Assuring Safe Autonomy

Proceedings of the 28th
Safety-Critical Systems Symposium
York, UK
11th-13th February 2020

SCSC

FOR EVERYONE WORKING IN SYSTEM SAFETY

Assuring Safe Autonomy

Related Titles

Making Systems Safer
Proceedings of the Eighteenth Safety-critical Systems Symposium, Bristol, UK, 2010
Dale and Anderson (Eds)
978-1-84996-085-4

Advances in Systems Safety
Proceedings of the Nineteenth Safety-critical Systems Symposium, Southampton, UK, 2011
Dale and Anderson (Eds)
978-0-85729-132-5

Achieving Systems Safety
Proceedings of the Twentieth Safety-critical Systems Symposium, Bristol, UK, 2012
Dale and Anderson (Eds)
978-1-4471-2493-1

Assuring the Safety of Systems
Proceedings of the Twenty-first Safety-critical Systems Symposium, Bristol, UK, 2013
Dale and Anderson (Eds)
978-1481018647

Addressing Systems Safety Challenges
Proceedings of the Twenty-second Safety-critical Systems Symposium, Brighton, UK, 2014
Dale and Anderson (Eds)
978-1491263648

Engineering Systems for Safety
Proceedings of the Twenty-third Safety-critical Systems Symposium, Bristol, UK, 2015
Parsons and Anderson (Eds)
978-1505689082

Developing Safe Systems
Proceedings of the Twenty-fourth Safety-critical Systems Symposium, Brighton, UK, 2016
Parsons and Anderson (Eds)
978-1519420077

Developments in System Safety Engineering
Proceedings of the Twenty-fifth Safety-critical Systems Symposium, Bristol, UK, 2017
Parsons and Kelly (Eds)
978-1540796288

Evolution of System Safety
Proceedings of the Twenty-sixth Safety-critical Systems Symposium, York, UK, 2018
Parsons and Kelly (Eds)
978-1979733618

Engineering Safe Autonomy
Proceedings of the twenty-seventh Safety-Critical Systems Symposium, Bristol, UK, 2019, SCSC-150
Parsons and Kelly (Eds)
978-1729361764

Mike Parsons • Mark Nicholson
Editors

Assuring Safe Autonomy

Proceedings of the 28th
Safety-Critical Systems Symposium
(SSS'20)
York, UK, 11th-13th February 2020

SCSC-154

The publication of these proceedings is sponsored by BAE Systems plc and Jaguar Land Rover

Editors

Mike Parsons
CGI UK
20 Fenchurch Street
London
EC3M 3BE
United Kingdom

Mark Nicholson
Department of Computer Science
University of York
Deramore Lane, York
YO10 5NG
United Kingdom

ISBN-9781713305668

Published by the Safety-Critical Systems Club 2020.

Individual chapters © as shown on respective first pages

Cover design by Alex King

Preface

This book contains the papers and posters presented at the twenty-eighth Safety-Critical Systems Symposium (SSS'20), held in York, UK. This year, like SSS'19, the symposium had a special focus on Autonomy and Machine Learning. The presenters produced important material covering topics relevant to safety-critical systems practitioners; we are very grateful to them for their contributions.

The first day themes were: Autonomy, AI and Machine Learning Part I and New Techniques. Dewi Daniels opened the symposium with a topical and highly relevant keynote covering the tragic air crashes in 2018 and 2019*: "The Boeing 737 MAX Accidents".* The second keynote of the day, looking at the evolution of satellite navigation was given by John Spriggs, a long-time supporter of the club, *"Satellite Navigation ~ Where Are We Going?"*. The third and final keynote of the day was given by Emma Taylor from SaRS entitled *"Safety in Space: A Changing Picture"*. The day finished with a "reverse panel" session led by Catherine Menon where the audience were asked to vote interactively on various topical issues related to system safety.

The two themes of the second day were Assurance and Security Informed Safety, including talks covering modular assurance cases and SACM. After the afternoon social activities in York there were drinks followed by a Poster Session. The day concluded with a fabulous symposium banquet with guest after-dinner speaker, Tim Kelly, a past Director of the club.

Data Safety, Human Factors and Autonomy, AI and Machine Learning Part II were the themes of the final day. The first two keynotes covered major infrastructure projects: Reuben McDonald gave the first talk of the day on the new high-speed rail link HS2. Alastair Crawford gave a fascinating talk about the construction and safety systems of the Hinkley C nuclear reactor. Jack Weast closed the symposium with a captivating talk related to his work on autonomous vehicles: *"An Open, Transparent, Industry-Driven Approach to AV Safety".*

We are grateful to our sponsors for their valuable support and to the exhibitors at the Symposium's tools and services fair for their participation. And we thank Alex King at York for the detailed event organisation.

Mike Parsons and Mark Nicholson

A message from the sponsors

BAE Systems and Jaguar Land Rover are pleased to support the publication of these proceedings. We recognise the benefit of the Safety-Critical Systems Club in promoting safety engineering and value the opportunities provided for continued professional development and the recognition and sharing of good practice. The safety of our employees, those using our products and the general public is critical to our business and is recognised as an important social responsibility.

The Safety-Critical Systems Club

organiser of the

Safety-Critical Systems Symposium

Safety-critical systems and the accidents that don't happen

When a plane crashes, it makes headlines. That thousands of flights each week do not crash is accepted as routine. Infusion pumps, railway signalling, security vetting systems, automatic braking functions, water treatment systems, radiation monitors and ambulance despatch systems are some of the critical systems in use, on which life and property depend. New autonomous systems, including road vehicles, maritime vessels and delivery drones will soon be with us; and they will be dynamically learning and adapting as they go. Safety and security are now intimately connected; it is a fact that you cannot be safe if you are not secure. Services are the way that many safety-critical systems are now delivered, meaning that people, processes, continuous change, service-level agreements and subcontractors must be included in the safety picture. Data, particularly training and testing data, is crucial in ensuring safety, especially in validating the new breed of autonomous vehicles arriving the next few years.

That safety-critical systems and services do work is because of the expertise and diligence of professional engineers, regulators, auditors and other practitioners. Their efforts prevent untold deaths and injuries every year. The Safety-Critical Systems Club (SCSC) has been actively engaged for over 27 years to help to ensure that this is the case, and to provide a "home" for safety professionals.

What is the Safety-Critical Systems Club?

The SCSC is the UK's professional network and community for sharing and developing knowledge about safety-critical systems and services. It brings together engineers and specialists working across a wide variety of industries, academics researching in the field, providers of the tools and services that help develop the systems, and the regulators who oversee safety. It provides, through working groups, publications, seminars, tutorials, a website, and at this annual Safety-Critical Systems Symposium, opportunities for them to network and learn from each other's experience in working hard at the accidents that don't happen. It focuses on current and emerging practices in safety engineering, software engineering, human factors, safety data and standards and guidance.

What does the SCSC do?

The SCSC maintains a website (scsc.uk), which includes a diary of events, working group areas and directories of tools and services. It publishes a regular newsletter, *Safety Systems*, three times a year. It organises seminars, workshops and training on general matters or specific subjects of current concern. Since 1993 it has organised the annual Safety-Critical Systems Symposium (SSS) where leaders in different aspects of safety from different industries, including consultants, regulators and academics, meet to exchange information and experience, with the content published in this proceedings volume. The SCSC supports industry working groups. Currently there are five active groups covering the areas of: Assurance Cases, Security Informed Safety, Data Safety, Autonomous Systems Safety and Service Assurance. These working groups provide a focus for discussions within industry and produce new guidance materials. Three new working groups are planned for 2020: (i) Safety Culture, (ii) Multi-core and Manycore and (iii) Systems of Systems (SoS). The SCSC carries out all these activities to support its mission:

> *... to raise awareness and facilitate technology transfer in the field of safety-critical systems ...*

Origins

The SCSC began its work in 1991, supported by the UK Department of Trade and Industry and the Engineering and Physical Sciences Research Council. The Club has been self-sufficient since 1994.

Membership

Membership may be either corporate or individual. Membership gives full web site access, the hardcopy newsletter, other mailings, and discounted entry to seminars, workshops and the annual Symposium. Membership is often paid by employers.

Corporate membership is for organisations that would like several employees to take advantage of the benefits of the SCSC. Different arrangements and packages are available. Contact alex.king@scsc.uk for more details.

More information on membership can be obtained at: http://www.scsc.uk/

Club Positions

The current and previous holders of the club positions are as follows:

Director

Mike Parsons	2019-
Tim Kelly	*2016-2019*
Tom Anderson	*1991-2016*

Newsletter Editor

Paul Hampton	2019-
Katrina Attwood	*2016-2019*
Felix Redmill	*1991-2016*

Website Editor

Brian Jepson	2004-

Steering Group Chair

Roger Rivett	2019-
Graham Jolliffe	*2014-2019*
Brian Jepson	*2007-2014*
Bob Malcolm	*1991-2007*

University of York Coordinator

Mark Nicholson	2019-

Coordinator/Events Coordinator/Programme Coordinator

Mike Parsons	2014-
Chris Dale	*2008-2014*
Felix Redmill	*1991-2008*

Manager

Alex King	2019-

Administrator

Alex King	2016-
Joan Atkinson	*1991-2016*

Working Group Leaders

Assurance Cases	Phil Williams
Security Informed Safety	Tom Turner
Data Safety, Service Assurance	Mike Parsons
Autonomous Systems	Rob Alexander

Contents

POSTER ABSTRACTS

The Boeing 737 MAX Accidents

Dewi Daniels

Software Safety Limited

Abstract *On 29 October 2018, a Boeing 737 MAX aircraft departed from Soekarno-Hatta International Airport, Jakarta, Indonesia. The aircraft was less than three months old. Twelve minutes later, it had crashed, killing all 189 passengers and crew on board. On 10 March 2019, another Boeing 737 MAX aircraft departed from Addis Ababa Bole International Airport, Ethiopia. This aircraft was less than four months old. Six minutes later, it too had crashed, killing all 157 passengers and crew on board. This paper presents an analysis as to why these two accidents happened.*

1 Introduction

In October 2018, a Lion Air Boeing 737 MAX aircraft crashed in Indonesia. Less than five months later, in March 2019, an Ethiopian Airlines Boeing 737 MAX crashed in Ethiopia. These two accidents killed 346 passengers and crew.

The two Boeing 737 MAX accidents came as a shock to the worldwide aircraft industry, which had hitherto enjoyed an outstanding safety record.

Modern airliners are exceptionally safe. Boeing's own Statistical Summary of Commercial Jet Airplane Accidents (Boeing 2017) reports that the 10-year average fatal accident rate for scheduled commercial passenger operations is 0.16 fatal accidents per million departures. Until these two accidents, the recent variants of the Boeing 737 had achieved an accident rate of 0.08 fatal accidents per million departures. A pilot or passenger could expect to experience 6.25 million flights on average before becoming involved in a fatal accident.

Everyone was shocked that a new aircraft design from a highly respected aircraft manufacturer could be involved in two fatal accidents so soon after entering service. The second accident resulted in the worldwide Boeing 737 MAX fleet being grounded. They remain grounded at the time of writing.

© Software Safety Limited 2019.
Published by the Safety-Critical Systems Club. All Rights Reserved

2 Abbreviations and Acronyms

Table 1. List of Abbreviations and Acronyms

AD	Airworthiness Directive
AF 447	Air France Flight 447
AMS	Amsterdam Airport Schipol
AND	Aircraft Nose Down
ANU	Aircraft Nose Up
AoA	Angle of Attack
ARR	Arrival
ATC	Air Traffic Control
ATSB	Australian Transportation Safety Bureau
BEA	Bureau d'Enquêtes et d'Analyses
CVR	Cockpit Voice Recorder
DFDR	Digital Flight Data Recorder
DOA	Design Organisation Approval
EASA	European Aviation Safety Agency
ETH302	Ethiopian Airlines Flight 302
FAA	Federal Aviation Administration
FL	Flight Level
ft	Feet
GPWS	Ground Proximity Warning System
HAAB	Addis Ababa Bole International Airport, Ethiopia
HKJK	Jomo Kenyatta International Airport, Nairobi, Kenya
IAS	Indicated Airspeed
KNKT	Komite Nasional Keselamatan Transportasi
kt	Knots
LNI610	Lion Air Flight 610
MCAS	Maneuvering Characteristics Augmentation System
MEL	Minimum Equipment List
MHz	Megahertz
NNC	Non-Normal Checklist
NTSB	National Transportation Safety Board
ODA	Organizational Design Approval
OMB	Operations Manual Bulletin
PAN PAN	The radiotelephony urgency signal
PF	Pilot Flying
PFD	Primary Flight Display
PIC	Pilot in Command
RA	Radar Altimeter
RAeS	Royal Aeronautical Society
RCAF	Royal Canadian Air Force
SIC	Second in Command
SOI	Stage of Involvement
TAB	Technical Advisory Board
TE	Terminal East
TK1951	Turkish Airlines Flight 1951
UTC	Coordinated Universal Time
V_{FC}/M_{FC}	Maximum speed for stability characteristics
V_{MO}	Maximum operating limit speed
VNAV	Vertical Navigation
WIII	Soekarno-Hatta International Airport, Jakarta
WIPK	Depati Amir Airport, Pangkal Pinang

3 Lion Air Flight 610

Fig. 1. Lion Air Boeing 737 MAX 8 PK-LQM. https://commons.wikimedia.org/wiki/File:Lion_Air_Boeing_737-MAX8;_@CGK_2018_(31333957778).jpg. This file is licensed under the Creative Commons Attribution-Share Alike 2.0 Generic license.

3.1 Accident Flight

Lion Air Flight 610 (LNI610) was a scheduled domestic flight from Soekarno-Hatta International Airport, Jakarta (WIII) to Depati Amir Airport, Pangkal Pinang (WIPK). The accident flight on 29 October 2018 was operated by a Boeing 737 MAX aircraft registered PL-LQP. This was a new aircraft whose Certificate of Airworthiness had been issued on 15 August 2018.

The aircraft departed at 06:20 local time, which is 23:20 Coordinated Universal Time (UTC). The left control column stick shaker activated during rotation and remained activated for most of the flight. The following table shows the timeline of the flight.

Table 2. Lion Air Flight 610 Timeline

Time (UTC)	Event
23:21:22	The Second in Command (SIC) made initial contact with the Terminal East (TE) controller. The TE controller instructed LNI610 to climb to 27,000 ft.
23:21:28	The SIC asked the TE controller to confirm the altitude of the aircraft as shown on his radar display. The TE controller responded that the altitude was 900 ft., which was acknowledged by the SIC.
23:21:53	The SIC requested approval "to some holding point". The TE controller asked what the problem was. The pilot responded, "flight control problem".
23:22:05	The Digital Flight Data Recorder (DFDR) recorded automatic Aircraft Nose Down (AND) trim active for 10 seconds followed by flight crew commanded Aircraft Nose Up (ANU) trim.
23:22:31	The TE controller instructed LNI610 to climb and maintain 5,000 ft and to turn left heading 050°. This was acknowledged by the SIC.
23:22:48	The flaps extended to 5 and the automatic AND trim stopped.
23:22:56	The SIC asked the TE controller the speed as indicated on the radar display. The controller responded that the ground speed was 322 kt.
23:24:51	The TE controller annotated "FLIGHT CONTROL TROB" on his display.
23:25:05	The TE controller instructed LNI610 to turn left heading 350° and to maintain 5,000 ft. This was acknowledged by the SIC.
23:25:18	The flaps retracted to 0.
23:25:27	The automatic AND trim and flight crew commanded ANU trim began again and continued for the remainder of the flight.
23:26:32	The TE controller instructed LNI610 to turn right heading 050° and to maintain 5,000 ft. This was acknowledged by the SIC.
23:26:59	The TE controller instructed LNI610 to turn right heading 070° to avoid traffic. LNI610 did not respond. The controller called LNI610 twice.
23:27:13	LNI610 acknowledged the instruction to turn right heading 070°.
23:27:15	The TE controller instructed LNI610 to turn right heading 090°, which was acknowledged by the SIC. A few seconds later, the TE controller revised the instruction to stop the turn and fly heading 070°, which was acknowledged by the SIC.
23:28:15	The TE controller provided traffic information to LNI610, which responded "Zero". About 14 seconds later, the controller instructed LNI610 to turn left heading 050° and maintain 5,000 feet. This was acknowledged by the SIC.

Time (UTC)	Event
23:29:37	The TE controller questioned LNI610 whether the aircraft was descending. The SIC advised the controller that they had a flight control problem and were flying the aircraft manually.
23:29:45	The TE controller instructed LNI610 to maintain heading 050° and contact the Arrival (ARR) controller. The instruction was acknowledged by the SIC.
23:30:03	LNI610 contacted the ARR controller and advised that they were experiencing a flight control problem. The ARR controller advised LNI610 to prepare for landing on runway 25L and instructed them to fly heading 070°. The instruction was read back by the SIC.
23:30:58	The SIC requested, "LNI650 due to weather request proceed to ESALA", which was approved by the ARR controller.
23:31:09	The Pilot in Command (PIC) advised the ARR controller that the altitude of the aircraft could not be determined due to all aircraft instruments indicating different altitudes. The pilot used the call sign LNI650 during the exchange. The ARR controller acknowledged then stated "LNI610 no restriction".
23:31:23	The PIC requested the ARR controller to block altitude 3,000 feet above and below for traffic avoidance. The ARR controller asked what altitude the pilot wanted.
23:31:35	The PIC responded, "five thou". The ARR controller approved the pilot request.
23:31:54	The DFDR stopped recording.

3.2 Previous Flight

The previous flight on 28 October 2018 had also experienced a problem with the Angle of Attack (AoA) sensor, which had been replaced immediately before the flight. The stick shaker activated during the rotation and remained active throughout the flight. The IAS DISAGREE warning appeared on the Primary Flight Displays (PFDs). This warning indicates that the airspeed has differed by more than 5 kt between the left and right PFDs for more than 5 seconds. The PIC handed over control to the SIC, cross-checked the PFDs with the standby instrument and determined that the left PFD was in error. The PIC noticed the aircraft was automatically trimming AND. The PIC declared PAN PAN. This is the radiotelephony urgency signal, which indicates a condition concerning the safety of an aircraft or other vehicle, or of some person on board or within sight, but which does not require immediate assistance. He moved the STAB TRIM CUTOUT switches to CUTOUT, which resolved the problem.

The PIC followed three Non-Normal Checklists (NNCs), none of which contained an instruction to land at the nearest suitable airport. The PIC therefore elected to continue to his destination. This decision seems surprising given that the stick shaker presumably remained active throughout the 1.5-hour flight. The aircraft landed at Jakarta at 15:56 UTC.

3.3 Response to the Accident

On 6 November 2018, Boeing issued Operations Manual Bulletin (OMB) TBC-19 to emphasize the existing procedures provided in the runaway stabilizer NNC. This OMB stated:

> In the event an uncommanded nose down stabilizer trim is experienced on the 737-8 /-9, in conjunction with one or more of the above indications or effects, do the Runaway Stabilizer NNC ensuring that the STAB TRIM CUTOUT switches are set to CUTOUT and stay in the CUTOUT position for the remainder of the flight.
> **Note:** Initially, higher control forces may be needed to overcome any stabilizer nose down trim already applied. Electric stabilizer trim can be used to neutralize control column pitch forces before moving the STAB TRIM CUTOUT switches to CUTOUT. Manual stabilizer trim can be used after the STAB TRIM CUTOUT switches are moved to CUTOUT.

On 7 November 2018, the Federal Aviation Administration (FAA) issued Emergency Airworthiness Directive (AD) 2018-23-51. This Emergency AD stated:

> **Runaway Stabilizer**
> Disengage autopilot and control airplane pitch attitude with control column and main electric trim as required. If relaxing the column causes the trim to move, set stabilizer trim switches to CUTOUT. If runway continues, hold the stabilizer trim wheel against rotation and trim the airplane manually.
>
> Note: The 737-8/-9 uses a Flight Control Computer command of pitch trim to improve longitudinal handling characteristics. In the event of erroneous Angle of Attack (AoA) input, the pitch trim system can trim the stabilizer nose down in increments lasting up to 10 seconds.
> ...
> Initially, higher control forces may be needed to overcome any stabilizer nose down trim already applied. Electric stabilizer trim can be used to neutralize control column pitch forces before moving the STAB TRIM CUTOUT switches to CUTOUT. Manual stabilizer trim can be used before and after the STAB TRIM CUTOUT switches are moved to CUTOUT.

The preliminary accident report (KNKT 2018), the OMB and the AD did not mention the Maneuvering Characteristics Augmentation System (MCAS), which was a new system fitted to the Boeing 737 MAX, but not to earlier variants of the Boeing 737. The AD alludes to MCAS when it states, “Note: The 737-8/-9 uses

a Flight Control Computer command of pitch trim to improve longitudinal handling characteristics".

4 Ethiopian Airlines Flight 302

Fig. 2. Ethiopian Airlines Boeing 737 MAX 8 ET-AVJ. https://commons.wikimedia.org/wiki/File:Ethiopian_Airlines_ET-AVJ_takeoff_from_TLV_(46461974574).jpg. This file is licensed under the Creative Commons Attribution-Share Alike 2.0 Generic license.

4.1 Accident Flight

Ethiopian Airlines Flight 302 (ETH302) was a scheduled international flight from Addis Ababa Bole International Airport, Ethiopia (HAAB) to Jomo Kenyatta International Airport, Nairobi, Kenya (HKJK). The accident flight on 10 March 2019 was operated by a Boeing 737 MAX aircraft registered ET-AVJ. This was another new aircraft, which had been delivered to Ethiopian Airlines on 15 November 2018.

The aircraft departed at 08:38 local time, which was 05:38 UTC. The following table shows the timeline of this flight.

Table 3. Ethiopian Airlines Flight 302 Timeline

Time (UTC)	Event
05:37:34	Air Traffic Control (ATC) issued take off clearance to ETH302 and to contact radar on 119.7 MHz. The takeoff roll began at approximately 05:38. The takeoff roll appeared normal.
05:38:44	Shortly after liftoff, the left stick shaker activated and remained active until near the end of the recording. Also, the airspeed, altitude and flight director pitch bar values from the left side deviated from the corresponding right side values.
05:38:46	At about 200 ft radio altitude, the Master Caution parameter changed state. The First Officer called out Master Caution Anti-Ice on the Cockpit Voice Recorder (CVR).
05:38:58	At about 400 ft radio altitude, the flight director pitch mode changed to VNAV SPEED, the Captain called out "Command" (the standard call out for autopilot engagement) and an autopilot warning is recorded.
05:39:00	The Captain called out "Command".
05:39:01	At about 630 ft radio altitude, a second autopilot warning is recorded.
05:39:06	The Captain advised the First Officer to contact radar and the First Officer reported SHALA 2A departure crossing 8400 ft and climbing Flight Level (FL) 320.
05:39:22	At about 1,000 feet the left autopilot was engaged and the flaps were retracted.
05:39:29	The radar controller identified ETH302 and instructed to climb FL 340 and when able right turn direct to waypoint RUDOL and the First Officer acknowledged.
05:39:50	The Captain asked the First Officer to request to maintain runway heading
05:39:55	Autopilot disengaged
05:39:57	The Captain advised again the First Officer to request to maintain runway heading and that they are having flight control problems.
05:40:00	The DFDR recorded an automatic AND trim activated for 9.0 seconds. The climb was arrested, and the aircraft descended slightly.
05:40:03	Ground Proximity Warning System (GPWS) "DON'T SINK" alerts occurred.
05:40:05	The First Officer reported to ATC that they were unable to maintain SHALA 1A and requested runway heading which was approved by ATC.
05:40:06	The column moved aft, and a positive climb was re-established during the automatic AND motion.

Time (UTC)	Event
05:40:12	Approximately three seconds after AND stabilizer motion ends, electric trim (from pilot activated switches on the yoke) in the ANU direction is recorded on the DFDR and the stabilizer moved in the ANU direction to 2.4 units. The Aircraft pitch attitude remained about the same as the back pressure on the column increased.
05:40:20	Approximately five seconds after the end of the ANU stabilizer motion, a second instance of automatic AND stabilizer trim occurred, and the stabilizer moved down and reached 0.4 units
05:40:23	From 05:40:23 to 05:40:31, three GPWS “DON’T SINK” alerts occurred.
05:40:27	The Captain advised the First Officer to trim up with him.
05:40:28	Manual electric trim in the ANU direction was recorded and the stabilizer reversed moving in the ANU direction.
05:40:35	The First Officer called out “stab trim cut-out” two times. Captain agreed and First Officer confirmed stab trim cut-out.
05:40:41	Approximately five seconds after the end of the ANU stabilizer motion, a third instance of AND automatic trim command occurred without any corresponding motion of the stabilizer, which is consistent with the stabilizer trim cutout switches were in the ‘’cutout’’ position.
05:40:44	The Captain called out three times “Pull-up” and the First Officer acknowledged.
05:40:50	The Captain instructed the First Officer to advise ATC that they would like to maintain 14,000 ft and they have flight control problem.
05:40:56	The First Officer requested ATC to maintain 14,000 ft and reported that they are having flight control problem. ATC approved.
05:40:42	From 05:40:42 to 05:43:11 (about two and a half minutes), the stabilizer position gradually moved in the AND direction. During this time, aft force was applied to the control columns which remained aft of neutral position. The left indicated airspeed increased from approximately 305 kt to approximately 340 kt (V_{MO}). The right indicated airspeed was approximately 20-25 kt higher than the left. The data indicates that aft force was applied to both columns simultaneously several times throughout the remainder of the recording.
05:41:20	The right overspeed clacker was recorded on CVR. It remained active until the end of the recording.
05:41:30	The Captain requested the First Officer to pitch up with him and the First Officer acknowledged.
05:41:32	The Captain asked the First Officer if the trim is functional. The First Officer replied that the trim was not working and asked if he could try it manually. The Captain told him to try.

Time (UTC)	Event
05:41:54	The First Officer replied that manual trim is not working.
05:43:04	The Captain asked the First Officer to pitch up together and said that pitch is not enough.
05:43:11	Two momentary manual electric trim inputs are recorded in the ANU direction. The stabilizer moved in the ANU direction.
05:43:20	Approximately five seconds after the last manual electric trim input, an AND automatic trim command occurred, and the stabilizer moved in the AND direction. The aircraft began pitching nose down. Additional simultaneous aft column force was applied, but the nose down pitch continues, eventually reaching 40° nose down.

4.2 Response to the Accident

The day of the accident, 10 March 2019, Ethiopian Airlines decided to suspend operation of the Boeing 737 MAX. The following day, 11 March 2019, China, Indonesia and Mongolia grounded all Boeing 737 MAX aircraft in their countries. On 12 March 2019, Singapore, India, Turkey, South Korea, Europe, Australia and Malaysia also grounded the Boeing 737 MAX. On 13 March 2019, Canada, the United States of America, Hong Kong, Panama, Vietnam, New Zealand, Mexico, Brazil, Colombia, Chile and Trinidad and Tobago all grounded the Boeing 737 MAX. The Boeing 737 MAX remains grounded at the time of writing.

The preliminary report was issued on 4 April 2019 (Ethiopian Civil Aviation Authority 2019). Again, the preliminary report does not mention MCAS.

The accidents have been widely reported and discussed in the general and specialist press. For example, the Seattle Times has published several newspaper articles on the two accidents (e.g. Seattle Times 2019a and Seattle Times 2019b).

5 The Role of MCAS

The prototype Boeing 737-100 made its first flight in April 1967. The Boeing 737 has undergone continuous evolution over the past 50 years. The Boeing 737 MAX is the latest variant, superseding the Boeing 737-700, -800 and-900.

The Boeing 737 MAX uses bigger engines, which are mounted further forwards than on previous variants of the Boeing 737. During wind tunnel testing, Boeing found that the Boeing 737 MAX tended to pitch up during certain flight

conditions. This violated Federal Aviation Regulation (FAR) 25.143(g), which states:

> When maneuvering at a constant airspeed or Mach number (up to V_{FC}/M_{FC}[1]), the stick forces and the gradient of the stick force versus maneuvering load factor must lie within satisfactory limits. The stick forces must not be so great as to make excessive demands on the pilot's strength when maneuvering the airplane, and must not be so low that the airplane can easily be overstressed inadvertently. Changes of gradient that occur with changes of load factor must not cause undue difficulty in maintaining control of the airplane, and local gradients must not be so low as to result in a danger of overcontrolling.

Boeing tried to fix the problem by changing the physical design of the aircraft, but no effective aerodynamic solution was found. Boeing decided to introduce a new system called MCAS. MCAS would push the aircraft's nose down at high AoA to compensate for the pitch up, resulting in a more linear control response. MCAS was developed by Rockwell Collins, now Collins Aerospace (Washington Post 2019).

MCAS is activated by a single AoA sensor. On the accident flights, failure of this AoA sensor resulted in a high AoA being reported to MCAS, which led to MCAS pushing the nose down to try and prevent a stall. The pilots applied electric trim to counter the effect of MCAS, but as soon as they stopped applying trim, MCAS reactivated and applied more nose down trim. This eventually resulted in MCAS applying so much nose down trim that both aircraft ended up diving into the ground despite the pilot pulling back on the yoke. This ratcheting effect does not appear to have been anticipated by the developers of MCAS nor was failure of the AoA sensor considered during simulator testing or flight test.

6 Fallacies About the Accidents

Several fallacies about the accidents have been widely reported in the news and on internet forums.

6.1 Fallacy #1: The Accidents Were the Result of Reduced FAA Oversight

Several commentators blamed the FAA for the Boeing 737 MAX accidents because they were alleged to have delegated oversight to Boeing (e.g. Politico

[1] V_{FC}/M_{FC} means the maximum speed for stability characteristics.

2019). In fact, the current FAA system of Organizational Design Approval (ODA) is very similar to the European Aviation Safety Agency (EASA)'s Design Organisation Approval (DOA), which has worked very successfully since EASA was created in 2002.

Furthermore, MCAS was determined to be DO-178C Level C. DO-178C defines 6 software levels, from Level A for software that can cause or contribute to a Catastrophic failure condition to Level E for software that has no safety effect. Level C software is software that can only cause or contribute to a Major failure condition.

FAA Order 8110.49A (FAA 2018) specifies how to determine the level of certification authority involvement in a software project. Even under the previous regulatory regime before Boeing was given ODA, the FAA would not have participated directly in Stage of Involvement (SOI) audits for a Level C software project by an established supplier such as Rockwell Collins. In this instance, Boeing ODA would have performed the independent oversight.

6.2 Fallacy #2: Pilots Trained in the United States Would Have Successfully Handled the Situation

This claim was made by Congressman Sam Graves in a hearing of the Subcommittee on Aviation on 15 May 2019 (United States House of Representatives 2019).

FAA Emergency AD 2018-23-51 reads, "Disengage autopilot and control airplane pitch attitude with control column and main electric trim as required. If relaxing the column causes the trim to move, set stabilizer trim switches to CUTOUT. If runway continues, hold the stabilizer trim wheel against rotation and trim the airplane manually".

The crew of ETH302 attempted to carry out all the actions required by the Emergency AD. The autopilot was disengaged at 05:39:55 UTC. The column was moved aft at 05:40:06. The main electric trim was used at 05:40:12. The stabilizer trim switches were set to CUTOUT at 05:40:35. The First Officer attempted to trim the aircraft manually at 05:41:32.

It appears that the main electric trim was not used for long enough and it is not clear how hard the First Officer tried to trim the aircraft manually. The difficulty experienced in turning the trim wheel and holding back the column suggests that the aircraft was considerably out of trim when the stabilizer trim switches were set to CUTOUT.

On 10 May 2019, Aviation Week had reported that a US-based Boeing 737 MAX crew tried to replicate the ETH302 accident in a simulator (Aviation Week 2019). The crew found that keeping the aircraft level required significant aft-column pressure by the captain, while aerodynamic forces prevented the first officer from moving the trim wheel a full turn. The crew were eventually able to recover the aircraft by using a technique known as the roller coaster procedure, which is not described in the Boeing 737 MAX flight manual. The pilot said he had only learned of the roller coaster procedure from excerpts of a Boeing 737-200 manual posted in an online pilot forum following the two Boeing 737 MAX accidents. Aviation Week concluded that "the Ethiopian crew faced a near-impossible task of getting their 737 MAX 8 back under control".

On 17 May 2019, the New York Times reported that "Boeing recently discovered that the simulators could not accurately replicate the difficult conditions created by a malfunctioning anti-stall system, which played a role in both disasters. The simulators did not reflect the immense force that it would take for pilots to regain control of the aircraft once the system activated on a plane traveling at a high speed" (New York Times 2019). This suggests that the task faced by the Ethiopian crew was even more difficult than had been suggested by the Aviation Week article.

A puzzling aspect is that the DFDR shows that at 05:43:11, about 32 seconds before the end of the recording, two momentary manual electric trim inputs were recorded in the ANU direction. This follows the Captain asking the First Officer to pitch up together at 05:43:04. The most likely explanation is that the crew re-engaged the stabilizer trim in an unsuccessful attempt to regain control of the aircraft.

Congressman Graves criticized the pilots for reactivating the automated system. He said, "No operating procedures that I know of direct a pilot to reactivate a faulty system". I think this criticism is unfair. The crew had already followed the procedure described in Emergency AD 2018-23-51. They were still struggling to control the aircraft, even with both pilots pulling back on the yoke.

Congressman Graves also criticized the pilots because they "never pulled back the throttles after setting them at full thrust for takeoff". This is a valid criticism, although the Emergency AD makes no mention of the throttles. However, the nose down attitude would have caused the aircraft to accelerate regardless of the throttle setting.

6.3 Fallacy #3: The Accidents Were Due to Outsourcing

On 28 June 2019, Bloomberg published a news article claiming that much of the work on Boeing 737 Max had been outsourced to workers making as little as $9 an hour (Bloomberg 2019).

There is no evidence that the Boeing 737 MCAS software development was outsourced or that the system safety engineering and system engineering failings that led to the accidents were caused by low-paid, outsourced workers.

7 Reasons for the Accidents

7.1 Reason #1: Failure of System Safety Engineering

The system safety review presented to the FAA assumed that MCAS was limited to commanding 0.6 degrees nose down from the trimmed position. The Boeing submission to the FAA included a safety analysis of the effect of possible MCAS failures. Boeing analyzed what would happen if, in normal flight, MCAS triggered inadvertently up to its maximum authority and moved the horizontal stabiliser the maximum of 0.6 degrees. Boeing also analysed what would happen if MCAS kept running for three seconds at its standard rate of 0.27 degrees per second, producing 0.81 degrees of movement, thus exceeding its nominal maximum authority. Three seconds was selected because this is the time that FAA guidance says it should take a pilot to recognise what's happening and begin to counter it. Boeing assessed both failure conditions as Major.

Boeing did not consider the possibility that MCAS could trigger repeatedly, therefore far exceeding its nominal maximum authority. This is what happened on the accident flights. Such a failure condition would have been considered Catastrophic.

Initially, it was thought that MCAS was only necessary at high AoA at high airspeeds. This original version of MCAS required two conditions – high AoA and high G-force – to activate.

During flight test, it was discovered that MCAS would also be required during certain low-speed flight conditions. Because at low speed, the stabilizer needs to be deflected more to achieve the same effect, the maximum authority of MCAS was increased from 0.6 degrees to 2.5 degrees each time it was activated.

Because there would be little G-force present during these low speed maneuvers, high G-force was removed as a condition for MCAS to trigger, meaning that MCAS would now be activated by a single AoA sensor.

The safety analysis was revisited after this change. However, Boeing considered that the effect of MCAS failure would be less at low speed than at high speed, so they assessed that the worst-case failure condition remained unchanged at Major.

7.2 Reason #2: Failure of Requirements Engineering

It is apparent from both preliminary accident reports that MCAS trimmed the aircraft nose down. On both accident flights, the Pilot Flying (PF) tried to counter MCAS using the electric trim. MCAS repeatedly reactivated, applying more nose down trim as soon as the PF stopped applying electric trim.

There is no evidence that the MCAS software failed to satisfy its requirements. It appears that the MCAS software behaved correctly according to its requirements, but that those requirements specified unsafe behaviour.

It seems that the requirements author only considered a single activation of MCAS. Like the system safety engineers, he or she does not appear to have considered the possibility that MCAS could activate repeatedly, eventually driving the stabilizer to a fully nose down position.

It is remarkable that the requirements were not validated in the simulator or during flight test. There is no evidence that Boeing ever flight-tested the accident scenario where a faulty AoA sensor caused an unintended activation of MCAS.

8 Further Thoughts about the Accidents

8.1 Thought #1: Procedural Mitigation Was Ineffective

Following the Lion Air accident, the FAA issued Emergency AD 2018-23-51. The Emergency AD revised the Operating Procedures Chapter of the airplane flight manual (AFM) to include:

> Disengage autopilot and control airplane pitch attitude with control column and main electric trim as required. If relaxing the column causes the trim to move, set stabilizer trim switches to CUTOUT. If runway continues, hold the stabilizer trim wheel against rotation and trim the airplane manually.

These instructions look reasonable but failed to anticipate the high control forces required in the situation in which the Ethiopian Airlines crew found themselves. The crew of ETH302 tried to follow the steps described in the Emergency

AD. The high forces required to move the stabilizer trim wheel led the First Officer to conclude, incorrectly, that manual trim was not working. The force required to pull back the yoke was so high that the aircraft pitched nose down 40° despite both pilots pulling back on the yoke.

A YouTube video (Mentour 2019) shows the control forces required to counter nose down trim at high airspeed.

The loss of ETH302 suggests that procedural mitigation cannot be relied upon to compensate for inadequate design integrity in a complex, software-intensive system.

8.2 Thought #2: Failure to Learn from an Earlier Accident

Sidney Dekker gave a keynote speech at the Safety Critical Systems Symposium 2019 on the earlier loss of a Boeing 737-800 (Dekker 2019). Turkish Airlines Flight 1951 (TK1951) crashed at Amsterdam Airport Schiphol (AMS) on 25 February 2009. Failure of the left Radar Altimeter (RA) resulted in the auto throttle believing that the aircraft had landed and entering retard flare mode. The aircraft crashed short of the runway, resulting in the death of nine passengers and crew, including all three pilots. Sydney Dekker identified two factors in the loss of TK1951 that were also present in the two Boeing 737 MAX accidents:

1. Dependence on a single sensor. The Boeing 737-800 auto throttle uses a single RA even though two RAs are fitted to the aircraft. The Boeing 737 MAX MCAS uses a single AoA sensor even though two AoA sensors are fitted to the aircraft.
2. Relevant technical detail withheld from the flight crews. Sydney Dekker explained that the Boeing documentation did not make it clear that the auto throttle always uses a single RA and implied that it could use either RA, just like the autopilot. The pilots were aware that the left RA had failed but believed that the auto throttle was using the right RA. Likewise, the Boeing 737 MAX documentation did not mention MCAS. The pilots were not even aware of the existence of the system.

It seems that Boeing repeated two mistakes on the Boeing 737 MAX that could have been avoided by learning from the loss of TK1951 in 2009.

8.3 Thought #3: Erosion of Airmanship Skills

8.3.1 Incidents Involving Exceptional Airmanship Skills

There have been several incidents where the flight crew demonstrated exceptional airmanship skills to save the lives of their passengers. Such incidents include, but are not limited to:

1. Air Canada Flight 143, also known as the "Gimli Glider" (Canadian Government 1985). On 23 July 1993, an Air Canada Boeing 767 ran out of fuel over a remote part of Canada. The aircraft's fuel gauges were inoperative due to an electronic fault, but the Minimum Equipment List (MEL) allowed the aircraft to take off provided enough fuel was loaded to reach the destination. However, due to a mix-up between pounds and kilograms by the ground crew, only half the amount of fuel required had been loaded. When the engines stopped, the crew decided to land at a former Royal Canadian Air Force (RCAF) base at Gimli, which had been converted into a racetrack. Ironically, the pilot, Captain Robert (Bob) Pearson was an experienced glider pilot, while the co-pilot, First Officer Maurice Quintal, had been stationed at Gimli while serving with the RCAF. Captain Pearson side-slipped the Boeing 767 on final approach and landed successfully at Gimli.
2. United Airlines Flight 232 (NTSB 1990). On 19 July 1989, a DC-10 flying from Denver, Colorado to Chicago, Illinois suffered a catastrophic failure of its tail-mounted engine. This resulted in total loss of the hydraulic flight controls. The flight crew, commanded by Captain Alfred Clair Haynes, discovered they could regain limited control of the aircraft by using only the throttles for the two remaining engines. The flight crew decided to attempt a landing at Sioux City, Iowa. They were unable to land the aircraft normally. The right wing touched the runway first, spilling fuel, which ignited. Sadly, 111 people died in the impact and subsequent fire. Remarkably, 185 people survived, including the four pilots, which was a testament to their remarkable determination and flying skills.
3. Qantas Flight 32 (ATSB 2013). On 4 November 2010, an Airbus A380 flying from London to Sydney via Singapore suffered an uncontained failure of one of its four engines. The aircraft was badly damaged by flying debris. In fact, Airbus had never flight-tested an A380 with so many systems inoperable because they believed it to be impossible that the aircraft would continue to fly after sustaining so much damage. The

flight crew, commanded by Captain Richard Champion de Crespigny, landed successfully at Singapore. The firefighters were reluctant to come near the aircraft because of the risk of fire! The Australian flight crew 'encouraged' the firefighters to come closer, which they did (RAeS 2010).

8.3.2 Accidents Involving Poor Airmanship Skills

However, in recent years, there have been a worrying number of avoidable accidents where the flight crew showed poor airmanship skills. Accidents where the flight crew's poor airmanship skills contributed to, rather than prevented, the accident include:

1. Colgan Air Flight 3407 (NTSB 2010). On 12 February 2009, a Bombardier Dash-8 Q400 was being flown from Newark, New Jersey to Buffalo, New York. While flying on autopilot, the aircraft had encountered icing conditions, which had caused the autopilot to push the nose down to compensate. When the aircraft was slowed down for landing, the autopilot disconnected, and the stick shaker activated. The aircraft pitched up and stalled. The pilot failed to push the yoke forward sufficiently to recover from the stall. He tried to stop the aircraft from rolling by using the aileron, which caused a wing drop that resulted in a steep descent. The aircraft crashing into a house, killing all 49 passengers and crew on board and one person on the ground.
2. Air France Flight 447 (BEA 2012). On 1 June 2009, an Airbus A330 enroute from Rio de Janeiro to Paris suffered icing of the pitot tube, which is used to measure air speed. The pilot should have been able to fly the aircraft using attitude, but instead pulled the stick back and caused the aircraft to stall. Rather than pushing the stick forward to recover from the stall, the pilot pulled back on the stick in a vain attempt to regain altitude. The pilot held the stick fully back nearly all the way to the ground. The aircraft crashed into the Atlantic Ocean, killing all 228 passengers and crew on board.

8.3.3 Airmanship Skills in the Boeing 737 MAX Accidents

The crew of ETH302 did follow the checklist but failed to save the aircraft and their passengers. However, they could have saved the aircraft by setting the STAB TRIM CUTOUT switches to CUTOUT before the aircraft had got too far out of trim or the air speed had increased to the point where they could no longer apply manual trim, or by using the electric stabilizer trim to neutralize the control column pitch forces before they moved the STAB TRIM CUTOUT switches to CUTOUT. They could also have retarded the throttles to reduce the rate of acceleration.

9 Conclusion

Airworthiness regulations have often been criticised for being too onerous, placing an undue burden on airframe manufacturers and equipment suppliers and resulting in unnecessary expense.

Critics point out there are over 26,000 certified jet airplanes in service worldwide, that scheduled commercial passenger operations enjoy an excellent accident rate of 0.16 fatal accidents per million departures and that not a single fatal accident in passenger service has been ascribed to software failure.

The Boeing 737 MAX was designed by a well-respected manufacturer (Boeing) and certified by a well-respected certification authority (the FAA). Nevertheless, this new design resulted in two fatal accidents resulting in the deaths of 346 people within two years of entering service.

This experience suggests that the airworthiness regulations are not too stringent after all.

Disclaimers This paper is based on the information publicly available at the time of writing. It is expected that more information will become available when the FAA Technical Advisory Board (TAB) report is published. Fresh information could affect the conclusions presented in this paper.

References

ATSB (2013) In-flight uncontained engine failure, Airbus A380-842, VH-OOA overhead Batam Island, Indonesia | 4 November 2010. Final – 27 June 2013. http://www.atsb.gov.au/media/4173625/ao-2010-089_final.pdf. Accessed 20 September 2019.

Aviation Week (2019) Ethiopian MAX crash simulator scenario stuns pilots. https://aviationweek.com/commercial-aviation/ethiopian-max-crash-simulator-scenario-stuns-pilots. Accessed 9 August 2019.

BEA (2012) Final Report on the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France, flight AF 447 Rio de Janeiro – Paris, published July 2012.

https://www.bea.aero/docspa/2009/f-cp090601.en/pdf/f-cp090601.en.pdf. Accessed 20 September 2019.
Bloomberg (2019) Boeing's 737 Max Software Outsource to $9-an-Hour Engineers, 28 June 2019. https://www.bloomberg.com/news/articles/2019-06-28/boeing-s-737-max-software-outsourced-to-9-an-hour-engineers. Accessed 9 August 2019.
Boeing (2017) Statistical summary of commercial jet airplane accidents, worldwide operations 1959-2017.https://www.boeing.com/resources/boeingdotcom/company/about_bca/pdf/statsum.pdf. Accessed 5 August 2019.
Canadian Government (1985) Final Report of the Board of Inquiry into Air Canada Boeing 767 C-GAUN Accident – Gimli, Manitoba, July 23, 1983. Published April 1985. http://data2.collectionscanada.gc.ca/e/e444/e011083519.pdf. Accessed 20 September 2019.
Dekker (2019) Automation Surprise in the 21st Century: Culture, Collaborative Cognition, Complexity and Legacy Systems. Proceedings of the Twenty-seventh Safety-Critical Systems Symposium, Bristol, UK. https://scsc.uk/rp150.1:1. Accessed 12 August 2019.
Ethiopian Civil Aviation Authority (2019) Aircraft accident investigation preliminary report, Ethiopian Airlines Group, B737-8 (MAX) registered ET-AVJ. Published March 2019. http://www.ecaa.gov.et/Home/wp-content/uploads/2019/07/Preliminary-Report-B737-800MAX-ET-AVJ.pdf. Accessed 28 October 2019.
FAA (2018) Order 9110.49A Software Approval Guidelines. https://www.faa.gov/documentLibrary/media/Order/FAA_Order_8110.49A.pdf. Accessed 9 August 2019.
KNKT (2018) Aircraft accident investigation report, PT. Lion Mentari Airlines, Boeing 737-8 (MAX), PK-LQP. Published November 2018. http://knkt.dephub.go.id/knkt/ntsc_aviation/baru/pre/2018/2018%20-%20035%20-%20PK-LQP%20Preliminary%20Report.pdf. Accessed 5 August 2019
Mentour Pilot (2019) Boeing 737 Unable to Trim!! Cockpit video (Full flight sim). https://www.youtube.com/watch?v=aoNOVlxJmow from 10:06 to 13:47. Accessed 12 August 2019.
New York Times (2019) Boeing 737 Max Simulators Are in High Demand. They Are Flawed. https://www.nytimes.com/2019/05/17/business/boeing-737-max-simulators.html. Accessed 9 August 2019.
NTSB (1990). Aircraft Accident Report. United Airlines Flight 232. McDonnell Douglas DC-10-10. Sioux Gateway Airport, Sioux City, Iowa. July 19, 1989. Published November 1, 1990. https://www.ntsb.gov/investigations/AccidentReports/Reports/AAR-90-06.pdf. Accessed 20 September 2019.
NTSB (2010). Loss of Control on Approach, Colgan Air, Inc., Operating as Continental Connection Flight 3407, Bombardier DHC-8-400, N200WQ, Clarence Center, New York, February 12, 2009. Adopted February 2, 2010. https://www.ntsb.gov/investigations/AccidentReports/Reports/AAR1001.pdf. Accessed 20 September 2019.
Politico (2019) How the FAA delegated oversight to Boeing. https://www.politico.com/story/2019/03/21/congress-faa-boeing-oversight-1287902. Accessed 9 August 2019.
RAeS (2010) Exclusive – Qantas QF32 flight from the cockpit. https://www.aerosociety.com/news/exclusive-qantas-qf32-flight-from-the-cockpit/. Accessed 20 September 2019.
Seattle Times (2019a) Flawed analysis, failed oversight: How Boeing, FAA certified the suspect 737 MAX flight control system, 17 March 2019. https://www.seattletimes.com/business/boeing-aerospace/failed-certification-faa-missed-safety-issues-in-the-737-max-system-implicated-in-the-lion-air-crash/. Accessed 9 August 2019.
Seattle Times (2019b) The inside story of MCAS: How Boeing's 737 MAX system gained power and lost safeguards, 22 June 2019. https://www.seattletimes.com/seattle-news/times-

watchdog/the-inside-story-of-mcas-how-boeings-737-max-system-gained-power-and-lost-safeguards/. Accessed 9 August 2019.

United States House of Representatives (2019), Ranking Members Sam Graves & Garret Graves Statements from Hearing on Status of the Boeing 737 MAX, 15 May 2019. https://republicans-transportation.house.gov/news/documentsingle.aspx?DocumentID=404188. Accessed 9 August 2019.

Washington Post (2019), Boeing's 737 Max design contains fingerprints of hundreds of suppliers, 5 April 2019. https://www.washingtonpost.com/business/economy/boeings-737-max-design-contains-fingerprints-of-hundreds-of-suppliers/2019/04/05/44f22024-57ab-11e9-8ef3-fbd41a2ce4d5_story.html?noredirect=on. Accessed 19 August 2019.

Challenges in Education and (Human) Training for Systems with AI

Nikita Johnson and Mark Nicholson

University of York

Abstract *The idea that systems are safe because humans can adapt their behaviour is a key tenet of Safety II (proposed by Erik Hollnagel). But what happens when humans in a system are largely replaced with AI components? This adaptability for safety must come from the system, and it requires engineers to encode people's ability to succeed under great uncertainty and complexity. This requirement drives a fundamental change in the competencies an engineer must possess. Similarly, there are competence profile changes for other stakeholders such as regulators, safety and security practitioners, and system operators. Using support from the literature and experience on the Assuring Autonomy International Programme (AAIP), this paper aims to enumerate some differences in competence, training and education for assuring system with AI components, discuss the difficulties of doing this and propose a way forward. In this way we can start to build a picture of what good practice is. It is argued that creating practical theories and training tools for AI systems in a safety-critical context is not just an exercise in intellectualism, but an integral part of the safety of future systems. A systematic approach for identifying competencies and creating training to match those competence requirements is proposed.*

© Nikita Johnson and Mark Nicholson, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

The Emergence of Accidental Autonomy

Alastair Faulkner

Abbeymeade Limited

Mark Nicholson

University of York

Abstract *The Boeing 737 MAX - Manoeuvring Characteristics Augmentation System (MCAS) accidents have demonstrated how cumulative factors may lead to accidental autonomy. Accidental autonomy emerges when differences in models compete over resources and control. In the operational domain, one manifestation is failure at the human-machine interface. Subtle, incremental changes in technology allied with downward economic pressures encourage reuse to create the system safety property of 'additionality'. Cumulative incremental changes occur that when taken together, are safety significant. Reuse of process, product or both gives rise to inappropriate design trade-offs. Assumptions about the completeness of process, design, implementation or context may lead, in extreme circumstances, to the creation of accidental autonomy - systems without human oversight that implement safety-related functionality or services. Oversight, assessment and approval of systems dependent on reuse are reliant on the familiarity of the assessor with the reused elements within their operational and use context. Incomplete, inadequate understanding and failures of comprehension, along with the allure of fast software development, create the potential for accidental autonomy*

1 Introduction

Systems engineering is subject to several capability and economic pressures. This has driven Systems Engineers to create systems from generic components. To become generic components are designed to depend on data to be configured, characterised and parameterised to the required behaviour. This data dependency is also evident in the broader system, its subsystems, interfaces and shared overlapping datasets.

Our collective understanding of autonomy [1] is bound to context [2] and era. It is context that gives legitimacy to a person or systems actions, the facts or pro-

© Alastair Faulkner and Mark Nicholson 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

cesses of doing something, typically to achieve an aim. [3] Automation often first appears as an aid to support existing practice. These aids are developed and evolve to support and reinforce changes – driving uniformity and consistency, but not necessarily improving reliability, performance or resilience. [4] The overall impact on the working environment is a dramatic increase in the volume of digital data and flow of that digital data around system elements and a concomitant decrease in understanding of the context and limitations of this autonomy. [5]

The potential for accidental autonomy arises when changes are implemented as 'islands' of functionality to support identified activities. We use the word accidental [28] as happening by chance, unintentionally, or unexpectedly. Existing system interfaces characterise the boundaries of these new functionalities. Operators become reliant on the autonomous actions of systems (and its assumed functional model). Even if they can intervene, they often do not, as they tend to trust the technology.

At the same time, a reliance on automated decision-making increases. At the lowest level of autonomy, [6] computers offer no assistance; they facilitate information acquisition. Later, they offer a set of decision alternatives, facilitating analysis. They provide increasing amounts of support for the decision-making process itself. The operator has a restricted time to overrule the autonomous decision before an action. Finally, the highest levels of autonomy provide implementation with no capability for the human to overrule and little, if any, information provided on what actions the autonomy has undertaken.

Safety risk arises where the operators understanding of the system and the role of autonomy is incomplete. A classic human-machine interaction failure may result. Clean design and efficiency are frequently used to justify autonomy. Autonomy may correct an underlying instability in the system model or design. It becomes accidental autonomy when its actions arise from incomplete design, implementation or its use is outside an acceptable design envelope with respect to safety. Further, it may induce additional human failures due to mismatches between the human mental model, the goals of autonomy, and what is detected, by both parties, of the real-world context.

Systems Engineering has progressed to the point where machines have the capability to undertake high-level decisions and enact consequent actions without recourse to direct human input. As a result, we should not assume the presence of a user; instead, we employ the term actor as 'an individual, entity, or combination of product (including autonomy), people, and process'. This raises the question of supervision [11], and to what extent humans remain involved in operational decisions.

2 System as the Fundamental Concept

A system is a (purposeful [7]) set of things working together as part of a mechanism or an interconnecting network; a complex whole. [8] Systems operate in increasingly open environments. These environments and the data exchanged within them are homogeneous or heterogeneous or a mixture of both. The nature of the environment influences the formation of the system boundary (porous or secure (as defined by an appropriate security model)). For modern complex systems, it is common to present several views of the system either across many sheets (possibly in a hierarchy) or to separate physical realisations from abstract (logical) models. Therefore, reviewing any classification system requires an appreciation of the context, including the role of the actors within each viewpoint.

Systems of Systems (SoS) bring together a set of systems for a task that none of the systems can accomplish on its own. Each constituent system keeps its management, goals, and resources while coordinating within the SoS and adapting to meet SoS goals. [9] Interconnected SoS give rise to accidental tasks associated with the mapping and representation of abstract entities and mapping of those entities onto the constraints of the solutions. [10] One implementation uses Internet of Things (IoT) devices as generalised platforms, enabling them to be adapted and configured to a range of solution areas. Often these include a highly capable Operating System (OS) that can provide a full range of communication and computational services. They are configured (and characterised) to a particular task (or range of tasks) through data. An additional dimension to SoS would be to add (joiners) or remove (leavers) to the system. A joiner introduces additional capabilities, capacities and tasks. A leaver removes them. The system is modified (adapted) to reflect these changes in the service catalogues.

There are several interrelated models at play here for a complex system.

- Security – one starting point requires all system entities to be assigned an identity so that an actors access privileges can be assigned (and revoked). This leads to consideration of the relative authorities between the system and the actors that use and are served by the system.
- Maintenance – it is common to use Permit to Work (tickets) as part of a formal maintenance procedure to isolate physical plant and equipment. How should autonomy and their respective services be controlled while maintenance is being enacted? How will the system be reconfigured to manage reduced capability whilst one or more systems (and their associated (and (mutually) dependent services)) are maintained?
- Operational – how big, complex or complicated should a system be before the risks associated with its failure demand the creation of an operational model? It is common to consider operational modes during system safety management activities. An operational strategy should be used to direct and

inform the creation and maintenance of the operational model. Issues associated with size and complexity require that a decision model is constructed and maintained over the operational model.

- Safety - In an ideal world, control and protection function differentiation would be applied universally to function, flow and service. The desirable safety features of hardware and software are well established. It is not clear that such requirements are applied to services. Issues associated with size and complexities require that a services model is constructed and maintained.
- Risk - in systems which are dependent on autonomy, the form and nature of risks are multi-dimensional, crossing many discipline and system boundaries. Many of these boundaries are indistinct. The risks associated with reliance on autonomy require the reappraisal of existing risk models.
- Supervision – the action of supervising someone or something. [11] A supervisory model is a scheme for specifying and enforcing supervisory policies.

The use of SoS and IoT technologies means that an integrated risk model across these models is required to address residual and unsecured functionality.

3 Autonomy

Autonomy [1] is not new. It is used to describe human political activity, for a region 'having its own laws'. In the modern sense, autonomy is readily adapted to address systems capable of operating without direct human control [12] but varying degrees of human supervision or oversight. At one extreme automaton [13] is confined to actions described by a predetermined set of coded instructions. At the other are learning systems that adapt their behaviour in response to changes in the operating environment – its context. [2]

It is context – 'the circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood' [2], that provide the basis of this paper. Context is not limited to the operational environment. For the engineered system, it reflects the designer's expectation of the operational environment. This position is further complicated by what learning systems 'understand' as the basis for the formulation of its response.

Consider an automatic [14] washing machine; it works by itself with little or no direct interaction. The use or introduction of multiple automatic devices into an existing context creates automation. [15] These definitions contain an implicit expectation of a static context. That, the context is known, and if it changes at all, it changes slowly under controlled conditions. What if, a new generation of automatic devices, providing a form-fit-function [16] replacement, are introduced based on IoT. Suppose the system has unused capability and capacity. An

unsecured 'discovery' function recognises other IoTs and connects to this residual and unsecured functionality. Taking a SoS perspective, this represents one or more emergent properties [17], possibly with unintended consequences.

Autonomy becomes multi-dimensional under Industry 4.0: [18, 19]

- The vertical integration of flexible and reconfigurable systems within businesses;
- The horizontal integration of inter-company value chains and networks;
- The product life-cycle integration of digital end-to-end engineering activities across the entire value chain of both the product and the associated systems.

3.1 Accidental Autonomy

In a connected system, automation creates a dramatic increase in the volume of digital data and flow of that digital data around system elements and a concomitant decrease in understanding of the context and limitations of that data. Change is a crucial factor in creating the potential for accidental autonomy. At one extreme are revisions of an existing product or model, as with an aircraft. At the other are a series of incremental changes in the pre-existing operational context. Existing boundaries may not constrain these new functionalities. An actor may become reliant on data produced by another actor (passed across one or more boundaries), with little ability to influence the data stream they have become reliant on. At the same time, reliance on automated decision-making increases.

Accidental autonomy results when differences between models of use, and context of use, are not sufficiently well understood in all operational modes. This includes misuse. The true nature of its inclusion in the system is omitted, or downplayed, in the safety ensurance and assurance process. As a result, insufficient safety mitigations are provided, and poor human-machine interactions may occur. We can conceive of accidental autonomy arising between two or more elements of a system, perhaps as a complete system, or SoS. Within any given context, these elements, systems or SoS have different responsibilities, of which some will be safety-related. The accidental autonomous system could co-exist with people-centered activities reliant on a predefined set of processes. A fundamental assumption is that the people within the system are trained, competent and experienced enough to deliver the required operation (including its safety management); this implies a maturity of definition and application. Therefore, the provision of the product or process is dependent on context.

Extending the concept of emergent properties [17] gives rise to the concept of emergent autonomy. Emergent autonomy is a consequence of the interactions

and relationships between system elements rather than the behaviour of individual elements. [7] The nature of emergent autonomy is linked to robustness and resilience and is a critical contribution to safety. Accidental autonomy is a subset of emergent autonomy and may be a result of incomplete development activities. Consider an existing design. This design has been in production and operation for many years undergoing successive revisions. Each revision 'refreshes' the technology, typically the control systems. The effects of seemingly minor changes become cumulative, giving rise to the safety property of 'additionality'. Budgetary, time and project management constraints limit the safety analysis to a subset of changes. Given these conditions, it is easy to see how the effects of automation and increasing levels of autonomy are overlooked. For example, an aircraft design will be revised over many generations of the airframe. It is not unusual for the aircraft model to evolve over 40 years as with Nimrod (as an extensive modification of the de Havilland Comet). We await, with interest, the two accident reports for the Boeing 737 MAX.

For visualisation, we reuse elements of [4]. Fig. 1 illustrates the footprint of accidental autonomy.

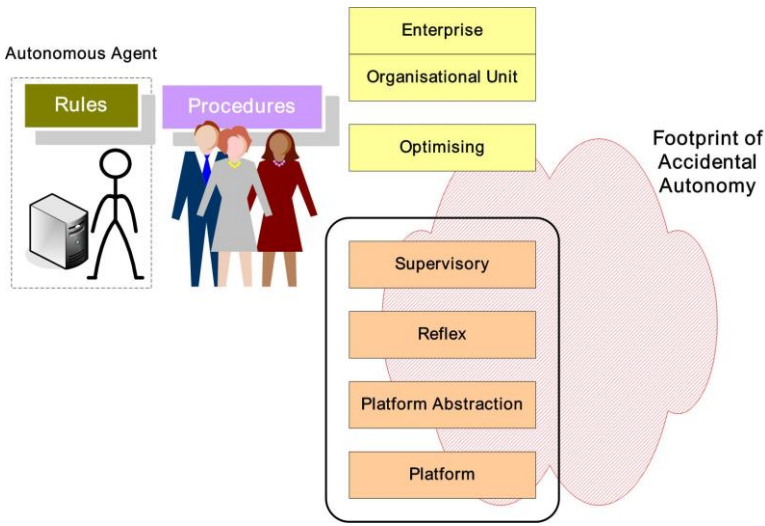


Fig. 1. A layered model for a hierarchy of systems and the footprint of accidental autonomy

How far up the hierarchy could accidental autonomy reach? Potentially, all the way to the top. Decision support systems are ever more reliant on data analytics, data science, data engineering and autonomy. Fig. 1 also illustrates the supervision within the hierarchy.

As autonomy moves through the hierarchy of abstraction, its responsibilities and authorities change. Similarly, the ability of humans to provide Safety-I [20] mitigation procedures needs to be addressed. Improved diagnostics, monitoring

and higher-level response, are required. This challenges the ability to design efficient procedures. Furthermore, it impacts on the ability of humans to execute Safety-II [20] dynamic mitigations, as their mental model of the system and how they interact with it is flawed.

The implementation strategy must include a means to impose a boundary to the propagation of the actions of the system and the impact of failures on the availability and safety of the system. Fig. 2 illustrates how Interface Agreements (IA) [5] provides that functionality for new and legacy systems.

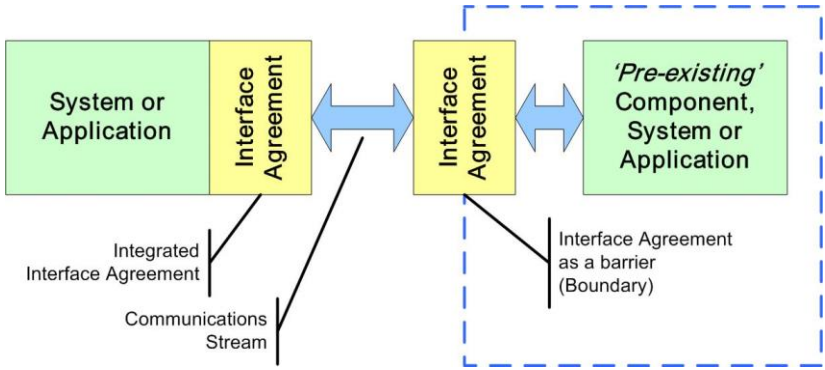


Fig. 2. Implementation Model for Interface Agreements

A series of interfacing elements can be envisaged. Transformations occur as the sequence is progressed. One or more actors supervise this series of transformations. These chains, and associated transformations can occur at multiple levels of abstraction. As a result, a number of different aspects to Autonomy can be identified. Any one of these organisational aspects may lead to developers creating accidental autonomy.

3.2 Vertical Autonomy

Vertical expansion is used to describe an organisation that grows through the acquisition of companies that produce the intermediate goods needed by the business. Economic pressures create management and organisational structures that become more rigid and inflexible. These companies become monolithic, often creating closed implementations in silos. Over time this inflexibility makes it difficult for a company to respond to changes in the marketplace. Implementations based on SoS and IoT offer an opportunity to break free from vertical silos.

This requires vertical integration of flexible and reconfigurable systems within businesses.

These structures also apply to systems. Consider a basic control system that consists of input-controller-output. In past implementations, the input would be wholly dependent on the physical properties of the sensor. For example, a bi-metallic strip is used to implement the function of a thermostat where specific temperature causes the differential expansion to deflect enough to close (or open) the contacts. These devices, once physically co-located, are now implemented using IoT on remote networks. They may even be replaced by more generic devices that sense a number of properties. The required data is then mined from the output of these sensors.

Here we use the following classifications of vertical autonomy:

- backward (upstream)
- forward (downstream)
- balanced (both upstream and downstream)

Fig. 3 illustrates vertical integration; supplier, manufacturer and distributor:

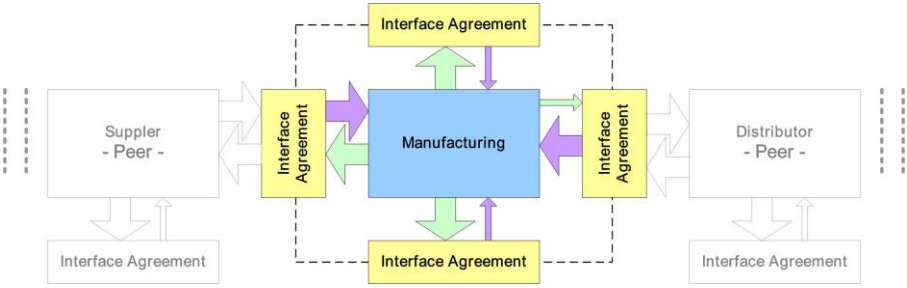


Fig. 3. Supplier, Manufacturer and Distributor

3.2.1 Backward (Upstream)

A manufacturer implements upstream expansion by purchasing a parts supplier. Upstream autonomy provides more data, command and control of the upstream systems. For example, accidental upstream autonomy contains an operational model, that when a candidate production schedule is interpreted, causes it to re-order stock for a future production run without seeking authority (or confirmation) that the production run would take place. Therefore, accidental upstream autonomy is consumption-led and may lag demand. In our control system example, upstream autonomy is where the controller reduces the frequency of updates from the sensor becoming less responsive.

3.2.2 Forward (Downstream)

A manufacturer implements downstream expansion by purchasing the distribution and sales network. Downstream autonomy allows the manufacturer to only produce what can be sold. Accidental downstream autonomy might misinterpret demand to produce too much or too little. Therefore, accidental downstream autonomy is demand-led. Sudden (step) changes in demand may create instability and lead to over and under production. In our control system example, downstream autonomy is where the controller checks the actuator (output), continually monitoring the energy required to assert the output has occurred. The demand is the energy (effort) required to assert the output.

3.2.3 Balanced (both Upstream and Downstream)

A manufacturer implements balanced expansion by purchasing a supplier's, distribution and sales network. Balanced autonomy contains a balance of consumption, demand and 'damping' (stabilising) elements. Accidental balanced autonomy may implement complex functions analogous to the Proportional-Integral-Derivative (PID) pattern in control theory. [21] Other patterns also apply. In our control system example, balanced autonomy is where the controller adapts to the operational requirement adjusting the input sensor rate to the best fit to the PID set point and deviations from it. At the same time, the controller calculates a predictive output anticipating the effort required to apply the output.

3.3 Horizontal Autonomy

A manufacturer implements horizontal integration through changes in capacity and capability. For example, the introduction of new technology replaces fixed function machines (manufacturing cells) using reconfigurable workstations clustered in super-cells into a production line. This creates higher capacity and requires integration of the value chains and networks. In our control system example, horizontal autonomy combines an array of identical *balanced autonomy controllers*. Horizontal autonomy uses sensor fusion over the input devices and load-balancing across the outputs. This implies the use of supervision [11] over the array of controllers.

The above characterisation implies that horizontal data transformation and vertical abstraction transformations (along the lines of fig. 1) need to be considered when identifying Critical Control Points (CCP) to ensure that development processes do not introduce accidental autonomy. [22]

Paths (physical product or data) will incorporate CCP and may involve multiple sources and multiple sinks. These issues are compounded when these paths are dynamic. This dynamism is not limited to changing numbers of sources or sinks or processes but also changes in demand (capacity) and availability. Paths may be transient synthesised on demand for single-use and then discarded. Incident investigation is eased where these CCPs provide controls and logged data. Therefore, CCPs provide evidence about the actions of the system, including accidental autonomy.

To implement dynamic paths, one analogy would be to use a standardised library of elements across a node and link network. This provides one means of implementing redundancy where nodes are unavailable. In network theory, links can be assigned a weight; path management is used to identify a route with the least weight. Accidental autonomy need not be persistent; it may arise from transient elements of the dynamic formation of the system. Therefore, implementation requires the definition of an Identity Model and Security Model.

3.4 Product-line Autonomy

A product line can be defined as a set of systems sharing a common, managed set of features that satisfy specific needs and are developed from a common set of core assets in a prescribed way. [23, 24] A product-line [25] development employs a life-cycle model and the process it contains to develop the system definition, into the system. The defined set of features can be reused within defined fit-form-function [16] contexts.

Accidental introduction of Autonomy via product lines presents two potential threats; that the fit-form-function replacement introduces residual and unsecured functionality, and secondly that the system has grown organically and cannot support digital end-to-end engineering activities with a reasonable certainty of outcome. This may lead to the accidental autonomy being presented with a range, sequence and timings in an ‘unfamiliar’ environment that it cannot manage safely. The autonomy cannot rely on the human to retrieve the situation.

4 Cyber Physical System Threats

When deciding what steps to take to prevent and respond to threats, we might immediately focus on the threat of hacking. However, the range of threats systems face is much broader than this, encompassing anything that can adversely affect their operation, including theft, destruction, disclosure, modification or unauthorized access.

We modify the definition of threat [26] to be ‘an actor likely to cause one or more hazards.’ This definition includes autonomy within the actor. Changes in system context require resilience and robustness from the autonomy to withstand threats arising from identity, security and sneak attributes.

4.1 Identity

‘Identity’ should be a unique labelling of attributes of the object (system resource) being accessed and of the actor requesting access in a given context. Threats arise from identity error, duplicate and missing identities. A malicious, deliberate identity-based (spoofing) act could be used as a means to gain control of the system. One means to counter identity-based threats is the use of a formalised and managed ‘identity model’. An identity model is a scheme for specifying and enforcing identity policies. An identity model is a key aspect of the security model.

Both emergent and accidental autonomy are sensitive to error, omission or duplicate identities, especially in dynamically reconfigurable systems as changes in system behaviour presents significant system safety management challenges. These issues are compounded where a system uses joiners and leavers as one means to satisfy operational demand, including capability and capacity. It cannot be assumed that identity theft applies only to users as it applies equally to actors, systems, assets, data and data paths.

4.2 Security

A security model is a scheme for specifying and enforcing security policies. A security model uses a formal model of access rights. Authorisation is implemented using identity and enforced through authentication. Potentially, failures of cyber-security provide the intruder with unbridled access to a system. Identity and its management is a critical feature of both Data Safety and information security.

Should each instance of autonomy be required to be assigned its own unique identity? Low-level systems often do not implement a security or identity model. Autonomy in such systems can act without authorisation, often acting with ‘super-user’ rights. More extensive systems required security models, and therefore, each actor or autonomy requires one or more identities.

4.3 Sneak Attributes

Systems may contain sneak (or hidden) attributes that may cause unwanted action or inhibit desired functions [27]. Sneak attributes arise where the physical realisation contains many more characteristics than the logical representation. Examination of simple network switches reveals capabilities to separate network traffic using configuration data. Errors in the configuration may permit 'mixed network' traffic, compromising the intended separation, and creating additional paths between entities.

These may include:

- Sneak paths: unintended paths within a system and its external interfaces.
- Sneak timing: unexpected interruption or enabling of a [function or service] due to timing problems which may cause or prevent the activation or inhibition of a function [or service] at an unexpected time.
- Sneak indications: undesired activation or deactivation of a [status] indication which may cause an ambiguous or false display of system operating conditions.
- Sneak [identity]: incorrect or ambiguous identity of a [function or service] which may cause actor error through inappropriate control activation.

As complex networks of autonomous actors embedded within systems emerge the ability to create accidental autonomy via a sneak, increases.

5 Discussion

There are many examples of automatic systems, from the washing machine to automobile automatic transmissions. The degree of possible automation increases by using SoS and IoT technologies. Economic pressures to increase efficiencies, such as fuel economy, drive change to the foundations of existing designs and the organisations that develop and operate them. The increased reliability, availability, capability and real-time response of control systems allow the designer to explore inherently unstable designs. These unstable designs offer potential operational efficiencies. This involves a design change from stable towards the edge of instability, where additional means are required to stay within the stability envelope. One means to achieve this is autonomy. Accidental autonomy contributes to the emergent property of incremental additionality. It may be unintentional, or unexpected, but it does not happen by chance. These emergent properties and autonomies are systematic. Their behaviours are repeatable and may be inclusions (impurities) from an incomplete development.

This paper has outlined how autonomy is multi-dimensional. It follows that accidental autonomy is also multi-dimensional. The system safety question, 'how could this possibly go wrong?' is more relevant than ever. Early indicators of the Boeing 737 MAX accidents show how organisational structures can interact with economic and engineering factors to create the potential for accidental autonomy. They illustrate how changes to a pre-existing model create a change in context where the users (pilots) are unaware of the underlying nature of change. One approach to addressing accidental autonomy is to increase the ability of pilots to address the unexpected. This will require them to become experts in diagnosing and addressing such issues. This higher competency is required to detect, diagnose and formulate a course of action during the operational event. This assumes that the actions of the user can result in a positive outcome. The more dimensions autonomy occupies, the more extensive, the more difficult - real-time - diagnosis becomes. We can no longer rely on the steady-state being a safe condition. Economics and human factors knowledge imply that this is not a credible approach. As a result, organisational and technical means must be found to identify and address potential accidental autonomy issues.

For all forms of autonomy, the permutations of threats, failures and latent hazards may be extensive but are foreseeable. Accidental Autonomy may result in unintended consequences. Merton [29] asserts that these are outcomes that are not the ones foreseen and intended by a purposeful action. The operational domain includes maintenance. What provision should autonomy make to include the statutory requirements for 'Permit to Work' (PtW) and its required 'Safe System of Work' (SSoW) based on one or more 'Safe Method of Work' (SMoW)?

Its hidden nature characterises accidental autonomy. Its use to manage the properties of an underlying design without adequate annunciation and user involvement contributes to the confusion of an ongoing incident. Its actions cannot be assumed to be benign or malevolent; they will be incomplete, in accidental autonomies pursuit of ill-defined, unknown goals.

6 Conclusion

No single approach resolves the difficulties associated with the essence [9] of engineered and accidental autonomy - those parts concerned with the fashioning of abstract conceptual structures of high complexity. Greater scale, scope and complexity give rise to an urgency to create strategies to manage the large-scale application of techniques and measures. In part, this urgency arises from the reliance placed on these systems and safety risks associated with their failure. Many systems are reliant on this connectivity and provide substantially reduced functionality when the interconnectivity fails. In contrast, previous generations

of system implementations operated as islands, separated and protected from external influences – and in that sense, self-reliant.

The first step in addressing accidental autonomy is recognition of its potential scale, scope and complexity. It will introduce new failure mechanisms due to differences in its required and the actual context. It is a multi-dimensional problem occupying vertical, horizontal and product-line axes. Its management will require many interrelated approaches and their associated techniques and measures. Its independence also provides new forms of latent failures. For example, two or more learning autonomous elements may adapt in different ways to changes in their operational environment. The new behaviours may introduce conflict and cause instability over many learning cycles. There is no guarantee that these differences will resolve into a stable state. Such variations will only be manifest in an incident.

References

1. Autonomy – "freedom from external control or influence; independence" www.lexico.com/en/definition/autonomy (visited 10 October 2019)
2. Context "The circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood." www.lexico.com/en/definition/context (visited 10 October 2019)
3. Action - "fact or process of doing something, typically to achieve an aim", www.lexico.com/en/definition/action (visited 10 October 2019)
4. Alastair Faulkner and Mark Nicholson, "An Assessment Framework for Data-Centric Systems", Proceedings of the Twenty-Second Safety-Critical Systems Symposium, Brighton, UK. Edited by Chris Dale and Tom Anderson. ISBN 978-1491263648. Safety Critical Systems Club, 2014
5. Alastair Faulkner and Mark Nicholson, "Data-Centric Safety: Challenges, Approaches, and Incident Investigation", Elsevier, to be published March 2020, ISBN: 978-0-12-820790-1
6. R. Parasuraman, T. B. Sheridan, and C. D. Wickens. A Model for Types and Levels of Human Interaction with Automation. IEEE Transactions on Systems, Man, Cybernetics—Part A, Vol. 30, No. 3, 2000
7. P. Checkland. Systems Thinking, Systems Practice. 1981 John Wiley & Sons
8. System –"A set of things working together as parts of a mechanism or an interconnecting network; a complex whole." www.lexico.com/en/definition/system (visited 10 October 2019)
9. BKCASE Governance and Editorial Board. Guide to the Systems Engineering Body of Knowledge (SEBoK). 2017
10. Frederick P. Brooks. No Silver Bullet — Essence and Accident in Software Engineering. Proceedings of the IFIP Tenth World Computing Conference, 1986, pages 1069–1076

11. Supervision - “action of supervising someone or something” www.lexico.com/en/definition/supervision (visited 10 October 2019)
12. Autonomous - “denoting or performed by a device capable of operating without direct human control.” www.lexico.com/en/definition/autonomous (visited 10 October 2019)
13. Automaton – “machine which performs a range of functions according to a predetermined set of coded instructions” www.lexico.com/en/definition/automaton (visited 10 October 2019)
14. Automatic – “(of a device or process) working by itself with little or no direct human control.” www.lexico.com/en/definition/automatic (visited 10 October 2019)
15. Automation – “use or introduction of automatic equipment in a manufacturing or other process or facility.” www.lexico.com/en/definition/automation (visited 10 October 2019)
16. Morris, R. The fundamentals of product design. AVA Publishing. 2009
17. SEBoK (Guide to the Systems Engineering Body of Knowledge): Emergence, www.sebokwiki.org/wiki/Emergence (visited 10 October 2019)
18. David Leal-Ayala, Jennifer Castañeda-Navarrete and Carlos López-Gómez, OK Computer? -The safety and security dimensions of Industry 4.0, Global Manufacturing and Industrialisation Summit (GMIS) and Lloyd’s Register Foundation (LRF). 2019
19. Mario Hermann, Tobias Pentek and Boris Otto, “Design Principles for Industrie 4.0 Scenarios” in 49th Hawaii International Conference on System Sciences (HICSS), pages 3928-3937, 2016
20. Erik Hollnagel. Safety-I and Safety-II. ISBN 978-1472423085. Routledge, 2014
21. Araki, M. "PID Control", Control Systems, Robotics and Automation, Vol II, 2011
22. Mark E. J. Newman. Networks: an Introduction. Oxford, 2010
23. Paul Clements and Linda Northrop. Software Product Lines: Practices and Patterns. Addison-Wesley Professional, 2001
24. Linda Northrop and Paul Clements. A Framework for Software Product Line Practice. Software Engineering Institute, 2012
25. André de Oliveira et al. Supporting the Automated Generation of Modular Product Line Safety Cases. Volume Theory and Engineering of Complex Systems and Dependability. ISBN 978-3-319-19215-4. Springer International Publishing, 2015, pages 319–330
26. Threat - “A person or thing likely to cause damage or danger.” www.lexico.com/en/definition/threat (visited 10 October 2019)
27. MIL-STD-785B Reliability Program for Systems and Equipment Development and Production. US Department of Defence, 1980.
28. Accidental – “happening by chance, unintentionally, or unexpectedly” www.lexico.com/en/definition/accidental (visited 10 October 2019)
29. Robert K. Merton "The Unanticipated Consequences of Purposive Social Action" American Sociological Review. 1 (6): 895. 1936

Quantifying Dataset Properties for Systematic Artificial Neural Network Classifier Verification

Darryl Hond, Alex White, Hamid Asgari

Thales Research, Technology & Innovation[1]

Abstract *Autonomous systems make use of a suite of algorithms for understanding the environment in which they are deployed. These algorithms typically solve one or more classic problems, such as classification, prediction and detection. This is a key step in making independent decisions in order to accomplish a set of objectives. Artificial neural networks (ANNs) are one such class of algorithms, which have shown great promise in view of their apparent ability to learn the complicated patterns underlying high-dimensional data. The decision boundary approximated by such networks is highly non-linear and difficult to interpret, which is particularly problematic in cases where these decisions can compromise the safety of either the system itself, or people. Furthermore, the choice of data used to prepare and test the network can have a dramatic impact on performance (e.g. misclassification) and consequently safety. In this paper, we introduce a novel measure for quantifying the difference between the datasets used for training ANN-based object classification algorithms, and the test datasets used for verifying and evaluating classifier performance. This measure allows performance metrics to be placed into context by characterizing the test datasets employed for evaluation. A system requirement could specify the permitted form of the functional relationship between ANN classifier performance and the dissimilarity between training and test datasets. The novel measure is empirically assessed using publicly available datasets*

[1] Thales UK, 350 Longwater Avenue, Green Park, Reading, Berkshire RG2 6GF, UK

© Darryl Hond, Alexander White, Hamid Asgari, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

1 Introduction

Verification and validation (V&V) are vital parts of the development of any engineering system. These processes are well-established in more mature sectors of engineering such as aerospace and automotive. However, they are not as well developed in areas such as autonomy and machine learning (ML), and the broader field of artificial intelligence (AI). Since ML technologies are being more widely adopted, questions will be asked as to whether they will behave in an expected manner, and whether any people they might interact with during their operation will be safe.

The focus of this paper will be the verification of artificial neural network (ANN) systems, which lie within the field of ML. More specifically, ANN classifiers will be considered. ANN classifiers have played a key role in the progress made by commercial ML systems in recent years, and so their verification is vital, especially for safety-critical systems. An ANN classifier is verified by acquiring evidence that it operates as expected.

Systems are verified with respect to the specified requirements. The obvious requirement for a classifier is: a given level of classification performance; the requirement can be verified by dynamic testing. However, this unelaborated requirement does not refer to the test dataset. An unspecified test dataset could be interpreted as being any arbitrary input set, in which case it might be inappropriate for the system incorporating the classifier. An additional requirement needs to be specified: the properties of the test dataset used to evaluate the classification performance. The test dataset might be characterised, for example, in terms of its relation to the dataset used to train the classifier, or in terms of its noise content, or in terms of the intrinsic separability of its component classes. System requirements addressing discriminative capability could then state the permitted form of a function mapping test dataset properties to classifier performance. If these requirements are specified and verified, we can have a degree of confidence that the classifier will perform at a certain level in an operational mode when applied to input instances of a certain type. Different classifier use cases might require different requirements to be drafted in terms of the stated test dataset properties.

This paper introduces a measure. This is a measure of the dissimilarity between a training dataset and a test dataset, and is formulated in terms of the separation of points in a representational space. This dissimilarity will henceforth be termed ‘dataset dissimilarity’. It can be used to quantify the properties of real-world classification datasets. Classifier performance for a particular test dataset might itself be measured in terms of say, accuracy. If so, classifier accuracy can then be given as a function of our dataset dissimilarity measure: each test dataset is assigned a dataset dissimilarity value, and this is mapped to an accuracy value. This in turn allows requirements to be formulated in terms of the necessary relationship between performance and the test dataset dissimilarity measure. If this

requirement is verified, evidence has been gathered that the classifier will perform at a certain level, in terms of classification accuracy for example, when processing test datasets which return particular values for the dataset dissimilarity measure.

The remainder of this paper is structured as follows. We briefly introduce some important verification and validation concepts derived from software engineering in section 2, and then review the relevant literature in section 3. We then discuss our contribution to the field in section 4, before presenting and analysing initial experimental work in section 5. Section 6 examines how the approach described could be extended. In the final section, we draw conclusions.

2 Key V&V Concepts Relevant to Machine Learning Algorithms

Verification methods can be defined to be: 'methods by which confidence can be gained in the correctness of a system with respect to its specification' (Hond, et al. 2018). These methods can be divided into formal verification and dynamic testing. Formal verification attempts to prove at least some degree of correctness of the system model with respect to requirements; dynamic testing employs test instances to gain confidence in the correctness of the actual, trained ML system.

ML training and test dataset instances correspond to points in an input space, the space in which all possible inputs can be represented, or, after feature extraction, as points in some feature space (as explained further in section 4). Sometimes it is assumed that the test data is distributed, in input or feature space, in the same way as the training data (Chung et al 2019). Often, this assumption will be incorrect. The form of these distributions and the identification of outliers relative to a distribution are significant for the evaluation or verification of ML systems.

In traditional software engineering, corner cases are a key aspect of verifying the correctness of a program's behaviour. A corner case is a state in which several factors reach the edge of their operating or behavioural range (each being an edge case) simultaneously (e.g. when several program inputs achieve their maximum or minimum values) (ChicoState, 2016). Such states might only occur rarely, and might also be difficult to generate or simulate. These situations are important, because in some sense they represent a high-stress scenario for the designed software: if the behaviour observed under such a set of inputs is correct or within expectations, then the designer derives confidence - at least for the range and number of corner cases tested - that the program they have written is robust.

The idea of a corner case naturally transfers to ML input data. Here, corner cases can be considered to be outliers with respect to training data. If each factor influencing or operating within a system corresponds to an axis, and there are a

large number of axes, then co-occurring factor values would correspond to points in a high-dimensional space, and could be modelled by some probability density function. A point in this space corresponding to some corner case combination of factor values, would tend to return very low values for such a probability density function relative to the values returned for more likely combinations. Such a point would therefore be an outlier, and outliers tend to pose a significant challenge for a ML algorithm. As ML algorithms enter deployment, we need to be able to state and verify the expected, or mandated, prediction or classification performance for corner cases.

3 Related Work

In (Asgari, et al. 2019), a review of selected aspects of RAS (Robotics and Autonomous Systems) from different sectors is covered. This review includes the defined level of autonomy, the technological and regulatory aspects, and current verification, validation, certification and assurance (VVCA) aspects. The current VVCA are mainly based on the standards and regulations for the safety and security of systems that are composed of deterministic functions. The incorporation of ML in RAS will generally necessitate new techniques for architecting, designing, developing, integrating, and testing of these systems. This includes adoption of techniques for achieving functionality in terms of adaptation and learning, accommodating reliability and resiliency, and performing verification of non-deterministic autonomy algorithms, using both formal verification and dynamic testing.

The formal verification of ANN classifiers is a new, though burgeoning, subject which has only established a limited range of results to date. Many studies have examined the extent to which test instances can be perturbed without yielding a change in the assigned class (Weng et al. 2018). This seam of research was a response to the observation that images which have been correctly assigned to a particular class by an ANN classifier, are sometimes assigned to an alternative, incorrect class when subject to minor modifications; if an image is only slightly perturbed, it is unlikely to be assigned a different class by a person, and so if it is assigned a different class by a classifier, the decision is likely to be erroneous. Such minimally modified images have been termed ‘adversarial’ (Szegedy et al. 2014) and might be produced for malicious reasons.

The field of ANN classifier dynamic testing has also been concerned with identifying adversarial images. In (Ma et al. 2018), (Tian et al. 2018), (Pie et al. 2017), the authors propose test coverage metrics which assess the extent to which neural networks are exercised by test datasets. It can be empirically demonstrated that adding adversarial images to a test image dataset tends to increase neural network coverage (Ma et al. 2018).

4 A Novel Measure of Test Dataset Dissimilarity: The Median Fractional Neuron Region Distance

The contribution made by this paper is the introduction of a novel measure which gauges the dissimilarity between a test dataset and a training dataset. This measure adopts and extends some of the concepts reported in the DeepGauge paper on testing criteria (Ma et al. 2018). DeepGauge aims to extend traditional software V&V dynamic testing methods to ANN architectures by defining ANN test coverage criteria. We mirror the DeepGauge paper in focussing on ANN classifiers applied to imagery.

The metrics proposed by the developers of DeepGauge are all based on the range of values output by the neurons within the neural network architectures under test. The paper defines a number of test coverage measures. At the core of the formulation of these measures are two complementary concepts. The authors term these the "major function region of a neuron" and the "corner-case region of a neuron", and provide formal definitions for both. When submitted to a trained network, the images used for training will generate an output value for each neuron. The major function region of a neuron is the interval between the greatest and least output values generated for that neuron by the set of training images. The corner-case region of a neuron is then defined as the complement of the major function region: it is the set of values outside of that interval, and within $-\infty$ and $+\infty$. If a test image is submitted to a network, then each neuron will produce an output value which might lie in the major function region or in the corner-case region. The set of neurons whose output values lie in the corner-case region can be considered to be 'corner-activated' (our term). This is illustrated in Figure 1. This conception is in keeping with the discussion presented in section 2, in that a corner-case is considered an outlier. This paper further designates a neuron as being 'upper' or 'lower' corner-activated based on whether the activation range is violated at the lower or upper bound. A network as a whole is considered to be located in its corner-case region, after processing some input image, if at least one of its neuron output values lies in a (neuron) corner-case region.

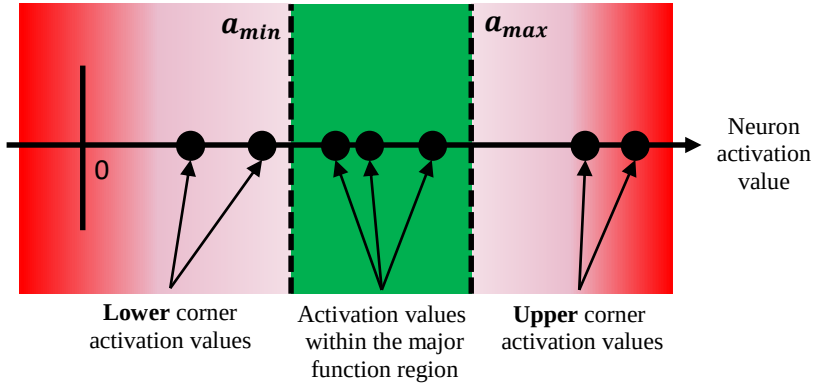


Fig. 1. Illustration of the definition of upper and lower corner activation values for a given neuron, based on the maximum and minimum activation values (a_{min} and a_{max}) recorded for that neuron over the set of training images. The green region spans the major function region, i.e. the range of neuron output values observed when processing the set of training images. By definition, only test images can produce neuron output values outside of the major function region and in the zone denoted the corner-case region. The axis has a zero marked to the left because it is being assumed that the neuron is a rectified linear unit (ReLU).

The DeepGauge test coverage criteria are stated in terms of statistics describing how many neurons are corner-activated in a network, and also the pattern of these activations, over a whole test dataset. For example, the DeepGauge paper introduces the "neuron boundary coverage" which is a measure of the extent to which a network has been exercised by a given test dataset. Given that a neural net has been applied to a test dataset, the measure is defined as the sum of the number of neurons which have been 'upper' corner-activated and the number of neurons which have been 'lower' corner-activated, with this sum being normalised by dividing by twice the number of the neurons in the network.

We have formulated a novel distance which returns a value for a given test instance and a particular training dataset. We have also defined a normalised form of this distance. Our novel dataset dissimilarity measure is the median of this normalised distance over a test dataset. These measures apply to ANN classifiers.

The first distance is termed the Neuron Region Distance (NRD), which is an extension of the DeepGauge measures, and is based on the major function and corner-case neuron regions. The NRD can be normalised to produce a Fractional Neuron Region Distance (fNRD). fNRD values can be interpreted as indicating the extent to which test instances differ from a training dataset; the values can be used to generate a measure of test dataset dissimilarity. Each test instance in a test dataset will return an fNRD value. Our dataset dissimilarity measure is the median of the set of fNRD values returned for an entire test dataset: the median fNRD. The fNRD and the median fNRD will be mainly discussed for the remainder of this section.

We show that the median fNRD relates monotonically to classification performance, and thus helps add context to a quoted performance figure (such as accuracy or recall). Our novel distances and measure can also be used as additional analytical tools for test coverage assessment, for example in addition to those offered by DeepGauge.

When an ANN classifier is applied to imagery, each test or training instance will take the form of an individual image. The fNRD is then a measure of the difference between a test image and a dataset of training images. The images comprising a training dataset can be treated as points in an input space, for example the space of all images, where each axis corresponds to a pixel. If each training image is mapped to a feature vector, then the dataset can also be represented in feature space, where each axis corresponds to a feature vector component. When training a classifier to perform some classification task, the set of training images employed will not, in general, be the optimal training set. The optimal population of training images will be distributed in input or feature space. The actual training set will comprise sample images drawn from this distribution. The performance of a classifier is partly dependent on the spatial relationship between the sample images in the actual training dataset and the optimal image distribution. A test image might arise in a region of space where the optimal image distribution has not been densely sampled. For example, the test image might be distant (by some measure in the space) from the majority of the training images. For such a case, misclassification is more likely. Therefore, since the distances between images, or between an image and an image dataset, can influence classification performance, the measurement of such distances has utility. There are many ways to measure the distances between images, and several spaces in which these measurements can be made1. The NRD and fNRD are additions to the set of measures for determining image proximity. In line with this discussion, a conjecture can be proposed - and ideally this notion would be theoretically and empirically underpinned - that test images with a greater fNRD, that lie further from the training dataset, will be more likely to be misclassified. If this were the case, then it can be further conjectured that the median fNRD would have a decreasing monotonic relationship with classification performance.

Suppose that a neural network is trained with an image dataset, and also that multiple test datasets are prepared. By finding the fNRD of each image, the median fNRD can be found for each test dataset. We suggest two potential behaviours in the context of object classification:

Type 1: If the performance of the network for each test dataset significantly decreases as the median test dataset fNRD increases, then the network is not generalising well to more distant data.

[1] There is some evidence for the counter-intuitive behaviour of distance metrics in high-dimensional spaces (Aggarwal, Hinneburg & Kiem, 1973).

Type 2: On the other hand, if performance is stable across a range of median dataset fNRD values, then classifier generalisation is good.

Having established this novel measure, the median fNRD, an experimental investigation is required to assess its effectiveness for placing performance evaluation in the context of test dataset dissimilarity. The issues to be addressed include:

Issue 1: Is there a positive relationship between the median fNRD of a test dataset and some alternative, perhaps intuitive, measure of the **dissimilarity** of the test dataset to the training dataset? If this is the case, then it provides evidence that the median fNRD is an effective indicator of significant test dataset dissimilarity, that it will reflect real-world, noticeable changes between datasets. This activity is intended to establish the measure's potential as a practical tool.

Issue 2: Can the measure be employed to evaluate the ability of an ANN classifier to **generalise**? One approach is to determine the relationship between classification performance and the median fNRD of multiple test datasets.

5 Experimental Investigation into the Utility of the Median fNRD Measure

The experiments described in this section were designed to reveal how the performance of an ANN classifier changes when the median fNRD of the datasets used to test the network is varied. Specifically, we were interested in manipulating the median fNRD for a given base test set, and observing whether this was related to any observed changes in the performance of a trained network.

5.1 Manipulating the median fNRD

Broadly, there are two approaches to preparing a collection of datasets which exhibit a range of median fNRD values: choosing a number of varied, existing datasets to use for testing, or applying a series of transforms to a fixed test dataset, for example by adding noise or blurring. The former is difficult to do, as understanding what makes two images "different enough" to increase the fNRD by a particular value, or to increase the chances of misclassification by a set amount, is not obvious at the outset, and requires knowledge of both the underlying datasets and how the network was trained. For example, a classifier for cars trained on images taken from a side-perspective would likely struggle to classify a car

from an aerial image. This probably would not be the case for a network trained on a large dataset of images with a range of perspectives. Rather than curating datasets which yield higher or lower median fNRD scores, we opted for the second approach and chose to apply simple transformations to distort the images. This was based on the knowledge that these perturbations typically reduce performance for both machine-vision algorithms and humans alike (Geirhos, et al. 2018). To be specific, we systematically generated a series of test datasets from a source dataset by adding noise of progressively greater variance to the source.

Our investigations centred around two well-known datasets: 1) the original Modified National Institute of Standards and Technology dataset (MNIST), which features images of handwritten digits (LeCun, Cortes & Burges 2019), and 2) Fashion MNIST (Fashion 2019), which features images of items of clothing. We refer to the MNIST data as the "Base" dataset.

5.2 Building and Training Classifiers

The LeNet5 convolutional neural network (CNN) architecture (LeCun et al. 1998) was used to build classifiers for both the Base MNIST and Fashion MNIST datasets. This choice was made to allow comparison of our computed DeepGauge-based statistics with those reported in the original paper (Ma et al. 2018). This architecture is also relatively lightweight, with 60k parameters, and therefore fast to train, even without access to intensive GPU processing.

In order to stabilise computed statistics, five class-balanced training, validation and test splits were drawn in the ratio 819: 81: 100 to create five copies of each network. Mean values for performance metrics and median fNRD scores were calculated over the five networks.

Training was carried out using the ADAM (Kingma Ba 2017) optimisation routine, and all networks were implemented using Keras with a Tensorflow backend. Hyper-parameters and learning rate schedule were set as per the original LeNet5 paper (LeCun et al. 1998).

To standardise training for the two networks, early-stopping was implemented based on the accuracy computed on a validation dataset. The criterion for halting training was an inter-epoch difference in validation accuracy of $< 0.5\%$. The model with the highest observed validation set accuracy was then retained.

5.3 The Effect of Transformations on the Median fNRD and on Classifier Accuracy

In line with the Issues listed at the end of section 4: in Issue 1, the potential role of the median fNRD for assessing test dataset dissimilarity needs to be assessed with regard to an alternative and intuitive method for measuring dissimilarity. To

that end, a series of test datasets featuring noise of progressively greater variance was generated from a source test dataset. A second series was also produced from an alternative source. The median fNRD was then measured for each test dataset to see if it rose with increasing noise variance. Such a rise would indicate that the median fNRD has an increasing monotonic relationship with the variance of additive noise.

Issue 2 is to investigate whether classification performance has a decreasing monotonic relationship with median fNRD, as conjectured. Since additive noise is known to reduce classification performance (Geirhos, et al. 2018), the same two test dataset series used to address Issue 1 were also used to investigate the relationship between the median fNRD and classification competence.

The two source test datasets were drawn from the Base MNIST and the Fashion MNIST datasets, and derivative datasets were generated by adding noise. Thus in addition to the two source datasets, for which the additive noise can be considered to be zero, 14 further datasets were produced, 7 generated for each of the two source datasets by adding noise of progressively greater variance. Figure 2 shows the effect of these 8 different additive noise variances on the recall2 for a selection of classes (as assigned in the Base MNIST and Fashion MNIST datasets). The blue trace corresponds to the Base MNIST family of datasets and the orange trace to the Fashion MNIST family of datasets. The full results for all classes can be found in Figure 5, 8 Appendix A.

We use recall to indicate classification success due to both source datasets being multiclass. If the datasets had been binary, recall could have been plotted against false alarm rate to produce a receiver operating characteristic (ROC) curve, but this approach does not naturally transfer to the multiclass case. Each graphed recall score is an average, calculated over 5 repeats, as discussed above. The plots show that as the additive noise increases in strength, recall degrades monotonically in practically all cases. There are small increases for classes 0, 2 and 9 for Fashion MNIST, though this is isolated to the least additive noise variance (for variances greater than zero), after which a decrease in recall is observed.

With the same gradated degrees of noise variance applied to the source test datasets, i.e. using the two series of test datasets prepared as described, we calculate the fNRD for each image in each dataset, and recover the median fNRD value per dataset. According to Figure 3, the median fNRD appears to increase with greater additive noise variance, and this is observed for both datasets. This makes intuitive sense: adding noise shifts the distribution of the test images away from that of the images used to train the network. This establishes, in this particular case, that the median fNRD has an increasing monotonic relationship with additive noise variance.

[2] Recall is defined as the number of true positives over the sum of the true positives and false negatives.

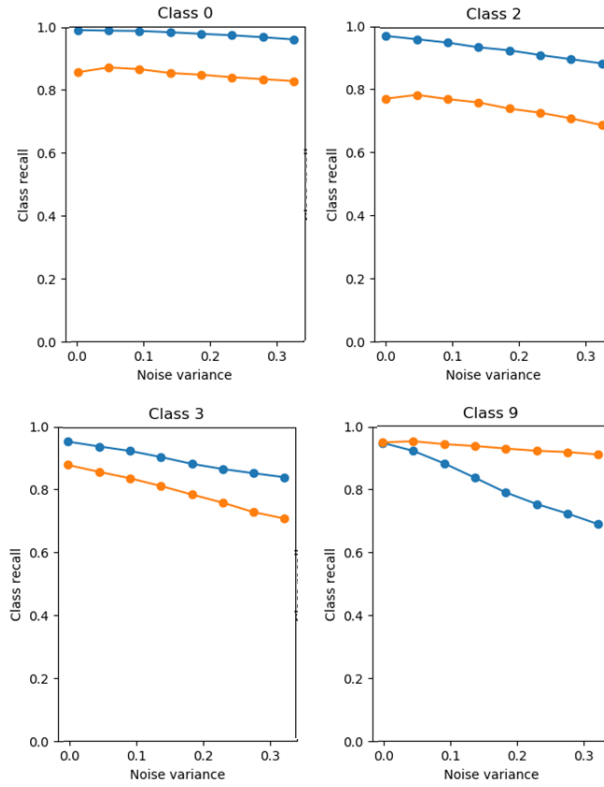


Fig. 2. Recall scores for a selection of classes in the Base MNIST (blue) and Fashion MNIST (orange) datasets as a function of additive noise variance.

Our next step is to determine the relationship between classifier performance and the median fNRD. The scatter plot of classification accuracy[3] against the median fNRD score (Figure 4) suggests that the relationship between the two quantities is monotonic and decreasing. In other words, test datasets for which the median fNRD is higher, are associated with reduced classification accuracy. This is an example of classification behaviour Type 1 as listed in Section 4: both networks' performance scores suffer as the median fNRD, and the additive noise variance, increases, which indicates that they are unable to generalise well to images which feature large amounts of added noise.

Classification behaviour Type 2, stable performance as the median fNRD varies, is not observed here. In Figure 4, the scatter plots for both datasets show very similar behaviour. If the performance curve for one dataset had deviated much

[3] Accuracy as a metric is defined as the fraction of correctly classified test instances.

less with increasing median fNRD than the curve for the other dataset, we would have gained some evidence that the network delivering the more stable performance would be better able to generalise to unseen, noisy data.

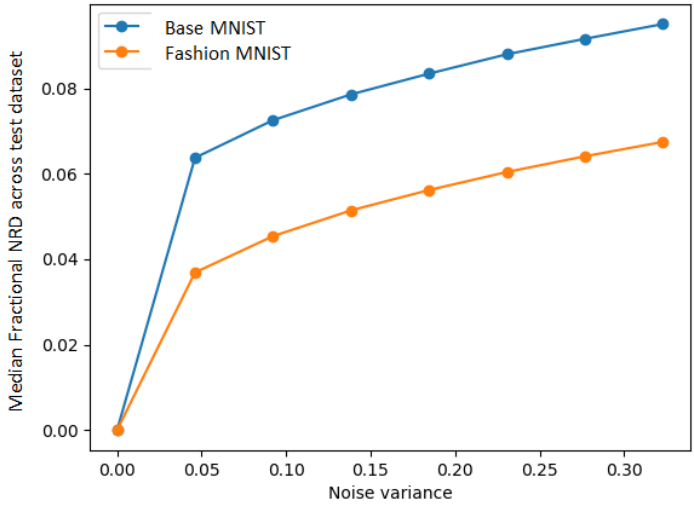


Fig. 3. Variation in Median Fractional Neuron Region Distance (fNRD) per dataset against the variance of additive noise applied to the Base MNIST or Fashion MNIST source test datasets.

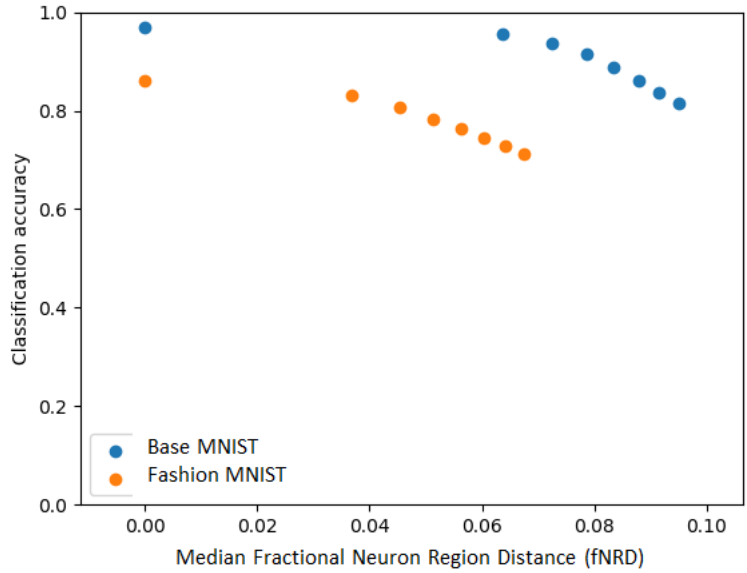


Fig. 4. Scatter plot showing the relationship between classification accuracy and median fractional Neuron Region Distance (fNRD) for the Base MNIST and Fashion MNIST series of noise-augmented datasets.

6 Future Work

We are currently undertaking a finer-grained examination of the relationship between the fNRD and classifier performance by producing results for subsets of datasets.

So far, we have examined classic datasets, using Base MNIST and Fashion MNIST as the source of training and test data. Applying the fNRD approach to more extensive public or industrial datasets would be an obvious next step, and would allow us to check whether our results hold for more useful application areas. Although imagery and CNNs have featured in this study, the measures can be applied to any form of input data, and other ANN architectures.

We also intend to apply other transformations to source datasets, in the same manner that additive noise was applied in the set of experiments described. These would include further image processing transformations that are known to degrade quality and perceptual performance, such as rotations, and blurring by means of Gaussian filters. As before, the focus would be on the relationship between classification accuracy and fNRD values.

7 Conclusions

The Assuring Autonomy International Programme (AAIP) is developing a Body of Knowledge which will serve as a reference for the safety assurance of autonomous systems (Hawkins 2019). Our proposed approach addresses several assurance objectives within the document such as: Sufficiency of training, Verification of the learned model, Using simulation, and Identifying ML deviations.

Novel approaches for verifying the correctness, performance, and behaviour of ANN classifiers will raise levels of confidence in their robustness and safe operation, and in their suitability for real-world deployment. Our contribution, developing a measure which allows classifier performance to be given as a function of test dataset dissimilarity, is a step towards this end. The measure, and the associated function, can be used to refine the expression of classifier requirements, enabling more systematic and informative verification.

In this paper, the neuron region distance (NRD) and fractional neuron region distance (fNRD) have been introduced. They indicate the difference between a test instance (for example a test image) and a set of training data. A statistic describing these distances, the median fNRD, has been employed as a novel measure to assess dataset dissimilarity. These distances and the measure can be used to evaluate ANN-based object classification algorithms.

The experiments conducted have shown that classification accuracy is a monotonically decreasing function of median dataset fNRD. This was established by determining the relationship between classification performance and median

fNRD for multiple test databases. This finding supports the conjecture made in section 4. The result also illustrates how the median fNRD could be used to assess the ability of ANN classifiers to generalise to test datasets of increasing dissimilarity.

Empirical evidence has also been presented which suggests that the median fNRD can be expressed as a monotonically increasing function of an alternative and intuitive measure of test dataset dissimilarity, namely the variance of noise added to a base training set. This indicates that the median fNRD could prove to be practical measure, where the values returned relate to significant differences between real-world datasets.

Requirements addressing the performance of ANN classifiers could make use of the fNRD. For example, the form of an acceptable relationship between classification performance and median test dataset fNRD could be specified in a requirement. Verification would then need to provide evidence that the relationship is in accord with the stated constraints.

Acknowledgment

The work undertaken to produce this paper was fully funded by Thales UK Research, Technology and Innovation (RTI). The authors would like to thank their colleagues, especially Dan Jeffery, Ben Pritchard and Sam Budgett, for their contributions towards this work.

References

Aggarwal Charu C., Hinneburg A., Keim D. A. (1973), "On the surprising behaviour of distance metrics in high-dimensional space", Lecture Notes in Computer Science book series (LNCS, volume 1973), [online]. Available: https://bib.dbvis.de/uploadedFiles/155.pdf.

Asgari Hamid, Farrell J., Pritchard B. (2019), "Review of Regulatory Issues of Robotic Autonomous Systems: Learning for Civil Nuclear Industry", Engineering Safe Autonomy - Proceedings of the 27th Safety-Critical Systems Symposium, pp. 135-155, 6th of Feb. 2019, Bristol, UK.

ChicoState (2016), ChicoState/SoftwareEngineering, "Software Testing - Edge & Corner Cases and Boundary Testing", Apr. 2016, [online]. Available: https://github.com/ChicoState/SoftwareEngineering/wiki/Software-Testing-Vocabulary.

Chung Yeounoh, Haas Peter J., Kraska Tim, Upfal Eli. (2019), "Learning Unknown Examples For ML Model Generalization", V2, 11 Oct 2019, [online]. Available: https://arxiv.org/abs/1808.08294v2 .

Fashion (2019), "Fashion MNIST", accessed Sept. 2019, [online]. Available: https://github.com/zalandoresearch/fashion-mnist.

Geirhos Robert, et al. (2018), "Comparing deep neural networks against humans: object recognition when the signal gets weaker", 2018. [online]. Available: https://arxiv.org/pdf/1706.06969.pdf.

Hawkins, Richard. (2019), "Body of Knowledge - Structure and Scope", Assuring Autonomy International Programme, University of York, v1-2, April 2019. [online]. Available: https://www.york.ac.uk/assuring-autonomy/body-of-knowledge/ .

Hond Darryl, et al. (2018), “Test Coverage Metrics for Artificial Intelligence”, Thales UK Technical Document, RN 78, June 2018.

Kingma Diederik P., Ba J. (2017), “ADAM: a Method for Stochastic Opimization”, Version 9, Jan. 2017, [online]. Available: https://arxiv.org/abs/1412.6980.

LeCun Yann., et al. (1998), “Gradient Based Learning Applied to Document Recognition”, Proceeding of IEEE, Volume: 86, Issue 11, Nov. 1998, [online]. Available: http://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf.

LeCun Yann, Cortes C., Burges C.J.C, (2019), “The MNIST Database of Handwritten Digits” accessed Sept. 2019, [online]. Available: http://yann.lecun.com/exdb/mnist/.

Ma Lei, et al. (2018), “DeepGauge: Multi-Granularity Testing Criteria for Deep Learning Systems”, V4, 14 Aug. 2018, [online]. Available: https://arxiv.org/abs/1803.07519

Pei Kexin, et al. (2017), “DeepXplore: Automated Whitebox Testing of Deep Learning Systems”, Version 4, [online]. Available: https://arxiv.org/abs/1705.06640.

Szegedy Christian, et al. (2014), “Intriguing properties of neural networks”, Version 4, [online]. Available: https://arxiv.org/abs/1312.6199 .

Tian Yuchi, et al. (2018), “DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars”, Version 2, [online]. Available: https://arxiv.org/abs/1708.08559.

Weng Tsui-Wei, et al. (2018), “Evaluating the Robustness of Neural Networks: An Extreme Value Theory Approach”, Jan. 2018, [online]. Available: https://arxiv.org/abs/1801.10578.

8 Appendix A

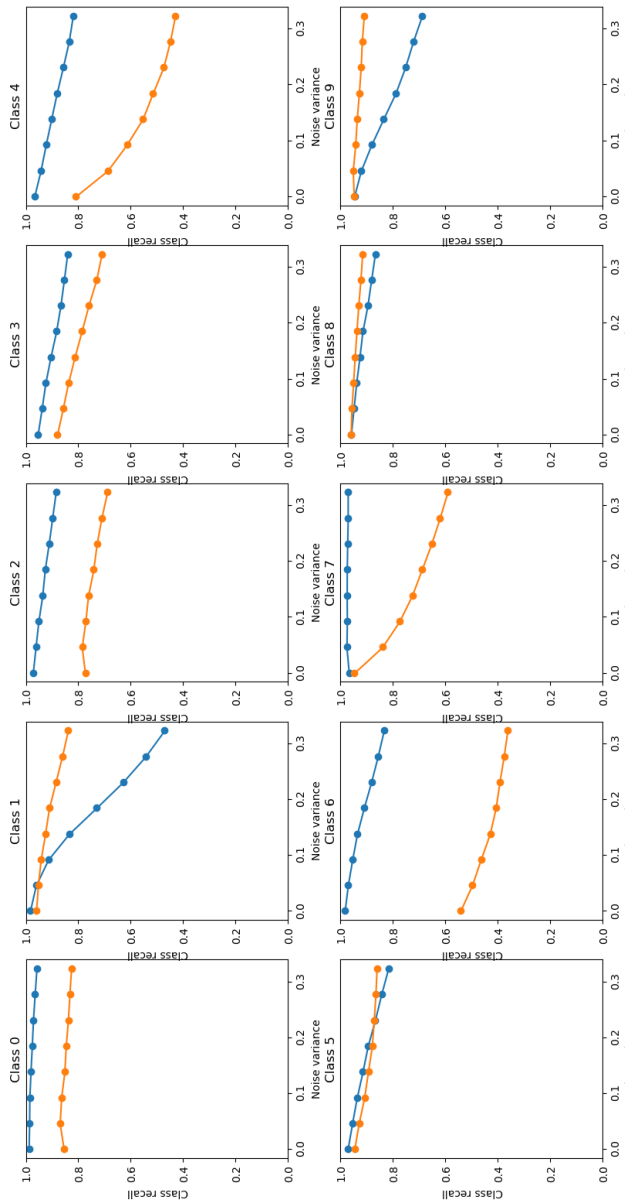


Fig. 5. Recall scores for each class in the Base MNIST (blue) and Fashion MNIST (orange) datasets as a function of additive noise variance.

Satellite Navigation ~ Where Are We Going?

John Spriggs

Independent Author and Presenter

Hayling Island, UK

Abstract *The International Civil Aviation Organization (ICAO) only accepted the original satellite navigation constellations (GPS and GLONASS) as a supplementary source of navigation data for civil air transport. This was not because of accuracy (although that is insufficient for some phases of flight), but because of the lack of integrity. Position errors due to a satellite fault, for example, can go undetected. This paper briefly summarises provisions specified by ICAO to make a trusted Global Navigation Satellite System, and looks forward to some new developments in providing trusted information to support the integrity of navigation solutions, which could also be used in other domains, e.g. autonomous vehicles.*

1 Introduction

Many people will attend the 2020 Safety-critical Systems Symposium in York having travelled by car, and assisted by a GPS-based navigational aid[1]. Such a satellite navigation receiver, or "Sat Nav", is something to check from time to time to confirm the planned route is still being followed; it may also provide advice such as, "Turn left in two hundred yards". It does not automatically direct the vehicle. Would it be 'safe enough' to direct the vehicle?

There is a stretch of motorway (freeway) in England that runs alongside a local road. Drivers on that route have reported being told by their Sat Nav to "Return to the motorway", because the navigation receiver perceives them to be suddenly on the local road. The navigation solution is not sufficiently accurate for following that particular road layout.

Some commercial aircraft autopilots are informed by satellite navigation, and so this technique *is* used to direct vehicles. Presumably, the pilots are not told to

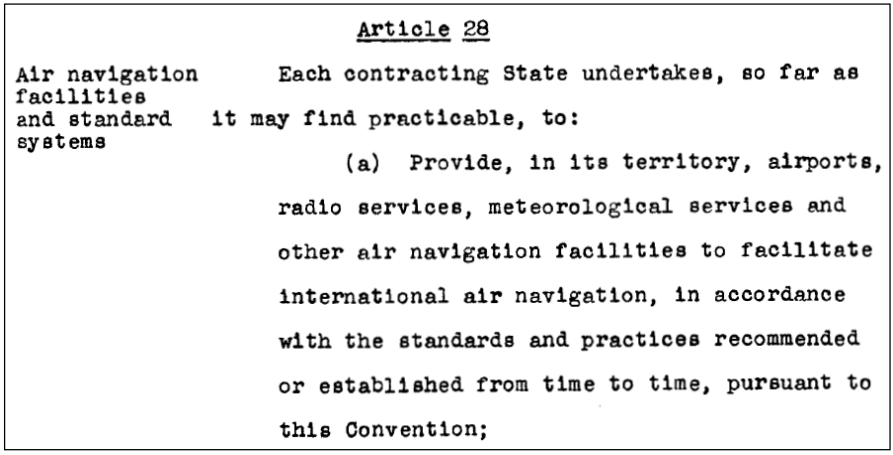
[1] GPS is the Global Positioning System, also known as Navstar, a satellite navigation system developed and managed by the United States Department of Defense.

© John Spriggs 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

"Return to the airway", so what is different? The avionics system is not just using GPS, it is using GNSS, the Global Navigation Satellite Service, as specified by ICAO, the International Civil Aviation Organization. GNSS is defined (ICAO, 2018a) as:

> A worldwide position and time determination system that includes one or more satellite constellations, aircraft receivers and system integrity monitoring, augmented as necessary to support the required navigation performance for the intended operation.

This definition of GNSS is from an Annex to the Chicago Convention on Civil Aviation, which established rules of airspace, and set up ICAO to become a United Nations agency to coordinate international air travel and maintain the rules.



Article 28

Air navigation facilities and standard systems

Each contracting State undertakes, so far as it may find practicable, to:

(a) Provide, in its territory, airports, radio services, meteorological services and other air navigation facilities to facilitate international air navigation, in accordance with the standards and practices recommended or established from time to time, pursuant to this Convention;

Fig. 1. Extract from the original Convention on International Civil Aviation (1944)

Notice that the GNSS definition does not explicitly mention accuracy. Vehicle system designers will make provisions to obtain the accuracy needed for their intended applications, be it keeping to a taxiway or an airway, or finding and staying on the glidepath into an airport runway. The parameter that is important for safety is *integrity*, the trust users can have that the system will provide the required performance. This is mentioned in the GNSS definition and explored further in Section 4.

A basic navigation service may be obtained from a single constellation of navigation satellites; this is not sufficient for most civil aviation applications, which require better accuracy and/or integrity. To establish this, the GNSS definition states that signals from a satellite constellation, currently GPS or GLONASS[2],

[2] GLONASS is the Globalnaya Navigazionnaya Sputnikovaya Sistema, a satellite navigation system originally deployed by the Soviet Union and now developed and managed by the Russian Roscosmos State Corporation for Space Activities.

are to be "augmented as necessary", i.e. used in combination with additional equipment and sources of information to provide the navigation services to aviation users.

The Standards and Recommended Practices of Annex 10 to the Chicago Convention specify three types of augmentation system particular to aviation: space-based, ground-based and aircraft-based (ICAO, 2018a). If you deploy or operate equipment to implement an augmentation scheme in an aircraft, it must satisfy the associated ICAO-promulgated requirements. Annex 10 defines:

Satellite-based Augmentation System (SBAS). A wide coverage augmentation system in which the user receives augmentation information from a satellite-based transmitter.

Ground-based Augmentation System (GBAS). An augmentation system in which the user receives augmentation information directly from a ground-based transmitter.

Aircraft-based Augmentation System (ABAS). An augmentation system that augments and/or integrates the information obtained from the other GNSS elements with information available on board the aircraft.

Understanding these alternative augmentation systems requires some knowledge of how position measurements are made.

2 An Aside on Position Measurement

Imagine taking a set of rulers and getting someone to use each of them to measure your x-y position from the corner of the room; they would get a scatter of answers, with errors arising from their method and from the rulers themselves. Plot them, and the resulting map may look like Figure 2 wherein the corner of the room is at (x, y) = (0, 0).

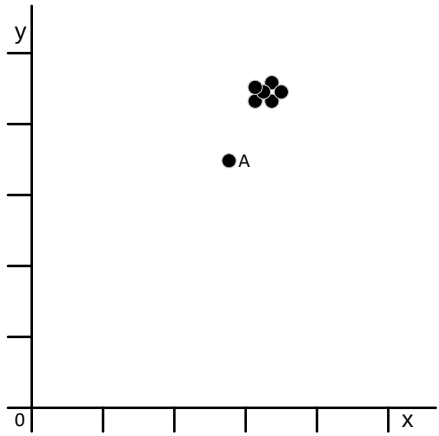


Fig. 2. A Scatter of Measurement Results

Now take the set of GPS satellites that are in view and use sub-sets of them to measure your position on a similar map. You may well get another scatter of answers looking like Figure 2.

It transpires that one of the rulers and one of the satellites each has a significant error. Point A in Figure 2 is obviously different from the others; it is an outlier. We would expect that Point A was the one obtained with the 'bad ruler'. However, Point A is very likely to be the best result obtained from the satellite measurements, because it is the result unaffected by the 'bad satellite'. This arises because satellite navigation is not done by separately measuring Cartesian coordinates with respect to an origin, like we did with the rulers; rather, it uses multilateration. The navigation receiver measures the distances from a set of known points, the satellites, to itself and then works out its position from there.

Now imagine that we are at a fork in the road (in a flat landscape), and the signpost says it is four miles to village A in one direction, five miles to village B in the other direction, and village C is six miles behind us. We could be anywhere on a six-mile radius circle around C, but that intersects with the five-mile radius circle around B at two points. Drawing the four-mile radius circle around A (appropriately scaled of course) on our map decides which point is the right one. See Figure 3.

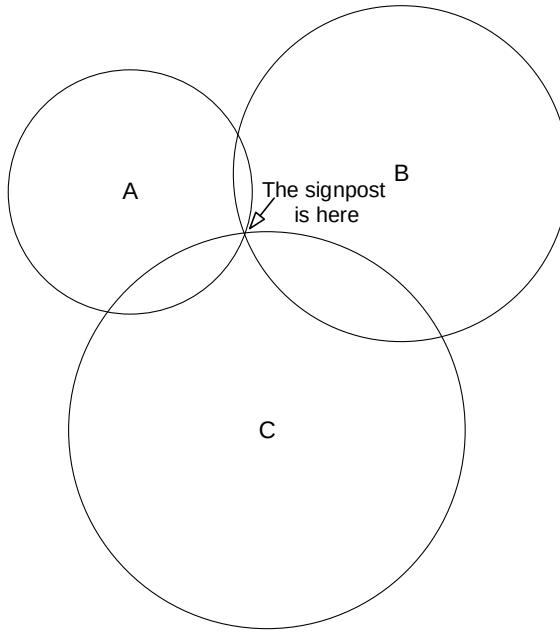


Fig. 3. The Intersecting Circles Model

The principle is the same with the satellites' measurements, except that we would take the solution down from a sphere to a circle, to two points, and then to one. Alternatively, you could construct notional circles/spheres around your own position, each of which intersects with one of the villages or satellites, and then solve for a common centre.

The problem is that in practice it does not work out quite that easily; the circles or spheres do not intersect at a point due to measurement uncertainty. The degree of uncertainty in the solution depends on, amongst other things, the geometry (and it is not just that the roads to villages A, B and C are not straight, or not knowing to which location in the village the signpost is pointing).

As with any measurement process, there will be uncertainty in the results and so we are not intersecting perfect Euclidean circles or Platonic spheres, but something with 'thickness', i.e. the radius is not just R but $R \pm e$, where e is a representation of the measurement error. In the left hand part of Figure 4 the diagonal shading shows the uncertainty in an individual measurement, and the hash shows the uncertainty in the location of the intersection point.

Fig. 4. Geometric Dilution of Precision

If the geometry were to be different, we would get a larger uncertainty in the solution as shown by the hashed area in the right hand part of the figure. This effect is called Geometric Dilution of Precision, i.e. the uncertainty in the result that is a function of geometry.

This is all straightforward, but glosses over a significant aspect. Irrespective of whether you use the intersecting circles model, or solve for a common centre, you will need to know the co-ordinates of the villages to specify the circles needed to work out your co-ordinates. How does a navigation receiver know *where* each satellite is? Furthermore, the actual distance measurement is done by timing how long a signal takes to get from the satellite at the speed of light, so how does the receiver know *when* the satellite was when the signal left it?

To look at the clock problem, we need to return to the fork in the road, where the signpost has been refurbished, removing the distances (see Figure 5).

Fortunately, the town clocks at A, B and C are synchronised to an atomic reference clock and so, if we were to measure the times of arrival of the sounds when they strike the hour, we would be able to work out the distances easily, knowing the speed of sound. At least, we would if the local receiver clock were also synchronised to the atomic reference, and it is not; giving us three equations for four unknowns, which is insufficient. Each equation can be rearranged to show that the distance from each point, divided by the speed of sound, is the time the sound arrived minus the time when it started out.

Fig. 5. The Refurbished Signpost Has No Distances

The time measurements made were arrival times against a different time-base, which give what is known as *pseudo-ranges*, rather than the required distance. There is a simple work-around; rather than working with the individual measurement, we can take differences between pairs of arrival time measurements. This cancels out the time of transmission, and gives us three equations for three unknowns, which is tractable despite losing us the intersecting circles model. There is still uncertainty due to the geometry, the measurement method and variations in the speed of sound from the cold, wet, September day on which Figure 5 was captured to the hot, clear, high-pressure, days earlier in the Summer.

The same approach can be taken with the satellite solution, but could it be less uncertain, because, as "everyone knows", the speed of light is constant? In fact, the speed of light is only a constant in free space and, unfortunately, the space between the satellite and receiver is not free in this sense, being encumbered by the troposphere, which slows down light (and radio waves, and sound), and by the ionosphere, which interferes more dynamically with the signals. Consequently, it is not just geometry and the method of measurement introducing uncertainty.

In practice, there are other error sources not addressed here, as this is just an overview. The interested reader can find them briefly discussed in the proceedings of a previous Safety-critical Systems Symposium (Spriggs, 2003).

Some errors are mitigated by system design. Others can be compensated for in the receiver using correction factors to improve accuracy. These data are pro-

vided by the satellites in addition to the ranging signals in a "Navigation Message", which also includes orbital parameters (almanacs[3] and ephemerides[4]), so that the positions of the satellites can be calculated, as required for the solution.

3 New Frequencies

One of the new developments in GNSS is being introduced specifically to reduce measurement errors due to the ionosphere. It has been observed that the ionospheric effects are a function of frequency, such that, if the ranging signals were to be transmitted on two frequencies, the receiver would be able to apply corrections largely removing the ionosphere-related error. The original GPS design includes two frequencies for precise position measurement, but only one of these is available for civilian use.

All navigation signals used for civil aviation are required to be within the frequency bands specifically allocated by the International Telecommunication Union to the Aeronautical Radio Navigation Services (ARNS). This is to ensure that all signals used for aviation purposes are not affected negatively by other transmissions, which are kept out of those bands. The ARNS allocation is a subset of frequency bands allocated to Radio Navigation Satellite Services in general.

The original GPS civilian frequency, known as "L1" and centred at 1575.42 MHz, has now been joined by "L5" at 1176.45 MHz. These frequencies are in separate ARNS sub-bands, but can both be used for aircraft navigation with a suitable receiver. Similarly, GLONASS offers a new frequency "L3", which is in the same band as GPS L5.

Note, however, that the Standards and Recommended Practices of Annex 10 (ICAO, 2018a), which specify, in line with the GNSS definition, receiver performance, augmentation systems, and integrity monitoring, will need to be updated to include specification details of L5 and L3 before such a receiver can be approved for use on airliners. Such an update is currently being developed by ICAO working groups.

---

[3] In antiquity, an almanac was a document published annually containing predicted events, such as sunrise and sunset times, eclipses, tide tables, etc. In contrast, a new GPS Almanac is uploaded for transmission by the satellites every six days; it contains information on the entire constellation, including coarse orbit data, and various correction factors.

[4] An ephemeris gives data on the trajectory of astronomical objects, and is used in astronomy and celestial navigation, i.e. the art of observing stars, etc., and using the time of the observations with their ephemerides to work out where the stars are and, hence, estimating your own position. The GPS Ephemeris transmitted by each satellite gives its current and predicted locations, clock corrections, etc. It is updated every two hours.

4 Integrity

Like Safety, Integrity means different things to different people. Even in my narrow context of international civil aviation, there is more than one definition enshrined in annexes to the Chicago Convention. Annex 10 itself has a number of integrity definitions specific to particular systems (ICAO, 2018a), but also has a general "Integrity" definition, which states[5]:

> A measure of the trust that can be placed in the correctness of the information supplied by the total system. Integrity includes the ability of a system to provide timely and valid warnings to the user (alerts) [when the system must not be used for the intended operation (or phase of flight)].

Whereas Annex 15 (ICAO, 2018b) has "Data Integrity":

> A degree of assurance that aeronautical data and its value has not been lost or altered since the origination or authorized amendment.

Note that the Annex 15 definition refers to loss, whereas that from Annex 10 does not; this is because of the context.

Annex 15 addresses aeronautical data, for example the surveyed position of a ground-based navigational aid. This is required to be correct, but it could have become corrupted. Some possible corruptions are believable, and so potentially dangerous, whilst others are obvious, in which case the data are considered lost to the user. For example, a corruption apparently putting the navigational aid on another continent would be obvious, and the data would be discarded.

In the Annex 10 context, it is the signals in space from our navigational aid that are considered. It does not really matter if the system is lost, i.e. ceases functioning, because that is obvious and other provisions (such as inertial navigation) will be in place, but it does matter if the signals are believable, but dangerously wrong. If a navigation aid is transmitting corrupted signals, it is required either to shut down or to provide an indication (known as an Alert) that the signal is "false guidance". That is why ground-based navigational aids are specified by ICAO to have separate local monitors to detect when the signals emitted are out of specification.

Two out of the three ICAO-specified GNSS augmentation systems extend this monitoring principle, but with a bit more independence, because the monitor is provided by a third party, not by the provider of navigation signals as is the case with ground-based navigational aids.

The next section gives a brief overview of the three types of augmentation system, and a subsequent section considers a new development that will improve ABAS performance.

---

[5] The text in square brackets does not appear in the original definition, but appears in Attachment D to the Annex, wherein the definition is restated in a discussion of satellite navigation

5 Augmentation Systems

Now, step away from the vehicle, and set up a GPS receiver at a fixed location on the ground. A surveyor can measure the position with greater accuracy than that of the navigation solution (position estimate). The solution will change over time as the various error sources change. It can be used in association with the surveyed position to produce 'corrections', which can then be shared with other users nearby so that they can derive better position estimates for themselves.

There must be some threshold over which it may be unwise to produce corrections; this is when the perceived error is so great that there must be something fundamentally wrong. In this case, the nearby users should be sent an Alert, saying that they should not trust their navigation solution. This example is a very local augmentation system; those defined by ICAO have a wider scope.

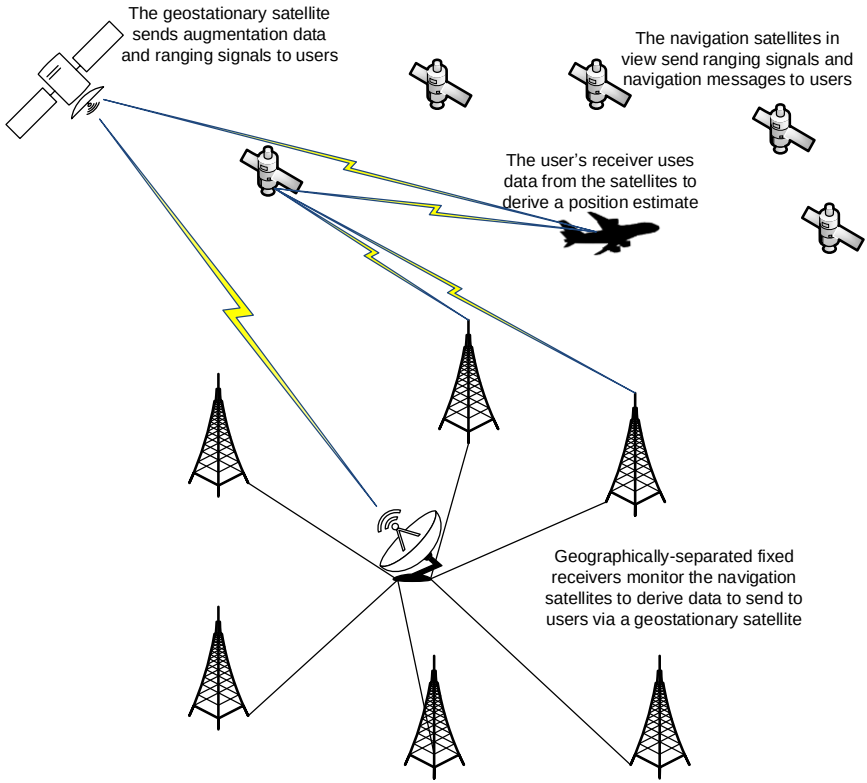


Fig. 6. Overview of SBAS Operation

The SBAS uses a set of monitoring stations whose locations are specified to cover the region of interest (as shown in Figure 6, which has most signals omitted for clarity). Data from these is processed to provide information with which the user

can improve their navigation solution, and this information includes 'flags' indicating whether or not each satellite can be used to generate a solution. The augmentation messages are usually passed to the users via a geo-stationary satellite (which can also be used as an additional ranging source, thus further improving the solution).

Other data in the messages, error estimates, can be used to improve integrity by calculating a 'protection volume' around the aircraft's true position in which the calculated position will lie. Separate vertical and horizontal protection levels are calculated to define this volume. Annex 10 (ICAO, 2018a) describe the protection levels thus:

> The horizontal protection level provides a bound on the horizontal position error with a probability derived from the integrity requirement. Similarly, the vertical protection level provides a bound on the vertical position.

The definition is more detailed in the Minimum Operational Performance Specification for airborne receivers (RTCA, 2016). There are two definitions; the horizontal protection level, stated here, and the directly equivalent vertical protection level (WGS-84 is the co-ordinate reference system used by GPS (World Geodetic System, n.d.)):

> The horizontal protection level is the radius of a circle in the horizontal plane (the plane tangent to WGS-84 ellipsoid), with its center being at the true position, that describes the region assured to contain the indicated horizontal position. It is the horizontal region where the missed alert requirement can be met. It is based upon the error estimates provided by SBAS.

Traditionally, civil aviation addresses vertical and horizontal aircraft separation separately, and measures vertical distance in thousands of feet, and horizontal distance in nautical miles; the protection volume is thus more like an ice-hockey puck (cylinder) in shape, rather than the American football (ellipsoid) that may have been expected (see Figure 7).

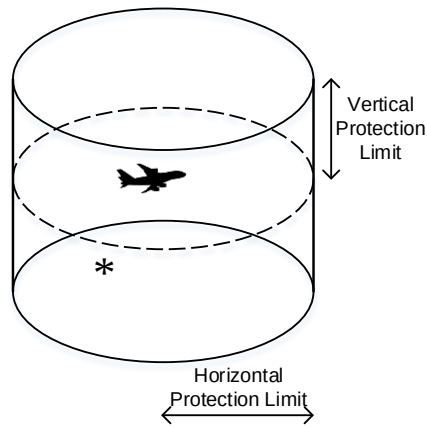


Fig. 7. A Representation of the Protection Volume

In Figure 7, '*' represents the computed solution, which, in this example, is a bit low and a little behind, but still in the volume. At first glance it may seem that, as it is in the protection volume, this solution could be used (and, if it were not in the volume, an Alert would have been raised). This ignores two significant factors:

1. To know where the protection volume is and, hence, whether the computed solution is inside it, we need to know the true position of the aircraft (and, if we knew that, we would not need satellite navigation).
2. The protection volume was defined such that it will always contain the computed position; it is the error bound.

Alert Limits are defined for an application as the maximum allowable position errors in both horizontal and vertical planes. The system is declared unavailable for that application if the protection volume extends outside the equivalent volume defined by the alert limits. The computed solution may in fact be within the alert limits, but we only know that the error bound exceeds the 'safe' limits and so we cannot use the solution.

The standards also specify the Time to Alert, which is the time (after the protection level breach event) in which the system shall declare an Alert. An integrity failure is thus when the protection volume extends outside one or both Alert Limits for greater than the specified Time to Alert. Some authorities use the term Hazardously Misleading Information for the computed solutions presented during an integrity failure.

Examples of SBAS implementation are the WAAS[6], covering most of North America, and EGNOS[7] covering Europe. These systems implement the ICAO requirements and so are managed and operated, on behalf of the system owners, by properly accredited Air Navigation Service Providers (ANSPs), the FAA[8] for WAAS and ESSP[9] for EGNOS. Other systems are available in other regions.

It should be noted that the owners and operators of the satellite constellations themselves, GLONASS and GPS, are not ANSPs and their services are used for many non-aviation applications too. The requirement is that the operator writes to ICAO a 'Letter of Commitment', offering their service for aviation use, and making various undertakings. Similarly, the geostationary satellite operators and providers of other constituent services are not ANSPs either, but are required (in European law at least) to have formal arrangements (such as a Service Level Agreement) in place with the augmentation system ANSP that uses them.

[6] The WAAS is the Wide Area Augmentation System

[7] EGNOS is the European Geostationary Navigation Overlay Service

[8] FAA Is the Federal Aviation Administration, a US Government agency

[9] ESSP is the European Satellite Service Provider, a commercial company whose shareholders are also ANSPs

GBAS is similar in concept to SBAS, but operates in a smaller region, i.e. specific to a single airport, e.g. Frankfurt am Main (or to a local group of airports). The augmentation information is not provided by satellite, but via VHF data broadcasts (in an Aeronautical Communications Band specified by the International Telecommunication Union). Such a set-up, because of the local monitors, can provide better (trustable) accuracy than the SBAS, and could be used for 'blind landing' operations if appropriate procedures were in place.

The third augmentation scheme defined by ICAO is more interesting in the context of this paper, because ABAS is focussed on integrity, rather than on accuracy. Currently, it comes in two flavours (both performed locally to the aircraft). One, Aircraft Autonomous Integrity Monitoring (AAIM), takes benefit from other on-aircraft data sources, such as a barometric altimeter; the other is Receiver Autonomous Integrity Monitoring (RAIM), and does not require any additional information or aircraft-external monitoring or signal processing.

RAIM uses GNSS satellite-derived information exclusively. It takes advantage of the navigation solution being over-determined[10] (when sufficient satellites are in view), and so multiple calculations may be done to identify a 'faulty satellite' (like in Figure 2). The first-generation receivers would raise an Alert, requiring use of alternative navigation systems. More-recent designs exclude the anomalous measurement from the solution, thereby allowing GNSS navigation to continue with confidence and without interruption.

For an aircraft receiver to perform RAIM, signals must be received from a minimum of five satellites (with satisfactory geometry). Users can predict satellite availability and geometry for their intended flight plan, and so will know in advance whether they can use GNSS with RAIM or not. This knowledge will inform the choice of departure and arrival routes and procedures.

The performance standard for airborne receivers (RTCA, 2016) also specifies a protection level for GNSS airborne equipment operating autonomously (this is just horizontal; there is no vertical analogue, as RAIM is to be used only for lateral navigation):

> The horizontal protection level is the radius of a circle in the horizontal plane (the plane tangent to WGS-84 ellipsoid), with its center being at the true position, that describes the region assured to contain the indicated horizontal position. It is a horizontal region where the missed alert and false alarm requirements are met for the chosen set of satellites when autonomous fault detection is used. It is a function of the satellite and user geometry and the expected error characteristics: it is not affected by actual measurements. Its value is predictable given reasonable assumptions regarding the expected error characteristics.

The 'reasonable assumptions' referred to here include such things as the probability of individual satellite failure, which were originally assumed for GPS based

[10] In Mathematics, a set of simultaneous linear equations is said to be over-determined if there are more equations than unknowns.

on reliability analyses, etc., and the predictions made then have since been borne out by experience.

6 New Constellations, New Concepts

Another factor that will improve performance in future is the presence of additional constellations. The original American and Russian ones have now been joined by Chinese and European examples, named BeiDou and Galileo respectively. Before they can be used for civilian air transport operations, they will need to be included in the ICAO Standards and Recommended Practices, i.e. Annex 10, update of which is triggered once Letters of Commitment are accepted by ICAO. The new material will take several years to generate, validate, agree, and publish (this work is currently in progress).

An overall concept of operations is currently being developed to enable use of the two frequencies mentioned previously and the multiple constellations that will soon be available (the new constellations also have additional frequencies). The original satellite navigation receivers used a single constellation, but now more are available, it is sensible to use more than one of them to improve accuracy and/or integrity of position measurements. The concept is known as Dual Frequency Multiple Constellation operations, DFMC. Airborne DFMC receivers are currently in development.

It is not just the Annex 10 changes (and the updated Minimum Aviation System Performance Standards and Minimum Operational Performance Specification, "MASPS & MOPS", with which to implement them) that will be required to operate these new concepts. Each ICAO member state has to publish their approvals for GNSS use in their airspace. This is done, in compliance with Annex 15 (ICAO, 2018b), in the state's Aeronautical Information Publication. The principle is that approvals should be at GNSS element level, i.e. constellation by constellation, frequency by frequency, and augmentation by augmentation. The United Kingdom's Aeronautical Information Publication, for example, currently allows use of GPS at L1 across all the airspace, and EGNOS at L1 as notified through individual aerodrome's Instrument Approach Procedures. Use of RAIM is not made explicit here, but other states include guidance on its use along with their approvals.

As mentioned, current GPS RAIM calculations use failure rate predictions that can now be backed by in-service performance data. Fewer such data will be available for the new constellations, or for their supporting systems, and so initially we will have less confidence in the RAIM solutions using them, but confidence will build over time as more data are collected.

New algorithms are in development (EU-US, 2016) for an advanced RAIM concept for DFMC receivers, 'ARAIM'. The RAIM implementations that are in

service now for horizontal navigation address only a single GNSS measurement failure at a time, whereas ARAIM is an improvement intended to support en-route flight, terminal area manoeuvring, and lateral and vertical guidance for airport approach operations, using dual frequencies from multiple constellations. The proposed ARAIM algorithm will have three main parts (EU-US, 2016):

1. It first checks that its satellite signal measurements are consistent with the nominal performance assumptions. (In conventional RAIM, these were fixed assertions regarding the nominal performance and failure rates of GPS or GLONASS. In contrast, ARAIM has a wider scope and allows data to be changed over time; a ground-based system will generate and provide updates, via an "Integrity Support Message" (ISM) including, for each constellation, the nominal performance and failure rates);
2. If those measurements were found to be consistent, it would compute parameters, such as the figures of merit associated with the geometry, for use in computing the protection limits and in other parts of the navigation solution;
3. Alternatively, if the measurements were found to be inconsistent, it would revisit the calculation excluding a particular satellite, and then repeat this for each satellite in sight until a consistent set is found upon which a trusted navigation solution can be based.

The ISM for each constellation would need to be regenerated as things change, e.g. as more in-service data are collected, allowing a better satellite failure rate prediction to be adopted. Readers may think that the ISM generator is just a (trusted) computer program that generates the ISM. In reality it may well be, but it also needs a resilient organisation wrapped around it to ensure continuity and integrity.

If it were just for failure rate updates, the ISM would not change very often, and could actually be done in updates to Annex 10, with updates to the avionics made as a ground maintenance activity. However, the ISM needs to be updated more often than that for some applications, which require, for example, frequent updates to satellites' ephemerides (EU-US, 2016), and so a means of in-flight update is required. Several means of transmission have been investigated; the current preference seems to be for the ISM to be sent to the user from the satellites themselves with the Navigation Message.

Some may argue that, for added confidence, the provision of the ISM should be entirely separate from the constellation provider, but it is a valid trade-off to reduce the complexity of using independent means of transmission. It has been proposed, however, that the ISM Generator organisation be kept independent of the constellation provider. In Europe, there is an existing incentive for this separation of concerns, regardless of whether you consider it a provider of GNSS

Signals or of Aeronautical Data[11], by law the ISM Generator has to be certificated as an ANSP (to ensure continuity and integrity). The incentive for separation is that the legal requirements for gaining and maintaining certification apply to the whole organisation, not just to the department providing the service in question, and they would be considered too onerous to apply to a complex satellite constellation operator organisation.

Work is still on-going in this area, but it can be assumed that the eventual ISM Generator(s) will set up formal arrangements with the satellite providers, and will develop a secure means of providing ISM updates with appropriate integrity. These updates may be in the form of Aeronautical Data, complying with the data quality and integrity requirements of Annex 15 (ICAO, 2018b). It is also assumed that the data required to generate the updates will be obtained from other organisations, e.g. monitoring stations, with appropriate service level agreements in place. To give more confidence to the user, these agreements and the internal operations will be audited by the pertinent state regulator, or other competent authority (in Europe, it will be the authority that issued the ANSP certificate).

7 Conclusion

Returning to the original question: would satellite navigation be 'safe enough' to direct a ground vehicle? We have established that it is appropriate for civil air transport, because they are required to use augmentation schemes to ensure the integrity of the navigation solution, and there are other techniques, such as inertial navigation, available if the solution is not good enough for use.

The ground vehicle problem is more difficult, and not only because motorways are much narrower than airways. Fewer satellites are likely to be in view at one time due to occlusion by buildings, etc. However, the same principles can be applied. There is the question of who establishes and maintains the rules, as there is no United Nations organisation to co-ordinate autonomous ground travel (yet). A reasonable argument could be made for the use in ground vehicles of augmentation systems that are intended for aircraft, but the usage and regulatory environments are different, so it is not as easy as it may look. Formal arrangements are likely to be needed, for example a Service Level Agreement with an SBAS supplier; but who shall agree, the manufacturer of the vehicle, its owner, its operator, its insurer, or even a governmental highways agency?

The new features discussed will also provide advantages to the ground-based user of satellite navigation services. In particular, taking advantage of more than one constellation will, to a degree, alleviate the 'urban canyon' problem, because

[11] The current view is that the ISM is Aeronautical Data, and so the ISM Generator would be certificated as a Data Services Provider

more usable satellites will be in view at one time. Using the new frequencies should improve accuracy, such that we would no longer be told to "Return to the motorway".

We can generalise (and for any service, not just satellite navigation):

- Establish the performance requirements, and how they can be achieved; can they all be satisfied consistently in practice?
- Can sufficient Continuity of Service be guaranteed for the application, or are fall-back provisions needed?
- If there is a fall-back (and the vast majority of vehicle applications should have at least one), what is the recovery time objective to get back to the original service; can it be achieved in practice?
- Most importantly, can the integrity of the service be established; will it alert in a timely manner when it is providing false data that must not be used?

If all these questions have been answered with a "yes", and compelling assurance arguments have been developed (Spriggs, 2019), then it will indeed be 'safe enough' – probably…

Disclaimers All sources used in the preparation of this paper are in the public domain, but note that some of the topics covered were proposals at the time of writing, and may subsequently not be taken up by ICAO and/or the member states. It should also be noted that this is just an overview, intending to highlight some common misconceptions, whilst glossing over much of the complication. The reader is urged to check the current official documentation if planning to deploy new systems based on these concepts.

References

World Geodetic System. (n.d.). In Wikipedia. Retrieved 26 September 2019, from https://en.wikipedia.org/wiki/World_Geodetic_System#WGS84

EU-U.S. Cooperation on Satellite Navigation. (2016). Working Group C - ARAIM Technical Subgroup, Milestone 3 Report, February 25th, 2016. Washington/Brussels: ARAIM TSG

ICAO. (2018a). Annex 10 to the Convention on International Civil Aviation - Aeronautical Telecommunications - Volume 1 - Radio Navigational Aids, 7th Edition, July 2018. Montréal: International Civil Aviation Organization

ICAO. (2018b). Annex 15 to the Convention on International Civil Aviation - Aeronautical Information Services, 16th Edition, July 2018. Montréal: International Civil Aviation Organization

RTCA. (2016). Minimum Operational Performance Standards for Global Positioning System/Satellite-Based Augmentation System Airborne Equipment, RTCA/DO-229E, December 2016. Washington DC: RTCA, Inc.

Spriggs, J. (2003). Developing a Safety Case for Autonomous Vehicle Operation on an Airport, in Redmill F, and Anderson T: Current Issues in Safety-Critical Systems, Proceedings of the Eleventh Safety-critical Systems Symposium, Springer-Verlag. ISBN 1-85233-696-x

Spriggs, J. (2019). Sufficient Assurance?, in Parsons M, and Kelly T: Engineering Safe Autonomy, Proceedings of the Twenty-seventh Safety-Critical Systems Symposium, Independently published. ISBN-978-1729361764

The Convention on International Civil Aviation done at Chicago. December 7, 1944. ICAO Doc.7300 https://www.icao.int/publications/Documents/7300_orig.pdf (Accessed 26 September 2019)

A Service Perspective on Accidents

Kevin King[1], Mike Parsons[2] and Mark Sujan[3]

[1] BAE Systems

[2] CGI UK

[3] Human Reliability Associates

Abstract *Major accidents that have impacted society, whether in aviation, healthcare, oil and gas, maritime, nuclear, defence or rail have all had a services element that played a part in the accident. This work utilises formal accident reports to identify and analyse these service aspects that contributed to recent accidents. Service elements include the people, training and procedures. These can both cause an accident or help recovery from it. Reference is made to the emerging Service Assurance Guidance produced by the SCSC Service Assurance Working Group (SAWG). The paper shows that service failures can cause accidents; often with fatal consequences.*

1 Introduction

This paper describes some recent accidents in the maritime, aviation, healthcare and rail sectors and identifies specific service aspects that are relevant to the accident and its aftermath[1]. It then ties these service elements to guidance being developed by the SCSC Service Assurance Working Group (SAWG).

The term "Service" is much overloaded; its definition is much discussed. This paper does not aim to provide a precise and constraining definition, instead it refers out to the Service Assurance Guidance (SAWG, 2020) which presents sev-

[1] The analysis presented in this paper has no legal standing whatsoever. The purpose of this paper is not to discredit, contradict or challenge any existing accident analysis; the aim is simply to view these incidents through the lens of service assurance. The analysis is the author's interpretation; they are not speaking on behalf of their employers.

© Kevin King, Mike Parsons and Mark Sujan 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

eral standard definitions, but more importantly, identifies characteristics of a service that may make the *Service-Based* approach to safety assurance appropriate. For more details on what constitutes a safety-related service, the reader is directed to the Service Assurance Guidance (SAWG, 2020).

A civil aviation example (AAIB, 2016) explains this *Services* perspective:

On 30 January 2016 at 1712 hrs, after take-off from London Heathrow Airport, the flight crew of a Boeing 747-436 G-CIVX passenger aircraft (figure 1) retracted the landing gear but were unable to move the landing gear lever in the cockpit from the UP to the OFF position. Concerned the landing gear may not be safely secured for their planned flight, the crew chose to return to Heathrow, where a safe landing was enacted with the nose and body landing gear lowered using the backup extension system.

Subsequent investigations identified that this was the first flight since the aircraft's Landing Gear Control Module (LGCM) had been replaced during a period of maintenance. The lever jamming was attributed to the omission of a rig pin during the installation of the replacement LGCM.

Fig. 1. Boeing 747-436 G-CIVX

Four significant service-related events were identified as the main causal factors:

1. Inadequate handover between night and day shifts;
2. Deficiencies in the task card system used by the maintenance organisation;
3. An engineer noticing the missing rig pin but being "seduced" by an overdue rest period and not warning his colleagues and:
4. An omission of the need to re-insert the rig pin in the Operator's Temporary Revisions to the Aircraft Maintenance Manual.

Combined these could have, were it not for the prompt actions of the flight crew, led to a far worse outcome for the 293 passengers and 17 crew on-board.

With this example it is worth identifying the specific services that could be considered as key factors in the chain of events that led to the near miss (note that these may or may not be explicitly identified as such in the actual situation):

1. "*Staffing*" service which supported the staff handover with suitably fresh and qualified staff;

2. "*Tasking*" service utilising cards and other procedures;
3. "*Staff Monitoring*" service, presumably run by managers/supervisors, and:
4. "*Maintenance Documentation*" service, either from within the maintenance organisation or further back down the supply chain.

These can all be usefully viewed as services as they are activities which do not produce any tangible "product" but do involve people, processes, etc. in maintaining a conforming "product".

Safety-related and safety-critical services are becoming the dominant way of delivering safety functionality to users, covering diverse services such as emergency medical response, air-traffic control and building maintenance. The key aspect is that there may be **no specific delivery of hardware or software involved** (however safety-critical and safety-related systems may be utilised by the overall service). It is essential that these services maintain conformance with the customer's requirement for safety to be assured.

These 'safety-related services' range from the initial provision of design expertise right through to disposal at the end of life, and include procurement and manufacture, in-operation maintenance and repair, such as in the Boeing 747 example, and all activities in-between. If the services work as intended, they can provide mitigation for potential threats and consequences of hazards, especially if they include highly trained and professional staff such as aircraft pilots or clinical staff who can recognize and adapt to evolving serious situations.

However, failure of such services can "pull the trigger" and cause an accident, potentially with fatal consequences and a significant time after the service was actually provided (e.g. a specialist radiographer missing indications of cancerous growth on a medical scan, leading to a diagnosis of cancer some weeks or months after the scan was analysed).

It should be noted that in many cases services can be highly robust and resilient, and accidents avoided or reduced in severity by suitable service design or staffing.

1.1 The Service Assurance Working Group

Aware of the importance of safety-related services and consequences of their failures, the Safety-Critical Systems Club saw a need for industry wide direction on the subject and formed the Service Assurance Working Group (SAWG, 2017) in 2017. This group has the aim of developing appropriate guidance that could aid providers in assuring their services in a safety context.

Armed with a greater understanding of the issues that services present and the characteristics that make up a safety-related service the SAWG has progressed to

the development of a pan-industry set of cogent principles for Service Assurance, set out in the Service Assurance Guidance (SAWG, 2020). That guidance is aimed at supporting safety-related service providers; in both reducing the contribution to safety risk of service failures, and in the development of failure mitigation approaches.

This paper supports that guidance by reviewing accidents that have occurred where the failure of one or more safety-related services can be considered to be a significant causal factor.

1.2 Sectors Considered

For this review, industry sectors have been chosen that have detailed and formal accident reports in the public domain, specifically: Marine, aviation, healthcare and rail.

The main sections of this paper consider these sectors in turn, providing analysis of accidents in the sector that have occurred since January 2017 where service failings could be considered significant factors.

Central to the review is a consideration of the benefits the developed Service Assurance Guidance may have provided in helping the service providers or system owners mitigate some or all of the outcomes for the accidents assessed.

The authors see service malfunctions as being significant across industrial accidents through history. The sample of accidents analysed herein was chosen to illustrate the continuing issue of service failures in contemporary society.

1.3 The Service Assurance Principles

The Six Service Assurance Principles devised by the SAWG are listed in table 1, together with further information. See the Service Assurance Guidance document SCSC-156 (SAWG, 2020) for more details.

Table 1. Service Assurance Principles

1	**Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours**
	There must be an overall definition of what the service is trying to achieve (formulated as requirements) and this must be within an expected usage scenario (e.g. concept of operations). There must be requirements addressing known behaviours that are unwanted or unsafe.

2	**The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces**
	This relates to the way the service hierarchy and service decomposition is constructed. It is saying that the intent of the assurance requirements must be shown to be met by the service elements comprising the service, and that the overall service architecture or hierarchy supports this flow down (i.e. that all service elements together meet the overall intent, and nothing is missing). Service elements can be of various types, including other services, systems, subcontracts, and agreements.
3	**Service assurance requirements shall be satisfied**
	Service requirements must be satisfied, i.e. verified as-is or decomposed into further requirements which are subsequently verified in some way. The methods by which service requirements are verified are wider than traditional systems, often including extensive use of proven-in-use (service history) and commodity-usage arguments, and also some specific contractual mechanisms. This principle (together with (4) below) creates the need for assurance "wrappers". (A wrapper is an assurance augmentation which addresses the assurance deficit inherent in the consumed service in some way.)
4	**Unintended behaviours of the SBS shall be identified, assessed and managed**
	All undesired or unintended behaviours which may impact safety properties or safe behaviour of the overall system must be identified and assessed within the usage context. They must be appropriately managed (e.g. mitigated, avoided or accepted in some way). This is not always possible to the extent desired, especially when commercial "commoditised" services are involved. Hence this may create the need for additional wrappers to make up the assurance gaps (see also principle (3) above).
5	**The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS**
	This is the proportionality principle: the level of (safety) risk must be used to determine the amount of effort (resources, time, etc.) put into assurance and mitigation activities. This principle can be used to underpin a set of levels of service assurance, where applicable activities are defined in bands derived from the risk level.
6	**These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing**

	Services may have a long lifetime and the service offering may evolve significantly over this time. These principles must be established and maintained throughout life: through e.g. usage change, technical change, subcontractor change, supplier or process and personnel change. This principle must also hold in service failure scenarios (contingency situations) where the service might temporarily employ manual or procedural activities to achieve its aims. It might be thought that this principle is implied by the others, but continuous evolution and change is a key property of services; in this they are different to (largely) static systems.

2 Marine

Marine accident reports covering the United Kingdom (UK) are publicly available from the UK's Marine Accident Investigation Branch (MAIB) website (MAIB, 2019). Accidents that have occurred since January 2017 were analysed to consider service failings and what positive benefits the Service Assurance Guidance could have bestowed to mitigate the incident. We consider here one such accidents in more detail.

2.1 Accident 1 – Catastrophic engine failure, resulting in a fire and serious injuries to the engineer on board Wight Sky, off Yarmouth, 12 Sep 2017

2.1.1 Accident Summary

According to MAIB Report 14/2018, at 21:33 on 12 September 2017, while approaching Yarmouth, Isle of Wight, the "ro-ro" passenger ferry Wight Sky (figure 2) suffered a major fire as a consequence of a catastrophic failure in one of its main propulsion engines. Although the fire was promptly contained, the vessel's engineer, who had been close to the event, was briefly engulfed in a ball of fire resulting in serious burns to his face and hands requiring 7 day's hospitalisation. He was also subsequently diagnosed with post-traumatic stress disorder.

Fig. 2. Wight Sky

Following investigation by the engine OEM, Volvo Penta UK, debris in the engine's oil channel following a recent rebuild was identified as the most probable trigger for the failure and subsequent fire. The MAIB determined:

1. "The engine had been completely rebuilt and failed after only 5½ hours of operation;
2. The vessel's soft patches[2] had not been removed, necessitating the engine to be lowered piecemeal into the engine room.
3. Debris could have entered the engine's oil channels in the rebuild or during the 3 days that the partially assembled engine had been exposed to the elements.
4. Analysis of oil samples from the engine indicated that accelerated wear had commenced before the engine failure;
5. The power supply to the essential services switchboard, which distributed power to critical equipment including the fixed fire-extinguishing system, was lost 27 minutes after the accident." (MAIB, 2018)

2.1.2 Analysis of Service Failures

The maintenance or the delay in re-installing the engine was the likely cause. "*Maintenance*" and "*re-installation*" are clearly service activities, and the sug-

[2] Soft patches: steel plates bolted down and sealed flush with the vehicle deck, that can be removed to allow large pieces of ship's machinery or equipment to be removed/inserted

gestion is that these were carried out in such a way to leave the engine oil contaminated. Hence the "*engine maintenanc*e" service can be considered to have operated deficiently.

The engine oil samples indicating accelerated wear were not acted upon – this can be considered a failure of the "*oil monitoring*" service.

2.1.3 Application of the Service Assurance Principles

Table 2 highlights where the service assurance principles could have mitigated the ensuing incident with the Wight Sky.

Table 2. Service Assurance Principles applied to the Wight Sky Incident

1	Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours
	The overall Service Based Solution in this case can be considered to be "Engine Maintenance". There were likely to have been detailed engine manufacturer replacement/installation instructions (effectively forming requirements) applying to this service. These likely included mechanisms to ensure the engine was kept in a clean environment during maintenance, again this was likely not followed. There were two earlier incidents with the engine on the Wight Sky, but these were probably unrelated to this failure. Three earlier incidents involving the same engine type were identified in the report. It is not known if the crew of the vessel were aware of these incidents.
2	The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces
	There were competence requirements on the vessel crew, and experience needs were met: *"The master and engineer held STCW4 certificates of competency appropriate to their ranks. The master had 7 years' experience and the engineer 26 years' experience in their respective roles on board Wightlink ferries"* Leaving the engine partially assembled would have left the possibility of ingress of foreign debris, hence the requirements flowed to each part of the maintenance process (involving method of reinstallation and cleanliness) were likely not complied with *"However, ME2's short block had been exposed to the elements for 3 days with only a loose plastic sheet for protection, and debris could have entered the oil channels during this time"* The way the maintained engine was lowered into position was likely against manufacturer recommendations *"The soft patches were not used to move the engines in or out of the engine rooms due to the disruption their removal would cause. Therefore, RKM planned its work around the use of the emergency hatch. This required the engines to be partially disassembled and rebuilt in the engine room, transporting the majority of the components in the short block."*
3	Service assurance requirements shall be satisfied

	There was a log book for the engine in use, but it is not mentioned how this was used except for *"…the engineer went down to the forward engine room…to note the machinery running parameters for the logbook"*. There was no recorded evidence that the correct process for replacing the engine was followed, hence the service assurance requirements were not explicitly satisfied. It would be expected that there would be (at least) formal test records and sign-offs.
4	Unintended behaviours of the SBS shall be identified, assessed and managed
	The possibility of dirt or particle ingress into the engine should have been considered. This would have led to this issue being monitored as part of the continuing maintenance service. The report mentions that *"As the engine was not fitted with a particle detector or other means of detecting rapidly progressing wear, there was no possibility of receiving an early warning before the engine failed".* The possibility that the power supply to the essential services could be lost should have been considered as credible failure scenario *"It shut down just after 2200 after the electrical power supply automatically switched over to the aft switchboard. As a result of an earlier oversight, the ES circuit breaker for the aft switchboard had been left in the manual mode, so the ES switchboard was left without power. This caused the loss of all essential services…"*
5	The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS
	There was some testing of the engine: *"The crew of the morning shift had tested … off load and had verified that all alarms and shutdowns were functioning correctly. During the afternoon shift, the vessel's engineer tested the engine on load, and on departure from Lymington that evening all four main engines were sharing the sea load"*, but clearly these tests were not sufficient to reveal the problem, i.e. they were not of the necessary duration type to reveal the contamination problem.
6	These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing
	It is hoped that the ferry operator will follow the recommendations in the report, learn the lessons and improve the overall engine maintenance service accordingly. The engine manufacturer, Volvo Penta has written to all dealerships providing guidance on good practice (effectively changing the specification of the maintenance service): *"Where appropriate, soft patches are removed to allow removal and reinstallation of complete engines.* *○ Engine assembly is completed in a clean environment to prevent debris being built into an engine.* *○ Following rebuilds, engines are load-tested on a dynamometer and certificates issued confirming the required performance.* *○ Records of component measurements are kept to confirm that they are within tolerance and fit for reuse"*

	There was an explicit recommendation on the engine manufacturer: *"Consider offering wear particle detection technology for Volvo Penta marine engines that cannot be easily serviced on board"*

2.1.4 Discussion

This incident can be viewed as involving several services from different service providers. The overall service is considered to be the "Engine Maintenance Service", which can be considered to consume services from:

1. The engine maintenance operator (RKM), "Engine Rebuild and Reinstallation",
2. The engine manufacturer (Volvo Penta), "Engine Maintenance Instructions and Guidance", and
3. Those services provided by the vessel staff, including "Staffing", "Engine Testing" and "Vessel Operation".

It can be seen that all of the sub-services failed in some way, so contributing to the incident. The lack of explicit documentation makes it hard to establish what the individual service failures were, but given the recommendations we can conclude that changes are required to all.

Note that the service assurance guidance goes on to suggest the use of service "wrappers" or assurance supplements; in this case the wrappers could include:

(i) the additional "soak" testing of the engine;
(ii) better manufacturer guidance;
(iii) more rigorous oversight of the engine re-installation process and
(iv) the production of additional documentation to enable more detailed fault investigation.

3 Aviation

Aviation accident reports covering the United Kingdom (UK) are publicly available from the UK's Air Accident Investigation Branch (AAIB) website (AAIB, 2019). When an aviation accident occurs in UK airspace it is the responsibility of the AAIB to investigate and report findings. The analysis they provide is primarily aimed not at apportioning blame but determining the causes of the accident and making recommendations directed at relevant stakeholders, from aircraft manufacturers and operators to maintenance organisations and even airports. We consider here one such accident in more detail.

3.1 Boeing 737-4Q8 (G-JMCR), loss of electrical power en route to East Midlands Airport, 12 Oct 2018

3.1.1 Accident Summary

Fig. 3. Boeing 737-4Q8 G-JMCR

According to AAIB Report EW/C2018/10/03 (AAIB, 2019), at 01:55 hrs on 12 October 2018 West Atlantic were operating Boeing 737-4Q8, G-JMCR (figure 3), on a night cargo flight en-route to Aberdeen from Leipzig via Amsterdam and East Midlands Airports. On commencing its descent into East Midlands, the flight-crew were surprised by an abnormal assortment of sporadic electrical failures on the co-pilots display screens and indication panels. Fortunately, this occurred when both the pilot and co-pilot had visual sight of the runway enabling a manual landing to be completed without further incident or injury.

3.1.2 Analysis of Service Failures

All West Atlantic flight crews are trained to provide an '*in-flight incident management*' service to analyse abnormal and emergency situations in line with Boeing's Quick Reference Handbook (QRH) and their employer's decision-making strategy. At no time leading up to the incident did the crew of G-JMCR seek to enact such a service. The AAIB determined that the flight-crew had sufficient time without impacting negatively on a safe landing.

For G-JMCR an Acceptable Deferred Defect (ADD) was in place for a faulty generator (Gen 1). This was permissible under European Union Aviation Safety Agency (EASA) Minimum Equipment List (MEL) rules provided a fully functional second generator (Gen 2) was available and an Auxiliary Power Unit

(APU) was operated during flight. EASA rules also allowed for the operator to approve a Rectification Interval Extension (RIE)3. Incorrectly, the operator saw the MEL and RIE as means of supporting continued operational commitments rather than prioritising defect resolution. Consequently, partial fault finding, and defect resolution occurred with aircraft wrongly pressed into operation with unresolved defects. Alas, the underlying fault in Gen 1 remained extant as the aircraft operator continued to overlook opportunities to fully enact a 'defect management' service.

Pertinent to this incident there were also instances of failings in the service of 'record keeping'. GCUs were swapped out during defect rectification work on Gen 1 in support of addressing the ADD in the days prior to the incident. These were recorded in the Operator's Flight Status Reporting system (FSR) but not the records specific to G-JMCR.

3.1.3 Application of the Service Assurance Principles

Considering the identified service failings, Table highlights where the various service provider organisations (both internal and external to West Atlantic) could have benefited from application of the service assurance principles to direct focus onto the assurance of their service provision. This could in turn have mitigated the Boeing 737-400 (G-JMCR) near miss incident.

Table 3. Service Assurance Principles applied to the Boeing 737-4Q8 (G-JMCR) Incident

1	Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours
	West Atlantic had an SBS in place. However, there is no evidence that they recognised it as such, with no formal record of service assurance requirements. Examining their activities from the perspective of the safety-related services they are accountable for would have provided valuable insight into their SBS and where it was deficient from an assurance perspective. Improving their SBS to provide clarity of the appropriate behaviours across all stakeholders within and beyond their organisation would have alleviated the issues of weak or non-existent communication that contravened their Part 145 approvalFollowing Principle 1 in developing a related set of service assurance requirements we believe would have re-focussed their priorities away from opera-

[3] The operator's procedures allow a one-time RIE where a defect cannot be cleared within MEL time limits. RIEs should only be used in 'exceptional circumstances' and must only be approved when '...it was not reasonably practical for the repairs to be made.'

	tional commitments and into a view where conformance of their services and regulatory compliance were seen as primary to ensure their continued safe operation.
2	The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces
	Sadly, without clarity of service requirements and the characteristics (both desirable and undesirable) of their service portfolio, West Atlantic lacked perception of their service accountability leading to poorly designed '*tasking*', '*defect management*' and '*record management*' services which we have shown were weak or failed in some way driven by a culture where operational commitments were prioritised over defect resolution. Refocus onto service assurance through a structured service architecture across the service provider organisations, including through West Atlantic's supply chain, would almost certainly have flipped that priority. Importantly, although aircraft downtime may have increased, we believe this service assurance centred approach would have impacted positively on West Atlantic's relationship with their customers, a 'safety first' message. In this incident the diversity, geographic spread and potential language issues across all stakeholders (e.g. Part M, LMC) would also have benefited had West Atlantic maintained a logical approach to service assurance with clear service handshaking across interfaces within their organisation and through their supply chain, what we call assurance 'Wrappers' (Durston et al, 2019).
3	Service assurance requirements shall be satisfied
	With a workable SBS and 'Wrappers' in place through its service hierarchy, we would like to think G-JMCR would not have been allowed to take off from Amsterdam as the ground engineer would not have been able to completely satisfy the requirements for his '*defect resolution*' service and cleanly hand-back to the LMC. Equally the LMC would not have been able to conclude their '*defect management*' and '*tasking*' services and allow the flight-crew to safely continue their journey.
4	Unintended behaviours of the SBS shall be identified, assessed and managed
	Incidents like this occur in an evolving service landscape populated by varying players at any one time. Overlaying the drive to meet operational commitments means unintended behaviours are more likely to occur. This makes an SBS architecture, service assurance requirements and 'wrappers' across all assurance boundaries invaluable assets.

	Key facets of service provision include evidence of regulatory compliance and clearly defined service procedures, both of which were lacking to some degree in this incident leading to unintended behaviours. For example, the '*defect management*' service was deficient with records of GCU swap outs not documented due to time pressures to return the aircraft to service. Another factor where this principle could have helped is related to the commander of G-JMCR who, new to West Atlantic, held working practices from his previous employer that may have been incongruent to the West Atlantic handbook. These should have been addressed during the design/enactment of the '*staff training*' service for the commander role, highlighting that contributory service failings can occur within in-direct support organisations and potentially sometime in the past.
5	The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS
	The evidence available in the AAIB report leads us to believe that none of the service provider organisations were fully cognisant of the contribution to safety risk posed by the services they were accountable for. Certainly not an understanding commensurate with their contribution to the level of risk associated with their service provision.
6	These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing
	Air freight is a fast-paced environment with prompt turnaround of aircraft being the norm. Consequently, this leads to pressures to deliver safe '*defect management*' and '*maintenance*' services in tight timescales. Without maintaining a structured focus on the assurance of the services being provided through life West Atlantic could easily find themselves delivering unsafe services in the future.

3.1.4 Discussion

The fast paced, low cost world of civil aviation in the 21st Century is putting pressures on aviation organisations to keep pace or perish. Even though standards and regulation are in place to promote a culture of safety management (e.g. ICAO Annex 19 to the Chicago Convention, 2019), incidents such as G-JMCR are still occurring with service failure a significant causal factor. Research over the past 20 years has also focused on safety in aviation maintenance organisations (e.g. McDonald et al., 2000 and Patankar and Taylor, 2016), but that too has not considered assurance of the service provision.

A service assurance thread needs to be woven into the approach aviation organisations take to developing their safety cases. We believe the Service Assurance Guidance, if publicised appropriately will support that aim. However, the guidance must not be flavoured too heavily towards a particular sector of industry to comply with that sectors regulation and standards, to the detriment of others.

4 Healthcare

Healthcare accident reports covering the United Kingdom (UK) are publicly available from the UK's Healthcare Safety Investigation Branch (HSIB) website (HSIB, 2019). We consider here one such accident in more detail.

4.1 Investigation into the transition from child and adolescent mental health services to adult mental health services,18 Oct 2017

4.1.1 Accident Summary

According to HSIB Report I2017/008, 18-year old Ben (not the person's real name) committed suicide during the period of transition from child and adolescent mental health services (CAMHS) to adult mental health services (AMHS). Ben had been diagnosed with Autism Spectrum Disorder (ASD) during childhood, and already had a documented history of attempted suicide. Owing to Ben's low moods, anxiety and suicidal tendencies, Ben was referred by his GP to CAMHS. Ben was put on medication and a care plan was established. Subsequently, Ben was seen by different professionals.

After about 3 months Ben's care coordinator went on sick leave, and Ben was allocated a new care coordinator. As Ben was approaching his 18th birthday, a transition request to AMHS was put in. The referrer noted in the transition request that Ben had expressed the intention of ending his life once he turned 18. Ben expressed great anxiety about the transition to AMHS, which was explained in part by his dislike of change associated with his ASD.

Over the next few months Ben's low moods and negative thoughts increased, and Ben's medication was increased further. Ben's mother informed his care coordinator that he had self-harmed. Ben met with his care coordinator, and he expressed again his anxiety about transitioning to AMSH, and his desire to continue to remain with CAMHS. Ben was told that he needed to transition to AMHS at the age of 18, but that a handover would be put in place.

The same night, Ben died by suicide.

4.1.2 Analysis of Service Failures

When a child or young person dies by suicide it is, by default, a failure of the health service that was supposed to look after and care for that person. This is especially true in Ben's case. Ben had a history of suicidal episodes and low moods, and had been in frequent contact with mental health services. There were many warning signs, and Ben had even announced his intention to end his life on his 18th birthday. And yet, it is hard – and misleading – to point, in hindsight, the finger at any one individual and assign blame or identify their actions as the cause of this tragic event.

However, adopting a service perspective might provide further insights that can help explain this death, and from which we might be able to identify lessons for improvement. The service element, which crucially failed in this case, is the '*transition*' service provided by CAMHS and AMHS in collaboration. This transition period is recognised as being critical, and it is known from the literature that young adults often disengage from the health service (not just mental health) during transition, which leads to suboptimal health outcomes, or death as in this case (Griffiths et al, 2017). The HSIB reports emphasises that Ben's case is not an isolated example, but that similar issues linked to failures in transition have occurred throughout England.

Even though CAMHS and AMHS are providing this transition service together, they are each very different services, and the transition is complex. This is further exacerbated by variability in practice, with some CAMHS providing care flexibly up to the age of 25, while others transition more rigorously to AMHS at the age of 18.

The transition request was initiated by CAMHS quite close to Ben's 18th birthday, and this was, in part, caused by the 3-months absence of Ben's initial care coordinator due to sickness. As a result, plans for handover and shared care arrangements were not in place, and this caused Ben significant additional anxiety. Ben's ASD diagnosis and his struggle to deal with change were known and documented, and the HSIB report suggests that a longer and better planned transition period would have supported Ben.

Shared care arrangements between CAMHS and AMHS are facilitated by joint meetings, but frequently these do not take place due to high workload, difficulties in managing and aligning diaries, and the young person's and their families' availability. This was the case with Ben, where no joint meeting was held in the run up to Ben's transition to AMHS.

The HSIB reports makes a range of reasonable recommendations, including training for staff in safe transitioning, mental health service configuration and ring-fencing budgets.

4.1.3 Application of the Service Assurance Principles

Considering the identified service failings, table 4 highlights how the service assurance principles could provide a useful framework and structure to support the organisations in reasoning about services and how they can be assured.

Table 4. Service Assurance Principles applied to the transition from CAMHS to AMHS

1	Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours
	One of the main problems with the transition service is that it was not properly recognised as a service. As a result, there was no specification of how the transition service would be delivered, even though each organisation (CAMHS and AMHS, but also GP surgery) had its own procedures in place. Principle 1 in essence stipulates that organisations reason explicitly about services, define them, consider both desired and undesired behaviours, and have mechanisms in place to learn from past experience. Arguably, none of these objectives were met for the transition service, even though it was recognised as a crucial and potentially high-risk service. It might be helpful to define explicitly how the transition service is set up and intended to work using the Service Assurance Objectives as scaffolding, and potentially documenting the arguments and evidence in a safety case (Sujan et al, 2015).
2	The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces
	No service assurance requirements had been defined, and hence subsequent service assurance principles were not met. Even if overall service assurance requirements had been defined, current practice within the health sector would make it unlikely that these would be decomposed and allocated consistently to the different service elements and actors. Practice is very variable, and organisations have their own processes and procedures, which do not necessarily align with those of other organisations. Principle 2 supports organisations in defining with greater clarity how roles and responsibility for meeting safety requirements are distributed among different actors.
3	Service assurance requirements shall be satisfied
	Some of the organization's procedures include targets and assurance requirements, such as initiating the transition several months prior to the actual transition, and having joint meetings involving all stakeholders. However, such requirements are routinely violated in practice, and it is unclear to what extent this is actively monitored, and whether any learning is drawn from this unless a serious adverse outcome necessitates investigation. Principle 3 provides

	guidance to organisations about processes and evidence for assuring that requirements are met. This then links back to Service Assurance Principles 1 and 2 to provide greater transparency and a logical flow of how specific evidence feeds into the overall argument for service assurance.
4	Unintended behaviours of the SBS shall be identified, assessed and managed
	Arguably, healthcare organisations are very poor at identifying unintended behaviour proactively, as there is a strongly reactive culture that considers safety too often in response to adverse events, i.e. after patients have been harmed (Sujan, 2015). Principle 4 suggests that organisations reason explicitly and in a systematic way about how services might fail and how unintended behaviours might have knock-on effects further downstream.
5	The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS
	Principle 5 supports organisations in reasoning about the strength of their service assurance evidence. This type of structured thinking about services, risk and evidence, and trustworthiness of evidence is not current practice in health services.
6	These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing
	Constant change is a characteristic of services that need to adapt to variations in demand and developments in healthcare brought about by medical and technological innovation. Health services frequently are not designed explicitly (Principle 1), but evolve, and there is no clear understanding or systematic approach for managing change safely Principle 6 suggests that organisations maintain an adequate record of how changes might affect the service.

4.1.4 Discussion

Structured reasoning about safety risks is still in its infancy in many parts of the health sector (Spurgeon et al, 2019). The Service Assurance Guidance could support healthcare providers in gaining a better understanding of how their services contribute to patient safety, and where the threats and vulnerabilities are. However, there is also a learning point for the Service Assurance Guidance if it is to be adopted in a sector like healthcare (Sujan et al, 2017). The guidelines need to consider the different organisational, institutional and cultural context in healthcare, and appreciate the specific norms, values and needs of healthcare stakeholders, such as clinical and professional autonomy, the nature of what is accepted as scientific evidence, and the particular ways in which organisations need to demonstrate accountability (Dixon-Woods et al, 2014).

5 Rail

Rail accident reports covering the United Kingdom (UK) are publicly available from the UK's Rail Accident Investigation Branch (RAIB) website (RAIB, 2019). The analysis they provide is primarily aimed at improving railway safety through identifying mitigations that could prevent future accidents across the UK rail net-work, they do not seek to determine liability or apportion blame but make recommendations around safety improvements directed at relevant stakeholders.

Accidents were analysed to consider service failings and what positive benefits the Service Assurance Guidance could have bestowed to mitigate the incident. Here we consider two such accidents in more detail.

5.1 Accident 2 – Members of the public struck by a flailing 240v AC cable at Abergavenny (Y Fenni) station, 28 July 2017

5.1.1 Accident Summary

Abergavenny (Y Fenni) station sits on the Newport to Hereford line. According to RAIB Report 06/2018, at about 18:05 on 28th July 2017, the roof of a northbound passenger train entering the station caught a 240V AC electrical cable hanging below the station's footbridge (figure 4) dragging the cable and causing it to become detached from both its fixings and an electrical distribution cabinet. The free end of the cable flailed in the air striking passengers climbing the footbridge stairs causing minor injuries to three of them. Continuing its trajectory, the cable also nearly struck a member of station staff on the platform. Collateral damage occurred to other cables and station infrastructure (figure 5).

The cable provided the main power source to the adjacent Abergavenny signal box and had become separated from the cable tray securing it over the footbridge.

Fig 4. Detached cable tray running across the footbridge (to left of image)

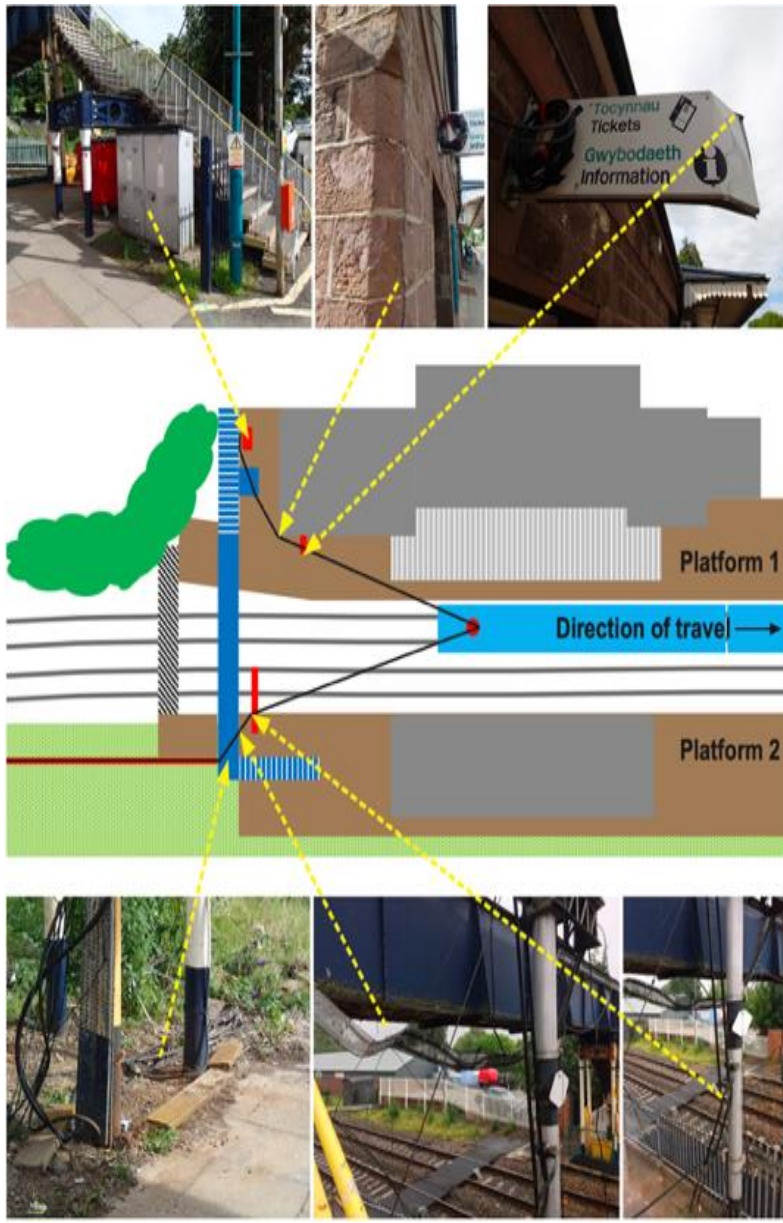


Fig. 5. Diagram showing the position of the supply cable as the train pulled it and images of the damage caused

It was drooping down and caught an antenna on the rear carriage of the train. The RAIB showed that nylon cable ties used to hold the cable in place had failed. The cable had not been inspected periodically as required and regular footbridge inspections had not highlighted the fact that the cable was hanging loose. Furthermore, the loose cable was not reported during routine station safety checks. The RAIB also identified the lack of controls in Network Rail protocols for the management of low voltage electrical supply cables where those cables cross over operational railway lines via overhead structures.

5.1.2 Analysis of Service Failures

Clearly the cable being dragged by the train and pulled from the electrical distribution cabinet was the immediate cause of the incident. But how the cable came to be hanging such that it snagged on the train can be tracked back directly to failings in safety-related services.

The primary service failure was in '*installation*' where black nylon cable ties were used that were unsuitable for exterior applications due to likely premature degradation from exposure to ultraviolet, excess moisture and variations of temperature. The situation was exacerbated by the tray to which the cable was fixed being hung vertically on the side of the footbridge, meaning the black nylon cable ties fixing the cable were also directly holding its weight. The RAIB determined the accident occurred at least 12 years after that installation service failure that set the trigger.

A further service failure was that of '*inspection*'. A clear requirement of the wiring regulations (BS 7671) is periodic inspection of all electrical installations, with periodicity influenced by among other things the environment. In this case, there was a split of responsibility with Network Rail accountable for inspecting the signalling and the signal box, and Arriva Trains Wales accountable for inspecting the stations electrical infrastructure. The last set of inspections in 2013 recorded that the cable in this incident (from the station distribution box to the signal box) was being inspected by the other organisation and as a consequence it was inspected by neither.

Sadly, other '*inspection*' service failures occurred: i) where the requirement for annual visual inspections was overlooked post-2016, notably because the last such inspection identified concerns with a sag in the affected electrical cable, and ii) Monthly station safety checks which relied on an ambiguous questionnaire.

5.1.3 Application of the Service Assurance Principles

Considering the identified service failings, table 5 highlights where the service assurance principles could have mitigated the cable drag incident at Abergavenny.

Table 5. Service Assurance Principles applied to the cable drag incident at Abergavenny

1	Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours
	It is well understood that rail networks are inherently high-risk environments, exemplified by Network Rail's Safety Vision (Network Rail, 2019). It would seem reasonable therefore to expect safety to be the first priority in all service activities that take place across the network. Sadly, although stakeholder organisations would like to hope that is the case, accidents such as that at Abergavenny are still occurring. It would seem greater focus is required on the assurance of safety-related service activities across the rail network to determine the behaviours of those services and what unique mitigations need to be considered. This is where Principle 1 of the Service Assurance Guidance can help. Particularly of value from a services perspective would be consideration of the impact from degraded modes of service and prior service failings (such as a previous cable drooping incident from the same bridge in 2002) in defining a robust service architecture.
2	The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces
	The RAIB identified a number of service failings in this incident related to a lack of ownership or accountability. Network Rail own the rail infrastructure including the signal box but are not accountable for the station infrastructure, owned/managed by Aviva Trains Wales. Presumptions were made as to which organisation was accountable for the service of inspecting the cable tray affixed to the bridge in which the drooping cable was meant to be secured. In the end neither organisation inspected it. Had an overarching SBS been in place with clear agreements across service interfaces we would like to think inspections of the tray would have occurred because accountability for it would have been clearly defined/recorded.
3	Service assurance requirements shall be satisfied

	Services come in many guises; equally identical services can be delivered by different providers. The UK rail network is a classic example of this kind of interwoven service/provider framework. Although requirements for services exist, such as the monthly requirement for a station safety inspection, the ambiguity over accountability for inspection of the cable tray led to that service not being wholly enacted. Had the requirements been more explicit from a service provision/accountability perspective in line with Principles 1 and 2 then satisfactory compliance with the service requirements would hopefully have mitigated the drooping cable.
4	Unintended behaviours of the SBS shall be identified, assessed and managed
	Failings in the '*inspection*' service can be seen as consequential mitigation failures. The main service failure in this incident was that of '*installation*' of the cable using inappropriate nylon cable ties. The RAIB cannot be certain of an exact date but the '*installation*' error certainly happened at least 4 years before the incident and potentially as far back as the early 1990s. Clearer understanding of their accountability for the '*installation*' service would likely have made the engineer involved and their parent company more diligent in ensuring more appropriate fixings were used.
5	The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS
	In such a high-risk industry one would like to think the service providers are fully cognisant of the safety risks they face and have confidence in their approaches to manage them. The reason a lot of rail accidents occur is that an equivalent level of risk understanding does not seem to exist around their service provision.
6	These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing
	As a service, '*inspection*' needs to be maintained at a level commensurate with the risk/impact of a failure of the item(s) being scrutinised. Periodic inspections, such as those mandated by BS7671, as detailed in the RAIB report (RAIB, 2018), need to be held. Although Arriva Trains Wales maintained the services they were accountable for, there is no certainty that a future franchise operator would adopt the same level of diligence. An overarching SBS considering the Service Assurance Principles would provide an enduring framework to support the assurance of extended services like '*inspection*'.

5.1.4 Discussion

It is important to recognise the complete suite of services that make up an SBS not only those that are currently being enacted but also those provisioned earlier

in the lifecycle of the affected safety-critical system, in this case a section of the UK rail network. Services can fail sometime after they were first delivered, as in the '*installation*' service at Abergavenny. Well-thought-out analysis of service provision, both spatially and temporally can be reinforced by use of the Service Assurance Guidance, which provides a framework to link the assurance of services across an industry sector identifying where weaknesses exist in the service assurance map, critical in industries where safety-related services predominate.

6 Further Work

Work has progressed to decompose the service assurance principles into a lower-level set of objectives. If the associated objectives are met, then the principle is deemed to have been achieved. The principles with their associated objectives are shown in table 6.

Further work would involve establishing whether the objectives were met in the particular accident scenarios. In some cases, the accident reports contain highly detailed information, (although usually not related to service aspects). This indicates an update to accident investigation methods is required.

Table 6. Service Assurance Principles and Objectives

1	Service assurance requirements shall be defined to address the Service-Based Solution's (SBS) contribution to both desirable and undesirable behaviours
	a. Context and intended use of the SBS SHALL be established
	b. States of the SBS SHALL be defined including normal, abnormal and degraded modes, as well as transitions between the states
	c. Key stakeholders of the SBS SHALL be identified
	d. Service assurance requirements for desirable behaviours, including service and performance levels, of the SBS SHALL be defined
	e. Service assurance requirements to mitigate undesirable behaviours of the SBS SHALL be defined
	f. A high-level service architecture SHALL be defined
	g. Historical accidents and incidents related to the service offering SHOULD be assessed and any relevant recommendations considered.

2	The intent of the service assurance requirements shall be maintained through the service definitions, service levels, the service architecture and the agreements made at service interfaces
	a. Service assurance requirements SHALL be decomposed and assigned to service elements within the service architecture of the SBS b. The service architecture including sub-services SHALL be defined c. Service assurance requirements SHALL be defined for each sub-service d. The agreements made at service interfaces SHALL be defined e. Service assurance requirements tracing through the service architecture SHALL be established f. Methods and techniques used to provide service assurance within each level of the service architecture SHALL be defined and implemented g. Assurance wrappers SHALL be identified and defined for service elements to make good any known assurance shortfalls
3	Service assurance requirements shall be satisfied
	a. Verification evidence SHALL be produced to show that service assurance requirements are met by the architecture and the elements of the SBS b. Assurance wrappers SHALL be implemented and verified c. Evidence SHOULD include proven in use and service history evidence
4	Unintended behaviours of the SBS shall be identified, assessed and managed
	a. Residual risks SHALL be identified and linked to service artefacts and service properties b. The residual risk of the SBS SHALL be reduced to an acceptable level c. Unintended behaviours resulting from the service architecture and service elements SHALL be identified, assessed and managed d. Unintended behaviours resulting from fault-free cases SHALL be identified, assessed and managed e. Service-service interactions SHALL be considered f. Service assurance artefacts SHALL be identified and produced

5	The confidence established in addressing these principles shall be commensurate with the level of risk posed by the SBS
	a. Service Assurance Levels (SALs) SHALL be established based on the level of risk that the service presents to the service users b. SALs SHALL be decomposed and assigned to service elements within the service architecture of the SBS c. Service assurance artefacts SHALL be produced according to the SAL d. Activities, methods, analyses and tools used to provide service assurance SHALL be appropriate for the SAL
6	These principles shall be established and maintained throughout the lifetime of the SBS, resilient to all changes and re-purposing
	a. All changes to the SBS that impact these objectives SHALL be assessed and managed b. Service assurance artefacts SHALL be maintained c. Use of the SBS SHALL be monitored for change and a safety impact analysis shall be undertaken d. Use of the SBS for a new purpose, or changed scope SHALL cause a re-evaluation of the compliance with the objectives e. Degraded and contingency modes of the SBS SHALL maintain defined set of these objectives f. Lessons learnt SHALL be incorporated in the SBS

7 Conclusion

Accident reports often list multiple causes or contributory factors; some of these causes will be related to the way the services (implicit or explicit) are constructed. Accident Reports do not currently consider service failings and the value of applying a services perspective, it is argued that they should. It can be seen from the service analyses of the accidents:

1. Consideration of the accident scenarios through “service eyes” can be a useful way to consider the situation. It helps to analyse the accident if the implicit and explicit services are identified together with the failures of those services, (even if there are no explicit service contracts in

place);
2. Different and varied ways in which safety-related services can fail leading to an accident (e.g. a "Lookout Service" in a marine context)
3. How some aspects of the failures could be mitigated by application of the Service Assurance Guidance.

It should be noted that as per "Safety II" many service failures never result in accidents because components of the service (often the staff) are resilient and prevent the chain of events from a specific failure turning into an accident.

Acknowledgments

Fig. 1: reproduced from jetphotos.com, https://www.jetphotos.com/photo/9485348 Photographer: Wael AL-Qutub.

Fig. 5: reproduced from flickr.com, https://flickr.com/photos/130961247@N06/43103501082 Photographer: Anna Zvereva

All other figures are reproduced from the respective referenced accident reports.

The efforts of the SAWG are the primary driver for this work. The group's contribution is much appreciated.

References

AAIB (2016), Air Accident Investigation Branch Bulletin, 10/2016, https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/567055/AAIB_Bullletin_10-2016.pdf, accessed 10 October 2019.

AAIB (2019) Air Accident Investigation Branch Bulletin 10/2019, Available at https://www.gov.uk/aaib-reports, accessed 2 September 2019.

Dixon-Woods, M., Martin, G., Tarrant, C., Bion, J., Goeschel, C., Pronovost, P., Brewster, L., Shaw, L., Sutton, L., Willars, J., Ketley, D. and Woodcock, T. (2014). Safer Clinical Systems: Evaluation Findings. London: Health Foundation.

Durston, N., Parsons, M., Scott, A. and Simpson, A. (2019). The Principles of Service Assurance in Kelly T and Parsons M, "Engineering Safe Autonomy", SCSC-150, 2019, https://scsc.uk/rp150.6:1, accessed 28 October 2019

Griffiths, F. E., Bryce, C., Cave, J. A., Dritsaki, M., Fraser, J., Hamilton, K., Huxley, C., Ignatowicz, A., Kim, S.-W., Kimani, P., Madan, J., Slowther, A.-M., Sujan, M. and Sturt, J. (2017). Timely digital patient-clinician communication in specialist NHS clinical services serving young people: findings from a mixed methods study (The LYNC study). Journal of Medical Internet Research.

HSIB (2019) Healthcare Safety Investigation Branch Report I2017/008, Available at https://www.hsib.org.uk/investigations-cases/, accessed 2 September 2019.

International Civil Aviation Organization (ICAO), Annex 19: Safety Management . Montreal, Quebec: ICAO, 2019.

MAIB (2019) Marine Accident Investigation Branch Report 6/2019, Available at https://www.gov.uk/maib-reports, accessed 2 October 2019.

McDonald, N., Corrigan, S., Daly, C., Cromie, S. (2000). Safety management systems and safety culture in aircraft maintenance organisations. Saf. Sci. 34 (1‑3), pp151‑176.

Network Rail (2019) Our safety vision. Available at: https://www.networkrail.co.uk/who-we-are/our-approach-to-safety/our-safety-vision/ , accessed 17 November 2019

Patankar, M. S., Taylor, J. C. (2016), Risk Management and Error Reduction in Aviation Maintenance. Routledge, London

RAIB (2019) Rail Accident Investigation Branch Report 19/2018, Available at https://www.gov.uk/raib-reports, accessed 2 September 2019.

SAWG (2017), Service Assurance Working Group web pages, Available at http://scsc.uk/groups.html?group=gs, accessed 2 October 2019

SAWG (2020), SAWG Service Assurance Guidance v1.0, Safety Critical Systems Club, SCSC-156, Available at http://scsc.uk/SCSC-156

Spurgeon P., Sujan, M. A., Cross, S. and Flanagan, H. (2019) Building Safer Healthcare Systems: A Proactive, Risk Based Approach to Improving Patient Safety. Springer, Cham

Sujan, M., Spurgeon, P., Cooke, M., Weale, A., Debenham, P. and Cross, S. (2015). The Development of Safety Cases for Healthcare Services: Practical Experiences, Opportunities and Challenges. Reliability Engineering & System Safety, 140, pp200-207.

Sujan, M. (2015). An organisation without a memory: a qualitative study of hospital staff perceptions on reporting and organisational learning for patient safety. Reliability engineering & system safety, 144, pp.45-52.

Sujan, M. A., Habli, I., Kelly, T. P., Gühnemann, A., Pozzi, S. and Johnson, C. W. (2017). How can health care organisations make and justify decisions about risk reduction? Lessons from a cross-industry review and a health care stakeholder consensus development process. Reliability Engineering & System Safety, 161, pp1-11.

Modular Safety Cases for the Assurance of Industry 4.0

Omar Jaradat[1]∗, Irfan Sljivo†, Richard Hawkins‡, Ibrahim Habli‡

∗National Electric Vehicle Sweden (NEVS) AB, Trollhattan, Sweden,

†Malardalen Real-Time Research Centre, Malardalen University, Vasteras, Sweden,

‡Department of Computer Science, University of York, York, UK

Abstract *The Internet-of-Things (IoT) has enabled Industry 4.0 as a new manufacturing paradigm. The envisioned future of Industry 4.0 and Smart Factories is to be highly configurable and composed mainly of the 'Things' that are expected to come with some, often partial, assurance guarantees. However, many factories are categorised as safety-critical, e.g. due to the use of heavy machinery or hazardous substances. As such, some of the guarantees provided by the 'Things', e.g. related to performance and availability, are deemed as necessary in order to ensure the safety of the manufacturing processes and the resulting products. In this paper, we explore key safety challenges posed by Industry 4.0 and identify the characteristics that its safety assurance should exhibit. We propose a modular safety assurance model by combination of the different actor responsibilities, e.g. system integrators, cloud service providers and "Things" suppliers. Besides the desirable modularity of such a safety assurance approach, our model provides a basis for cooperative, on-demand and continuous reasoning in order to address the reconfigurable nature of Industry 4.0 architectures and services. We illustrate our approach based on a smart factory use case*.

1 Introduction

The Internet-of-Things (IoT) can be seen as a system of inter-connected cyber-physical objects that collect and exchange data. More formally, IoT is defined as "a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving

[1] omar.jaradat@nevs.com, irfan.sljivo@mdh.se,
{richard.hawkins, ibrahim.habli}@york.ac.uk

© Jaradat et al 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

interoperable information and communication technologies" [25]. This infrastructure allows the Things to be sensed and controlled remotely so that their integration into the physical world leads to different ways to utilise the Things in various reconfigurable applications. Cloud Computing is a fundamental infrastructural element for IoT, enabling different types of X as a Service (XaaS)[1] [19], where X is a software, platform, infrastructure, etc. In this paper, we adopt the NIST definition of Cloud Computing:

> "*a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*" [17].

The marriage of the IoT and Cloud services (e.g., cloud XaaS) has paved the way towards the fourth industrial generation[2], Industry 4.0, as a new trend of automation and data exchange in the manufacturing industry. This new industrial paradigm is characterised by its ability to reconfigure and often optimise autonomously, particularly during the operational stages. Moving certain manufacturing services, e.g. scheduling and data storage and analytics, to the Cloud has potential benefits in cost reduction, energy efficiency, sharing of resources and increased flexibility. The use of Cloud Computing in critical applications has been highlighted as a significant area of research, especially for production and manufacturing systems [3], [7], [12], [26].

However, factories are often categorised as safety-critical systems as failures of these systems, under certain conditions, can lead to human harm or damage to property or the environment, e.g. due to the use of heavy machinery or hazardous substances. As such, the risk associated with the manufacturing processes and the resulting products has to be analysed, controlled and monitored. However, the reconfigurable, modular and dynamic nature of Smart Factories pose significant safety assurance challenges. For example, designers or operators of factories do not have much control over the design and evolution of the 'Things' or Cloud-based services that are increasingly being used in manufacturing processes. This potentially weakens confidence in the safety of the factory and can undermine the overall safety case [21], i.e. due to high degrees of uncertainty about the actual performance or behaviour of these 'Things' or Cloud-based services.

Most of the reviewed published literature on IoT and Cloud Computing reveals focus on security in particular and dependability in general but without much focus on safety. For example, the German automation technology supplier 'PILZ' [18] stated that the Industry 4.0 vision entails modular plants being reconfigured quickly and flexibly. They view the control and decision-making process

[1] Key IoT terms are described in the last section.

[2] aka Industrie 4.0

in Industry 4.0 becoming more decentralised and highlight safety, in particular, as a fundamental challenge, with emphasis on the necessary modular certification of the individual factory devices (PILZ uses the term Safety 4.0 to indicate modular safety solutions).

In this paper, we introduce a common Industry 4.0 architectural style (Section 2) and explore its safety assurance characteristics (Section 3). We then propose a modular safety assurance model by diffusion of the different actor responsibilities, e.g. system integrators, cloud service providers and 'Things' suppliers (Section 4). Our model aims to provide a basis for cooperative, on-demand and continuous safety reasoning in order to address the reconfigurable and compositional nature of Industry 4.0 architectures. We illustrate our approach based on a smart factory use case (Section 5) and conclude in Section 6.

2 Industry 4.0 Architecture

In this section, we introduce a generic architecture for Industry 4.0. This architecture comprises three levels, as depicted in figure 1, where the Things and Fog/Edge levels typically represent the local part of the system, while the Cloud represents a remote infrastructure that is usually owned by a third-party service provider:

- *The Things Level* is composed of a set of Things that enable interaction with the physical environment via different sensing/actuating devices. We consider a Thing as an object capable of communicating with other networked devices [2]. Due to the limited storage and processing power, devices from this level rely on the Fog or Cloud infrastructures for storage and processing services.
- *The Fog Level* is composed of a set of Fog/Edge devices that are directly connected to Things or/and Cloud infrastructure. We consider Fog devices to be local computational devices that offer advanced storage and processing power to the Things and rely on remote Cloud infrastructure for high-power computing and storage. The Fog devices receive data from the Things and, depending on the system configuration, might forward the data to the Cloud infrastructure. Moreover, the Fog devices may perform partial processing of the data and directly instruct commands to the Things.
- *The Cloud Level* is composed of a set of remote servers providing on-demand capabilities. The Cloud infrastructure typically receives data from the Fog devices, processes the data and forwards commands to the Things via Fog devices.

The distribution of control, authority and responsibility between the Things and the Fog and Cloud infrastructures depends on factors such as (1) performance,

e.g. avoiding the Cloud for hard real-time requirements, (2) global and adaptive services, e.g. Big Data analytics via the Cloud and (3) local situational awareness, e.g. via smart IoT-based devices. Understanding the behaviour and integrity of the individual Things and infrastructural elements, and their interactions, is a pre-requisite for assuring the safety of Industry 4.0.

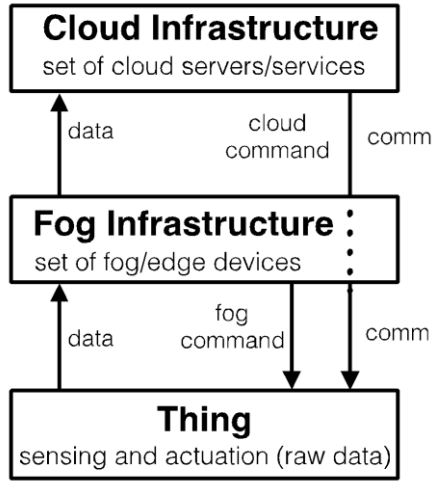


Fig. 1. Industry 4.0 Generic Architecture

3 Safety Characteristics for Industry 4.0

Considering the capabilities of Industry 4.0, in this Section, we explore key characteristics of its safety assurance.

1) *Modular and Cooperative:* The safety assurance for Industry 4.0 will often have to be cooperative in a sense that a safety or assurance case cannot be built by a single stakeholder or organisation. Since the implementation of the business models is shifting from a single company to a network of service providers [14], so does the resulting system shift from a standalone system to a network of devices and services, performing, cooperatively, a number of functionalities. Each business participating in the integrated system, e.g. as a Thing supplier (be it a "dumb" or a "smart" connected device), should accompany the provided Thing with a set of safety assurances for different usages. However, since the suppliers cannot provide all the needed safety assurances out-of-context, certain properties should be assured by the integrator in the context of the particular usage of the Thing.

2) *Continuous:* Safety cases are used to justify how the risk of each identified

hazard has been eliminated or adequately mitigated. Industry 4.0 assumes that a modular factory can be reconfigured quickly and flexibly. The safety assurance of such a factory is expected to be in a position to accommodate this widening of flexibility. For safety cases, they should comprise evidence to make a convincing argument to support the relevant safety claims [15]. However, some claims and pieces of evidence might get invalidated due to reconfigurations that commonly take place in the factory, e.g. changes to the manufacturing processes and services. Hence, safety cases might be out of date and no longer reflect the actual safety performance of the system. To this end, the safety cases should be proactively reviewed and continuously maintained in order to justify the evolving status of the factory [6].

3) *On-demand:* As motivated in the previous characteristic, safety cases should be maintained after changing the associated factory to continuously demonstrate the status of the safety performance. Sometimes, however, updating the safety cases is not feasible because of the nature of the changes. That is, there might be drastic changes to the factory that could introduce new and different types of hazards that require repeating the entire safety assurance process and generating more and/or new pieces of evidence. Here, re-constructing the safety cases might be necessary as a more cost-effective option compared to updating the existing cases [22].

In this paper, we limit our focus to the modular and cooperative characteristics of safety assurance for Industry 4.0, considering the overall safety case for Smart Factories and future needs for continuous and on-demand assurance.

4 Industry 4.0 Safety Assurance Approach

Assurance can be defined as justified confidence in a property of interest. In high-risk domains, assurance is typically demonstrated through the provision of an assurance case, consisting of a structured argument, i.e. justification, supported by evidence [15]. In this paper, the assurance case is for safety properties (aka safety case). As discussed in Section 3, due to the co-operative nature of IoT, it is not possible for any single stakeholder to provide the assurance case for the entire system.

The constituent Things, and the required infrastructure elements will be developed and provided by different organisations. It is these separate organisations that have the knowledge of the properties and characteristics of their components (i.e. Things or infrastructure elements). However, these suppliers are only able to reason about the assurance of their own components and can say little about the assurance of the IoT system as a whole, especially with regard to system- level conditions such as hazards, accidents and harm. The system integrator must

therefore consider what is required for safety assurance and then show that the Things or infrastructure elements being used are able to support this.

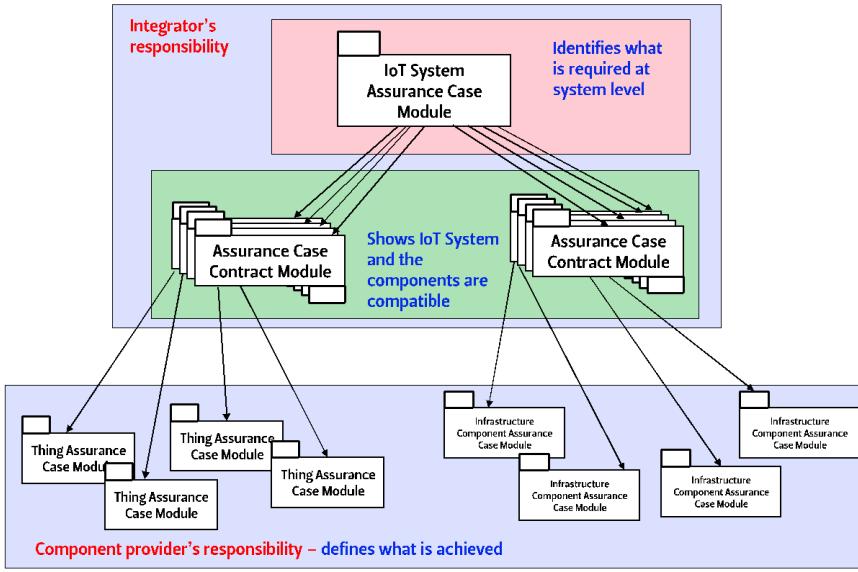


Fig. 2. Proposed IoT Assurance Case Architecture

This leads us to propose a modular approach to assurance for IoT-based systems as indicated in figure 2. The figure shows the overall assurance case structure for the IoT-based system, split into a number of modules, where each module reasons about a different aspect of the system. There are assurance modules for each of the Things and infrastructure elements, and modules dealing with the assurance of the integration of these into an IoT system. The different stakeholders have assurance responsibilities within the structure in figure 2 in order to ensure that a compelling overall assurance case for the IoT system can be created. These responsibilities are discussed below.

4.1 Responsibilities of Things or Infrastructure Providers

Each of these providers must define an *assurance contract*. This contract defines the set of properties that they are able to assure and a definition of potential failure behaviour of their Things or infrastructures. In order to be usable as part of the integrated assurance case for the IoT system, each of the identified properties should be defined with the following assume-guarantee reasoning form:

if {condition} then {Thing or infrastructure} shall provide {property} with confidence of {confidence}

The *condition* and *property* represent the assumptions and guarantees of an assume-guarantee contract [23]. The *condition* and *confidence* of this assume-guarantee contract specification is crucial to our approach. For any *Thing* or *infrastructure*, there exist limitations on the circumstances under which it can perform its function. For example, an assurance contract for a pressure sensor may include:

If temperature is greater than -20°C then pressure sensor shall provide air pressure value with accuracy of 0.001% with confidence of 99%.

It should be noted that, unless some failure has occurred, the pressure sensor is expected to provide an air pressure value. However, at temperatures below -20°C the confidence in that value will be reduced. If this confidence is not defined at these lower temperatures, then the property cannot be assured outside that temperature range. This may then require alternative pressure sensing capabilities (or some other guarantee of temperature range) in order to create the assurance case.

Knowing the level of confidence with which a Thing or infrastructure can guarantee a particular property is also crucial to the integration process as it enables the overall level of assurance for the system properties to be determined. Further, each Thing or infrastructure provider must be able to reason about the completeness and correctness of the failure behaviour definition provided as part of the contract. These definitions of such failure behaviour are also taken into account when assessing the assurance of the integrated system. It should be noted that the information required of the Thing or infrastructure provider described above is specific to the Thing or infrastructure, but in no way specific to the particular IoT system of which that Thing or infrastructure may become a part. This facilitates the use of independently, commercially developed and reusable components as part of the safety assurance framework.

4.2 Integrator's Responsibilities

The integrator has responsibility for creating the IoT system by utilising the Internet-enabled Things and infrastructure elements. The integrator therefore also has responsibility for demonstrating the overall safety assurance of the IoT system. As previously discussed, the integrator should have available to them information about the assurance of the individual Things or infrastructures through the assume-guarantee contract specifications. The integrator must show how the assurance provided for the Things or infrastructures can be used to demonstrate

the assurance of system- level properties. In particular, the integrator must identify the hazards, i.e. sources of potential harm, and their associated risks, posed by the system, e.g. unsecured loads, laser radiation or heavy machines operating in the presence of operators. For any configuration of Things or infrastructures, the integrator must then determine the safety requirements for each of these by identifying how the Things or infrastructures may contribute to hazards (this could for example be done through considering deviations on the functionality or interactions).

Once these requirements are known, the safety assurance case for the IoT system can be created if it can be demonstrated that 1) the properties in the contracts are able to satisfy the assurance requirements defined for the IoT-based system with sufficient confidence, and 2) the contracts of the relevant Things or infrastructure elements are satisfied (the properties and conditions are met and the failure modes are mitigated). As discussed, the Thing or infrastructure element provider has responsibility for specifying the contract for that element and ensuring the properties are met, however it is the responsibility of the integrator to ensure the conditions are satisfied, and the identified failure modes of the element are mitigated in the context of the overall IoT system (through a variety of mechanisms such as redundancy, monitoring, operational constraints etc.).

In order to facilitate this integration of an overall safety case, we propose the use of *assurance case contracts*. Assurance case contracts provide a mechanism for recording and justifying the agreed relationship between assurance case modules. Figure 2 shows assurance case contracts being established between the IoT-based system assurance case module and the individual component modules. The structure that such a contract module might have is illustrated as a pattern in figure 3, using the Goal Structuring Notation (GSN). Readers who are unfamiliar with this notation are referred to the GSN Standard [1] for more detailed information.

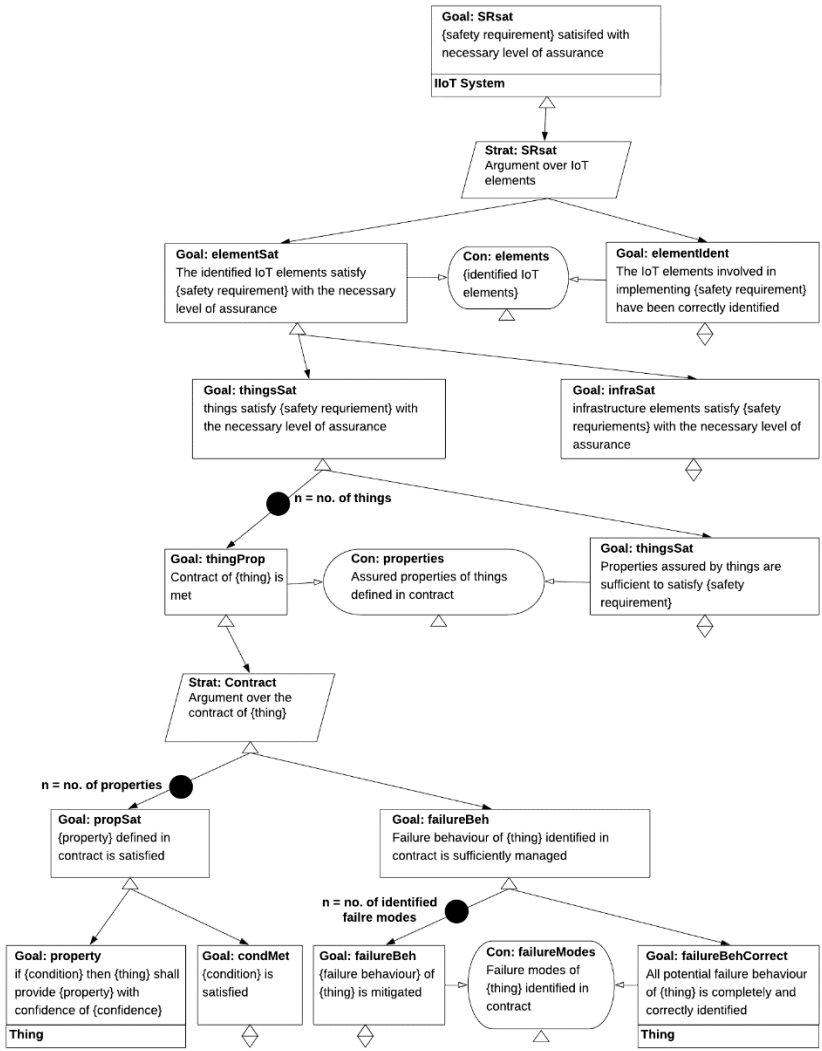


Fig. 3. Structure of an Assurance Case Contract for IoT System

Figure 3 shows how in order to assure a safety requirement identified by the integrator, a number of the Things or infrastructure elements may need to be considered. For each of these, the contract defined for those elements is used to make the assurance argument. In Figure 3 we show only how this is done for Things, but a similar argument structure would be used for infrastructure elements as well. In order to form the assurance case contract, it must be demonstrated that the properties defined in the contract for each element are sufficient to satisfy the

safety requirement. It must then be demonstrated that each aspect of the contract for each element is satisfied. Claims about the satisfaction of the properties, and the identification of failure behaviour, are supported by a safety case module developed by the provider of that element and provided to the integrator along with the element itself.

Needless to say, establishing and justifying assurance case contracts is a challenging task. The specification of the assurance model and clear definition of the supplier's assurance responsibilities are merely a first step towards this. A contract- based assurance approach is potentially desirable for an IoT- based system as the contract helps to determine whether the relationship between the assurance case modules continues to hold and the (combined) safety assurance case remains valid when Things or infrastructures are altered or substituted in the system. This issue is discussed further in Section 6.

5 Use Case

In this section we present a fictitious, yet representative, Smart Factory and focus on a single part of the factory to illustrate safety assurance for Industry 4.0. We focus on a Warning Light System (WLS) as a safety measure that includes IoT-related elements. We demonstrate our approach by performing safety analysis of the WLS and developing a corresponding argument for the system based on the assurance case contract structure presented in Section 4.

5.1 Smart Factory Description

Our use case considers scenarios where the requirements and design specification for the manufacturing of a product are provided via a Cloud-based service. Some of the manufacturing control capabilities reside remotely on the Cloud, e.g. scheduling and design reconfiguration. Others are managed locally either at the Fog or Things levels. More specifically, our use case considers a manufacturing factory in which a number of computer-based machine tools make a range of gearbox shafts from metal blanks. The blanks, which weigh about 4kg each, are delivered in pallets of 50, and stored in an automated warehouse until they are required. Finished products are also packed into pallets and taken to a holding area before being shipped to the main assembly plant.

The movement of pallets around the plant is managed using an Automatic Guided Vehicle (AGV) system. The system consists of a number of battery-powered vehicles, each fitted with pallet handling equipment, whose movements are

directed by an AGV Central Control Fog. This is interfaced to a Warehouse Control, Holding Area Control and Machining Control Fogs, so that stock movement requirements can be fulfilled. Each AGV will carry only one pallet at a time. The conceptual flow of materials is illustrated in figure 4.

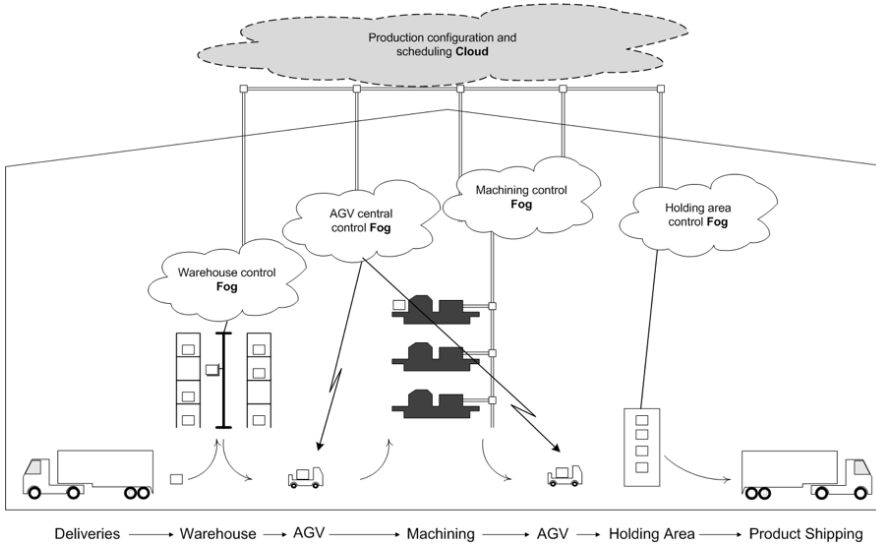


Fig. 4. Flow of Materials and Information through the Factory

To manage different automated activities in the factory, Light Imaging, Detection, and Ranging (LIDAR) sensors are positioned to cover the whole factory. Such a setup allows the Smart Factory to "see" what is going on, i.e. in real time, and to manage the activities accordingly. A Cloud service is used for the integration of the LIDAR inputs and for modelling the activities in the factory. This Cloud service allows for customisable features to be implemented specific to different factory operations. Special docking stations are provided for the AGVs, each weighing about 0.8 tonnes. The vehicles will normally be directed by the central Fog to return to these charging stations when they are not required to move pallets. The factory is not fully automated, and people cannot be excluded from the areas where the AGVs operate.

5.2 Hazard Analysis

Since the factory employs both human workers and machines of different autonomy levels, there are many factory-level hazards, e.g. proximity to heavy moving objects. One general safety measure is to define restricted areas for the different

factory configurations to protect the human workers from both the moving machinery and the dangerous goods they transport. In this use case, we focus on a single factory-level hazard: *"Unauthorised AGV vehicle enters the restricted area"*. Due to the noise protection procedures that human workers may be using in certain configurations, audio warning is not sufficient, so a visual warning light system is also needed. Amongst the different safety requirements specified to address this hazard, we focus on the following requirement: *"A warning light shall be signalled when an unauthorised AGV enters the restricted area"*. This requirement is allocated a Safety Integrity Level (SIL) 2, based on the likelihood and severity of the considered factory-level hazard.

To achieve this requirement, several other sub-requirements should be specified. We mention only some:

R1: The system shall distinguish between authorised and unauthorised AGVs.
R2: The scope of the restricted area shall be specified to 5cm degree of precision.
R3: The signalling of the warning light shall occur within 0.5sec from an unauthorised AGV entering the restricted area.

The main objective of the proposed Warning Light System (WLS) is to monitor restricted areas where certain types of objects (humans, robots, vehicles, etc.) are prohibited due to safety reasons. The system is intended to trigger a warning light if an object classified as prohibited under the given factory configuration appears in the designated restricted area.

The high-level architecture of the WLS is presented in figure 5. WLS is implemented using the Cloud service and the factory LIDARs. We focus on a particular configuration and a specific restricted area for that configuration, as presented in figure 5. The considered restricted area includes 3 LIDARs, 4 access points and 4 warning light lamps. The gateway and local control device are located within the factory, but outside of the restricted area. The access points facilitate wireless communication between the sensors/lamps and the gateway, while the gateway enables connection to the cloud service that acts as a control node.

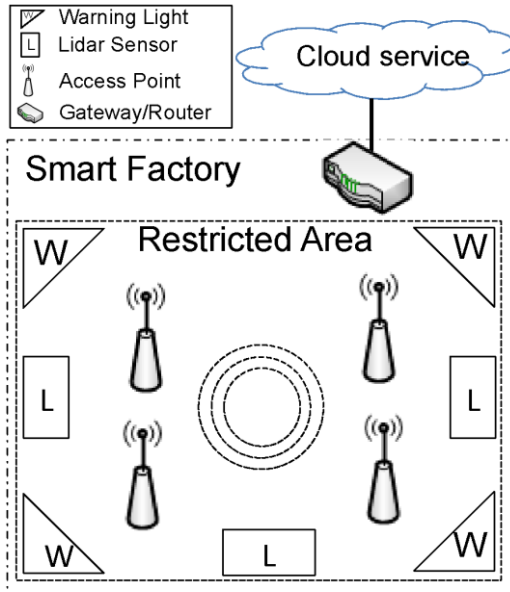


Fig. 5. Smart Factory Use Case: WLS Configuration

The considered WLS is composed of:

Things:

1) LIDARs 1-3 (identical)
2) Warning Light 1-4 (identical)

Infrastructure elements:

3) Gateway/Router (local control node)
4) Access Point 1-4 (identical)
5) Cloud Service (control node)

The Cloud service is responsible for processing the data and commanding the activation of the warning light via the local network. The Cloud is also responsible for monitoring all moving objects in the factory. The local controller is only responsible for the most severe restricted area violations. As such, it only monitors certain objects entering the area. The initial requirements are further decomposed and allocated to the IoT system elements to more clearly specify their function. For example, for the R1 requirement we specify sub-requirements such as:

R1.1: LIDARs shall detect all objects entering the restricted area.
R1.2: The cloud service shall analyse and classify all detected objects.
R1.3: The gateway shall analyse and classify only the most dangerous objects.
R1.4: The gateway shall transmit all the sensor data to the cloud service.

Similarly, for the requirement R3, we decompose it to allocate the timing requirements on the operations of the different elements. For example, a sub-requirement R3.1 can be specified as: "The warning lights shall engage on receipt of the engage command within 0.2sec".

5.3 WLS Failure Analysis

So far, we have defined safety requirements for WLS without considering failures of the individual elements. In this section we consider hazardous contributions of all the WLS IoT system elements and their contributions to the considered hazard. Some of the identified hazardous failures for the IoT system elements are as follows:

- LIDARs
    - No signal provided
    - Unable to detect unauthorised object entering restricted area
    - Signal reports incorrect light conditions
- Warning light lamps
    - The warning light does not turn on when requested
    - The warning light turns on with a delay greater than 0.2sec
- Access Points
    - Access point fails to route data to the Gateway
    - Access point takes longer than intended to route data
- Cloud Service
    - Cloud does not generate warning signal request
    - Cloud generates an incorrect warning signal request
    - Cloud takes longer than intended to generate warning signal request.

We have also derived safety requirements to address the above hazardous failures. For example, these requirements include the following:

1) "Each restricted area shall have at least two warning lamps visible from every position in the area",
2) "Each moving object in the factory shall have a marking detectable by LIDARS", and

3) "Human workers shall be notified of the WLS failures".

All the derived requirements are assigned with at least SIL 2, based on the corresponding higher-level requirement.

5.4 Assurance Case Contract Example for WLS

The application of the assurance case contract, as defined in Section 4, is presented in figure 6.

In the presented argument we focus on the safety requirement R3 of WLS and detail in particular the warning light lamp element. The supplier of the lamp is able to provide an assurance case for the lamp that supports various claims about the lamp as detailed in the assume-guarantee contract. In the example in figure 6 we see that the lamp assumes a constant power supply and working temperature in a predefined range in order to provide assurance of maximum light intensity within 0.2 seconds during the promised lifespan.

The confidence in this claim is provided by the lamp assurance case. In forming the assurance case contract shown in figure 6, this claim about the lamp is used to support a safety requirement as part of the higher-level factory assurance case (in other words, the assurance case contract reasons that this lamp is sufficient, from a safety perspective, for its use as part the factory operations).

Figure 6 shows how the assurance case contract also must consider the known failure behaviours of the lamp as detailed by the supplier. The effects of the failure behaviours are shown to be mitigated by the AGV and the Smart Factory configuration.

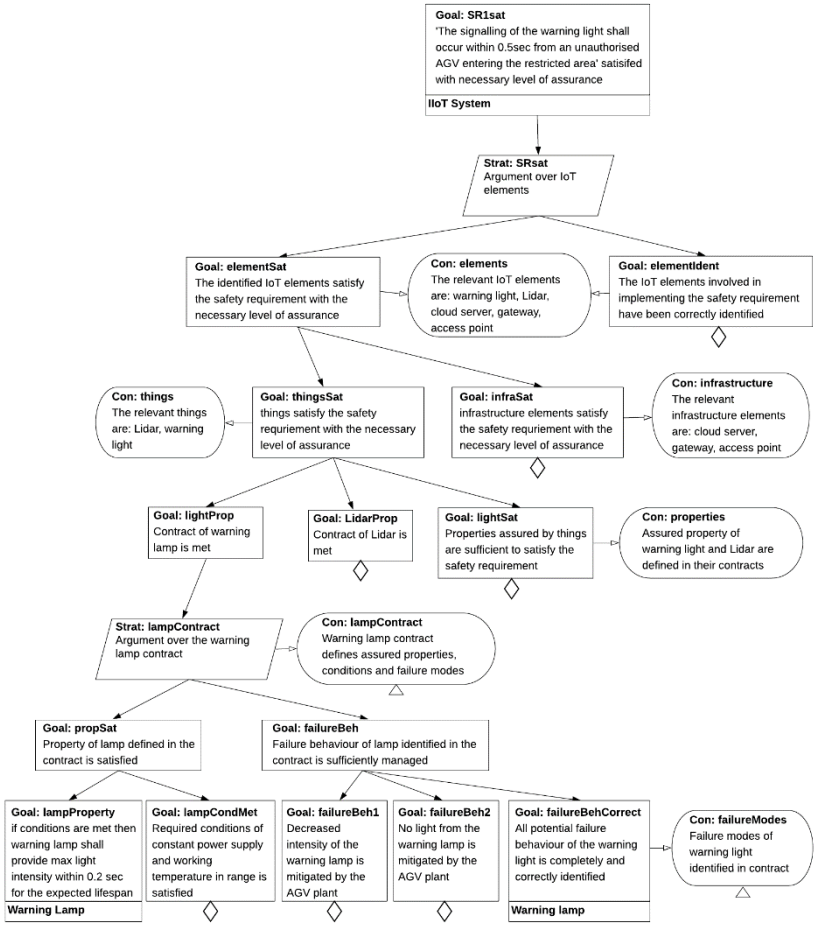


Fig. 6. The Warning Light Lamp Assurance Case Contract

6 Discussion and Conclusions

We have highlighted a number of safety challenges posed by Industry 4.0 and proposed a modular assurance approach that has the potential to address some of these challenges, particularly with regard to the compositional and configurable nature of IoT-based architectures. In essence, our approach builds on past and current research on assume-guarantee reasoning, contract-based assurance and modular certification for safety-critical applications [20] [16] [8]. Historically, these approaches formed the basis for safety cases and certification for systems

in various domains including automotive [24] and aviation [5]. However, some fundamental safety assurance problems remain and have to be addressed as a prerequisite for realising the general-purpose vision of Industry 4.0. We explore, and reflect on, these in the rest of this section.

A. Industry 4.0 Safety Validation Challenge

We discussed the potential for modular and contract-based reasoning to drive the structure of the overall safety case for Industry 4.0 architectures and meet the safety requirements. However, the fundamental problem does not lie in how the configurable architectures meet the safety requirements. Rather, the issue lies in the *generation* of these safety requirements in the first place. The ad hoc assemblage of Things and infrastructures for Industry 4.0 architectures will likely result in new hazards and/or risk ratings and as such new safety requirements. These *emerging* hazards are due to expected, yet unpredictable, reconfigurations or redeployments of the architecture in multiple contexts (i.e. we cannot assume that the world is stable, and variation only lies within our system). This will often mean that the hazard analysis, or at least a large part of it, will have to be manually repeated for each reconfiguration or deployment and should produce an updated set of safety requirements (i.e. each of these changes might be considered as a new factory). This can be seen as undermining the general-purpose and reusable nature of Industry 4.0 architectures, i.e. where rapid reconfiguration and deployment is seen as a unique selling point. In other words, modularity and contract-based reasoning largely deal with the *verification* issue whereas hazard analysis of the whole system addresses the *validation* problem. Safety validation, against the intended real-world usage, is the essence of safety assurance and how risk and harm are assessed, perceived and accepted.

B. Industry 4.0 Safety Confidence Challenge

In our example definition of assurance contracts for Things and infrastructures within Industry 4.0 architectures, we highlighted the need to specify necessary properties that have to be provided (e.g. measurement of air pressure values) to a particular level of integrity (e.g. accuracy of 0.001%) and confidence (e.g. 99%). For large socio-technical IoT systems such as Smart Factories, confidence will inevitably be measured using different qualitative [11] and quantitative [6] indicators. Propagating confidence from the different qualitative and quantitative measures associated with the various Things in an infrastructure is necessary to assess confidence in the safety of the overall configured system [10]. This has to be performed dynamically and on-demand to address the particular reconfigurable characteristics of Industry 4.0 architectures. This is a grand safety challenge for Industry 4.0 (and safety engineering generally). Current approaches to specifying confidence and associating it with assume-guarantee contract specification for individual components is relatively straightforward compared to the challenge of assessing, dynamically, confidence for the different reconfigurations.

C. Industry 4.0 Commercial Pressure Challenge

The financial appeal of commercially available Things and infrastructures, which *appear* to be dependable although they are not developed for safety-critical applications, should not be undermined. The business pressure is mounting on safety engineers to accept the use of, relatively *cheap*, consumer electronics and commercially available cloud-based services. Resistance from the safety community on the basis of difficulty or novelty could be counter-productive. This might result in alienating or excluding safety engineers when design decisions are made or more likely, and sometimes rightly so, appealing to reduction in overall risk despite increases in technological risks (e.g. a typical risk-benefit argument in clinical applications in which clinical benefits outweigh technological risks [13]).

D. Industry 4.0 Security-Informed Safety Challenge

There is now almost a consensus on the necessity to address cyber security in safety assurance [4]. This issue takes a greater significance for Industry 4.0 where remote connectivity and the use of commercially available infrastructures and Things expose the system to a wide range of cyber threats (particularly Distributed Denial of Service [9]). Security risks tend to be more dynamic than safety risks. As such, exploring the extent to which an Industry 4.0 architecture might have to reconfigure in the event of a security breach is a significant challenge, particularly in how it might compromise safety assurance (i.e. a typical trade-off between safety and security that has to be made more explicit in the safety assurance case).

In conclusion, in this paper, we explored a number of characteristics for the safety assurance of Industry 4.0 and focused on modularity as a key aspect of the overall assurance case for safety. We also highlighted some grand challenges that remain and will be a focus for our future work.

Terminology

XaaS – Anything (X) as a Service

Internet of Things (IoT) - a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.

Thing devices – enable interaction with the physical environment via different sensors/actuators.

Cloud Computing – a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Edge Computing – a decentralized infrastructure in which parts of applications, management and data analytics are moved to the end devices such that computing is performed as close as possible to the data source.

Fog Computing - a decentralised infrastructure in which parts of applications, management and data analytics are moved into the network itself using a distributed computing model.

Fog/Edge Devices – local computational devices that offer advanced storage and processing power to the Things and rely on remote Cloud infrastructure for high-power computing and storage.

Acknowledgments This work is supported by the Swedish Foundation for Strategic Research (SSF) via the project Future factories in the Cloud (FiC).

References

[1] Goal Structuring Notation working group, November 2011.

[2] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[3] A. Bessani, R. Kapitza, D. Petcu, P. Romano, S. V. Gogouvitis, D. Kyriazis, and R. G. Cascella. A look to the old-world sky: EU- funded dependability cloud computing research. *Operating Systems Review*, 46(2):43–56, July 2012.

[4] R. Bloomfield, K. Netkachova, and R. Stroud. Security-informed safety: if its not secure, its not safe. In International Workshop on Software Engineering for Resilient Systems, pages 17–32. Springer, 2013.

[5] P. Conmy, M. Nicholson, and J. McDermid. Safety assurance contracts for integrated modular avionics. In *Proceedings of the 8th Australian workshop on Safety critical systems and software-Volume 33*, pages 69–78. Australian Computer Society, Inc., 2003.

[6] E. Denney, G. Pai, and I. Habli. Dynamic safety cases for through-life safety assurance. In *Proceedings of the 37th International Conference on Software Engineering-Volume 2*, pages 587–590. IEEE Press, 2015.

[7] B. Esmaeilian, S. Behdad, and B. Wang. The evolution and future of manufacturing: A review. *Journal of Manufacturing Systems*, 39:79 – 100, 2016.

[8] J. Fenn, R. Hawkins, P. Williams, and T. Kelly. Safety case composition using contracts-refinements based on feedback from an industrial case study. In *The Safety of Systems*, pages 133–146. Springer London, 2007.

[9] Guardian. DDoS attack that disrupted internet was largest of its kind in history, experts say. www.theguardian.com/technology/2016/oct/26/ ddos-attack-dyn-mirai-botnet.

[10] J. Guiochet, Q. A. Do Hoang, and M. Kaaniche. A model for safety case confidence assessment. In *International Conference on Computer Safety, Reliability, and Security*, pages 313–327. Springer, 2015.

[11] R. Hawkins, T. Kelly, J. Knight, and P. Graydon. A new approach to creating clear safety arguments. In *Advances in systems safety*, pages 3–23. Springer, 2011.

[12] W. He and L. Xu. A state-of-the-art survey of cloud manufacturing. *Int. J. Comput. Integr. Manuf.*, 28(3):239–250, Mar. 2015.

[13] ISO. *ISO 14971: medical devices-application of risk management to medical devices*. ISO, 2012.

[14] H. Kagermann, J. Helbig, A. Hellinger, and W. Wahlster. *Recommendations for Implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry*. Forschungsunion, 2013.

[15] T. P. Kelly. *Arguing safety: a systematic approach to managing safety cases*. University of York, 1999.

[16] T. P. Kelly. Concepts and principles of compositional safety case construction. *Contract Research Report for QinetiQ COMSA/2001/1/1*, 34, 2001.

[17] P. Mell, T. Grance, et al. The nist definition of cloud computing. 2011.

[18] PILZ. Industrie 4.0 – safe and smart (white paper), June 2016.

[19] B. P. Rimal, E. Choi, and I. Lumb. A taxonomy and survey of cloud computing systems. In *2009 Fifth International Joint Conference on INC, IMS and IDC*, pages 44–51, Aug 2009.

[20] J. Rushby. Modular certification. Technical report, Sept. 2001.

[21] J. Rushby. The interpretation and evaluation of assurance cases. Technical Report SRI-CSL-15-01, Computer Science Laboratory, SRI International, Menlo Park, CA, July 2015. Available at http://www.csl. sri.com/users/rushby/papers/sri-csl-15-1-assurance-cases.pdf.

[22] J. Rushby. Trustworthy self-integrating systems. In N. Bjørner, S. Prasad, and L. Parida, editors, *12th International Conference on Distributed Computing and Internet Technology, ICDCIT 2016*, volume 9581 of *Lecture Notes in Computer Science*, pages 19–29, Bhubaneswar, India, Jan. 2016. Springer-Verlag.

[23] A. Sangiovanni-Vincentelli, W. Damm, and R. Passerone. Taming dr. frankenstein: Contract-based design for cyber-physical systems. *European journal of control*, 18(3):217–238, 2012.

[24] D. Schneider and M. Trapp. Conditional safety certification of open adaptive systems. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 8(2):8, 2013.

[25] Telecommunication standardization sector of ITU. *Overview of the Internet of things*, Y.2060 edition, 6 2012.

[26] D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer. Cloud manufacturing: Strategic vision and state-of-the-art. *Journal of Manufacturing Systems*, 32(4):564 – 579, 2013.

Safety in Space: A Changing Picture?

Emma Ariane Taylor

Safety and Reliability Society (SaRS)
Manchester, U.K.[1]

Abstract *Space exploration and utilisation is increasingly focussed through the lens of private space activities. Whilst international treaties and agencies provide the framework for access to, and utilisation of, space, the rapidly increasing activities of private entities is leading to new challenges, both legislative and technical. Governments are responding in a range of different ways to meet the goal of supporting this growing sector whilst ensuring that their national and international obligations are met. Based on the review of current and near future trends, some suggestions are made on risk assessment methodologies that may help provide clarity when assessing safety in space.*

1 Introduction

With the increase in space exploration and utilisation, increasingly driven through the growth of private space activities, it can be stated that the established governmental-led space sector no longer acts as a standalone arbiter of space activities. Based on that assumption, it is worthwhile evaluating what changes can usefully be made to the way in which those activities are managed in order to ensure that safety in space is maintained. The increasing societal interest in this field and the increasing number of countries involved in space system manufacture, launch and operations may also influence how safety in space is perceived and so societal expectations on how it should be managed. This paper provides a snapshot of this rapidly changing field and notes some assessment frameworks which might usefully provide a practical perspective on safety in space. The space sector is of course somewhat different from most other high hazard sectors. Explosions of launch vehicle, whilst not of the same scale as oil and gas explosions or nuclear containment failures in terms of potential fatalities, occur relatively frequently. Some risks are international e.g. uncontrolled re-entry of large

[1] Safety and Reliability Society (SaRS), Albert Street, Oldham, Manchester, OL8 3QL. info@sars.org.uk & emma.a.taylor@googlemail.com.

© Emma Ariane Taylor 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

satellite crossing low altitude orbits and high altitude airspace, then landing in either international waters or national territories.

For consistency, the term safety in space is considered to apply to all elements of a space mission i.e. an activity with the purpose of placing an object into orbit including launcher (example in Figure 1), satellite, and all other supporting systems that may leave the surface of the Earth as part of achieving orbit. Safety in space also considers the safety of space objects in orbit, whether manned or unmanned, as well as safety of objects on their return to Earth, controlled or uncontrolled. These common definitions are made for the purposes of this paper; as a caveat they may not align fully interpretations of current and future legal definitions within national legislation. For clarity, this paper cannot be used as setting the legal context for safety in space assessments and the contents are provided for discussion only. Other definitions of 'space safety' also exist including from the European Space Agency's European Centre for Space Law (ESA-ECSL): "Space Safety: Sustainability of Space Activities, Space Situational Awareness (SSA) and Space Traffic Management". These three areas fall under the general category of 'space debris' (ECSL 2019).

Fig. 1. Ariane 6 launcher (Wikipedia, 2019a)

2 International context

The United Nations (UN) provided the context for the first international discussions on the use of space, leading to the signing of the Outer Space Treaty (OST) in 1967 (UN 1967) (full name: *Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space, including the Moon and Other Celestial Bodies*) by more than 100 nations. Subsequent treaties and agreements

have been established through the co-ordination framework of UNCOPUOS (the UN Committee for the Peaceful Uses of Outer Space). One treaty, the Liability Convention, builds on the liability provisions articulated in the OST (Art VII). It sets out absolute liability for physical damage suffered on the surface of the Earth, or to aircraft in flight, and establishes a fault-based liability regime for space objects in outer space. The Registration Convention signed a few years later, also builds on the OST (Art VIII). It requires national registries of space objects and international registration of those space objects (UN 1971, UN 1974). Not all spacefaring nations are signatories to all agreements (UN 2019a) although all who have signed OST are bound by Art VII and Art VIII even if they have not signed the respective conventions. The UN General Assembly Resolution (UN 1962) established a basis for the OST in customary international law which is binding on all states regardless of whether or not they've signed up to the OST.

The two UN specialised agencies ICAO (International Civil Aviation Organization) and IMO (International Maritime Organisation) will continue to play a role in establishing common norms accepted by states for use of air and sea, ICAO now acting as a hub for exchange of information on space transportation (ICAO 2019). The International Telecommunications Union (ITU) was founded in 1865 and is a now UN agency whose Space Services Division allocates radio spectrum frequencies for ground and satellite use so maintaining a 'master international frequency register' to prevent signal interference (ITU 2019).

More recently UN have discussed the long-term sustainability of space at the General Assembly level, as part of UN COPUOS submissions (UN 2019b). The message provided is clear:

> "The Earth's orbital space environment constitutes a finite resource that is being used by an increasing number of States, international intergovernmental organizations and non-governmental entities. The proliferation of space debris, the increasing complexity of space operations, the emergence of large constellations and the increased risks of collision and interference with the operation of space objects may affect the long-term sustainability of space activities. Addressing these developments and risks requires international cooperation by States and international intergovernmental organizations to avoid harm to the space environment and the safety of space operations".

During the preparation of this paper, the International Institute of Air and Space Law (IIASL) Hague International Space Resources Governance Working Group (IIASL, 2019) published their 'building blocks', an international framework on space resource activities. It evaluates concepts that are being discussed and explores how it might be ensured that associated activities meet existing treaty obligations regarding on-orbit operations and space resource rights. Space resources "*include mineral and volatile materials, including water, but excludes (a) satellite orbits; (b) radio spectrum; and (c) energy from the sun except when collected from unique and scarce locations*". Whilst the term 'space resources' is typically used to refer to exploration and utilisation of minor bodies or planetary systems in the solar system (i.e. not the Earth's orbital regions), the statements made on

the prevention of space debris, the use of sustainable technologies and other topics may influence indirectly the Earth orbital environment, potentially through changes in standards and interpretation and evolution of existing international treaties and space law.

3 Earth orbital environment

A focus on preservation of the orbital environment has gained importance in recent years, and space sustainability is now a recognised concept (e.g. Newman and Williamson 2018). Due to the characteristics of the orbital environment, a satellite may remain in orbit indefinitely, or be brought down to Earth through planned or unplanned measures. In orbit it is vulnerable to impact from other objects. There are of course other (natural) hazards in space, some of which also affect high altitude air travel and impact the surface e.g. space weather, asteroids (large objects) through to micrometeoroids, gamma radiation exposure on commercial flights etc.

This problem of 'space debris' (also known as 'orbital debris') is discussed within various international fora including the International Space Debris Coordination Committee (IADC) and the International Organization for Standardization (ISO). From the first issue in 2010 of ISO24113 on mitigation of orbital debris (ISO 2019a), a further 40-plus space debris-related standards are either in publication or preparation, including safety requirements for launch site operations (ISO 2019b). The UN have also produced space debris mitigation guidelines (UN, 2010). As for the master international frequency registry, there is a launching object registry. This however only covers the initial launch, high accuracy in-orbit position and tracking information is not readily available to all, the commonly used Two Line Element (TLE) sets have known limitations (Celestrak, 2019) although additional modelling can be carried out (e.g. Racelis and Joerger 2018). However, information about potential hazards on an orbiting object and its configuration is not available, so leaving knowledge gaps for any organisation dealing with an in-orbit accident, in-orbit retrieval or other management of hazards.

The space debris standards are mapped to a number of the UN Sustainable Development Goals, showing the greater emphasis on sustainability of space operations as well as the benefits that space-based resources can bring towards combating global issues (UN 2019c). Orbital debris will not only limit the lifetime of operating orbital satellites but may also impact missions leaving Earth's orbital regions. The European Space Agency (ESA) have in place a comprehensive and growing programme on Safety & Security, covering monitoring and safeguarding of space, along with protecting our planetary environment (ESA 2019a). In-orbit

space environment operational decision making support is available (ESA, 2019b). This builds on many years of space activities which, as they remain primarily non-private sector in nature (i.e. the basis of the European Space Agency's remit is unchanged) have not been examined in detail in this paper.

4 Implementation of treaties

Over time States have implemented a range of interpretations and approaches to implementation of UN treaties and other international agreements) into national law. As an object launched into space will always carry obligations of the launching state and the state that registers it, this tailoring is not unexpected. The obligations in the OST mean that states which are active in space operations need primary domestic legislation in order to discharge effectively their obligations under the OST. To date, governmental entities ('states') have in effect authorised all national activities from launch through to operations and decommissioning as well as international co-operative programmes. Again, this is implemented under the obligations of the OST i.e. they must authorise such activities.

Launchers of course will be launched from either the ground or the surface of the sea, with air launch to orbit capabilities now more available. All will travel through airspace. The complexity of interfacing international air law and space law, and the interface between the two remains the subject of some discussion (Dempsey and Manoli 2017). As for space law, international agreements on air law and maritime law are implemented as appropriate within national frameworks. At this time, due to the limited number of space launches and the location of some of the launch sites, the potential for tension between space operators and air operators is limited, but clarification is likely to be needed as the situation changes. Similarly, progress towards formulation of voluntary, non-binding instruments and their inclusion in national legislation is being made on the topic of space debris (Popova and Schaus 2018).

Design and operational requirements for launchers and satellites will be derived from those developed for mitigation of orbital debris and are therefore likely to impact how safety in space is achieved. The ISO committee TC20/SC14 (ISO 2019b) publishes a wide range of relevant standards: Design engineering and production; system requirements, verification and validation, interfaces, integration, and test; operations and support systems; materials and processes; space environment (natural and artificial).

As noted in a system safety study of the Columbia and Challenger disasters, "Safety considerations are especially critical during system development because it is very difficult to design or "inspect" safety into a system during operation" (Dulac et al. 2007). In a nutshell, once a system has been launched it cannot easily

be inspected or modified. (A different approach may apply for reusable launch vehicles.) Similarly, licensing of radio telecommunications frequencies (international and national level) plays an integral part in ensuring safety in space, as stakeholders are keen to ensure that a frequency is not blocked by a malfunctioning satellite nor an orbital slot (particularly in the high value telecommunications geosynchronous orbital region) disrupted by space debris (ITU 2019) or 'zombiesats' (Weedon 2010). Note also the potential impact of recently-published international framework on space resource activities (IIASL, 2019). These factors set the scene for the large scale entry of private operators into the space sector, principally over the past decade.

5 Private operators

Whilst private entities have been involved in the manufacture and launch of space objects for a number of years, they have typically operated under direct and/or sole supplier subcontract from government entities and are licensed under that basis. Now that an increasing number of privately-owned launch providers are launching space objects owned by commercial entities the picture is changing on who is responsible for the practical steps to be taken towards implementation of measures towards the challenge of maintaining safety in space. The US has led the way with missions by Blue Origin, SpaceX (Figure 2) and Virgin Galactic having a particularly high profile (Grady, 2017).

Whilst private companies 'own' the space assets, states are still liable for damage caused. States under whose registry the objects operate retain jurisdiction and control. As noted in OST (Art VI) authorisation and supervision of non-governmental entities by states is required.

Across the globe, there is a plethora of non-governmental entities who are recognised as currently offering (or planning to offer) equipment and services geared towards spaceflight e.g. the number of launch vehicle makers listed is approaching 40 (Wikipedia, 2019b), with a more comprehensive list of small satellite launchers also available (New Space Index, 2019). The growth of asteroid mining companies has received a boost through 2015 US legislation "that grants property rights to the resources on a planetary body (though not to the body itself) to whoever "gets there first" (Weinzierl 2019).

Stepping back, it appears as if many organisations are forming and failing within relatively short periods of time, as evidenced by the numbers of dormant or cancelled missions (New Space Index 2019). In particular companies focussing on exploiting planetary resources and deep space industries have both experienced significant financial difficulty due to the very high costs of initial infrastructure. Funds are being raised through private sources including venture capi-

talists (Wilson 2018); private equity holdings could supersede future public listings if wider Silicon Valley trends apply, although Virgin Galactic recently floated on the US Stock Exchange (Henderson 2019). It is unclear whether such financial ups and downs will influence an organisation's perspective on, or investment in, safety in space, although it is reasonable to assume that any investor would want to be reassured that a license to launch and operate would be granted on their investment. At its core, every mission is defined on the basis of a user receiving some 'value' from a space system or service, with another organisation providing that 'value' in exchange for revenue or other benefit. Ensuring safety in space will be part of exchange.

Fig. 2. SpaceX Dragon (Creative Commons, 2019a)

6 Parallels

Some parallels can perhaps be drawn with the rapid growth of oil extraction in the mid-1800s in the US, across states such as Pennsylvania, Texas and California, then moving to Alaska in the 1900s. New technologies were tried, tested and improved and the number and size of organisations changed rapidly through mergers, failures and new discoveries. Some accidents occurred, and it is fair to as-

sume that the licensing regime ("permit to drill") was likely lighter than the current day. Another more recent example, Egypt's gas "gold rush" has led to well in excess of 50 separate offshore gas fields now being mapped. Following a long period of state ownership and slow growth, followed by political turbulence, a new offshore gas development containing 30tn ft³ (trillion cubic feet) has been brought online in only three years (Stephen, 2019). These are just two somewhat different examples of how rapid growth of a market can occur when financial incentives, political context and technology availability combine in favourable circumstances. With the ongoing growth in launch services providing access to orbit (and reducing launch costs per Kg), the broadening availability and decreasing costs of resources (knowhow and equipment), perhaps similar growth conditions exist for the space sector. It is perhaps also remembering back to the early stages of civilian aircraft design and test i.e. a 'fly-fix-fly' approach – are there parallels to be drawn here with the growth elements of parts of the space industry?

7 New Space

Other trends to note include the use of language as part of the broader social commentary (Varghese, 2018): "fundamentally, space has very few rules… the relative lawlessness of space" and the delineation of "old space" (aka governmental programmes) and "new space" (everything else, often called New Space) in referring to the two types of markets (Williams, 2017). The balance between the two continues to shift, one self-described libertarian think-tank evaluates a future where space-based essentials, such as management of energy, materials and waste, are developed and owned privately, along with full private ownership of commercial orbital transportation services (Greason and Bennett, 2019). The authors outline "three historical examples (development of the western United States, ocean shipping, commercial aviation)" and comment on how "private sector's entrepreneurial innovations and large-scale investments enabled sustainable development".

The US's NASA would, in this scenario, shift their focus to research and exploration, and the US government could support initial infrastructure development e.g. transcontinental railroad was developed as a public-private partnership between the federal government and railroads. Such commentary, and the associated drive for a more investor-friendly regime, might not have been made only a few decades ago. The balance of influence between NASA and the US commercial launch providers who stepped in to provide services after the end of the Space Shuttle programme has already been acknowledged (in NASA's own words) as moving from "Contingency to Dependency" (Weinzierl 2019).

The term "New Space" is sometimes used to describe this ongoing sea change in the space industry's ecosystem, with various industry fora creating open source publications e.g. Handbook for New Actors in Space (Secure World Foundation, 2017). The strong growth in cubesats and smallsats has been a catalyst "for the implementation of new space laws as governments seek to regulate and transfer some of the responsibility down to the satellite operators" (Wheeler 2018). Any satellites so designed will work on their own but thousands have been announced that will launch in large constellations. These constellations are huge networks of satellites flying together in relatively low orbits designed to provide global, close-range coverage (observation or communications). It is reasonable to assume that the ongoing requirement to insure and indemnify the UK government against its obligations in the OST will perhaps influence the satellite insurance market, given the significantly larger numbers of satellites planned for these constellation launches (New Space Index, 2019).

Similarly, in the UK there is a step change in the number of space ports being proposed (vertical and horizontal). Sub-orbital flights, whether for the purposes of space tourism or other activity, are an increasing, as are high altitude balloons (e.g. Red Bull Stratos) and other activities that are above commercial aircraft operations. There is no clearly agreed definition of where the boundary of space starts, although general media take a view (Gould and Kane 2017), with a more recent in-depth discussion on this topic also available (McDowell 2018).

Taken overall it can be said that there is a sense that boundaries are being stretched, both in new systems being designed and launched, and also perhaps a slight strain at the regulatory constraints, as identified in a recent case study (Wheeler 2018):

> "Many technology startups in Silicon Valley adopt a business strategy of risk first, where regulatory compliance is often not prioritized until they achieve financial stability. This approach was seen in January when a satellite start-up, Swarm Technologies, launched four cubesats into orbit from India after the domestic regulator, the U.S. Federal Communications Commission (FCC), had denied Swarm's application for a launch licence. The FCC has granted temporary authorization allowing Swarm to reactivate its satellites "for the sole purpose of collecting orbital and tracking data" for six months from Aug. 24, 2018. However, the Swarm satellites still face a continuous ban on commercial use, and the FCC is likely to use the case as an example to deter against future breaches of regulations".

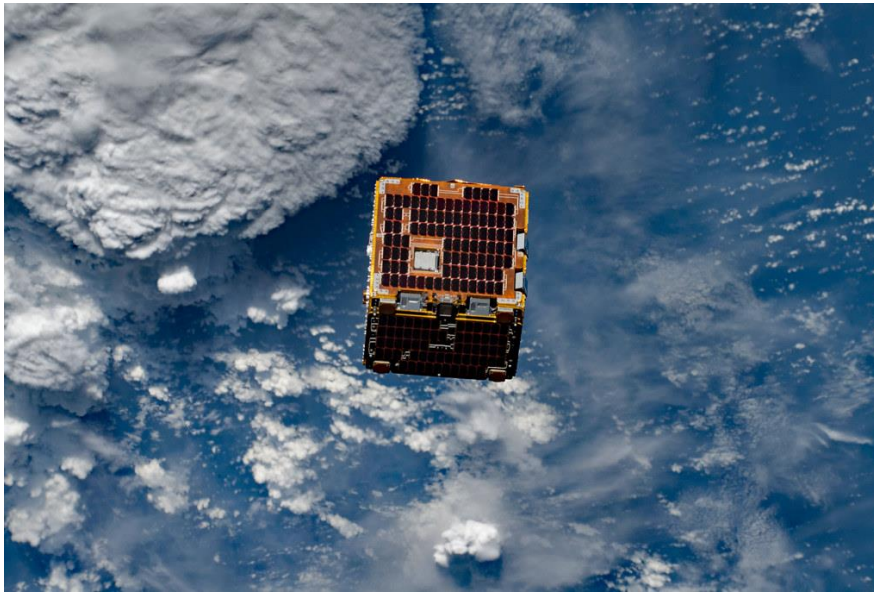


Fig. 3. Cubesat (Creative Commons, 2019b)

8 Legislative changes and a UK perspective

National governments are adapting to these changes through their legislative and licensing regimes, as well as continuing to be (in some cases) key customers of the products and services provided by these companies. For example, the US Congress passed legislation HR2262 otherwise known as the US Commercial Space Launch Competitiveness Act (US 2015) which limits the Federal Aviation Administration's (FAA) scope from issuing standards for commercial spacecraft through to 2023 at the earliest (in contrast to its role with the commercial airline industry) so allowing a learning period without regulatory constraints (Grush 2015). During this time the FAA continue to license launches and re-entering spacecraft (where possible) with a focus on avoiding immediate threats to uninvolved people and property.

Within the FAA regime, astronauts fly on the basis of 'informed consent' (discussed later in this paper). Further streamlining of launch licensing has recently been proposed through a 2019 rule under consultation. It sets out a performance-based regulatory approach, "creating flexibility for operators to meet safety requirements" (FAA 2019a). It is understood that US organisations align with FAA regulations worldwide, irrespective of the country from which they are launching. It is likely that clarification on 'informed consent; will be required within the

framework of the UK's Space Industry Act (SIA), discussed below. The US export regulations ITAR are another example of how US legislation influences UK space activities (Wheeler 2019).

The UK's interest in launch technologies has been in flux for a number of decades, but the practicalities of commercial implementation have only recently been explored in more detail (CAA 2014). Noting the continued growth in commercial space activities, and the established role that the UK plays in the space sector (through its role in the European Space Agency, scientific research, specialist equipment and satellite manufacture and home to many telecommunications and Earth observation companies), the UK government issued into law the Space Industry Act in 2018 (UK 2018). Designed to provide a legislative basis for the authorisation and supervision of launch activity conducted from UK territory, it allows for the first time commercial launches from the UK. Licensing activities are assigned to either the UK Space Agency (UKSA) or the UK Civil Aviation Authority (CAA), depending on whether the spaceport launch is vertical or horizontal (UK 2019). The UK HSE will be the regulatory authority for ground operations.

Balloons and sub-orbital activities will also be licensed, with balloon spaceports under the remit of the CAA. Where multiple types of launches are undertaken from a single location the regulatory authorities will co-ordinate. Some activities (e.g. UK entity procuring overseas launch or operating a satellite overseas) will continue to be licensed under the existing Outer Space Act (UK 1986). Further secondary legislation on "orbital and suborbital licensing regulations" is planned to clarify legal issues and so support further developments of spaceports and other commercial activities (BIS 2018). This ongoing regulatory change is also seen in other countries with geographies favourable to launch, such as New Zealand and Portugal (Holmes et al. 2019).

The private sector drive for growth and diversification of the space sector, combined with the ongoing high level of change, means that the question of safety in space is likely to be considered by a broader and more rapidly changing range of stakeholders than might be present in more mature 'high hazard' sectors, such as nuclear or transportation (e.g. road, rail, aviation). These more mature sectors are of course of public interest, with significantly different risk profiles and legislative frameworks that are managed appropriately (including considering societal perception of risk). However, barring major accidents that end up reframing societal views and legislation (e.g. the Nimrod aircraft disaster) these sectors are not subject to rapid rates of change in either methodology for assessment of safety or the legislative regime within which those assessments are made.

Due to the current location of many launch sites (location driven by operational needs) multi-fatality accidents impacting the public may only test the "reasonably foreseeable" threshold. However, launch failures and launch successes receive high levels of media interest, and this will increase as privately funded

sub-orbital and orbital astronaut programmes come online. Private sector social media and general public engagement programmes can include the narrative of "explorer risk" (i.e. keep trying until you get it right, accidents are part of it). It is also reasonable to assume that societal perception of risk and expectations of safety for participants and public may yet shift, concerns on private launch safety being raised a decade ago (Milstein 2009).

A priority for UK-based projects is that the current UK Health and Safety (H&S) legislative regime covers a number of topic areas, all of which will need to be considered for assessment of launch safety:

- Generic H&S legislation, including HASWA (Health and Safety at Work Act) and MHSWR (Management of Health and Safety at Work Regulations)
- H&S legislation applicable to work environments, including DSEAR (Dangerous Substances and Explosive Atmospheres Regulations) and COSHH (Control of Substances Hazardous to Health)
- H&S legislation for specific processes and complex activities, including COMAH (Control of Major Hazards), LUP (Land Use Planning) and HSC (Hazardous Substance Consent)

These and other applicable legislation are summarised and referenced in (McBeth 2018), this will necessarily be supplemented by the UK Space Industry Act (UK 2018) and secondary legislation, as issued in 2020 and beyond. The delineation between air and space regimes will require clear definition as will further clarification on the roles played by relevant agencies (including UKSA, DfT (Department for Transport), CAA and HSE). This is anticipated to be addressed in secondary legislation. UK legislation covers the employer's obligation to protect people from these risks to their safety in the workplace, and to members of the public who may be put at risk by the work activity, which could include spaceport passengers (assuming no legal clarification required).

For comparison, the US FAA, which licenses US spaceflights, has produced a guidance note on informed consent for crew and space flight participants. This clearly sets out the information to be provided and also a pro forma to be signed by the astronaut to confirm they have received and understood the information (FAA 2017), with the context set as follows:

> "Although [US] Congress has charged the FAA with certificating aircraft, it has not provided the agency the authority to certificate launch or re-entry vehicles…The non-certification statement informs the crew that the FAA's oversight responsibilities…are intended to protect the public and do not extend to the safety of crew or space flight participants…operators are encouraged to explain that the statement means the U.S. government does not ensure the safety of flight crew, government astronauts, or space flight participants (individually and collectively, "occupant").

Further changes to the US legislative framework are anticipated, given President Trump's policy statement on streamlining regulations on commercial use of space. The focus is on moving to simplify the licensing of launch and re-entry

operations by relying on performance-based regulations rather than prescriptive regulations (US Government, 2015). Over 150 submissions were received (FAA 2019b), and the publicly available position of key organisations varied as to the value of the 'streamlining' achieved (Space News 2019). Some comments highlight the aviation industry's record levels of safety through collaboration and information sharing and suggest this as a model to follow, as well as an integrated approach to air traffic control, including collaborative decision-making processes. It remains to be seen how much influence the aviation sector will have on the streamlining of regulations on the commercial use of space.

Whilst the topic of asteroid mining, and ownership of resources, is outside the scope of this paper, recent legislative changes and the rapid growth of companies are starting to influence the legislative context. As noted by Cheney and Newman (2019), "space mining might be the start of the divergence of space law", and the international treaties and national legislation may adapt to that, thus influencing the broader topic of safety in space.

9 Is space safe?

What do these numerous and changing factors mean when it comes to answering the question "is space safe?". How influential is the legal context? What techniques and methodologies can reasonably be implemented by safety and reliability professionals? The analysis presented so far influences how safety in space can be assessed, what the context should be and what methods should be considered to help scope out the picture. The following section discusses the UK perspective and suggests two approaches to assessing risk which might provide a practical basis.

No one risk analysis method can provide a complete picture. This is particularly for emerging sectors such as ensuring safety in space (e.g. commercial launch operators, for manned orbital and sub-orbital craft, and unmanned satellite systems across a range of altitudes). What is evident from the current situation, perhaps most readily characterised as VUCA (volatility, uncertainty, complexity and ambiguity of general conditions and situations), is that a broad and extended socio-technical perspective ('socio-technical-plus'?) will need to be taken in order to establish a baseline. One way to implement this will be to use a PESTLE analysis (Political, Economic, Sociological, Technological, Legal and Environmental) to identify the external factors influencing any decision making process around a particular space activity, both by the private entity and the governmental licensing agencies. Perhaps it might seem unusual to take such a view on what in

essence is a technical activity, but the demographics and topics covered at a recent leading UK industry space conference indicate many of these factors are already part of the narrative (UKSEDS, 2019).

Whether it is through lack of access to suitable resources for project success (from allocation of radio frequencies to import of specialised equipment) to competitive pressures influencing the legislative process across multiple nations, it is not readily evident that an unchanging regime (say over a decade or more) for establishing acceptable level of 'safety in space' can be guaranteed. Changing societal factors (perception or actual risk) may influence government approaches to licensing. Previous spaceflight multi-fatality incidents that shaped US approaches to space safety should continue to influence governmental thinking (Rogers 1986, Gehman 2003).

10 Approaches to safety in space

Whilst demonstrating compliance with some of the UK H&S legislation can be achieved through the use of guidance (e.g. that provided to advise the UK's Health and Safety Executive (HSE) inspectors) the question of how to demonstrate overall ALARP (the UK legal principle of 'As Low As Is Reasonably Practicable'0 may be somewhat challenging. It's reasonable to assume that full and comprehensive quantitative risk assessment may prove to be a significant task, due to the relative lack of prior operational data for various launch systems (particularly horizontal and vertical systems under development) and limited in-orbit quantitative data and modelling on the likelihood of damage to systems that control satellite operations. Re-entry and breakup simulations (from aborted launch through to end of satellite life) can be used to provide some indication on the re-entry survivability and impact risk, but accurate simulations of impact location can be challenging. Clearly a similar reference system approach can be taken to bridge the gap (i.e. applying an existing quantitative risk assessment from a similar system) but only if suitable data for a closely-related system exists.

10.1 Risk assessment

Given the limitations inherent in implementing quantitative risk assessment, qualitative and semi-quantitative methods may be an initial way forward for carrying out a preliminary assessment of risk at all stages of a space mission in order to establish the high level of objective of achieving safety in space. They can be used for unmitigated (i.e. no control measures implemented) and mitigated (i.e.

control measures identified and implemented) risk assessment. Care must be taken to avoid confusion between the two.

Within this context, risk matrices to assess safety in space can be used in a number of different ways, including:

- To rank risks in order to identify priorities for risk reduction;
- Carry out a high level screening of whether hazards are managed by existing control measures;
- Identify where risks can be considered to be broadly acceptable i.e. no further risk reduction measures are required.

A balanced approach will need to be taken between qualitative, semi-quantitative and quantitative analyses.

10.2 Hazard assessment

Hazard identification is typically carried out as part of a workshop activity, and events (realisation of the hazard) can then be assessed using a pre-calibrated risk matrix as part of risk assessment. All reasonably foreseeable hazards should be considered as part of the analysis, including new hazards created from innovation and new technologies (satellites and launchers). Calibration requires clear definitions of numerical bounds for each likelihood category and avoiding terms such as frequent (which are open to a wide range of interpretations if assumptions and definitions not established). Guidance is available on the use of logarithmic scaling on both consequence and likelihood categories ("basically (logarithmic)" as defined in Duijm 2015). In establishing the consequence categories, several approaches can be taken (adapted from Duijm 2015):

1. "The potential event". An event that has the potential to cause damage (worst case); the associated probability is the probability that the event (irrespective of the actual damage) occurs;
2. "The representative or most likely event". The event that leads to the most likely or most representative damage; the associated probability is the probability that the event (irrespective of the actual damage) occurs;
3. "The distribution of possible outcomes event". Events representing a number of alternative, discrete damage outcomes, each in another consequence category; the associated probabilities are the probabilities that each of those damages occur.

If quantitative modelling of scenarios which may cause fatalities, environmental impact or reputational damage cannot readily be made and compared with existing (e.g. aborted launch trajectories or satellite breakup and re-entry paths) it may provide a first pass approach to ranking risks requiring further assessment. Care must be taken to avoid dividing an event into different categories, with slightly different consequences, and then assessing the risk of each individual event, so providing an overly optimistic picture. Some guidance can be given as follows: "risk matrices are not seen as a complete risk analysis tool…risk matrices provide some of the information relevant for decision making" (Thomas et al. 2013, also Flage and Røed 2012). Clearly approach 1 is more conservative than others, using the worst credible event is a useful starting point (ISO 2010). The standard BS31010 also highlights the importance of communication and consultation (as well as monitoring and review) as an integral part of risk management. Given the many stakeholders in the space sector, this activity will likely require notable effort, a 'who's who' roles and responsibilities should be defined as part of establishing the context for the risk assessment.

In order to provide a suitably broad range of inputs, many people who are not risk assessment specialists will need to be involved in the process of hazard identification and use of risk matrices for risk ranking. One example of a lightly calibrated risk matrix is given in a US MIL-STD-882B, showing provides only qualitative definitions of the severity and frequency of accidents for the purpose of risk assessment (FAA 1995). (The qualitative guidance has been updated in US MIL-STD-882E). For this to be used by a broad range of people, all must have the same understanding of frequencies described as "frequent, probably, occasional, remote, improbable" and consequences defined as "catastrophic, critical, marginal, negligible". With a suitably prepared risk matrix, the challenge of removing unconscious bias on the part of the participants remains but the models identified in behavioural economics may provide some guidance (Kahnemann and Tversky 2011). Given the relatively large number of organisations and potential methodologies and existing good practice being blended into a single risk assessment, those preparing and running workshops using expert judgement inputs will need to take care to tease out underlying assumptions and clarify sources of reference data.

10.3 Organisational failings

One common factor underpinning many major multi-fatality accidents is organisational structure, another the notion of an ineffective 'safety culture' under budget and schedule pressures. The STAMP method (Systems-Theoretic Accident Modelling and Processes) is an approach to accident causation, including organizational and social aspects and has been applied to NASA in consideration

of Challenger and Columbia incidents, highlighting structural factors that limited effective intra-organisational co-operation. (Dulac et al. 2007). Organisational failings can also occur across national and international organisational lines, including the medical sector. As shown through a STAMP analysis, stage 1 drug trial led to multi-organ failure in six healthy human volunteers despite having passed regulatory approval. Interactions between a relatively small number of national and international agencies, coupled with the use of standardised analyses and assumptions, led to a drug trial that created a life-threatening 'cytokine' (auto-immune) storm response (Vacher et al. 2018). As reported by Lord Justice Haddon-Cave (Haddon-Cave 2009), complex organisational failures are not new, stating that "the organisational causes of the loss of Nimrod XV230 echo other major accident cases, in particular the loss of the Space Shuttles Challenger and Columbia, and cases such as the Herald of Free Enterprise, the King's Cross Fire, the Marchioness Disaster and BP Texas City". Given the highly complex and interactive nature of organisational structures involved in evaluating safety in space, both at national and international level, a STAMP-based evaluation of either the launch or spacecraft safety licensing process might identify areas requiring further scrutiny, potentially at the licensing and legislative interfaces. Others to be covered will include licensing (and safety cases as appropriate) for operators, designers and other facilities.

10.4 Further work

Consideration of risk assessment scenarios on specific space projects (e.g. vertical and horizontal UK launchers) is beyond the scope of this paper, which is limited to considering factors influencing the current and changing context for establishing safety in space. Generally, the operative regulator is responsible for the ultimate 'Go/No' assessment, where a licence for orbital activity will not be granted if it will jeopardise public health or safety of persons or property. Through the SIA, the discretionary granting of license for sub orbital and UK launch is regulated by two agencies (UKSA and CAA), with HSE regulating ground operations. Information on the application of the SIA is available (UK, 2019). Overall a regulator has the duty to secure public safety through the licensing process.

It is reasonable to assume that an assessment of space safety will have some element of a risk based safety regime, also and that there will be interfaces with other national and international regulatory bodies (e.g. FAA, EASA), potentially supplemented by international fora (e.g. ICAO, COPUOS) taking the lead. The question of how much this approach will need to vary from the current norms of regulation for other sectors will need to be explored. As part of this process it will

be useful to explore how the approach to aircraft system engineering and system safety assessment (e.g. ARP4761 and 4754A (SAE 1996, SAE 2010) will influence space safety. A safety case based approach, where the licensee is specifically required to think about "reasonably foreseeable" scenarios (and measures to manage) could be well suited to anticipating the broad range of potential outcomes for broad (not bounded) potential uses of space. (The ways in which civilian airliners can be used is fairly mature, the same cannot be said for the utilisation of space). These questions will need to be considered alongside further development of the licensing and regulatory regimes, along with the broader question of how such a regime can best influence the further development of a safety culture and mindset across the space industry.

Also not addressed here, but also of increasing importance is the role of space-based and space-broadcast data products in day to day life, GPS and telecommunications being two obvious examples. Those data products will also be used to ensure intra-satellite constellation coordination and communication. The maintenance of data safety, an emerging perspective on the safety of digital information, will be key, both for information used to operate space systems safely, and the fidelity of the information provided by those systems as input to ground-based activities (SCSC, 2019). Safety in space may in the future have a much greater software and data component, eclipsing the focus on maintenance of physical integrity and functionality of launchers and satellites.

Finally, the question of how to disseminate the results of risk assessment needs to be raised, given the importance of creating common understanding across the space sector and an engagement to maintain a live 'risk picture'. Bowties have been used by the UK CAA (and referenced worldwide) to provide a common framework for the results of their 'significant seven' study, which set out the seven top safety risks. That original study provided input material to workshop-based development of bowties (CAA, 2015). As noted by the CAA, '*the expert judgement of the subject matter experts was an important component for decisions related to the control effectiveness ratings*', it is reasonable to assume that factors influencing the implementation of risk matrices (e.g. in section 10.2) will also be present. Bowties (methodology, output and tools) complement other approaches.

11 Conclusions

The increasing growth of the commercial space industry, including the diversification of satellite manufacturers, rapid expansion of space-based operations (including cubesat and smallsat constellations) and market-leading private launch companies (including well known examples such as SpaceX) is driving the need for clarity of obligations and efficiency of licensing. This fast-evolving industry

wave of activity, sometimes referred to as New Space, is also shaping how different nations are approaching their obligations under international treaties and existing national law.

Access to space through launch services and the operation of orbital space resources (e.g. to provide telecommunication and Earth observation services) serve both national interests, civilian and military. Access to space is framed by legislation, regulatory supervision and guidance (e.g. standards) and custom and practice. It remains to be seen how this greatly increased stakeholder community, broadening and diversifying of disciplines and sectors at the table and the greater engagement of the public (and their views on acceptable levels of safety and sustainability as applied to space) will in combination end up shaping how safety in space is assessed and managed, preserving access to this valuable resource.

Acknowledgments The support of the Safety-Critical Systems Club, as well as the valuable discussions with the speakers at the Safety and Reliability Society's 'Safety in Space' event in March 2019 (held at the Institution of Civil Engineers through the Hazards Forum), is much appreciated. Reviewer contributions and guidance from Thomas Cheney, Rob Garner, Ron Macbeth, Christopher Newman, Mike Parsons, Heidi Thiemann and Simon Whiteley are gratefully acknowledged.

Disclaimers The views expressed in this paper are personal and do not reflect the Safety and Reliability Society's position on this or any other matters. The material presented in this paper should not be used to inform a legally binding or commercial position and is presented for information only.

References

BIS (2018) UK space puts on a show. British Interplanetary Society. Spaceflight 60:10

CAA (2011) Safety regulation group 'significant seven' task force reports. CAA paper 2011/03

CAA (2014) Title UK Government Review of commercial spaceplane certification and operations. Civil Aviation Authority technical report CAP 1189.

CAA (2015) How were the bowties created?. https://www.caa.co.uk/Safety-initiatives-and-resources/Working-with-industry/Bowtie/Bowtie-templates/How-were-the-bowtie-templates-created-/. Accessed 30 September 2019

Celestrak (2019) Satellite times: More FAQs. https://celestrak.com/columns/v04n05/. Accessed 30 September 2019

Cheney T and Newman C J (2018) Managing the resource revolution: space law in the new space age. Frontiers of space risk: natural cosmic hazards & societal challenges, Eds. Wilman R J and Newman C J, CRC Press

Creative Commons (2019a) SpaceX Dragon propulsive descent landing test. Licensed under CC PDM 1.0. https://ccsearch.creativecommons.org/photos/f51509f1-24c4-479b-89e3-a9cd4728eecc. Accessed 30 September 2019

Creative Commons (2019b) Over Seas and Clouds, processed image. Licensed under CC BY-NC 2.0. https://ccsearch.creativecommons.org/photos/1e97887b-faf8-4d7e-a0f3-ab43d7f0fd00. Accessed 30 September 2019

Dempsey PS and Manoli M (2017) Suborbital flights and delimitation of air space vis-à-vis outer space: functionalism, spatialism and state sovereignty. Submission to the United Nations Office of Outer Space Affairs (UN-OOSA). OOSA/2017/19 12 September 2017.

http://iaass.space-safety.org/wp-content/uploads/sites/24/2018/03/Definition-Delimitation-Air-Outer-PSDMM-16Jan2018.pdf. Accessed 30 September 2019

Duijm N (2015) Recommendations on the use and design of risk matrices. Safety Science 76, 21–3

Dulac N, Owens B, Leveson N, Barrett B, Carroll J, Cutcher-Gershenfeld J, Friedenthal S, Laracy J, Sussman J (2007) Demonstration of a new dynamic approach to risk analysis for NASA's Constellation Program. MIT. http://sunnyday.mit.edu/ESMD-Final-Report.pdf. Accessed 30 September

ECSL (2019) 2019 ESA-ECSL (European Centre for Space Law). Workshop on Space Debris. http://www.esa.int/About_Us/ECSL_European_Centre_for_Space_Law/2019_ESA-ECSL_Workshop_on_Space_Debris. Accessed 22 Nov 2019

ESA (2019a) A full summary of ESA's ongoing and planned activities is available on the blog " Road to ESA's Council at Ministerial Level". http://blogs.esa.int/space19plus/. Accessed 22 November 2019

ESA (2019b) SEISOP Space Environment Information System for Operations. http://www.esa.int/Enabling_Support/Operations/SEISOP_br_Space_Environment_Information_System_for_Operations. Accessed 30 September 2019

FAA (1995) Hazard analysis of commercial space transportation. US FAA. https://www.faa.gov/about/office_org/headquarters_offices/ast/licenses_permits/media/hazard.pdf. Accessed 30 September 2019

FAA (2017) Guidance on informing crew and space flight participants of risk. FAA. https://www.faa.gov/about/office_org/headquarters_offices/ast/regulations/media/Guidance_on_Informing_Crew_and_Space_Flight_Participants_of_Risk.pdf. Accessed 30 September 2019

FAA (2019a) Office of Commercial Space Transportation regulations. https://www.faa.gov/about/office_org/headquarters_offices/ast/regulations/. Accessed 30 September 2019

FAA (2019b) Streamlined launch and reentry licensing requirements; notice of availability and extension of comment period. https://www.federalregister.gov/documents/2019/07/22/2019-15465/streamlined-launch-and-reentry-licensing-requirements-notice-of-availability-and-extension-of. Accessed 30 September 2019

Flage A and Røed W (2012) A reflection on some practices in the use of risk matrices. Proceedings of the 2012 ESREL conference. Helsinki, Finland

Gehman H (2003). Columbia accident investigation report. UK Government Accounting Office. https://www.nasa.gov/columbia/home/CAIB_Vol1.html. Accessed 30 September 2019

Gould S and Kane S (2017) Here's where outers space actually begins. Business Insider. https://www.businessinsider.com/where-does-space-begin-2016-7?r=US&IR=T. Accessed 30 September 2019

Grady M (2017) Private companies are launching a new space race – here's what to expect. The Conversation. https://theconversation.com/private-companies-are-launching-a-new-space-race-heres-what-to-expect-80697. Accessed 30 September 2019

Greason J and Bennett J C (2019) The economics of space: an industry ready to launch. Reason Foundation. https://reason.org/policy-study/the-economics-of-space/. Accessed 30 September 2019

Grush L (2015) Private space companies avoid FAA oversight again, with Congress' blessing. The Verge. https://www.theverge.com/2015/11/16/9744298/private-space-government-regulation-spacex-asteroid-mining. Accessed 30 September 2019

Haddon-Cave C (2009) An independent review into the broader issues surrounding the loss of the RAF Nimrod MR2 aircraft XV230 in Afghanistan in 2006. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/229037/1025.pdf. Accessed 30 September 2019

Henderson R (2019). Virgin Galactic hits $2.3bn valuation in public launch. Financial Times. https://www.ft.com/content/8e86eb5a-f992-11e9-a354-36acbbb0d9b6. Accessed 22 Nov 2019

Holmes M, Cocco M and Mendonça H C (2019) Analysis on the Portuguese Space Act. Via Satellite Digital. http://interactive.satellitetoday.com/via/february-2019/analysis-on-the-portuguese-space-act/. Accessed 30 September 2019

ICAO (2019) Space transportation. https://www4.icao.int/space. Accessed 30 September 2019

IIASL (2019) Building blocks for the development of an international framework on space resource activities. The Hague international space resources governance working group. https://www.universiteitleiden.nl/en/law/institute-of-public-law/institute-of-air-space-law/the-hague-space-resources-governance-working-group. Downloaded 2 December 2019

ISO (2010) Risk management. Risk assessment techniques. International Organization for Standardization (ISO). BS ISO 31010: 2010

ISO (2019a) Space systems — Space debris mitigation requirements. International Organization for Standardization. ISO 24113:2019

ISO (2019b) Standards by ISO/TC 20/SC 14 “Space systems and operations”. https://www.iso.org/committee/46614/x/catalogue/. Accessed 30 September 2019

ITU (2019). Space Services Department (SSD). https://www.itu.int/en/ITU-R/space/Pages/default.aspx. Accessed 30 September 2019

Kahneman D and Tversky A (2011) Thinking fast and slow. Farrar, Straus and Giroux. 978-0374275631

McBeth R (2018) Spaceports: keeping people safe. UK Health and Safety Laboratory. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/780509/Spaceports_keeping_people_safe.pdf. Accessed 30 September 2019

McDowell J C (2018) The edge of space: revisiting the Karman Line. Acta Astronautica 151:668-677

Milstein M (2009) Is it safe?. Air and Space Magazine. https://www.airspacemag.com/space/is-it-safe-57287715/?all . Accessed 30 September 2019

Newman C J (2018) Rediscovering UK sovereign launch capability. Room. Journal of Asgardia 16(2):88-92

Newman C J and Williamson M (2018) Space sustainability: Reframing the debate. Space Policy.46: 30-37

New Space Index (2019) Concise original overview of commercial satellite constellations, small satellite rocket launchers and NewSpace funding options. https://www.newspace.im/. Accessed 2 December 2019

Popova R and Schaus V (2018) The legal framework for space debris remediation as a tool for sustainability in outer space. Aerospace 5:55

Racelis D and Joerger M (2018) High-integrity TLE models for MEO and GEO satellites. Proceedings of the 2018 AIAA Space and Astronautics Forum. AIAA 2018-5241

Rogers W P (1986) Report of the Presidential Commission on the Space Shuttle Challenger Accident. US Government Accounting Office. https://spaceflight.nasa.gov/outreach/SignificantIncidents/assets/rogers_commission_report.pdf. Accessed 30 September 2019

SAE (1996) ARP 4761: Guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment

SAE (2010) ARP 4754A: Guidelines for development of civil aircraft and systems

SCSC (2019) Data safety guidance version 3.1. https://scsc.uk/r127D:2. Accessed 30 September 2019

Secure World Foundation, Handbook for new actors in space. https://swfound.org/handbook/. Accessed 30 September 2019

Stephen C (2019) Egypt's gas gold rush. Petroleum Economics. https://www.petroleum-economist.com/articles/upstream/exploration-production/2019/egypts-gas-gold-rush. Accessed 30 September 2019

Thomas P, Bratvold R and Bickel J (2013) The risk of using risk matrices. SPE Economics & Management. 6. 10.2118/166269-MS.

UK Government (1986) Outer Space Act 1986. UK Public General Act. http://www.legislation.gov.uk/ukpga/1986/38/contents. Accessed 30 September 2019

UK Government (2018) UK General Public Act: Space Industry Act 2018. http://www.legislation.gov.uk/ukpga/2018/5/contents/enacted. Accessed 30 September 2019

UK Government (2019) Applying for a future licence under the Space Industry Act. https://www.gov.uk/guidance/applying-for-a-future-licence-under-the-space-industry-act. Accessed 30 September 2019

UKSEDS (2019) National Student Space Conference 2019. https://ukseds.org/aurora/?p=programme&event_id=46. Accessed 30 September 2019

UN (1962) Declaration of Legal Principles Governing the Activities of States in the Exploration and Use of Space. https://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties/travaux-preparatoires/declaration-of-legal-principles.html. Accessed 30 September 2019

UN (1967) Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space, including the Moon and Other Celestial Bodies (Outer Space Treaty). http://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties/introouterspacetreaty.html. Accessed 30 September 2019

UN (1971) Convention on International Liability for Damage Caused by Space Objects (Liability Convention). http://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties/introliability-convention.html. Accessed 30 September 2019

UN (1974) Convention on Registration of Objects Launched into Outer Space (Registration Convention). http://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties/introregistration-convention.html. Accessed 30 September 2019

UN (2010) UN Space debris mitigation guidelines of the Committee for Peaceful Uses of Outer Space (COPUOS). http://www.unoosa.org/pdf/publications/st_space_49E.pdf. Accessed 30 September 2019

UN (2019a). Status of international agreements relating to activities in outer space. http://www.unoosa.org/documents/pdf/spacelaw/treatystatus/AC105_C2_2019_CRP03E.pdf. Accessed 30 September 2019

UN (2019b) Guidelines for the long-term sustainability of outer space activities. https://undocs.org/pdf?symbol=en/A/AC.105/C.1/L.366. Accessed 30 September 2019

UN (2019c) United Nations Sustainable Development Goals. https://www.un.org/sustainabledevelopment/sustainable-development-goals/. Accessed 30 September 2019

US Department of Defense (2012) Standard practice system safety. MIL-STD-882E.

US Government (2015) US Commercial Space Launch Competitiveness Act. HR2262. https://www.congress.gov/bill/114th-congress/house-bill/2262. Accessed 30 September 2019

US Government (2018) Space policy directive-2, streamlining regulations on commercial use of space. https://www.whitehouse.gov/presidential-actions/space-policy-directive-2-streamlining-regulations-commercial-use-space/. Accessed 30 September 2019

Vacher A, Salvo F, Pollina M, Whiteley S, Daridan M and Edwards B (2018) The contribution of causal analysis using systems theory to the analysis and prevention of serious incidents in clinical research involving healthcare products: preliminary results of its application to the TGN1412 first-in-human clinical trial. Proceedings of the 2018 International cross-industry safety conference (ICSC) and European STAMP workshop and conference (ESWC). Amsterdam

Varghese S (2018) . One small step for private companies: how the future of space travel is being redefined. New Statesman. https://www.newstatesman.com/science-

tech/space/2018/01/one-small-step-private-companies-how-future-space-travel-being-re-defined. Accessed 30 September 2019
Weedon B (2010) Dealing with Galaxy 15: Zombiesats and on-orbit servicing. The Space Review. https://thespacereview.com/article/1634/1. Accessed 30 September 2019
Weinzierl M (2019) Space, the final economic frontier. J Econ. Perspectives 32(2):173–192
Wheeler J (2018) Regulation: an enabler, not an enemy for emerging space companies. Via Satellite Digital. http://interactive.satellitetoday.com/via/november-2018/regulation-an-enabler-not-an-enemy-for-emerging-space-companies/. Accessed 30 September 2019
Wheeler J (2019). When ITAR Met Brexit. Via Satellite Digital. http://interactive.satellitetoday.com/via/february-2019/when-itar-met-brexit/. Accessed 30 September 2019
Wilson R (2018) A look at venture capital investment in the space industry in 2018. https://martinwilson.me/venture-capital-investment-space-industry-2018/. Accessed 30 September 2019
Wikipedia (2019a) Image of Ariane 6 launcher. Licensed under Creative Commons Attribution 3.0. https://commons.wikimedia.org/wiki/File:Ariane_62_and_64.svg. Accessed 30 September 2019
Wikipedia (2019b). List of private spaceflight companies (non-governmental entities). https://en.wikipedia.org/wiki/List_of_private_spaceflight_companies. Accessed 30 September 2019
Williams A (2017) Space — the final frontier for investors. Financial Times. https://www.ft.com/content/05f24014-07e1-11e7-97d1-5e720a26771b. Accessed 30 September 2019

Making Modular Assurance Cases Work Using Structured Assurance Case Metamodel (SACM)

Jane Fenn BAE Systems

Yvonne Oakshott Leonardo Helicopters

Richard Hawkins University of York

Ran Wei University of York

Abstract *It has long been postulated that the use of modularity in assurance cases has the potential to bring extensive benefits through its ability to manage technical and organisational complexity, provide a scalable solution and facilitate re-use in future large complex systems. Previous work such as that undertaken by the Industrial Avionics Working Group (IAWG), has shown how a modular assurance case approach could be adopted for real systems, however despite this, its uptake by industry has been slow. The Object Management Group, (OMG), recently published Version 2.1 of their standard for assurance cases, called the Structured Assurance Case Metamodel (SACM). By providing a standardised metamodel for assurance cases, SACM also supports the integration and interchange of different assurance artifacts and controlled terminology. This makes SACM the ideal mechanism to support modular assurance cases through the development of assurance case packages, interfaces and integration bindings. In this paper we describe the state of the art for modular assurance cases through an example from the IAWG project, expressed using the modular GSN notation. We show how this example could be developed using SACM. We go on to discuss the key challenges that are preventing the wide-spread adoption of modular assurance cases and discuss the extent to which SACM may be able to help address these challenges*

1 Introduction

The Assurance Case Working Group of the Safety Critical Systems Club is currently developing guidance material which is expected to be published in 2020. It includes a guidance paper on modularity as a way of dealing with large and complex assurance arguments, a summary of which is provided in section 2. Fragments of the modular software safety case developed as a case study by the Industrial Avionics Working Group, (IAWG), (Fenn et. al., 2007) are introduced

© BAE Systems (Operations) Ltd, Leonardo MW Ltd & University of York 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

in section 3, which utilised Goal Structuring Notation, (GSN), (SCSC-141B 2018). We discuss some of the notational challenges identified during the IAWG case study and potential reasons why the approach has not been more broadly adopted. The GSN fragments are then re-expressed using the SACM notation, (OMG - Structured Assurance Case Metamodel), and we discuss the advantages of SACM in addressing some of this issues arising during the IAWG case study.

2 Modular Assurance Arguments

Large and complex assurance arguments can be difficult to follow and comprehend. A reviewer may find it difficult to determine whether all the necessary aspects have been considered and whether their consideration is sufficient within the context of use of the system or service. This is particularly apparent where multiple teams, within the same organisation or in external organisations, collaborate to generate an assurance argument. In this scenario, the relationships between independently developed aspects of the assurance argument can be left implicit and unstructured, leading to an incoherent overall argument where it is not easy to determine whether the interfaces between elements of the argument constitute complete coverage of the scope, or whether duplication or omission has occurred.

2.1 Rationale for Modular Assurance Arguments

Modularity supports two key types of structuring, basic structuring and structuring for compositional arguments, which may be used individually, or in combination.

Basic structuring allows the assurance argument author to focus the reader's attention on the intended part of the argument structure. For example, one module of an argument might deal with the failure reporting mechanism for an in-service system, or another may present an argument about the configuration management system in place. Similarly, the author may wish the reader to focus on the main thrust of an argument whilst handling potentially distracting 'side' arguments in a separate argument module. This is commonly referred to as 'separation of concerns'.

Many systems are developed by integrating new or pre-existing components, software modules/applications or similar, which are sourced from disparate teams. Those teams could be within or external to the organisation that is developing the overall assurance argument. When an overall system or service is to be composed by integrating separate parts, which may have been developed by third

parties, it would be advantageous if the assurance argument could be similarly composed from elements of assurance arguments produced for those parts. The independence introduced also promotes and supports re-use of the separate parts.

A necessary consequence of 'modularising' an assurance argument is that interfaces in the argument will be created between the modules. These interfaces need to record any needs, dependencies or shared assumptions between the linked modules. The assurance required from the overall argument will dictate the proportional response to the rigour of definition of the interface.

Interfaces should consider not only functional behaviour that needs to be assured, or depended upon between modules, but also should cover other explicit or, often, implicit information. A check of compatible assumptions and presumptions must be made at any interface when composing the overall argument.

It is very easy to overlook quite simple interface compatibility issues, such as units of measure, "endiansim", height reference datum, cited assurance standard compliance, etc. Being explicit about interfaces encourages and facilitates checking of these issues.

2.2 Guidance on Modularising Assurance Arguments

Use of modularisation appears to be a simple and logical decision, however, there are a number of areas that strongly impact on the effectiveness of the approach. As with all aspects of assurance arguments, the level of effort expended in determining and optimising the argument structure should be proportionate, not only to the level of risk presented by the system, but also, in this instance, to the potential opportunity for re-use.

The structure of an assurance argument should be considered early in the design lifecycle, and, for complex systems, is ideally considered as part of the selection criteria for deciding upon the design architecture, such that mutual optimisation between design architecture and assurance argument architecture can take place. A range of techniques may be used to optimise a system architecture, such as the SEI Architecture Trade-Off Analysis Method (SEI 2000), which can be extended to consider assurance argument architecture.

The ideas of 'separation of concerns' described earlier could also be achieved by a hierarchical structure, which may hide information that is not required by the flow of the current argument. When creating an argument composed from other modules of argument, developed in isolation, it is necessary to consider whether the product, in operation, also operates in an isolated way. If, say, the behaviour of one component can interfere with the operation of another component, it may not be valid to simply present the assurance arguments made independently. For example, if one component can utilise resource that would then be unavailable to

another component, that is a form of interference that would compromise the associated assurance argument. As well as being clear on resource requirements at the interface, in this example, a high level argument and evidence would typically be required about how resource allocation in the system would be policed and how each component is intended to behave when it has insufficient resource available. Other common non-interference arguments that may need to be made, include interference from modules of different safety assurance or security accreditation levels. This might be addressed by partitioning arguments from the operating system, for example.

End-to-end system performance measures are typically difficult to handle in a modular way. Timing properties could be handled by budgeting for each individual component of a software system, for example, but is likely to yield a pessimistic outcome. Probabilistic timing analysis may be sufficient for low assurance systems or higher assurance systems with soft deadlines, but will be challenging where hard timing deadlines exist. This needs to be considered as early as possible in the design lifecycle.

3 State of Practice in Modular Assurance Argumentation

In 2006/7, the Industrial Avionics Working Group (IAWG), undertook an industrial case study, developing a modular software safety case for a complex avionic system, publishing results at a number of conferences, e.g. (Fenn et. al., 2007). The continuing IAWG activities culminated in the publication of a public process, the IAWG Modular Software Safety Case Process (IAWG 2012). Subsequently, it has been difficult to track the adoption of this approach. Although there has been some up-take in the Defence industry, this is a domain where it is unusual to publish material and hence meaningfully track adoption. Modular GSN has certainly been presented at, and has appeared in presentations by those working on Safety Case approaches and authoring standards for the European automotive industries, (Birch, J. 2019).

Without access to more up to date examples of usage, fragments of the IAWG case study are presented below to represent extant practice. The IAWG modular assurance case activities were instigated to address the issues arising from the introduction of Advanced Avionic Architectures (AAA) and the resultant need to handle the introduction and the subsequent updates of software safety cases for such large and complex systems. Although targeted at software safety cases much of the process is applicable at system level and can also be applied more generally to assurance cases. Indeed, the PETER programme, (PETER), is currently recruiting for PhD students to investigate the development of modular Electro-Magnetic Interference Safety Cases.

Challenges with use of the process and notation were identified during the IAWG programme, however, and recommendations were also made to improve the process, subject to appropriate tool support.

3.1 IAWG Case Study

During 2006/7, IAWG developed a modular software safety case for a military Integrated Modular Avionics System, (IMS), as defined by Defence Standard 00-74. Such systems contain a 'three-layer stack', as shown in figure 1, comprising an Application Layer, an Operating System Layer (OSL) and a Module Support Layer (MSL). A 'Run-Time Blueprint' configures the connections in the system to reduce the impact of change of a layer of the system.

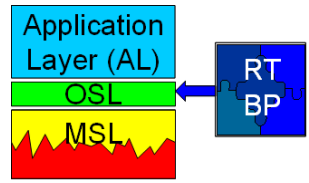


Fig. 1. ASAAC IMS Architecture

As part of that study, IAWG identified an optimised architecture for the assurance argument for the system, which is represented, in Figure 3, in modular Goal Structuring Notation (GSN) which is summarised in Figure 2 for those unfamiliar. More detail is available from the GSN Standard (SCSC-141B 2018).

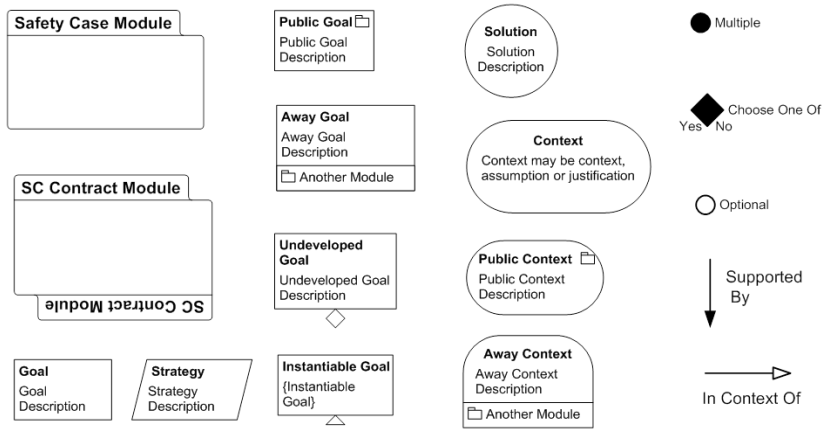


Fig. 2. GSN Notation Summary

The assurance case architecture for the case study is shown in figure 3. Arguments about 'real world' design elements are present, such as each of the applications in the Application Layer, but also integration arguments were added to limit the impact of change, to argue over integration concerns such as the interference between parts and to address system-wide issues such as end-to-end timing. Individual applications can be changed without impacting on the arguments made about other applications or about the OSL and MSL. The 'Architecture Integration' argument integrates those from the OSL and MSL which was expected to facilitate changes to either element whilst minimising impact on the application layer.

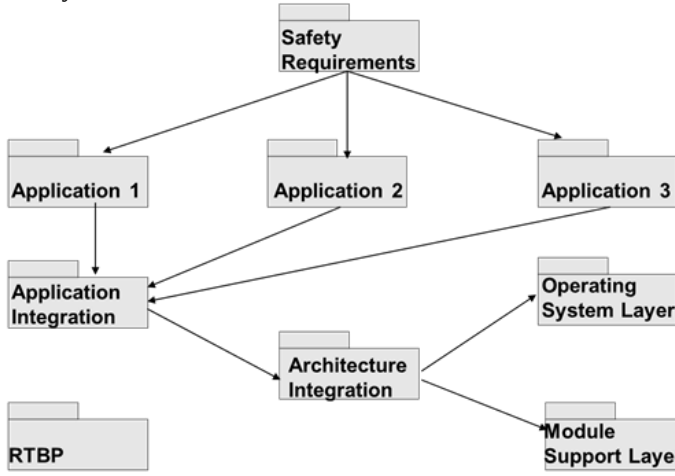


Fig. 3. Assurance Case Architecture for IMS

3.2 Example Argument Fragment from the IAWG Case Study

The specific fragment of the argument that has been selected to analyse in this paper is focussed around a specific type of partitioning that is used in the OSL to prevent interference between applications, 'Temporal Partitioning'. The 'Application Integration' argument identifies a requirement for non-interference between applications which is supported by the 'Architecture Integration' module. An outline of part of the argument thread within 'Architecture Integration' module is shown in figure 4.

The argument has a goal which is supported by a safety case contract. The safety case contract construct provides isolation from change when a goal in one module requires support from argument and evidence in another module, as only the contract needs to be updated in response to the change, rather than the safety case modules themselves. The safety case contract between 'Architecture Integration' argument module and the 'Operating System' argument module is provided in figure 5.

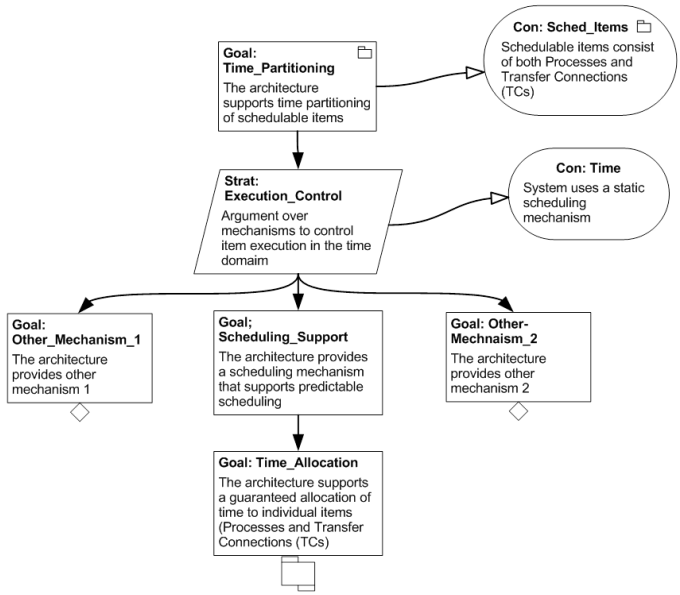


Fig. 4. 'Architecture Integration' argument fragment

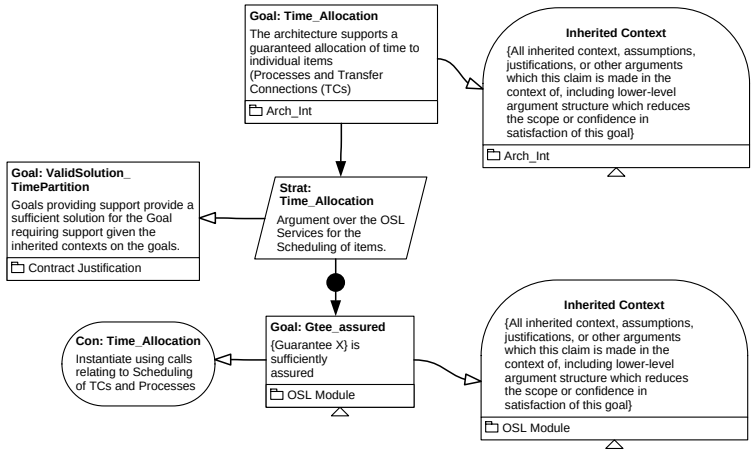


Fig. 5. 'Architecture Integration' to 'Operating System Layer' Safety Case Contract

The notational form used for 'Goal: Gtee assured' in Figure 5 should be noted. This is a recommendation of the IAWG work, and was described as a 'template'. Where an argument structure is to be duplicated across many properties of interest, for notational expediency, IAWG utilised the pre-existing pattern notation in GSN to indicate the replicated fragment of GSN, e.g. 'Guarantee {X}', where the instantiation data for X is provided in the context attached to the goal. This allows the reader to see a clearer, simpler GSN structure, to determine the adequacy of the claim, but the expectation was that, with tool support, this could be extended by asking the tool to create the significantly larger, full GSN structure, by automatically instantiating the repeated fragment with the provided instantiation data.

These fragments provide a full range of modular GSN notational types and were selected to provide a credible example for re-expression.

4 Challenges to Adoption of Modular Assurance Cases

As part of the evaluation of the IAWG study, a number of issues were discussed around potential barriers to adoption of the IMSSC process, with proposals made as future research and/or implementation requirements. Firstly, additional or amended GSN notation was proposed, some of which was incorporated into issue 2 of the GSN standard, or is scheduled to be incorporated at issue 3. Tool support for GSN was also seen as a barrier, and recommendations were made on functionality such as dual representation of repeated Safety Case fragments created

from 'Templates', with instantiation from a provided data tables, with provisioning for optional use of the pattern and the data or the full GSN representation. Interpretation of both formats by the tool is required to handle the argument, check validity of input and to perform change impact analysis.

Anecdotally, a number of issues have also been reported to the authors regarding the reasons for lack of perceived up-take. A key theme that the authors need to reflect upon when presenting modular safety cases is the perception of complexity. Although modularity, in general, is a vehicle to deal with system complexity, modular approaches can, in themselves, appear complex. Many systems in use today are characterised by attempts to minimise complexity for high assurance functions, for which a way to handle complexity is not required and counter-intuitive.

There are undoubtedly costs associated with handling the interfaces between modules and these need to be outweighed by the benefits. Practitioners typically have established localised procedures for handing off safety related interfaces between parties which, it is perceived, would receive no material benefit from recording in a more rigorous way, as necessitated through modular safety cases. This may be a valid position for simpler systems and established programmes, but creeping system complexity is likely to lead to a need for more rigour in the future.

Similarly, the authors have been advised that many industries deal predominantly with change cases rather than having the luxury of working from a 'greenfield site'. This can limit the ability to create an optimised assurance case architecture, which provides the benefits of ease of change impact assessment and hence change containment.

Other issues encountered by the authors when fostering the adoption of modular assurance cases are:

- modular assurance cases are generally not perceived as accepted practice, accompanied by lack of awareness of state of the art in this area and little or no access to what has already been achieved
- lack of evidential information regarding the benefits, leading to a low acceptance levels
- for those who have seriously considered adoption, the general lack of tool support for modular assurance cases and in particular lack of support for handling interfaces and integration, results in manual checking and resolution

The authors have been pleased to hear from colleagues who recognise potential benefits of modular assurance cases in the future as they begin to imagine how to deal with design techniques such as Model-Based Systems Engineering and Agile processes. Particularly any technique that can facilitate evaluation of change impact is welcomed. The authors agree that modular safety cases do provide this

advantage, however, until the safety engineering 'modelling', by which we mean the safety case modules, or at least safety case requirements and dependencies, can be modelled in such a way as to be meaningfully integrated within the design tool, we expect that the benefits will be quickly eroded under cases of rapid change, such as one might expect with agile processes. A more recent development that has the potential to address some of the issues raised above is the development of the SACM. In the next section we discuss SACM and how it applies to modular assurance cases.

5 SACM

Over ten years ago the Object Management Group (OMG) established the Systems Assurance Task Force to improve standardisation and interoperability of assurance cases. The task force brought together the developers of existing assurance case approaches (e.g. GSN and CAE) to develop an agreed standard for the interchange model of assurance cases. This work resulted in the specification of the Structured Assurance Case Metamodel (SACM) (OMG 2019). SACM provides a formal metamodel, and a means of exchange between different assurance case notations (e.g. translation of CAE arguments to GSN). In addition, SACM is more powerful than existing approaches in a number of ways, such as providing more fine-grained modularity, support for the use of a controlled vocabulary, and improved traceability between arguments and evidence, see (Wei et al, 2019). The SACM notation is summarised in Figure 6.

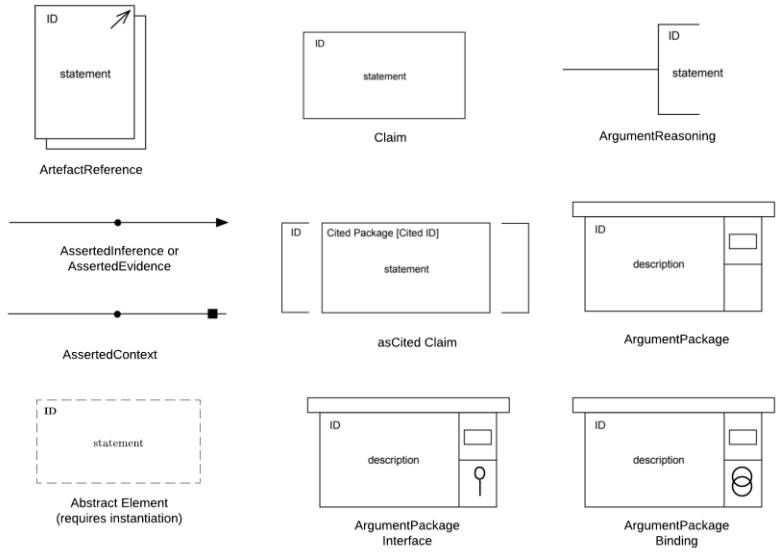


Fig. 6. SACM notation summary

SACM consists of a number of different 'packages' that can be used to create assurance case models. An Assurance Case package can contain a number of Argumentation packages, Terminology packages and Artifact packages. The Argumentation package captures the core structure of the assurance argument; The Artifact package captures the concepts used in providing evidence to support the arguments; The Terminology package captures the concepts used in expressing the arguments. The relationships between the packages that make up an assurance case are captured in the SACM model.

SACM also provides mechanisms that are particularly useful for creating modular assurance cases. Assurance case package Interfaces can be defined in order to identify elements of an assurance case package that are exposed externally. Defined interfaces can then be related using an assurance case package Binding in order to create an overall assurance case. Figure 7 illustrates how SACM interfaces and bindings could be used to structure a modular assurance case in SACM.

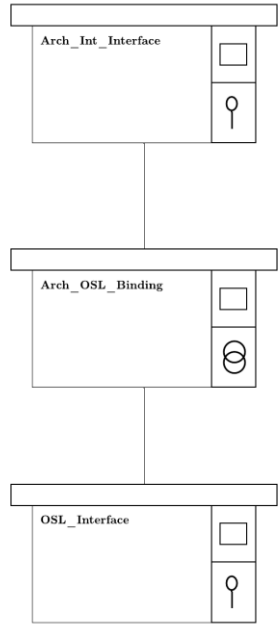


Fig. 7. Using an SACM package Binding to link the interfaces of the Arch_Int and OSL argument packages

Figures 8 and 9 show the example argument fragments from figures 4 and 5 as they would be represented using the SACM visual notation. Figure 8 shows a 'needsSupport' claim (Claim: Time_Allocation) within this argument. To provide support for this claim from another argument package this claim will be included as part of an interface for the Architecture Integration package. Figure 9 shows this claim being referenced as an 'asCited' claim in the Interface of Arch_Int. This is being supported by another 'asCited' claim in the Interface of OSL in order to providing a binding between the two packages.

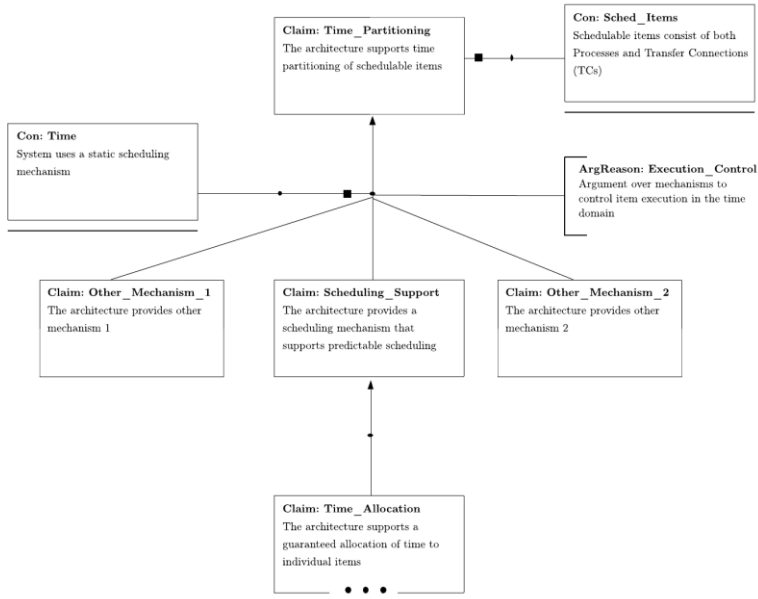


Fig. 8. 'Architecture Integration' argument package

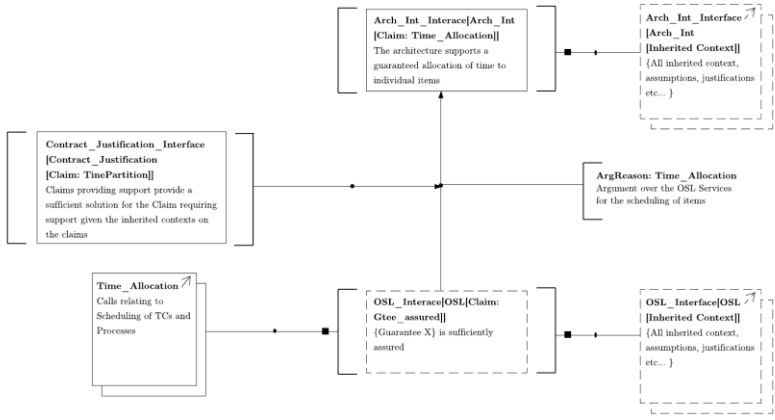


Fig. 9. 'Architecture Integration' to 'Operating System Layer' Assurance Case Package Binding

5.1 Using SACM for Modular Assurance Cases

Clearly, the SACM visual representation seen in Figures 8 and 9 differs from GSN as represented in Figures 4 and 5, but the authors found it reasonably easy to transfer between the two visualisations and expect familiarity to be quickly gained by other existing GSN users. In reality, visual presentation is not the area where SACM provides significant advantages.

The predominant advantage of SACM is the potential for integration with other tools. Although some GSN tools have already provisioned to link into design artifacts, their integration is limited, typically tracing to documents and reports within artifact document repositories, such as in (Denney, et al, 2012). SACM facilitates much deeper integration through linking directly to elements in other models, such as linking to a specific gate within a fault tree analysis model. This facilitates significantly improved traceability into the design and evidence artifacts but also provides for the possibility of, for example, the assurance case model interrogating the design models using a defined relationship or transform. This brings closer the potential for real-time re-evaluation of assurance cases.

SACM also provides the facility, through the use of the Terminology Package, to enable the equivalent of the IAWG-requested GSN template instantiation of 'pattern language' elements. From the examples in Figures 5 and 9, the instantiation of {Guarantee X} is related to the requirements of the OSL for the scheduling of processes and Transfer Connections. For the GSN example the instantiation data is manually incorporated in the associated context. Using SACM, 'Guarantee X' could be explicitly defined as a term in the terminology package and then related to the relevant elements of other models that provide the instantiation data. Implementation constraints can be defined within SACM to specify the rules for instantiation. An SACM tool could then automatically instantiate all the instances of 'Guarantee X'. The reverse would also be true, which would be particularly useful when assessing the impact of change, through-life, e.g. querying what were all instances of the generic 'Guarantee X' in the assurance case model. These automated and tool-assisted options represent a significant improvement on what is currently a largely manually evaluated assessment of the impact of change.

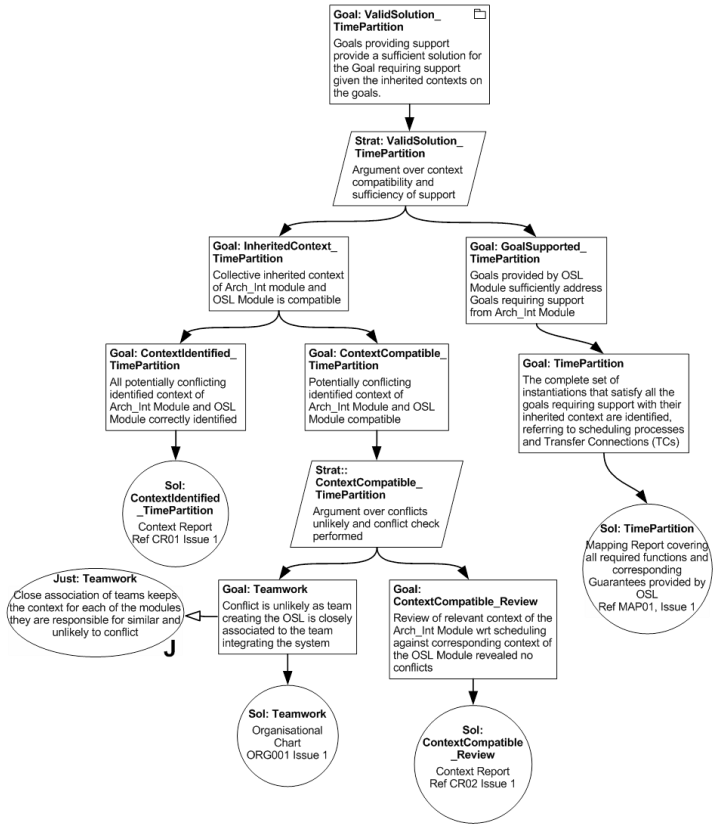


Fig. 10. 'Architecture Integration' to 'Operating System Layer' Safety Case Contract Justification

Figure 10 shows the Justification argument that is referenced from the safety case contract in Figure 5. This includes the solution TimePartition, which refers to a mapping report that requires manual input. SACM can be used to generate a solution by automatically relating relevant model elements, rather than having to manually generate a report. The mapping report referred to by solution TimePartition covers the functions required by Arch_Int and the corresponding 'Guarantees' provide by the OSL. This could be automatically generated by transitioning between a formally specified model that covers the functional requirements for the integrated Architecture and a formally specified model that covers the functional requirements for each of the services provided by the OSL that relate to each of the 'Guaranteed' behaviours.

An additional benefit is that the rather crude mechanism of public and private elements, in GSN has been superseded in SACM. In GSN, all elements that are

marked public are visible to all users of that interface. When an interface is complex, containing many elements and with many customers or providers using that interface, this visibility may not be appropriate. In SACM, argumentation elements that are required to be shared are not marked as public, but instead are included within an argument package interface. SACM supports the creation of multiple interface packages, each of which may contain different elements. This allows different interfaces to be shared with different stakeholders who may require different visibility.

6 Conclusion

In this paper we identified the rationale for the use of modular assurance arguments and provided guidance on their construction. The state of practice in modular assurance argumentation is discussed and a modular software safety case previously developed for a complex avionic system is presented as a representation of current practice.

The challenges to adopting modular assurance cases were identified using experience and anecdotal evidence. These can be summarised as:

- The inadequacy of tool support for modularity in assurance cases
- The perception of complexity associated with modularity
- Minimal requirement for new assurance cases
- Lack of understanding of accepted practice
- Lack of evidential information regarding the benefits of modularity

Example GSN arguments from the IAWG case study were re-expressed using the SACM visual notation. It was identified that the real advantages that come from the use of SACM are not in the visual representation, but in the underpinning capabilities, particularly:

- traceability included in the underlying models
- the ability to define implementation constraints within the SACM models
- separation of multiple elements in multiple interfaces

Through these mechanisms, SACM provides a centralised formal interface between models. The transformations and transfers between models can be achieved using the formal interface between those models. This means that the relationships between the system models and the assurance case models can be

captured to facilitate automatic generation of argument that supports, for example, pattern instantiation and evidence generation. This also brings closer the potential for real-time re-evaluation of assurance cases.

Modular assurance cases are beneficial in addressing ever increasing complexity. When expressed in SACM, design techniques such as Model-Based Systems Engineering and Agile processes can be dealt with effectively, supporting realisation of the benefits of those techniques. Modular assurance cases also promote and support re-use of the separate parts.

The authors believe that modular assurance cases and model integration of notations such as SACM will be fundamental to dealing with the complexity of modern systems and realising the benefits of modern design tools and methodologies.

References

Birch, J – (2019) - "What's the case for safety in Automotive?" – SCSC Seminar: Evolution of Assurance Case Practice, available from http://www.scsc.org.uk

DEF STAN 00-74: ASAAC Standards Part 1: Standards for Software, 2008 – available via: https://www.dstan.mod.uk/

Denney, E, Pai, G and Pohl, J - "Advocate: An assurance case automation toolset". in proc. Workshop on Next Generation of System Assurance Approaches for Safety Critical Systems (SASSUR), pp 8-21, 2012.

Fenn J, Hawkins RD, Williams PD, Kelly TP, Banner MG, Oakshott Y (2007) - The Who, Where, How, Why And When Of Modular And Incremental Certification – IET Systems Safety Conference

IAWG, Modular Software Safety Case Process Description, MSSC 201, Issue 1, November 2012, available via: https://www.amsderisc.com/resources/www.capability-agility.co.uk/capability-agility/work-package-3/

OMG, 2019. Structured Assurance Case Metamodel version 2.1. available at: https://www.omg.org/spec/SACM/2.1/Beta1/PDF

PETER – EU Horizon 2020 Marie Sklodowska-Curie Project – Pan-European Training, Research & Education Network on Electromagnetic Risk Management - https://ec.europa.eu/research/mariecurieactions/

SCSC-141B (2018) Goal Structuring Notation Community Standard Version 2 – available from http://www.scsc.org.uk

SEI Architecture Trade-Off Analysis, available via https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=513908

Wei, R., Kelly, T.P., Dai, X., Zhao, S. and Hawkins, R., 2019. Model based system assurance using the structured assurance case metamodel. Journal of Systems and Software, 154, pp.211-233.

A Practical Assurance Approach for Multi-Cores (MCs) Within Safety-Critical Software Applications

Mark Hadley, Mike Standish

Defence Science and Technology Laboratory (Dstl)

Portsdown West, UK

Abstract *Single-core processors are increasingly difficult to source with micro-processor manufacturers moving into a Multi-Core (MC) arms race for energy efficiency and performance improvement. However, performance gains by MC utilisation of many cores and shared resources brings challenges for qualification; e.g. interference paths that impact Worst-Case Execution Time (WCET). For high-integrity aviation systems (e.g. DO-178B/C level A and B) these challenges need to be re-solved for confidence to be gained to accept these MC based systems. MC is the future and we need a way to qualify and accept MC based safety-critical systems into service. This paper illustrates a practical implementation strategy for MCs on a safety-critical system within a UK airborne system that is currently undergoing an external qualification assessment. This paper documents the strategy in terms of recommendations based upon the development, verification, and validation activities undertaken. The strategy has been refined based upon our experiences*[1]*. The approach is based upon a diverse strategy which adopts quantitative and qualitative evidence.*

[1] *It should be noted the strategy pre-dates any guidance which currently exists for MC (e.g. CAST-32, CAST-32A, and MIL-HDBK 516c MC extensions).*

© Crown copyright (2020), Dstl. This material is licensed under the terms of the Open Government Licence except where otherwise stated. To view this licence, visit http://www.nationalarchives.gov.uk/doc/open-government-licence/version/3 or write to the Information Policy Team, The National Archives, Kew, London TW9 4DU, or email: psi@nationalarchives.gsi.gov.uk. DSTL/CP119120.
Published by the Safety-Critical Systems Club. All Rights Reserved

1 Introduction

Multi-Core (MC) Micro-processors (MCMPs) are common in the private and commercial sector; however, in the defence sector Single-Core Micro-processors (SCMPs) are generally used. Within large data centres (e.g. those provided by Google, Amazon, and Microsoft) they seek energy efficiency and power-load management. This can be achieved with load shedding between MCMP cores. This enables significant cost savings due to energy efficiency of MCMPs. Benefits such as these are resulting in SCMPs becoming obsolete due to the MP manufacturers increasingly not producing them. However, in part, it is the dynamic features (such as dynamic load shedding between MCMP cores) that reduces deterministic behaviour. These could be a cause of potential interference channels (these are activities that could interfere with the function being executed correctly). They are mainly related to timing functions which therefore can have an impact on any Worst-Case Execution Times (WCETs).

Interference channels are numerous depending on the MCMP properties. Examples include, but are not limited to: unused Peripheral Component Interconnect (PCI) interrupts[1] and conflicts in shared cache or external memory[2]. Interference channels are not the only qualification issue relating to MCMPs. Other issues include failure modes of MCMPs, such as what occurs if a single-core or localised cache fails? Are all cores treated equally when shared resources are used, e.g. cache and external memory, or does core bias exist[3]?

Difficulties obtaining detailed design knowledge of a MP are not a new issue due to Intellectual Property Rights (IPR); however, MCMPs increases the complexity of the architecture. This leads to increased hidden lower-level registers which are used for debugging, monitoring, and health checking. Combine this with the increased complexity of the software architecture that the MCMP resides in, leads to increased implementation, qualification, and accreditation complexity.

This paper provides an outline qualification strategy by defining recommendations that, in part, are being applied to a current military platform and subsequently refined based upon our experiences in the qualification of MCMP-based systems. This paper does not go into a fine level of granularity of the technical detail but hopefully is at a sufficient level for a non-technical reader to understand the issues surrounding MCMP.

---

[1] The Xilinx PCI Express Interrupt Debugging Guide contains further information which can be generalised to other MCMPs – see Xilinx (2014).

[2] ScienceDirect (2019b) contains a number of resources to gain further information on these topics.

[3] See Koufaty (2010) for an overview of core bias within MC architectures.

While MCMPs are not new, the qualification of MCMP for safety-critical systems is, and the qualification challenges for MCMP need to be addressed. This paper addresses these issues. This paper encapsulates the philosophy of multiple levels of assurance and diversity of evidence. This is built from the initial selection of the MCMP to the on-target MCMP testing with a special focus on stress testing in order to test the System-on-Chip (SoC) properties[4].

2 Adopting a Diverse Assurance Approach

In order to gain a suitable level of assurance for any MCMP implementation there is a need to gather evidence from a diverse range of sources. There is empirical evidence that it is the *combination* of approaches which increases assurance confidence, e.g. as demonstrated within the software testing domain (Hadley 2013). The term *diverse* in this context uses a general definition, i.e. it is to have variety (or to be assorted) and to be distinct in kind (Collins 1995). Thus, *diversification* of evidence may reduce dependencies on certain evidence types and increase confidence in the MCMP implementation through a number of evidential strands. This approach assists with negating qualification shortfalls (if they exist) by adopting equivalent, more relevant, or supporting evidence. The judgements on any evidence (including counter-evidence) is based upon Subject Matter Expert (SME) opinions. It should be noted that any SMEs must be Suitably Qualified and Experienced Personnel (SQEP).

The philosophy of the diverse assurance approach is to review a broad range of *quantitative* and *qualitative* evidence. This evidence includes, but is not limited to: process-based evidence (e.g. life-cycle artefacts and suitable development); in-service reliability arguments (pre-deployment based upon other MCMP implementations and post-deployment based upon the specific MCMP solution); and testing activities of varying levels of granularity (e.g. specific MCMP testing and system integration testing). This philosophy is in keeping with the diverse evidence approach which can be targeted at an overall platform level. This has been demonstrated via an assurance approach termed the *Wheel of Qualification* (Hadley and Standish 2019). This paper implements an assurance approach for MCMPs which is in keeping with the diverse evidence philosophy.

[4] See Techopedia (2017) for a very brief description of a SoC.

3 The MC Problem

A MCMP should be seen as a SoC with cores connected together by a form of databus that links the cores to either external memory or supporting low-level devices, e.g. Universal Serial Bus (USB), Peripheral Component Interconnect Express (PCIe), Power Management etc. Level 1 (L1), Level 2 (L2), and Level 3 (L3) are types of cache; this is a collection of memory that can be accessed more quickly than external memory. In short, L1 is smaller but has quicker access times then L2, and likewise compared to L3. L1 and L2 are sometimes local to the core, while L3 is often external and shared between the cores. Figure 1 shows a typical basic MCMP architecture.

Figure 1 indicates that the L3 cache is shared between the 4 cores. This is common in all major MCMPs and for some MCMPs the L2 cache is shared between cores, e.g. the QorIQ T2080 (NXP 2018). A number of potential interference channels and bottlenecks for conflict can be seen from the abstract hardware view in Figure 1; e.g. L3 cache, the databus, and low-level device support.

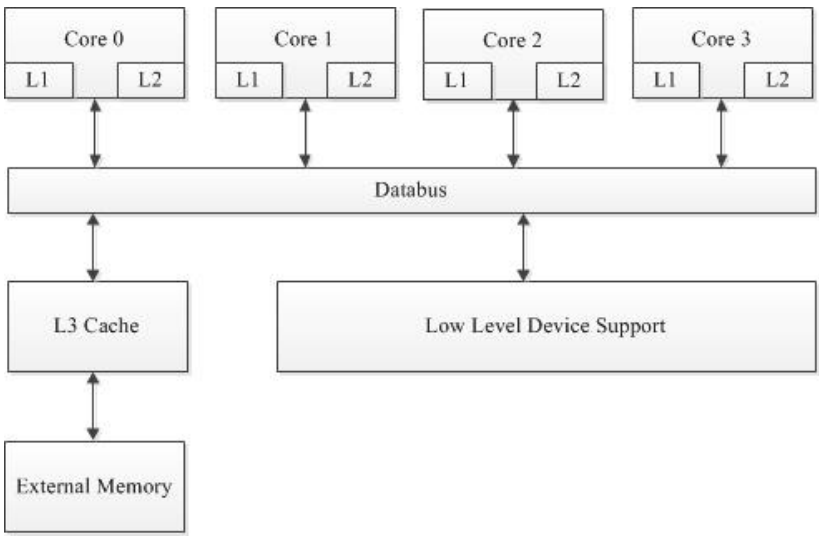


Fig. 1. Basic MCMP Architecture

Scenario based assessment could be adopted[5], e.g. Software Architecture Analysis Method (SAAM) (see Bass et al. (2012)), or finer grained approaches, e.g. Architecture Trade Assessment Method (ATAM) (see Kazman (1998)). These

[5] ScienceDirect (2019a) contains a number of resources to gain further information on this technique, including a comparison and evaluation of the methods.

would assess and evaluate possible interference channels on the MCMP architecture and allow judgements to be made on any mitigation. For example, the assessment could indicate a number of low-level devices that are not required and could be disabled, e.g. by the Board Support Package (BSP)[6] for the USB or PCIe.

While L3 shared cache is often the concern, L1 and L2 could be sources of interference when it is configured as Static Random Access Memory (SRAM). Therefore, local cache could be shared between cores. However, concurrent access to SRAM impacts WCET. If multi-threaded processing is implemented on the core, then the L1 and L2 cache could be shared by the threads. Similarly, with L2 and L3 cache, concurrent access could interfere and increase WCETs.

Gaining detailed design information on the MCMP is unlikely since all MP manufacturers are protective of their IPR. Although historically some vendors, but not many, do provide open source information and are supportive of high-level design meetings. For example, NXP leads a consortium looking at MCMP qualification, the Multicore for Avionics (MCFA) Working Group (WG). The WG membership includes hardware, software, and system suppliers, e.g. NXP, Rapita, Boeing, and BAE.

The following sections will provide an overview of some of the MCMP features which cause challenges from a qualification perspective. A number of suggestions in the form of recommendations are also stated to ameliorate such challenges.

4 MCMP Suitability Assessment and Intended Use

There is a requirement to assess the MCMP for suitability to the applied problem domain and to understand the initial shared resources of the MCMP.

4.1 MCMP Justification

All MPs contain low-level registers for debugging and monitoring of the cores. Design information of these registers is unlikely to be gained due to IPR limitations. This is also applicable to SCMPs as well as MCMPs. These features could increase possible interference channels and impact safety and/or security. There are known MCMP features which could, in principle, pose a risk to scheduling (e.g. interrupting a Real-Time Operating System (RTOS) partition) which could impact WCETs. Therefore, these features should be disabled or the integrator

[6] See TechTarget (2012) for a very brief description of BSPs.

who is implementing the MCMP within the architecture should justify why having certain features are acceptable (e.g. does not cause interference) and that the specified WCETs are still achievable.

Another factor that needs to be considered when selecting MCMPs is core bias where one core may have priority access to the resources. This can only be determined with the low-level design of the MCMP or with discussions with the MCMP developers. Since neither may be possible due to IPR limitations one can only determine this by direct evaluation of the MCMP.

Other features that could also cause possible interference include PCI or USB support which are not required for use within the system architecture. Therefore, the integrator should determine the features that are not required based upon the functional requirements. Other features that are not required should be disabled at the board level.

Recommendation 1. Feasibility analysis, on target evaluation of the MCMP, and justification for the MCMP being used should be documented. This includes, but it not limited to: properties of the MCMP to be down-selected; identification of interference channels (and how these are being mitigated or evidence that they do not impact the selected solution), identification of features that are not required and will be disabled; and failures modes and failure re-configuration of the cores (if applied). In addition, the feasibility analysis should indicate that the selected MCMP can meet the performance requirements for the developed system. This should include WCET requirements and any growth margin requirements.

4.2 Product Service History (PSH)

It is unlikely that a process (life-cycle) compliance argument could be generated fully for MCMPs or even for a SCMP. Within the aviation domain it has been suggested that DO-254 (RTCA 2000) could be adopted in terms of MCMP qualification; however, SCMPs have normally been qualified by in-service use arguments. MP manufacturers can provide vendors with usage data to support the qualification of the MPs (including failure reports). Whilst there is no direct guidance for the use (and level) of in-service hours for MCMP qualification it is likely that the quantity of service hours will be from a range of domains (specifically within telecommunications). Therefore, vendors need to generate a PSH argument and present it to the qualification/certification bodies.

Recommendation 2. Vendors should gain product service evidence and generate a qualification argument for the MCMP selected.

4.3 Errata Reviews

Corrections to the silicon often occur based upon the in-service use of the MPs. Errata sheets are used to detail the corrections between the silicon updates. While purchasing the latest MCMP may have little PSH, a process needs to be in place to review the errata sheets and determine the impacts on reported failures from the wider in-service use of the MCMP.

Recommendation 3. Errata sheets should be obtained and reviewed on a regular basis through-life for the MCMP. A corrective action strategy should be developed if any of the errata sheets document issues that impact safety or security of the MCMP used in the system.

4.4 Shared Resource Analysis

The majority of MCMPs include L1, L2, and L3 cache. Depending on the RTOS and/or configurations of the MCMP the interference channels may be reduced by static configurations of the MCMP or by partitioning the cache etc. Whilst the latter has the benefit of reducing interference this could impact on performance of the MCMP. For example, Cache Allocation Technology (CAT) is offered by Intel on some of their MPs, e.g. Intel Xeon processors E5 v4 family (Intel 2016).

Recommendation 4. The vendor should identify the resources (e.g. external memory and cache) which are shared between cores and determine how the interference is mitigated and managed.

5 Symmetric Multi-Processing (SMP), Asymmetric Multi-Processing (AMP), and Hypervisors

There are two broad configurations of MCMP in terms of RTOSs: either SMP or AMP. In short, SMP have one RTOS managing more than one or more cores, often a single instance of an RTOS managing all cores. This enables better use of shared resources and is often used to maximise Central Processing Unit (CPU) processing power. This is very similar to desktop computers.

An AMP configuration is where each core is managed by an independent instance of an RTOS and therefore each software process is locked to one core. This is very similar to legacy SCMPs. AMP is suited when legacy code is ported and allows developers to manage each core independently. Figure 2 and Figure 3 show the AMP and SMP configurations. As an example, Figure 2 illustrates two

instances of the WindRiver (WR) ARINC 653 RTOS on cores 0 and 1. The other two cores are not used and could be "switched off" or held in Power On Reset by the BSP. Figure 3 illustrates one RTOS, a WR VxWorks 6.9, that manages all the cores.

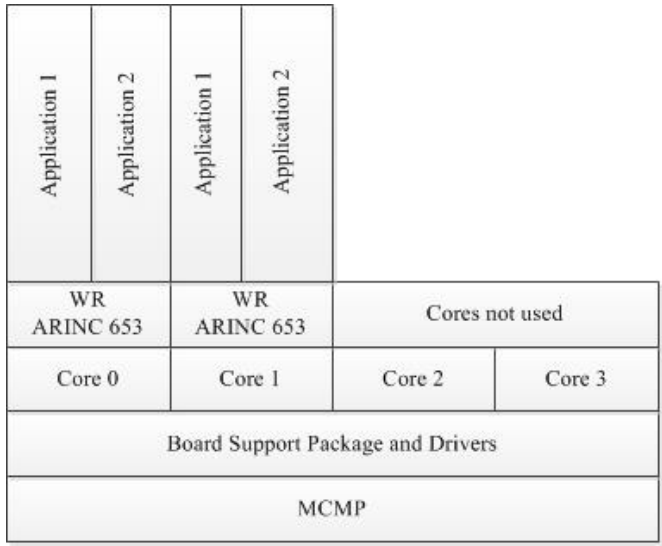


Fig. 2. AMP Cluster Configuration

Virtualisation enables multiple independent operating systems (OS) (often referred to as a Guest OS) to be executed concurrently on a shared hardware system. Hypervisors are used to manage the actual physical hardware interactions with the Guest OS. There are two types of hypervisors: Type 1 is often referred to as "bare metal" which runs directly on the hardware; and Type 2 hypervisors which run architecturally above the RTOS. This therefore, leads to dependencies on the RTOS to ensure separation and increase safety and security qualification needs.

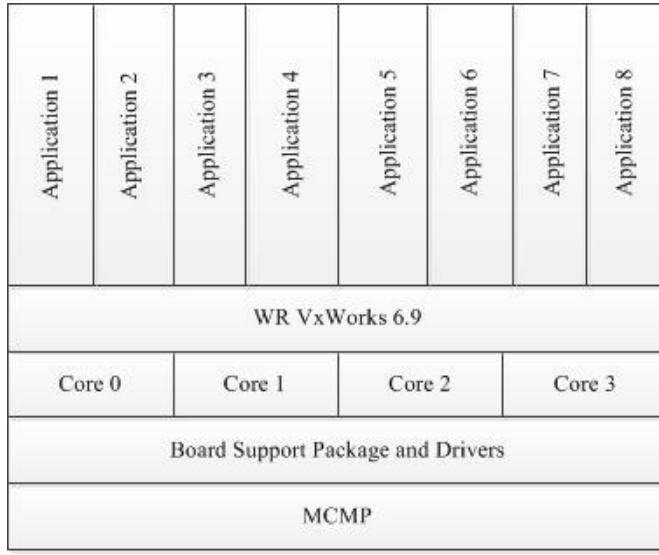


Fig. 3. SMP Cluster Configuration

Type 1 hypervisors are smaller in terms of functions and are specifically designed to ensure robust partitioning; therefore the qualification needs should be less. The major issue is obtaining the low-level implementation detail of a Type 1 hypervisor. Figure 4 illustrates this with three virtual machines, one for each Guest OS, with two RTOSs in an AMP configuration (WR and Green Hills) and LynxOS in an SMP cluster managing two cores. The RTOSs provide their own robust partitioning in terms of time and space partitioning.

Recommendation 5. The hypervisor's robust partitioning between the Guest OSs and the physical hardware layer should be verified.

6 RTOS, BSP, and Low-Level Device Driver Qualification

An RTOS which is certifiable for a safety-critical software application requires analysis to ensure that the specific intended solution is appropriate. There are also considerations for the lower-level features of the architecture.

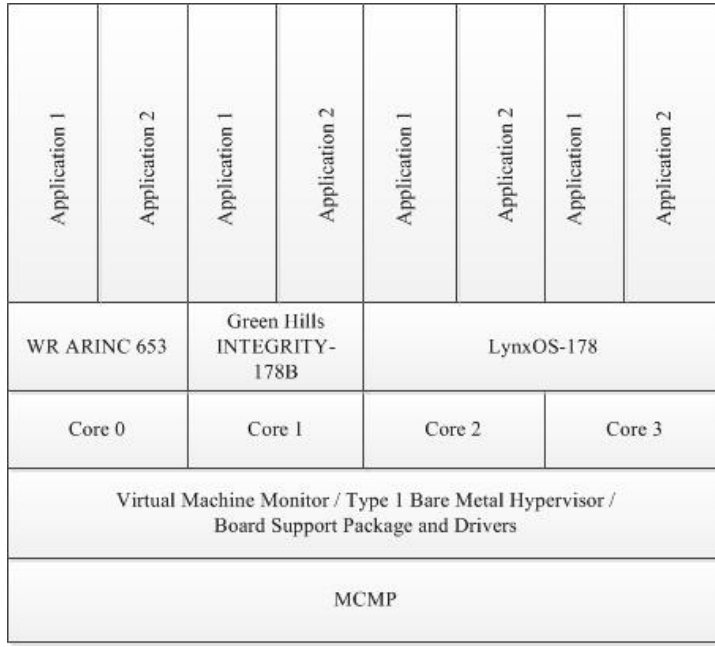


Fig. 4. SMP and AMP Cluster Configuration Including Hypervisor

6.1 Comply with Certification Data Packages

A range of RTOSs are provided with certification data packages; e.g. the WR VxWorks Cert Platform (WindRiver 2019). The certification data pack provides the life-cycle data items to demonstrate compliance against the objectives in DO-178B/C (RTCA 1992 & RTCA 2011). These certification data packages are normally based upon a pre-defined BSP and assumptions on the use of the RTOS. The RTOS certification data packs normally have a Software Vulnerability Analysis (SVA) document (WindRiver 2019) and a Certification Evidence Integration Guide. SVA notes should be provided to the System Integrator. The SVA notes define a number of additional verification steps the Integrator should conduct for the RTOS qualification assumptions to remain valid. For example, re-running the entire RTOS test procedures and confirming the low-level features are used in a defined way (as defined by the RTOS certification data pack, e.g. memory allocation). The RTOS certification is dependent on the BSP configuration. If no changes are made to the BSP which is provided by the RTOS manufacturer then no additional verification activities are required (however, this is rarely the case). Due to this, the qualification evidence of an RTOS on one system cannot be solely read across when applied to a different system with different BSP configurations.

Recommendation 6. The certification data pack for the RTOS being considered should be used and benchmarked against the appropriate standard/guideline, e.g. DO-178C.

Recommendation 7. The System Integrator should review and assess the Software Vulnerability Analysis (SVA) and the integration guide and re-run the certification test procedures. This will ensure the original RTOS certification evidence remains valid.

6.2 Robust Time and Space Partitioning

Certified RTOSs provide robust time and space partitioning. For example, the ARINC 653 Specification (ARINC 2015) defines the requirements for time and space partitioning and scheduling. Space partitioning involves the memory being allocated to each program and only that program can access their memory space. Therefore, in Figure 5 the *Tactical* partition can only access the memory allocated to it, i.e. C800 to EFFF. Time partitioning involves program threads/tasks being separated by time. For example, in Figure 6 the *Displays* partition has been allocated 20ms of processing time with a 10ms gap which could be used for network related activities before the *Tactical* thread gets the processing time and so on.

While time and space partitioning concepts are not new they are fundamental in laying the foundation in the partitioning of programs and reducing conflict between programs.

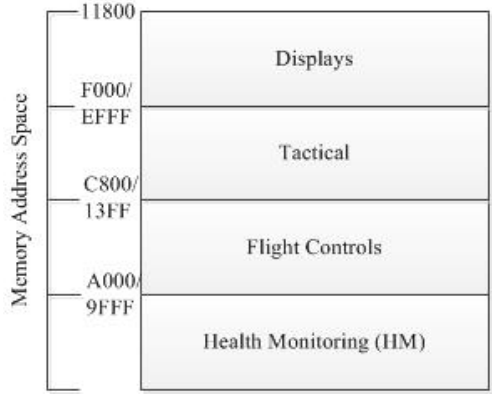


Fig. 5. Sample Space/Memory Partitioning

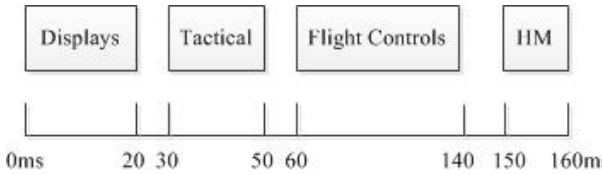


Fig. 6. Sample Time Partitioning

Recommendation 8. Robust partitioning (i.e. time and space) should be implemented within MCMP systems. If this is not the case the vendor needs to provide justification and assurance for how any interference is being avoided between program threads on the same core (and on different cores). In addition, it should be determined how WCET is being guaranteed.

6.3 Failure and Error Management

In Figure 5 and Figure 6 there is a *Health Monitoring* (HM) partition. HM can be bespoke or used by the RTOS HM handling capability which many RTOSs provide. HM provides a framework to raise and manage alarms (errors) in a system. How failures and errors are managed (including recovery) needs to be justified and documented.

Recommendation 9. The vendor should determine the failure and error reporting system and how will these failures will be managed and recovered. This should include the whole SoC properties and the software architecture which the MCMP resides in.

6.4 Suitable BSP and Complex Electronic Hardware (CEH) Assurance

If the BSP and low-level supporting device drivers are amended to support features on the MCMP hosted board then the features should be qualified to the same level as the required Development Assurance Level (DAL). Similarly, if Complex Electronic Hardware (CEH) (e.g. Field Programmable Gate Arrays (FPGAs) or Application Specific Integrated Circuits (ASICs)) is used then DO-254 should be applied.

Recommendation 10. The developer should ensure that any modification to the BSP and low-level device drivers should be developed and qualified to the appropriate DAL.

Recommendation 11. If CEH is used on the MCMP hosted board then DO-254 (or in the case of non-aviation domains another CEH development guideline) should be applied.

7 Specific MC Testing

Within the aviation domain European Union Aviation Safety Agency (EASA) CM-SWCEH 001 (EASA 2018) and CAST-32A (CAST 2016) refer to on-target testing with DO-178C referring to related testing requirements. However, these requirements would appear insufficient to provide confidence in the MC properties and the shared properties of MCs. MC should be considered as a SoC. Many of the requirements in DO-178B and DO-178C provides confidence that the functional requirements/interfaces have been correctly implemented and robustness testing provides some direct MC confidence. However, CAST-32A provides little additional software testing guidance beyond that which already exists in DO-178C.

We believe this is insufficient and propose the use of stress testing based upon mission testing scenarios to stress the underlying SoC properties on the target hardware. Stressing the underlying system properties could be achieved via greedy algorithms that consume resources; e.g. processor, local and external cache, and external memory. We prefer here to use stress testing as defined in the classical testing literature in the 1980s, e.g. Bezier (1984), and use mission system profiles to stress the system under high-load.

Stress testing is testing the System-Under-Test with high background loads with the aim of overloading one, several, or all, of the resources. Resources could be hardware (e.g. memory, micro-processor, network architecture, external storage devices) to software (e.g. loops, interrupts, buffers). Stress testing should be based upon the usage profile of the system (stress test use-cases should be generated). While the stress testing may not be physically feasible (e.g. concurrently pressing each key on the keyboard), stress testing distorts the normal load order of processing; more so when processing occurs at different priority levels. Stress testing forces race conditions, distorts normal processing loading, and overloads the system boundaries. In doing so it tests the control and thresholds of the system to manage these situations and perhaps, most importantly, depletes resources in sequences that may be undetected by manual verification and oversight activities.

Stress testing is often applied at the system level and is normally conducted late in the development life-cycle. However, there is a preference to capture failures *early* in the life-cycle stages. It is essential for MCMP based systems for stress testing to be performed to validate the interference analysis so that performance and WCET requirements can be met under extreme stress.

Recommendation 12. Stress testing should be conducted with test cases based upon the requirements of the system to stress the underlying SoC properties of the MCMP. Results can support the validation of the MCMP interference analysis, performance requirements, and show how the system manages with extreme load levels.

8 MCMP Qualification

CAST-32A and MIL-HDBK-516c MC extensions (Jackson 2019) provide a level of guidance/requirements for MCMPs. CAST-32A is a position paper for MCMP qualification and is published for educational and informational purposes only by the Federal Aviation Administration (FAA). CAST-32A can be summarised into the following broad technical areas:

- Planning settings of resources.
- Interference channels and resource usage.
- Software verification.
- Error detection, handling, and safety nets.

This section of the paper does not re-state or provide additional guidance already stated in CAST-32A. Instead it provides the type of areas that should be covered by any MCMP qualification strategy, in many ways it is the summary of the recommendations already stated in this paper. The UK Ministry of Defence (MOD) Military Aviation Authority (MAA) has stated that the qualification of MCMP based systems is on a case-by-case basis. A Military Critical Review Item (MCRI) process defined in RA5810 (MAA 2018) is used to document the qualification strategy for MCMP based systems.

WR and Rockwell Collins documented the qualification of a mixed criticality DAL system which is waiting a final Stage of Involvement (SOI) assessment. A Technical Standard Order (TSO) was to be granted in early 2019 (Radack 2019). To date no virtualised airborne based system has been put forward for regulatory qualification, e.g. to FAA/EASA, to the best of our knowledge.

This paper outlines a MC strategy that, in part, encapsulates some of the guidance in CAST-32A:

- Characterise the on-target evaluation and select an appropriate MCMP. Gain usage evidence to support the MCMP qualification (recommendations 1 and 2).
- Review MCMP errata sheets during the life of the MCMP (recommendation 3).
- Evaluate and *design out* interference paths (recommendations 1 and 4).

- Design in static configurations in terms of memory and setting of resources (recommendation 4).
- Design in robust time and space partitioning (recommendations 5 and 8).
- Design in error handling and HM (recommendation 9).
- Ensure that all layers architecturally above the MCMP have been developed to the appropriate DAL. If the high-level supporting component is level A then all the supporting software, including the RTOS, should be developed to that level. Gain life-cycle data items for independent review (recommendations 6, 7, 10 and 11).
- In addition to robustness testing (as defined in DO-178/C) the tests should focus on the verification of the space and time partitions. If hypervisors are used to implement virtual machines then methods of hypervisor verification also need to be conducted (recommendation 5).
- Conduct on-target testing for all formal software acceptances.
- Conduct stress testing to stress the SoC properties (not stated within CAST-32A) (recommendation 12).
- Use the wider qualification evidence in terms of System-of-Systems (SoS) integration, rig testing, and final acceptance of the total system evidence if that is applicable (not stated within CAST-32A).

9 Wider MCMP Qualification Considerations

MCMP qualification should not be seen in isolation to the wider qualification that the system may reside within. For example, many systems are integrated into wider systems (e.g. SoS) (e.g. aircraft platforms and telecommunication systems). These systems are integrated and tested together along with the final acceptance and validation of the total SoS, e.g. flight trails. An approach is to develop a solution and conduct flight trails *before* formalising the development of the system. These activities; such as formal integration, rig testing and flight trials, all provide additional evidence to support the qualification of the MCMP and the system which it resides within. This adheres to the philosophy of adopting diverse assurance evidence.

10 Summary

This paper has set out the main implementation and qualification challenges of MCMPs within an aviation context. However, the lessons and recommendations are transferrable to other safety-critical software domains. To overcome the chal-

lenges and to generate evidence to support MCMP qualification a number of recommendations have been proposed (see Table 1). These have been based upon a qualification strategy applied on an airborne system that is currently undergoing external qualification assessment.

The strategy was derived before any guidance was available in terms of CAST-32, CAST-32A, or the Mil-HDBK-516C MC extensions. The qualification domain is now generating requirements for MC qualification; however, it has been almost a decade since the introduction of MCs into an airborne based system.

Table 1. MCMP Qualification Recommendations

#	Recommendation
1	Feasibility analysis, on target evaluation of the MCMP, and justification for the MCMP being used should be documented. This includes, but it not limited to: properties of the MCMP to be down-selected; identification of interference channels (and how these are being mitigated or evidence that they do not impact the selected solution), identification of features that are not required and will be disabled; and failures modes and failure re-configuration of the cores (if applied). In addition, the feasibility analysis should indicate that the selected MCMP can meet the performance requirements for the developed system. This should include WCET requirements and any growth margin requirements.
2	Vendors should gain product service evidence and generate a qualification argument for the MCMP selected.
3	Errata sheets should be obtained and reviewed on a regular basis through-life for the MCMP. A corrective action strategy should be developed if any of the errata sheets document issues that impact safety or security of the MCMP used in the system.
4	The vendor should identify the resources (e.g. external memory and cache) which are shared between cores and determine how the interference is mitigated and managed.
5	The hypervisor's robust partitioning between the Guest OSs and the physical hardware layer should be verified.
6	The certification data pack for the RTOS being considered should be used and benchmarked against the appropriate standard/guideline, e.g. DO-178C.
7	The System Integrator should review and assess the Software Vulnerability Analysis (SVA) and the integration guide and re-run the certification test procedures. This will ensure the original RTOS certification evidence remains valid.
8	Robust partitioning (i.e. time and space) should be implemented within MCMP systems. If this is not the case the vendor needs to provide justification and assurance for how any interference is being avoided between program threads on the same core (and on different cores). In addition, it should be determined how WCET is being guaranteed.
9	The vendor should determine the failure and error reporting system and how will these failures will be managed and recovered. This should include the whole SoC properties and the software architecture which the MCMP resides in.
10	The developer should ensure that any modification to the BSP and low-level device drivers should be developed and qualified to the appropriate DAL.
11	If CEH is used on the MCMP hosted board then DO-254 (or in the case of non-aviation domains another CEH development guideline) should be applied.

#	Recommendation
12	Stress testing should be conducted with test cases based upon the requirements of the system to stress the underlying SoC properties of the MCMP. Results can support the validation of the MCMP interference analysis, performance requirements, and show how the system manages with extreme load levels.

Many of the qualification recommendations stated in this paper are now embodied in CAST-32A or the MIL-HDBK-516c MC extensions; however, stress testing and the consideration of wider qualification evidence are not. Also, we have attempted to guide the reader into approaches to assess interference and generate PSH to generate evidence that makes up the wider qualification argument for MCMP.

All systems and MCMPs are different, but the recommendations and strategy should remain valid. This paper also indicates the use of broader evidence outside the system qualification to support the qualification of the MCMP. A system is often part of a SoS and the wider diverse evidence needs to be considered when making a qualification argument. This is not just true for MCMP based systems but for all systems. The approach adopts both *quantitative* and *qualitative* diverse evidence.

Some of the qualification challenges for MCMPs are not new, e.g. ensuring separation of programs and external memory shared between programs. However, many are new in terms of managing more than one core and the removal of interference channels from the SoC. We are still in the early stages in terms of qualification of MCMP for high level DAL systems (i.e. A and B) and gaining civil regulatory approval for aviation (e.g. FAA/EASA). However, we have limited choice but to embrace MC since MP manufacturing is heading solely down that path. A diverse assurance approach can assist in gaining the required levels of confidence for SQEP SMEs.

References

ARINC (2015) Avionics Application Software Standard Interface. Parts 1-3.

Bass L, Clements P, Kazman R (2012) Software Architecture in Practice. Addison Wesley.

Beizer B (1984) Software System Testing and Quality Assurance. Van Nostrand Reinhold.

CAST (2016) Certification Authorities Software Team: Position Paper CAST-32A. Multi-Core Processors. November 2016.

Collins (1995) English Dictionary and Thesaurus. HarperCollins.

EASA (2018) EASA CM-SWCEH-001: Development Assurance of Airborne Electronic Hardware. Issue 1, Revision 02. January 2018.

Hadley M (2013) Empirical Evaluation of the Effectiveness and Reliability of Software Testing Adequacy Criteria and Reference Test Systems. PhD thesis. University of York. http:// etheses.whiterose.ac.uk/ 5861/. Accessed 23 October 2019.

Hadley M, and Standish M (2019) Using Tiers of Assurance Evidence to Reduce the Tears! Adopting the "Wheel of Qualification" for an Alternative Software Safety Assurance Approach. High Integrity Software (HIS) Conference, 5th November 2019, Bristol, UK.

Intel (2016) Introduction to Cache Allocation Technology in the Intel Xeon Processor E5 v4 Family. https://software.intel.com/en-us/articles/introduction-to-cache-allocation-technology. Accessed 23 October 2019.

Jackson C (2019) Verification of Computer Systems Utilizing Multicore Processors From an USAF Airworthiness Certification Perspective. Multi-Core Processors Test and Validation Workshop, April 30 – May 1 2019, Dayton, OH, USA.

Kazman R, Klein M, Barbacci M et al (1998) The Architecture Tradeoff Analysis Method. 4th International Conference on Engineering of Complex Computer Systems (ICECCS '98), 10-14 August 1998, Monterey, CA, USA.

Koufaty D, Reddy D, Hahn S (2010). Bias Scheduling in Heterogeneous Multi-Core Architectures. 5th European Conference on Computer Systems (EuroSys '10). 13-16 April 2010, Paris, France.

MAA (2018) Regulatory Article (RA) 5810 - Military Type Certificate. Issue 2. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/712833/RA5810_Issue_2.pdf. Accessed 23 October 2019.

NXP (2018) QorIQ T2080 and T2081 Multicore Communications Processors. https://www.nxp.com/products/processors-and-microcontrollers/power-architecture/qoriq-communication-processors/t-series/qoriq-t2080-and-t2081-multicore-communications-processors:T2080. Accessed 24 October 2019.

Radack D, Tiedeman Jr H, Parkinson P (2019) Civil Certification of Multi-core Processing Systems in Commercial Avionics. 27th Safety-Critical Systems Symposium, 5-7 February 2019, Bristol, UK.

RTCA (1992) DO-178B: Software Considerations in Airborne Systems and Equipment Certification.

RTCA (2000) DO-254: Design Assurance Guidance for Airborne Electronic Hardware.

RTCA (2011) DO-178C: Software Considerations in Airborne Systems and Equipment Certification.

ScienceDirect (2019a) Architecture Evaluation. https://www.sciencedirect.com/topics/computer-science/architecture-evaluation. Accessed 24 October 2019.

ScienceDirect (2019b) Cache Conflict. https://www.sciencedirect.com/topics/computer-science/cache-conflict. Accessed 24 October 2019.

Techopedia (2017) System on a Chip (SoC). https://www.techopedia.com/definition/702/system-on-a-chip-soc. Accessed 24 October 2019.

TechTarget (2012) Board Support Package. https://whatis.techtarget.com/definition/board-support-package. Accessed 24 October 2019.

WindRiver (2019) VxWorks Cert Platform Product Overview. https://www.windriver.com/products/product-overviews/vxworks-cert-product-overview/. Accessed 24 October 2019.

Xilinx (2014) Xilinx Answer 58495 – PCI-Express Interrupt Debugging Guide. https://www.xilinx.com/Attachment/Xilinx_Answer_58495_PCIe_Interrupt_Debugging_Guide.pdf. Accessed 24 October 2019.

Disclaimer This article is an overview of UK MOD sponsored research and is released for information purposes only. The contents of this article should not be interpreted as representing the views of the UK MOD, nor should it be assumed that they reflect any current or future UK MOD policy. The information contained in this article cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.

Demystifying Functional Safety in Road Vehicles – ISO 26262

Rajiv Bongirwar

HEMRAJ CONSULTING, UK[1]

Abstract *Understanding the need to implement and address some challenges posed by ISO 26262.*

1 Introduction

The second edition of ISO 26262:2018 Road vehicles – Functional safety, was released in Dec 2018 and now includes Truck, Bus and Motorcycle manufacturers within its scope; Mopeds being the only road vehicles for general public use that are excluded. The standard poses many challenges to the industry and this paper addresses some key aspects that will enable the industry in not only adopting the standard but also leverage its compliance to deliver quality vehicles to the end customer. Vehicles designed and developed by a process compliant with this standard will avoid unreasonable risk to human injury, are more reliable and less prone to failures.

Key questions addressed in this paper include:

- Need for automotive Original Equipment Manufacturers (OEM's) and suppliers to comply with ISO 26262
- Relationship between functional safety and product safety
- Common myths or questions concerning ISO 26262
- Actions an organisation needs to take to deliver ISO 26262 compliant products

[1]Trading name for Hemraj Consultants Limited, UK
Email: director@hemraj.co.uk
Website: www.hemraj.co.uk

© Rajiv Bongirwar of HEMRAJ CONSULTING 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

The rise of electronic and software content in vehicles has been exponential in the last 10 years. Premium passenger vehicles have a few hundred Electronic Control Units (ECU's) and implement few thousand functions. Figure 1 shows location of typical ECU's in a car. Automotive electronics cost as a share of total car cost rose from 5% in 1970 to an estimated 50% in 2030 (35% in 2010) (Deloitte, 2019).

The factors driving this growth include regulatory requirements, emissions control aerodynamics, CO2 footprint, sustainability, increasing customer demands & expectations and market competitiveness. This explosive growth of E/E (Electrical & Electronics (includes Software and Firmware)) in vehicles has significantly increased the opportunities for failures and their occurrence has resulted in human fatalities, injuries or adverse impact on health. For example, in 2014, faulty ignition switches in General Motors vehicle resulted in 124 deaths and more than twice as many injuries costing the company USD 4.1 billion and 30.4 million cars to be recalled worldwide (Burrows D, 2018).

Amongst the latest known examples of an E/E failure is the grounding of the entire Boeing 737 Max 8 Fleet worldwide after two crashes killed 346 people primarily caused by the faulty Angle of Attack Sensor and Manoeuvring Characteristics Augmentation System (MCAS) relying on a single AOA source amongst other contributing factors (KNKT, 2019). Boeing's pledge to support families of victims alone has costed them US Dollar (USD) 100 million (Kent G, 2019). It is therefore not surprising that International Standards Organisation released the first version of ISO 26262 in 9 parts in Nov 2011 setting the threshold of best practices automotive OEM's and suppliers needed to adopt to ensure Functional Safety. This was closely behind the Toyota's recall of their hybrid vehicles including the bestselling 2010 Prius model due to faulty anti-lock braking software resulting in the recall of 436,000 vehicles globally on 9th Feb 2010 – (Reuters, 2010 and Various authors in Wikipedia, 2019). This discipline is very challenging as it deals with high complexity in the presence of prevailing component-driven vehicle design as against the required top-down systems engineering approach and this paper explains the "Why's" and "What's" for compliance with this standard.

Fig. 1. Location of typical ECU's in a car (Magazine, 2010)

2 The rising contribution of Functional Safety related incidents within overall Product Safety

2.1 Relationship between Functional Safety and Product safety

At the outset, let us first understand the various categories of overall Product Safety and how Functional Safety relates to Product Safety. Safety as defined in (ISO 26262, 2018) is 'absence of unreasonable risk'. Safety could therefore be broken down into three 3 components mentioned below, as further detailed in (Ward D, 2019):

a) Functional Safety – 'absence of *unreasonable risk* due to *hazards* caused by *malfunctioning behavior* of *E/E systems*' as defined in (ISO 26262, 2018)
b) System Safety – To make it mutually exclusive from Functional Safety and make it applicable only to E/E systems to maintain a common scope of applicability with respect to engineering disciplines, 'System Safety' could be defined as 'absence of unreasonable risk due to hazards caused by intended

behavior of E/E systems'. The following have been taken into consideration to propose this definition:

- 'System safety' is mentioned in (Squires A, 2019) 'As an engineering discipline, *system safety* is concerned with minimizing hazards that can result in a mishap with an expected severity and with a predicted probability.'
- 'System safety' as defined in (MIL STD 882E, 2012) is "The application of engineering and management principles, criteria, and techniques to achieve acceptable risk, within the constraints of operational effectiveness and suitability, time, and cost, throughout all phases of the system life cycle" (DoD 2012)
- *'The absence of unreasonable risk due to hazards resulting from functional insufficiencies of the intended functionality or by reasonably foreseeable misuse by persons is referred to as the Safety Of The Intended Functionality (SOTIF)' – (*ISO/PAS 21448:2019)

c) Product Safety – If we now were to think of the overall Safety as a superset, the two sub-sets (a) and (b) above leave only the non-E/E related engineering disciplines of a system thus leading to this possible definition – 'absence of unreasonable risk due to hazards caused by non E/E systems (such as chemical reactions – corrosion & explosion, mechanical vibrations, structural integrity, thermal events) OR impact of environment (Takata Airbag recall mentioned in (Burrows D, 2018) AND any hazard resulting in *harm* to equipment or environment that does not *harm* humans'. Following inputs also considered in proposing this definition:

- IATF 16949 defines *Product Safety* as 'standards relating to the design and manufacturing of products to ensure they do not represent harm or hazards to customers'

From the above, we infer that Functional safety is a subset of System Safety which in turn is a subset of Product safety. The reason to explicitly differentiate these subsets is different competencies required to address these disciplines. Further, it helps to categorize safety related failures reported so as to better assess the root cause and remedial actions to resolve and prevent recurrence of similar failures in the future.

2.2 Rising contribution of Functional Safety in Vehicle recalls

In absolute terms, as of 2016, the Takata airbag recall is estimated to be the costliest recall at 24 billion USD (US Dollar). This recall started in 2008 and is expected to conclude in 2023 - as of Mar 2018 (Burrows D, 2018). Although it could be argued that this recall is outside the scope of Functional Safety, it gives an indication of the magnitude of exponential costs of vehicle recalls that could be attributed to Functional Safety.

The adoption of safety-related electronics systems has grown explosively. Semiconductor components that make up these electronic systems will cost USD 600 per car by 2022 (Deloitte, 2019). Electronic systems as a % of total car cost is estimated to rise to 50% in 2030 compared to 35% in 2010 and 10% in 1980 when Electronic fuel injection was introduced. With onset of Camera Monitoring Systems replacing mirrors in cars and trucks and stringent Vulnerable Road User's (VRU's) regulations being introduced in EU soon, this cost per vehicle will be much higher.

According to a study 'The Auto Industry's Growing Recall Problem -- and How to Fix It' referenced in (Jibrell A, 2019), the number of recalls related to electronic and electrical systems have risen nearly 30 percent per year since 2013, compared with 5 percent annual increases from 2007 to 2013, hence the more sophisticated infotainment and safety systems are drawing more attention. . The study in this reference also noted that the number of vehicles using similar systems is increasing due to adoption of global rather than local platforms, resulting in recall notices increasingly likely to involve not thousands but millions of vehicles.

The rise in Vehicle recalls related to E/E content is due to both Functional Safety related and many other causes of failures. Amongst the two categories, our endeavor should be to prioritize prevention of human injuries and fatalities over vehicle or infrastructure damage and hence minimizing functional safety related failures.

3 Can we defer ISO 26262 compliance?

Complying with ISO 26262 helps deliver safe and reliable products with enhanced quality – on time and within budget when implemented appropriately. If this reason alone is not enough, section 2.2 above gives a quantitative indication[1]

[1] The reason being not all E/E failures resulted in a hazardous situation. For example, if a car does not start, it results in aborting the mission but cannot cause harm, hence not a functional safety related problem.

of the rising costs of addressing inadequacies in functional safety. Lastly, from a product liability perspective, if evidence of compliance with current revision of ISO 26262 prevailing at the time of start of product development can be evidenced in a court by providing a robust Safety Case, the OEM and / or supplier are very likely to be removed from any liability as they would be deemed to have applied due diligence, reduced risk to an acceptable level and designed and developed products according to state-of-art technology[2] and industry best practices. It is extremely difficult if not impossible to prove this without complying with ISO 26262.

Section 3.1 below lists the components of reduced costs that you will have to incur if you choose the pro-active approach of embodying quality upfront, else, the exponentially higher costs of not investing in quality upfront and paying later instead of paying now.

Section 3.2 provides statistical information that evidences increased costs of fixing failures that are rapidly rising and yet the budget for proactive vehicle quality continues to take a hit due to cost cutting measures being adopted by the automotive industry.

3.1 Pay now or very likely pay more later

Automotive OEM's and suppliers will have to pay upfront (cost of proactive quality) or pay much more later in the event of vehicle recalls, warranty claims or law suites. It's better to build safety, compliance & quality in the product upfront and save time money, resources aggravation and rework. The various cost components in each of the two scenarios are listed below:

a) Pay now (Proactively planned measures to ensure quality[3])
   - Estimation and planning of resources
   - Quality Assurance
   - Quality Control
   - Continuous monitoring

   OR

b) Very likely, pay more later (Cost of inadequate quality)
   - Vehicle recalls
   - Liability claims
   - Warranty claims
   - Brand reputation

[2] With additional supporting evidence relevant to the technology used

[3] Safety is a subset of Quality!

- Reduced market share price
- Ethical & social responsibility
- Loss of jobs
- Bankruptcy
- Rework and repair
- Scrap

3.2 Paradox of the need and action

The following statistics taken from (Jibrell A, 2019) indicate the magnitude of cost of inadequate quality:

- It cost automakers and suppliers approximately USD 11.8 billion in claims and USD 10.3 billion in warranty accruals for USA recalls in 2016. The USD 22.1 billion total is an estimated 26 percent increase over the previous year USD 17.5 billion
- The number of vehicles recalled in the U.S. in 2016 rose 4.5 percent to 53.1 million, from 50.8 million in 2015 --making 2016 the highest year on record, the study says (Almost 50% of those recalled vehicles were attributed to Takata Corp.'s defective airbag inflators or General Motors' faulty ignition switches, which combined for 23 million.)
- Automotive suppliers' share on the rise:
    - total recall costs have tripled from 5 to 7 percent from 2007 to 2013, to 15 to 20 percent since 2013 and
    - the frequency that suppliers are named in recall notices has doubled

Despite the above facts, automakers and suppliers remain focused *on innovation and cost cuts while vehicle quality takes a hit*. The study mentioned in (Jibrell A, 2019) estimates that automakers and suppliers have slashed spending on quality functions by 30 to 50 percent since the economic recession in October 2008.

To clarify, section 2.2 focuses on increased proportion of functional safety within product safety - so increased cost of E/E systems related vehicle recalls, increased E/E content in the vehicles etc. whereas this section is about overall increased cost of vehicle recalls, warranty claims, and degrading quality.

4 Some common myths around ISO 26262

'We can't solve problems by using the same kind of thinking we used when we created them.' – Albert Einstein

4.1 Root cause of E/E failures

The root cause of many of the problems[4] are not related to electrical, electronic or software (even though they can only manifest here), instead, they are often related to the effort estimation, resources deployed, organisation structure, roles and responsibilities, competency matrix and defining, communicating and getting a buy-in of an unambiguous 'Responsible-Approves-Supports-is Informed-is Consulted' responsibility matrix (RASIC) for the various work products and activities in product development, manufacturing, servicing and decommissioning processes.

Additional examples of root causes include, but are not limited to the following:

- inadequate, conflicting, ambiguous or misinterpretation of requirements
- inappropriate impact analysis due to:
    - not consulting all relevant stakeholders and Subject Matter Experts (SME's)
    - not analysing ALL changes between old and new Product
- not adhering to the correct chronology of activities
- incorrect assumptions about system behaviour

In many instances, the risk reduction phase post safety analysis either does not happen or is performed too late and does not find its way into the design. This results in the correct safety measures including safety mechanisms not being defined and hence the violation of Safety Goals (SG's, as defined in ISO 26262:2018 Part 1).

[4] 15 common mistakes in implementing ISO 26262 are mentioned in (Hilderman V)

4.2 Where is the time and money for this added rigor?

We don't spend the money when and where it is most beneficial and pay much more later when we are forced to, a simple example is grounding of Boeing Max 737 8 fleet almost worldwide is costing the company more than USD 1 billion (The Guardian, 2019) whereas proactive spend on quality could have avoided loss of human lives, equipment and money at a very small fraction of this cost.

Analysis of statistical data has revealed (Jibrell A, 2019) that most companies do not know their real cost of quality and are therefore unable to prioritise quality activities for product development leading to series production of components or vehicles for use on public roads. Sections 2.2, 3.1 and 3.2 have quantified the exponentially higher costs automotive OEM's and suppliers have had to pay for E/E failures. It pays rich dividends to spend a fraction of that time and money upfront, deploy the right resources under able guidance, provide them with enough time (enabled by proper planning) and a mandate to diligently follow a compliant process that is also adapted to the unique circumstances of each Project, Department or Organisation. Tailoring is most likely required for all Projects as each is different in some way from others.

4.3 Work products are frozen at Project Gateways

It is common experience that Project Managers demand work products, e.g. Hazards Analysis and Risk Assessment (HARA) to be released and remain unchanged for the remaining duration of the Project. What is required to be done is to baseline HARA at the end of concept phase and keep it within the scope of the Project's configuration and change management. Once the SFMEA (System Failure Mode Effects Analysis) is performed, to which outputs of HARA are an input, the SFMEA may uncover new failure modes not earlier known and hence not considered in the HARA – so the HARA will have to be updated. Later on, when supplier of a component within the Item boundary is performing DFMEA of their components, they identify new failure modes which may require the HARA to be updated and vehicle level effects of failure modes of this component to be analysed again because this is new information that was not available at the point in time the first version of HARA was performed. The ISO 26262 standard also refers to this as *refined XXX where XXX is a work product – it could be HARA, Safety Plan, FSC (Functional Safety Concept) etc.*

4.4 This Project has only mechanical changes – so no need to follow ISO 26262!

Taking the example of a new truck series production Project where only the Cabin (cab) external and interior dimensions are changing to make it more aerodynamic, thus reduce drag and CO2 emission and increase internal space for the cab occupants to work and move around. All E/E components are carried over from previous Truck Project with no E/E changes. If we ignore any electrical routing changes, one could be easily tempted to assume that there are no ISO 26262 activities required to be performed in this Project. Unfortunately, this is not true for two reasons:

a) As per ISO 26262:2018 Part 3, Impact analysis is mandatory for *all* Projects and results of this activity must be confirmation reviewed according to I3[5] independence as per ISO 26262:2018 Part 2 *even for ASIL (Automotive Safety Integrity Level) rating of QM* – this is easy to miss. *This ensures that the Impact analysis has been correctly performed by involving all relevant stakeholders and SME's to correctly assess the impact of change or newness of this Project compared to existing ones and scope of ISO 26262 can be correctly applied - aligned with the scope of the change or newness of the Project compared to the previous one.*

b) Changes to cab structure requires re-calibration of the Passive Safety Air-Bag deployment criteria due to its sensitive nature. *This requires very expensive safety validation (crash) tests to be performed on multiple Trucks as per ISO 26262 Part 4 as well as comply with ISO 26262 Part 6 Annex C related to Software Calibration data and revalidate the SG's because previous safety validation on the older Truck model is no longer valid.* The very high costs of destructive tests to be performed on many Trucks and many man months of efforts could have been easily missed out had the right people not be involved while performing the Impact analysis.

4.5 What does QM (Quality Management as referenced in ISO 26262:2018 Part 1) really mean?

QM does not imply any procedures that your organization has currently implemented. On the contrary, it expects the minimal systems engineering and ISO 9001 processes to be in place with the addition of special disciplines applicable

[5] I3 is the highest level of independence required by ISO 26262 standard

to the product being developed. In the context of Automotive E/E systems, this implies ISO 9001:2015 with automotive specific addition of ISO / TS 16949[1], ISO 15288 (Systems Engineering), ASPICE (Automotive software development) and special disciplines such as ISO/PAS 21448:2019 (SOTIF), ISO 21434:2019 Cybersecurity to name a few. Other ISO standards for EMI/EMC, REACH regulations, UN ECE Regulation 100 for Electric Vehicles will also have to be complied where applicable. A robust Quality Management System (QMS) that embodies all these relevant standards and defines processes that are optimized for the context of their respective organisations provide a very solid foundation for systems engineering. It then becomes fairly easy to design functional safety from start of product development rather than trying to shoe-horn it in when it is already late and this significantly increases the likelihood of SG violation. ISO 15288 is not explicitly referenced in ISO 26262 - this recommendation is based on my personal experience. Embodying it appropriately within the product design & development process will integrate the necessary systems engineering discipline across the organisation - a prerequisite to integrate functional safety into products that customers actually want and are delivered on time and on budget.

5 Six steps to become ISO 26262 compliant

a) Align Roles, Responsibilities and Competencies of people in the Organisation with work products and processes defined in a refined Quality Management System (QMS) that includes unambiguous RASIC.
b) Update the QMS to implement a top down Holistic Systems Engineering approach optimised to the context of the Organisation, Department and Project and regularly monitor that it is complied with – this lays the foundation to inherently design safety within the product.
c) Secure a complete understanding of not only how a system behaves, but also how it fails, perform safety analyses and incorporate these learning in the system design, safety measures and safety mechanisms at various levels. This should be integrated in the existing systems engineering / product design & development process.
d) Design Functional Safety upfront during Product Development. This includes making a Safety Plan, effectively implementing the concept phase (OEM) or SEooC (safety Element out of Context as defined in ISO 26262: 2018 Part 1) approach (Tier n supplier) and tailoring the QMS to the scope of the Project. The Safety Plan should include how Safety Case will be constructed and identifying the Development Interface Agreements (DIA's) required to be created and signed off to deliver a validated Item integrated within the vehicle.

e) Create DIA's identified in the Safety Plan encompassing all stakeholders delivering the "Item". Things to consider include RASIC for the 106 unique work products listed in the ISO 26262 standard.
f) Work with a Coach who can also be your Mentor – a competent person who can not only help you reach your strategic goals but also help with accomplishing short-term objectives along the way.

6 Conclusion

Complying with ISO 26262 is no longer a choice. It is only a question of when, with the costs of quality rapidly increasing with the time to implement. It is noteworthy that despite release of ISO 26262 first edition in Nov 2011; both the costs of fixing failures and frequency of suppliers named in vehicle recalls has increased by three-fold and two-fold respectively since 2013 compared to the 2007 to 2013 period as mentioned in section 3.2. In the context of rising E/E content in vehicles mentioned in section 2.2, this could imply that the increased E/E content increases opportunities of E/E failures and introduction of ISO 26262 standard since the first edition in Nov 2011 has either not been effective in reducing Functional Safety related E/E failures due to inadequate implementation or will take more time to impact all E/E applications and majority of the automotive OEM's and suppliers, not taking system related failures into account. Any deterministic conclusion will require more analysis and capturing the required information of E/E failures to effectively categorize them. The STAMP Framework to analyze Automotive Recalls is a good step in this direction (Hommes, 2014) and comprehensive data for a conclusive analysis could be available in the future.

Practical experience cannot be substituted with theoretical training. One could easily get tempted to just read the standard, go through classroom training and apply it straight away on real-world practical projects and get caught in myriad of trying to understand the real meaning of what is written and how it is applied in practice.

There is a light at the end of the tunnel. Have a holistic view and integrated approach of the overall product development, systems engineering and functional safety for the entire life cycle of the product, carefully select and take a Coach who can also Mentor you in the long term and tread along with you in this journey of achieving repeatable functional safety in products that can, as a spin-off, also result in better quality, on time delivery, significant reduction of vehicle recalls, warranty and liability claims and minimize, if not eliminate, non-safety related failures. A coach and mentor can help you avoid very expensive mistakes people have already done in the past and help you get it right the first time.

Acknowledgments My sincere gratitude to Mr. Devansh Mehta, Functional Safety Manager & Subject Matter Expert - ISO 26262 Competency Centre, Volvo Trucks AB, Gothenburg,

Sweden and Mr. Michael S Parsons of Safety Critical Systems Club, UK for taking the time and promptly reviewing this paper prior to publication at short notice despite their busy schedule.

Disclaimers These are authors personal views and recommendations based on his experience. The author or the organisation he represents do not take any responsibility or liability for the impact of implementing any of the proposed actions, suggestions or recommendations mentioned in this paper in the organisations you represent or in your personal businesses or in any other way. They also do not accept responsibility or liability of correctness of information or data contained in any of the references cited or in any way endorse or recommend their authors, publishers, organisations or copyright owners.

References

Burrows, D (2018) 10 Biggest Product Recalls of All Time. Kiplinger https://www.kiplinger.com/slideshow/investing/T052-S000-10-biggest-product-recalls-of-all-time/index.html. Accessed 24 Nov 2019

Deloitte (2019) Semiconductors – the Next Wave Opportunities and winning strategies for semiconductor companies. https://www2.deloitte.com/content/dam/Deloitte/cn/Documents/technology-media-telecommunications/deloitte-cn-tmt-semiconductors-the-next-wave-en-190422.pdf. Accessed 24 Nov 2019

Department of Defense (2012) MIL-STD-882E.

Hommes, QVE (2014) Applying STAMP Framework to Analyze Automotive Recalls US Department of Transportation, Volpe Center

Hilderman V Top 15 ISO 26262 Snafus How to Prevent & Mitigate the Mistakes Teams Make. Jama Software White paper https://resources.jamasoftware.com/paper/top-15-iso-26262-snafus. Accessed 25 Nov 2019

International Standards Organization (2011) ISO 26262:2011 Road vehicles - Functional safety.

International Standards Organization ISO 26262-1:2018 Road vehicles - Functional safety - Part 1: Vocabulary.

International Standards Organization ISO/PAS 21448:2019 - Road vehicles — Safety of the intended functionality.

Jibrell A (2019) Auto recall bill grew 26% to $22 billion in 2016, study says. Automotive News https://www.autonews.com/article/20180130/RETAIL05/180139974/auto-recall-bill-grew-26-to-22-billion-in-2016-study-says, Accessed 25 Nov 2019

Kent G (2019) A year after the first 737 Max crash, it's unclear when the plane will fly again. c|net https://www.cnet.com/news/best-early-black-friday-2019-deals-available-today-199-ps-or-xbox-airpods-pro-discounts-and-more/. Accessed 24 Nov 2019

KNKT.18.10.35.04 (KOMITE NASIONAL KESELAMATAN TRANSPORTASI CHAIRMAN) (2019) Aircraft Accident Investigation Report http://knkt.dephub.go.id/knkt/ntsc_home/ntsc.html. Accessed 02 Dec 2019.

Magazine (2010) How computers took over our cars in BBC News http://news.bbc.co.uk/1/hi/magazine/8510228.stm Accessed 02 Dec 2019.

Reuters (2010) Toyota to recall 436,000 hybrids globally-document. Metals News 1 MIN READ https://af.reuters.com/article/metalsNews/idAFTKG00664220100209. Accessed 24 Nov 2019

Squires A, Fairley D, (2019) SEBoK. https://www.sebokwiki.org/wiki/Safety_Engineering. Accessed 24 Nov 2019

Utilising MBSE for Safety Assurance of COTS devices with embedded software

Waleed N Chaudhry[1]

EDF Energy Nuclear Generation Ltd.

Abstract *Commercial off the shelf (COTS) devices with embedded software offer flexible and wide-ranging benefits recognised from technological advancements. Their use in nuclear safety systems has become prevalent but this has come with a difficult challenge for safety assurance. These new devices are complex and restricted access to evidence from the product developer to support a functional safety audit can make their justification in safety-critical systems difficult. This paper presents a novel nuclear safety justification strategy termed 'Model Based Safety Assurance' (MBSA), which requires less invasive questioning and is thus less resource intensive for the developer. It uses concepts from Model Based Systems Engineering and applies them in the context of safety assurance, to achieve qualification of COTS devices for use in safety systems. The strategy utilises established techniques for software development (e.g. Model Based Testing) but ex-tends their scope to support safety assessments. The paper also discusses the advantages and limitations of MBSA compared with the traditional safety demonstration approach currently used by the civil nuclear industry. Finally, with the help of a case study (based on a real system), it seeks to demonstrate the strength of the approach when combined with software safety assurance techniques such as Statistical Testing and Goal Structuring Notation.*

1 Introduction

Using Commercial off the Shelf devices (COTS) in safety systems provides substantial commercial and technological benefits which have been well utilised within the nuclear industry *EPRI (1996)*. Low cost instrumentation and control (I&C) equipment available from multiple specialist suppliers is usually the preferred choice for nuclear power plant designers, as it comes with advantages such as reduced in house design/development and reproducibility in multiple applications. Since the last British nuclear power plant was commissioned in the 90's, these COTS devices have seen substantial improvements as their designs have

[1] waleed.chaudhry@edf-energy.com

© EDF Energy Nuclear Generation Ltd. 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

evolved (see figure 1 for example). One such technological improvement is the replacement of electronic hardware with embedded software, which provides the benefit of improved functionality at reduced costs. This has however resulted in highly complex designs for even the simplest safety functions, consequentially increasing the potential for systematic faults that could result in failure of safety functions fulfilled by the I&C. International standards such as *IEC61508* provide sufficient guidance on techniques and measures that should be put in place to reduce systematic failures introduced from the design of programmable electronic devices intended for use in safety systems. The popularity of such standards has also meant there are numerous I&C products compliant with these processes to choose from in the market. However, for industries such as nuclear which need to fulfil *'Intelligent Customer'*[1] requirements, these standards do not provide the required safety assessment structure to evaluate such products.

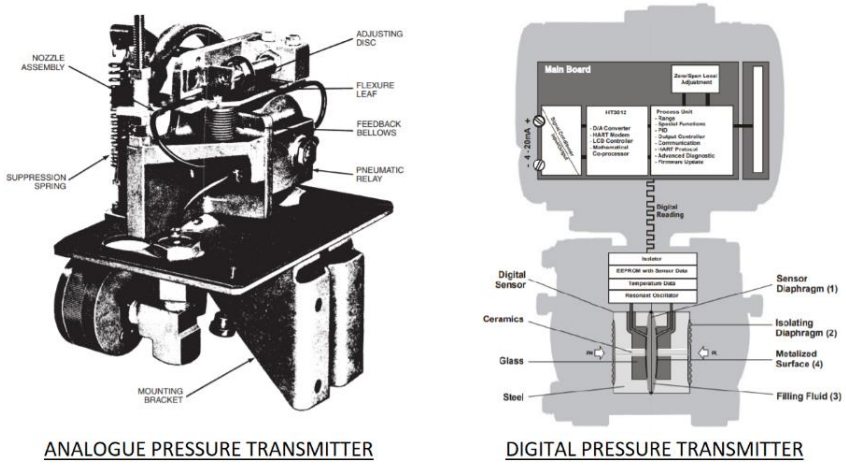


Fig. 1. Analogue vs Digital Instrument Design[2]

[1] Nuclear licensees in UK are required to demonstrate they are 'Intelligent Customers' i.e. '…an organization (or individual) that has the competence to specify the scope and standard of a required product or service and subsequently assess whether the supplied product or service meets the specified requirements' IAEA (2011).

[2] Information from product manuals of Ashcroft Series 4000 and Smart LD-400 pressure transmitters has been reproduced in figure 1.

The nuclear industry's current practice to fulfil the 'Intelligent Customer' requirements is a two-legged SMART3 device qualification structure, which is described in section 2 below. Briefly described also is the Emphasis methodology which is used to demonstrate confidence in the first leg. The current methodology is adequate for the purpose of safety assurance but comes with a number of challenges, which are described in the same section. Section 3 focuses on the alternative approach, which is first introduced as Model Based Safety Assurance4 (MBSA) and then described using an example COTS device. Section 4 demonstrates how a safety case could be put together using Model Based System Engineering (MBSE) techniques and then as a conclusion, challenges and pitfalls of the approach are discussed.

2 The current approach

An illustration of the nuclear industry's current approach is described in Office for Nuclear Regulation's (ONR) nuclear safety technical assessment guide for computer-based safety systems, *ONR (2019*). It recommends the multi-legged approach utilising Production Excellence (PE) assessments and performance of Independent Confidence Building Measures (ICBMs). The principles are described in the aforementioned ONR document, and below is an interpretation of its objectives as related to embedded software devices. These objectives are used as the basis for utilising MBSE to satisfy the multi-legged approach.

Production Excellence
In assessment terms, this involves wide ranging and searching audits of documentation related to the specification, design, development, integration and validation of the device to ascertain high quality production. This assessment provides confidence that the impact of random hardware faults has been considered and that measures have been taken to minimise residual systematic faults as appropriate for the safety function reliability. PE assessments are intended to demonstrate that the:

1. device satisfies the requirements set out during its specification stage (or those claimed on its technical literature)

[3] SMART instruments are defined by the nuclear industry as electronic units, usually COTS, which contain intelligence in the form of a microcontroller that is not programmable by the end user.

[4] Within this paper, Model Based Safety Assurance is a SMART device qualification regime that makes use of MBSE tools and techniques. The device end user utilises it to construct a safety justification by substantiating the use of a SMART device in a safety-critical system application.

2. failure modes of the device are understood and mitigated to negate any impact on the device functionality
3. device development is adequate for the safety function Category and associated equipment Class (as defined using nuclear standards *IEC61226 (2010)*) OR Safety Integrity Level (SIL) (defined using functional safety standards *IEC61508 (2010)*)

The nuclear industry has developed a methodology termed *Emphasis* to demonstrate the PE of a SMART device. This is essentially an assessment of a SMART device using question banks targeted at all lifecycle stages of a V-model according to the SIL claimed. It bears much resemblance to *IEC61508 (2010)*, with questions developed using recommended tools and techniques for programmable electronic system development within that standard. It is supplemented with questions derived from nuclear industry best practice. A SMART device manufacturer is requested to provide responses with supporting evidence to each of the questions which are related to manufacturer's quality management processes, detailed hardware/software designs, validation/verification techniques, cybersecurity etc. The responses and evidence are then reviewed by a team of *Emphasis* assessors who judge its adequacy for the SIL and then discuss them with the manufacturer's engineering team during an audit. They then report on any gaps against the *Emphasis* guidelines and suggest compensatory measures as required. The compensatory measures range from performance of independent design reviews to static/dynamic analysis of source codes. *Emphasis* is a very thorough analysis of each lifecycle phase of the device. However, this comes with a resource and monetary burden for both the nuclear licensee and Original Equipment Manufacturer (OEM) (who are most likely to have been through a similar assessment for SIL qualification). As such, use of *Emphasis* poses the following challenges, which we seek to address through MBSA:

1. Personnel who perform *Emphasis* assessments need a broad range of knowledge to allow them to judge adequacy of the hardware, software, testing regimes, quality assurance and reliability of the embedded software devices
2. The OEM has to make available significant time of their product developers to support evidence provision. This means lost time in product development and thus an "unknown" cost penalty
3. *Emphasis* is used to assess documentation which already exists and which may have been reviewed already by independent functional safety assessors thus duplicating work
4. OEMs are concerned about making their intellectual property available to third parties

Specifically, for points 2, 3 & 4 above, manufacturers can be unwilling to provide the required support for an *Emphasis* assessment. Where this support is provided, it tends to come with an associated cost to the nuclear licensee. High assessment costs coupled with the need for multiple devices can result in engineering misjudgements within safety justifications e.g. the high costs favour use of devices without software and consequentially lack of functionality provided by most modern devices (such as self-diagnostics).

Independent Confidence Building
This is the performance of techniques and measures independently of the device production process and of the OEM to scrutinize the device and its configuration and to demonstrate high quality in the outcome of the production process, i.e. the product itself. In the ICBM leg, confidence needs to be evaluated by utilizing diverse methods from those used in the PE leg. ICBMs are often application specific and are designed to demonstrate:

1. Device satisfies the requirements of the safety system within which it is incorporated
2. Failure modes of the device are understood and mitigated to negate any impact on the system safety function
3. The confidence gained in the quality of the production process through the PE leg remains unchallenged through insights gained from performing the ICBMs

Device type testing, acceptance testing, independent tool reviews and reliability analyses are some examples of ICBMs employed by the nuclear industry.

3 Utilising MBSE

The International Council of Systems Engineering (INCOSE) defines MBSE as *"the formalized application of modelling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases." INCOSE (2007).* Widely used in systems engineering, visual representations of a given system are produced in a general-purpose modelling language such as OMG Systems Modelling Language (OMG SysML) which itself is based on a subset of Unified Modelling Language (UML) with engineering specific extensions. Specific modelling languages for systems engineering provide benefits over their software-centric UML type counterparts (such as requirements modelling, automated Verification & Validation (V&V) etc.). However, their primary goal remains design visualisation of a system to support various lifecycle phases

of a system's development. For safety critical applications, their most important benefit perhaps is the means for providing structured and traceable development from requirements into design and finally complete validation of a system. These are the benefits we will seek to use in MBSA for satisfying the objectives of the nuclear industry's multi-legged SMART device qualification approach.

3.1 An alternate Safety Assurance concept

Contrary to the traditional thinking in safety assurance, we start by assuming that the COTS device for assessment has been developed using a recognised development lifecycle. The V-model development lifecycle shown in figure 2 is used to structure the assessment of the COTS device. In reality, the development lifecycle may be completely different. However, if the device was developed using best practices to ensure minimisation of residual errors, it should be possible to retrospectively and successfully perform activities relevant to each lifecycle phase and use the outputs to evaluate its strength. For example, parallel safety assurance activities (as shown in figure 2) can independently verify that the system specification, development and validation meet the required standard expected of a safety critical system.

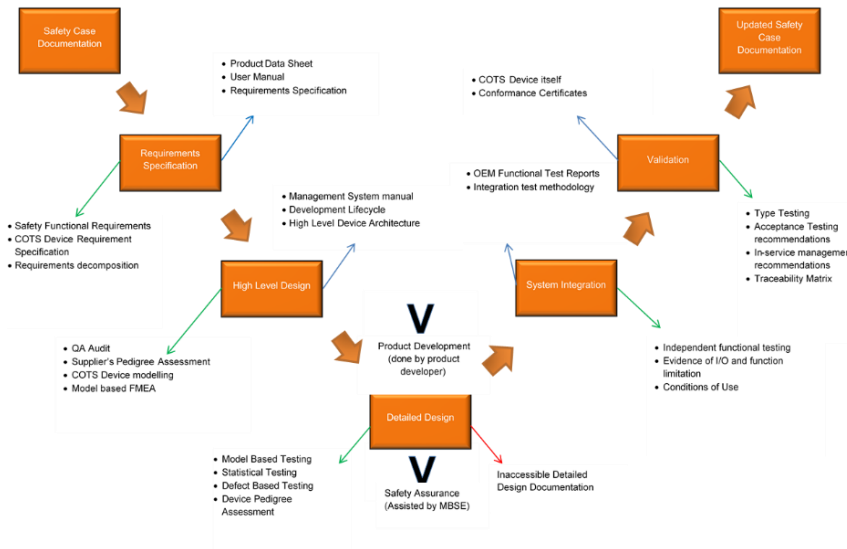


Fig. 2. Adapted V-model for Model Based Safety Assurance

Carrying out these safety assurance activities also confirms our original assumption above as well as providing a confidence level of the integrity of the supplier's processes. One can thus assess the PE of the COTS device without detailed evidence from the OEM, and through independent means thus providing some benefits over the *Emphasis* approach. Furthermore, the automation aspects within MBSA can provide significant safety assurance benefits. For instance, a process step that was previously undertaken by a human still has human assurance over it at some point in the lifecycle. In the case of MBSA, both the automated step has to be wrong and human oversight has to fail to identify gaps and errors in the development phase.

Retrospectively taking a device through a V-model would be more or less reverse engineering it. Of course, this would mean substantial costs and if one were to implement it, why not manufacture a bespoke device for your application? This is where MBSE helps as the device can be taken through a V-model development lifecycle in a modelling environment thus drastically reducing the associated costs. In fact, the models only need to be simplified abstracts encompassing important properties of the device and their fidelity adjusted according to the level of assessment required for a particular safety application. Section 3.2 shows how specific techniques from MBSE can demonstrate confidence in each of the lifecycle stages. There are limitations and challenges of this approach. Some of these are mitigated through support by other techniques as shown in the safety argument presentation in section 4. The remaining are summarised under section 5 with the conclusion.

3.2 MBSE supported Safety Assurance

This section demonstrates how MBSE techniques can be used to provide confidence in some of the V-model lifecycle stages of a COTS device development. A Pressure Transmitter (digital version from figure 1) is used as a simplified example which is partially reconstructed in SysML. *OMG (2015)* provides information on SysML and Fig. 3 below shows which aspects of SysML are demonstrated in this paper). The limited modelling has been done using publicly available information from the OEM to demonstrate how the techniques can be used with limited or no support from the OEM. It is worth noting here that MBSA is not used to replace or compensate for other techniques that should be performed in a lifecycle phase. They are used as safety assurance techniques to provide confidence that adequate measures were employed by the COTS device developer to minimise potential failures.

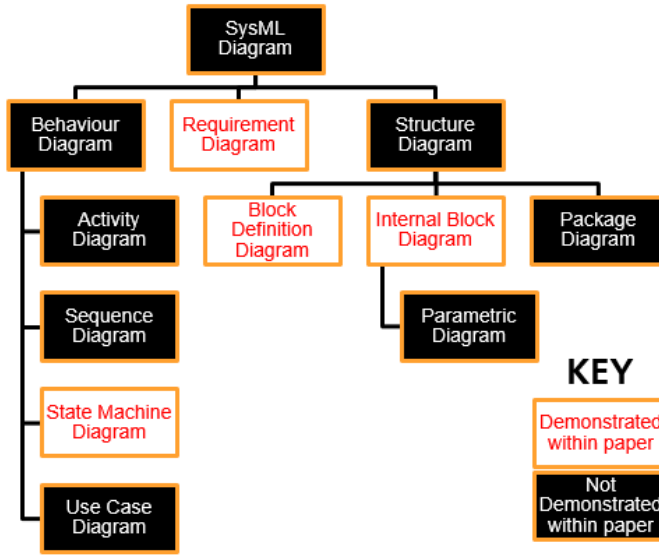


Fig. 3. OMG SysML diagram types demonstrated in this paper

The approach that we describe here is to use a set of modelling activities making use firstly of information available on the device in the public domain or via easily accessible information from the OEM and secondly supplemented by discussions with the OEM.

Within this paper, we only go into limited detail of how individual steps can be achieved. We aim to outline the principles, tools and methods employed and provide examples of the outcome of these steps. We expect to provide further technical details in proceeding academic work on the topic.

Note: For simplicity, the lifecycle phases have been limited to only three i.e. Requirements Specification, Design and Validation.

3.2.1 Requirements Specification

As with any Safety Critical System (SCS) application, the first activity is to determine the safety functional requirements (SFRs) of the SCS. Once the SCS requirements are specified and a COTS device identified for use within the SCS, the next step is to ensure that the COTS device can fulfil each requirement. SysML provides built in requirements management functionality. *Roques (2015)* provides a good description of how SysML can be used to specify and trace requirements. For our purpose, the relationships created between the requirements

and other model elements provide a powerful representation of the system requirements decomposed into the COTS device. This is done by first breaking down the SFR into individual requirements for the COTS device. Figure 4 is an example of how the pressure transmitter requirements can be modelled (we have utilised the requirements diagrams within SysML to achieve this). We first model the two SFRs for our SCS and then using the 'derive requirement' feature of SysML, link them to the COTS device requirements. The derived requirement relationship can thus be used to systematically ensure each SFR is satisfied by a function of the COTS device. More importantly however, it allows a formal link to be made between the SCS and COTS device in the modelling environment. This also provides a visual representation of SFR satisfaction by the COTS device thus ensuring completeness of COTS device requirements against the SFRs. Furthermore, the link between the two is used in the next phase of the modelling to ensure the traceability into structural and behavioural elements of the COTS device (demonstrated in section 3.2.2 figure 5).

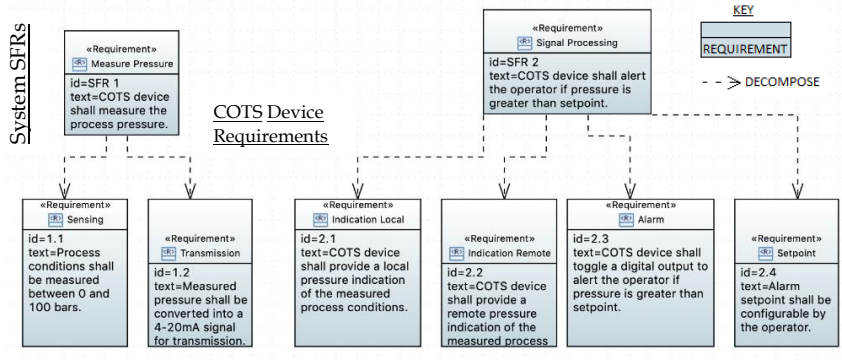


Fig. 4. Example Requirements Modelling in SysML

Using the Pressure Transmitter example, the COTS device requirements were derived from the product datasheet and modelled using SysML. Even with the limited review of the product datasheet, it was found that the COTS device offered more functionality than was required for our SCS application. Although not shown in our limited example, for a real application the unwanted functionality could be modelled and would remain 'disconnected' from the SFRs. We limited ourselves here to addressing only those device requirements that directly relate to the SRFs. However, when developing the structural and behavioural models it could demonstrate how the unwanted functionality impacts the SFRs themselves e.g. the pressure transmitter provides 3-term control functionality which quite possibly uses the same central processing capability as the display and trip functions associated for safety. From the requirements model the unwanted functionality may look simply superfluous but through analysis one can evaluate impacts on the SFRs and justify non-detriment arguments.

For our objective of COTS device safety assurance, within this phase, the requirement aspects of MBSE can thus be used to:

- Specify safety functional requirements of the safety critical system
- Model functionality provided by the COTS device
- Trace the SFRs into COTS device functions using requirement derivation
- Identify unwanted functionality offered by the device with respect to SCS
- Provide structure for detailed analysis to be performed and traced back to their effects on the SFRs

3.2.2 Design

Whilst developing a COTS device in the design phase, the developer tries to ensure the device fulfils all requirements identified in the specification and sufficient mitigation exists for ensuring they cannot fail because of credible failure mechanisms. In order to provide confidence in the design phase, the safety assurance task is thus to ensure independently that all requirements are in fact met and no perceivable failure mechanisms (specifically those related to the SCS application) can cause their failure.

In order to achieve this, we will continue our system modelling by developing structural and behavioural models. These will both lead to identification of failure modes and relevant test cases.

We start by developing a *'structural model'* of our COTS device i.e. describe its internal structure in terms of its parts, ports, connectors etc. Figure 5 shows the pressure transmitter example that has again utilized information from the product manual.

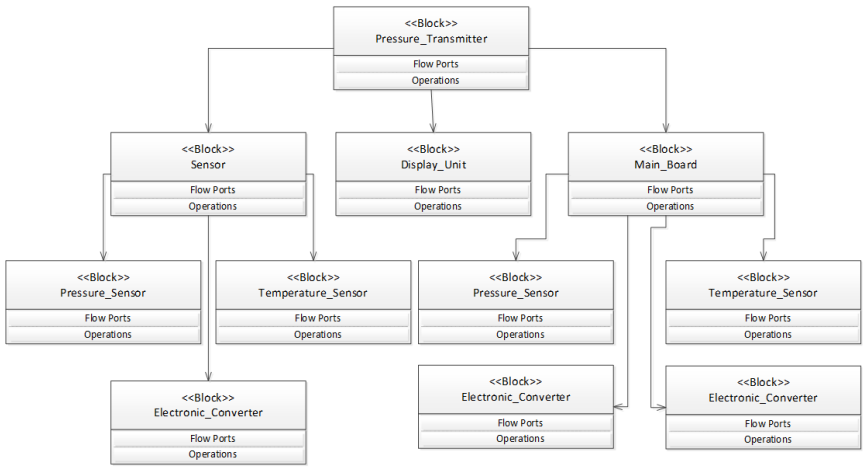


Fig. 5. Example SysML Structural Diagram

The finished structural model allows requirement decomposition i.e. each of the requirements modelled in the previous phase can now be linked to individual structural elements of the device. The granularity to which this can be done is dependent on the information made available by the OEM as well as the technical expertise of the modeller.

Figure 7 shows the internal block diagram (IBD) of the Main Board in the pressure transmitter example. This is an example of the next level of detail that could be achieved. However, as is obvious from the example, the greater the fidelity the more assumptions we start to make about the device. This is perhaps one of the most challenging aspects of this strategy as one may have to carefully manage where direct/indirect inputs from the OEM are required whilst maintaining the independent modelling of the device[5]. For instance, we need to discuss assumptions made during the device modelling with the OEM developers to ensure the model is a close enough representation of the COTS device. However, we may also need to maintain sufficient independence to ensure effective Safety Assurance. For this reason, we have introduced the two staged modelling, approach described later in this section.

The action of modelling provides a systematic way of understanding how each requirement is fulfilled. These requirements are also associated to specific subsystems of the COTS device, highlighting any assumptions or queries related to the technical aspects of the design itself. This is further complemented by a guide word assisted Failure Mode and Effects Analysis (FMEA) of the structural model. Figure 7 shows an example of the pressure transmitter IBD supplemented by a brief FMEA. Rather than actions for the design, we have introduced test cases within the FMEA table. These are actions that can be undertaken to verify that the failure mechanism will not affect the safety functionality. Some of these can be obvious tests such as the impact of I/O module disconnection. Others, which cannot be tested by the end user, would have to be discussed with the COTS device developers (or seek evidence from them, such as unit test records). It is recommended that all queries and assumptions from the modelling and test cases from the FMEA are recorded for discussion with the COTS device developer, as they could be key to recognizing false positives in the validation stage. Furthermore, positive discussions informed through the modelling would also enable an engineering judgment to be made about the quality of the development. For example, if the modelling identifies a failure mechanism related to inadequate I/O connections, it would be reasonable to expect that the developers have already considered it and are able to demonstrate its mitigation in the design or provide a reasonable explanation for not doing so.

[5] Independence between the COTS device developers and safety assurance modellers is useful as an ICBM. It has some benefits to offer in the PE leg. However, it is unknown if the gains are proportional to the cost and effort of maintaining independence. This is one of the topics of our current research on MBSA.

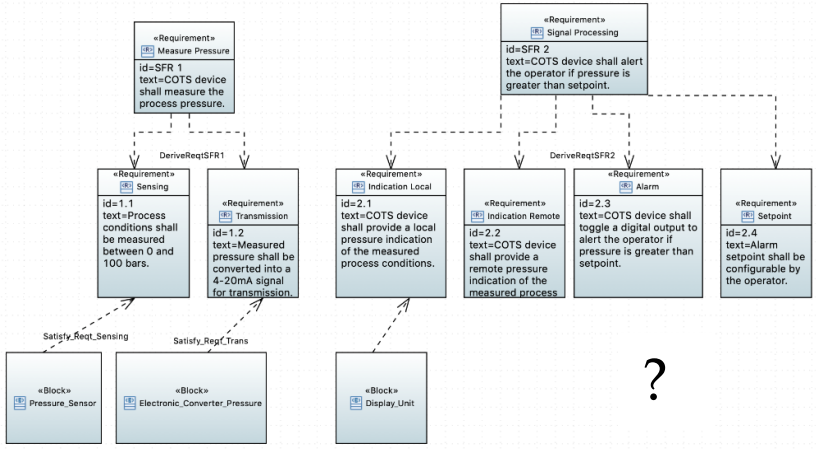
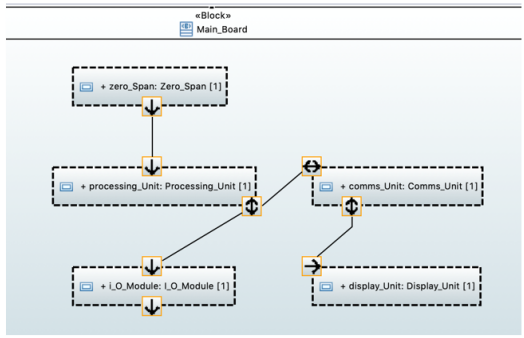


Fig. 6. Example Requirements Decomposition in SysML

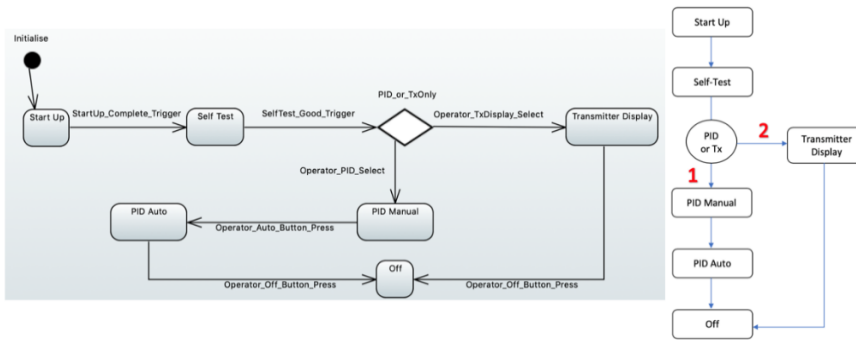
Following on from the structural models, we start developing "*behavioural models"* of the system. As it is the desired behaviours (and conditions under which they are exhibited) of the components within the COTS device that are directly related to the SCS functionality required, it is important they are modelled, analysed and verified. SysML allows expression of these in the form of Use Cases, Activity Diagrams, Sequence Diagrams and State Machine Diagrams. In safety product design and development, models such as state machines and state transitions are used to achieve design completeness, consistency, reachability and absence of endless loops (*IEC61508* Part 7). For the safety assurance model, the objective is to achieve the same for the model of the COTS device. The initial input to this stage is expected to be the product manual to understand the operational modes, their entry/exit conditions and the effects they have on the safety functions. As with the structural modelling, numerous assumptions and queries are logged followed by a FMEA exercise. These are used in a similar manner to structural analysis, i.e. assist discussions with COTS device developers. Figure 8 shows a state machine for the pressure transmitter example, including determination of two test cases that can validate the Display and 3-term control functionality.



Guide Words
No - Port is missing
As well as - Signal allows wrong signal type to flow
Other than - Does not allow correct signal type to flow
Part of – Does not allow all required signals to flow

Item	Guide word	Issue	Cause	Effect	Detection	Impact on SF	Test Case
I/O to CPU port	No	No input to CPU	Physical port wear	Pressure not measured	Self Diag. alarm	Yes	Disconnect I/O module
	Part of	Unknown	Design Error	Unknown	Unknown	Unknown	Discuss with OEM

Fig. 7. Example SysML Structural Diagram with FMEA



#	Test Case	Acceptance Criteria	Impact on SF	Comments
1	PID Select	Display PID Manual Display PID Manual Switch Off	Unlikely as function not used	Assess non-determent to SF
2	Tx Display Select	Display Pressure Reading Switch Off	Failure results in loss of SF	None

Fig. 8. Example SysML Behavioral Diagram with Test Cases

The model and the COTS device are still two very different things. In order to ensure the model provides an adequate abstraction of the COTS device, there needs to be a level of input from the COTS device developer. Some of this can be achieved by using the requirement specification of the COTS device (from the OEM), as well as technical literature about the device (e.g. product manuals, data

sheets etc.). This maintains the necessary independence from the COTS device developers and generates a set of manual test cases and technical queries. However, in order to ensure the model is a reasonable abstraction of the COTS device, the next step is to discuss the models, test cases and technical queries with the COTS device developers. Noting the challenges from *Emphasis*, the OEM may not be forthcoming to discuss their Intellectual Property. However, the targeted set of queries and models produced independently could be much more attractive to the OEM, as it provides assessment of the device through independent means (thus not duplicating work). Furthermore, it does not require the same level of evidence provision and is less of a resource burden for the OEM's developers. It is thus recommended that once models are finalized and the set of initial queries and test cases produced, they are discussed with the OEM to ascertain confidence in the model's representations of the COTS device. Before the validation phase, they would need to be further informed based on the discussion findings. Figure 9 below shows how the different activities performed bring the model closer to the COTS device.

For our objective of COTS device safety assurance, within this phase, aspects of MBSE can thus be used to:

- Develop an understanding of the behaviours (COTS device states, entry/exit conditions, sequence of activities etc.)
- Develop sufficient understanding of the internal structure of the COTS device
- Analyse failure mechanisms and their effect on the COTS device functions
- Analyse unwanted functionality and their effects on the SFRs
- Provide a structure for informed discussions with the COTS device developers
- Develop a set of test cases which can be used to gain confidence in the device PE or independently validate SFR aspects to be fulfilled by the COTS device which is application specific

3.2.3 Validation

Following on from the modelling in the Requirements and Design phase and discussions with the COTS device developers, one would have a simplified yet well-informed model of the COTS device along with a set of manually generated test cases. The model should contain all aspects effecting the SFR and the test cases would have started as queries/assumptions and/or failure mode analyses of the various model elements. For a developer, the objective of the validation phase is to generate the necessary evidence, which shows that each specified requirement has been met by the design. For the safety assurance, the objectives remain very

similar except that we use the following inputs to drive testing of the actual COTS device:

- Requirement specification of the COTS device (if using in PE leg)
- SRS for the SCS (if using in ICBM leg)
- Test Cases generated from the failure mode analyses
- Test Cases based on model assumptions and technical queries
- Test Cases generated using automated test case generation (ATCG) tools

After iterations and finalization of the model, further development of test cases using ATCG tools (such as Conformiq Designer described in *Conformiq (2011)*) is desirable. These should be run against the COTS device requirement specifications (if used in PE leg) and SRS for SCS (if used in ICBM leg) as the inputs and determined using structural and behavioural models. It is expected these would return a large set of test cases, which could be costly, and some unpractical for the assessors to perform. However, considering the model now is a realistic representation of the COTS device, it would be reasonable to expect the OEM considered the test cases during the device development. Evidence from the OEM in the form of test records or design analysis addressing the ATCG findings would thus provide a good level of confidence in the PE leg. For ICBMs, the test cases generated are against SCS requirements (which should be testable!) thus, the same issue does not generally apply.

Use of ATCG is seen as an important aspect of this strategy as it not only provides completeness to the validation but also correctness of the model itself (*Conformiq (2011)*). Like the fidelity of the models use of ATCG, the test coverage offered or the number of tests actually run on the COTS device can all be scaled based on the integrity requirements of the safety function.

Essentially, use of model based testing is being suggested to validate COTS device or SCS requirements. The results of this testing should help evaluate confidence in the PE leg or generate evidence for the ICBM leg. For instance, if the tests pass and sufficient independence has been maintained between the COTS device developer and system modeller, it shows a device of reasonably good PE. However, if the tests fail, it demonstrates either poor PE or a lack of understanding about the device by the modeller. Considering the *'intelligent customer'* requirement discussed in section 2, both outcomes indicate gaps exist which need to be addressed before the device can be or should be used. In case of ICBMs, test failures show the device cannot perform the required safety function and thus should not be used in its current configuration.

For our objective of COTS device safety assurance, within this phase, aspects of MBSE can thus be used to:

- Independently gain confidence in the COTS device developers design and production processes

- Validate any assumptions made during the modelling
- Generate the necessary testing evidence to demonstrate all SFRs can be fulfilled by the COTS device
- Generate the necessary testing evidence to demonstrate adequate mitigations are in place to mitigate failure mechanisms identified through modelling analyses

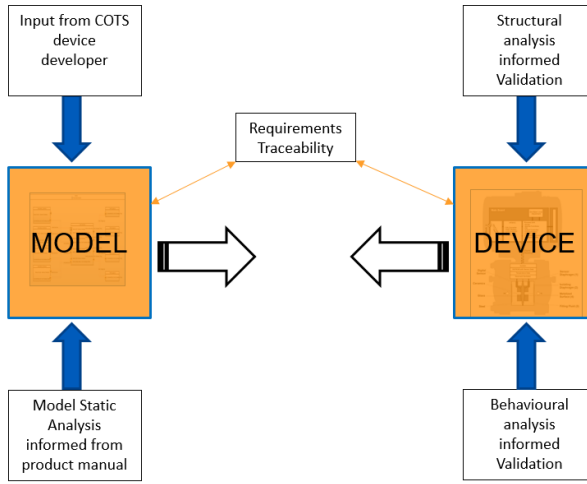


Fig. 9. Informing the model for better abstraction

4 Presenting a safety argument

This section suggests how a safety argument for a COTS device with embedded software to be used within an SCS can be presented. It is done using the Claims, Arguments and Evidence notation and seeks to justify the Claim 'The correct device has been selected for the application'. It is expected that this Claim used in conjunction with other safety claims of the system would form the safety case of the system incorporating the SMART device.

Software aspects of computer based systems performing Category B/C nuclear safety functions are identified in *IEC62138 (2009)*. The standard sets out the requirements for selection and use of dedicated devices with embedded software. Below, these are translated into Arguments to derive the necessary evidence required for the SMART instrument justification of use.

The approach used is to carry out a set of safety assurance activities that complement each lifecycle phase expected from a safety development lifecycle of a SMART instrument. The activities have been derived (as evidence) from tools

and techniques within *IEC61508* Part 7 focussing specifically on MBSE techniques. Commentary is provided to demonstrate how the evidence can be used to satisfy the Claim and Arguments identified and as a result show compliance with the aforementioned *IEC62138* requirements. Figure 10 is a pictorial representation of the example safety argument presented below.

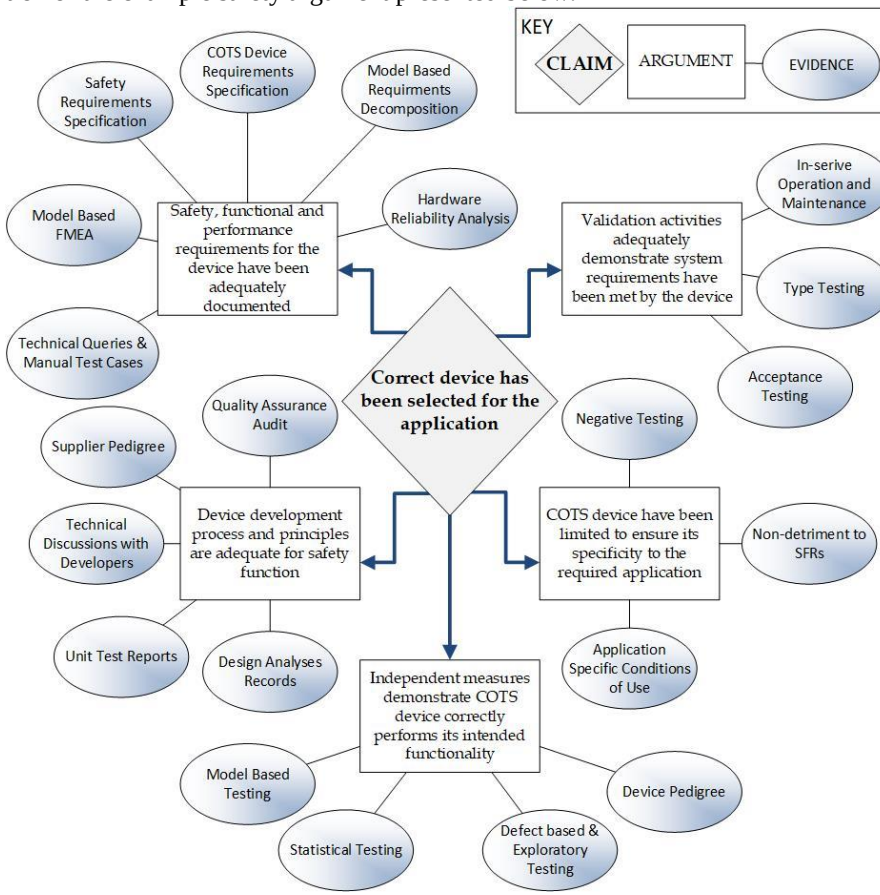


Fig. 10. CAE structure for Model Based Safety Assurance

Claim A – The correct device has been selected for the application

Argument A.1 – Safety, functional and performance requirements for the device have been adequately documented

This argument should seek to justify the following so that the end user can use this information in selection and application of the COTS device.

- The safety functional, interface, performance and hazard withstand requirements of the system pertaining to those of the subsystem being modified are identified and documented
- The functionality provided by the COTS device is clearly identified and documented
- There is traceability between requirements of the SCS to those offered by the COTS device
- Failure modes of the COTS device are identified and documented
- Effects of COTS device failure on the overall system functionality have been identified and documented as far as is reasonably practicable in this stage of the lifecycle

The evidence below can support the argument:

- SRS for the SCS
- Requirement specification for the COTS device
- Model based requirements decomposition (integrated requirements, structure and behaviour models)
- Model based FMEA
- Hardware reliability analysis (e.g. parts count)
- Set of technical queries for the OEM developer
- Set of test cases generated from FMEA and reliability analysis

Argument A.2 – Device development process and principles are adequate for safety function

This argument concerns the development processes and principles used by the COTS device developer. As such, it is expected that the evidence is gathered through an audit of the OEM's device development processes and which seeks to justify the following:

- Development of the COTS device has been performed according to a recognised development lifecycle with activity outputs produced and verified by competent persons
- Configuration control has been implemented throughout the lifecycle of the COTS device
- The COTS device developer is of good pedigree
- Software design incorporates measures to ensure errors or failures of the software are detected early, do not propagate beyond specified limits and do not go undetected

The evidence below can support the argument:

- Quality Assurance audit of COTS device developer, noting this is less onerous than an audit against *IEC61508*.
- Supplier Pedigree (e.g. independent certifications, functional safety accredited development etc.)
- Discussions with COTS device developer informed by the models created to support A.1
- Evidence demonstrating test cases developed to support A.1 have been considered by the OEM (e.g. unit test reports, design analyses etc.)

Argument A.3 – Independent measures demonstrate COTS device correctly performs its intended functionality

It is intended to report on the confidence gained in the production process through independent measures. This is used as a complementary means to address the issue of limited access to detailed design of the COTS instrument, as discussed in section 2. The argument should seek to justify the following:

- Complementary testing demonstrates that COTS device can meet the functions specified in its requirement specification under all specified conditions and within the specified acceptance criteria
- Operational experience data provides confidence in the ability of the COTS device to perform its specified functions reliably

The evidence below can support the argument:

- Model Based Testing carried out using test cases generated from requirements, structural and behavioural modelling, failure mode analyses and ATCG
- Operational Profile based Statistical Testing (see C.5.1 of *IEC61508* Part 7 for information on statistical testing)
- Defect based and exploratory testing informed by the choice of programming language used
- COTS device pedigree such as field use data, fault reports, soak testing etc. which are specific to the COTS device

Argument A.4 – Functionality and interfaces of the COTS device have been limited to ensure its specificity to the required application

As the COTS device utilizes embedded software, it is reasonable to expect that its complex. Complexity limits the functional test coverage that can be achieved for the device and as such this argument is concerned with limiting the functionality of the device utilized by the SCS to only those functions that are associated with SFRs. The argument should seek to justify the following:

- Any unwanted functionality not essential to perform a SFR is restricted on the COTS device
- Any unwanted functionality of the COTS device with respect to the SCS SFRs does not undermine the SFRs
- The COTS device input and output interfaces are limited as far as possible to those essential for the SFRs
- Ability to configure the COTS device is limited to programming of process specific variables e.g. alarm thresholds, calibrations etc.

The evidence below can support the argument:

- Application specific conditions of use
- Functional testing to demonstrate no detriment to safety functions under the conditions of use
- Negative testing to demonstrate no detriment to safety functions from unwanted functionality that cannot be restricted

Argument A.5 – Validation activities adequately demonstrate system requirements have been met by the device
Objective of this argument is to demonstrate the COTS device satisfies SFRs of the SCS. The argument should seek to justify the following:

- Functional, interface, performance and hazard withstand requirements of the SCS are met by the COTS device
- Operability and maintenance aspects of the COTS device pertaining to its application are adequate
- Product lifetime is adequate for its operating environment

The evidence below can support the argument:

- Type testing of COTS device for application specific environmental conditions
- Acceptance testing ensuring functional and performance requirements are met
- Recommendation and implementation of in service operation and maintenance requirements e.g. proof testing, in service inspections etc.

5 Conclusion

In the preceding sections it is demonstrated how MBSA can provide an alternative method for safety assurance of a COTS device with embedded software. It

also shows how this can support the objectives of the nuclear industry's multi-legged approach when supplemented with other software safety assurance techniques. Furthermore, we implicitly show how the use of MBSE allows us to contextualise our questioning, documentation etc. through association with the model elements. This powerful aspect of MBSE provides a massive advantage in MBSA by removing ambiguities created through personal human models and providing a single platform for communication between the product developers and safety assurance engineers. Although we have overcome some of the aforementioned challenges posed by the current approach, MBSA also has its disadvantages:

- The COTS device modelling is based on publicly available information about the device. Although this has the advantage of independence from the COTS device developers, it can limit the achievable model fidelity. As mentioned before, there is trade-off between independence and input from the OEM or perhaps even no need to maintain this independence. However, this is something that can be addressed through further research.
- The model will always be an abstraction of the COTS device and there will always be differences between the modelled design and the real design of the COTS device (as different developers design them). This could result in large numbers of technical queries or false test cases that pose a high workload for the safety assurance team. However, it transfers the effort from the OEM to the safety assurance team thus addressing challenge 2 discussed in section 2.
- It is hard to judge the granularity of the model elements required and their representation of the real device. This can be adjusted according to the integrity requirement of the safety function but requires research and subsequently guidance to be put in place. Without these, it could remain a debatable aspect of this approach.
- It is expected ATCG could result in an unmanageable number of test cases (specifically for devices which provide multiple functions e.g. process controllers). Thus, it is recommended pre-developed sentencing strategies[6] and test Oracles[7] are used to decide which test cases are of most concern and add the most value to the safety assurance. Furthermore, the credibility of the test cases is highly dependent on the quality of the modelling against the actual COTS device. This is also subject of further research and study.
- The strategy proposed addresses the challenges discussed in section 2, with the exception of challenge 1. There is a change in the set of skills required by the assessors. Unlike *Emphasis* assessors who need a broad set of

[6] Pre-developed sentencing strategies are rules designed to refine and prioritise test cases based on their effectiveness of finding errors and practicality of running the tests.

[7] Test Oracles are mechanisms for determining if a test has passed or failed.

knowledge at audits, modellers (who are usually skilled at system/software engineering) can be supported by hardware and reliability specialists to inform their models. However, it does mean a diverse team of specialists is required to support the safety assurance task, as it is unlikely the modeller alone will be technically competent to reverse engineer a specialist I&C device in a modelling environment.

MBSA can provide an alternative way of assessing COTS devices with embedded software. It addresses many of the challenges faced by the current qualification strategy used in the nuclear industry, primarily in the area of OEM developer resource. It also generates much of the evidence required for putting a safety argument together and is strengthened when accompanied with traditional software assurance techniques. There however remains a large amount of work to be done to evaluate its effectiveness and address some of the challenges discussed. These are recommended as subjects of further research on the topic.

Acknowledgments I would like to thank various colleagues for supporting the production of this work. In particular Mr Sam Moody, Dr Silke Kuball, Mr Andrew Jennings, Mr Sam Robinson and Mr Martin Gale for reviewing the paper and/or providing guidance, encouragement and useful critiques of this research work. I would also like to express my deep gratitude to EDF Energy's C&I Systems branch for making available their support and resources to progress this work.

Disclaimers The views expressed within this paper are of the author and not those of the nuclear industry or EDF Energy. They are based on the author's experience gained from working on nuclear safety-critical system assessments. The example used is based on a real instrument but does not represent what is being or was done specifically for a particular application.

References

Conformiq (2011) A Conformiq Technology Brief - The Conformiq Approach to Algorithmic Test Generation

EPRI (1996) Guideline on Evaluation and Acceptance of Commercial Grade Digital Equipment for Nuclear Safety Applications

IAEA (2011) IAEA Nuclear Energy Series – Workforce Planning for New Nuclear Power Programmes NG-T-3.10

IEC61226 (2010) International Standards – Nuclear power plants – Instrumentation and control important to safety – Classification of instrumentation and control functions

IEC61508 (2010) International Standard – Functional safety of electrical/electronic/programmable electronic safety-related systems

IEC62138 (2009) International Standard – Nuclear power plants – Instrumentation and control important to safety – Software aspects for computer based systems performing category B or C functions

INCOSE (2007) International Council on Systems Engineering, Systems Engineering Vision 2020 INCOSE-TP-2004-004-02 Version 2.03

OMG (2015) OMG Systems Modeling Language (OMG SysML ™) formal/2015-06-03 https://www.omg.org/spec/SysML/1.4 (Last accessed on 05 Nov. 19)

ONR (2019) ONR GUIDE: Nuclear Safety Technical Assessment Guide - Computer Based Safety Systems NS-TAST-GD-046 Revision 5

Roques (2015) Requirements Engineering Magazine – Modelling Requirements with SysML https://re-magazine.ireb.org/articles/modeling-requirements-with-sysml (Last accessed on 05 Nov. 2019)

Independent Co-Assurance using the Safety-Security Assurance Framework (SSAF): A Bayesian Belief Network Implementation for IEC 61508 and Common Criteria

Nikita Johnson[1], Youcef Gheraibia and Tim Kelly

University of York, UK

Abstract *For modern safety-critical systems we aim to simultaneously maintain safety whilst taking advantage of the benefits of system interconnectedness and faster communications. Many standards have recognised and responded to the serious security implications of making these connections between systems that have traditionally been closed. In addition, there have been several advances in developing techniques to combine the two attributes, however, the problem of integrated assurance remains. What is missing is a systematic approach to reasoning about alignment. In this paper, the Safety-Security Assurance Framework (SSAF) is presented as a candidate solution. SSAF is a two-part framework based on the concept of independent co-assurance (i.e. allowing separate assurance processes, but enabling the timely exchange of correct and necessary information). To demonstrate SSAF's application, a case study is given using requirements from widely-adopted standards (IEC 61508 and Common Criteria) and a Bayesian Belief Network. With a clear understanding of the trade-offs and the interactions, it is possible to create better models for alignment and therefore improve safety-security co-assurance.*

1 Introduction

In systems engineering the tension between system safety (the desire to protect people from harm) and cyber security (the desire to protect assets of a system) has increased in recent years. This is primarily due to the increased size and interconnectivity of modern systems. In order to maximise productivity and capability we allow systems to communicate with each other and share their services.

[1] Corresponding author nlj500 <at> york.ac.uk

© Nikita Johnson 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

This interconnectedness presents greater security risk as there is a larger attack surface and many more ways for 'things to go wrong'. This threatens to undermine the goals of an entire system, including safety goals. From an assurance perspective, it is therefore no longer acceptable to assume complete separation of safety and security risk. Nor is it acceptable to treat that risk solely in the comfortable isolation of each domain's[1] practices, knowledge, and culture.

In an attempt to address the issue of isolated practice, many solutions have been created by extending existing safety techniques. These are partial solutions at best because their focus is predominantly on safety and much of the security information is discarded. Many of these techniques have previously been critically examined (Johnson & Kelly, 2019a) and found insufficient for through-life co-assurance.

There are *technical approaches* that aim to combine risk concepts across domains; *organisational structures* that allow for better communication between experts; and also legal and *regulatory initiatives* to align[2] safety and security at national and international levels. However, these changes do not address all the concerns that arise from bringing the two attributes together, therefore issues with misalignment remain.

This paper presents a candidate solution that enables and supports full alignment of safety and security – the Safety Security Assurance Framework (SSAF). At the core of the Framework is the concept of *independent co-assurance* and *synchronisation points*. This allows for some separation to be maintained in order to make the most of differing expertise, knowledge and practice, whilst ensuring that the right people get the right information at the right time.

This paper is laid out in three parts. Part 1: Section 2 attempts to characterise the differences between safety and security. Part 2: Section 3 and 4 present the Safety Security Assurance Framework (SSAF) and apply it to a case study using a Bayesian approach and requirements found in standards. Finally, Part 3: Sections 5, 6 and 7 examine the rationale of the decisions made during the case study, discuss related work, and provide a conclusion.

[1] Here *domain* refers to technical area (safety or security) rather than application domain.

[2] The term *alignment* will be used interchangeably with co-assurance because of its widespread use in industry. It is worth noting that co-assurance (argument and process) is achieved through aligning safety and security.

2 Characterising Challenges in Safety and Security Assurance

Due to the similarities between safety and security assurance, namely that both processes are concerned with reduction of risk and prevention of loss, it is tempting to equate the two notions of risk and have one representative (quantitative) value. Whilst this is certainly feasible, as demonstrated by many techniques that combine risk analysis into a single methodology, these methodologies often disregard the fact that there exist differences that make the attributes incommensurable[3]. This may be one of the reasons that none of these methodologies has been widely adopted for co-assuring safety-critical systems.

Two characterisations of safety and security assurance will be discussed in this section with the aim of making their differences clear. The emphasis is on what makes co-assurance difficult, and the requisite qualities of an effective alignment solution. The first characterisation will look at the contribution of security concerns to safety risk; the second will use widely-adopted standards to establish differences between the attributes.

2.1 Effects of Security on Safety Risk

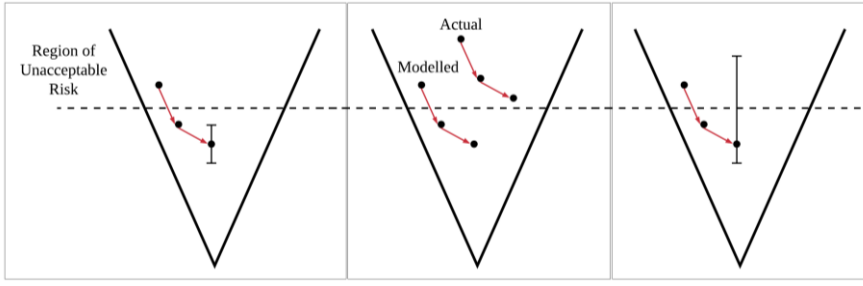


Fig. 1. ALARP Representation of the Problems of Uncertainty for Safety Risk

In the UK, the Health and Safety at Work Act 1974 (HSE, 1974) states it is the duty of employers to ensure the safety of its employees "so far as is reasonably practicable". This philosophy is often enacted as the ALARP principle: safety risk should be As Low As Reasonably Practicable. Depicted in Figure 1(a) is the ALARP "carrot diagram". The idea is to identify the level of a particular risk, then systematically reduce that risk until it is ALARP. It is possible to reduce risk

[3] Definition: not able to be judged by the same standards.

in one of three ways: *i.* Designing it out of a system, *ii.* Engineering in controls, or *iii.* Having procedural mitigations.

Alongside the risk value is a window of variation which is analogous to a statistical confidence interval; this represents the uncertainty in the estimation of risk. Several factors affect this interval such as the competence of the practitioners, the rigour of their processes, the limitations of the tools they use, *etc.* Safety is concerned with the higher portion of this interval, and the potential for variance into the unacceptable risk region. Thus, it is often a requirement by regulators for a confidence argument to be provided with the safety risk argument or safety case.

Figure 1(b) shows the first problem of safety-security alignment. Practitioners and engineers might follow an ALARP process and use their expert judgement to estimate the level of a particular risk; however due to the presence of an intelligent and motivated adversary the level of risk might be substantially higher in reality. Therefore, models and artefacts used to support a safety case are inaccurate and the safety argument is fundamentally undermined. There are ways that this can be minimised, for example verifying estimates made at design time against operational data, however this is not always feasible.

Figure 1(c) shows the second problem for the safety-security interaction: there may exist an estimation of risk, but the level of uncertainty may be high due to security concerns. This could be the result of socio-technical factors, such as inadequate processes, or the judgement of a practitioner with insufficient training.

Whilst the underlying reason for these two co-assurance problems is the uncertainty introduced by security concerns, there are different treatments of uncertainty. Most existing technical approaches focus solely on the uncertainty introduced in Problem 1 above, *i.e.* they attempt to improve the accuracy of risk level by considering security sources of risk, but do not consider the implications of other assurance factors. This leads to:

Solution Criterion 1 (SC1) A solution for co-assurance *must* facilitate reasoning about the diverse ways in which uncertainty is introduced into assurance, and how it is handled on multiple levels of abstraction.

2.2 Safety and Security Assurance Characteristics

In (Alexander et al. 2004), characteristics of Systems-of-Systems are identified with the aim of prompting analysts to proactively consider likely failure modes in relation to the characteristics; thus, another way to gain knowledge about risk, besides learning from accidents, is created.

In a similar fashion, in this section, the safety and security characteristics for which there is conflict are identified from three of the most commonly used standards across both domains. The aim of identifying the characteristics is to understand the ways in which safety and security can diverge, and so proactively consider ways to prevent these *co-assurance failures*.

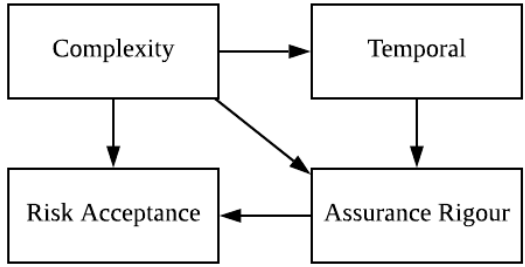


Fig. 2. Characteristics of Co-Assurance Factors

The source of safety assurance characteristics is IEC 61508 (International Electrotechnical Commission, 2010). For security, both ISO 27005 (International Organization for Standardization, 2011) and Common Criteria (International Organization for Standardization, 2017) were used. Figure 2 shows the shared characteristics of safety and security assurance.

Risk Acceptance – This describes the attributes' attitude to risk. First and foremost, safety is about preventing death and injury so naturally it is very conservative and risk averse; made clear from the prescriptive and conservative safety requirements. This is in stark contrast to security, where acceptable risk is a lot more strategic; in most systems (that are not safety-critical) there is an element of gameplay[4] in an attempt to balance security risk against potential benefits. This flexibility is reflected in the standards, where unlike safety standards, no universal notion of acceptable risk is given. It is worth noting that as security incidents begin to have a greater impact on society, regulatory authorities and lawmakers are attempting to define acceptable risk for security so that if it is not maintained, punitive action can be taken for those responsible.

Assurance Rigour – Both attributes' assurance processes aim for order, reasoned arguments supported by evidence, predictable behaviours, and repeatable outcomes, *etc.* Due to safety's risk aversion, the operational environment of safety-critical systems is often defined and strictly bounded, thereby making it easier to achieve higher levels of assurance rigour at system level. For security however, the intelligent adversary alone means that there exists higher levels of uncertainty, which makes it much more difficult to achieve predictable outcomes.

[4] Gameplay describes Game Mechanics, bounds of decision making, and rules of interaction permitted for agents. Ryutov et al. (2015) conceptualise cyber security as a three-way adversarial game between attackers, defenders and users.

The approach is therefore to achieve assurance rigour at lower levels of the system (*e.g.* component level) with clearly defined assumptions to test against.

Temporal – This is the time required to perform assurance activities and the expected rate of change. For safety, the conservative risk stance and the need to provide detailed argumentation about acceptable risk takes time. It is also expected that once a system has been certified[5], the safety case remains valid unless there is a major change to the system or its usage. In contrast, the security assurance environment is more agile and fast-paced. This is reflected in the "classes" approach to requirements in the standards that aims to protect against groups of vulnerabilities, so that change can be responded to more effectively – such as when a new vulnerability is discovered. Security arguments need to be robust to change at a faster rate than safety.

Complexity – Both safety and security are emergent, but they handle different types of complexity in different ways. For example, complexity due to number of components is not an issue in and of itself for safety[6]; what is important is that complexity does not interfere with the assurance factors for safety. In contrast, redundancy and diversity creates more work for security, as there are a greater number of potential attack options to consider. This has a significant impact on the time resource available and the level of assurance rigour that can be achieved.

These are the four characteristics where the most significant alignment failures are likely to occur. Having defined these, it is now possible to find mitigations to prevent the specific mismatches. Therefore:

Solution Criterion 2 (SC2) A solution for co-assurance *must* address the alignment trade-offs with respect to each of the assurance characteristics (risk acceptance, assurance rigour, temporal and complexity).

It is clear that the co-assurance solution criteria stated in this section are not inherently technical. Although there will most certainly be an element of technical analysis required for alignment, a solution that considers only this aspect and not the wider context would be a partial solution at best. What is needed is an approach that has the capability to address both the technical risk alignment and the socio-technical assurance factors.

[5] "certified" used in the broadest sense. Based on the assumption that *all* safety-related sectors have some form of formal safety acceptance before use that does not necessary involve a regulator.

[6] For example, if a system is built with an infinite number of components whose behaviour is formally understood, then we can create a mathematical model of this infinitely large system, and argue safety.

3 The Safety-Security Assurance Framework

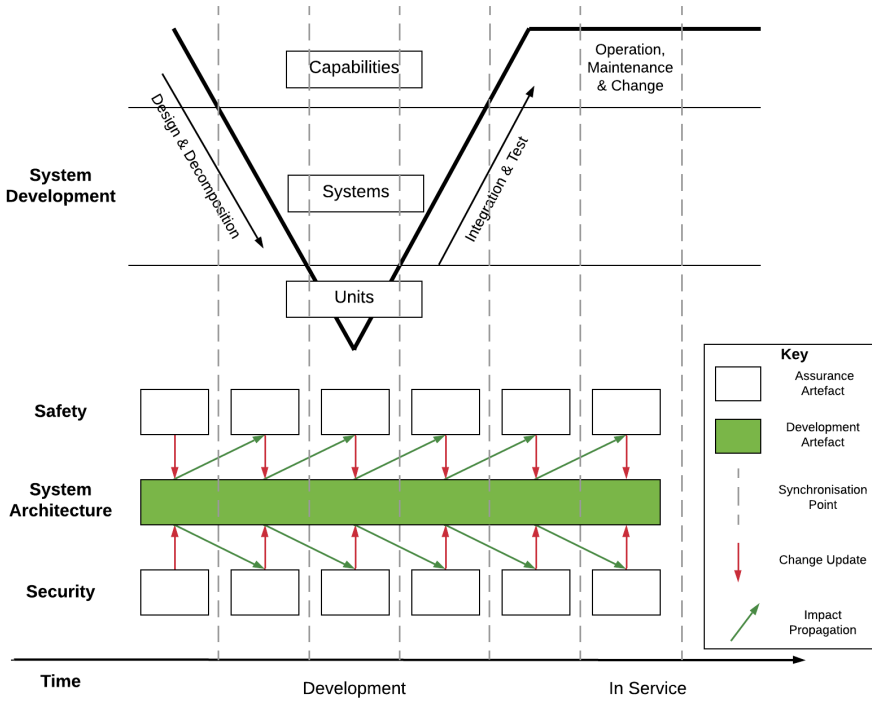


Fig. 3. SSAF: Independent Co-assurance and Synchronisation Points

The Safety-Security Assurance Framework (SSAF) is a solution for co-assurance that addresses the criteria (SC1 and SC2) stipulated in the challenges section. It consists of a process and model for systematically reasoning about the alignment of system safety and cyber security throughout the life of a system. SSAF is built on the new paradigm of *independent co-assurance* – that is, keeping the disciplines and expertise separate but having key synchronisation points where required information is exchanged across the discipline boundaries *i.e.* "the right information is given to the right people at the right time". Figure 3 illustrates this concept during system development and deployment. Note that SSAF was developed with the assumption of a model-based design, however that does not preclude its use on non-model-based systems. More work is required to establish the synchronisation (sync) points in this case.

SSAF is comprised of two models – the *Technical Risk Model* allows for the communication of risk and impact across disciplines; and the *Socio-Technical Model* which recognises that co-assurance is an inherently human activity that

involves many judgements at different levels that could constrain the technical solution to alignment.

3.1 SSAF Technical Risk Model (TRM)

The first SSAF model – the **Technical Risk Model** has three major parts:

1. An *ontology* for cross discipline communication
2. A *5-step process* for creating synchronisation points and links
3. A *causal model and patterns* for different conditions and their relationships

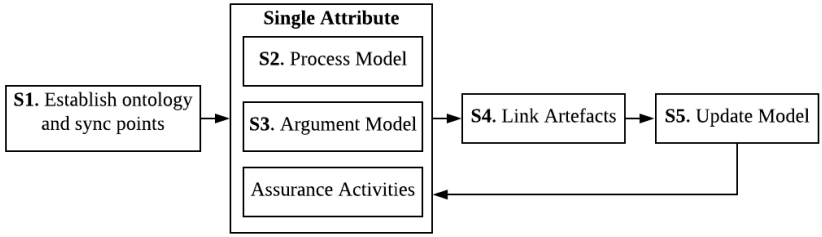


Fig. 4. SSAF TRM Process for Synchronisation

The SSAF TRM processes in Figure 4 consists of five steps. Steps 1, 4 and 5 are where the unique contribution of the TRM lies. Steps 2 and 3 are assumed to be the standard best practice in each of the domains[7].

At the core of making independent co-assurance, and therefore SSAF, work is establishing sync points and understanding the causal relationships between conditions in safety and security. Many standards have started to include synchronisation points where safety and security must communicate as an integral part of their processes *e.g.* ARP 4754A (SAE International, 2010) and DO-326A (RTCA, 2014) for aerospace, ISO 14971 (International Organization for Standardization, 2007, p. 200) and AAMI TIR 57 (Association for the Advancement of Medical Instrumentation, 2016) for medical devices, *etc.* However, it can be argued that these are the absolute minimum needed for alignment. In order to add more sync points and improve alignment, it is important to understand what information needs to be exchanged and when.

Figure 5 shows the (partial) TRM causal meta-model that enables relationships between conditions across domains to be modelled. By systematically understanding relationship between conditions and the synchronisation points

[7] In (Johnson & Kelly, 2019b) a worked example of the application of the SSAF TRM process to a wearable medical device is given.

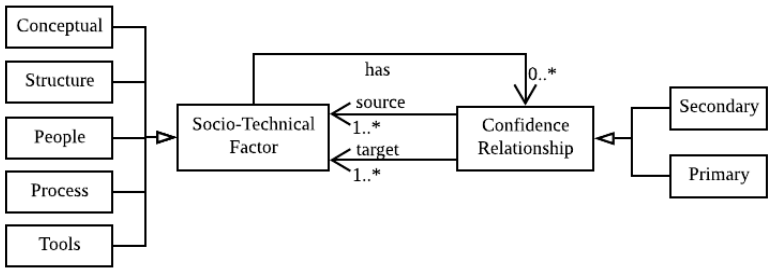


Fig. 6. SSAF STM Confidence Relationships

Primary confidence describes all the factors that could affect the technical risk argument directly, examples of this include the analysis approach being used, and the competence of practitioners. *Secondary confidence* describes all the factors that may influence primary confidence. This includes organisational structures, engineering processes, individual cognitive biases, *etc.*

The STM has roots in the socio-technical systems domain. The categories of assurance factors are based on Bostrom and Heinan's model of socio-technical systems design (Bostrom & Heinen, 1977). The categories are *Structure*, *Process People* and *Tools*. An additional category, *Conceptual*, was added to the model. It is orthogonal to the other types of assurance factor. For safety-security alignment communication of mental models is one of the biggest factors affecting assurance. The communication of concepts was not adequately represented in any of the other four categories. By explicitly representing points where uncertainty can be propagated across domain boundaries, we can start to reason about *why* it is not a problem, and still have confidence in the risk argument. Without articulating these relationships it is almost impossible to do this.

3.3 Assurance surface

Another useful concept introduced by the SSAF model is that of the assurance surface. The underlying concept is borrowed from the security domain - *attack surface i.e.* all the vectors that an attacker might exploit to launch an attack on a system (Howard, Pincus, & Wing, 2005). The *assurance surface* by analogy is all the ways in which uncertainty can be propagated across domains. The concept necessitates an important shift in thinking for co-assurance. It implies that there are multiple ways in which uncertainty can be propagated, and therefore it is highly unlikely that any one technical approach to integration will address all uncertainty propagation. Instead we must think in terms of *assurance coverage*,

and use the best possible combination of approaches and argumentation to minimise uncertainty propagation across domains. Figure 7 shows the relation between SSAF TRM and STM to existing safety-critical system development. More detail about the tiers is provided in (Johnson & Kelly, 2019c).

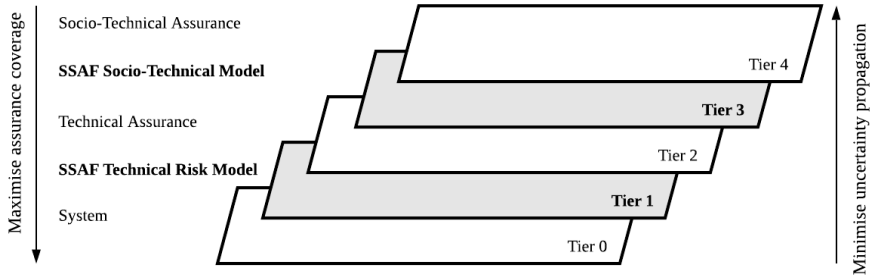


Fig. 7. SSAF Assurance Surface Concept

4 Case Study: SSAF TRM Bayesian Approach to Standards

Thus far, SSAF independent co-assurance has been presented as a potential solution to safety-security alignment problems. The reasoning behind why the approach of explicitly modelling relationships is the best way forward has been provided. However, still lacking is the means by which to achieve this.

That is the purpose of this case study. The motivation is twofold – first, to provide empirical evidence for the SSAF models and methodology. Secondarily, to show the creation of a re-useable alignment pattern. To maximise applicability of the case study, requirements from two of the most commonly used single-domain standards were selected: IEC 61508 and Common Criteria.

4.1 IEC 61508 and Common Criteria Brief

IEC 61508 is arguably the most widely adopted safety standard. It has been adapted to multiple domains including healthcare, rail, automotive, aerospace, and nuclear. It consists of seven parts that define the safety process for a system. It is the software design and development (software architecture design) requirements found in Table A.2, p48 IEC 61508-3:2010 that were selected for this case study.

Common Criteria is a widely adopted security standard that has been adopted across many types of systems in many domains, including some that are safety-

critical. It consists of three parts. For this case study, functional requirements from Common Criteria Part 2 were selected.

4.2 'Link and Sync' Methodology

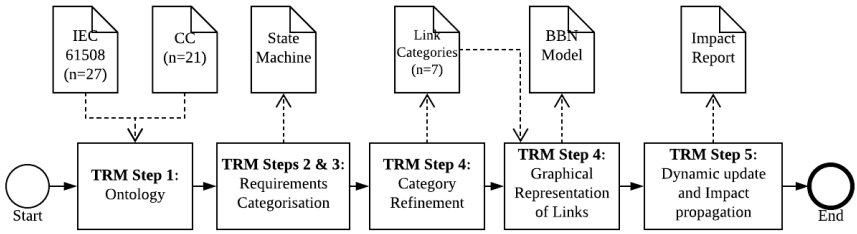


Fig. 8. BPMN[11] Model of Methodology for Linking Safety and Security Requirements

The main contribution of SSAF TRM is the explicit modelling of and reasoning about the causal relationships that exist at different synchronisation points. This is encapsulated in Steps 1, 4 and 5 of the SSAF TRM Process. The objective of this case study is to demonstrate *how* this process functions, emulate how industrial system requirements could be linked, and show how the links could be implemented on a project. Figure 8 shows the process steps followed for the case study.

Step 1 – Ontology. Using 27 functional design requirements from IEC 61508 (found in Part 3 Annex A Table A.2) and 21 functional requirements from Common Criteria Part 2 – commonalities and general categories were identified.

Steps 2 & 3 – Requirements Categorisation. These steps were performed independently within each domain, with respect to either safety or security. The ontology and categories established in Step 1 were used to categorise the requirements according to type. In addition, a state machine was created to explain the impact on safety in the absence of a safety argument (further detail in Section 5.2).

Step 4a – Category Refinement. Once the requirements had been through initial categorisation, the categories were jointly refined further which resulted in 7 types of requirements. These were mapped to four states in a state machine that showed which requirements were violated. The four states were St0 None, St1 Resource & Timing Violated, St2 Failure Behaviour Violated, St3 Communication Violated.

[11] Business Process Model and Notation – process modelling notation

Step 4b – Graphical Representation. Using the refined categories, requirements from safety and security which were in the same categories were linked to each other. These links were then modelled as a Bayesian Belief Network (BBN).

Step 5 – Dynamic Update and Impact Propagation. The leaf nodes of the BBN are the security classes of requirements from Common Criteria. A practitioner provides details if a security requirement class has been "violated" or not. The BBN then outputs the probabilities of being in state St1, St2 or St3.

4.3 Results

Figure 9 shows the state machine that was output from TRM Steps 2 and 3. It consists of four states. S0 where no safety requirements have been violated, and three other states where at least one safety requirement from the IEC 61508 set was violated. Transitions occur according to the type of safety requirement that has not been satisfied[12], for example not satisfying requirement "13a Guaranteed maximum time" would transition to state S1. To return to S0 that violation would need to be resolved. The states were formed by grouping the seven requirements types in groups which were highly cohesive, *i.e.* {Resource Use and Timing}, {Failure Behaviour, Failure Detection, Recovery}, and {Communication and Trust}.

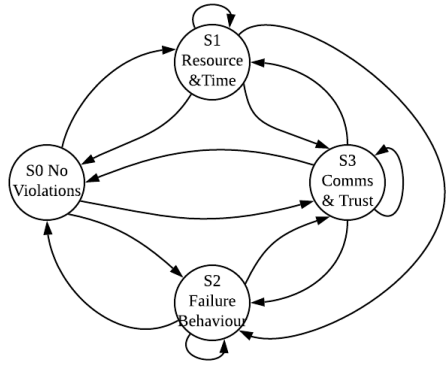


Fig. 9. State Machine for Safety Requirements Violation

[12] It does not add to the analysis in this case to distinguish between "requirements violation" and "not satisfying a requirement". They are viewed as equivalent, however it might be necessary to make the distinction in operational systems where violation during operation may carry more severe consequences than requirements not being met pre-deployment at design.

Figures 10(a) and 10(b) show the model of the causal links that were established during the linking process in TRM Step 4. Figure 10(a) provides a summary conceptual model to communicate the content and structure of the BBN. Figure 10(b) shows the real-world implementation of the BBN in the GeNIe modelling tool[13].

The leaf nodes of the BBN are the requirements classes taken from Common Criteria. The driving concept that makes this model successful, is the idea that if any of the security requirements in that class are violated, it can be input into the BBN leaf nodes. The impact then propagates through the classes and related safety requirements to the safety output *i.e.* the impact report which is the probability of being in a particular safety state.

As knowledge is contained in the state machine about how to transition back to a state where no safety requirements have been violated, it is now possible for a safety practitioner to take the output impact report from the BBN, and use that to determine the state, then resolve the issue more efficiently without needing to know specific information about the security requirements.

This model would be most useful during operation where security violations can occur at a fast rate. However, the model has some utility during the requirements phase to reason about impact – similar to sensitivity analysis.

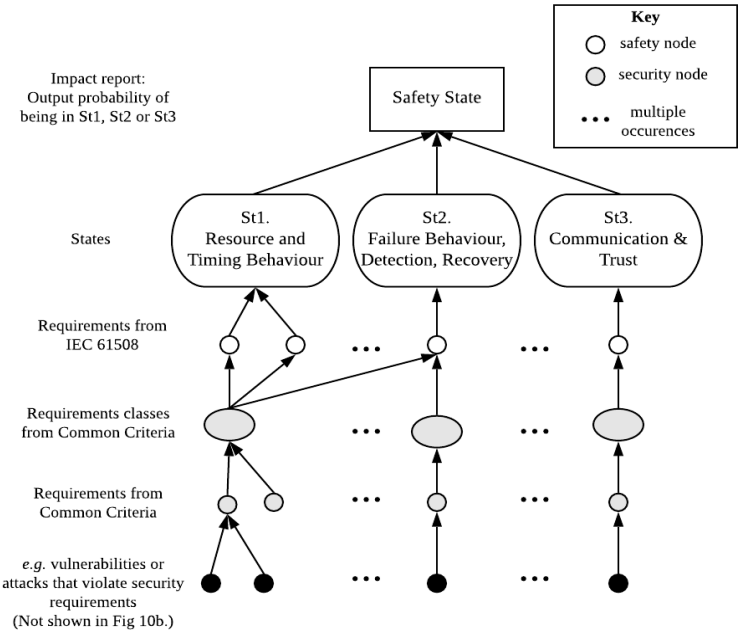


Fig. 10(a). Conceptual Model of BBN Links between Safety and Security

[13] Tool description can be found at: https://www.bayesfusion.com/genie/

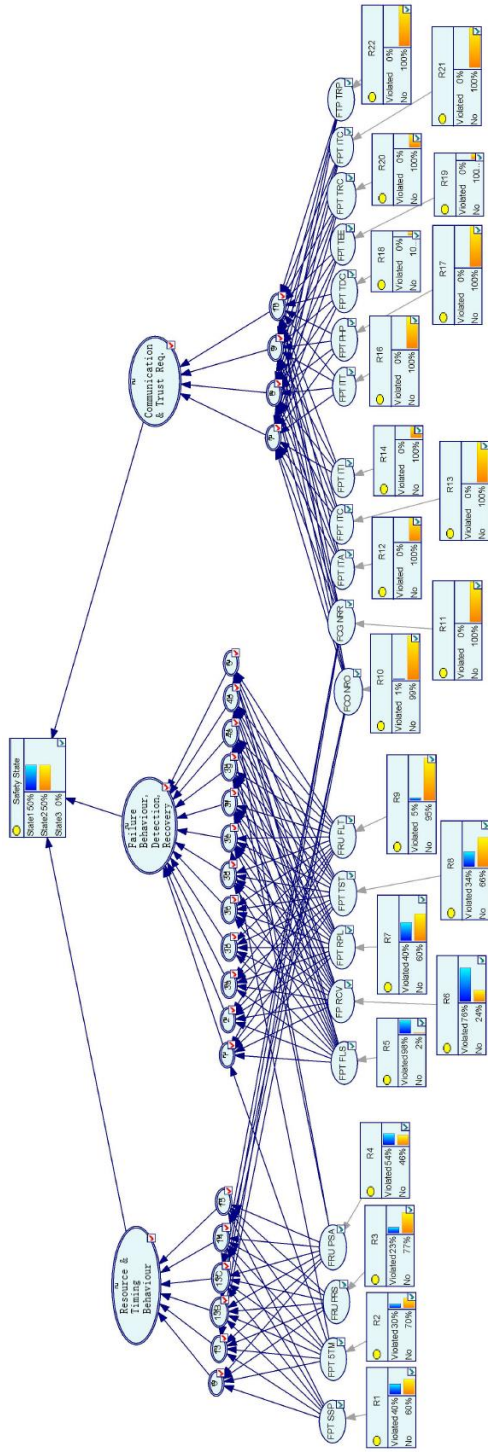


Fig. 10(b). GeNIe Implementation of the BBN from Fig. 10(a)

5 Discussion

The quality of this implementation of SSAF is dependent on the quality of the links *i.e.* between the safety requirements from IEC 61508 and security requirements from Common Criteria. The links were determined by sorting them in to cohesive groups. Table 1 shows an example of the requirements from both safety and security for Resource Use and Timing. If performed on an industrial project, the group categories could be decided beforehand, practitioners could classify the artefacts in each domain separately, subsequently link tables can be created.

Table 1. Example Grouping for Resource Use and Timing Requirements

Domain	ID	Requirement
Safety	A2.6	Dynamic reconfiguration
	A2.13a	Guaranteed maximum time
	A2.13b	Time-triggered architecture
	A2.13c	Maximum response to events
	A2.14	Static resource allocation
	A2.15	Static synchronisation of access
Security	FPT_SSP	State synchrony protocol
	FPT_STM	Time stamps
	FRU_PRS	Priority of service
	FRU_RSA	Resource allocation

5.1 Deciding the Causal Relationships

Deciding on group categories is a non-trivial task. Most unified methodologies such as security-informed fault trees usually specify the syntax of how artefacts should be linked, but not the semantics, *e.g.* linking the top event of an attack tree to the base event of a fault tree. This SSAF Case Study implementation goes some way to answering the question about semantics of causal links. In this case, expert judgement, experience and concept cohesion were used to make the groupings.

It would have been possible to create links with less complex reasoning behind them, such as linking all safety and security requirements per component; however, the aim of co-assurance is to argue alignment using these links, therefore a more structured and strategic approach is needed for link creation. Preferably, an approach that can be examined, contested and repeated if necessary.

Grounded Theory (GT) research methodology (Glaser & Strauss, 2017) was used to address this problem because its function is to build connections that are

grounded in data. Ordinarily, GT is used in social science and humanities research. It was selected for application in this case study because it structures the thinking of the person performing it. It also has several distinct phases such as initial sampling, populating, memoing, constant comparison, *etc.* that can be documented and reviewed. This explicit documentation of reasoning could be used as evidence to support an alignment argument. Another advantage of using a GT approach is that existing connections can be extended or new ones created. This is especially important when considering multiple, complicated notions of causality that are present for safety and security. Having a causal model that theory of interaction that can be expanded is essential.

5.2 Action After Determining the Impact

In the TRM process described in previous work (Johnson & Kelly, 2019b) there exists the assumption that the argument structures for each attribute are known. In addition, there is the assumption that the artefacts (*e.g.* analysis models used for evidence) are linked to the argument (*e.g.* safety case) and the TRM model. So when a change occurs, impact can be traced from the TRM model to the claims in the argument. However, modelling the argument structures for Common Criteria and IEC 61508 was beyond the scope of this case study which is concerned mainly with the creation of causal links.

Instead, a state machine was presented as a way to understand the security impact on safety; *i.e.* by construction, the states communicate to the safety practitioner which types of safety requirement have been violated. This allows safety practitioners and decision makers to respond to change more effectively because they are not required to reason about security requirements to understand impact. Knowledge about particular states and how to transition is encapsulated in the model. This approach enables resources to be applied proportionally to the impact. For example, from a safety perspective, moving to a state where an availability-related requirement has been violated (S2) is probably of more concern than if a confidentiality-related requirement has been violated (S3).

There are, of course, a few limitations to using the state machine for the purposes of determining impact. The first is the assumption that the transitions modelled are possible and accurate, *i.e.* once a safety requirement has been violated then a suitable and timely resolution can be found to transition back to state S0 where there are no violations; this is unlikely to be true in all cases. However, even if transitions are not possible it is still important to capture the reasoning and assumptions in a systematic way.

Another limitation is the simplicity of the model. Only four states were modelled for comprehensibility, but many more states could be captured with many

more complex transitions. States could be included to represent multiple violations, partial violations, *etc.* - this would risk a possible state explosion that would be counterproductive to the aim of using the model, *i.e.* for practitioners to understand impact and what to do next.
Although there are limitations with this approach to handling impact, the state machine is understandable and helps practitioners to respond proportionally.

5.3 Socio-Technical Considerations

The Case Study was a controlled application of SSAF TRM whose focus was on establishing causal links and propagating technical risk. Thus, it is quite difficult to evaluate the socio-technical factors as they would have occurred in an industrial project such as evaluating preparatory tasks for the alignment meetings, *etc.*

SSAF STM provides a structured model to help practitioners reason about the socio-technical factors that would affect technical risk. Table 2 shows the kinds of claims that might be examined. Some of these, labelled primary, are claims that would affect confidence of the technical risk argument. All other factors affect the socio-technical factors that impact risk, those are labelled secondary.

Table 2. Example SSAF Socio-Technical Claims

STM Factor	Confidence	Claim
People	Primary	Practitioners are sufficiently competent to perform the methodologies.
Process	Secondary	Timing is bounded for information exchange and issue resolution
Structure	Secondary	The responsibility and authority to manage safety-security interactions has been designated
Tools	Primary	BBN is sufficient for the purpose of modelling the interactions between requirements

In the SSAF Case Study example, the alignment argument with primary and secondary confidence claims would need to be provided to show that the BBN was fit-for-purpose. A possible rebuttal for this particular instance, is that BBNs do not include a time dimension, therefore another complementary methodology might be used to align temporal factors.

6 Evaluation and Related Work

In Section 2, two criteria were identified for an alignment solution:

Solution Criterion 1 (SC1) A solution for co-assurance *must* facilitate reasoning about the diverse ways in which uncertainty is introduced into assurance, and how it is handled on multiple levels of abstraction.

Solution Criterion 2 (SC2) A solution for co-assurance *must* address the alignment trade-offs with respect to each of the assurance characteristics (risk acceptance, assurance rigour, temporal and complexity).

The SSAF Case Study presented here was limited to finding causal links between requirements for risk impact propagation. It is important to note that this is just one type of information exchange between safety and security, at one very particular point in the development lifecycle. The advantage of using the SSAF approach is that it does not limit practitioners to using just this one synchronisation point and methodology for alignment. Instead, it allows for multiple complementary approaches to be used. SC1 is therefore satisfied because SSAF provides a structure for reasoning at multiple synchronisation points throughout the life of a system. SC2 is much more difficult to demonstrate without a full industrial application and evaluation because the assurance factors are context dependent. However, even without this context, SSAF provides the mechanisms needed to address each of the factors.

Other than some emerging standards that include cross-domain considerations, SSAF is the only framework that allows for bi-directional impact propagation. Even then, the standards have a limited number of synchronisation points which would reflect minimum best practice at the time that the standard was written. SSAF is flexible enough to support practitioners if alignment approach requires many synchronisation points. SSAF also provides the structure to argue about the alignment of both safety and security from both a technical and socio-technical perspective. Another advantage, is that its models and output can provide evidence for an explicit co-assurance argument which could support other certification and accreditation activities.

Making the causal links and reasoning behind the co-assurance argument explicit has many advantages as discussed earlier in this section, but it also presents new questions such as whether or not those links or reasoning are correct, what the relationship is between links on different levels of abstraction and how to review the quality of the links with resources that could be spent engineering the system rather than on assurance.

6.1 Related Work

The case study showed one implementation of one causal model at one particular synchronisation point. This model will clearly not be universally applicable even though it is useful. It is possible that models such as this one could be used as the

basis for alignment patterns which would allow practitioners and engineers to reuse the knowledge.

There are many existing techniques for safety-security co-assurance and co-engineering. If the underlying causal model was made explicit, then more patterns could be created, thus creating a catalogue of possible alignment strategies. Some patterns could also be used as a requirement of standards. This is likely to be a much more practical approach to alignment than stating a synchronisation point but not saying how information should be exchanged, or providing one unified technique that is limited to only partial analysis of co-assurance factors.

Table 3 (discussed in Johnson & Kelly (2019b)) shows methodologies for alignment, and the possible causal links they represent:

Table 3. Causal Relationship Examples

Condition		**Causal Relationship**	
Source	**Target**	**Label**	**Methodology**[14]
Vulnerabilities	Failure	causes	FFA
Vulnerabilities	Hazards	contribute to	SAHARA, DDA, UML, FTA
Safety Effect	Attack	motivates	ADT
Threat	Hazard	safety impact	Standard
Threat	Safety Requirements	Influence	STPA-Sec, STPA-SafeSec
Safety Requirements	Security Requirements	trade-off	ATAM
Security Requirements	Safety Requirements	trade-off	ATAM
Security Controls	Safety Requirements	conflict with	ad-hoc

7 Conclusion

The Safety-Security Assurance Framework was presented as a candidate solution to the existing alignment problem between system safety and cyber security. SSAF is based on a new paradigm of independent co-assurance which allows for

[14] Full references and some discussion about these methodologies is given in Johnson & Kelly 2019b. The methodologies refer to Functional Failure Analysis (FFA), Security-Aware Hazard Analysis and Risk Assessment (SAHARA; HARA from ISO 26262 Part 3), Dependability Deviation Analysis (DDA), Unified Modelling Language (UML), Fault Tree Analysis (FTA), Attack Defense Tree (ADT), Systems Theorectic Process Analysis – Sec/SafeSec (STPA-Sec and STPA-SafeSec), and Architecture Tradeoff Analysis Method (ATAM).

separate domains, expertise, ways of working, *etc*. but requires that predetermined synchronisation points are established where information is exchanged. Multiple methodologies can be used at these synchronisation points, commensurate with the needs for alignment.

Much like the role of a systems integrator, it is likely that a new role will be created to manage the co-assurance argument and artefacts, and to ensure that the necessary activities occur; otherwise the co-assurance goals that are not covered by either safety or security goals will be overlooked.

The vision for the future of safety and security co-assurance is that the knowledge and practice for alignment is captured and modelled explicitly so that it can be examined, reasoned about, contested and reused. SSAF provides the structure to make that possible.

Acknowledgments Research and development of SSAF supported by the University of York, the Assuring Autonomy International Programme (AAIP), and BAE Systems. Research Award Ref: EPSRC iCASE 1515047.

References

Alexander, R., Hall-May, M., & Kelly, T. (2004). Characterisation of systems of systems failures. *Proceedings of the 22nd International System Safety Conference*. Citeseer.

Association for the Advancement of Medical Instrumentation. (2016). *AAMI TIR57:2016 Principles for medical device security—Risk management*.

Bostrom, R. P., & Heinen, J. S. (1977). MIS problems and failures: A socio-technical perspective. Part I: The causes. MIS quarterly, 17-32.

Glaser, B. G., & Strauss, A. L. (2017). *Discovery of grounded theory: Strategies for qualitative research*. Routledge.

Howard, M., Pincus, J., & Wing, J. M. (2005). Measuring Relative Attack Surfaces. In D. T. Lee, S. P. Shieh, & J. D. Tygar (Eds.), *Computer Security in the 21st Century* (pp. 109–137). https://doi.org/10.1007/0-387-24006-3_8

HSE. (1974). *Health and Safty at Work etc. Act 1974*. Retrieved from UK Health and Safety Executive website: http://www.legislation.gov.uk/ukpga/1974/37/contents

International Electrotechnical Commission. (2010). *IEC 61508-1:2010 Functional safety of electrical/electronic/programmable electronic safety-related systems. Part 1: General requirements* [Standard]. Geneva, CH.

International Organization for Standardization. (2007). *ISO 14971:2007 Medical devices – Application of risk management to medical devices* [Standard]. Geneva, CH.

International Organization for Standardization. (2011). *ISO/IEC 27005:2011 Information technology – Security techniques – Information security risk management* [Standard]. Geneva, CH.

International Organization for Standardization. (2017). *ISO/IEC 15408:2017 Common Criteria for Information Technology Security Evaluation. Part 1: Introduction and general model* [Standard]. Geneva, CH.

Johnson, N., & Kelly, T. (2019a). An Assurance Framework for Independent Co-assurance of Safety and Security. In C. Muniak (Ed.), *Journal of System Safety*. International System Safety Society.

Johnson, N., & Kelly, T. (2019b). Devil's in the detail: Through-life safety and security co-assurance using SSAF. *International Conference on Computer Safety, Reliability, and Security*, 299–314. Springer.

Johnson, N., & Kelly, T. (2019c). Structured Reasoning for Socio-Technical Factors of Safety-Security Assurance. *International Conference on Computer Safety, Reliability, and Security*, 178–184. Springer.

MISRA. (2019). *Guidelines for Automotive Safety Arguments*. ISBN 978-1-906400-24-8 (PDF), September 2019.

RTCA. (2014). *RTCA DO-326:Revision A Airworthiness Security Process Specification*. Washington, DC, USA.

Ryutov, T., Orosz, M., Blythe, J., & von Winterfeldt, D. (2015, September). *A game theoretic framework for modeling adversarial cyber security game among attackers, defenders, and users.* In International workshop on security and trust management (pp. 274-282). Springer, Cham.

SAE International. (2010). SAE ARP4754:Rev A Guidelines for Development of Civil Aircraft and Systems.

Developments in Safety & Security Integration: Remotely Piloted Unmanned Aircraft Systems Command and Control

Paul Hampton, Jonathan Pugh, Richard Ball

Independent Consultants

London, UK

Abstract *Unmanned Aircraft Systems (UAS) are forcing a rethink on how traditional safety and security assessment processes are conducted. Traditional concerns have been with the safety and security of the crew and passengers on the aircraft, but with the advent of UASs, these shift to the risks that the system presents to people and infrastructure on the ground, and other air users. This shift is presenting challenges to a large body of stakeholders, including: the rule makers, the UAS designers, the operators, safety and security assessors and the regulators. This paper provides a case study focussing on the command and control link to the aircraft, and describes the challenges experienced and developments made. Also, as a test case the paper aims to lay down a framework for a generalised approach for the harmonious integration of safety and security disciplines.*

1 Introduction

1.1 Remotely Piloted Aircraft Systems

Remotely Piloted Aircraft Systems (RPAS) is a technology that is anticipated to experience significant growth in the coming years. An RPAS is defined as *"A remotely piloted aircraft, its associated remote pilot station(s), the required command and control links and any other components as specified in the type design"* JARUS (2015). An RPAS is an aircraft operated without the possibility of direct

© Paul Hampton, Jonathan Pugh, Richard Ball 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

human intervention from within or on the aircraft. It includes the airframe, propulsion unit, flight controls, health monitoring systems, data communications, electrical system, navigation system, sensors and any other component on-board or attached to the aircraft. This unmanned aircraft is also known as a "drone", but more generally, it is referred to as Unmanned Aerial Vehicle (UAV[1]) or sometimes just an Unmanned Vehicle (UV).

Fig. 1. Example Unmanned Aerial Vehicle (UAV)[2]

The RPAS has a pilot located remotely to the aircraft who controls and manages the aircraft during the total duration of the flight (take off, air manoeuvres and landing). The entire system comprising the pilot control station, the data link and airborne vehicle is termed the Unmanned Aerial System (UAS).

Remote communication with the RPAS systems has traditionally been conducted through line-of-sight radio control, but communication like SATCOM are now being considered to support operations beyond line of sight (BLOS). Use of BLOS communications allows much wider ranges of operation but presents additional challenges in that the operator cannot see the aircraft, and remote communications can introduce latency in the control of the aircraft.

[1] UAV is a broader term that encompasses an autonomous vehicle as well as those used to support RPAS. An 'autonomous aircraft' is defined JARUS (2018) as "An unmanned aircraft that does not allow pilot intervention in the management of the flight". For the purposes of this work, the expression UAV is being used only in the context of remotely piloted vehicles.

[2] Image reproduced with permission from Alpha Unmanned Systems www.AlphaUnmannedSystems.com

The remote pilot does not have direct control of the aerodynamic surfaces but rather operates the vehicle in a 'fly by mouse' or 'pilot on the loop[3]' mode. In this mode, the pilot instructs the vehicle to manoeuvre to a particular point or along a predetermined route, but the vehicle's own flight management and navigation systems will be responsible for controlling the aerodynamic surfaces and propulsion system to execute the instruction. This also applies to take-off and landing phases where the pilot will initiate the instruction but the vehicle will automatically execute the take-off/landing procedure.

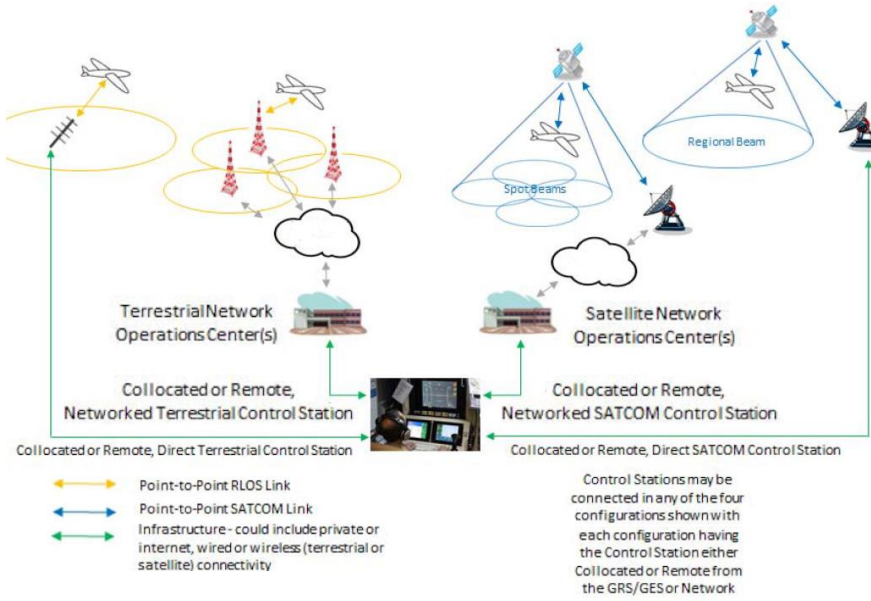


Fig. 2. RPAS BLOS Possible Connectivity[4]

Figure 2 shows some possible BLOS connectivity options. These can, for example, be through ground-based terrestrial links (top left of figure) or satellite-based links (top right of figure).

UAVs can be fitted with Detect And Avoid (DAA) subsystems to mitigate risks associated with beyond line of sight usage. The DAA subsystems can, for example, detect the presence of other aerial vehicles or terrain that present a danger of collision and inform the pilot so that avoidance action can be taken.

The link to the UAV involved with the command and control is called C2 and also sometimes the Non Payload Communication Link (NPCL). This link is solely related to the control of the vehicle, voice communications to the UAV's

[3] As distinct from 'Pilot *in* the loop' where the pilot has direct control.

[4] Image taken from DO-377 RTCA (2019) and used with permission and copyright of RTCA, Inc.

radio[5] (if present) and status reporting of its subsystems and does not include a video feed to the pilot from the UV's point of view. C2 is separate from "Payload" links used to convey data related to the mission function such as video images (e.g. hotspots on powerlines) and environmental monitors (eg. images of the spread of forest fires).

A typical use case for an RPAS Operation (based in US airspace) is shown in figure 3:

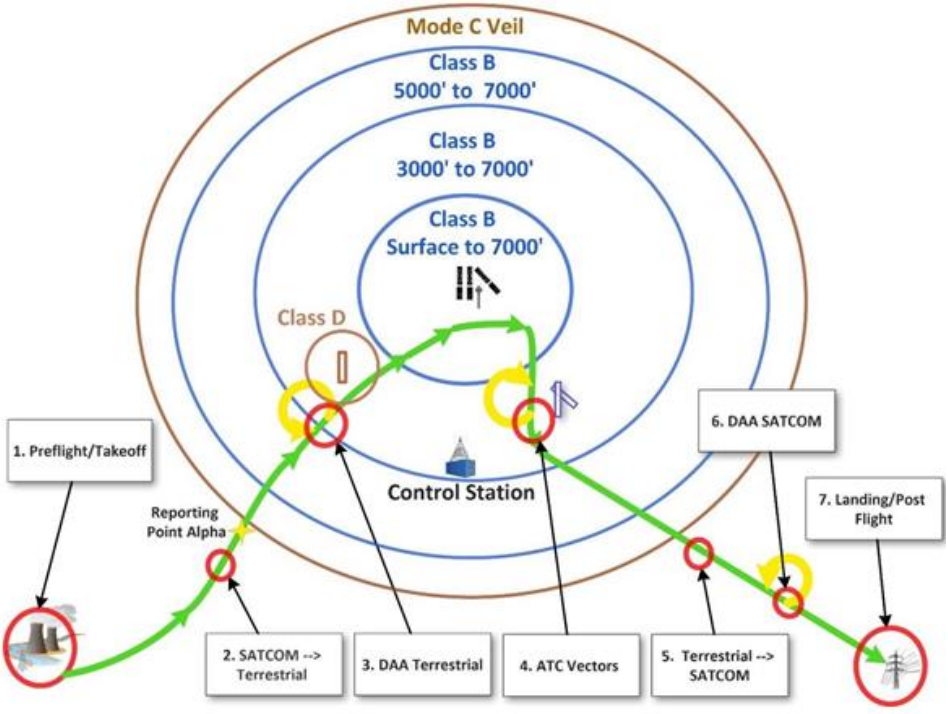


Fig. 3. RPAS Powerline Inspection[6]

In the diagram, the main route is shown in green with a yellow route indicating a temporary divergence to avoid conflicts with other air-users. The Classes refer to US airspace designations with Class B surrounding the busiest airports and Class D surrounding smaller aerodromes. The Mode C Veil surrounds Class B

[5] Although there is no pilot on the UAV, voice communications can be conveyed to the aircraft for onward broadcast via VHF so that other pilots in the vicinity can hear ATC voice exchanges with the UAV pilot.

[6] Image taken from DO-377 RTCA (2019) and used with permission and copyright of RTCA, Inc.

and requires aircraft to be fitted with transponders. In this scenario, the UAV inspects a powerline running from a power station (far left – point 1) through to an urban substation (far right – point 7). During the mission there is a switch between SATCOM and terrestrial links (points 2 and 5) and DAA activity (points 3 and 6) including responding to Air Traffic Control vectoring instructions (point 4) when in controlled airspace.

An operation such as this has a number of safety and security challenges:

- The remote pilot has no visual sight of the vehicle and is solely reliant on the accuracy and availability of status reporting to determine position and manoeuvring;
- Although the vehicle is unmanned, it still presents safety risks to other air users. For example, a mid-air collision with another manned aircraft would have catastrophic outcomes even for the smallest of UAVs;
- The vehicle presents safety risks to people and infrastructure on the ground, including ground crews involved in the take-off/landing procedures and the general public;
- As well as inadvertent errors in the UAS that might lead to loss of control of the UAV (e.g. link loss or uncommanded manoeuvres), C2 links could be subjected to security attacks that could lead to hazardous behaviour of the vehicle.

This paper discusses safety and security challenges in the context of the C2 link from the perspective of an organisation that wants to provide a C2 link service to RPAS operators. This paper discusses the approach of conducting both safety and security assessments and the framework that has been developed to integrate the two activities.

1.2 Standards Development

1.2.1 Safety Standards

RPAS standards have been slow to evolve as RPAS does not fit neatly into traditional aviation regulations that govern, say, airworthiness of aircraft (cf. CS.23 1309 EASA (2012) et al). This is because UAVs cannot be considered in isolation from the ground-based systems that would traditionally be covered by a separate set of regulations cf. EASA Commission Implementing Regulation 2017/373, EASA (2017) in Europe.

In Europe, standards for certified UASs are still some way off and the focus is on an intermediate category of use called “Specific” where approvals will only be granted for specific well-defined operational use cases. To this end, EUROCAE SC-105 and JARUS have been supporting EASA in developing a safety

assessment process: The Specific Operations Risk Assessment (SORA) process JARUS (2019).

In the US, RTCA SC-228 working group are developing minimum safety and performance requirements (MASPS) for C2 Link Systems in support of RPAS and current work is embodied in DO-377 RTCA (2019)[7]. Other standards development activities are being undertaken for DAA systems and type certification for the vehicle itself.

The certification process for the entire UAS is still unclear. In Europe, the European Aviation Safety Agency's (EASA) vision is to have an additional UAS Certification Specification (CS)[8] that spans most of the existing CSs but the work is still in early development.

As DO-377 is the most advanced in terms of providing quantitative performance requirements for the C2 Link System, this has been used as the basis for the Functional Hazard Assessment for the case study.

1.2.2 Security Standards

As with the safety domain, there is no single standard that governs the management of security risks for a UAS as, again, it spans both ground and airborne systems. For example, ED-203A EUROCAE (2018) is a relevant aviation security standard but is only intended to support aircraft airworthiness.

A number of techniques were therefore used in the case study as follows:

- Domain security risk analysis, based upon UK Government's National Cyber Security Centre (NCSC) Information Assurance Standards (IAS) 1 and 2 HMG (2012);
- STRIDE Threat Modelling Microsoft (2009);
- Security Control Objectives from Common Criteria ISO (2009);
- ED-203A Airworthiness Security Methods and Considerations EUROCAE (2018);
- ISO 27005 Risk Management ISO (2018).

Information Assurance Standard 1 & 2 (IAS 1&2) and its supporting documents is a legacy suite of information risk management guidance, produced by the predecessor of the NCSC.

The security risk analysis technique used was based on IAS 1 covering information security risk management. This standard used to be mandatory for UK

[7] The complete document may be purchased from RTCA, Inc. 1150 18th Street NW, Suite 910, Washington DC 20036, (202) 833-9339 www.rtca.org

[8] There are several CS's that are aligned with particular aircraft types such a large body aircraft, utility aircraft, helicopters etc.

public sector organisations. Whilst this ceased to be the case in 2015, the risk assessment method within IAS 1 and the accompanying risk management standard IAS 2 are still available for use. The guidance they contain is still valuable and suitable for broader application.

For the aviation case study, this methodology was adopted and simplified. It was used to assess risks via a combination of impact levels and threat severities applied to assets, also known as Business Domains, in order to arrive at a list of risks.

- The Microsoft STRIDE Threat Model allows threats to be modelled from an adversary's perspective. This was used to provide a systematic method of focusing security review effort and to categorise applicable system functions as security measures.
- ED-203A Airworthiness Security Methods and Considerations was used to provide an aerospace specific quantitative Risk Assessment. This was used to articulate risk as the function of impact and likelihood of threats.
- ISO 27005 guideline for information security risk management was used to define how risks are managed through design and operations. Where risk was identified as not acceptable, Security Control Objectives are derived to reduce risk.
- Security Control Objectives allowed for the inclusion of other standards and frameworks such as Common Criteria ISO (2009) and ISO/IEC 27001 ISO (2013).

2 Case Study

2.1 Context

A number of operational requirements set the context for the Case Study under consideration. The UV is:

- fitted with a DAA subsystem, but this is advisory only. It will detect and advise the pilot of a route to resolve the conflict but will not take any automatic control of the vehicle;
- expected to fly within controlled airspace and so will be fitted with a transponder and the pilot will need to respond to Air Traffic Control (ATC) instructions;
- fitted with a VHF radio for communication with ATC and other air users;
- a rotorcraft or fixed, wing typically of a size in the range of 3m-8m. This can include purpose-built UVs as well as retrofitted mainstream aircraft. Examples of this class of vehicle are as follows.

Fig. 4. Fixed Wing UAV[9]

Fig. 5. Rotorcraft UAV G-Air Schiebel S-100[10]

2.2 Case Study CONOPS

The case study is based on certified use by an operator allowing a variety of uses for this class of vehicle. Figure 6 provides examples of scenarios that may be typically supported.

[9] Photo reproduced with permission from Robot Aviation www.robotaviation.com

[10] ID 158361999 © Dikiiy | Dreamstime.com

Fig. 6. Typical RPAS Scenarios (reproduced with permission)

2.3 C2 Link System

Figure 7 shows the conceptual architectural diagram and the boundary of concern for the C2 Link System.

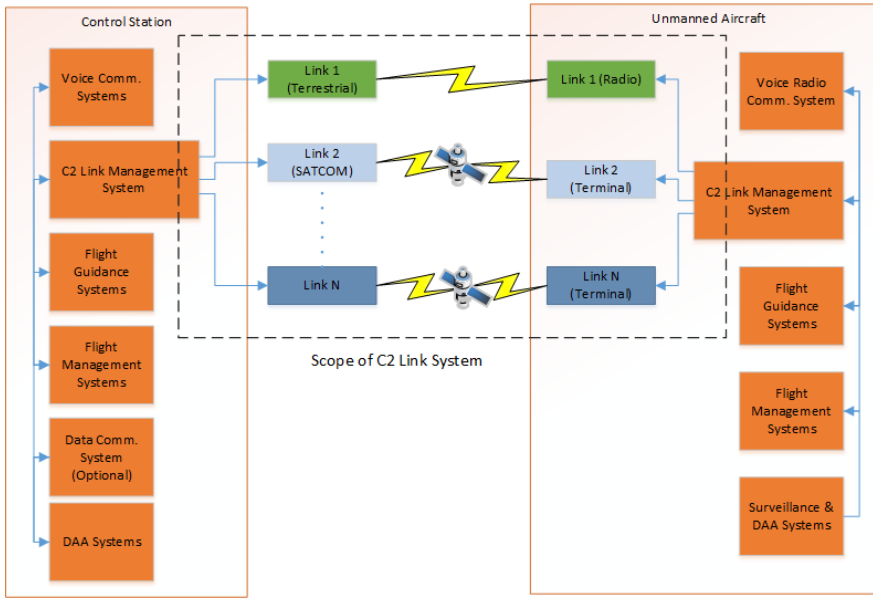


Fig. 7. C2 Link System[11]

The C2 Link System comprises one or more links (which may be terrestrial or satellite based) and those components of the Control Station and the UAV responsible for data transfer, security and routing of C2 messages. This sets the scope for safety analysis and the RTCA MASPS define requirements applicable to that scope only. These requirements will however be informed by the wider context in which the UAS will operate. Similarly, with security, the main focus is on establishing requirements for the C2 Link System but the analysis will consider the wider operational context to inform the refinement of requirements at lower levels.

[11] Adapted from an image taken from DO-377 RTCA (2019) and used with permission and copyright of RTCA, Inc.

3 Assessment Processes

3.1 Safety Assessment

3.1.1 Safety Assessment Process

The link system connects ground and air systems and so there is no existing safety assessment methodology that can be applied in its entirety to this scenario. The methodology chosen for this case study was the EUROCONTROL Safety Assessment Methodology (SAM). This provides a framework and tooling for the assessment and is broadly based on SAE ARP4754A EUROCONTROL (2010), which is used for aircraft certification.

Figure 8 shows the outline of the process applied. The process flows from top to bottom as the system progresses from concept through to design and build.

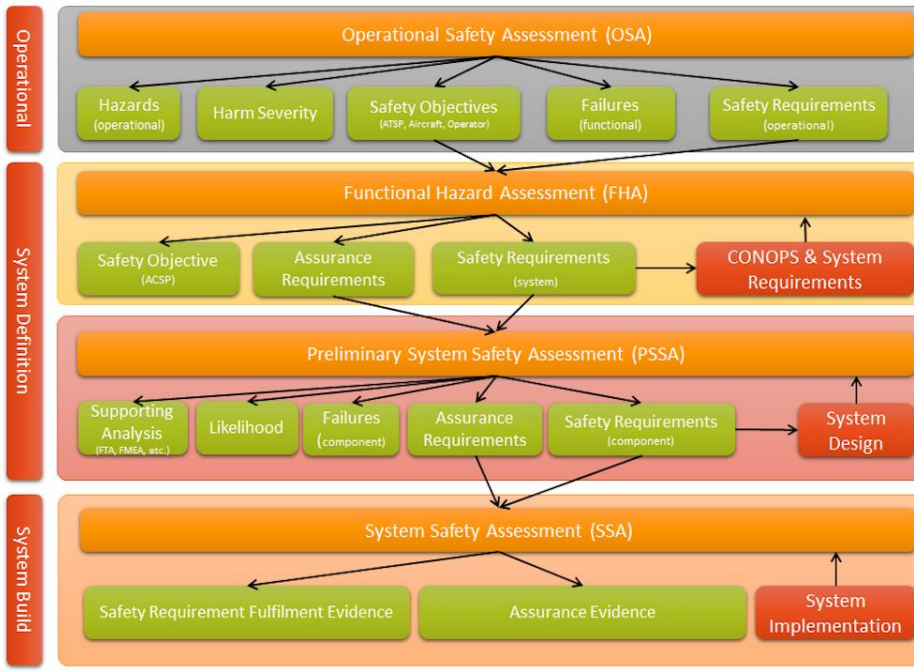


Fig. 8. Safety Assessment Process

Operational Safety Assessment (OSA)

The first step is the OSA shown in orange, which looks at the full operational context in order to determine the tolerability of identified operational hazards and to then set safety objectives and requirements to the C2 Link System as a whole.

Note that in this and subsequent text, terms marked in bold are key ontological terms that will become important when safety and security integration is considered later.

Outputs of this OSA process are shown in green:

- The operational **Hazards** the C2 Link System can give rise to along with the **Harm Severity** and the functional **Failures** that can cause the **Hazards**. This might be for example: *"Corrupted C2 Link System information exchange results in the UA not following the expected route."* Depending on the situation, the **Hazard Severity** may typically be qualitative, such as: "Hazardous".
- **Safety Objectives** for various **Stakeholders** are then derived from the **Harm Severity** so that, for example, the likelihood of a "Hazardous" situation should be "Extremely Remote";
- This then results in **Safety Requirements** required to meet the **Safety Objectives**, for example, quantitative integrity, latency and availability figures for the C2 Link system as a whole.

Functional Hazard Assessment (FHA)

The next step looks at a particular system to assess the hazards that the specific system can contribute to. For this case study, the system in scope was a single link provider's system, such as that covered by the light blue "Link 2" box in figure 7. That is, it includes the ground and air terminal components but does not cover the CS and UV Data Transfer, Security and Routing (DTSR) components or other link provider systems. This scope was chosen because, in a commercial setting, the C2 link is likely to be provided as a service separate from the RPAS operator and UV manufacturer. The study aimed to determine the safety requirements that would apply to such as link provider's service provision.

The FHA takes as input, the system Concept of Operations (CONOPS), the safety objective and system requirements. This process arrives at:

- **Safety Objectives** applicable to the link provider's system acting as the Air Ground Communications Service Provider (ACSP);
- **Safety Requirements** for the system such as quantitative integrity, latency and availability figure apportioned to the link itself;
- **Assurance Requirements** determining the level of rigour required in assuring that Safety Requirements have been met. These assurance requirements are proportional to the degree of **Harm Severity**;

As safety requirements augment the system requirements definition, the process may iterate a number of times as indicated by the flow of arrows into, and out of, the red CONOPS & System Requirements block.

Preliminary System Safety Assessment (PSSA)

The next step in the process is to consider whether the resulting design will, at least on paper, meet the **Safety Requirements** as specified in the Functional Hazard Assessment (FHA). This process analyses the System Design using a number of Supporting Analysis techniques such as Fault Tree Analysis (FTA) and Failures Mode and Effects Analysis (FMEA) that consider where component **Failures** may give rise to Operational Hazards and the **Likelihood** of those occurrences. The resulting analysis may determine that additional **Safety Requirements** need to be applied to particular system components, for example, that a particular component needs to be implemented in a diverse redundant configuration.

The process will also define assurance requirements for individual components where these are identified as contributors to Operational Hazards. For example, whether a software subsystem is to be assured to a particular software assurance level such as AL3/DAL C in ED-109A/DO-178C terminology.

System Safety Assessment (SSA)

The final step assesses the system "as built" and determines whether the implemented system will meet its Safety Requirements. This step takes as input, the System Implementation (eg. executable software / physical equipment) and assesses whether there is sufficient Fulfilment Evidence for the Safety Requirement and Assurance Evidence that the required level of rigour in building and testing the system has been adopted.

3.2 Security Assessment

3.2.1 Security Assessment Process

The Security Assessment Process is conducted over a number of stages using a number of different techniques. As recommended by the National Cyber Security Centre, NCSC (2016), different techniques are used to ensure variety in the assessment process allowing different perspectives of risk to be established and to avoid inherent biases that may exist in a single method.

Figure 9 shows the outline of the process applied. The process flows from top to bottom as the system progresses from concept through to design and build.

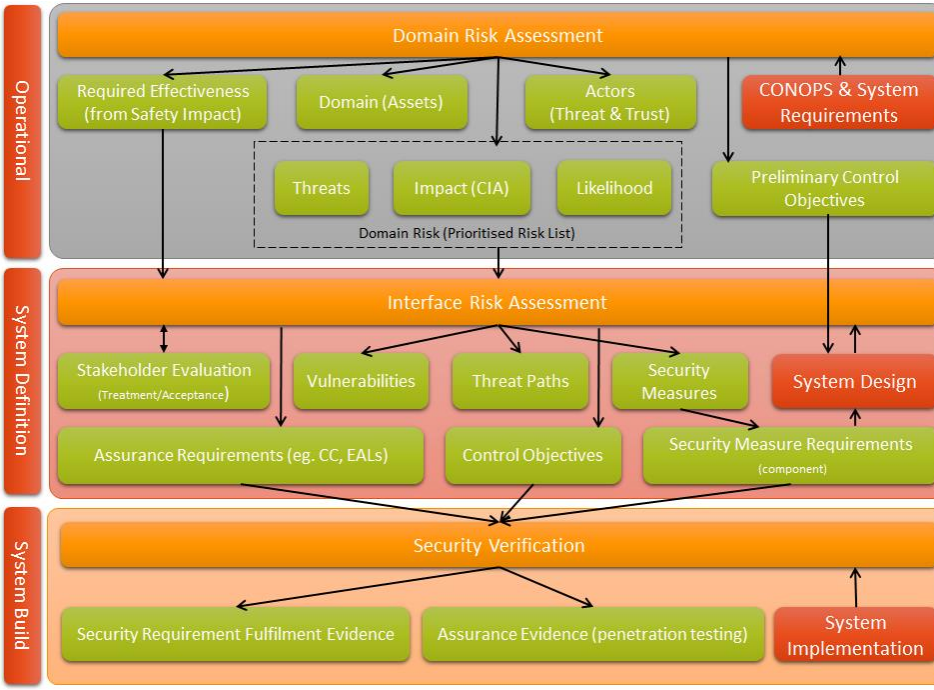


Fig. 9. Security Assessment Process

Domain Approach

This methodology assesses **Risk** via a combination of impact levels and **Threat** severities applied to **Assets**, also known as business **Domains**, in order to arrive at a list of risks graded in six levels ranging from Very Low to Very High. This method focuses on the people who may pose a threat (known as **Threat Actors**) as well as the **Threat Sources** who may influence and assist the Threat Actors to carry out an attack.

Domains are identified and their impact levels assessed in terms of Confidentiality, Integrity and Availability. **Domains** are grouped into Foci of Interest (FoI), in order to make the risk assessment more efficient, where the same threats are expected for a set of domains within a Focus of Interest. The Impact Levels of a FoI take the highest values for the domains it contains.

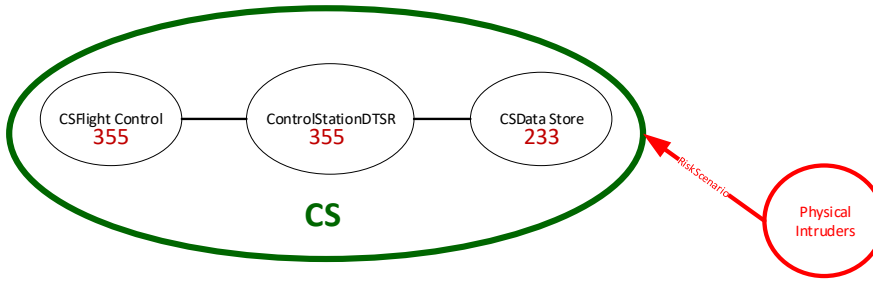


Fig. 10. Example Domain Risk Assessment

In the diagram, the ovals show how architectural assets are grouped into domains. **Threat Actors** are identified in circles and the severities of the threats (expressed as 3 single digit numbers in a sequence, eg. 355, for Confidentiality, Integrity and Availability respectively) they pose to the Foci of Interest are assessed, by judging their capabilities, motivations and influences.

This then leads to a 'Prioritised Risk List', describing all of the threats, impact levels and risk levels applying to each Focus of Interest. **Risk** can be described as being a function of: **Threat**, **Impact** and **Likelihood**. A typical domain risk would be as shown in table 1:

Table 1. Example Domain Risk

Threat Source	Threat Actor	Threat Actor Type	Compromise Method	Asset	Impact Type	Impact Level	Risk Level
Influence by Terrorists	Physical Intruder	As a physical intruder	Tampers with equipment in	Control Station	Availability	5	Very High

The HMG IAS 1 & 2 framework provides tools that allow assessments of likelihood to be calculated based on typical Threat Actors and the tool calculates the resulting Risk Level.

STRIDE Threat Model

System threat modelling allows analysis of a system from an adversary's perspective. It assumes that an adversary cannot attack a system without a method of supplying it with data. It further assumes that an adversary will not attack the system without assets of interest. The model assesses the system's entry points to determine the functionality that an adversary can exercise and what assets can be adversely affected. This approach allows development teams to enumerate attack goals or threats. **Vulnerabilities** are discovered when a threat is investigated and safeguards are proven to be insufficient.

System threat modelling is used in a systematic manner to focus security review effort and to categorise applicable system functions as security measures.

This categorisation informs and focuses security testing efforts in such a manner that penetration testing and automated code review efforts can be focused on these security measures. The steps within the assessment include:

- Define all domains and their level of trust;
- Determine trust boundaries and entry points;
- Determine interfaces and data flows between domains;
- Define possible types of threat on the defined interfaces;
- Determine **Threats** to each entity within the data flow: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service and Elevation of Privilege (STRIDE – see table 2);
- Describe security risks posed on each interface and identify preventative and detective security measures that can be implemented to mitigate the risk.

Table 2. STRIDE Threat Categories

Common Method	Security Characteristics	Description
Spoofing	Identity	Spoofing is where one entity claims an electronic identity of another entity. An example of identity spoofing is illegally accessing and then using another system's authentication information, such as certificate and private key.
Tampering	Integrity	Tampering involves the malicious modification of data or systems. Examples include: - Unauthorized changes made to data as it flows over a network, - Attempting to perform an injection attack or buffer overflow against online services, - Physically tampering with a device to remove private keys; - Installation of unapproved binaries and services.
Repudiation	Accountability	Repudiation is where an entity claims not to have performed an action on a system.
Information Disclosure	Confidentiality	Information Disclosure is where an entity views information that they are not supposed to have access to. An example of Information Disclosure is where an intruder reads data in transit between two devices on a network.
Denial of Service	Availability	Denial of Service is where an entity renders a service unavailable either intentionally or unintentionally.

Common Method	Security Characteristics	Description
Elevation of Privilege	Separation of Duty/Least Privilege	Elevation of Privilege is where an attacker intentionally increases their permissions within a system. This is typically performed immediately after exploiting another vulnerability, which provided the attacker limited access to a system; such as an injection attack or buffer overflow.

Security Control Objectives

Where a risk needs to be treated, Security **Control Objectives** treat the risk, and address domain and interface-related risks. A **Control Objective** may, for example, be:

> "The system must ensure that cryptographic keys and functions are adequately protected".

The Security Control Objectives are determined from the aggregation of all risk assessment techniques and will also be influenced by expert judgement and experience and the **Risk Owner's** risk appetite and risk treatment strategy.

The next step in the process is to develop tangible security requirements that will fulfil those Control Objectives in a manner that is quantifiable so that the sufficiency of the measures adopted can be demonstrated.

ED-203A Airworthiness Security Methods and Considerations

ED-203A provides a quantitative Risk Assessment methodology that is Aerospace specific and uses an approach where attackers use **Threat Paths** to facilitate attacks on Targets within a system. The following steps outline the process:

- **Security Measures** are defined and assigned quantitative risk reduction values. For example, a typical strong Security Measure could be attributed risk reduction values of 2,1,6 (scores for preparation means, window of opportunity and execution means respectively) and added together to give a risk reduction score of 9;
- **Security Measures** are attributed to the **Threat Path** and the combined risk reduction values are calculated. Note that combination is possible only where the measures are independent, eg. using security products from different vendors on different operating platforms;
- The combined risk reduction values should exceed the 'Required Effectiveness' values that are derived from the Safety Impact. For example, a **Difficulty of Attack** of 'Very High' would require the demonstration of a combined quantitative score of at least 25 (see figure 11)

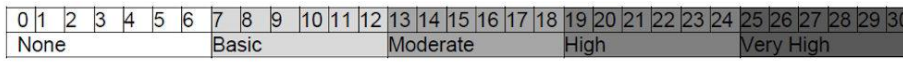


Fig. 11. ED-203A Difficulty of Attack Levels

This then provides a quantitative measure of confirming that sufficient measures have been established for the given Threat Path.

Risk Evaluation, Treatment and Acceptance

The ISO27000 set of standards for Information Security, specifically ISO27005:2018 - guidelines for information security risk management, has been extended to incorporate the application of system threat modelling onto the data flows between **Domains**. The **Threats** are then annotated with a system impact to describe risk.

Risk needs to be managed in two phases of the project: during design and during operations. During design, the threat assessment identifies candidate preventative and detective controls; these controls mitigate risk.

During design, risk acceptance and treatment determines if a risk is acceptable and if not, how the threat should be mitigated. The decisions relating to acceptable risk and treatment is taken in conjunction with the **Service Provider**. Where a risk is evaluated to be not acceptable, security control objectives to reduce risk are drafted and agreed.

During operations, the need for ongoing risk management falls within the scope of the Information Security Management System (ISMS), owned and run by an **Operating Organisation**. Where a risk is evaluated as being either acceptable or unacceptable this should be documented by the Operating Organisation in their Information Security Risk Register and communicated to all stakeholders, via a Security Management Forum.

4 Safety and Security Integration

From the safety and security processes described earlier, it can be seen that there are some similarities in terminology as both are effectively risk management processes, but there are differences in the terms used to express related or similar concepts.

One of the initiatives being conducted by the Safety-Critical Systems Club (SCSC) Data Safety Initiative Working Group (DSIWG) is to develop an ontology for data safety. This was initiated to validate the consistency of the concepts and their relationships used in the material published in the Data Safety Guidance document DSIWG (2019). Dave Banham created the original risk terms for the ontology simplifying work drawn from other sources that are developing risk models, such as the OMG "Operational Threat and Risk Information Sharing and Federation Model" OMG (2016). Dave also aligned the terms with the established standard for risk management: ISO31000 (ISO 2018) and the work was further refined from input from DSIWG members.

Although the focus is on data safety, the results of the analysis has required a general model of risk to be developed first, which provides a common "lingua franca" to which both risk-based Safety and Security disciplines can relate.

4.1 An Ontology for Risk

The ontological model is still currently under development, but the following figures illustrate emerging relationship diagrams:

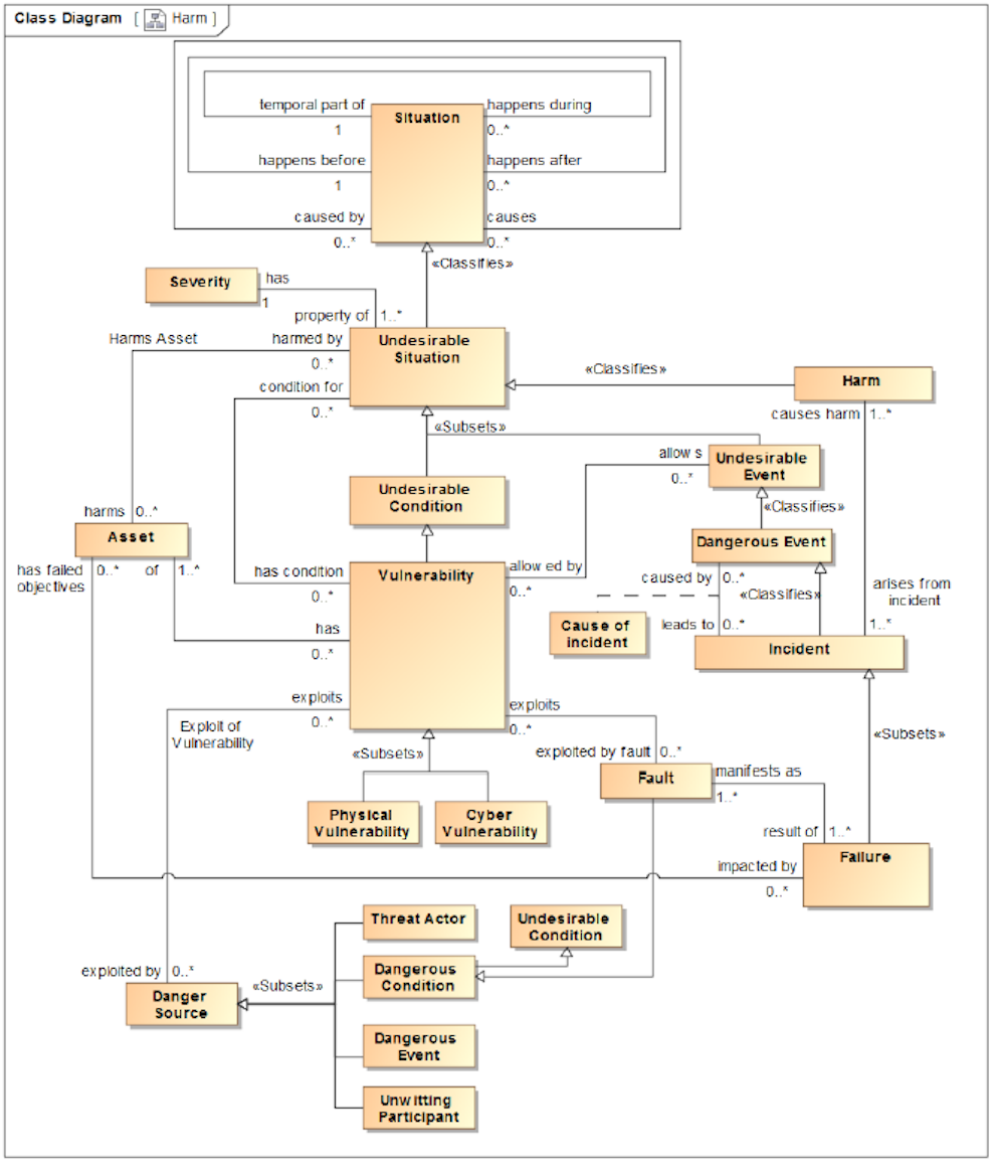


Fig. 12. Example relationship model for an Ontology of Risk (Harm)

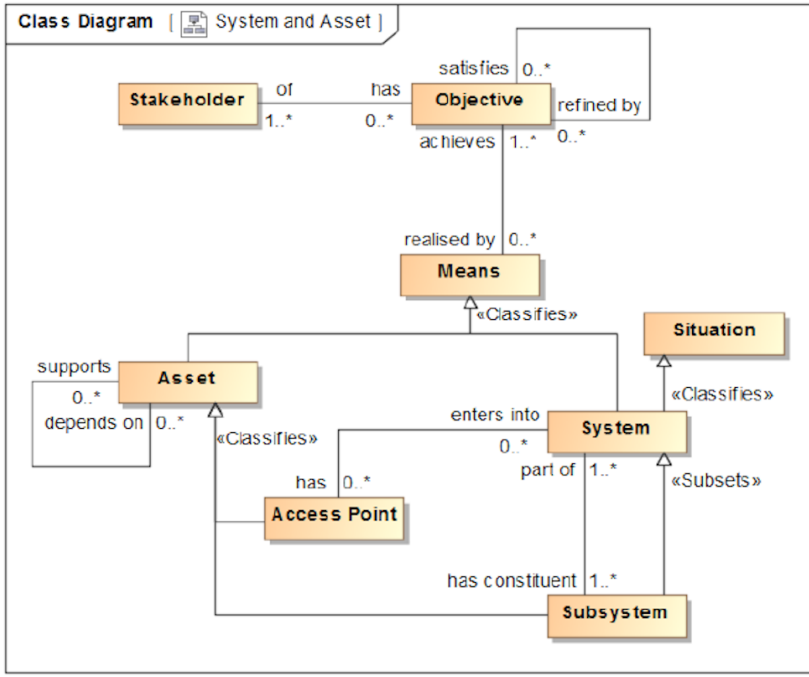


Fig. 13. Example relationship model for an Ontology of Risk (System and Asset)

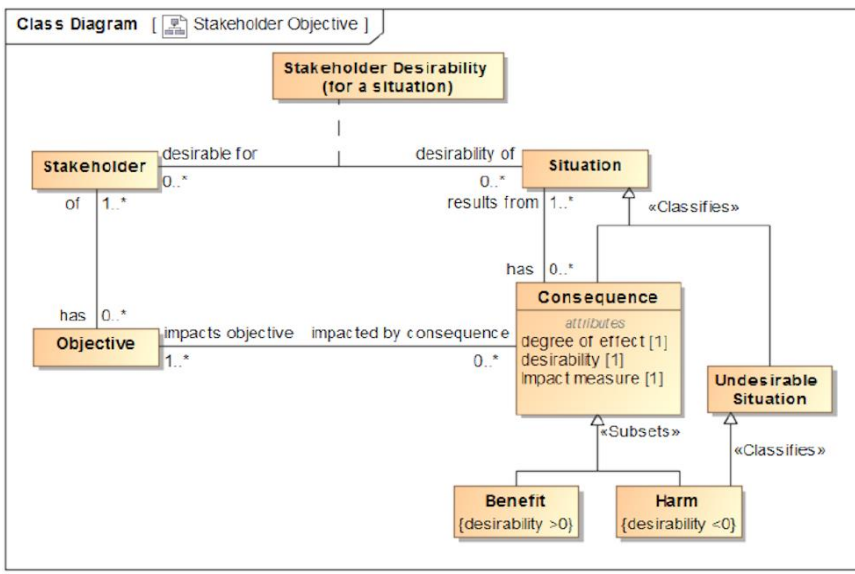


Fig. 14. Example relationship model for an Ontology of Risk (Stakeholder Objective)

From these models, statements about risk can be constructed in a precise and consistent manner. For example, a failure can be expressed as follows:

A **Failure** is a class of **Incident** that can impact a **System** and its **Subsystems** and the resulting **Harm** is that of a failure to meet the **Objectives** set out by the **Stakeholder** for the **System**.

4.2 Application of the Ontology

By expressing both the Safety and Security process terminologies using this common ontology, it is possible to show where the two processes can be integrated. The following tables expresses both the Safety (table 3) and Security (table 4) concepts in terms of the ontology.

Note that some terms used in the ontology do not import the exact same meaning that may be currently understood by practitioners in the safety/security domains so these need to be carefully defined and understood:

- **Desirability**: A metric describing the desirability of a situation or its resulting consequences. May be positive or negative where positive is desirable and negative is undesirable. Negative desirability for a consequence is equatable to **Severity** when the consequences are undesirable;
- **Harm**: a consequence of a situation that negatively impacts the objectives of stakeholders and therefore has negative desirability for those stakeholders. This definition encompasses harm to assets that may, for example, arise from security incidents as well as safety-related accidents. Note that this is broader than the expression "harm" as typically used in the safety discipline where it related specifically to the injury or death of people; in the ontology, such harm is handled as a Harm Category called Health Impact.
- **Incident**: An incident is a class of dangerous event (since more generally danger leads to harm), which is a class of undesirable situation. This definition applies equally to safety (e.g. a near-miss is a type of Incident) and security (a security breach is a type of Incident).
- **Vulnerability**: Assets can have vulnerabilities where a **Vulnerability** defines the conditions for an undesirable situation that may allow a dangerous event to occur that leads to an **Incident** or **Failure**. Failures manifest in the failed objectives of assets. Although more common in security terminology, vulnerability as expressed here encompasses aspects that are of concern to safety, for example, physical things are vulnerable to breaking because they are subject to fatigue, wear, and stress. Safety analysis often considers how frequently an asset may succumb to such a vulnerability (e.g. Mean time between failures).

Table 3. Ontological Expression of Safety Terms

Safety Process Term	Ontology Equivalent	Typical Ontological Expressions
Severity Class Scheme	RISK TREATMENT STRATEGY	A Risk Treatment Strategy is required by a Risk Owner who leads Risk Treatment and has Risk Treatment Objectives.
Failure	FAILURE	A Failure is a class of Incident that can impact a System and its Subsystems and the resulting Harm is that of a failure to meet the Objectives set out by the Stakeholder for the System.
Hazard	HAZARD	A Hazard is a type of Danger Source that exploits a Vulnerability.
Safety Objective	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy. A Safety Objective is a further classification of a Modify Risk Objective where the associated Harm has a Health Impact.
Assurance Requirement	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy. This objective defines the level of assurance to be applied when applying risk controls.
Safety Requirement	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy. A Safety Requirement is a class of risk modification objective that supports the fulfilment of a Safety Objective.

And the same process can be carried out for the Security terminology:

Table 4. Ontological Expression of Security Terms

Safety Process Term	Ontology Equivalent	Typical Ontological Expressions
Domains (Assets)	VALUED ASSET	A Valued Asset is a class of Asset that a Risk Owner values enough to establish Risk Reductions Objectives.
Actor (Threat)	THREAT ACTOR	A Threat Actor is a type of Actor, Stakeholder and Danger Source that exploits a Vulnerability. The Threat Actor perpetrates an Attack targeted at an Asset that has a Vulnerability.
Actor (Trusted)	STAKEHOLDER	A Stakeholder is a type of Responsible Agent that has Objectives and has desirability for a Situation.

Safety Process Term	Ontology Equivalent	Typical Ontological Expressions
Required Effectiveness	RISK TREATMENT OBJECTIVE	A Risk Treatment Objective is used to fulfil a Risk Treatment Strategy required by a Risk Owner.
Threat	THREAT	A Threat is a Dangerous Event that is a class of Undesirable Event (and therefore a Danger Source) that leads to an Incident that causes Harm.
Impact (CIA)	SEVERITY	Severity is a property of a Consequence arising from an Undesirable Situation the causes Harm to an Asset as a result of an Incident.
Likelihood	LIKELIHOOD	Likelihood is a property of a Potential Situation that also has a desirability for a Stakeholder.
Control Objective	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy.
Stakeholder Evaluation	RISK TREATMENT STRATEGY	A Risk Treatment Strategy has a Risk Treatment Objective that is subtyped by Avoid/Accept/Modify Risk Objective.
Vulnerability	VULNERABILITY	A Vulnerability defines the conditions for an Undesirable Situation that may allow a Dangerous Event to occur that leads to an Incident or Failure.
Threat Path	RISK SEEKING TECHNIQUE	A Threat Path is a Risk Seeking Technique used by a Threat Actor as a Risk Control specified by their Risk Treatment Objective[12].
Security Measure	RISK MITIGATION TECHNIQUE	A Risk Mitigation Technique is used by a Risk Mitigation Control to contain (lessen the impact of harmful events), protect (prevent the occurrence of harmful events) or recover (bring a failed/or harmed asset back to nominal state).
Assurance Requirement	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy. This objective defines the level of assurance to be applied when applying risk controls.
Security Measure Requirement	MODIFY RISK OBJECTIVE	A Modify Risk Objective is a class of Risk Treatment Objective that is used to fulfil a Risk Treatment Strategy.

[12] While a stakeholder establishes risk reduction objectives to decrease the likelihood and/or impact of harm to valued assets, a Threat Actor will also have risk modification objectives to increase the likelihood and/or impact of harm to a valued asset. A Threat Actor may also have objectives to decrease the likelihood of detection, thus reducing the risk of harm to themselves, as a valued asset.

4.3 Integration

From the tables it can be seen that, when expressed in terms of the risk ontology, there are some areas of cross-over and commonality between safety and security. While these may have been suspected or intuited, the previous section demonstrates equivalency in a more rigorous manner.

A key observation from the tables is that **Vulnerability** emerges as a key concept. The term works well across both safety and security. More colloquially, in security, it is vulnerabilities in assets that malicious 3rd parties exploit in order perpetrate attacks that will result in losses (or more precisely "harm") for a stakeholder. In safety, a vulnerability can be considered as a state, feature or issue with a system that in combination with a particular event (which effectively inadvertently exploits the vulnerability), could give rise to injury or death.

Similarly, the concept of **Danger Source** that exploits vulnerabilities, works equally as a threat path for malicious attackers and safety hazards. Also arising from this analysis is a common definition of Risk that applies equally in both safety and security domains:

- **Risk** is a measure of a Stakeholder's desirability for a Potential Situation that has a potential Consequence impacting a Valued Asset. The Potential Situation has a Likelihood, and the Consequence has a Desirability.

For example, a hacker (the stakeholder) takes risks in attempting to gain access to a system (a valued asset). Being caught (a potential situation) has a likelihood (low, for example, if conducted anonymously but higher if attempting to gain access to a secure property person). The consequences of being caught could be confiscation of assets, fines and imprisonment, which will have very low desirability for the hacker.

In both disciplines, where risk is not acceptable, Risk Modification Objectives are established. Risk Mitigation Controls will then be defined to meet those objectives and these will be implemented by using a Risk Mitigation Technique.

In both safety and security, it is usual to keep a log of risks in the form of a hazard log and risk register respectively. It is proposed that these can be combined to give a single picture of the overall risk. To be consistent with the ontology, this would be called a Danger Source Vulnerability Exploitation Log (DVE Log) and would have the following core fields:

Table 5. Danger Source Vulnerability Exploitation Log (DVE Log)

Field	Description
Identifier	A unique identifier for an individual DVE entry.
Valued Asset	A description and scoping of the asset that is at risk.
Danger Source	A description of the hazard, a known danger source, or threat path that may exploit a vulnerability. For example, loss of availability for a link or physical connection to ground based CS system by a 3rd party.
Vulnerability	The undesirable condition that can be exploited by a danger source. This might, for example, be software defects/programming errors or unguarded network equipment in public areas.
Likelihood	The likelihood that a situation could potentially arise that has harmful consequences to a valued asset.
Severity	A measure of the severity of each consequence of harm arising from an exploited vulnerability.
Risk Class	A classification of the Likelihood and Severity, in accordance with the risk holder's risk treatment strategy.
Risk Mitigation Control	The risk mitigation controls that are to be used to reduce the identified risk (these typically reduce the likelihood and/or severity of an exploit).
Residual Risk Class	A reassessment of the classification of risk assuming risk mitigation control are in place.
Risk Mitigation Control Evidence	Evidence that confirms that risk mitigation controls are in place (e.g. review, test, evaluation certificates etc.).

4.4 Safety and Security Interactions

Having a common risk language means that Safety and Security disciplines have a collaborative basis on which to discuss and assess risk. A shared repository of concerns (in the DVE Log) also promotes the continual sharing and close integration of Safety and Security risk considerations.

As is the case already with both safety and security disciplines, the process is highly iterative. If a security measure is added to reduce risk of an identified threat path, the revised design will then be reassessed from a safety perspective to consider if failures in that new feature could result in a form of safety-related harm (health impact). Similarly, if a safety requirement modifies or introduces new features into the design then security risk assessments will be undertaken to assess the risk that the feature could be exploited and result in harm (in the broader sense – defined earlier) to assets.

Note also that the impact levels are informed in both directions between safety and security. In the case study, the safety assessment informs the security assessment when assessing the impact of a security incident. For example, if an inadvertent loss of link has Major severity then this is, as least, the same impact from

a security perspective as a 3rd party can also cause link loss. (eg. through a denial of service attack). The security assessment can however, then refine this initial assessment. For example, it can be argued that a corrupted link could cause a UAV to move in an uncommanded direction that may result in ground collision or bring the vehicle into conflict with other air users in controlled airspace. From a safety perspective, a corrupted command to manoeuvre the UAV is arguably Hazardous as the pilot can terminate the link to force the UAV into link loss protocols. ATC will also ensure other aircraft are separated from a UAV so that separation minima are maintained. However, a 3rd party accessing a link and masquerading as the pilot could result in more serious consequences as the UAV could be deliberately directed into a populated area, critical infrastructure or be directed to 'chase' another air user causing potentially Catastrophic harm in these cases. Suitable security measures are therefore required to address the security risks to mitigate against this level of outcome.

4.5 Assurance Considerations

The case study has an interesting outcome in that the security concerns lead to more severe outcomes than inadvertent failures. In other words, a malicious 3rd party is likely to be able to do more harm than a latent non-goal-seeking[13] software defect. However, it does not follow that this should automatically result in a higher level of required safety assurance. Safety assurance is, in essence, to demonstrate with sufficient confidence that safety-related functions will operate, as intended, and are sufficiently free from non-goal-seeking failures. This is not always suitable for assuring against security risk that are not aleatory in nature, Alexander et al (2011). For example, an attacker, having breached one defence, may go on to exploit it to breach another. The assurance approach for this case study is therefore based on the following principles:

- Safety assurance shall be conducted at level that is commensurate with the impact severity of non-goal-seeking failures of the system, including non-goal-seeking failures of the security infrastructure itself;
- Security assurance shall be conducted at a level that is commensurate with the impact severity of failures arising from goal-seeking threat actors as well as inadvertent security failures of non-malicious actors (eg. sending an email to wrong recipient).

[13] Any specific inadvertent software defect implicitly has no associated Threat Objective.

5 Conclusions

This case study has provided an example of an emerging technology where there are currently no definitive standards for safety and security risk management. As the technology under study links both ground and air systems, the boundary of concern now spans a larger domain than is covered by prevailing standards, many of which consider only the safety and security of the aircraft or ground systems in isolation.

The study shows that safety and security considerations cannot be easily separated and treated in isolation; for example, the security risk assessment needs to consider the safety impacts of security breaches, and the safety assessment needs to consider the safety impact of failures of the resulting security infrastructure.

In lieu of any definitive guidance, the study describes the analysis approaches for safety and security that were adopted; these were based on current industry safety standards development and security best practices.

The study then uses a risk ontology being developed by the Data Safety Initiative Working Group to recast the terms used in both the safety and security analysis processes. By expressing safety and security terms using a normalised and consistent language for safety and security risk management, the study draws clearer correlations between the concepts and activities employed by the two disciplines.

In doing so, the study shows how the outputs of the two disciplines can be integrated and how risks can be captured in a shared log called a Danger Source Vulnerability Exploitation Log (DVE Log). It is argued that a log based on a common risk language provides a much better collaborative basis on which safety and security personnel can discuss and assess risk.

The study concludes with the recommended principles to be applied when assuring the safety and security activities taken to manage identified risk.

Although the scope of the system under consideration for the study is novel and has a specific focus on C2 links for RPAS, it is hoped that the documented approach to safety and security integration provides a more general framework that provides much wider applicability for other types of developments and operations.

6 Further Work

The ontological model that this work is based on is still being developed and so there will be further work required to align with the model as it is refined. It is also hoped to conduct further detailed field studies of the use of the DVE Log and to assess how well this promote collaboration and alignment of safety and security processes.

References

Alexander, et al. (2011) Security Assurance Cases: Motivation and the State of the Art, 07/04/2011

DSIWG (2019) Data Safety Guidance, The Data Safety Initiative Working Group, ISBN: 9781793375766

EASA (2012) CS-23 Certification Specification for Normal, Utility, Aerobatic and Commuter Aeroplanes

EASA (2017) Commission Implementing Regulation (EU) 2017/373, laying down common requirements for providers of air traffic management/air navigation services and other air traffic management network functions and their oversight, 01/03/2017.

EUROCAE (2010) EUROCAE ED-79A / SAE ARP 4754A Guidelines For Development Of Civil Aircraft And Systems, December 2010

EUROCAE (2018) ED203A - Airworthiness Security Methods and Considerations, June 2018

HMG (2012) UK HMG NCSC Information Assurance Standards 1 & 2 (including Supplement), April 2012.

ISO (2018) ISO/IEC 27005:2018 Information technology — Security techniques — Information security risk management

ISO (2009) (ISO/IEC 15408 Information technology — Security techniques — Evaluation criteria for IT security (Common Criteria)

ISO (2013) ISO/IEC 27001:2013 Information Security Management System

ISO (2018) ISO 31000:2018 Risk management – Guidelines, provides principles, framework and a process for managing risk, 2018

JARUS (2015) AMC RPAS.1309 Safety Assessment of Remotely Piloted Aircraft Systems, Issue 2, November 2015.

JARUS (2018) JARUS Glossary, JAR_DEL_Glossary_D.4 v0.7, 18/07/2018

JARUS (2019) JARUS guidelines on Specific Operations Risk Assessment (SORA), JAR-DEL-WG6-D.04, v2.0, 30/01/2019

Microsoft (2009) The STRIDE Threat Model, https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20), accessed 17/10/2019

OMG (2016) Operational Threat and Risk Information Sharing and Federation Model, May 2016.

NCSC (2016) Variety in risk information, https://www.ncsc.gov.uk/collection/risk-management-collection/essential-topics/variety-risk-information, accessed 19/10/2019

RTCA (2019) DO-377, Minimum Aviation System Performance Standards for C2 Link Systems Supporting Operations of Unmanned Aircraft Systems in U.S. Airspace, 21/03/2019

IEC TR 63069, Security Environments and Security-Risk Analysis

Peter Bernard Ladkin

CAUSALIS LIMITED / CAUSALIS IngenieurGmbH

Bielefeld, Germany

Abstract *In May 2019, the IEC published a guide to combining cybersecurity and safety for industrial automation and control systems (IACS), IEC TR 63069. I consider critically two main concepts in the guide: an overly-strong notion of "Security Environment" (SE), and an accompanying incomplete type of security-risk analysis called "threat-risk assessment <security>" [sic]. A simple example from experience illustrates the weaknesses.*

1 Introduction

IEC TR 63069 *"Industrial-process measurement, control and automation – Framework for functional safety and security"* [IEC 63069] proposes a "framework" for accommodating the interaction between functional safety and cybersecurity in industrial automation and control systems (IACS). A key proposal is that a "security environment" (SE) shall be established. Within that SE, it is suggested that safety engineers are to perform their risk analysis[1] (RiskAn) and risk evaluation under the assumption that cybersecurity is assured by the SE. This recommendation is explicitly formulated as "Guiding Principle 1":

1 Risk analysis is defined in [IEC 51]; see in particular Figure 2. Formally, it is the systematic use of available information to identify hazards and to estimate the risk [IEC 51 subclause 3.9]. "Risk" is defined to be the combination of the "probability" (I would say likelihood) of occurrence of harm with the severity of that harm [IEC 51 subclause 3.8]. It is understood that the probability of occurrence encapsulates the exposure to a hazardous situation, the occurrence of a hazardous event, and any possibility thereby to avoid or limit the harm.

© Peter Bernard Ladkin 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

> 1) Guiding principle 1: protection of safety implementations. Security countermeasures should effectively prevent or guard against adverse impacts of threats to safety-related systems and their implemented safety functions. *Evaluations of safety functions should be based on the assumption of effective (security) countermeasures.* [IEC 63069 Clause 5, my emphasis]

The definition of an SE is an "area of consideration where all relevant security countermeasures are in place and effective" [IEC 63069 subclause 3.1.5]. This all raises a number of questions.

2 Some Questions

In this paper, I am concerned not with security in general (involving, say, physical security such as fences, guard-dogs, personnel validation and so on, which have their place in IACS security) but alone with cybersecurity. Cybersecurity is often considered to be founded on the so-called "CIA triad" or some extension of it: Confidentiality, Integrity and Availability [Lad 17, Chapter 8]. IACS are nowadays called Operational Technology, OT, to contrast with Information Technology, IT, because their primary purpose is to cause and regulate system actions, not to organise data (except in so far as that data is necessary for appropriate system action).

For safety-related OT, the "IA" parts of "CIA", that is, the integrity of IACS systems and the continuing availability of safety-related functionality, are argued to be primary, with any confidentiality concerns of lesser interest [op.cit.]. In the circumstances in which (positive) safety is achieved by means of safety functions (as required in the functional safety standard IEC 61508 – see below), we want safety functions to continue to behave as designed and implemented (a version of integrity) when they are operative, and we want them to continue to do their job during system operation (a version of availability).

It is often helpful to speak of the obverse to cybersecurity, namely cyberinsecurity. Safety-related cyberinsecurity consists of vulnerabilities by means of which safety-related functions may be caused to behave differently from designed, intended, and/or required. There are many types of vulnerabilities. Some of them are avoidable – for example, so-called buffer-overflow vulnerabilities in key software, in which the execution stack may be overwritten by outside input, and which overwriting may contain code which does something other than the system "should" be doing. Such buffer-overflow vulnerabilities are avoided by appropriately careful software development. Some vulnerabilities are unavoidable – for example, where human controllers affect the operation of the system, a malfeasant person can impersonate a controller by giving identification credentials which have been purloined and which the system determines to be valid, and then proceed to subvert the proper functioning of the system. Or the controller himherself can be malfeasant. Any circumstances in which human input is required inevitably remain vulnerable to malfeasance.

I am concerned, as is IEC TR 63069, with cybersecurity phenomena as they interact with system safety concerns. In the conception of system safety proposed by the international standard for functional safety of E/E/PE-based[2] systems, IEC 61508 [IEC 61508], hazards are to be identified, analysed and evaluated. Hazards are, roughly, circumstances (either states of the system + system environment, or events) which could result in harm (to people, to the general environment, or to property). The analysis of a hazard consists in determining the magnitude of the harm which could result (its severity) and the likelihood of such harm occurring. If this combination of likelihood and severity ("risk") is deemed to be unacceptable[3], then mitigation of the hazard must be engineered. Mitigation could consist in a redesign, or it could consist of a protective function, a so-called "safety function", which reduces the severity or likelihood of harm or both to an acceptable level. Such a safety function is assigned a reliability requirement, a so-called "safety integrity level" (the terminology here is problematic, but this is not the topic of this paper). Much of the rest of IEC 61508 is devoted to specifying how safety functions are to be built and assessed to have attained the specified level of reliability. For example, the software-development part of the standard has between 50 and 60 documentation requirements. The standard does not say how such software is to be put together, but it does require the accompanying documentation. There are similar constraints on E/E/PE hardware.

There are three ways in which cyberinsecurity can adversely affect safety, under such a conception. The first two stem from the possibility that exploitation of such insecurity could increase the frequency or intensity of certain dangerous failures[4].

- First, suppose that the rate of such dangerous failure is judged to be acceptable without such exploitation, and then becomes unacceptable when such exploitation is considered. Then a safety function will be required, which was not required when malfeasant exploitation was not considered.
- Second, a safety function aimed at mitigating specific types of dangerous failure could fail to cope acceptably with an increase of the rate of occurrence of such failures through exploitation of vulnerabilities.

---

2 "E/E/PE" stands for "electrical/electronic/programmable electronic" and is intended to include systems dependent for their function on digital-electronic devices. The coinage stems from the functional-safety standard IEC 61508 [IEC 61508].

3 Acceptability of a risk is in IEC 61508 a social judgement, outside of the scope of the standard itself. It is not specified how this judgement is to be arrived at.

4 A dangerous failure is a "*failure of an element and/or subsystem and/or system that plays a part in implementing the safety function that: a) prevents a safety function from operating when required (demand mode) or causes a safety function to fail (continuous mode) such that the EUC is put into a hazardous or potentially hazardous state; or b) decreases the probability that the safety function operates correctly when required.*" [IEC 61508]

- Third, exploitation could subvert the operation of a safety function, so that it is no longer reliably functional to the required level.

The focus of cybersecurity+safety considerations are, then, on considering these possibilities and what can be done about them. This is supposedly the aim of IEC TR 63069 and its concept of Security Environment. It is thus a good idea to try to determine what a "Security Environment" may be. For this, we go to the definition. An SE is defined to be an "*area of consideration where all relevant security countermeasures are in place and effective*" [IEC 63069 subclause 3.1.5]. Questions immediately arise as to what this may mean.

Inter alia, an SE is supposed to establish an environment which satisfies Guiding Principle 1. I shall consider below the definition of the notion "Security Environment" in [ETSI 1021651], and suggest this notion is more appropriate, but the ETSI notion is insufficient to satisfy Guiding Principle 1 (see below).

Let me consider the definition in subclause 3.1.5 phase by phrase.

2.1 "Area of consideration"

First of all, what here is an "area of consideration"? Is it a geographical location (things called "security areas" usually are)? Is it some sort of geometrical object imposed upon the ground or in a working volume, as in some recent safety standards in robotics? Is it something like a "zone", a "grouping of logical or physical assets that share common security requirements" [IEC 62443 subclause 3.2.117]? The only (partial) clarification is that an SE is not a zone (Note 2 to Subclause 4.2).

2.2 "Effective" countermeasures

In an SE, relevant "security countermeasures"[5] are to be "in place and effective". What is it for cybersecurity measures to be "effective"? Part of it must surely mean that they are active (there would be little point in having cybersecurity measures in place but not activated). But what else?

Consider, for example, a measure S intended to provide only authorised access to control-system-command execution. Suppose that Person A is "authorised". And that Person M, who is not authorised, obtains A's credentials (those belongings assigned to and characteristics of A which S checks in order to grant

5 The notion of "security countermeasure" is an odd choice of phrase. A measure to counter security is the opposite of what is meant. What is meant is "cybersecurity measure" or (if one must) "cyberinsecurity countermeasure". I shall say "cybersecurity measures".

A command-execution access). M can use these credentials to obtain access to command execution. This can theoretically happen with any such S.

So S cannot infallibly guarantee access to those people and only to those people whom the owners/operators intend should have it. Is S to be counted as "effective"? Or is S ineffective because it is not perfect? Does "effective" mean "pretty d**n good" (that is, devised and implemented by the top engineering scientists in the field); does it mean "fulfils its specification perfectly" (which notoriously raises the further question: how good is the specification?); does it mean "fulfils the design intent perfectly" (that is, it invariably excludes all M's)? We are not told.[6]

3 The elaborations

3.1 Security environments (SE)

In IEC documents, an explicit definition in clause 3, Terms and Definitions, is (supposed to be) a precise source of meaning. But besides the explicit definition of SE in subclause 3.1.5, there is a further elaboration in IEC TR 63069 of what an SE is intended to be. It is just a set, namely a set of countermeasures:

The security environment .... is understood as the overall collection of countermeasures required to ensure an efficiently protected environment for operations of the safety functions, however it is not limited to protect the safety functions only.

The SE includes but is not limited to the following countermeasures:

- all countermeasures protecting the perimeters of the security environment under evaluation,
- all countermeasures concerning the interaction between different functional Units at the security environment,
- all countermeasures to be applied at the functional units within the security environment.

NOTE 1: In practical applications, countermeasures might not be exclusive for the safety functions.
NOTE 2: The security environment is not the same as a "zone" as described in [IEC 62443] series.
NOTE 3: The security environment can incorporate the "defence in depth" strategy [IEC 62443-1-1 Chapter 5.4] for achieving sufficient resilience of an application.

6 A reviewer suggested I might try to offer a "workable definition". There indeed needs to be some notion of an appropriate implementation/activation of a cybersecurity measure. I would propose that the measure fulfil its explicit requirements specification (RS), that additionally the RS be checked explicitly for situations it may not cover, and that the behaviour of the measure during system operation be monitored, and likely some other things. I would not talk of an "effective cybersecurity measure" because I find such a formulation inexact and facile.

This elaboration is, however, confused. An SE is a set (or "collection"). But it has a "perimeter", according to the first bullet point above. Sets have no perimeter. A zone has a perimeter, but an SE is not a zone (see NOTE 2 in the quote). Since a SE "includes … countermeasures", it would be reasonable to suppose it is consequent to identifying such countermeasures. Countermeasures are usually the result of some kind of analytical cyberinsecurity identification and mitigation process. In this paper, I shall call such a process a "security-risk analysis" (SRA)[7]. So an SRA defines or helps to define the collection of cybersecurity measures that constitute the SE[8].

3.2 Assessing Cyberinsecurity Risks

The term used in IEC TR 63069 for a form of SRA is "threat-risk assessment <security>" [IEC 63069 subclause 7.3]. This is one of many such terms in the standards literature.

3.2.1 Cybersecurity

In the engineering-scientific literature, cyberinsecurity analysis is often called "security risk analysis". ETSI[9] calls it "risk-based security assessment" [ETSI 203251]. Elsewhere, ETSI calls it "Threat, Vulnerability, Risk Analysis (TVRA)" [ETSI 1021651]. Some standards associated with IEC TR 63069 call it a "security risk assessment", and purport to explain how to do one [IEC 62443-3-2]. But even within that (draft) document it speaks instead of "cybersecurity risk assessment".

It is not just a matter of the terms chosen for such analysis, but of the content of such analyses. Different processes and procedures are proposed as constituting such analyses. I consider below the particular procedure defined (in part) in IEC TR 63069. First, I introduce another notion of SE, namely that defined in [ETSI 1021651]:

> 5.1.1.1 Security environment
> The security environment describes the security aspects of the environment in which

7 I regard the hyphen as essential. I have argued in [La17 Chapter 14] that security-risk is something other than a combination of probability and severity, because chances of occurrence change rapidly and discretely with the stages leading towards an attack. Hence the standard IEC concept of "risk" as defined in the International Electrotechnical Vocabulary [IEV] cannot be used unmodified.

8 A reviewer asked whether this was made clear in the document itself. In my reading, the answer is no, it is not made clear.

9 ETSI is "a European Standards Organization (ESO). We are the recognized regional standards body dealing with telecommunications, broadcasting and other electronic communications networks and services." (from www.etsi.org)

the asset is intended to be used. It shall include:
• Security assumptions:
- the intended use of the implementation;
- the physical, user and connection aspects of the environment in which an implementation will operate.
• Assets:
- the assets with which the asset under analysis will interact with;
- the nature of the asset's interaction with other assets.
• Threats and threat agents:
- all threats against which specific protection is required within either the implementation of a standard or its expected environment;
- the threat agents that will be used to enact the identified threats.
• Organizational security policies:
- any security policies or rules with which an implementation of a standard shall comply.

This definition is more helpful to those implementing cybersecurity measures than considering an SE to be a possibly-arbitrary collection of measures. It relativises the SE to an explicit collection of threats and threat-agents: these-and-these threats/agents are countered by the SE, but not necessarily other threats. So, to take my access-purloining example above, in the ETSI conception you consider explicitly A and M and A's set of credentials when considering this threat, and you would consider whether those credentials suffice for the current purpose, or whether they need to be strengthened. In any case, the "residual vulnerability", as we may call it, is explicit in the ETSI formulation.

It is prima facie clear in the ETSI conception that you cannot necessarily expect protection against an unlisted threat or agent. Furthermore, there is some hope of showing relative completeness of your SE, for you could perform, for example, BAN-logic[10] analysis of your measures to try to assure they suffice (but use of such logics is not necessarily easy for non-trivial protocols).

Notice that, on the basis of the ETSI definition of SE, one cannot assume that cybersecurity is assured, for there might well be threats not listed in your SE and thereby maybe not countered by measures in it. Hence, a safety engineer performing a risk analysis can at most presume that the system is secure against the enumerated threats, and will have to consider the possibility of other threats unforeseen in the SE definition and the possible effects of these threats upon safety functions. That is, cyberinsecurity must still be considered by a safety engineer performing a "Hazard and Risk Analysis" according to [IEC 61508-1 subclause 7.2].

3.3 What IEC TR 63069 Recommends

3.3.1 "Threat-risk analysis <security>"

10 BAN-logic is a logic explicitly designed by Mike Burroughs, Martín Abadi and Roger Needham to demonstrate the adequacy of cryptographic protocols for authetication [BuAbNe 89].

IEC TR 63069 specifies some conditions on security-risk analysis in subclause 7.3.2[11]:

> The following should be applied during the threat-risk assessment process:
> 1) The detailed safety description should be prepared;
> 2) The threats and their potential impacts to [sic] the security environment should be identified;
> 3) Hazards that can result from attacks should be identified;
> 4) The system operations and/or the system architecture and how they could introduce vulnerabilities should be identified;
> 5) Based on the information collected, the necessary security countermeasures establishing an effective security environment should be defined and identified.

[IEC TE 63069 subclause 7.3.2]. Various phenomena in this scheme raise immediate questions, some of which I address below. First some brief comments.

First, it is unclear to what artifact the term "detailed safety description" refers. Such a term is unknown from (say) IEC 61508 or other associated standards.

Second, there is a specific IEC meaning to the word "should". To say something "should" be done means "it is highly recommended". To indicate something is mandatory, the term "shall" is used. Taken literally, there is nothing you *have* to do in the type of security-risk analysis described in this quote, just things which are recommended.

Third, 7.3.2 step 2 seems to imply that there exists a "security environment" already, such that one can identify "threats and their potential impacts" on it. But if the "security environment" is to be identified with the "collection of all countermeasures", as discussed above, it is surely defined as and when those countermeasures are defined, which is in the *later* 7.3.2 step 5. The "cart is put before the horse".

3.3.2 The overall analysis process

There seem to me to be three key components to the IEC TR 63069 process. First, a security-risk analysis is required. Second, on the basis of such analysis, a SE, defined as a collection of formulated cybersecurity measures, is defined. Third, this SE is to fulfil the condition ("Guiding Principle 1") that it should be adequate to allow safety engineers to perform their analyses and system development under that assumption that cybersecurity is assured by the SE.

If we follow this guidance, it seems to me we are led to something like the following scheme:

> i). Perform a SRA. Formulate cybersecurity requirements on the basis of the SRA, as well as cybersecurity measures to assure that the cybersecurity requirements are fulfilled.

11 I shall refer below to the five steps in the subclause as "7.3.2 step 1" to "7.3.2 step 5".

ii). Define a SE (= the explicit collection of those cybersecurity measures).
iii). Perform a (safety) RiskAn assuming that cybersecurity is assured by that SE.
iv). Then follow the rest of your system development based as usual on the results of that RiskAn.

Steps i) and iv) here are not explicitly listed in the document IEC TR 63069; this is understandable in the case of step iv), because such development is handled in other standards such as IEC 61508. It is less understandable for step I). It seems to me that, before one formulates cybersecurity measures, one needs to figure out what the need is for those measures; such a need is typically expressed in what are called "requirements"; hence the formulation of "security requirements" as a result of the SRA and precursor to derivation of cybersecurity measures. There seems to be little or nothing in IEC TR 63069 about security requirements as a result of SRA and precursor of the formulation of cybersecurity measures.

I consider step iii) very dubious. It is only reasonable to assume in a RiskAn that cybersecurity is assured if indeed there has been an attempt to ensure that this is so, and the attempt has been successful. But there is no suggestion, anywhere in IEC TR 63069 that I can see, that the SRA as performed has to be evaluated for such completeness. Just assuming that the defined SE suffices, without making an explicit effort to check it, I regard as inappropriately rash. A further step needs to be inserted:

ii-a). Show that the SE is complete. That is, that it thwarts all cyberattacks.

Worldwide, one or two such systems based on a successful step ii-a) are believed to exist in special environments [Sch 16]. For civilian systems in everyday use, I and others consider such a step in the current state of the art to be unrealistic [Tho 19]. I suggest it is more realistic, even though possibly very difficult, to perform an ETSI-type TVRA and achieve the following:

ii-b). Show that the (ETSI-type) SE is relatively complete. That is, it successfully thwarts cyberattacks deriving from the enumerated threats.

But then, if step ii-b) is followed, Step iii) becomes inadequate: as noted above, a safety engineer performing a RiskAn will need to consider possible cyber-insecurities which are not listed in the (ETSI-type) SE.

There seems to be no requirement for any check on completeness or relative completeness in the analytical process described in subclause 7.3.2. Relative completeness can maybe be attained if one relativises the SE explicitly to those threats identified in 7.3.2 step 2, as is done in the ETSI conception. That would be an extra step in 7.3.2, formulated as in step ii-b) above. But it would then follow that Guiding Principle 1 must (at best) be relativised, to only assume "effective" countermeasures to those explicitly-identified threats.

Additionally, it would be appropriate for the "effectiveness" of the

countermeasures to be assessed and assured[12]. Such an assessment activity does not explicitly occur in 7.3.2 (nor have I formulated it – yet – in my steps i) - iv)). Such assessment would occur after the vulnerabilities have been identified (7.3.2 step 4)) and measures formulated (7.3.2 step 5)). It follows there needs to be a step for the assessment which follows these two, say:

> 6) The adequacy ("effectiveness") of the measures identified in 5) for counteracting the security vulnerabilities in 4) should be assessed.

Going back to 7.3.2 step 3), it says that a (limited) hazard identification (HazID) is to be performed. A HazID is a process which is performed by safety engineers. It follows that the process of "threat-risk assessment <security>" defined in IEC TR 63069 involves some safety analysis and thereby safety engineering and thus safety engineers. It is not an ETSI-defined TVRA, which has no place for hazard identification. As far as I can see, it is not a SRA as elaborated elsewhere in the literature, for similar reasons – to my knowledge, no other SRA analyses safety properties (and, one might argue, neither should it). What is being suggested in IEC TR 63069 is somehow a safety-for-security-for-safety analysis. As we have seen, this process as expressed is inadequate to the task.

4 Practice and Best Practice

Standards documents are generally supposed to represent "best practice". In this case, that would be best practice in combining cybersecurity and safety concerns in safety-related IACS. In determining what should be best practice, it is often wise to see how practices might pan out on simple examples. I relate one from my personal experience; most, maybe all, sysadmins have such tales.

4.1 An Experience

Once upon a time, I established my computer network for use of my research group RVS, separate from my faculty network at Bielefeld University. The RVS network gateway was connected to the University backbone in a nearby room used by the university computing centre (UCC) for such connections. There is mutual dependence on appropriate behaviour and configuration implied by such an arrangement; what used to be called "rely-guarantee conditions". RVS guaranteed to UCC that RVS network activity would comply with UCC's defined terms of use. The UCC network-administrator told us in response that we could rely on the university backbone only containing legitimate traffic, and that no one unauthorised would/could "tap in" to eavesdrop or perform man-in-the-middle activity. We accepted at that time that they had the University-internal traffic

12 As noted above, such assessment and assurance can theoretically be carried out through the use of such inference systems as BAN-logic.

and behaviour under control.[13]

One evening very late, a user of an RVS subnetwork noticed a system administrator logged on and tried to chat, but got no response. He called the admin by telephone, who it turns out was not logged on. The admin went in to look, and the intruder beat a hasty retreat, destroying about 3GB of logs and associated files. Extensive forensic analysis over the next few weeks gave us a pretty good idea of who it had been and what he had been trying to do [WiTa 02].

The immediate question that evening was: how had he come in?

The sysadmin entered his subnetwork often over a modem from his home. He had used login processes in cleartext. This should have been OK according to our rely-guarantee conditions with the UCC: the traffic to the University was carried over telephone lines using ATM (even then an old protocol) which was not susceptible to eavesdropping, and the reception kit in the university converted this signal into university-backbone IP traffic, which we had understood by our rely-guarantee condition to be free of intrusion. The university backbone was the only place where someone could have eavesdropped on my colleague's credentials. It had apparently been breached. I called up the UCC network-admin on a Saturday morning: “you've been breached”. He responded, “could well be – we can't do everything.” An honest and reasonable reply, but also an admission that our “rely” condition had been violated.

In retrospect, RVS should have analysed the “rely” condition more carefully: we were safe from University-generated malfeasance on the backbone. Outsiders were a different situation.

Suppose IEC TR 63069 had existed then, and say we had thought to use it for guidance as to how to deal with our cybersecurity concerns (there are no safety functions in the RVS network, but there are still RVS-critical functions which we held important not to subvert). What would have been our SE? A university backbone with careful internal control, carrying only legitimate traffic; no snoopers (at that time). According to Guiding Principle 1, we would have been entitled to assume that on the guarantee offered by the UCC, and devise our network behaviour accordingly, which indeed my sysadmin had done. Result: we were breached.

Suppose, rather, that we had followed the ETSI conception. We would have been able to rely upon no University-generated malfeasance, which was the threat explicitly considered in the original negotiation[14]. But then it would have been

13 No longer. Then, it was the mid-1990's: the Internet had only just started. Most people had never heard of buffer-overflow attacks and root kits. Unix security was thought to be pretty well dealt with in [GaSp 96], which it was at the time. It was plausible to think that a University sysadmin could ensure to a high degree of confidence that only legitimate traffic was passing over a network.

14 Of course, once a University asset is compromised by an outsider, then the outsider's activity on that asset *appears to be* University-generated, so this distinction between internal and external is not exclusive. However, if such activity does not comply with the University's terms of use, which in many cases it will not, then alert University sysadmins can be expected to notice and investigate.

up to us to have considered our residual vulnerability. The threat of outsider malfeasance on routine university and other networks was at that time just becoming manifest (it was of course already well-known on "interesting" networks). The ETSI view might well have led us in RVS to internal discussion of our vulnerabilities and likely led us to implement end-to-end encryption for all authentication processes.

Conclusion: ETSI would have given us better guidance than IEC TR 63069.

4.2 What Practice Could Be Best?

If the ETSI conception would have given us better guidance in the case above than IEC TR 63069, why would we assume that IEC TR 63069 is adequate for more complex cases?

Let us consider, informally, what best practice could be. Suppose you are an engineer responsible for certain critical functions of a system. The same concepts of hazard and risk analysis can be applied to such critical functions, but here one replaces the notion of "harm" with that of "loss", which is a more general term – it is up to you to say what a "loss" is, and then your "risk" is your expectation of loss (in the probabilistic sense of "expectation"). This is the original 1711 De Moivre concept of risk [Ber 98].

As an engineer responsible for designing and implementing a system in which risk is to be kept to an acceptable level, it is surely best to rely on what you know your cybersecurity colleagues can reasonably accomplish, and not on an assumption of perfection. In an ideal world, colleagues can say what they can protect against and how well. Given that they so protect, you then have to figure out, presumably in consultation with them, where your residual vulnerabilities may lie. Some of those residual vulnerabilities may constitute possible causes of hazards, through which you might experience loss. Maybe you have recognised and mitigated those hazards already (for they might arise outside of cyberinsecurity causes), maybe not. The ETSI concept of SE leads to discussion and, one hopes, identification of what residual vulnerabilities there may be pursuant to an (ETSI) SE. It is difficult to see how the IEC TR 63069 concept of SE encourages a similar negotiation. The key is a mutual understanding of protections and residual vulnerabilities, and then for safety engineers explicitly to consider such residual vulnerabilities. It is hard to see how one could get from here to a principle anything like Guiding Principle 1.

5 Conclusion

I conclude that the analysis proposed by IEC TR 63069 in subclause 7.3.2, the "threat-risk assessment <security>" is underspecified, and the result unrealistically overconfident. Such a process necessarily involves cybersecurity analysis as well as safety analysis and therefore necessarily involves both security engineers and safety engineers. Lacking precision, it remains a vague and

incomplete guide to doing something which might help safety analysis in a cyber-insecure environment. Furthermore, the IEC TR 63069 concept of "security environment" does not appear as adequate to the task as the ETSI concept.

Acknowledgements

Many thanks to Ross Anderson, Bev Littlewood, Mark Nicholson, Mike Parsons, Harold Thimbleby and Martyn Thomas for their helpful comments on drafts of this paper.

References

[Ber 98] Peter L. Bernstein, Against the Gods: The Remarkable Story of Risk, John Wiley & Sons, 1996/1998.

[BuAbNe 89] Michael Burroughs, Martín Abadi and Roger Needham, A Logic of Authentication, Research Report 39, Digital Equipment Corporation Systems Research Center, 1989, revised 1990. Available from https://www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-39.pdf , accessed 2019-10-22.

[ETSI 1021651] European Telecommunications Standards Institute, ETSI TS 102 165-1 V5.2.3 (2017-10), CYBER; Methods and protocols; Part 1: Method and pro forma for Threat, Vulnerability, Risk Analysis (TVRA), Technical Specification, ETSI 2017. Available from https://www.etsi.org/deliver/etsi_ts/102100_102199/10216501/05.02.03_60/ts_10216501v050203p.pdf , accessed 2019-01-11.

[ETSI 203251] European Telecommunications Standards Institute, ETSI EG 203 251 V1.1.1, Methods for Testing & Specification; Risk-based Security Assessment and Testing Methodologies, Final Draft, ETSI Guide, ETSI 2017. Available from https://www.etsi.org/deliver/etsi_eg/203200_203299/203251/01.01.01_50/eg_203251v010101m.pdf , accessed 2019-01-11.

[GaSp 96] Simson Garfinkel and Gene Spafford, Practical Unix and Internet Security, 2nd Edition, Revised and Expanded, O'Reilly & Associates, 1996.

[IEC 51] International Organization for Standardization/International Electrotechnical Commission, Guide 51, Safety aspects – Guidelines for their inclusion in standards, Edition 3, ISO/IEC 2014.

[IEC 61508] International Electrotechnical Commission, IEC 61508, Industrial-process measurement, control and automation – Functional safety of electrical/electronic/programmable electronic safety-related systems. Seven parts. Edition 2, IEC 2010.

[IEC 61508-1] International Electrotechnical Commission, IEC 61508-1, Industrial-process measurement, control and automation – Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 1: General requirements, Edition 2, IEC 2010.

[IEC 62443-1-1] International Electrotechnical Commission, IEC TR 62443-1-1, Industrial-process measurement, control and automation – Network and system security – Part 1-1: Terminology, concepts and models, IEC 2009.

[IEC 62443-3-2] International Electrotechnical Commission, IEC, Security for industrial

automation and control systems – Part 3-2: Security risk assessment and system design, Committee Draft, 2018. Note this document is, according to IEC procedures "not to be used for reference purposes". A Committee Draft is, however, clear evidence of an IEC project with an intent to produce such a document.

[IEC 63069] International Electrotechnical Commission, IEC TR 63069, Industrial-process measurement, control and automation – Framework for functional safety and security, IEC 2019.

[IEV] International Electrotechnical Commission, IEC 60050, International Electrotechnical Vocabulary. Many parts, various dates. Available in part at www.electropedia.org , accessed 2019-09-07.

[Lad 17] Peter Bernard Ladkin, A Critical-Systems Manifesto: Issues Arising from IEC 61508, 2017. Available from https://rvs-bi.de/publications/RVS-Bk-17-01.html , accessed 2019-01-11.

[Sch 16] Roger R. Schell, Cyber Defense Triad for Where Security Matters, Comm. ACM 59(11):20-23, November 2016.

[Tho 19] Martyn Thomas, comment to the Safety Critical Systems List, 2019-01-09 at 1913 MET. Available at http://www.systemsafetylist.org/4480.htm , accessed 2019-01-11.

[WiTa 02] I Made Wiryana and Avinanta Tarigan, Forensic Analysis on Nakula and Antareja Machine Incidents on 18th January 2002, Research Report RVS-RR-02-02, RVS Group, Bielefeld University. Available at https://rvs-bi.de/publications/Papers/serangnakula.pdf , accessed 2019-09-07.

A Comment on IEC TR 63069

Martyn Thomas

Gresham College

Abstract *IEC TR 63069 is a misleading guide to what is needed for cybersecurity in safety-critical digital systems*

1 Introduction

IEC TR 63069 *"Industrial-process measurement, control and automation – Framework for functional safety and security"* seems to me to be trying to solve the near-impossibility of assuring the safety of critical systems in the face of security threats. It does so by separating the assurance of safety from the assurance of security. That is a sensible way of passing the problem to someone else to solve, but it doesn't in itself solve the overall problem of safety. It does, at least, acknowledge that you cannot consider a system to be safe unless you have assurance that it is secure.

The requirement for a "security environment" (SE) means that the reliability requirements for safety functions become reliability requirements for the security of the SE. As expressed in the excerpts from 63069 in (Ladkin 2020), the requirement for the security of the SE appears to be for total security. That is unhelpful, because no safety engineer will have access to a perfect SE, as Stuxnet demonstrated (Langner 2013). So the (let us call it) "security-integrity level" of the SE must be part of the safety argument for the system whose safety is being assured.

Meeting and assuring this "security-integrity level" will involve rigorous system and software engineering. The central question for the authors of IEC TR 63069 is therefore whether the introduction of the SE has made it easier, quicker or less expensive to assure the safety and security of the total system than it would have been had the safety and security threats been considered together.

It seems likely that the overall task has actually been made more difficult because:

© Martyn Thomas 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

(a) all the same subtasks will need to be carried out (those who wish to doubt this can try to exhibit an example of a key subtask that is no longer required, and will be unable to do so);

(b) some of the subtasks will have to be repeated in analysing safety and in analysing security (such as: describing the system architecture and design)

(c) the design decisions that are taken in the safety engineering (such as choosing specific COTS components and communications protocols) may affect the feasibility of establishing the SE (and vice versa), so that safety engineers and security engineers will have to interact to resolve the conflicts.

It is well accepted that it is not cost effective to add security to an insecure system late in its design, even if it is feasible, because that imposes major design changes that would have been better considered earlier in the development. It is also not cost effective to develop an SE without detailed knowledge of the system that it has to protect, because (for example) you cannot assure the security of the supply chain for critical components unless you know what components are being used.

On this analysis, IEC TR 63069 increases rather than reduces the difficulty and costs of solving the problems that it is designed to address. One can, if necessary, construct a worked example for the simplest safety-critical system in the following way. A component might contain trojan functionality that was designed to cause a safety function implemented by that component to fail. Inside an SE of the IEC TR 63069 variety, the safety engineers would supposedly have no need to search for that trojan functionality or to consider the supply chain security when selecting components, according to Guiding Principle 1. However, the SE cannot be established without knowing what components the safety engineers have decided to include in their design.

References

International Electrotechnical Commission, IEC TR 63069, Industrial-process measurement, control and automation - Framework for functional safety and security, May 2019.

Ladkin, Peter Bernard, IEC TR 63069, Security Environments and Security-Risk Analysis, this volume.

Langner, Ralph, To Kill a Centrifuge, Langner Communications GmbH, November 2013. Available at https://www.langner.com/wp-content/uploads/2017/03/to-kill-a-centrifuge.pdf accessed 2019-09-13.

Application of Engineering Safety & Security Management across HS2: From automatic trains to concrete

Reuben McDonald

High Speed Two (HS2) Ltd

Abstract *HS2 is a huge and complex programme, which involves building new rail and road infrastructure, civil works, stations, railway systems, trains and creating new organisations to operate these. From initial design to completed delivery over a timescale of over 20 years. Within this, a consistent and comprehensive system safety approach has been developed that is flexible enough to cover cyber security threats to on-board systems through to fire safety of concrete. Reuben will present his approach and note key issues and learning points that have evolved.*

© HS2 Ltd, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

Safety Critical Integrity Assurance in Large Datasets

G S Sutherland[1], A Hessami[2]

[1]Ikon Riskconsulting Limited, Cambridge, MA, United States

[2]Vega Systems Limited, London, U.K

Abstract *Historically, data types such as standing, configuration, and other data types have had to be proven correct before application in a safety-critical environment. Usually, this has been achieved by rigorous manual or automated checking and system testing before first use, and is feasible because the data sets are relatively small. However, a "safety by compliance" strategy for data does not adequately deal with sources of errors leading to accidents. As AI is based on the availability of huge quantities of data, such approaches become increasingly useless at scale. Three problems therefore must be overcome. First, by ensuring that large data sets contain sufficiently granular detail to correlate to events associated with identified accident potential or other rare events, and validated using appropriate principles; second, to assess whether related, but diversely sourced data sets could be cross-validated by identifying and quantifying the probability of encountering missing features in the data and, finally, to provide assurance that any capacity of an AI-driven function to incorrectly extrapolate from data within the existing data set is minimised. This paper is concerned with possible approaches to address these problems in greater detail.*

1 Introduction

This paper investigates the assurance of Artificial Neural Networks with respect to control systems that have the potential to influence, either directly or indirectly, the safety of human beings.

The use of such systems is likely to expand rapidly and encompass activities and processes once the exclusive preserve of human users, operators and experts. These systems cannot be assured using traditional means, presenting a problem to regulatory authorities, users, owners, developers and the public in being able to gain sufficient confidence in their use and operation.

Such systems are based on Big Data, but large-scale data curation appears to suffer from diminishing economic returns as the corpus size increases; rare events can be overlooked and the collection effort tends to the repetitive. Such effort therefore needs to be expended carefully to ensure deviations from normal or expected behaviour are fully captured (Lin, et al, 2005) whilst avoiding excessive focus on collection of data inferred from known or expected rules.

A number of new approaches to verification and safety assurance of these functions are investigated. The importance of data diversity needs to be acknowledged as important when building a corpus of data. The well-known and widespread use of the 'beta-factor' approach, traditionally used to quantify the effectiveness of redundancy in hardware-based systems, is re-introduced for the data driven, deep learning application era.

Although there are severe theoretical and practical disadvantages in attempting ANN verification at the neuron level, much greater confidence in output correctness could be derived from structures of competing or co-operating ANNs structures. These are proposed for verification and validation of data, namely; game theoretic and adversarial networks.

Although potentially vast and seemingly unbounded, the data underpinning these functions should in fact be regarded as a limited resource requiring carefully targeted effort to correctly represent relevant experience from which ANNs can reliably extrapolate. Ensuring an appropriate level of granularity in the data is also key. The dataset should therefore be appropriately sized for the function and contain 'surprisal' value. The need for an assurance framework based on principles is derived from the above considerations.

1.1 Problem Space

Machine Learning Systems, more commonly known as Artificial Intelligence, utilise networks of very simple mathematical functions known as 'neurons', provide a significant opportunity to greatly increase the reach and scope of automation to reduce human operator supervision, control or intervention as an objective. This can be primarily for economic reasons, to reduce perceived or actual human error, or both. It is almost certain that a great many fields of human endeavour will be affected, as such machines will soon have the capability to enhance decision-making, increase accuracy and repeatability (thus improving quality), whilst reducing costs and otherwise increasing efficiency and availability. Examples include autonomous (driverless) vehicles on public roads, at sea or in the air, as well as automated medical diagnostics and AI-assisted legal opinion, amongst others. The main resource consumed is data, from which networks of neurons statistically infer relationships and hence 'intelligence'.

Of course, it is very important to ensure that any system having the capacity to make decisions that directly or indirectly affect human safety are both thoroughly tested and assured to ensure that risk is reduced to an acceptable level.

- Many end-users (and the public) are exposed to an unquantifiable and therefore potentially unacceptable risk of accidental fatality whilst the corpus is built;
- Emergent risk may be inherent as the body of users and devices grows and unexpected situations appear more quickly than new data can be amassed, understood, verified and validated;
- Large-scale data curation appears to suffer from diminishing economic returns in the steady state as the corpus approaches a certain size, such that rare events could be easily or inadvertently overlooked and so remain unforeseen. There appear to be no benchmarks to quantify data sufficiency and as a result, no data 'ALARP' principle, for example;
- Rules-based compliance might not necessarily be sufficiently representative of complex socio-technical systems involving complexity;
- The data curation programme could tend to the repetitive and oft experienced, missing out important edge-cases, without proper care;
- Manipulation of large datasets at runtime can present exponentially increased computing overheads at runtime and critically limit performance.

A geometrical arrangement (Russell, et al, 2009) of neurons and the corresponding data as a matrix (or more correctly, a tensor) enables statistical inferences to be drawn. Connections between factors enable insights into the nature of the real world from where the data originated and provides the basis of 'intelligence'. The ANN relates variances in any number of variable dimensions (>>3), representing any naturally occurring or man-made phenomena; time series, natural language, image or other sensor data to any other feature contained within that dataset and conclude something from a given hypothesis. In essence, an ANN is able to make simultaneous comparisons between every feature within such data to determine the degree of association between variables using a richly connected array of neurons within the tensor. A process of training the neurons weights and biases with known use-cases or examples gives that ANN a capacity to do something useful, e.g. recognise classes of objects, provide a functional output, given an input or recognise hazards. The ANNs intelligence appears to come about because of optimisation algorithms which minimise the error between input state and output conclusion, because of this training.

As ANNs are not amenable to standard analytical techniques (Sutherland, Hessami 2019) and the field of assurance of AI-based systems involving real-world safety-critical control implications is an emergent one. Specific problems concern the size of the training data set, complexity and uncertainty, all of which

make human-based, traditional inspection and checking of neuron weights and biases of any practical ANN all but impossible.

For example, autonomous vehicles might use a digitised optical image obtained from a full-colour RGB camera system to classify objects. Such a device might yield an image 256 pixels by 256 pixels (256 × 256 × 3), in total containing over 196,000 variables, equating to a 196,000-dimensional array. Real-world application image sizes can considerably exceed this size, making verification implausible and probably a pointless exercise anyway, because the loss of context about the calculation; only the network 'knows' why a calculation result is the way it is. Further, complex structures of 'networks of networks' are needed to recognise features and provide sophisticated decision-making capability in many practical applications. Figure 1 shows one such application for an Autonomous Vehicles (AVs):

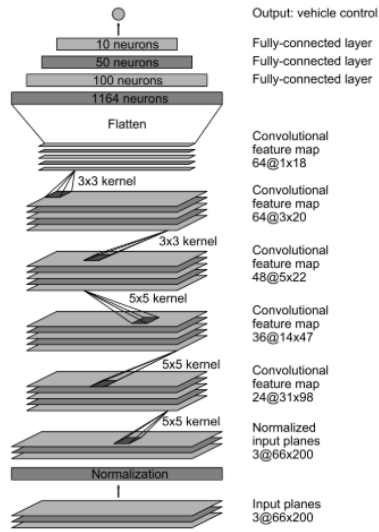


Fig. 1. ANN Neural Network using Convolutional Neural Network Feature Detection for AVs (Simplified) (Pal, 2019)

The end goal of any ANN is to reduce input dimensionality by layering 'richly interconnected' neurons. As noted, these connections are realised practically by tensor-multiplication. A final-layer neuron expresses a singular, conditional numerical probability. The end result is readily understandable to human operators, and to other machines. For example, a network may provide an output, such as '0.892258', with that scalar number representing the probability of a given hypothesis. This number can be thought of as a figure of merit, or a degree of belief in something being true, or not.

In binary classification type problems, (yes/no, stop/proceed, etc.,) anticipated to form the bulk of such safety-related and safety-critical AI based decision-making, that scalar output is interpreted as the probability of the condition being true or false. If the probability is equal to or above 50% then the condition is indeed 'true', whilst, under 50%, it represents 'false'.

Importantly, it should be stressed again that the ANN is not programmed with rules in the traditional sense, but instead determines the likelihood of the existence of rules from knowledge of its surroundings (Russel, et al 2009). In turn, that knowledge is obtained from a representation of the data in the environment. The following question arises: How do we know that such a definitive, singular answer is correct if the network decision process cannot be understood? The following sections are an attempt to providing insights into answers to this question.

2 Data Verification and Validation

It is necessary to ensure that data underpinning an ANN decision process is verified and validated; because without trustworthy data, the process of assurance cannot begin for ANNs.

The current approach indirectly stresses the importance of obtaining sufficient, perhaps very large quantities of data, assumed representative of all possible situations that the system might conceivably experience. Theoretically, as more data is collected, more experience is integrated within the corpus of learning and the probability of an ANN not previously encountering something tends to zero as the dataset size tends to infinity. This is in line with the Big Data ethos; the collection of ever greater quantities of data from which increased confidence can be inferred, because it is known that ANNs perform better as the dataset increases in size.

These issues need to be addressed by a framework of principles and a variety of structures to begin to build confidence for safety-critical applications of ANNs. These are introduced and examined below:

2.1 Structural Dataset Cross Validation

In addition to the preparation and use of data in practice, structural means of controlling hazard risk is necessary. The use of redundant architectures in safety engineering is well-known, with the benefits (and costs) well-documented. However, redundancy as a technique to achieve functional integrity in ANN-based applications is not yet widespread and as ANNs are, for all intents and purposes, in themselves unverifiable, the investigation of architectures in the field of safety

assurance is worthwhile. Two structures are considered: Adversarial Neural Networks and Game Theoretic Structures.

Adversarial Neural Networks

The concept of adversarial neural networks has many practical applications at design time for V&V or at runtime. Two networks play opposing roles to improve the accuracy in a given outcome or function from each network. Classical discriminative networks classify outcomes into categories, for example, 'true' or 'not true', based on features in the data set. Such discriminative networks work purely on forward mapping of labels to features[1]. Generative Neural Networks (GNNs) (Goodfellow, et al, 2014) on the other hand, work in the opposite sense; they attempt to predict the data automatically from the labels. One network, known as the *generator*, produces new instances of data, perhaps random, or pseudo-random data, which are evaluated for authenticity by a *discriminator*, which attempts to classify data as belonging to a training set, or not.

The generator is incentivised (to find a mathematically optimal solution) to produce new, synthetic data which is designed to fool the discriminator into accepting it as authentic. In effect, this is to encourage it to 'lie without being caught'. Both networks continually learn (not simultaneously) from the process of inauthentic data production and evaluation, so both get continually better at these tasks with each training epoch.

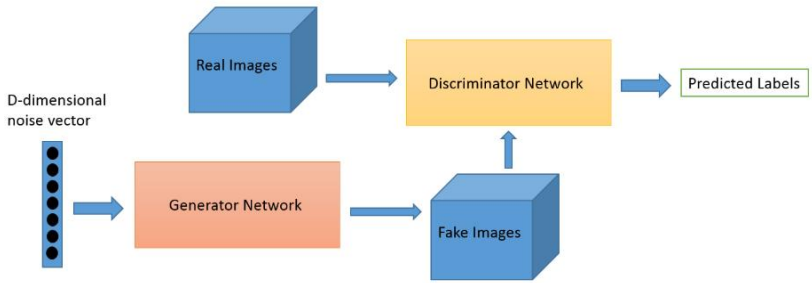


Fig. 2. Generative Neural Network Example (After Goodfellow, et al)

This approach is likely to have significant application in the field of data verification and validation for safety functions. One way of testing this would be to seed a generator with data known to be representative of a given scenario. Then, deliberate but known errors introduced to the data. As adversarial networks work by optimising their outputs relative to the role they play (i.e. either as 'validator'

[1] Data features are manually mapped to labels (e.g. 'Car_Present, Car_Not_Present) at training time.

or 'forger') the usefulness of the error catching function is evaluated. After several epochs of error detection, the data errors should be revealed.

Game theoretic Structures

Architectures for the implementation of game-playing neural networks have previously been proposed (Sutherland, Hessami 2019), including the use of two (or groups of two) independent networks, each playing a role in a MINIMAX (e.g. Maschler, et al) game in order to enhance the utility of a safety function.

These can be visualised as mutual checks for plausibility through the decision process; each network checks an outcome is performed correctly and is achieved at each 'turn'. The intermediate results are stored for online, diagnostic assessment or for verification purposes.

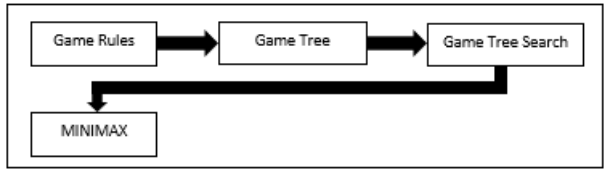


Fig. 3. MINIMAX Algorithm

Each network player takes a 'turn' according to the rules of the game as defined. One ANN player (MAX) tries to maximise the probability of its outcome, the other (MIN) attempts to minimise the maximum error inherent in the output, in a way similar to a check or balance. Its formal definition (Osborne, et al 1994) is given by:

$$\bar{v}_i = \min \max v_i(a_i, -a_{-i})$$

Where:

i is the index of the first player;
-*i* denotes all other players except player i;
a_i is the action taken by player i;
a_{-i} denotes all actions taken by the other players;
v_i is the value function of player i. and $\bar{v}_i$ is the MINIMAX of the payoff.

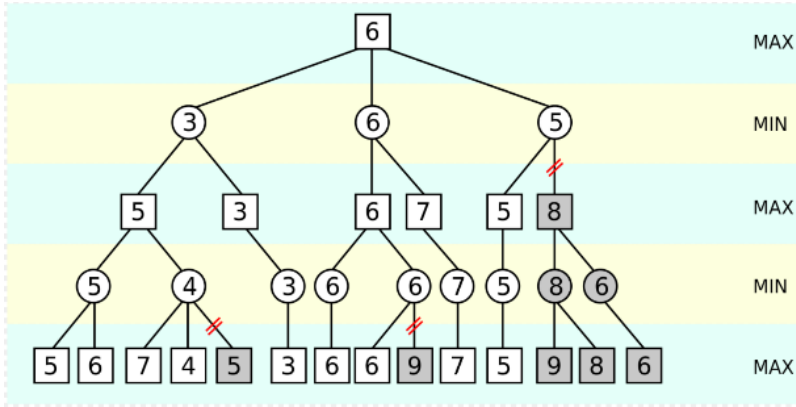


Fig. 4. Game Tree Search with Alpha-Beta Pruning[2] (MIT, 2003).

In the example above for traffic light aspect identification, one means of beginning to tell whether it is a red light signal (from any other type of sign) is by discrimination by shape. An algorithm expands these rules into a game tree, representing all possible combinations (e.g., whether the signal is round, or not, is red 'enough' or not, etc.) evaluated at every 'turn'. The resultant tree could be very large, with many possible valid combinations of parameter. The MINIMAX solution contains the lowest possible maximum risk, associated with a valid combination ('move'). Searching the tree for this combination is output. Shortcuts (alpha-beta pruning) to save evaluating all branch nodes of the tree may be necessary (Ortiz, 2006) to maintain performance of the search algorithm.

The tree rules are executed in high-integrity *software* (as opposed to evaluation in a network). Each outcome at every decision node in the game tree can be subject to a verification check, either in real-time or at design time. This provides the verification required, if necessary.

The Nash Equilibrium (Maschler, et al, 2013) exists if no player gains an advantage in the event of any of player changing their strategy. This equilibrium condition also applies to network players. If a Nash Equilibrium is encountered during a computation within the game tree, then the algorithm may stop. Diversity in the dataset (beta is a non-zero value) should result in no zero-sum games. The approach would therefore seem only work when the networks are trained with diverse data; using the same data would result in 'blind spots'.

[2] *Jez9999 CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/), via Wikimedia Commons*

https://commons.wikimedia.org/wiki/File:AB_pruning.svg

2.2 Controlling Information Leakage in ANNs

The information content in a network, expressed as entropy, is a measure of the unpredictability of the state, or equivalently, of its average information content. This is given by the Shannon Entropy (Shannon, 1948):

$$H(X) = H(p_1, ..., p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Where *H(X)* is the Entropy of a random variable, *X* and p_i is the probability of X_i. Entropy is a maximum when a value is most unexpected and carries news value (or 'surprisal') about the state of an event. It is a measure of how juxtaposed a data sample is, in general, uniform datasets contain lower entropy than highly randomized ones. In the case any given high-integrity safety function in a low-demand application, no ***new*** information would be generated for nearly all of its life, as 'nothing much' is expected to happen for the time that the function is expected to operate. When the safety function intervenes, a great deal of information needs to be imparted very quickly, and is a reason why performance of hardware is important. A highly non-linear response is therefore necessary.

Entropy as a measure of the expected surprisal, or 'news value' quantum of information in that process therefore needs to be very high.

Although the fixed data content in a given network for a real-world safety function is likely to be very large indeed (many tens or hundreds of megabytes), the change in that output remains close to zero, as a function of time. The challenge therefore is for the function to safely contain all relevant information and prevent leakage when there is no real demand. In summary, we are seeking two properties for an effective safety function; first, that it remains non-uniform in terms of data predictability, and second, that it reliably contains information unless and until there is a real and credible demand.

Large scale information leakage, e.g. inappropriate responses to given stimulus or low-level leakage over time, inevitably weakens the response of the system to the point of rendering its response useless. Either the network would have to be continually strengthened with new data, that duplicate information is somehow detected within the network, or perhaps that the function is itself randomised by reference to other elements within the same network to prevent wasteful leakage.

Structural means to ensure integrity or adversarial seeding of failure detection are two possible ways of achieving this. Mutual information may provide another way forward. This states that whenever information is known about one variable, uncertainty in the other is reduced. So, if two redundant datasets concerning the same subject are available, the information in one should improve certainty in the system of both. This has a basis in the degree of randomness in the dataset. If

such datasets are well-matched and completely mitigate the absence of information in the other, this might represent a common-mode 'beta' factor of 0 (they are completely independent) and when the opposite is true and they do not, this is represented by a beta factor of 1 (completely dependent, with no new information content at all), with intermediate, fractional values possible (See Jones, 2016).

2.3 Minimising Incorrect Extrapolation

Incorrect extrapolation refers to those cases when there is insufficient data at a given juncture, rendering the output of the neural network unreliable. Using the example of autonomous vehicles (AVs), which rely on AI perception algorithms that are trained to make safety-critical driving decisions, it is noted that an AV may fail to detect a hazardous scenario, known as an "edge case" (Zimmerman, 2012), because the training data does not recognise the specific unusual circumstance encountered in the real world. In this and in similar cases, the extent of the data represents the reliable limit of experience from which extrapolation remains valid. Any realistic dataset is finite in nature and practical size will be defined by having a reasonably foreseeable set of cases which the AV will encounter in use. Edge cases can be viewed as that set of prior experience which can be trained into the Sense, Understand, Decide and Act model of control. Simulation of such scenarios is a viable way forward; commercial providers have developed systems to design and train ANNs by complementing traditional computer simulation with road testing of perception systems.

Perception software is tested against existing sensor data and the adversarial approach is of practical value. The difficulty appears to reside in the ability to efficiently locate representative edge cases, though once identified, they can be quickly assimilated into training scenarios, with such cases appearing to exhibit two characteristics; first, of being rare and second, being closely aligned with unexpected behaviour, including that associated with human error.

One way to identify further edge cases could be via a combination of data analysis and simulation, to assist in the understanding of sequences involving a human actor's role in accident avoidance. The most useful analysis would involve 'near-miss' sequences, i.e. those accidents with lower severity (e.g. no fatalities), where avoidance action taken by human drivers has a positive effect. Use of a naming convention or standardised typology would also be of benefit. Figure 5 shows classes of nomenclature and standards used in automated driving.

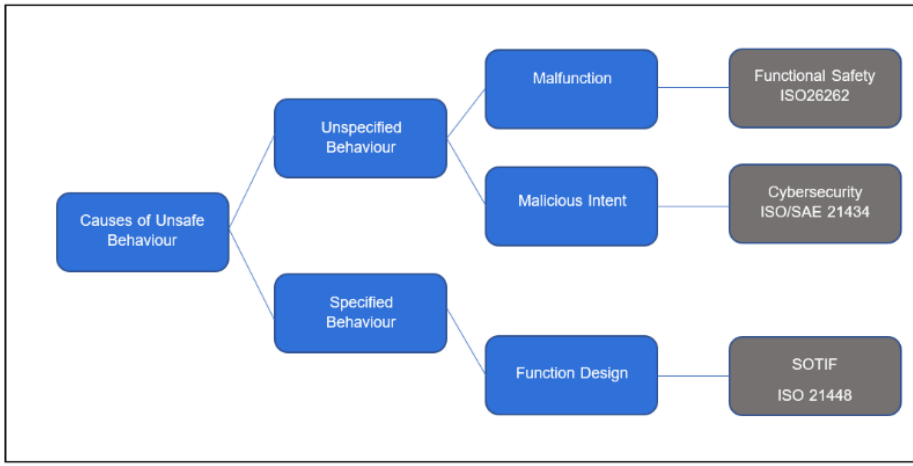


Fig. 5. Typology of Causes of Unsafe Behaviour (Birch 2019)

2.4 Data Safety Assurance Principles

We may now begin to outline solutions to some of above issues. The following principles relate to data safety assurance.

Principal 1: Ensure Sufficiently Diverse Data

The question of sufficiency of dataset size in absolute terms is almost impossible to define, as the generality of range, size and complexity of applications is equally enormous and will give rise to very large training sets. However, the effectiveness of a large dataset alone is no guarantee of functional safety for a given ANN. It may instead be possible to define what 'sufficiently diverse' may mean for a given application. Diversity will arise from a number of different sources, including sensor type, sensor location, geographical location, historical vs. current, etc.

Principal 2: Scenario Test All Data

The response of a network trained with data curated from the environment or elsewhere needs to be carefully determined before any release. Accelerated scenario testing is proposed for this purpose. Instead of the test data being tested on the target hardware, higher performance (perhaps static) testing is envisaged. Deliberate errors are injected into various areas of the overall system to determine the response.

Principal 3: Explicitly Maximise Data Entropy in a Dataset

Means to increase the surprisal value inherent within the dataset should be examined at every stage in the data curation process. This may be by analytical means, perhaps by random, automated sampling of data to determine the degree of similarity between samples, or by means of deliberate insertion of known edge-cases. It may be possible to express a data sample as a spectrum (how many units in each interval, for example).

In general, the noisier, less uniform, less predictable, less unlike another cooperatively used and shorter, the better.

Principal 4: Dataset Quality Assurance

Although a network may be trained on data obtained from the environment and later tested on a sub-set of that data, the issue of quality assurance in the curation process must not be overlooked. As insight into the role and use of data is extremely hard to comprehend once in the domain of an ANN, taking into account the circumstances in which the data was obtained enables the ability for human insight and oversight to be retained.

Principal 5: Ensure a Wide Range of Data Curation Models

Additional to Principal 1 above, the need for a wide range of curation methods should be considered.

Principal 6: Optimally Size a Given Dataset

Given the trade-off between system performance (being able to execute a function within a certain timeframe), the complexity of the function and the size of the dataset, it appears important to ensure that an optimal size of dataset is sized appropriately. Ways in which the dataset could be compressed or otherwise reduced in size should be considered.

Principal 7: Benchmark a Performance Level for an ANN-based Safety Function.

Similar to the concept of SIL and ASIL used in the industrial and automotive domains, the use of a benchmark for ANN based systems is also desirable. Although there is no known method for accurately measuring or predicting the performance of ANNs currently in terms of functional safety, the objective would be to ensure that the overall system objectives are set out for later analysis, when this becomes possible. For example, for a AV system, an overall safety target could be set at 1% of current human driver fatalities.

2.5 Example: Generation of an Assured Dataset for an AV System

This section is concerned with imagining the process to derive an assurable dataset for AV applications. Following the principles above, the first step is to consider a diverse data as a key component and is obtained from completely independent sources.

AVs learn from a number of sources. These include any combination of, rules-based learning, driving experience, sensor-fusion, maps, manually labelled features and edge-cases. These data sources are not of themselves necessarily independent because they represent the body of knowledge in the present and immediate past. Further, some data could be traced to a common-point, but most importantly they may miss significant accident experience. Much of the required knowledge is distributed, tacit and dynamic in nature; and likely to be dependent on weather, time of day, traffic, location, seasonality and many other factors. The *unknowability* of all combinations of contingent rules and possibilities results in a highly ambiguous and dynamic problem space. Preventing deviations from rules, for example, is not a sufficient condition for an understanding of road accidents and safety, because many accidents still occur when following rules.

A different perspective is required to capture this aspect, but is already available. There were over 37,000 road transport related fatalities in the US in 2014, with all accidents investigated (NHTSB, 2019). This corpus contains information about the nature of accidents and can be analysed by searching for features in the data that represent what should not occur when driving. If a 'suite' of appropriate accident models is developed, for example a set of event trees, then by searching the data to find the best fit between a given event tree and the accident. The process could be done manually, but the ubiquitous ANNs ability to deal with high dimensionality of factors and the number of cases to be considered, would be desirable. The product of this exercise would also be a collection of many, perhaps several hundred or thousands of event trees, each also representative of a type of vehicle accident. Some will contain information on how a human driver reacted to avoid more serious consequences, with this information used to train the on-board neural networks how *they* should behave to avoid similar circumstances.

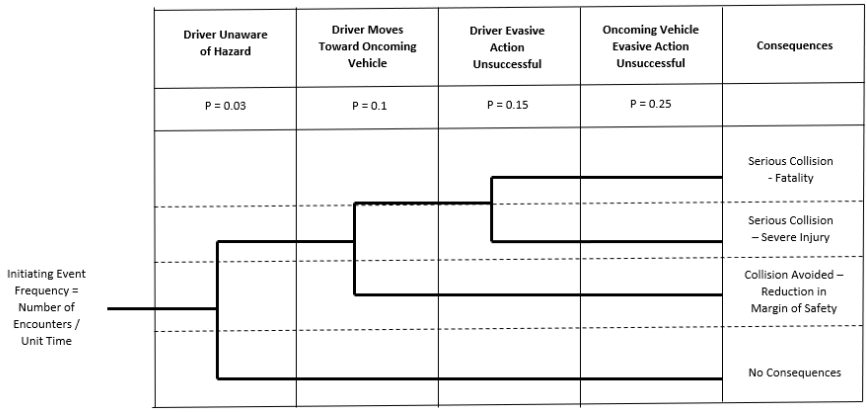


Fig. 6. Accident Analysis using Event Trees

Undertaking scenario tests is then necessary. Deliberately injected, known faults are introduced and by use of adversarial networks, determine whether errors are identified and to what extent they are corrected. Issues not nominally associated with deliberately injected errors may represent real configuration or validation errors and should be investigated. Following this, measures of data entropy are made and the system challenged with simulated intervention demands (‘apply brake’) etc. Changes to the data configuration may increase measures of entropy and are investigated. Structuring data for analysis and execution in a game tree could be used as an intermediate or final verification check on the correct ‘reasoning’ that the data supports.

For automated systems, it will also necessary to consider systematic and random failures. Systems can and will suffer from deviations from necessary behaviour for a variety of reasons, including; failure, mis-specification, configuration errors, performance issues, etc. It is likely that emergent effects, perhaps due to simultaneous ‘over the air’ software or standing data updates to large numbers of vehicles, unintended human-machine interaction or integration issues are possible and would constitute a non-negligible source of malfunction. These also need to be included in the modelling as described.

3 Summary and Conclusions

The assurance of ANN-based systems at the neuron level is acknowledged to be extremely challenging, with safety by compliance strategies ineffective when assuring such systems. Complex structures of networks are required in many real-world, practical applications, adding to the difficulty and cost of assurance.

Overall, properties for ANN-based safety functions have been identified, some of which seem counter-intuitive. In conclusion:

- Data should contain sufficient granularity in data such that it does not perform incorrectly when demanded;
- Data correctly reflects rare events to avoid incorrect extrapolation;
- When demanded, the response of a safety function should be configured to be highly non-linear;
- Entropy is a measure of the news value contained within data and can be quantified. For a safety function, entropy needs to remain high. Datasets are appropriately sized and optimised for entropy as a parameter;
- Does not exhibit data leakage, or contain uniform data for the purpose of maintaining 'surprisal' value;
- Data diversity is paramount as a means of verification. The concept of beta-value is re-introduced to express the level of independence between datasets;
- Cross-validation by independent ANNs is a means of identifying and quantifying the probability of encountering missing features within data. Structural means to achieve verification and validation (V&V) can be undertaken using game theoretic or adversarial neural network structures;
- Data curation should include processes for addressing quality and facilitate updating (by embracing change) and proving data correct;
- Testing by modelling and simulation is required. Results should be fed back into the curation process;
- Diversity and independence between data sets is critical. Traditional 'beta factors' as used in hardware analysis are reintroduced for data to signify the degree of independence between datasets.

References

Birch, et al.	What's the Case for Safety in Automotive? Safety Critical Systems Club, April 2019: https://scsc.uk/file/583/02---Birch---SCSC-Presentation-April-2019.pdf By kind permission.
Goodfellow, et al.	Generative Neural Networks. Cornell University, 2014 arXiv:1406.2661 [stat.ML]
Jones, H.W.	Common-cause Failures and Ultra Reliability. American Institute of Aeronautics and Astronautics. (2016) https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20160005837.pdf
Lin, Zhang	Artificial neural network modelling of driver handling behaviour in a driver-vehicle-environment system. International Journal of Vehicle Design 37(1), January 2005. DOI: 10.1504/IJVD.2005.006087 http://bit.ly/2ldjcz4
Maschler, M. Solan, E. & Zamir, S.	Game Theory. pp. 176–180. ISBN 9781107005488. Cambridge University Press 2013.
Massachusetts Institute of Technology (MIT)	Notes on the Complexity of Search: Department of Electrical Engineering and Computer Science, Cambridge, MA, September 2003. http://www.ai.mit.edu/courses/6.034b/searchcomplex.pdf Retrieved 11th September 2018.
National Highway Transportation Safety Board (NHTSB)	Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey. *Traffic Safety Stats*, February 2015. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812115 Retrieved 25th August 2019.
Ortiz, L.E.	For example: Practice Problem on Search in Games. MIT, Department of Electrical Engineering and Computer Science, (2006). Accessed 20th August 2018.
Osborne, M. J. Rubinstein, A.	A Course in Game Theory: MIT. Cambridge, MA. pp. 14. ISBN 9780262150415 (12 July 1994).
Pal	Deep Learning for Self-Driving Cars, https://towardsdatascience.com/deep-learning-for-self-driving-cars-7f198ef4cfa2
S. Russell, P. Norvig.	Artificial Intelligence: A Modern Approach, 3rd Ed. Pearson 2009.
Shannon, C.E.	*Shannon, Claude E. (July–October 1948). A Mathematical Theory of Communication. Bell System Technical Journal* (PDF). ***27*** *(3): 379–423.* doi:10.1002/j.1538-7305.1948.tb01338.x.
Sutherland G.S. Hessami A.	Potential Methods to Enhance Safety within Neural Network Based Systems, Proceedings of the 25th Safety Critical Systems Club, Bristol, 2019
UK Department for Transport (DfT)	The Highway Code, 2019, https://www.gov.uk/guidance/the-highway-code
Zimmerman, J	Principles of Imperative Computation, 2012. Retrieved 25th September 2019. https://www.cs.cmu.edu/~rjsimmon/15122-s13/rec/07.pdf

Human Factors of Using Artificial Intelligence in Healthcare: Challenges That Stretch Across Industries

Mark Sujan[1], Dominic Furniss[1], Richard Hawkins[2], Ibrahim Habli[2]

1 Human Reliability Associates, UK

2 University of York, UK

Abstract *The use of artificial intelligence (AI) in healthcare is one of the fastest growing industries worldwide. AI is already used to deliver services as diverse as symptom checking, skin cancer screening, and recognition of sepsis. But is it safe to use AI in patient care? However, the evidence base is narrow and limited, frequently restricted to small studies considering the performance of AI applications at isolated tasks. In this paper we argue that greater consideration should be given to how the AI will be integrated into clinical processes and health services, because it is at this level that human factors challenges are likely to arise. We use the example of autonomous infusion pumps in intensive care to analyse the human factors challenges of using AI for patient care. We outline potential strategies to address these challenges, and we discuss how such strategies and approaches could be applied more broadly to AI technologies used in other domains.*

1 Introduction

Expectations for the use of artificial intelligence (AI) in healthcare are high. In the UK, as well as world-wide, politicians and policy makers are quick to highlight the potential health and economic benefits that the widespread adoption of AI can bring. This is underpinned by the establishment of new dedicated bodies, such as NHSX[1] in the UK and significant government funding to facilitate and speed up the development and adoption of AI in health services. AI is a major disrupter to health systems, and it will transform the way healthcare is delivered and accessed by patients (Coiera, 2018).

Examples of the use of AI in healthcare include machine learning algorithms that rely on pattern recognition, classification and prediction. For example, deep learning is particularly well suited to the interpretation of radiological images

[1] https://www.nhsx.nhs.uk/

© Mark Sujan, Dominic Furniss, Richard Hawkins and Ibrahim Habli 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

because of the complexity and richness of the data (Saria et al., 2018). Deep neural networks (DNN) have been used to interpret head CT scans (Chilamkurthy et al., 2018), to identify skin cancer (Haenssle et al., 2018) and to recognise diabetes (Avram et al., 2019). AI-driven chatbots are another popular application domain, e.g. patient-facing symptom checkers (Semigran et al., 2015) or artificial agents delivering cognitive behavioural therapy to mental health patients (Fitzpatrick et al., 2017).

Evaluation studies of such AI algorithms have produced encouraging results. The evaluation of a bedside computer vision algorithm to identify and monitor behaviours of clinicians, such as hand washing, suggests that the algorithm can achieve 95% accuracy (Yeung et al., 2018). Skin cancer detection using algorithms might outperform dermatologists at this task (Esteva et al., 2017). Similarly, the developers of a DNN to detect diabetic retinopathy[2] found their algorithm achieved over 95% accuracy on two test sets (Gulshan et al., 2016). For the management of sepsis, the evaluation carried out by the developers of an algorithm trained by reinforcement learning found that on average patient mortality was lower when clinicians' management decisions matched those suggested by the AI (Komorowski et al., 2018).

However, looking across these studies, the focus of the evaluation is usually on the performance of the AI on a narrowly defined task. The evaluation is typically undertaken by the developers, and independent evaluation remains the exception. For example, the above evaluation of AI sepsis management has been criticised because the algorithm seemingly "learned" not to treat very ill patients – a strategy that fits with the training reward function, but is hardly suitable in a real clinical environment (Jeter et al., 2019). Sample sizes are often small, and prospective trials are infrequent. As a result, the evidence base to date about the actual performance of AI in real-world settings remains weak (Yu and Kohane, 2019).

There is relatively little evidence about the safety of using AI for patient care, and we argue that this is, in part, due to the focus on performance of the algorithms. The real challenges for the adoption of AI will arise when algorithms are integrated into clinical systems to deliver a service in collaboration with clinicians as well as other technology (Sujan et al., 2019d). It is at this clinical system level, where teams consisting of healthcare professionals and AI systems cooperate and collaborate to provide a service, that human factors challenges will come to the fore (Sujan et al., 2019b).

In this paper we analyse the human factors challenges of using AI for patient care as part of a clinical system, and we identify potential strategies for addressing these. The next section describes the scenario of autonomous infusion pumps in intensive care, which we use to illustrate the concepts. In section 3 we analyse

[2] Diabetic retinopathy is a condition of the eye that can affect people with diabetes. It is a leading cause of sight loss and blindness in the UK.

the scenario for human factors challenges and develop example strategies for dealing with them. In Section 4 we discuss how the identified strategies could be applied more broadly to AI systems used in other domains. We conclude the paper with a summary and outlook.

2 Scenario: Autonomous Infusion Pumps in Intensive Care

As a reference case we use a scenario developed within the Safety Assurance of Autonomous Intravenous Medication Management Systems (SAM) project (Sujan et al., 2019a). The SAM project[3] is funded under the Assuring Autonomy International Programme (AAIP)[4], and it is a collaboration between Human Reliability Associates (a human factors and safety consultancy), NHS Digital (an arms-length body of the Department of Health), and clinicians based at Royal Derby Hospital. The project explores safety assurance strategies for novel, highly-automated or autonomous infusion pumps within the intensive care setting. Figure 1 provides an illustration of the intensive care setting.

Fig. 1. Simulated patient in intensive care. The patient is on a ventilator. The stack of infusion pumps is on the left, next to the screen that charts the patient's data.

The motivation for considering the use of AI for intravenous medication management is twofold: to reduce medication errors, and to improve efficiency and

[3] http://www.humanreliability.com/casestudies/sam_project/

[4] https://www.york.ac.uk/assuring-autonomy/

effectiveness. Medication errors are a significant problem for the National Health Service (NHS), and health systems world-wide. A 2018 report estimates that as many as 237 million medication errors occur in England every year, and that these cause over 700 deaths (Elliott et al., 2018). Intravenous medication preparation and administration are particularly vulnerable activities, and therefore such infusion errors represent a considerable burden to patients and the health system (McLeod et al., 2013, Furniss et al., 2019).

In order to reason about the capabilities of automated and autonomous infusion pumps we took inspiration from the automotive domain, where a 6-level taxonomy of driver-vehicle control was developed by the Society of Automotive Engineers (SAE), which ranges from no automation (level 0) through to full automation (level 5). We used this approach and developed analogous levels of automation for infusion pumps, as shown in figure 2. Level 2 represents current smart pump capabilities, where the pump is able to undertake a number of automated checks, e.g. drug and patient identification. At level 5, which represents the scenario of future AI technology considered in this paper, an autonomous infusion pump is able to take clinical guidelines (e.g. for insulin administration) as a starting point, but has the ability to learn and modify these based on continuous monitoring of the patient's physiological response to the drug. We consider the reference scenario described in Table 1.

Table 1. Reference Scenario

Reference Scenario: L5 Infusion Pump
The patient is a 68-year old type II diabetic with sepsis secondary to pneumonia. The patient's blood sugars require insulin control via IV actrapid insulin infusion. Patient identity, nurse identity, prescription and syringe formulation checks are all done by barcode. If checks match, the pump automatically programmes itself to start the infusion, displays medication identity and selects hard and soft programme infusion rate limits without further or final human confirmation. The pump controls the IV infusion rate of insulin in response to continuously measured blood sugar from a central venous sampling device. Within the programmed limits it is able to "learn" the patient's actual insulin requirements and formulate an individualised protocol for the infusion rate based on the sugar readings to optimise sugars control through pre-emptive changes in infusion rates.

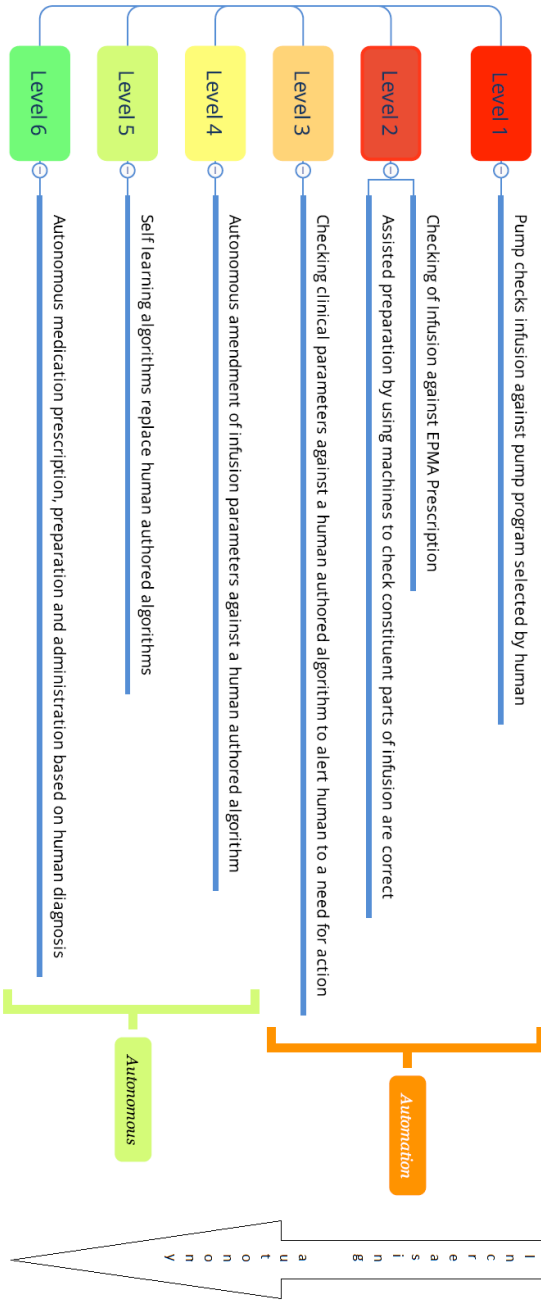


Fig. 2. Levels of Automation - Infusion Pumps

3 Human Factors Analysis

We undertook a human factors analysis of the reference scenario in order to identify human factors challenges that might impact on safe and effective care. The focus of the analysis was the clinical system, which includes consideration of how clinicians interact with the AI infusion pump, other tools and systems that might communicate with the infusion pump and clinicians, the impact on teamwork and the organisation of work, and the impact on communication with patients and patient experience. This socio-technical unit of analysis is shown in figure 3.

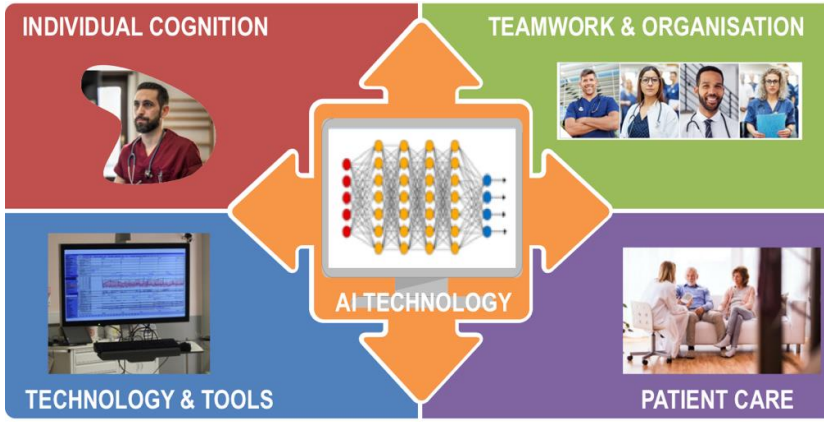


Fig. 3. Socio-technical system unit of analysis

As part of the analysis we undertook a task analysis of the current process as baseline. This was mapped using the Hierarchical Task Analysis (HTA) approach (Stanton, 2006). We undertook a human failure analysis using the Systematic Human Error Reduction and Prediction Approach (SHERPA) methodology (Embrey, 1986). We then mapped the future state that incorporates the autonomous infusion pump. The process map is shown in figure 5 in the appendix. The analysis involved a clinical team consisting of a consultant anaesthetist, an intensive care nurse and a pharmacist. We also interviewed 10 further clinicians about their views on the potential impact of using autonomous infusion pumps in intensive care.

The human factors analysis identified a number of human factors challenges that need to be considered and addressed in order to provide assurance that the AI can be integrated safely into a clinical system, and that the overall service is safe. An overview of the human factors challenges is given in table 2. The table contextualises the identified human factors challenges within the autonomous infusion pump example.

Table 2. Human factors challenges

HF Challenge	Description	Example
Handover	The autonomous system needs to be able to recognise its own performance boundaries, project into the future clinical scenarios that will be beyond its performance boundaries, and identify suitable ways to hand over control to the clinician. Handover includes consideration of: (a) when to hand over; (b) whom to hand over to; (c) what to hand over; and (d) how to hand over.	The patient's blood sugar levels do not respond sufficiently to the insulin given by the autonomous infusion pump. The pump predicts and recognises that it will not be able to control the patient's blood sugar. The pump triggers an alert on the electronic health record, raises an audible alarm, and requests the nurse to take over. The nurse can review the reason for the alert, the history of the pump's insulin management, and its projection into the future, and act accordingly.
Performance Variability	Clinicians need to manage competing organisational priorities and operational demands. They use their experience and judgement to make trade-offs based on the requirements of a specific situation. The autonomous system needs to support rather than constrain this performance variability and adaptive capacity.	The nurse realises that insulin has not yet been prescribed for the patient even though they will likely need it. The nurse goes and finds the doctor, explains the situation, and the doctor issues a verbal medication order and will follow this up with the written prescription later (performance variability). The autonomous system requires an electronic medication order, but allows for a manual override. The autonomous system sends reminders to the doctor with a request for completing the electronic medication order.
Automation bias	When a system works well most of the time, clinicians start to rely on it. In some situations, this can lead to overreliance, for example when the system takes an inappropriate action but the clinician does not recognise this because they trust the system.	Due to sepsis the patient requires tighter control of blood sugar levels than usual. The autonomous system has managed successfully septic patients before but, in this instance, fails to recognise the need for tighter glycaemic control. The autonomous system provides clinician interpretable justification and explanation of its decisions, and the clinician, who has received training on potentially inappropriate behaviours of the autonomous system, is able to spot the discrepancy and act accordingly.

Supervision	Clinicians are both users and supervisors of the autonomous system. They need to understand not only how to operate the autonomous system (e.g. loading a syringe), but also how to recognise potential failure modes or deviations from appropriate behaviour or changes in the environment that might move the autonomous system outside of its design envelope.	The autonomous infusion pump is operating on the sliding scale algorithm for administering insulin. It classifies the patient's response to the current insulin infusion as requiring transition to another scale with 70%, as opposed to 30% for staying within the current scale. The autonomous system initiates and the transition, and activates an "uncertainty marker" to alert the clinician.

3.1 Handover

Handover between clinicians has long been recognised as a safety-critical and particularly vulnerable activity (Sujan et al., 2015b). Handover is not simply the transfer of information from a sender to a more or less passive receiver, but involves collaboration, negotiation and coordination (Sujan et al., 2015c). The introduction of AI and autonomous systems complicates handover even further, as the AI needs to identify appropriate trigger points for handover, it needs to determine the appropriate person or persons to hand over to and understand their information needs, and it needs to use adequate communication channels for the handover. Such trigger points could be the self-detection of an internal fault or the recognition of situations outside of the system's design envelope. For example, should the autonomous infusion pump wait to raise an alert until it fails to control the patient's blood sugar levels or should it communicate its potential failure much sooner to allow clinicians to prepare for taking back control? Should it just raise an alert or should it communicate a history of its actions and a prognosis of the patient's physiological development? Should it sound an audible alarm so that the nurse can pick this up, or should it send a text message to doctors not close to the bedside?

All of these considerations are human factors concerns, and a look at the wider human factors body of knowledge can provide insight into potential approaches for designing adequate handover strategies. For example, the autonomous infusion pump would ideally initiate a form of graceful handover, where the trigger points are determined considering human performance characteristics as well as the specific clinical scenario. The design of the infusion pump should consider the information needs of different types of stakeholders that allows them to build an adequate situation awareness (Endsley, 1995). Alarm prioritisation and alarm

management are further strategies that have been developed in control room operations to prevent operator overload (e.g. EEMUA 191 Alarm Systems[5]), and these should also inform the way the handover between the autonomous infusion pump and clinicians is designed.

3.2 Performance Variability

Within the resilience engineering (RE) community performance variability is regarded as an asset that enables complex systems to deal with disturbances, conflicting goals, and unforeseen situations (Hollnagel et al., 2006). People continuously adapt their behaviour and make trade-offs, often based on some form of subjective risk assessment, and in this way they are able to cope with competing demands, uncertainty, and everyday disturbances such as staff shortages and peaks in demand (Sujan et al., 2015a). This work-as-done (WAD) is necessarily different from work-as-imagined (WAI) by people who design and manage systems (Hollnagel, 2015).

Our human factors analysis provided several examples of such everyday local adaptations. For example, nurses would sometimes administer drugs without a prescription and then chase the doctor to issue a prescription later. They do this depending on the perceived risk category of the drug, the urgency of administering the drug, the availability of the doctor, and their own workflows. This violates the protocol, which requires that a prescription is issued prior to administration of any drug, but it enables smoother functioning of the intensive care unit as a whole, and it can adapt better to patient needs.

Safety assurance of new technology focuses frequently only on the failure modes of the technology and the associated risks. However, from a Safety-II perspective it is equally as important to consider the impact on the resilience abilities of the (clinical) system, i.e. the impact on the ability to anticipate, to adapt, to monitor, and to learn (Hollnagel, 2014).

In the autonomous infusion pump scenario, it is easy to envisage how the static implementation of procedures and protocols might disrupt existing workflows and, in this way, create the need for other workarounds. The design of the infusion pump should consider WAD, e.g. with data collected through observations, interviews and task analysis. There is also a need to look beyond the immediate impact on human – machine (i.e. clinician – infusion pump) interaction, towards the potential impact the introduction of technology has on human – human relationships. Building and maintaining such relationships is an important aspect of

[5] Standard available for a fee from the Engineering Equipment and Materials User Association (EEMUA): https://www.eemua.org/Products/Publications/Print/EEMUA-Publication-191.aspx.

resilient health care (Sujan et al., 2019c). The introduction of technology should not prevent opportunities for building relationships and trust among clinicians.

3.3 Automation Bias

Automation bias describes the phenomenon that people tend to trust and then start to rely on automation uncritically (Parasuraman and Riley, 1997). An interesting recent study in the automotive domain found that even with training and specific instruction on the limitations of an autonomous vehicle, study participant drivers came to rely on the autonomous car within a week, and were spending most of their time on their smartphones or reading (Burnett et al., 2019). Examples of automation bias have also been found in healthcare, for example in mammography reading, where the introduction of a computer algorithm can decrease the performance of radiologists for certain difficult cases, where the algorithm provided incorrect classification (Alberdi et al., 2004, Lyell and Coiera, 2016).

Many, if not most, AI systems will be advertised as having ultra-high reliability, and it is to be expected that in due course clinicians will come to rely on these systems. However, the studies on automation bias suggest that the reliability figures by themselves do not allow prediction of what will happen in clinical use, when the clinician is confronted with a potentially inaccurate system output. It is important that clinicians are informed not only about the accuracy of algorithms, but also about their potential weaknesses and what to look out for.

Guarding against automation bias is not easy. While studies have suggested a number of strategies such as explainability and transparency of decision making, clear accountability and adaptive interfaces and task allocation, the evidence base for these is far from clear (Goddard et al., 2012). Different people might have different mental models and assumptions of the autonomous infusion system, which might be partial and even contradictory, because the behaviour may be too complex for anyone to understand what is going on. Technology developers, healthcare providers and clinicians need to have an awareness of this challenge, and find solutions that work in their specific setting so that users can build a good picture of the behaviour of the autonomous system in a way they can comprehend.

3.4 Supervision

With current infusion pumps (at L2 in our model of automation) clinicians are users of the infusion pump, i.e. they need to know how to load and program the

infusion pump. Failure modes of the infusion pump are fairly limited and reasonably well understood. The training provided to clinicians is about the functionality of the infusion pump and how to use it, e.g. how to navigate the interface.

The situation changes dramatically when we move to L5, because at this level the infusion pump becomes an autonomous system capable of taking decisions independently. The clinician needs to understand not only how to use the infusion pump, but also what potential weaknesses are and how the safe envelope is defined, maintained or breached. Clinicians need to be able to make sense of the pump's actions and provide clinically-based checks. In this sense, the role of the clinician changes from user of a passive pump to that of supervisor of an autonomous system. Consideration needs to be given to how clinicians can fulfil this role, and what kind of novel training needs might arise. It is even conceivable that a new role is created, e.g. that of an AI nurse specialist, who is specifically trained in managing AI and autonomous systems within their care setting.

4 Cross-Domain Discussion

This paper has identified a number of key human factors challenges through consideration of automation and autonomy in healthcare. Although the specific issues highlighted relate to the introduction of autonomous infusion pumps, the general challenges are not unique to this application and domain, and are likely to be more broadly applicable. In this section we discuss the generalisability of a number of the challenges presented through consideration of examples in other domains where the level of autonomy of systems is also increasing. We believe that there is great benefit that can be gained through sharing knowledge between domains on how to address these challenges.

4.1 Handover

The problems associated with handover from autonomous operation to a human operator are well known in other domains and have been widely studied. None more so than in the automotive domain where recent high-profile incidents have highlighted concerns around the use of so-called safety drivers. Current self-driving cars are only capable of driving autonomously under limited conditions such as defined geographical areas, types of roads or specified scenarios and environmental conditions. This means that the vehicle must hand back control to a human driver if the required conditions are not met, or if the car is in a situation that it

cannot resolve safely. Studies such as (Gold et al., 2013) have shown that, depending on the complexity of the situation at handover, it can take up to 8 seconds for a driver to take back full control of the vehicle, particularly if they are distracted at the time of handover. When driving on a motorway, 8 seconds may correspond to over 200 metres travelled. It does not seem unreasonable to take the high-end of this estimate. Given that the handover to a safety driver will often occur because the vehicle is in a difficult or dangerous state it is likely that the situation is complex. The ability of drivers who are not actively engaged in driving the vehicle to avoid distraction is also challenging, as discussed in (Merat et al., 2014). This is an area of active research, but there are certainly strongly held views, such as by Waymo (Waymo, 2018) that ultimately human drivers should be removed, as they cannot be relied upon to react quickly enough to ensure safety. This view has been given additional weight by accidents such as that in Tempe, Arizona in March 2018 (National Transportation Safety Board, 2018), where for various reasons the safety driver was unable to intervene quickly enough to prevent a fatality.

Human factors strategies such as graceful handover, situational awareness and designing with consideration of performance influencing factors, can also be seen to be crucial for ensuring safe handover in autonomous driving, and are also more broadly applicable to other domains.

4.2 Automation bias and impact on working practices

Aircraft have been highly automated for a long time, prompting research to investigate what the consequences of this might be on pilots' ability to fly the aircraft manually if the automated systems fail. This has been particularly motivated by a number of crashes that may have involved some element of de-skilling on the part of the pilots, such as Colgan Flight 3407 in 2009 when 50 people died when the pilots were found to have done the opposite of what they were trained to do when the aircraft entered an aerodynamic stall (National Transportation Safety Board, 2009), see figure 4 for an image of the crash site (Clarence Centre, New York). The paradox is that it would seem that although automation has made it increasingly unlikely that airline pilots will face critical problems during flight, it is also perhaps making it less likely they will be able to cope if such problems do arise.

Fig. 4. Crash site of Colgan Flight 3407 in 2009
(copyright: Bureau of Aircraft Accident Archives)

A study from 2014 (Casner et al., 2014) set out to understand how the prolonged use of cockpit automation has been affecting pilots' manual flying skills. They did this through experiments on 16 Boeing 747-400 airline pilots in a simulator, where they systematically varied the level of automation used to fly routine and non-routine flight scenarios. What they found was that pilots' instrument scanning and manual control skills to be mostly intact, even when pilots reported that they were infrequently practiced. However, when pilots were asked to manually perform the cognitive tasks needed for manual flight (e.g., tracking the aircraft's position without the use of a map display, deciding which navigational steps come next, recognizing instrument system failures), more frequent and significant problems were observed, and this seemed to depend on the degree to which pilots remain actively engaged in supervising the automation. Such observations in a domain where the use of high levels of automation are long established clearly bring knowledge that may be important for domains such as healthcare where high levels of autonomy are novel.

4.3 Supervision

In the maritime domain there are ambitious plans for autonomous operation of ships. For example, Rolls-Royce plan to operate a fleet of unmanned ships across

the world using a small number of operators in shore-based control centres (SCCs), which could be located thousands of miles away[6]. Unmanned shipping does not mean removing humans completely from operations, but moving them to a role more focussed on monitoring and supervision, requiring entirely new kinds of work roles, tasks, tools, training and environments. Crucially, to assure safety, people may need to be able to take some level of control over the ship at any time. Many of the issues relating to supervision for autonomous medical systems are therefore relevant here.

One of the big challenges of such a shift to SCCs is the loss of direct ship-sense. An investigation conducted in (Man et al., 2016) highlighted how critical ship-sense is in ship manoeuvring. They consider how operators in a remote operating centre will be able to effectively perceive the ship's movements and manoeuver the ship without ship-sense since there will be no physical connection between the human and the vessel, and no directly perceived information from the ship's environment. In (Wahlström et al., 2015), an overview of the human factors challenges that might concern future monitoring and control of unmanned ships from SCCs is presented. They identify the challenges through consideration of autonomous and remote operation across a number of domains including aviation, forestry, subway systems, space and military operations, and contrast these to the maritime context. The most prominent issues they identify include information overload, boredom, mishaps during changeovers and handoffs, lack of feel of the vessel, constant reorientation to new tasks, delays in control and monitoring, and the need for human understanding in local knowledge and object differentiation (e.g., in differentiating between help-seekers and pirates).

5 Conclusion

There is significant enthusiasm for the use of artificial intelligence in healthcare as well as in other industries, and there is no shortage of promise by technology developers of how AI can transform overstretched health services and improve patient care. There is also political will to support the development of new technologies with funding and by opening up relatively closed health systems such as the NHS. On the one hand this is good news, because these developments recognise the great potential that AI technologies undoubtedly bring. On the other hand, from a safety assurance perspective there is cause for concern because the evidence base on whether and how the introduction of such technologies might impact on patient safety is very thin. Largely, evaluation studies to date have considered performance of AI on specific tasks, but have neglected the

[6] Rolls Royce video available at:
https://www.youtube.com/watch?time_continue=1&v=vg0A9Ve7SxE

wider impact on clinical systems. One way forward might be to look not at algorithms in isolation, but rather consider the services AI systems are contributing to, and how the introduction of novel technologies will change the ways in which services are provided.

Standards and guidance exist, which could form a starting point for more rigorous safety assurance of AI technologies, such as established standards for risk management of medical devices (ISO 14971) and health information technology (NHS Digital clinical safety standards). However, these standards focus predominantly on technical aspects and do not cover human factors and service issues. In addition, many of the technology developers entering the AI healthcare market do not come from a safety-critical system engineering background and might be largely unfamiliar with existing guidance and best practice.

There is an opportunity for national bodies such as the Chartered Institute of Ergonomics and Human Factors (CIEHF) and the newly established NHSX to raise awareness of human factors and safety challenges for the use of AI in healthcare, and to develop and disseminate appropriate guidance. Funding should be made available not only for the development of AI technologies, but also for their rigours evaluation to ensure we understand from the outset how AI will impact on patient care and patient safety, and how potential hazards and human factors challenges can be addressed.

Appendix

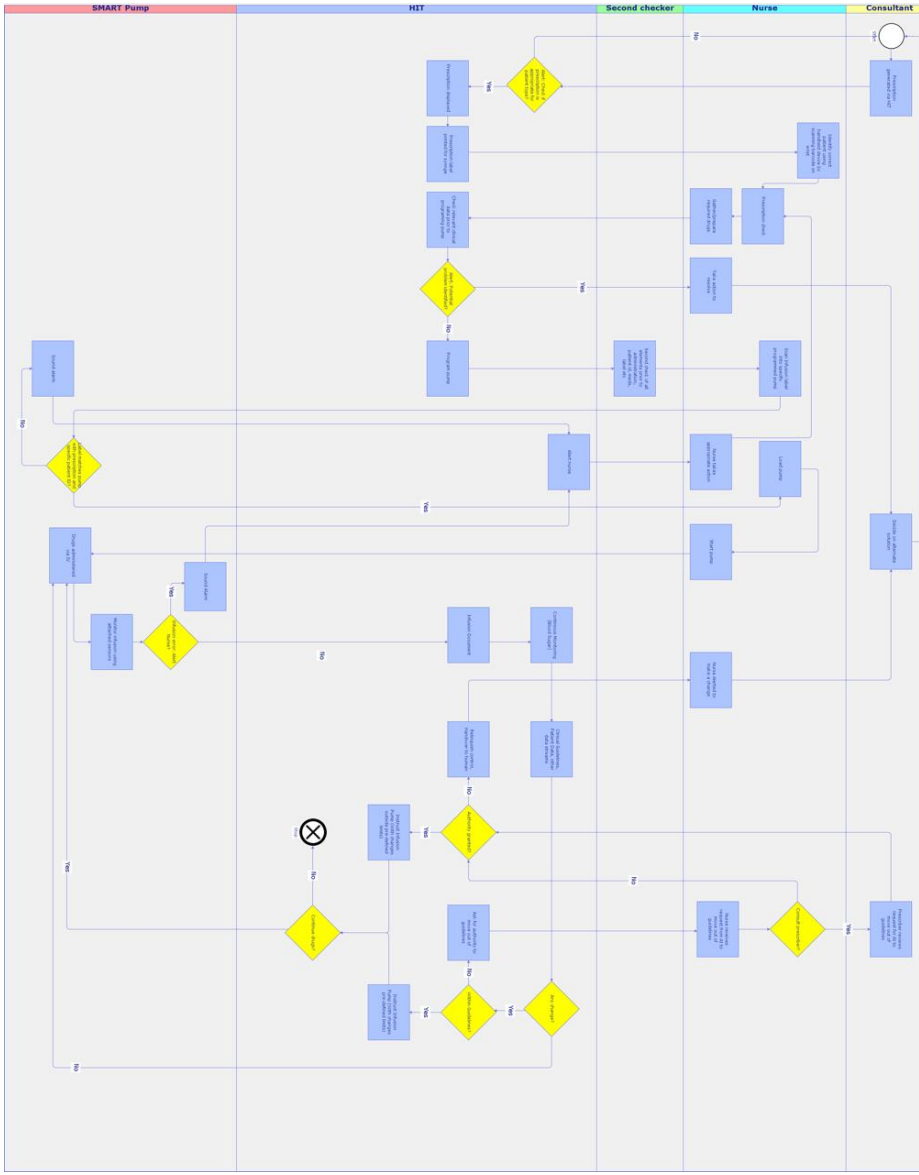


Fig. 5. Process map of infusion process with L5 autonomous infusion pump

Acknowledgments This work is supported by the Assuring Autonomy International Programme, a partnership between Lloyd's Register Foundation and the University of York. The process map in the appendix was developed by Shakir Laher using the NHS Digital SMART software. We are grateful to the clinical team at the Royal Derby Hospital (David Nelson, Matthew Elliott and Nick Reynolds) and the clinical safety team at NHS Digital (Sean White and Shakir Laher) for their contribution to the SAM demonstrator project.

References

ALBERDI, E., POVYAKALO, A., STRIGINI, L. & AYTON, P. 2004. Effects of incorrect computer-aided detection (CAD) output on human decision-making in mammography. *Academic Radiology,* 11**,** 909-918.

AVRAM, R., TISON, G., KUHAR, P., MARCUS, G., PLETCHER, M., OLGIN, J. E. & ASCHBACHER, K. 2019. PREDICTING DIABETES FROM PHOTOPLETHYSMOGRAPHY USING DEEP LEARNING. *Journal of the American College of Cardiology,* 73**,** 16.

BURNETT, G., LARGE, D. R. & SALANITRI, D. 2019. How will drivers interact with vehicles of the future? London: RAC Foundation.

CASNER, S. M., GEVEN, R. W., RECKER, M. P. & SCHOOLER, J. W. 2014. The Retention of Manual Flying Skills in the Automated Cockpit. *Human Factors,* 56**,** 1506-1516.

CHILAMKURTHY, S., GHOSH, R., TANAMALA, S., BIVIJI, M., CAMPEAU, N. G., VENUGOPAL, V. K., MAHAJAN, V., RAO, P. & WARIER, P. 2018. Deep learning algorithms for detection of critical findings in head CT scans: a retrospective study. *The Lancet,* 392**,** 2388-2396.

COIERA, E. 2018. The fate of medicine in the time of AI. *The Lancet,* 392**,** 2331-2332.

ELLIOTT, R. A., CAMACHO, E., CAMPBELL, F., JANKOVIC, D., ST JAMES, M. M., KALTENTHALER, E., WONG, R., SCULPHER, M. J. & FARIA, R. 2018. Prevalence and economic burden of medication errors in the NHS in England. Sheffield: Policy Research Unit in Economic Evaluation of Health & Care Interventions.

EMBREY, D. 1986. SHERPA: A systematic human error reduction and prediction approach. *Proceedings of the International Topical Meeting on Advances in Human Factors in Nuclear Power Systems.* Knoxville, Tennessee: American Nuclear Society.

ENDSLEY, M. R. 1995. Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors,* 37**,** 32-64.

ESTEVA, A., KUPREL, B., NOVOA, R. A., KO, J., SWETTER, S. M., BLAU, H. M. & THRUN, S. 2017. Dermatologist-level classification of skin cancer with deep neural networks. *Nature,* 542**,** 115.

FITZPATRICK, K. K., DARCY, A. & VIERHILE, M. 2017. Delivering Cognitive Behavior Therapy to Young Adults With Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial. *JMIR Ment Health,* 4**,** e19.

FURNISS, D., DEAN FRANKLIN, B. & BLANDFORD, A. 2019. The devil is in the detail: How a closed-loop documentation system for IV infusion administration contributes to and compromises patient safety. *Health Informatics Journal***,** 1460458219839574.

GODDARD, K., ROUDSARI, A. & WYATT, J. C. 2012. Automation bias: a systematic review of frequency, effect mediators, and mitigators. *J Am Med Inform Assoc,* 19**,** 121-127.

GOLD, C., DAMBÖCK, D., LORENZ, L. & BENGLER, K. 2013. "Take over!" How long does it take to get the driver back into the loop? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting.* SAGE Publications

GULSHAN, V., PENG, L., CORAM, M., STUMPE, M. C., WU, D., NARAYANASWAMY, A., VENUGOPALAN, S., WIDNER, K., MADAMS, T., CUADROS, J., KIM, R., RAMAN, R., NELSON, P. C., MEGA, J. L. & WEBSTER, D. R. 2016. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus PhotographsAccuracy of a Deep Learning Algorithm for Detection of Diabetic RetinopathyAccuracy of a Deep Learning Algorithm for Detection of Diabetic Retinopathy. *JAMA,* 316**,** 2402-2410.

HAENSSLE, H. A., FINK, C., SCHNEIDERBAUER, R., TOBERER, F., BUHL, T., BLUM, A., KALLOO, A., HASSEN, A. B. H., THOMAS, L., ENK, A., UHLMANN, L., LEVEL-I, R. S. & GROUPS, L.-I. 2018. Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists. *Annals of Oncology,* 29**,** 1836-1842.

HOLLNAGEL, E. 2014. *Safety-I and Safety-II,* Farnham, Ashgate.

HOLLNAGEL, E. 2015. Why is Work-as-Imagined different from Work-as-Done? *In:* WEARS, R., HOLLNAGEL, E. & BRAITHWAITE, J. (eds.) *The Resilience of Everyday Clinical Work.* Farnham: Ashgate.

HOLLNAGEL, E., WOODS, D. D. & LEVESON, N. 2006. *Resilience Engineering: Concepts and Precepts,* Aldershot, Ashgate.

JETER, R., JOSEF, C., SHASHIKUMAR, S. & NEMATI, S. 2019. Does the" Artificial Intelligence Clinician" learn optimal treatment strategies for sepsis in intensive care? *arXiv* 1902.03271.

KOMOROWSKI, M., CELI, L. A., BADAWI, O., GORDON, A. C. & FAISAL, A. A. 2018. The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine,* 24**,** 1716-1720.

LYELL, D. & COIERA, E. 2016. Automation bias and verification complexity: a systematic review. *Journal of the American Medical Informatics Association,* 24**,** 423-431.

MAN, Y., LUNDH, M. & PORATHE, T. 2016. Seeking Harmony in Shore-Based Unmanned Ship Handling - From the Perspective of Human Factors, What Is the Difference We Need to Focus on from Being Onboard to Onshore? *In:* AHRAM, T., KARWOWSKI, W. & MAREK, T. (eds.) *5th Int Conf on Appl Human Factors and Ergonomics AHFE.* Krakow, Poland: CRC Press.

MCLEOD, M. C., BARBER, N. & FRANKLIN, B. D. 2013. Methodological variations and their effects on reported medication administration error rates. *BMJ Quality & Safety,* 22**,** 278-289.

MERAT, N., JAMSON, A. H., LAI, F. C. H., DALY, M. & CARSTEN, O. M. J. 2014. Transition to manual: Driver behaviour when resuming control from a highly automated vehicle. *Transportation Research Part F: Traffic Psychology and Behaviour,* 27**,** 274-282.

NATIONAL TRANSPORTATION SAFETY BOARD. 2009. Accident report - Loss of Control on Approach Colgan Air, Inc. Operating as Continental Connection Flight 3407 Bombardier DHC-8-400, N200WQ, NTSB/AAR-10/01. Available: https://www.ntsb.gov/investigations/AccidentReports/Reports/AAR1001.pdf.

NATIONAL TRANSPORTATION SAFETY BOARD. 2018. Preliminary Report Highway HWY18MH010. Available: https://www.ntsb.gov/investigations/AccidentReports/Reports/HWY18MH010-prelim.pdf.

PARASURAMAN, R. & RILEY, V. 1997. Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors,* 39**,** 230-253.

SARIA, S., BUTTE, A. & SHEIKH, A. 2018. Better medicine through machine learning: What's real, and what's artificial? *PLoS Med,* 15**,** e10002721.

SEMIGRAN, H. L., LINDER, J. A., GIDENGIL, C. & MEHROTRA, A. 2015. Evaluation of symptom checkers for self diagnosis and triage: audit study. *BMJ,* 351**,** h3480.

STANTON, N. 2006. Hierarchical task analysis: Developments, applications, and extensions. *Appl Ergon,* 37.

SUJAN, M., FURNISS, D., EMBREY, D., ELLIOTT, M., NELSON, D., WHITE, S., HABLI, I. & REYNOLDS, N. 2019a. Critical barriers to safety assurance and regulation of autonomous medical systems. *In:* BEER, M. & ZIO, E. (eds.) *29th European Safety and Reliability Conference (ESREL 2019).* Hannover: CRC Press.

SUJAN, M., FURNISS, D., GRUNDY, K., GRUNDY, H., NELSON, D., ELLIOTT, M., WHITE, S., HABLI, I. & REYNOLDS, N. 2019b. Human factors challenges for the safe use of artificial intelligence in patient care. *BMJ Health & Care Informatics,* 26**,** e100081.

SUJAN, M., HUANG, H. & BIGGERSTAFF, D. 2019c. Trust and Psychological Safety as Facilitators of Resilient Health Care. *In:* BRAITHWAITE, J., HOLLNAGEL, E. & HUNTE, G. (eds.) *Resilient Health Care V: Working Across Boundaries.* CRC Press.

SUJAN, M., SCOTT, P. & CRESSWELL, K. 2019d. Digital health and patient safety: Technology is not a magic wand. *Health Informatics Journal*.

SUJAN, M., SPURGEON, P. & COOKE, M. 2015a. The role of dynamic trade-offs in creating safety—A qualitative study of handover across care boundaries in emergency care. *Reliability Engineering & System Safety,* 141**,** 54-62.

SUJAN, M. A., CHESSUM, P., RUDD, M., FITTON, L., INADA-KIM, M., COOKE, M. W. & SPURGEON, P. 2015b. Managing competing organizational priorities in clinical handover across organizational boundaries. *Journal of Health Services Research & Policy,* 20**,** 17-25.

SUJAN, M. A., CHESSUM, P., RUDD, M., FITTON, L., INADA-KIM, M., SPURGEON, P. & COOKE, M. W. 2015c. Emergency Care Handover (ECHO study) across care boundaries: the need for joint decision making and consideration of psychosocial history. *Emergency Medicine Journal,* 32**,** 112-118.

WAHLSTRÖM, M., HAKULINEN, J., KARVONEN, H. & LINDBORG, I. 2015. Human factors challenges in unmanned ship operations–insights from other domains. *Procedia Manufacturing,* 3**,** 1038-1045.

WAYMO. 2018. Waymo Safety Report - On the Road to Fully Self-Driving. Available: https://storage.googleapis.com/sdc-prod/v1/safety-report/Safety%20Report%202018.pdf.

YEUNG, S., DOWNING, N. L., FEI-FEI, L. & MILSTEIN, A. 2018. Bedside Computer Vision — Moving Artificial Intelligence from Driver Assistance to Patient Safety. *New England Journal of Medicine,* 378**,** 1271-1273.

YU, K.-H. & KOHANE, I. S. 2019. Framing the challenges of artificial intelligence in medicine. *BMJ Qual Saf,* 28**,** 238-241.

Psychological safety - facilitating self-reporting of error, mistakes and non-compliance: A rapid review for the Energy Institute

Michael Wright*, Sam Opiah* and Suzanne Croes†

*Greenstreet Berman Ltd. London, UK[1]

†Shell

Abstract *The Energy Institute, on behalf of Shell, commissioned a rapid review of psychological safety. Psychological safety can be described as the willingness of people to express an opinion, admit mistakes or unsafe behaviours, without fear of being embarrassed, rejected or punished. Psychological safety plays a role in facilitating the reporting of errors and unsafe behaviours – thereby enabling these to be identified, learnt from and improvements made to prevent repetition of errors and unsafe behaviours. Psychological safety is particularly important in hierarchical organisations, often with complex systems, where error may have serious safety consequence, and where individuals or organisations are held responsible for adverse consequence. Interventions to enhance psychological safety include tools to support analysis of causes of error and behaviours, displayable effort by management and the organisation to build trust and teamwork between staff and themselves as well as supporting and encouraging safety-related behaviour.*

1 Introduction

The Energy Institute (EI) Human and Organisational Factors Committee (HOFCOM) aims to help the energy and allied industries to understand and apply human and organisational factors to its operations by commissioning studies, providing information and sharing knowledge. The committee, at Shell's request asked for a review to summarise evidence regarding the association between psychological safety and health and safety performance and how to build psychological safety. The focus was on high hazard and safety-critical sectors, including

[1] 10 Fitzroy Square, London, W1T 5HP. info@greenstreet.co.uk www.greenstreet.co.uk

© Greenstreet Berman Ltd, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

but not limited to energy, mass transportation, defence, medical, mining, shipping, emergency services and aerospace. The review (Wright and Opiah, 2018) primarily covered peer reviewed and published literature. The EPPI Centre's Weight of Evidence (WoE)[1] framework was used to assess the reliability and weight of evidence and to ensure the review was limited to quality evidence, with 104 evidential items cited. There was a high number of studies indicating a link between psychological safety and human/HSE performance, especially on human performance such as reporting error.

2 What is psychological safety?

Psychological safety can be described as:

> "The willingness of people to express an opinion, admit mistakes or unsafe behaviours, without fear of being embarrassed, rejected or punished." Wright and Opiah (2018)

The concept of psychological safety was initially developed (Schein and Bennis 1965) in the context of complex work environments with high levels of human interaction, particularly in healthcare, where there is a need to promote reporting of human error, learning and positive (as opposed to defensive) responses to error.

Two specific lines of work that have cited similar concepts are 'safety climate' and Professor James Reason's work on 'Just cultures' (Reason 1997 and 1998). These other lines of work have used terms that are analogous to elements of psychological safety, such as openness; trust; just culture and speaking up. Psychological safety goes beyond reporting near misses and unsafe conditions to focus on reporting one's own errors and unsafe behaviours. It is noted that reporting one's own errors of commission is a greater challenge, due to the fear of reprisal, than reporting unsafe equipment (Edmondson, 1996), as are social barriers in speaking up about colleague's behaviour (Martinez et al. 2015).

Whilst psychological safety is reported to be a sub-element of safety culture, Tuyl (2016) asserts that:

> "...cultures of non-report exist within some organisations in spite of noble efforts to foster a supportive safety culture" (p15).

Tuyl (2016) cites a study of safety programs in five construction companies that, whilst representing good practice, reported ongoing non-reporting practices. This implies that there are specific factors that have a particular impact on psychological safety and that a 'good' safety climate does not necessarily lead to a sense of psychological safety. Psychological safety does not necessarily emerge as a product of a positive safety climate.

[1] http://eppi.ioe.ac.uk/cms/Default.aspx?tabid=88

The authors of this review would also note that as many models of safety climate omit psychological safety, safety climate interventions based on these models may not target the factors specific to psychological safety. Indeed, Martinex et al (2015) found in a healthcare study that teamwork and safety climate scales were not associated with self-reported speaking up behaviour. The evidence suggests that there are specific factors related to psychological safety.

3 Why is psychological safety considered important?

Psychological safety is thought to be a critical attribute of a team and organisation climate, and plays a role in:

- Facilitating the reporting of errors and unsafe conditions – thereby enabling these to be identified, investigated, learnt from and improvements made to prevent their repetition;
- Enabling people to challenge other people and query their performance;
- Facilitating effective investigation and understanding errors and unsafe behaviours, such as being able to perform valid behavioural analysis and solicit truthful statements of actions, behaviours and decision making that may have contributed to incidents.

The capacity of an organisation to identify areas of weakness in safety performance (as indicated by errors and unsafe behaviours) and effectively resolve these proactively is considered to contribute to the prevention of incidents and / or prevention of the repetition of the same or similar errors and unsafe behaviours. This is said to be particularly so where there is a need to improvise, in situations of uncertainty and / or making decisions without specific protocols (i.e. where people may err), Edmondson et al (2016).

Psychological safety was cited as a factor in the March 2005 Texas City explosion shown in figure 2, below. A hydrocarbon vapour cloud was ignited and exploded at the ISOM isomerization process unit at the refinery killing 15 workers, injuring 180 others and severely damaging the refinery. A raffinate splitter tower was mistakenly over filled resulting in hydrocarbons entering an overhead vapour line and via a relief valve system to a blowdown drum and stack from which the hydrocarbons were released to atmosphere and ignited. Communications errors, defective equipment and operational errors contributed to the incident, with a background of fatigue, shortcomings in safety management and preventative maintenance. BP had conducted its own audits and reviews prior to the explosion which had identified shortcomings but, according to the enquiry without adequate follow up.

Figure 1 shows the relevant recommendation from the Baker report (2007):

"BP should involve the relevant stakeholders to develop a positive, trusting, and open process safety culture within each U.S. refinery.

"develop a positive, trusting, and open process safety culture"—

(f) distinguish more clearly between acceptable and unacceptable employee acts such that the vast majority of unsafe acts or conditions can be reported without fear of punishment. A strong process safety culture facilitates the sharing of information that will reduce safety risks. As a result, BP's refineries should operate in such a way as to permit the reporting of the vast majority of unsafe acts or conditions by employees and contractors without fear of punishment. While unsafe acts that are reckless or particularly egregious may warrant some type of sanctions, the culture of each U.S. refinery should promote sharing of information relevant to safety even when that information indicates that workers have made mistakes;

(g) establish a climate in which:

workers are encouraged to ask challenging questions without fear of reprisal, and

workers are educated, encouraged, and expected to examine critically all process safety tasks and methods prior to taking them

Fig. 1: Baker report, 2007, recommendation 4, p249

Fig. 2. BP Texas City refinery explosion, 2005

4 What factors influence psychological safety?

Psychological safety is reported (e.g. Edmondson et al. 2016) to be a product of a team and organisational environment where people feel safe to express opinions and report matters such as mistakes without adverse consequences. Key attributes of a climate that engenders psychological safety include:

- Trust in how people will respond to your opinion and statements, such as whether you might be ridiculed, criticised, embarrassed, ostracised or blamed etc. instead of being thanked and supported;
- Trust in the actions people or your employers may or may not take in response to your opinions or admissions, such as whether they take disciplinary action or not, downgrade your performance assessment etc. versus positively rewarding your openness.
- The individual's sense of vulnerability in the event that they make a mistake, particularly whether an accountability culture leads to adverse consequences as opposed to being offered support and using the mistake as a learning opportunity.

Table 1. Factors influencing psychological safety

Factor	Summary
Individual and team factors	• Number of years employed, and years employed in current team. • The extent of social affiliation within a team (which can increase fear of social stigma from reporting or challenging).
Organisational attributes:	• Hierarchy – the degree of authority and respect afforded to individuals based on their position. • The extent to which professions are siloed. • Hierarchical status – higher grades such as supervisors may have higher levels of psychological safety.
Accountability culture	The extent to which individuals are accountable or have a sense of vulnerability if they share (for example) a need to learn.
Organisational climate	The organisational climate with respect to whether "speaking up" is an aspect of professional behaviour.
Leadership	• Acknowledging fallibility and proactively seeking input. • Explicit display of openness, availability and accessibility. • Staff perceptions that leaders acknowledge their contribution. • Staff provided with opportunity to contribute ideas that may challenge norms and may be seen as risky.

5 Psychological safety and learning from error

There is extensive advice on learning cultures, especially in the context of Prof Reason's research and guidance on Just Cultures. This research (e.g. Reason 1997, 1998) is not limited to responding to self-reports of error, including responding to reports of unsafe equipment, safety concerns and lessons learnt from incidents. The work focuses on the well reported topics of organisational learning and a learning culture, including:

- An organisational motivation to learn and a willingness change;
- A focus on identifying and resolving underlying causes of error.

Specific to learning from error, the research notes the importance of:

- An organisation viewing error as a latent hazard to be learnt from; and
- To collectively act to avoid the same errors in the future.

This has led to the concept of an error management culture. Guchait et al. (2014) define error management culture as one that:

> "involves organizational practices related to communicating about errors, sharing error knowledge, quickly detecting and handling errors, and helping in error situations."

Indeed, Krauss and Casey (2014) argue that:

> "…error management climate creates an opportunity for aligning and improving both safety and operational performance".

They refer to error management climate as:

> "…employees' perceptions of the extent to which the organization encourages communication about and management of errors and mistakes in the workplace."

Error is seen as a positive learning opportunity and a means by which teams and organisations can improve their performance. A collective response to error is thought to help move attention away from the individual and towards a shared sense of responsibility that in turn leads to a focus on the "system" related causes of error.

6 Concerns about the implementation of a Just Culture

There has been widespread adoption of Just Culture in US healthcare and the aviation sector. For example, Edwards (2018) reported that 79% of US acute hospitals have adopted a Just Culture model. The evidence regarding the success of Just Culture interventions is inconsistent. In practice, the success of Just Culture is, in part, a product of:

- Whether the form of Just Culture includes a "consequence management" policy of holding people 'culpable' if they knowingly do not follow policy or procedure, instead of a 'restorative' form of Just Culture (see Dekker and Breakey, 2016) that aims to learn from mistakes and behaviour and views these as a product of organisational systems and culture.
- The extent to which an organisation embeds the principles and practices through training and education of directors, managers and staff.
- The effectiveness of arrangements for reporting, analysis and feedback of actions.
- Whether staff feel they can more easily resolve an error locally than report it, and whether reporting a locally resolved error has a purpose.

The healthcare research cites ongoing fears of reporting with a suggestion that the remaining fear of adverse consequences (no granting of 'immunity') curtails the impact of Just Culture initiatives. This raises the question of how effective Just Culture interventions are without the granting of 'immunity'.

In addition, Edwards' (2018) discussion casts doubt on the practical application of a Just Culture algorithm for evaluating 'blame worthy' vs 'blameless 'acts (i.e. does it lead to blame) and whether the Just Culture 'culpability' model takes sufficient account of the need for trust and the factors influencing organisational learning. Bitar et al. (2018) found that the 1997 formulation of Just Culture algorithm did not always lead to outcomes consistent with the intent and contributed to instances of inappropriate blame. They reformulated the algorithm to place greater focus on the organizational and cultural antecedents of behaviour and less on the notion of individual culpability.

7 Interventions to promote psychological safety?

There is a substantial body of guidance on how to increase psychological safety. This suggests it is necessary to overcome hierarchical barriers and fears about peer group (social) or organisational reactions by:

- Team building (developing trusting interpersonal relationships);
- Having (inclusive) leaders and role models facilitating learning through adoption of a set of supportive behaviours, accessibility, neutral language and positive reinforcement of reporting;
- Generating a sense that error is a shared learning opportunity and a collective responsibility aimed at performance improvement;
- Demonstrating the value of speaking up by acting effectively on feedback and reporting actions back to people.

The guidance focuses on:

- Building trust and teamwork amongst peers and between staff and management;
- Supportive managerial and organisational processes and behaviours.

The research indicates that a multi-faceted approach is needed, including the tactics shown in figure 3:

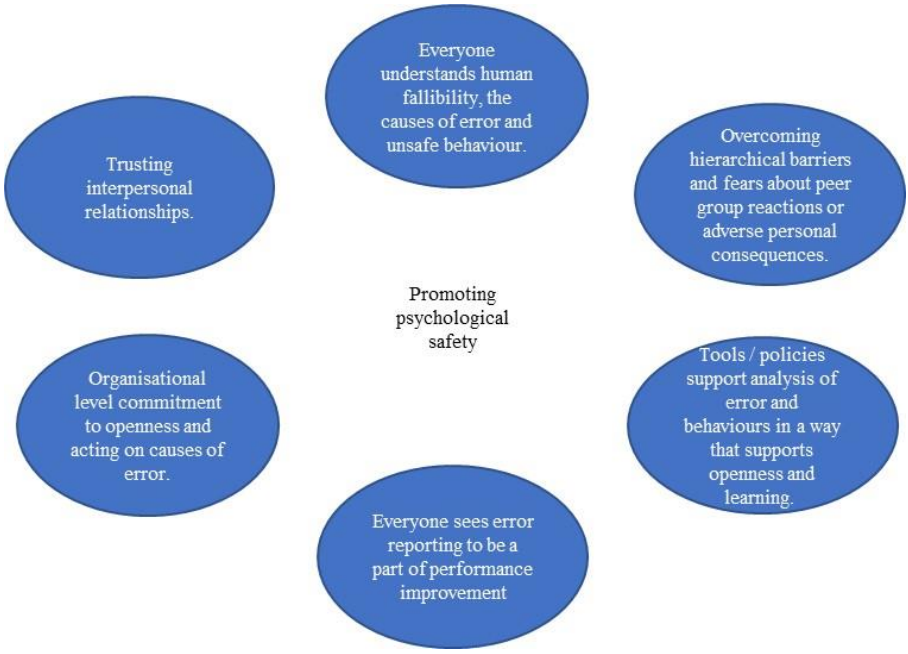


Fig. 3. Overview of tactics for promoting psychological safety

A fear of adverse organisational reactions to reports of error can undermine reporting behaviour even where Just Culture initiatives have been introduced. A summary of good practice for building psychological safety and learning from error is given below in table 2.

Table 2: Guidance on developing psychological safety

Good practice	Guidance
Interactive education	A re-orientation of "hearts and minds" forms of engagement and training at all organisational levels to achieve:

Good practice	Guidance
	• A common recognition and acceptance throughout the organisation of the value of reporting error and unsafe behaviours in respect of improving performance. • Recognition that employees may fear reporting/speaking up and that specific steps must be taken to facilitate reporting and learning.
Team building and trust building behavioural interventions	Development of inclusive leadership skills through non-technical skills training specific to the creation of psychological safety, including: • Inclusive leadership and supportive facilitation of employee engagement; • Accessible, respectful, collegiate, open, neutral, positive language, etc.
Supportive organisational environment	A supportive environment created by: • Positive response to reporting of error; • Demonstration of commitment to learning from error; • Learning performed as a collective exercise.
Demonstrating value of reporting and speaking up	The value of reporting and evidence of management commitment should be reinforced by timely and effective responses; and feedback on actions taken.
Learning from error	The following attributes are cited at the team and at the organisational level: • Learning and improvement are objectives; • Error is seen to be an opportunity to learn and improve; • Learning is a shared and collective responsibility; • Awareness and acceptance of a systems approach to causation of error and unsafe behaviour and need to address underpinning causes; • Feedback to personnel on actions taken; • Openness to change.
Tracking success	Tools such as surveys of psychological safety, employee perceptions of the risk of reporting and reporting behaviour may be used to track and measure success of interventions.

8 Gaps in the evidence

Key areas for further research include:

- Benchmarking levels of psychological safety and associated levels of error self-reporting in safety critical industries

  This review did not identify any published assessments of the current extent to which personnel in the oil and gas sector are willing to report error nor of the effectiveness of organisational responses to self-reported error. The added value of new interventions will depend in part on the baseline level of psychological safety.

- Evaluations of the effectiveness of interventions in increasing psychological safety and HSE performance

  There are few, if any, real world evaluations of the effectiveness of interventions aimed at increasing psychological safety and HSE performance in the oil and gas sector. Such evaluations should ideally include before and after longitudinal evaluations, preferably with control groups, and use a combination of measures covering psychological safety, reporting behaviour and improvements in safety from learning.

- Measures of psychological safety / Just culture

  The extent to which assessment tools, such as safety climate questionnaires, include psychological safety/ Just culture and/or the need to adopt discrete psychological safety measures could be further researched. Whilst some measures have been developed (for example Petschonek et al. (2013) for a measure of Just Culture), their application and validation in the context of oil and gas or other high hazard operations could be further researched.

- Guidance on psychological safety

  Whilst there are many guides on psychological safety, these are not specific to HSE performance.

9 Conclusions

The evidence indicates that building psychological safety leads to a higher rate of reporting of error and unsafe behaviours, thereby enabling learning from error and preventing incidents. In particular:

- A sense of vulnerability if a person reports an error may reduce the benefit of interventions such as facilitative leadership and reporting schemes; and
- A sense of trust and facilitative leadership may overcome hierarchical barriers to speaking up.

The ability to learn from error is reported to be related to:

- Error being seen by the organisation as a learning opportunity and a shared experience about what works and what does not work;
- Learning from error being seen to be a collective responsibility.

The confidence in effective action is in turn influenced by perception of organisational commitment to safety and past responses to reported errors or unsafe behaviour. There are validated questionnaire measures of psychological safety (see Edmonson 1999) that have been correlated with measures of team performance. Further work could usefully measure the level of psychological safety in high-hazard organisations and evaluate the success of interventions in improving their safety performance.

Acknowledgments This review was completed by Greenstreet Berman Ltd on behalf of the EI HOFCOM and Shell. We would like to acknowledge the advice and contributions of the members of HOFCOM and Suzanne Croes of Shell to this review.

References

Baker, J., N. Leveson, F. Bowman, S. Priest, G. Erwin, I. Rosenthal, & L. D. Wilson. (2007). The Report of the BP U.S. Refineries Independent Safety Review Panel.

Bitar, F.K, Chadwick-Jones, D., Nazaruk, M., & Boodhai, C. (2018). From individual behaviour to system weaknesses: The re-design of the Just Culture process in an international energy company. A case study. *Journal of Loss Prevention in the Process Industries*. 55, 267-282.

Dekker, S and Breakey, H. Just culture:' Improving safety by achieving substantive, procedural and restorative justice. *Safety Science*, Volume 85, (June 2016), Pages 187-193

Edmondson, A. Psychological Safety and Learning Behavior in Work Teams. (1999) *Administrative Science Quarterly*. Volume: 44 issue: 2, page(s): 350-383

Edmondson, A.C., Singer, M., Weiner, J., & Higgins, M. (2016). Understanding Psychological Safety in Health Care and Education Organizations: A Comparative Perspective. *Research in Human Development*. 13 (1), 65-83.

Guchait, P., Paşamehmetoğlu, A., & Madera, J. (2016). Error management culture: impact on cohesion, stress, and turnover intentions, *The Service Industries Journal*, 36 (3-4), 124-141.

Krauss, A. D., & Casey, T. (2014). Error Management Climate as a Way to Align Safety Objectives with Operational Excellence. Society of Petroleum Engineers.

Martinez, W., Lehman, L.S., Thomas, E.J., Etchegaray, J.M., Shelburne, J. T., Hickson, G.B., Schleyer, A.M., Best, J.A., May, N.B., & Bell, S.K. (2015). 'Speaking up' about patient safety concerns and unprofessional behaviour among residents: validation of two scales. *British Medical Journal Quality & Safety*, 24 (11), 671-680.

Petschonek, S., Burlison, J., Cross, C., Martin, K., Laver, J., Landis, R. S., & Hoffman, J. M. (2013). Development of the just culture assessment tool: measuring the perceptions of health-care professionals in hospitals. *Journal of patient safety*, 9 (4), 190-7.

Reason, J. (1997). Managing the Risks of Organizational Accidents.

Reason, J. (1998). Achieving a safe culture: theory and practice. Work & Stress.12 (3), 293-306.Schein, E. H., & Bennis, W. G. (1965). Personal and organizational change through group methods: The laboratory approach. New York, NY: Wiley.

Van Tuyl, R. M. (2016). Safety Culture in oil and gas: Factors that contribute to cultures of non-report. Masters of Arts in professional Communication Thesis.

Wright, M. and Opiah, S. (2018). Literature review: the relationship between psychological safety, human performance and HSE performance. Published by Energy Institute, 2018. https://heartsandminds.energyinst.org/research

Safety Systems and Defence in Depth in Nuclear New Build

Alastair Crawford

EDF Energy

Abstract *Hinkley Point C in Somerset is the first new nuclear power station to be constructed in the UK in a Generation. It is a light water "EPR" reactor, based on a design very similar to those which have just begun commercial operation in China and are nearing completion in France and Finland. This is one of the largest infrastructure projects in the World, costing around £20bn, employing 1000s of people on site and has a truly international supply chain. As a "Third Generation" reactor, it includes several safety and efficiency improvements compared to the previous generations of reactors which were designed and constructed over the last 50 years. The safety systems have been developed using a Defence in Depth approach with multiple redundant and diverse systems to reduce the frequency of an event leading to core melt significantly lower than previous generation reactors. There are additional design features to ensure that in the extremely remote event of a "severe accident", the resultant core melt is managed and cooled using engineered systems. This design philosophy of engineered redundant and diverse mechanical and electrical systems is mirrored in the I&C systems. There are two independent digital control and protection systems, and a third non-computerised system which are largely independent of each other but act in a hierarchical manner to provide very high levels of reliability. This key note speech will describe how the design has achieved very high safety and reliability levels using the defence in depth approach, explain how this is justified in the safety case and provide some insight as to how independent oversight is provided on such a complex project*

© Copyright Alastair Crawford, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

Site Images

Fig. 1. Hinkley Point C Image 1

Fig. 2. Hinkley Point C Image 2

Issues with Rules for Autonomous Vehicle Safety

Michael Ellims*, John Botham

Ricardo UK Ltd.

Cambridge, UK

Abstract *In this paper we examine the question of whether a vehicle that is following the "rules of the road" can always be regarded as operating safely, especially in an environment where human actors are operating. We do this in two ways: first we pose the question, "what is a 'lane'?" to highlight the problem of defining sys-tem behaviours in terms a human would understand; we then look at a pair of well-formed rules and assess the possible consequences of applying those rules in a traffic environment with a mixture of human-driven and autonomous vehicles (AVs). Overall the paper highlights the multitude of problems associated with defining how AVs should behave in a mixed human AV environment.*

1 Introduction

One significant issue with increasing automation in vehicles is that we do not really understand in any "deep" sense how people drive, in much the same way that we don't in any deep sense currently understand the nature of conscious (McGinn 1989).

Driving is an activity that for any human driver is learnt by practice and, in general, if we ignore specific instances of "bad" drivers, the more one practises the better one becomes – up to some limit. What driving is not, is the direct application of the rules of the road. Rather we should consider driving to be an activity that is constrained by the rules of the road.

This is important when we are discussing autonomous[1] vehicles because it shows that there are distinctions between different exemplars of driving. For example, consider the following descriptors of driving:

- Driving legally
- Driving safely

* Corresponding Author: Michael.ellims@ricardo.com

[1] Autonomous: where some or all the driving functions are automated to a high degree

© Ricardo UK Ltd. 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

- Driving timidly
- Driving dangerously

If we use these descriptions as templates, we could form logically correct phrases such as "they were driving dangerously but driving legally". For example, obeying the speed limit on a motorway in dense fog obeys the law in respect of speed limits, but is probably not following the "due care and attention" law, nor a good idea in the Darwinian sense[2]. Likewise, the phrase "they were driving timidly but also driving dangerously" can be correct in that "they" might be so timid as to incite risky behaviour from other road users.

There are several publications that suggest "rules" or guidelines for autonomous vehicles (AVs); two notable examples are TR 68 (Enterprise Singapore 2019) and Safety First for Automated Driving (Committee 2019). Some are framed as "formal" models (Shalev-Shwartz et al. 2017). The purpose of this paper is to draw attention to the issue that currently these rules are quite loosely worded and, in some cases, possibly framed badly for an environment that contains a significant proportion of vehicles controlled by humans.

2 What is a "lane"?

When we talk about a "lane" what do we mean? There are numerous examples currently where partly autonomous vehicles have apparently misinterpreted the road layout and ended up driving in the wrong place (Lambert, 2019). Some of them are superficially humorous, possibly because they appeal to our innate feeling of superiority in decision making. At least one resulted in a death when a Tesla Model X drove into a concrete divider NTSB (2018).

It should be noted that any sense of superiority is misplaced and that humans do not always get it right. For instance, the incident with the Model X noted above can be partly attributed to the fact that the crash barrier was not fully replaced after another driver, in a non-autonomous vehicle, had collided with it.

As an example of how a human can fail to correctly detect a "lane," images from dash-cam video taken by one of our colleagues on the A14 (a dual-carriageway road with multiple lanes) is shown in Figure 1. The top frame shows the confusing state of the lines at some points on the road. The middle frame shows that the vehicle ahead has settled in to drive in what the driver believes is a lane, but is actually the hard shoulder. The bottom frame shows the point where the driver corrects the trajectory of the vehicle, to avoid impacting a wall at the end of the hard shoulder.

[2] As there is a high probability of an accident, it may result in the driver removing themselves from the gene pool.

There are of course mitigating circumstances; this occurred in a particularly confusing area of road works. Old lines exist but have been painted out in black and the direction of travel is toward the setting sun, so reflections make it hard to distinguish between the black and white painted lines.

Fig. 1. A14 Junction with M11 west-bound, August 2019

Defining "lane" is important as it is a term that appears frequently in literature setting out approaches to the behavioural safety of AVs. For example, TR68 (Enterprise Singapore 2019) has the following: "A total failure of the AV's sensor system is detected… decelerating to a stop in the current lane" (Enterprise Singapore 2019 pg. 21) and "An AV shall keep within its lane unless it is performing a lane change or overtaking manoeuvre" (Enterprise Singapore 2019 pg. 22). The term is used 198 times in the document but there does not appear to be a definition, whether formal, semi-formal or informal. It is assumed that this is because lanes as marked on the road appear to be especially well defined and maintained in Singapore, at least in comparison with the UK. A search with Google Street View shows no roads without at least painted centre lines, except in the local cemeteries. However, on a recent visit one of the authors found that maintenance is not perfect and located several areas where wear was apparent.

Likewise, the concept of "lane" is used widely in Safety First for Automated Driving (Committee 2019), which gives a safe state or minimal risk condition as "Vehicle is stopped in-lane". Again, no definition of "lane" is found.

The Highway Code for the UK (2019) is slightly better as it identifies the markings that define lanes. However, beyond that it assumes that the reader understands the concept of lane.

The lane issue is in many ways central to many of the safety concepts that are used when discussing AVs. For example, it is been proposed that the "Trolley Problem" (Foot 1967) is of no interest because in an emergency situation the default action of the AV will be to brake in lane unless it is unambiguously safe to move out of a lane (Nilsson 2018).

There are at least two issues with this: first, as we are discussing, we need to define what is meant by "lane"; and secondly, and slightly more pedantically, with a fallible sensor set[3] there is probably no case where anything is completely unambiguous.

So, what does brake in lane mean?

In a straight line this is possibly reasonable simple, you brake without deviating from the path that is currently being taken.

On a curve it becomes a little more difficult, in that the path is curved, so the compensating action should ensure that the original intended path should be followed.

This get more complex when one considers what happens if the emergency action is required at the start of, or partway through, a manoeuvre. At the start of a lane change manoeuvre this could mean that the AV should return to the lane that it was exiting. Nearer the end of the manoeuvre it is unclear what it should mean and is highly dependent on what surrounds the vehicle and where danger

[3] That is to say, a sensor set that may be as reliable as is feasible for the technologies involved, but which is nevertheless fallible due to inherent limitations or to the potential failures.

lies. It's possible with a vehicle half-in and half-out of a lane that braking hard puts two lanes of traffic in danger.

As a starting point the idea of "brake in lane" has considerable merit. However, the details are complex and not always immediately obvious, and finding a solution or solutions may involve working through all the scenarios to work out what the best options are. This approach is itself problematic (Koopman et al. 2019).

This highlights what we believe to be a general problem: that there is a tendency to specify behaviours of an autonomous vehicle in terms that a human can understand, using basic concepts such as "lane". The issue is that an AV has no understanding of any concepts aside from what they have been programmed or trained to recognise. Even then we are probably stretching the idea of "understanding".

A two-year-old child when presented with a toy truck can identify it as a truck, but not necessarily as a toy. By the time they are old enough to drive they will usually have had at least 16 years of practice identifying objects in the real world and understanding the relationship between them, e.g. trucks drive on roads, and when trucks drive on roads, they drive in their "lane" on their side of the road. Autonomous vehicles, and the systems and software that control them, may have none of this deep ontological understanding.

3 On speed limits

There seems to be a "consensus" that autonomous vehicles should honour the speed limit and indeed there is perhaps currently little leeway on this matter in many jurisdictions. However, it can be argued that this is not the ideal solution for several reasons.

First, we need define what we mean when we talk about speed. For example, by law it is not permitted for the speedometer of a vehicle to read a value lower than the actual speed of the vehicle. In days of old the speedometer in a passenger vehicle was driven by a Bowden cable from a wheel and mechanically the speedometer was designed so that it always read high by a small percentage.

With the advent of electronic systems, speed over ground is still measured from the rotational speed of a wheel; only now the Bowden cable is replaced by electronic sensors on all the wheels, the information is also used by the brake control systems and the speedometer displays an "average" value.

Why is this a problem? Measuring speed over ground this way is quite difficult as it depends on the nominal radius of the wheel which changes depending on

factors such as tyre wear, inflation state and load on the wheel.[4] For example, on one of the authors' vehicles 70 mph (31.29 m/s) as measured by GPS corresponds to a speedometer display speed of 74 mph (33.08 m/s) in loaded conditions and 73 mph (32.63 m/s) unloaded.

The first issue arises if the vehicle and the driver (or, rather, occupant of the driver's seat) are working to two different values, i.e. the AV to the actual best estimate of speed over ground and the driver to what's displayed on the dashboard. It is possible, albeit probably unlikely, that this could result in unpredictable behaviour.

It is also certain that among a set of vehicles on the road, each will have a slightly different idea of how fast it is going in relation to the speed limit. With this in mind we can examine some scenarios and what the implications might be.

For our first example consider urban streets with a nominal speed limit of 30 mph (13.41 m/s). In this situation a small difference in speeds is most likely not an issue as, in a busy urban environment, the speed limit is often not the primary limit on how fast a vehicle can travel. In less busy environments, for example villages and side streets, it may annoy some drivers that a vehicle keeps to the limit; however, the limit is low for a reason, to protect other vulnerable road users (pedestrians and cyclists) and to keep local noise levels down. In this situation it appears clear that allowing a vehicle to travel faster than the speed limit is not readily defensible on grounds other than the convenience of road users who wish to travel faster than the limit.

There are situations where it may be advantageous to have an AV overtake, for example this would seem defensible when the vehicles in front is slow-moving but in general keeping to the speed limit in unconstrained road situations.[5]

A different situation exists on roads such as motorways where pedestrians and cyclists are not usually present. If we assume that an autonomous vehicle can overtake then we arrive at the same situation in which speed-limited heavy goods vehicles find themselves: it can take a considerable distance for an overtaking manoeuvre to be completed.

Calculations show that if an AV travelling at a speed over ground of 71 mph approaches another car moving at a speed over ground of 70 mph[6], assuming that the gap in-lane between the two vehicles is always at least the minimum recommended braking distance (96 m), then the manoeuvre will take on the order of 7 minutes. This is unlikely to be acceptable for many reasons, including but not limited to the following:

[4] There are of course other significant reasons why measuring speed this way is problematic due to under and over rotation of the wheels because of the surface friction however we ignore those for the purpose of this discussion.

[5] Unconstrained in that pedestrians, cyclists, dogs, cats, horses and tractors etc. can all be present

[6] The difference in speed is 1 mph or 0.45 m/s.

- It would probably be stressful for the AV occupants
- It would be very stressful or annoying for vehicles that found themselves behind the overtaking AV
- It may be uncomfortable for occupants of the vehicle being overtaken
- It would reduce the road capacity, which is a sociological harm

The potential to annoy vehicles finding themselves behind the AV is possibly unsafe (as well as anti-social) in that at the start of the manoeuvre there might be the opportunity and temptation for a driver behind the AV to undertake. This is not only illegal but also dangerous[7] and could result in harm to persons in all three vehicles. Thus, while the AV has been acting in a legal manner and it is not the direct cause of the accident it is not necessarily safe. In the event of an accident occupants of the AV would take scant comfort from the fact that their car had followed the letter of the law with respect to speed limits. That, however is not the end of it, as the law requires that drivers take due care and attention; and it is possible in a trial that the AV could be held to be a contributing cause by a judge.[8]

4 Keeping your distance

It has also been suggested by Shalev-Shwartz et al. (2017) that an AV should be required to maintain a minimum separation from the vehicle directly in front; though it was suggested by the authors that the distance would have to be determined by authorities. The study "Safe Distances Between Vehicles" (Breyer et al. 2010) gives a summary of practice from around the European Union, where the norm appears to be something similar to a time gap of two seconds. Note however that this is not always given as a legal requirement, but rather recommended practice. For example, the legal requirement in the UK is "driving with due care and attention" and in Germany the road traffic act requires drivers to leave a distance that will allow them to stop if the car in front brakes.

Again, we can perform a simple thought experiment. If the recommended distance is a time gap of two seconds and traffic is light, there is probably not an issue. However, if traffic is heavy, (Ayres et al. 2001) indicates that the time gap tends to approach one second, half of what would normally be considered safe. Here an AV could find itself impeded if it attempted to maintain the two second gap simply because other drivers cut in front of it.

[7] One of the authors has a similar situation on the M11 where the person undertaking misjudged where they could move back into the right-hand lane; this required evasive action from the author.

[8] Personal communication, Stephen Mason Oct 2019.

If the AV is also rigorously maintaining the speed limit the follow-on effects are not difficult to envisage: one might expect that vehicles following the AV will attempt to overtake, possibly also cutting in front of the AV. We thus get the situation where the AV has effectively become a mobile chicane which serves to increase the number of vehicle-to-vehicle conflicts (in the widest sense of the term) and hence increases the likelihood of an incident occurring.

There are also possibly secondary effects, e.g. if the AV is compelled to slow each time a vehicle cuts in front then the braking effect may ripple backwards and cause traffic to come to a halt some way behind the AV.

Now we turn our attention to what might be considered safe in terms of a following distance based on braking levels. It is suggested by Shalev-Shwartz et al. (2017) that an AV is "safe" if it is far enough behind the lead vehicle that a collision can be avoided if:

- the lead vehicle applies full braking force,
- for the duration of its response time (i.e. until it recognizes and reacts to the lead vehicle's braking), the AV is accelerating at maximum acceleration, and
- the AV then brakes by at least the minimum reasonable maximum braking force expected of a human driver.

If we ignore the condition with the AV accelerating and assume constant speed, then we have two unknowns: the maximum deceleration at full braking force and what can reasonably be expected as a maximum response from a human driver. What might be reasonable values here?

We can get a partial answer by looking at studies on how human drivers brake "in the wild". One of the first of these (Lechner and Perrin 1993) suggested that normal human driving occupies a bounding circle of +/- 0.3g in longitudinal and lateral acceleration. However, this is a small-scale study and does not have any data for crash avoidance scenarios.

The "100 Car" study (Neale et al. 2005) is much larger and ran over a period of one year. Lee et al. (2007) examined 25% of the data for braking patterns of drivers and found 190,000 usable braking events, of which 160 could be classified as near-crash and 2,713 of which were incidents which required evasive actions from the driver.

If we consider normal driving, the 95^{th} percentile lies at a deceleration of 0.37g. For incidents, the 90^{th} percentile is at 0.68g and for near crashes it is at 0.94g. The latter figure is a little surprising as the usual braking capacity of an "average" road vehicle is usually considered to be around 0.8g (Bosch 2014).

What are the implications? If we assume both vehicles are traveling at identical constant speeds at the start of the scenario, that the AV takes 200ms to perceive and react to the vehicle in front's braking and does not decelerate during that time, and that thereafter it decelerates at 0.37g after a linear ramp of one

second, we can calculate how large a separation is required at 70 mph to avoid a collision at 0.68g and at 0.94g.

Where the lead vehicle brakes at 0.68g, a time gap of 2 seconds (approximately 63m) is inadequate, but a time gap of 3 seconds (95m) appears. At 0.94g, a 3 second time gap is not acceptable but 4 seconds (125m) appears adequate.

The case involving deceleration of the lead vehicle at 0.68g is illustrated by Figure 2, which shows the separation of the vehicles (in meters) over time.

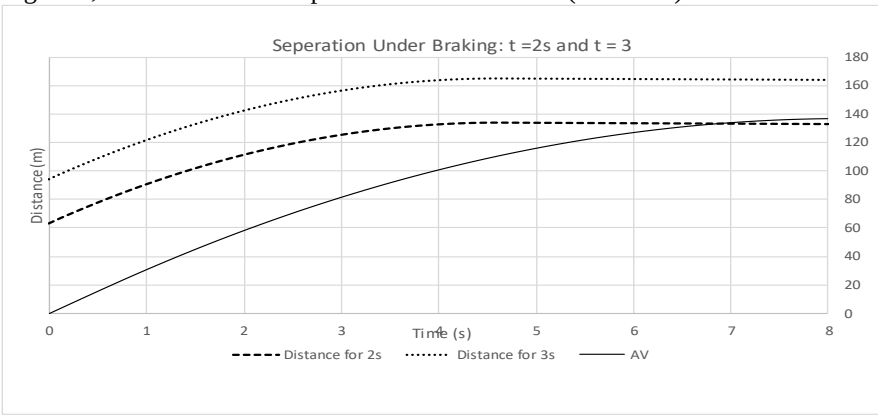


Fig. 2. Separation of AV from the lead vehicle with a starting gap of t = 2 seconds (63m) and t = 3 seconds (9m). Collision can be expected at approximately 6.6 seconds in the first case.

However, as previously described, in heavy traffic the typical time gap approaches 1 second, so it is almost certain that a time gap as large as three or four seconds could not actually be maintained in practice.

However, if the lead vehicle and the AV decelerate only at 0.37g then one second appears to be (just) adequate to avoid a collision given the fast response time of the AV. This would however mean that some crashes will be unavoidable, but the framing of the problem by Shalev-Shwartz et al. (2017) is highly artificial and there is no reason to assume that an AV should in practice be limited to the suggested deceleration limit; a more nuanced approach that more closely mimics the human response of a linear ramp in deceleration followed by a plateau can avoid most collisions (Kusano and Gabler 2011, Markkula et al. 2016).

From the discussion above we deduce that there appears to be no satisfactory simple answer to the question of what an adequate separation distance between vehicles should be, that can take into account both normal traffic flows and the need for emergency action. As noted, this is partly due to the way in which the problem and its solution has been framed. The primary issue is that, at least in the near future, roads will be dominated by human drivers with AVs a distinct minority. Over time this is expected to change as new vehicles are introduced into vehicle fleets, but AV dominance could take a decade or more.

5 Discussion

In this paper we have tried to point out that, in general, there is a multitude of problems associated with defining how AVs should behave in an environment dominated by vehicles that are under the control of the distractible, inattentive, capricious control system equipped with an inadequate set of sensors that is the human animal; an animal that has not evolved to cope with speeds higher than around 10 m/s for appreciable periods of time.

We have pointed out that the use of the term "lane" is burdened with what experienced human drivers understand a lane to be. It is almost certain that there are other terms that are burdened in the same way.

Another example could have been used: the "stop sign" for instance has been investigated in quite some detail. But exactly what does (and does not) constitute a stop sign is not understood either in terms of recognition or, possibly, in law. Humans have an intrinsic understanding of "stop sign" and are reasonably good at recognising one even if up-side-down (e.g. due to a rusted-through bolt). However, there is a stop sign at the roundabout outside our Cambridge offices that is smaller than normal and is around only a meter off the ground – and most drivers appear to be completely oblivious to it (Figure 3).

In a similar way, we have examined a set of extremely well-formed rules formulated by Shalev-Shwartz et al. (2017). The rules are rational and internally self-consistent, but quite possibly prove problematic at the current time. We think that this may be an on-going problem as AVs interact with human-driven vehicles and it is almost certain that situations and interactions will come to light in practice, of which no-one has thought in advance. For this reason we have tried to avoid the trap of suggesting simple modifications to the rules examined.

Fig. 3. Stop sign at the roundabout at the Cambridge Science Park entrance from Kings Hedges Road, Cambridge, October 2019

As a final note, the reader should be aware that the authors are as guilty as anyone of misusing terms such as "lane" and defining behaviour in terms that are simply comprehensible to experienced humans.

Acknowledgments The authors would like to thanks to Ahmed Meza for providing the dash-cam footage and for posing at the stop sign.

Disclaimers The work presented in this paper represents a snapshot of thoughts that the authors have about how rules for autonomous vehicles need to be formulated and does not reflect any official position of Ricardo UK Ltd. or other companies in the Ricardo group.

References

Ayres, T. J., Li, L., Schleuning, D., & Young, D. (2001, August). Preferred time-headway of highway drivers. In ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585) (pp. 826-829). IEEE.

Bosch (2014) Automotive Handbook.

Breyer, G (2010) Safe Distance Between Vehicles, Conference of European Directors of Roads.

Committee (2019) Safety First for Automated Driving, https://www.daimler.com/innovation/case/autonomous/safety-first-for-automated-driving-2.html accessed 30 Oct 2019.

Enterprise Singapore (2019) TR 69 Technical Reference Autonomous Vehicles – Part 1: Basic Behavior.

Foot, Philippa (1967). The problem of abortion and the doctrine of double effect. _Oxford Review_ 5:5-15.

Highway Code. http://www.highwaycodeuk.co.uk/uploads/3/2/9/2/3292309/the-official-highway-code-with-annexes-uk-en-12-04.pdf accessed 30 Oct 2019.

Koopman, P. Aaron Kane, A. Black, J. (2019) Credible Autonomy Safety Argumentation, Proc. Safety-Critical Systems Symposium 2019.

Kusano, K.D. and Gabler, H. (2011) "Method for Estimating Time to Collision at Braking in Real-World, Lead Vehicle Stopped Rear-End Crashes for Use in Pre-Crash System Design", doi:10.4271/2011-01-0576

Lambert, F. (2019) https://electrek.co/2019/05/20/tesla-fly-guard-rail-funny-balancing-pack/ accessed 21 May 2019.

Lechner, D., & Perrin, C. (1993). The actual use of the dynamic performances of vehicles. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 207(4), 249-256.

Lee, S. E., Llaneras, E., Klauer, S., & Sudweeks, J. (2007). Analyses of rear-end crashes and near-crashes in the 100-car naturalistic driving study to support rear-signaling countermeasure development. DOT HS, 810, 846.

McGinn, C. (1989). Can We Solve the Mind--Body Problem?. Mind, 98(391), 349-366

Markkula, G. Engstrom, J. Lodin, J. Bargman, J. Trent, V, (2016) A farewell to brake reaction times? Kinematics-dependent brake response in naturalistic rear-end emergencies, Accident Analysis & Prevention, Vol. 95, Part A, October 2016, Pages 209-226.

Neale, V. L., Dingus, T. A., Klauer, S. G., Sudweeks, J., & Goodman, M. (2005). An overview of the 100-car naturalistic study and findings. National Highway Traffic Safety Administration, Paper, 5, 0400.

Nilsson, J. (2018) Safe Self-driving Cars: Challenges and Some Solutions, Safety-Critical Systems Symposium 2018, (in presentation).

NTSB (2018) Preliminary Report Highway HWY18FH011, https://www.ntsb.gov/investigations/AccidentReports/Reports/HWY18FH011-preliminary.pdf access 30 Oct 2019.

Shalev-Shwartz, Shai, Shaked Shammah, and Amnon Shashua (2017) On a formal model of safe and scalable self-driving cars. arXiv preprint arXiv:1708.06374 (2017).

Typical Stopping Distances. https://assets.publishing.service.gov.uk/media/559afb11ed915d1595000017/the-highway-code-typical-stopping-distances.pdf accessed 30 Oct

Generating the Evidence Necessary to Support Machine Learning Safety Claims

James McCloskey[1], Rose Gambon[1], Chris Allsopp[1], Thom Kirwan-Evans[1], Richard Maguire[2]

Frazer-Nash Consultancy[1], Defence Science and Technology Laboratory (Dstl)[2]

Abstract *Machine Learning is making rapid progress in a variety of applications. It is highly likely to be used in safety-related and possibly safety-critical systems. As a logical next step to work presented at the Safety-Critical Systems Symposium 2019 on developing a safety argument structure for an autonomous system that uses machine learning, this paper focuses on generating the underpinning safety evidence. This is achieved through the representation of the machine-learnt software development life-cycle as a model which articulates constituent artefacts, information flow and transformations. This life-cycle model is then used to facilitate the systematic identification of the potential for the introduction of hazardous errors during development. Product, process and goal-based control measures are proposed to reduce and manage these potential errors. The feasibility and practicality of implementing these control measures and generating associated safety evidence is also discussed.*

1 Introduction

1.1 Overview

Artificial Intelligence (AI), including approaches such as Machine Learning (ML), is making rapid progress in a variety of applications. It is highly likely to be used in safety-related systems. This means there is a need to consider how to make safety arguments for systems that exploit ML techniques; and for the Autonomous Systems (AS) that make use of them.

© Crown copyright (2019), Dstl. Parts of this material are licensed under the terms of the Open Government Licence except where otherwise stated. To view this licence, visit http://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/ or write to the Information Policy Team, The National Archives, Kew, London TW9 4DU, or email: psi@nationalarchives.gsi.gov.uk.
Published by the Safety-Critical Systems Club. All Rights Reserved

Any safety argument must include evidence and this paper focusses on the practicality and feasibility of generating the evidence necessary to demonstrate the safety of an AS that uses ML. This paper:

- Summarises a generic safety argument structure for an AS that uses ML (see section 2). The work to develop this safety argument was presented at the Safety-Critical Systems Symposium 2019 (McCloskey et al. 2019).
- Discusses each of the key aspects of the safety argument and provides examples of the evidence types necessary to support the safety goals (see sections 3 – 8); and,
- Provides the conclusions of the work (section 9).

1.2 Project Background

As part of a Dstl funded project, Frazer-Nash Consultancy Ltd (Frazer-Nash), in partnership with industry and academia, developed a generic safety argument structure for an AS that uses ML. Table 1 lists the industry and academia partners that supported the project.

Table 1. The industry and academia partners supporting the project

Partner	Area of Specialism on the Project
SeeByte Ltd	Integration of Autonomy and Artificial Intelligence on maritime Platform
Bristol Robotics Laboratory	Embodied Artificial Intelligence within Autonomous Air Systems
Montvieux Ltd	Artificial Intelligence and Machine Learning Techniques
University of the West of England	Artificial Intelligence and Machine Learning Techniques
Ricardo Ltd	Autonomous Systems
University of Bristol	Functional Verification and Validation for Safety
University of York	Safety and Autonomy

2 The Safety Argument

2.1 The Nature of the Safety Argument

The nature of the safety argument is similar to that of a traditional product design safety case. It argues that:

- The design is sufficiently free from failure modes that contribute to a hazard for its given operating context; and,
- Arrangements are in place for the safe operation of the AS, such as human machine interface considerations and establishing both the operating limitations and user information necessary for safe use.

The safety argument has been developed to be platform agnostic. It focuses on the key claims and underpinning evidence necessary to demonstrate the safety of a system that has autonomous capabilities that are supported by ML functionality. The scope of the argument is limited to a system that has learnt offline – that is the ML aspects of the system only changes its parameters (i.e. learns) when the system is not in use. This may be prior to deployment, or periodically after use (i.e. recalibration). The argument may be applied across the range of autonomy levels. Online learning and optimisation were specifically excluded in this project to simplify the development of the safety argument in a nascent field. The inclusion of operational learning will require further research.

2.1.1 ML Techniques that are Within Scope

There is no single, coherent definition of what ML is or what exactly it encompasses. ML is a broad term that can be summarised as a mathematical model whose parameters are optimized by a computer to map an input to an output. Table 2 gives the ML techniques that are within the scope of the safety argument along with corresponding descriptions used by the partners within the consortium.

Table 2. The ML techniques that are within the scope of the safety argument

Technique	Description
Supervised Learning	In supervised learning, the ML model is provided with input data and labelled 'true' output data from which to 'score' its predicted output. The ML model is then optimized against this set of labelled data (e.g. classification systems).
Unsupervised Learning	In unsupervised learning, the ML model does not know what the 'true' desired output is and makes its best estimate according to internal rules (e.g. clustering systems).
Reinforcement Learning	Reinforcement learning is, essentially, a ML model that interacts with an environment. It reads inputs, generates outputs, acts on them, and includes the outcome of its actions in the optimization process (e.g. DeepMind's Go playing system). The environment is often simulated during the training phase.
Deep Learning	A general term to capture neural networks with many layers. Deep learning models typically have millions of tunable parameters, and cover other techniques such as Convolutional Neural Networks, Recurrent Neural Networks, Long Short Term Memory, and Generative Adversarial Networks.

2.2 A 'Hazard-Centric' Safety Argument

The safety argument is 'hazard-centric'. That is, the core of the argument involves demonstrating that each system hazard is acceptably managed. This is achieved by the imposition and satisfaction of Safety Requirements (SR) that manage the identified risks. This hazard-centric approach is also encapsulated in the '4+1' software safety principles (Hawkins et al. 2013) and already has corresponding safety argument patterns for traditional software (Hawkins and Kelly, 2013).

Additionally, the hazard-centric approach is consistent with previous work undertaken by Dstl (Ashmore and Lennon, 2016) which proposes a version of '4+1' principles which could be applied to ML (see Table 3).

Table 3. The 4+1 principles applicable to ML (Ashmore and Lennon, 2016)

Principle	Description
P1	ML Software (MLSW) Safety Requirements (SR) shall be defined to address the software contribution to system hazards.
P2	The MLSW detailed design shall embody the intent of the software SR.
P3	MLSW SR shall be satisfied.
P4	Hazardous behaviour of the MLSW shall be identified and mitigated.

Principle	Description
P4+1	The confidence established in addressing the MLSW safety principles shall be commensurate to the contribution of the software to system risk.

2.3 Summary of the Safety Argument

Figure 1 provides an overview of the safety argument developed by the Frazer-Nash led consortium. The core of the argument is that the AS is safe because each design or operating hazard is acceptably managed by addressing MLSW contributions to system hazards through the imposition and satisfaction of MLSW Safety Requirements (SR) that manage the risk.

In support of the core argument it is necessary to demonstrate that:

- The system hazards and their contributors are sufficiently identified;
- The intent of each MLSW SR is maintained during decomposition; and,
- The potential for hazardous design or implementation errors that could be introduced during the MLSW development process are managed.

Figure 1 also cross-refers to the corresponding sections of the paper where the elements of the safety argument are discussed in more detail.

3 The Top Level Argument

The Top Level argument for the AS argues that the system is sufficiently safe for operation in its environment because system hazards are:

- Sufficiently identified (see section 4); and,
- Each hazard is acceptably managed. It is argued that each hazard is acceptably managed because its contributors are in turn sufficiently identified and acceptably managed (see section 5).

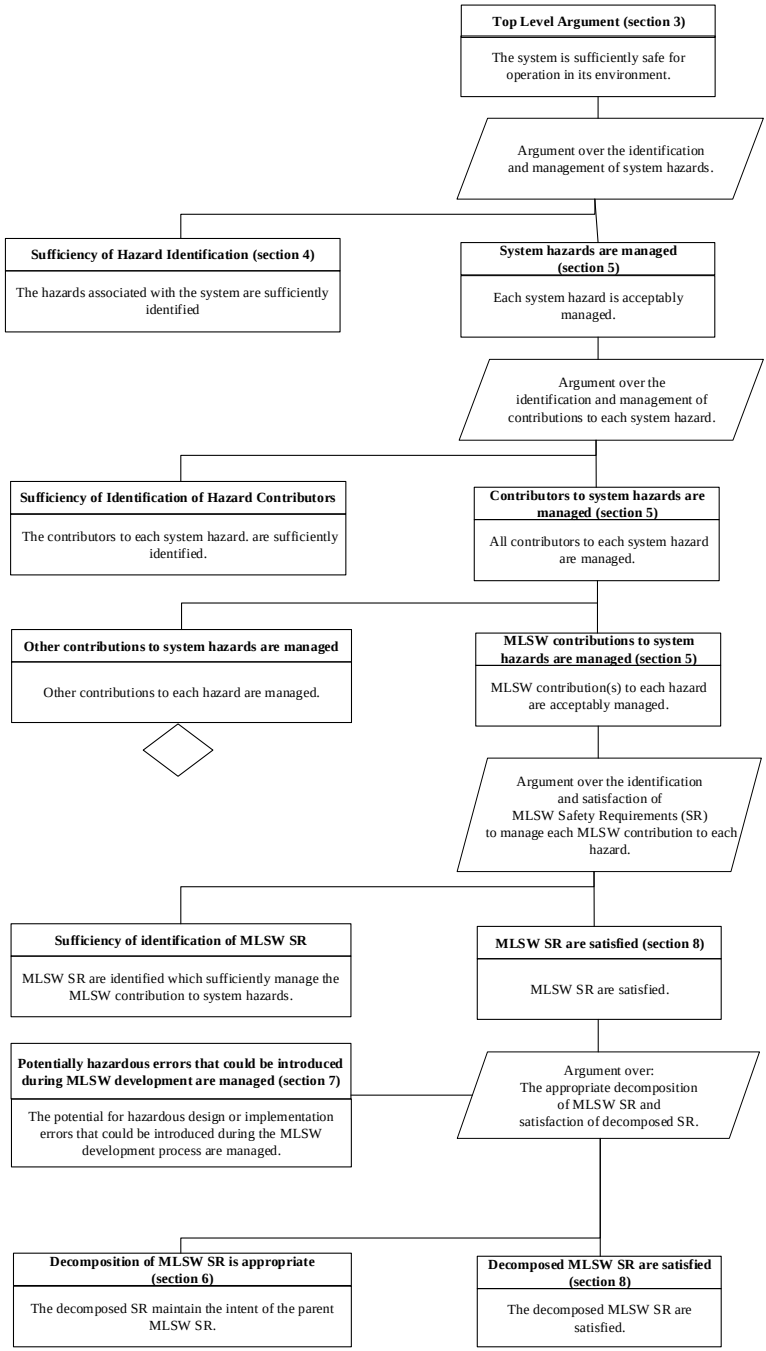


Fig. 1. A more detailed summary of the safety argument

4 Arguing Sufficiency of Hazard Identification

The argument that system hazards are sufficiently identified is based on:

- Application of appropriate techniques to identify different types of hazard (e.g. physical hazards, functional failure hazards, data flow hazards, misuse hazards, hazards arising from intended functionality and emergent hazards);
- The use of diverse hazard identification techniques for each hazard type; and:
- Independent assessment of the identification process and/or hazard logs.

4.1 Evidence of Sufficiency of Hazard Identification

Demonstrating completeness of hazard identification is challenging, in particular due to the emergent behaviours that result from complex interactions present in AS. Table 4 summarises evidence that can be used to support sufficiency of hazard identification.

Table 4. Supporting evidence for sufficiency of hazard identification

Solution	Example of Evidence	Discussion
Evidence of hazard identification. Evidence that hazard identification is sufficient. Evidence that diverse techniques have been applied.	Use of a diverse and complementary range of techniques e.g: • Energy Trace Barrier Analysis (Leveson, 1995); • Scenario-based Functional Failure Analysis (Alexander et al. 2009); • Hazard and Operability Study (Alexander et al. 2009); • Modelling and simulation; • Systems Theoretical Process Analysis (Leveson and Thomas, 2013); • Safety Of The Intended Functionality Analysis (The British Standards Institution, 2019); • Responsibility Analysis (Lock et al. 2010); and: • Functional Resource Analysis Method (Hollnagel, 2017).	AS that interact with the environment are presented with an infinite set of scenarios. Because the relative immaturity of AS, there is limited in-service experience to draw upon. As such, it is not possible to prove that a list of the identified hazards is comprehensive. Whilst this is the case for most systems, for AS it is presently not possible to estimate the potential gap in the comprehensiveness.
Independent assessment of hazard identification sufficiency.	Independence is achieved when the review activity is performed by person(s) outside of the hazard identification activity.	It is necessary to demonstrate that independent assessors are suitably independent and qualified and experienced to undertake the activity.

5 Managing MLSW Contributions to Hazards

As mentioned in the Section 3, it is argued that each system hazard is acceptably managed because:

- Its contributors are sufficiently identified. Contributions to a system hazard can be identified through techniques such as fault tree analysis or failure mode and effects analyses; and,
- Each MLSW contribution[1] to the system hazard is acceptably managed through the stipulation and satisfaction of appropriate MLSW SR. Such MLSW SR could, for example, specify requirements for the elimination, reduction in probability of occurrence or handling of the specific MLSW contribution to the hazard (Weaver et al. 2002).

5.1 Evidence of MLSW SR

It is noteworthy that specifying MLSW SR for AS can be challenging because advanced functionality (e.g. perception/interaction with the environment) may not be completely specifiable (Salay and Czarnecki, 2018). In particular, identifying the complete range of exceptional circumstances and identifying appropriate behaviours may not be possible (Koopman and Wagner, 2016). Table 5 gives some example approaches that can be used to in response to this issue.

[1] As the safety argument structure focuses on the safety of an AS that uses ML, a differentiation is made between contributor to hazards which are due to MLSW and those which are due other causes. The safety argument goes on to focus on demonstrating that each MLSW contribution to each identified hazard is acceptably managed.

Table 5. Types of evidence sources that can be used in specifying advance functionality

Solution	Example of Evidence	Discussion
Evidence of MLSW SR which may manage the MLSW contribution to system hazards	Orthogonal SR (Koopman and Wagner, 2016) e.g. a safety envelope approach.	Enforcing a safety envelope through a separate mechanism, for example through an independent, deterministic monitor architecture, reduces the demand placed on the MLSW. For instance, training MLSW to learn a safety envelope such that it exhibits a step change in its behaviour at the edges of the envelope may result in performance trade-offs elsewhere. In practice the safety envelope is usually less permissive than the ML/autonomy i.e. it under-approximates the safe state space of the system in exchange for simplifying computation (Koopman et al. 2019).
	Use of a partial specification (Salay and Czarnecki, 2018). Constraints that form a partial specification may include: • Pre-conditions; • Post conditions; • Invariants (e.g. an object of class x in an image will still be classified as x if translated to a different part of the image); • Equivariants (e.g. a rotation of an input image will result in the same rotation of the output image); • Probabilistic constraints (e.g. an object of class x has dimensions of probability distribution y. Unexpected classifications, while not necessarily incorrect, can be flagged for further analysis); and, • Contextual constraints (e.g. cars must be within x metres of the road, not flying in the sky).	While the use of partial specifications may not provide guarantees of the full behaviour of the ML-based system, the properties and constraints we can define can influence the development of ML software and support both verification and run-time monitoring of the system. Whilst the partial specifications define the necessary conditions for MLSW behaviour, they may not define sufficient conditions (Salay and Czarnecki, 2018). For AS that interact with the environment, a key issue is how to discretize the real world, justifying where the boundaries are, and that there are no gaps between the boundaries.

6 Decomposition of MLSW SR

This section of the safety argument involves demonstrating that the MLSW SR are adequately allocated, decomposed, apportioned and interpreted into lower level requirements. Here it is recognized that:

- MLSW requirements (including MLSW SR) are implemented through the combination of inter-dependent elements such as training data, learning algorithm, model, training process to be followed, and the application of evaluation criteria (see section 6.1);
- The decomposition of MLSW SR goes 'hand-in-hand' with design decisions or implementation choices. That is, design decisions/implementation choices will give rise to further detailed requirements which will influence further design decisions/implementation choices (Hawkins et al. 2010), (Aravantinos and Diehl, 2019)[2]; and,
- MLSW development (and hence decomposition of requirements) occur in an incremental manner through a series of steps or 'tiers' (see section 6.2).

6.1 The Inter-Dependent Elements in MLSW Development

The ML development process is complex; by definition, it involves trial and error with many feedback loops. Broadly, the ML development process relies heavily on developer experience and can be split into the following three phases:

- The specification, development and/or selection of the types of artefacts and processes which are capable of satisfying the requirement;
- Undertaking iterations of training, evaluation and refinement using the artefacts and processes until a particular instance of the combination of artefacts and processes is achieved that results in code which satisfies the requirement[3]; and,
- Verification that the code satisfies the requirement.

Artefacts include items such as the dataset, data labels, the architecture (for a Deep Neural Network (DNN)), the loss function, learning parameters and search strategies, etc. Processes include documented processes for developing artefacts,

[2] (Aravantinos and Diehl, 2019) *'In general, refining High Level Requirements into Low Level Requirements goes hand in hand with architectural decisions: the requirements can be decomposed only once the function is decomposed into smaller functions, to which one can assign more concrete requirements. This is why the DO-178C, for instance, refines the HLR into two artefacts: the LLRs on one hand, and the Software Architecture on the other hand.'*

[3] It is noteworthy that, for MLSW, design decisions may be informed by experimentation or trial and error (Aravantinos and Diehl, 2019).

executing steps in the MLSW development process, making design decisions, etc.

Following an approach which aligns with the ISO 42010, Systems and software engineering — Architecture description (ISO, 2011), each MLSW SR can be decomposed into a series of inter-dependent requirements (Douthwaite and Kelly, 2018). Therefore, by selecting to use MLSW to implement requirements, the requirements will be realised through the combination of inter-dependent viewpoints, including: training data, learning algorithm, model; a training process to be followed and the application of evaluation criteria. Figure 2 presents the ML development process, including the inter-dependent viewpoints: Data, Algorithm, Model, Evaluation Criteria and Training Process Requirements.

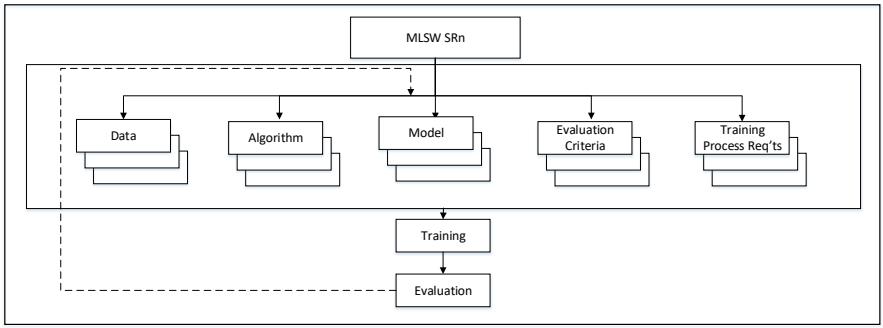


Fig. 2. MLSW development process including inter-dependent viewpoints

Each viewpoint will contain a series of more detailed 'concerns' to be taken into consideration when developing MLSW. For example: bias, sufficiency and accuracy are all concerns within the data viewpoint. Figure 3 illustrates how a MLSW SR would be decomposed into concern-based requirements. Concern SR are represented by the smaller boxes in Figure 3. The dashed arrows between the Concern SR are intended to show the inter-dependent nature of the concerns SR. That is each viewpoint cannot be viewed as a standalone silo. Therefore, concern SR should not be decomposed independently from the other viewpoints.

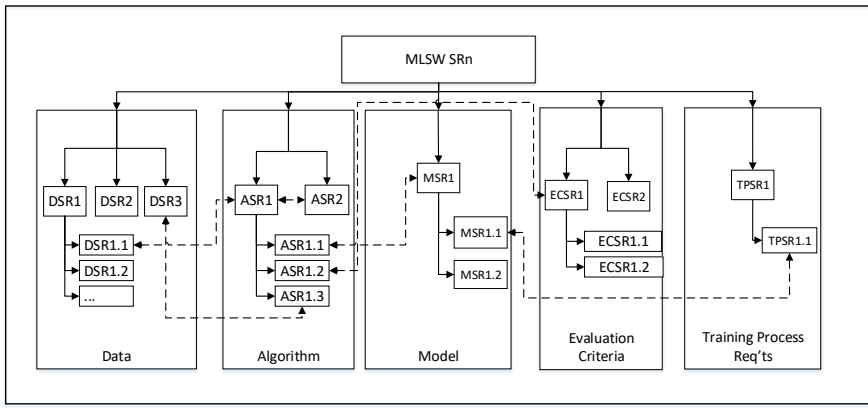


Fig. 3. In ML, a MLSW SR is satisfied through the interaction of a series of inter-dependent, decomposed concern-based requirements

6.2 Tiered Development of MLSW

The MLSW development process can be broken down into a series of incremental steps. Within the safety argument, each increment is described as a tier by the Frazer-Nash led consortium. A tier can involve one of the following activities:

- Decomposing a SR from the preceding tier;
- Making a design decision (including a design decision to undertake trial and error experiments);
- Implementing a design decision; and/or,
- Training, evaluating or refining the algorithm/model.

The input to each tier consists of a MLSW SR or concern SR along with the design commitments to date. The output from each tier can be artefacts for use in the MLSW develop process, design commitments or further decomposed requirements. This continues until all concern SR are addressed through design decisions and implementation. At this point the MLSW will be ready for verification.

The safety argument provides a structure to argue that the MLSW SR from the current tier have been adequately allocated decomposed, apportioned and interpreted at the next tier. This involves demonstrating that:

- The decomposed SRs are equivalent to their parent MLSW SR;
- There is traceability between SRs across tiers;
- The decomposed SRs are of appropriate quality (unambiguous, verifiable, consistent with other requirements and necessary for the requirement set); and,
- The design decisions taken and their implementation are appropriate to ensure that the original intent of the parent MLSW SR is maintained.

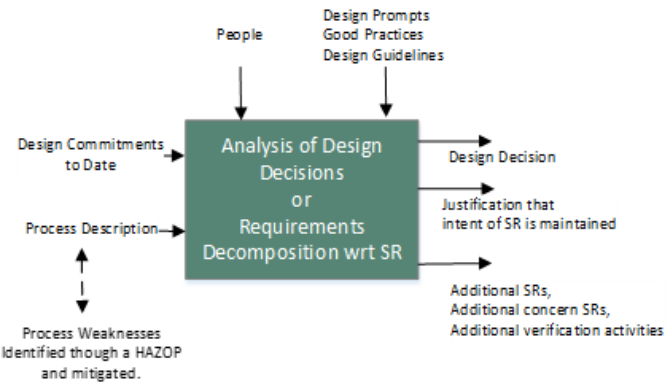
6.3 Taking Verification Requirements into Account when Decomposing SR

Achievable levels of verification for MLSW are affected by the interpretability of the MLSW (Salay and and Czarnecki, 2018). ML models contain knowledge in an encoded form, but this encoding is more interpretable by humans in some types of models than others. Taking into account the levels of verification required during the Concern SR decomposition and when making design decisions will support achieving the required levels of confidence in satisfaction claims.

6.4 Evidence that the Intent of the MLSW SR is Maintained During Decomposition.

Table 6 gives examples of evidence that can be used to support an argument that the intent of the parent MLSW SR is maintained during decomposition.

Table 6. Types of evidence sources that can be used in supporting MLSW SR decomposition claims

Solution	Example of Evidence	Discussion
Evidence that the intent of a MLSW SR is maintained during decomposition and design decisions	• Evidence that a systematic process was followed when the MLSW SR is decomposed. The diagram below presents a process model developed by the consortium for the decomposition of a MLSW SR or review of MLSW development design decisions.  • Evidence that, at each step of the process, reviews were undertaken: – To ensure the intent of the MLSW SR is maintained; and, – If possible, formal analysis of Concern SR decomposition.	• Manual reviews involve reliance on humans which introduces the potential for human 'bias'. • Manual reviews are subjective and not repeatable. • It may not always be possible to apply analysis methods (e.g. formal specifications) to the decomposition of MLSW SR or concern SR.
Traceability	Using a network or ontology to: • Provide traceability between the detailed MLSW SR, concern SR and/or design decisions; and, • Identify sources of potential undesired behaviour (by highlighting requirements/design decisions/artefacts that cannot be traced to a higher-level requirement).	The development of an ontology and identification of inter-relationships between MLSW concerns will involve manual review. Such reviews are subjective and not repeatable. There will need to be sufficient confidence that any tools used to record or manage inter-relationships are appropriate.

7 Managing Development Hazards

Although the MLSW SR placed on a design can capture the intent of the high-level SR, this cannot guarantee that the requirements have taken account of all the potentially hazardous ways in which the MLSW might behave (Hawkins et al. 2013). There will often be unintended hazardous behaviour, resulting from the way in which the MLSW has been developed, which could not be appreciated through simple requirements decomposition. These hazardous MLSW behaviours could result from either:

- Unanticipated behaviours and interactions arising from MLSW design decisions; or,
- Systematic errors introduced during the MLSW development.

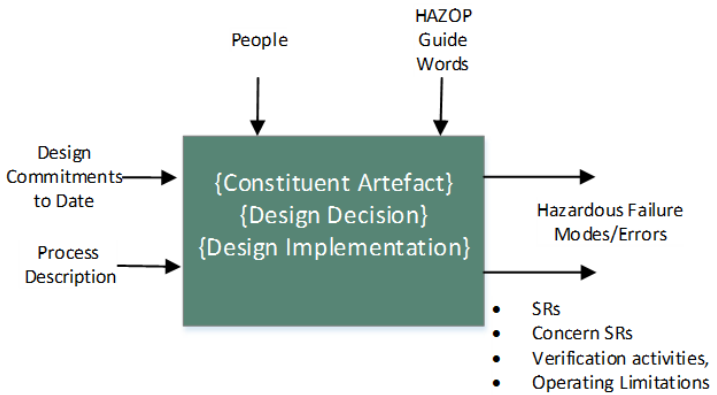
The safety argument provides a structure to demonstrate that potential hazardous failures that could be introduced during development are acceptably managed. It does so by arguing that:

- Potentially hazardous design errors are not introduced during development. This is demonstrated by arguing that:
  - The potential for the introduction of hazardous errors due to the development process has been identified through a hazard and operability study of the development process and rectifying measures implemented; and,
  - Each development artefact used in the MLSW development process has been reviewed or analysed and identified hazardous errors rectified.
- Hazardous MLSW failure modes that could result from design or implementation decisions are:
  - Identified through review of or experimentation with each design decision; and,
  - Addressed through the implementation of concern SR.

7.1 Evidence that Development Hazards are Managed

Table 7 gives examples of evidence that can be used to support an argument that the potential for the introduction of hazardous errors during the MLSW development process is managed.

Table 7. Examples of evidence that can be used to support an argument that the potential for the introduction of hazardous errors during the MLSW development process is managed

Solution	Example of Evidence	Discussion
MLSW constituent artefacts are sufficiently free from hazardous errors.	• Review or analysis of the MLSW constituent artefacts. The diagram below gives an example simple process model developed by the consortium for the identification of the potential of design decisions and implementation to introduce hazardous errors.  • As with traditional safety related software development, static analysis techniques can be applied with the aim of detecting implementation errors or deviations from expected software quality standards and dynamic analysis techniques can be applied to detect a subset of MLSW implementation errors.	These techniques may identify issues with the implementation of ML algorithms or concern SR but will not be able to highlight deficiencies with respect to high level functional requirements as there is not necessarily a transparent and traceable implementation in code.

Solution	Example of Evidence	Discussion
Hazardous MLSW Failure Modes that could result from design or implementation decisions are managed.	• Evidence that a systematic process is followed when the MLSW SR are disaggregated into Concern SR. • Evidence that a process is robust and systematic can be provided by demonstrating that the process has been assessed for potential weaknesses and those weaknesses addressed. (Hawkins and Kelly, 2010) propose a means of identifying potential weakness in a process by (see row above): – Representing the process as a simple model (with inputs and resources); and, – Considering the effect of potential deviations from process (in the form of HAZOP-like guidewords to each input and resource identified in the simple model). • Evidence that at each step of the process reviews were undertaken (Hawkins and Kelly, 2010) to identify and manage unintentional hazardous errors.	Weaknesses include: • Reliance on humans – introduces the potential for human 'bias'; • Manual reviews are subjective and not repeatable; and, • Analysis methods may or may not be applicable to the decomposition of concern SR. MLSW/concern-based errors may be several layers removed from the MLSW behaviour, so determining the effect of these errors can be difficult. Lack of transparency and traceability may reduce the effectiveness of implementation reviews.

8 Verifying that MLSW SR are Satisfied

Having demonstrated that MLSW SR are adequately allocated, decomposed, apportioned and interpreted into lower level concern SR, it is necessary to demonstrate that they are satisfied. This section of the new safety argument structure developed by the Frazer-Nash consortium does so by arguing that:

- Test results demonstrate that concern SR is satisfied;
- Analyses demonstrate that concern SR is satisfied; and,
- Reviews demonstrate that concern SR is satisfied.

8.1 Evidence that MLSW SR are Satisfied

Table 8 gives examples of evidence that can be used to support an argument that the concern SR are satisfied.

Table 8. Examples of evidence that can be used to support an argument that the MLSW SR are satisfied

Solution	Example of Evidence	Discussion
Test results	Test cases demonstrate that each test case results in output sufficient to demonstrate the achievement of the SR.	Potential weaknesses include: • Test cases may not be sufficient to trigger all possible outputs. • There is no recognised consensus on the verification techniques required for MLSW of different levels of safety-criticality. • Non-deterministic test scenarios combined with non-deterministic system behaviours and opaque system designs means it is difficult to know if a system has passed a test, because there is no single correct answer (Koopman et al. 2019). • Because of the inductive nature of ML, where training examples are used to derive a model, it may not be possible/practical to deduce that a test has been passed for valid reasons. • The complex problems that ML techniques are applied to often involve large numbers of inputs, leading to high dimensionality of the state space.
Analysis results (including formal analyses).	Verification by analysis can be used to demonstrate that input will always result in expected output.	Potential weaknesses include: • Analysis cases are reliant upon the accuracy of model and hardware assumptions. Non-formal methods may not be repeatable. • Lack of interpretability of MLSW adversely affects the ability to verify by analysis. • Scalability is a limiting factor for many analysis techniques, such that their application may not be practical. • There is no recognised consensus on the verification techniques required for MLSW of different levels of safety-criticality.
Results of reviews.	Review cases check that errors which affect the achievement of the requirement are not made in the design/code.	Potential weaknesses include: • Reviews cannot directly demonstrate achievement of the requirement. Reviews are subjective and not repeatable. • Lack of interpretability of MLSW adversely affects the ability to verify by review. • There is no recognised consensus on the verification techniques required for MLSW of different levels of safety-criticality.

8.2 Arguing Sufficiency of Verification

To put forward a comprehensive argument, it is also necessary to argue that tests, analysis and review activities are sufficient and appropriate. Table 9 gives examples of evidence that can be used to support an argument that the MLSW verification activities are sufficient and appropriate.

Table 9. Examples of evidence that can be used to support an argument that the MLSW verification activities are sufficient and appropriate

Solution	Example of Evidence	Discussion
Sufficiency of Verification Activities	• Manual justification of the sufficiency of verification activities to demonstrate that each Concern SR is satisfied to an adequate level of confidence. This will involve case by case justification of the selected coverage metrics cognisant of their limitations when applied to MLSW. Example metrics include: – Modified Condition/ Decision Coverage (RTCA and EUROCAE, 2012). – Parameter value coverage[4]. – Multiple condition coverage (RTCA and EUROCAE, 2012). • Manual justification of the extent of verification activities with respect to: – Brittleness. – Corner Cases and Edge Cases. – Coverage of the Input Domain. • Establishing Confidence during testing that the Behaviour of the System is consistent. • Manual justification of coverage of known limitations associated with Design Decisions (e.g. to verify against known limitations of a selected ML model).	• Justification may rely on judgement and experience and therefore may be subjective and not repeatable. • It is difficult to determine the efficacy of these coverage metrics applied to MLSW and therefore it will be difficult to justify their usage. • There is no consensus as to what constitutes sufficiency.

[4] Unit Testing with Parameter Value Coverage (PVC) - Parameter Value Coverage (PVC) is the ability to track coverage of a method based on the common possible values for the parameters accepted by the method (https://www.rhyous.com/2012/05/08/unit-testing-with-parameter-value-coverage-pvc/)

9 Conclusions

This paper summarises the key elements of a 'hazard-centric' safety argument for an AS that uses ML which has learnt offline and gives examples of the types of evidence that may be necessary to substantiate the argument. This is achieved through the representation of the machine learnt software development life-cycle as a model which articulates constituent artefacts, information flow and transformations. Product, process and goal-based control measures have been proposed to reduce and manage these potential errors. The feasibility and practicality of implementing these control measures and generating associated safety evidence was also discussed. The key conclusions are:

- Demonstrating completeness of hazard identification is challenging, in particular due to the emergent behaviours that result from complex interactions present in AS. To address this challenge, a diverse range of hazard identification techniques is proposed. As yet no trials have been conducted.
- Specifying MLSW SR for AS can be challenging because advanced functionality may not be completely specifiable. In particular, identifying the complete range of exceptional circumstances and identifying appropriate behaviours may not be possible (Salay and Czarnecki, 2018). The use of orthogonal safety requirements and partial specifications may be helpful in overcoming the challenges that are presented in attempting to specify autonomous functionality.
- MLSW SR will be implemented through the combination of inter-dependent elements or 'concerns'. MLSW development and decomposition of requirements occur in an incremental manner through a series of steps or 'tiers'. It is not possible to use analysis alone to prove that equivalence is maintained when a MLSW SR is decomposed into a series of requirements applicable to ML constituent concerns. Rather, developer experience is used to:
  - Derive a series of inter-dependent requirements applicable to the constituent artefacts that are judged to be capable of meeting the parent requirement; and,
  - Undertake iterations of training, evaluation and refinement using the artefacts and processes until a particular instance is achieved that satisfies the parent requirement.
- The application of systematic reviews and justifications of each design decision or requirements decomposition step in the MLSW process is recommended to substantiate claims that:
  - The intent of each MLSW SR is maintained; and,
  - The potential introduction of unintentional hazardous errors are identified and managed;

- The use of analysis techniques for the verification of MLSW may not be possible and may not provide the same level of confidence as can be achieved in the verification of traditional software. This is due to:
  - The challenges with interpreting and explaining MLSW; and,
  - Scalability issues.

References

Alexander R, Kelly T, Herbert N (2009) Deriving Safety Requirements for Autonomous Systems. University of York

Aravantinos V, Diehl F (2019) Traceability of Deep Neural Networks. Fortiss GmbH arXiv:1812.06744

Ashmore R, Lennon E (2016) Progress towards the Assurance of Non-Traditional Software. Defence Science and Technology Laboratory (Dstl). Safety-Critical Systems Club rp135.3:1

The British Standards Institution (2019) International Standards Organization/Publically Available Standard 21448:2019; Road vehicles — Safety of the intended functionality; first edition 2019-01.

Douthwaite M, Kelly T (2018) Safety-Critical Software and Safety-Critical Artificial Intelligence: Integrating New Practices and New Safety Concerns for AI systems. University of York. Safety-Critical Systems Club rp140.6:1

Hawkins R, Kelly T (2010) A Structured Approach to Selecting and Justifying Software Safety Evidence. The University of York

Hawkins R, Kelly T (2013) A Software Safety Argument Pattern Catalogue. The University of York, YCS-2013-482

Hawkins R, Habli I, Kelly T (2013) The Principles of Software Safety Assurance. The University of York

Hollnagel, E (2017) FRAM: The Functional Resonance Analysis Method - Modelling Complex Socio-technical Systems, 1st Edition, ISBN 9781351935968, CRC Press, London.

Holloway M, Graydon P (2018) Explicate ’78: Assurance Case Applicability to Digital Systems. Federal Aviation Administration, DOT/FAA/TC-17/67

ISO/IEC/IEEE 42010:2011 (2011) Systems and Software Engineering - Architecture Description. ISO

Koopman P, Kane A, Black J (2019) Credible Autonomy Safety Argumentation. Carnegie Mellon University. Safety-Critical Systems Club rp150.22:1

Koopman P, Wagner M (2016) Challenges in Autonomous Vehicle Testing and Validation. Carnegie Mellon University

Leveson N (1995) Safeware: System Safety and Computers. ACM Press New York, NY, USA.

Leveson N (2004) A New Accident Model for Engineering Safer Systems. Massachusetts Institute of Technology

Leveson N, Thomas J (2013) An STPA Primer. http://sunnyday.mit.edu/STPA-Primer-v0.pdf

Lock, R, Storer, T, Sommerville I (2010) Responsibility Modelling for Risk Analysis. CRC Press

McCloskey J, Allsopp C, Smith D, Lennon E, Jenkins S, Ramsay L (2019) Towards a Safety Argument for Autonomous Systems that Use Machine Learning. Safety-Critical Systems Club rp150.20:1

RTCA and EUROCAE (2012) RTCA DO-178C: Software Considerations in Airborne Systems and Equipment Certification. RTCA and EUROCAE

Salay R, Czarnecki K (2018) Using Machine Learning Safely in Automotive Software: An Assessment and Adaption of Software Process Requirements in ISO 26262. University of Waterloo

Weaver R, McDermid J, Kelly T (2002) Software Safety Arguments: Towards a Systematic Categorisation of Evidence. Department of Computer Science, University of York; York, UK.

Disclaimer This article contains UK MOD sponsored research and is released for informational purposes only. The contents of this article should not be interpreted as representing the views of the UK MOD, nor should it be assumed that they reflect any current or future UK MOD policy. The information contained in this article cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.

An Open, Transparent, Industry-Driven Approach to AV Safety

Jack Weast

Intel and Mobileye

Abstract *At Intel and Mobileye, saving lives drives us. Since joining forces, we've spread the word on the need for a safety standard for autonomous vehicles (AV), and how consumers and regulators alike demand transparency not offered by existing metrics used in AV safety claims. We proposed Responsibility-Sensitive Safety as a potential solution, a formal, mathematical model that defines what safe driving looks like. It was our first step towards building consensus in the industry. Today we take the next step in that journey, diving deeper into the makeup of RSS: What is this model, how does it work under the hood, and how can RSS help us balance the tradeoff between safety and usefulness of AV's? Higher levels of safety may result in overly conservative AVs that nobody wants on the road. So where should industry and the public draw the line to answer the question "How safe is safe enough"? Help us drive the conversation today that will enable the autonomous tomorrow.*

© Jack Weast, 2020
Published by the Safety-Critical Systems Club. All Rights Reserved

POSTERS

Applying reverse engineering to perform integrated safety and cybersecurity analyses of system functionality

Xinxin Lou[1], Peter B. Ladkin[1], Karl Waedt[2], Ines Ben Zid[1]

1 Bielefeld University

2 Framatome GmbH

Abstract *We propose an approach to perform a safety-with-cybersecurity analysis of existing industrial safety-critical systems. These were originally designed without essential cybersecurity considerations, while being used in today's operating environment. As an example, we consider a refuelling machine (RFM) that is commonly used to perform the fuel assembly loading and unloading in Nuclear Power Plants (NPPs). The objective of the analysis is to identify (and allow the chance to mitigate) those cybersecurity vulnerabilities which have implications for the safe operation of the RFM. First, we reverse engineer a functional specification (FS) of a legacy RFM, based on the manuals and handbooks, e.g. the system operation description, the controller and components handbooks. We also consult current designers. Additionally, we evaluate general information that we collected from publicly available RFM descriptions. The FS is expressed in the form of Hoare Triple, illustrated with preconditions and post-conditions of each function. Then we identify the hazards in this FS, by utilizing event tree analyses (ETA). Last, we analyse the possible causes of each hazard from considering the preconditions and post-conditions of each function. The detailed steps of each key part are described in the respective sections.*

Keywords*: reverse engineering, cybersecurity analysis, functional specification, hazard identification, risk analysis, Hoare Logic, event tree analysis*

1 Introduction

Safety and cybersecurity are increasingly important with Industry 4.0. Recent attacks, such as those on the Ukrainian Critical Infrastructure (Booz A. H. 2016) and the City of Joburg (BRI 2019) led to inconvenience on the daily life and

© Xinxin Lou, Peter B. Ladkin, Karl Waedt, Ines Ben Zid 2020
Published by the Safety-Critical Systems Club. All Rights Reserved

safety concerns. Some legacy safety-critical systems were designed without considering cybersecurity issues originally, while they are still running today. Given the growing cybersecurity challenges, how to prevent cyber attacks from happening on these systems, and how to involve cybersecurity considerations into new designs are worth investigating. The best idea to solve these situations is to avoid them. This can be achieved through analysing the system cybersecurity vulnerabilities before they are leveraged by attackers. Therefore, the intention of this work is to identify (and allow the chance to mitigate) those cybersecurity vulnerabilities which have implications for the safe operation of an existing system, for example a Refuelling Machine (RFM). An abstracted RFM is shown in Fig. 1. The FAI is fuel assembly Insert, CPIs (Chausse-Pieds Intégrés), is used to assist with lowering FA into a target cell accurately, Video Cameras are used for monitoring and checking the refuelling process, the Bridge is used for moving the RM in X axis (0°-180°) and Trolley is in charge of moving the RM in Y axis (90°-270°) [Xinxin 2018].

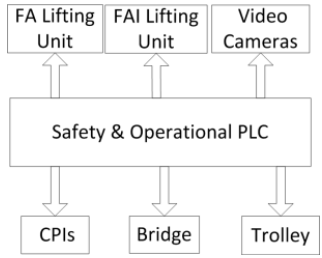


Fig. 1. General components of a Refueling Machine [Xinxin 2018]

The analyses only consider causes that arise from cybersecurity vulnerabilities. As illustrated in Fig. 2, there are three main steps in this analysis approach. We first reverse-engineer a Functional Specification (FS) of the RFM, then identify the hazards by combining the HAZOP (Hazard and Operability) (IEC 2011), OPRA (Objects, Properties, Relations, Assertions) (Peter 2017) and ETA (Event Tree Analysis). Finally, the analysis of possible causes of each hazard is performed to identify the cybersecurity vulnerabilities of the system. This is achieved by analysing the system architecture, by referring to open source vulnerability library (e.g. the CVE (CVE 2019), NVD (NVD 2019)), and the practical results of publications. Then we build attack models on this specific system, and finally assess the feasibility of such cyber attacks. Resultant mitigations are not included in this work, which is purely analytic. Each step will be illustrated in detail in subsequent sections.

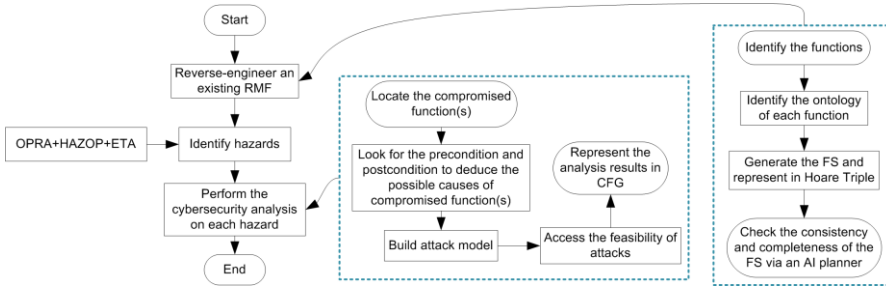


Fig. 2. The overall approach

The two main novel parts to this approach are:

(1) The reverse-engineering technique. Such reverse-engineering has often been thought to be infeasible for real systems. We show in contrast that it is feasible. In addition, an approach of using an artificial intelligence planner to check the correctness, completeness and consistency of the FS is proposed (already published in our previous work).

(2) The formal analysis method. The analysis is developed on the basis of the FS that is expressed in Hoare Triple. Potential causes are deduced by examining the status of each condition in the preconditions and postconditions of a function. In comparison with fault tree or attack tree analyses, a stricter representation, the Casual Fault Graph (CFG) (Ladkin 2015) with Counterfactual Test (CT) (Peter 2017) is utilized to illustrate the analysis result. This allows the causal-factor relations to be checked. The arrangement for the rest of this paper is as follows: the main process of reverse-engineering a system is illustrated in section 2, an example is also included in this section. In section 3, we describe the approach of identifying the hazards. The idea of cybersecurity analysis based on the FS is reported in section 4. Finally, we conclude the overall paper.

2 Reverse-engineering a functional specification (FS)

Reverse engineering is the process of analysing a subject system to identify the system's components and their interrelationships and to then create representations of the system in another form or at a high level of abstraction (Eillot and James 1990). Researchers have used reverse engineering on various domains, such as: (i) hardware analysis (Eillot and James 1990), (ii) communication protocol analysis (Stephan et al. 2019), e.g. Narayan et. al. (John 2015) investigate the tools to reverse engineer a protocol, understand and modify software (Lionel et al. 2006), (iii) electronic circuits (especially the integrated circuits) (Burcin and

Sharad 2018) (Cunxi and Maciej 2019) (Marc et al. 2018). In our work, we propose to apply reverse engineering on a safety critical system by producing formal system functional specifications.

2.1 Express the FS in Hoare Logic

To reverse-engineer a FS of the RFM, we collect information from system handbooks and component manuals. In addition, we also consulted the current designers. Some of the general information is collected from various RFMs. The FS is expressed in the form of Hoare Triple, illustrated with preconditions and postconditions of each function. The format of a Hoare Triple is as shown in Equation (1):

$$\{P\}S\{Q\} \tag{1}$$

P indicates the preconditions of executing a function, Q indicates the postconditions executing a function, and the S represents the command that is used to execute the function. P and Q are expressed in predicate logic. We collect all necessary requirements of executing a function, represent them with FOL and put them into preconditions in a Hoare Triple. We proceed similarly with postconditions.

The detailed steps of getting the FS are illustrated in Fig. 2. The system functions should be identified first. Then the ontology of each function has to be identified. Then they are represented with Hoare Triple. After the FS of all functions are generated, the correctness, consistency and the completeness of the FS will then be checked. An example of a functional specification is illustrated in the left part of Fig. 3. It is necessary to note that, each FS is represented as a Hoare Triple, but when we perform checking of the FS, this is done by translating them into the PDDL (Planning Domain Definition Language) (Maria and Derek 2003) to be compatible with the Fast Forward (FF) (FF 2011) planner. Therefore, there are two main differences between the FS in Hoare Triple and the FS that are checked by FF planner:

- The expression format of the FS is different, in addition, there are more conditions in the FS that are expressed in PDDL (for specification checking). For example, we use only one condition to indicate a situation in Hoare Triple, while it needs more conditions to represent this situation in PDDL during the specification checking, otherwise the planner cannot “understand” this situation clearly.
- Another reason we simplify the conditions in the Hoare Triple is that, each condition in the preconditions and postconditions will be analysed later on (in the cybersecurity analysis stage). Therefore, if there are too many similar descriptions of one restriction of a function, the workload will be increased.

As an example of a FS, the "Engage FA gripper" function in Hoare Logic and in PDDL (only the precondition) is illustrated in Fig. 3. The Hoare Logic style description is on the left of Fig. 3, the status of D1 (Door1) and D2 (Door2) are not necessary to indicate, while they are needed during the specification checking process (the PDDL description, on the right side of Fig. 3).

$$
\begin{aligned}
&Precondition\\
&\Longleftrightarrow\\
&((Above(RM, TargetCell)\\
&\wedge In(RM, ReactorCore)\\
&\wedge Full(TargetCell)) \vee (Above(RM, FTF)\\
&\wedge In(RM, TransferZone) \wedge Full(FTF)\\
&\wedge Vertical(FTF))) \wedge Empty(RM)\\
&\wedge Placed(FAGripper, Set-down\_Axis)\\
&\wedge Disengaged(FAGripper)\\
&\wedge Inside(FAGripperFingers, FATopNozzle)\\
&\wedge Closed(FAGripperFingers) \wedge LowLevel(FAGripper)\\
&\wedge Stopped(RM, H) \wedge Stopped(RM, Z) \wedge (NoAlarm)\\
&\\
&Postcondition\\
&\Longleftrightarrow\\
&Engaged(FAGripper) \wedge Open(FAGripperFingers)\\
&((Above(RM, TargetCell) \wedge In(RM, ReactorCore)\\
&\wedge Full(TargetCell)) \vee (Above(RM, FTF)\\
&\wedge In(RM, TransferZone) \wedge Full(FTF)\\
&\wedge Vertical(FTF))) \wedge Empty(RM)\\
&\wedge Placed(FAGripper, Set-down\_Axis)\\
&\wedge Onload(RM)
\end{aligned}
$$

```
:parameters (?cell -cell ?r -room
             ?cell_axis -align_axis)
:precondition
(and
    (InRoom RM ?r)
    (CellInRoom ?cell ?r)
    (AxisInRoom ?cell_axis ?r)
    (RMAbove ?cell)
    (GripperAbove FAGripper ?cell)
    (PlacedHoistAlong ?cell_axis)
    (Inside FA ?cell)
    (GripperLowLevel FAGripper)
    (Disengaged FAGripper)
    (RMStopped Z)
    (RMStopped H)
    (EmptyRM)
    (NoAlarm)
    (DoorClosed D1)
    (DoorClosed D2)
    (RotationLocked FAGripper)
    (FingerClosed FAGripperFingers)
    (Inside FAGripperFingers FATopNozzle)
)
```

Fig. 3. Example of a FS of "Engage FA gripper function"

2.2 The FS consistency and completeness checking

The consistency and the relative completeness of each FS has been checked by using an opensource AI planner (Xinxin 2018). It calls a Fast Forward planner (FF 2011). The reason we say that it is with regard to relative completeness is that all relevant necessary sub-functions of a main function are identified, but not the functions that are not directly relevant with this main function. Part of the updated and refined FS after our previous work (Xinxin 2018) is illustrated in Fig. 4. It is the result of checking the specification of the "Reloading FA (Fuel Assembly)" process. Each step in Fig. 4 is a sub-function of the reloading FA function.

```
ff: found legal plan as follows

step    0: ROTATIONLOCK FAGRIPPER
        1: MOVE_NEXTTO_CELL FTF FTFNEXTTOFROM RMORIGINNEXTTO TRANSFERAREA RMORIGINNEXTTO_AXIS FTFNEXTTOFROM_AXIS
        2: REALIGNMENT_FAHOIST_SET-DOWN_AXIS FTF_AXIS FTF FTFNEXTTOFROM_AXIS TRANSFERAREA FTFNEXTTOFROM FTFNEXTTOTO
        3: LOWER_FAGRIPPER FTF TRANSFERAREA FTF_AXIS FTFNEXTTOTO
        4: ENGAGE_FAGRIPPER FTF TRANSFERAREA FTF_AXIS
        5: LIFT_FA FTF TRANSFERAREA FTF_AXIS
        6: MOVE_NEXTTO_DOOR_L D1 D1_LEFT TRANSFERAREA INTERMEDIATEAREA FTFNEXTTOTO FTF_AXIS D1_LEFT_AXIS
        7: OPENGATE D1 D1_LEFT TRANSFERAREA INTERMEDIATEAREA D1_RIGHT
        8: GOTHRUDOOR_L_R D1 D1_LEFT D1_RIGHT D1_LEFT_AXIS D1_RIGHT_AXIS TRANSFERAREA INTERMEDIATEAREA
        9: CLOSEGATE D1 D1_RIGHT TRANSFERAREA INTERMEDIATEAREA
       10: MOVE_NEXTTO_DOOR_L D2 D2_LEFT INTERMEDIATEAREA COREAREA D1_RIGHT D1_RIGHT_AXIS D2_LEFT_AXIS
       11: OPENGATE D2 D2_LEFT INTERMEDIATEAREA COREAREA D2_RIGHT
       12: GOTHRUDOOR_L_R D2 D2_LEFT D2_RIGHT D2_LEFT_AXIS D2_RIGHT_AXIS INTERMEDIATEAREA COREAREA
       13: CLOSEGATE D2 D2_RIGHT INTERMEDIATEAREA COREAREA
       14: MOVE_NEXTTO_CELL TARGETCELL TARGETCELLNEXTTOFROM D2_RIGHT COREAREA D2_RIGHT_AXIS TARGETCELLNEXTTOFROM_AXIS
       15: REALIGNMENT_FAHOIST_SET-DOWN_AXIS TARGETCELL_AXIS TARGETCELL TARGETCELLNEXTTOFROM_AXIS COREAREA TARGETCELLNEXT
TOFROM TARGETCELLNEXTTOTO
       16: LOWER_FA TARGETCELL COREAREA TARGETCELL_AXIS TARGETCELLNEXTTOTO
       17: DISENGAGE_FAGRIPPER TARGETCELL COREAREA TARGETCELL_AXIS
       18: LIFT_FAGRIPPER TARGETCELL COREAREA TARGETCELL_AXIS


time spent:    1.00 seconds instantiating 41 easy, 126 hard action templates
               0.00 seconds reachability analysis, yielding 58 facts and 160 actions
               0.00 seconds creating final representation with 54 relevant facts
               0.00 seconds building connectivity graph
               0.02 seconds searching, evaluating 45 states, to a max depth of 2
               1.02 seconds total time

root@XS:/mnt/d/FF#
```

Fig. 4. The checking result of FS of the “Reloading FA” function

3 Hazard identification

A hazard is a potential source of harm (IEC61508-4-3.1.2). PHA (Preliminary Hazard Analysis) (Jeffrey 2014) and HAZOP (IEC 2011) are typical methods to identify hazards. The overview of identifying hazards in our work is shown in Fig. 5. As it is represented in right part of Fig. 5, we first get the deviations of system status, then deduce the possible consequences of these deviations. Hazards are identified from these deviations according to their possible consequences. The detailed idea is illustrated in the left part of Fig. 5.

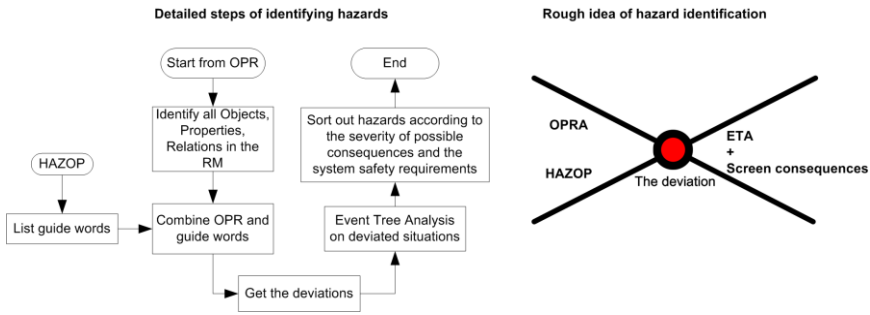


Fig. 5. Hazard identification by combining HAZOP, OPRA and ETA

To get the deviations, we first identify the ontology of a system. It is proposed to use the "OPRA" (Objects, Properties, Relations, Assertions) (Peter 2017). Once the ontology of a function is identified, we then combine the guide words in HAZOP with object properties and relations of objects to derive deviations (Peter 2011). Then the ETA is performed on each deviation, to deduce the possible consequences. The severity of potential consequences will be classified to various levels according to specific criterion, e.g. the International Nuclear Event Scale (INES). The hazards are finally screened out from the deviations according to the severity of possible consequence and system safety requirements.

4 Cybersecurity analysis

Once the hazards are identified, we then investigate possible causes of each hazard. Only causes that could potentially be triggered by cyber attacks will be included in this work. The detailed steps of performing cybersecurity analysis are illustrated in Fig. 2. The analysis starts from locating the relevant compromised functions, as we are analysing the cybersecurity of system functionality. Then we look for potential reasons of compromised functions from the FS. Potential causes are deduced by examining the status of each condition in the preconditions and postconditions before and after a function is executed. Which is to identify the causes that: (i) a function is executed while not all preconditions are satisfied, and (ii) the causes that not all postconditions are achieved after a function is executed legally.

On the basis of violation of each condition (precondition or postcondition), we deduce the possible attacks that could cause each violated situation, for example, a violated situation can be that the "Go through the Door" function is executed while the door is "Closed" (the door should be open if "Go through the Door" function is executed according to the FS). This is achieved by examining the system architecture vulnerability, investigating the components vulnerability libraries, e.g. the CVE and NVD. Some practical work such as (Garcia & Sadeghi, 2017) and (Spenneberg et al. 2016) are also referenced. The attack models are built to illustrate potential detailed attacks. However, not all vulnerabilities in the vulnerability libraries are possible to occur on this RFM. Then the feasibility assessment of potential attack models is performed based on our specific system structure, e.g. in previous work (Xinxin et al. 2019).

The overall analysis result of the system is represented in CFG (Peter 2015). The reason that we choose CFG instead of Fault Tree or Attack Tree is that, the Counterfactual Test (CT) is involved during producing a CFG to illustrate the analyse result. This allows the causal-factor relations to be checked. Subsequent mitigation is not included in this work, but to some extent, the mitigation can be found out from the final analysis results.

5 Conclusion

In conclusion, an approach of applying reverse engineering to perform safety and cybersecurity analysis is proposed. This includes three key parts: (i) reverse-engineering a functional specification of a refuelling machine, (ii) identification of system hazards, and (iii) analysis of the causes of hazards (only causes that are led by cybersecurity vulnerabilities). Each key part is illustrated in this paper.

Acknowledgments This work is partially funded by German Ministry BMWi, elaborated as part of Framatome GmbH's participation in the SMARTEST R&D with German University partners. We thank some of industry engineers who support this work and all reviewers.

References

Booz A. H. (2016) When the lights went out. Available from https://www.boozallen.com/content/dam/boozallen/documents/2016/09/ukraine-report-when-the-lights-went-out.pdf, accessed 2019-10-25.

BRI Germany (2019) Available from https://brica.de/alerts/alert/public/1283450/city-of-joburg-banks-under-cyber-attack/, accessed 2019-10-25.

Burcin C. and Sharad M. (2018) Reverse Engineering Digital ICs through Geometric Embedding of Circuit Graphs, ACM Trans. Des. Autom. Electron. Syst., vol. 23, no. 4, pp. 1–19.

Cunxi Y. and Maciej C. (2019) Formal Analysis of Galois Field Arithmetic Circuits-Parallel Verification and Reverse Engineering, IEEE Trans. Comput. Des. Integr. Circuits Syst., vol. 38, no. 2, pp. 354–365.

CVE (2019) Common Vulnerabilities and Exposures. Available from https://cve.mitre.org/, accessed 2019-11-10.

Eillot J. Chikofsky and James H. C. (1990) Reverse engineering and design recovery: a taxonomy, IEEE Softw., pp. 1–5.

FF Homepage (2011) Available from https://fai.cs.uni-saarland.de/hoffmann/ff.html, accessed 2019-10-25.

Garcia, L. A., & Sadeghi, A. R. (2017) Hey, My Malware Knows Physics! Attacking PLCs with Physical Model Aware Rootkit. NDSS Symposium. Available from https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/hey-my-malware-knows-physics-attacking-plcs-physical-model-aware-rootkit/, accessed 2019-10-25.

IEC TC56 (2016), IEC61882: Hazard and operability studies (HAZOP studies)-Application guide.

Jeffrey W. V. (2014) Chapter 6 Preliminary Hazard Analysis, in Basic Guide to System Safety (3rd Edition). John Wiley & Sons, Inc.

John N., Sandeep K. S., and Charles C. C. (2015) A Survey of Automatic Protocol Reverse Engineering Tools, ACM Comput. Surv., vol. 48, no. 3, pp. 1–26.

Lionel C. B., Yvan L. and Johanne L. (2006) Toward the reverse engineering of UML sequence diagrams for distributed Java software IEEE Trans. Softw. Eng., vol. 32, no. 9, pp. 642–663.

Maria F., Derek L. (2003) PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains, Journal of Artificial Intelligence Research 20 (2003) 61-124.

Marc F., Sebastian W., Pawel S. et al (2018) HAL- The Missing Piece of the Puzzle for Hardware Reverse Engineering, Trojan Detection and Insertion, IEEE Trans. Dependable Secur. Comput., vol. 16, no. 3, pp. 498–510.

NVD (2019) National Vulnerability Database. Available from https://nvd.nist.gov/vuln/categories/, accessed 2019-11-10.

Peter B. L. (2011) Kapitel 9 OHA Beispiel Automobil Kommunikationsbus in Safety of Computer-Based Systems, e-textbook, RVS Group, Bielefeld. Available from https://rvs-bi.de/publications/books/ComputerSafetyBook/index.html, accessed 2019-10-25.

Peter B. L. (2015) Risks People Take and Games People Play, RVS Group, Bielefeld. Available from https://rvs-bi.de/publications/Talks/LadkinSSS2015.pdf, accessed 2019-10-25.

Peter B. L. (2017) Digital System Safety, e-textbook draft, RVS Group, Bielefeld. Available from https://rvs-bi.de/publications/RVS-Bk-17-02.html, accessed 2019-10-25.

Spenneberg, R., Brüggemann, M., and Schwartke, H. (2016) PLC-Blaster: A Worm Living Solely in the PLC, Black Hat. Available at https://www.blackhat.com/docs/asia-16/materials/asia-16-Spenneberg-PLC-Blaster-A-Worm-Living-Solely-In-The-PLC-wp.pdf, accessed 2019-10-25.

Stephan K., Lisa M., and Frank K. (2019) Survey of protocol reverse engineering algorithms: Decomposition of tools for static traffic analysis, IEEE Commun. Surv. Tutorials, vol. 21, no. 1, pp. 526–561.

Xinxin L., Karl W. and et al. (2018) Combining AI planning advantages to assist preliminary formal analysis on ICS cybersecurity vulnerabilities. ECAI 2018, Iasi, Romania.

Xinxin L., Karl W. and et al. (2019) Cybersecurity Analysis of Industrial Control System Functionality. IEEE ICPS2019, Taiwan.

A Step Towards Harmonising IEC Terminology

Dieter Schnäpp[1], Peter Bernard Ladkin[2], Holger Lange[3]

IVA, Technical University of Braunschweig/Causalis IngGmbH/VDE

Braunschweig/Bielefeld/Frankfurt, Germany

Abstract *The Harbsafe[4] project analysed the technical terminology defined in an array of IEC standards and guides concerning functional safety and cybersecurity. 460 terms were defined in the documents surveyed, most given in Clause 3 Terms and Definitions[5] of IEC documents. IEC publishes guidelines for terminology; terminology conformant with these guidelines is said to be "harmonised". We found that terminology in the documents reviewed was not well harmonised. We devised three techniques to aid harmonisation: an application of machine-learning "word embedding" analysis to identify related concepts, possibly synonyms, which were not overt; SemAn (a variety of Semantic Analysis) for analysing and possibly harmonising the definiens of homonyms and almost-synonyms; ConcAn (a variety of Conceptual Analysis) for analysing terms whose overt definition did not seem to us to fit well with everyday engineering use. The first author wrote a WWW-based tool, the Terminology Dashboard, now on-line at VDE, to aid engineers in navigating a database of terms and definitions and their relations to each other.*

1 Corresponding Author: d.schnaepp@tu-braunschweig.de

2 ladkin@causalis.com

3 holger.lange@vde.com

4 The authors gratefully acknowledge the support of the German Federal Ministry for Economic Affairs and Energy (BMWi) (Grants: 03TNG006A and 03TNG006B).

5 An exception is IEC 61508 [IEC 61508], in which technical terminology is defined in a separate document, Part 4.

© Dieter Schnäpp, Peter Bernard Ladkin, Holger Lange 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

1 The Importance of Harmonised Terminology

Terminology is important. An engineer designing and building a piece of safety-related equipment for application in an industrial system considers her job to ensure that there is "freedom from unacceptable risk" (IEC 61508 Part 4, subclause 3.1.11, also IEC Guide 51), where "risk" is a "combination of the probability of occurrence of harm and the severity of that harm" (op. cit., subclause 3.1.6, also IEC Guide 51), and "harm" is "physical injury or damage to the health of people or damage to property or the environment" (op. cit., subclause 3.1.1, also IEC Guide 120). Whereas in autonomous driving it seems to be harder to discover what exactly is meant by "*acceptable safety*" (Uber 2018).

If the IEC definition of "safety" is used, then one must evaluate risk, the combination of likelihood and severity of harm. While Uber is "...deeply regretful for the crash in Tempe, Arizona, this March [2018]" (op. cit., Letter to the Reader), one can scan the 70pp document in vain for an assessment of likelihood (the severity of the harm is known, of course – 1 human fatality, 1 bent car, 1 live driver with her aftereffects).

If we are to consider just that part of the harm that is the human fatality, and consider there to have been roughly a million miles logged by autonomous road vehicles on public roads at the time of the accident (a popular estimate), then before the fatal accident we could have been around 63% confident that the fatal accident rate for autonomous vehicles was better than 1 in a million miles, but after the one fatal accident our confidence in that rate reduced to 26%, although we could still be about 60% confident that the rate was better than 1 in 500,000 miles (Bishop 2018). For comparison, for human driving the rate in which one can have about 63% confidence is 1 fatality in 100,000,000 miles (op. cit.)[1].

It seems clear from this that when a functional-safety engineer talks about "s*afety*" and when an autonomous-driving engineer working for Uber talks about "s*afety*", they are not talking about the same thing.

We might wish that they were. We might also wish to compare the two uses of the word "safety" in the context of functional safety engineering, and in the context of autonomous driving, to see how similar and different they are.

[1] Of course, one could take a view that this rate is objectively known: it is the number of fatalities divided by (an estimate of) the number of miles driven. But this number could have been different – many if not most accidents have a component of chance; the observed rate has a stochastic component and the figures are more appropriately represented as <rate, confidence-level> pairs. In other words, if we want to claim that the system of human driving has some intrinsic rate of fatal accidents, observed accidents yield an estimate of that rate expressed as <rate, confidence-level> pairs. Usually one would be interested in rates at which the confidence level was 90%, or 95%. The ∽60% rate arises here as a suitable comparison point simply from the sparse data on autonomous driving. You cannot be very confident if data are sparse; a million miles is not a lot here. So why do the numbers? The answer is that it is part of the IEC definition of safety, and we are trying to judge that safety, even if we cannot do so to 90% confidence.

But even within specific engineering sectors, commonly used technical terms mean different things: in IEC Technical Committee 65 (and its Subcommittees), the functional safety standard for industrial process plant (IEC 61511) defines many key terms differently from the basic safety standard IEC 61508, which comes from the same Subcommittee (SC65A, System aspects): 12 key terms contain a note: "*NOTE This definition differs from the definition in IEC 61508-4 to reflect differences in the process sector*", that is, the document is overtly defining homonyms. Also, at least one term is different from the term used for the same concept (given by the same definiens) in IEC 61508, that is, it is an overt synonym. There seem to be further non-overt similarities and differences (IEC 61511). Such overt occurrence of homonyms and synonyms conflicts with ISO/IEC guidelines (ISO 10241-1).

2 The Task, the Corpus

In the authors' experience, terminology and harmonisation tasks are currently undertaken in national and IEC standardisation committees through participants thinking and discussing the words in an informal way. The question for us was: could some intellectual and/or software-assisted technology be useful here, obtaining equivalent or, we hoped, more satisfactory results than traditional informal discussion? We devised some methods and tried them out on a limited corpus. We believe the analytical techniques we devised are indeed an improvement and will be helpful in future harmonisation activity.

The Harbsafe project analysed terminology in electrotechnical standards documents associated with functional safety and cybersecurity. The organisation which produces the international electrotechnical standards is the International Electrotechnical Commission, based in Geneva, which is one of three global "sister" organisations which develop international standards (the others are the International Organisation for Standardisation, ISO, and the International Telecommunications Union, ITU). Harbsafe classified such standards into priority groups, and researched the first group, which consisted of the ISO/IEC Guides to the inclusion of safety aspects, resp. cybersecurity aspects in electrotechnical standards (ISO Guides 51, 120), the "Basic Safety Standard" ISO 61508 (ISO 61508), and the cybersecurity standards series for industrial automation and control systems, aka IACS (IEC 62443). Other IEC standards dealing with these topics were classified as lower priority.

This corpus is orders of magnitude smaller than corpori for which machine-learning word-embedding (ML-WE) techniques have previously been shown to work. Nevertheless, non-trivial results were obtained in Harbsafe through use of ML-WE techniques on this small corpus. These results are displayed in part in the Dashboard SW developed in the project. The size of the corpus, while small for ML-WE, is nevertheless somewhat daunting for manual techniques, such as

the semantic analysis SemAn and conceptual analysis ConAn techniques we devised, in terms of the effort required.

3 Similarity and Difference: Word-Embedding Studies

Word-embedding analysis is a recent machine-learning technique, which considers words in a corpus and derives lengthy vectors of characteristics (in our case, some 300+) of contexts in which the words occur, in order to derive assessments of semantic similarity and difference of the words as they are used in the corpus (Mikolov 2013). This is, in other words, a concrete application of Wittgenstein's dictum that "meaning is use" (Wittgenstein 1953). The first author wrote word-embedding assessment software based on the techniques of Arora (Arora 2017) and used it to derive similarity/difference assessments for various terms, both those defined explicitly in Sections 3 (and IEC 61508 Part 4) as well as for those "common" engineering terms which were used in the documents but not regarded by the authors as worthy of definition. The resulting "word vectors" were used to construct a pseudo-3-dimensional similarity plot in the Dashboard SW (Arndt, Schnäpp 2018).

These techniques we have found difficult to illustrate at a length suitable for this extended abstract: we thereby refer to the full paper (Arndt, Schnäpp 2018).

4 Semantic Analysis: SemAn

The second author compared homonyms found in the 460 definitions manually. There were 63 multiply-defined concepts. About a third (22) were repetitions of definitions (same definiendum and same definiens) which could be replaced by one explicit definition to which all other occurrences refer. 11 terms exhibited minor differences (largely punctuation or grammatical variation). 28 terms, a little under half the total, exhibited moderate to substantial differences – that is, they are true homonyms (Ladkin 2019-1).

Such definitions may be formally parsed to indicate explicitly the similarities and differences (omissions, extra parts, variations of parts). A definiens is thereby translated into a formal language, the language of sorted predicate logic (LSL), and the resulting formal descriptions compared. We found that this accurately localised and highlighted the issues to be resolved. We devised a process for pursuing this form of analysis effectively, SemAn (Ladkin 2019-3).

We present a simple example, rendered in "controlled English", that is, a formal language with the surface structure of English, rather than LSL. "*Harm*" is defined in IEC 61508 and ISO/IEC Guide 120 as "*physical injury or damage to the*

health of people or damage to property or the environment" (IEC 61508 subclause 3.2.1, ISO/IEC Guide 120 subclause 3.7). We expand this into the controlled-English term:

(DEF1) *physical injury to a person OR damage to the health of a person OR damage to property OR damage to the environment*

In ISO/IEC Guide 51, the definition is syntactically different: "*injury or damage to the health of people, or damage to property or the environment*" (ISO/IEC Guide 51). Do these two definitions mean the same thing or not (are they semantically equivalent)?

Since (Meaning Postulate 1) "*people*" means "*one or more persons*", any action on a person is ipso facto an action on one or more persons and thus on "*people*", conversely any action on "*people*" is an action on at least one "*person*". It follows that the syntactic difference "*person*"/"*people*" is semantically insignificant. Also (Meaning Postulate 2) *"physical injury"* of a living being is "*injury*" to that being, which in turn is "*damage to the health*" of that being. It follows under these two Meaning Postulates that both definitions turn out to be semantically equivalent to

(DEF2) *damage to the health of people OR damage to property OR damage to the environment*

and therefore semantically equivalent to each other. Of course, any audience for this observation reaches this conclusion, intuitively, rather more quickly than we did here! But speed is not the issue; there are subtler, more involved cases in which the SemAn principles are not only helpful but decisive.

According to IEC guidelines, the issues highlighted by SemAn are for the nominated subject-matter experts to resolve, those working in the Project Teams/Working Groups who authored the document, in consultation with IEC TC 1, the Terminology TC.

5 Conceptual Analysis: ConcAn

Some key concepts turn out to have very different meanings in different documents, for which possible resolutions are not helpfully indicated through SemAn.

For example, "*Integrity*" is defined variously as a "*quality of a system ...*" (IEC 62443-1-1 subclause 2.1.55), or as a "*probability*" (IEC 61508-4 subclause 3.1.4), which is then further confused by saying, of part of "*safety integrity*", that it "*...cannot usually be quantified*" (Ladkin 2017 Chapter 5). Probabilities are numbers between 0 and 1, or percentages between 0% and 100%. First, numbers are not "*qualit[ies] of a system*", so these two definitions are clearly homonyms.

Second, how can a probability, a number, "*[not] [be] quantified*"? If a probability cannot be quantified, it cannot be a number. What is it, then? A number or a quality? A further source, outside the corpus, says "*integrity*" is an "*absence*" (namely, "*.... of improper system alterations*") (Avizienis et al. 2014). To stir things up even more, IEC 62443-1-1 subclause 2.1.55 has a "Note to entry: In a formal security mode, integrity is often interpreted more narrowly to mean protection against unauthorized modification or destruction of information". So, a number, or a quality, or "*protection*", or an absence – which?

These are differences in what is known as (metaphysical) category (Ryle 1949). Resolving them cannot simply be accomplished by translating everything into a logical language and discriminating subconcepts, as in SemAn. Answering the question whether integrity is a system quality, "*protection*", a number, or an absence is a task which has been at home in analytical philosophy for some hundreds – even thousands – of years. ConcAn gives some guidelines as to how this philosophical skill can be used in the analysis of technical terminology (Ladkin 2019-2).

Acknowledgments We thank our former Harbsafe colleagues Susanne Arndt of the Technical Information Library (Technisches Informationsbibliothek) in Hannover, and Sven Müller of Rail Power Systems GmbH in Offenbach for substantial contributions to the work reported here.

References

Avizienis, A., Laprie, J.-C., Randell, B., and Landwehr, C., (2004), Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Trans. Depend. Sec. Comp. 1(1):1-23. Available from https://www.nasa.gov/pdf/636745main_day_3-algirdas_avizienis.pdf , accessed 2017-12-01.

Arndt, S.; Schnäpp, D. (2018) Harbsafe-152 – A Domain-Specific Data Set for the Intrinsic Evaluation of Semantic Representations for Terminological Data. Submitted for publication.

Arora, S. et al. (2017) A Simple but Tough-to-Beat Baseline for Sentence Embeddings. https://openreview.net/pdf?id=SyK00v5xx.

Bishop, P. (2018) Personal communication with the second author, 2018-04-11.

International Electrotechnical Commission, IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related systems, 2nd Edition, 7 parts, 2010.

International Electrotechnical Commission, IEC 61511, Functional safety – Safety instrumented systems for the process industry sector, 2nd Edition, 3 parts plus AMD1:2017, 2016/2017.

International Electrotechnical Commission, IEC 62443, Security for industrial automation and control systems, many parts, various dates 2009-2019.

International Organisation for Standardization, ISO 10241-1, Terminological Entries in Standards – Part 1: General Requirements and Examples of Presentation. ISO, 2011.

International Organisation for Standardization/International Electrotechnical Commission, ISO/IEC Guide 51 Edition 3, Safety aspects – Guidelines for their inclusion in standards, 2014.

International Organisation for Standardization/International Electrotechnical Commission, ISO/IEC Guide 120 Edition 1, Security aspects – Guidelines for their inclusion in publications, 2018.

Ladkin, P.B., (2017) A Critical-System Assurance Manifesto: Some Issues Arising from IEC 61508, RVS-Bk-17-01, RVS Group, Bielefeld University. Available from *https://rvs-bi.de/publications/RVS-Bk-17-01.html* , accessed 2019-11-13.

Ladkin, P. B., (2019-1) Multiply-Defined Concepts: Diff Notes, unpublished manuscript, Causalis Ingenieurgesellschaft mbH.

Ladkin, P. B., (2019-2) Conceptual Analysis: ConcAn, unpublished manuscript, Causalis Ingenieurgesellschaft mbH.

Ladkin, P. B., (2019-3) Semantic Analysis: SemAn, unpublished manuscript, Causalis Ingenieurgesellschaft mbH.

Mikolov, T. et al. (2013) Efficient Estimation of Word Representations in Vector Space, arXiv:1301.3781.

Ryle, G. (1949) The Concept of Mind, Hutchinson & Co. Reprinted University of Chicago Press, 2000; Penguin Classics, 2000.

Uber Advanced Technologies Group, A Principled Approach to Safety, 2018. Available at *https://uber.app.box.com/v/UberATGSafetyReport* , accessed 2019-11-13.

Wittgenstein, L. (1953) “Philosophical Investigations (PI)”. 4th edition, 2009, Oxford: Wiley-Blackwell.

Formal verification of relative safety for autonomous decision making

Hoang Tung Dinh

KU Leuven, Belgium

Abstract *An autonomous system should only make decisions that are safe. However, since the system only has partial control over the environment, achieving absolute safety is impossible. If a person jumps in front of a fast-moving autonomous car, the car may not be able to stop in time. For certification and liability assignment, the decision making logic should be able to state explicitly on which assumptions it relies and provide guarantees that, under these assumptions, safety properties hold. Although generally conceived as crucial, assumptions are typically not dealt with explicitly. State-of-the-art decision making is often the result of learning or advanced planning techniques, encoding many implicit assumptions on the operating environment. We propose an approach to reveal assumptions and verify relative safety for decision making policies. Relative safety provides conditional guarantees - given that the explicitly-specified assumptions are valid during the operation, we can provide solid guarantees. We use the highly expressive formal logic FO(.) to specify assumptions and desired properties. We employ the state of the art knowledge base system IDP as a model checker to verify desired properties and reveal missing assumptions. In our approach, one can discover and add assumptions in an iterative process. A thorough validation of the approach in two case studies is included: an autonomous UAV system for pylon inspection and a semi-autonomous car for highway driving.*

An Effective Approach to Meeting the Challenges of RTCA DO-326A

Elizabeth Lennon

Dstl

Abstract *RTCA's DO-326A describes a Security Airworthiness Process that aligns the security process to the safety process with the intent of identifying the impact of malicious cyber security attacks on the safety of an airborne system. Dstl have developed an incremental process for the engagement with stakeholders such that available supporting evidence to the DO-326A objectives can be identified and reasoned with. Specifically, the process defines an approach to utilise pre-existing security evidence from alternative security engineering processes, and exploits the Goal Structuring Notation (GSN) to store and explain the argument as to whether an acceptable means of compliance can be determined based on the available evidence. Key to the value of this work is the ability to identify any fundamental shortfalls in meeting the intent of DO-326A, that is, in addressing the challenge of security-informed safety. By systematically assessing the potential for existing evidence to meet some or all of DO-326A, the dialogue with stakeholders can be focussed to the development of mitigations where it is required.*

© Crown copyright (2019), Dstl. This material is licensed under the terms of the Open Government Licence except where otherwise stated. To view this licence, visit https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3 or write to the Information Policy Team, The National Archives, Kew, London TW9 4DU, or email: psi@nationalarchives.gsi.gov.uk

Published by the Safety-Critical Systems Club. All Rights Reserved

Safety approaches for autonomous mobile machines in industrial environments

Risto Tiusanen, Timo Malm, Eetu Heikkilä, Janne Sarsama[1]

VTT

Abstract *The development of automated mobile machines towards autonomous operation is proceeding rapidly in many industrial sectors. New technologies and their increasing complexity set challenges for designing safe and reliable machinery and systems. One big challenge is to manage system-level safety and reliability risks with cost-effective solutions in applications where autonomous machinery, manual machines and employees aims to work in the same area. Three safety design approaches for autonomous machinery are introduced and discussed through the requirements and constraints imposed by system operating concepts and operating environments. The key element in machine autonomy is adaptability to dynamically changing environment based on the available information. Current safety engineering methods developed for automated machinery do not cover or consider autonomy aspects like dynamic risk assessment or independent decision-making. Safety standards for the design of autonomous machinery applications are also discussed.*

Keywords: autonomous work machine, standard, safety approach, safety requirement.

1 Introduction

The development towards a higher automation level in material handling and logistic systems in industry is a clear global trend. This means that machine manufacturers' interests are changing towards system level aspects also regarding safety issues. Instead of optimizing single machine's operational capability and

[1] Risto Tiusanen: VTT Technical Research Centre of Finland Ltd, E-mail: risto.tiusanen@vtt.fi
Timo Malm: VTT Technical Research Centre of Finland Ltd, E-mail: timo.malm@vtt.fi
Eetu Heikkilä: VTT Technical Research Centre of Finland Ltd, E-mail: eetu.heikkilä@vtt.fi
Janne Sarsama: VTT Technical Research Centre of Finland Ltd, E-mail: janne.sarsama@vtt.fi

© Risto Tiusanen, Timo Malm, Eetu Heikkilä, Janne Sarsama 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

ensuring its safety, the system suppliers have to take into account the overall system constraints and goals, work processes, and the operational environment. Safety-critical systems are based more and more on software solutions and security functions use information from several interacting systems [1].

Different operating environments and work processes require different solutions to ensure safe operation of the automated or even autonomous systems. VTT is conducting research on different safety approaches and concepts in Finland in a jointly funded AUTOPORT project (https://autoport.fi) together with University of Tampere and a group of companies. The project is focusing on terminal and port operations and logistic robotics. Its main goal is to pave the way towards business renewal, and operational excellence by developing ecosystem level approaches for logistic robot systems. One interesting and challenging research topic in the AUTOPORT project is advanced safety concepts in logistic systems to enable automated machinery, manual machines and manual workers to operate and collaborate in the same open work area.

In general level safety strategies for automated mobile machinery can be categorised into three groups [2]:

1. **Use of safe separation distance to an automated machine**. In this strategy on-board sensors have capability to detect persons and obstacles in front of an automated machine or near it. The control of the separation distance can also be arranged with central control system, which know all the time the locations of all machines and persons in the operating area.

2. **Use of an isolated area for automated machinery.** The area is isolated and persons enter there through an access control system, which stops or limits the operation of the automated machines at the area. The isolated area may consist of several sectors and the automated machines are stopped only if person or a manual machine and the automated machines are at the same sector.

3. **Use of rules for working in a restricted automated area.** This strategy means that only authorised persons and machines may enter the restricted area. The entering persons must know the rules. This strategy can be applied only in low risk applications. Safe operation depends mainly on the persons at the area.

Derived from these general safety strategies three safety approaches have been identified and characterised for autonomous machinery. They are based on different needs, operating environments and industrial background. The first approach is for systems, where the autonomous mobile machine carries an on-board safety system and the safety system is not dependent on other external systems

in the infrastructure. The second approach aims to isolate the autonomous machinery, to control and restrict the access to the automated area and to track other actors (persons of vehicles) in the automated area. The third approach is for autonomous machinery systems, where system safety relies mainly on a human operator who is assisted with monitoring and warning systems.

An autonomous machine carries the safety system

The first approach operating concept is where the machine carries a sensor system and safety system is contained within a machine. This allows non-separated working areas for humans, machines and autonomous machines to operate in the same area. The detection of an object may cause a reduced speed, stop or rerouting of the machine to avoid collision with the detected object. Typical sensors to be applied in this approach are LIDARs, laser scanners, RADARs, UWB sensors, 3D-cameras, IR-cameras, proximity detectors (ultrasonic, optical, capacitive) and tactile bumpers. So far this approach has been used in indoor applications as the sensor systems needed are only suitable and certified for indoor use.

Isolated operating area, access control and tracking systems

The second approach aims to separate and isolate the autonomously operating machinery from other operations nearby, to control the access and to monitor other vehicles or persons in the autonomous operating area. This approach could be applied for autonomous operations in quite clear and unchanging outdoor environments. The approach has been applied in automated machinery systems, but so far, the safety functions simply stop the machinery if someone enters to the automated area. Currently there are no reliable sensor solutions for outdoor use, which limits the wider and more advanced use of this approach.

Human operators assisted with situational awareness information

The third approach relies strongly on machine safeguarding, monitoring functions and skilled and experienced operators. When a problematic or hazardous situation is detected, the operation could be stopped and the automated control is switched to the local or remote operator. The approach relies heavily on the operator's ability to understand the operational situation, machine functionalities and operator's capability to react correctly. The approach is suitable for working environments where there are relatively few simultaneous activities and a low likelihood of hazardous situation, and where there is enough time to warn the operator and transfer responsibility.

2 Guidelines from safety standards in different industrial sectors

Different industrial sectors have different safety strategies and approaches. There is a big difference in safety strategies in industrial environments compared with e.g. cars and public transport. Machine safety engineering approaches have currently rather a narrow view of autonomy aspects. Many of the standards related to safety requirements for autonomous machine systems are still in draft phase and the current existing standards will evolve. However, many activities are going on related to safety requirements and risk assessment processes in the context of machine autonomy. An overview of the current situation in safety standards in different industrial sectors and in different system levels is illustrated in figure 1.

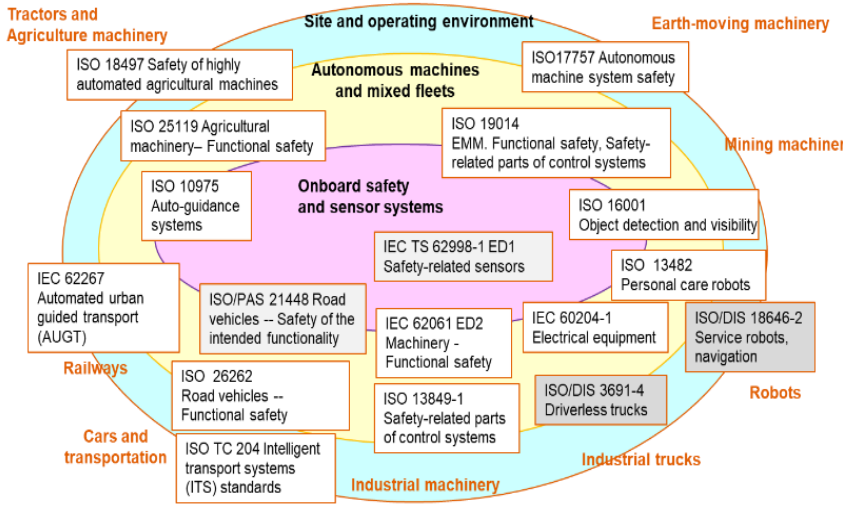


Fig. 1. An overview of the current situation in standardization [3].

In the context of autonomous machines some interesting standards are highlighted here. In connection to the first approach introduced above, ISO/DIS 3691-4 [4] defines requirements for the operation of driverless forklifts in different operating areas and the requirements for safety-related functions. The standard requires that any access to the automated area must be controlled. It also defines speed limits for specific operating conditions and functional safety requirements for safety functions. In connection to the second approach ISO 17757 [5], in the earth-moving and mining machine sector, gives guidelines how the safety risks should be assessed and how the system safety requirements should be defined in autonomous mobile machine applications. The approach emphasizes the risks related to the actual operating concepts and actual operating environment at the site

and the uncertainties related to the safety-related functions and technologies. The standard introduces the concept of an autonomous operating zone (AOZ), controlled by the access control system, where monitored manned machines and monitored persons can work at the same time with autonomous machines (see figure 2). In connection to the third approach, ISO/FDIS 18497.2 [6] defines the requirements for the deployment and implementation, monitoring and remote monitoring of highly automated agricultural machinery (HAAM) and their safety systems. The standard specifies requirements for starting, movements and tool movements of a HAAM. It also sets test methods for human detection systems.

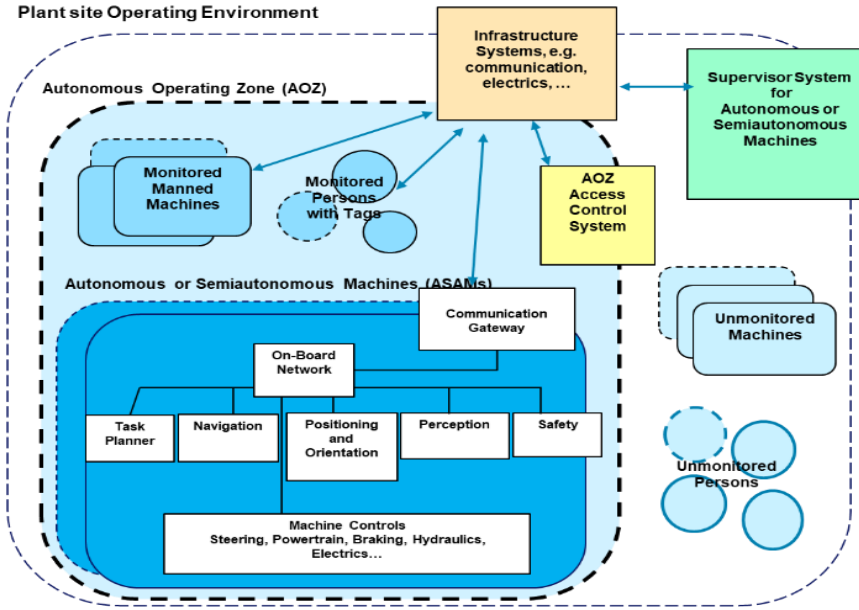


Fig. 2. The difference between the monitored and un-monitored persons and machinery in relation with the safety concept introduced in ISO 17757 [5]

3 Discussion

Autonomy is a systems engineering challenge that need to be solved together with all stakeholders involved. Robust development practices and design tools are needed to develop and verify safe and efficient autonomous mobile machinery systems. Consideration and design of the human – technology - interaction in

system level will become essential in autonomous machinery systems. It is important to understand what can be required from the operator and what safety functions must be automatic. It is not clear how the autonomous machine system should indicate its actions or intentions for the system operators and people working on site [7].

The pressure to use the approach based on on-board safety systems in industrial applications will increase due that the safety solutions in robot cars are based on this type of approach. For the second approach introduced here there is need for innovative production-fitted safety concepts such as adaptive safety functions based on reliable situational awareness information.

Experiences from current automated machinery system show that different modes of operation as well as different safety approaches will be needed in highly automated machinery systems. Fully autonomous modes, remote modes, manual modes and failure modes will need different safety approaches, safety layers and capabilities for independent decision-making to ensure safe operation.

The above mentioned standards ISO 17757, ISO / DIS 3691-4 and ISO / FDIS 18497.2 have their own (different) definitions and different approaches to defining safety requirements for autonomous machinery systems. Operational environments and conditions form a different basis for the requirements. In all of them there seems to be a gap between the requirements set in standards and the state of the art in technology. Another issue is that the standards are for machine, device and component manufacturers. The worksite operators or owners need to be involved when creating an autonomous system and environment, where machines operate.

Acknowledgements The research work on safety of autonomous mobile machinery in VTT has been funded by Business Finland, VTT, FIMA and participating companies.

References

1. Leveson Nancy G. Safety Analysis in Early Concept Development and Requirements Generation. 28th Annual INCOSE International Symposium. July 7-12. 2018. Washington, DC, USA.
2. Malm, T. & Ahonen, T., Safety concepts for autonomous and semi-autonomous mobile work machines. In the proceedings of the 9th International Conference on Safety of Industrial Automated Systems (SIAS 2018), Nancy, France. INRS, 2018, p. 103-108
3. Tiusanen R., Malm T. & Ronkainen A. An overview of current safety requirements for autonomous machines - review of standards. Automaatiopäivät23 conference, 15.-16.5.2019, Oulu Finland. To be published in Automation in Finland 2019 Special Issue by Open Engineering.

4. ISO/DIS 3691-4 Industrial trucks -- Safety requirements and verification -- Part 4: Driverless industrial trucks and their systems. Under development 2018.
5. ISO 17757:2017 Earth-moving machinery and mining -- Autonomous and semi-autonomous machine system safety
6. ISO/DIS 18497.2 Agricultural machinery and tractors — Safety of highly automated agricultural ma-chines — Complementary element. Under development 2018.
7. Tiusanen, R., Heikkilä, E., Malm, T. & Ronkainen, A. System safety engineering approach and concepts for autonomous work-machine applications. In the proceedings of the 2019 World Congress on Resilience, Reliability and Asset Management, Singapore. Future Resilient Systems (FRS), Singapore 2019. p. 144-147

Applicability of systems-theoretic methods in the safety assessment of autonomous port logistics

Eetu Heikkilä, Risto Tiusanen[1]

VTT

Abstract *Increasing autonomy of operations is a major development trend in port logistics. Large container terminals have already automated various parts of their operations. In the future, also smaller terminals aim for increased efficiency using automation and autonomous systems. In such use, the machinery used for container handling needs to be highly adaptive and able to conduct a multitude of tasks in changing environmental conditions. This results in a complex and dynamic operating environment where manual and autonomous machines, as well as humans, may work simultaneously in the same area.*

In this paper, we focus on addressing the systemic safety hazards resulting from the interactions between various actors in the context of a small container terminal. Selected existing systems-theoretic hazard analysis methods are reviewed, specifically covering the following:

- *AcciMap*
- *Functional Resonance Analysis Method (FRAM)*
- *System-Theoretic Accident Model and Processes (STAMP), and the associated STPA hazard analysis method.*

The methods are studied particularly from the point of view of their suitability for addressing the challenges relevant for autonomous machine systems in small container terminals. Based on the findings, we present a preliminary safety assessment approach applicable in the concept design phase of such systems.

[1] Eetu Heikkilä, M.Sc. (Tech.), VTT Technical Research Centre of Finland Ltd, Risto Tiusanen, D.Sc. (Tech.), VTT Technical Research Centre of Finland Ltd.

© Eetu Heikkilä, Risto Tiusanen, 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

1 Introduction

Increasing automation of operations is a major development trend in port logistics. Recent advances have been achieved especially in container terminals, as many large terminals have already automated various parts of their operations. In the future, also smaller terminals aim for increased efficiency using automation and increasingly autonomous systems. In such use, the machinery used for container handling needs to be highly adaptive and able to conduct a multitude of tasks in changing environmental conditions. This results in a complex and dynamic operating environment where manual and autonomous machines, as well as humans, may work simultaneously in the same area.

The new operating concepts enabled by autonomy bring great promises of efficiency increases. However, new safety and security risks also arise. The reasons for risks introduced by autonomous systems include, but are not limited to, the following:

- Increasing system complexity: Autonomy introduces new combinations of technologies, as well as various interactions between humans and technology.
- Increasing software-intensity: Increasing use of software e.g. in perception and decision-making may lead to hazards that are difficult to identify in the development phase. In many cases, software failures can be traced back to system requirements.
- Verification & validation: Testing for all plausible scenarios is a major issue in autonomous systems development.

Partly due to the issues listed above, arguments have been presented against the use of traditional safety analysis methods, which often focus on analysis of component failures or linear chains of events, rather than identifying problems arising from unsafe interactions between system elements (Leveson, 2012). To address the above issues, systems-theoretic approach to safety has been proposed as a potential basis for performing more comprehensive safety analyses.

2 Systems-theoretic approaches

Systems theory is a set of principles that can be applied to comprehend complex systems and their behaviour. Based on systems theory, a number safety hazard analysis approaches, as well as accident analysis methods have been proposed. In literature, comparisons of these methods have been performed mostly in terms of accident analysis (see e.g. Yousefi et al. 2019). However, the applicability of systems thinking and systems-theoretic approaches has not been widely studied in the context of new product development. In the following, three qualitative

systems-theoretic approaches (AcciMap, FRAM, and STAMP) are briefly compared, especially from the perspective of their applicability for supporting the design of autonomous machine systems.

- AcciMap method, originally developed by Rasmussen (1997), provides a structured, graphical representation of a causal scenario. Although designed as a part of an active risk management approach, it has mostly been applied for accident analysis purposes in safety-critical domains.
- FRAM (Functional Resonance Analysis Method), developed by Hollnagel (2012), is based on identification and analysis of system functions. It provides a graphical language for modelling the system functions, and focuses on providing insights on how the functions interact.
- STAMP (Systems-Theoretic Accident Model and Processes), developed in the MIT by Leveson (2012) describes the system as a hierarchical control structure. The STAMP approach is accompanied with the STPA (Systems-Theoretic Process Analysis) hazard analysis method, which aims to identify flaws within the safety controls.

Based on comparative reviews of the methods available in literature (e.g. Karanikas & Roelen, 2019), and considering the characteristics of application in autonomous systems, we can point out some of the advantages and disadvantages related to each approach as shown in table 1.

Table 1. Potential advantages and disadvantages of selected systems-theoretic safety analysis approaches from the perspective of R&D of autonomous machine systems.

Approach	Advantages	Disadvantages
AcciMap	- Provides a clear way of graphically representing accident scenarios.	- Focuses mainly on accident analysis, not at hazard analysis in R&D. - Describes only events and actions, not e.g. system components.
FRAM	- Provides a visual language for describing system functions and the interactions between them.	- Focuses on specific accident scenarios. - Lacks a defined hazard analysis procedure.
STAMP	- Visually describes the system hierarchy. - Provides a defined, systematic hazard analysis process (STPA). - Instrumental for generating design recommendations, as system flaws are directly pointed out.	- Findings are heavily text-based.

3 Conclusions

None of the systems-theoretic approaches presented here have been widely applied to R&D activities of industrial systems. Additionally, their earlier applications in the context of logistic systems are unknown to the authors. Thus, we cannot make conclusions regarding their previous applications in this context.

However, systems-theoretic approaches seem to provide a promising basis for expanding the coverage of traditional hazard analyses, especially for considering the issues typically arising from increasing autonomy. The brief comparison of the three methods presented in table 1 suggests that STAMP/STPA would be most suitable towards R&D activities. AcciMap and FRAM both provide means for system modelling but they lack the systematic hazard analysis methodology, whereas STAMP approach is complemented with the STPA method.

As future work, the application of STAMP approach within a defined container logistics R&D project is planned. Additionally, the application of STAMP in combination with relevant traditional methods will be further studied.

References

Hollnagel, E. (2012). FRAM: The functional resonance analysis method: Modelling complex socio-technical systems. Ashgate Publishing Ltd.

Karanikas, N., & Roelen, A. (2019). The Concept Towards a Standard Safety Model (STASAM v.0). MATEC Web Conf., 273, 02001.

Leveson, N. (2012). Engineering a safer world: Systems thinking applied to safety. MIT Press.

Rasmussen, J (1997). Risk management in a dynamic society: A modelling problem. Safety Science. 27 (2–3)

Yousefi, A., Rodriguez Hernandez, M., & Lopez Peña, V. (2019). Systemic accident analysis models: A comparison study between AcciMap, FRAM, and STAMP. Process Safety Progress, 38(2).

Acknowledgement The research presented in this paper is a part of the AUTOPORT project funded by Business Finland under the Smart Mobility programme

Safety Aspects of Complex Human-Robot Interaction in Healthcare Robotics

Daniel Delgado Bellamy Daniel.Delgadobellamy@uwe.ac.uk

Chris Harper Chris.Harper@brl.ac.uk

Sanja Dogramadzi Sanja.Dogramadzi@uwe.ac.uk

Praminda Caleb-Solly Praminda.Caleb-solly@uwe.ac.uk

Bristol Robotics Laboratory, University of the West of England, UK

Abstract *As our society faces the prospect of an ageing population, with ever greater demands on healthcare and social care and personalised treatments, the body of nursing and care support professionals will experience some difficulties looking for personnel and addressing emerging later life conditions as the life span increases. To prevent this from becoming unmanageable, researchers are investigating the use of robotic technology to relieve the tasks assigned to carers. Ideally, robots will undertake the performance of mundane, physically risky and frequently demanded tasks such as assistance with moving about so that care staff can concentrate on care activities that require human-to-human interaction. For example, those requirements involving significant emotional interaction and support such as recreational activities, conversation or counselling*

1 Introduction

The idea of designing effective, safe and long-term psychologically supportive mobility assistance is highly challenging, requiring complex capabilities and analysis. The capabilities required are detailed below:

- Operation for extended periods with or without supervision from professional therapists such as Occupational Therapy (OT) staff.
- Being in direct but safe physical contact with users for extended periods of time.
- Control of the user and robot as a combined mechanical system; some assistance tasks, such as sit-to-stand or stand-to-sit, require the performance

© Chris Harper, Daniel Delgado Bellamy, Sanja Dogramadzi, Praminda Caleb-Solly 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

of sequences of control actions (poses) to achieve the therapeutically correct protocol. The robot must be able to learn, adapt or develop strategies for unseen instances when such protocols cannot be applied.

- Keep and update a history of the user's profiles. For example, all that comprises cognitive and physical impairments, pain issues, type and level of medication, tendency to become fatigued, clinically advised body poses, heat signatures, heart rate trends over extended time, breathing patterns, and so on.
- Track and store the details about the environment, to which the robot must adapt and consider to fulfill the required human-robot interaction behaviour. Variations can include layout and lighting levels, obstacles and distractions due to objects in the environment, agents with complex behaviour (e.g. people, animals).
- Ensuring a dialogue equivalent to the one between carers and users to maintain trust and consistency of the assistance. A robot may need to be capable of reproducing such dialogue at least to some degree of fidelity, so that users can be reassured and maintain confidence in the process of interaction (loss of confidence may affect their behaviour, inducing deviations in the human-robot physical interaction that cause the very accidents over which they may be so concerned). The robot may need to be capable of natural linguistic communication, as well as other modes of interaction (e.g. display indications), to cope with users of differing levels of capability or impairment.

2 Safety Issues and Hazards

Safety issues and hazards can include:

- Many different and unforeseen modes by which a user can fall over (forwards, backwards, crumpling, keeling over when standing up, clinging onto the robot in desperation, etc.)
- The presence of pets/animals or infants of which the former represents a significant fraction of fall accidents.
- The loss of control if the user releases contact with the robot with one or both hands.
- Presence of hindering issues other than obstacles on the floor, for instance, the user's own clothing (inadequate shoes, dangling clothing, etc.).

- The consideration of physical and cognitive impairment that could alter the user's predicted behaviour abruptly, for example, if they forget something or change their minds.
- Pace matching issues when the robot moves too quickly or too slowly for the user.
- Failure of the user (muscle power loss) or failure of the robot.

The severity of accidents can also be highly variable, ranging from mild (e.g. light bruising) through to more severe injuries such as hip fractures or worse. Risk assessment may need to assess potential hazards based on the distribution of outcomes rather than on single-point values.

3 Deriving Specifications

The system specification process for the robot is complex as the skills of professional occupational therapists in providing physical assistance support are often of a highly specific sensorimotor nature. In other words, they are developed subconsciously by practice over long periods of time, and are not amenable to the linguistic expression that is often required for the development of software or control system specifications.

We argue that although the existence of tasks can be identified by careful dialogue with professionals, through processes such as Focus Groups or Interviews, the exact specification of the required states or behaviour patterns (state trajectories) of a given task cannot be fully captured in this manner. Consequently, the extensive use of machine learning techniques becomes essential to capture this information from user interactions, (co-)supervised by OT professionals who can guide the process.

4 Methodology

This poster describes a user-centred methodology for capturing the complexities of the human-robot interaction indispensable for this class of robotic application. The methodology consists of a mix of:

1. User-centred domain analysis involving dialogue (focus groups, interviews, etc.) with domain subject matter experts (SMEs), who are primarily occupational therapy practitioners working in care establishments such as hospitals and care homes.
2. Use Case Analysis to reduce this domain knowledge into a set of scenarios capturing the high-level generalised goals of the robot.

3. Task Analysis to refine the use cases into specifications for robot functional processes.
4. Machine Learning processes to extract precise models of task behaviour, supervised both by engineers and the domain SMEs to ensure that the models remain plausible and valid.

The poster will review current progress of this process and its effectiveness on the project to date.

Using Task Analysis and Environmental Survey Hazard Analysis to identify requirements of autonomous systems

Chris Harper Chris.Harper@brl.ac.uk

Daniel Delgado Bellamy Daniel.Delgadobellamy@uwe.ac.uk

Chris Harper Chris.Harper@brl.ac.uk

Sanja Dogramadzi Sanja.Dogramadzi@uwe.ac.uk

Praminda Caleb-Solly Praminda.Caleb-solly@uwe.ac.uk

Bristol Robotics Laboratory, University of the West of England, UK

Abstract *Two characteristic features of autonomous systems (distinguishing them from systems that are merely 'automatic') are (1) that they are usually required to perform and achieve complex situated behaviour patterns, and (2) that they are required to perform non-mission tasks as well as the tasks that define their mission or purpose*.

1 Introduction

Automatic systems are suitable for highly structured or constrained environments, in which there is nothing to do other than the task associated with the system's intended mission, or alternatively are supervised or managed by human operators who can take control if necessary. Therefore, the fact that they do nothing else does not present an unmanageable operational problem (such as a safety problem) – as long as the intended tasks are performed correctly, which for safety related applications can be assured with existing design assurance methods. Systems such as domestic central heating systems, washing machines, industrial

© Chris Harper, Daniel Delgado Bellamy, Sanja Dogramadzi, Praminda Caleb-Solly 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

CNC machines, or even more sophisticated examples such as automatic train operation (ATO) systems and older generations of manufacturing robots (which operate inside industrial work-cells surrounded by protective barriers) are all instances of automatic systems. They operate inside well-shielded environments and only perform tasks related to their intended purpose.

2 Autonomous Systems

In contrast, autonomous systems are generally conceived as operating entirely without the possibility of human intervention and, in addition to performing their design mission, must survive in unbounded environments containing features that require interaction for the purpose of maintaining safety but are not directly associated with the intended mission. Systems such as unmanned aircraft, driverless cars, or mobile robots all have to operate in environments in which there is significant variation of situations induced by features of the environment that are not directly associated with their mission; these systems are autonomous, at least to some degree or 'level'. In the project within which we have been applying the methods defined in this paper, we are developing the requirements for a mobility assistance robotic system, which assists elderly or frail people to move around in residential environments, such as domestic or nursing homes, or hospitals.

3 Automatic to Autonomy

Previous generations of automatic system have typically been required to perform simple tasks to a high degree of precision, fidelity, accuracy, and so on. As the move from basic automation into outright autonomy has progressed, the tasks required of robotic and autonomous systems has increased towards complex interaction patterns (trajectories) which are only specified indirectly (as task goals, not exact trajectories), or which evolve over time thus requiring the system to adapt to changing circumstances. Other complex characteristics, such as ethical behaviour, also become essential as well as the basic achievement of function. And by their very nature, autonomous systems usually operate in noisy and unpredictable environments, so resilience is also an essential safety requirement.

4 Design Methods

Design methods are needed to specify complex situated behaviour in an abstract manner, such that operational characteristics such as safety, resilience, and performance (stability, rate of convergence, etc.) are captured adequately. This paper focuses on Task Analysis methods, and describes our application of some variations of Hierarchical Task Analysis (HTA) [Ref.1]. The method was originally developed for Ergonomics analysis and design, to identify and design user interactions with a system. In this paper and previous references, we have been applying the technique as a means of specifying the situated behavior required of an entirely autonomous system. This has necessitated the use of HTA extensions proposed by Huddlestone and Stanton [Ref.2] to address non-linear sequencing of tasks, and the Goals-Means Task Analysis (GMTA) method of Hollnagel [Ref.3] as a scheme for capturing the specification each individual task.

5 Hazard Analysis

Traditional methods of preliminary hazard analysis do not encourage the identification of non-mission interactions and their potential hazards. The design perspective taken by HAZOP, FHA, and other techniques tends to be internally focused, looking only at the effect of deviations of internal mechanisms or at boundary interfaces (e.g. human errors) on the intended mission. Only by chance or by imaginative insight do non-mission interactions get identified.

6 ESHA Methodology

To overcome this methodological deficiency, we have developed a variant of preliminary hazard analysis called Environmental Survey Hazard Analysis (ESHA). This method sets up an externally focused analysis perspective explicitly, thereby ensuring that such aspects are analyzed systematically rather than leaving their discovery to chance. The ESHA methodology is still under development, and is being matured as it is applied to design problems. The initial studies [Ref.4] were based on a somewhat contrived design problems developed for workshop-type exercises or as a feasibility check of the method. In this poster/paper we review the first use of the extended HTA and ESHA methods to a full-scale application, the design of the aforementioned autonomous robot for mobility assistance and support for elderly or frail people. We present updates and refinements to these methods, based on the experience gained with the application.

References

[Ref.1] Kirwan & Ainsworth, A Guide to Task Analysis, Taylor & Francis 1992, ISBN 0-7484-0058-3

[Ref.2] Huddlestone J.A., Stanton N.A., New graphical and text-based notations for representing task decomposition hierarchies: towards improving the usability of an Ergonomics method, Theor. Iss. in Erg. Sci., Vol.17 Nos.5-6, Taylor & Francis 2016

[Ref.3] Hollnagel E., A Goals-Means Task Analysis method, originally published in an ESA-ESTEC project report, dated 1991; URL: http://erikhollnagel.com/onewebmedia/GMTA.pdf

[Ref.4] Dogramadzi S., Harper C. et al, Environmental hazard Analysis – a Variant of Preliminary Hazard Analysis for Autonomous Mobile Robots, Jrn. Intell. Robot Syst. (2014) 76:73-117, Springer

A Gamified Prototype Design for Software Safety Requirements Engineering

Helen Partou, Vito Veneziano, Trevor Barker & Catherine Menon

University of Hertfordshire

Abstract *Communication gaps remain a challenge for stakeholders involved in software requirements engineering, particularly when eliciting and refining safety requirements. These are difficult to express and elicit using standard techniques such as CRUD-based (Create, Retrieve, Update, Delete) methodologies. This difficulty can manifest across contractual boundaries and act against effective validation. For instance, the use of ambiguous and emotive language when discussing safety requirements can increase the complexity of these requirements, compromise software safety and lead to costly revisions. This paper presents the design of a gamified prototype, which aims to explore whether these challenges for eliciting safety requirements can be minimised via the use of a competitive collaboration technique. The prototype's design allows stakeholders to document and manage requirements through agile-based user stories. It includes customisation of De Bono's 'Six Thinking Hats' from the field of cognitive psychology as a mechanism for gamification, and an emotive word bank based on the OCC (Ortony, Clore, Collins) 'Model of Emotions' to support stakeholder communication around safety.*

The MSCA ETN Safer Autonomous Systems project

Davy Pissort

KU Leven

Abstract *SAS - SAFER AUTONOMOUS SYSTEMS: The coming of autonomous systems doesn't just mean self-driving cars. Advances in artificial intelligence will soon mean that we have drones that can deliver medicines, crew-less ships that that can navigate safely through busy sea lanes, and all kinds of robots, from warehouse assistants, to search-and rescue robots, down to machines that can disassemble complex devices like smartphones in order to recycle the critical raw materials they contain. As long as these autonomous systems stay out of sight, or out of reach, they are readily accepted by people. The rapid and powerful movements of assembly-line robots can be a little ominous, but while these machines are at a distance or inside protective cages we are at ease. However, in the near future we'll be interacting with "cobots" – robots intended to assist humans in a shared workspace. For this to happen smoothly we need to ensure that the cobots will never accidentally harm us. This question of safety when interacting with humans is paramount. No one worries about a factory full of autonomous machines that are assembling cars. But if these cars are self-driving, then the question of their safety is raised immediately. People lack trust in autonomous machines and are much less prepared to tolerate a mistake made by one. So even though the widespread introduction of autonomous vehicles would almost eliminate the more-than 20,000 deaths on European roads each year, it will not happen until we can provide the assurance that these systems will be safe and perform as intended. And this is true for just about every autonomous system that brings humans and automated machines into contact.*

A ‘Z’ specification of the concepts of Data Safety Assurance

Divya Atkins

MCA Ltd

Abstract *The SCSC 'Data Safety Guidance' provides a framework and a process for assuring the safety of modern data-driven computer systems providing safety-related functionality. In this poster we use the ‘Z’ notation to capture the essential content of the existing SCSC Data Safety Guidance. The formal specification of the entities and the relationships between them will make it easier for a practitioner to understand the Guidance and to apply it to real projects.*

Formalising the Language of Risk

(Full paper)

Dave Banham[1]

Abstract *The use of natural language in engineering is often problematic due to assumed meanings and usage contexts of domain terms with the result that misunderstandings arise and uncertainty abounds. Somewhat ironically, this language uncertainty is just as present in systems (and project) engineering risk management. In this paper, a formalised structure of words – an ontology – is proposed by which, at least, the risks arising from system safety and security concerns can be described.*

1 The Problem

The SCSC Data Safety Initiative Working Group (DSIWG) found that the language of safety and risk is contextually dependent with terms having multiple meanings, different terms being used with the same implied meaning, and, as is often the case in English, terms being used without qualification. Add to this situation the risk terminology used by cyber security professionals, because *security informed data safety* is a useful adjunct, and the result is a long glossary of terms with no self-consistency. It is therefore hard to produce guidance for data safety that is clear and accessible. Moreover, often this variability of meaning, introduces subtle misunderstandings that take conscious effort to detect and resolve.

Take the word "risk" as an example. In common usage, "risk" means an activity or situation that has a chance of a significantly unpleasant outcome in the worldview of the observer making the statement. Risk is generally used as the adjective "risky" to qualify the sense of uncertainty being expressed about the named activity: *parachute diving is risky*; *driving fast is risky*; *betting on slot machines is risky*; etc. Risk can be used as a noun when conceptualising it: *I accept the risks involved in free climbing*; *the risk of injury in rugby is high*. The

[1] Dave Banham can be contacted at dave.banham@gmail.com

© Dave Banham 2020.
Published by the Safety-Critical Systems Club. All Rights Reserved

two forms can be combined to yield sentences such as: *there is a risk of injury from risky driving*. However, note how easily the implied meaning of risk shifts from one of uncertainty in outcome (when used as adjective), to that of likelihood (when used as noun). Moreover, the outcomes and likelihoods that are often inexplicitly stated as an assumed shared understanding, are significantly undesirable in the worldview of the person making the statement, but may be considered otherwise by somebody else. Free climbing is one person's *horror story*, but another's *pleasant sport*. The common language use of the word "risk" is completely inadequate for engineering where assumptions need to be eliminated in preference for precise terms, calculations, and a shared (and agreed) worldview.

For this reason, engineering makes use of standardised terms. The international standard for risk management is ISO 31000 [1], with the compendium ISO vocabulary of risk terms, ISO Guide 73 [2].

ISO Guide 73 defines risk as *the effect of uncertainty on objectives of stakeholders*. Objectives are things that stakeholders seek or want to avoid. We don't want harm to arise from the use of our goods and services; conversely, we want to make money from selling our goods and services. Uncertainty exists in the fulfilment of these objectives due to phenomena such as natural processes, unforeseen circumstances, competition, etc. (See figure 1) [1] When things are certain (perhaps because they have already happened), there is no risk.

This leads to the idea of positive risk (*seeking a benefit*) and negative risk (*avoiding a harm*). The common use of "risk" is in the sense of negative risk; harmful (often physical) situations that need to be avoided. However, risk arises from the worldview, frame, or context that the stakeholder has since they own the objective. (See figure 3.) Consider theft. The owner of a valuable asset wants to protect that asset from, amongst other things, theft. Theft results in a harm that creates a loss, to the owner, of the stolen asset and is thus a negative risk concept to the stolen asset's owner. Whereas to the criminal, theft is the means by which value is gained (a benefit) and is thus a positive risk concept to them, notwithstanding the negative risk of being caught.

To bring this discussion back to safety and especially data safety, the challenge was to address data safety in an industry sector neutral way where some sectors have established functional safety[2] standards and regulations, and others where there is very little maturity of safety engineering practice. For these reasons, the group used the generic framing of risk management, as set out by ISO 31000 [1], on the basis that even where a functional safety standard is in use, it is implicitly using a safety risk management driven approach. Our use of ISO 31000 is however contextualised to that of systems (especially cyber systems) and security informed system safety. (See figure 2.)

[1] The figure references relate to a formal ontology of risk terms that is introduced further on in the text.

[2] Note: Functional safety concerns the provision of active (as opposed to passive) methods of achieving safety.

2 Articulating Risk

The language of data safety is formalised around that of risk. If a situation exists where data can contribute to a system failing such that a safety-related harm results, then there is a data safety-related risk.

Let us start by defining "harm". Harm is the *consequence* of a *failure* to meet stakeholder objectives when the consequential situation is *undesirable* to them. The converse is a "Benefit", when the *consequence* is *desirable* to them. (See figure 1.) A subset of the total set of possible harms is the set of safety-related harms. A safety-related harm is generally defined as a physical harm that impacts the health or life of a person or persons, or impacts the wellbeing of the natural environment. (See figure 5.) Although a stakeholder may include other impacts such as the loss of an asset, loss of reputation, etc. in their definition.

How can harms arise? Since harms are generally not certain, specific situations need to occur to allow them to arise as a *consequence*. These causal situations are referred to as *incidents*. An incident is a *dangerous event* (i.e. a moment in time) and is therefore a *danger source*. (That is, danger may lead to harm.) A near miss is an incident that did not lead to harm, but had the potential to do so. An accident arises from an unintentional incident that leads to harm; that is, an accident arises from unintentional sources of danger. Incidents can be intentionally created and sometimes maliciously; for example, by arsonists, thieves, or by misguided misuses of a system (i.e. by incompetent users). As such, the term "incident" is more useful than purely safety terms such as "accident", as it allows the safety analysis to consider a wider set of concerns that have traditionally been, for example, the reserve of security specialists. (See figure 4.)

One purpose of a system safety analysis is to theorise about what potential harms a system may cause and to identify the potential danger sources that may lead to them. An identified danger source is called a *hazard*; a hazard is a known danger source that may lead to an incident that causes harm. (See figure 6.)

A risk score is a metric arising from a function of a potential incident's likelihood and the desirability of the potential outcome. Hence, numerically:

$$\text{risk} = \text{likelihood} \times \text{desirability}$$

where *likelihood* is a probability of occurrence, *desirability* is a positive score when the objective is sought and a negative score when it is to be avoided, and where × is a binary operator (i.e. a function) taking two parameters. Hence, a negative risk score indicates the risk of a harm, which conforms to the ISO 31000 framework. In the context of harms, desirability is often stated as a severity score and the equation of negative risk can be stated as:

$$\text{risk} = -(\text{likelihood} \times \text{severity})$$

To understand how harm may arise we either start with a harm and ask the deductive question of *how can it arise*, or we start with some other aspect of the system such as a system input, or a subsystem and ask the inductive question of *what would happen if*. Where a weakness or susceptibility to failure is found in a constituent part of a system (which also includes people when they form an active part of the system) then a *vulnerability* is said to exist. A danger source is something that can exploit a vulnerability to create an incident. An incident that is a system failure can result in harm, although typically what happens is that the incident is a failure that is more localised to a constituent part of the system; that is, a part no longer completely fulfils its objectives. (See figure 2.) A localised failure manifests as a fault (failure condition) that can propagate through a system exploiting other vulnerabilities (that is, triggering other failures) until potentially the system fails with some harmful outcome. (See figure 4.)

Risk management consists of several activities, but those pertinent to the previous discussion are those of:

- Risk Identification
- Risk Analysis
- Risk Evaluation
- Risk Treatment

A top-down risk identification requires the identification of the potential system incidents that could result in harm. As noted previously, the class of all possible harms can be subdivided in to various subclasses of which safety-related harms is just one such subclass. However, because incidents that lead to safety-related harms are also called hazards, this specialised form of risk identification is generally referred to as a hazard analysis. Nevertheless, the general definition of *hazard* is that of a foreseeable danger source, which covers the broader definition of harm. (See figure 6.)

A top-down risk identification is generally performed early in a project's life cycle to provide a clear indication of the potential incidents and consequential harms the system may be liable to create.

Risk analysis provides a more detailed understanding of the nature of risk and its characteristics. ISO 31000 summarises this as "Risk analysis involves a detailed consideration of uncertainties, risk sources, consequences, likelihoods, events, scenarios, controls, and their effectiveness. An event can have multiple causes and consequences and can affect multiple objectives."

Risk evaluation considers the understanding of risk arising from the risk analysis and determines what needs to be done (but not how). This can range from requiring more risk analysis, to triaging and prioritising the treatment of risks, to actively ignoring risks.

Risk Treatment requires the selection and implementation of *controls* for addressing risk. ISO 31000 outlines this as the iterative process of:

1. formulating and selecting risk treatment options;
2. planning and implementing risk treatment;
3. assessing the effectiveness of that treatment;
4. deciding whether the remaining risk is acceptable;
5. if not acceptable, taking further treatment.

In the context of system safety, the top-down risk identification should identify safety-related risks (that is, risks related to potential safety-related harms). This then leads to safety requirements as the primary means of controlling the associated risk to an acceptable level. However, as it is well known that design and technology choices have associated vulnerabilities, a further more detailed "bottom-up" inductive risk identification is required on the design (or design options when they are being evaluated). From this, we arrive at a suitable level of design risk control such that the system can be assured to be acceptably free of design borne risk; that is, the design does not contain vulnerabilities that could lead to incidents that result in harm. (See figure 8.)

3 A Model of Risk Terminology

We can describe this terminology formally in an ontology and use UML class diagrams to represent aspects of that ontology through a series of diagrams. The ontology captures the ISO 31000 concept of desired and undesired stakeholder objectives through the consequences of benefit and harm. However, from a safety and security point of view, our main interest is in the risks associated with harms and, as a result, the ontology is significantly more refined in this area. Nevertheless, it is important to understand the opportunity and benefit cases that malicious threat actors may have towards a system.

The following diagrams and their associated descriptions provide an illustration of some of the ontology that has been constructed and focuses around the risk terms that have been introduced in the preceding sections.

Our experience with the development of this ontology has been challenging. The complexity of language is revealed and the precise meaning of terms in relation to the other terms has been, at times, difficult to define. Out of brevity, the definitions of each term have been omitted and we instead call upon your understanding of the words, in the context of the provided diagram descriptions. You may find as you study these descriptions and their associated diagrams that our use of the terms conflicts with your normal use of them. What we ask is that you consider both the context, as previously outlined, for the construction of our ontology and its purpose of defining a self-consistent unambiguous domain language for security informed safety risk analysis. If something is amiss, please let us know. An introduction to the UML class diagram notation for ontology modelling is provided at the end of this paper.

In figure 1 **consequences** impact **objectives** of **stakeholders**. A **consequence** is the outcome of a **situation** that impacts (affects) **objectives** of a **stakeholder**. **Consequences** can be good (objectives were met) or bad (objectives were not met).

This leads to the idea of **stakeholder** *desirability* of the **situations** that impact their **objectives**. Since *desirability* can be for **situations** that are sought, as well as for **situations** that are to be avoided, we introduce the idea that *desirability* is a metric where positive values describe **consequences** that are desired and negative values describe **consequences** that are to be avoided. We call the former a "**benefit**" and the latter a "**harm**".

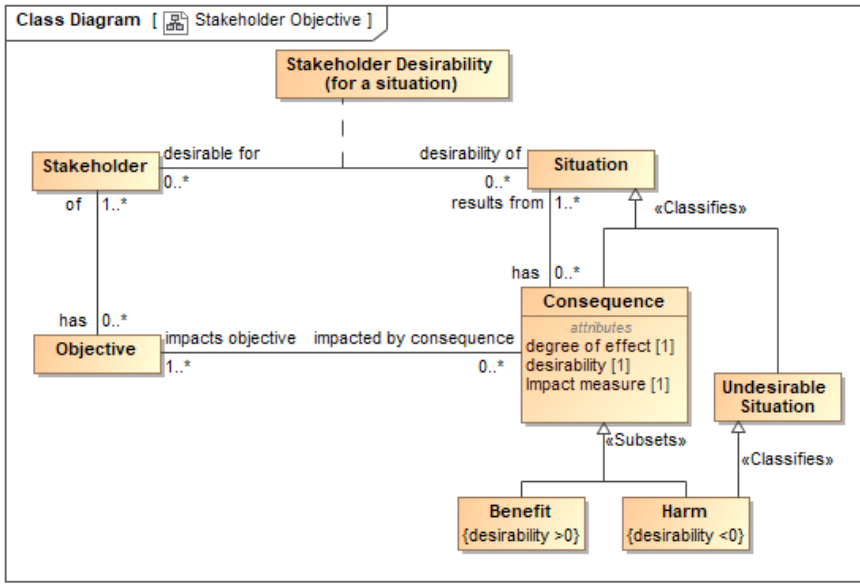


Fig. 1. Stakeholder Objective related terms

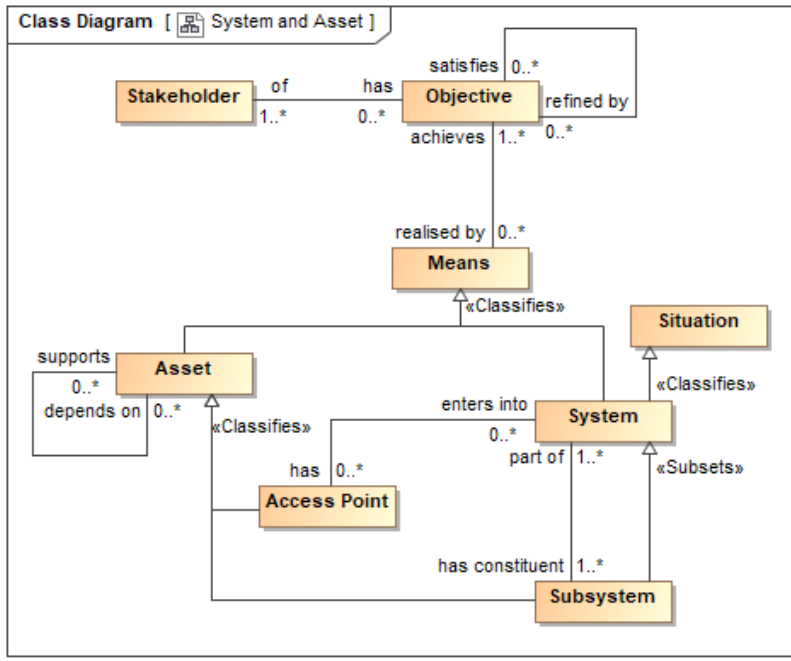


Fig. 2. System and Asset related terms

In figure 2, whilst there are many definitions of what **system** means, the essential concepts required for risk management are that a **system** is an abstract concept for the composition of one or more **subsystems**. As such, **subsystems** are treated as tangible **assets**. Systems interact with their environment via **access points** (that for product systems can be thought of as being **interfaces**). Consequentially, **access points** are also treated as **assets**. Moreover, since a **subsystem** is also a **system**, a **subsystem** can be further composed out of **subsystems** recursively.

Systems and **assets** are the **means** by which **stakeholders** achieve their **objectives**. Furthermore, since, in general, a **situation** is a particular arrangement of entities in a defined time interval, and a **system** is an arrangement of its constituent **subsystems**, a **system** is hence a particular kind of **situation**.

Since **risk** is defined in terms of a **stakeholder's** uncertainty in their **objectives**, the above model allows us to apply a risk based approach to the black box specification of a **means** as well as to the white box definition of a **system** (with its **subsystems** and **access points**), and to do this recursively.

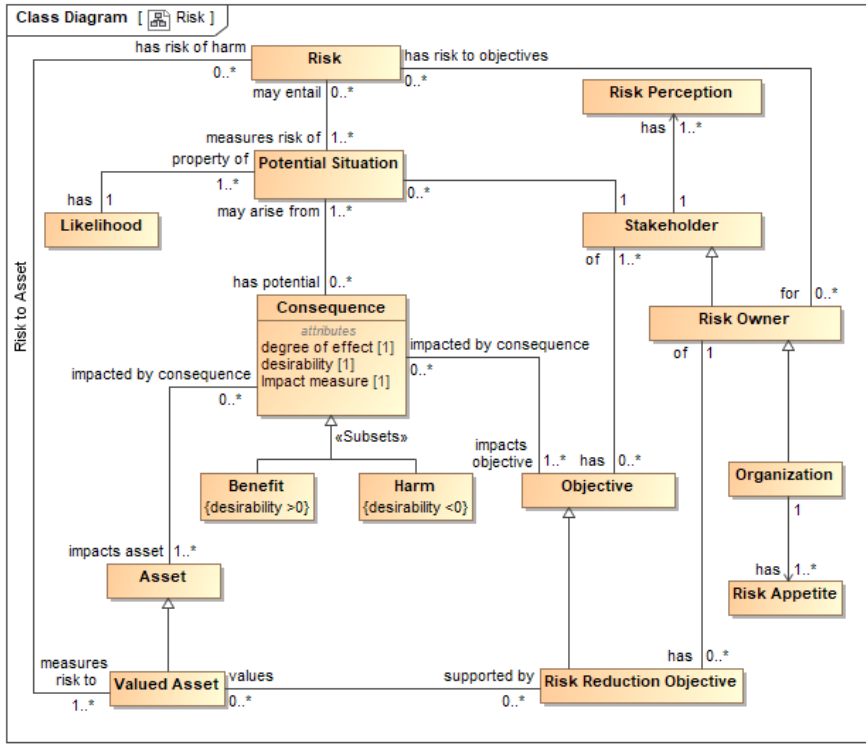


Fig. 3. Risk related terms in the context of the impact of Potential Situations on Stakeholder Objectives

In figure 3, **risk** is a measure of the uncertainty in attaining **stakeholder** objectives. **Objectives** are things that **stakeholders** want or do not want to happen, which are not certainties. **Risks** are contextualised against the **asset** or assets that are impacted by the **objectives**, given the *likelihood* of the **potential situations** that may arise from them and the *desirability* of the **situation** that may arise as **consequence**. Such **consequences** can be *desired* or not *desired* and we call this a **benefit** and a **harm** respectively.

In terms of managing risk, there needs to be an identified **risk owner** that has **risk reduction objectives** that relate to the subset of **assets** that are considered to be of value (i.e. **valued assets**). This corresponds with the pragmatic view that the formulation of a **risk treatment strategy** needs to be targeted to be cost effective.

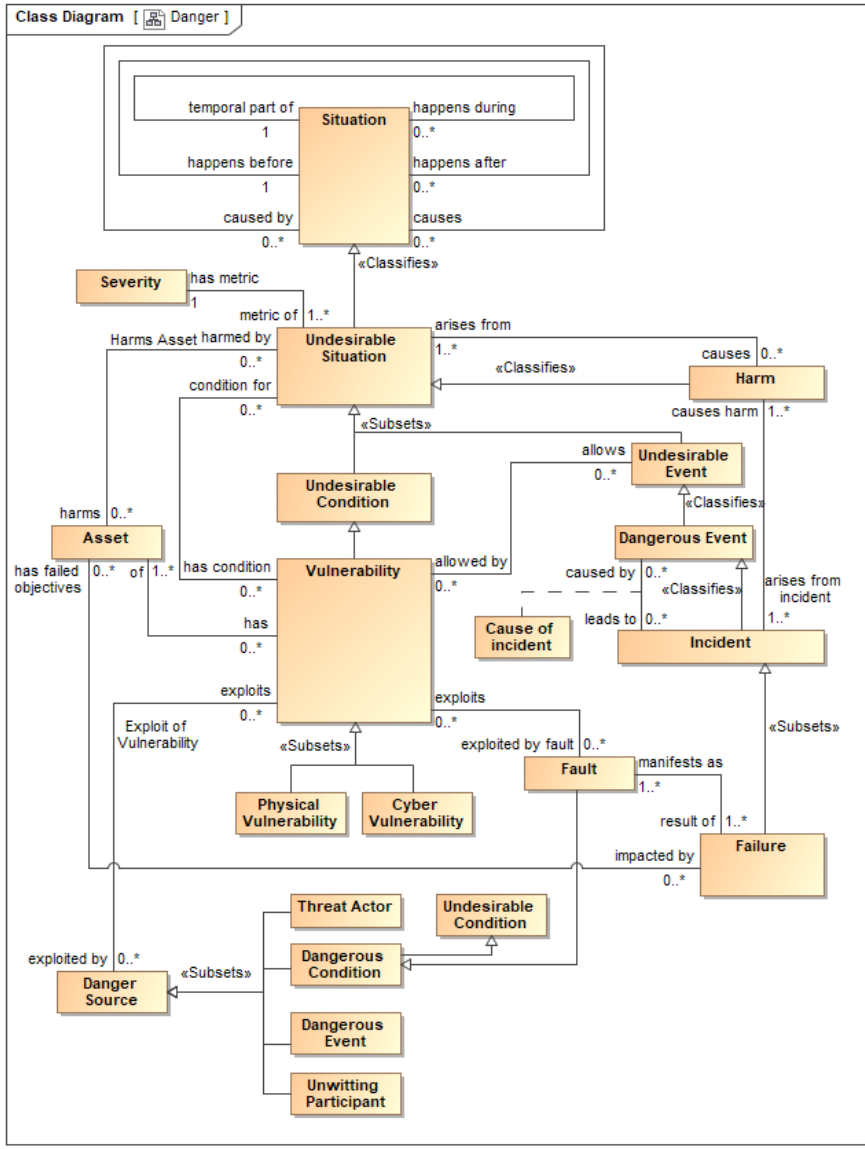


Fig. 4. Danger arises from Undesirable Situations that have the possibility of causing Harm

In figure 4, danger is described by the *possibility* that an **undesirable situation** may cause **harm**, as denoted by the relationship between these two terms. More specifically, we can state that **harm** arises from **incidents** that cause it; an **incident** is the cause and **harm** is the **consequence**.

An **incident** is both a **danger source** (a **dangerous event**) and an **undesirable situation** (an **undesirable event**). Hence, the *possible* relationship between **undesirable situation** and **harm** becomes a substantiated one between **incident** and **harm**.

The degree of *danger* posed by an **undesirable situation** is captured by its **severity** property.

Assets can have **vulnerabilities**, where a **vulnerability** defines the conditions that *allow* an **undesirable event** to occur. A **danger source** is the generic term for something – natural, systematic, or intentional – that can *exploit* a **vulnerability** to create an **undesirable event**. Since **undesirable events** can be classified as **dangerous events**, which is a **danger source**, a chain of events can be created whereby a series of **vulnerabilities** are exploited until an **incident** occurs and **harm** arises.

The term "exploit" is used here with both its "use" and "abuse" meanings. Physical things have **physical vulnerabilities** that are subject to the laws of physics and particularly the law of entropy; physical things break. They break through the *wear and tear* of natural use, and they break by being abused and misused. Complex systems have both physical vulnerabilities and vulnerabilities arising out of design limitations and design flaws (i.e. systematic defects). In computer-based systems, these design vulnerabilities are called **cyber vulnerabilities**.

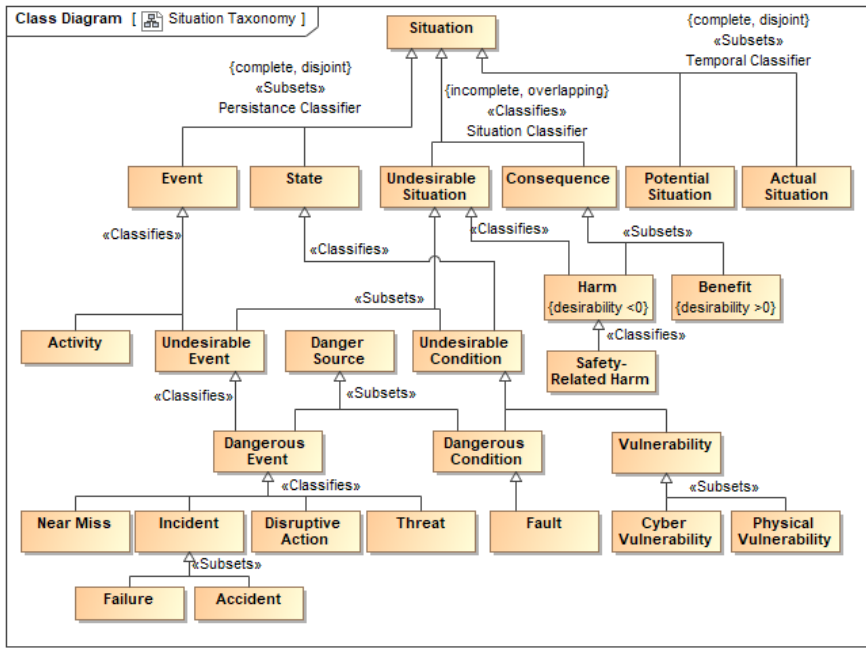


Fig. 5. Taxonomy of Situation terms

For a **system** that is composed out of a hierarchy of **subsystems**, a localised **incident** in a **subsystem** results in the **harm** of that **subsystem** not fulfilling its intended purpose. Here we use the term **failure** rather than **incident**. The manifestation of a **failure** is a **fault** and since a **fault** can create other **failures** (i.e. fault propagation), we can consider that a **fault** is itself a **danger source** and hence **faults** *exploit* **vulnerabilities**.

Figure 5 shows a taxonomy of the **situation** terms in the ontology. The three top-level divisions of **situation** provide three separate and overlapping primary classification schemes for it, covering temporal classification, persistence classification, and sub-types of situation. This allows classifiers from the three divisions to be freely combined; we can thus have **potential** or **actual situations** that are **events** (occurrences in time) or **states** (persistent conditions in time). Some useful combinations are shown in figure 5, such as **undesirable event**, but given the number of possible combinations, it is impracticable to show them all. From a risk analysis point of view, we are concerned with the possible set of **potential situations** that may arise given the **actual situation** we find ourselves currently in. From an **incident** investigation point of view, we are dealing with the **actual harm** (which is both a **consequence** and an **undesirable situation**) and trying to determine the sequence of **undesirable situations** (states and events) that gave rise to the **actual incident** that resulted in it.

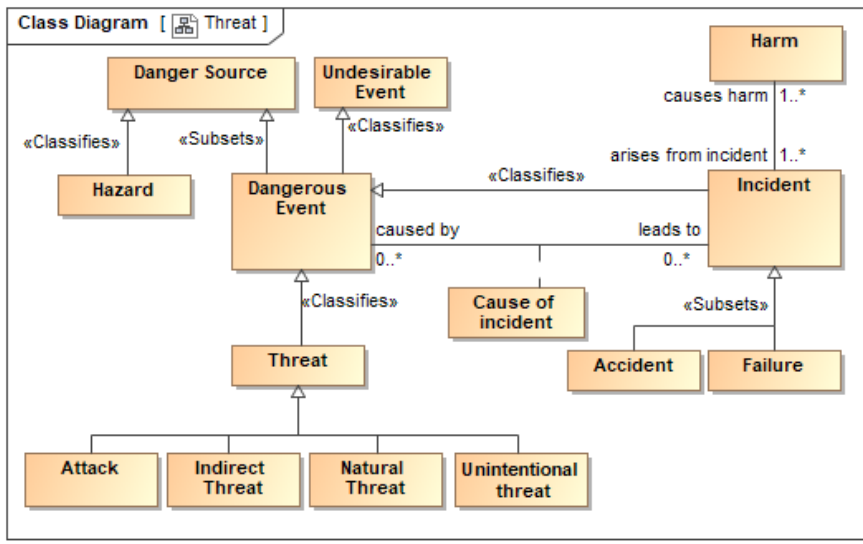


Fig. 6. Taxonomy of Threat related terms as classifiers of Dangerous Events that may lead to Incidents

In figure 6, **threat** is shown as being a **dangerous event** that can happen to bring about an **undesirable situation**, which is called an **incident**.

A taxonomy of **threats** is proposed.

Attacks are malicious and are perpetrated by a **threat actor**. (See figure 7.)

Unintentional threat, **indirect threat**, and **natural threat** are what might be considered to be *accident sources*; that is, non-malicious, non-intentional, threats.

Since **threat** is a form of **dangerous event**, we can see that this term covers both accidental sources of danger and malicious sources of danger.

A **hazard** is defined as a known **danger source**. One aspect of risk identification is to determine the sources of danger that could lead to harm, and for this reason, such methods are often referred to as hazard assessment or hazard identification.

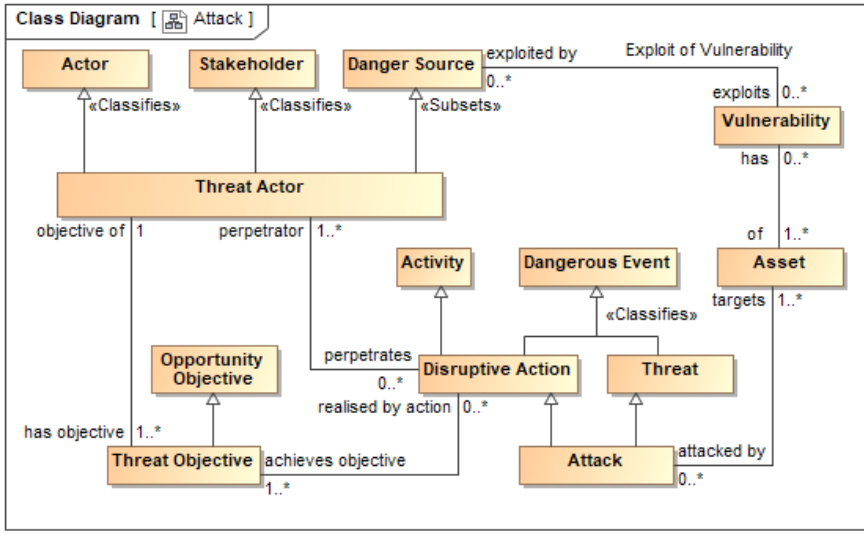


Fig. 7. Attack related terms contextualising Threat Actors perpetrating Disruptive Actions to achieve Threat Objectives

In figure 7, a threat actor is both an actor and a danger source.

A threat actor has some (1 or more) threat objectives that are their opportunity objectives.

A threat actor perpetrates disruptive actions, which is a dangerous event that may *exploit* the vulnerabilities of assets. (See figure 4.)

An attack is a disruptive action that is also a threat. Attacks target assets and exploit their vulnerabilities.

In figure 8, a **risk treatment strategy** gives rise to **risk treatment objectives** that may require a **risk control** to be implemented to achieve the desired changes in the parameters of the target risks that results in the actual level of **controlled risk**.

A **risk treatment strategy** defines the framework and targets for the treatment of risks within a defined scope of application of the organisation's overall risk

management plan. **Risk treatment strategies** can therefore be specifically targeted in the context of individual risks, whilst also setting out the strategies required for conformance to organisational and external standards and practices, and regulatory target.

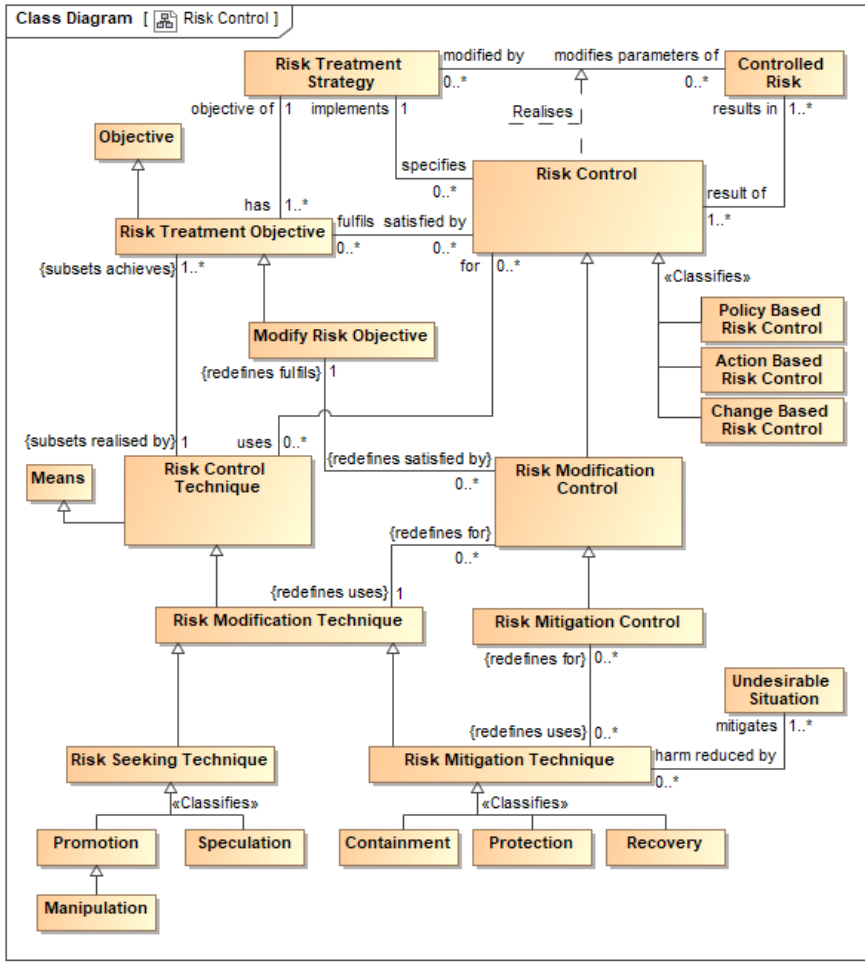


Fig. 8. Risk Control and Risk Mitigation related terms

A **risk treatment objective** defines what risk treatment is required for a specific risk. There are five main forms of **risk treatment objective**:

- **Avoid risk** – objective to avoid the source of the risk to lower the likelihood of occurrence;
- **Accept risk** – objective to take the risk;

- **Modify risk** – objective to modify the parameters of the risk, generally by objectively modifying the situations that affect these parameters;
- **Transfer risk** – objective to transfer the impact of the risk's occurrence to another, e.g. insurance;
- **Retain risk** – objective to retain a *residual risk*,[3] typically arising when no further worthwhile actions can be devised.

A **risk control** defines how a risk treatment objective will be satisfied and remains satisfied for the life of the associated risk. Given the different forms of risk treatment objective and the individual circumstances of each risk, only a broad classification of risk control methods can be outlined. Moreover, actual risk controls will typically be a mix of these methods, which in themselves are not intended to be a definitive list.

Risk controls can be:

- **Policy based** – aim to constrain risk-taking and risk exposure, use risk-related indicators and risk measures; define risk decision authorities and limits; trigger alternative control procedures or escalation, and influence decision-making in other ways;
- **Action based** – actions, such as active checks and decisions, performed regularly or in response to events;
- **Change based** – changes to plans, procedures, designs, etc.

A specific subset of **risk controls** are **risk modification controls** that fulfil **modify risk objectives**. A specific subset of **risk modification control** is a **risk mitigation control**, which seeks to reduce the consequential impact on assets of an actualised risk.

A **risk mitigation technique** provides a **means** for **risk mitigation control**. **Risk mitigation techniques** can be classified as follows:

- **Protection** techniques seek to detect the onset of potentially undesirable situations and then react with a countermeasure means to prevent harmful consequences.
- **Recovery** techniques seek to bring a failed or harmed asset back to non-failed or non-harmful state.
- **Containment** techniques seek to lessen the impact of a harmful (or potentially harmful) situation.

The converse to **risk mitigation techniques** are **risk seeking techniques**:

- **Promotion** techniques are used when the uncertainty in attaining desirable objectives (opportunities) needs to be improved. However, not all stakeholders hold the same point of view and the exploitation of a **vulnerability** is a

[3] A *residual risk* being a risk that has been controlled down to a residual level of risk that the organisation is prepared to accept. C.f. accept risk objective where no risk control is applied.

manipulation, which is a type of risk **promotion** from the **threat actor's** point of view.

- **Speculation** techniques attempt to improve the outcome (i.e. the impact) by investing in the **means** to improve that outcome. For example, by investing in preliminary work that is in itself of limited direct value, the benefit in achieving the primary objective can be increased for a positive outcome and the cost of failure reduced for a negative outcome.

4 Conclusion

This paper has set out to explain the core language defined in the Risk Ontology that the Data Safety Initiative Working Group has assembled. The language is self-consistent (by virtue of the ontology formalism), and provides the means for describing causal situations that can result in harm to assets. The power of expression in the language is aided by the ontology classification meta-language as it allows terms to inherit higher order concepts. An example being that whilst *harm* is a *consequence* that results from some other *situation*, *harm* is also a *situation*. This allows causal modelling to show, for example, how harms can propagate. For example, a fire in a bin (a localised harm) spreads (due to lack of adequate containment and/or proximity to other combustible materials) to destroy the building (a larger scale harm). Moreover, the causality modelling afforded by the situation related terms can be used in incident investigation (i.e. after the fact) where evidence is being assessed to determine why and how something occurred.

The ontology attempts to find common ground between the safety and security risk analysis by using unified terms such as "incident" and "vulnerability". The language described may not cover all aspects of safety or security analysis, but we hope that it provides enough common language to enable greater productivity in achieving security informed system safety.

The ontology has been extended to derive a sizable amount of data safety terminology, but in doing so we have found some issues with the existing terminology in the current Data Safety Guidance (version 3.1) that will need to be reconciled as the DSIWG works towards a version 4 release.

Work is also required to ready the Risk Ontology for a formal publication. One area of significant outstanding work is the need to reach agreement on a single definition for each term and, as such, achieve a self-consistent set of definitions.

Acknowledgements

The author would like to acknowledge the significant work of the "Threat and Risk Community"[4] in creating an ontology that paved the way for the creation of our own Risk Ontology. In that respect, our ontology shares many of the same concepts and terms as the Threat & Risk Ontology, although ours is smaller, in part because it lacks some of the foundational terms that the Threat & Risk Ontology formally defines. Moreover, we have aligned our ontology around the ISO 31000 terminology.

The author would also like to thank the following people for their contribution to this paper and to the Risk Ontology: Paul Hampton, Divya Atkins, Martin Atkins, and Mike Parsons.

Lastly the work of the SCSC WG is gratefully acknowledged.

References

[1] "Risk Management - Guidelines," ISO 31000, 2018.

[2] "Risk Management - Vocabulary," ISO Guide 73, 2009.

[3] "Information technology - Object Management Group Unified Modeling Language (OMG UML), Superstructure," ISO/IEC 19505-2, 2012.

Appendix A UML Class Diagram Notation Guide for Ontology Models

Figure 9 shows the graphical notation subset of UML 2 [3] class diagrams that are used to model the risk ontology. The rectangular shapes within the diagram frame represent classifiers, which are used to describe the language terms in the ontology. The *is-a* relationship denoted by the hollow closed arrow headed edge (▷) describes a taxonomical relationship between classifiers where the specialised classifier is a subset concept of the more general classifier at the arrowhead end of the edge. The ontology optionally clarifies the specialisation with an annotation text shown with double angle quotes next to the generalisation edge:

- «Classifies»
- «Subsets»

A «Classifies» relationship denotes a set of specialisations that can be used in an additive fashion; that is, they are overlapping and additive concepts. Whereas

[4] See http://threatrisk.org/drupal/ (accessed 25 October 2019)

a «Subsets» relationship denotes a set of specialisations that are distinct from each other; that is, they are non-overlapping concepts.

For example, a vehicle can be classified by its means of its source of power, its means of motion, and its colour (to name just a few). Each of these classifiers can be subset, so for example for power, we could have diesel engine, electric engine, gas turbine, etc., and for the subsets of means of motion we could have, wheels, wings, hull, etc., and for colour some set of colours. From this set of classifiers and their subsets we can describe a vehicle as red, with diesel engine and wheels, or as red with gas turbine and wings. The classifiers are additive and individually describe an aspect of the thing (a vehicle in this example) they classify. They also provide discrimination by class, so in this example all the red vehicles can be identified, irrespective of their other classifiers.

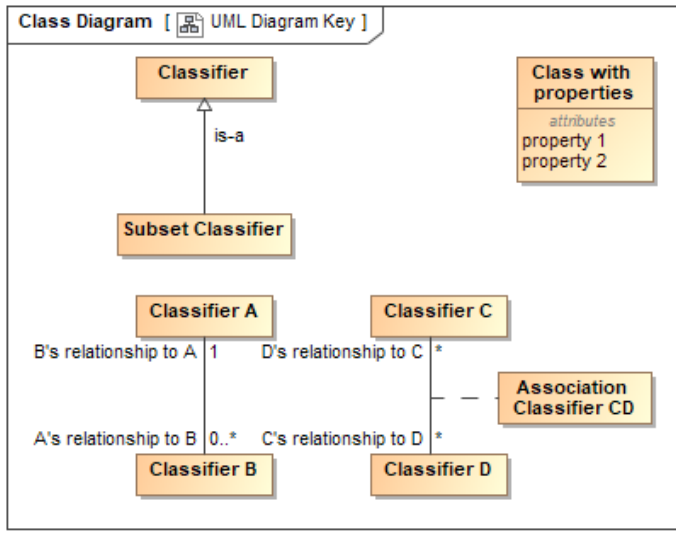


Fig. 9. UML Class Diagram Notation (Simplified)

A class can relate to other classes in non-taxonomical ways and these are denoted by edges with either no arrowheads, where the relationship is bidirectional (as shown in figure 9 for classifiers A and B), or with a single open arrowhead end (→) to denote a unidirectional relationship. The meaning of the relationship is denoted by the verb phrases at each end of the edge for bidirectional relationship, or just at the arrow headed end for unidirectional relationship.

A relation end multiplicity quantifies the permissible number of relationships that may exist when the related terms are being used. Table 1 lists the typical multiplicities that have been used and their corresponding meaning. The combination of the multiplicity and the associated end verb phrase combine to provide a quantified relation from one classifier to the related classified.

An association can be further qualified by an association class (as shown in figure 9 for classifiers C and D with association class CD). The purpose of an association class is to provide a class based definition of the association; that is, an association class is a class with edges. One benefit this provides in ontology modelling is that it allows relationships to be defined terms by virtue of the association class name.

Table 1. UML Class diagram relationship multiplicity designators

Multiplicity designation	**Meaning**
1	One
0..1	May have (i.e. none or one)
1..*	Some (i.e. one or more)
*	May have some (i.e. none, one, or more)

A class may have properties, which are displayed as a UML "attribute" compartment in the class's rectangle, as shown for the class at the top right hand side of figure 9. Classifier properties provide specific detail about the classifier and when the classifier is used, these properties will take on usage specific values.

Ontologies typically make significant use of taxonomical relationships, where there is a need to constrain the relationships that are defined between the more general terms for use between the more specific subset (specialised) terms (i.e. taxa). The problem is set out in Venn diagram notation in figure 10.

In figure 10 two primary sets are shown, Set A and Set B. In each primary set, there is a subset, denoted SA and SB. Some small shapes are used to denote things that are members of these four sets, where the circles are members of the set of A and the triangles are members of the set of B. There is a defined relationship between set A and set B, which is simplistically denoted by the line labelled "A-B relationship". Hence, every member of set A can be related to a member of Set B. Some of these relationship pairings are show as lines between the small set-member shapes. Now the problem is that for the members of the two subsets, SA and SB, the general "A-B relationship" must be constrained to be only between members of the two subsets. Hence, the relationship between subset-members denoted as "cSA" and "bSB" is allowed, but the relationship between subset member "aSa" and non-subset member "wB" is not allowed.

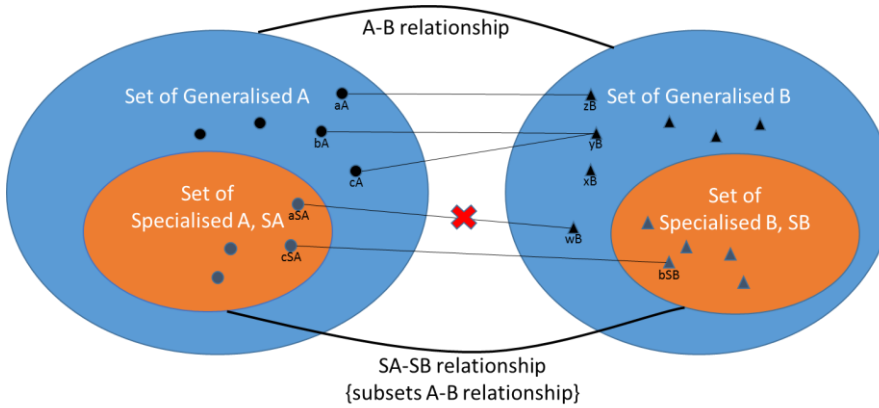


Fig. 10. Venn diagram illustration of relationship subsetting

To solve this problem, an additional relationship between the subsets must be added, as shown with the line denoted "SA-SB relationship". However, this relationship must replace the more general relationship for members of the subsets; otherwise, the example relationship between member's aSA and wB would remain valid. This is achieved by *subsetting* the general relationship, as denoted by the curly bracket annotation on the subset relationship.

The UML class diagram representation of the Venn diagram set classifiers shown in figure 10 is shown in figure 11.

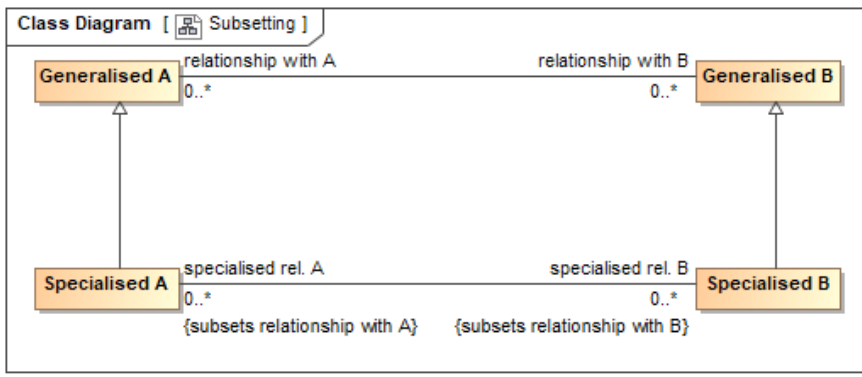


Fig. 11. Subsetted relationships in UML

Subset relationships are strongly defined to be strict subsets of the more general relationship they are constraining. This extends to the meaning of the relationship and its quantification (multiplicity).

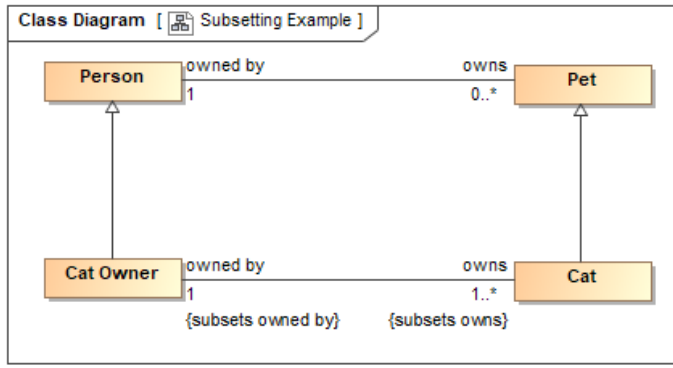


Fig. 12. Example of subsetted relationships

An example of the use of subsetted relationships is shown in figure 12. This illustrates the application of subset relationships, as terms in an ontology are refined. Here a "cat owner" is defined as owning at least one "cat", based on the optional relationship that a "person" may own a "pet". Whilst in this example the subsetted relationship retains the original's intent of "ownership", it has further constrained the quantification from an optional *may have some* ("0..*") to a required *has some* ("1..*") on the "cat owner" "owns" relation side, as well as further constraining the "cat" "owned by" relationship to exactly one ("1") "cat owner".

There are occasions where more flexibility in classifier subset relationships is required and strict subsetting is relaxed. This is called a *redefinition* and is illustrated in figure 13. In this example the "person" – "organisation" relationship of "has role in" is redefined for the specialisation of "engineer" – "engineering department" relationship of "works for". Conversely, the relationship of "employs" becomes "has" because in the context of "engineering department" the "has Engineers" concept implies more than the base relationship of employment.

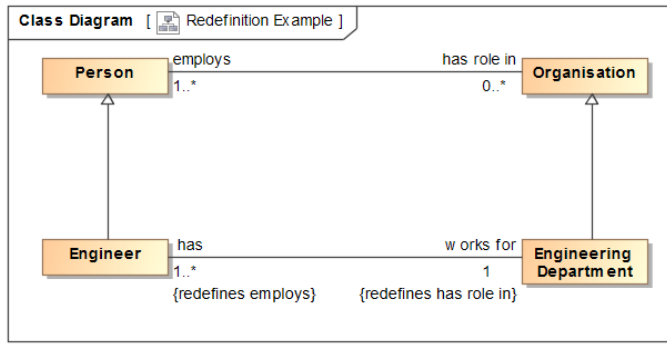


Fig. 13. Example of redefined relationships

AUTHOR INDEX

Mike Parsons • Mark Nicholson **Editors**

Assuring Safe Autonomy

Proceedings of the 28th
Safety-Critical Systems Symposium
York, UK
11th-13th February 2020

Assuring Safe Autonomy contains papers presented at the 28th annual Safety-Critical Systems Symposium, held in York, UK, in February 2020.

The Symposium is for engineers, managers and academics in the field of system safety, across all industry sectors, so the papers making up this volume offer wide coverage of current safety topics and a blend of academic research and industrial experience. They include both recent developments in the field and discussion of open issues and questions.

The topics covered in this volume include: Assurance Cases, Autonomy, AI and Machine Learning, Data Safety, Human Factors, New Techniques and Security Informed Safety.

This book will be of interest to practitioners, managers and academics working in the safety-critical and safety-related areas.

SCSC

FOR EVERYONE WORKING IN SYSTEM SAFETY