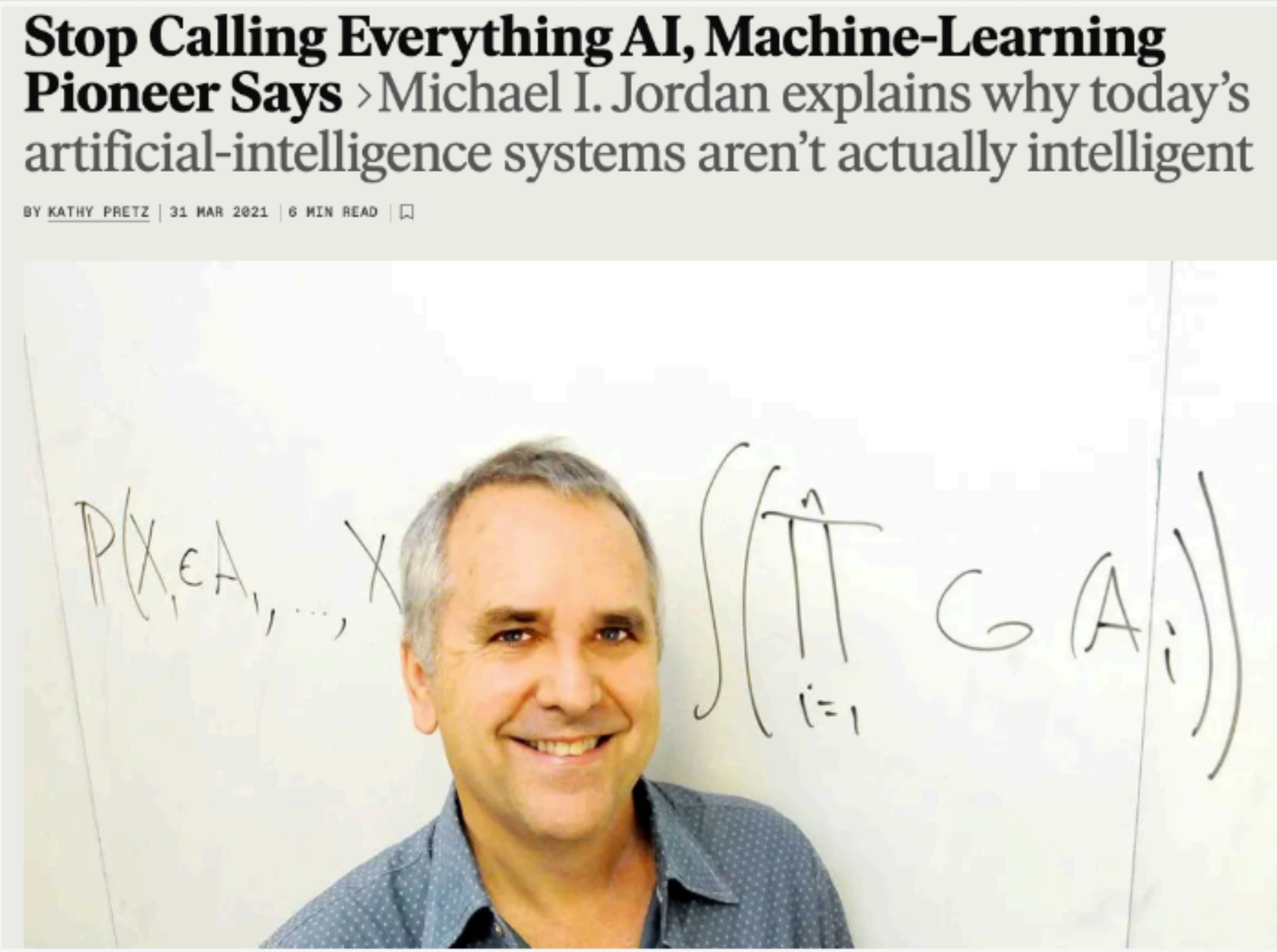
Escola Politécnica da USP- Depto. de Enga. Mecatrônica

PMR-3510 Inteligência Artificial Aula 8 - Sist. Especialistas em Automação

Prof. José Reinaldo Silva
reinaldo@usp.br

d-l@b

→ C4AI

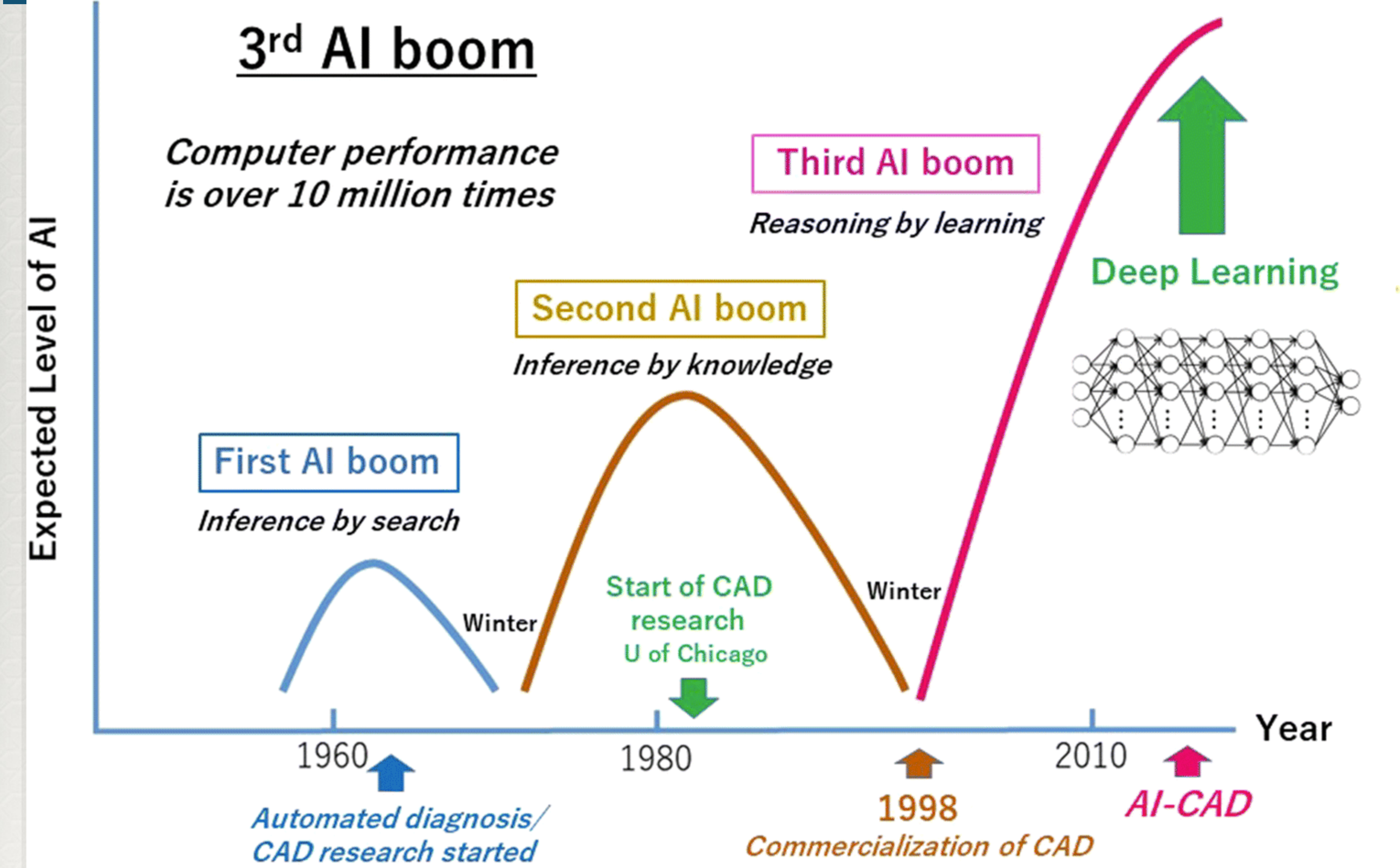


Stop Calling Everything AI, Machine-Learning Pioneer Says › Michael I. Jordan explains why today's artificial-intelligence systems aren't actually intelligent

BY KATHY PRETZ | 31 MAR 2021 | 6 MIN READ |

$P(X_1 \in A_1, \ldots, X$
$\int \left(\prod_{i=1} G(A_i) \right)$

3^{rd} AI boom



Computer performance is over 10 million times

First AI boom

Inference by search

Second AI boom

Inference by knowledge

Third AI boom

Reasoning by learning

Deep Learning

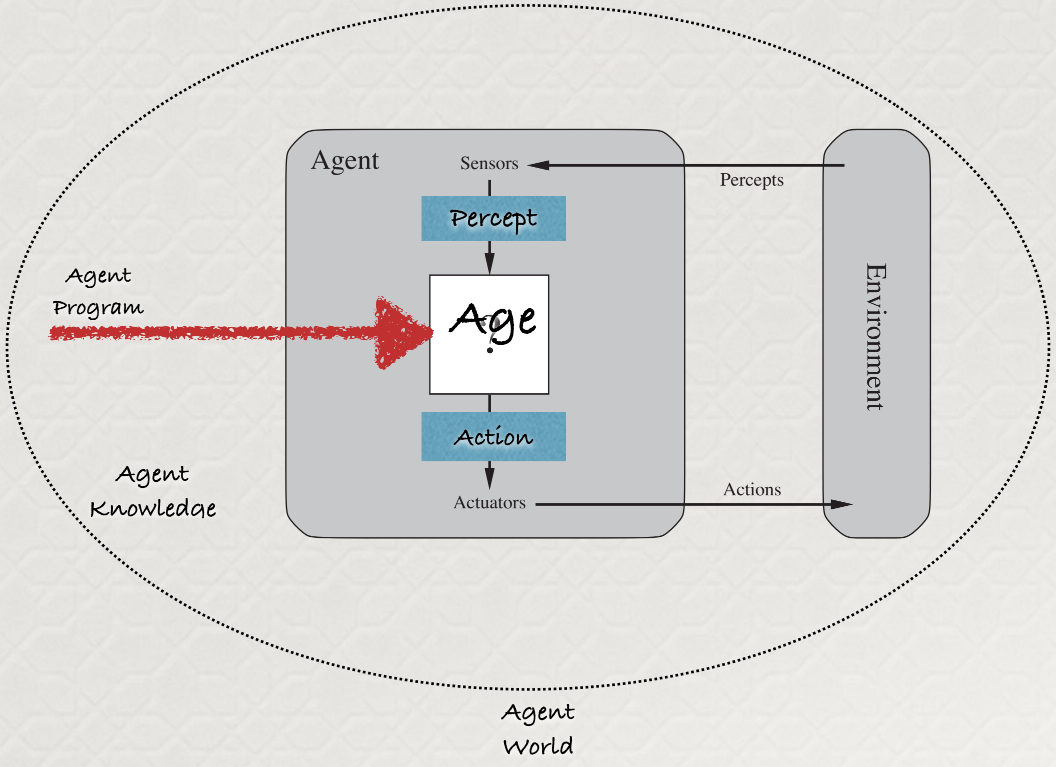
Start of CAD research U of Chicago

Automated diagnosis/ CAD research started

1998 *Commercialization of CAD*

AI-CAD

A perspectiva é que seja possível aplicar agentes inteligentes e sistemas especialistas em processos de automação.



Diferentes tipos de agentes:

Reflexivos (simple réflex agents)

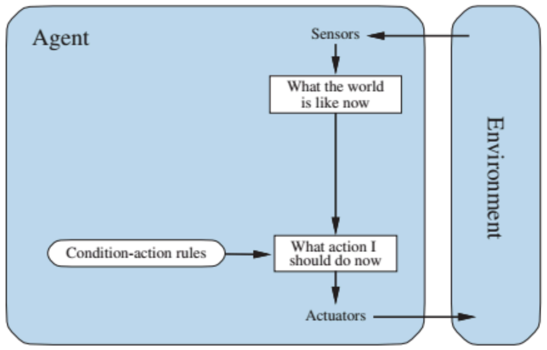


Figure 2.9 Schematic diagram of a simple reflex agent. We use rectangles to denote the current internal state of the agent's decision process, and ovals to represent the background information used in the process.

Agentes baseados em modelos (Model-based agents)

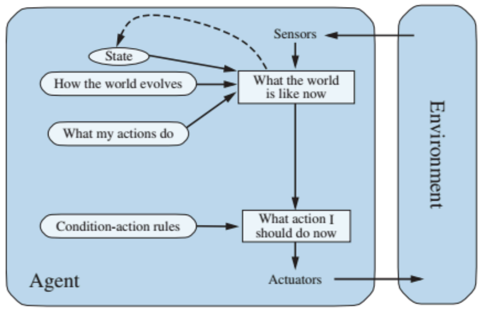


Figure 2.11 A model-based reflex agent.

Agentes baseados em objetivos (Goal-based agents)

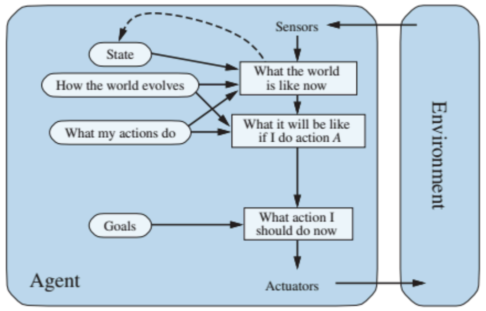


Figure 2.13 A model-based, goal-based agent. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

Agentes baseados em serviço (Utility-based agents)

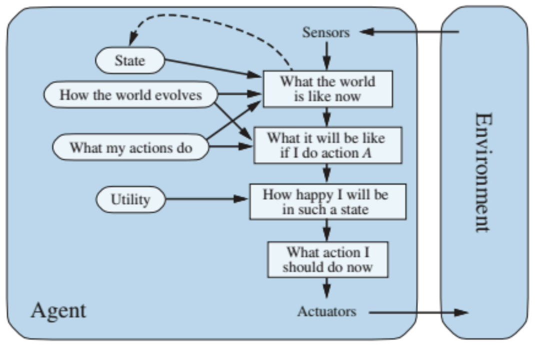


Figure 2.14 A model-based, utility-based agent. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility is computed by averaging over all possible outcome states, weighted by the probability of the outcome.

Agentes que aprendem (Learning agents)

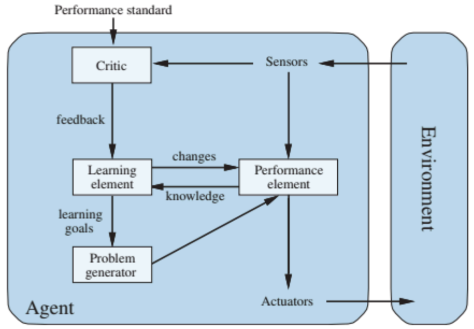
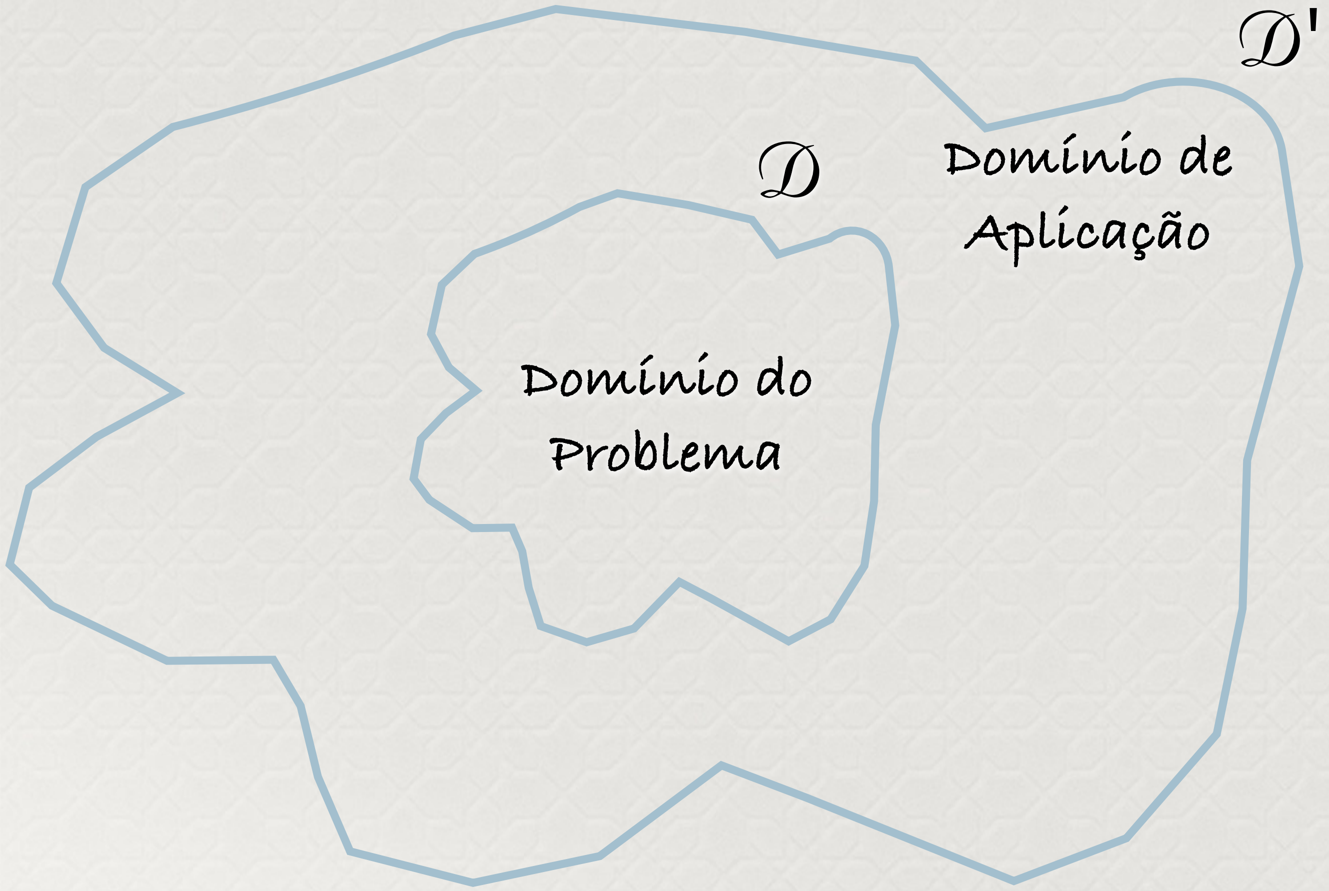
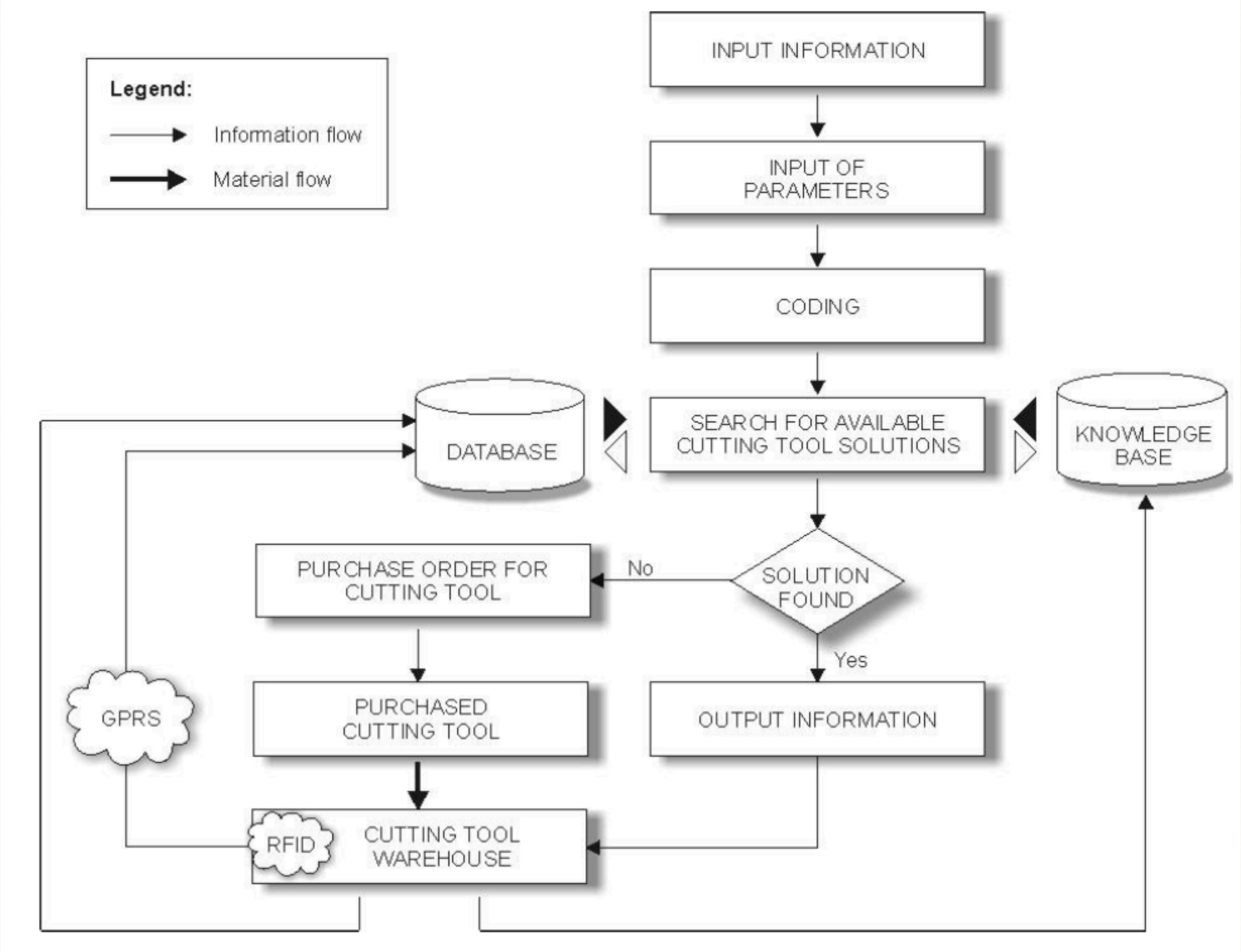


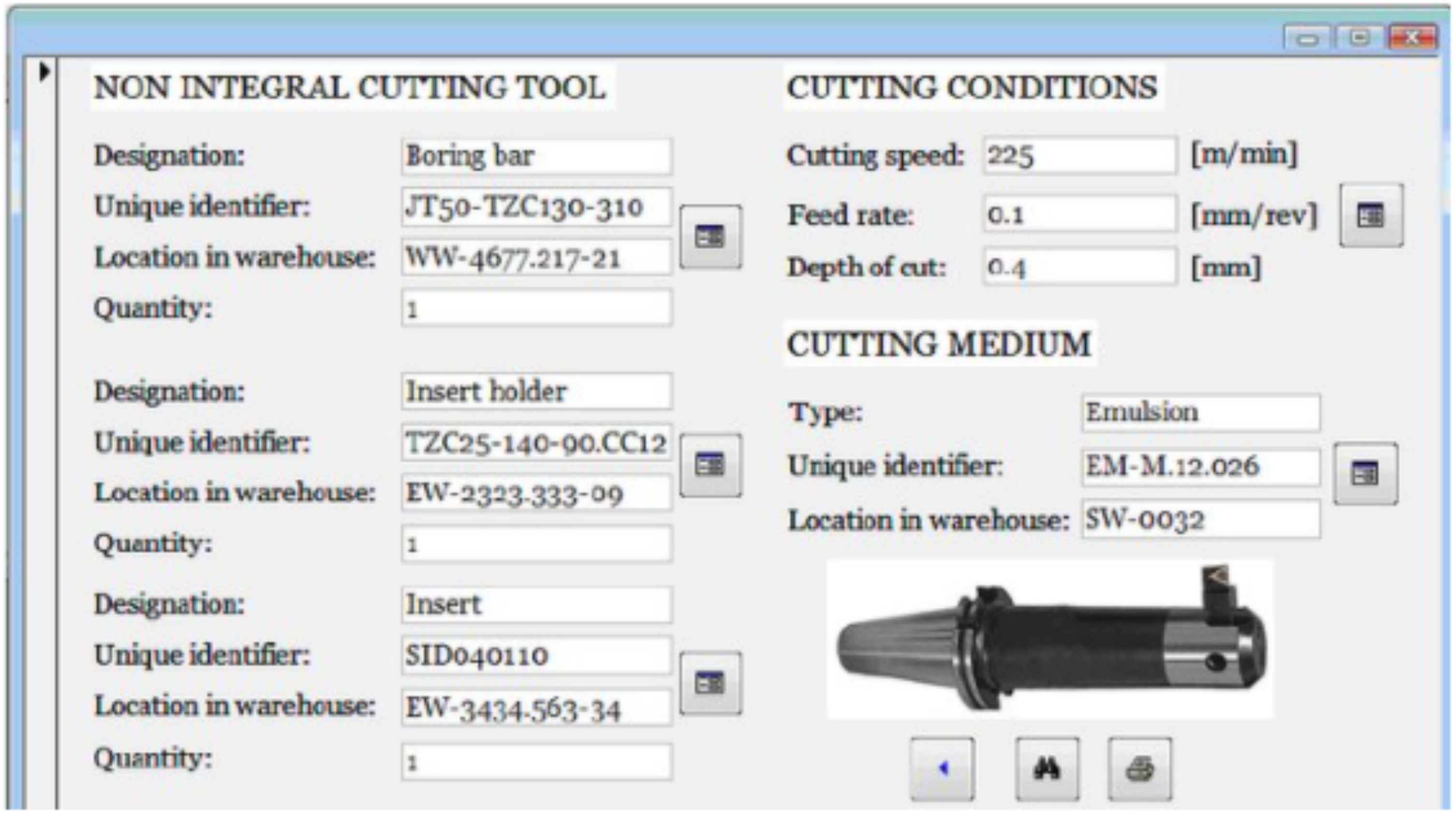
Figure 2.15 A general learning agent. The "performance element" box represents what we have previously considered to be the whole agent program. Now, the "learning element" box gets to modify that program to improve its performance.

Podemos mapear o domínio do problema, a aplicação (em automação) de sistemas baseados em conhecimento que usam regras de produção.



An integral system for automated cutting tool selection





NON INTEGRAL CUTTING TOOL

Designation:	Boring bar
Unique identifier:	JT50-TZC130-310
Location in warehouse:	WW-4677.217-21
Quantity:	1
Designation:	Insert holder
Unique identifier:	TZC25-140-90.CC12
Location in warehouse:	EW-2323.333-09
Quantity:	1
Designation:	Insert
Unique identifier:	SID040110
Location in warehouse:	EW-3434.563-34
Quantity:	1

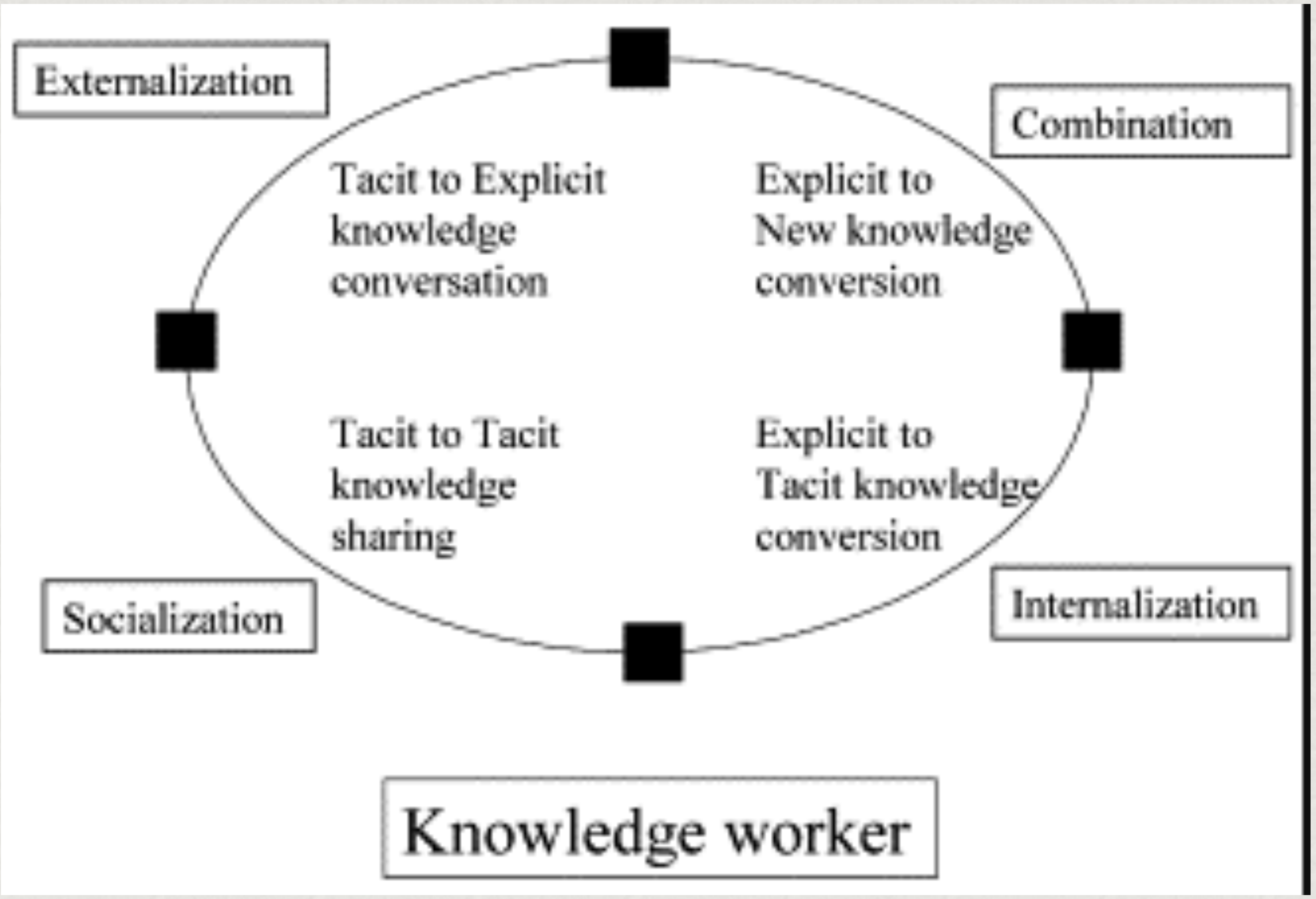
CUTTING CONDITIONS

Cutting speed:	225	[m/min]
Feed rate:	0.1	[mm/rev]
Depth of cut:	0.4	[mm]

CUTTING MEDIUM

Type:	Emulsion
Unique identifier:	EM-M.12.026
Location in warehouse:	SW-0032

Knowledge Warehouse

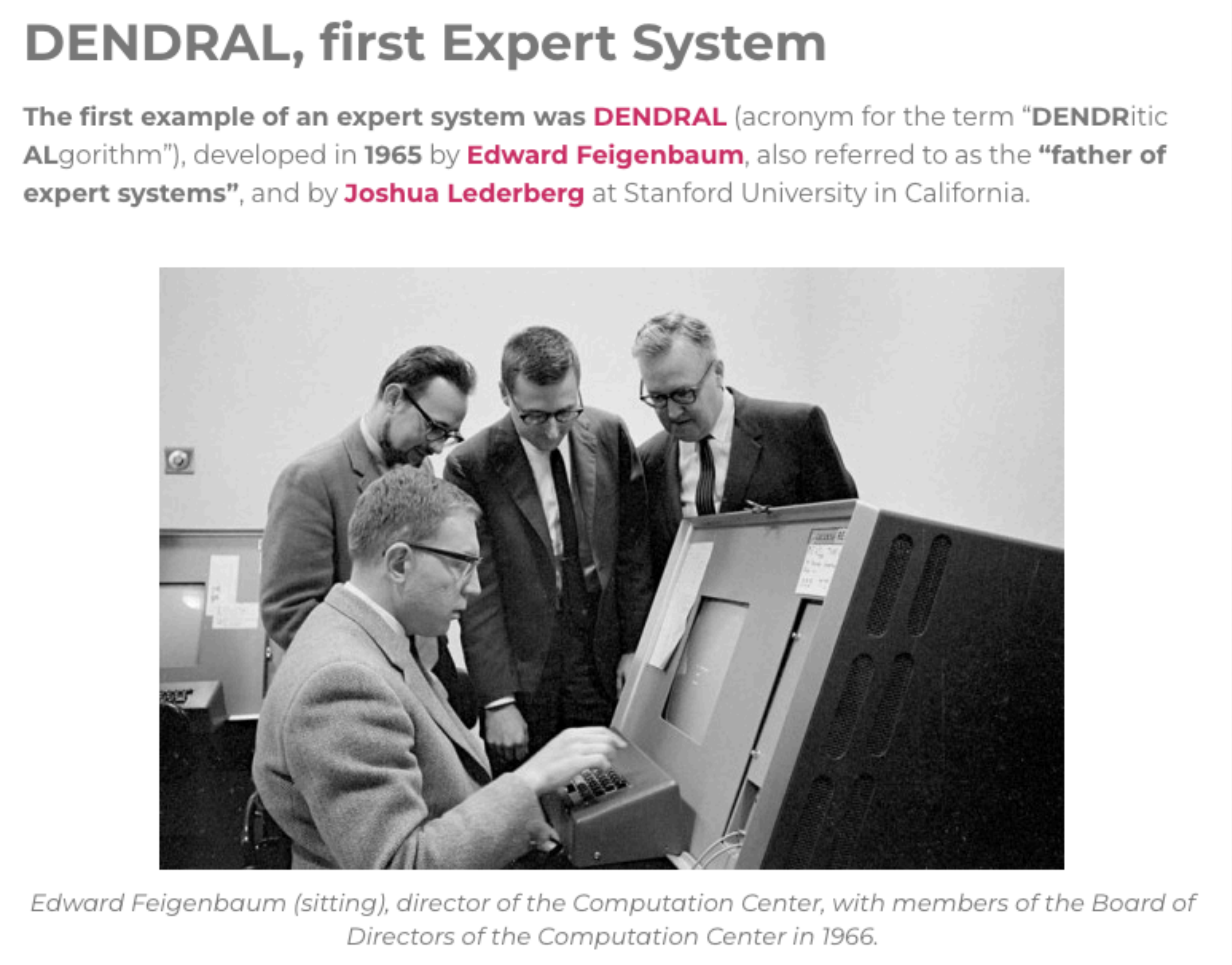

Externalization
Combination
Tacit to Explicit knowledge conversation
Explicit to New knowledge conversion
Tacit to Tacit knowledge sharing
Explicit to Tacit knowledge conversion
Socialization
Internalization
Knowledge worker

H. Nemati et al., Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence and data warehousing, Decision Support Systems, vol. 33, no. 2, 2002.

SCANNING
IN
PROGRESS

Sistemas (especialista), baseados em regras de produção podem ser usados para direcionar robôs armazenadores em warehouses.

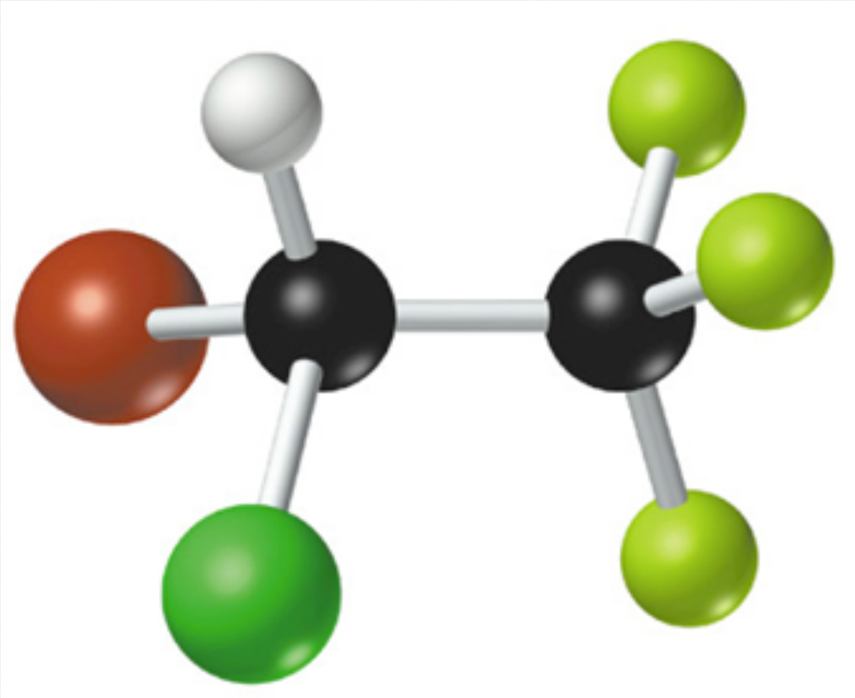
O histórico dos sistemas de diagnóstico: a primeira aplicação dos sistemas especialistas.



DENDRAL, first Expert System

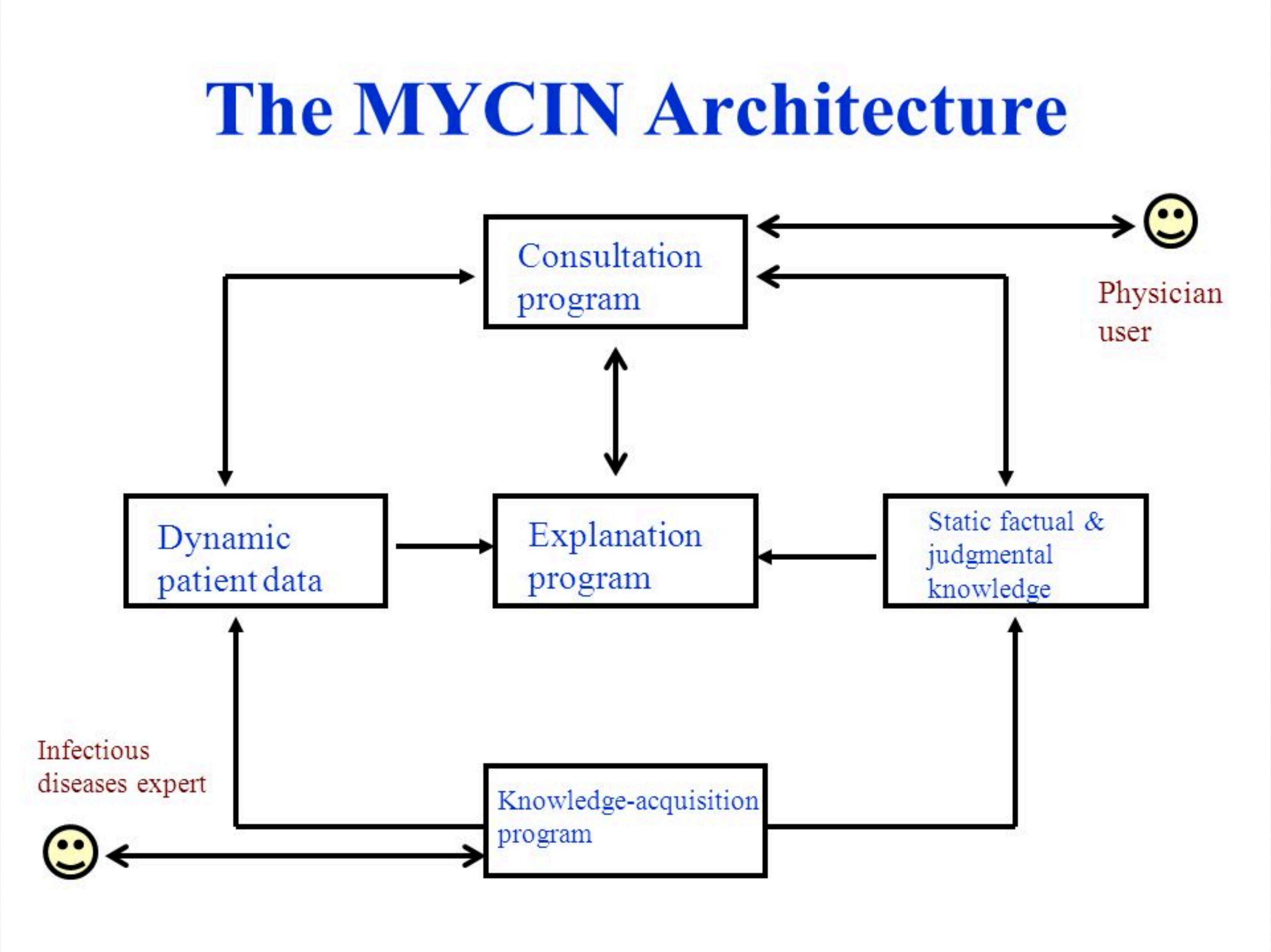
The first example of an expert system was DENDRAL (acronym for the term "**DENDR**itic **AL**gorithm"), developed in **1965** by **Edward Feigenbaum**, also referred to as the **"father of expert systems"**, and by **Joshua Lederberg** at Stanford University in California.

Edward Feigenbaum (sitting), director of the Computation Center, with members of the Board of Directors of the Computation Center in 1966.



O DENDRAL, primeiro sistema especialista, tinha como função identificar a estrutura química e compostos, a partir de análise espectral.

Sistemas (especialista) para diagnóstico

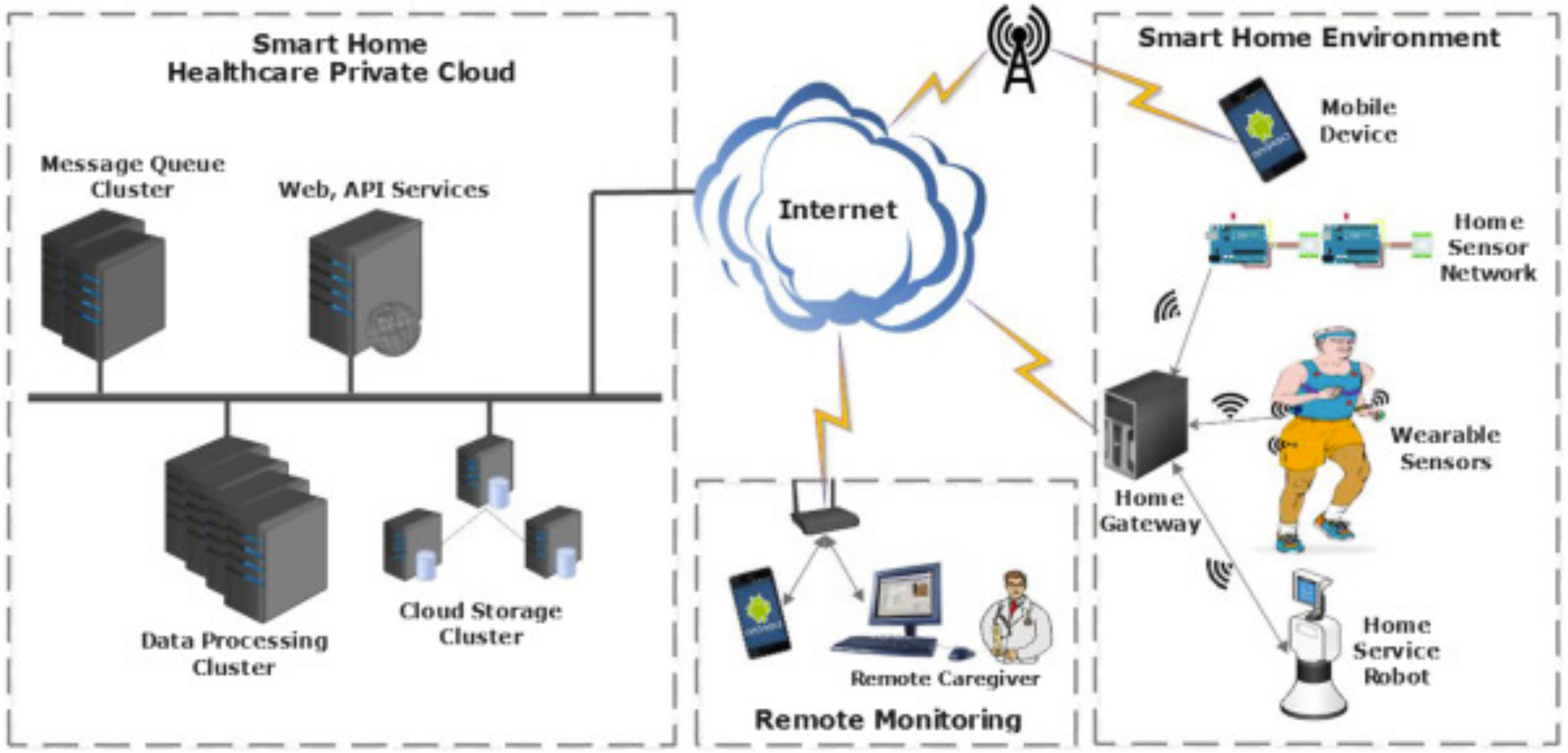


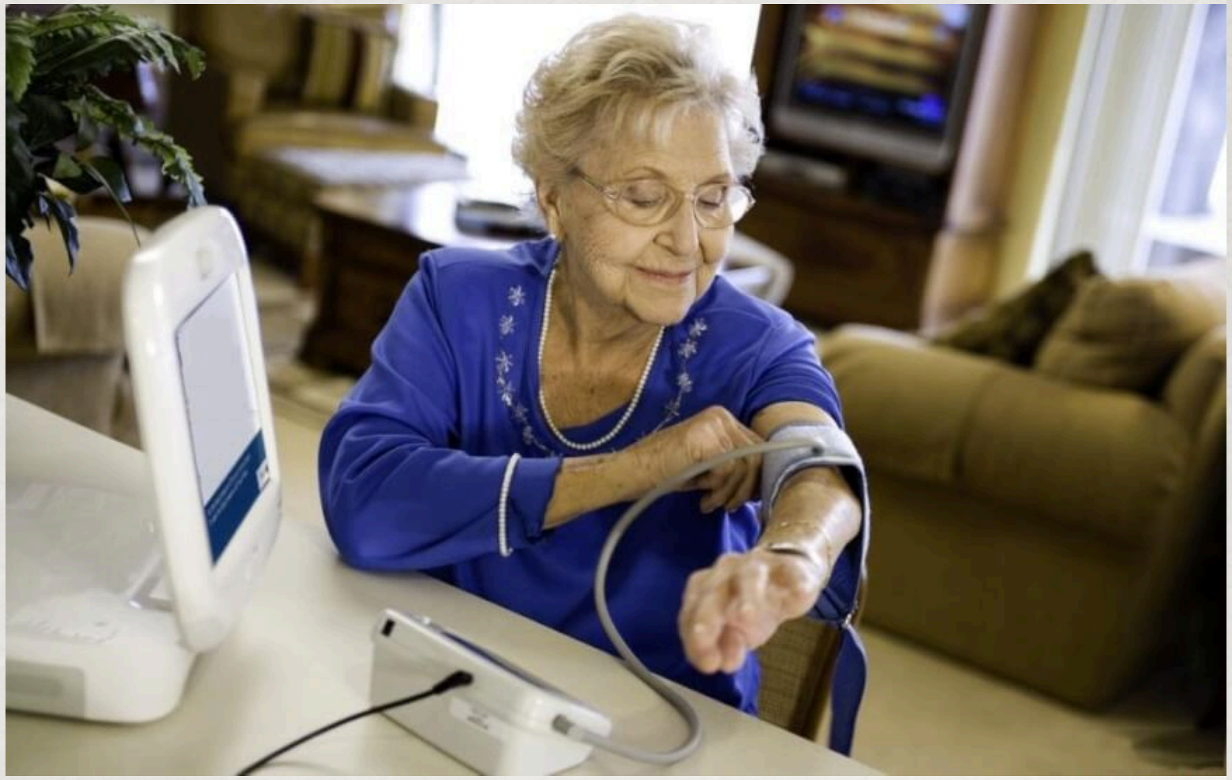
Symbolics workstation

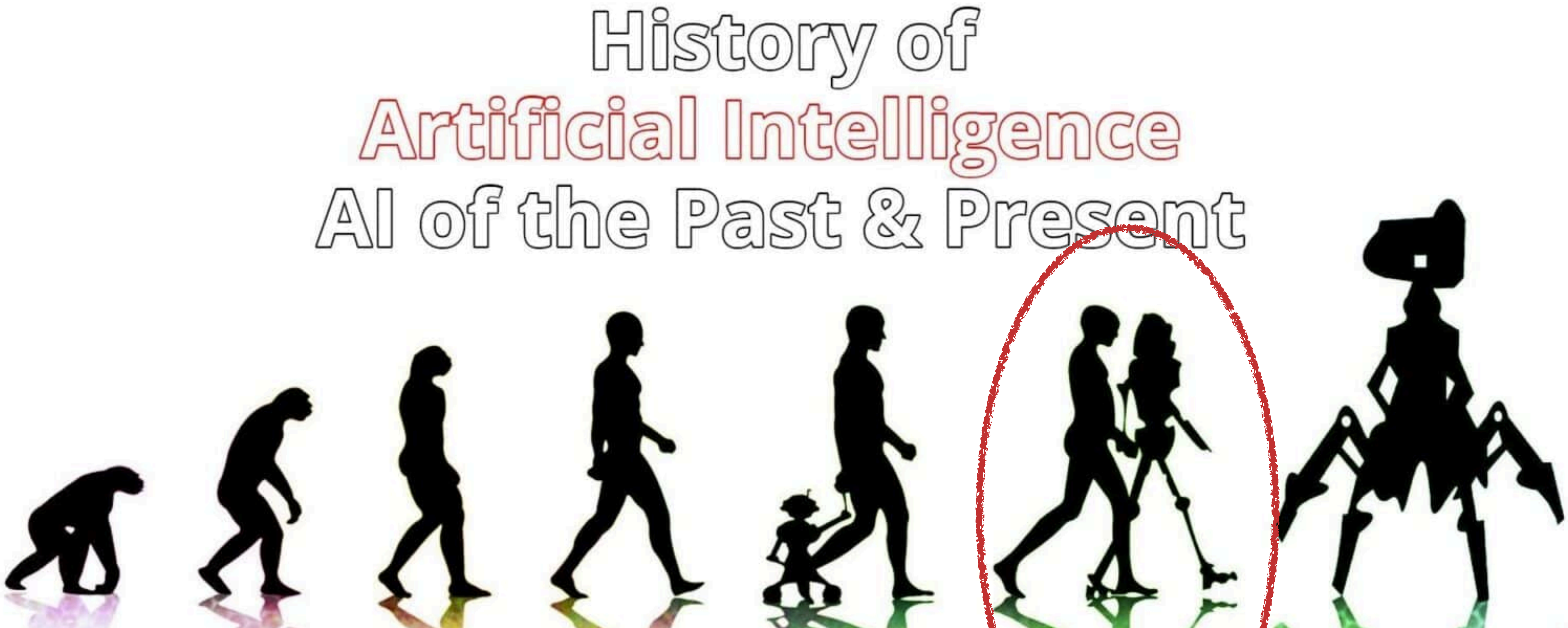
HP-RISC 9000

ZEUS
Murketts Vauxhall
ZEUS INTELLIGENT DIAGNOSTICS

AI Applications to Health Care


Smart Home
Healthcare Private Cloud
Message Queue
Cluster
Web, API Services
Data Processing
Cluster
Cloud Storage
Cluster
Internet
Remote Caregiver
Remote Monitoring
Smart Home Environment
Mobile
Device
Home
Sensor
Network
Wearable
Sensors
Home
Gateway
Home
Service
Robot





Dispositivos e mecanismos que reagem "como humanos"

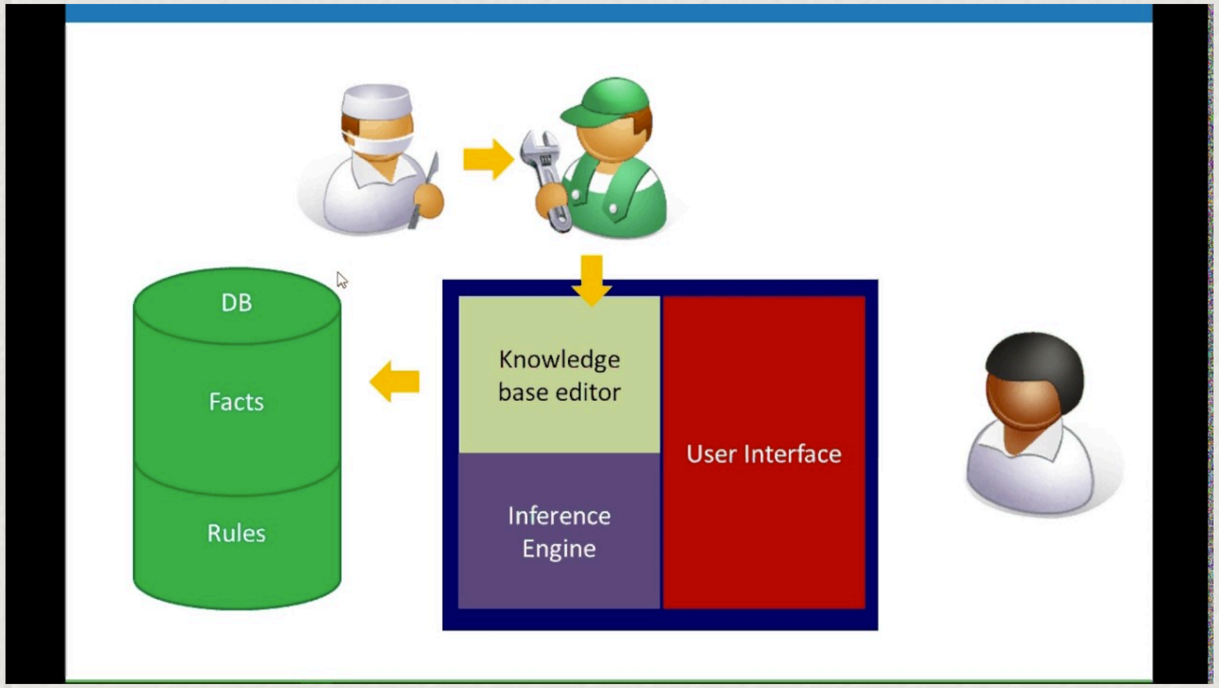
Contraponto da imitação

Resolução de problemas

Captura do conhecimento especialista

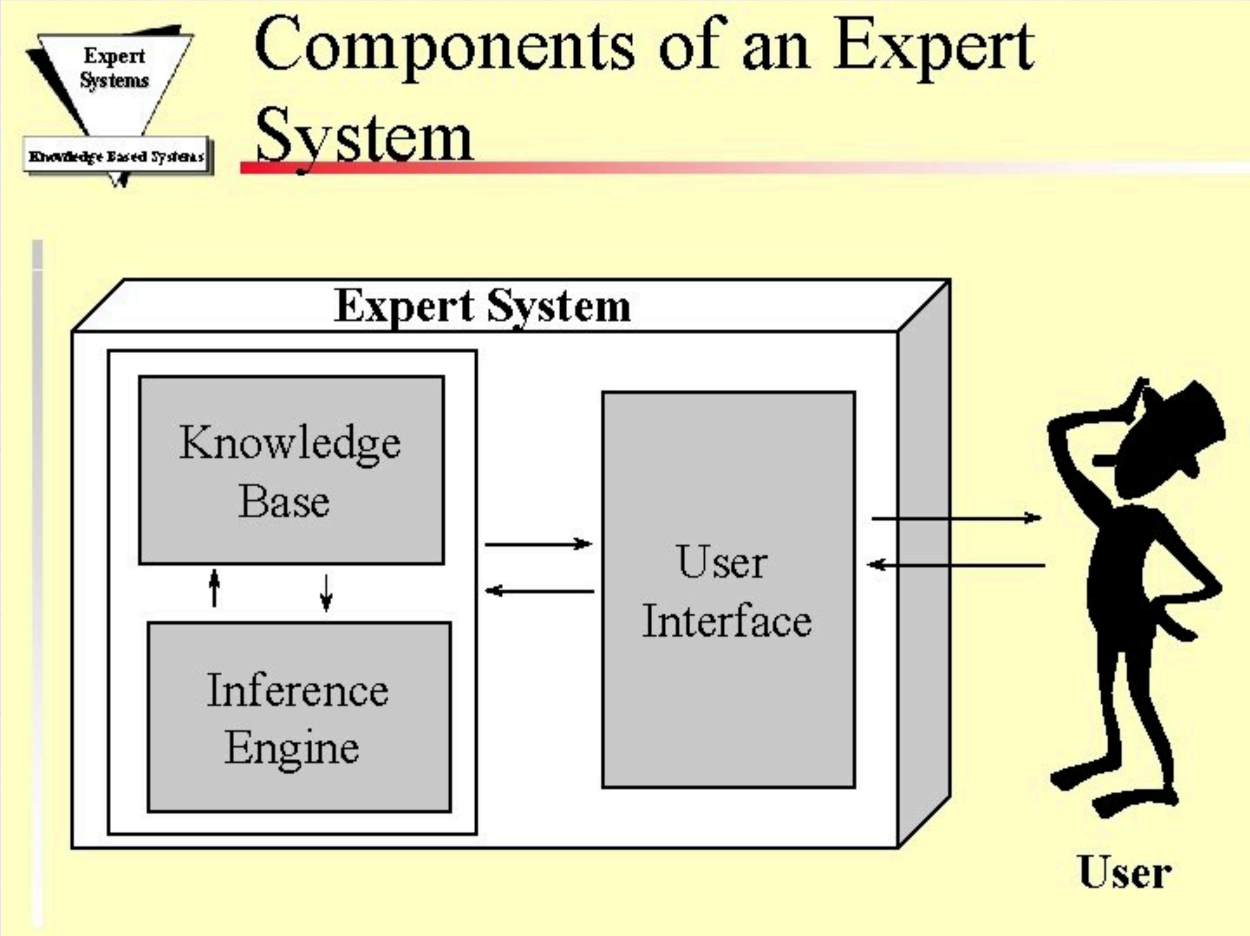
Deep Knowledge and Deep Learning

Tanto as aplicações em logística quanto em sistemas de diagnóstico se enquadram na característica de serem baseados na captura de conhecimento tácito e explícito em uma base de conhecimento na forma de regras de produção. Esta base de conhecimento (e de dados) é um elemento fundamental dos sistemas especialistas.

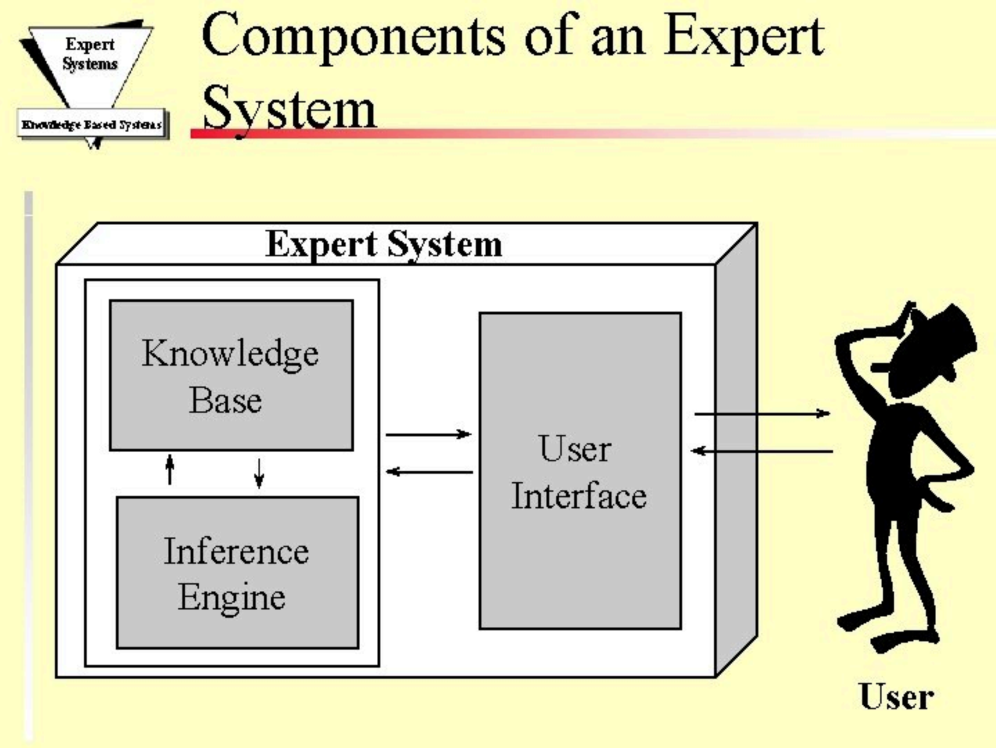


Dado o número e a variedade de aplicações vale a pena entender mais profundamente como funciona um sistema baseado em conhecimento (KB system), o que significa entender o funcionamento da base de conhecimento e da máquina de inferência.

Um KB-system é composto de fatos e regras associados a algum “environment" (domínio). Uma base envolvendo conhecimento tácito e explícito pode ser associada a um “sistema especialista”.

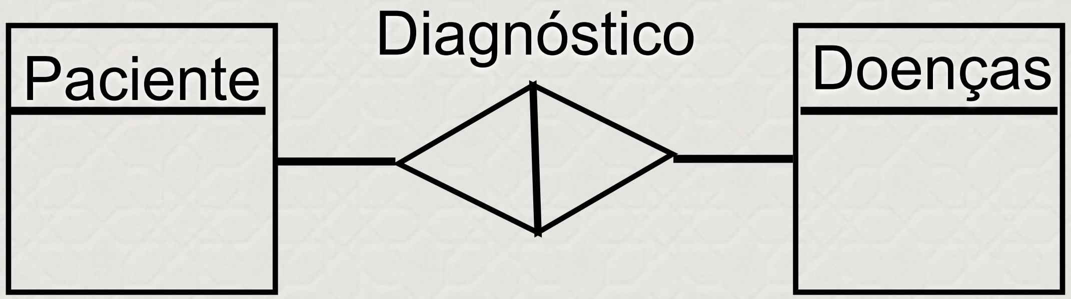


Components of an Expert System



Components of an Expert System

Fatos em um BD



Fatos em Prolog

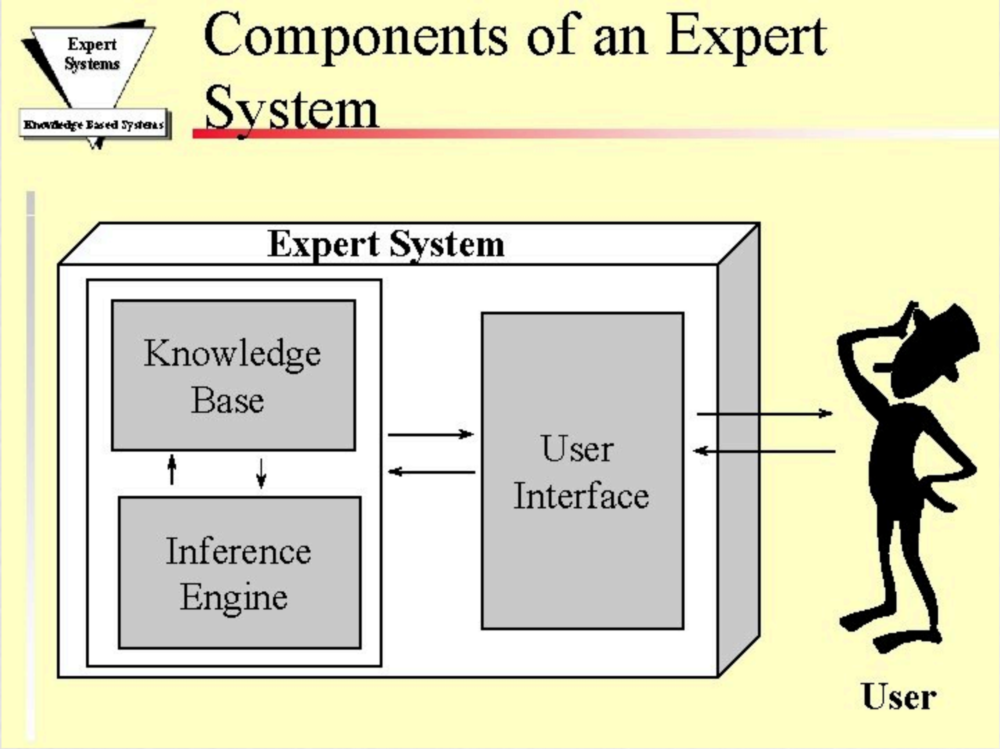
```
diagnosticado(pacienteX, doençaY)
```

Regras:

If-then (-else)

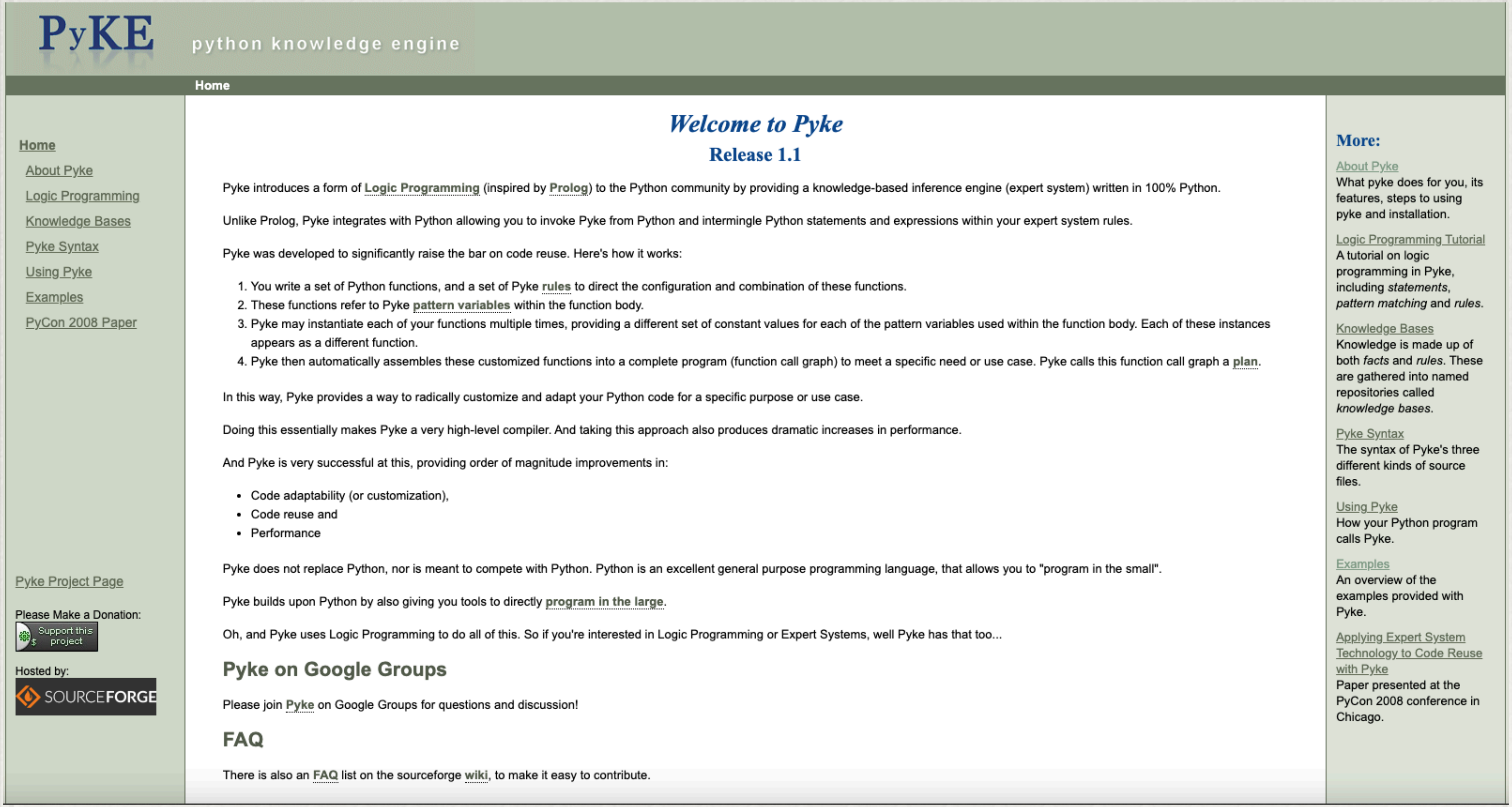
Regras:

Cláusulas de Horn



Components of an Expert System

https://pyke.sourceforge.net



PyKE python knowledge engine

Home

- Home
- About Pyke
- Logic Programming
- Knowledge Bases
- Pyke Syntax
- Using Pyke
- Examples
- PyCon 2008 Paper

Pyke Project Page

Please Make a Donation:

Support this project

Hosted by:

SOURCEFORGE

Welcome to Pyke

Release 1.1

Pyke introduces a form of Logic Programming (inspired by Prolog) to the Python community by providing a knowledge-based inference engine (expert system) written in 100% Python.

Unlike Prolog, Pyke integrates with Python allowing you to invoke Pyke from Python and intermingle Python statements and expressions within your expert system rules.

Pyke was developed to significantly raise the bar on code reuse. Here's how it works:

1. You write a set of Python functions, and a set of Pyke rules to direct the configuration and combination of these functions.
2. These functions refer to Pyke pattern variables within the function body.
3. Pyke may instantiate each of your functions multiple times, providing a different set of constant values for each of the pattern variables used within the function body. Each of these instances appears as a different function.
4. Pyke then automatically assembles these customized functions into a complete program (function call graph) to meet a specific need or use case. Pyke calls this function call graph a plan.

In this way, Pyke provides a way to radically customize and adapt your Python code for a specific purpose or use case.

Doing this essentially makes Pyke a very high-level compiler. And taking this approach also produces dramatic increases in performance.

And Pyke is very successful at this, providing order of magnitude improvements in:

- Code adaptability (or customization),
- Code reuse and
- Performance

Pyke does not replace Python, nor is meant to compete with Python. Python is an excellent general purpose programming language, that allows you to "program in the small".

Pyke builds upon Python by also giving you tools to directly program in the large.

Oh, and Pyke uses Logic Programming to do all of this. So if you're interested in Logic Programming or Expert Systems, well Pyke has that too...

Pyke on Google Groups

Please join Pyke on Google Groups for questions and discussion!

FAQ

There is also an FAQ list on the sourceforge wiki, to make it easy to contribute.

More:

About Pyke
What pyke does for you, its features, steps to using pyke and installation.

Logic Programming Tutorial
A tutorial on logic programming in Pyke, including *statements*, *pattern matching* and *rules*.

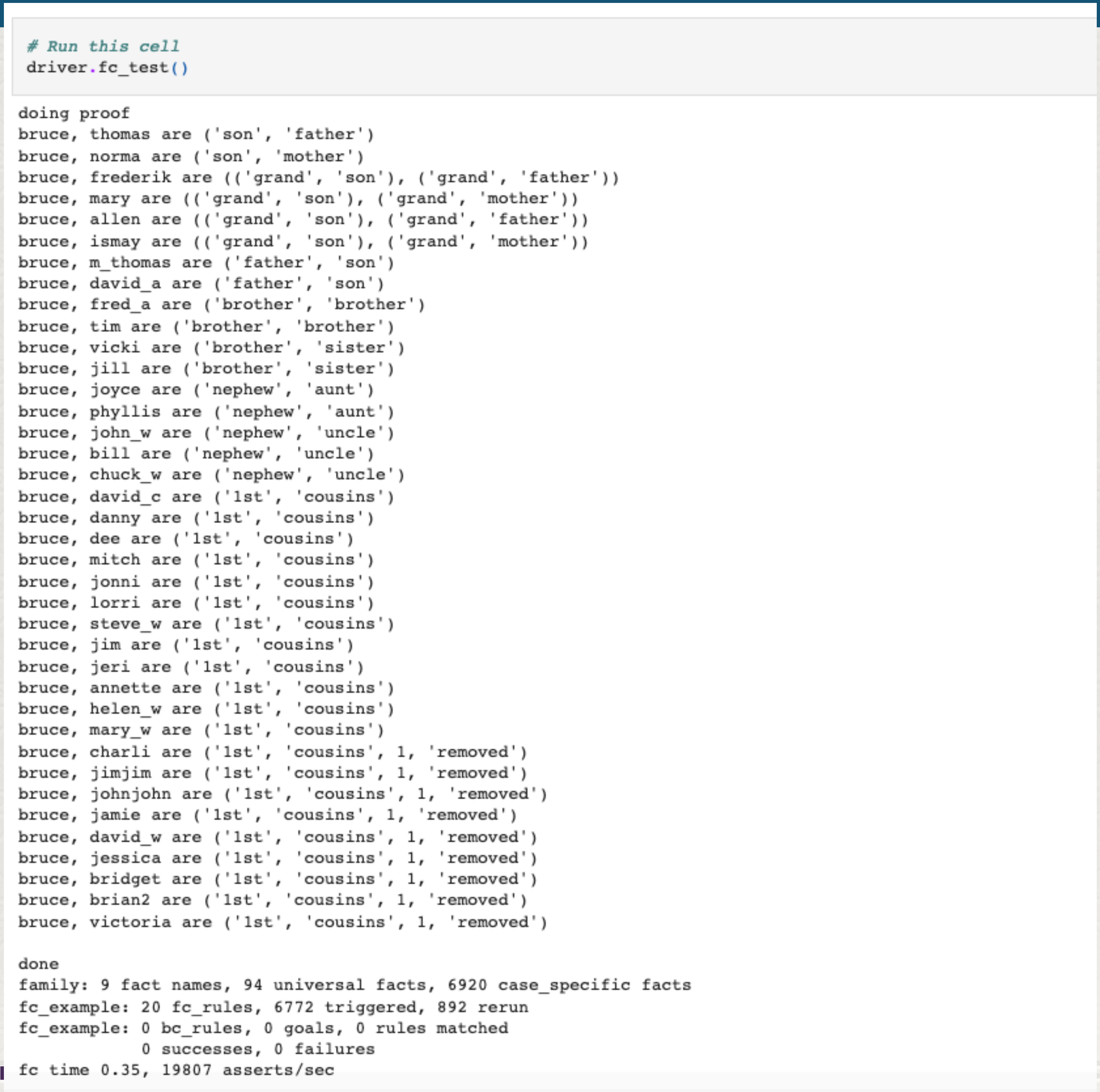
Knowledge Bases
Knowledge is made up of both *facts* and *rules*. These are gathered into named repositories called *knowledge bases*.

Pyke Syntax
The syntax of Pyke's three different kinds of source files.

Using Pyke
How your Python program calls Pyke.

Examples
An overview of the examples provided with Pyke.

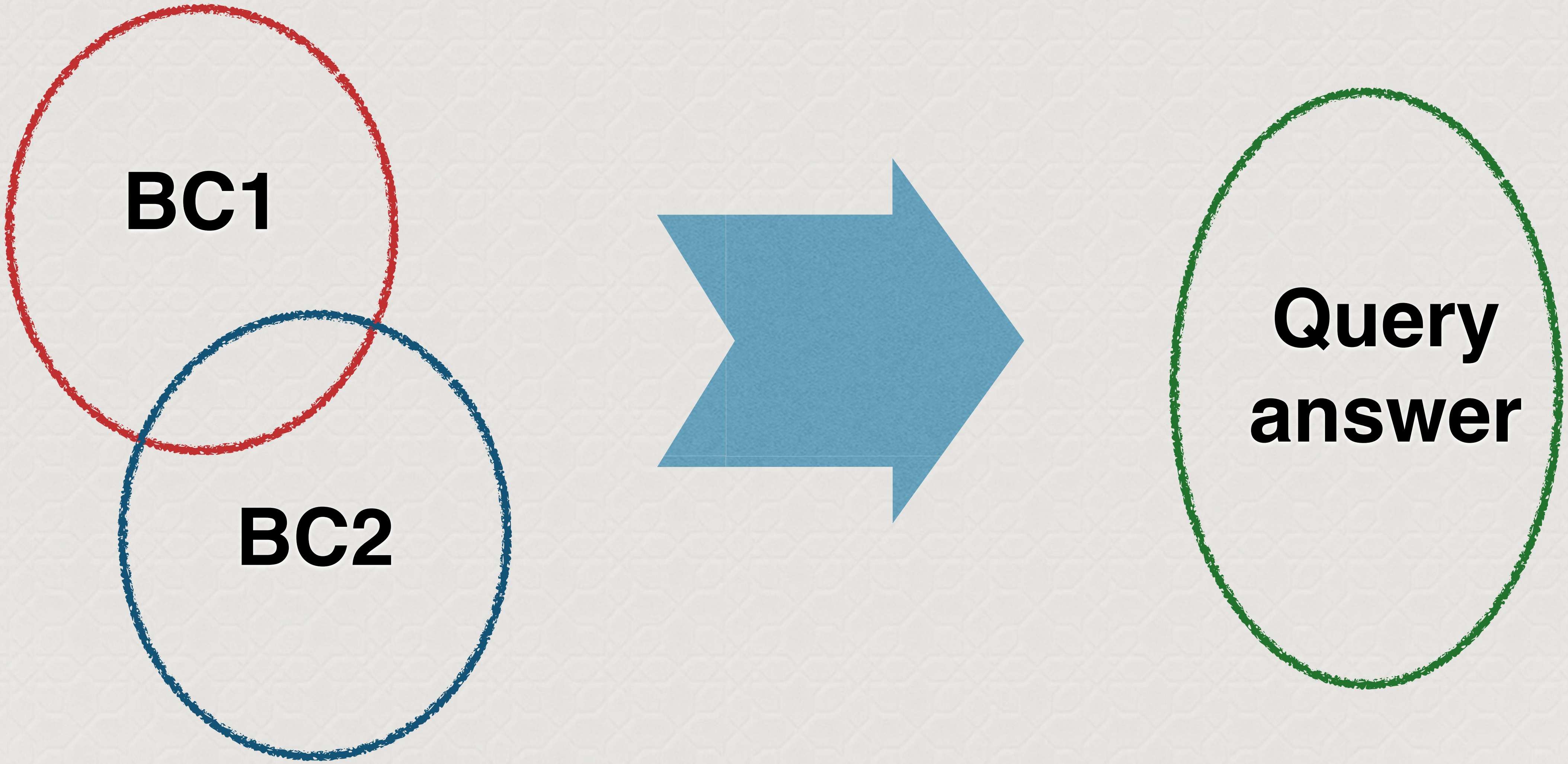
Applying Expert System Technology to Code Reuse with Pyke
Paper presented at the PyCon 2008 conference in Chicago.



```
# Run this cell
driver.fc_test()
```

```
doing proof
bruce, thomas are ('son', 'father')
bruce, norma are ('son', 'mother')
bruce, frederik are (('grand', 'son'), ('grand', 'father'))
bruce, mary are (('grand', 'son'), ('grand', 'mother'))
bruce, allen are (('grand', 'son'), ('grand', 'father'))
bruce, ismay are (('grand', 'son'), ('grand', 'mother'))
bruce, m_thomas are ('father', 'son')
bruce, david_a are ('father', 'son')
bruce, fred_a are ('brother', 'brother')
bruce, tim are ('brother', 'brother')
bruce, vicki are ('brother', 'sister')
bruce, jill are ('brother', 'sister')
bruce, joyce are ('nephew', 'aunt')
bruce, phyllis are ('nephew', 'aunt')
bruce, john_w are ('nephew', 'uncle')
bruce, bill are ('nephew', 'uncle')
bruce, chuck_w are ('nephew', 'uncle')
bruce, david_c are ('1st', 'cousins')
bruce, danny are ('1st', 'cousins')
bruce, dee are ('1st', 'cousins')
bruce, mitch are ('1st', 'cousins')
bruce, jonni are ('1st', 'cousins')
bruce, lorri are ('1st', 'cousins')
bruce, steve_w are ('1st', 'cousins')
bruce, jim are ('1st', 'cousins')
bruce, jeri are ('1st', 'cousins')
bruce, annette are ('1st', 'cousins')
bruce, helen_w are ('1st', 'cousins')
bruce, mary_w are ('1st', 'cousins')
bruce, charli are ('1st', 'cousins', 1, 'removed')
bruce, jimjim are ('1st', 'cousins', 1, 'removed')
bruce, johnjohn are ('1st', 'cousins', 1, 'removed')
bruce, jamie are ('1st', 'cousins', 1, 'removed')
bruce, david_w are ('1st', 'cousins', 1, 'removed')
bruce, jessica are ('1st', 'cousins', 1, 'removed')
bruce, bridget are ('1st', 'cousins', 1, 'removed')
bruce, brian2 are ('1st', 'cousins', 1, 'removed')
bruce, victoria are ('1st', 'cousins', 1, 'removed')

done
family: 9 fact names, 94 universal facts, 6920 case_specific facts
fc_example: 20 fc_rules, 6772 triggered, 892 rerun
fc_example: 0 bc_rules, 0 goals, 0 rules matched
            0 successes, 0 failures
fc time 0.35, 19807 asserts/sec
```


BC1
BC2
Query answer

Diferentes combinações de fatos e das regras (e de representação de conhecimento) levam a implementações distintas para uma mesma máquina de inferência.

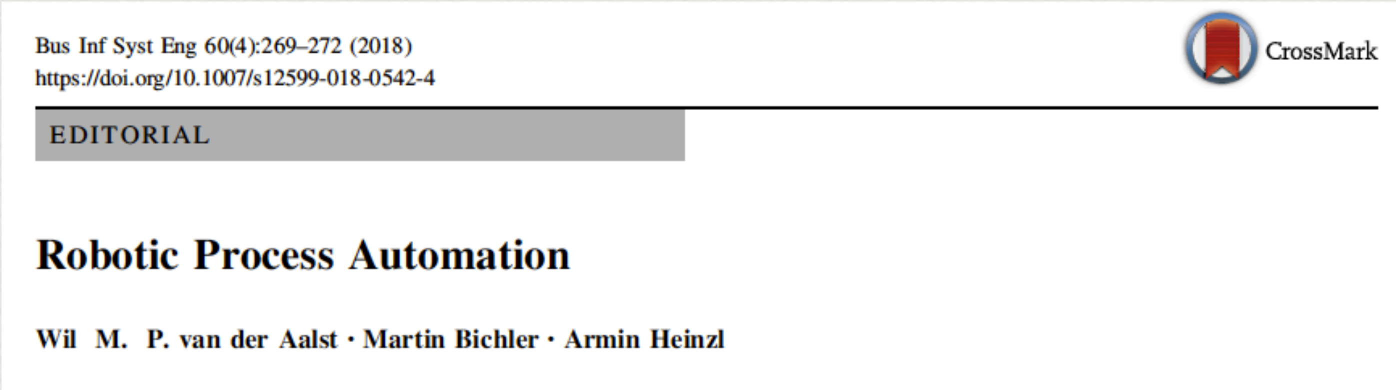
Sistemas (especialista) & RPA (Robotic Process Automation)

Uma outra possibilidade, surgida mais recentemente, é associar sistemas especialistas com RPA's e ter a possibilidade de ter sistemas de agentes inteligentes distribuídos colaborando para realizar diferentes funções.

Definição de RPA

Um sistema RPA é composto por vários agentes (inteligentes) que desempenham tarefas de conexão em vários sistemas colaborativos mantendo regras de consistência ou objetivos pré-definidos.

No mundo dos negócios, entretenimento, sistemas de saúde, etc. o RPA é instanciado com um conjunto de agentes de software (bots) que desempenham funções (repetitivas) normalmente executadas por humanos.



Bus Inf Syst Eng 60(4):269–272 (2018)
https://doi.org/10.1007/s12599-018-0542-4

CrossMark

EDITORIAL

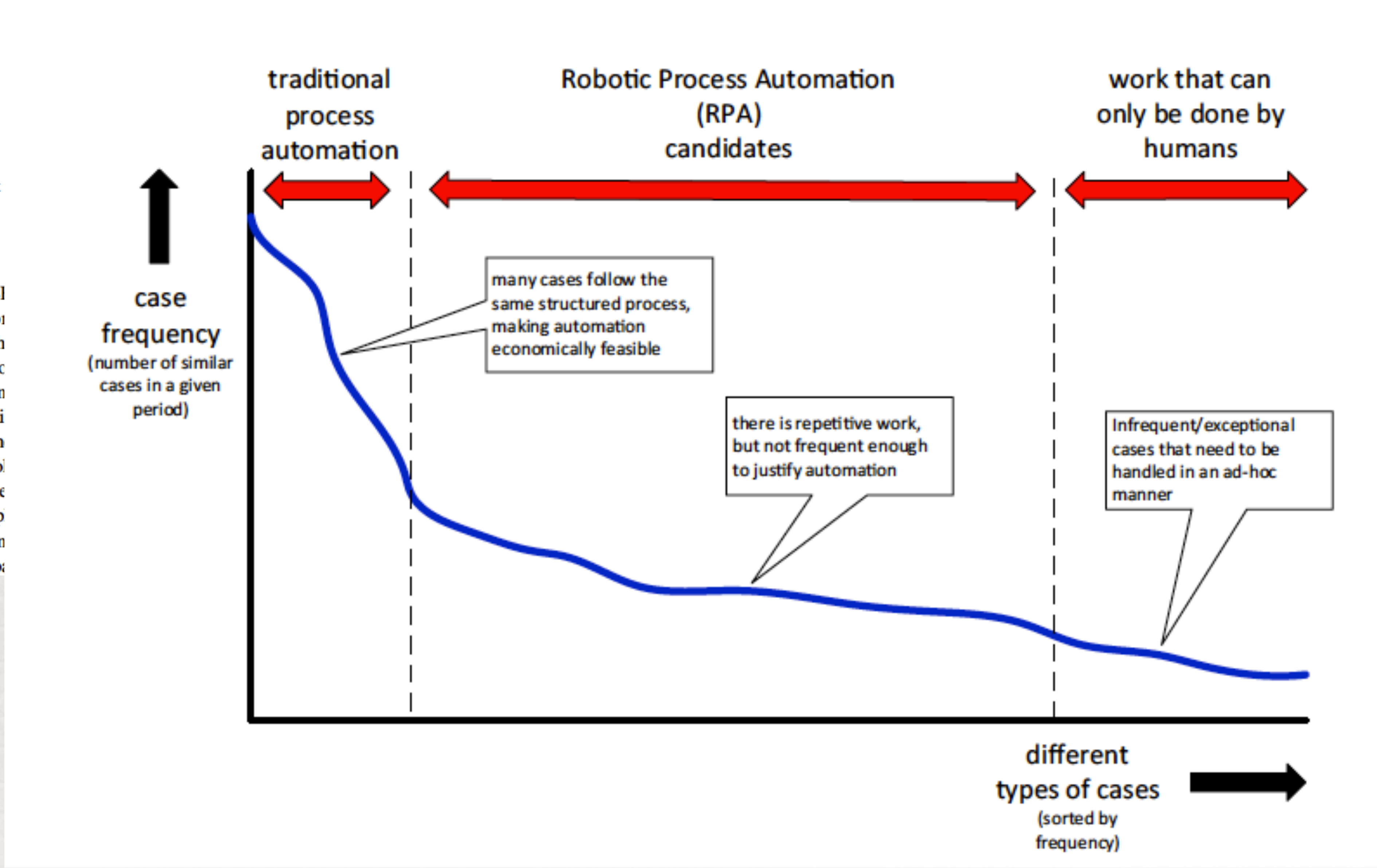
Robotic Process Automation

Wil M. P. van der Aalst · Martin Bichler · Armin Heinzl

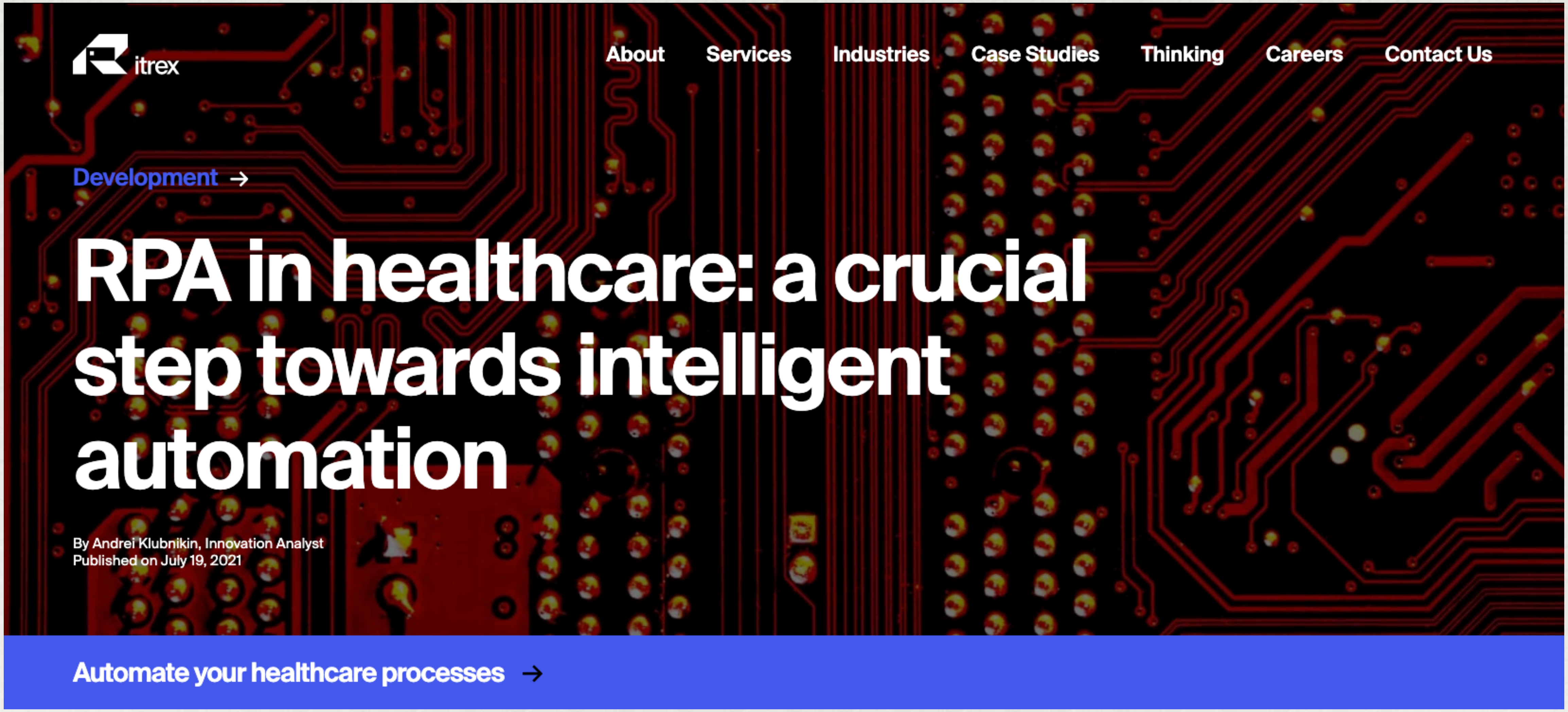
Published online: 14 May 2018

1 Introduction

A foundational question for many BISI Information Systems Engineering) authoi "What should be automated and what sh humans?" This question is not new. Hc ments in data science, machine learnin intelligence force us to revisit this questi Robotic Process Automation (RPA) is on opments. RPA is an umbrella term for tool the user interface of other computer syste human would do. RPA aims to rep automation done in an "outside-in" man from the classical "inside-out" approa



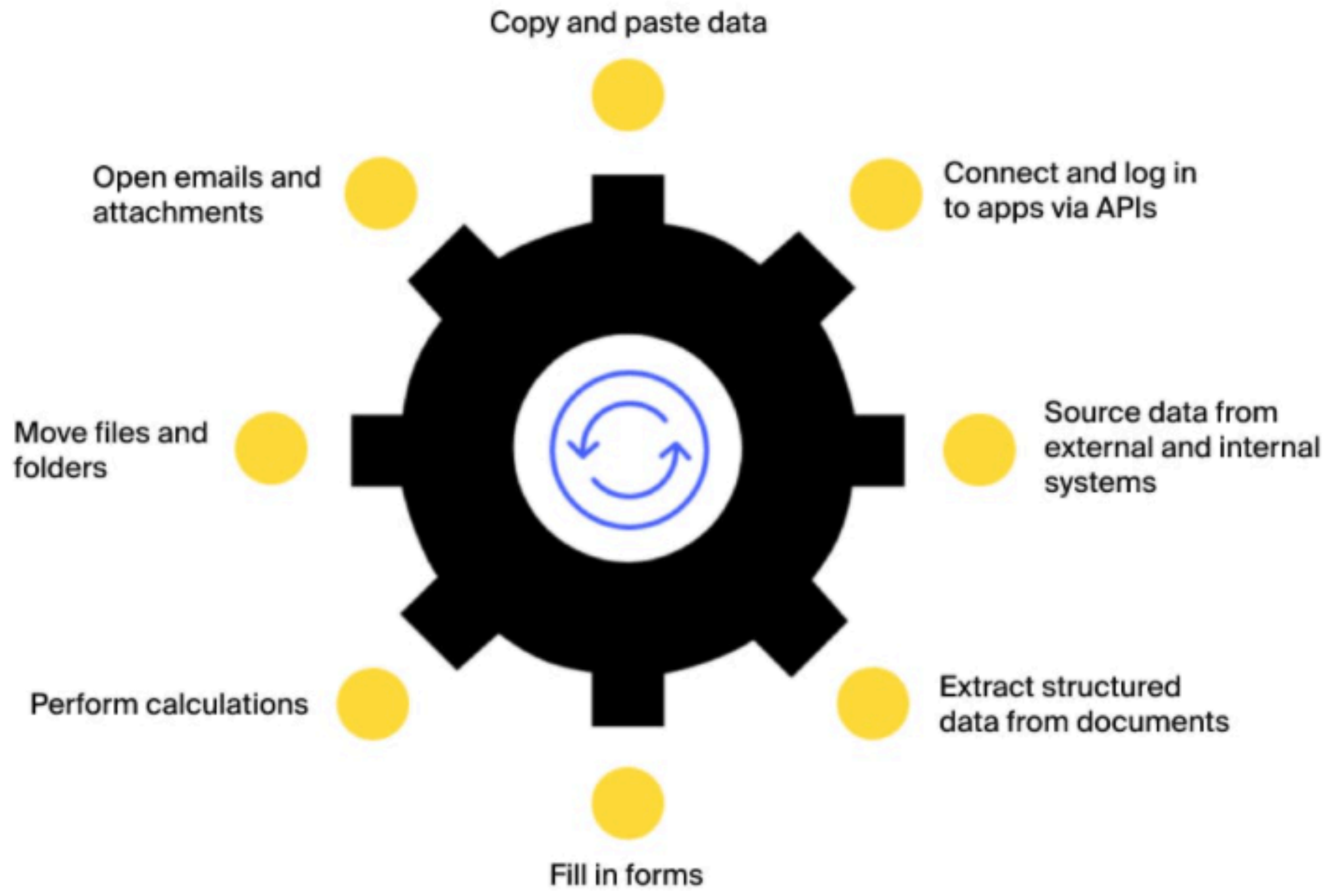
https://itrexgroup.com/blog/rpa-in-healthcare/#header



RPA in healthcare: a crucial step towards intelligent automation

By Andrei Klubnikin, Innovation Analyst
Published on July 19, 2021

RPA can be configured to perform rules and event-driven tasks



Copy and paste data

Connect and log in to apps via APIs

Source data from external and internal systems

Extract structured data from documents

Fill in forms

Perform calculations

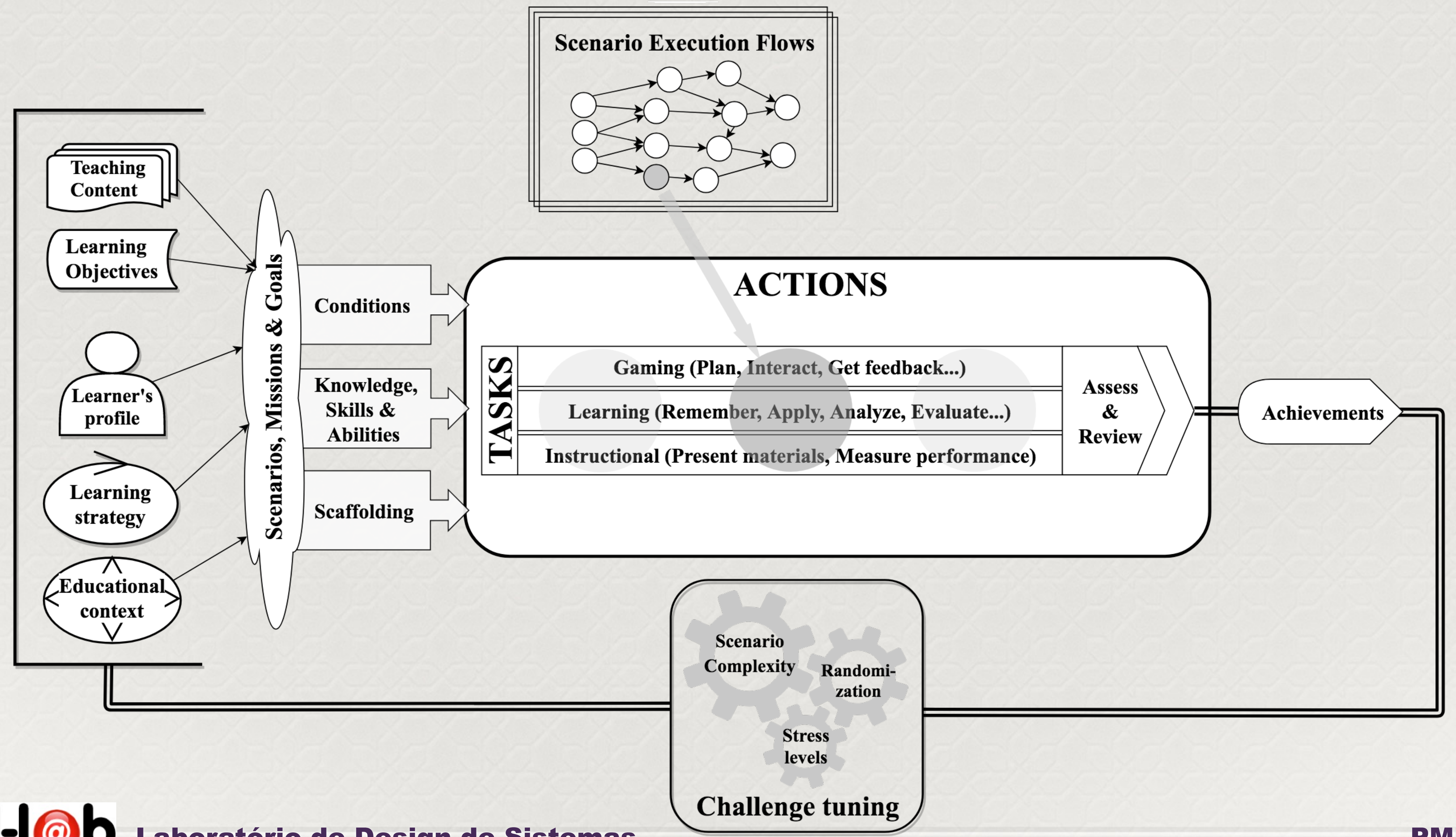
Move files and folders

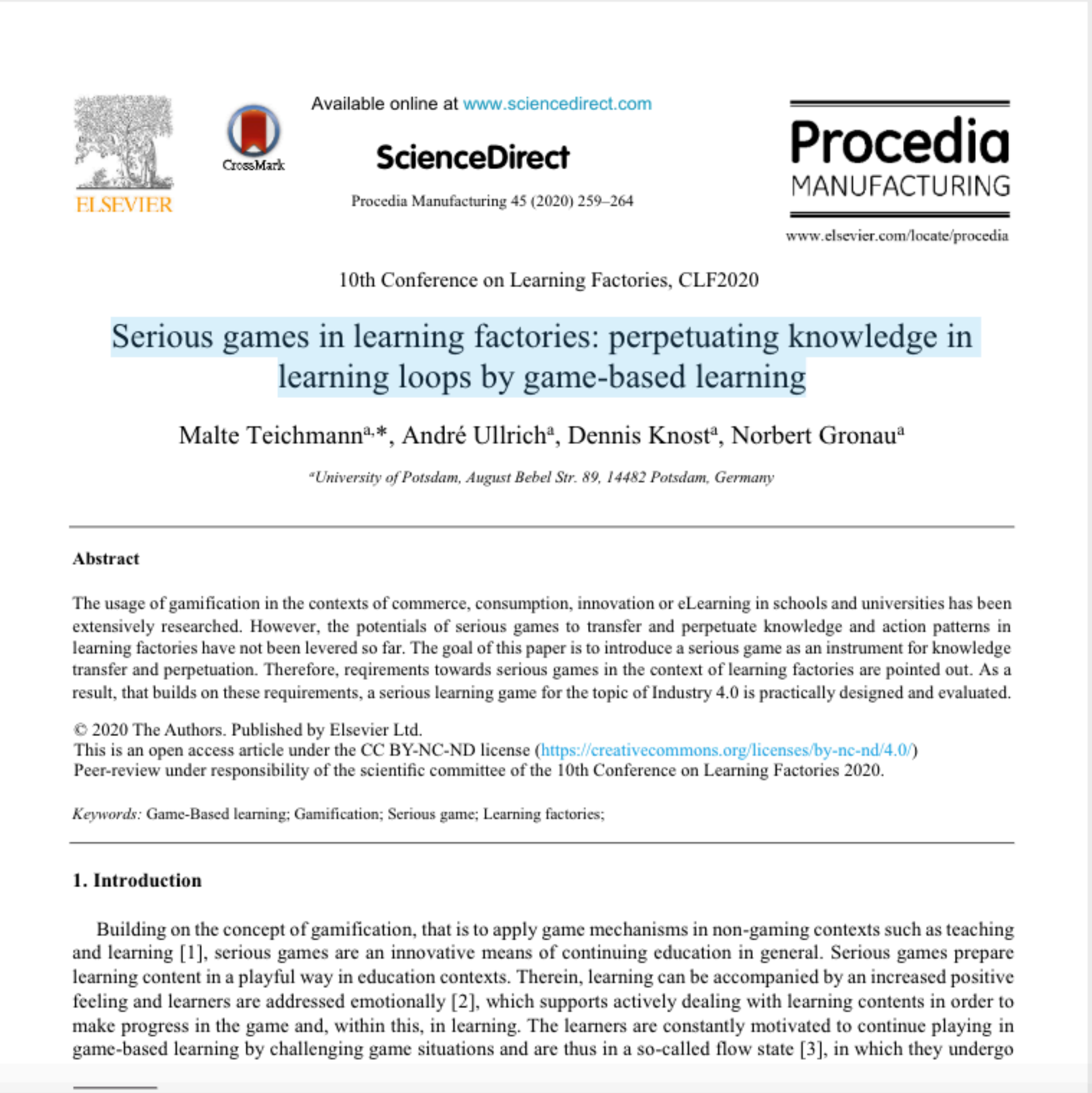
Open emails and attachments

Sistemas RPA inteligentes podem fazer a compatibilização de prontuários, receituários, dados de internação, uso de farmácia popular, etc. de pacientes que usam o serviço público de saúde. Podem ainda checar dados gerados no sistema de saúde complementar ou privado.

Em pesquisa e desenvolvimento, agentes inteligentes de RPA podem ser usados para monitorar, coletar dados e relações transacionais para gerar requisitos em novos projetos de sistemas de grande porte e de com um número muito elevado de usuários.

Sistemas KB e os "serious games"





Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 45 (2020) 259–264

Procedia
MANUFACTURING

www.elsevier.com/locate/procedia

ELSEVIER

CrossMark

10th Conference on Learning Factories, CLF2020

Serious games in learning factories: perpetuating knowledge in learning loops by game-based learning

Malte Teichmann[a],*, André Ullrich[a], Dennis Knost[a], Norbert Gronau[a]

[a]University of Potsdam, August Bebel Str. 89, 14482 Potsdam, Germany

Abstract

The usage of gamification in the contexts of commerce, consumption, innovation or eLearning in schools and universities has been extensively researched. However, the potentials of serious games to transfer and perpetuate knowledge and action patterns in learning factories have not been levered so far. The goal of this paper is to introduce a serious game as an instrument for knowledge transfer and perpetuation. Therefore, reqirements towards serious games in the context of learning factories are pointed out. As a result, that builds on these requirements, a serious learning game for the topic of Industry 4.0 is practically designed and evaluated.

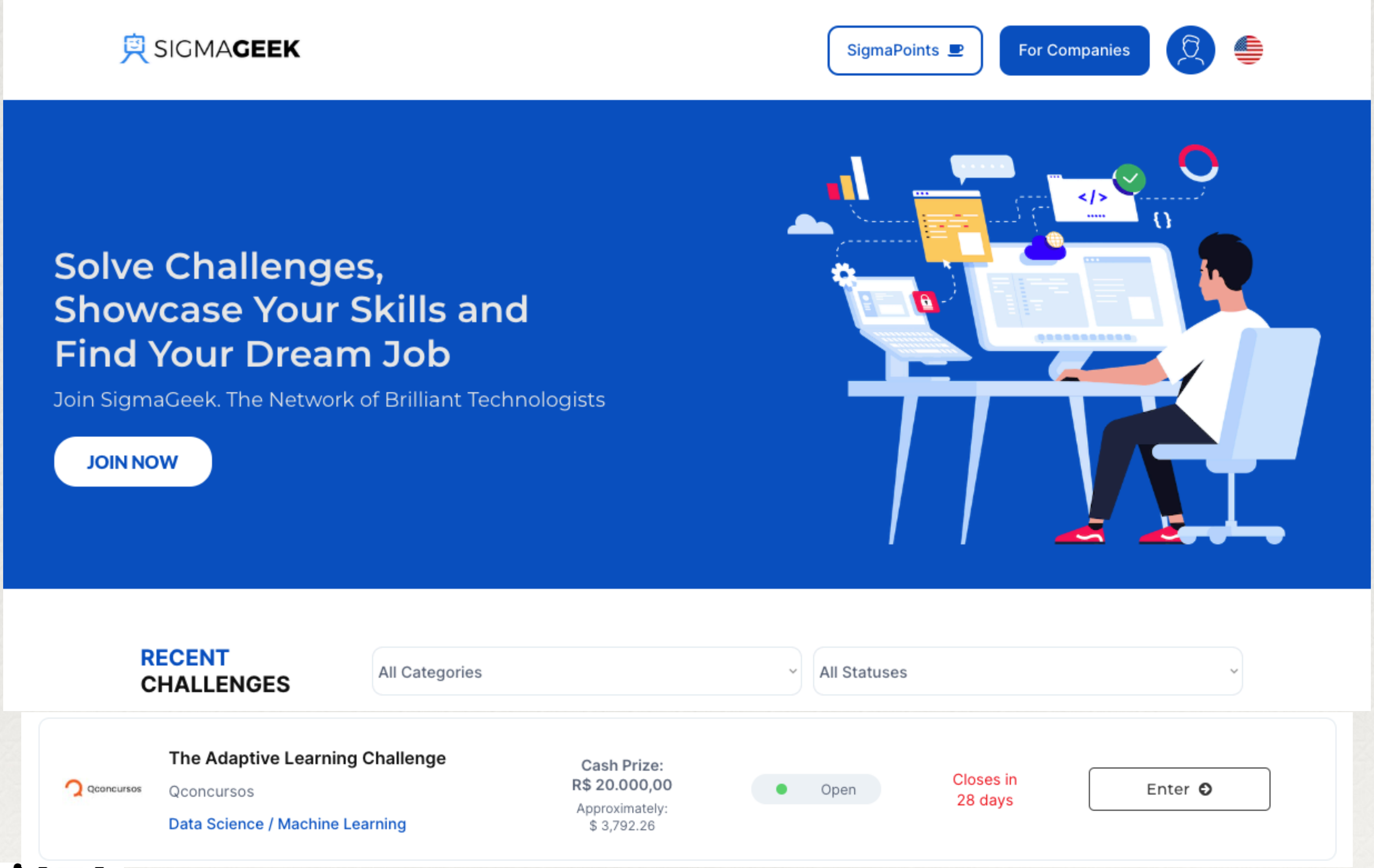
Keywords: Game-Based learning; Gamification; Serious game; Learning factories;

1. Introduction

Building on the concept of gamification, that is to apply game mechanisms in non-gaming contexts such as teaching and learning [1], serious games are an innovative means of continuing education in general. Serious games prepare learning content in a playful way in education contexts. Therein, learning can be accompanied by an increased positive feeling and learners are addressed emotionally [2], which supports actively dealing with learning contents in order to make progress in the game and, within this, in learning. The learners are constantly motivated to continue playing in game-based learning by challenging game situations and are thus in a so-called flow state [3], in which they undergo

Nas próximas aulas vamos tratar com um pouco mais de detalhes sobre a representação de conhecimento e sobre a máquina de inferência e seu funcionamento.

www.sigmageek.com



Solve Challenges, Showcase Your Skills and Find Your Dream Job

Join SigmaGeek. The Network of Brilliant Technologists

RECENT CHALLENGES

Qconcursos	The Adaptive Learning Challenge Qconcursos Data Science / Machine Learning	Cash Prize: R$ 20.000,00 Approximately: $ 3,792.26	● Open	Closes in 28 days	Enter

Perguntas?