Some Preliminary Ideas Towards a Theory of Digital Preservation

Giorgos Flouris
ISTI-CNR
Via Giuseppe Moruzzi, 1
56124, Pisa PI Toscana, Italy
flouris@isti.cnr.it

Carlo Meghini
ISTI-CNR
Via Giuseppe Moruzzi, 1
56124, Pisa PI Toscana, Italy
meghini@isti.cnr.it

ABSTRACT

The problem of digital preservation is one of the most challenging research problems faced by the community of digital libraries today, receiving growing interest by researchers and practitioners alike. One of the major gaps in the related research is the lack of a general agreement on a formal model to describe the problem or on a formal description of the required properties of a good solution to the problem. This work aims to fill this gap by presenting a number of ideas towards a formal, mathematical, logic-based description of preservation as a scientific discipline, to the end of deriving a methodology resting on solid theoretical grounds. We will present and justify a number of desired properties of such a formalism and introduce a model that handles the static aspects of the problem; some ideas related to the dynamics of preservation will be presented as well.

1. INTRODUCTION

The rapid obsolescence of large volumes of digital (especially "born-digital") data is one of the most challenging problems faced by modern archivists. This problem is commonly referred to as the problem of *digital preservation* [10, 15] and deals with the problem of retaining the meaning of a digital object (file, image, database, document, etc) unaltered for an evolving community of readers. Such readers are usually referred to as the *Designated Community* (DC) of the digital object [3, 12].

The problem of digital preservation is not fully understood to date; even though there is a number of ongoing efforts on the practical and methodological aspects of preservation (e.g., [5, 13, 14]), there are very few efforts in the direction of a formal description of the problem [4]. The introduction of such a formal description would in many ways contribute to the research field of digital preservation. For example, a formal theory could allow the development (and proof) of impossibility and existential results: given the inherent difficulties associated with the problem, we intuitively expect some limitations on what types of digital objects can be preserved; we also expect certain types of DC evolution to be such that no preservation is possible. In addition, a formal theory could allow the grounding of existing (and future) preservation methods upon a common formalism for comparison, and could result to a set of formal desirable properties for evaluating such methodologies [8].

Motivated by the above considerations, we propose certain definitions which are part of a larger ongoing effort towards the development of a formal, mathematical, logic-based description of preservation as a scientific discipline, to the end of deriving a methodology resting on solid grounds.

We begin with a general discussion on digital preservation, addressing some general properties of the problem (section 2). This discussion includes some thoughts on the relationship of our ideas with existing standards, such as OAIS [3]; establishing such a relationship is necessary, as it would eventually allow the connection of this work with existing efforts (such as the CASPAR project [2]). Following that, we introduce a formalism that handles the static aspects of the problem (section 3) and present some thoughts related to the dynamic aspects of the problem (section 4).

2. DISCUSSION ON PRESERVATION

2.1 Types of Preservation

As already mentioned, digital preservation refers to the problem of retaining the meaning of a digital object unaltered for an evolving DC. Let us suppose that the digital object under question is an image, say I, created by a particular individual (called the *producer* [12]), say P; moreover, consider a particular member of the DC (called the *consumer* [12]), say C.

The ultimate goal of preservation is to ensure that C understands I despite the many changes that can intervene as time passes by. Understanding in this context implies accessing, of course, but access alone is (usually) not enough. Below, we sketch the general steps required for C to understand I; notice that most of these steps require the use of some artificial agent (software program, hardware device etc) to apply the relevant transformation:

1. The original input is the physical storage (on some form of long-term storage media) of the sequence of bits which encodes the image I in some format.
2. By *reading* these bits from the storage media, C obtains a sequence of bit values representing the image.
3. By *rendering* these bits, C obtains an image that is

some form of light that C's eyes can take in. Rendering presupposes some knowledge on the image *format*.

4. By *interpreting* the image, C figures out its meaning, i.e., the worlds in which the portrayed scene can occur.

The fundamental divide in the above discussion is the separation between rendering the object, the image in our case, and understanding the object. In other words, we regard the above process as *the interpretation of the rendering of the bit stream*. Preservation implies the ability to perform this process *at any time*. This leads to our informal definition of preservation as: *the ability to perform the interpretation of the rendering of a bit stream at any time*. Notice that this involves three steps: producing the bit stream, rendering the produced bit stream and understanding the rendered object. This results to a *decomposition* of the preservation task into sub-tasks, each corresponding to one *preservation type*.

The first type, called *bit preservation*, refers to the ability to produce a particular sequence of bits from a storage media at any time; this can be achieved using error correction techniques, backups, RAID or mirrored disks, media refreshment and other technologies.

The second type, called *data preservation* or *object preservation*, refers to the ability to render the produced bit stream and produce a meaningful output from it at any time. This is the focus of most current approaches to the problem.

The third type, called *information preservation*, refers to the ability to understand the rendered object at any time, i.e., to be able to understand its content by understanding the terms, concepts or other information that appears in it, by placing it in its correct context etc. This is the toughest type of preservation, and is often ignored by existing preservation approaches.

We argue that a complete preservation system should handle all three preservation types. Notice that information preservation applies also for physical objects, whereas the other preservation types only make sense for the realm of digital objects. In what follows, we will not consider bit preservation; for some relevant discussion, refer, for example, to [17]. Our work focuses on information preservation, even though most of the approaches presented here can be easily amended to apply for data preservation as well.

2.2 Preservation in Time and Space

Normally, the process of digital preservation applies when the passage of time renders some digital object incomprehensible by a particular DC. However, we can view preservation as the more general process of allowing an object to be understood by some target DC. The ability of the DC to understand an object may be hindered by several factors, including, but not limited to, the passage of time; the intelligibility of a certain digital object may also depend, for example, on a number of software or hardware modules, or on some background knowledge regarding some particular domain, which may or may not be available to the target DC. This gives rise to two "preservation dimensions": the space dimension and the time dimension.

In the space dimension, the producer needs to formulate the created digital object is such a manner so that the various DCs that he is addressing his data for (which, in general, may have different background knowledge, rendering abilities, hardware, software etc) can understand it.

The time dimension represents the evolution of the knowledge of the DC in time. Such evolution may be, e.g., due to some new discovery, in which case the changes are easy to capture, well-documented and noticeable. However, this is not always the case, as it is possible that the evolution could be due to slight changes in knowledge, jargon, terminology etc, which usually go by unnoticed, but accumulate through time. Thus, the knowledge of the DC should be checked at regular intervals, and, if changes are found, an explicit knowledge shift should be performed to guarantee preservation. This shift consists in the specification of the new knowledge of the DC (i.e., the currently used knowledge) and the change that resulted in this shift.

In fact, both "preservation dimensions" can be essentially reduced to the following problem: given a digital object, carrying a particular meaning, format, etc, as well as a target DC, with some given rendering abilities, software and hardware modules, background knowledge etc, determine the changes required upon the original digital object so that the DC can understand the meaning intended by the object's original producer.

Notice that this formulation makes no reference to the time element, so it avoids the problem of not knowing what a future DC will be like. This way, the preservation problem becomes in many respects similar to a communication problem between two agents and its recursive character is eliminated: we only need to devise a way through which an agent can adequately amend a digital object so as to be understandable by another agent. Once we achieve this, by repeating this process once per agent (i.e., DC), the problem is solved in the space dimension. Moreover, by repeating this process once per agent (i.e., DC) evolution, the DC at time t can play the role of the producer, so as the next-generation DC (at time $t + 1$) will be able to correctly understand the meaning of the digital object, as it was understood by the DC at time t (which is hopefully identical to the meaning intended by the producer at time 0).

2.3 Questions and Answers

In order for preservation to be possible, it is generally necessary for the producer to include in the digital object a certain amount of information on how the object should be interpreted, as well as, possibly, a certain amount of redundancy that will help consumers decipher its meaning. One of the major problems that need to be resolved for preservation is to determine what this information is and how it should be formally represented.

A related issue is which part of the digital object is worth preserving. For example, if the digital object is a text document, then it is composed of various information, including its content, format, fonts, pagination information, attached images or other objects, etc; depending on the context, we may be interested in only a part of this information. Thus, we argue that it is not usually necessary (or possible) to preserve the entire information carried by a digital object; instead, we could isolate and preserve the object's most "useful" or "important" information.

To formalize the above requirements, we will consider that a digital object is a set of *questions* (or *properties*) whose *answers* (or *property values*) will help the consumer understand the (interesting part of the) meaning of the object. Notice that this viewpoint is sufficiently general, as it allows us to include in the preserved digital object some, or all, of the information in the original object, as well as to include additional, external to the original object, associated information that may be useful for preservation purposes.

2.4 Relevant Questions and the OAIS Model

Determining the information (i.e., questions) worth preserving for the object at hand is not an easy task; it depends on the object type, its content, legal issues as well as on the producer's and consumer's needs, among other things. A great aid in this task is provided by preservation models, such as the OAIS standard [3]. The role of such a model in this respect is to provide a methodological framework and a "best practices" approach towards the aim of determining the most important information related to a digital object.

As an illustration, the categories of information that OAIS prescribes are the Content Information (which is in turn divided into Content Data and Representation Information) and the Preservation Description Information (which is in turn divided into Provenance, Reference, Context and Fixity); the Representation Information is further divided into Structural Information and Semantic Information (see [3] for details). Each of those types of information could be modeled as questions about the object.

3. PRESERVATION STATICS

3.1 Required Model Properties

Before performing any preservation activity, we need to formalize a way to represent a digital object as a set of questions and answers. These should be expressed in some language, let's call it $\mathcal{L}$, which will formally determine the syntactical and semantical rules that can be used for formulating such questions and answers.

We will define $\mathcal{L}$ to be a formal language of a logical nature. There are various arguments in favor of this choice. First, $\mathcal{L}$ has to be formal, like logics are, otherwise no scientific theory of preservation can be developed; second, it must be able to express knowledge, and formal logic has been developed for exactly this purpose; third, it must be suitable to capture question-answering, and the inference relation of mathematical logic allows precisely that; and, finally, logic is a very well studied field of science, offering a very rich set of results from which to draw.

There is an overwhelming array of mathematical logics we could use; at this stage, we do not embrace any of them, because this is not necessary for developing a theory of preservation. The only assumptions made about $\mathcal{L}$ is that it allows us to state queries by talking about otherwise unspecified individuals and that it comes with a formal semantics and an associated inference relation $\models$.

Informally, $\mathcal{L}$ can be viewed as the language which must be "spoken" (understood) by someone in order to be able to understand the (questions and answers related to the) digital object under question. In the process of "reading" a digital object (say a text document), we are often able to draw conclusions that are not direct consequences of the document's content, but are partly based on some background or commonsense knowledge. Such background knowledge is necessary for the correct understanding of a digital object, so $\mathcal{L}$ should be coupled with some domain knowledge, represented by a logical theory $\mathcal{T}$, which is expressed in terms of the language $\mathcal{L}$. Following intuition, $\mathcal{T}$ will be assumed finite and consistent.

Notice that a digital object is nothing more than a bunch of symbols unless coupled with some formal structure that provides the semantics to these symbols. This formal structure is the pair $\langle \mathcal{L}, \mathcal{T} \rangle$ which allows us to understand the "meaning" of a digital object; this pair will be called the *Underlying Community Knowledge* (UCK) of the digital object and each digital object will be considered to be associated to a single UCK, which provides the framework for understanding it.

Notice that the content of the UCK depends on the context. For example, if we are interested in data preservation, the UCK would be a formal description of the underlying format of the digital object; if we are interested in information preservation, the UCK would be a formal description of how the rendered object should be interpreted. Moreover, both the producer and the consumer have a UCK of their own; if this UCK is the same, they can both understand the digital object, and no preservation is necessary. Problems emerge when the UCKs of the producer and the consumer are different, in which case a digital object that carries a particular meaning for the producer may carry a totally different meaning for the consumer, or, more likely, be totally unreadable; this is where preservation comes into play.

As mentioned above, $\mathcal{L}$ allows the statement of queries; such queries will be used to formalize questions. Similarly, the individuals being the answers to such queries will be used to formalize the answers to such questions. Answers to questions should normally encode genuine information about the digital object, in the sense that this information is not implicit in the underlying theory $\mathcal{T}$; however, we can imagine situations where this is not necessarily the case. On the other hand, answers cannot contradict our knowledge (i.e., $\mathcal{T}$). Finally, all answers are assumed to be given by a knowledgeable person, which could be either the producer himself or some other person who can understand the digital object well enough to provide information on it.

3.2 Formal Embodiment of our Requirements

We now have all the ingredients we need to fulfill our goal of determining a formal model for the statics of digital preservation. As mentioned above, such a model should contain a UCK (consisting of a formal language, $\mathcal{L}$ and a logical theory $\mathcal{T}$ from $\mathcal{L}$), as well as a digital object (consisting of a set of queries from $\mathcal{L}$, say $\mathcal{Q}$, and a set of answers to each such query, formalized using a function, say *ans*).

More formally, we define the *Underlying Community Knowledge* (or UCK) as a pair $\mathcal{U} = \langle \mathcal{L}, \mathcal{T} \rangle$, where:

- $\mathcal{L}$ is a logical language, or, more formally, a tuple $\mathcal{L} = \langle \mathcal{L}^L, \mathcal{V}, \mathcal{V}^I, \mathcal{P}, \mathcal{P}^C, \models \rangle$, consisting of the following elements:
  - The set of logical symbols of the language, denoted by $\mathcal{L}^L$.
  - The vocabulary $\mathcal{V}$, which is a set of symbols.
  - A set $\mathcal{V}^I$, which is the subset of $\mathcal{V}$ that contains the individuals of the language, defined as all the elements of the vocabulary $\mathcal{V}$ that can be produced as answers to queries ($\mathcal{V}^I \subseteq \mathcal{V}$).
  - A set of well-formed formulas $\mathcal{P}$, which is a nonempty set containing all the formulas that are allowed in $\mathcal{L}$.

- The set $\mathcal{P}^C$ which is the set of closed formulas of the language $\mathcal{L}$. Obviously $\mathcal{P}^C \subseteq \mathcal{P}$. $\mathcal{P}^C$ in effect splits $\mathcal{P}$ into two disjoints sets, namely the set of closed formulas (i.e., $\mathcal{P}^C$ itself) and the other formulas called open formulas and denoted by $\mathcal{P}^O$; obviously $\mathcal{P}^O = \mathcal{P} \setminus \mathcal{P}^C$. Closed formulas will be used to represent facts (e.g., in the theory $\mathcal{T}$), while open formulas represent queries (for $\mathcal{Q}$).
- A binary relation $\models$ between elements of $\mathcal{P}$ (the inference relation of the logic).

- $\mathcal{T}$ is a finite and consistent theory in $\mathcal{L}$: $\mathcal{T} \subseteq \mathcal{P}^C$.

Each *digital object* is associated to a certain UCK $\mathcal{U} = \langle \mathcal{L}, \mathcal{T} \rangle$ and is defined as a pair $\mathcal{D} = \langle \mathcal{Q}, ans \rangle$ where:

- $\mathcal{Q}$ is a finite, non-empty set of queries in $\mathcal{L}$: $\mathcal{Q} \subseteq \mathcal{P}^O$.
- *ans* is a function associating each query $q \in \mathcal{Q}$ with an answer, that is a set of tuples $\vec{a}$ of individuals in $\mathcal{L}$.

We impose a further requirement on *ans*, by asking that the answers, taken all together, do not break consistency. This means to ask the consistency of the theory: $\mathcal{T} \cup \{q(\vec{a}) \mid q \in \mathcal{Q}$ and $\vec{a} \in ans(q)\}$.

Notice that the structure $\mathcal{D} = \langle \mathcal{Q}, ans \rangle$ contains all the questions and answers that were chosen for preservation (see subsection 2.3). Thus, the set of sentences: $\{q(\vec{a}) \mid q \in \mathcal{Q},\ \vec{a} \in ans(q)\}$ is all the information required to enable the interpretation of the part of the digital object that was considered useful for preservation purposes.

Since each preserved digital object is associated to a UCK, we can define the pair $\langle \mathcal{U}, \mathcal{D} \rangle$, or equivalently the 4-tuple $\mathcal{S} = \langle \mathcal{L}, \mathcal{T}, \mathcal{Q}, ans \rangle$, as the *Information Preservation Structure* (IPS) of the digital object. The IPS contains all the information related to the preservation of the digital object, because it contains both the digital object itself (i.e., the questions and answers in $\mathcal{D}$), as well as the description of the meaning of the symbols in $\mathcal{D}$ (i.e., the UCK $\mathcal{U}$).

4. PRESERVATION DYNAMICS

4.1 Preliminary Discussion on the Dynamics

As already mentioned, preservation comes into play when producer's background knowledge is different from the respective consumer's knowledge. Thus, using the terminology introduced so far, the problem of preservation can be defined as follows: given a digital object $\mathcal{D}_O$ whose content (meaning) is understandable using some UCK $\mathcal{U}_O$, a different UCK $\mathcal{U}_N$, and a description of the differences (evolution) between $\mathcal{U}_O$ and $\mathcal{U}_N$, find a digital object $\mathcal{D}_N$, whose content (meaning), understood using $\mathcal{U}_N$, is identical to the content (meaning) of $\mathcal{D}_O$, understood using $\mathcal{U}_O$.

The first problem we have to face in the above process is the identification of the exact changes that led to the new UCK from the old. We argue that the complexity of the UCK structure implies that the changes might be so subtle (or so great) that no automated system (or human being) can determine them by just looking at $\mathcal{U}_O$ and $\mathcal{U}_N$; for example, it is possible that complex changes may overlap and "hide" the effects of each other from an external observer. Therefore, we will make the (reasonable) assumption that preservation takes place while there are still people (human experts) who are knowledgeable of both the new and the old UCK and have kept track and can pinpoint the exact changes that occurred during the UCK evolution.

Given the detailed description of those changes, the purpose of preservation is to determine the changes to apply to the digital object $\mathcal{D}_O$, in order to get the new object, $\mathcal{D}_N$. Such changes should be calculated as a function of the old digital object ($\mathcal{D}_O$), the two UCKs ($\mathcal{U}_O, \mathcal{U}_N$) and the UCK change specification. Notice that this viewpoint allows us to generalize any solutions found, because, once we have found how to preserve an object of some type (i.e., an object associated with some particular UCK) against some particular UCK evolution, we can apply the same solution (function) to all objects associated with the same UCK. For example, if we want to preserve a large number of images of the same format against format obsolescence, all we have to do is determine the correct transformation for one image; then, the same transformation can be applied to the other images.

Our definition makes it clear that, in preservation, the exact syntactical formulation of a digital object is irrelevant; what we are interested in preserving is the *meaning* of the digital object, as derived from the associated UCK.

A final note on the above definition is that it is not always desirable (or possible) to achieve perfect preservation; in some cases, the new DC language ($\mathcal{L}_N$) may be less expressive than the old one ($\mathcal{L}_O$) so the exact meaning of the digital object may not be expressible using $\mathcal{L}_N$; in other cases, part of the meaning of the original digital object may be inconsistent with our current background knowledge ($\mathcal{T}_N$), so, by our definitions and constraints (subsection 3.2), this part should not be preserved.

Combining the above ideas, we conclude that a solution to the problem of preservation should, first, determine a powerful enough formal structure that can describe UCK evolution and, second, define a formal process that will determine the new digital object $\mathcal{D}_N$, as a function of the old ($\mathcal{D}_O$), the two UCKs ($\mathcal{U}_O, \mathcal{U}_N$) and the UCK evolution specification. This function should be such that the meaning of the old digital object is preserved as much as possible, so we should formally define what constitutes "preservation of the meaning" as well.

In the next subsection, we will present some examples that will lead us to some preliminary ideas towards resolving the above issues; a more concrete answer to the above concerns is part of our future work.

4.2 Desired Properties and Examples

Let us consider the example of the evolution of our symbolism from the Roman numerals (I, II, ...) to the Arabic ones (1, 2, ...). An informal description of this evolution could be something like: "the old symbol 'I' evolved to the new symbol '1', the old symbol 'II' evolved to 2, ... etc".

An immediate observation that can be made from this example is that the "language" used to describe the UCK evolution contains terms from both the old (e.g., 'I') and the new (e.g., '1') UCK. Thus, any attempt for a formal description of the UCK evolution should be expressed in a language (i.e., UCK, say $\mathcal{U}_E$) that is at least as expressive as either of $\mathcal{U}_O$ and $\mathcal{U}_N$.

As a second example, let us consider a recent terminology change in the field of astronomy. In August, 2006, during a meeting in Prague, astronomers decided to change the definition of the term "Planet"; in addition, they introduced a new term, "Dwarf Planet" [1]. As a consequence of these changes, Pluto is no longer classified as a planet, but as a dwarf planet.

The main difference of this example with the previous one is that there is no direct 1-1 correspondence between the meaning of the terms of the two UCKs, because there are new terms that don't correspond to any term in the old UCK (e.g., "Dwarf Planet"), there are terms which don't change name but change meaning (e.g., "Planet") and there are terms that change neither name nor meaning, but, due to other terminological changes, their status with respect to other terms does change (e.g., "Pluto").

The above change types are only a small list of the various changes that could occur to terms; thus, the UCK evolution structure should allow fine-grained information to be captured. If there is a term in the new terminology corresponding to a term in the old (like in the first example), we should be able to denote so; if not, we should be able to express as much as we know about the relationships between the old term and the new terminology.

In addition, even though the above discussion is largely limited to vocabulary changes, this is not the only type of change that a UCK may undergo. More difficult are the cases where the logic itself changes where similar problems may occur. According to [7], the changes that our knowledge may undergo can be classified in three broad categories (levels). The first level (level 1, or logic changes) corresponds to changes in the logical formalism used to describe our knowledge (e.g., removal of a logical operator); the second level (level 2, or language changes) corresponds to changes that affect the vocabulary that is relevant to the domain (e.g., the addition of a concept name or predicate name); the third level (level 3, or KB changes) corresponds to changes that affect our knowledge on the relations between the vocabulary elements (e.g., the addition of logical propositions). To the authors' knowledge, preservation is the only real-world problem in which all three change levels are relevant.

4.3 Ideas Towards a Possible Solution

A possible way to resolve the above problems is to use a mapping from each UCK ($\mathcal{U}_O$, $\mathcal{U}_N$) to the expanded one ($\mathcal{U}_E$). The semantics of this mapping, say f, is that an element x from $\mathcal{U}_O$ (or $\mathcal{U}_N$) "corresponds" to (i.e., has the same meaning as) the element $f(x)$ from $\mathcal{U}_E$. Abusing notation, we will use the same f regardless of whether x is a term, a language symbol, an open formula etc. In effect, f corresponds to a mapping from each of the structures comprising $\mathcal{L}_O$ and $\mathcal{L}_N$ to the respective structure in $\mathcal{L}_E$.

Thus, a structure describing the evolution of the UCKs should consist of an expanded UCK ($\mathcal{U}_E$) and a mapping (f) that provides the correspondences between the various elements of $\mathcal{U}_O$, $\mathcal{U}_N$ with $\mathcal{U}_E$. Using f, we can define what it means to retain the meaning of an element: an element y of $\mathcal{U}_N$ *retains the meaning of* x of $\mathcal{U}_O$ iff $f(x) = f(y)$.

To capture more complex interrelationships between elements of $\mathcal{U}_O$ and $\mathcal{U}_N$, we will use the theory of $\mathcal{U}_E$ (namely $\mathcal{T}_E$) and the $\models_E$ relation of $\mathcal{U}_E$. In particular, to capture a complex terminological relationship between the terms x (of the old UCK) and y (of the new UCK), we represent this relationship using a formula relating $f(x)$, $f(y)$ in terms of $\mathcal{U}_E$ and include it in $\mathcal{T}_E$. Similarly, to capture complex logical relationships between formulas x (of the old UCK) and y (of the new UCK), we include the respective relationship (between $f(x)$, $f(y)$) into the $\models_E$ relation.

The next step is to define what it means for a digital object to *preserve* another. A straightforward definition that uses the notion of "retaining the meaning" is too restrictive, as it is based on both the syntax and the semantics of the involved objects (rather than just the semantics).

Thus, it would make more sense to use some notion of "equivalence" that will allow us greater flexibility on how to preserve a digital object. This idea leads to a number of different definitions, depending on how we formally interpret the term "equivalence". Probably the most interesting way to define this notion is as follows: a digital object $\mathcal{D}_N = \langle \mathcal{Q}_N, ans_N \rangle$ associated to $\mathcal{U}_N = \langle \mathcal{L}_N, \mathcal{T}_N \rangle$ *preserves* $\mathcal{D}_O = \langle \mathcal{Q}_O, ans_O \rangle$, associated to $\mathcal{U}_O = \langle \mathcal{L}_O, \mathcal{T}_O \rangle$ iff $\mathcal{T}_E \cup D_{OE} \equiv_E \mathcal{T}_E \cup D_{NE}$, where: $D_{OE} = \{f(q(\vec{a})) \mid q \in \mathcal{Q}_O, \vec{a} \in ans_O(q)\}$, $D_{NE} = \{f(q(\vec{a})) \mid q \in \mathcal{Q}_N, \vec{a} \in ans_N(q)\}$.

According to this definition, to determine whether $\mathcal{D}_N$ preserves $\mathcal{D}_O$, we take each question-answer pair of the old digital object and map it into its "corresponding" formula in $\mathcal{U}_E$ (using f); the results, taken together, constitute D_{OE}, which is combined with the information on the relationships between the terminology of the old and the new UCK (i.e., the background knowledge of the expanded UCK, $\mathcal{T}_E$). The same process is followed for the new digital object. The definition states that preservation is achieved iff the respective results (for $\mathcal{D}_O$, $\mathcal{D}_N$) are equivalent (under $\models_E$).

As already mentioned, preservation cannot always be perfect; to capture such cases, we would also need to define some notion of *partial* or *approximate* preservation; this is part of our future work.

The final step in the definition of a preservation model is the development of a formal process that will determine the new digital object (i.e., the one that preserves the old) as a function of the old digital object, the two UCKs and the description of the evolution between the two UCKs. To resolve this problem, we need to identify those formulas from $\mathcal{U}_E$ which (a) have an equivalent in $\mathcal{U}_N$ (through f), and, (b) taken together, they satisfy the condition for preservation given above. The exact determination of a step-by-step process for this task is also part of our future work.

4.4 Representing Evolutions

The above structures are useful for theoretical manipulations, but are rather cumbersome in practice without some adequate compact representation. In this respect, the two well-established fields of ontology evolution [11] and belief revision [9] may be of use; these fields are dealing with the representation and determination of changes upon a corpus of knowledge, which could be an ontology (in ontology evolution) or some formal logical theory (in belief revision).

Even though this is a valid option, it should be emphasized that it would only partly cover our preservation needs. The first reason for this is that neither of these fields deals with level 1 changes [7]. In particular, belief revision only deals with level 3 changes, while ontology evolution deals with changes in levels 2 and 3. This restricts the types of UCK evolutions that these fields can describe and handle.

In addition, most of the developments in these fields are based on certain assumptions on the underlying logic; should the UCK logic be different, most of the relevant literature would be inapplicable. For a recent attempt to (partially) overcome this problem, in a different context, see [6].

Another problem that invalidates this option in certain contexts is the "infiniteness" issue. Both belief revision and ontology evolution use a simple, explicit and straightforward way to represent changes as a list of operations; unfortunately, this would not work in all cases. The example with the Roman and Arabic numerals (subsection 4.2) is an excellent manifestation of this fact: as is obvious from the informal description of that evolution, there is an infinite number of evolutions that took place, one per Roman numeral. Thus, it is not possible to explicitly describe such an evolution in a finite way using the standard methodology; a more compact implicit specification is required.

Unfortunately, this "infiniteness" problem appears more often than not in real-world applications. An everyday example is conversions from one currency type to another, or from one unit of measurement to another (e.g., Celsius degrees to Fahrenheit degrees); in such cases, every symbol (e.g., 18^oC) should be transformed to its equivalent (90^oF) and there is a potentially infinite number of different temperatures (symbols) that could be measured.

One way to address this problem is to describe evolution as the output of a certain algorithm which can be finitely expressed using one of the formalisms developed in computer science (e.g., Turing Machines) [16]. Of course, this option invalidates the use of all representations and methodologies employed in belief revision and ontology evolution.

Despite these deficiencies, we argue that the fields of belief revision and ontology evolution could (and should) be applied for certain types of UCK evolution. Such an option would relieve us from dealing with problems already addressed in these fields, so we believe it's worthwhile to consider it. For example, ontology evolution could handle the astronomy example presented in subsection 4.2.

5. EPILOGUE

This paper reports on an ongoing effort with the ultimate goal of formally modeling the process of digital preservation. We started with a general discussion on the problem, which allowed us to determine the basic properties that such a model should have. This discussion also led to the definition of the vital steps that need to be performed towards this aim, as well as to a number of preliminary proposals that satisfy most of the required properties of such a formalism.

We argued that the process of digital preservation should be described using a model that describes both the digital object under preservation itself (using the questions-answers mechanism) and the general context (semantical, syntactical etc) in which this object is placed (i.e., background knowledge, captured by the UCK structure).

Using these notions, we described the problem of preservation in terms of UCK evolution and argued that, in order to formally model it, we need to define the process that would determine the new digital object as a function of the old digital object, the old and the new UCK, as well as the information on the UCK evolution; the new digital object should be such that the meaning of the old digital object is preserved, so a formal definition of this notion was provided.

We believe that the refinement of those initial ideas will lead to a formal model of digital preservation; such a model would be a significant contribution to the research efforts in the field, as it would allow the development (and proof) of formal results, the grounding of preservation methods upon a common formalism for comparison and the development of a set of formal desirable properties for evaluating preservation methodologies.

6. ACKNOWLEDGMENTS

This work was carried out during the first author's tenure of an ERCIM "Alain Bensoussan" Fellowship Programme; it was partially supported by CASPAR (FP6-2005-IST-033572).

7. REFERENCES

[1] 2006 definition of planet. http://en.wikipedia.org/wiki/2006_redefinition_of_planet.

[2] Caspar: Cultural, artistic and scientific knowledge for preservation, access and retrieval. eu funded project (fp6-2005-ist-033572). http://www.casparpreserves.eu.

[3] *ISO 14721:2003: CCSDS 650.0-B-1: Reference Model for an Open Archival Information System (OAIS). Blue Book, Issue 1*, 2002. available at: http://ssdoo.gsfc.nasa.gov/nost/isoas/ref_model.html.

[4] J. Cheney, C. Lagoze, and P. Botticelli. Towards a theory of information preservation. In *Proceedings of the 5^{th} European Conference on Research and Advanced Technology for Digital Libraries*, 2001.

[5] M. Factor, D. Naor, S. Rabinovici-Cohen, L. Ramati, P. Reshef, and J. Satran. The need for preservation aware storage: A position paper. *ACM SIGOPS Operating Systems Review*, 41(1):19–23, 2007.

[6] G. Flouris. *On Belief Change and Ontology Evolution.* PhD Thesis, University of Crete, Greece, 2006.

[7] G. Flouris. On the evolution of ontological signatures. In *Proceedings of the Workshop on Ontology Evolution*, 2007.

[8] G. Flouris and C. Meghini. Steps towards a theory of information preservation. In *Proceedings of the International Workshop on Database Preservation*, 2007. Invited Talk.

[9] P. Gärdenfors. Belief revision: An introduction. In P. Gärdenfors, editor, *Belief Revision*, pages 1–20. Cambridge University Press, 1992.

[10] H. Gladney. *Preserving Digital Information.* Springer-Verlag, 2007.

[11] P. Haase and Y. Sure. D3.1.1.b state of the art on ontology evolution, 2004. available at: http://www.aifb.uni-karlsruhe.de/WBS/ysu/publications/SEKT-D3.1.1.b.pdf.

[12] B. Lavoie. The open archival information system reference model: Introductory guide. In *DPC Technology Watch Report 04-01*, 2001.

[13] C. Lynch. Canonicalization: A fundamental tool to facilitate preservation and management of digital information. *D-Lib Magazine*, 5(9), 1999.

[14] P. Mellor, P. Wheatley, and D. Sergeant. Migration on request, a practical technique for preservation. In *Proceedings of the 6^{th} European Conference on Research and Advanced Technology for Digital Libraries*, pages 516–526, 2002.

[15] A. Pace. Coming full circle, digital preservation: Everything new is old again. *Computers in Libraries*, 20(2), 2000.

[16] C. Papadimitriou. *Computational Complexity.* Addison Wesley, 1994.

[17] D. Rosenthal. Engineering issues in the preservation of databases. In *Proceedings of the International Workshop on Database Preservation*, 2007.