A Case Study of Long-Running Business Processes: Digital Information Preservation

Yannis Tzitzikas

Assistant Professor, Department of Computer Science, University of Crete
Associate Researcher, Institute of Computer Science (FORTH-ICS)

First SSME Workshop and Summer School
Outline

• What is Digital Information Preservation?
• Why it is important?
• Aspects of Preservation
• Preservation Approaches (/Strategies)
• The OAIS Reference Model
• The CASPAR Project
• On preserving the Intelligibility of Digital Objects
  – Formalizing Intelligibility and Intelligibility Gaps
  – Intelligibility-aware processes
• Concluding Remarks and Directions for Further Research
What is Digital Information Preservation?
Phaistos disk (dated to 1700 BC)

We still cannot understand it (the meaning has not been preserved)
We still don’t know how the pyramids were constructed. (the process has not been preserved)
Digital Objects

How can we be sure that in the future one would be able to understand this byte stream?

100110110000110111011011101110010111100111

089097110110105115

&#89;&#97;&#110;&#110;&#105;&#115

It is “Yannis” in ASCII

How we will preserve the meaning of digital objects?
Digital Objects
The need for preserving the process that created a digital object

How we will preserve the digital process?

- How this image has been derived?
- When and by whom it was taken?
- How the satellite image was processed (by what algorithms and with what parameters)?
I know UML but what this diagram specifies?

- A person cannot start a job before his/her birth
- A promotion cannot lower the salary of an employee

⇒ Now I can develop the system or I can guess how the existing system operates

Yannis Tzitzikas, SSME'07
everything flows nothing stands still

[Heraklitus]
The need for tackling changes

We need to tackle changes in software/hardware and community knowledge

Suppose a tourist agency which keeps a web site where a large number of touristic brochures (for various destinations all over the world) are made available in electronic form. All the material is stored in a digital repository.

Notice that:
- The Flag is no longer valid
- The Country … “does not exist” any more
- The currency is not valid
- We may want to change the **image format** (e.g. gif -> .png)

Metadata
- Format: gif
- City: Maribor
- Country: Yugoslavia
- Currency.type: Yugoslav dinars (YUM)
- Currency.Value: 5
We need to tackle changes because
... everything flows nothing stands still  [Heraklitus]
Tackling changes

Tour of Maribor with only:

Metadata
- Format: giff
- City: Maribor
- Country: Yugoslavia
- Currency.type: Yugoslav dinars (YUM)
- Currency.Value: 5

1977

Format migration

Knowledge update

Tour of Maribor with only:

Metadata
- Format: png
- City: Maribor
- Country: Slovenia
- Currency.type: Slovenian Tolar
- Currency.Value: 3.4

July 2006
Preservation of Digital Information

Why it is important?

• The world produces around 2 exabytes ($2^{60}$) of unique information per year,
  – 90% of which is digital and with a 50% annual growth rate.

• “Everything flows, nothing stands still” [Heraclitus]

• \(\rightarrow\) Digital information **has to be preserved** not only against hardware and software technology changes, but also against changes in the knowledge of the community.
Aspects of Preservation

But what should we preserve?

- For sure we have to preserve the **bits** of the digital objects

We should also try to preserve the **information** carried by the digital objects
  - Their accessibility
  - Their integrity
  - Their authenticity
  - Their provenance
  - Their intelligibility (by human or artificial actors)

Preservation has been termed “**interoperability with the future**”
What are the current preservation approaches and initiatives?
Current preservation approaches

Approaches

• Replication
  – Keep multiple copies

• Refreshing
  – Copy data onto newer media or systems

• Migration
  – Replace digital objects of old formats with "equivalent" objects of new formats.

• Emulation
  – An emulator duplicates (provide an emulation of) the functions of one system with a different system, so that the second system behaves like (and appears to be) the first system.

Standards

• OAIS
  • (will be discussed next)

Ongoing EU Projects

• PLANETS
  • Objective: Support humans in deciding what preservation policy (emulation, migration) to adopt based on criteria like cost, loss of information.

• CASPAR
  • (will be discussed next)
OAIS: Open Archival Information System
(ISO 14721:2003)
OAIS: An archive, consisting of an organization of people and systems, that has accepted the responsibility to preserve information and make it available for a Designated Community (OAIS 1.7.2)

- Development led by the Consultative Committee for Space Data Systems (CCSDS)
- Published in early 2003 as ISO 14721:2003
- Delivers two high-level models:
  - Information Model
  - Functional Model
• **Representation Information**
  - objective: for taking a collection of bits and convert it to something useful
  - key notions: *Structure, Semantics, Algorithms,...*

• **Preservation Description Information**
  - objective: for considering the origins and relevance of any digital information
  - key notions: *Provenance, Fixity, Reference* and *Context*

• **Descriptive Information**
  - role: important for data management, discovery and access
OAIS Information Model

- Information Object
  - Data Object
    - Physical Object
    - Digital Object
  - Representation Information
    - Bit Sequence
    - 1+
  - Interpreted using
- Representation Information
  - Interpreted using
- Structure Information
- Semantic Information
- Other Representation Information
- Software
- Standards
- Algorithms
- Access Software
- Representation Rendering Software
- Bit Sequence
- 1+
- Reference Information
- Provenance Information
- Context Information
- Fixity Information
- Package Description
- Archival Information Package
- Packaging Information
- Content Information
- Further described by
- Preservation Description Information

Yannis Tzitzikas, SSME'07
OAIS Information Model
Kinds of Metadata

class OAIS Information Model

Information Object

Data Object

Representation Information

Physical Object

Digital Object

Bit Sequence

Structure Information

Semantic Information

Software Information

Algorithms Information

interpretedUsing

0..*

1..*

1..*
OAIS Functional Model

Functional Model of OAIS (6 entities):
- Ingest
- Archival Storage
- Data Management
- Administration
- Preservation Planning
- Access

- **SIP**: Submission Information Package
- **AIP**: Archival Information Package (e.g. format) which consist of
  - **IO** (Information Object): Data Object + Representation Information
  - **PDI** (Preservation Description Information): provenance, context, fixity
- **DIP**: Dissemination Information Package
  - is the version of the information package delivered to the Consumer in response to an access request. May differ in form (e.g. TIFF to JPEG) or content (e.g. amount of metadata supplied) to that which resides in the archival store.
The CASPAR project

CASPAR:
Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval

• Ongoing FP6 Integrated Project
• Start: April 2006.
• Duration: 42 months
• EU Funding: € 8 800 000
• Total planned budget: € 16 000 000
CASPAR Objectives

Pioneering framework to support the end-to-end preservation “lifecycle” for scientific, artistic and cultural information based on existing and emerging standards

- to establish the foundation methodology for covering all preservation aspects
- to research, develop and integrate advanced components
- to create the CASPAR framework
- to demonstrate the validity of the CASPAR though testbeds
  - Cultural (UNESCO)
  - Contemporary Arts (CNRS, INA, IRCAM, UofLeeds, …)
  - Scientific (European Space Agency, CCLRC)
The partners of this project are:

- Council for the Central Laboratory of the Research Councils – UK (Coordinator)
- Foundation for Research and Technology - Hellas GR
- European Space Agency, ESRIN - IT
- UNESCO
- Centre National de la Recherche Scientifique - FR
- Institut de Recherche et Coordination Acoustique/Musique – FR
- Institut National de l’Audiovisuel - FR
- Consiglio Nazionale delle Ricerche – IT
- IBM Haifa Research Laboratory - IL
- University of Leeds - UK
- International Centre for Art and New Technologies - CZ
- University of Glasgow - UK
- Università di Urbino - IT

and 4 companies:

- Advanced Computer Systems S.p.A. - IT
- @semantics S.r.l. - IT
- Metaware S.p.A. - IT
- Engineering – Ingegneria Informatica S.p.A. - IT
The project has to tackle a number of problems and we are just in its first year.

Hereafter we will focus on the notion of **intelligibility** of digital objects.


According to OAIS, metadata are distinguished to various categories.

One very important is that of **Representation Information**

- Aim at enabling the conversion of a collection of bits to something useful
In order to abstract from the various domain-specific and time-varying details, we introduce the general notions of **Module** and **Dependency**.

- **Module**
  - We adopt a very general definition. A module could be:
    - a piece of software/hardware module.
    - a knowledge model expressed explicitly and formally (e.g. an Ontology)
    - a knowledge model not expressed explicitly (e.g. GreekLanguage)
  - (the only constraint is that modules need to have a unique identity)

- **Dependency**
  - A module t depends on t’, written t > t’, if t requires t’
  - The meaning of a dependency t > t’
    - t cannot function/be understood/managed without the existence of t’

Note: We model the RI requirements of OAIS as *dependencies* between modules.
Modules and Dependencies: Examples

(a) README.txt
   TEXT EDITOR
   WINDOWS XP
   ENGLISH LANGUAGE

(b) README.txt
   TEXT EDITOR
   WINDOWS XP
   ENGLISH2GREEK DICTIONARY
   GREEK LANGUAGE
• **Scientific Data**

   ![Diagram showing Modules and Dependencies: Examples]

   - **FITS FILE**
     - **FITS STANDARD**
       - **PDF STANDARD**
         - PDF s/w
     - **FITS JAVA s/w**
     - **JAVA VM**
   - **FITS DICTIONARY**
     - **DICTIONARY SPECIFICATION**
       - **XML SPECIFICATION**
       - UNICODE SPECIFICATION

Yannis Tzitzikas, SSME'07
Modules and Dependencies: Examples

• Performing Arts Data

MULTIMEDIA PERFORMANCE DATA

- C3D
  3D motion data files

- DirectX
  3D scene data files

- MAX/MSP
  motion to music mapping strategy
• Cultural Data

Metadata Record

CIDOC CRM CORE

CIDOC CRM STANDARD

CRM CORE XML Schema

RDF STANDARD

XML SPECIFICATION
Modules and Dependencies: Examples

• Semantic Web Data
Formalizing **Modules** and **Dependencies**

- **Objects:** \( \text{Obj}=\{o_1, \ldots, o_n\} \)
- **Components:** \( \text{C}=\{t_1, \ldots, t_k\} \)
- **Modules:** \( T = C \cup \text{Obj} \)
- **Dependencies:** A binary relation over \( T \) (i.e. \( > \subseteq T \times T \))
- **Dependency graph:** \( G = (T, >) \)

**Notations**
- \( S \): a subset of \( T \)
- \( >^+ \): the transitive closure of \( > \)
- \( >^* \): the reflexive and transitive closure of \( > \)
- \( \text{Nr}(t) = \{ t' \mid t > t' \} \)
- \( \text{Nr}^+(t) = \{ t' \mid t >^+ t' \} \)
- \( \text{Nr}^*(t) = \{ t' \mid t >^* t' \} \)
- \( \text{Max}(S) \): the maximal elements of \( S \) w.r.t. \( > \)
Formalizing **Actor/Community knowledge** (in terms of modules and dependencies)

- Each actor or community \( u \) can be characterized by a *profile* \( T_u \) that contains those modules that are assumed to be available/known to \( u \).
- Formalization: \( T_u \subseteq T \)

**Examples**

- \( u \) is an artificial agent
  - \( T_u \) may include the software/hardware modules available to it
- \( u \) is a human,
  - \( T_u \) may include modules that correspond to implicit knowledge

**Unique Module Assumption** (UMA)

- Each module is uniquely identified by its name and its required modules are always the same (more practical: different modules have different identities)
The notion of **closure**
(of modules and profiles)

- Closure of a module $t$: $C(t) = Nr^*(t)$
- Closure of a set of modules $S$: $C(S) = \bigcup \{ C(t) \mid t \in S \}$
- Required modules of $t$: $C^+(t) = C(t) - \{t\} = Nr^+(t)$
- Closure of a profile $Tu$: $C(Tu) = Nr^*(Tu)$

$\mathcal{T}$

- $C^+(tx) = C(tx) - \{tx\}$ // modules required by $tx$
- $C^+(ty) = C(ty) - \{ty\}$

- Closure of $Tu$

- It is assumed that $u$ knows $C(Tu)$. 
Intelligibility and Intelligibility Gap

• **Intelligibility**
  – Definition (dictionary)
    • 1. Capable of being understood: *an intelligible set of directions*.
    • 2. Capable of being apprehended by the intellect alone.

• **Intelligibility Gap**
  – Definition:
    • The smallest set of modules *u* needs to have in order to understand a module *t*.
  – Notation
    • **Gap(t,u)**: The intelligibility gap between a user *u* with profile *Tu* and a module *t*.
Intelligibility and Intelligibility Gap (I)

- u can understand t iff: $C^+(t) \subseteq C(T_u)$
- The intelligibility gap: $\text{Gap}(t, u) = C^+(t) - C(T_u)$

Gap$(t_y, u) = \emptyset$

Gap$(t_x, u) = \{t_1, t_2, t_4, t_5\}$
Intelligibility and Intelligibility Gap (II)

• u can understand t iff: $C^+(t) \subseteq C(T_u)$

Due to UMA we can write:

• $C^+(t) \subseteq C(T_u) \iff \max(C^+(t)) \subseteq C(T_u)$

• In our example
  – $\max(C^+(t_y)) = \{t_3\} \in C(T_u)$
  – $\max(C^+(t_x)) = \{t_1\} \not\in C(T_u)$
We can capture emulation and migration by introducing converters (as a different kind of edges). Intelligibility gaps can be filled with converters and finding the appropriate converters reduces to the problem of REACHABILITY in directed graphs.
The extension of the filename gives us a hint about the type of the digital object, so we may write \( \text{type(mypage.html)} = \text{HTML} \), and as \( \text{mypage.html} > \text{HTML} \), we can in general assume that: for every \( t \) it holds: \( t > \text{type}(t) \), if \( \text{type}(t) \) is known.

However **only if HTML is intelligible** we can realize that: \( \text{mypage.html} > \text{HTML} \)

We need to have an HTML parser. If we cannot understand HTML then we cannot deduce the dependency \( \text{mypage.html} > \text{JPG} \)

In general, \( \text{type}(o) = \text{type}(o) \cup \text{type} \left( \text{contents}(o) \right) \)

To compute \( \text{contents}(o) \) we need to be able to understand \( \text{type}(o) \)
The extension of the filename gives us a hint about the type of the digital object, so we may write
\[
\text{type}(\text{mypage.html}) = \text{HTML}
\]
and as \(\text{mypage.html} > \text{HTML}\), we can in general assume that:
for every \(t\) it holds:
\[
\text{type}(t) \subseteq \text{type}(t)
\]
However only if HTML is intelligible we can realize that:
\[\text{mypage.html} > \text{HTML}\]

We need to have an HTML parser. If we cannot understand HTML then we cannot deduce the dependency \(\text{mypage.html} > \text{JPG}\)
In general, \(\text{type}(o) = \text{type}(o) \cup \text{type}(\text{contents}(o))\)
To compute \(\text{contents}(o)\) we need to be able to understand \(\text{type}(o)\)

\[
\begin{align*}
\text{Mypage.html} & \quad \text{HTML} \\
\text{HTML} & \quad \text{JPG}
\end{align*}
\]
A Preservation Information System could adopt the following policies

- **Input Policy**
  - The **input** (e.g. data objects to be archived) **should be intelligible by the system**

- **Output Policy**
  - The **output** (e.g. returned answers) **should be intelligible by the recipients**

The notion of **profile** could be used as gnomon in these policies

Intelligible wrt the **profile of the system** (say Tp)

Intelligible wrt the **profile of the user** (say Tu)
Intelligibility-aware Interaction Schemes

Consider the classical **query-and-answer** interaction scheme between:

- an information provider \( p \) and
- an information consumer \( u \)

(1): \( u \rightarrow p: \text{query}(q) \)
(2): \( p \rightarrow u: \text{answer}(A) \)

We will extend the query-and-answer interaction scheme with intelligibility-related concerns
Intelligibility-aware Interaction Schemes
Case: p stores the dependency graphs and the profiles
Intelligibility-aware Interaction Schemes
For Delivering Intelligible Answers > with fixed Number of Messages

Scheme (I)
Answers are accompanied by their closure

(1) \( u \to p: \text{query}(q) \)
(2) \( p \to u: \text{return}(A, C(A)) \)

Scheme (II)
u sends \( Tu \) with the query (or registers it), \( p \) returns answers accompanied by the intelligibility gap

(1) \( u \to p: \text{query}(q, Tu) \)
(2) \( p \to u: \text{return}(A, \text{Gap}(A,u)) \)

Step 1 can be replaced by
(1’) \( u \to p: \text{query}(q, \text{Max}(Tu)) \)
Intelligibility-aware Interaction Schemes
For Delivering Intelligible Answers > with fixed Number of Messages

Scheme (I)
Answers are accompanied by their closure

1. \( u \rightarrow p: \text{query}(q) \)
2. \( p \rightarrow u: \text{return}(A, C(A)) \)

Scheme (II)
u sends \( Tu \) with the query (or registers it), \( p \) returns answers accompanied by the intelligibility gap

1. \( u \rightarrow p: \text{query}(q, Tu) \)
2. \( p \rightarrow u: \text{return}(A, \text{Gap}(A, u)) \)

Step 1 can be replaced by
1. \( u \rightarrow p: \text{query}(q, \text{Max}(Tu)) \)

• May expensive to compute and large in size
Intelligibility-aware Interaction Schemes
Case: p does not store user profiles

dependency graph only
Intelligibility-aware Interaction Schemes
For Delivering Intelligible Answers > \textit{progressive method}

Scheme (I')
Gradual identification and completion of the intelligibility gap
The provider does not know Tu.

Answers are accompanied by their direct requirements
(1) $u \rightarrow p$: query($q$)
(2) $p \rightarrow u$: return($A$, $\max(C(A))$) // $\equiv$ return($A$, $\text{Nr}(A)$) $\equiv$ return($A$, $\text{directReqOf}(A)$)
(3) $u$: repeat
(4) $u$: $M := \text{recmsg} - Tu$ // or $M := \text{recmsg} - C(Tu)$
(5) $u$: if $M \neq \emptyset$ then
(6) $u \rightarrow p$: getDirectReqsOf($M$)
(7) $p \rightarrow u$: return($\max(C(\text{recmsg}))$) // $\equiv$ return($\text{Nr}(\text{recmsg})$)
(8) $u$: until $M = \emptyset$

\textbf{Fast and small in size}
It is analogous with the previous case (we revert the roles of p and u):

- we ignore the query submission step
- we consider that the user u is the preservation system who wants to ingest the set of objects A that user p sends to u.
Preservation-related processes and intelligibility-related concerns

- Ingest and Archive
  - Ensure the intelligibility by the system, adopt a self-describing (wrt a profile) packaging approach.
- Disseminate
  - Deliver intelligible information packages
- Curate
  - Identify risks of obsolescence, react to changes, select preservation policy to adopt
- Clean
  - Estimate what is worth preserving. Delete the rest
Summary and Concluding Remarks
Intelligibility of Digital Objects

• Intelligibility is an important notion of preservation.
• We formalized this notion on the basis of dependencies. The notion of dependency is ubiquitous and dependency management is an important requirement that is subject of research in several (old and new emerged) areas, from software engineering to ontology engineering.

• A modern digital information preservation system should be generic, i.e. able to preserve heterogeneous digital objects which may have different interpretation of the notion of dependency.

• Contribution
  – Abstract notion of module and dependency
  – The notion of DC Profile: gnomon for deciding intelligibility
    • representation information adequacy (during input)
    • intelligibility (during output).
  – Intelligibility Gap
  – Intelligibility-aware processes
Future research

- Extend the theoretical framework with Converters (for capturing migration/evolution): they can be considered as a specialization of the notion of module.
- Study the effects of changes (on modules, dependencies) and notification services
- Study modules and dependencies of different granularity
- Study properties of dependency relations (transitivity, acyclicity, …)
- Relax the notion of identify (incorporate the notion of similarity and the notion of Diff)

For more see


Proof-of-concept prototype

- Based on Semantic Web Technologies.
Summary and Concluding Remarks

General

- Digital preservation is an endless-process which poses a number of challenging problems

Thanks for your attention

Yannis Tzitzikas, SSME'07