Royal Netherlands Academy of Arts and Sciences (KNAW)

KONINKLIJKE NEDERLANDSE AKADEMIE VAN WETENSCHAPPEN

Epimenides: Interoperability Reasoning for Digital Preservation

Kargakis, Yannis; Tzitzikas, Yannis; van Horik, M.P.M.

published in
PRES 2014 - Proceedings of the 11th International Conference on Preservation of Digital Objects
2014

document version
Publisher's PDF, also known as Version of record

Link to publication in KNAW Research Portal

citation for published version (APA)
Kargakis, Y., Tzitzikas, Y., & van Horik, M. P. M. (2014). Epimenides: Interoperability Reasoning for Digital Preservation. In *PRES 2014 - Proceedings of the 11th International Conference on Preservation of Digital Objects* (pp. 110). State Library of Victoria.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the KNAW public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain.
• You may freely distribute the URL identifying the publication in the KNAW public portal.

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:
pure@knaw.nl

Download date: 07. Oct. 2025

iPRES 2014

MELBOURNE | 6-10 OCTOBER

Proceedings of the 11th International Conference on Digital Preservation

Epimenides: Interoperability Reasoning for Digital Preservation

Yannis Kargakis
Institute of Computer Science, FORTH-ICS
Greece
kargakis@ics.forth.gr

Yannis Tzitzikas
Institute of Computer Science, FORTH-ICS
Computer Science Department, University of Crete, Greece
tzitzik@ics.forth.gr

René van Horik
Data Archiving and Networked Services, DANS
Netherlands
rene.van.horik@dans.knaw.nl

ABSTRACT

This paper presents `Epimenides`, a system that implements a novel interoperability dependency reasoning approach for assisting digital preservation activities. A distinctive feature is that it can model also *converters* and *emulators*, and the adopted modelling approach enables the *automatic reasoning* needed for reducing the human effort required for checking (and monitoring) whether a task on a digital object (digital collection in general) is performable. Finally, the paper describes (in the form of scenarios) concrete preservation activities of a research data archive (DANS) and elaborates on how `Epimenides` could be used and the benefits that would bring.

Keywords

Conversion/Emulation, Dependency Management, Automated Reasoning, Case Study

1. INTRODUCTION

Can we achieve interoperability without necessarily having to rely on standards, but by combining existing software? This question is complex and difficult to answer, therefore the adoption of (or at least assistance from) an automated reasoning approach is beneficial. This is the objective of the migration and emulation-aware dependency reasoning that was presented in [12] (more in [6]). This paper describes the system `Epimenides`, the first system that implements this automated reasoning approach for digital preservation. The paper also elaborates on how it can be used in practice by a research data archive such as DANS (Data Archiving and Networked Services, NL).

We can convey the main message of our approach through an example. Consider a user, say Yannis, who would like to compile and run on his mobile phone, software source code written many years ago, e.g. software code written in the Pascal programming language, stored in a file named

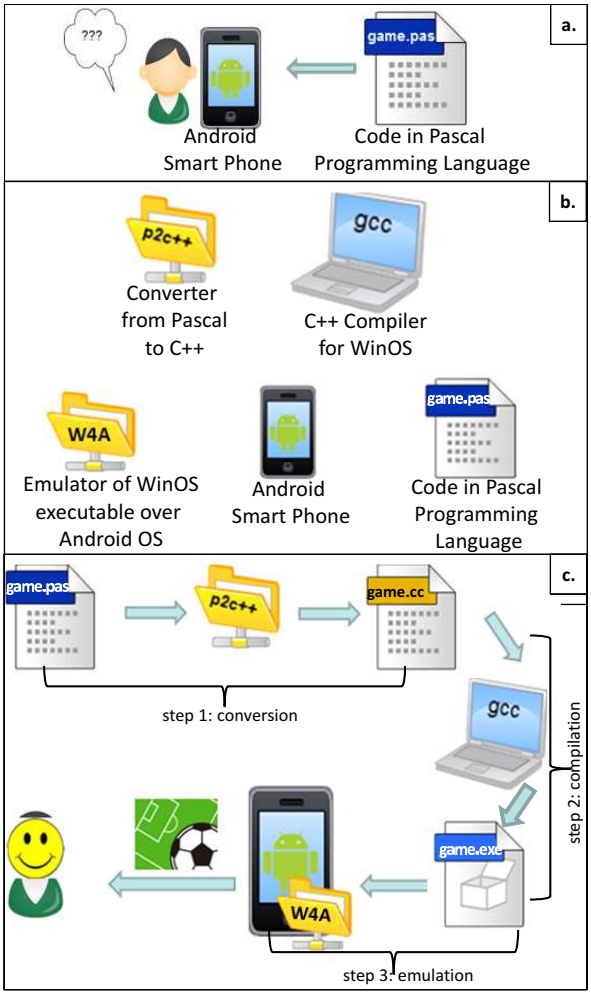

Figure 1: Running example. (a) The problem, (b) The available modules, (c) A series of conversions/emulations to achieve our objective

`game.pas`. For example consider the situation illustrated in Figure 1a. *What can Yannis do? (to achieve his objective), What should we (as a community) do?, Do we have to develop a Pascal compiler for Android OS?, Do we have to standardize programming languages?* The direction and answer of the above questions (according to the approach that `Epimenides` follows), is that it is worth investigating whether

iPres 2014 conference proceedings will be made available under a Creative Commons license. With the exception of any logos, emblems, trademarks or other nominated third-party images/text, this work is available for re-use under a Creative Commons Attribution 3.0 unported license. Authorship of this work must be attributed. View a copy of this licence at `http://creativecommons.org/licenses/by/3.0/legalcode`

it is already possible to compile and run that code on android by "combining" existing software, i.e. by applying a series of transformations and emulations. To continue this example, suppose that we have at our disposal only the modules that are shown in Figure 1b. Someone could then think that we could run `game.pas` on his mobile phone in three steps: by first converting the Pascal code to C++ code, then compiling the C++ code to produce executable code, and finally by running over the emulator the executable yielded by the compilation. Indeed, the series of conversions/emulations shown in Figure 1c could achieve our objective. However, one might argue that this is very complex for humans. Indeed this is true. We believe that such reasoning should be done by computers, not humans. `Epimenides` enables this kind of *automated reasoning*.

The contributions of this paper are:

- its presents `Epimenides`, a system offering novel interoperability reasoning services for digital preservation
- it presents an analysis of digital preservation scenarios of DANS, and shows how `Epimenides` could be used in these scenarios.

The rest of this paper is organized as follows. Section 2 discusses the context and the direction of this line of research. Section 3 presents the system `Epimenides`. Section 4 describes the scenarios provided by DANS and what `Epimenides` could do in each of them. Finally Section 5 concludes the paper.

2. CONTEXT, DIRECTION & RELATED WORK

The proposed methodology aims at offering a coherent approach for handling *interoperability dependencies*. Digital objects and digital collections should remain usable, i.e. one (human or artificial agent) should be able to understand and *use* the digital material over time. This is related to *interoperability*, and for this reason digital preservation has been termed *"interoperability with the future"*. Each interoperability objective or challenge (like those that were listed in [5], [9]) can be considered as a kind of demand for the *performability of a particular task* (or tasks). We can identify various tasks, which in many cases are layered. Examples of tasks include: *rendering* (for images), *compiling* and *running* (for software), *getting the provenance* and *context* (for datasets), etc. In every case the performance of each task has various *prerequisites* (e.g. operating system, tools, software libraries, parameters, representation information etc). We call these *dependencies*. The definition and adoption of standards (for data and services), aids interoperability because it is more probable to have (now and in the future) systems and tools that support these standards, than having systems and tools that support proprietary formats. From a dependency point of view, standardization essentially reduces the dependencies and makes them more easily resolvable; *even though it does not eliminate dependencies*. In all cases (standardization or not), we cannot achieve interoperability when the involved parties are not aware of the dependencies of the exchanged artifacts. However, the ultimate objective is the ability to perform a task, not the compliance to a standard, nor the availability of extra information. An important observation is that even if a digital object is not compliant to a standard, there may be tools and processes that enable the performance of a task on that object. However, as the scale and complexity of information assets and systems evolves towards overwhelming the capability of human archivists and curators (either system administrators, programmers or designers), it is important to aid this task, by offering services that can check whether it is feasible to perform a task over a digital object. For example a software written in 1986 could be executed on a 2013 platform, through a series of conversions and emulations. The process of checking whether this is feasible or not is too complex for a human and this is where *automated reasoning services* could contribute. Such services could greatly reduce the human effort required for periodically checking (monitoring) whether a task on a digital object is performable.

Towards this vision, in the context of APARSEN (Deliverable D25.2 [6]), past rule-based approaches for dependency management ([10], [8], [11]) were advanced for being able to capture converters and emulators. GapMgr[1] and PreScan[2] [7] are two systems that have been developed based on the dependency management model of past approaches [8], [11]. The new proposed modeling [6] enables the desired reasoning regarding task performability taking also into account the capabilities offered by *converters* and *emulators*. The prototype system `Epimenides` (which is the focus of the current paper) is the first system that realizes this approach and demonstrates its functionality.

Another related work is the TIMBUS[3] project. TIMBUS [2] is an EU co-funded project focuses on the preservation of business processes. It employs reasoning-based enterprise risk management to identify preservation risks, mitigation options and to determine the options' cost-benefit. It determines the metadata that needs to be captured and the dependencies (software and hardware components) of relevant process. However there are currently no publicly available TIMBUS software products that exploit this reasoning. In addition, there are several works that can assist various task of the digital preservation area. For example there are tools for the identification of file formats (e.g. DROID[4], Jhove[5], Apache Tika[6]), for getting the details about a technical environment (e.g. TOTEM [1], Preservation Network Model (PNM) [4]) and for getting assistance in preservation planning (e.g. Plato[3]). However none of the aforementioned works offers an automated reasoning for checking whether a task can be performed over a digital object, which is the ultimate objective in a digital preservation strategy.

3. THE EPIMENIDES PROTOTYPE SYSTEM

As stated `Epimenides` is the first system that realizes the approach described in [12, 6]. Its implementation is based on W3C standards (e.g. HTML, CSS, RDF, SPARQL), and its Knowledge Base (expressed in RDF/S) contains information about all MIME types and the modeling of various quite common tasks. Since it is based on Semantic Web technologies it can be straightforwardly enriched with in-

[1]http://athena.ics.forth.gr:9090/Applications/GapManager/
[2]http://www.ics.forth.gr/isl/PreScan
[3]http://timbusproject.net/
[4]http://digital-preservation.github.io/droid/
[5]http://jhove.sourceforge.net/
[6]http://tika.apache.org/index.html

formation coming from other external sources (i.e. other SPARQL endpoints).

Epimenides is a web accessible system[7], it can be used by several users (and each of them can define and maintain his/her own profile). Fundamental notions of Epimenides are *module*, *dependency* and *profile*. A module can be a software/hardware component or even a Knowledge Base (KB) expressed either formally or informally, explicitly or tacitly, that we want to preserve. A profile is the set of modules that are assumed to be known (available or intelligible) by a user, and this notion allows controlling the number of dependencies that have to be recorded formally.

3.1 Use Cases

In brief Epimenides offers the following services: (a) Task-Performability Checking, (b) Consequences of a Hypothetical Loss and (c) Identification of Missing Modules. A Use Case Diagram providing an overview of the supported use cases is given in Figure 2.

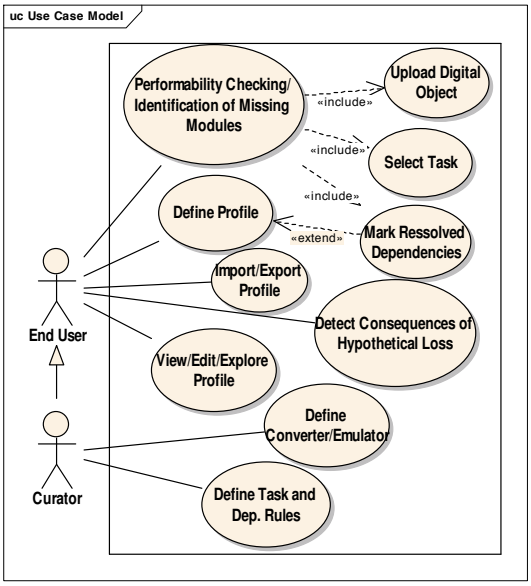

Figure 2: Use Case Diagram of Epimenides

3.2 User Interface

The user interface contains a menu divided in three sections as shown in Figure 3. The first section contains the option "Upload Digital Object" which is the core functionality of Epimenides. The "Manage Profile" section contains options for adding/deleting modules to/from a profile. Finally, the "Manage System" section contains options for curators that allow them to define Tasks, Emulators and Converters.

3.3 Performability Checking

To perform a task we have to perform other subtasks and to fulfil the associated requirements for carrying out these tasks. Epimenides is able to decide whether a task can be performed by examining all the necessary subtasks, exploiting also the possibilities offered by the availability of converters and emulators. In our example of Figure 1, the availability of a converter from Pascal to C++, a compiler of C++ over Windows OS and an emulator of Windows OS over Android OS, allows the inference that the particular Pascal file is runnable over Android OS.

[7]http://www.ics.forth.gr/isl/epimenides/

Figure 3: Main functionality of Epimenides

The core service of Epimenides, performability checking, is illustrated in the screenshots of Figure 4. After logging in to Epimenides, *the user can upload a digital object* (file or zipped files) and select the task whose performability he or she wants to check. The system checks the dependencies and computes the corresponding gap. To identify the dependencies of the uploaded objects, the system exploits the extension of the object (e.g. .pdf, .doc, .docx). An alternative way to identify file types that could be supported by Epimenides is to use *file format identification* tools like those that mentioned in Section 2. The KB of Epimenides contains the dependencies of some widely used file types. The identified dependencies are then shown to the user. The user can *add* those that (s)he already has, and this is the method for defining his/her profile *gradually*. In this way the user does not have to define a profile in one shot. The system stores the profiles of each user (those modules marked as "I have them") to the RDF triplestore.

3.4 Architecture and Current Deployment of Epimenides

The server side of Epimenides is implemented in Java and it uses the Apache Tomcat[8] 7.0.3 web server. The used triple store is the OpenLink Virtuoso[9] 06.01.3127 version, and the Virtuoso Jena RDF Data Provider[10] is used for the communication with the triplestore. Figure 5 shows the component and deployment diagram of Epimenides. The architecture of Epimenides is based on the MVC (Model View Controller) pattern, meaning that all business logic is implemented in Java Servlets and all communication and data transfer issues are addressed with the use of Java Beans. The presentation of data is specified using JSP pages in order to separate the presentation design from the application logic, making easier the extension and modification of the system.

[8]http://tomcat.apache.org/
[9]http://virtuoso.openlinksw.com/
[10]http://www.openlinksw.com/dataspace/doc/dav/wiki/Main/VirtJenaProvider

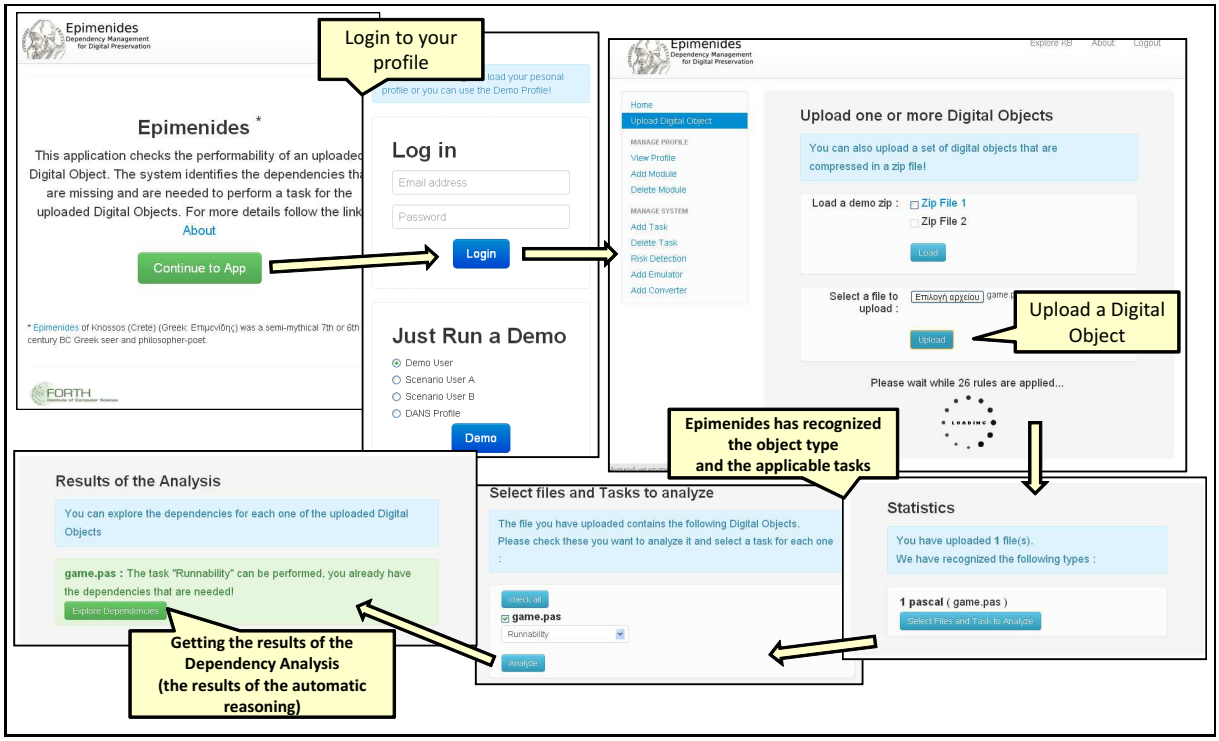

Figure 4: Checking the performability of a digital object

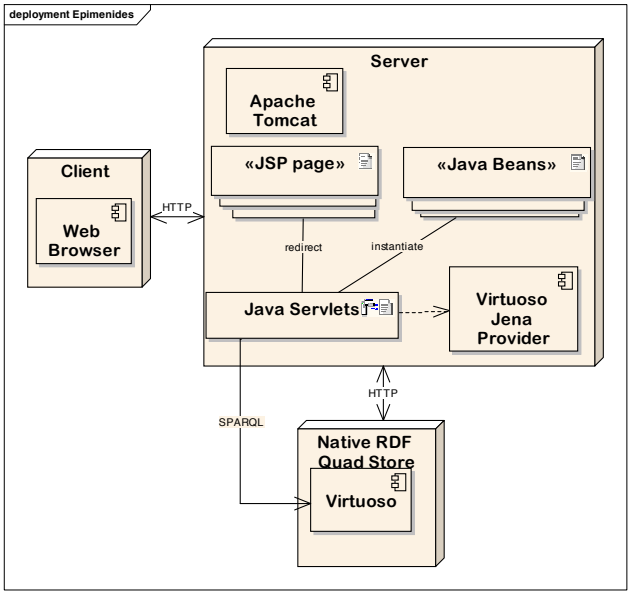

Figure 5: The deployment diagram of `Epimenides`

More information about the architecture of the Knowledge Base is given in [6].

4. EPIMENIDES USED BY A RESEARCH DATA ARCHIVE

We have conducted a case study in which the reasoning service of `Epimenides` is applied in the research data archive of DANS (Data Archiving and Networked Services, NL)[11].

[11]http://www.dans.knaw.nl/en

DANS aims at promoting sustained access to digital research data. For this purpose, it encourages researchers to archive and reuse data in a sustained manner, e.g. through the online archiving system EASY[12]. DANS also provides access, via NARCIS[13], to scientific datasets, e-publications and other research information in the Netherlands. Apart from these, the institute provides training and advice, and performs research into sustained access to digital information.

Table 1 describes some of the common practices that are followed by curators of DANS in order to archive a file in the digital repository.

4.1 Scenarios

In collaboration with DANS, we have defined a number of scenarios that indicate where and how automatic reasoning related to long-term access to digital objects could be used. The analysis yielded five scenarios, whose description follows. In brief, the desired (for DANS) tasks are mainly related to the notion of *acceptable/preferred formats*, and with the runability of DANS software (including computability of checksums).

For each scenario there is a short *description* and an *applicability* subsection that discusses how the dependency management approach can be applied and how it can be realized by `Epimenides`.

4.1.1 *Scenario 1: Supporting the notion of Preferred/ Acceptable Formats for Ingestion*

[12]http://easy.dans.knaw.nl
[13]http://www.narcis.nl

Table 1: Common practices that DANS follows in order to archive a file

Type of Data:	Common Practices
Documents	All documents (and also presentations - Powerpoint) are converted to PDF/A. For the conversion Adobe PDF convertor of Adobe Acrobat Professional is used.
Images/Illustrations	Both JPEG and TIFF (archival format) are used. Managing software: Adobe Photoshop.
Windows Metaformat (WMF) & Encapsulated Postscript (EPS)	Are converted by Adobe Illustrator to SVG (Scalable Vector Graphics) files.
Databases	dBASE (.dbf), Access (.mdb) and MS Excel Openoffice Calc are converted to CSV format. The export function of MS Access is used for the conversion. Some specific rules are applied (decimal delimiter, memo fields, double quotes in text fields). DBase (.dbf) files are imported in MS Access and exported in comma-separated values (.csv) files. Excel (.xls) files are exported to tab-delimited text files, then imported in MS Access and subsequently exported to comma-separated values (.csv) files.
Geographical Information files	Images such as Mapinfo Workspaces are converted to PDF/A. MapInfo TAB files are converted to MID and MIF files. ArcGIS Shapefiles are converted to MIF/MID by the Data Interoperability Extension of ArcGIS. Grid-fles are converted to ASCII-text files. MIG files are converted by the MAPINFO MIG-Toolbox. Surfer .grd and .srf files are converted by Golden Software Surfer to GS ASCII. Georeferenced images are converted by ArcMAP to a standard bitmap; this file is converted by Adobe Photoshop to JPEG and TIFF.
Computer Aided Design	AutoCAD files are stored as AutoCAD R12/LT2 DXF.

Description: For a number of data types (tables, text, images, etc.), specific file formats are considered to be durable at least into the near future. DANS maintains a list of *acceptable* and *preferred* formats. These lists are the basis for data archiving activities. The list that DANS currently uses is shown in Figure 6.

Applicability: If the converters (or emulators) that are in use by DANS for carrying out the migration activities, are registered in a system like `Epimenides`, then the system can be exploited not only for checking whether a newly ingested file is in an acceptable/preferred format, but also for checking whether it is migratable to one preferred or acceptable format using the migration/emulation software that DANS uses and has registered.

To realize this scenario, one has to define a profile (say $profile\_DANS$) that consists of:

i. The list containing the software that DANS uses for managing a file having an acceptable/preferred file format (e.g. `AcrobatReader` for rending PDF files, `VLC player` for playing mpg/mpeg/mp4/avi/mov files). At least one software tool per format is required.

ii. For each file type in the list of acceptable/preferred list, a task has to be associated (the one usually applicable to such file types) and the dependencies for that task have to be delivered in a way so that they are satisfied by the list of software described in (i). (e.g. for a pdf file type we can identify the *Rendering* task, and the need of (a) a pdf file, (b) an `AcrobatReader`).

iii. The list of tools that DANS uses for migration/conversion purposes (e.g. the tool `doc2pdf` for converting doc files to pdf).

Type of data	Preferred format(s)	Acceptable format(s)
Text documents	• PDF/A (.pdf)	• OpenDocument Text (.odt) • MS Word (.doc, .docx) • Rich Text File (.rtf) • PDF (.pdf)
Plain text	• Unicode TXT (.txt, ...)	• Non-Unicode TXT (.txt, ...)
Spreadsheets	• PDF/A (.pdf) • Comma Separated Values (.csv)	• OpenDocument Spreadsheet (.ods) • MS Excel (.xls, .xlsx)
Databases	• ANSI SQL (.sql, ...) • Comma Separated Values (.csv)	• MS Access (.mdb, .accdb) • dBase III or IV (.dbf)
Statistical data	• SPSS Portable (.por) • SAS transport (.sas) • STATA (.dta)	• R (*)
Pictures (raster)	• JPEG (.jpg, .jpeg) • TIFF (.tif, .tiff)	
Pictures (vector)	• PDF/A (.pdf) • Scalable Vector Graphics (.svg)	• Adobe Illustrator (.ai) • PostScript (.eps) • PDF (.pdf)
Video	• MPEG-2 (.mpg, .mpeg, ...) • MPEG-4 H264 (.mp4) • Lossless AVI (.avi) • QuickTime (.mov)	
Audio	• WAVE (.wav)	• MP3 AAC (.mp3) (**)
Computer Aided Design	• AutoCAD DXF version R12 (.dxf)	• AutoCAD other versions (.dwg, .dxf)
Geographical Information	• MapInfo Interchange Format (.mif/.mid)	• ESRI Shapefiles (.shp and accompanying files) • MapInfo (.tab and accompanying files) • Geographic Markup Language (.gml)

(*) under investigation
(**) please contact DANS before depositing MP3 audio files

Figure 6: DANS: Preferred and acceptable formats

Having completed these steps, the end user (or archivist) could use `Epimenides`. Whenever he uploads a file, `Epimenides` prompts the applicable task and directly informs the user if it is in an acceptable format or migratable to an

acceptable format using the software that DANS has.

Without such facility it is difficult for a curator to (a) determine that an archived dataset is formatted in a durable format and (b) to have an overview of the applicable file format migration procedures that can be carried out to convert a file into a preferred file format (given that the list of preferred file formats will change over time as file formats become obsolete).

4.1.2 Scenario 2: Managing the set Preferred/ Acceptable File Formats

Description: As the usability and durability of file formats tend to change over time, for DANS it is important to periodically monitor and assess the applicability of the list of preferred formats and if it is necessary to replace a file format that became obsolete with a new one. Also new preferred formats can be introduced in the list. Specifically, say every year, the specifications on the list of preferred file formats have to be assessed based on a number of criteria (e.g. discussions in literature, consensus of organizations that provide guidelines in this field, etc.).

Applicability:

i. To add a new format in the list of acceptable/preferred file formats, the archivist can register it to the Knowledge Base of `Epimenides`. The check performed at ingestion will then function as expected (i.e. in accordance with the revised list of acceptable formats).

ii. Before deleting a file format (or managing software) from the list of acceptable/preferred file formats (or available software respectively), the archivist can check the impact of that deletion, i.e. the impact that this deletion will have on the performability of tasks over the archived files. This service (risk detection) is described in detail in [12].

iii. To remove a file format (or managing software) from the list of acceptable/preferred file formats (or available software respectively), the archivist can delete the corresponding entries from the system. After doing so, the checking at ingestion (Scenario 1) will function as expected, i.e. in accordance with the revised list of acceptable formats.

Without such services it is difficult to identify all the consequences of file format's obsolescence. It is also difficult to identify what will happen if managing software that is able to convert to/from a preferred file format, is lost or will become obsolete.

4.1.3 Scenario 3: Migration

Description: Research datasets are submitted in a number of formats to the data archive by the depositors. The data archive stores and manages these datasets in the format as submitted by executing the so-called "bit-preservation" (more about bit preservation in a next scenario). The data archive manages all formats but only commits itself to the long-term usability of files that are formatted according to the so-called preferred formats, described in the previous scenarios. In two situations a file format migration is required: (a) as part of the ingest procedure, files not formatted according to the preferred file format are migrated to a suitable preferred file format, (b) in case in the future a preferred file format becomes obsolete the files have to be migrated to this new format. The migration process requires using certain tools. Quality features of these tools are: speed, accuracy, level of completeness, and usability of the tool.

Applicability: The dependency management approach can show the archivist whether a file format migration is possible using the software that DANS has (recall Scenario 1). Also since a migration can be performed with different tools (or execution plans in general), the proposed system can assist the archivist by showing him/her, the possible actions/tools and this can be achieved by exploring the dependencies that the system offers.

Without this approach it is difficult for a human to identify all possible migration plans.

4.1.4 Scenario 4: Software Preservation

Description: Despite the fact that research data archives aim for durable access of datasets, there are cases where specific software is required to be able to use the datasets. For such cases, activities have to be undertaken to guarantee that this software is usable over time. Software preservation involves much more dependencies than research data preservation (e.g. changing operating systems, proprietary source code, etc.). Research data archives currently have no general accepted software preservation strategy.

Applicability: The example described in section 1 (Figure 1) falls in this scenario. Also [12] demonstrated this scenario with various examples.

4.1.5 Scenario 5: Authenticity of digital objects

Description: The bit preservation scenario involves activities to guarantee that digital objects do not become corrupted. This means that not one bit is changed over time. Thus the integrity of the data objects is guaranteed. This can be achieved by creating checksums on the occasion where the digital objects are ingested in the data archive and periodically checking whether a checksum is still valid. Dependencies in the scenario are the strength of the checksum procedures and the time interval the checksum is checked as part of the bit preservation activities.

Applicability: If checksums are supposed to ensure that the data have not been corrupted, an archive can model as task the computation of checksums for being sure that in the future the archiving organization will be able to recompute and compare them with the stored ones. Note that there are several tools for computing checksums[14]. We can say that this is a special case of Scenario 4.

4.2 Consolidation of the Scenarios

[14] `http://en.wikipedia.org/wiki/Checksum#Checksum_tools.`

Table 2: Application of the Methodology for the case of DANS

General Step	Specialization for the case of DANS
1. Identify the desired tasks and objectives	The desired tasks are: a. those related to the list of the acceptable/preferred formats, e.g. `render` (for pdf, txt, pictures), `play` (for video, aurio), `getTheRelationalModel` (for spreadsheets, databases), etc. b. those related to the runability of DANS software (including computability of checksums).
2. Model the identified tasks and their dependencies (check hierarchy)	Model the tasks using the list of software described in Scenario 1 (i). (Section 4.1.1). Moreover the dependencies of the runability of the tools that DANS uses for migration have to be modeled. Model the software dependencies that are required for running the software that DANS uses. In general the required modeling is quite simple, analogous to the examples given in [12].
3. Specialize the rule-based approach	It seems that there is not need for any particular specialization.
4. Identify Ways to capture dependencies (manual, auto, semi-automatic)	The file types are detected automatically (when one uses the upload feature of `Epimenides`). For applying this approach in big collections of files, various tools could be used for automating this process. Surely, in an operational setting the proposed functionality could extend or complement the functionality of the ingestion procedures of the systems that DANS currently uses.
5. Customize, use and exploit the dependency management services	For demonstration purposes this can be done using `Epimenides`, i.e. no need for customization or integration with the other systems of DANS. However, in an operational setting the processes and systems of DANS (EASY, NARCIS) should be considered.
6. Evaluate	This can be done using `Epimenides`.

Table 2 consolidates the key points of the above scenarios describing them based on the steps of a general methodology introduced in [12], for modeling, capturing and managing dependencies for the needs of digital preservation.

4.3 Defining the Profile of DANS in Epimenides

Following the implementation requirements of the scenarios that were described in Section 4.1, we defined a profile for the case of DANS. Specifically:

- We have registered (using `Epimenides`) the managing software that DANS uses in order to manage the preferred/acceptable files.
- We have identified and registered to the KB of `Epimenides` the tasks that make sense to apply in the list of the preferred/acceptable files.
- Finally the migration tools of DANS have also been registered to the DANS profile.

This profile is available in the registry of `Epimenides` and can be used by the archivists of DANS to exploit the benefits of the automatic reasoning approach that are described in the above scenarios. It defines 21 converters, 11 managing software tools, 4 tasks, and 44 rules. The representation of the profile as RDF triples is around 2,405 RDF triples. The numbers are summarized also in Table 3.

Considering the practices shown in Table 1, note that `Epimenides` with the DANS profile behaves as expected. For example, the practices of DANS for excel database files as described in Table 1 are: *"Excel (.xls) files are exported to tab-delimited text files, then imported in MS Access and subsequently exported to comma-separated values (.csv) files"*. Now suppose that we want to check if DANS could manage a database excel file, say `mydb.xls`. Two conversions should be applied according to the practice that is described before (.xls $\xrightarrow{MSExcel}$ .tab $\xrightarrow{MSAccess}$ .csv). Having defined (as shown in Figure 7a) in `Epimenides` that DANS holds the needed converters (`MS Excel` and `MS Access`) and uploading the `mydb.xls` to the system we can see in Figure 7b that the proposed automated reasoning has been applied and the appropriate tasks can be performed for this file.

Table 3: DANS profile & Numbers

Component:	#
Converters	21
Managing Software	11
Tasks	4
Defined Rules	44
Triples in Repository	2,405

5. CONCLUDING REMARKS

Digital material has to be preserved not only against loss or corruption, but also against changes in its ecosystem. In this paper we described `Epimenides`, a system that realizes an automatic reasoning approach for assisting this digital preservation problem. The approach is based on the description of *dependencies* that are required in order to achieve a *task*. `Epimenides` can be used by digital archives and digital libraries to help archivists in checking whether the archived digital artifacts remain *intelligible* and *functional*, and in identifying the consequences of probable losses.

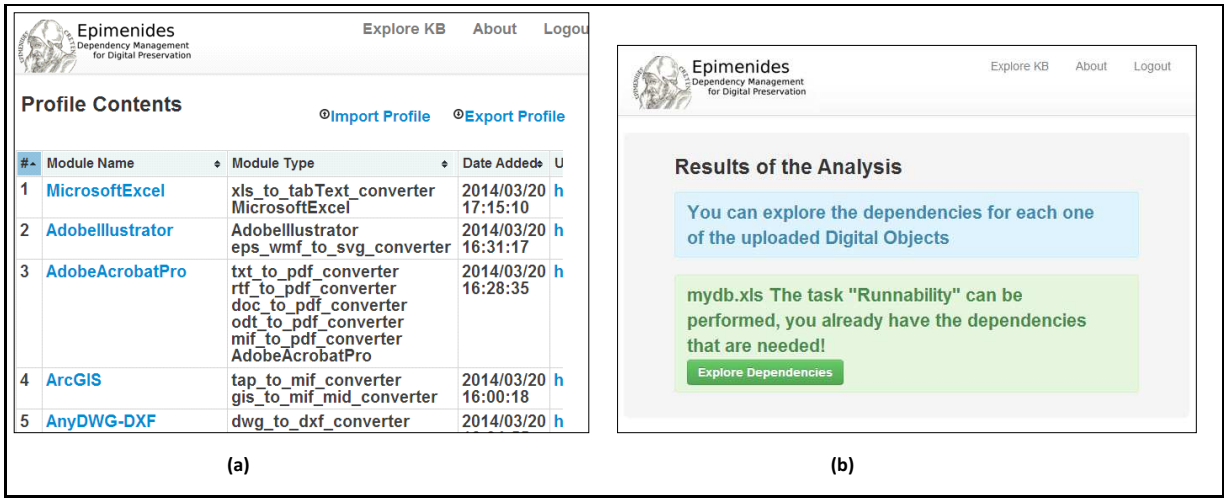

Figure 7: a)Contents of DANS profile as shown in `Epimenides` b)Checking the performability of an excel file in DANS profile

In this paper we described (in the form of scenarios) how the reasoning service of `Epimenides` can be applied in the DANS data archive. We showed how various real activities are actually dependency management activities. Finally for the realization of the scenarios, we defined in `Epimenides` a profile for DANS.

From the technical side, an objective for future research is to develop quality-aware reasoning for enabling quality-aware preservation planning.

Acknowledgements

Work done in the context of NoE APARSEN (Alliance Permanent Access to the Records of Science in Europe, FP7, Proj. No 269977).

6. REFERENCES

[1] David Anderson, Janet Delve, L Konstantelos, A Ciuffreda, and M Dobreva. Totem: Trusted online technical environment metadata: a long-term solution for a relational database/rdf ontologies. 2011.

[2] José Barateiro, Daniel Draws, Martin Alexander Neuman, and Stephan Strodl. Digital preservation challenges on software life cycle. In *Software Maintenance and Reengineering (CSMR), 2012 16th European Conference on*, pages 487–490. IEEE, 2012.

[3] Christoph Becker, Hannes Kulovits, Andreas Rauber, and Hans Hofman. Plato: a service oriented decision support system for preservation planning. In *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*, pages 367–370. ACM, 2008.

[4] Esther Conway, Matthew Dunckley, Brian McIlwrath, and David Giaretta. Preservation network models: Creating stable networks of information to ensure the long term use of scientific data. *Proc. PV2009, Madrid, Spain*, pages 1–3, 2009.

[5] Alliance for Permanent Access to the Records of Science Network (APARSEN). "D25.1 Interoperability Objectives and Approaches", 2013. (http://www.alliancepermanentaccess.org/wp-content/uploads/downloads/2013/03/APARSEN-REP-D25_1-01-1_7.pdf).

[6] Alliance for Permanent Access to the Records of Science Network (APARSEN). "D25.2 Interoperability Strategies", 2013. (http://www.alliancepermanentaccess.org/wp-content/uploads/2013/10/APARSEN-REP-D25_2-01-1_7.pdf).

[7] Y. Marketakis, M. Tzanakis, and Y. Tzitzikas. PreScan: Towards Automating the Preservation of Digital Objects. In *Procs of the International Conference on Management of Emergent Digital Ecosystems MEDES'2009*, Lyon, France, October, 2009.

[8] Y. Marketakis and Y. Tzitzikas. Dependency Management for Digital Preservation using Semantic Web technologies. *International Journal on Digital Libraries*, 10(4), 2009.

[9] Y. Tzitzikas and B. Bazzanella. Interoperability Objectives and Approaches:Results from the APARSEN NoE . In *Proceedings of the 10th Annual International Conference on Digital Preservation (iPres2013)*, 2013.

[10] Y. Tzitzikas and G. Flouris. "Mind the (Intelligibily) Gap". In *Procs of the 11th European Conference on Research and Advanced Technology for Digital Libraries, ECDL'07*, Budapest, Hungary, September 2007. Springer-Verlag.

[11] Y. Tzitzikas, Y. Marketakis, and G. Antoniou. Task-based Dependency Management for the Preservation of Digital Objects using Rules. In *Procs of 6th Hellenic Conf. on Artificial Intelligence, SETN-2010*, Athens, Greece, 2010.

[12] Y. Tzitzikas, Y. Marketakis, and Y. Kargakis. Conversion and Emulation-aware Dependency Reasoning for Curation Services . In *Proceedings of the 9th Annual International Conference on Digital Preservation (iPres2012)*, 2012.