Yannis Marketakis

PreScan (Preservation Scanner)

slide 1 The Preservation Scanner, which is presented in this speech, is a tool that automates the extraction of metadata of digital files.

slide 2 After having described the background, the architecture of the PreScan and its components are shown, followed by some measurements regarding the time performance of PreScan, a comparison with other similar systems, and finally a description of some future extensions of the system.

slide 3 Behind this tool there is the need to create and maintain metadata, which is a laborious and expensive task, and the consequent need to automate this process as much as possible.
PreScan is a tool for automating the extraction and maintenance metadata.
In the FORTH approach, some metadata is extracted automatically and then updated periodically; users being allowed to handle metadata and define dependencies between files.

slide 4 Metadata can be stored internally or externally: in the first case metadata is called *embedded* and is stored in the digital file; in the second case metadata is called *detached* and is stored in a separate file.
These approaches have both advantages and disadvantages: embedded metadata is tightly coupled with files and therefore it is transferred with files; however, there will obviously be redundancy between files. On the other hand, detached metadata can be stored in special repositories and linked to digital files; this approach does reduce redundancy, but the way in which metadata is linked to data should be treated with particular attention since inconsistency may arise from it.

slide 5 Four components constitute PreScan: the Scanner, the Metadata Extractor, the Repository Manager and the Controller.

slide 6 The Scanner is responsible for scanning the file system. The user can choose the folders that should be scanned and where the metadata should be stored.
The Scanner also allows the re-scanning of a project that is aware of file movements, file addition and human provided metadata.

slide 7 The Extractor extracts the embedded metadata of the scanned file. PreScan can choose an external metadata extractor for this purpose; it currently uses JHOVE

which supports various kinds of metadata, as shown in the table.

slide 8 The Controller is responsible for the entire process. It organizes the other components; performs the file system scan; when the scan is required, it tries to identify new files that did not exist at the time of the previous scan, files that have been moved to another location and those that are unnamed. The identification of these files is essential so that human provided metadata will be preserved and linked with the current files.
The user has also the ability to confirm a file movement or reject it. This process also permits only new or modified files to be scanned, instead of scanning the whole folder again.

slide 9 The Repository Manager is responsible for storing and managing the metadata records of digital files.
Metadata, both automatically extracted and human provided, is organized in metadata records which are created for every file that is scanned.
There is more than one choice regarding where these metadata records may be stored: the first option is to organize and store metadata records in a specific directory which is chosen by the user, which makes it easier to find the metadata record of a specific file; the second option is to store the metadata record in the same folder with its original file; the third option is to store the metadata record in a Semantic Web-based Knowledge Base which allows the querying and updating of the metadata of a file.
These three options are not exclusive.

slide 10 If one wants to use a Semantic Web-based Knowledge Base, metadata must be recorded in a hardier format and in order to do that, an ontology could be used, a good choice for this ontology being the CIDOC-CRM with its extensions.
CIDOC-CRM is a core ontology used by libraries, archives and museums; one of its extensions being the CIDOC-CRM Digital Ontology which is appropriate for capturing digital objects and their provenance.
Underneath this there is a Core Ontology for Dependencies (COD) which is used to express dependencies between digital objects.
Any other domain specific specializations can be used under this architecture of ontologies.
The instance layer contains the automatically extracted and the manually provided metadata or dependencies.

slide 11 The diagram shows the complete set of ontologies which are used: the boxes represent classes and the arrows represent subclasses of relations; the yellow boxes are the CIDOC-CRM and CIDOC-CRM Digital classes, the red and green

boxes are the COD and COD typologies classes.

slide 12 This is a diagram of the RDF output of PreScan.
Every metadata record would be a digital object and information regarding its size, its MD5 checksum, its last modification date and last scan date, would be instantiations of the ontologies presented above.

slide 13 Analysing the time performance of the PreScan for various data sets, it is possible to see that the most time-demanding process is the extraction of metadata which takes about 10 seconds to scan a folder containing 10 files and amounts to approximately 10 hours for a folder containing 100 thousands files of various types.
It is possible to see also the time it takes to calculate the MD5 checksum of the files and the time to store the metadata records of the files.

slide 14 PreScan is quite similar in spirit to the crawlers of Web Search Engines. It allows one to scan the file system, extract the embedded metadata and build an index; but it also supports more advanced extraction services, the manual addition of metadata, more expressive representation frameworks to store and to exploit the metadata (that is possible by using for example Semantic Web languages), does periodic scans which do not scan a folder from scratch but exploit the previous scan in association with external resources (e.g. registries).
PreScan can aid automating the ingestion process for file system-based archives.

slide 15 There are several other related works: Empirical Walker is probably the most similar tool to PreScan since it scans files, determines file formats, analyzes the contents of files and associates external metadata.
Another similar tool is Catalogue-File metadata miner which also extracts metadata and exports it in RDF, identifies several file formats but does not recognise its specific version.
PreScan offers more functionalities than other tools, i.e. the PreScan with preservation of manual metadata, the identification of file movements, the export to RDF ability, the exploitation of format registries and the possibility to realize the compliance with dependency management.

slide 16 Some of the future extensions of the PreScan will be to use more metadata extractors in order to recognize more formats, and the generation of RDF metadata according to the CIDOC-CRM Digital in a more flexible way.

slide 17 The releases of PreScan available are: the Alpha version released in June 2009; this has some known bugs that were correct in the Beta version which was released in September 2009 and supports the generation of instances of CIDOC-

CRM Digital which allow the browsing of metadata through the GUI of GapManager.

slide 18 The developers and the people responsible for the PreScan project are Yannis Tzitzikas, Yannis Marketakis and Makis Tzanakis.
On one of the web pages of PreScan http://www.ics.forth.gr/PreScan it is possible to find the releases of the tool and some useful documents as well as examples concerning its usage.

demo Supposing one wants to create a new project, the first step is to add the name of the project and the location of the file, after which one defines where the metadata records should be stored either in a specific directory or in the folder with the original file, and then finally selects the folder that needs to be scanned.
During the scanning of the folder, information concerning the scanning process is visualized in a window; at the end of it, a quantity of information about the project may be looked at, e.g. the folder which has been scanned, the location of the metadata records, the amount of the total scanned files and the last scanning date.
Moreover, it is possible to see the metadata records which are in XML format.
In the case of a file modification, one can add some metadata, find the source file for this metadata record, rename it, move it to another location, add new files, rescan the project and after look at the new preservation file that the scan has found. One can also control that the old file no longer exists, individuate the new one and control the metadata of the project.
By using the Exporter, the metadata of a file can be exported in the RDF format; moreover, this is exported as CIDOC-CRM instances and one can choose to export all the metadata in a single file or in multiple files, choosing the destination folder in either case.