

A Repository for 3D Model Production and Interpretation in Culture and Beyond

M. Doerr¹, K. Tzompanaki¹, M. Theodoridou¹, Ch. Georgis¹, A. Axaridou¹, S. Havemann²

¹Institute of Computer Science, FORTH-ICS, Crete, Greece

²Institute of Computer Graphics and Knowledge Visualization, Graz University of Technology, Austria

Abstract

In order to support the work of researchers in the production, processing and interpretation of complex digital objects and the dissemination of valuable and diverse information to a broad spectrum of audience there is need for an integrated high performance environment that will combine knowledge base features with content management and information retrieval (IR) technologies.

In this paper we describe the design and implementation of an integrated repository to ingest, store, manipulate, and export 3D Models, their related digital objects and metadata and to enable efficient access, use, reuse and preservation of the information, ensuring referential and semantic integrity. The repository design is based on an integrated coherent conceptual schema that models complex metadata regarding provenance information, structured models, formats, compatibility of 3D models, historical events and real world objects. This repository is not implemented just to be a storage location for digital objects; it is meant to be a working integrated platform for distant users who participate in a process chain consisting of several steps. A first prototype, in the field of Cultural Heritage, has already been implemented in the context of 3D-COFORM project, an integrated research project funded by the European Community's Seventh Framework Programme (FP7/2007-2013, no 231809) and the results are satisfactory, proving the feasibility of the design decisions which are absolutely new, ambitious, and extraordinarily generic for e-science.

Categories and Subject Descriptors (according to ACM CCS) : I.3.3 [Computer Graphics]: Picture/Image Generation— Digitizing and scanning, I.3.4 [Computer Graphics]: Graphic Utilities—Meta files, I.3.8 [Computer Graphics]: Applications—

1. Introduction

The advances in digital technology provide scientists with powerful tools to digitize, document and manage artifacts of scientific interest. However, the challenge still lies in the integration of the available technologies in an homogeneous, high performance working environment to support the researchers' workflow and the dissemination of information. Such an environment must combine knowledge base features with content management and information retrieval (IR) technologies.

Scientists need to create and use 2D and 3D digital representations of artifacts, physical objects or geographical places and to link these representations to other relative information. Often the processes that lead to the construction of a 3D model consist of several steps that are performed by

different people, possibly in distant locations. Handled digital objects can be complex in nature, demand large storage resources, high transfer ratio and carry significant, highly interrelated, semantic and provenance metadata information. Hence, the efficacious exchange of the metadata information on the processing workflow as well as the efficient large scale data exchange and data management are of great importance.

In certain scientific production workflows, objects are created and ingested in the repository gradually rather than "imported" at the end of the workflow. The implicit assumption in existing working repositories is that the object to be stored has undergone all possible processing steps and reached its final stage. Repositories that store only the final product are well supported by models such as OAIS [OAI02] and ORE

[ORE08]. However, this is not the case in a research workflow that is not predefined, and which cannot be described as a production pipeline, but rather as a trial and error procedure with several back and forth steps, which may be performed by different people, in distant places. In this type of workflow it is desirable to store all the steps of a given process chain, i.e. possible experimental data, intermediate data, used software and procedures information, parameters of interactive processes etc. The final product is only a part of a given process and might be the result of multiple reorganization and reaggregation of information. The repository can thus be viewed as a directed graph of temporary (intermediate) products and acts as the integration point for all the tools used by the researchers.

Following the above mentioned requirements, we have designed an integrated repository to ingest, store, manipulate and export complex digital objects, their components and related metadata and enable efficient access to the information. The preserved metadata can be used to reconstruct intermediate files that have been deleted either manually or by a "garbage collection" mechanism that is applied periodically. This repository provides an integrated management of:

1. any type of digital objects, including digital representations of artifacts,
2. related metadata regarding model structure and parameters and relations of 3D models to modeled objects and their coordinates,
3. digital object provenance information, i.e. employed sources, processes, tools and parameters,
4. descriptions and semantic classification of the modeled objects, their parts and employed analogies, their location, their history and other associations, sources and expert annotations about modeling and related historical data.

A well defined Application Programming Interface (API) to the repository functionality allows external tools to access, enhance and/or use the knowledge and the information contained in the repository.

The repository design was based on the following features:

- a) **Scalability** of the system in order to handle the growing amount of distributed data. The management of distributed large volume information needs to be associated and harmonically blended with the management of centralized metadata information.
- b) **Safety** of the expensive and unique data require replication of information.
- c) Support of **long term preservation** which addresses the challenges of final result definition out of trial and intermediate data, of obsolescence and inaccessibility posed to digital artifacts by technological (rapidly changing S/W, format obsolescence), organizational, legal, and marketplace influences, and by the loss of means for interpreting,

understanding and representing those artifacts (data about lost objects, excavation layers etc).

- d) **Confidentiality** support, which requires reliable user and access management and controlled replication.
- e) **Sustainability** of the repository data, which is achieved by persistent versioning. Persistent versioning means that any change to the binary of a content file is associated with a new ID and the change is described in persistent metadata information.
- f) Support of **provenance metadata**. The processing of data in "atomic" per processing step-principle of "historical order" guarantees exception-free processing. By enforcing constraints, the repository ensures referential integrity between processing step metadata. At the same time, relaxed constraints allow for legacy data import. By summarizing metadata from all precursor steps it is possible to produce DIP/AIP exports (Open Archives Initiative)[ORE08]. Metadata export is necessary for communication with other repositories such as Europeana [Eur] and can be achieved through the use of persistent URIs and Linked Open Data [LOD] compatibility.
- g) Support of an **annotation mechanism** that provides referential integrity of links into content segments of any kind and dimension (areas), with no modifications on the original object. The annotation areas should be preserved across processing, at any time in the life-cycle.
- h) Support of a **co-reference mechanism** that acts as a "mending" mechanism of the semantic network. It should detect and correct co-reference at any time (e.g. identifiers of acquired objects) and manage authority, personal knowledge, truth and inconsistency.

In order to fulfill the aforementioned requirements, we have surveyed existing content and media management repositories. Some widely used systems are MediaHub [MED], Alfresco [ALF] and Fedora [FED]. MediaHub is an integrated system mostly used by enterprises to manage and facilitate team activities by sharing notes, interactions and results within the tool, whereas Alfresco is mostly an open source document and web-content management solution. Fedora is a repository for long-term preservation of digital assets, but lacks high performance for large storage volumes and does not support handling of very complex metadata. Several existing state-of-the-art 3D cultural heritage repositories are described in [KFH09]. However, there is no system in our knowledge, that provides an integrated high performance working environment to support the management of complex digital objects together with their significant, highly interrelated, semantic and provenance metadata information.

The paper is structured in 3 further sections. Section 2 describes the design and the architecture of the Repository Infrastructure (RI) along with the implementation details. Section 3 introduces the metadata model developed. Finally, section 4 contains concluding remarks.

2. Repository Infrastructure

The Repository Infrastructure (RI) is designed to allow different kinds of users (researchers, academics, technicians, museum users, web users, etc) to store, query and retrieve digital objects which have been produced by digitizing physical objects or by processing other digital objects. These actions are initiated by external tools that interact with the RI through a well-defined API. Such an integrated system deployed in the field of Cultural Heritage, the 3D-COFORM infrastructure, is being developed in the context of 3D-COFORM project [3DC], an integrated research project funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 231809.

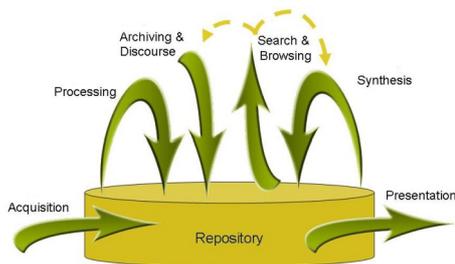


Figure 1: A typical workflow. All external tools, no matter in which part of the workflow are involved (e.g. acquisition, processing, synthesis, presentation, etc.) generate and maintain semantic and provenance metadata and interact with the RI. At all steps users may access, use and refer to information in the repository. The repository provides persistency and rich access services for the highly interlinked content which includes 3D artifacts, multi-media data, metadata, and for storing and maintaining semantic relationships. The repository also manages links to, and data exchange of contents and metadata with other existing repositories and collection management systems by open interfaces and formats.

The integrated repository system plays a central role in the digital object production workflow. Figure 1 presents the typical workflow and shows how the RI is used in order to form the base of a wide range of user actions. The repository is the main integration actor of most of the digital outcomes which the users derive and/or store as either data producers, data consumers or both.

2.1. Repository Infrastructure Design

The repository is the place where all digital artifacts are physically stored supporting all textual, 2D and 3D artifact representations (e-science technical outcomes). It provides a range of services: ingestion, retrieval, deletion, update, grouping, replication, querying as well as mark-up and indexing.

During the RI design specific issues were taken into consideration:

- a) The object repository follows the distributed Data Warehouse logic, since interinstitutional (i.e. distant and individual) processing is a requirement. It provides controlled replication, partial location transparency and URI abstraction.
- b) The metadata repository is a centralized semantic network that manages integrated metadata information. The metadata source files are stored, for safety reasons, as objects in the object repository along with the content objects. Thus we can recreate the metadata repository information from these metadata sources at any time.
- c) There are general constraints to be taken into account for the design of the repository. These constraints include compliance to standards, interoperability requirements, data and metadata repository and distribution requirements, security requirements and network communications.

2.2. Repository Infrastructure Architecture

The RI is composed of several modules which connect and interact with one another as illustrated in Figure 2. The most important modules are the Object Repository (OR), consisting of a number of distributed OR-nodes and a Central Replica Location Service (CRLS), and the central Metadata Repository (MR). The OR-nodes contain all the content data-files, while the CRLS is responsible for holding information about the storage location of data-files and metadata-files and their replicas. The MR contains the related semantic metadata information. The OR and the MR are assisted by the Content Retrieval Indices module (CRI) containing pairs of indices about material/shape and object-identifiers, that provides the mechanism to search the RI for digital objects, using material/shape-based queries.

These modules are coordinated by the RI-API central service. This central service forwards the requests from the clients that need to access the repository to the RI modules e.g. the MR, the OR-nodes etc. Data transfer requests to and from the OR-nodes, such as ingest and retrieve, are initiated by the central service RI-API, whereas they are executed and controlled by the specific OR-node. Query requests concerning data, metadata and material/shape information are redirected to the respective RI modules (OR-nodes, CRLS, MR, CRI) by the Query Manager (QM). These modules are presented in the following sections.

2.3. Object Repository

The Object Repository (OR) consists of the distributed OR-nodes and the Central Replica Location Service (CRLS).

The OR-nodes contain different classes of binary datasets: raw data, normalized data, intermediate data of processing

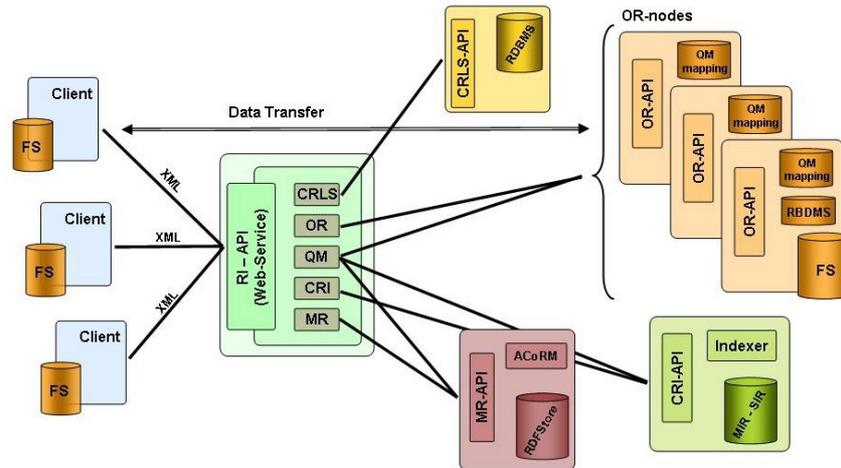


Figure 2: Overview of the Repository Infrastructure architecture. The clients communicate with the repository through a central WEB-Service, via XML messages that follow the SOAP standard. This central service is passing the requests to the RI components (e.g. the Metadata Repository, the OR-nodes etc). Data transfer is a direct file transfer between clients and OR-nodes.

steps, scenes composed of several datasets, presentation objects, 3D models, 2D-Images and any kind of digital document (text, multimedia). The OR understands the notion of complex objects, having datasets as parts and can support any data type. Each OR-node (Figure 2) consists of three modules: the distributed file system (FS) that is the physical storage location for the data-files, the relational database that holds information regarding the stored files e.g. their hierarchical dependencies, date of creation, etc. and the QM mapping module that holds the relational database schema in RDF format.

The CRLS is a central directory management system, that records the storage location of all OR files and their replicas.

2.4. Metadata Repository

The Metadata Repository (MR) is a semantic network that aims to provide a common place to ingest, store, manipulate and export metadata concerning objects stored in the OR, other related objects or information relative to them. Metadata are recorded in metadata files, physically stored in the OR together with their related content files, while in the MR a semantic network is built with the integrated metadata information. In this way, data integrity is satisfied and it is guaranteed that a failure in the MR will be resolved by re-ingesting metadata files retrieved from the OR.

The MR is based on an integrated, coherent conceptual schema that models the complex metadata comprising a range of different concepts such as provenance metadata, object descriptions, annotations and co-reference information, structured models, formats, compatibility of 3D models, his-

torical events, and real world objects (also see section 3). The MR provides a variety of actions to the users like ingestion, updating, querying and deduction.

The MR contains as an integral part, the Annotation and Co-reference Manager (ACoRM) which handles the links between 3D models, their metadata and all related resources and knowledge. ACoRM supports duplicate detection to find matches between references to places and people, semi automatic detection of references to the same thing and conflict resolution consistency enforcement algorithms (*is the same as/not the same as*).

2.5. Content Retrieval Indices

The Content Retrieval Indices module (CRI) contains pairs of indices about material (Material Index Repository - MIR), shapes (Shape Index Repository - SIR) and object identifiers.

Material search: The purpose of the material search is to enable the user to search and browse the RI for objects, using material-based queries (e.g. a sample of a Bidirectional Texture Function - BTF). The search is performed based on material descriptors.

Shape-Model search: The purpose of the shape-based search is to enable the user to search in the RI for the 3D objects with similar shapes to the one selected by the user as an example. The search is performed using the query service based on shape descriptors that could include global and local descriptors.

These materials/shapes relevance descriptors are pre-calculated for the objects stored in the OR by an indexing-service (Indexer) that will be run periodically and will store

the computed relevance descriptors (for newly ingested OR objects) in CRI repositories (MIR and SIR).

2.6. Query Manager

One of the most important aspects in e-science is to be able to search and combine information regarding objects, persons, places, events, procedures etc. Of great usefulness is also the ability to find and compare objects relatively to specific materials/shapes. This functionality is supported by the Query Manager (QM) which is responsible to analyze the query and redirect the specific parts of the query to the respective modules (OR, MR, CRI).

The user formulates a query by providing a URI, word, or term and the query may consist of two parts: the SPARQL and the CRI parameters.

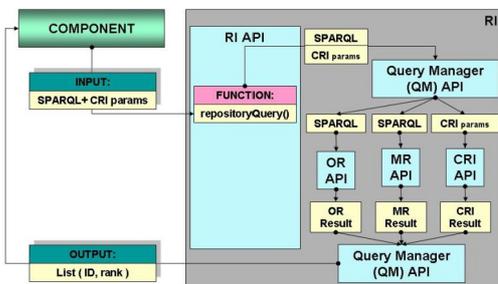


Figure 3: Query Manager mechanism

The QM (Figure 3) receives the SPARQL query statement, divides it into two statements, one SPARQL for the OR and one for the MR, forwards the two statements to the respective repositories and receives the results from the two repositories.

In order to be able to query the relational database of the OR-node by SPARQL we use a mapping from the relational database schema to RDF. In this way we achieve a unified, efficient and easy to handle querying mechanism to both the relational and semantic databases.

Additionally, the QM forwards the CRI parameters to the CRI component and receives the results. Due to the periodic indexing, the result of a material/shape based search may not be complete since it will not contain results that were ingested after the last indexing process.

The overall query result is a unified list of the three result sets and is presented as a list of object UUIDs, or a list of object UUIDs and values for the other queried parameters, or a ranked list (according to the provided CRI parameters) of object UUIDs, depending on the query.

2.7. Implementation Issues

In the context of the 3D-COFORM project, the Repository Infrastructure (3D-COFORM RI) is being developed based on the following decisions:

- The integrated repository platform is implemented using existing Open Source technologies.
- The Object Repository FS is implemented on Andrew File System (AFS) [MSC⁺ 86, Ope], which is a distributed file system featuring user management, access control lists, Kerberos-5-based user- and service-authentication, efficient caching mechanism to reduce bandwidth usage and optionally encrypted network traffic. The relational database implementation is based on MySQL [MyS] and the D2R server [D2R] is selected for the mapping of the relational database schema to RDF in order for the QM to direct access the relational database with SPARQL query statements.
- The Metadata model is based on CIDOC-CRM schema version 5.0 (ISO21127) [CDG⁺ 09], and CRM_{dig} version 2.3 [TTD⁺ 10] which includes the extensions regarding metadata about digital objects and processes. The technology used to write the metadata files is Resource Description Framework (RDF) [KC04].
- The Metadata Repository is implemented on Sesame [SES]. Sesame is an open source Java framework for storing, querying and reasoning with RDF and RDF Schema. It can be used as a database for RDF and RDF Schema, or as a Java library for applications that need to work with RDF internally.
- The first prototype of the RI-API was implemented as Java API while the final interface is being implemented as a Web-service [BHM⁺ 04].

3. Metadata

Cultural and scientific data cannot be understood without knowledge about the meaning of the data and the ways and circumstances of their creation. For this reason it is essential to have metadata created for physical objects as well as for digital objects that bear cultural or scientific interest. These metadata are actually data describing objects, people, places, times which are causally related by events while other relations are either deductions from events or found by observation events. Consequently, metadata are event centric and must be described in a historical order in order to ensure that there are no references to non-existent (non-recorded) events or objects. Generally, we use metadata to assess meaning (view, experimental setup, instrument settings), relevance (depicted objects, their status, their conditions), quality (calibration, tolerances, errors, artifacts) and possibilities of improvement and reprocessing.

Creating and saving metadata is critical for processes such as the 3D modeling process for which it is essential to maintain the processing chain that led from the physical object to

its 3D model, as well as to ensure the correlation between the objects involved in the processing chain and the information kept in each stage (provenance metadata). In this way the provenance information is secured and can be retrieved any time, in order to re-perform a processing step. Except for metadata concerning objects, metadata may also exist in the form of annotations on 3D models or other digital objects or areas on them and can depict modeling assumptions (evidence, analogous continuation, cultural parallels), source references, model comparisons (model-model, model-source), pros and cons and related events.

Facing the need to record processing steps individually but in a common way and then connect them to one another, we came across the lack of an existing metadata standard format. The already existing solutions are function oriented, partial, overspecialized and without understanding of generalizations. The solution lays on using a common core ontology to explain the meaning of various data structures and support the complex metadata requirements of the e-science domain. The selected CIDOC-CRM core conceptual schema provides a common and extensible semantic framework that any procedural information can be mapped to. Instances of the CIDOC-CRM model can be merged to huge meaningful networks of knowledge about historical facts and contextual relationships [DI08]. The CIDOC-CRM model is intended to be a common language for domain experts and implementers to formulate requirements for information systems and to serve as a guide for good practice of conceptual modeling. In this way, it can provide the "semantic glue" needed to mediate between different sources of information, such as that published by museums, libraries and archives.

Subsequently we introduced *CRM_{dig}*, an extension of CIDOC-CRM ontology, able to capture the modeling and the query requirements regarding the provenance of digital objects, as described in the next subsection. In a similar way, the model can be extended to describe other e-science procedures.

Metadata are generated following a specific schema and structure, manually or automatically. We chose to use RDF as the metadata encoding language, because RDF offers several assets such as independence, interchange and scalability. To ensure the integrity and correctness of the metadata files and also the consistency of the produced semantic web, we have produced an RDF schema that checks and validates the files according to specific rules [CID, CRM].

3.1. Modelling acquisition, processing and annotation events

Digitizing moveable and immovable objects and processing digital objects in order to produce further representations are common and very important procedures which need to be well and under certain rules documented.

For this reason, we assume a form of more or less con-

straint workflow. We assume that there is a finite set of physical objects to be digitized with an acquisition process which will produce digital outputs (e.g. images or point clouds) for each object, in one or more steps. The digital objects can then be further used by appropriate 3D model production processes in order to produce their 3D representations.

Additionally, the produced digital objects can serve as anchors for annotation links. Annotations are associations between digital objects, areas in them and/or persons, places, events, etc. In that way they are considered to be another form of metadata and can be recorded under a standard structure.

Depending on the kind of the association, we distinguish the following types:

- **Comment** is used to link a comment with an object.
- **Relationship** is used to relate objects with each other or with persons, places, events etc.
- **Analogy** is used to describe the analogical relationship on a specific common feature of two or more objects
- **Same-As** is used to declare that two or more objects are exactly the same
- **Reference** is used to create a reference from one object to another.
- **Classification** is a special case of reference that is used to classify an object.

Our modeling approach is event centric and follows an hierarchical workflow structure. The target is to avoid redundancy of information so we suggest holding the common information as high as possible in a hierarchy of nested activities. For example it is preferable to store the information about the acquired object in the highest level of an Object Acquisition Event so that it can be inherited by the sub-events, than repeating this information for each sub-event separately. So, the highest level includes some constants in the process setup while the lowest level includes the information about the machine event that produced the digital object.

Metadata can be structured by answering the following four main questions about:

- **WHO:** the persons or organizations playing role in the event
- **WHERE:** the place the event occurred
- **WHEN:** the time the event occurred
- **WHAT:** the things involved in the event

Since we support multiple instantiation, an action may simultaneously be of multiple types and thus a new processing class can be introduced specializing the more generic ones. Each processing step is described with its own metadata, and is interconnected with its *historically previous* step.

In the following a simplified example is presented in order to describe our modeling approach. The physical object with inventory number Bo036 is digitized using the Multiview

Dome [Dom]. The digital object produced (File A) is further processed by a different person to produce a 3D model (File B). A third user annotates the 3D model by relating it to the creator of the depicted physical object. The metadata are produced independently in each step (acquisition, processing, annotation) and are ingested in the repository separately and in a historical order. Classes and properties of the CRM are identified by codes such as "E55" and "P12". Codes such as "D2" and "L12" represent the classes and properties of *CRM_{dig}*. All values are identified by UUIDs or URIs to avoid mistyping and to ensure common reference to the same thing.

Dome Acquisition Event

D2.Digitization Process:("Acquisition uuid")

- P2F.has type: E55.Type("http://..DOME photography")
- L30F.has operator: E21.Person("http://..John Papas")
- L31F.has starting date-time: 2009-08-12T09:38:00Z
- L32F.has ending date-time: 2009-08-12T10:15:00Z
- P7F.took place at: D23.Room("http://..Multimedia Lab")
- L1F.digitized: E18.Physical Thing("http://..Bo036")
- L12F.happened on device: D8.Digital Device("http://..Dome")
- L11F.had output: D1.Digital Object("File A uuid")
- ...

3D Mesh-Modeling Event

D3.Formal Derivation:("Processing uuid")

- P2F.has type: E55.Type("http://..3D-Mesh Modeling")
- L30F.has operator: E21.Person("http://..Thomas Miller")
- L31F.has starting date-time: 2009-09-12T08:30:00Z
- L32F.has ending date-time: 2009-09-12T09:15:00Z
- P7F.took place at: D23.Room("http://..3D Lab")
- L2F.used as source: D1.Digital Object("http://..mm3d.exe")
- L21F.used as derivation source: D1.Digital Object("File A uuid")
- L22F.created derivative: D1.Digital Object("File B uuid")
- ...

Annotation Event

D30.Annotation Event:("Annotating uuid")

- P2F.has type: E55.Type("http://..Annotation")
- L30F.has operator: E21.Person("http://..Petra Friger")
- L31F.has starting date-time: 2009-10-12T08:30:00Z
- L32F.has ending date-time: 2009-10-12T09:15:00Z
- L48F.created annotation:
 - D32.Relationship ("Annotation uuid")
 - L44F.relates: D1.DigitalObject("File B uuid")
 - L45F.relates to: E39.Actor("http://..Nollekens")
- ...

3.2. How to make use of the created semantic web

It is of high importance to record and store semantic information concerning scientific procedures in an integrated semantic network under a common schema which allows for useful inferences on indirect or derived relationships, frequently following deep data paths in the network. If the metadata ingested in the repository are consistent and the relationships are well described, chains or data paths are created that correlate events, objects, places, times, people. Queries to the repository can traverse these paths in order to find distant relations relevant for understanding and managing the content.

For example, the physical object represented in a given 3D model may not be described in the model's metadata, but may exist in the metadata describing the acquisition process. Under the assumption, that the subsequent processing events are declared as "subject preserving", which means that the physical object depicted in the derivatives remains the same as the one in the derivation source, it is possible to infer the subject of the model by traversing the following path:

3D model -> L21B.was_derivation_source_for -> source files used to create the 3D model -> L11B.was_output_of -> events that created the source files -> P9B.forms_part_of -> object acquisition event -> L1F.digitized -> physical object

Another issue that can be addressed by queries on the metadata of the processing chain, is tracing device flaws, e.g. when observing that the same pixels are destroyed in several digital objects that may have been produced by different processes, using the same device. Also, the user may find where an annotation made on a digital object should appear on derivative digital objects that maintain the same subject, by inferring the propagation of anchor links depending on the classification and parameters of the derivation processes. It is even possible, to infer annotation anchors from a derived object to the original one.

4. Conclusions and Further Work

In this paper we have described the implementation of an integrated repository, that has achieved to:

- a) implement a repository for *digital objects* and metadata files emerging from *scientific processes* of measurement, processing or simulation.
- b) implement a semantic repository to store all individual metadata information about each measurement or processing step and about related things and contexts in a searchable format and to connect all metadata into one coherent *semantic network*.
- c) integrate the whole repository to serve as a *working environment*, temporary and permanent storage medium for different and *distant* users, in order to *collaborate* in *integrated* workflows and other scientific activities, such as annotation based discourses.

- d) establish a sustainable repository of 3D digital models and a 3D interchange format suited for *long term preservation*
- e) to enable the *management* of temporary and final *products*, garbage collection and generation of *export packages* to other repositories or clients for distribution, dissemination and archival of results.
- f) provide an integrated query mechanism on data and metadata (semantic and provenance information) which enables useful and complex inferences to support context management, comprising even diverse content retrieval mechanisms for 3D objects.

The idea of the integrated Repository Infrastructure described here is innovative, ambitious, and extraordinarily generic for e-science. It has currently successfully passed the first prototype state and elementary testing. Enough further work remains to be done:

The Metadata Repository search should be enhanced by reasoning and rules concerning deductions from the metadata. For the time being we are considering as reasoning platform OWLIM [OWL] which is a high-performance semantic repository developed in Java and is packaged as a Storage and Inference Layer (SAIL) for the Sesame RDF framework. Adding reasoning to our semantic database will make queries more effective and simple. Further, the enforcement of constraints and other rules on the metadata content at ingest and update time will ensure the consistency and correctness of the information described in the semantic web. In this way, rich provenance information can be sustained, enabling many useful and complex inferences to support the content management.

For the Object Repository further work must be done for the off-line operational mode. The current implementation assumes that all the system modules are on-line. However, for field data acquisition campaigns it is useful if an OR-node can go off-line. In this case the OR must be able to provide elementary services, like ingest. This ability will also provide more robustness under unreliable networks. The Metadata Repository synchronization (automatic or manual) will then be performed as a deferred action when the OR node goes on-line again, providing error handling, recovery or log reporting following the triggered action.

5. Acknowledgements

We gratefully acknowledge the generous support from the European Commission for the Integrated Project 3D-COFORM (grant no. 231809).

We would also like to thank our partners from TU Graz, P. Beckmann and X. Pan, for the fruitful collaboration in the implementation of the presented system.

References

- [3DC] 3D-COFORM: Tools and expertise for 3D collection formation. <http://www.3d-coform.eu>. 3

- [ALF] Alfresco. <http://www.alfresco.com>. 2
- [BHM⁺04] BOOTH D., HAAS H., MCCABE F., NEWCOMER E., CHAMPION M., FERRIS C., ORCHARD D.: *Web Services Architecture*. Tech. rep., World Wide Web Consortium, February 2004. 5
- [CDG^{*}09] CROFTS N., DOERR M., GILL T., STEAD S., STIFF M.: *Definition of the CIDOC Conceptual Reference Model*. Tech. rep., Mar 2009. http://www.cidoc-crm.org/docs/cidoc_crm_version_5.0.1_Mar09.pdf. 5
- [CID] CIDOC CRM v5.0.1 Encoded in RDFS. http://www.cidoc-crm.org/rdfs/cidoc_crm_v5.0.1.rdfs. 6
- [CRM] CRMdig 2.3 Encoded in RDFS. <http://www.cidoc-crm.org/rdfs/CRMdig2.3.rdfs>. 6
- [D2R] D2R Server: Publishing Relational Databases on the Semantic Web. <http://www4.wiwiw.fu-berlin.de/bizer/d2r-server>. 5
- [DI08] DOERR M., IORIZZO D.: The dream of a global knowledge network—a new approach. *ACM J. Comput. Cult. Herit.* 1, 1 (2008), 1–23. 6
- [Dom] Multiview Dome. <http://3dcoform.org/index.php/tools/multiview-dome>. 7
- [Eur] Europeana. <http://europeana.eu/portal/>. 2
- [FED] Fedora Repository Project: general-purpose, open-source digital object repository system. <http://www.fedora-commons.org>. 2
- [KC04] KLYNE G., CARROLL J. J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. *World Wide Web Consortium, Recommendation REC-rdf-concepts-20040210* (Feb. 2004). 5
- [KFH09] KOLLER D., FRISCHER B., HUMPHREYS G.: Research challenges for digital archives of 3D cultural heritage models. *ACM J. Comput. Cult. Herit.* 2, 3, Article 7 (December 2009), 17 pages. 2
- [LOD] Linked Open Data. <http://esw.w3.org/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>. 2
- [MED] MediaHub. <http://www.marketwire.com/mw/include.do?module=MEDIA&pageid=661>. 2
- [MSC^{*}86] MORRIS J. H., SATYANARAYANAN M., CONNER M. H., HOWARD J. H., ROSENTHAL D. S., SMITH F. D.: Andrew: a distributed personal computing environment. *Commun. ACM* 29, 3 (1986), 184–201. 5
- [MyS] MySQL: Relational Database Management System. <http://www.mysql.com>. 5
- [OAI02] Reference Model for an Open Archival Information System (OAIS). <http://public.ccsds.org/publications/archive/650x0b1.pdf>. 1
- [Ope] OpenAFS: open source implementation of the Andrew distributed file system (AFS). <http://www.openafs.org>. 5
- [ORE08] Open Archives Initiative - Object Reuse and Exchange, Specifications and User Guides, Oct. 2008. <http://www.openarchives.org/ore/1.0/toc.html>. 2
- [OWL] OWLIM. <http://www.ontotext.com/owlim>. 8
- [SES] Sesame: RDF Schema Querying and Storage. <http://www.openrdf.org>. 5
- [TTD^{*}10] THEODORIDOU M., TZITZIKAS Y., DOERR M., MARKETAKIS Y., MELESSANAKIS V.: Modeling and Querying Provenance by Extending CIDOC CRM. *Distributed and Parallel Databases* (2010). 5