An ontology-based model for preservation workflows

Michalis Mikelakis
Dept. of Informatics
Athens University of Economic and Business
Athens, Greece
mikelakism@aueb.gr

Christos Papatheodorou
Dept. of Archives and Library Science
Ionian University
Corfu, Greece
papatheodor@ionio.gr

ABSTRACT
This paper aims to propose an ontology for the main digital preservation workflows carried out by an organization or an archival system. The proposed ontology covers the entire preservation life cycle, starting from the ingestion of digital resources and including internal functions, such as system administration and preservation planning policies, and access control. Fifty workflow models have been represented using the ontology, which takes into account the special characteristics and features specified by the international standards, as well as the existing metadata schemas for preservation. The proposed ontology supports the decision making of the collection managers, who design preservation policies and follow practices, by providing a knowledge-based tool able to guide, encode and (re)use their reasoning and choices.

Categories and Subject Descriptors
H.3.7 [**Information Storage and Retrieval**]: Digital Libraries – *standards, systems issues*.

General Terms
Design, Documentation, Standardization.

Keywords
Digital Preservation Workflows, Ontology, OAIS Model.

1. INTRODUCTION
Digital preservation has attracted the interest of the scientific community during the last decade since it addresses crucial issues for the future of digital data and information stored in large repositories or published on the World Wide Web. The production of digital data nowadays has grown rapidly and it concerns all aspects of human activity, such as health, science, culture, public functions and political decisions. At the same time, the fast changes in technology have shortened the lifespan of digital objects, which, in contrast to analog ones, have no meaning outside the technical environment that they have been designed for. The danger of information loss is even greater for digitally born objects, where the original information cannot be retrieved from any other source in case of media failure, format or tool obsolescence or loss of metadata.

The systems that have been implemented in the area of digital preservation focus mainly on particular preservation activities such as planning, migration or emulation and follow workflows inspired by OAIS model [6]. Some of them integrate a set of tools trying to provide a preservation framework and support organizations to develop policies and workflows for preserving their own material [3, 8, 9, 12]. However these systems do not offer a model that expresses explicitly and analytically the workflows they perform in order to (i) guide the user throughout the preservation process and (ii) be potentially reused by other implementations.

This paper proposes an ontology that provides a new conceptualization of the OAIS preservation workflows describing the concepts associated with the structure and form of a digital object as well as the complex relationships involved in the preservation process. The choice of creating an ontology was grounded on the expressive power of such knowledge organization and representation schemes. Moreover, the use of an ontology facilitates information reuse. It could easily be used in its entirety by an organization interested in representing information for digital preservation workflows, or integrated with other internal ontologies of the organization. Furthermore, it can be extended by defining new concepts and relationships or even redefining existing ones in order to fit to one's specific needs. The proposed ontology was developed using OWL, a language for authoring ontologies, which has been endorsed by the World Wide Web Consortium (W3C). The use of a language that has been established as a standard agrees with the concept of long-term preservation and ensures that the model will not become obsolete in the future. Thus the paper exploits semantic web tools to contribute to the systematic aggregation and formal expression of the preservation workflows. Hence the preservation workflows for particular collections and digital objects are represented as instances of a conceptual model and formulate a semantic network. These instances can be retrieved (using SPARQL queries), re-used and interlinked to each other or with other metadata concerning the collections and digital objects.

The next section describes the current standards and tools related to workflow management and used by well known initiatives aiming at the development of digital preservation tools. Section 3 presents the proposed model providing a description of the classes and properties of the developed ontology. Section 4 presents how the ontology is used to represent preservation workflows and provides a detailed example concerning the implementation of a specific preservation workflow model. Section 5 describes the user guidance throughout the preservation process with the utilization of the model and the representation of user interactions with the archival system. In the last section we conclude with summarizing the present work and providing directions for future expansion.

2. BACKGROUND
A workflow is defined as the computerized facilitation or automation of a business process, in whole or part [5]. A workflow is a model of an activity, which is consisted of a set of operations or steps. It defines various objects participating in the

flow of the process, such as documents, roles, information exchanged and tools needed for the completion of each step. Every step is generally described by some basic features, which are input information, output information and transformations made by a person or a machine playing a specific role [4].

Workflow management is a technology that has demonstrated a very large expansion and has been adopted in various industries. Organizations develop and use workflow management systems, which are designed according to their internal processes or adjusted to fit their specific needs. A Workflow Management System is defined as "a system that completely defines, manages and executes workflows through the execution of software whose order of execution is driven by a computer representation of the workflow logic" [5].

The vast spread in the development of workflow management products has lead to the need for a common framework, which will define the basic aspects of a workflow management system and provide standards for the development of systems by different vendors. The Workflow Management Coalition (WfMC[1]) is a consortium, comprised of adopters, developers, consultants, analysts, as well as university and research groups, whose purpose is to identify common characteristics among workflow management systems and to define standards for the interoperability of such systems. The WfMC has developed the Workflow Reference Model, in order to define a workflow system and to identify the most important interfaces for the interaction between such systems. Under the scope of the Workflow Reference Model, XML Process Definition Language (XPDL) [13] was defined, which is a format to interchange definitions of business process workflows between different workflow products, including both their structure and semantics. XPDL defines an XML schema for specifying the declarative part of a business process. XPDL is not an executable programming language, but a process design format that visually represents a process definition. Another standard created under the WfMC is Wf-XML, which provides web service operations to invoke and monitor a process that might need a long time to complete, so as to facilitate the communication between a process editing tool and a process execution tool, which may be provided by a different vendor.

The mentioned standards focus mainly on providing a representation of a business process. On the other hand, there are executable languages for representing processes. Business Process Execution Language (BPEL) [7] is one such language, which specifies actions within business processes. BPEL uses an XML-based language and provides the capability of interconnecting with outside systems. Processes in BPEL export and import information by using web service interfaces exclusively. BPEL does not provide a strict protocol and there are no explicit abstractions for people, roles, work items, or inboxes. Instead it is a process-centric model that focuses on the interactions and message exchanges that take place in a process.

Another popular business process management tool is jBPM[2]. jBPM is a flexible Business Process Management Suite which models the business goals by describing the steps that need to be executed to achieve a goal and the order of the steps. It uses a flow chart, where a process is composed of tasks that are connected with sequence flows. There are a lot of other implementations based on the above models, such as Apache OFBiz Workflow Engine[3], Apache Agila[4], Open Business Engine[5], wfmOpen[6] and ActiveBPEL[7].

A suite of tools created for building and executing workflows is Taverna[8], a domain-independent workflow management system that uses its own definition language. It provides a graphical designer enabling the addition and deletion of workflow components. Taverna does not provide any data services itself, but it provides access and integration of third party services. The SCAPE project[9], a recently European founded project on preservation, has chosen Taverna as the tool for representing workflows. Preservation processes are realized as data pipelines and described formally as automated, quality-assured preservation Taverna workflows.

The SCAPE working group continues the efforts of the PLANETS project[10], also co-funded by the European Union, which addresses digital preservation challenges. The project's goal was to build practical services and tools to ensure long-term access to the digital cultural and scientific assets. In general the project provides a detailed implementation of the preservation functions of an OAIS compliant digital repository. The Planets Functional Model is broken down into three Sub Functions: Preservation Watch, Preservation Planning and Preservation Action [10]. These Sub Functions have been mapped to the functions of the OAIS Reference Model. Especially the Planets Preservation Planning Sub Function is based on the OAIS model to describe the functions and processes of a preservation planning component of a digital repository [11, 12].

The project specifies its own workflow description language and execution engine. A preservation workflow consists of a sequence of invocations of services, where the output parameters of one service are mapped to the input parameters of the next one. Furthermore, the Planets Workflow Execution Engine (WEE) introduces the concept of workflow templates, which are predefined workflow definitions. The user interacts with a set of Web Service interfaces through which he can browse the available templates and choose to instantiate and execute those that meet his specific needs [1].

The proposed approach is designed to cover exclusively and with completeness the needs for representing and manipulating preservation workflows. Therefore it should use a language able to express consistently the semantics of the OAIS Reference Model. An additional requirement would be the subsumption of the information for preservation workflows under the linked data framework. For this purpose OWL was opted for the description of the proposed model.

[1] http://www.wfmc.org/

[2] http://www.jbpm.org/

[3] http://incubator.apache.org/ofbiz/

[4] http://wiki.apache.org/agila/

[5] http://obe.sourceforge.net/

[6] http://wfmopen.sourceforge.net/

[7] http://www.activebpel.org/

[8] http://www.taverna.org.uk/

[9] http://www.scape-project.eu/

[10] http://www.planets-project.eu/

3. THE PROPOSED MODEL

As mentioned the design of the model was mainly based on the specifications of the OAIS Reference Model. The entities and the messages exchanged among the different functions specified in the OAIS model were combined into logical sequential steps which constitute the basic workflows. In addition, these workflows were enriched with information provided outside of the OAIS model, especially operations defined within the scope of the Planets project [11] [2, 9]. These operations focus on specific functions of the preservation process, such as preservation planning, and provide more details refining the steps of the process.

For the design of the ontology, we used Protégé[12] (version 4.1.) an open-source ontology engineering tool, developed at Stanford University. Protégé has been widely used for ontology development, due to its scalability and extensibility with a large number of plug-ins. The classes and properties of the proposed ontology are described in the next sections, while the whole model is presented in Figure 1.

3.1 Preservation Workflows

The OAIS Reference Model has been established as a fundamental design reference model for an archival system and has been widely adopted as a basis in digital preservation efforts in many areas, such as digital libraries, commercial organizations and government institutions. The OAIS model defines the basic entities and functions required by an organization responsible for the preservation of digital information and its availability to a Designated Community and it provides a minimal set of responsibilities for an archive to be called an OAIS. It consists of six main entities, which are Ingest, Archival Storage, Data Management, Administration, Preservation Planning and Access. Each entity plays a specific role in the preservation process.

The OAIS model also defines specific roles which describe the way that external users interact with an archival system and the way that internal users can manage the broader policy of a system. These roles are referred to as Producer, Consumer and Management. Every user can take specific actions according to the available interfaces. A Producer is the person or system which provides the data products to be preserved. An object submitted to the system must have specific characteristics and meet some minimum requirements in order to be accepted. OAIS makes an extensive description concerning the ways for representing information and the structure of a digital object, as well as the forms that it can take inside and outside the scope of an archival system. Before an information package is accepted, the archival system should make sure that it has the required control and rights to ensure the long-term preservation of the information.

Thus the preservation of a digital object is a complex procedure, which follows specific policies and a general strategy defined by the archive management in agreement with the users. It consists of several steps, each of them operated by a number of internal functions of the archival system. Several functions should cooperate sequentially or in parallel via the exchange of objects for a complete preservation process.

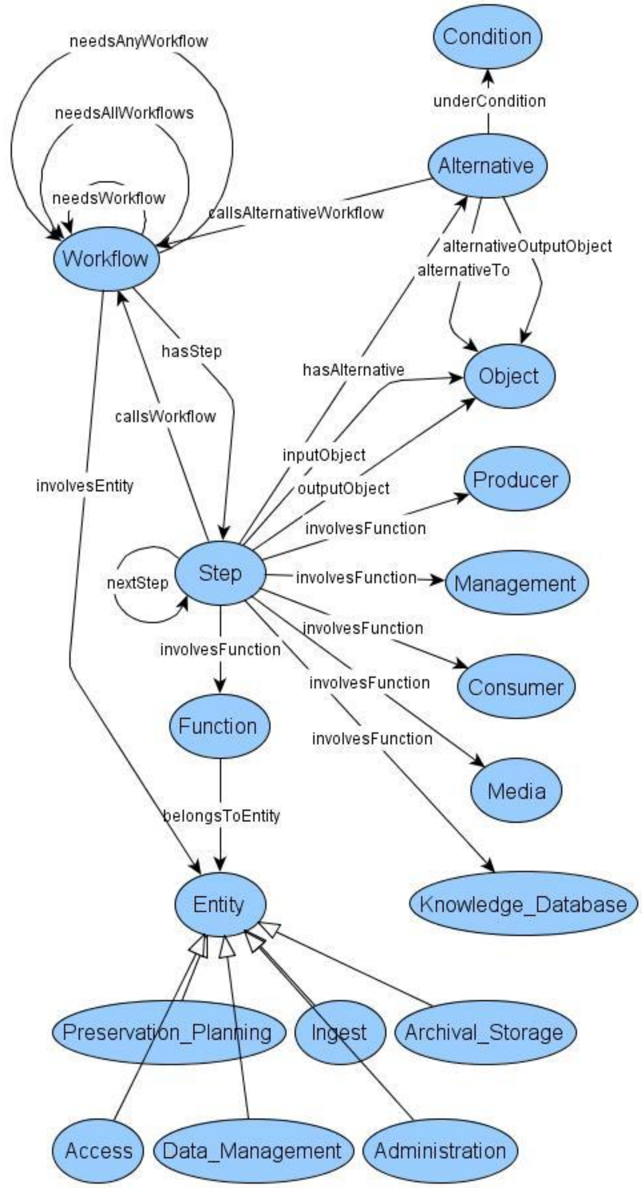

Figure 1. The proposed ontology

An important aspect of an archival system is the way it makes the preserved information available to external users, also referred to as the Designated Community. It should provide a Consumer with search functionalities on metadata kept by the archive or even on the preserved objects themselves. This is accomplished by the iterative submission of queries and the return of query responses.

Based on the above description, some basic concepts that describe the structure of an archival system and the interactions with the users can be concluded. The workflows are divided into six groups, in accordance to the functional entity that is responsible for their execution. Specifically, the workflows are related to Ingest, Archival Storage, Data Management, Administration, Preservation Planning and Access. A workflow may be executed directly and therefore be considered as a primitive workflow, or it may have to wait for another workflow to be completed in order to be able to start. Each workflow consists of one or more steps,

[11] http://www.ifs.tuwien.ac.at/dp/plato/intro.html

[12] http://protege.stanford.edu/

which are executed consecutively or may be executed in parallel. A step has an input and/or output object and is executed by a specific function. After a step is completed it may call the next step(s), call another workflow or end the workflow. The exact classes and properties that constitute the proposed ontology are introduced in the next sections.

3.2 Classes

The classes of the ontology are defined as follows:

Entity: It encompasses the functional entities as described in the OAIS Reference Model. Hence its subclasses are: Ingest, Access, Administration, Archival_Storage, Data_Management and Preservation_Planning.

Function: The entities perform particular functions; according to the OAIS model the subclasses of this class are the following: Activate_Requests, Administer_Database, Archival_Information _Update, Audit_Submission, Co-ordinate_Access_Activities, Co-ordinate_Updates, Customer_Service, Deliver_Response, Develop_Packaging_Designs_and_Migration_Plans, Develop_ Preservation_Strategies_and_Standards, Disaster_Recovery, Error_Checking, Establish_Standards_and_Policies, Generate_ AIP, Generate_DIP, Generate_Descriptive_Info, Generate_Report, Manage_Storage_Hierarchy, Manage_System_ Configuration, Monitor_Designated_Community, Monitor _Technology, Negotiate_Submission_Agreement, Perform_ Queries, Physical_Access_Control, Provide_Data, Quality_ Assurance, Receive_Data, Receive_Database_Updates, Receive_ Submission, Replace_Media.

Role: It includes the main roles of the external entities, as described by the OAIS Reference Model; hence its subclasses are Management, Producer and Consumer.

Object: Every object that may be exchanged between two functions during a digital preservation process. Each object is represented as a subclass of the Object class. According to the OAIS model these subclasses are: AIP, AIP_request, AIP_review, Advice, Alternatives, Appeal, Approved_standards, Assistance, Assistance_request, Audit_report, Bill, Billing_information, Budget, Change_requests, Commands, Cost_estimate, Customer_comments, Customisation_advice, DIP, Data_Formatting_standards, Database_update_request, Database_Update_response, Descriptive_information, Disaster_ recovery_policies, Dissemination_request, Documentation_ standards, Duplicate_AIP, Emerging_standards, Error_logs, External_data_standards, Final_ingest_report, Inventory_report, Issues, Liens, Migration_goals, Migration_package, New_file_format_alert, Notice_of_data_transfer, Notice_of_ shipping_order, Operational_statistics, Order, Payment, Performance_information, Policies, Potential_error_notification, Preservation_requirements, Procedures, Product_technologies, Proposal, Prototype_request, Prototype_results, Quality_ assurance_results, Query_request, Query_response, Receipt_confirmation, Recommendations, Report, Report_request, Request_accepted_notification, Request_ rejected_notification, Requirements_alerts, Resubmit_request, Review-updates, Risk_analysis_report, SIP, SIP_design, SIP_review, SIP_templates, Schedule_agreement, Security_policies, Service_requirements, Status_of_Updates, Storage_confirmation, Storage_management_policies, Storage_ request, Submission_agreement, Survey, System_evolution_ policies, System_updates, Technology_alert, Template, Tools, Unanticipated_SIP_Notification.

Media: According to OAIS this class represents hardware and software settings within the archive.

Workflow: This class is defined as the set of all the preservation workflows. Each entity involves a subset of workflows; the workflows in each entity are modelled as subclasses of the class *Workflow*.

Step: Each workflow consists of a set of distinct steps. The steps of each workflow are modelled as subclasses of the class Step.

Alternative: An alternative out of the normal flow in a step, depending on a specific condition, which leads to an alternative output object and may also result in an alternative workflow being called.

Condition: A condition that must be satisfied so as for an alternative to take place. This class is the set of all the conditions that must hold before the execution of alternatives.

Knowledge_Database: The database that stores the gained experience and knowledge from preservation planning activities.

The instances of the mentioned classes correspond to particular functions, steps, workflows, etc. applied by the administrators of a digital repository for the preservation of the objects of particular collections. The ontology provides a rich vocabulary to express in detail and explicitly the actions and the dependencies between them.

3.3 Properties and their constraints

The properties of the ontology correlate its classes defining reasoning paths. The proposed object properties of the ontology are defined as follows:

involvesEntity: This property correlates a workflow to the entity involved in it. Hence the domain of this property is the class *Workflow* and its range the class *Entity*. A constraint is defined on the property imposing that every workflow must be related with exactly one entity.

hasStep: This property denotes that a workflow includes at least one step; it correlates a workflow with all the steps that are needed for the workflow to be completed. Thus the domain of the property is the class *Workflow* and its range is the class *Step*.

involvesFunction: The domain of this property is the class *Step* and its range is the union of the classes *Function*, *Consumer*, *Producer*, *Management*, *Media* and *Knowledge_Database*. Every step must be related to exactly one Function, Consumer, Management, Media, Producer or Knowledge_Database with the property in hand.

belongsToEntity: This property relates a function with the entity it belongs to; thus the domain of the property is the class *Function*, while its range is the class *Entity*. Every function must be related to at least one entity with this property.

inputObject: It defines that a step requires as input an object. Its domain is the class *Step* and its range the class *Object*.

outputObject: It relates a step with an object produced by the step as an output. Its domain is the class *Step* and its range the class *Object*.

nextStep: It correlates a step to all the steps that immediately follow after it. Thus the domain and the range of this property is the class *Step*.

callsWorkflow: It correlates a step with the workflow that is called after its completion, denoting that a workflow might follow a step of a preceding workflow. The domain of the property is the class *Step* and the range the class *Workflow*.

needsWorkflow: It correlates a workflow with the required workflows for its completion. The required workflows must be completed before the beginning of the current workflow. This property has two subproperties, the *needsAllWorkflows* and *needsAnyWorkflow*. The first subproperty means that all the required workflows must be completed before the execution of the workflow in hand and the second subproperty implies that a workflow can begin after the completion of any one of the required workflows.

hasAlternative: Its domain is the class *Step*, while its range is the class *Alternative* and denotes an alternative of a step.

alternativeOutputObject: The property identifies the output object of an alternative step of the given step. Its domain is the class *Alternative* and its range is the class *Object*.

alternativeTo: The domain of this property is the class *Alternative* and its range is the class *Object*. The property defines the output object that has been substituted by the alternative output object (defined by the previous property).

underCondition: The domain of this property is the class *Alternative* and its range is the class *Condition* and denotes that the execution of an alternative step pre-supposes the satisfaction of a condition.

callsAlternativeWorkflow: It denotes that an alternative workflow is called during a step, instead of the workflow that would normally be called. Its domain is the class *Alternative* and its range the class *Workflow*.

Table 1 concludes the ontology object properties along with their constraints. Moreover three datatype properties are introduced that attribute the names and identifiers of the ontology instances, as follows:

workflowId: It is a data type property correlating a workflow with its identifier, which is a unique string. Every workflow must have exactly one identifier.

objectName: It is a data type property correlating an object with a name, which belongs to the string datatype. Every object must have exactly one object name.

stepId: It is a data type property and denotes that every step must have exactly one identifier; thus the domain of the property is the class Step and its range the datatype string.

alternativeId: It is a data type property correlating an alternative with its identifier, which is a unique string. Every alternative must have exactly one identifier.

conditionId: It is a data type property correlating a condition with its identifier, which is a unique string. Every condition must have exactly one identifier.

Table 1. The ontology Properties

Name	Domain	Range	Constraints
alternativeOutputObject	Alternative	Object	
alternativeTo	Alternative	Object	
belongsToEntity	Function	Entity	cardinality =1
callsAlternative Workflow	Alternative	Workflow	
callsWorkflow	Step	Workflow	
hasAlternative	Step	Alternative	
hasStep	Workflow	Step	min cardinality =1
inputObject	Step	Object	
involvesEntity	Workflow	Entity	cardinality =1
involvesFunction	Step	Function or Consumer or Management or Media or Producer or Knowledge_Database	cardinality =1
needsWorkflow	Workflow	Workflow	
nextStep	Step	Step	Asymmetric, Irreflexive
outputObject	Step	Object	
underCondition	Alternative	Condition	min cardinality =1

4. IMPLEMENTING THE MODEL

The ontology represents each preservation workflow as a subclass of class *Workflow*. It involves exactly one entity that consists of a number of steps, modeled as subclasses of the class *Step*. Totally 50 workflow models have been created covering every possible internal function of an archival system or user interaction with the system incorporated in OAIS. An example that demonstrates the way the proposed ontology reveals explicitly all the characteristics of a preservation workflow, is given in regard to the Ingest entity as follows.

Figure 2 presents the Ingest entity as it is described in the OAIS Functional Model. Ingest provides the services and functions to accept Submission Information Packages (SIPs) from Producers (or from internal elements under Administration control) and prepare the contents for storage and management within the archive [6]. According to Figure 2, the Ingest entity consists of five functions, Receive Submission, Quality Assurance, Generate AIP, Generate Descriptive Info and Co-ordinate Updates. Each function performs specific tasks and exchanges a number of messages and objects. The sequence of the message and object exchanges defines the basic workflows that are specified by the proposed model.

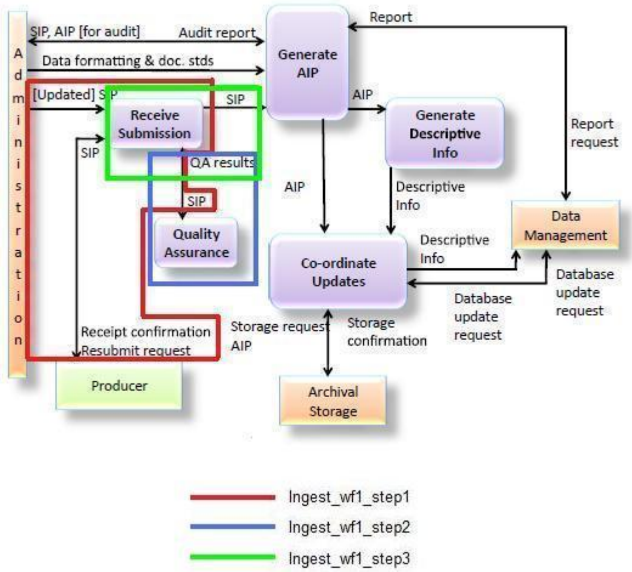

Figure 2. OAIS Ingest Functional Entity

The model decomposes the Ingest entity to four workflows. The first of them is named Ingest_wf1 and consists of three sequentially executed steps, highlighted in Figure 2 as differently coloured frames. Each frame encloses the functions and objects participating in the respective step.

The representation of the workflow Ingest_wf1 by the ontology is shown in Figure 3. The workflow needs any one of the three workflows, namely Ingest_wf4, Administration_wf6 and Administration_wf10, in order to start executing. During the first step, the Receive_Submission function receives a SIP as input from any one of the above workflows and produces a Receipt_Confirmation and a SIP as output to the second step. Alternatively, it can output a Resubmit_request and call the fourth workflow, named Ingest_wf4. During the second step, the Quality_Assurance function receives the SIP, it outputs a Quality_Assurance_results object and continues to the third step, where the Receive_Submission function receives the Quality_Assurance_results as input, outputs a SIP and calls the second workflow named Workflow Ingest_wf2.

An indicative representation of the workflow using the classes and properties of the ontology is shown below. The following fragment from Protégé editor defines that the workflow refers to the Ingest entity and consists of three steps:

```
Ingest_wf1 involvesEntity exactly 1 Ingest
Ingest_wf1 hasStep exactly 1 Ingest_wf1_step1
Ingest_wf1 hasStep exactly 1 Ingest_wf1_step2
Ingest_wf1 hasStep exactly 1 Ingest_wf1_step3
```

The definition of the three steps, encoded in OWL, is given by the following fragment:

```
<SubClassOf>
  <Class IRI="#Ingest_wf1"/>
  <ObjectExactCardinality cardinality="1">
    <ObjectProperty IRI="#involvesEntity"/>
    <Class IRI="#Ingest"/>
  </ObjectExactCardinality>
</SubClassOf>
<SubClassOf>
  <Class IRI="#Ingest_wf1"/>
```

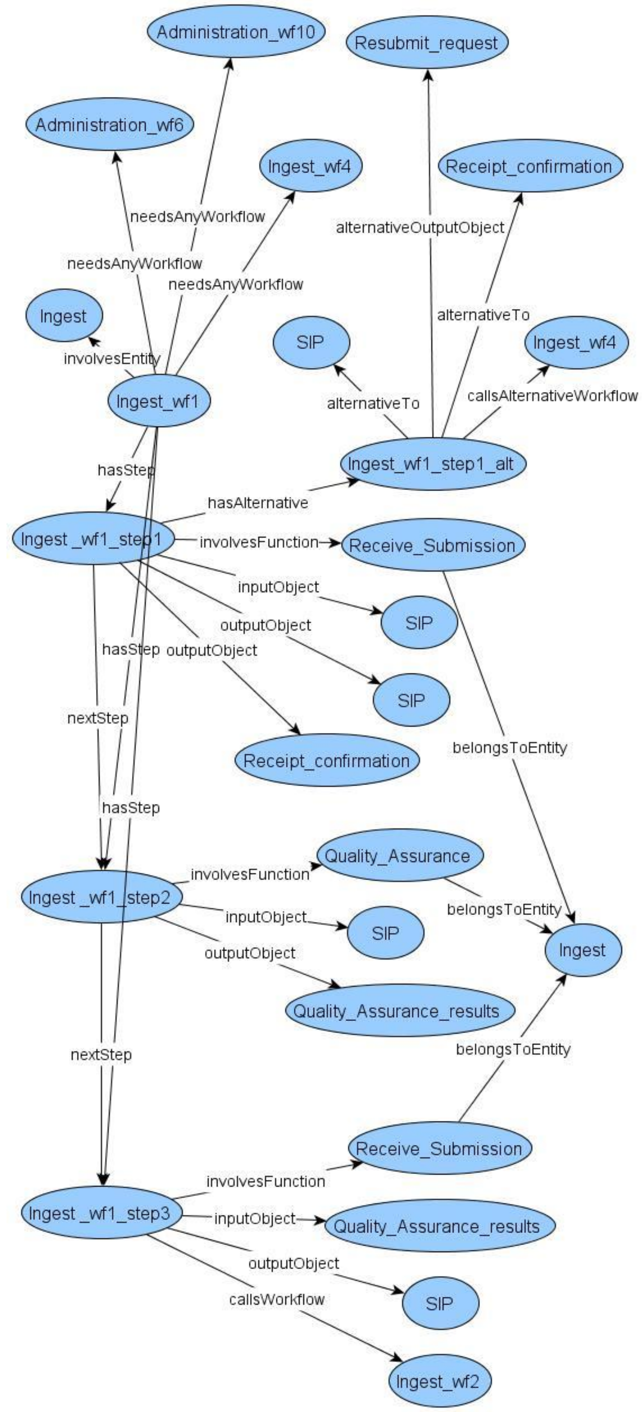

Figure 3. Ingest first workflow

```
  <ObjectExactCardinality cardinality="1">
    <ObjectProperty IRI="#hasStep"/>
    <Class IRI="#Ingest_wf1_step1"/>
  </ObjectExactCardinality>
</SubClassOf>
<SubClassOf>
  <Class IRI="#Ingest_wf1"/>
  <ObjectExactCardinality cardinality="1">
```

```
    <ObjectProperty IRI="#hasStep"/>
    <Class IRI="#Ingest_wf1_step2"/>
  </ObjectExactCardinality>
</SubClassOf>
<SubClassOf>
  <Class IRI="#Ingest_wf1"/>
  <ObjectExactCardinality cardinality="1">
    <ObjectProperty IRI="#hasStep"/>
    <Class IRI="#Ingest_wf1_step3"/>
  </ObjectExactCardinality>
</SubClassOf>
```

Due to space limits the rest definitions are not given in OWL but as Protégé fragments. The fact that the workflow Ingest_wf1 starts after the completion of any of the workflows Ingest_wf4, Administration_wf6 and Administration_wf10, is declared by the following fragment:

```
Ingest_wf1 needsAnyWorkflow exactly 1 Ingest_wf4
Ingest_wf1 needsAnyWorkflow exactly 1
Administration_wf6
Ingest_wf1 needsAnyWorkflow exactly 1
Administration_wf10
```

The definition of the first step of the workflow, named Ingest_wf1_step1, which refers to the subclass Receive_Submission of the class *Function*, as well as its inputs and outputs are presented in the following fragment:

```
Ingest_wf1_step1 involvesFunction exactly 1
Receive_Submission
Ingest_wf1_step1 inputObject exactly 1 SIP
Ingest_wf1_step1 outputObject exactly 1
Receipt_confirmation
Ingest_wf1_step1 outputObject exactly 1 SIP
```

The next step is named Ingest_wf1_step2. However the step Ingest_wf1_step1 has an alternative, named Ingest_wf1_step1_alt. The alternative step produces as output the object named Resubmit_request (instead of a Receipt_confirmation and a SIP) and of course it calls an alternative workflow named Ingest_wf4. These statements are presented in the Protégé fragment:

```
Ingest_wf1_step1 nextStep exactly 1
Ingest_wf1_step2
Ingest_wf1_step1 hasAlternative exactly 1
Ingest_wf1_step1_alt
Ingest_wf1_step1_alt alternativeTo exactly 1
Receipt_confirmation
Ingest_wf1_step1_alt alternativeTo exactly 1 SIP
Ingest_wf1_step1_alt alternativeOutputObject
exactly 1 Resubmit_request
Ingest_wf1_step1_alt callsAlternativeWorkflow
exactly 1 Ingest_wf4
```

The mentioned example constitutes just one indicative case of the set of the encoded workflows that come across during a preservation process. The rest of the workflows are modeled similarly and are available at the URL: http://www.ionio.gr/~papatheodor/papers/PreservationWorkflows.owl.

5. GUIDING THE WORKFLOWS

The proposed ontology constitutes a generic model for the representation of preservation workflows. An organization can use the ontology to tailor its own workflows and model its internal structure and functions. The choice of the workflows to be implemented depends on the nature of the organization, its own needs and internal functions as well as the specifications of its archival system. After the selection of the needed workflows, the organization officers should define the instances of the chosen workflows, their steps, the input and output objects, etc. Given that a subset of the ontology classes have been populated with instances, then a user, who interacts with the archive under a specific role and can execute a number of workflows according to the rights given to this role, could be navigated to the specified paths and monitor the execution of a set of workflows.

The interaction of that user with the archival system can start by selecting the execution of a primitive workflow, i.e. a workflow which is not related to any other workflows through the property *needsWorkflow*. Such a workflow can be executed at any time, regardless of other processes running simultaneously. Then, the user input is combined with information, which is provided to the archive by the prior periodical or on demand execution of other workflows and is stored in the archive database. This information may consist of standards, procedures, templates, statistics or internal policies. The ontology ensures the continuation of the data flows and guides the user by recommending what workflows and steps should be performed at each time point. Moreover, the workflow execution process may ask for the user interaction by providing the user with feedback and requesting additional input.

For instance, a Producer can send a submission information package (SIP) to the Receive_Submission function and call the workflow Ingest_wf1 to accept the SIP and manage the required processing. The person having the role of the producer is modeled as an instance of the class *Producer* and the object provided by the producer is modeled as an instance of the SIP subclass of the class Object. The ontology guarantees that the user will follow the processing paths specified by the properties of the ontology and their constraints, presented in Figure 3. The Receive_Submission function receives the SIP provided by the Producer and forwards it to the Quality_Assurance function, while it sends a Receipt_Confirmation object back to the Producer. Alternatively, if there are errors in the submission, a Resubmit_Request is sent back to the Producer and the appropriate workflow is called in order for the proper resubmission of the SIP. Quality_Assurance in turn receives the SIP and send back a Quality_Assurance_Results object. Finally, Receive_Submission, after getting the Quality_Assurance_Results, sends the SIP to the Generate_AIP function and ends workflow Ingest_wf1. The accomplishment of Ingest_wf1 activates the second workflow of the Ingest entity. After the successful performance of a sequence of workflows the object, i.e. the instance of the subclass SIP, is stored in the database of the archival system.

Hence the ontology guides precisely the user to perform the workflows needed to manage the preservation actions for its repository. Concluding, the ontology covers the whole spectrum of the registered workflows and encourages the preservation policy makers and administrators to experiment by either adding new workflow models or by selecting and populating the most appropriate from the existing ones that satisfy the needs of their organization.

6. CONCLUSIONS

Throughout this paper we proposed a model for the representation of the digital preservation workflows, as they can be found in an archival system. Our goal was to cover the entire preservation process and provide a common language to organizations concerned in the field of digital preservation. Therefore the development of the proposed model was mainly based on the OAIS Reference Model. OAIS is a general framework for understanding and applying concepts needed for long-term digital information preservation. The OAIS Reference Model does not specify a design or implementation. It provides a basis for organizations that aim to implement an archive, by defining general concepts related to long-term preservation. The proposed model provides a tool for specifying the desired preservation activities of an organization as well as it can recommend particular steps and alternatives to a user who runs a preservation activity. Its main advantageous design parameters are the expressiveness to define clearly the preservation workflows, as well as the interoperability and openness ensured by the usage of semantic web languages and open standards.

The present work can be treated in a more detailed way and constitute the basis for a future more elaborated study. The ontology can be used as groundwork for implementing a recommendation system enhanced with a graphical user interface, which will be used by organizations with large volumes of information. Such a system could be fed with a knowledge base, depending on the organization's data and needs, and provide a guide for the entire preservation process.

7. REFERENCES

[1] ARC. 2009. Guidelines for Creating and Installing IF Preservation Workflows and Templates. IF5-D1. Retrieved June 6, 2012: http://www.planets-project.eu/docs/reports/Planets_IF5-D1_Creating&Install_IF_Pres_Workflows.pdf.

[2] Becker, C., Kulovits, H., Rauber, A., and Hofman, H. 2008. Plato: A Service Oriented Decision Support System for Preservation Planning. In *Proceedings of the 8th ACM IEEE Joint Conference on Digital Libraries, JCDL'08,* (Pittsburgh, Pennsylvania, USA, June 16-20, 2008).

[3] Farquhar, A., and Hockx-Yu, H. 2007. Planets: Integrated Services for Digital Preservation, *The International Journal of Digital Curation, Vol 2* (2) 88 – 99. Retrieved June 6, 2012: http://www.ijdc.net/index.php/ijdc/article/view/45.

[4] Fischer, L. 2003. Workflow handbook. Future Strategies Inc., ISBN 0-9703509-4-5.

[5] Hollingsworth, D. 1995. Workflow Management Coalition. The Workflow Reference Model.

[6] ISO. Space data and information transfer systems – Open archival information system – Reference model (ISO 14721:2003), 2003.

[7] OASIS. 2007. Web Services Business Process Execution Language Version 2.0: OASIS Standard. Retrieved June 6, 2012: http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf.

[8] Rechert, K., Von Suchodoletz, D., Welte, R., Van Den Dobbelsteen, M., Roberts, B., Van Der Hoeven, J., Schroder, J. 2009. Novel Workflows for Abstract Handling of Complex Interaction Processes in Digital Preservation. In *Proceedings of the Sixth International Conference on Preservation of Digital Objects, IPRES'09,* (San Francisco, California, October 5-6, 2009).

[9] Schmidt, R., King, R., Jackson, A., Wilson, C., Steeg, F., and Melms, P. 2010. A Framework for Distributed Preservation Workflows, *The International Journal of Digital Curation, Vol 5* (1) 205 – 217. Retrieved June 6, 2012: http://ijdc.net/index.php/ijdc/article/view/157.

[10] Sierman, B., and Wheatley, P. 2010. Evaluation on Preservation Planning with OAIS, based on the Planets Functional Model. PP7-D6.1. Retrieved June 6, 2012: http://www.planets-project.eu/docs/reports/ Planets_PP7-D6_EvaluationOfPPWithinOAIS.pdf.

[11] Strodl, S., Becker, C., Neumayer, R., and Rauber, A. 2007. How to Choose a Digital Preservation Strategy: Evaluating a Preservation Planning Procedure. In *Proceedings of the 7th ACM IEEE Joint Conference on Digital Libraries, JCDL'07,* (Vancouver, British Columbia, Canada, June 18-23, 2007), 29-38.

[12] Strodl, S., and Rauber, A. 2008. Preservation Planning in the OAIS Model. *New Technology of Library and Information Service* (Beijing, China), International Book Trading Corporation, 61 – 68.

[13] The Workflow Management Coalition Specification, Workflow Management Coalition Workflow Standard: Process Definition Interface -- XML Process Definition Language, Document Number WFMC-TC-1025, 2008.