Towards a Reference Architecture for Archival Systems: Use Case With Product Data

Raphael Barbau[1]
Le2i, Université de Bourgogne,
BP 47870, 21078 Dijon, France;
Systems Integration Division,
National Institute of Standards and Technology,
Gaithersburg, MD 20899

Joshua Lubell
Systems Integration Division,
National Institute of Standards and Technology,
Gaithersburg, MD 20899

Sudarsan Rachuri
Systems Integration Division,
National Institute of Standards and Technology,
Gaithersburg, MD 20899

Sebti Foufou
CSE Department,
CENG, Qatar University,
Doha, Qatar;
Le2i, Université de Bourgogne,
BP 47870, 21078 Dijon, France

Long-term preservation of product data is imperative for many organizations. A product data archive should be designed to ensure information accessibility and understanding over time. Approaches, such as the Open Archival Information System Reference Model (OAIS RM) and the Audit and Certification of Trustworthy Digital Repositories (ACTDR), provide a framework for conceptually describing and evaluating archives. These approaches are generic and do not focus on particular contexts or content types such as product data. Moreover, these approaches offer no guidance on how to formally and comprehensively describe archival systems. Such descriptions should include the business activities that a product data archive has to support and the systems that interact with the archive. Enterprise architecture provides a means to describe systems in their potentially complex environments. This paper proposes a holistic approach to formally describe the architecture and the environment of archival systems. This approach relies on the formal representation of the preservation terminology, including OAIS concepts, using the Department of Defense Architecture Framework (DoDAF). The approach covers the various interactions of other business functions with the archive and the information models necessary to ensure preservation and accessibility of product data. This approach is a step towards a reference architecture for the formal description of archival systems. To demonstrate the approach, we formally describe the ingest of product data related to a ship. The resulting description uses the preservation terminology defined in the OAIS Reference Model. It facilitates the understanding of how the preservation solution is actually implemented and provides evidence that the solution is able to preserve product data and make it accessible. [DOI: 10.1115/1.4027150]

1 Introduction

A large amount of digital information is produced and consumed every day. Although most of this information is for immediate consumption, many organizations have an interest in long-term preservation [1]. The main motivations for preserving information are reusing existing knowledge or keeping proofs of past events. Besides typical digital data management, specific information and activities are needed to ensure data's long-term preservation and accessibility [2,3]. These information and activities are part of a dedicated entity: the archive. Design of an archive is a key factor in successful preservation, especially when complex information and activities are involved. Product data preservation is a good example of such complexity.

A complex product may be composed of numerous systems and parts, and for each part, various product data may be produced from conception to disposal. Product data may be formally represented through large and complex information models. The metadata necessary to organize, interpret, or prove the authenticity of the information may also be complex. Finally, the interactions between the different product data repositories and the archive may be complex and may involve many different stakeholders at different product lifecycle stages.

One issue of this complexity is that the production and the consumption of preserved information are usually part of business functions not dedicated to preservation. This means that the archive has to be well integrated within the organization.

Digital preservation involves computer systems that can automatically process information. In this paper, such systems in the context of preservation are referred to as archival systems, while the term archive designates the people and systems involved in the preservation. The design of archival systems should include the information, activities, systems, and other concepts needed to carry out the preservation mission. The objective of this paper is to propose a way to formally describe these elements.

The modeling of systems and their environment in the context of an enterprise is addressed by the Enterprise Architecture (EA). EA establishes a link between the missions of an organization and the actual activities undertaken to achieve these missions. EA typically supports the description of systems, services, activities, information, and constraints related to systems and their environment within an organization. EA can be leveraged to detail how the preservation strategy is implemented. In particular, it is possible to describe in a coherent manner the various activities and the information involved in the preservation.

Different efforts have attempted to determine the common elements of archival systems. The Reference Model for an Open Archival Information System (OAIS RM) [4] is a mature conceptual framework for describing and comparing archives. It defines a common terminology for information preservation and defines a functional model that shows the generic activities an archive performs, whatever the preserved content is. The OAIS RM also defines an information model for describing preserved content and additional metadata in order for this content to stay understandable and accessible over time.

The OAIS RM is widely accepted in the product data preservation community [5,6]. However, its functional and information models are generic and conceptual: they are not meant to be

[1]Corresponding author.

Contributed by the Computers and Information Division of ASME for publication in the Journal of Computing and Information Science in Engineering. Manuscript received February 10, 2014; final manuscript received February 25, 2014; published online April 28, 2014. Editor: Bahram Ravani.

directly implemented, but rather to serve as guidance for preservers to develop their own solutions. So, the OAIS RM neither targets specific content nor does it prescribe any technology for the implementation of the archival system. Moreover, it does not prescribe how to incorporate the OAIS concepts into an actual archival system design.

This paper presents an approach that combines the concepts and terminology defined in the OAIS RM with those used in EA to allow formal description and comparison of archival system architectures. By using a formal description, the preservation concepts are explicitly referred to, which increases the understanding of the design, ensures consistency among the various elements described, and ultimately leads to an implementation of high quality. This approach is a first step towards the definition of a generic reference architecture to guide and constrain the description of archival systems.

Besides the OAIS RM, another standard was developed to help preservers designing archival systems. The Audit and Certification of Trustworthy Digital Repositories (ACTDR) provides specific criteria for the certification of repositories. ACTDR helps to determine whether the repository can be trusted with regards to long-term access. Our approach addresses some of the ACTDR criteria, by generating certification evidence.

We demonstrate our approach by formally describing a scenario where product data from a Product Data Management (PDM) system is sent to an archive. The data used in this scenario are actual ship design data, from the US Navy Torpedo Weapon Retriever (TWR). Through the use of OAIS RM concepts and ACTDR evidence, the description aims to demonstrate the archive's ability to make product data understandable and accessible over time.

This paper is organized as follows. Section 2 presents background information on preservation and enterprise architecture. Section 3 presents the approach for combining the OAIS RM and EA to enable the description of an archival system. Although the approach does not focus on a particular content type, it can be used to represent, in a coherent fashion, the complex interactions and information models involved in product data preservation. Section 4 presents how this approach is used to describe the ingest of product data. Finally, Section 5 presents our conclusions.

2 Background on Digital Preservation and Enterprise Architecture

This section provides background information about the conceptualization, development, and certification of archival systems. It also introduces EA, and particularly the Department of Defense Architecture Framework our approach uses.

2.1 Conceptual Frameworks and Certification of Archival Systems. The OAIS RM (ISO 14721) [4] proposes a conceptual framework for describing and comparing archives. It defines the terminology related to information preservation, including the types of information required to ensure preservation and accessibility of the content, and the main functions that an archive should support.

The OAIS RM defines the different kinds of information in an archive. This information, composed of content and metadata, is encapsulated in information packages. Three kinds of information packages are defined. The Submission Information Package (SIP) refers to what the producer sends to the OAIS. The Archival Information Package (AIP) refers to what the archive stores. The Dissemination Information Package (DIP) refers to what the archive delivers to the consumer. Preservation Description Information (PDI) refers to information added to the content to ensure its preservation. Descriptive Information (DI) is a subset of PDI used to locate the desired information.

The OAIS RM describes the main functions of an archive: Ingest, Access, Archival Storage, Data Management, Preservation Planning, and Administration (see Fig. 1). The Ingest function receives the SIPs and generates AIPs to be sent to Archival Storage and descriptive information to be sent to Data Management. The Data Management function sends some descriptive information to the Access function when needed, and the Archival Storage function sends the desired AIP to the Access function. Then, the Access function returns a DIP to the consumers. The Preservation Planning function monitors the environment of the OAIS and makes recommendations regarding the evolution of the archive. The Administration function, directed by the management, establishes the overall preservation strategy of the OAIS. Each function is further decomposed into smaller functions in the OAIS RM.

Both information and functions are presented in a conceptual and generic way: they are not tied to a particular domain or implementation method. The OAIS RM explains which aspects should be considered to develop an archive, but solutions still need to be tailored to the specific content to be preserved and the context of the preservation. OAIS information and functions need to be elaborated upon, and implementation technology needs to be chosen. Also, the OAIS RM does not make the distinction between functions performed by humans and functions performed by computers. The approach presented in this paper identifies functions performed by computers.

ACTDR [7] is a standard for the certification of an OAIS. It addresses organizational aspects that are not considered in the OAIS RM, and it gives more details about what is expected from the archive. The certification concerns three different areas: the organizational infrastructure, the digital object management, and the security risk management. Each area is composed of requirements, and each requirement gives examples of how to demonstrate that the organization meets that requirement. For example, ACTDR requires describing the whole organizational unit related to the archive and providing financial information to demonstrate the financial sustainability of the archive. Because our approach addresses computer science aspects, our approach considers only ACTDR requirements that can be demonstrated by the architectural description of the archival systems (such as system functions and composition, information models, systems interactions).

2.2 Introduction to Enterprise Architecture and its Use for Information Preservation. Our approach to design archival systems is to rely on EA. EA is the discipline of formally describing an enterprise, in particular, the systems that compose it. An enterprise can be defined as an organization or a subset of an organization. EA describes how the objectives of an enterprise are realized through systems, services, and activities [8]. EA provides an abstract view that makes it easier to understand how the enterprise works and how the systems are integrated. The actual description of an enterprise or of one of its parts is called architectural description.

Using EA for representing systems requires two components: a method that provides the steps in the development of the architecture and the tools to concretely describe this architecture, for example, by providing a metamodel. Different Enterprise Architecture Frameworks (EAFs) propose varying approaches to describe enterprises, and sometimes they focus on the aspect they judge the most important. For example, The Open Group Architecture Framework (TOGAF) [9] is an EAF well known for its Architectural Development Method (ADM). TOGAF has not incorporated a metamodel until recently. On the other hand, the Ministry of Defence Architecture Framework (MODAF) [10] and the US Department of Defense Architecture Framework (DoDAF) [11] focus on defining a metamodel and a set of views to formally represent architectural descriptions. Other EAF include the Generalized Enterprise Reference Architecture and Methodology (GERAM), developed by the IFIP-IFAC Task Force [12–14] and adopted as an Appendix of ISO15704:2000 [15] is a generalized EAF for enterprise integration and business process engineering. GERAM defines all the components required for use in enterprise engineering. Other well-known reference architectures are the

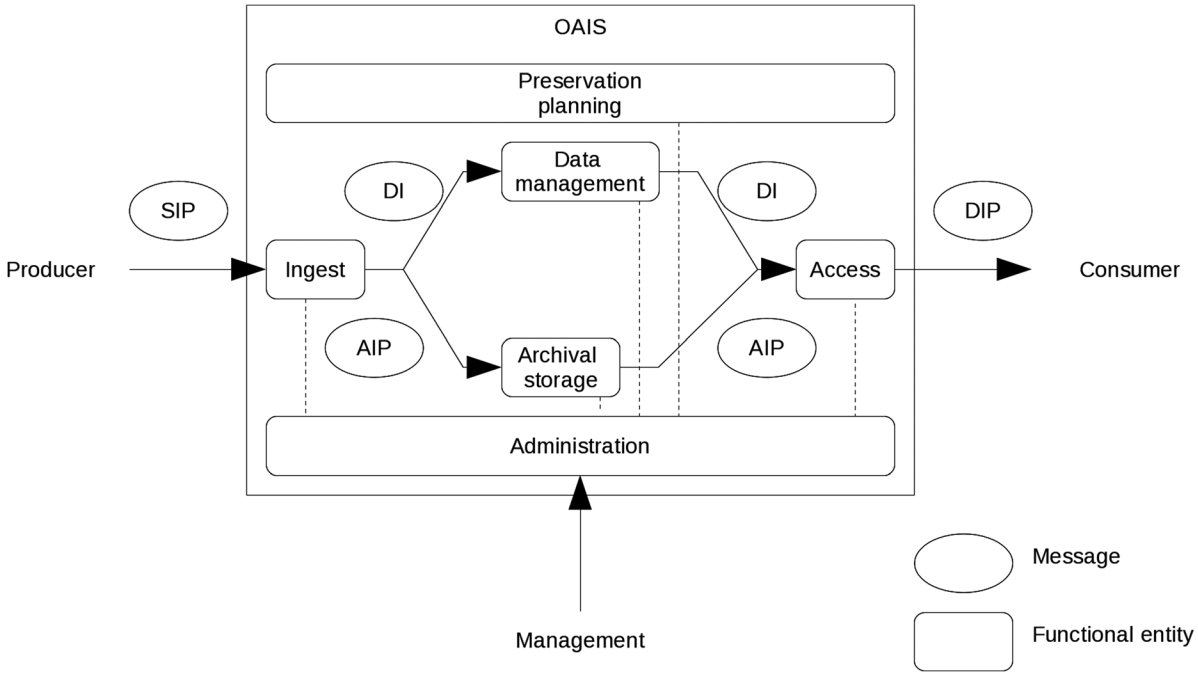

Fig. 1 OAIS functional model

Purdue Enterprise Reference Architecture (PERA) [16], CIMOSA [17], and GRAI-GIM [18].

Becker et al. presented a reference architecture approach for archives, which emphasized the development process rather than the description of the actual implementations [19]. Approach of Becker et al. was to highlight the different recommendations and standards that need to be considered during the design of the archive. Becker et al. took into account preservation recommendations and information technology standards related to risk management, data quality, or security. However, they did not address the formal description of the archival system architecture.

This paper presents an approach that focuses on the formal description of archival system architectures. Our approach consists in (1) formally representing the preservation terminology to be used in architectural descriptions and (2) defining a set of views to describe important aspects of an archival system. We rely on DoDAF to provide the core enterprise architecture concepts to which preservation concepts relate. A formal description offers several benefits: (1) the preservation concepts are explicitly referred to, which increases the understanding of the design, (2) the same preservation concepts can be reused across multiple descriptions, (3) the different elements of the description can be consistently described and reused under different perspectives, and (4) parts of the formal description can lead to software implementation using a model-driven architecture approach.

DoDAF proposes a framework for creating architectural descriptions [11]. It was created to meet the needs of US Department of Defense, but it is generic enough to be used in any organization. It provides two major components to build formal architectural descriptions. The first component is a metamodel called DoDAF Meta-Model (DM2) that defines the concepts used in architectural descriptions as well as the relationships among these concepts. The second component is a collection of views to describe an architecture under different perspectives. The views are organized in viewpoints: the All Viewpoint, the Capability Viewpoint, the Operational Viewpoint, the Services Viewpoint, the Systems Viewpoint, the Data and Information Viewpoint, the Standards Viewpoint, and the Project Viewpoint.

The DoDAF meta-model is implemented as an extension of the Unified Modeling Language (UML) [20] in the Unified Profile for DoDAF/MODAF (UPDM) [21]. UPDM makes it possible to develop architectural descriptions with generic UML modeling tools. DoDAF concepts are implemented as stereotypes, and views are implemented as UML diagrams.

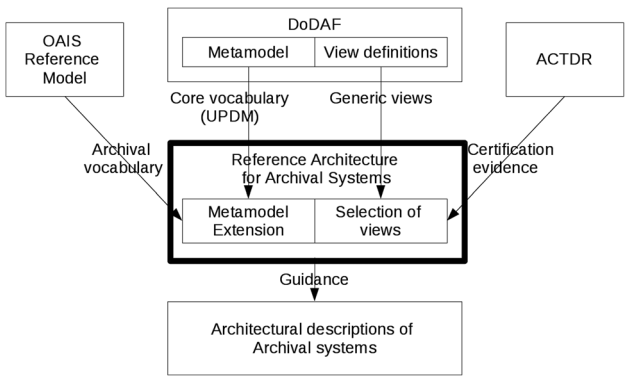

Fig. 2 Presentation of the approach

3 Architectural Description of Product Data Archives

The approach presented in this paper relies on enterprise architecture to describe archival systems, to formally describe the interactions involving the archival system, and the functions performed by this system.

This approach, depicted in Fig. 2, is split into two parts. The first part extends the metamodel defined in DoDAF to incorporate a new archival vocabulary derived from the OAIS RM. The expanded terminology can then be used to represent archival system elements in architectural descriptions. The second part selects DoDAF views to represent specific aspects of the preservation solution. These views can serve as ACTDR evidence to demonstrate the ability of the archival system to preserve information. Note that a complete archival system description would go beyond what is in the scope of our approach, so other views may be used to address other aspects.

3.1 Representation of Preservation Concepts. DoDAF provides a generic implementation model to describe architectures,

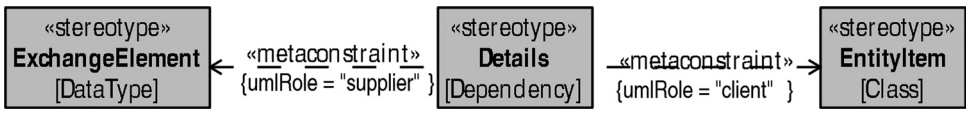

Fig. 3 Notation used for stereotypes

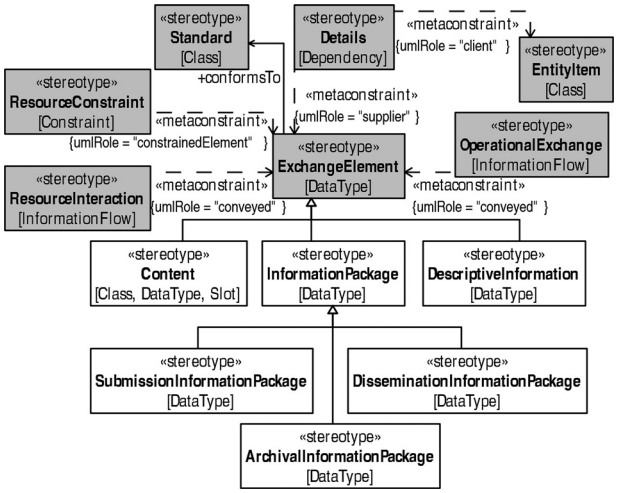

Fig. 4 Extension of ExchangeElement

and the OAIS RM provides conceptual model to describe archival systems. Our approach combines both to allow a formal description of archival systems using enterprise architecture and by incorporating the preservation terminology. Only the terminology related to the archival system and the interactions between this system and the environment are considered. Preservation functions, such as customer monitoring or technology monitoring, are present in the OAIS RM, but they are not in the scope of our approach. The OAIS RM describes archives in a conceptual and technology-independent manner, while DoDAF describes concrete implementations within an organization. So, more preservation concepts can be inferred by determining what DoDAF concepts would be used in an archival system description. The approach uses UPDM, an implementation of DoDAF as a UML profile, which allows using the DoDAF terminology in UML tools.

To understand our approach, knowledge of the profiling mechanism and of the UML metamodel is required [22]. UML defines a metamodel, which is composed of metaclasses such as Class, Property, Operation, or Activity. These metaclasses are then instantiated to create UML models. The profiling mechanism enables the creation of stereotypes that extend these metaclasses. For example, if someone creates a stereotype S that extends the Class metaclass, they can then create a Class C and apply the stereotype S to it. The profiling mechanism is generally used to add more semantics to the models. In the case of UPDM, the stereotypes correspond to enterprise architecture concepts defined in DoDAF. In this paper, we introduce new stereotypes which correspond to preservation concepts.

In the following diagrams, UPDM concepts have gray background and preservation concepts have white background.

3.1.1 Graphical Notation to Define the Concepts. We use the same notation as the UPDM specification to define stereotypes. This notation shows in one diagram the stereotypes, the extended metaclass, and the constraints that impact the metaclass if the stereotype is applied. These constraints, called metaconstraints, are represented as UML dependencies along with the UML property being constrained.

To illustrate this notation, we present three UPDM stereotypes in Fig. 3, which will be reused and explained afterwards, in Fig. 4: *Details*, *ExchangeElement*, and *EntityItem*. The *Details* stereotype extends the *Dependency* metaclass, the *ExchangeElement* stereotype extends the *DataType* metaclass, and the *EntityItem* stereotype extends the *Class* metaclass. Two metaconstraints are defined for *Details*. The first one goes from *Details* to *ExchangeElement* and has *supplier* as *umlRole*. This means that a UML element stereotyped by *Details* must have a UML element stereotyped by *ExchangeElement* as supplier. The second one goes from *Details* to *EntityItem* and has *client* as *umlRole*. This means that a UML element stereotyped by *Details* must have a UML element stereotyped by *EntityItem* as client.

3.2 Adapting the DoDAF Concepts for Preservation. The following paragraphs present the preservation concepts considered in our approach. The objective is to provide a way to formally describe the preserved content, the information packages, the representation information, the preservation description information, the operational nodes, the activities, the system functions, the constraints, and the standards.

3.2.1 Content and Information Packages. Content and information packages are OAIS concepts relating to information that is exchanged during preservation-related activities. Content is the information that is meant to be preserved. Information packages encapsulate content information as well as additional information required to ensure a long-term preservation and accessibility. The SIP, AIP, and DIP represent the information packages, respectively, as they are received, preserved, and disseminated. In UPDM, the concept *ExchangeElement* represents a resource exchanged during an activity, so both content and information packages are defined as the following specializations of *ExchangeElement*:

- SubmissionInformationPackage
- ArchivalInformationPackage
- DisseminationInformationPackage
- Content

In the context of product data preservation, the *SubmissionInformationPackage*, the *ArchivalInformationPackage*, and the *DisseminationInformationPackage* represent the container of product data during ingest, retention, and dissemination activities, respectively. A *Content* may represent the actual product data, or a piece of information that corresponds to the target of the preservation. This paper does not consider only data (file) as content, but also the information (logical entities) represented by the data.

As it can be seen in Fig. 4, an *ExchangeElement* can have a *Details* relationship to an *EntityItem*. *EntityItems* are UML *Classes*, and the *Details* relationship shows how a piece of information is implemented by a logical structure (class or data type). *Details* are UML *Dependencies*, which have an *ExchangeElement* as *supplier* and *EntityItems* as *client*. An *ExchangeElement* can also be the subject to a *Constraint* and may have to conform to a particular *Standard*. Finally, an *ExchangeElement* can be conveyed during an activity (*OperationalExchange*) or during an interaction between resources (*ResourceInteraction*).

3.2.2 Representation Information and Presentation Description Information. Within information packages, the OAIS RM defines PDI and representation information that can be attached to the content. Representation information represents information that gives more meaning to the data: it could be everything that allows computers or humans to interpret the data such as format specifications, software, or dictionaries. PDI is further detailed in

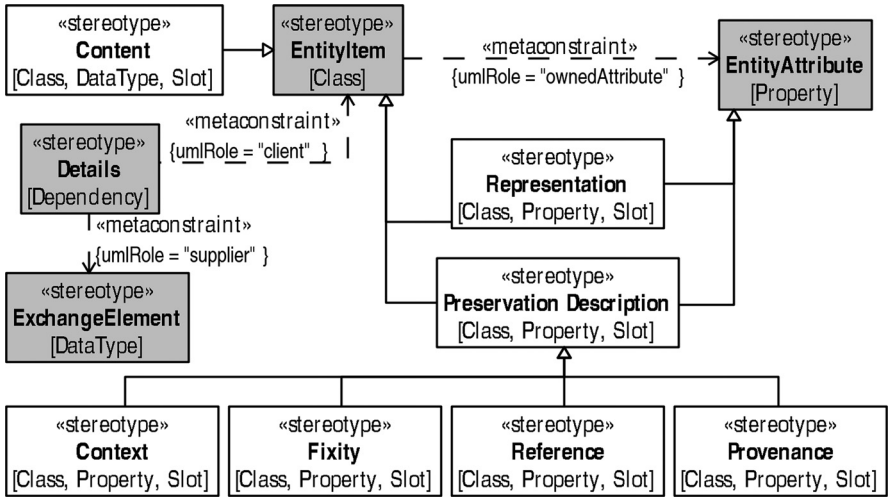

Fig. 5 Extension of EntityItem

four categories. Reference information identifies the content. Provenance information describes the content history. Fixity information represents the information used to check that the content is not altered. Finally, Context information provides the relationships between a *Content* and the other various contents.

In UPDM, these types of information can be seen either as *EntityItems*, or as *EntityAttributes*. *EntityItems* are UML *Classes*, and *EntityAttributes* are UML *Properties*. An *EntityItem* has *EntityAttributes* as owned attributes. It is possible to refer to PDI and representation information either for the definition of a particular type, or for the use of a type. Here are the specializations of *EntityItem* and *EntityAttribute*, as shown in Fig. 5:

- Representation
- Reference
- Fixity
- Context
- Provenance

In the context of product data preservation, some of the preservation description information can be extracted from Product Data Management or Product Lifecycle Management (PLM) systems such as identifiers, creators, or relationships among product data.

3.2.3 Nodes. Producers and consumers can also be seen as *nodes* instead of physical persons. A *node* is a logical abstraction, meaning that it may correspond to people or systems. The following specializations of nodes are added:

- Producer
- Consumer
- Preserver
- Archive

In the context of product data preservation, *Producer* may correspond to the data source from which the product data originate (e.g., PDM systems), *Archive* abstracts the physical realization of the archival system, and *Consumer* may correspond to where the product data is used over time. *Preserver* can represent the persons in charge of the preservation of product data.

As shown in Fig. 6, Nodes can be connected to OperationalActivities by IsCapableOfPerforming. OperationalActivities are UML Activities, and IsCapableOfPerforming relationships are UML Dependencies, which have OperationalActivities as supplier and Nodes as client. Implements is a UML Dependency that has a Node as supplier and Software as client. Software is a UML Class. An OperationalExchange represents an exchange of information between two Nodes. OperationalExchanges are UML InformationFlows, which have elements stereotyped by Node as informationSource and informationTarget.

3.2.4 Activities. Three kinds of activities can be identified: the interaction between the archive and the producers, consumers, and management constitute, respectively, ingest, access, and management activities. In addition, the activities that are within the OAIS are also defined, in particular, the preservation activities that include update and disposal of the preserved content. All of these activities are defined as specialization of *OperationalActivities* in UPDM:

- IngestActivity
- PreservationActivity
- AccessActivity

In the context of product data preservation, *IngestActivity* may represent the activities of taking product data from their original place, preparing a *SubmissionInformationPackage*, and sending it to the archive. *PreservationActivity* may represent the activities undertaken for preserving the product data over time by accessing the archival system. *AccessActivity* may represent the activities that request product data from the archive.

As seen in Fig. 7, an *OperationalActivity* is a UML *Activity* that has many relationships. *SupportsOperationalActivity* is a UML *Dependency* that has an *OperationalActivity* as supplier and a *ServiceInterface* as client. *IsCapableOfPerforming* is a UML *Dependencies* that has *OperationalActivities* as supplier and *Nodes* as client. *OperationalActivities* can be the subject of *OperationalConstraints*.

3.2.5 Services. Services constitute another concept that is important in the implementation of archival systems. Nowadays, many software development approaches rely on a Service-Oriented Architecture. UPDM supports this approach by defining the notion of *service*. Regarding the archive, different services are considered, as seen in Fig. 8:

- *IngestServices* are the services exposed to the producers for the ingest
- *ManagementServices* are the services exposed to the preservers to make sure the content stays interpretable
- *AccessServices* are the services exposed to the consumers for accessing the preserved content.

ServiceInterface are UML Interfaces. SupportsOperationalActivity is a UML Dependency that has an OperationalActivity as supplier and a ServiceInterface as client. Service is a UML Port which is owned by a Software, and which has a ServiceInterface as type. A ServiceInterface has ServiceOperations, which are

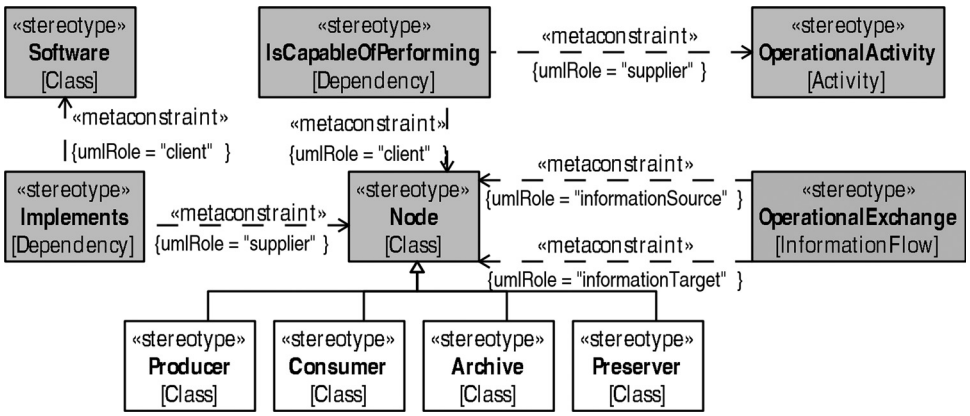

Fig. 6 Extension of nodes

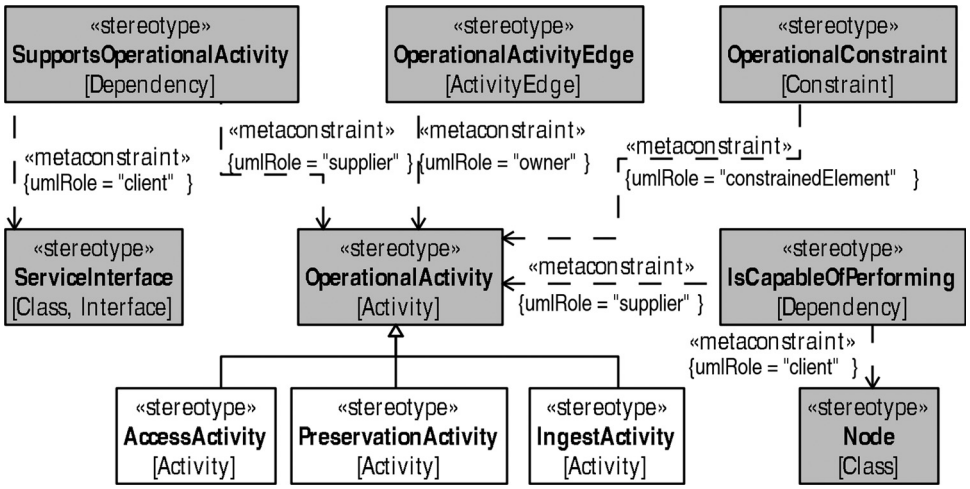

Fig. 7 Extension of operational activities

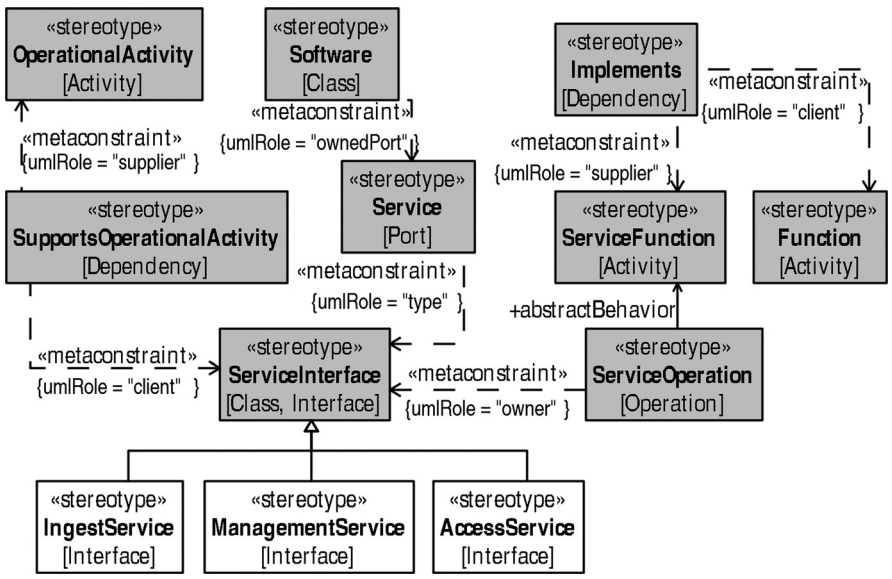

Fig. 8 Extension of services

UML Operations and are abstracted by ServiceFunctions. Implements is a UML Dependency with a ServiceFunction as supplier and a Function as client.

3.2.6 System Functions. The OAIS RM defines various functions that an OAIS performs. Two kinds of functions can actually be differentiated: those intended to be performed by humans and

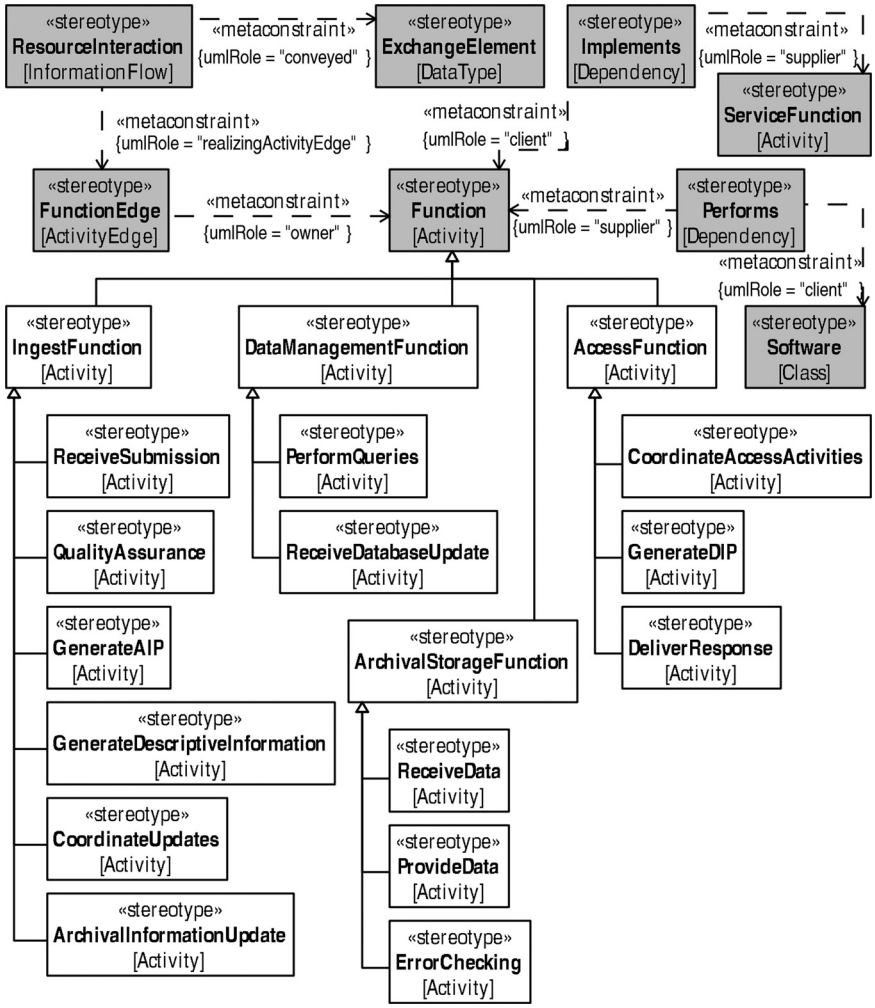

Fig. 9 Extension of function

those intended to be performed by computers. UPDM makes the distinction between these two types, and calls them, respectively, *OperationalActivities* and *Functions*. The *Functions* that are likely to be implemented by systems are the following (see Fig. 9):

- *IngestFunction* ingests the content
- *ManagementFunction* manages the preserved content
- *AccessFunctions* makes the preserved content accessible

Each of these functions uses OAIS functions:

- *ReceiveSubmission* receives the SIP
- *QualityAssurance* makes sure the SIP is correct
- *GenerateAIP* generates an AIP from the SIP
- *GenerateDescriptiveInformation* extracts the descriptive information from the SIP
- *CoordinateUpdates* synchronizes the storage of the SIP and the descriptive information updates
- *ReceiveData* receives an AIP and stores it
- *ProvideData* retrieves a stored AIP
- *ReceiveDatabaseUpdate* updates the database with descriptive information
- *ErrorChecking* checks that a stored AIP is not altered
- *ArchivalInformationUpdate* modifies an AIP to preserve the information
- *PerformQueries* queries the descriptive information to discover preserved content
- *CoordinateAccessActivities* manages the consumers queries and asks for a stored AIP
- *GenerateDIP* generates a DIP from an AIP
- *DeliverResponse* gives a response to the consumer

Functions are UML *activities*. *Performs* is a UML *Dependency* that has a *Function* as supplier and a *Software* as client. *Implements* is a UML *Dependency* that has a *ServiceFunction* as client, and a *Function* as supplier. A *Function* may own *FunctionEdges*, which can be involved in *ResourceInteractions*. *ResourceInteractions* is a UML *InformationFlow* that can convey *ExchangeInformation*.

3.2.7 Constraints. UPDM includes the notion of constraint that may apply to various UPDM concepts. Two types of constraints are actually used: *OperationalConstraints* applies to *OperationalActivities*, and *ResourceConstraints* applies to *ExchangeElements*. Both constraints are UML *Constraints*. The proposed approach includes representing the following constraints (see Fig. 10):

- *ContentAccess* constraints apply to a content and restrict the access activities
- *ContentModification* constraints apply to a content and define what manipulations the archive can perform
- *ArchiveAccess* constraints apply to the services and constrain who is allowed to access the archive
- *ContentValidation* constraints apply to the contents and define what makes it valid

3.2.8 Standards. UPDM defines the notion of standard. Standards in this use are not restricted to international standards, but are

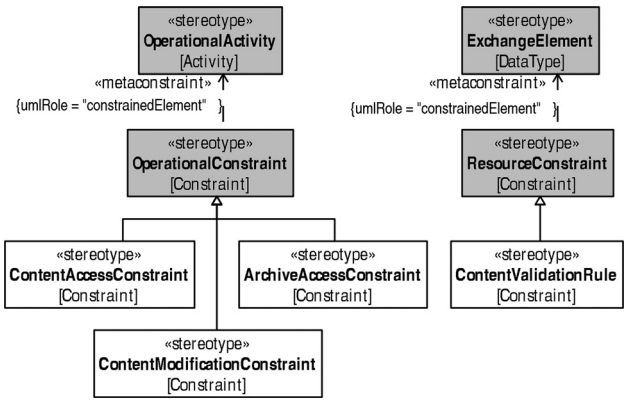

Fig. 10 Extension of constraints

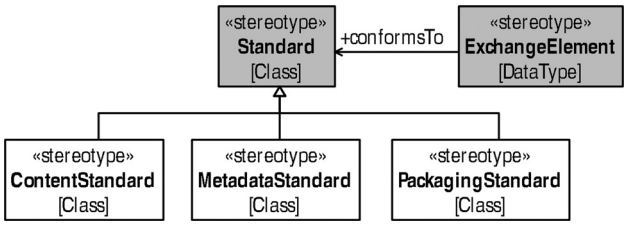

Fig. 11 Extension of standards

defined broadly as formal agreements. So, they can be used to represent the following aspects (see also Fig. 11):

- *ContentStandards* are formats used for the content
- *MetadataStandards* are formats used for PDI and the descriptive information
- *PackagingStandards* are formats used for the information packages

All these *Standards* are UML *Classes*, and apply to *ExchangeElement*.

In the context of product data preservation, content formats may be product data formats such as Computer-Aided Design (CAD) or PDM formats.

3.3 List of DoDAF Views and Their Use for Archival System Descriptions. Architectural descriptions are basically a set of views that, taken together, provide a comprehensive description. DoDAF proposes a list of views, and each view focuses on a particular aspect of the architecture. In the context of the architectural description of an archive, a view can describe how a particular aspect of the archive is implemented.

The ACTDR certification requires proofs that criteria have been met, including descriptions of how the archive functions. DoDAF views can serve as such evidence. However, ACTDR, unlike the OAIS RM, has a broader scope than the archival system. For example, it includes organizational aspects that are not considered in this approach.

Here is a nonexhaustive list of views that can be used in the proposed approach. When a view can serve as evidence for ACTDR, the relevant section of ACTDR is indicated as well as the role of the view.

Operational Resource Flow Description (OV-2) presents the resource exchanged within an enterprise. It can be used to show the information exchanged between the archive and its environment (through SIP or DIP). OV-2 can be useful to represent the flow of product data between the different systems over time.

Operational Activity Model (OV-5b) details the enterprise activities. The activities that can be detailed in the context of the archive are the ingest, access, update, audit, and disposal activities. For example, an OV-5b diagram can be created for each ingest scenario. Each scenario would detail the steps between the extraction of product data from their original data source and the submission of the product data to the archival system.

Event-Trace Description (OV-6c) represents interactions that occur during an activity. An OV-6c can represent the interactions with the archive, and can show the ingest, access, and management scenarios. An OV-6c diagram can represent how a person will interact with the original data source to retrieve product data and then interact with the archival system. This diagram can display how the services provided by the archival system are used in particular cases.

Conceptual Data Model (DIV-1), Logical Data Model (DIV-2), and Physical Data Model (DIV-3) represent conceptual, logical, and physical data models, respectively. They can serve various purposes with regards to the archive. One use could be to show what type of information is preserved, and what the important characteristics of the preserved content are. Another use could be to represent the SIP, AIP, and DIP data models. Yet another use is to define the metadata schemas supported by the archive, including the Provenance, Reference, Context, Fixity, and Representation information used for every type of content.

In the case of product data, these diagrams can represent the information models of the content, or the metadata for this content.

ACTDR evidence: Collection policy [7, Sec. 3.1.3], content information and information properties [7, Sec. 4.1.1], SIP definition [7, Sec. 4.1.2], and descriptive metadata definition [7, Sec. 4.5.1] including provenance [7, Sec. 4.1.4], reference [7, Sec. 4.2.4], fixity [7, Sec. 4.4.1.2], representation [7, Sec. 4.2.5] information

Standards Profile (StdV-1) is used to represent the standards, recommendations, or guidance used in the architectural description. It can be used to define, for example, the standards and formats required to recognize and parse information packages and their content.

Standards for product data can include information models such as those defined in ISO 10303, informally known as the Standard for the Exchange of Product model data (STEP) [23], or languages used to represent the data such as eXtended Markup Language (XML) [24].

ACTDR evidence: SIP parsing technologies [7, Sec. 4.1.3], content formats used [7, Sec. 4.3.2].

Operational Rules Model (OV-6a) defines the business rules that apply during the activities. In the context of the archive, many different rules can apply during the ingest, management, and access activities. For example, property rights, access rights, or security rules, can be defined.

ACTDR evidence: deposit agreement [7, Sec. 3.5.1], intellectual property rights [7, Sec. 3.5.2].

Systems Functionality Description (SV-4) is used to represent system functions. It can be used to show what actions are performed, and also which subsystem or component performs these actions.

ACTDR evidence: SIP verification [7, Sec. 4.1.5], AIP generation [7, Sec. 4.2.2], AIP disposal [7, Sec. 42.3], Error checking [7, Secs. 4.2.8, 4.2.9, and 4.4.1.2], Receive data [7, Sec. 4.4.1], Generate DIP [7, Sec. 4.6.2].

4 Use Case: Architectural Description of Ship Product Data Ingest

In this section, we show how the proposed approach is used to describe the ingest of product data. The objective is to describe the major architectural elements (such as activities, actors, information, and functions) in a coherent manner. Thanks to the approach defined in Sec. 3, we can describe these architectural elements under different perspectives, and formally relate them to a core preservation terminology.

This description has several benefits. First, it shows how the preservation concepts defined in the OAIS RM are applied to

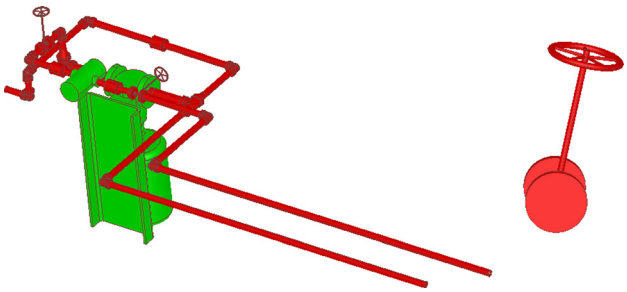

Fig. 12 3D representation of a fuel oil system

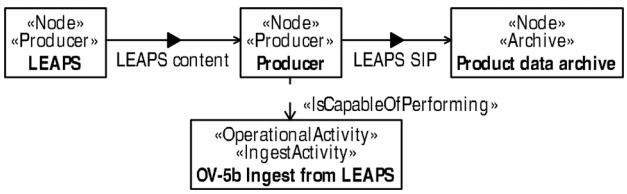

Fig. 13 OV-2 view for ingest and access of product data

concretely design implementations. The use of these concepts facilitates the understanding of the preservation strategy. The description contributes to demonstrating that the archival system can actually meet the preservation needs. Second, the description provides traceability from business requirements and preservation concepts to system design. If only a high-level description is presented here, more detailed software architectures can be derived from this description.

The target of the preservation used in this use case is product design data from a ship. The architectural description uses parts of the views and parts of the vocabulary defined in Sec. 3, and relies on DoDAF and UPDM.

Ship product data may be preserved for various reasons, for example, disassembly or maintenance. Maintenance activities often need product data that was created long ago. Ships often remain in service for more than 30 years, so the product data required to support the ship during its lifetime may have been produced decades before. A maintenance activity may require location of the part, photos, and maintenance procedure. For this information to remain accessible for such a timeframe, the information has to be first retrieved from the initial product data management system and then gathered into an archive that will enable the long-term preservation.

Information models are used to precisely describe what content and what metadata are involved in the preservation. The architectural description presented in this section defines the product information models that cover the content and the metadata for this case. The data that conforms to these product information models are called product models. Product models can be particularly complex, so the product information models and formats need to be well defined and documented to guarantee that computers can interpret the data. In particular, it is important to clearly identify the Reference, Provenance, Context, and Fixity information for product models.

The following subsections discuss the use case in detail. Section 4.1 discusses the data set, information flows, and information packaging. Section 4.2 presents the conceptual metadata model used for the architectural description. Section 4.3 describes the data formats used for content information. Section 4.4 describes ingest and access activities.

4.1 Dataset, Information Flow, and Information Packaging. The target of the preservation is product data representing a TWR ship. This data comes from a PDM system called Leading Edge Architecture for Prototyping Systems (LEAPS) [25]. LEAPS was developed by the Navy to be used as a product data repository for any ship. The TWR dataset includes a large amount of product data such as engineering drawings, 3D CAD files, photos, part manuals, or catalogs. The description of a ship involves several types of product data.

This example focuses on the 3D representation and PDM information of a gate valve (see Fig. 12, on the right side), part of the fuel oil system of the ship (see Fig. 12, on the left side).

PDM information includes information such as bill of material, configuration management, document management, product properties, or system breakdown. All of this information needs to be preserved, so it represents content information for the archive.

The objective of this section is to describe the ingest of this dataset. This description should include the steps required to prepare and send the data to the archival system, and the definition of the content and metadata, in particular descriptive metadata [26]. The ingest involves translating some data to formats more suitable for preservation such as open and freely available international standards [27].

The architectural description of the ingest activity is composed of several views. Each view proposes a different aspect of the overall preservation strategy. This subsection describes the information flows related to the use case, the composition of the SIP, the definition of the metadata, and the definition of the ingest activity.

By using this approach, it is possible to formally identify and relate back to the OAIS terminology:

- What systems and what producers send information to the archive
- What content composes the SIP sent to the archive
- What metadata has to be present within the SIP
- What steps are necessary to go from the raw data to the SIP

An initial approach to the ingest of TWR data was proposed in Ref. [26], where the authors discussed the role of descriptive metadata for product model preservation. In this paper, we present a formal representation for the descriptive metadata and the TWR SIP template.

Three distinct *nodes* are identified in this use case: the *Producer*, the *LEAPS* system, and the *Product data archive*. The *Producer* retrieves *LEAPS content* from the *LEAPS* system, creates a *LEAPS SIP*, and sends it to the archive. The OAIS terminology identifies the producer and the archive (see Fig. 13). The stereotypes used are defined in Fig. 6.

Figure 14 provides a conceptual description of the *LEAPS SIP* exchanged in the previous figure. The different files that can compose the SIP are presented, as well as the main characteristics of the content. The PDM information contains the bill of material, system and zone breakdowns, document management, and parts connections. The CAD files contain geometry. Note that PDM information is seen both as content and as metadata. This is because this PDM information includes metadata for the CAD files. The stereotypes used in this diagram are presented in Fig. 4.

This view can serve as ACTDR evidence, as it shows what type of content is being preserved and what information properties are expected from the content.

4.2 Conceptual Metadata Model. The notion of metadata needs a further clarification. Metadata is commonly defined as data about data. Metadata can provide additional information about different things. For example, some metadata applies to documents, while other metadata applies to objects defined within the documents. Within CAD documents, metadata connects geometrical shapes to product concepts. PDM models can be seen as metadata that connects documents, including product data, to product concepts. In both cases, metadata is part of the product data.

All documents have the same core of metadata. The generic metadata needed in this use case is shown in Fig. 15. Each of these

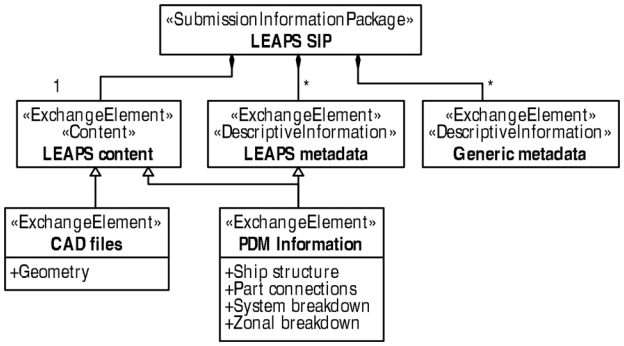

Fig. 14 DIV-1 view of the LEAPS SIP

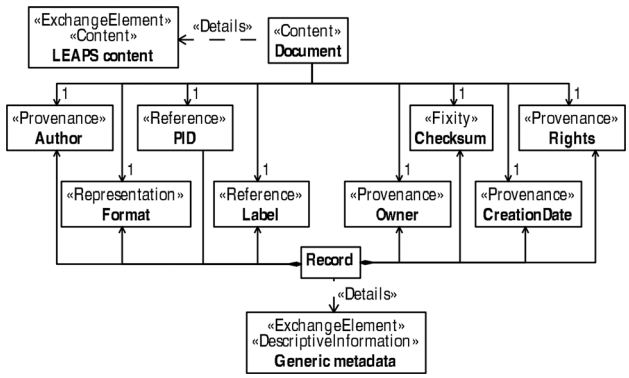

Fig. 15 DIV-1 view of generic product metadata

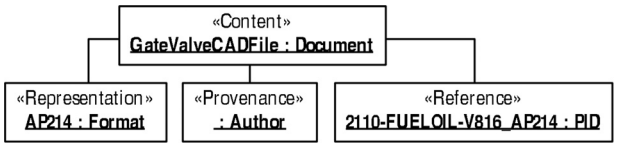

Fig. 16 Generic product metadata for a particular CAD file

types of metadata is categorized according to the OAIS RM. This metadata includes author (*Provenance*), format (*Representation*), persistent identifier (*Reference*), label (*Reference*), creation date (*Provenance*), owner (*Provenance*), rights (*Provenance*), and checksum (*Fixity*). The stereotypes used in this diagram are presented in Fig. 5.

Figure 16 shows how these concepts would be used for the CAD file of the gate valve we are focusing on. The format is declared as AP214 [28] (a CAD format defined in STEP), the author is declared as John Doe, and the identifier is declared as 2110-FUELOIL-V816_AP214.

Product models have particular needs regarding metadata. The information present in the LEAPS file is PDM information, so the information model representing the content is metadata. Figure 17 shows what this metadata comprises. In the LEAPS system, documents can be associated with parts, zonal breakdown, or system breakdown. The metadata should support these concepts, as well as the connections among them. So, additional information includes connections among parts, connections among breakdowns, and connections between parts and breakdowns. Each breakdown has breakdown elements, which can be hierarchically structured using a breakdown structure construct.

Figure 18 shows how these concepts would be used for the PDM information regarding the gate valve. The LEAPS file contains the definitions of the fuel oil system, of the gate valve identified by 2110-FUELOIL-V816, and of a compartment identified by 2110. The fuel oil system relates to a ship system breakdown, and the compartment 2110 relates to a shop compartment breakdown.

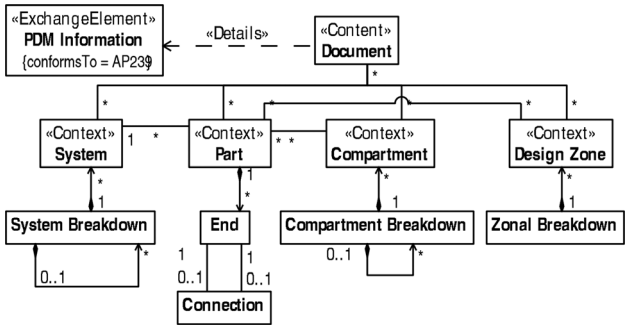

Fig. 17 DIV-1 view of conceptual metadata for LEAPS information

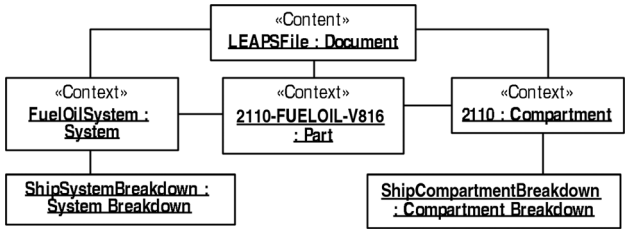

Fig. 18 LEAPS information related to a part

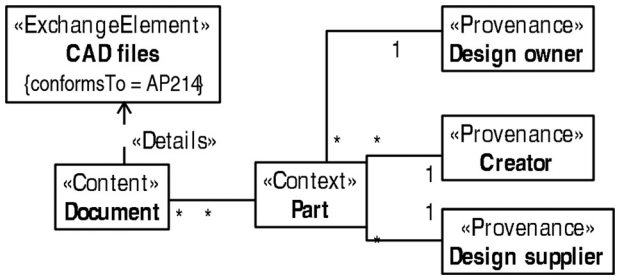

Fig. 19 DIV-1 view of conceptual metadata for CAD files

Figure 19 shows the metadata for the CAD document. The CAD document itself is represented as well as the product it describes. But the CAD file also includes metadata associated with the product such as the design supplier, the design owner, and the creator of the design. This metadata is the provenance information for the content.

Figure 20 shows how these concepts would be used for the CAD file presenting the gate valve. The CAD file relates to the gate valve, which has John Doe as a creator.

The conceptual metadata model has to be further detailed as a logical information model. Logical models take implementation into consideration, but without choosing a particular language. Regarding the metadata presented in this paper, the logical models rely on Dublin Core [29] for generic metadata, and ISO 10303-239 (STEP AP239) [30] for product-specific metadata.

The conceptual models and their associated logical models can serve as ACTDR evidence, as they show the different kinds of preservation description information for the content.

4.3 Formats Used for the Content. Figure 21 shows the standard formats accepted for each type of content. Part of the verification of the SIP is to make sure the files conform to the submission agreement. If a file does not conform to this standard, it will be rejected by the archival system. It is necessary to translate the files into accepted formats.

The choice of formats is crucial to ensure the preservation of information. The United States Library of Congress has proposed

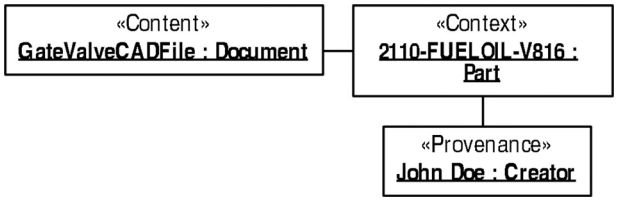

Fig. 20 CAD metadata related to a part

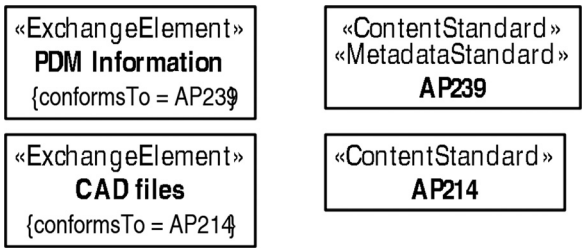

Fig. 21 StdV-1 view of formats used in the SIP

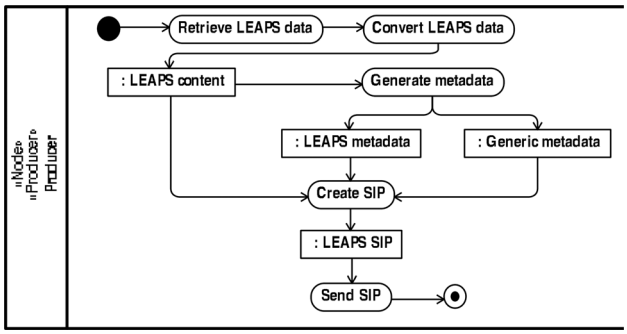

Fig. 22 Ingest activity (OV-5b)

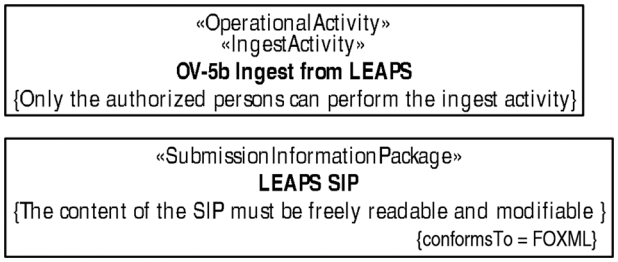

Fig. 23 Ingest and access rules (OV-6a)

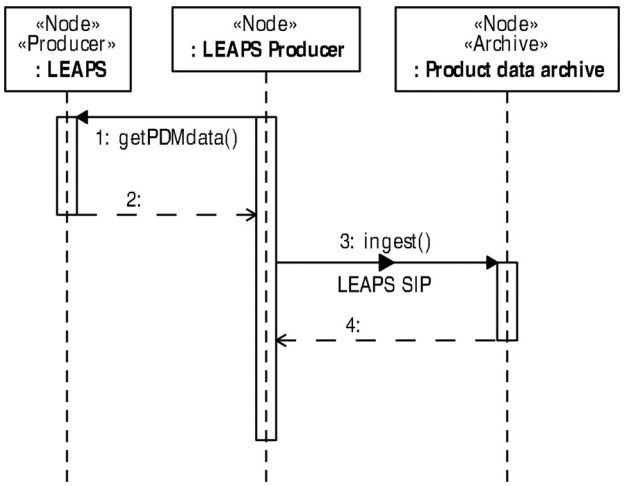

Fig. 24 Sequence diagram of the ingest scenario (OV-6c)

seven sustainability factors for digital formats: disclosure, adoption, transparency, self-documentation, external dependencies, impact of patents, and technical protection mechanism [31]. These factors should be considered for the selection of formats.

CAD files are present in both proprietary and ISO 10303-214 (STEP AP214) [28] files. Following the sustainability factors, the proprietary format generally offers low disclosure (specification not available), low transparency (binary files), low self-documentation (binary files), strong external dependencies (proprietary software), unknown impact of patents, and unknown technical protection mechanisms. The adoption is limited to the user of specific software. STEP files offer high disclosure (open formats), high transparency (ASCII or XML files), high self-documentation (various metadata available), low external dependencies (supported by multiple software), no impact of patents, and no technical protection mechanisms. However, in practice, proprietary CAD formats are still more widely used than STEP formats [32]. From a long-term preservation perspective, it is better to choose STEP formats. In addition to the benefits listed in the previous paragraph, STEP formats have greater longevity, by virtue of being nonproprietary international standards. Over time, STEP data will need to be translated less often, which reduces the cost of preservation.

In addition to STEP formats, product models are also converted into Web Ontology Language (OWL) [33] to supplement STEP. The OWL version of the product models provides two benefits. First, it facilitates the extension of the product information model to use domain-specific terminology, so product knowledge can be easily defined more precisely. Second, it allows semantic connections among product models to be established. These advantages are documented in Ref. [34]. As a result, from a preservation standpoint, OWL provides a stronger basis for improving the understanding and the discovery of product information.

So, CAD files are stored in STEP Application Protocol 214 – Core data for automotive mechanical design processes (AP214) and OWL formats. Similarly, LEAPS information is stored in STEP AP239 and OWL formats.

This view can serve as ACTDR evidence, to show formats are used for the information packages and for the content.

4.4 Ingest and Access Activities Decomposition. The ingest activity is depicted in Fig. 22. The diagram shows the decomposition of the activity into actions. The diagram does not detail how the actions are implemented, but it shows the ingest as a 5-step process:

- Retrieving the LEAPS data. This data is composed of PDM information and CAD files.
- Convert the data to the format suitable for preservation. The PDM information is converted into AP239, to represent the conceptual information presented in Fig. 17. The CAD data are already in STEP, so no conversion is needed. Therefore, an OWL version of both PDM information and CAD files is generated.
- Generating the metadata for the content, as defined in Figs. 17 and 19.
- Creating a SIP for the content, using a packaging information model.
- Sending the SIP to the archive for preservation.

Various business rules may apply during the ingest and access activities. Figure 23 shows an example of these rules. The constraints are as defined in Fig. 10. For the ingest activity, a rule state that only the LEAPS producer is allowed to perform the

activity. A second rule states that the system must have the right to modify the content. Also, the LEAPS SIP should conform to the packaging format Fedora Object XML (FOXML) [35].

The ingest activity involves an information exchange between the different actors. This exchange is depicted in Fig. 24 to show the interaction between the producer, the LEAPS system, and the archival system.

Once the archival system receives the SIP, it performs specific functions to store and manage the content. These functions can be presented as different activity diagrams. The OAIS RM proposes a set of system functions. These functions can be taken and adapted to support the specific content of LEAPS. Some functions are generic, while others are specific to the content. Decomposing the ingest function into multiple diagrams makes it possible to define generic functions, and to provide specific guidance when necessary.

The diagrams representing the system functions were omitted for brevity, more details about systems functions can be found in Ref. [36].

5 Conclusion

This paper discussed an approach to formally describe the information and activities related to archival systems. This approach relies on the DoDAF enterprise architecture framework. The approach uses a preservation terminology inspired by the Reference Model for an Open Archival Information System, to describe the main elements of the archival system. A selection of DoDAF views was also presented to describe various aspects of the archive, and possibly to serve as evidence that the archive meets certification criteria.

This approach was demonstrated by describing a product data ingest. The product data comes from a ship data set. In this use case, the activities and information involved in the ingest were formally defined. The ingest activity was formally described using the OAIS RM terminology, showing the transfer from the original system to the archive. From the information perspective, the content and the metadata were defined.

This approach is generic enough to be used in other domains besides ship product data. It can lead to the definition of a comprehensive reference architecture for archives, which provides a way to formally describe archive architectures. However, the proposed approach does not include all the aspects that must be considered for the development of a complete standalone archival system. Risk management, security, and many other organizational aspects have not been covered in this work.

Acknowledgment

No approval or endorsement of any commercial product by NIST is intended or implied. Certain commercial software may be identified in this report to facilitate better understanding. Such identification does not imply recommendations or endorsement by NIST nor does it imply the software identified are necessarily the best available for the purpose.

References

[1] Beagrie, N., 2006, "Digital Curation for Science, Digital Libraries, and Individuals," Int. J. Digit. Curation, **1**(1), pp. 3–16.
[2] Kuipers, T., and Hoeven, J., 2009, "Insight into Digital Preservation of Research Output in Europe," Survey Report.
[3] Blue Ribbon Task Force, 2010, "Sustainable Economics for a Digital Planet: Ensuring Long-Term Access to Digital Information," Final Report of the Blue Ribbon Task Force.
[4] International Organization for Standardization, 2012, Space Data and Information Transfer Systems – Open Archival Information System (OAIS) – Reference Model ISO.
[5] British Standards Institute, LOTAR (Long Term Archiving and Retrieval of Digital Technical Product Documentation Such as 3D, CAD, and PDM Data).
[6] Verband der Automobilindustrie e. V., 2005, "VDA 4958: Long-Term Archiving of Digital Product Data, Which are not Based on Technical Drawings," Technical Report.
[7] Consultative Committee for Space Data Systems, 2012, "Audit and Certification of Trustworthy Digital Repositories," International Organization for Standardization.
[8] Bernard, S., 2005, *An Introduction To Enterprise Architecture*, AuthorHouse, Bloomington, IN.
[9] Group, T. O., ed., 2011, *TOGAF Version 9.1, Enterprise Edition*, Van Haren Publishing, Zaltbommel, Netherlands.
[10] Ministry of Defence, MOD Architecture Framework (MODAF) 1.2.
[11] Department of Defense, DOD Architecture Framework (DoDAF) 2.0.
[12] IFIP-IFAC Task Force, 1993, "IFIP-IFAC Task Force on Architectures for Integrating Manufacturing Activities and Enterprises," IFIP/IFAC Newsletter.
[13] Bernus, P., Nemes, L., and Schmidt, G., 2003, *Handbook of Enterprise Architecture* (IFIP-IFAC Task Force, GERAM: The Generalized Enterprise Reference Architecture and Methodology), Springer-Verlag, Berlin, Germany, pp. 40–82.
[14] Williams, T., Bernus, P., Brosvic, J., Chen, D., Doumeingts, G., Nemes, L., Nevins, J., Vallespir, B., Vlietstra, J., and Zoetekouw, D., 1994, "Architectures for Integrating Manufacturing Activities and Enterprises," Comput. Ind., **24**(2–3), pp. 111–139.
[15] International Organization for Standardization, 2000, ISO/IS 15704:2000, Industrial Automation Systems Requirements for Enterprise Reference Architectures and Methodologies.
[16] Williams, T., 1994, "The Purdue Enterprise Reference Architecture," Comput. Ind., **24**(2–3), pp. 141–158.
[17] Chin, K.-S., Lam, J., Chan, J. S. F., Poon, K. K., and Yang, J., 2005, "A CIMOSA Presentation of an Integrated Product Design Review Framework," Int. J. Comput. Integr. Manuf., **84**(4), pp. 260–278.
[18] Doumeingts, G., Chen, D., Vallespir, B., Fenie, P., and Marcotte, F., 1993, "GIM (GRAI Integrated Methodology) and its Evolutions—A Methodology to Design and Specify Advanced Manufacturing Systems," Proceedings of the JSPE/IFIP TC5/WG5. 3 Workshop on the Design of Information Infrastructure Systems for Manufacturing, North-Holland Publishing Co., Amsterdam, Netherlands, pp. 101–120.
[19] Becker, C., Antunes, G., Barateiro, J., Vieira, R., and Borbinha, J., 2011, "Modeling Digital Preservation Capabilities in Enterprise Architecture," 12th Annual International Conference on Digital Government Research, College Park, MD, June 12–15.
[20] Object Management Group, 2005, Unified Modeling Language (UML) 2.0.
[21] Object Management Group, Unified Profile for the Department of Defense Architecture Framework (DoDAF) and the Ministry of Defence Architecture Framework (MODAF).
[22] Gogolla, M., and Henderson-Sellers, B., 2002, "Analysis of UML Stereotypes Within the UML Metamodel," The Unified Modeling Language, pp. 63–81.
[23] International Organization for Standardization, 1994, Industrial automation Systems and Integration—Product Data Representation and Exchange—Part 1: Overview and Fundamental Principles.
[24] World Wide Web Consortium, 2008, Extensible markup language (XML) 1.0.
[25] Naval Sea Systems Command, Leading Edge Architecture for Prototyping Systems.
[26] Lubell, J., Kassel, B., and Rachuri, S., 2009, "Descriptive Metadata Requirements for Long-Term Archival of Digital Product Models," Proceedings of the Indo-US Workshop on International Trends in Digital Preservation.
[27] Kassel, B., and David, P., 2007, "Long Term Retention of Product Model Data," J. Ship Prod., **23**(2), pp. 118–124.
[28] International Organization for Standardization, 2003, ISO 10303-214:2003 Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 214: Application Protocol: Core Data for Automotive Mechanical Design Processes.
[29] Initiative, D. C. M., 2012, Dublin Core Metadata Element Set, Version 1.1.
[30] International Organization for Standardization, 2012, ISO 10303-239:2012. Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 239: Application Protocol: Product Life Cycle Support.
[31] Arms, C., and Fleischhauer, C., 2005, "Digital Formats: Factors for Sustainability, Functionality, and Quality," IS&T Archiving Conference, Society for Imaging Science and Technology, Washington, DC, pp. 26–29.
[32] LongView Advisors, 2008, Collaboration & Interoperability Market Report.
[33] World Wide Web Consortium, 2012, OWL 2 Web Ontology Language.
[34] Barbau, R., Krima, S., Rachuri, S., Narayanan, A., Fiorentini, X., Foufou, S., and Sriram, R., 2012, "OntoSTEP: Enriching Product Model Data Using Ontologies," Comput.-Aided Des., **44**(6), pp. 575–590.
[35] Davis, D., 2011, Fedora Object XML (FOXML).
[36] Barbau, R., 2013, "A Reference Architecture for Archival Systems With Application to Product Models," Ph.D. thesis, Université de Bourgogne, Dijon, France.