# Preserving records in the cloud

Jan Askhoj, Shigeo Sugimoto and Mitsuharu Nagamori
*Graduate School of Library, Information and Media Studies, University of Tsukuba, Ibaraki, Japan*



## Abstract

**Purpose** – The purpose of this paper is to examine the characteristics of managing records in a cloud computing environment and compare these with existing archiving models, exemplified by the open archival information system (OAIS) reference model.

**Design/methodology/approach** – The authors compare the functional entities in OAIS with a layered model of cloud computing, in which services are abstracted and shared between layers.

**Findings** – It is concluded that there are a number of areas where OAIS does not integrate well with cloud computing systems. Based on the findings, a new layered model for a cloud archiving system is defined using the concepts and information types from the OAIS reference model. The proposed model allows the sharing of functionality and information objects by making them available as services to higher layers. The model covers the entire document lifecycle, making archive functionality such as preservation planning possible at an early stage and helping to simplify records transfer.

**Research limitations/implications** – The model provides a simple, OAIS compatible approach to representing how digital objects and necessary metadata can be transferred from content creation systems to archives systems.

**Originality/value** – Whereas a lot of research has been done on the technical aspects of cloud storage, there is a lack of focus on how to comprehensively integrate records transfer and preservation in cloud systems. This paper fills in some of the gaps.

**Keywords** Electronic records management, Archiving, Systems design, Information systems, Digital storage, Computer applications, Document management

**Paper type** Research paper

## 1. Introduction

A shift has started in the computing landscape, where cloud computing is becoming an increasingly popular choice for many organizations. IDC predicts that by 2020 as much as 15 percent of the information in the Digital Universe could be part of a cloud service. Whereas much of this information may not be considered preservation worthy, an increasing amount of business and administrative records are being stored in the cloud (Gantz and Reinsel, 2010).

In cloud computing, data, software and sometimes the entire technical infrastructure of an organization is moved from corporate data-centers to Cloud Service Providers. This transfer of service and responsibility to one or more third parties can make it hard to guarantee the reliable storage and preservation of records. Changes in available services, difficulty in emulation and migration, inadequate preservation and so on may result in the loss of information (Jain and Bhardwaj, 2010). Existing archive models, such as OAIS (the Open Archival Information System Reference Model developed by the Consultative Committee for Space Data Systems is an extensively used reference model for archiving systems) have been established

The study presented in this paper was in part supported by Grant-in-Aid for Scientific Research.

Records Management Journal
Vol. 21 No. 3, 2011
pp. 175-187

0956-5698
DOI 10.1108/09565691111186858

specifically to deal with archiving and preservation, but they may be inadequate or hard to apply when it comes to the cloud. It is necessary to discover now if existing archive models are applicable in a cloud-computing environment or if new solutions are needed before it is too late (CCSDS Secretariat, 2002).

## 2 Background

### *2.1 Definition of cloud computing*

Cloud computing is a relatively new and somewhat "hyped" concept. This means that there is not yet a single, agreed definition. Indeed, Vaquero *et al.* (2008) identify several co-existing definitions. As a definition for the purposes of this paper, we have come up with four characteristics of cloud computing:

(1) cloud computing is an abstracted, scalable platform for service delivery;

(2) cloud computing makes use of existing technologies that can be described via a layered model;

(3) access to both platform and services is available via the internet on a pay-per-use basis; and

(4) the availability, quality and number of services are offered according to agreements with a Cloud Service Provider.

### *2.2 Characteristics of archiving in the cloud*

A common way of distinguishing between different cloud services is by dividing them into layers, such as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) (Lenk, 2009). In a layered model, each layer builds on services offered by the layer below, and in turn offers services to the layer above. Each layer uses its own information types (data classes and properties) to provide specific functionality. A well-known example of a layered model is the Open System Interconnection reference model for network communication (Wetteroth, 2001). The benefit of a layered model is that once data types and services are defined, layers can be abstracted. This means that when considering any specific layer it is possible to disregard the inner workings of the layer below (Youseff *et al.*, 2009).

It is equally possible to apply a layered model to archiving systems. This can be illustrated by a simple two-layered model for an archive framework (Figure 1). The bottom "Platform Layer" represents a trusted digital cloud repository offering guaranteed Digital Object integrity at the bit-level. The top "Software Layer" represents applications such as office software using the repository below. In this scenario, applications are able to save Digital Objects directly to the repository, trusting that secure storage is provided (Sugimoto, 2007).

Because of their scalability and networked nature, cloud services can also be easily shared between a number of systems. For example, the records producing institution may share storage with the archive it is submitting records to. This would bring benefits to both parties by reducing the need to duplicate services.

## 3. Objective

The ability to abstract and share archiving services opens up exciting opportunities for archiving and records management system design, such as the sharing of computing

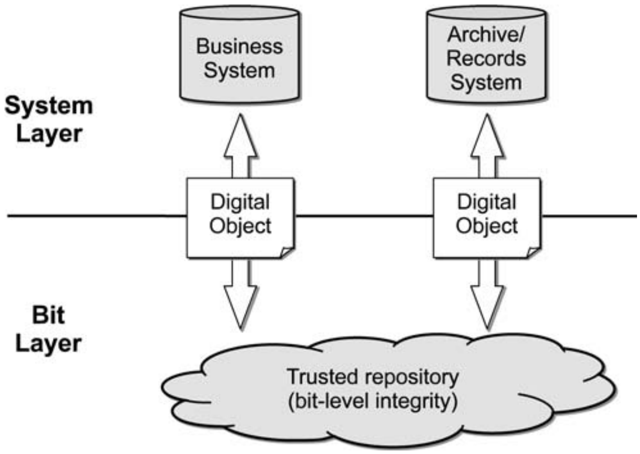


**Figure 1.**
Simple layered model for an archive system

resources between producers of information and archives and better coordination of record submission. Our objective is to make it possible for organizations to enjoy the benefits of moving to cloud computing, while at the same time ensuring that important records are properly stored and preserved. We therefore set out to examine the characteristics of cloud computing from an archiving perspective and examine any differences with the OAIS model. Based on our findings we want to create an overall model for a cloud computing archiving system, including information types, classes and properties. Finally, using the new model, we plan to show how archiving services can be delivered via the cloud, using an example where both records creation and archiving takes place in the cloud and records are transferred automatically. It is our goal to use the proposed model as the base for the development of a dedicated cloud archive.

## 4. Mapping the OAIS model to layered services

It would be ideal if the OAIS model with its Functional Entities could be applied directly to a cloud environment, where services can be shared and abstracted in layers and where services such as storage and data management can be outsourced to a third-party and paid for on-the-fly. There are, however, some areas in the OAIS that make integration with cloud computing difficult.

- The fact that the functional entities in OAIS are interdependent, makes it is difficult to transfer responsibility for parts of an OAIS archive to an external service provider. In other words, it may be difficult to get an external party to provide the exact functionality specified by OAIS for a functional entity or to make sure that any functionality not provided by the external provider is covered elsewhere. For example, if an organization is looking for a storage solution offering bit-level integrity for Digital Objects to use as a back-end for an archiving system, this would involve overlapping functionality from the Management, Data Archiving and Archival Storage entities.

- In OAIS, the burden of creating Submission Information Packages (SIP) is left to producers, who must meet the requirements of the OAIS archive. This task can be very resource intensive, depending on the strictness of the requirements and how many archives the producer submits to. This can lead to producers holding on to records for long periods, before submitting them in bulk. This can significantly delay preservation planning. With cloud computing and the shared platform it offers, Digital Objects can be made accessible to an archive without delay, allowing early preservation planning.
- With cloud computing, the need to include Digital Objects and metadata in information packages disappears. With a shared, trusted platform, producers only need to provide the information (URI or similar) of where the Digital Objects are stored. However, the OAIS Model does not specify the requirements and functionality of such a shared platform.
- The OAIS doesn't cover the initial stages of the Document Lifecycle (the Create, Use, Manage stages). It can be argued that these stages lie outside the scope of an archive. However, the nature of the events in these stages and how well they are documented can have a huge impact on how easy it will be to carry out preservation work later on.

## 5. Proposed layered model for archiving services

The basic information types in OAIS can be arranged according to information complexity. There is an increase in complexity from the relatively simple Digital Object to the more comprehensive information package. This progression from simple to complex is comparable to how information flows in a layered model, where information in one layer is used, manipulated and passed to a higher layer. We believe that a model showing the development in complexity of information objects corresponds to the document lifecycle, where a document goes through a number of stages over time (Jia-sun, 2001). We have used this progression as the base of our model.

The simple two-layered model in Figure 1 shows that an archive system can be broken down to three basic entities. A repository for Digital Objects, the objects themselves and systems used to access and manage the objects. We have expanded on this model in two ways. First, since preservation is more than just ensuring safe storage, we have added a Preservation Layer that provides the information types necessary for long-term preservation and creates information packages for archive systems. Second, we have added a top layer called the Interaction Layer. This layer contains the different systems used to manage documents and records over time. Figure 2 shows our proposed layered model.

We have chosen to make the PaaS layer the lowest layer of our model. If a trusted, long-term repository for Digital Objects can be guaranteed, and if that platform can be migrated to another environment, there would be no need for archivists to be concerned with what infrastructure (IaaS) lies below that repository. This is similar to how organizations use PaaS for their business systems and leave infrastructure services to a provider. In the PaaS layer objects do not carry meaning, they are simple bit-strings and can be of any type or format.

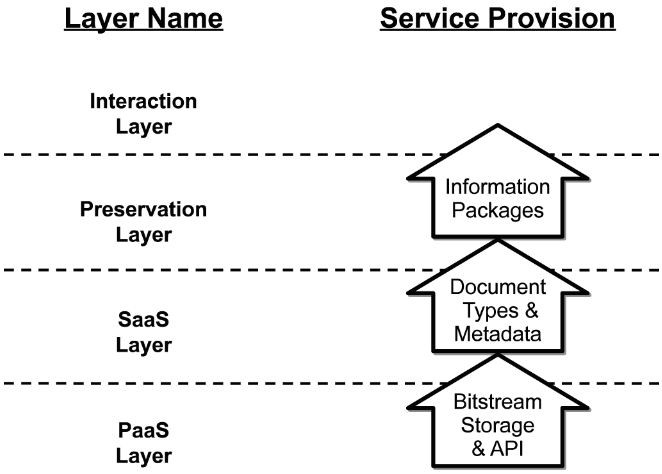


**Figure 2.**
Layered model

The next layer is the SaaS Layer. The creators and users of content do not access bit level data directly. They use documents; tables and other represented information types, referred to as Digital Objects in the OAIS Model. The functionality to create and present Digital Objects in manageable form is provided by business software in the SaaS Layer. Examples of business software are online office software and e-mail clients.

The Preservation Layer is responsible for receiving Digital Objects from SaaS systems and turning them into information packages that can be managed by archive systems. Packages need to contain information necessary for preservation, i.e. Representation Information, Preservation Description Information etc. Apart from creating packages, the Preservation Layer is responsible for the registration of the information used in package creation, such as information about business systems, metadata schema registers and crosswalks. Finally, the Preservation Layer generates reports to the Interaction Layer.

The Interaction Layer is the layer where users or external systems access cloud systems to create, manage or archive records. In this layer are the front-end systems for Content creation, using a browser or other client for accessing cloud systems and Archives and Records Management Systems.

In the second column of the model, we have specified a number of Service Provisions. These contain information on how two layers interact and descriptions of the services provided by one layer to the layer above. (See Chapter 8, Information Classes and Properties) about the services provided.

The definition of services also forms the base of a migration path. If one layer of the cloud solution fails or if the systems in a layer need to be migrated, a clear migration path must be specified. By defining services and information types required for any one layer, specification of requirements and facilitation of migration becomes possible.

We believe that our model is helpful for a number of reasons. First, it corresponds to both the layered nature of cloud systems and to the increasingly complex OAIS information types as the document lifecycle progresses. Second, it allows sharing of functionality and information types by dividing these into layers and providing them as services. Finally, it allows migration of any one layer by defining functionality and information types.

## 6. Information classes and properties

### *6.1 OAIS information types*

Using the proposed layered model for a cloud archiving system, the next step is to define the information and services provided by each layer. As explained earlier, we believe that it is difficult to apply the OAIS Model to a cloud environment with layers of service. However, we have found that the OAIS Information Types themselves can be successfully mapped to our model. Figure 3 shows the model with corresponding information types.

Mapping the OAIS information types to the layered model is an important step in defining the information flow between layers, and by extension in defining what functionality must be provided by each layer.

### *6.2 Breakdown of information types*

In the following section, we present an overview of the different information types, according to layer.

*6.2.1 PAAS layer. Bit-strings.* The PaaS layer provides common storage for archive and business systems. All objects in the PaaS Layer are simple strings of bits. They are primitive units that make no sense as information outside the context of a system that can read and represent them. The objects can be parts of the Digital Objects that are

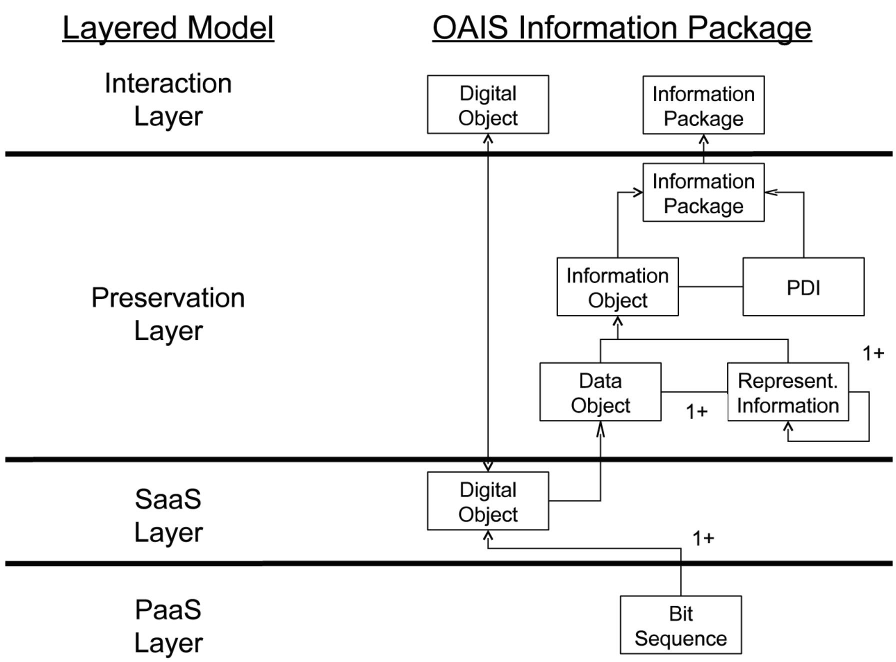


**Figure 3.**
Information types and layers

the target for archiving, parts of metadata or system configuration data. The integrity of the data is guaranteed by the layer, preventing media failure, update errors or "bit decay". One important aspect of data in the PaaS layer is that it can be presented to higher layers as a coherent unit and be referred to by a single, permanent URI. These requirements are necessary because higher layers rely on the PaaS layer to provide long-term access to objects, even if these should change due to factors such as migration.

*6.2.2 SAAS layer*. In the SaaS Layer are the server-side systems that turn bit-strings into information objects that are understandable by and can be manipulated by users. An example of such a system is Google Docs that sits on top of a platform and storage back-end hosted by Google.

*Digital objects (DO).* We refer to these information objects as Digital Objects. For example, in a SaaS word processing application, a DO can be a document that consists of text, formatting data, representation data etc. If a DO is declared as a record, it needs to be made available to layers above it in an agreed format that is accepted by these layers. In other words, if the software in the SaaS Layer is proprietary or uses proprietary formats (which is often the case with existing SaaS solutions), there must be functionality to convert the DO into a format that can be understood by higher layers both now and in the future, without having to rely on functionality in the SaaS Layer

*6.2.3 Preservation layer*. The Preservation Layer contains functionality to manipulate and add information to Digital Objects, turning them into information packages. The role of this layer is basically to form a bridge between the archive systems in the Interaction Layer and SaaS systems.

Digital Objects with added Representation Information are known as Data Objects.

*Representation information (RI).* The purpose of Representation Information is to ensure the understandability of Digital Objects to a Designated Community. Both structure information and semantic information about Digital Objects are necessary. RI may need to be referenced by other RI. For example a text document with RI saying that text is in an ASCII table, may further includes a definition of ASCII.

We imagine this to take the form of a Representation Information Registry (RIR), i.e. "...a systematic collection of representation information objects or locatable references to objects held elsewhere." (Giaretta, 2005).

Using a RIR in a cloud based system carries with it two advantages. First, it becomes possible to specify RI for Digital Object types and store this in lower layers. This allows systems managing RI to simply reference stored RI instead of providing it for each Digital Object. Second, a RIR takes the burden of providing representation information away from individual systems in the Business Layer, simplifying records submission. In order for this to work, the types of DO that can be declared records must be registered in the RIR in advance.

*Preservation description information (PDI).* OAIS specifies four types of Preservation Description Information needed for adequate preservation:

(1) Reference Information is a unique identifier that allows systems to refer to a particular Data Object. It can be a Record ID Serial Number or similar. In the SaaS Layer, a URI is assigned to identify Digital Objects. This URI follows DIs in the higher layers, serving as Reference Information.

(2) Provenance information documents the history of a DO and changes during its lifecycle up to and including archival. Depending on content there can be different Provenance information.
(3) Context information describes the relationship between a DO and its environment. Like Provenance Information, it needs to be provided by the SaaS Layer in a usable format.
(4) Fixity information is added to Data Objects to ensure that these do not change, due to system errors or tampering.

*Packaging information.* Information packages are central to the OAIS Model. Archives ingest, store and disseminate records as different types of packages. The concept of packages still exists in the layered model, but the content of packages is different. Because functionality and information objects provided by one layer can be used by one or more entities in higher layers, the need to include all the information mentioned above is no longer necessary. Instead, a URI pointing to the object in question is enough. This means that Packaging Information therefore becomes a container for URIs, pointing to information in the PaaS Layers. Using URIs rather than transferring the objects themselves means that packages can be made much smaller, reducing data transfer overhead.

*Package description.* The package description provides searchable information about an information package and makes it available to access aids. This information is normally provided at the time of dissemination, where a user may use it to locate a specific package. In such cases, Package Description depends on the intended audience, and becomes difficult to provide. In the Preservation Layer, a Package Description can be provided as a generic service to help access by archiving systems in the above layer.

*6.2.4 Interaction layer*. This layer holds user-facing systems used to directly manipulate information objects. We have divided these into two categories:

(1) front-end systems for content creation; and
(2) archive systems.

The front end systems (browsers etc.) are used by users to create and manage Digital Objects. Since systems for content creation do not make use of preservation related functionality from the Preservation Layer, Digital Objects in the SaaS Layer are accessed directly as shown in Figure 3.

Archive systems in the Interaction Layer access standardized information packages from the Preservation Layer. It is important to note that while archive systems in the Interaction Layer can access and read information referred to in Information Packages (Digital Objects, RI, PDI and so on), they cannot make changes or overwrite this information. When additional information is needed (such as an Associated Description and Archive Administrative Data) or if the archive requires information in another format, archives are responsible for saving and maintaining this information themselves.

The interaction between the Business and Preservation Layers corresponds to the way that one OAIS may take the role of Producer to another OAIS. The packages provided by the Preservation Layer are in effect complete Archival Information Packages (AIP), however they act as SIPs to archive systems in our model.

## 7. Model information flow

### *7.1 Provision of preservation metadata*

We regard the provision of adequate preservation metadata for the creation of information packages as one of the biggest challenges, when implementing our proposed model. In other words, the Preservation Layer must provide sufficient preservation metadata to archive systems. If this cannot be done, then archives are burdened with the task of making up or complementing for missing metadata. This can be resource intensive and sometimes an impossible task (Gartner, 2004).

We have set out to examine how sufficient metadata could be provided by the services in our model. To represent "metadata necessary for long-term preservation", we decided to use the required metadata elements, as defined in the PREMIS data dictionary (OCLC/RLG, 2002). Whereas PREMIS does not cover all aspects of preservation metadata, it defines the information most preservation repositories need to guarantee long-term preservation of digital materials (Caplan, 2009).

Initially, we have divided the mandatory elements of PREMIS into functional categories to fit the functionality in our layered model. Using this approach, we found that all PREMIS elements fit into one of three categories:

(1) Metadata generated by business systems at the time of document creation or record declaration: as a minimum, the descriptive metadata for preservation that cannot be supplied elsewhere must be provided by the business systems.

(2) Information that can be pre-registered: static information provided in advance by systems in the cloud model. That is, information about registered system entities and information types (representation information, metadata schemas, crosswalks etc.).

(3) Event Related Information: information describing changes to Digital Objects and metadata taking place during the preservation process.

That PREMIS metadata elements can be divided into the above categories in turn suggests what functionality must be provided by the Preservation Layer: management of business systems objects and metadata (import modules, validation and formatting tools, cross-walking tools, etc.); registration of information about business systems (simple representation information, metadata schema information, technical system information, etc.), and documentation of changes to Digital Objects and metadata occurring during the package creation process.

Based on this breakdown it becomes possible to define requirements for a cloud service that can provide sufficient preservation metadata, provided the requirements above for information and functionality are met.

### *7.2 Information flow example*

To illustrate how our model may be applied in the real world, we have created a scenario presenting a cloud system using our model. In the scenario we show how an e-mail with an attachment created in a SaaS System passes through the Preservation Layer and is stored in an archive system.

First a presentation of the entities in the cloud system. The PaaS Layer providing the execution environment with data storage support consists of a number of virtual images on top of Amazon S3 or similar XEN based hyper-visor (Amazon, 2011). Higher Layers access the PaaS Layer via the Simple Cloud API, using a REST interface to call application services (Zend, 2009).

Business software in the SaaS Layer in this case consists of an online office software suite. SaaS software has been registered in the Presentation layer in advance, with information on system type, digital object types, metadata schemas and formats, using a machine readable format, such as XML.

The Preservation Layer contains a Packager for the creation of information packages. This has three types of functionality:

(1) A Registration module holding information on entities in the SaaS Layer;

(2) an Event Documentation Module to document changes in Digital Objects and Metadata during the packaging process; and

(3) Data Management functionality for the actual creation of packages.

The Archive Systems in the Interaction Layer have been registered with basic information in the Preservation Layer, ensuring that only authorized systems can access packages. Archive systems receive complete virtual information packages directly from the Preservation Layer, but can generate and save additional metadata to the PaaS layer as required.

In the above system, a typical information flow would be as follows:

- A user belonging to an organization creates an e-mail, using an online e-mail client (SaaS Business System) accessed via a browser (User Facing Business System). The user attaches a HTML document to the e-mail and sends it to a recipient.
- Based on organizational policy, the e-mail is declared a record in the business system. The bit-strings making up the e-mail and its attachment are locked in the PaaS layer. An XML notification of the bit-strings making up the e-mail is sent to the Packager.
- Based on the notification, the Packager retrieves the relevant data from the PaaS layer. After the data is validated, parts may need to be converted (for example, metadata may need to be cross-walked to a different schema). Based on the converted data, an XML file of PREMIS Preservation metadata is created based on Business System Metadata, Pre-registered information and Event Related Information. This information and the corresponding Data Object is saved in the the PaaS layer, as a virtual package. An notification pointing to the relevant bit-strings is sent to the Interaction Layer.
- Archive systems in the Interaction Layer receive packages from the Preservation Layer. Based on archive policies, archive systems may also save additional metadata, such as Access Aid specific data.

## 8. Model evaluation

Our layered model serves as a common, conceptual frame of reference for archives and cloud providers when planning the storage and preservation of records in the cloud. The information flow in Figure 4 helps illustrate a number of benefits of applying a layered model when constructing a cloud archive. The biggest benefit of these is that the model makes it possible to provide preservation metadata, storage, packaging etc. as services. Instead of defining these functions as entities inside the remits of an OAIS archive, our model allows for abstraction and sharing. For example, a Cloud Service Provider could use the model to demonstrate that its services have the functionality of a level 1 and 2 (PaaS and SaaS) system for archiving purposes. Archives could build on such a description of functionality and provide the required level 3 and 4 (Preservation and Interaction) functionality themselves, or from yet another provider). In other words, providing such OAIS services outside an archive can help to simplify system design, by making it possible to rely on shared services from Cloud Service Providers. However, the parties involved in such sharing must have common knowledge of the resources they are sharing and of each other. This is where the proposed model becomes especially useful.

Another benefit comes from having the business system and archive sharing the same PaaS platform. In the example above, this makes it possible to start automatically providing preservation metadata as soon as the e-mail is declared a record. In the OAIS, the burden of providing the necessary SIP metadata is with the Producer, and if SIPs are sent to more than one archive, different metadata may need to be supplied to

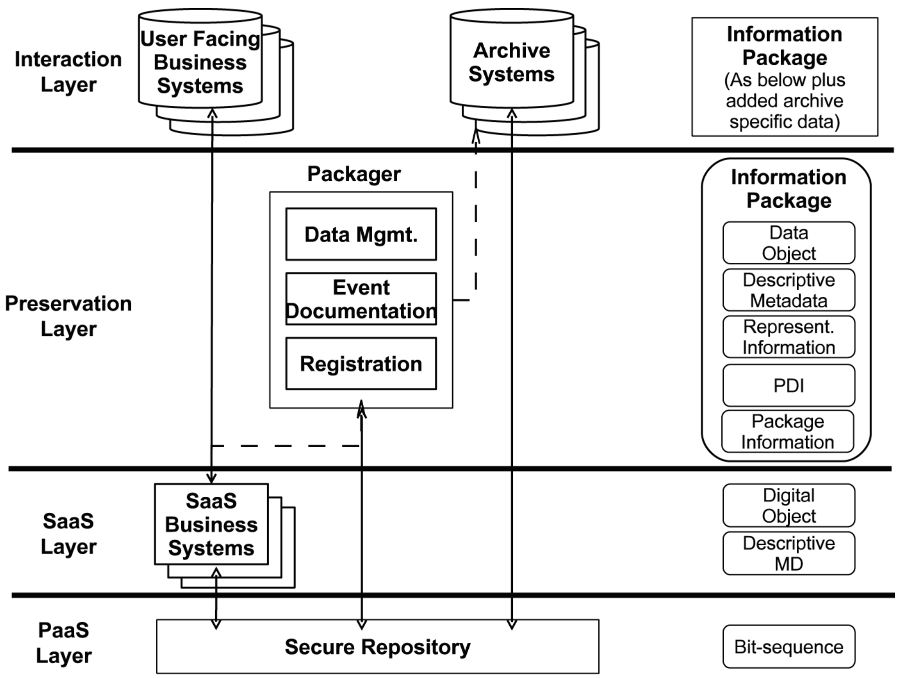


**Figure 4.**
Information flow

each archive. Using our model, a Provider only needs to provide one type of metadata, as long as it meets the requirements of the preservation service.

When examining the information flow in Figure 4, it becomes clear that there are a number of conditions that must be met to ensure successful implementation. First of all, no matter what cloud platform is used, it must be truly trustworthy, with no change or loss of service. While no system is 100 percent secure, not having data in-house makes this point doubly important. Second, the SaaS systems used in the cloud system must be able to export content and metadata in usable formats and of sufficient quality to be useful for preservation.

## 9. Concluding remarks and future work

In this paper, we have presented a layered model for archiving in a cloud computing environment. We believe that the strength of our model is that it builds on existing OAIS concepts, while being compatible with cloud computing systems. With our model, it becomes possible to abstract services, such as the provision of a trusted repository and preservation metadata. The model also shows how these services can be easily shared in a way that makes it simpler to construct archiving systems, by making it possible to rely on services from lower layers.

We believe the proposed model serves as a good starting point for the development of new archive system frameworks. It is our plan to apply the proposed model to construct an actual cloud archiving system, and work has already started on the development of specific data classes and properties. We hope that by building a cloud archive system using the concepts in our model we can demonstrate benefits to both records producers by the automatic provision of preservation metadata and to archive system managers by providing a reliable, trusted repository and comprehensive, standardized information packages.

## References

Amazon (2011), *Amazon Simple Storage Service (Amazon S3)*, available at: http://aws.amazon.com/s3/ (accessed 2011).

Caplan, P. (2009), *Understanding PREMIS*, available at: www.loc.gov/standards/premis/bibliography.html (accessed November 17, 2010).

CCSDS Secretariat (2002), *Reference Model for an Open Archival Information System (OAIS)*, available at: public.ccsds.org/publications/archive/650x0b1.PDF (accessed November 2, 2011).

Gantz, J. and Reinsel, D. (2010), *The Digital Universe Decade – Are You Ready?*, IDC iView, available at: http://idcdocserv.com/925 (accessed 2011).

Gartner, R. (2004), "PREMIS—Preservation metadata implementation strategies update 2: core elements for metadata to support digital preservation", *RLG Diginews*, Vol. 8 No. 6.

Giaretta, D. (2005), "Representation information for interoperability now and with the future", *Proceedings of the MSST 2005. Local to Global Data Interoperability - Challenges and Technologies, 2005, Sardinina, Italy, 20-24 June*, pp. 42-46.

Jain, L. and Bhardwaj, S. (2010), "Enterprise cloud computing: key considerations for adoption", *International Journal of Engineering*, Vol. 2 No. 2, pp. 113-7.

Jia-sun, H.E. (2001), "Life cycle of electronic records", *Journal of Zhejiang University (Humanities and Social Sciences)*, p. 4.

Lenk, A. (2009), "What's inside the cloud? An architectural map of the cloud landscape, *Proceedings of the Software Engineering Challenges of Cloud Computing, 2009. CLOUD'09. ICSE Workshop*, pp. 23-31.

Sugimoto, S. (2007), *Ensuring the Preservation and Use of Electronic Records*, available at: www.archives.go.jp/english/news/pdf/sugimoto.pdf (accessed 2011).

The OCLC/RLG Working Group on Preservation Metadata (2002), *Preservation Metadata and the OAIS Information Model - A Metadata Framework to Support the Preservation of Digital Objects*, available at: www.oclc.org/research/activities/past/orprojects/pmwg/pm_framework.pdf (accessed 2011).

Vaquero, L.M. (2008), "A break in the clouds: towards a cloud definition", *ACM SIGCOMM Computer Communication Review*, Vol. 39 No. 1, pp. 50-5.

Wetteroth, D. (2001), *OSI Reference Model for Telecommunications*, McGraw-Hill Professional, New York, NY.

Youseff, L., Butrico, M. and Da Silva, D. (2009), "Toward a unified ontology of cloud computing". paper presented at the Grid Computing Environments Workshop, 2008, GCE08, pp. 1-10.

Zend (2009), *Simple Cloud API Reference*, available at: http://simplecloud.org/api (accessed 2011).

**Corresponding author**
Jan Askhoj can be contacted at: janaskhoej@gmail.com