Chapter 4. MadPK: An Integrative Architecture for Digital Preservation

4.1. Introduction

In this chapter, we construct an architecture to connect the KM approach and the digital preservation approach in a dynamical way. We propose a multi-layer architecture under the concept of web services. Thus, the interoperabilities of the business processes in various approaches are ensured. Besides, the interoperabilities of the knowledge objects (digital objects) are also ensured.

4.2. Proposed Architecture for Digital Preservation

Our research work includes the KM approach as well as the digital preservation approach. The KM approach identifies enterprise knowledge and produces digital knowledge objects. The digital preservation platform performs the whole digital preservation process. These two parts of long term knowledge preservation are both necessary and important. However, the KM environments are complex and diversified in enterprises, and compared to them, the digital preservation environment lasts longer than KM. Therefore, in order to achieve dynamic preservation to connect the two major parts, we need to develop an architecture for long term knowledge preservation. This architecture should allow the two parts to perform smoothly in their own environments, and allow the communication and connections between them to be accessible and dynamic. As we have introduced in Chapter 2, this architecture appear as the Architecture Model of our extended CommanKADS methodology.
4.2.1. Multi-layer Architecture for Dynamic Preservation of Knowledge – MadPK

The KM approach must selectively acquire comprehensive knowledge for long term preservation. On one hand, we need to adapt the ways of KM and develop a methodology, which is based on KM methodologies, for long term knowledge preservation. On the other hand, digital preservation needs a platform for the repository of knowledge (i.e. digital content). Digital preservation has correlations with KM, but presents a very different context compared to data mining. In order to achieve dynamic preservation and interoperability between knowledge sources and digital preservation platforms, we intend to construct a universal architecture for long term digital preservation.

There are plenty of projects in Europe which work on digital preservation. In the researches and projects of digital preservation, the KM approach is not usually integrated. Meanwhile, although the long term knowledge preservation research issues have been noticed by researchers after the year 2000, the existing researches and projects on long term preservation focus on specific field (mainly on production field). In their approaches, the confinement of “knowledge” narrows to product’s engineering data (i.e. 3D, CAD data, etc.). The core work of these existing long term preservation researches and projects is the development of standards or approaches (e.g. STEP, OntoSTEP [Krima 09], etc.), which is not a fully KM process. The digital preservation platform is just one carrier or one box for the “knowledge”. These researches and projects’ critical work focus on knowledge identification, acquisition and encapsulation, and they integrate the digital preservation platform during the knowledge retention (engineering data retention) process.

In real life, there are different KM approaches in enterprises. Data acquisition in different information systems is also involved in different processes and adapted to different standards or principles. In some departments, there is even an integrated Knowledge Management System (KMS) dealing with the acquisition of knowledge. On the other hand, KM and digital preservation have different purposes. The data objects in KM (i.e. knowledge) are not always well structured for digital preservation, and vice versa. If we focus on just certain departments such as the previous long term preservation projects (e.g. manufacturing data), we have to redefine models, processes or systems in both KM and digital preservation. Nevertheless, this way narrows the utility of the long term preservation approach and lacks dynamic features.
Therefore, we propose a more adaptable architecture for long term preservation – Multi-layer Architecture for Dynamic Preservation of Knowledge (MadPK), which separates KM and digital preservation. This architecture will allow us to keep the existing various KM approaches or systems, while providing dynamically connected digital preservation platform. The MadPK architecture is shown in Figure 4.1. Our proposed architecture is different from any other existing digital preservation architecture. And we use this “separated-but-connected” structure of MadPK to benefit from existing technologies and models of both KM and digital preservation. The MadPK architecture is proposed for establishing the dynamic connections and communications. The layers in our MadPK are:

- **Enterprise Layer**: KM approaches among information systems and domain experts in a designated organization. This layer handles “knowledge source” (i.e. the information systems and people in enterprise) and produces knowledge from the enterprise business processes. There is more than one KM approach in this layer.

- **Digital Preservation Layer**: digital preservation platform. This layer captures, packages, transfers, stores knowledge and provides retrieving features for knowledge reuse. One of the responsibilities of the digital preservation platform is to ensure that the preserved information is in an independently understandable and reusable form to the end users in the long term. We adapt the OAIS reference model in our proposed preservation platform. Knowledge in this platform is formed on the basis of a formal Knowledge Model (i.e. Information Package of OAIS).

- **Mediation Layer**: integration methods and tools for digital preservation. This layer connects the other two layers, in a dynamic perspective. The data transferred between the two other layers are digital knowledge objects. So we name this layer as an enterprise knowledge bus (EKB), which enables the other two layers to call and acquire corresponding knowledge. Although this layer is named EKB, in the same way as the Enterprise Service Bus (ESB), more content and missions will be introduced in this layer (e.g. knowledge structure modeling, definition of knowledge transfer rules, storage and management of unfulfilled digital content for digital preservation, etc.). Therefore, our architecture does not only involve two layers (KM and digital preservation). The third layer (EKB) is the key to support the other layers’ interoperabilities as well as enable them to adapt complete existing models with technologies and without worrying about the interoperability issues.
In Figure 4.1, there are three layers that we have defined in our research work. The two-direction arrows between the layers state that there are forward and backward interactions and data exchanging among the layer. In general, MadPK works as:

- The digital preservation platform is stable to a certain extent, and stability is one key for long term data accuracy and security. This is the reason why we separate the knowledge repository from the knowledge production (Enterprise Layer, KM approaches).
- The reason why digital preservation platform can be stable to a certain extent is that its input and output knowledge are always packed in formal model, even if the knowledge and data from the knowledge source can be different in models and in formats. We develop all the model transfer activities inside Mediation Layer (EKB).
- In Enterprise Layer, changes of information systems lead to the change of KM approaches, and consequently the change of output knowledge model, which is the input of EKB. The output knowledge model of KM approach can be different depending on the complexity of the information systems in enterprise. However, whenever some or all of these knowledge models change, knowledge model conversion rules are upgraded correspondingly in the Mediation Layer (EKB).
4.2.2. Interoperability vs. Integration in MadPK

In our design of MadPK, we try to enhance the interoperability of the KM system and the digital preservation platform. Interoperability is the ability or the aptitude for two systems to understand each other and function together. The word “inter-operate” implies that one system performs an operation for another system [Chen 02, 04]. Generally, interoperability has the meaning of a coexistence, autonomy and federated environment, whereas integration refers to the concepts of coordination, coherence and unification. However, on the other hand, according to our proposed MadPK architecture, we have also developed an integrated platform (consisting of KM and digital preservation). From the Table 4.1, we notice that integration and interoperability represent different levels of coupling.

<table>
<thead>
<tr>
<th>INTEGRATION</th>
<th>INTEROPERABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consistency of local objectives to global ones</td>
<td>No consistency between local and global objectives</td>
</tr>
<tr>
<td>Tightly coupled Components interdependent</td>
<td>Loosely coupled Components independent</td>
</tr>
<tr>
<td>Unification (languages, methods, tools, etc.)</td>
<td>Identity &amp; diversity preserved</td>
</tr>
<tr>
<td>Intra enterprise Fusion, re-structuration, etc.</td>
<td>Inter enterprise (Virtual enterprise, etc.)</td>
</tr>
</tbody>
</table>

Table 4.1: Integration vs. Interoperability [Chen et al. 06]

Separated systems, which are interoperable with each other, will reduce the coupling of the information systems. However, whether this way is better than an integrated system in the long term is not certain, according to different situations in enterprises. Generally, lower coupling will ease the update of systems and be more dynamic. Nevertheless, in long term digital preservation, one integrated system for KM and digital preservation may ensure the data availability from a technical and legal point of view.

Therefore, we will not determine that the Enterprise Knowledge Bus carries out internal communications within a system, or external communications between systems. Instead, this communication will provide services of knowledge transfer between the KM approach and the digital preservation approach.
4.3. Roles and Requirements of MadPK

4.3.1. Roles in MadPK

As our research is under the context of KM, the roles, which interact in our work, are extensions of actors in the KM process. In [Rao 05], knowledge roles are suggested for the online communities, and we adapt this suggestion of roles to our research work. We have added one more role as “Knowledge Storage Manager” and redefine the other roles and to make them fit for our MadPK:

- **Knowledge Consumer**: the activities of this role are searching, browsing, accessing, applying, and learning knowledge. In our MadPK, this role is the domain experts and information systems in enterprise, who consume knowledge in the long term.

- **Knowledge Creator**: the activities of this role are to publish, improve, classify, and discuss knowledge. As we have introduced in Chapter 3, the domain experts and information systems in enterprise are also the knowledge source, from which the preserved knowledge is extracted. Analogously, in the OAIS reference model, the roles Producer and Consumer are represented by Knowledge Creator and Knowledge Consumer. Nevertheless, in MadPK, the two roles could be fulfilled by the same individuals, who are in different positions in the temporal dimension. For example, the designer of one product preserved the design data five years before, as the Knowledge Creator. After five years, the same designer (or other individuals but in the same position) retrieves the preserved product design knowledge for the maintenance of the product, or for designing a similar new product, in the same way as the Knowledge Consumer.

- **Knowledge Editor**: the activities of this role are interviewing experts, storytelling, and content management. In MadPK, this role collects information from Enterprise Layer, and submits the extracted knowledge into EKB. This role also edits the information in digital preservation platform (Digital Preservation Layer).

- **Knowledge Expert**: the activities of this role are validating, certifying, and legitimizing knowledge. In MadPK, this role works in all the three layers. The Knowledge Expert designs the knowledge models according to different environments.
in Enterprise Layer and Digital Preservation Layer, and determines the knowledge transfer rules in EKB (Mediation Layer).

- Knowledge Broker: the activities of this role are locating experts and knowledge, identifying gaps, organizing the whole preservation architecture, filtering knowledge, and coordinating communities of practice (CoPs). In MadPK, this role plays in Mediation Layer, maintains the communications between the other two layers, and transfers knowledge into different forms.

- Knowledge Leader: the activities of this role are shaping KM agenda and aligning with enterprise business objectives. This role is the decision-maker of MadPK. Thus, this role makes knowledge preservation plans and strategies in each layer of the MadPK architecture.

- Knowledge Storage Manager: this role was not introduced in Rao’s research, which mainly focused on KM. However, as we have stated that the MadPK is a combination of KM and digital preservation, the knowledge preservation approach and platform require a role to manage and operate. This role manages and maintains the archived data and metadata, performs queries, checks for knowledge obsolescence, and configures the digital preservation platform.

We have identified the roles that interact in our proposed MadPK architecture. Our purpose of the role identification is to determine the usage requirements of each layer of MadPK and consequently to propose the functional design and models. Table 4.2 shows the roles in each layer of MadPK. The definition of roles will lead to the interactivities between the roles, and consequently lead to the requirements of the MadPK.

<table>
<thead>
<tr>
<th>Role</th>
<th>Enterprise Layer</th>
<th>Digital Preservation Layer</th>
<th>Mediation Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Consumer</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knowledge Creator</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Knowledge Editor</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Knowledge Expert</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Knowledge Broker</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Knowledge Leader</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Knowledge Storage Manager</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 4.2: Roles in each layer of MadPK
4.3.2. Usage Requirements of MadPK

We introduce the use cases of the architecture by layers and by interactions between layers in MadPK. The usage requirements will illustrate the basic functional needs of the MadPK architecture. Moreover, the interactions between different roles will help to identify the business processes with the MadPK architecture.

4.3.2.1. Requirements of Enterprise Layer

Enterprise Layer supports the KM approach with information systems in designated organization. In our research work, we aim to achieve the knowledge preservation with the production information systems in industries or enterprises. This is why we named this layer as Enterprise Layer. This layer produces product, process, and organizational data. By adapting KM approach and Context Level models, knowledge will be generated and packaged for the following preservation. Figure 4.2 shows the use cases of this layer.
Figure 4.2: Use Case diagram of Enterprise Layer
The Knowledge Leader makes the KM planning, which determines KM strategies and identifies the critical changes in the long term. According to the KM planning, the Knowledge Expert generates the knowledge model, which is the structure of the exacted data. Then, the Knowledge Creator extracts knowledge from the domain experts or from information systems, based on the knowledge model. The knowledge packages will be sent to Mediation Layer, and the Knowledge Broker will operate these packages.

When the Knowledge Consumer would like to retrieve information from the preserved knowledge, queries need to be sent to the Knowledge Broker, too. By going through the communications with Digital Preservation Layer, the KnowledgeBroker sends back the query responses and the knowledge packages, in order that the Knowledge Consumer can use the required information. Similarly, when critical changes (i.e. policies, resources, software changes, etc.) in Enterprise Layer are captured by the Knowledge Expert, signals (e.g. queries, notifications, etc.) also need to be sent to the Knowledge Broker, in order to inform and cooperate with the Digital Preservation Layer.

4.3.2.2. Requirements of Digital Preservation Layer

Digital Preservation Layer is the digital preservation platform. This layer captures, packages, transfers, stores knowledge and provides features such as search and browse for the knowledge reuse. The digital preservation platform in our research work is an instance of OAIS. The use cases of this layer are shown in Figure 4.3.

The Knowledge Leader also makes plans for digital preservation in this layer. As the knowledge is packaged according to the Information Package model of OAIS, the Knowledge Expert creates, audit and modify the Information Package structure and content in this layer. Normally, the Information Package structure is firmed. However, in certain circumstances, Information Package should be restructured according to critical changes in Enterprise Layer. Once the Information Package structure is settled, the Knowledge Storage Manager gets submission from the Knowledge Broker (from Mediation Layer). Then, the Knowledge Storage Manager validates submission, generates Information Packages, stores the packages in certain media (i.e. physical media such as hard disk, flash disk, optical disc, etc.) and updates database of the digital preservation platform. Moreover, according to the OAIS recommendation, the Knowledge Storage Manager handles the queries from the Knowledge Broker (originally from the Knowledge Consumer) and delivers responses. The dynamic
feature of this layer is that the Knowledge Storage Manager updates the preserved knowledge according to the real time critical change signals from Enterprise Layer.

![Figure 4.3: Use Case diagram of Digital Preservation Layer](image)

4.3.2.3. Requirements of Mediation Layer

Mediation Layer is the integration of methods and tools for dynamic communications. This layer connects the previous two layers together. For the purpose of dynamic perspective, we acquire Service-Oriented Architecture (SOA) theory and aspect in our proposal. Thus, the digital preservation platform connects to the knowledge source through enterprise knowledge bus (EKB). The use cases of this layer are shown in Figure 4.4.
The Knowledge Leader also makes EKB plans in this layer. The planning includes the model transfer rules and the configuration of the dynamic connections. As there are different forms of knowledge between the other two layers, the primary job of this layer is the creation and modification of model transfer rules, which allow the knowledge of different form to be regenerated and recognized by either layer. At the same time, the configuration of the connection between layers has to be done by negotiating with the other two layers. Once the connection and the model transfer rules are determined, the Knowledge Broker manages the communications between the other two layers, and delivers knowledge, queries and the critical change signals. Sometimes, a critical change signal will also appear as a knowledge package or as a query message when being submitted.

Figure 4.4: Use Case diagram of Mediation Layer

4.3.2.4. Communications between Layers

Through the synthesis of the previous use cases, we have noticed that there are some communications between different layers. In order to make the use case diagram readable, we will not put all the roles and use cases together to show the interactions and communications
between layers. Instead, we extract the activities of interactions between layers. Figure 4.5 and Figure 4.6 show the communication between Mediation Layer and the other two layers. The interactions mainly are exchange of information. The dynamic feature is that all cross-layer information and the knowledge are transferred through the Knowledge Broker (and through the EKB of Mediation Layer), without worrying about the formats and data structure. In other words, both KM and digital preservation approaches will pay more attentions on their businesses and benefit from the existing methods, technologies and tools. This kind of architecture looses coupling of different information systems and makes the digital preservation platform and preserved knowledge be stable in certain extent.

![Diagram](image)

**Figure 4.5: Use Case diagram of communication between Enterprise Layer and Mediation Layer**
However, in some circumstances, the Knowledge Consumer communicates with the digital preservation platform directly. In our previous research on digital preservation platforms, there is a browsing feature in most of the digital preservation platforms. Thus, the Knowledge Consumer may connect to digital preservation platform for browsing of preserved knowledge (Figure 4.7). Nevertheless, when the Knowledge Consumer finds appropriate knowledge in digital preservation platform, the knowledge objects may also need to go through the knowledge transfer process of Mediation Layer in order to be used correctly in Enterprise Layer (e.g. need to be generated into certain form).
These usage requirements that we have introduced in this section lead to the functional design for the MadPK architecture. We will introduce the functional design in the following sections.

4.4. Functional Design of MadPK

The structural design of MadPK is represented by Architecture Model, while the functional design is realized by Function Models. As we have stated in Chapter 2, the functional design appears as the Function Model of our extended CommonKADS methodology. In order to identify the Function Model, we have to identify the basic functionalities and the sub-functionalities that construct the MadPK architecture. We have defined the relations between the Function Model and the Architecture Model as:

- The functions are basic elements within the architecture, and can be seen as the decomposition of the architecture;
- The functions are organized and determined by the architecture;
- The functions are determined by the duty of the architecture, as well as the requirements of long term knowledge preservation.
4.4.1. Enterprise Layer

Figure 4.8 shows the Function Model of Enterprise Layer, while we consider the other two layers as black boxes, which provides sufficient capacities for appropriate inputs and outputs. Generally, the Knowledge Acquisition and Knowledge Reuse are both directed by the Knowledge Model that defined in KM Planning. According to the KM strategy developed by KM Planning, KM Acquisition captures knowledge from information systems or domain experts, and then submits it into Mediation Layer. The Knowledge Reuse function is triggered when an end user attempts to use preserved knowledge or when the critical changes are detected in KM Planning. After Knowledge Reuse is triggered, it sends queries to Mediation Layer and waits for the responses. When the responses (i.e. query results, error messages, etc.) are sent back from Mediation Layer, Knowledge Reuse handles the response objects, get use of the query results, sends more queries or terminates the knowledge reuse process.

Figure 4.8: Function Model in Enterprise Layer

4.4.1.1. Knowledge creation

We have introduced the KM approach in Chapter 2, and we have stated that the functional aspect “strategy alignment” is the first and primary task in KM approach, for it will lead to the following each step or function in a KM project.

Business process management (BPM) and KM is two major parts in company. As the trend of enterprise has changed from industrial economy to knowledge economy, KM is required to be
able to interoperate with BPM. Automating business process is able to preserve business rules. The repository of business rules makes companies easy to change business logic and to manage process knowledge. In other words, one requirement of coordinating business logic and knowledge is automating process culture (process automation culture).

If considering a knowledge related project, different business processes have different knowledge intensities. Some processes may mostly benefit from the knowledge sharing culture, while others mainly benefit from the automating process. No matter what sort of business process is concerned in KM, the knowledge workflow must have correlations with the business processes. Both the knowledge and business process are supervised by the corporation strategy and rules. Of course, in corporation strategy, BPM and KM have different objectives. Therefore, in the knowledge creation process, we have adapted the PPO model concept, and we acquire not only the data object (product), but also capture the organizational and business process information, which is associated with the data object (Figure 4.9).
Knowledge Acquisition

KM Planning

Knowledge Acquisition

Acquire IS info.

Design Knowledge Model

Set KM objectives

Capture Business Process

Identify knowledge object - Product

Synthesize system and organizational structure

Identify KM problems and solutions

Describe organization focal area

Breakdown business process

Identify product knowledge

Judge KM feasibility

Acquire product knowledge

Launch Information system

Migrate product data

Analyze tasks (sub-processes)

Identify information systems

Send knowledge to Mediation Layer

Terminate the KM process for LTKR

KM project is feasible for LTKR

(true)

Figure 4.9: Activity diagram of knowledge creation
This process produces several documents. According to the Context Level model set, which we have introduced in Chapter 2, the sub-function of “design knowledge model” in KM Planning determines the detailed structure of the Organization, Task and Product Models. Based on this structure, the function of Knowledge Acquisition performs interview to the Knowledge Creator and extract knowledge from the knowledge source. After the extraction of organizational and environmental information, Knowledge Acquisition function will judge the feasibility of the KM process (performed by the role of the Knowledge Leader). If the knowledge objects are not fit for long term preservation, the KM process will be terminated. For example, a KM objective of a long term preservation project is preserving knowledge for reengineering and redesign. By analyzing the information systems for workshop planning and scheduling, the Knowledge Leader considers the scheduling information is not significant for reengineering and redesign in long term. Thus the KM process in this information system may be terminated.

The acquired knowledge (several documents) will be packaged and be submitted to Mediation Layer. In Mediation Layer, the Knowledge Broker will transfer the knowledge into Information Package form and submit the knowledge to digital preservation platform. The Knowledge Broker will send notifications to Enterprise Layer to inform the Knowledge Editor, in order to confirm the submission or request for resubmission.

4.4.1.2. Knowledge query

We place “knowledge query” as the title of this section, rather than “knowledge reuse”. The reason why we use knowledge query is that the requests from Enterprise Layer are not always from the Knowledge Consumer. In some circumstances, when the critical changes are identified in Enterprise Layer, the Knowledge Expert will send query requests (critical change signals) in order to check and update the knowledge in digital preservation platform. In either case, Enterprise Layer throws queries into Mediation Layer and waits for responses from digital preservation platform. After the Knowledge Expert sends the critical change signals, a backward query would be received from digital preservation platform, in order to get enough information for the knowledge update. In other words, the sending query processes by the Knowledge Consumer and by the Knowledge Expert are identical (Figure 4.10). In any case, Enterprise Layer sends “something” to Mediation Layer.
In the function of Knowledge Reuse, after sending queries, responses will be sent back from Mediation Layer. At times, the required knowledge may not be found in digital preservation platform, and the responses message will suggest changing some information on queries. This happens when the Knowledge Consumer want to reuse the preserved knowledge. In this circumstance, the Knowledge Consumer will choose re-send modified queries or stop the query process.
4.4.1.3. Knowledge evaluation

The Knowledge Evaluation function plays in Enterprise Layer. However, its purpose is to evaluate whether to terminate the knowledge, which is preserved in digital preservation platform. This is a dynamic feature for MadPK in Enterprise Layer. The Data Management function of Digital Preservation Layer checks the preserved Information Package periodically, and filters the Information Packages, which are not been used in a designated long term (e.g. one year, etc.). From the Digital Preservation Layer side, the Knowledge Storage Manager synthesizes the summaries of these filtered Information Packages and throws the Descriptive Information of these Information Packages as queries to the Knowledge Broker. The Knowledge Broker transfers the queries into appropriate forms and sent them to Enterprise Layer. This is how the Knowledge Evaluation function is triggered.

The Knowledge Evaluation function handles the queries from digital preservation platform and tries to retrieve knowledge by the Knowledge Creator from current knowledge source (i.e. information systems or domain experts).

- If the Knowledge Creator provides certain knowledge corresponding to the queries, the Knowledge Expert will decide whether the preserved knowledge in digital preservation platform should be updated or not. If yes, new submissions will be made and be thrown to the Mediation Layer; if not, a simple notification will be sent to indicate that the preserved knowledge is up-to-date and still be valuable in Enterprise Layer. In other words, the knowledge should be kept.

In some case, there is not corresponding knowledge retrieved according to the queries, because the checking period (designated long term in Digital Preservation Layer) may longer than the product life cycle of a certain product in this organization. Of course this is the main reason why we need to perform long term preservation. When there is no query results from the Knowledge Creator, the Knowledge Expert must go back to check the original documents of KM planning. Here is a manual function or human task, which should be performed by the Knowledge Expert and the Knowledge Leader. They will decide whether the preserved knowledge, which is not been used in a designated long time, should be preserved for longer time or should reach its obsolescence.
Figure 4.11: Activity diagram of knowledge evaluation
The submission for long term preservation comes from the KM processes in Enterprise Layer, with pre-performed analysis and KM objectives. However, the Digital Preservation Layer could not see the KM processes and may argue the efficiency of utilization of the resource for maintaining Information Package, which is not been touched for a long time. Sometimes, due to the organizational changes or technical issues, it is quite possible that certain Information Packages, which should have been updated or terminated, are not handled correctly in time. Therefore, the Knowledge Evaluation function is necessary and is one significant feature in MadPK.

4.4.2. Digital Preservation Layer

The Function Model of the Digital Preservation Layer has already presented in Chapter 3 (Table 3.3). We notice that the main knowledge objects exchanged in digital preservation platform are submission information package (SIP), archival information package (AIP), and dissemination information package (DIP). Therefore, we distinguish the whole process of OAIS data workflow into four phases: “Receiving submission and validating SIP”, “Generating and storing AIP (with Descriptive Information)”, “Generating DIP and sending results”, and “Updating AIP”. Thus, we describe the digital preservation platform not only from a functional viewpoint, but also from a business process viewpoint. This will help us in the following development by adapting BPM (Business process management) and SOA (Service-Oriented Architecture) technologies and applications.

![Figure 4.12: Function Model in Digital Preservation Layer](image-url)
4.4.2.1. Receiving submission and validating SIP

As we have introduced previously, we identify the “Producers” as the production information systems or domain experts in the KM approach, but they submit knowledge or data in their own knowledge model, which is not an exact Information Package. The Ingest function of the digital preservation platform gets knowledge in Information Package form. (Figure 4.13)

- When acquiring SIPs, there is a sub-function to validate and audit the submission (sub-function: Validate Submission). Although SIPs are actually produced by going through the KM approach, within an OAIS, these SIPs should be proved and validated. This sub-function of “validate submission” will check the submission to make sure that there is no error (the definition of error depends on the policies made by the Knowledge Expert, e.g., missing description information, wrong format of package title, etc.) in the submission. However, whether errors are detected or not, the digital preservation platform will send a Confirmation of Receipt to the role who controls the KM approach, in order to confirm that the SIPs have been received. Then in case of errors resulting from the submission, a “re-submit” request will be included in the Confirmation of Receipt.

- In the OAIS recommendation, despite the fact that it has developed multiple functions to deal with multiple situations, we still need to provide more extra feature in our real implementation. In this receiving submission case, Knowledge Expert, the Knowledge Editor and the Knowledge Broker will have to negotiate and make an agreement on a definite lead time of sending Confirmation of Receipt. If the definite lead time has expired, the Knowledge Editor will once more send the submission and inform the OAIS of the resubmission in the Descriptive Information of the SIPs.
Once the SIPs (without errors) are obtained, all these SIPs will be sent to go through the Quality Assurance process. For digital submissions, these mechanisms (of Quality Assurance) might include Cyclic Redundancy Check (CRC) or checksums. Checksum is one simple way ensuring the security of the transferred digital objects. Checksums are associated with the data files or written in the system logs [CCSDS 650.0-B-1; ISO 14721:2003].

Before they are utilized to generate AIPs, the SIPs have to be verified by the sub-function of “audit” in the entity Administration. The “audit” sub-function makes sure that the submissions meet the specifications of the Submission Agreement, which is negotiated by both the Knowledge Editor and Knowledge Expert [CCSDS 650.0-B-1; ISO 14721:2003].
ISO 14721:2003]. After the audit process, an Audit Report will be sent to the entity Ingest to generate AIP, and a Final Ingest Report will be generated, too.

4.4.2.2. Generating and storing AIP (with Descriptive Information)

The validated SIPs are transferred into Archival Information Packages (AIPs) for storage (sub-function: Generate AIP). Besides, descriptive metadata (Descriptive Information) should be extracted from the AIPs (sub-function: Generate Descriptive Information), simultaneously AIPs will be retrieved by this Descriptive Information. (Figure 4.14)

- Although proper SIPs are obtained and validated by the previous process, more information on existing information packages in the OAIS is still needed for generating new AIP. Thus to generate AIPs, validated SIPs as well as reports from Data Management are required. There is a Generate Report sub-function in Data Management, which will deal with all the report requests. Once the Generate AIP function has sent report request to the entity Data Management, the Generate Report sub-function will provide report, which may include summaries of archive holdings by categories, or usage statistics for access to archive holdings. With both the SIPs and the reports from the entity Data Management, the Generate AIP sub-function transforms SIPs into the AIPs.

- Generating AIPs may involve the conversions of the file formats, the data representation or just a simple re-organization of the content in the SIPs. The content of the report from Data Management is used to generate Descriptive Information that complete the AIPs. The mapping between SIPs and AIPs is not one-to-one, and it depends on the Data Formatting and Documentation Standards as well as the type of SIPs. According to the OAIS recommendation, this mapping between SIPs and AIPs can be one-to-one, many-to-one, one-two-many, many-to-many, one-to-zero (Annex 4).
Similar like the SIPs, the AIPs also need to be verified by the sub-function of “audit” in Administration. The AIPs will go through the same process of auditing to make sure that the AIPs meet the specifications of the submission agreement. After the audit process, an Audit Report will be sent to the entity Ingest to confirm that AIPs are approved and ready to be stored.
After the AIPs are generated and verified, the Ingest process will go through two parallel paths. One is contacting the Archival Storage entity and store the AIPs in certain sort of media; the other is getting Descriptive Information and transferring it to the entity Data Management to update the database.

When transferring the AIPs to Archival Storage, a Storage Request will be attached to the AIPs. The Storage Request may represent an electronic, physical, or virtual transfer. Moreover, in the Storage Request, the frequency of utilization of the data objects comprising the AIP may be indicated. Thus, after the AIPs and the Storage Request have been sent to the Receive Data function in Archival Storage, appropriate storage devices or media will be selected for storing the AIPs according to the requirements. This function performs the real physical storage activities and stores the AIPs. After the storing of AIPs has been accomplished, the Receive Data function will send a Storage Confirmation message to Ingest, including the storage identifications of the AIPs.

The result of the Generate Descriptive Information function proposed in the OAIS recommendation has more content than the extracted Descriptive Information from AIPs. Still, it is not the final Descriptive Information that will be stored in the OAIS database. The Generate Descriptive Information function extracts Descriptive Information from AIPs, collects Descriptive Information from other sources (information for searching and retrieving) and mixes them together. After the Storage Confirmation message has been sent by Data Management, the storage identification information also needs to be added into Descriptive Information. In other words, the generating of final Descriptive Information will go through the three following phases: extracting it from AIPs, collecting it from other sources, and obtaining it from Storage Confirmation.

As soon as the final Descriptive Information is generated, it will be sent into the sub-function of “archive data” in Data Management, along with a Database Update Request. Although the Descriptive Information requires the Storage Confirmation from Archival Storage, Data Management updates may take place without a corresponding Archival Storage transfer when the SIP contains Descriptive Information for an AIP already in Archival Storage. In this case, since the AIP is already stored in Archival Storage, a Storage Confirmation with storage identification is still available. The Receive Database Updates is partial of the sub-function
“maintain storage”, which adds, modifies or deletes information in the Data Management persistent storage. In any case (updates succeed or fail), the “maintain storage” sends a Database Update Response back to Ingest, indicating the status of the update.

4.4.2.3. Generating DIP and sending results

This process corresponds to the Access entity in the OAIS reference model. In the Knowledge Reuse process, end users send queries to the digital preservation platform, and search for appropriate information. According to the knowledge model we have defined in our KE approach, the descriptive information of one information package represents all the critical information corresponding to the data content. Therefore, when searching, only the predefined descriptive information is searched, while other information related to the data content is packaged in the information package. (Figure 4.15)

- After searching of information, the preservation platform should convert information package again (i.e. convert AIP to DIP) and deliver the DIP to end user. This is because inside of the digital preservation platform, the information package form is fit for knowledge and storage management. As the supported information may not be sufficient for end users, the conversion process is necessary. The other reason for the conversion of information package is that we try to develop business process models which are dynamic and reusable. Besides, the conversion process could also be use in the knowledge changing/updating process.

- Three categories of Knowledge Consumer requests are distinguished: Query Requests, Report Requests and Orders. Query requests are performed in the Data Management entity and the responses of query requests are query results of the stored information (AIPs). Report requests could be considered a combination of a series of queries, and the query results need to be structured as a “formal” report, whose format is defined by end users. Orders represent the stored information held by the Archival Storage entity, and the AIPs will be generated as “formal” DIPs according to the requests. These three categories are transferred by KM approach After receiving the requests, estimates should be done in order to determine whether the existing resources are available for performing the requests and assuring the users are authorized to access and receive the requested items, etc. At last the Knowledge Consumer will be informed whether the requests are accepted or not. The response for the Knowledge
Consumer can also be a report, illustrating the information about the estimates or the reasons for the rejection of requests.

Figure 4.15: Activity diagram of generating DIP

- The type of requests must be checked. Both a Query Request and a Report Request are simpler than an Order. When an accepted request is a query request, the digital preservation platform just sends the query/report request to the entity Data Management, and within Data Management there are sub-functions performing
queries among the stored Descriptive Information and/or generating specific formatted reports using the query results. These queries are performed quite fast and efficiently. And the results sets or reports will be sent back to the Knowledge Broker, and consequently to the Knowledge Consumer.

- The Dissemination Requests, which are used to request the DIPs, may not come from the Knowledge Consumer but from internal activity of the digital preservation platform. Generally, an Order from the Knowledge Broker can be an Ad hoc Order, which is executed only once, or an Event-based Order that will be maintained by the entity Administration. When Access receives and accepts an Ad hoc Order, it will generate a Dissemination Request. And when Access receives and accepts an Event-based (Event-driven) Order, it will send the order to the entity Administration. The entity Administration holds the event-driven requests and checks the request and the storage periodically in order to make sure the event-driven orders are still fulfillable. And if the designated “event” is reached, the entity Administration generates the Dissemination Requests and sends them to the Access entity.

- The Generate DIP sub-function generates DIPs from the AIPs, which come from the Archival Storage entity, according to the Dissemination Requests. The reason why we need both the AIP and Descriptive Information to generate DIP is that the AIP contains less Descriptive Information than the final Descriptive Information stored in Data Management. To acquire AIP, the Generate DIP sub-function calls the Provide Data sub-function in Archival Storage with an AIP Request. And Provide Data sub-function will provide the requested AIPs and transfer them to the entity Access or to a check area. Meanwhile, a Notice of Data Transfer will be sent so as to Generate DIP and make sure the quest is accomplished. As we have discussed before, if the receiver could get the notice in a designated time period, the request would be sent once more. To acquire Descriptive Information, the Generate DIP sub-function sends a report request to Generate Report sub-function in Data Management. The latter will send back a report with all the required Descriptive Information. The Generate DIP sub-function will generate DIP with all the required data, and the mapping between AIP and DIP is just similar as that between SIP and AIP. In other words, the detailed generating DIP process depends on the policies of the digital preservation platform and the Dissemination Request.
As the Dissemination Request come from the outside of the preservation platform, The Deliver Response sub-function handles both on-line and off-line deliveries of responses (DIPs, result sets, reports and assistance) to the KM approach. As we model the information packages workflow, the delivery can be considered as the end of the workflow. The delivered response will be handled in the Mediation Layer.

4.4.2.4. Updating AIPs

Updating AIPs in the OAIS is a self-submission cycle process. When the updating process is triggered, the entity Administration sends a Dissemination Request to Access and gets the resulting DIPs. The contents of the DIPs will be updated and submitted to the digital preservation platform itself. When Ingest catches the “DIPs”, it recognizes them as SIPs, which are submitted to the digital preservation platform.

- Although we can consider the archival information updating process as the learning evolution within an OAIS (based on the Preservation Planning), the process also goes through the normal submission process (only the SIPs are from the submitted DIPs of the OAIS). In the OAIS recommendation, there is no Confirmation of Receipt requested in this kind of SIP submission.

- Although there may not need a Resubmit Request when SIPs inside the OAIS, we may need to confirm the receipt of the packages. Thus, in implementation, sending a Confirmation of Receipt in both situations when receiving SIPs should be sufficient. Nevertheless, when within the OAIS, we have a huge amount archival information to be updated; the sending confirmation information will be an insufficient function wasting also a huge amount of resources.

- The Figure 4.16 shows the sketch of the archival information updating process. This is not a formal activity diagram, but it can show much clearer the self-submission cycle of the digital preservation platform.
4.4.3. Mediation Layer

This layer should be developed based on the development of the previous layers. The purpose of Mediation Layer is to enable and enhance the interoperability between Enterprise Layer and Digital Preservation Layer. Interoperability is “the ability of two or more systems or components to exchange information and to use the information that has been exchanged” (IEEE Standard Computer Dictionary). As the digital preservation process includes multiple systems and digital models, we propose to adapt Service-Oriented Architecture (SOA) to support our multi-layer structure of MadPK. SOA configures entities (services, registries, contracts, and proxies) to maximize loose coupling and reuse [McGoven, 03]. SOA stresses interoperability, the ability of systems using different platforms and languages to communicate with each other. Each service provides an interface that can be invoked through a connector. With SOA, we will define the interfaces in terms of protocols and functionalities. This enhances not only the interoperability between digital preservation platform and
information systems in enterprises, but also the interoperability between the digital preservation platform and web technical services. The former is the critical point when implementing a preservation platform into an enterprise, and the latter is a solution to overcome the issue of rapid development of technologies, which is against the capacity of knowledge retention.

The Mediation Layer connects the other two layers together, in a dynamic way. In our proposal, we build an Enterprise Knowledge Bus (EKB), which handles the knowledge objects and handles the communications between Enterprise Layer and Digital Preservation Layer. As the services provided by Mediation Layer are all dedicated for knowledge objects, we propose the EKB instead of the Enterprise Service Bus (ESB). The EKB also aims to adapt service concept to reduce the coupling of the platform and consequently to enhance the interoperabilities.

The Mediation Layer may develop multi-level ontology framework to support the structure and the evolution of existing models as well as to ensure the alignment between knowledge models in the other two layers. Ontology is one way to make the digital preservation to be dynamic. However, in our design, we do not specify the methodology for establishing model transfer rules in Mediation Layer, because the model transfer rules should be differ and specified according to the knowledge source types and company’s KM strategy. Therefore, we just propose the functional design of Mediation Layer (Table 4.3).

<table>
<thead>
<tr>
<th>Architecture Info.</th>
<th>Function</th>
<th>Sub-function</th>
<th>Associated Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mediation Layer</td>
<td>EKB Planning</td>
<td>Develop Model Transfer Rules</td>
<td>Organization Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Task Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Product Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Transformation Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configure Connection</td>
<td>Transformation Model</td>
</tr>
<tr>
<td>Knowledge Integration</td>
<td></td>
<td></td>
<td>Organization Model</td>
</tr>
<tr>
<td>Knowledge Distribution</td>
<td></td>
<td></td>
<td>Task Model</td>
</tr>
<tr>
<td>Knowledge Retrieval</td>
<td>Transfer Query</td>
<td></td>
<td>Product Model</td>
</tr>
<tr>
<td></td>
<td>Capture Response</td>
<td></td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Transformation Model</td>
</tr>
</tbody>
</table>

Table 4.3: Function Model of Mediation Layer
The functions are published as services according to the connection between the Enterprise Layer and Digital Preservation Layer. The main functions of the Mediation Layer are:

- **EKB Planning:** EKB Planning should configure the connections between the other two layers. Although the information systems in enterprises and digital preservation platform share and connect to an EKB, the connection configuration should be performed when there are organizational or technical changes of these information systems. In fact the basics of the connection configuration is the model transfer rules, which identify the methods of transferring enterprise knowledge into Information Package form or conversely.

- **Knowledge Integration:** Knowledge Integration transfers the knowledge from the form in Enterprise Layer to the form in Digital Preservation Layer (Information Package). We call this function “integration”, because the formal form of knowledge object in digital preservation platform is Information Package, and the knowledge from Enterprise Layer can be various and in different forms.

- **Knowledge Distribution:** the Knowledge Distribution function seems like the converse of the Knowledge Integration, but it is not. It provides knowledge for the Knowledge Consumer by distributing and reediting the Information Package. Most of the time, there is no need to transfer the Information Package into certain for knowledge reuse, because normally preserved knowledge in Information Package form is readable and reusable by end users. Therefore the Knowledge Distribution function sometimes is just a connector to transfer the query results.

- **Knowledge Retrieval:** this function handles queries between the other two layers and sometimes invokes Knowledge Integration or Knowledge Distribution for transferring knowledge.
Figure 4.17 shows the Function Model of Mediation Layer. The EKB may represent a piece of software that lives between the business applications (information systems) and enables communication between them. We use EKB to replace the direct contacts with the applications on the bus, so that all the communications take place via EKB. This needs the Knowledge Broker to handle the knowledge model definition and model transfer rules in EKB. We have defined Knowledge Integration and Knowledge Distribution models in EKB, and they recognize the input knowledge and encapsulate or reorganize it into an acceptable knowledge object for either layer. The logic of invoking functions or services in Mediation Layer is shown in Figure 4.18.
Receive knowledge input

<<decisionInput>> Knowledge object is in Information Package form

Invoke Knowledge Distribution service

<<decisionInput>> The knowledge input is query or response

Invoke Knowledge Retrieval service

Send knowledge object to its destination

[true]

[false]

Invoke Knowledge Distribution service

Send knowledge to Digital Preservation Layer

[true]

[false]

Invoke Knowledge Integration service

Send knowledge to Enterprise Layer

[false]

[true]

Transfer knowledge object into Information Package

Figure 4.18: Activity diagram of Mediation Layer
One knowledge input of the EKB is encapsulated as a knowledge object, or a simple message. The message that EKB receives can be the queries or the responses to the queries. When receiving these messages, EKB just clarifies the submitter and the destination, and sends the message directly. Nevertheless, when EKB receive encapsulated knowledge object package, it should estimate the form of the package, in order to use model transfer rules to transfer it.

- If a knowledge object package is in an Information Package form, which is the formal structure in digital preservation platform, EKB invokes the Knowledge Distribution function or service, in order to distribute the Information Package to a correct destination (this destination is not always the Enterprise Layer, and sometimes it is the Digital Preservation Platform itself, e.g. in “updating AIPs” process).

- If a knowledge object package is not in Information Package form, EKB must invoke the Knowledge Integration function or service, in order to construct the knowledge object in an Information Package way, based on the model transfer rules.

The Function Model of Enterprise Layer and Digital Preservation Layer have already been shown in Chapter 2 and Chapter 3. Table 4.4 shows the general view of the Function Model of the three layers of MadPK, including Mediation Layer. The functions, sub-functions and associated models in Table 4.4 are identical regarding Table 2.4, Table 3.3 and Table 4.3. Table 4.4 illustrates all the proposed functions together, and sorted by layers.

<table>
<thead>
<tr>
<th>Architecture Info.</th>
<th>Function</th>
<th>Sub-function</th>
<th>Associated Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Layer</td>
<td>KM Planning</td>
<td>Develop KM Strategy</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Capture Critical Change</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Design Knowledge Model</td>
<td>Organization Model, Task Model, Product Model</td>
</tr>
<tr>
<td>Knowledge</td>
<td>Acquisition</td>
<td>Capture Organizational Info.</td>
<td>Organization Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Identify Business Process</td>
<td>Organization Model, Task Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Idnetify Product Info.</td>
<td>Organization Model, Product Model</td>
</tr>
<tr>
<td>Knowledge</td>
<td>Evaluation</td>
<td>Handle Query</td>
<td>Organization Model, Task Model, Product Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Determine Knowledge Obsolescence</td>
<td>-</td>
</tr>
<tr>
<td>Knowledge</td>
<td>Reuse</td>
<td>Create Query</td>
<td>Organization Model, Task Model, Product Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Distribute Knowledge</td>
<td>-</td>
</tr>
<tr>
<td>Architecture Info.</td>
<td>Function</td>
<td>Sub-function</td>
<td>Associated Model</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------</td>
<td>--------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Digital Preservation Layer</td>
<td>Ingest</td>
<td>Validate Submission</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Generate AIP</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Generate Descriptive Information</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td>Archival Storage</td>
<td>Archive Data</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintain Storage</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide Data</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Terminate Data</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Data Management</td>
<td>Archive Data</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Generate Report</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Perform Query</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Filter Obsolescent Data</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Administration</td>
<td>Audit</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configure Platform</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configure Policy</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Authorize Data Evaluation</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Preservation Planning</td>
<td>Develop Preservation Strategy</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Capture Critical Change Signal</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Design Information Package Model</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td>Access</td>
<td>Handle Query</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Generate DIP</td>
<td>Knowledge Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deliver Response</td>
<td>-</td>
</tr>
<tr>
<td>Mediation Layer</td>
<td>EKB Planning</td>
<td>Develop Model Transfer Rules</td>
<td>Organization Model, Task Model, Product Model, Knowledge Model, Transformation Model</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configure Connection</td>
<td>Transformation Model</td>
</tr>
<tr>
<td></td>
<td>Knowledge Integration</td>
<td>-</td>
<td>Organization Model, Task Model, Product Model, Knowledge Model, Transformation Model</td>
</tr>
<tr>
<td></td>
<td>Knowledge Distribution</td>
<td>-</td>
<td>Organization Model, Task Model, Product Model, Knowledge Model, Transformation Model</td>
</tr>
<tr>
<td></td>
<td>Knowledge Retrieval</td>
<td>Transfer Query</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Capture Response</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4.4: Function Model of MadPK

### 4.5. Conclusion

We have introduced the detailed design of Architecture Model and Function Model for MadPK (Multi-layer Architecture for Dynamic Preservation of Knowledge) in this chapter. According to the long term knowledge preservation requirements, we analyze the usage of the MadPK. We have identified the roles play in MadPK, according to the actors in KM approach.
Then, interactions between the MadPK roles are presented, and the interactions of roles consequently lead to the interactions and data communications between the layers of MadPK. According to the use cases of each layer, we propose Function Model within MadPK. In Enterprise Layer, KM approach is performed. The functional design achieves the knowledge creation, knowledge query, knowledge reuse, and knowledge evaluation processes. In Function Model of Enterprise Layer, the Context Level model set (Organization Model, Task Model and Product Model) is adapted as formal documentation structure of extracted knowledge. In Digital Preservation Layer, long term digital preservation approach is performed. The functional design in this layer achieves the submission, archival storage, retrieval of archived knowledge, and knowledge update processes. In Function Model of Digital Preservation Layer, the Concept Level model (Knowledge Model) is adapted as the formal form in repositories. At last in the Mediation Layer, the connections and communications of the other two layers are achieved. The functional design in this layer enables the interoperability of the other two layers and thus achieves the dynamic preservation goal.