

Invited Paper

A Study on Process Description Method for DFM Using Ontology

K. Hiekata¹ and H. Yamato²

¹Department of Systems Innovation, Graduate School of Engineering, The University of Tokyo, 5-1-5, Kashiwanoha, Kashiwa-city, Chiba 277-8563, Japan

²Department of Human and Engineered Environmental Studies, Graduate School of Frontier Sciences, The University of Tokyo, 5-1-5, Kashiwanoha, Kashiwa-city, Chiba 277-8563, Japan
hiekata@nakl.t.u-tokyo.ac.jp, yamato@k.u-tokyo.ac.jp

Abstract

A method to describe process and knowledge based on RDF which is an ontology description language and IDEF0 which is a formal process description format is proposed. Once knowledge of experienced engineers is embedded into the system the knowledge will be lost in the future. A production process is described in a proposed format similar to BOM and the process can be retrieved as a flow diagram to make the engineers to understand the product and process. Proposed method is applied to a simple production process of common sub-assembly of ships for evaluation.

Keywords:

Production process, DFM, Ontology, Computer system

1 INTRODUCTION

There are many research and computer program for supporting optimization of production process in shipyards. And knowledge of experienced engineers is embedded into the system. Once the knowledge is embedded into computer system, the knowledge is not transferred to young engineers and will be lost in the future. And in shipyards, initial design department is separated from production design and fabrication shops, so engineers in initial design department don't know production process very well. So a new system for describing knowledge, especially knowledge about production, is needed.

In some research work of knowledge management in industries, New Energy and Industrial Technology Development Organization's 'Digital Meister Project' was an project to explore this field [1]. The knowledge base system to transfer the knowledge is discussed [2].

In this paper, a method to describe process and knowledge in computer system is proposed. This method is based on RDF (Resource description framework) [3] which is one of ontology description languages and IDEF0 [4] which is a formal process description format. The production process is described in a proposed format similar to BOM (Bill of material) system [5]. Production process can be edited by interactive user interface to design the product and production process and can be retrieved as flow diagram to make the engineers understand the product and production process. Proposed method is applied to simple production process for common sub-assembly of ship and evaluated.

(Unified Resource Identifier) is assigned to all the objects, and metadata is defined for all objects using the URI. Metadata is defined as a statement with subject, predicate and object. The statement is called triple in RDF. Two kinds of elements, Literal or Resource, can be a component of a statement. One is Resource and the other is Literal. Resources are objects with URI, and can be subject, predicate and object in RDF statements. Literals are plain text and must be used only in object. Complicated products or processes can be described by iteration of definition of resources and literals. Usually RDF can be stored in XML or other format, and is visualised in RDF graph. In RDF graph, subject, predicate and object are shown as in Figure 1.

Sample of RDF graph with some information is illustrated in Figure 2. The RDF graph indicates "creator" of "http://www.nakl.t.u-tokyo.ac.jp/" is "Kazuo Hiekata", the date of the website is "2006-04-13" and updated information is at "http://www.nakl.t.u-tokyo.ac.jp/update.html". Prefix "dc" and "ex" used in Figure 2 are XML namespace defined in other files or website. "dc" is a standard vocabulary defined by DCMI (Dublin Core Metadata Initiative) [6]. Figure 3 shows XML format of RDF metadata of the RDF graph in Figure 2.

The RDF graph in Figure 2 is shown in XML format. The rest of RDF expressions in this paper will be illustrated in RDF graph because the graphs are easier to understand.

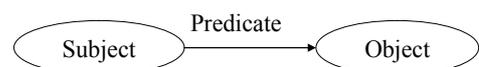


Figure 1 Simple RDF graph

2 METHODOLOGY FOR DESCRIBING PRODUCTION PROCESS

2.1 RDF and RDF Schema

RDF is one of a language for ontology and is a framework for describing metadata for some object. Metadata is data for explaining the object. For example author, date and publisher are metadata for a book. In RDF format, URI

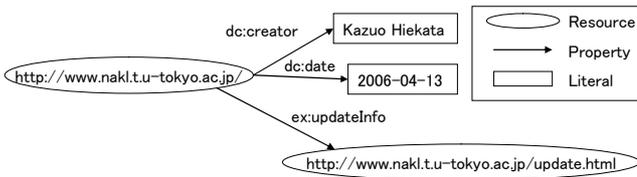


Figure 2 Example of RDF graph

```

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://www.nakl.t.u-tokyo.ac.jp/">
    <dc:creator>Kazuo Hiekata</dc:creator>
    <dc:date>2006-04-13</dc:date>
    <ex:updateInfo rdf:resource="http://www.nakl.t.u-
  tokyo.ac.jp/update.html" />
  </rdf:Description>
</rdf:RDF>

```

Figure 3 Example of RDF in XML file

Then RDF schema is introduced in this paragraph [7]. Metadata can be defined by RDF, but meta-languages are needed to define vocabularies and data schema of RDF. Resources in RDF are described as the class and instance concept same as in object oriented programming. To create models of real world using RDF, vocabularies to describe the object must be defined by RDF schema. Classes are defined by RDF schema and the relationships between instances are described by RDF. By defining new vocabularies based on RDF schema, RDF can be applied to any specific domains or specifies application.

2.2 Process description based on RDF and RDF Schema

Production process of building blocks of a ship is picked up in this paper. Production process is described in RDF metadata using newly defined vocabularies by RDF schema. At first, vocabularies required to describe production process are defined. Production process will be presented in formal manner using the vocabularies. To describe the process, parts and materials, "Operation", "Assembly" and "Primitive" are defined by RDF schema. "Duration", "input" and "output" are defined for relationships between the resources. The vocabularies are shown in Table 1.

"Operation" is a process in shipyards. And instances of "Operation" require clear input materials and/or parts and output assembly. To define the vocabulary, "Operation", "input" and "output" must be defined. "Input" can be the predicate of an RDF statement if the subject is an instance of "Operation". And "output" can be the object of an RDF statement. "Duration" is also defined for instances of "Operation". "Duration" must be Literal and be assumed as dimensionless number in this paper.

"Primitive" is a material which cannot be divided into multiple parts, or parts whose components design engineers don't need to consider.

"Assembly" is a member which has one or more instances of "Primitive" and "Assembly".

Part of the definition is shown in Figure 4. The first part is definition of "Assembly" class and the second part is definition of "duration". In the definition of "duration", some restriction is defined by RDF schema. "rdfs:domain" indicates that "duration" must be a property of specified class. So, the duration must be defined for instances of

"Operation" class. And the "rdfs:range" specifies the value of the property. The value of "rdfs:range" is Literal, so "duration" must be text string for man hour of a specified instance of "Operation" class. The definition by RDF schema strictly defines each term and is useful for formal description.

Process and product/intermediate product can be described using these vocabularies. This description method is based on the concept of IDEF0.

Basic process and product description by defined six vocabularies are shown in Figure 5. Figure 5 illustrates joining two steel plates to create a skin plate and the duration of joining process is 10. The left side of the figure is definition of basic members. The right side shows the process and parts. Two "Steel plate" are instances of Primitive class and "Joining plates" is process with two "input" and create "Skin plate" which is an instance of "Assembly" class. This schema is applied to all the production process.

```

<rdfs:Class
  rdf:about="http://www.sharefast.org/2008/09/productionproc
  ess#Assembly">
  <rdfs:isDefinedBy
  rdf:resource="http://www.sharefast.org/2008/09/productionp
  rocess#" />
  <rdfs:label>Assembly</rdfs:label>
  <rdfs:comment>The class of assemblies of primitives.
</rdfs:comment>
</rdfs:Class>

<rdf:Property
  rdf:about="http://www.sharefast.org/2008/09/productionproc
  ess#duration">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-
  rdf-syntax-ns#Property" />
  <rdfs:label xml:lang="en-US">duration</rdfs:label>
  <rdfs:comment xml:lang="en-US">Standard duration of
  an operation.</rdfs:comment>
  <rdfs:isDefinedBy
  rdf:resource="http://www.sharefast.org/2008/09/productionp
  rocess#" />
  <rdfs:domain
  rdf:resource="http://www.sharefast.org/2008/09/productionp
  rocess#Operation" />
  <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-
  schema#Literal"/>
  <dc:type
  rdf:resource="http://dublincore.org/usage/documents/princip
  les/#element" />
</rdf:Property>

```

Figure 4: Definition of vocabularies by RDF schema

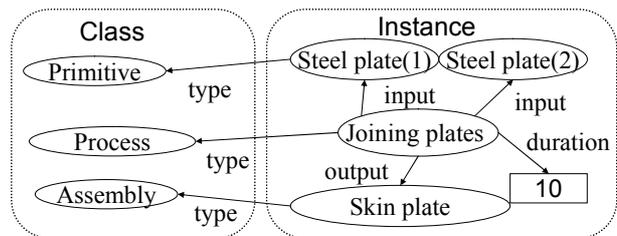


Figure 5: RDF Graph for process description

Name	Description
Operation	Class for operations in fabrication shop
Primitive	Class for parts or materials which cannot be divided into sub-components
Assembly	Class for parts consists of one or more "Primitive"
duration	Property for describing man hour of specified "Operation"
input	Property for describing input "Primitive" or "Assembly" for specified "Operation"
output	Property for describing output "Assembly" for specified "Operation"

Table 1: List of vocabularies for describing production process

3 SOFTWARE SYSTEM

3.1 System overview

Prototype system is implemented to evaluate proposed production process description method. The prototype system is developed ShareFast system which is an open source client/server document management system based on workflow [8]. ShareFast system manages document files using RDF metadata based on Jena framework [9][10]. The client program is developed under C# language, and the server program is developed under Java based on servlet technology. Metadata in the server can be accessed by several interfaces through servlets. The server has RDF model management component and the client program accesses to the metadata on the server by the componts via several interfaces. Figure 6 shows the system overview.

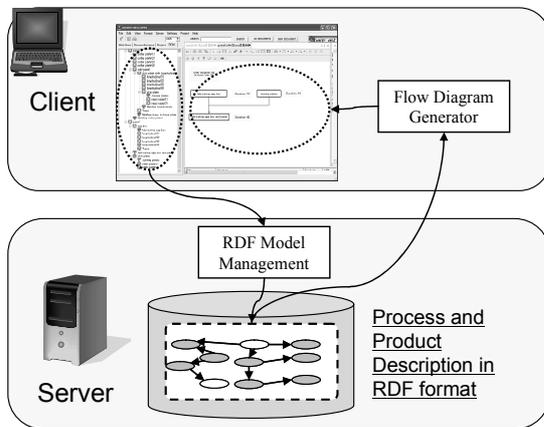


Figure 6: System overview

Figure 7 shows the main window of client program. The user can edit relationships between production process and parts using the left side of the program. Instances of "Operation", "Primitive" and "Assembly" class are shown in tree view in the left side. The system generates flow diagram of any node of "Assembly" instances in the tree view. Generated flow diagram will be shown in the right side of the program.

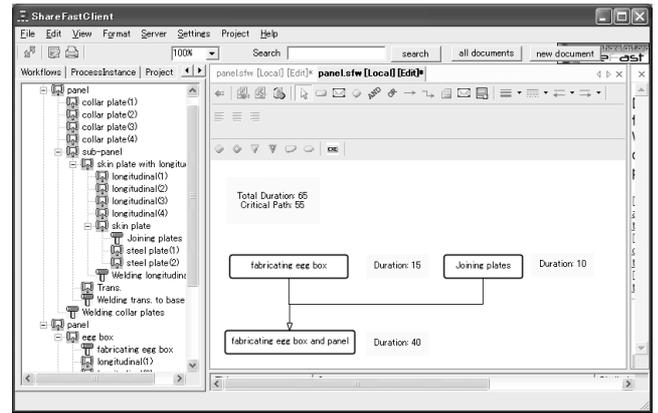


Figure 7: Interface of client program

3.2 Assembly definition

Proposed process description method consists of iteration of "Assembly" definitions. Figure 8 shows the input dialog box for defining "Assembly" instances. This dialog shows by right click menu of tree view of client program. Required field of this dialog box is the name of assembly, parts list of the assembly and the name and duration of the operation. The engineer is required to fill the name, and then the server creates instances in the metadata database in the server by assigning URI to defined objects. Figure 8 shows the same process and parts as Figure 5.

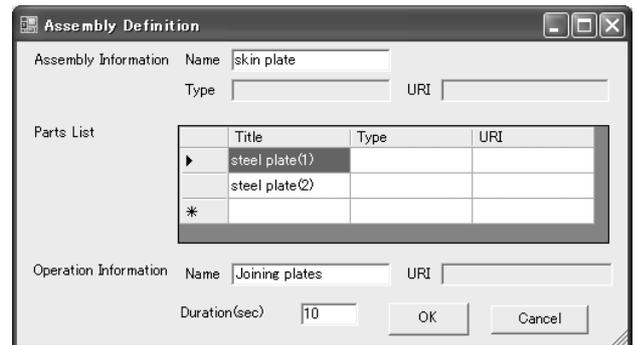


Figure 8: Assembly definition dialog

Figure 9 is the same production process and parts in a tree view of the client program. Output assembly is located in the parent node, and operation and input parts are put as children of the output assembly to make it easy to understand the relationship of each object. The concept of BOM system is introduced to this user interface.

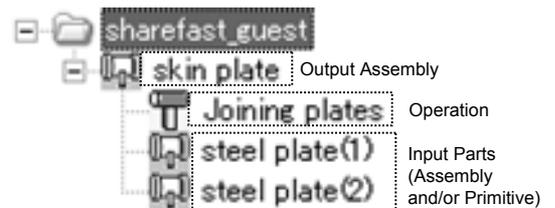


Figure 9: Representation in BOM tree view

3.3 Production process editing

After defining assemblies, design/production engineers have to organize the whole products. When the user edit the parts (“Assembly” and “Primitive”) hierarchy, the production process is also changed simultaneously because the assembly has process information as an instance of “Operation” class.

The client program provides drag and drop feature to edit the hierarchy same as other common software. A drag and drop operation sends a request to the server of updating metadata. Figure 10 shows the hierarchy of defined assemblies, and then the assemblies are organized into one assembly.

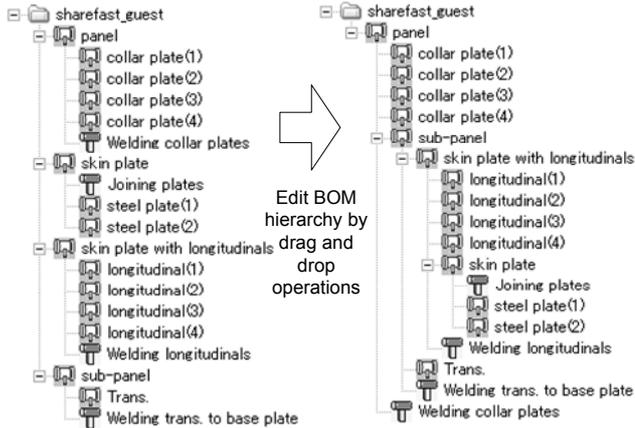


Figure 10: Production process editing

3.4 Generating flow diagram

Production process and parts information are stored in RDF format. The data has lists of parts of each assembly and dependency information of parts. Using the information of the tree view, a flow diagram can be generated. The hierarchy of parts in right side of Figure 10 is converted to flow diagram shown in Figure 11. In this process, four assemblies are fabricated sequentially, so the flow diagram has no branches and one track. Duration defined in assembly definition is put as an annotation in the flow diagram. And the total duration and the duration of critical path are calculated automatically by the system, and shown in the diagram. With this feature, engineers can get flow diagram with duration just after planning production process in graphical user interface like BOM system.

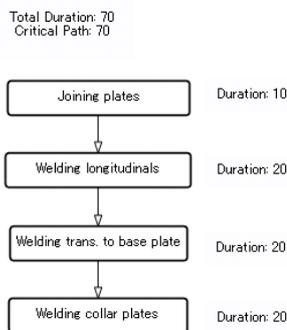


Figure 11: Flow diagram generated by proposed system

4 EXPERIMENT

4.1 Production process for experiment

To evaluate proposed method, a case study of planning sample production process is described by this system. The process is illustrated in Figure 12. A plate, four longitudinals and a transverse member are fabricated to a panel in this process. Two steel plates are joined to form the base plate. Several ways can be applied to fabricate longitudinals and transverse members. Two production processes for the same panel are described in this case study as shown in Figure 12. Duration is assigned to each process and shown in the figure by case arc.

In case (a) in Figure 12, large slits are made in transverse member to weld the transverse member and longitudinals. A transverse member and longitudinals are easier to weld but the slits must be closed after fabrication, so collar plates must be welded. So, the whole process is as follows.

1. Joining plates (10).
2. Welding longitudinals to a base plate (20).
3. Welding a transverse member to a base plate (20).
4. Welding collar plates to slits in transverse member (20).

In case (b) in Figure 11, a transverse member and longitudinals are fabricated to an egg box prior to be welded to the base plate. So the slits in transverse member are not needed. But the welding process of the egg box and the base plate is a complicated operation.

The process of this case is shown as follows.

1. Joining plates (10).
2. Fabricating an egg box (15).
3. Fabricating an egg box and a base plate (40).

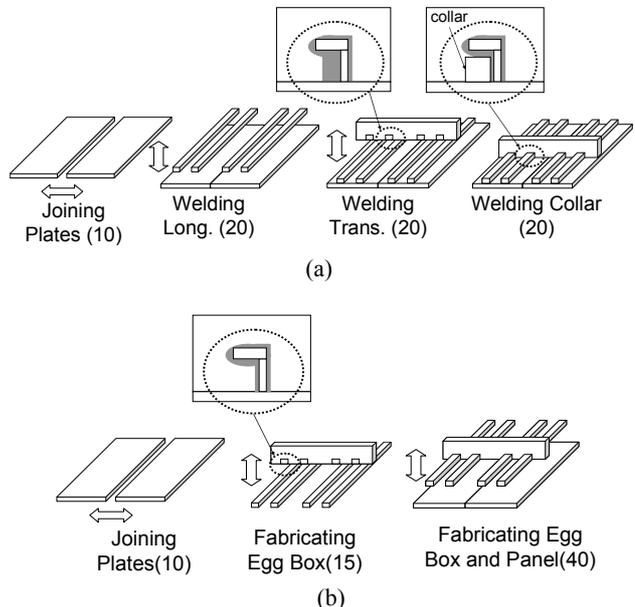
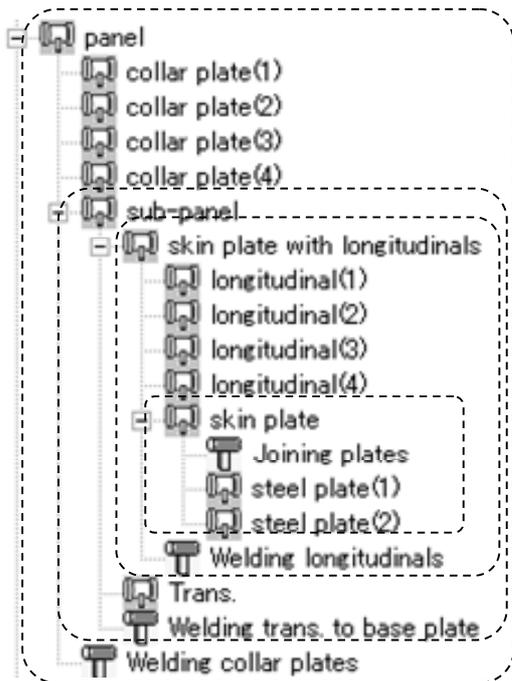


Figure 12: Sample panel production process

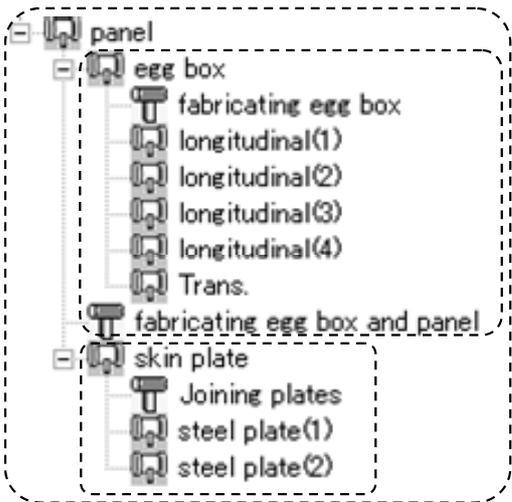
4.2 Description results

Two cases shown in Figure 12 are described by the proposed system. The hierarchy of parts is shown in Figure 13. In case (a), all the intermediate products which are instances of “Assembly” class are input of the next operation. So the four operations must be performed sequentially. By the way, in case (b), two assemblies are fabricated in the first step, and those two assemblies are

joined afterwards. The depth of the hierarchy is smaller than in case (a).



(a)

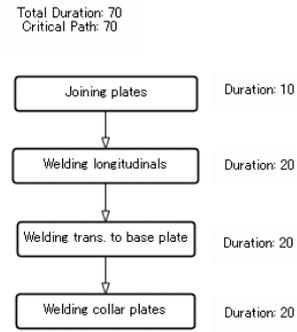


(b)

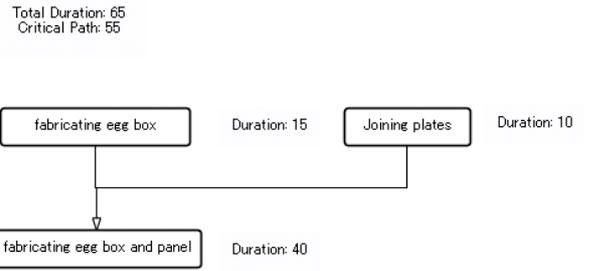
Figure 13: BOM tree for Sample panel production process

4.3 Results of flow diagram generation

Flow diagrams for these two cases are generated by the proposed program. The diagrams are shown in Figure 14. The flow diagram of case (a) is straight forward and there are no branches as explained in section 4.1. Total duration is 70 and the duration of critical path is also 70. As for case (b), the flow diagram has a branch and “fabricating egg box” and “Joining plates” can be performed simultaneously. The total duration is 65 and the duration of critical path is 55. Flow diagram can be generated using the result of editing the BOM tree.



(a)



(b)

Figure 14: Flow diagram generated by the system

4.4 Summary

Two cases can be described by this software and the translation from BOM tree to flow diagram is demonstrated. The tree view has dependency information of processes and duration of each task. Engineer can get the information of the flow diagram and duration of the product when planning the production process.

5 DISCUSSION

Proposed method introduces only three classes and three properties to describe production process. This simple data schema can describe dependency of assemblies and also basic process information. In this case study, very limited information is stored in the server and the generated flow diagram is not very large. But the case study shows editing the data for product configuration changes the flow diagram of the production process. This mechanism to provide multiple views for single data is evaluated in this experiment. Using this system, Engineers can create parts lists by the tree view.

UML [11] and IDEF are useful, but these methodologies are too difficult for engineers to describe the production process in most cases. BOM system is common to production process engineers, so the system adopts the graphical user interface similar to BOM system. Engineers are expected to use the software by themselves.

The proposed system can describe dependency relationships between production process and parts in single database thanks to ontology based on RDF and RDF schema. This integrated description format can be extended to many other applications.

6 CONCLUSION

A method for describing production process based on RDF schema is proposed. And computer program for describing the production process is developed. Sample production process of shipbuilding is described by the system and proved that a BOM tree with process

information can be translated to a flow diagram. The system tells the users about the total duration and critical path.

7 FUTURE WORK

The prototype system must be applied to large production process to evaluate if the system is useful for detecting problem in production process. Visualization of a BOM tree and a flow diagram might be useful, but must be evaluated in real production design process.

The data is stored in RDF format. More complicated and detailed modelling of production process and parts can be done by defining other properties by RDF schema. As URI are assigned to all the parts in the products, this system can work with existing information systems if appropriate properties related to CAD or BOM system are defined by RDF schema.

8 ACKNOWLEDGEMENTS

This work was supported by Grant-in-Aid for Young Scientists (B) No. 20760556. We'd like to thank engineers in shipyards for their valuable discussion.

9 REFERENCES

- [1] New Energy and Industrial Technology Development Organization, Japan, 2005, 'Product Fabrication and IT Integration Promotion Technology (Digital Meister Project)', NEDO
- [2] Guus Schreiber, Hans Akkermans, Anjo Anjewierden, Robert de Hoog, Nigel Shadbolt, Walter Van de Velde, and Bot Wielinga, 2000, 'Knowledge Engineering and Management The CommonKADS Methodology', The MIT Press
- [3] Klyne, G., Carrol, J. (eds.), 2004, Resource Description Framework (RDF): Concept and Abstract Syntax, W3C Recommendation, <http://www.w3.org/TR/rdf-concepts/>
- [4] National Institute of Standards and Technology , 1993, INTEGRATION DEFINITION FOR FUNCTION MODELING (IDEF0)
- [5] Dave Garwood, 1997, Bills of Material, Dogwood
- [6] DCMI (Dublin Core Metadata Initiative), 2003, Dublin Core Metadata Initiative, Dublin Core Metadata Element Set, Version 1.1: Reference Description, DCMI Recommendation, <http://dublincore.org/documents/dces/>
- [7] Brickley D, Guha R., 2004, RDF Vocabulary Description Language 1.0: RDF Schema., <http://www.w3.org/TR/rdf-schema/>
- [8] Hiekata K, Yamato H, Oishi W, Nakazawa T., 2007, Ship Design Workflow Management by ShareFast, Journal of Ship Production, 23(1), 23-29.
- [9] Hewlett-Packard Development Company, 2005, 'Jena 2 - A Semantic Web Framework', Hewlett-Packard Development Company
- [10] B. McBride, 2001, 'Jena: Implementing the RDF Model and Syntax Specification', Proceedings of the Second International Workshop on the Semantic Web
- [11] Grady Booch, James Rumbaugh , Ivar Jacobson, The Unified Modeling Language User Guide 2nd Edition, 2005, Addison-Wesley Professional