DEVELOPMENT OF A FEATURE LIBRARY FOR THE EXTRACTION OF MANUFACTURING FEATURE AND MANUFACTURING INFORMATION

Hendry Muljadi[1], Hideaki Takeda[1], Koichi Ando[2]
[1]National Institute of Informatics, Japan
[2]Shibaura Institute of Technology, Japan

ABSTRACT

In a manufacturing feature-based process planning system, for the generation of process plans, it is necessary to develop a feature library that consists of pre-defined manufacturing features and the manufacturing information to create the shape of the features. This paper describes the development of a feature library for a process planning system. The feature library is developed to enable the automatic extraction of manufacturing features with their proper manufacturing information that will lead to the generation of process plans. In this paper, the implementation of Wiki for the development of the feature library is proposed.

1. INTRODUCTION

Computer Aided Process Planning (CAPP) integrates the automation of product design with that of manufacturing by linking the design representation of Computer Aided Design (CAD) systems with the manufacturing process representation of Computer Aided Manufacturing (CAM) system. In order to allow the CAD system to interface with the CAPP system, feature technology has emerged as the enabling technology to convert CAD product data to manufacturing information. Many different approaches have been developed to extract features from the CAD product data (Subrahmanyam 2002).

In a manufacturing feature-based process planning system, for the generation of process plans, it is necessary to develop a feature library that consists of pre-defined manufacturing features and the manufacturing information to create the shape of the features (Kanamaru et al 2004). The manufacturing information consists of the required machine and tool data, the estimated cost and time data, etc (Scallan 2004). Considering the rapid progress of manufacturing technologies, a feature library needs to be developed in a way that is easy to modify or to customize.

On the other hand, in the context of collaboration on the web, Wiki has proven itself to be an easy to modify, user-friendly interface. For example, the community of Wikipedia.org, the free content encyclopedia is becoming larger and larger. There have been more than 600,000 people who gave their contributions, either by creating or editing articles in Wikipedia. Wiki is a discussion medium, a repository of ideas and a tool for collaboration. It is a simple publishing system that is easy to learn and quick to use (Leuf et al 2001). In Wiki, people can create or edit a Wiki page using a simple syntax to write content. So, it is normal to make an assumption that an extended Wiki will be useful for the development of a feature library which can be easily modified.

In this paper, the modification of MediaWiki, the Wiki software used for the Wikipedia, and the implementation of the modified MediaWiki for the development of the feature library is described.

The structure of this paper is as follows. Section 2 discusses the need to consider the designer's intention for the realization of the automatic extraction of manufacturing features with their proper manufacturing information. In section 3, the structure of the feature library to enable the automatic extraction of manufacturing features with their proper manufacturing information is described. Section 4 describes the modification of MediaWiki and the implementation of the modified MediaWiki for the development of the feature library. Section 5 states the conclusions drawn from the research.

2. CONSIDERING THE DESIGNER'S INTENTION FOR THE AUTOMATIC EXTRACTION OF MANUFACTURING FEATURES WITH THEIR PROPER MANUFACTURING INFORMATION

A manufacturing feature can be defined simply as a geometric shape and its manufacturing information to create the shape. However, most feature recognition methods only deal with the automatic extraction of geometric shape from the CAD product data, and do not deal with the automatic extraction of proper manufacturing information to create the shape.

For the extraction of proper manufacturing information to create the shape of the manufacturing features, it is important to understand the designer's intention. For example, a thru-hole feature may require a cylindrical grinding to create the shape, while the other thru-hole feature may require threading to create the shape, depending on why the designer designed the geometric shape.

In this research, the designer's intention is represented by the functions of the face elements that construct the features. The face element is defined as a geometrical entity that is bounded by a set of edges. The functional data of the face elements can be described as basic function, mechanism utilized for realization of the basic function, and condition and direction of the motion. The detail explanation of the functional data elements is given in other reports (Yoshikawa et al 1987) (Ando et al 1989). Table 1 shows the contents of the functional properties of face elements that are used in this research.

Fig.1 shows an object with a screw thru-hole. Table 2 shows the representation of function of the face elements that construct the screw hole. The basic function is fixation of motion. The mechanism utilized for realization of the basic function is bolt-only. The condition and direction of the motion is stationary-object.

Fig.2 illustrates that by dealing with the geometrical data and the functional data of the product design data, the automatic extraction of manufacturing features with their proper manufacturing information can be realized (Muljadi et al 2004).

Table 1. Contents of functional properties

Basic Function	Mechanism utilized for realization of the basic function	Condition and direction of the motion
Transmission of motion	1: friction-mech. 2: gear-mech. 3: link-mech. 4: cam-mech.	1: liner 2: smooth-liner 3: very-smooth-liner 4: round 5: smooth round 6: very smooth round
Constraint of motion	1: rigidity-mech. 2:ball-bearing-mech. 3: sliding-mech.	1: liner 2: weak-radial 3: strong-radial 4: weak-thrust 5: strong-thrust
Fixation of motion	1: bolt-and-nut 2: bolt-only 3: friction-mech. 4: bearing-fit 5: key-fit 6: rivet-fit 7: shrinkage-fit	1: stationary-object 2:revolutionary-object

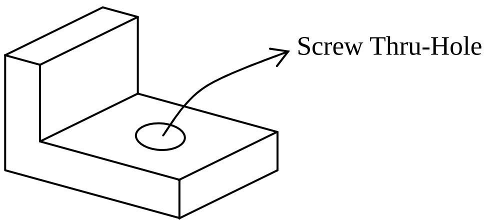

Fig. 1 An object with a screw thru-hole

Table 2. The representation of function of the face elements that construct the screw hole

Basic Function	Mechanism utilized for realization of the basic function	Condition and direction of the motion
Fixation of motion	bolt-only	stationary-object

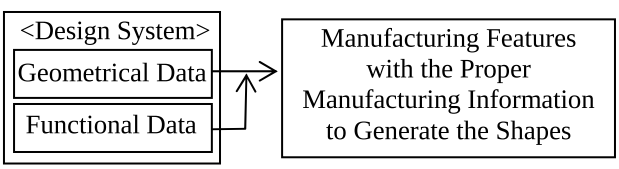

Fig.2 Geometrical data and functional data for the automatic extraction of manufacturing features with their proper manufacturing information

3. THE STRUCTURE OF THE FEATURE LIBRARY

As mentioned in the previous section, for the automatic extraction of manufacturing information to create the shape of the manufacturing features, it is important to understand the designer's intention, which is represented by the functions of the face elements that construct the features. For the development of the feature library, the use of function feature ontology is proposed. A function feature is defined here as a geometric shape and its functions as intended by the designer.

Fig. 3 shows the ontology of function features, the manufacturing feature ontology, and the relation between the two ontologies.

For the creation of the function feature ontology, first, features such as thru-slot, step etc are listed up. This research uses the list of features proposed in (Butterfield et al 1988). Then sub-classes of these features are created by describing the required function of the face elements that construct the features.

For the creation of the manufacturing feature ontology, first, manufacturing features such as thru-slot, step etc are listed up. Sub-classes of these manufacturing features are created by describing the general manufacturing methods to create the parent classes. Sub-classes of these sub-classes are created by describing the more specific manufacturing methods to create the manufacturing feature. For example, a round thru hole feature can be manufactured by drilling, milling etc. So, the sub-classes of the round thru-hole feature will be "milled thru-hole" feature class, "drilled thru-hole" feature class etc. The sub-classes of the "drilled thru-hole" feature class will be "threaded drilled thru-hole" feature, "precise drilled thru-hole" feature etc.

The relation between the classes in the lowest level of the manufacturing feature ontology and the function feature ontology represents how the manufacturing features should be manufactured to fulfill the required functions of the face elements that construct the manufacturing features. For example, a "threaded drilled thru-hole" class of the manufacturing feature ontology can have a relation with the "round thru-hole with both face elements require: Basic function: fixation. Mechanism:bolt-only. Motion: stationary-object" class of the function feature ontology, since threading is the manufacturing method used to create the round thru-hole with the intended functions.

Instances of the classes in the lowest level of the manufacturing feature ontology should also be created, where each instances should have its own unique manufacturing information. The creation of the instances of the manufacturing feature classes can be done by considering the specific size of the manufacturing features.

By developing a feature library based on the proposed structure, the feature library can be useful for the automatic extraction of manufacturing features with their proper manufacturing information. For example, when a round thru-hole feature is extracted by a feature recognition method, and the functional properties of the face elements that construct the round thru-hole feature lead to the extraction of the "threaded drilled thru-hole" feature class, a proper manufacturing information can be extracted automatically from the instance of the manufacturing feature class.

The automatic extraction of manufacturing features with their proper manufacturing information is a

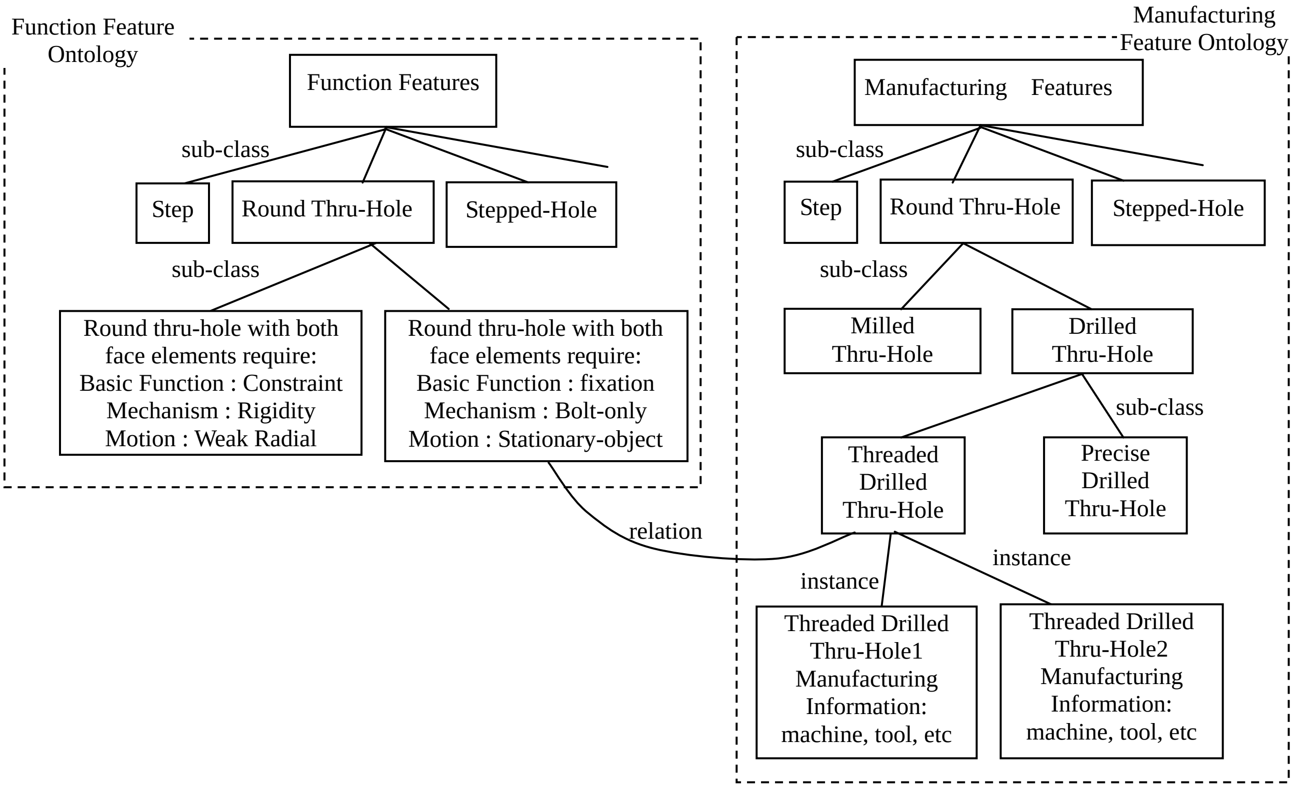

Fig.3 The structure of the feature library

significant step for the realization of the automatic generation of process plans.

4. THE WIKI-BASED FEATURE LIBRARY

4.1 SEMANTIC EXTENSION OF MEDIAWIKI

MediaWiki is a Wiki software that is written in PHP and uses MySQL database. It is being used to run the Wikipedia and also other web-based encyclopedias and dictionaries. MediaWiki is a very useful tool for the collaborative content management.

An extension of MediaWiki to enable the writing of the labeled link has been proposed in other report (Muljadi et al 2005). This extension has enabled the Wiki to write Resource Description Framework (RDF) statement, which consists of subject-predicate-object triple. RDF is a language to express metadata about information resource on the Web proposed by the WWW Consortium (W3C) (Miller et al 2004). RDF has a simple data model that is understandable by human and is easy for computer applications to process and manipulate.

The Wiki syntax to write the RDF triple is [[Term: target_page|property]]. The RDF triple is <source_page> <property> <target_page>. Each time the Wiki syntax is used, the Wiki engine will store the RDF triple into a table in the Wiki database. By directly querying the table, the labeled link relation will be displayed as follows.

1) On the source_page: -> property -> target_page
2) On the target_page: <- property <- source_page
3) On the property: source_page -> target_page

Fig.4 illustrates the RDF triple construction and the relation of pages displayed on the Wiki pages in the extended MediaWiki. Fig.5 shows the overall structure of the extended MediaWiki. The extended MediaWiki as an extension of MediaWiki has the benefit of having all the functions available in MediaWiki as a content management system, and can be used as an editor of metadata according to simple RDF statement. The RDF triples which are stored in a table, can be exported to RDF Database for mapping to other Semantic Web applications. The running system of the extended MediaWiki is available at http://semanticwiki.jp.

The development of the Wiki-based feature library is based on the extended MediaWiki proposed in (Muljadi et al 2005).

4.2 DEVELOPING A WIKI-BASED FEATURE LIBRARY

For the development of the Wiki-based feature library, further extension of MediaWiki is done. New namespaces are created. Namespace (“FF:”) is created to deal with the function feature ontology, and namespace (“MF:”) for the manufacturing feature ontology. New tables are also created in the Wiki database to deal with the new namespaces.

For the creation of the function feature ontology, the Wiki syntax [[FF:feature_subclass|subclass]] is used (see Fig.6). When the Wiki syntax is written on the parent class page, the Wiki engine will store the RDF triple into a table which deals with the namespace (“FF:”) in the Wiki database. By directly querying the table, the labeled link relation will be displayed as follows.

1) On the parent class page: -> subclass -> feature_ subclass (see Fig.7).
2) On the feature_subclass page: <- subclass <- parent class (see Fig.8).
3) On the subclass page: parent class -> feature_subclass (see Fig.9).

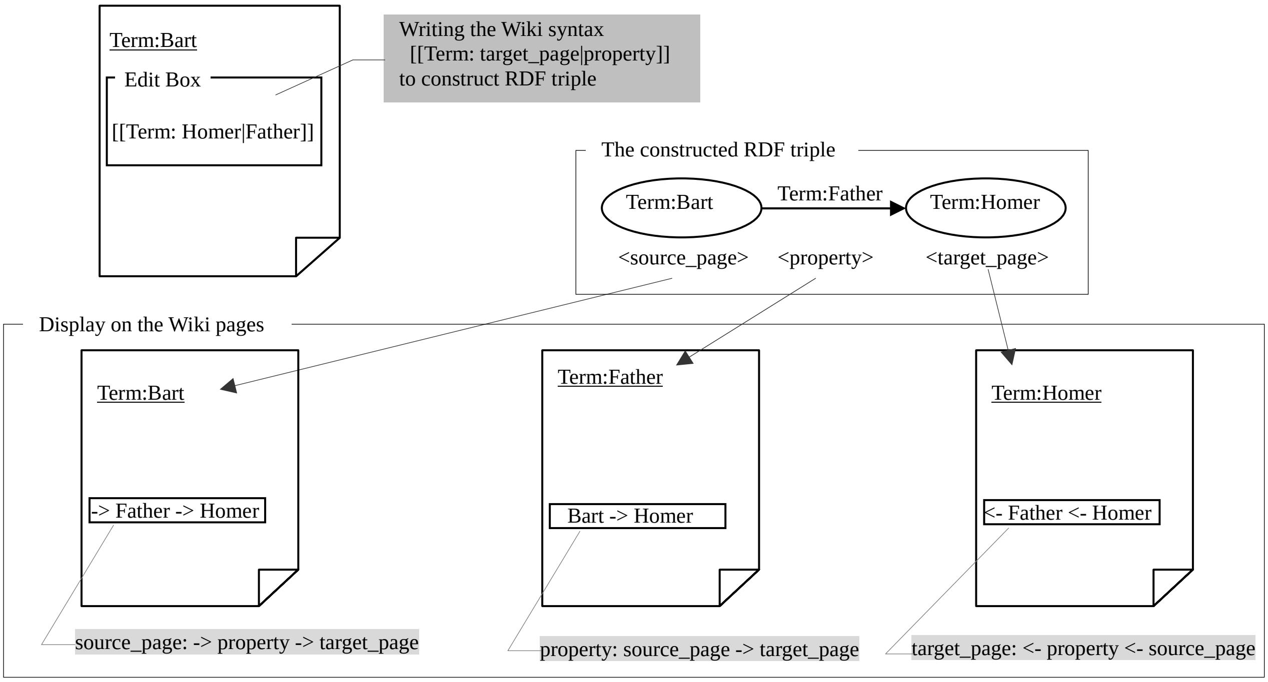

Fig.4 RDF triple construction and the display on the Wiki pages in the extended MediaWiki

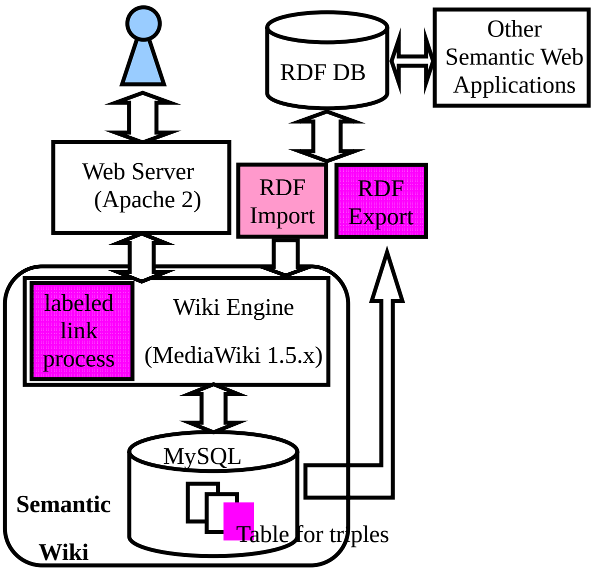

Fig. 5 Overall structuore of the extended

Fig.7 Display on the parent class page

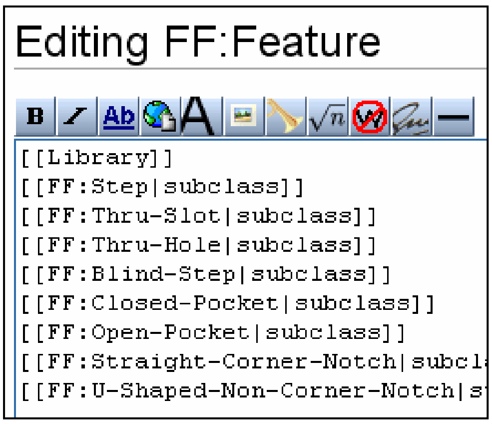

Fig.6 Writing the Wiki syntax
[[FF:feature_subclass|subclass]]

Fig. 8 Display on the feature_subclass page

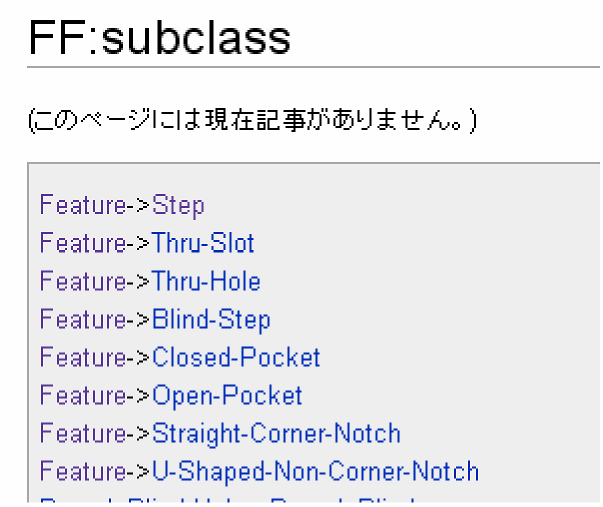

Fig. 9 Display on the "FF:subclass" page

The "FF:subclass" page can be used to see all the class-sub-class relations of the function feature ontology.

Fig.10 illustrates the page relations in the manufacturing feature ontology. For the creation of class-sub-class relations in the manufacturing feature ontology, the Wiki syntax [[MF:feature_subclass| subclass]] is used. When the Wiki syntax is written on the parent class page, the Wiki engine will display the labeled link relations in the same way as in the function feature ontology. The "MF:subclass" page can be used to see all the class-sub-class relations of the manufacturing feature ontology. For the class-instance relation in the manufacturing feature ontology, the Wiki syntax [[Term: instance_page|property]] can be used. Feature size etc can be used as the property. When the Wiki syntax is written on the manufacturing_feature_class page, the Wiki engine will display the labeled link relation as follows.

1) On the manufacturing_feature_class page: -> property -> instance_page
2) On the instance page: <- property <- manufacturing_ feature_class
3) On the property page: manufacturing_feature_class -> Instance_page

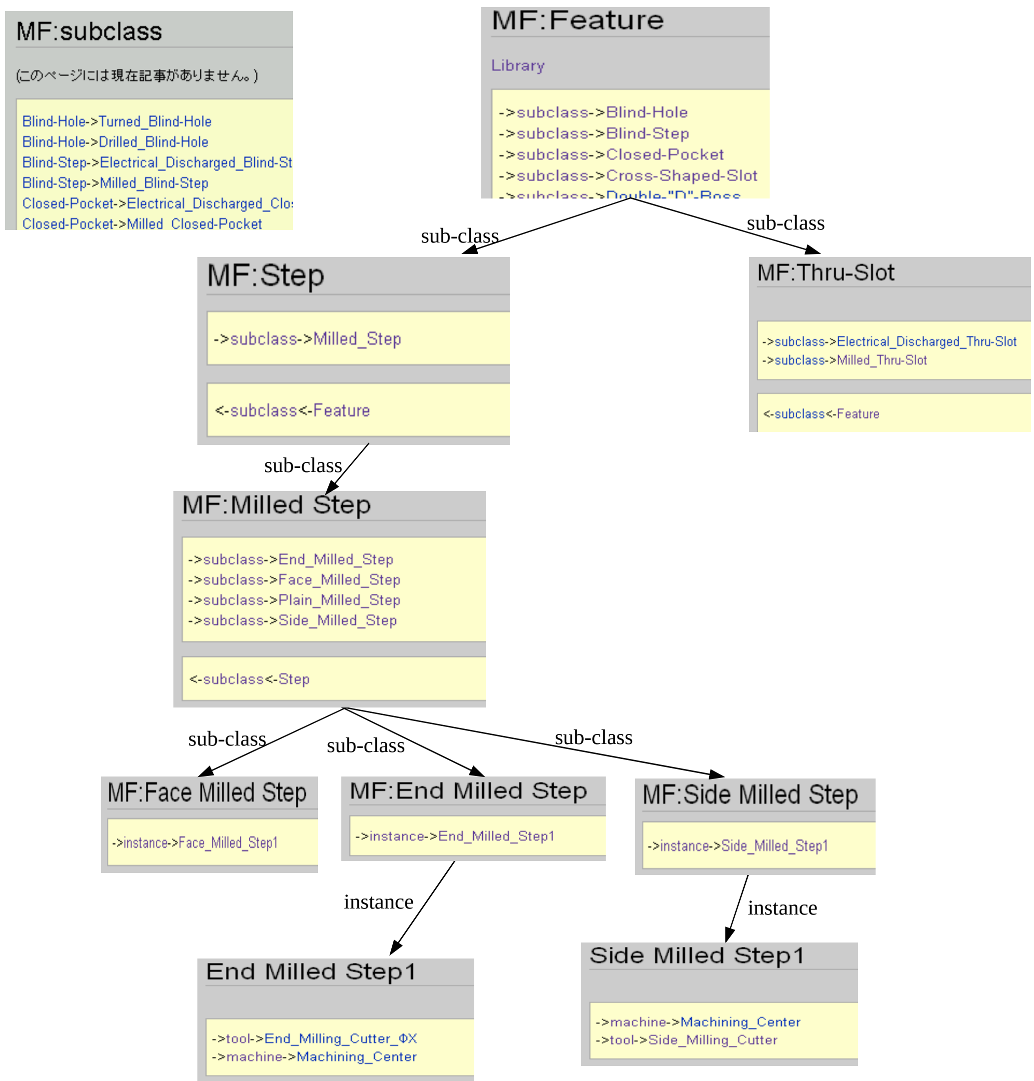

Fig.10 Page relations in the manufacturing feature ontology

To write the manufacturing information such as machine types on the instance page, the Wiki syntax [[Term:machine_name|machine]] is used. Fig.11 and Fig.12 show the Wiki syntax writing and the display on the instance page respectively.

To make the relation between the lowest sub-class of the function feature ontology and the lowest sub-class of the manufacturing feature ontology, the Wiki syntax [[MF:manufacturing_feature_class|related]] is used. Fig.13 shows the Wiki syntax writing on the function_feature_class page. When the Wiki syntax is written on the function_feature_class page, the Wiki engine will display the labeled link relation as follows.

1) On the function_feature_class page: -> related -> manufacturing_feature_class (see Fig.14)
2) On the manufacturing_feature_class page: <- related <- function_feature_class (see Fig.15)
3) On the related page: function_feature_class -> manufacturing_feature_class (see Fig.16)

The “MF:related” page can be used to see all the relations between the lowest sub-class of the function feature ontology and the lowest sub-class of the manufacturing feature ontology.

The Wiki-based feature library is able to construct the function feature ontology, the manufacturing feature ontology, and the relation between the two ontologies. It also can be used to manage the manufacturing information to create the manufacturing feature. The Wiki-based feature library can be easily, visibly and collaboratively modified. And since the page relation can be represented in an RDF triple representation, the page relation can be processed automatically by computer applications. Thus, the Wiki-based feature library can be used to enable the automatic extraction of manufacturing features and their proper manufacturing information.

Fig.11 Writing the manufacturing information on the instance page

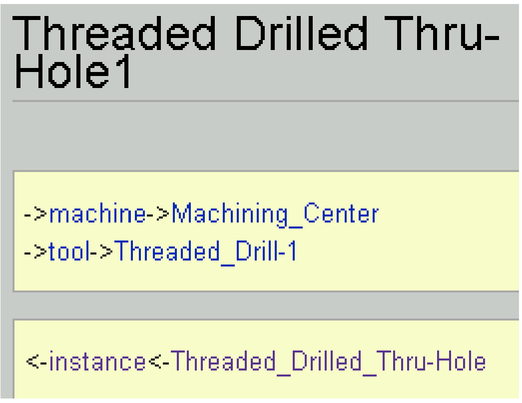

Fig.12 The display on the instance page

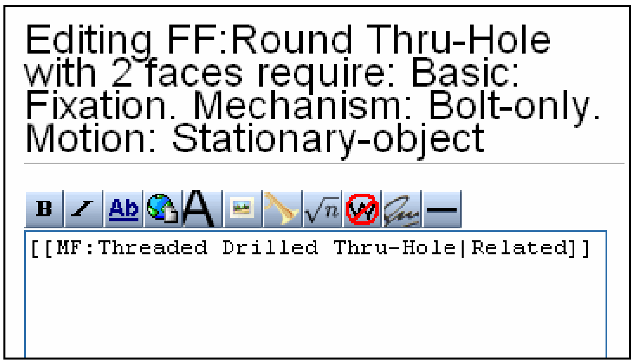

Fig.13 Writing the Wiki syntax [[MF:manufacturing_feature_class|related]]

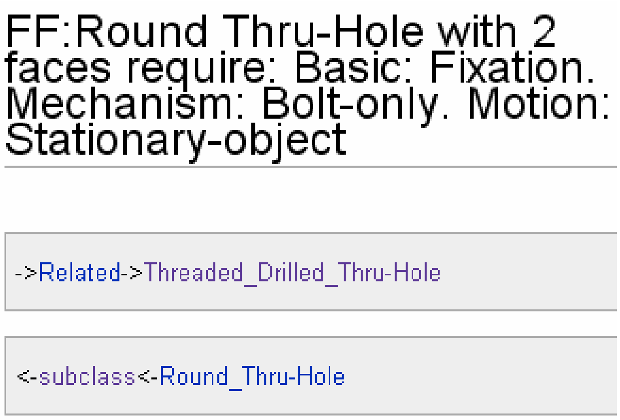

Fig.14 Display on the lowest sub-class of the function feature ontology

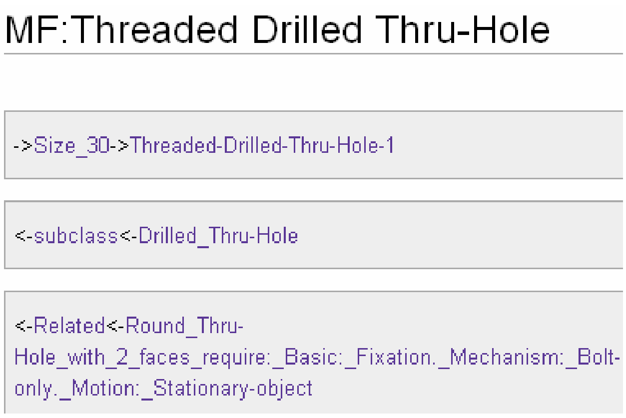

Fig.15 Display on the lowest sub-class of the manufacturing feature ontology

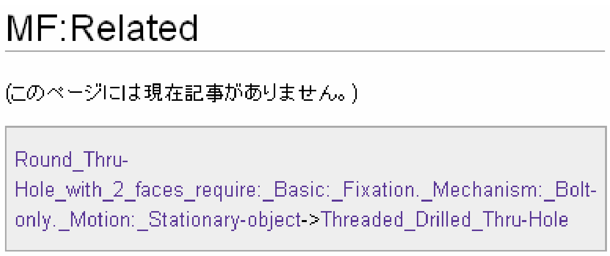

Fig.16 Display on the “MF:Related” page

5 CONCLUSION

This research can be summarized as follows.

1) The feature library consists of the function feature ontology and the manufacturing feature ontology. The relation between the classes in the lowest level of the manufacturing feature ontology and the function feature ontology represents how the manufacturing features should be manufactured to fulfill the required functions of the face elements that construct the manufacturing feature. By developing the feature library based on the proposed structure, the extraction of manufacturing features with their proper manufacturing information for the generation of process plans is made possible.
2) MediaWiki is modified for the development of the feature library. The Wiki-based feature library is able to construct the function feature ontology, the manufacturing feature ontology, and the relation between the two ontologies. It also can be used to manage the manufacturing information to create the manufacturing feature. The Wiki-based feature library is a very simple system, and as one tries to modify the feature library, one may enjoy the visible modification of the feature library. And since the page relations are written in RDF triples, the page relations can be processed automatically by computer applications. Thus, the Wiki-based feature library can be used to enable the automatic extraction of manufacturing features and their manufacturing information.

REFERENCES

Ando K., Yoshikawa H., (1989), “Generation of manufacturing information in Intelligent CAD”, *Annals of CIRP*, Vol.38, No.1, p.133-136.

Butterfield W.R., Green M.K., Scott D.C., Stoker W.J., (1988), “Part Features for Process Planning” , *Computer Aided Manufacturing International (CAM-I)*, Document R-86-PPP-01.

Kanamaru M., Ando K., Muljadi H., Ogawa M., (2004), “Manufacturing feature library for the machining process planning”, *Proceedings of the Japan Society of Precision Engineering Autumn Conference*, CD-ROM C-74 (in Japanese).

Leuf B., Cunningham W., (2001), *The Wiki Way: Quick Collaboration on the Web*, Boston: Addison-Wesley.

Miller E., Swick R., Brickley D., (2004), “Resource Description Framework (RDF), http://www.w3.org/RDF/

Muljadi H., Ando K., Takeda H., (2004), “Considering designer’s intention for the extraction of manufacturing feature”, *Proceedings of the 18th International Conference on Production Research*, CD-ROM.

Muljadi H., Takeda H., (2005), “Semantic Wiki as an integrated content and metadata management system”, *Poster & Demonstration Proceedings of the 4th International Semantic Web Conference*, PID-44.

Scallan P., (2004), *Process Planning: The Design/ Manufacture Interface*, Oxford:Butterworth-Heinemann.

Subrahmanyam S.R., (2002),“A method for generation of machining and fixturing features from design features”, *Computer in Industry*, Vol.47, No.3, p. 269-287.

Yoshikawa H., Ando K., (1987), “Intelligent CAD in manufacturing,” *Annals of CIRP*, Vol.36, No.1, p. 77-80.