WEB SERVICE ORIENTED STANDARD PRODUCT LIBRARY

Ma, Y. S.[1*], Rajeshbabu, T. S.[1] and Deng, Y. M.[2]
[1]*School of MAE*
Nanyang Technological University
Singapore
[2]*Faculty of Engineering*
Ningbo University
Zhejiang, China

ABSTRACT

Using standard components or modules is vital in product design, development and manufacturing. The current book type catalogues and those, in the form of electronic files together with CAD drawings distributed on disks or via web-based publication systems, can not meet industry requirements for collaborative engineering. Ideally, users could source competitive offers from different vendors and use their information over the Internet. The drawbacks in the current catalogues are platform dependency, and difficulties in version management, CAD file translation, and catalogue content updating. In this research, a novel "Web Service" oriented approach is proposed to address these limitations. Based on an in-house standard component and assembly library, a new method is proposed to engage a CAD modeler with Java Web Service technology. With such an approach, all the aforementioned drawbacks are eliminated. Moreover, the new method avoids hard coding of catalogues within CAD systems, facilitates the catalogue vendors to update their databases at any time. Taking injection molding design as an example, the authors discussed key mechanisms in its implementation and the feasibility of industrial applications.

Keywords: Web Service; Collaborative Engineering, Standard Product Library, CAD

1. INTRODUCTION

New product introduction (NPI) with short lead-time is vital among OEMs (Original Equipment Manufacturers) for competitiveness and profitability. Due to globalization of economies, new products are often developed in various countries with increasing use of standard components or modules because in such a way product development cycle time and cost can be reduced. On the other hand, internal machining workshops and external suppliers have to follow different sets of standard part or module requirements from different customers and for the different product structures. CIMdata proposed a blueprint for product lifecycle management strategies in a collaborative approach [5] in which collaborative product commerce is emphasized. It is suggested that purchasing decision making and ordering for standard components or external modules should be carried out at the time of product design. This demand motivates vendors to release their catalogues in a more effective way, comparing to the present book type catalogues or the existing electronic distribution. To do so, it is essential to have a common system, which seamlessly gives the most updated information of standard products from as many providers as possible; such information includes prices, ordering procedures, views of components (pictures), and generic 3D models that could be directly used and edited with different CAD systems.

The objective of this research is to study the feasibility of a reference architecture that links the existing in-house electronic catalogue system with Java Web Service technology. Theoretically, the proposed system can provide services to end users with the choices of selection among different catalogues. By using the Java Web Service technology, common architecture between catalogue providers and the end users across different platforms over the Internet can be achieved; so that, third-party developers and the end users can integrate their applications with the standard components or modules provided from multiple suppliers. With the proposed system architecture, flexible customized CAD models can be delivered efficiently instead of hard-coded and CAD specific catalogues. The system also facilitates the catalogue vendors to update their databases at any time hence misleading and non-updated catalogue versions can be avoided.

* **Corresponding author: mysma@ntu.edu.sg**

In the next section, the drawbacks of the existing component/product libraries/catalogues are discussed. Subsequently a Web service oriented method is proposed. A prototype system is then introduced which was developed based on a standard component library for injection mold design. Its implementation demonstrates the feasibility and practicability of the proposed method.

2. REVIEW OF THE EXISTING APPROACHES

Standard component libraries are very useful for reusing design knowledge and engineering data, and reducing product development time and cost [6]. Traditionally, suppliers deliver book type catalogues to the product designers, which have to be printed once in every 2-3 years in order to update the designs, configurations, prices and ordering or delivering information. Book type catalogues are still commonly used for publishing the standard components and updating the relevant information periodically. The catalogue updating is scheduled based on the number of new or modified components to be introduced; it also depends on the company's growth. More recently, suppliers provide CAD specific electronic libraries in the form of files delivered on disks, or via internet WEB portals. There are some major disadvantages in the current practices as follows.

- Manufacturing companies accept catalogues often from the major vendors and thus are not friendly for the new entries and niche component providers.
- It is difficult for the catalogue providers to publish and market their new products timely due to the periodical updates.
- The designer needs to create the CAD models by themselves in their designs; this task is a very time-consuming and sometimes hurdles the designer to use third party standard products.
- Many small companies support major vendors with standard components according to the customers' preferences. Thus, they must create different CAD models according to different vendors, and they have to maintain expensive customized CAD models and different CAD software packages.
- There is a high possibility of using the outdated catalogues in end users' side.
- Catalogue vendors face difficulty in changing the information of catalogue items before the next update.
- Finally, huge cost is involved in the catalogue book development and distribution all over the world.

Computer-based libraries are definitely superior in these aspects [6]. Several design packages and commercial Web sites offer component libraries recently [2][18]. Almost all of them provide predefined 3D components only, not parametrically designed with explicit features. Most of them are developed either in IGES, STEP or other neutral formats or for one specific CAD system [13][14]. They generate or load-in components in solid models with predefined geometry and dimensions, and are very limited in editing functions; this limitation makes them difficult to be adopted by different vendors due to the enormous variations in component definitions. We refer such libraries and CAD models as hard-coded because they cannot be used in a flexible manner [14]. Due to the numerous sizes and configurations, such hard coded method results in numerous CAD files and they are not easily editable according to key patterns [16].

Some CAD vendors implement standard components from major catalogue providers as an add-on module of their systems. We call this method as "CAD system approach". As can be seen, although this approach reduces the excessive work and investment needed by end users to develop CAD models for standard components or modules, there are still major disadvantages as follows.

- For updating catalogue models, suppliers depend on the CAD vendors or developers. Usually CAD vendors care only their CAD software updates while developers demand extra significant resources.
- CAD vendors implement only the products of major suppliers but not the small or niche ones.
- CAD vendors implement standard component or module CAD models for the purpose of making it easier to use their software and eventually for their CAD system marketing purpose. Thus, the suppliers have no control on this type of catalogues.
- Very often, due to the cost of development involved, CAD vendors implement very limited standard products from different catalogues. There is no interest for them to update those catalogues when there is any updating requirement initiated from suppliers.
- Since it is the CAD vendors who implement the catalogues, they validate the catalogues by themselves. The standard product supplier does not have control or validation for the libraries. Very much non-geometric information, such as design patterns, feature configuration, ordering forms, prices, etc, is striped off.
- Due to the hard-coded approach, there is a limitation of space set by the CAD vendors for each type of catalogues from different suppliers.
- Each CAD vendor develops its standard component or module library in its own CAD platform; this practice limits the usage of the implementation only to their customers.
- If end users or suppliers do not update, or invest in, their CAD systems to obtain the

newest version, then the design may end up to using outdated libraries.

A few suppliers developed their catalogues and published them in the Internet. We refer to this practice as "Web-based approach". There are two types of implementation; the first type refers to those being only catalogue text contents downloading systems while the other with the corresponding CAD model libraries.

In the catalogue-only downloading system, catalogue vendors publish their catalogues documents on their website instead of book type and inform end-users by email or other means of notification. End users download the catalogue from the catalogue vendor's website. The purpose of using this system is to avoid using the outdated version and to reduce the usage of book type catalogues. However, there are still some major disadvantages:

- End users are likely to overlook the updates.
- End users only get catalogue and price details but not the CAD model library that can be directly used in their specific CAD systems. So the end users need to either create their own CAD models or still depend on the CAD vendors to update their CAD system libraries.

The second type, with web-based CAD model libraries, is relatively new where all the data and CAD models are implemented with a database system, and interfaced with user interfaces in the website. This method is flexible for the catalogue providers to add/modify components easily [20]. End users can select and get details of the required components or modules by browsing the Web site; they can download the information relating to the catalogue and the CAD models which can be used for their downstream application. Even so, the following observations are made:

- It is difficult to provide CAD models for different CAD platforms available in the market. Usually, the CAD models are hard-coded. Only those users who have the supported CAD systems could use the CAD models for their applications. Although it is possible to provide generic CAD objects in neutral format like STEP files, but the end user do not willing to accept the format due to the well-known interoperability problem. To avoid this, native CAD models are preferred. Commonly, all these formats can only support pure geometry; hence, they are 'dead blocks' without engineering and business semantics.
- There are still problems, if the CAD models of libraries are developed in a neutral format (e.g. STEP, ACIS), because the neutral format definition scheme is followed closely for creating generic CAD objects, and whenever the neutral format provider makes some changes, Web data contents and model generators have to be revamped thoroughly. Also, end users need to update their CAD design models as well as system versions in order to avoid translation problems.
- Regardless the CAD model format (e.g. STEP, ACIS) that is used, deep knowledge and skills are required for the end users to extract the required data and to associate with their downstream applications.
- There is tremendous pressure on end users to follow up with the updates even though they are informed.
- Standard component suppliers need to invest huge amount of resources to develop and maintain the web-based catalogue system.

To conclude, the drawbacks in existing standard component or product libraries are the platform dependency, the difficulties in version management, heavy load of CAD file translation, and catalogue content updating.

To address these limitations, this research uses a new technology, namely "Web Service-Oriented" product library approach. This method leverages some of the authors' previous research results [12][14][11]. The first one is the CSG-based architecture for standard component library [12] where the group technology method was adopted. The primitives of standard component features from different supplier catalogues are modeled in a matrix form with particular reference to a classification coding system. Such a matrix scheme is designed for identifying and grouping related or similar shape features in order to take advantage of their similarities in generating the final geometrical model. The second related work is the standard component library (SCL) for mould design specifically [14]. The SCL allows designers to select, load, identify and edit standard components. The component representation includes rich information, such as suppliers, major types, sub-types, alterations, sizes, constraints, tolerances, etc. It is flexible to deal with both dimensional and topological variations of CAD objects. The third related work is the standard assembly library implemented for modular mechanical development [13]. The concepts used in these researches are feature-based rich CAD objects which provide good engineering informatics support for the current work [16].

3. SYSTEM DESIGN AND ARCHITECTURE

3.1 Web Services

The term "Web service" describes a standardized way of integrating web-based applications using XML (Extensible Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Description, Discovery and

Integration) open standards over an Internet protocol backbone. A major difference between web application and web service is that web application is a "user to program" type application, but web service is a "program to program" type application. XML is used to tag the data; SOAP is used to transfer the data; WSDL is used for describing the services; and UDDI is used for listing what services are available. Unlike traditional client/server models, such as a web server/web page system, web services do not provide the user with a GUI. Web services instead share business logic, data and processes through a programmatic interface across networks. The developers can associate the web service to a GUI (such as a web page or an executable program), which is one aspect of the current work, to offer specific functionality to users [19][21][22].

Web services are becoming the basis for electronic commerce of all forms [8]. In manufacturing enterprises, web services have great potential [9]. Companies invoke the services of other companies to accomplish a business transaction [3]. The content of a web service [9], that is, its business logic, can be coded in any programming language; but for reasons such as portability, Java programming language is ideal for building web services and developing related applications [15]. Java APIs are available for these purposes. In essence, these APIs provide the mapping between the Java programming language and XML [15] so that Java technology can be used with the XML-based technologies that underlie web services. To avoid all the drawbacks of the existing standard component/product libraries, the following new system is proposed with the web service oriented architecture [9].

3.2 Web Service Oriented Catalogue System Architecture

Ideally, suppliers can provide catalogue data and CAD models to catalogue providers, who can publish their WSDL to a web service registry. This service registry is categorized into many varieties like telephone yellow-page books. By using the UDDI information that presents in the registry, the end user can develop a link with service provider and use the information in the WSDL to send and get the SOAP messages with data. Apart from overcoming the disadvantages of the traditional as well as the existing catalogues, the main advantage of this web service system is that end user or third-party service providers can access many catalogues from different suppliers via a uniform platform so as to compare the specifications, prices, quality etc. for a particular mechanical component or a sub-module required. This will provide great opportunities for the competitiveness in cost and quality for OEMs as well as the suppliers [5]. At the same time, a new third-party web catalogue service business can be created by connecting the suppliers according to a common protocol.

Catalogue vendors should have full control on the entire system. End users or CAD vendors are not required to invest and develop separate system for their specific CAD systems. Figure 1 shows the overall framework of the system. The framework is divided into three major modules which are briefly introduced as below:

- End user station implementation. In this end, user applications, their interfaces, transactions, uploading and down-loading functions, are hosted. This module is also called Request module.
- Web service module. This is the portal server module implemented with the Java web services technology.
- CAD application module. This module is integrated with CAD solid modelers, which provides fine-grain geometric services and data processing capability.

These three modules are further discussed in detail in the following sub-sections respectively.

3.2.1 Request Module

This module generates the request code through end user selection. Basically, all the catalogues are generated with the reference catalogue number, which is the key for the customer to order according to their requirement. This module generates the catalogue number as request key. The catalogue selection and catalogue key generations are made through the server. The implementation may be done with simple JavaScript, HTML programming languages and ActiveX components. For the actual implementation, developers can use the Java Server Pages (JSP) and Java bean technology to connect the client selection and data retrieval from database through the Java programming.

3.2.2. Web Server Module

For this research, the Appache Tomcat web-server is selected, which is free to use. Tomcat is the Servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies. Tomcat is developed in an open and participatory environment and released under the Apache Software License. In the proposed system, catalogue providers publish their services and WSDL information in the UDDI registry under the category of standard components for the precision industries. The service requesters (end users) discover the above service by using the UDDI registry with the client application, or manually, and uses the published WSDL to generate the client proxy. At runtime the client uses the client proxy to construct and send SOAP messages to the Web service module.

Figure 2 illustrates the process sequence of the system upon reception of the end user requirements. First, the user's selection of component type, sub-type, alterations, and dimensions is recorded and coded according to a product configuration convention. This request file is used to check whether a CAD model with the same configuration exists. If there is an existing CAD model, then it is reused by updating the dimensions according to the user's selected dimensional data (expression values for CAD feature parameters). After parametric updating with solid modeler API functions, the new CAD model is saved and tagged as the required item, and it is then sent back to the client straight.

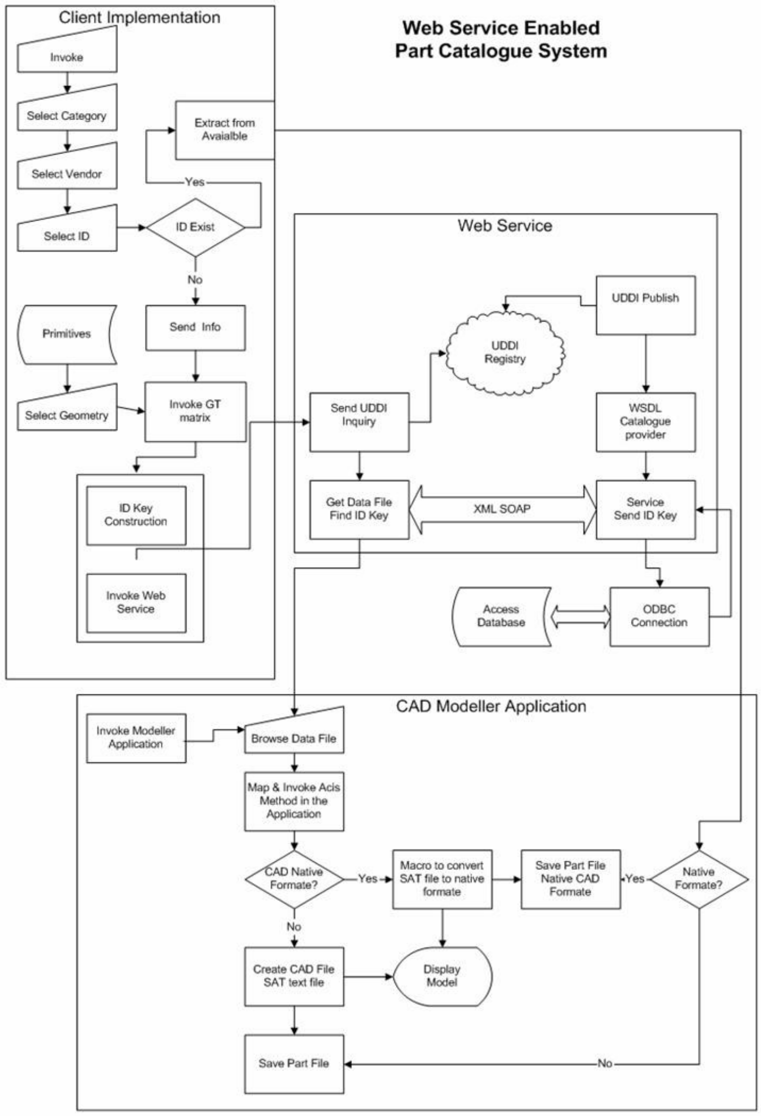

Figure 1: Web Service enabled part catalogue system

If there is no suitable CAD template existing in the client side, a request for generating a new CAD model is issued from the client module to the catalogue web service module via 'sendKeyRequest' method. The vendor-specific configuration file is then generated according to the user's selection (or request), and sent to the vendor catalogue database module for verification via the '*selectData*' method. At the same time, available sizes together with the corresponding dimension values are obtained from the vendor's catalogue database via the vendor's client module running '*returnData*' method. Based on the returned configuration and dimensional data files, the catalogue web service module activate the 'generateModel' method which in turn calls solid modeler to create the corresponding CAD object which is then sent back to the user's client side. Via the 'accessModel' method, the user then could use the newly generated standard component or sub-assembly model in his design environment.

One of the important modules in this system is the database. All the catalogue data have to be kept in the database for the electronic catalogues. Since enormous amount of research and development have been done in this area, this research does not focus on this area and a small database had been developed for testing purpose only.

3.2.3 CAD Object Modeler

When generating new CAD objects, the return XML SOAP from catalogue database module contains the data to generate the CAD object and there should be an application engine to handle this XML data to for CAD object generation.

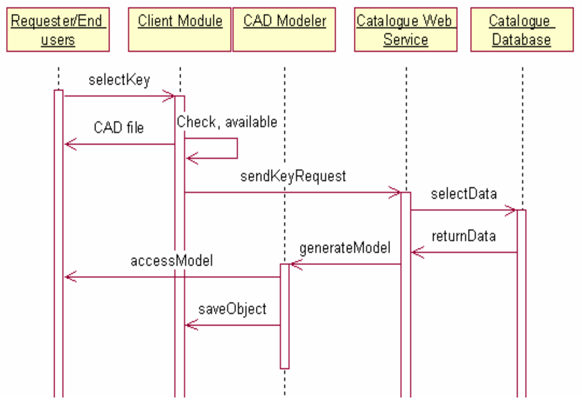

Figure 2: Process sequence for the web service based catalogues

There are a few 3D modelers available in the market. For the actual implementation, catalogue vendors can use any 3D modeler according to their requirements [12][13][14][11]. There is a feasible solution to achieve feature-level uniformity for CAD model generation by using a neutral feature operation scheme as reported in [1][4]. For this research, ACIS 3D CAD modeler has been chosen simply for the availability of research license.

ACIS integrates wireframe, surface, and solid modeling by allowing these alternative representations to coexist naturally in a unified data structure, which is implemented in a hierarchy of C++ classes. ACIS bodies can have any of these forms or combinations of them. Linear and quadric geometry is represented analytically, and non-uniform rational B-splines (*NURBS*) represent free-form geometry. ACIS is relatively platform independent and freely available for the university research project development. All the necessary basic objects are available with functions, and user can easily integrate the basic objects to create the required CAD objects. ACIS is written in C++ and consists of a set of C++ classes (including data and member functions, or methods) and functions. As such, we can use these classes and functions to create the generic CAD modeler, which is used to generate ACIS 3D files to be inserted into the end user CAD system.

Figure 3 illustrates the CAD modeler sequence of the system. Continuing from the scenario described in the previous sub-section, if a new CAD model is required, the data flow starts when the end user sends the catalogue request key to vendors' web service module. After it processes internally to find the values from the database and return as XML file to the client (end user) side, which contains the CAD geometry generation instructions and the relevant dimensional data in aneutral format. It is then interpreted by an interface developed on top of a CAD (ACIS) solid modeler. Those two portions of information obtained are then converted into solid modeler specific representation, i.e. feature configuration and the relevant dimensional data to be applied corresponding to the used features and their special positions and orientations. Presently, the solid modeler is proposed to be installed in the client side. According to the feature definitions contained in the configuration file, a configuration code is obtained from the Group Technology (GT) matrix module. This configuration code (ID) is unique representing the features and their special relations to be generated. Once this code is returned back to the solid modeler constructor, a geometry generation module developed in an object oriented manner, based on this ID together with the dimensional data provided, CAD 3D models are then generated automatically by calling ACIS API functions which are encapsulated into the constructor's methods. Note that if the ACIS CAD file is the final target format, this file is then sent to the client CAD system directly. Otherwise, another CAD format translator is called to covert the generated ACIS file into the target CAD system format or STEP neutral format first before the resultant file being sent to the client CAD system. When editing is required by end users, regeneration has to be carried out.

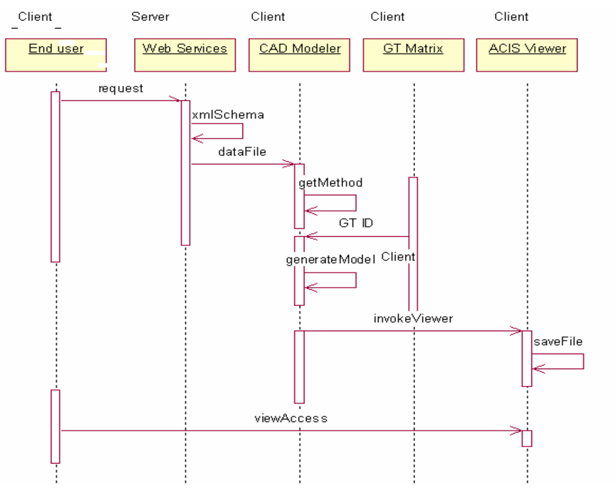

Figure 3: CAD modeler sequence

4. SYSTEM IMPLEMENTATION

4.1 Programming Languages

The prototype system implementation takes into account the major aspects discussed in the previous sections. Several software modules have been developed, including web service creation and deployment, web server deployment, user interfaces, database system, CAD modeler application and integration interfaces. As mentioned before, Java technology has been used to create the Client/Server system and also the Web Services. The details are as follows:

- Programming language-The Java 2 Standard Edition JDK 1.4.2, containing all the basic Java libraries and tools.
- Web Service development tool-The Java Web Services Developer Pack, version 1.3, an integrated toolkit that allows Java developers to build, test and deploy XML applications, Web Services, and Web applications.
- Web Server/Servlet-Tomcat 5.0 servlet/JSP container for Java Web Services.

4.2 System Architecture

From system implementation point of view, there are three layers in the system architecture. The upper layer is a SOAP interface layer, which contains interface classes defined for SOAP calls. This interface layer is interacted directly through SOAP calls. It represents the gateway of the main functionalities offered by this catalogue service system. The middle layer implements business logic for catalogue service. During application process, certain dynamic source codes are created by this layer according to the application requirements. The lower layer is the database of the application. There is a package named "Data" providing some common functionalities of repository needed for this system. Figure 4 illustrates the components and web services used in a cycle.

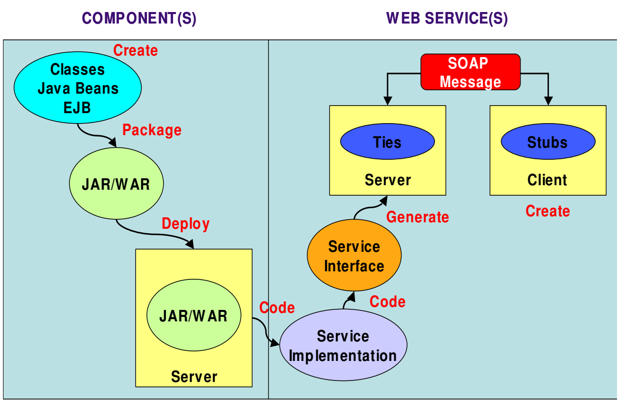

Figure 4: Components and web services used in a cycle

4.3 A Case Study with Injection Mold Design

Plastic injection mold design covers a very important industrial sector. Due to the pressure for quick delivery and high precision, intelligent mold design with advanced CAD tools has been widely accepted [10][12][14][11]. One of the major aspects is to use standard components or modules as much as possible. This work is based on the existing standard components and assembly libraries developed in house to explore their integration with web service technology. Because of high development effort required, this research has simplified the prototype system with emphasis on the information flow and application integration. In the prototype system, the user interface contains the following aspects:

- The available product types/variants, and sizes.
- The picture of the product alongside with its master parametric dimensions.
- Some detailed information about the configurable parameters.
- Some other important information such as price.

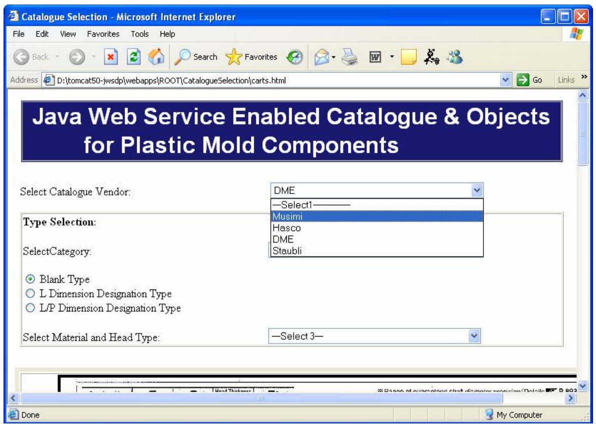

Figure 5: Display & selection of category from the UI

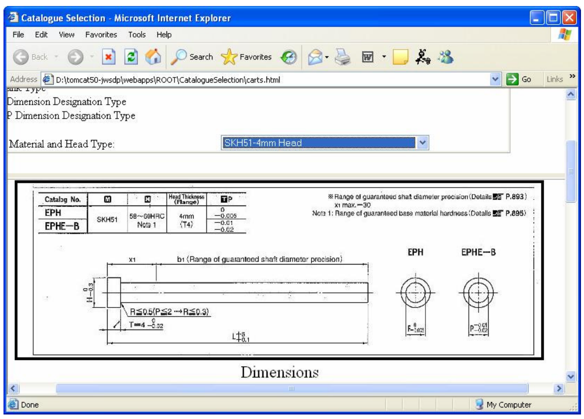

Figure 6: Display of product major details in the UI

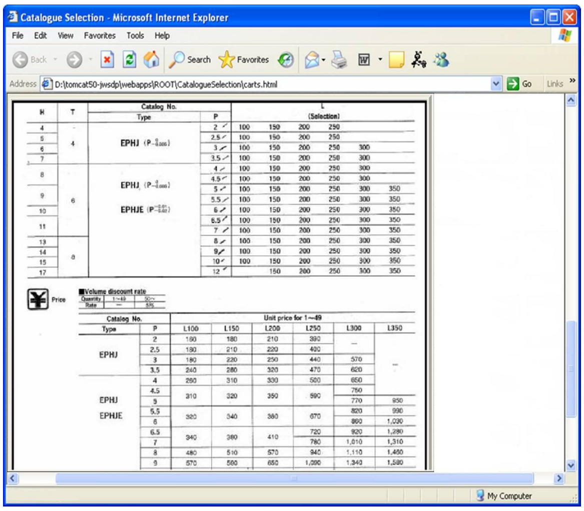

Figure 7: Display of product's parameters in UI

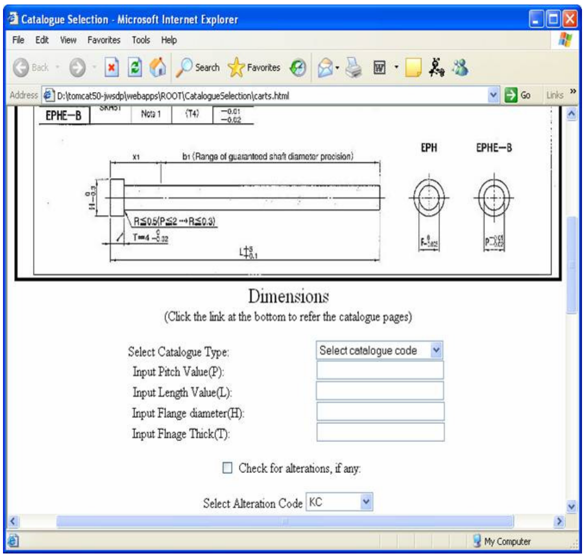

Figure 8: Input the catalogue values in the UI

The following figures show some of the UI screens captured from a case demonstration with the prototype system developed. Figure 5 shows that the user selects a vendor from the catalogue provider list, and defines the component configuration method. Figure 6 shows the user's selection of major type from the customized component types; here 'SKH51-4mm Head' indicate the type is 'a pin with head; major type is 'EPH' or 'EPHE-B'. User also needs to indicate the dimensions required from specification. Figure 7 shows the available sizes and dimensional values usable. Figure 8 shows the UI where the user could specify the available sizes or major parameters. Figure 9 uses the available configurations for a center pin component. Note that a UI can make use of images scanned from the existing hard copies of catalogues; hence, the UI could have the same convention in coding of configurations, dimensions, etc. This technique would make the implementation more user-friendly and economical. Figure 10 shows a generated center pin from the CAD modeler module. Figure 11 shows the environment of QuickMould, a productivity tool developed in NTU, including the main menu and the standard component library (SCL) UIs.

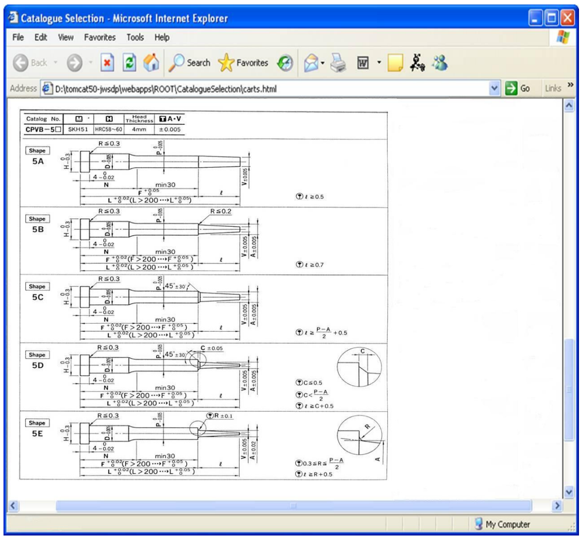

Figure 9: Display of various alterations in the UI

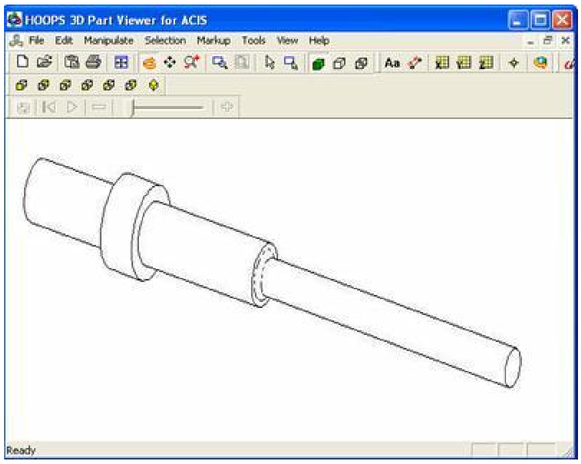

Figure 10: Display of CAD object in the HOOPS viewer

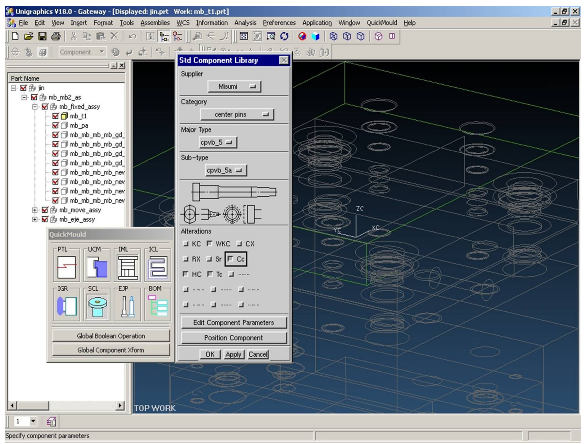

Figure 11: Application of CAD object in a mould design environment

From Figure 11, it can be seen that because standard components CAD models are generated automatically, CAD models can embed the allowed type/sub-type configurations and variations into the standard component object so that the component can be flexibly configured upon use or modifications required from time to time.

5. CONCLUSIONS

Although traditional book type catalogues are still being used in current industrial practice, some standard component and assembly suppliers have already implemented different computer-based electronic catalogues which are either standalone or web-based. However they have built-in drawbacks as identified in Section 2. The informatics technology for standard components and assembly libraries is still in the evolution and development stage. This proposed web service oriented catalogue system overcomes those drawbacks concerned to the currently existing systems. The web service technology is an entirely new concept and the application developments are still at the research stage [9]. There is no web service enabled application for the mould industry and CAD systems. The presented approach gives architecture to develop such systems by integrating with a solid modeler and a feature-based geometry configuration file which is in platform independent XML formats. Also, catalogue providers will have full control on their catalogues and the CAD objects, thus avoiding the intermediate CAD vendor's involvement, which in turn makes the catalogues more timely updated.

Another advantage of this web service oriented approach is that the end user or the third-party service provider can integrate many catalogue providers to compare the price, quality, etc. for a particular component/product. This will provide a great opportunity for the competitiveness in price and quality of component providers and also for the new third-party catalogue service business.

The presented work has provided only an initiative and reference architecture for the future system. Further development on this proposed system is necessary. This is because that web service is a new technology [7] that has limited proofed applications in areas of CAD and product lifecycle management so far.

REFERENCES

1. Bidarra, R. and Bronsvoort, W. F., 2000, "Semantic feature modeling," *Computer-Aided Design*, Vol. 32, pp. 201-225.
2. Cadalog.com, http://www.cadalog.com/index.php.
3. Calladine, J., 2004, "Giving legs to the legacy-web services integration within the enterprise," *BT Technology Journal*, Vol. 22, No. 1, pp. 87-98.
4. Chen, J. Y., Ma, Y. S., Wang, C. L. and Au, C. K., 2005, "Collaborative design environment with multiple CAD systems," *Computer-Aided Design and Applications*, Vol. 2, No. 1-4, pp. 367-376.
5. CIMdata Inc., 2001, "Collaborative product definition management (cPDM): An overview", http://www.CIMdata.com.
6. Culley, S. J, Webber, S. J., 1992, "Implementation requirements for electronic standard component catalogues," *Journal of Engineering Manufacture*, Vol. 206, No. 3, pp. 253-260.
7. Davies, N. J., Fensel, D. and Richardson, M., 2004, "The future of web services," *BT Technology Journal*, Vol. 22, No. 1, pp. 118-130.
8. Domenico, B., Francesco, G., Stefania, M. and Maurizio, V., 2004, "A web service based framework for the semantic mapping amongst product classification," *Journal of Electronic Commerce Research*, Vol. 5, pp. 114-127.
9. Estrem, W. A., 2003, "Toward the information utility: Service-oriented architectures in the next generation manufacturing enterprise," CASA/SME blue book series, Society of Manufacturing Engineers, http://www.sme.org.
10. Fu, M. W., Fuh, J. Y. H. and Nee, A. Y. C., 2001, "Core and cavity generation method in injections mould design," *International Journal of Production Research*, Vol. 39, pp. 121-138.
11. Ma, Y. S., Britton, G. A., Tor, S. B. and Jin, L. Y., 2005, "Associative assembly design features: Concept, implementation and application," *International Journal of Advanced Manufacturing Technology*.
12. Ma, Y. S., Britton, G. A., Tor, S. B., Gunawan, E. and Lee, C. H., 2003, "Standard component library design and implementation for plastic injection mold design with CAD tool," *The Fourth International Conference on Control and Automation*.
13. Ma, Y. S., Britton, G. A., Tor, S. B., Jin, L. Y., Chen, G. and Tang, S. H., 2004, "Design of a feature-object-based mechanical assembly library," *Computer-Aided Design and Application*, Vol. 1, No. 1-4, pp. 397-403.
14. Ma, Y. S., Tor, S. B. and Britton, G. A., 2003 "The development of a standard component library for plastic injection mould design using an object oriented approach," *International Journal of Advanced Manufacturing Technology* Vol. 22, pp. 611-618.
15. Mok, C. K., Chin, K. S. and Ho, J. K. L., 2001 "An interactive knowledge based method in injection moulding process," *International*

Journal of Advanced Manufacturing Technology, Vol. 17, pp. 27-38.
16. Raghothama, S. and Shapiro, V., 2002 "Topological framework for part families", *Proceedings of ACM SM'02*.
17. Sun Micro Systems Inc., "The Java^{TM} Web Services Tutorial".
18. Thomas Publishing Inc, http://www.cadregister.com/.
19. UDDI Web Site, http://www.uddi.org.
20. Varlamis, I., Vazirgiannis, M., 2001, "Bridging XML-schema and relational databases: A system for generating and manipulating relational databases using valid XML documents," *Proceedings of the ACM Symposium on Document Engineering*, Vol. 1, pp 105-114.
21. W3C XML and related recommendations, http://www.w3.org/TR.
22. Website: http://www.webopedia.com/TERM/W/Web_services.html.
23. Xue, D. and Dong, Z., 1997, "Coding and clustering of design and manufacturing features for concurrent design," *Computers in Industry*, Vol. 34, No. 1, pp. 139-153.

ABOUT THE AUTHORS

Y.-S. Ma is an Associate Professor at the school of Mechanical and Aerospace Engineering, Nanyang Technological University (NTU), Singapore. His main research areas include product lifecycle management, feature-based product and process modeling. Dr. Ma received his B. Eng. from Tsing Hua University, Beijing (1986). He obtained both his M. Sc. and PhD degrees from Manchester University, UK in 1990 and 1994 respectively. He worked in Ngee Ann Polytechnic as a lecturer from 1993 to 1996. Then he joined Singapore Institute of Manufacturing Technology (SIMTech) as a Research Fellow. Before joining NTU in 2000, he was a research group manager and a Senior Research Fellow.

T.S. Rajeshbabu is a Senior Technical Support Engineer with Legrand IPO Asia Pte Ltd. He graduated with Msc degree from School of MAE, Nanyang Technological University, Singapore in 2005. His research experience covers CADCAM development and Web-oriented industrial IT solutions. He had been with Singapore Institute of Manufacturing Technology (SIMTech) from 2002 to 2005.

Y.M. Deng is a Professor of Ningbo University, Faculty of Engineering, Ningbo, Zhejiang,China. His research areas include plastic injection moulding simulation, CADCAM, intelligent manufacturing and micro-device design.

(Received May 2006, revised November 2006, accepted April 2007)