

# A New Web Enabled Model for Product Structure and Collaboration in Product Development

K Prakasan  
{\*prakasanpsg@gmail.com }

Department of Production Engineering, PSG College of Technology, Tamilnadu, Coimbatore-641004, India<sup>1</sup>

**Abstract.** Product development is a collaborative effort, involving multi-disciplinary functions, heterogeneous databases and applications. Product structure (PS) is the key element with which several variants of a given product can be instantiated with appropriate effectivity. Intelligence built into the product structure decides the reusability of this entity to generate variants for different customer requirements. This situation is comparable to that of the evolution of organisms caused by genes. The basic biochemical unit required to drive all biological processes is protein. Amino acids are the structural elements of proteins and each amino acid is produced by a specific gene. The groups of codons that are arranged in grammatically correct sequence are called the genes. Such attempts to model product development as a genetic process can be found in literature. The availability of functions like views, bill of materials (BOM) and product visualization/markup for a given product structure is critical for precise communication among different teams. A seamless integration of these functions and a proper mapping of the physical elements in the product development process with the information model are necessary to track and control the process throughout the product life cycle. The model for product structure developed in this work is based on genetic approach and domain theory. The model is equipped with efficient view-management, generation of bill of materials and visualization functions. Java 2 Enterprise Edition (J2EE), Autovue solid model Application Program Interface (API) for Visualization and eXtensible Markup Language (XML) for data access are used in its development.

**Keywords:** J2EE, XML, API, BOM, PS.

## 1 Introduction

Related documents finally pointing to a physical product. While realizing the final product, this product structure will be assigned with values for several attributes of those components which constitute the product structure through effectivity. This process can be compared to the evolution of organisms to assimilate ways to handle customization of products. A change in the sequence of genes in an organism produces a new variant. In product development, many a time, existing products

undergo change with reference to some of the components and other components remain unchanged still making a meaningful relationship. This leads to the concept of modeled attributes and instance-based attributes (IBAs). Recombinant Deoxyribo Nucleic Acid (rDNA) models can be applied for evolving powerful models for the product structure.

The model proposed in this paper uses genetic approach to develop model for part and relation among parts. Domain theory is used for providing these objects with interfaces for life cycle management. For other functions like views, BOM's and visualization, this model is equipped with Autovue solid model API for java [1] for accessing information relating to the part. This API provides a framework where various formats of product data can be visualized within a single application.

## 2 New Product – Genetic Approach And Domain Theory

In genetic approach, according to Wendy Czika, [2], the new product development can be studied from the evolution process of organisms. The basic units of the living beings are the genes which are responsible for the characteristic features of the being. Similarly, the basic units of the product are the components which make up the product. To change a particular trait of a living being, the gene which promotes that kind of trait is manipulated and a desired trait is produced. Similarly, to produce a customized product the components which are necessary for the required product are manipulated to realize the product. Though in an organism some of the genes are manipulated, there are some genes which are kept undisturbed. These genes are required for the functional performance of the being. Like this there are some components in the product which form the main functional unit of the product and that cannot be customized for the prevailing conditions. These components form a fixed position in the product structure and only those components which are customizable within the frame work of the enterprise vary from variant to variant.

The structural ways of chromosome model consist of technology, organ, and part. The interaction between the product and the environment or user is explained by technology. The properties and mode of action for a product is explained by organs. The physical realization of the product is known as part [3]. The four system points of view of a product is shown in Fig.1[4].

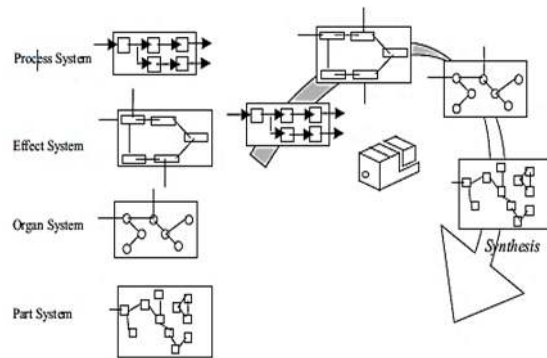


Fig. 1 Four system points of view [4]

The scenarios where transformation of materials, energy, and information of the product related to its use or function is required demands process view. Process view describes the process structure. The effect or function of a structure that is in focus is enabled by effect view. The purpose of effect view is to facilitate the necessary transformation. The organ structure is nothing but the result of design considerations in it. The technical principles of the

product that are in focus are related organ view. The principles are called organs. The organs create the desired effects by their functionality. The conditions that are required for the organs and their functionality are created by determining the material, tolerances on form, and surface quality.

The four systems of a product explained above appears that they are interrelated in a casual chain. It is required to establish a procedure for the synthesis of a product so that the four systems of a product will be in a sequence following the four domains.

Variations in the assortment of the products is demanded due to different market segments, users, and use situations. The variations generally act as a cost driver for the organization as the benefits of high volume is lost. This drawback could be addressed by incorporating flexibility in the production or by creating certain commonality between the products that are identical to the manufacturing system. In general, there are four variances. They are process level, organ level, effect level and part level variances [5]. Work related to product life cycle support [6] and collaboration in product development [7] were also done in this domain.

From the wide study of the domain theory and genetic approach towards product structure modeling, it may be concluded that genetic approach gives a pathway for the modeling of a part, product structure and the relationship between them and the domain theory gives the pathway for modeling the part, its views and its interaction with the life cycle through the workflow interface. The model developed in this paper was based on these two concepts.

### 3 Model For Part

The part was developed with consideration to relationship with other parts, processes and other value addition interfaces. Object oriented software (java) was used to define the part, its attributes and its interfaces with life cycle and workflow.

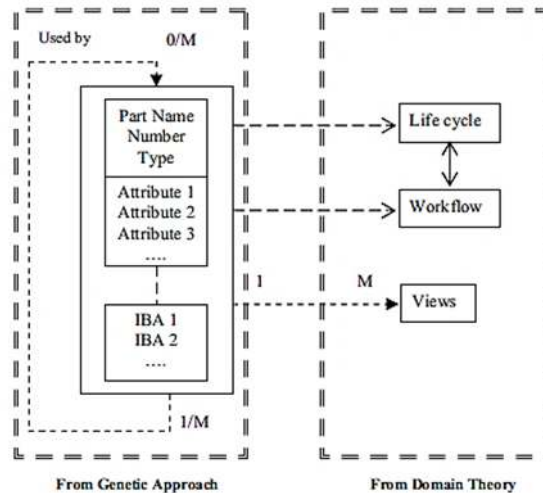


Fig. 2 Model for the part

The database schema was defined in a relational database (oracle). The part model developed based on the above approaches is shown schematically in Fig 2 and the interface for

creation of a part is shown in Fig 3. User interface is through a browser which is powered by servlets.

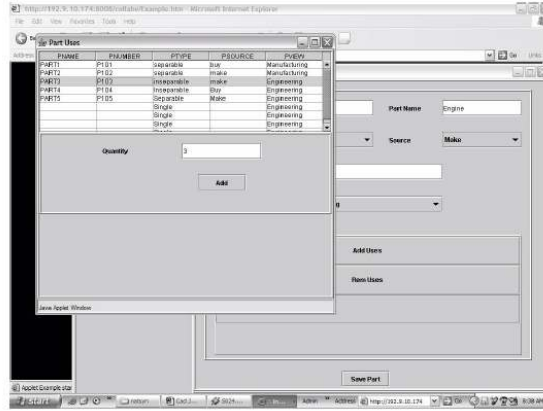


Fig. 3 GUI for creation of part

#### 4 Model For Product Structure

A gene is formed by a particular sequence of codons derived from DNA. Any change in this sequence produces a new gene. This pattern can be correlated to a set of parts that form the component with modeled attributes and IBAs. Let the attributes of component C1 be A1, A2, and A3. Then it may be mathematically represented as

$$C1 = \{A1, A2, A3\} \quad (1)$$

Likewise, other components can be represented with their attributes. Two components become similar only when they have the same set of attributes. Some representation of components are given below.

$$C2 = \{A1, A4, A3\} \quad (2)$$

$$C3 = \{A5, A2, A4, A1\} \quad (3)$$

A difference has to be introduced between the actual gene and the component at this stage. According to Timothy Caulfield, [8], the attributes of the gene are decided at the design stage itself but an additional facility is given to the component of being updated with attributes at any stage of its life. But it can be correlated to the dormant genes, i.e those genes that remain dormant till a particular phase of the life and gets into active role after that. For example, the gene that controls the trait of a moustache remains dormant till the early adolescent age and gets active after that. Similarly, a component can have attributes which are not having any value till a particular phase of the life cycle and is provided with a value after that. These attributes are called instance based attributes (IBA's). A genetic model of product structure is shown in Fig 4.  $X_1, X_2, \dots, X_N$  are the attributes each of which can have values ranging from  $x_{m1}, x_{m2}, \dots, x_{mn}$  for  $m = 1, 2, \dots, N$ .

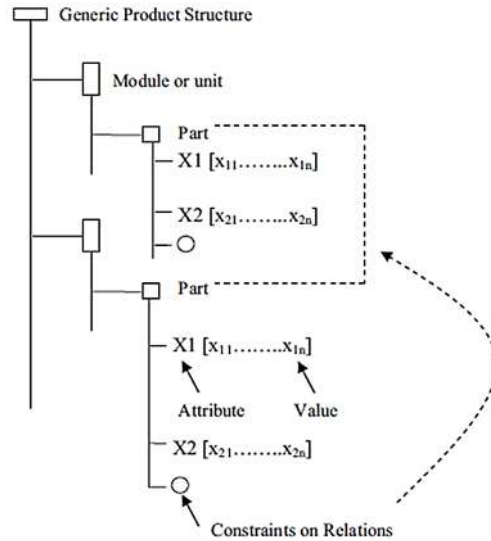


Fig.4 Genetic model of the product structure  
 Building of a product structure is shown in Fig 5 and 6.

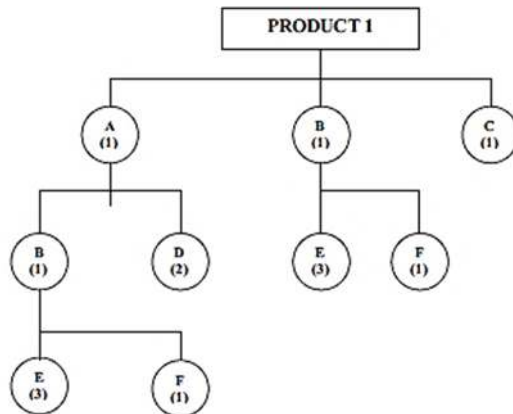


Fig.5 Building of product structure

This model uses a separate database schema for storing the relationship between the parts (components, subassemblies, assemblies). This schema was designed keeping in mind that each part is a product by itself. This part may have sub-parts or it may be a sub part for another part. The model discussed in this paper is made flexible so that the product structure can be derived for any level of the product, i.e., structure of the assembly, sub assembly and the entire product. When a particular product structure is requested, iterations are executed through all the relationships defined for that part till no more relations are found. The parts at the leaf nodes of the product structure are the parts which does not use any other part as its subpart.

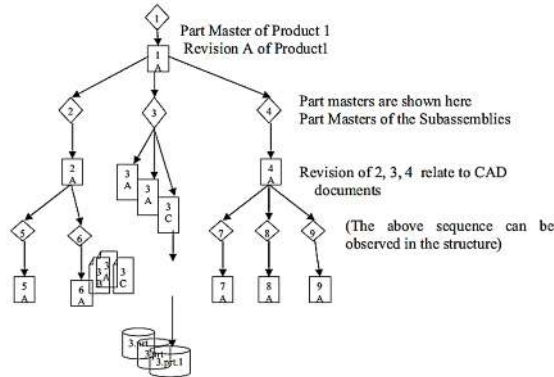


Fig.6. PS represented as an association among part versions and part masters [9]  
 The entity relationship (ER) diagram for this database schema is shown in Fig.7.

## 5 View Management

A good product structure should help the engineer in maintaining various configurations and views. The model developed should be capable of tracking versions, effectivity, and design changes when the configuration change over time to enhance the value of the product structure for various activities across the extended enterprise. It should be possible to generate BoM of different configuration of products automatically from the product structure to support design, purchasing, manufacturing and other activities of the organization. The following Fig.8 shows various views of the product structure.

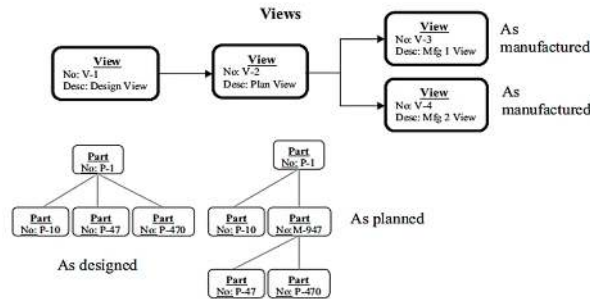


Fig. 8 Product structure and views

The work presented in this paper is to empower the users and applications to link various kind of product data like drawings, documents and process plans with the parts and assemblies within a product structure. In this model, various views are defined by the user at the time of creation of the part. Default views like engineering view and manufacturing view can be assigned to the part at the time of creation. If a new view has to be created, then the administrator creates a new value for the view array. Different views are then assigned to parts and the views are saved in the database. The views can be set with access control rights. When an authenticated user requests for a particular view of a product structure, the database is

queried for the parts using java servlets and data is added to a tree structure sequentially. At any point of time when need arises to add a part to the system under a particular view, a synchronization module is available to synchronize this data with the other views based on the configuration set by the administrator. The views are in turn connected to the visualization tool by getting the link of location of the element from the database and then setting it as an attribute to the visualization domain. The data access diagram for the generation of various views of the product structure and the corresponding methodology used are shown in Fig 9.

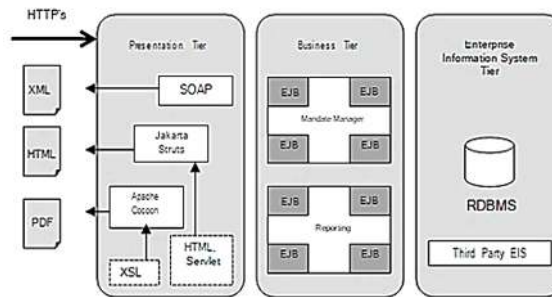


Fig.9 Data access for product structure generation  
The snapshots of screen for generation of views are shown in Fig 10 and Fig 11.

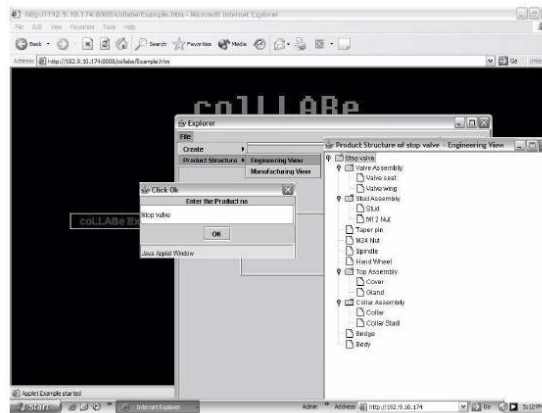


Fig.10 Product structure-engineering view of stop valve  
Simple Object Access Protocol (SOAP) is used to retrieve data in XML format and eXtensible Styling Language (XSL) is used for formatting XML file into a presentable format. Hyper Text Transfer Protocol (HTTP) is the protocol used for network access. Enterprise Java Beans (EJB) is used for managing distributed data. The Enterprise Information System (EIS) is modeled using a Relational Database Management System (RDBMS).

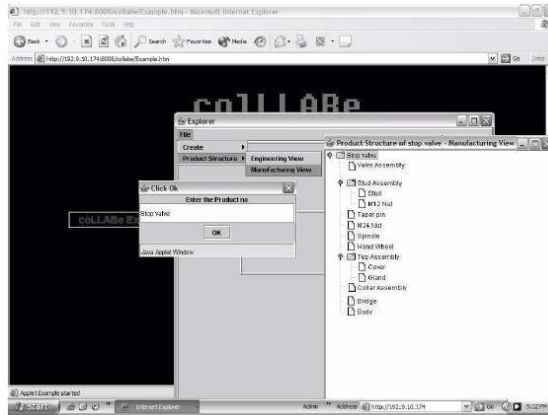


Fig.11 Product structure - Manufacturing view of stop valve

## 6 BOM Generation and BOM Management

The most important deliverable that an engineering team presents to the manufacturing team is the bill of materials (BOM). It is developed and used by a team within the organization and throughout the supply chain. Managing BOM efficiently requires a multi-user relational data base that can support multitudes of data types, entries and relationships. It should also be possible to access the BOM by the stakeholders regardless of their geographic location.

Although the manufacturing department appreciate the use of BOM, they do not create it. It is created by the engineering project manager who is responsible for estimating the costs and analyzing the engineering tasks. The BOM becomes highly complex and difficult to maintain as the design process transforms from early development stage to the production stage. The engineering BOM is created and maintained in a spreadsheet program. The BOM is closely related with the product structure views. The manufacturing BOM gets its data from the manufacturing view of the product structure and the other BOM views from their corresponding product structure. The quantity of each element can be obtained from the database for each node of the corresponding product structure. The generation of BOM in this model at the data access level is well described in the Fig 12.

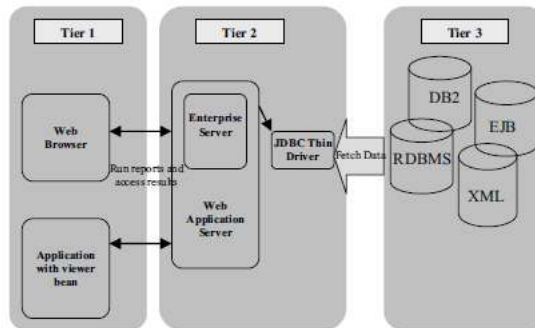


Fig. 12 Data access for BOM generation



In this model the management of product structure is made very simple. When the user requests for BOM of particular view, in the background the product structure for that particular view is actually created and the quantity of each element in the product structure is obtained from the database and appended to this data and the BOM is generated in a table structure. Any additional information requested by the user if residing in a different database, is also fetched and is appended to this data and shown to the user. Java servlets are used for retrieving data from various databases and merging them according to user requests. An interface to manage data synchronization is provided in the model which in turn be interfaced with the life cycle. The snapshots of the BOM generation for different views are shown in Fig 13 and 14.

Visualization of product data can be done at two levels. The first one is the visualization of the CAD model of the product and its components/parts and the second is the visualization of information pertaining to the product and its elements as described by notes, instruction sheets and manuals.

Part No	Part Name	Quantity	Location	Source
P101	Valve seat	1	Natraja	Buy
P245	Valve wing	1	Natraja	Buy
P097	Stud	2	Natraja	Make
P011	M12 Nut	8	Natraja	Make
P137	Taper pin	1	Natraja	Make
P913	M24 Nut	4	Natraja	Make
P330	Spindle	1	Natraja	Make
P712	Handle wheel	1	Natraja	Make
P901	Cover	2	Natraja	Make
P835	Gland	1	Natraja	Make
P221	Collar	1	Natraja	Buy
P014	Collar stud	1	Natraja	Buy
P452	Bridge	1	Natraja	Make
P113	Body	1	Natraja	Make

Fig. 13 Engineering BOM for stop valve

Part No	Part Name	Quantity	Location	Source
P100	Valve assembly	1	Natraja	Buy
P097	Stud	2	Natraja	Make
P011	M12 Nut	8	Natraja	Make
P137	Taper pin	1	Natraja	Make
P913	M24 Nut	4	Natraja	Make
P330	Spindle	1	Natraja	Make
P712	Handle wheel	1	Natraja	Make
P901	Cover	2	Natraja	Make
P835	Gland	1	Natraja	Make
P221	Collar assembly	1	Natraja	Buy
P452	Bridge	1	Natraja	Make
P113	Body	1	Natraja	Make

Fig. 14 Manufacturing BOM for stop valve

## 7 Visualization Of Product Data

User can achieve these objectives by using a thin client developed with Autovue. This is done using an API of Autovue which enables visualization of information, irrespective of the application with which it was created.

The additional feature of using such an API is to provide a collaborative session between various users of the data management system. Users can work in the local CAD system and can upload files to the server. Using the Autovue API implementation they can view the file within the client system. The user can invite other users to participate in the collaboration system through which online discussion like chatting, markup and other collaborative tools can be used to make the design faster and easier. The CAD platform used in this work is Pro/Engineer 2001.

The sequence of operations that has to be carried out for a successful collaboration session are described in Fig 15 through 17.

Sequence of operations followed are

1. The administrator creates users who can access the system.
2. The administrator defines the access control policies for each user which will define the level of usage of the system.
3. The administrator creates workflows using the work flow templates and maps the users and rolls of the work flow.
4. The administrator structures and defines the various phases of the life cycle through which the information can be managed.
5. The administrator creates the parts.

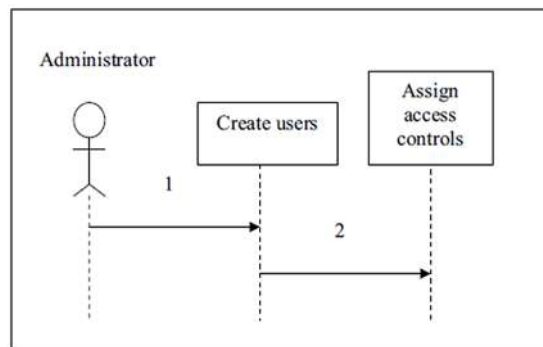


Fig.15 Users creation

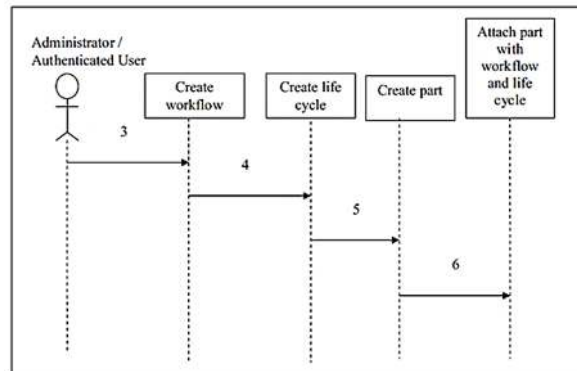


Fig. 16 Creation of business elements

6. The administrator attaches the part with one of the already defined life cycle and thus its corresponding workflow.
7. One of the users (user 1) logs into the system.
8. User 1 checks in the part to his local system.
9. User 1 designs and does modifications of the part that is checked in.
10. User 1 creates a collaboration session for the part.
11. User 1 sends a request to another user (user 2) to join the session.
12. User 2 receives the request.
13. User 2 logs into the collaboration session
14. User 2 receives the part file and is opened in the AutoVue API
15. User 1 and User 2 do chatting, markup and viewing.
16. User 2 leaves the collaboration session.

In this work, multiplexing of servlets is utilized for incorporating new databases of the enterprise. It can be achieved by specifying the address of database server in the configuration file. The multiplexing of servlets is shown in Fig.20.

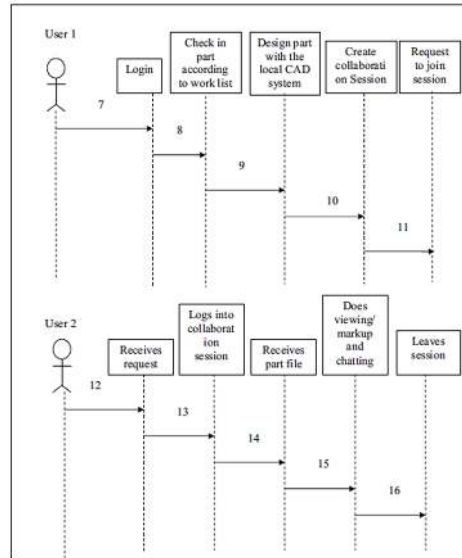


Fig.17 Collaborative session

The snapshot of the collaborative environment is shown in the Fig 18 and 19.

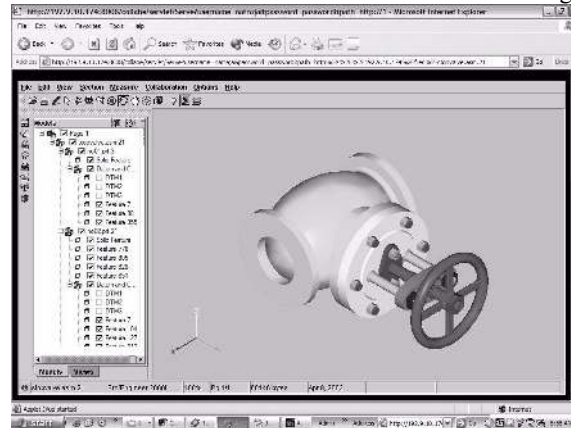


Fig.18 Multi-format information viewer

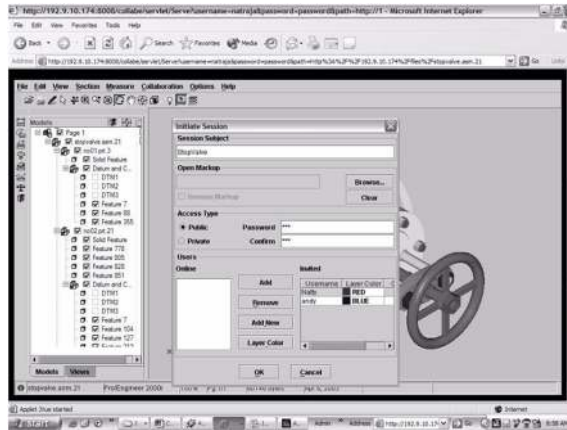


Fig.19 Collaborative session

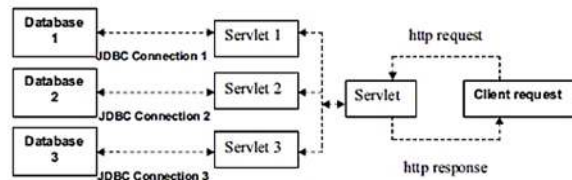


Fig.20. Multiplexing of servlets

## Conclusion

The complete knowledge of the product is inherent in the product structure and thus a robust modeling of product structure invites attention. The model developed in this work tries to use the genetic concepts and domain theory. An object oriented model using Java and Oracle is developed. To enhance its capabilities for handling engineering functions like BOM generation, view management and visualization/markup options are included. The working of the model is found to be satisfactory. This model can be used by small and medium enterprises (SMEs) with limited resources. As multiplexing of servlets is used in this model, a new database of the enterprise can be easily included by incorporating the address of the new database server in the configuration file of this model.

## References

- [1] Czika, Xiang Yu, and Russell D. Wolfinger, "An Introduction to Genetic Data Analysis Using SAS/Genetics", Wendy SAS Institute Inc. Cary, North Carolina, USA
- [2] Mortensen, N.H.: Product Modelling in Configuration Systems, Proceeding of Design Day, Department of Control and Engineering Design, Technical University of Denmark, 1999.
- [3] Andreasen, M.M.: Machine Design Methods Based on a Systematic Approach – Contribution to a Design Theory, Diss., [In Danish], DMD, Lund Institute of Technology, Sweden, 1980.
- [4] Andreasen, M.M.& Ahm, T.: Flexible Assembly Systems, IFS Publications/Springer Verlag, London, 1986.

- [5] M. Rezayat, "The Enterprise – Web portal for life-cycle support", *Computer aided design*, 32, p85-96, 2000
- [6] Lihui Wang, Weiming Shen, Helen Xie, Joseph Neelamkavil, Ajit Pardasani, "Collaborative conceptual design – state of the art and future trends", *Computer aided design*, 34, p981-996, 2002
- [7] "Genohype and the Genomic Revolution", Timothy Caulfield, Canada Research Chair in Health Law and Policy Research Director, Health Law Institute, University of Alberta
- [8] M. Tholkapiyan, A.Mohan, Vijayan.D.S , "A survey of recent studies on chlorophyll variation in Indian coastal waters", *IOP Conf. Series: Materials Science and Engineering* 993 (2020) 012041, doi:10.1088/1757-899X/993/1/012041.