Collaborative 3D Product Development with
Multiple Levels of Detail in Visualization of Design Features

Chih-Hsing Chu[1], Ping-Han Wu[2] and Yu-Chiung Hsu[3]

[1]National Tsing Hua University, chchu@ie.nthu.edu.tw
[2]National Tsing Hua University, g923822@oz.nthu.edu.tw
[3]National Tsing Hua University, g933830@oz.nthu.edu.tw

ABSTRACT

This research develops a novel representation scheme for collaborative 3D product development with multiple levels of detail (LODs) based on integration of feature model and 2D meshes. Different features are selectively hidden in the product model from certain collaborators, depending on their individual authorities and actual requirements in collaboration. The feature switch face is used to control display/hiding of each feature. A tree-like data structure is introduced for characterizing the feature hierarchy, 2D mesh, and the relations of each feature to the switch faces and LODs in a product model. Algorithms are also provided for quick generation of the mesh data from the data structure for displaying the 3D model corresponding to any LOD. An agent-based system, which allows multi-user synchronous collaborative 3D assembly over the Internet, is implemented to demonstrate the effectiveness of this work. A real scenario of 3D mold design using the system shows that the proposed LOD concept enhances the security of information sharing and the efficiency of data transmission while ensures the practicality in network-based 3D design collaboration.

Keywords: Collaborative product development, Feature model, Mesh, Levels of detail, Design collaboration

1. INTRODUCTION

The challenge most enterprises facing today is to assure their core competences and outsource less-skilled activities in order to remain competitive in global competition. This is especially important for the medium and small sized enterprises (SME's), which rely on close collaborations with customers, suppliers, and other partners of its supply chain, to compete with large companies. Collaborative product development (CPD) has thus emerged as an effective business approach in many industry sectors [6]. CPD involves a wide spectrum of product development activities, in which design is one of the most complex tasks. The design process is referred to as collaborative design when a product is designed through the collective and joint efforts of many designers [5]. There have been a number of studies concerning distributed collaborative design. Li et. al [13] performed a survey and discussed current research status in this field. Integration of horizontal and hierarchical collaboration, new feature-based streaming technology, and security/interoperability are identified as future research challenges.

Multiple-resolution techniques [11,14] have been identified for years as an effective method to achieve good performance in terms of shape preservation and data transfer. The most common approach is mesh refinement, or simplification, adopted for polygon meshes with various LODs, i.e. near objects are rendered with a higher resolution while distant ones with a lower resolution. Most mesh generation based on this approach have been focused on graphics rendering with less emphasis on LOD of design semantics. On the other hand, there have been studies that deal with compression and de-compression of 3D CAD in B-Rep in distributed environment. Koo and Lee [12] developed a "wrap-around" algorithm for producing multi-resolution representations from a B-Rep model, which is simplified according to the resolution level. This simplified model can be converted into a higher resolution while needed. With the wrap-around operation, Song and Lee [15] proposed a software framework for streaming the B-Rep model through the network. Users see a CAD model with incremental shape details. However, this streaming technique has an inherent restriction that only the space inside concave inner loop can be deleted but not convex inner loop. Moreover, it requires rendering environment implementing the warp-around procedure and other specialized CAD functions.

Fuxin and Edlund [8, 9] first put forward the concept of levels of detail (LODs) for the product development process. They argue that geometry-based product information is not only used at the design stage but also regarded as communication tools for other downstream activities in order to achieve integration of the entire development process. Therefore, they proposed a methodology that decomposes the requirements on Geometry Based Product Information (GBPI) to separate and categorize geometry users. However, the previous studies did not mention any implementation details. Cera et. al [3] developed a framework for information assurance (IA) within collaborative design based on role-based viewing. IA is required in collaborative design with multiple participants to protect engineering information by ensuring its availability, integrity, authentication, and access control. The FAÇADE (Framework for Access-control in Computer-Aided Design Environments) system demonstrates a new approach for achieving this through integration of collaborative graphics, feature-based model, and multiple resolution geometry. This approach mainly adopts the mesh simplification algorithm based on the quadric error metrics (QEM) [10] along with topology-preserving simplification technique. The degree of visibility is used to control the reduction of LOD.

This research develops a new representation scheme for visualization with multiple levels of detail (LODs) in collaborative 3D product development based on integration of feature–based model and 2D meshes. The scheme controls display or obfuscation of a design feature by the feature switch face(s). The product owner or the one leading the collaboration process can selectively hide certain features for each LOD and determine which LOD is accessible to each participant. A special tree data structure is introduced to store the mesh data grouped according to the relationships between feature/meshes, feature/switch face, and features/LOD. This work also provide algorithms for quick generation of the meshes corresponding to a given LOD from the data structure. Finally, an agent-based prototyping system is implemented based on the scheme, which allows synchronous 3D assembly and design review with multiple users over the Internet. A scenario of 3D mold design verifies the feasibility of the proposed scheme. It also demonstrates that the LOD concept enhances the security of product information sharing as well as the efficiency of data transmission in the Internet-based collaborative 3D design.

2. LEVELS OF DETSIL (LOD) IN PRODUCT MODEL

Recent research work [9] has shown that users' requirements should be used to define configurable product view. Different types of established product views need to be presented to them, depending on where they are active in the organization [8]. This idea is applicable throughout most stages of a product lifecycle. In collaborative product development, not only designers geographically dispersed are likely involved, but other stakeholders such as customer, suppliers, and technology providers may also participate the development process at different phases. The findings obtained in the empirical studies [6] indicate that different collaborators need to access product models with different levels of design information, depending on individual roles and the time they engage in the process, in order to assure the security of information sharing. For example, an ODM (Original Design and Manufacturing) manufacturer certainly does not want to disclose the complete product model when doing online design review with a supplier only responsible for development of a single component. It is to the manufacturer's advantage that the supplier can only gain access to the part model being outsourced and possibly some others interfacing with it. Sensitive design information or critical components need to be hidden from the supplier. The product owner should be able to freely choose the design content to be obscured. Intellectual properties can be well protected in this manner. In addition, the size of data transmission over the network is reduced during design collaboration if the models can be simplified as a result of eliminating the unnecessary or hidden information.

Therefore, this research argues that a product model should be categorized into variants rendered for different stakeholders considering their individual role and the time they join the collaboration. Fig. 1 illustrates such an idea about level of detail (LOD) for 3D model. LOD in this regard indicates assess control of design features at different levels. In contrast, the previous LOD methods that are derived from mesh refinement treat CAD model as a whole in geometry categorization. As a result, they cannot control display/hiding of design features in an explicit manner. Our empirical studies show that protection of information sharing in distributed collaborative product development should be put into the context of design semantics. Certain design features must be completely obfuscated in order to avoid leak of company intellectual properties. The mesh model only visually in a coarse resolution, cannot serve the purpose well.
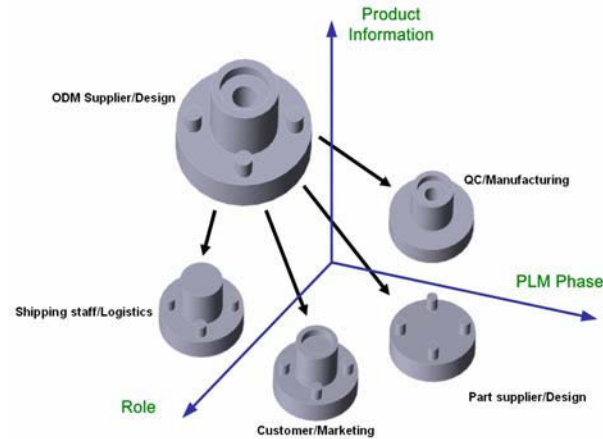
Fig. 1. Levels of detail (LODs) in 3D product model.

3. MULTIPLE LEVELS OF DETSIL (LODs) IN FEATURE MODEL

3.1 Feature Hiding with Switch Faces

This research develops a novel representation scheme for 3D product model with multiple LODs of design features based on integration of feature-based model and 2D mesh generation. The control mechanism of LOD is accomplished by the idea of "feature switch face". The following assumptions are made to simplify the scheme:

1. The representation scheme is proposed for form feature models.
2. 3D CAD model is constructed with a series of feature creation steps and each step involves only one feature creation.
3. Effective feature volume is generated by sweeping a 2D profile along a given direction
4. The display/hiding of a feature is controlled by its corresponding switch face (or faces).
5. 2D triangular meshes can be computed for each composing face of an effective feature volume.

(a)

remove composing faces → add the feature switch face →

pocket feature   opening to be filled   pocket is hidden

(b)

remove composing faces → add the feature switch face →

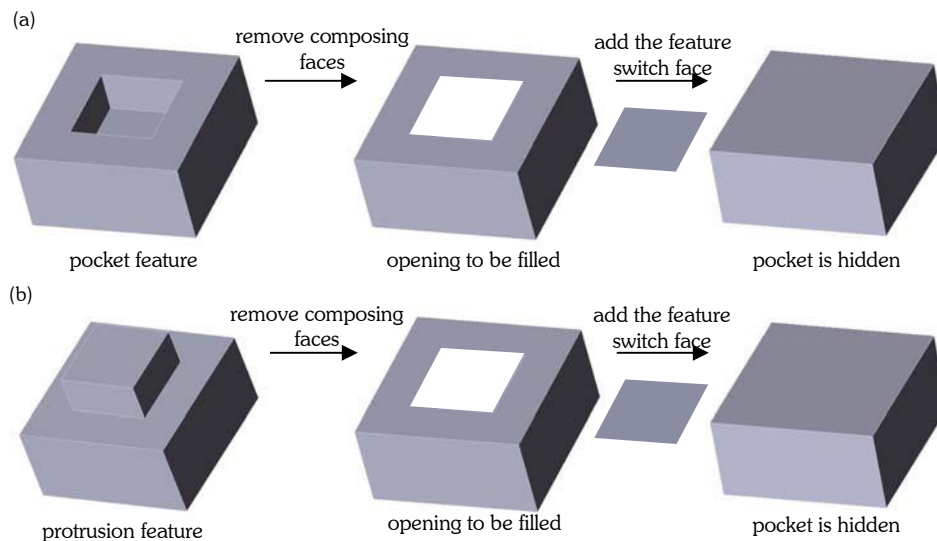protrusion feature   opening to be filled   pocket is hidden

Fig. 2. Procedure of hiding features.

Fig. 2(a). shows the procedure for hiding a pocket feature in the 3D model. The effective volume of the pocket is composed of five planar faces. The first step is to remove all the faces making up the feature. An opening is generated

as a result and makes the model visually invalid. The next step is thus to fill the opening with a new face, referred to as the feature switch face. The pocket disappears from the 3D model when the switch is on. The feature corresponding to a positive volume can be obfuscated similarly, as shown in Fig. 2(b).. Hence, displaying or hiding a form feature is accomplished by removing faces that make up it and filling up with the feature switch face, which then controls whether the feature displays or not. Notice that it is possible that a feature is controlled by multiple feature switch faces. As shown in Fig. 3., the step feature around the corner is controlled by three feature switch faces (the hatched faces). The feature becomes invisible when these switches display at the same time.
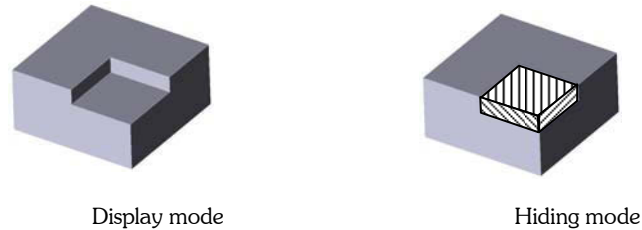
Display mode   Hiding mode

Fig. 3. Feature hiding with multiple feature switch faces.

Notation	Definition
L_m	Collection of all faces in the B-Rep model at stage m
V_n	Collection of all faces in the B-Rep of the effective volume of feature n
$F_{n,b}$	Face b in V_n
$F_{(pq,rs)}$	Boolean difference between $F_{p,q}$ and $F_{r,s}$
$\Lambda_{m,n}$	Collection of the (partial) overlapping face pairs in L_m and V_n
$\oplus_{m,n}$	Boolean intersection of each pair in $\Lambda_{m,n}$
$\Theta_{m,n}$	Boolean intersection of each pair in $\Lambda_{m,n}$

Tab. 1. Notations in the algorithm of multi-LOD product models.

3.2 Algorithms for Generating Multi-LOD Product Models

After introducing the idea of feature hiding with the switch face, this section describes algorithms that generate multi-LOD 3D product models. Tab. 1. lists the notations used in the algorithms. L_m represents all the faces of the 3D model in B-Rep at the construction step m. V_n represents the n-th form feature of the product model and a set of faces that make up the effective volume of the feature in B-Rep. $F_{n,b}$ stands for the b-th face of the feature V_n. $F_{(pq,rs)}$ indicates the Boolean difference between $F_{p,q}$ and $F_{r,s}$. $\Lambda_{m,n}$ is the collection of the (partial) overlapping face pairs in L_m and V_n. $\oplus$ and Θ indicates Boolean intersection and difference operators, respectively. The following algorithm is proposed for generation of the faces required to hide a feature n in the product model.

Algorithm 1

Generate the face collection L_m for the B-Rep model at the construction step m.

Repeat the following steps for the effective volume of feature n:

Generate the face collection for V_n.

Compute $\Lambda_{m,n}$ for L_m and V_n.

Calculate $\oplus_{m,n}$ and $\Theta_{m,n}$

The switch face(s) is expressed as $\oplus_{m,n}$

The other faces needed for hiding V_n in L_m are expressed as $\left[(L_m \cup V_n) - \Lambda_{m,n} \right] \cup \Theta_{m,n}$

3.3 Data Structure for Multi-LOD Product Models

The product models with LODs are designed for collaborative product development in a networked environment. Special data structures are required to facilitate model transfer during the collaboration process. Hence this work proposes a tree-like data structure that characterizes the feature hierarchy in the modeling process, i.e. the relation between features and the modeling steps, and the composing faces as well as the switch faces for each feature. Note that each node corresponds to a 2D face. For simplicity, we define a special code number, FaceID, for each node in the tree. FaceID consists of three digits. The first digit specifies the node level of the data structure. The second digit is the index number at a given level. The values of the last digit are listed in Tab. 2.. In addition, a set of symbols represent different types of nodes in Tab. 3..

Value	Definition
1	Face forming an additive feature
2	Face forming a subtractive feature
3	A transient face
4	A switch face of an additive feature
5	A switch face of a subtractive feature

Tab. 2. The implications of the third digit in FaceID.

Symbol	Name	Definition
▱	Root node	No parent, one or more children
▭	Leave node	One parent, no child
◯	Independent node	No parent, no child
◇	Transient node	One parent, one or more children

Tab. 3. The symbols for the tree nodes.

The following example explains how the data structure works. Fig. 4. is a simple block composed of six faces, which correspond to single nodes with FaceID 001, 011, 021, 031, 041, and 051. Suppose a step feature V1 is added in the next step. The previous nodes 011, 021, and 031 are divided into two child nodes, as they overlap with the effective feature volume being added. Their ID's also change to 013, 023, and 033, respectively, see Fig. 5.. Moreover, three nodes (101, 111, and 121) are produced from the difference operation. Three feature switch faces (104, 114, and 124) are also produced. The remaining nodes (132, 142, and 152) contain the faces making up V1 and they are represented as independent node. Note that the updates nodes after V1 has been created can be generated with Algorithm 1. Fig. 6. illustrates the changes in the data structure after another hole feature V2 has been constructed.
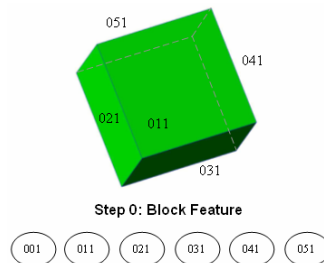

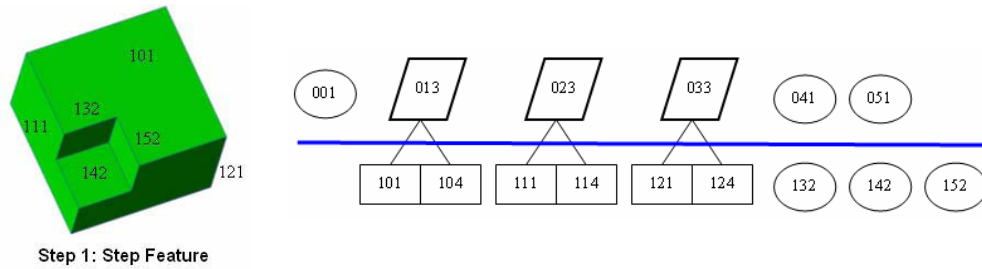
Fig. 4. Nodes corresponding to the block feature.

Step 1: Step Feature

Fig. 5. Nodes updated with respect to the creation of the step feature.
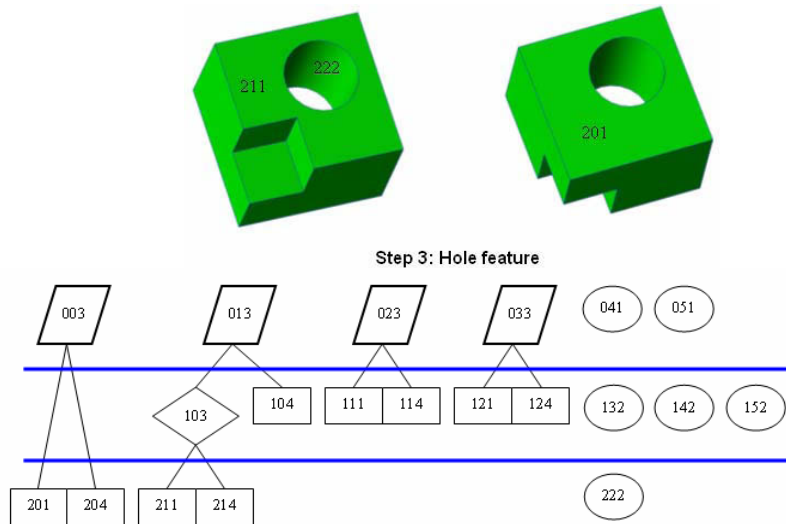
Step 3: Hole feature

Fig. 6. Nodes updated with respect to the creation of the hole feature.

The proposed data structure serves as a kernel in the process of generating multi-LOD models. The creation of a form feature changes the whole hierarchical structure and possibly the FaceID's as well as the type of some related nodes. New nodes may be created and their relations to the other nodes have to be determined simultaneously. Algorithm 2 provides a systematic approach for updating the data structure.

Algorithm 2
L_i represents the model at modeling step i. (L_0 is the initial model)
There are n form features ($V_1 \sim V_n$) be created in sequence.
get_FaceID($F_{m, n}$) returns the FaceID of the node $F_{m, n}$.
set_FaceID($F_{m, n}$) sets the FaceID of the node $F_{m, n}$.
Let $i = 0$, $L_{tool} = L_i$, if_overlap = false;
For each V_j ($j = 1$ to n)
For each element $F_{tool, a}$ of L_{tool}
If $i == 0$

    set_faceID($F_{tool, a}$) ;

EndIf
For each element $F_{j, b}$ of V_j

  set_faceID($F_{j, b}$) ;

If $F_{tool, a}$ and $F_{j, b}$ overlap

```
    if_overlap = true ;
    Set the third digit of FaceID as 3 ;
        EndIf
            EndFor
        EndFor
    Update_Intersection (⊕( _{tool, j})) ;
    Update_Difference (Θ( _{tool, j})) ;
    i = i + j ;
    L_{tool} = L_i ;
EndFor
Function set_faceID(F_{m, n})
    FaceID.first_digit = m ;
    FaceID.second_digit = n ;
    If form feature with positive effective volume
        FaceID.third_digit = 4 ;
    Else form feature with negative effective volume
FaceID.first_digit = 5 ;
    EndIf
EndFunction
Function Update_Intersection(⊕( _{m,n}))
    If form feature with positive effective volume
        Set the third digit of FaceID as 4 for each element of ⊕( _{m,n}) ;
    Else belong to a form feature with negative volume
        Set the third digit of FaceID as 5 for each element of ⊕( _{m,n}) ;
    EndIf
EndFunction
Function Update_Difference (Θ( _{m,n}))
    For each element of Θ( _{m,n})
        FaceID.first_digit = get_FaceID(the feature switch node).first_digit ;
        FaceID.second_digit = get_FaceID(the feature switch node).second_digit ;
        FaceID.third_digit = get_FaceID (parent node).third_digit ;
    EndFor
EndFunction
```

In the tree data structure, only leaf and independent nodes correspond to the actual faces that make up a 3D model in B-Rep. The root and transient nodes mainly connect the nodes containing the actual faces and maintain the hierarchy after each feature creation. With these information, the data structure can produce 3D models presented at different levels of detail according to collaboration requirements. From the viewpoint of data segmentation, the multi-LOD technique divides a complete mesh-based model into smaller data fragments of design implication. Proper combination of these fragments creates simplified variants of the original 3D model. Such a mechanism provides an effective way of protecting design contents, both semantically and visually. On the other hand, it serves as an foundation for CAD streaming, which is vital for real-time data transmission and visualization required in collaborative CAD [2]. CAD streaming does not require a complete download of the whole geometric model before the user can see the model, i.e. the model at a coarse resolution can be visualized while the data transfer is still in process. A possible use of the proposed LOD scheme is that the user first receive the data fragments corresponding to a simplified 3D model without any features. Depending on individual needs and the Internet quality, some or all of the features are transmitted in sequence at a later stage. Such a progressive transmission through incremental refinement reduces the workload of the

Internet traffic. Fig. 7. illustrates display of the 3D model at one level of detail by proper selection of the data fragments. Hiding the step feature is achieved by removing nodes 132, 142, and 152 and rendering the data corresponding to nodes 001, 041, 051, 101, 104, 111, 114, 121, and 124. Replacing nodes 104, 114, 124 with 132, 142, 152 discloses the feature in the model. Algorithm 3 describe the process of selecting the nodes for display/hiding a feature from the tree. The similar approach applies to the LODs in which more than one feature are to be hidden. Note that 2D mesh generation (Constrained Delaunay triangulations [7] in this work) will be performed on the faces selected according to the algorithm.
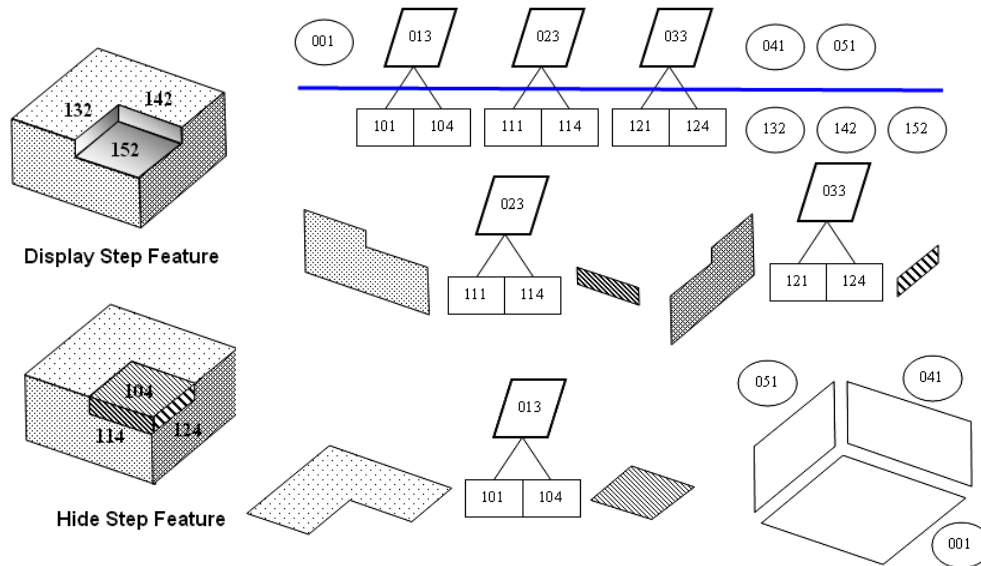

Fig. 7. Display/Hiding of the step feature from the data structure.

Algorithm 3
If the feature is set as Display
Find all the faces (nodes) belonging to the feature.
For each face, determine the following conditions according to the third digit of FaceID:

    The digit value = 1 or 2
        If Independent node   Discard the node
        else   Output the node
    The digit value = 3
        If Root node   Discard all the descendant nodes
        else   Discard the node
    The digit value = 4 or 5   Output the node
If the feature is set as Hidden
Find all the faces (nodes) of the feature.
For each face, determine the following conditions according to the third digit of FaceID:
    The digit value = 1 or 2   Output the node
    The digit value = 3   Discard the node
    The digit value = 4 or 5   Discard the node

4. IMPLEMENTION AND REAL SCENARIO

An Internet-based 3D collaborative assembly system is implemented based on the scheme and algorithms proposed in this research. This prototyping software adopts agent technologies [1] and the client-server structure. Fig. 8. illustrates the systematic framework   with JADE 3.2 (Java Agent Development Environment) [16] as the major implementation platform. Multiple users can participate the assembly process of 3D product simultaneously over the Internet with this system. A real scenario of 3D mold design is used to demonstrate the feasibility of the LOD concept for collaborative

product design. The mold consists of three major components: a connector, a piston, and a mold base. The connector is made up of two parts, a connecting rod and a cap, joint by bolts. The connector and the piston are joint together with a pin. The base mold is another single component that works with the piston. There are five different companies involved in the product development. C1 is a leading role in the collaboration process. In order to reduce cost and time, C1 outsources the piston and connector as a subsystem to a supplier D1, an ODM supplier, who is responsible for both the design and manufacturing of the subsystem. Moreover, D1 focuses on the development of the piston and subcontracts the connector to a next-tier supplier D2 due to its limited production capacity. In addition to C1, D1 and D2, there are two other part providers involved in this project. S1 is a fixture provider specialized at designing precision fixture to secure the base during the molding process. S2 is a standard part provider who supplies all the standard parts. S2 is invited to the collaboration because the company is a consign supplier [4] specified by the product owner C1.
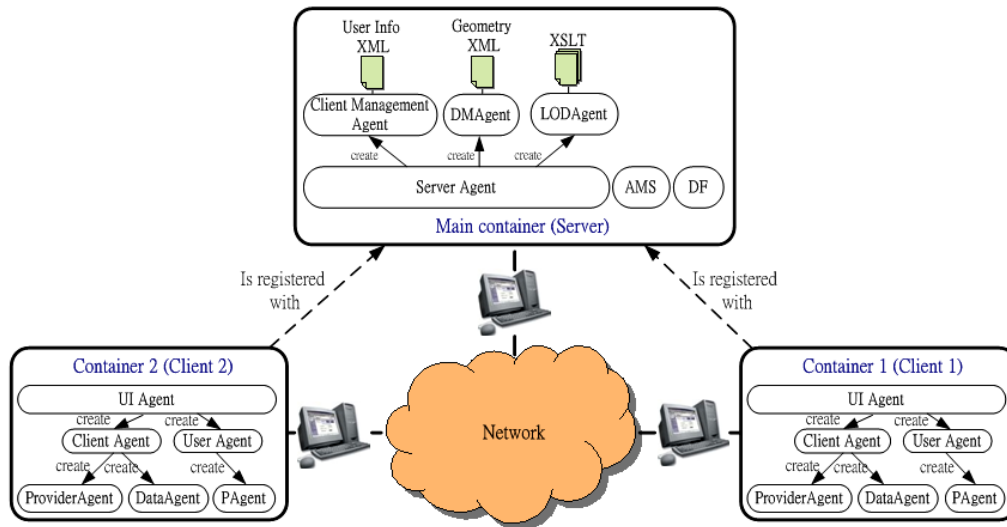

Fig. 8. The agent-based collaborative assembly system with LODs.

Such a complex scenario is commonly seen in distributed product development in Asia Pacific Region nowadays [6]. Tab. 4. summarizes all the participants in the collaboration and their individual roles. In addition, the product structure is also shown in Fig. 9. along with the owner of each component/subsystem. The product owner, D1, usually hosts the collaboration platform, initiates the process, and invites the other players in the supply chain for design review and quality assurance of the product. Obviously each company is responsible only for certain components and thus should be limited to the corresponding information. Disclosure of intellectual properties and interest conflicts may arise in the case where the entire product information is open to everyone. This is inevitable without a proper mechanism of information sharing. The LOD concept is proposed exactly to fulfill this requirement. Fig. 10.-13. show the connection rod, the cap, the piston, and the mold base at different LODs, respectively.

Collaborator	Tasks
Product Owner C1	Lead the entire product development Design and manufacturing of the base
First-tier Supplier D1	ODM supplier of C1 Design and manufacturing of the piston
Second-tier Supplier D2	ODM Supplier for D1 Design and manufacturing of the connector
Fixture Supplier S1	Design fixture for the base
Standard part provider S2	Supply all standard parts

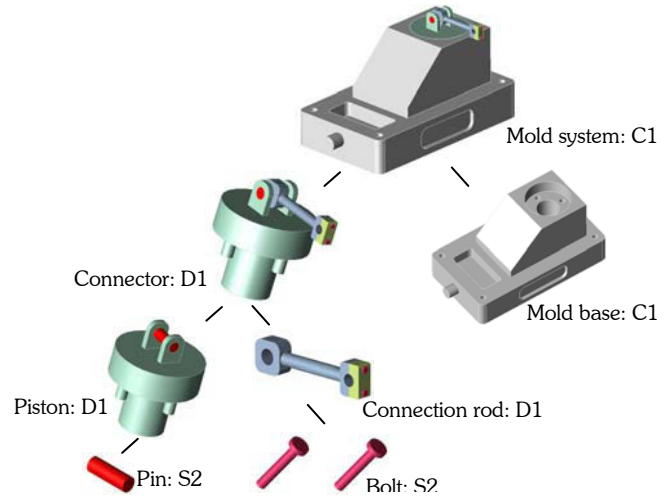Tab. 4. Collaborators and their individual tasks in the collaboration.

Fig. 9. The mold structure and the owner of each component.

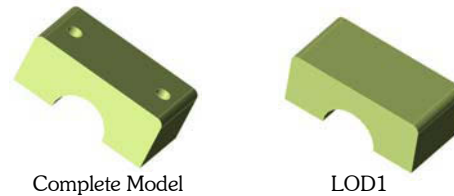Fig. 10. Connection rod at different LODs

Fig. 11. Cap at different LODs.

With the models at different LODs shown above, each collaborator can only access the product information related to individual development tasks. As the product owner, C1 has the highest viewing and modeling authorities. Details of every part are open to this leading role. Moreover, C1 need to determine the mapping between feature/LOD, part/collaborator, and LOD/collaborator. The ODM supplier D1 is allowed to view the piston and connector in full detail. The mold base contains design know-how of C1 and thus the complete model should not be available to the supplier. However, some portions of the part needs to be shared with D1, as the piston being developed by D1 interfaces with the mold base. One purpose of the collaborative assembly is to verify the interface between parts. To resolve the dilemma, a simplified 3D model of the base in which sensitive features have been hidden is offered to the supplier. Such a model at a lower LOD enables effective communication between C1 and D1 while the information security is guaranteed.

Similarly, the supplier D2 owns the complete model of the connector. The fixture supplier S1 is allowed to access to the product information only related to the task of the base fixturing. In addition, C1 grants S2 the permission to view some standard parts, which may affect the fixturing design. The online synchronous assembly starts after each company has obtained the appropriate product models according to individual roles and collaboration requirements. The overall

assembly process is shown in Fig. 14.. The horizontal axis represents the five collaborators and the vertical axis highlights the result at different phases during the assembly process. There are two major activities in the collaboration: the assembly of the piston and base and the assembly of the piston and connector.
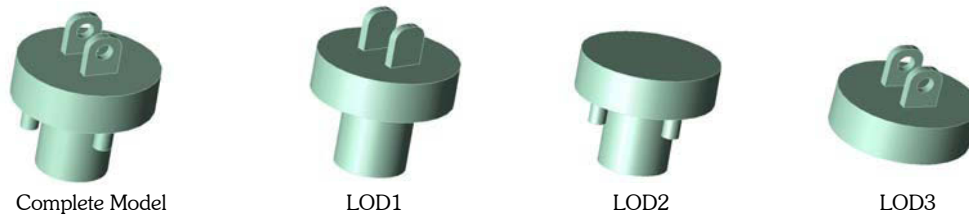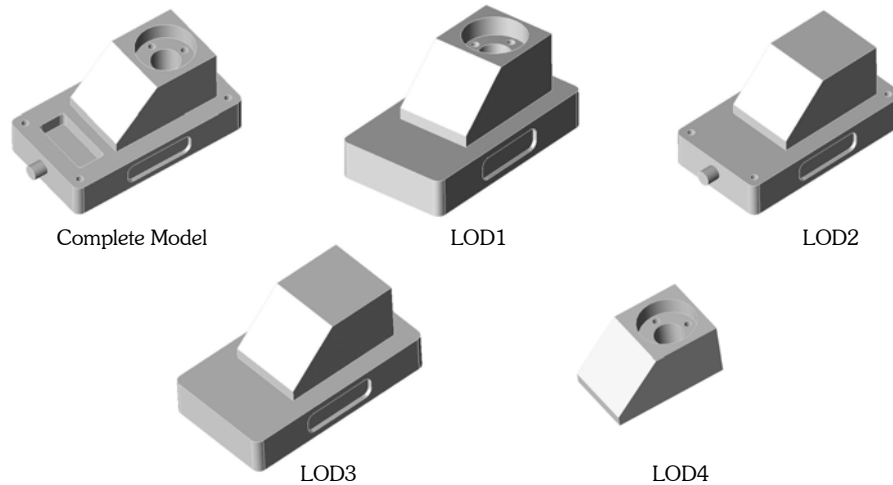

Fig. 12. Piston at different LODs.

Fig. 13. Mold base at different LODs.

5. CONCLUSIONS AND FUTURE WORK

This research proposes a new representation scheme for 3D product models with different levels of detail (LODs) based on integration of feature-based model and 2D mesh generation. Product information defined in terms of design features can be selectively hidden with the feature switch faces. Algorithms are developed for generation of the mesh data corresponding to any given LOD from a tree-like data structure, which characterizes the feature hierarchy, the relationships between features and LODs, and the switch faces of each feature. These technologies provide an effective method to ensure the security of information sharing and to improve the efficiency of data transmission in network-based collaborative 3D product development. An agent-based system is implemented to validate this work. This system allows multiple users to perform collaborative 3D assembly in real time. However, each user may access to product models at different LODs, depending on individual roles and collaboration requirements in the product development. A mold design scenario demonstrates the realization of the LOD concept.

This paper only addresses the preliminary result of applying the LOD concept to design collaboration. Non-geometric attributes such as dimensions and tolerances need be included in the LOD to enhance its practicality. The proposed scheme should be readily extendable to this. One interesting topic is to investigate the resolving mechanism of the conflicts induced by information imbalance as a result of different LODs. In addition, multi-agent technologies provide a promising implementation method for real world situations in collaborative design and commerce. More sophisticated functions should be developed into software agents like automatic negotiation and induction learning. Development of 3D CAD streaming techniques with the LOD scheme is also worth of pursuing in the future work. From a practical aspect, the computational performance of the proposed method needs to be improved to handle complex models.
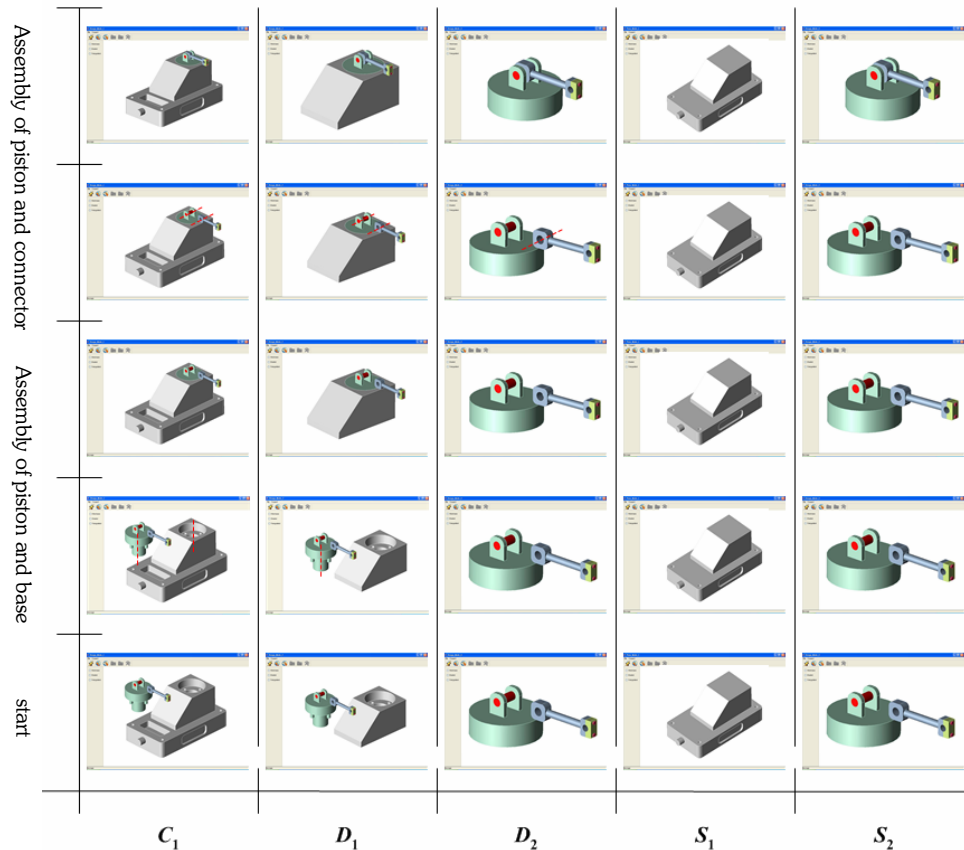
800

Fig. 14. Collaborative mold assembly based on the LOD scheme.

7. REFERENCES

[1]   Anumba, C.F., Ugwu, O.O., Newnham, L., Thorpe, A., A multi-agent system for distributed collaborative design. *Logistics Information Management*, Vol. 14, 2001, pp 355-366.
[2]   Bok, S.H., Senthil Kumar, A., Wong Y.S., Nee, A.C.Y., Model compression for design synchronization within distributed environments. *Computer-Aided Design and Applications*, Vol. 1, Nos. 1-4, 2004, pp 331-338.
[3]   Cera, C.D., Kim, T., Han, J.H., Regli, W.C., Role-based viewing envelopes for information protection in collaborative modeling. *Computer Aided Design*, Vol. 36, 2004, pp 873-886.
[4]   Chang, J.R., Chu, C.H., Collaborative product development in PCB industry. *International Journal of Electronic Business Management*, Vol. 2 ,No. 2, 2004, pp 108-116.
[5]   Chen, X.L., Fuh, J.Y.H., Wong, Y.S., Lu, Y.Q., Li, W.D., Qiu, Z.M., An adaptable model for distributed collaborative design. *Computer-Aided Design and Applications*, Vol. 2, Nos. 1-4, 2005, pp 47-55.
[6]   Chu, C.H., Chang, C.J., Cheng, H.C., Empirical studies on inter-organizational collaborative product development. to appear, *ASME Journal of Computing & Information Science in Engineering*, 2006.
[7]   Domiter, V., Constrained Delaunay triangulation using plane subdivision. *Proceedings of the 8th central European seminar on computer graphics*, Slovakia, 2004, pp 105-110.
[8]   Fuxin, F., Configurable product views based on geometry user requirements. *Computer-Aided Design and Applications*, Vol. 1, Nos. 1-4, 2004, pp 377-386.

[9]   Fuxin, F., S. Edlund, Categorization of geometry users. *Concurrent Engineering: Research and Application*, Vol. 9, No. 1, 2001, pp 15-22.

[10]   Garland, M., Heckbert, P.S., Surface simplification using quadric error metrics. *Proceedings of the 24th annual conference on computer graphics and interactive techniques*. 1997, pp 209-216.

[11]   Hoppe, H. Progressive meshes. *Proceedings of the 23rd annual conference on computer graphics and interactive techniques*. 1996, pp 99-108.

[12]   Koo, S., K. Lee, Wrap-around operation to make multi-resolution model of part and assembly. *Computers & Graphics*, Vol. 26, 2002, pp 687-700.

[13]   Li, W.D., Lu, W.F., Fuh, J.Y.H., Wong, Y.S., Collaborative computer-aided design – research and development status. *Computer Aided Design*, Vol. 37, 2005, pp 931-940.

[14]   Rossignac, J., Edgebreaker: connectivity compression for triangle meshes. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No. 1, 1999, pp 47-61.

[15]   Song, Y., K. Lee, Incremental transmission of B-Rep models through the network. *Computer-Aided Design and Applications*, Vol. 1, Nos. 1-4, 2004, pp 523-530.

[16]   JADE, http://jade.tilab.com/.