Journal of
Mechanical
Science and
Technology

Journal of Mechanical Science and Technology 21 (2007) 1235~1243

Web-Based CAD Viewer with Dimensional Verification Capability Through the STEP Translation Server

In-Ho Song[a], Sung-Chong Chung[b,*]

[a]*Graduate Student, Department of Mechanical Design and Production Engineering, Hanyang University, Seoul 133-791, Korea*
[b]*School of Mechanical Engineering, Hanyang University, Seoul 133-791, Korea*

(Manuscript Received November 20, 2006; Revised May 7, 2007; Accepted May 9, 2007)

Abstract

To design a CAD viewer on the Web, a file translation system which converts a native CAD file of a commercial CAD to the simplified CAD file is required. However, commercial CAD viewers have used simplified geometric data similar to the VRML format. Thus, it has been difficult to apply such data to the accurate design and dimension examination procedures. In this paper, a STEP translation server supporting most CAD native formats is devised. To apply the STEP file to the precise CAD viewer, an accurate geometric data extraction method from the STEP file is studied. A STEP translator classifies and defines geometric shapes according to the boundary information. Viewing and measurement functions of the developed system verify the geometries, interferences, dimensional errors, human factors, and form errors of the CAD data. In addition, design modification messages are transferred through the markup functions of the system. Collaborators are able to share their design ideas and opinions rapidly and remotely through the XML (eXtensible Markup Language) open architecture. The effectiveness of the developed system is confirmed through case studies.

Keywords: CAD native formats; Dimensional verification; Design verification; Standard for The Exchange of Product (STEP); eXtensible Markup Language (XML)

1. Introduction

Most manufacturing companies are trying to develop competitive products by improving quality, shortening the time to market, and reducing production costs. Collaborators in product development need to examine geometric forms and dimensions during the design process, and verify the dimensional errors of products during the fabrication process. Therefore, the concept of remote design and manufacture based on concurrent engineering is strongly required in manufacturing processes.

Since CAD models do not contain or represent final results required to design or fabricate a product, design examination and modification of the CAD models should be performed occasionally during the design process. These are performed through iterative verification and modification by designers, clients, manufacturing engineers, etc. They require close cooperation of product developers, but such cooperation is difficult for developers who are geographically separated.

Due to the evolution of electronic verification tools and the availability of the Internet as a medium for sharing and distributing CAD models, collaborators no longer need to be located in geographical proximity. Furthermore, collaborators may resolve design conflicts in the early design stage, which reduces the lead-time of product development as well as the manufacturing cost.

*Corresponding author. Tel.: +82 2 2220 0444, Fax.: +82 2 2298 4634
E-mail address: schung@hanyang.ac.kr

There are many collaborative design and verification systems. Huang et al. (2002) developed a standard Internet-based Design for X (DFX) shell that provides a framework in which many types of DFX tools are able to operate. Ahn et al. (2002) proposed an Internet-based CAD/CAM system for concurrent engineering. A real-time collaborative system was also developed for product design on the Internet environment with VRML and Java applet (Kan et al., 2001). In addition, several commercial Web-version viewing tools have been developed, such as Spinfire of Actify corporation (2006) and VisView of UGS corporation (2006). However, the commercial viewing tools only offer simple and rough dimensional examination functions. Moreover, as they do not support neutral formats, these tools have difficulty connecting the viewers to various CAD systems. To share and view design ideas over the Internet, Pang and Wittenbrik (1997) developed Cspray to enable distributed users to view data from independent or shared camera positions. Another development offers asynchronous two-way communication and interactive 3D views, along with a markup system for STEP data (O'Grady et al., 1998). However, this method uses polygonized CAD models (VRML) to represent the data format for visualization, and accurate dimensional information is impossible to obtain from the VRML models.

In this paper, a collaborative Internet-based design and dimension examination system is developed as shown in Fig. 1. After designing a product on a specific CAD system, designers register the CAD data to the Web server. The integrated server then automatically translates the native CAD file into a STEP file, and distributes the STEP file over the Internet. After finishing the first registration procedure of the design, remote collaborators, designers, clients, manufacturers, and so on who are related to the development of a new product are able to verify and confirm the geometric forms and dimensions of the design data on the Internet.

In conventional Web-based design verification systems, such as Spinfire and VisView, CAD data are translated into polygonized CAD models for the design examination. In the polygonized CAD representation, the CAD model is represented with triangular patches. As information losses occur during the translation process, native CAD files of various CAD systems should be translated into STEP files through a STEP translation server, and the STEP files must be applied directly to the design verification procedure. A STEP translator performs classification and design of geometric shapes for the precision design and dimension examination. The dimensional examination and markup data are stored in XML format to expand the information sharing capability among multiple users. The proposed system is executed on Internet Explorer without expensive commercial CAD/CAM hardware or software.

Since the ActiveX-server architecture is used in the system, no installation process is required and the maintenance is very convenient. Moreover, collaborators are able to conduct the design and dimension examination by using the Web-browser. This paper also describes the design methodology of the dimensional verification functions and markup modules applicable to the practical design examination. The performance of the developed system is confirmed and verified through case studies

2. System architecture

2.1 Overall structure

Figure 1 shows the structure of the proposed design and dimension verification system. Several users are accommodated simultaneously with the adoption of the ActiveX-server architecture. The integrated server consists of a database server, a STEP translator and an application server. The application server contains a dimensional verification module, so that the client software has the ability to access the server via the Internet (Chung and Song, 2003). ActiveX controls visualization of the STEP files, dimensional examination, and markup results. The search, file upload,

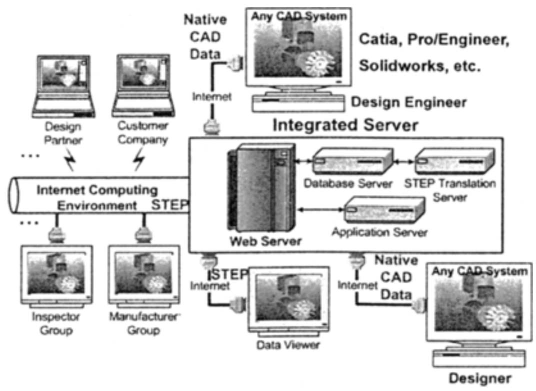

Fig. 1. Structure of the Web-based design and dimension verification system.

and user authentification functions in the proposed system are realized through Microsoft's Active Server Page and Structured Query Language mechanisms.

2.2 Significance of implemented techniques

Java is an ideal platform for network computing environments. It enables the creation of platform-independent software tools, and its ready availability enables users to load Java-based objects and link them together to create full applications. However, Java is unsuitable for developing full-scale CAD/CAM applications. There are no native Java libraries or APIs to support solid and geometric modeling or other infrastructures needed for CAD/CAM systems (Birngruber et al., 1999).

Microsoft's ActiveX is a set of technologies that enables software components to interact with one another in a networked environment, regardless of the language in which the components were created. An ActiveX control is a user interface element created with the ActiveX technology, which is also a technology based on a component object model (COM). COM is a type of software architecture that enables applications to be built from binary software components. In addition, COM is the underlying architecture that forms the foundation for higher-level software services, such as those provided by OLE, a technology for transferring and sharing information among applications. The ActiveX controls are small, fast, and powerful, and they make it easy to integrate and reuse software components. As they have lots of graphic libraries, they are more suitable than Java for representing complex 3D features. In this paper, the ActiveX technology is used to represent and verify 3D CAD data.

2.3 Structure of the client ActiveX

Client ActiveX of the design and dimension verification system is realized by using OpenGL and VC++. It consists of the following five modules:

(1) The CAD data translation module, which extracts features of the CAD/CAM system from the neutral data in STEP format.

(2) The 3D viewing module, which is realized by using the graphic libraries of Silicon Graphics for CAD data visualization.

(3) The dimensional verification module, which classifies the verification functions according to CAD objects by characterizing geometric entities for high-precision dimensional examination. This module has an open architecture so that measurement functions are easily incorporated into the system.

(4) The markup module, which manages annotation texts and markup notations.

(5) The XML file saving module, which contains both the markup and saving functions.

Figure 2 shows the configuration of the developed client ActiveX.

3. Translation of native CAD data

Drawings designed by various CAD systems are stored as the native files of each CAD system. Nowa-

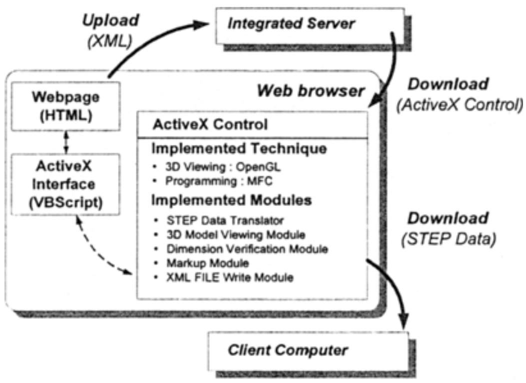

Fig. 2. Configuration of the client ActiveX for CAD data verification.

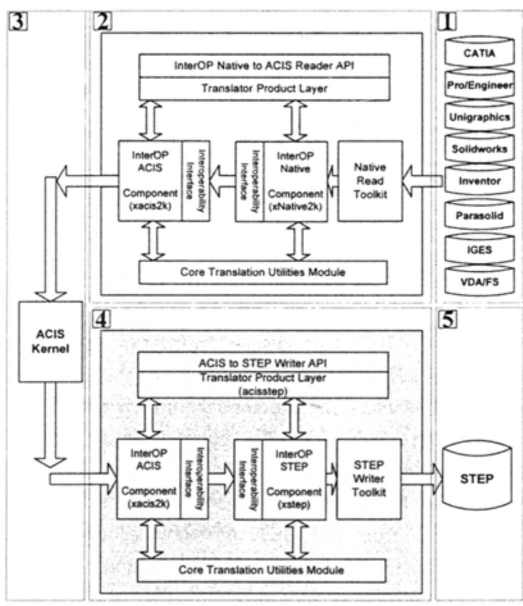

Fig. 3. Architecture of the STEP translation server.

Table 1. Mapped entities: CATIA, ACIS and STEP.

Native CAD entities	ACIS entities	STEP entities
CATLine	STRAIGHT	Line
CATCircle	ELLIPSE	Circle
CATEllipse	ELLIPSE	Ellipse
CATNurbsCurve	INTCURVE	Bspline curve
CATPlane	PLANE	Plane
CATCylinder	CONE	Conical surface or Cylindrical surface
CATNurbsSurface	SPLINE	Bspline surface
CATTorus	TORUS	Toroidal surface
CATSphere	SPHERE	Spherical surface
CATBody	BODY	Body

days, several CAD systems are used to develop a pro-product. For example, CATIA is used to design car bodies and Pro/Engineer is mainly used for the powertrain design. However, CAD systems cannot exchange native files with each other. In order to resolve this problem, a translation server that converts native files of various CAD systems into neutral files is required.

Figure 3 shows the structure of the translation server. Figure 3 $\boxed{1}$ shows native files produced from several CAD systems. Figure 3 $\boxed{2}$ represents the InterOp Reader of Spatial Inc., which converts native files into the ACIS kernel. Figure 3 $\boxed{3}$ shows the 3D modeling ACIS kernel. Finally, Fig. 3 $\boxed{4}$ represents the InterOp STEP Writer module, which generates STEP files through the ACIS kernel. All the processes described so far are automatically managed through the translation server. Various commercial CAD files are converted into STEP files through the proposed system. Table 1 shows the mapped entities of CATIA, ACIS and STEP. The CATIA entities are mapped to the ACIS kernel entities and then to the STEP entities (Spatial, 2006; Corney and Lim, 2001).

4. Geometric data extraction from STEP

STEP AP203 provides various models of expre-ssion for part geometry. The object of part geometry in STEP AP203 indicates an appropriate expressive object inherited from the object of the geometry expression, where the practical geometry information is linked to the object of the geometry expression. In addition, the basic model, which is related to geo-metry and topology, is described in part 42 of the EXPRESS language. There are three modes of geo-metry expression: the solid model, which is a boun-dary representation; the surface model; and the wire-frame model. The solid model is a type of geometry

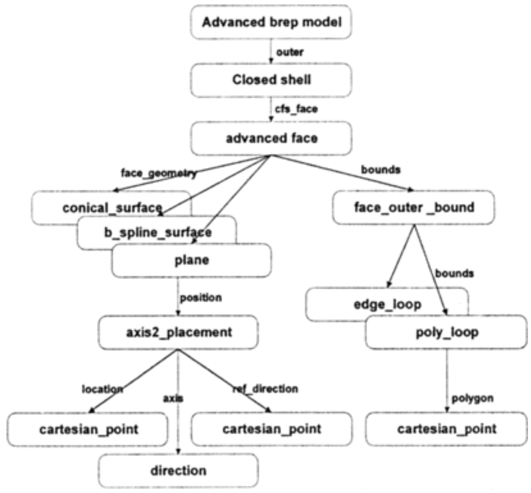

Fig. 4. Structure of the advanced brep model.

expression that includes all the topological informa-tion, and it contains the terms advanced_brep_model and facetted_brep_model, etc. The objective of this paper is to translate the term advanced_brep_model, which is widely used in CAD systems.

Figure 4 shows the overall structure of advanced brep model in STEP AP203. The basic concept of extracting geometric features and dimensional infor-mation depends upon whether the data instances in the physical STEP file correspond to the model schema. The data extraction process of the advanced brep model searches for lower entity information from the top entity according to the recorded pointers. The detailed processes are as follows: Firstly, the address pointers of a shell are obtained from the pointers of the advanced brep model, and the face information is extracted from the recorded pointers of the shell. The face is then classified into a plane and a spline surface according to the shape of the face. If the face is a plane, the boundaries of the face are given by poly lines composed of 3D curves or lines. The poly lines are represented as equations of 2D parametric curves on the u-v plane. If the face is a spline surface, boundaries are given by spline curves composed of both 3D spline curves and 2D para-metric curves on the u-v plane. Hence, all the boun-dary information of the face is obtained from the spline surfaces and the parametric curves. Secondly, the 3D coordinate values are derived from lattices obtained from the boundaries of the splines and the 2D parametric curves. Consequently, the STEP data is visualized in terms of coordinate values (ISO, 1994; Owen, 1997).

5. Classification of STEP data for dimensional verification

The analyzed STEP data must be classified into entities which are applied to the dimensional examination. In the STEP data, geometric entities applied to the dimensional verification are boundary data, which are composed of geometric elements, such as points, lines, circles and curves. For the dimensional examination, edge information must include either equations of the geometric elements or the essential parameters from which the equations are derived.

The point element includes the position information, and the line element includes the position information of both the start and end points. The position information is then applied to the line verification. The circle and arc elements, which are composed of conic curves are visualized through the parametric equations as

$$\begin{aligned} x &= R\cos\theta + x_c \\ y &= R\sin\theta + y_c \end{aligned} \tag{1}$$

where (x_c , y_c) and R are the central point and the radius of the conic element, respectively. Those parameters are extracted from the STEP data. The points on the conic element are calculated from the parametric equation in relation to a constant increment of $\Delta\theta$ from 0. When the central point and the radius of the conic element are used, the circle and arc examination is accomplished.

The spline curve information extracted from the STEP data is applied to the visualization of a curve element. As shown in Eq. (2), the proposed system visualizes the curve element by using a NURBS(non-uniform rational B-spline) equation. The length of the curve element is calculated as the summation of the length between two points on the curve element. The points on the curve are selected as the nearest point to the cursor (Chung and Song, 2004).

$$P(u) = \frac{\sum_{i=0}^{n} h_i P_i N_{i,k}(u)}{\sum_{i=0}^{n} h_i N_{i,k}(u)}, \quad 0 \le u \le n-k+2 \tag{2}$$

where

k : order of the NURBS curve

n : number of control points minus 1

P_i : control points of NURBS curve

h_i : weight of control points

$$N_{i,1}(u) = \begin{cases} 1 & t_i \le u < t_{i+1} \\ 0 & otherwise \end{cases}$$

$$N_{i,k}(u) = \frac{(u-t_i)N_{i,k-1}(u)}{t_{i+k-1}-t_i} + \frac{(t_{i+k}-u)N_{i+1,k-1}(u)}{t_{i+k}-t_{i+1}}$$

An efficient data structure is essential for the design of the dimensional verification through the analysis of STEP data composed of various entities. In this paper, a CAD entity is connected with the binary tree struc-

```
struct Group{
CString Name;      /* Name of group */
Solid *gsolid;     /* Pointer to list of solid */
Group *prevg;      /* Pointer to previous Group */
Group *nextg;      /* Pointer to next Group */
...
}

struct Solid{
CString Name;      /* Name of solid */
Group *sgroup;     /* Back pointer to group */
Face *sface;       /* Pointer to list of face */
Solid *prevs;      /* Pointer to previous solid */
Solid *nexts;      /* Pointer to next solid */
...
}

struct Face{
CString Name;      /* Name of face */
Solid *fsolid;     /* Back pointer to solid */
Loop *floop;       /* Pointer to list of loop */
Face *prevf;       /* Pointer to previous face */
Face *nextf;       /* Pointer to next face */
...
}

struct Loop{
Face *lface;       /* Back pointer to face */
Edge *ledge;       /* Pointer to list of edge */
Loop *prevl;       /* Pointer to previous loop */
Loop *nextl;       /* Pointer to next loop */
...
}

struct Edge{
Type Edgetype      /* Type of edge */
Loop *eloop;       /* Back pointer to loop */
Edge *preve;       /* Pointer to previous edge */
Edge *nexte;       /* Pointer to next edge */
...
}

...
```

Fig. 5. Examples of STEP translation data for the dimensional verification.

ure through the use of a self-reference structural pointer and the topological information of the STEP data. The tree structure with the self-reference strucural pointer is effective and fast for modification, insertion, and deletion of geometric data. In addition, besides enabling the data to be searched efficiently and quickly, the structure is extended to huge design works through the dynamic memory allocation.

In the binary tree structure, the terms *prev and *next are pointers representing topological relations of the tree structure. The proposed system facilitates the search for information and is extensible through the storing of additional data at the *next pointer. Figure 5 shows example entities of the dimensional verification.

6. Method of dimensional verification and markup modules

When the point selected on the 2D monitor coordinates matches the corresponding point of the 3D data obtained from an object selection method of OpenGL, an entity selection for the dimensional examination is accomplished. The object selection method of OpenGL is conducted through the selection of a small region specified by the cursor. In the small region, the specified object near the cursor is selected as an object.

The line length is the distance between a start point and an end point of the line. The distance between two points is calculated by computing the difference of the two points selected by cursors. The distance between two lines is computed from the minimum distance between the two lines selected by the cursors. The circle radius is extracted from the selected data of a circle. The angle and the radius of curvature of three points are calculated by using the point coordinates selected by cursors on an object. The distance between two circles is extracted from the difference between the centers of the two circles.

Using the markup function, someone on the Internet is able to upload review messages about verification results. Anyone who wishes to review the messages of others about the verification results confirms the markups through simple mouse clicks. Figure 6 shows the available markup functions. Using them, designers, manufacturers and clients are able to collaborate on the design and manufacturing in the distributed environment. They are not only able to share their opinions on the product design, but also able to verify the dimensions and geometries of the product. For improved the sharing capability among collaborators, the XML file type is incorporated into the proposed system. In addition, separators are applied to distinguish markup messages from the dimensional verification results. The new dimensional examination results or markup messages are then included in the markup file and followed by a separator. For numerous consultations with clients, a hyperlink function is also used in the markup function. Hence, the markup file is easily expanded by the use of the separators and hyperlinks (Chung and Song, 2005).

7. Storing dimensional verification and markup results

The dimensional verification and markup functions enable distributed clients to collaborate each other by expressing and sharing their opinions on the visualized geometric features. As shown in Fig. 7, clients are able to select the markup type such as a line, arrow, rectangle and so on. They leave markups on a visualized feature and express their thoughts in writings. In addition, a viewport of a client, which is

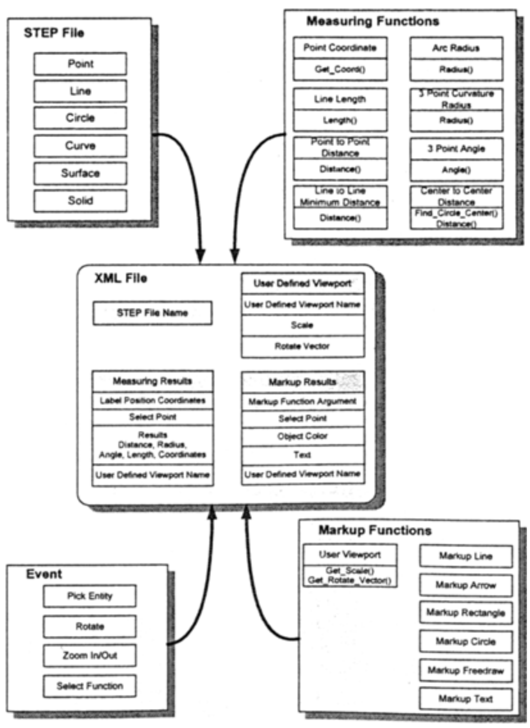

Fig. 6. Architecture of the dimensional verification and markup functions.

represented by a viewing scale, viewing center and rotational vector mathematically, is stored together with dimensional verification and markup results through the user viewport function. Other clients are not only able to understand another client's opinion, but are also able to look at the viewport when they select the user viewport name registered on the administrative tree-view. The selected viewport screen shows the markup and dimensional verification results from the previous verifiers. Using the markup function, collaborations between designers, manufacturers and buyers are available on the Internet.

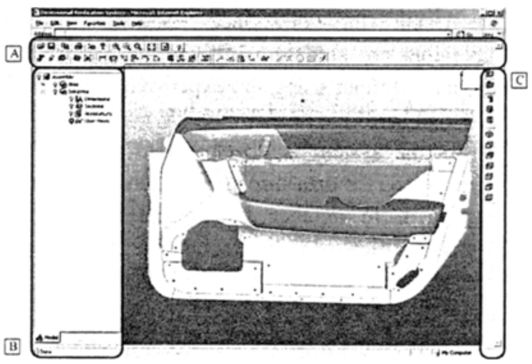

Fig. 7. Graphic user interface for design and dimensional verification module.


```
<?xml version="1.0" encoding="UTF-8" ?>
<Dimension_Markup>
<BaseDocument topNode="Drawing">"c:\stepdata\door.stp"
<UserViewPort_Name>ihsong</UserViewPort_Name>
<Viewing_Range>106.736</Viewing_Range>
<OriginCen>
<Vector x="1.776" y="1.065" z="14.689" /> </OriginCen>
<ViewingCen>
<Rotate ViewZoom="153.949" ViewCenterX="104.914"
 ViewCenterY="148.414"> </Rotate>
<Measuring_Result>
<Measuring_Function_Argment>3</Measuring_Function_Argment>
<Style styles="Color:255,255,255; Show:1"></Style>
<UserView_Name>3</UserView_Name>
<Point_to_Point_Dist unit="mm">651.539</Point_to_Point_Dist>
  <Point3d x="840.145" y="745.807" z="472.750" />
<Point3d x="1491.096" y="753.071" z="494.773" />
</Measuring_Result>
 :
</Dimension_Markup>
```


Fig. 8. XML document of markup result for dimensional verification.

In order to share information over the Internet, a standard data format that supports information exchange is required. XML is a simple and flexible text format used to exchange data on the Web. Figure 8 shows an XML example of the dimensional examination and markup results. Stored data in the XML file are STEP file name, user defined viewport information, dimensional verification results, markup results, etc. The verification results, composed of the labeled position coordinates, selected points, resultant values and user viewport name, have different information according to verification functions such as coordinates, line length, point to point distance, etc. Therefore, the verification results are classified and saved according to the verification function argument that identifies verification functions. A label, including a result value, is generated when the verification process is finished. The label position coordinate is a coordinate value shown on a monitor. The markup results are classified according to the markup function argument identifying markup types. Each result contains the selection of the point, color, text and user viewport name.

8. Case studies

Effectiveness of the proposed system is verified through design examination of a car door. Figure 7 shows a Graphic User Interface (GUI) of the developed viewing system. A, B and C of Fig. 7 are the dimensional verification and markup toolbar, the management treeview, and the graphic set up toolbar, respectively.

A design and dimension examination procedure is as follows: After receiving user authentication through the Web-page, collaborators, such as designers, clients and manufacturers, are able to use the viewing system. In order to start the system, they input their names, affiliations and date on the starting window. If they select the search file mode on the tree view, the system looks at other collaborators' review files. If they select a specific file, it is possible to review the verification results through the execution of ActiveX.

Figure 9 shows the dimensional examination results of a car door. If we select the start point A and the end point B after selecting the icon to compute the length between points, a dimensional verification result of the door panel is given as the length of 651.539 mm as shown in C. Collaborators are able to confirm the design and manufacturing errors. If a collaborator is not satisfied with certain design results, the collaborator may leave verification messages by using the markup function.

The radius of curvature of a car door is one of human factors. By using the curvature measurement function, a user can verify the design result. After selecting the icon for curvature measurement, the radius of curvature is calculated as 91.572 mm followed by the selection of D, E and F.

Figure 10 shows the user-defined viewport and markup managed by a viewport name specified on the tree structure in A. If a collaborator selects a viewport name in the treeview, a particular viewport with a specific viewing direction and zoom in/out window appear. Then, a collaborator is able to look into the dimensional verification results and markup contents through the viewport as shown in B of Fig. 10. Circles and the markup function's arrows shown in C deliver another verifier's message about the current design. Since this markup function is able to be added on the verification file as a hyperlink function, collaborators may deliver their opinions in detail and attach related documents and pictures, too.

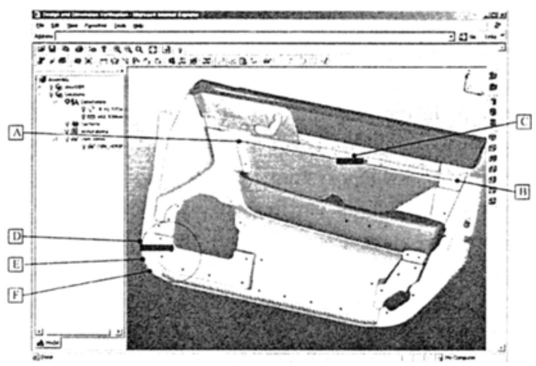

Fig. 9. Case study: dimensional verification.

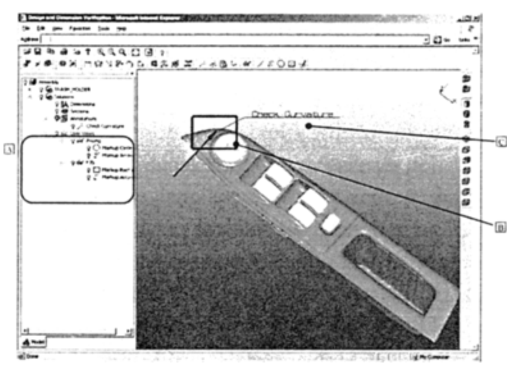

Fig. 10. Case study: user viewport and markup.

After reviewing the markup files written in XML format, the supervisor of the design is able to modify the current design and upload the improved design data on the Internet for further reviews.

9. Conclusions

Design methodology of a Web-based CAD Viewer with dimensional verification capability through the STEP translation server is proposed in this paper. Following conclusions have been obtained:

(1) Using the proposed STEP translation server, it is possible to examine design and dimension data constructed by various CAD systems on the Internet.

(2) Using the developed viewing system, collaborators are able to verify design results and dimensions without using the expensive commercial CAD/CAM hardware or software.

(3) As the system is developed to process and embody the STEP data directly in Active X without translating it to another file format, various dimensional verifications (dimensional verification with points including circles, lines, and curves) are possible on the Web environment.

(4) As the ActiveX-server architecture is applied to the developed system, installation of the developed system is conducted automatically through the Internet. Automatic updating of the ActiveX reduces the maintenance and upgrade cost of the system on the distributed environment. It is better than the conventional client-server architecture.

(5) Design and dimension examinations are conducted through the markup module. In order to expand the information sharing capability among collaborators, the dimensional verification data, markup data and user defined viewport are stored in XML formats.

(6) The validity and effectiveness of the developed system have been verified through case studies. Various parts of a car are examined to find out whether geometric shapes and dimensions coincide with design specifications or human factors.

After developing a native file converter suitable for the real-time environment, a real-time collaboration system will be studied for the future work.

References

Actify Inc., Available online at: www.actify.com (accessed 30-10-2006).

Ahn, S. H. et al., 2002, "Web-Based Design and Manufacturing Systems for Automobile Components: Architectures and Usability Studies," *International Journal of Computer Integrated Manufacturing*, Vol. 15, No. 6, pp. 555~563.

Birngruber, D., Kurshl, W. and Sametinger, J., 1999, "Comparsion of JavaBeans and ActiveX A Case Study," *STJA*, pp. 36~42.

Chung, S. C. and Song, I. H. 2004, "Web-based Design and Dimension Verification System Using STEP Files," *Transactions of the KSME(A)*, Vol. 28, No. 7, pp. 961~969.

Chung, S. C. and Song, I. H., 2003, "Web-based Precision Dimensional Verification System for Rapid Design and Manufacture," *ASPE's 18th Annual Meeting*, pp. 359~362.

Chung, S. C. and Song, I. H., 2005, "A Collaborative Design and Dimension Verification System on the Internet," *Transactions of NAMRI/SME*, Vol. 33, No. 1, pp. 493~500.

Corney, J. and Lim, T., 2001, 3D Modeling with ACIS, UK, Saxe-Coburg Publications.

Huang, G. Q., 2002, "Web-based Support for Collaborative Product Design Review," *Computer in Industry*, Vol. 48, pp. 71~88.

ISO, 1994, Industrial Automation Systems and Integration Product Data Representation and Exchange Part 203: Configuration Controlled Design, Geneva.

ISO, 2000, Part 28: XML Representation of EXPRESS Schemas and Data, Geneva.

Kan, H. Y., Duffy, V. G. and Su, C. J., 2001, "An Internet Virtual Reality Collaborative Environment for Effective Product Design," *Computers in Industry*, Vol. 45, pp. 197~213.

O'Grady, O. et al., 1998, "Distributed Concurrent Engineering: Internet-Based Interactive 3-D Dynamic Browsing and Markup of STEP Data," *Concurrent Engineering Research and Applications*, Vol. 6, No. 1, pp. 53~70.

Owen, J., 1997, STEP: An Introduction, 2nd ed, Information Geometers.

Pang, A. and Wittenbrik, C., 1997, "Collaborative 3D Visualization with Cspray," *IEEE Computer Graphics and Applications*, pp. 32~41.

Spatial Technology Inc., 2006, ACIS Geometric Modeler.

UGS Inc., Available online at: www.ugs.com (accessed 30-10-2006).

W3C, Extensible Markup Language (XML), Available online at: www.w3.org/XML/ (accessed 30-10-2006).