CyberEye: An Internet-enabled Collaborative Design Environment

by

Yi Zhuang

A thesis submitted in conformity with the requirements for the Degree of Master of Applied Science

Department of Mechanical and Industrial Engineering, in the University of Toronto

© Copyright by Yi Zhuang 2000
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-50427-1
Abstract

CyberEye: An Internet-enabled Collaborative Design Environment

An M. A. Sc. Thesis by Yi Zhuang

Department of Mechanical and Industrial Engineering, University of Toronto

2000

This thesis presents an Internet-enabled, platform-independent, distributed collaborative design environment by applying cutting-edge Java3D technologies and Internet technologies (such as ASP and Java Servlet). The proposed environment integrates the Product Visualization Module (the 3D CAD browser), the Product Information Management Module and the Team Management Module to support the execution of multidisciplinary team design on the Internet.

A novel approach adopting Java3D API and Java servlet technology is proposed to implement the 3D CAD browser. The Product Information Management Module is used to achieve shared understanding of product data and provides the facilities to control access to shared information and two-way information exchange between clients and server. The Team Management Module not only manages the activities of team participants, but also supports both asynchronous and synchronous distributed interactions amongst them.
To my beloved parent, grandmother and uncle
Acknowledgements

I would like to thank my supervisors Professors Li Chen and Ron Venter for their invaluable help in my research and in writing of the thesis. I would extend my thanks to my wife Becky, whose love and support made this thesis possible.

Special thanks are due to my colleagues in the Design and Manufacturing Integration Laboratory and the Mechanical and Industrial Engineering Department for their helpful discussions. I also want to particular thank Mr. Rex J. Ecclesto from Visteon (an enterprise of Ford Motor Company) for his kind assistance.

Finally, I gratefully acknowledge the financial support that I received from the University of Toronto.
# Table of Contents

Abstract .................................................................................................................................................. i

Acknowledgements ............................................................................................................................... ii

Table of Contents ................................................................................................................................. iii

List of Figures ......................................................................................................................................... ix

List of Tables ......................................................................................................................................... xiv

Nomenclature ......................................................................................................................................... xv

Chapter 1 Introduction ......................................................................................................................... 1

1.1 Motivation and Background .......................................................................................................... 1

1.1.1 3D Display and Manipulation on the Internet ......................................................................... 4

1.1.2 Collaboration and Coordination on the Internet ...................................................................... 5

1.2 Literature Survey ............................................................................................................................. 9

1.3 Thesis Overview ............................................................................................................................... 19

1.3.1 Scope and Contributions ........................................................................................................... 19

1.3.2 Thesis Objective .......................................................................................................................... 20

1.3.3 Thesis Organization ..................................................................................................................... 21
# Table of Contents

1.4 Concepts and Terminologies .......................................................... 22

Chapter 2 Enabling Technologies and Tools........................................ 28

2.1 Internet and WWW ...................................................................... 28

2.2 Client-Side Web Technologies......................................................... 31

2.2.1 HTML .................................................................................. 32

2.2.2 Client-Side Scripting ............................................................... 33

2.2.3 Java Applet ........................................................................... 35

2.2.4 ActiveX Control ..................................................................... 37

2.3 Server-Side Web Technologies ......................................................... 38

2.3.1 CGI ......................................................................................... 38

2.3.2 Active Server Pages ............................................................... 40

2.3.3 Java Servlet and JSP ............................................................... 42

2.4 Internet-based 3D Technologies ..................................................... 45

2.4.1 VRML .................................................................................... 46

2.4.2 Java and Java 3D ................................................................. 48

2.5 Supporting Tools ......................................................................... 53

2.5.1 Web Servers ......................................................................... 53
# Table of Contents

2.5.2 Web Browser ................................................................. 57

2.5.3 Development Tool .......................................................... 58

2.6 Summary ............................................................................. 61

## Chapter 3 Overview of CyberEye ................................................. 62

3.1 Infrastructure ........................................................................ 62

3.1.1 Product Visualization Module ............................................. 66

3.1.2 Product Information Management Module .......................... 68

3.1.3 Team Management Module ............................................... 69

3.2 Architecture ........................................................................ 73

3.2.1 Client-Side Frame ............................................................. 73

3.2.2 Server-Side Frame ............................................................ 75

3.3 Security ................................................................................. 77

3.4 Summary ................................................................................. 78

## Chapter 4 Product Visualization Module ...................................... 80

4.1 Architecture ........................................................................ 81

4.2 Implementation Details .......................................................... 84

4.2.1 Data Format ..................................................................... 84
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.2 Data Storage</td>
<td>85</td>
</tr>
<tr>
<td>4.2.3 3D Display</td>
<td>85</td>
</tr>
<tr>
<td>4.2.4 Communication</td>
<td>86</td>
</tr>
<tr>
<td>4.3 Interface and Operation</td>
<td>89</td>
</tr>
<tr>
<td>4.3.1 User Interface Description</td>
<td>89</td>
</tr>
<tr>
<td>4.3.2 User Activities Description</td>
<td>95</td>
</tr>
<tr>
<td>4.4 Data Input</td>
<td>98</td>
</tr>
<tr>
<td>4.4.1 I-DEAS</td>
<td>99</td>
</tr>
<tr>
<td>4.4.2 Pro/Engineer</td>
<td>100</td>
</tr>
<tr>
<td>4.5 Summary</td>
<td>100</td>
</tr>
<tr>
<td><strong>Chapter 5 Product Information Management Module</strong></td>
<td>102</td>
</tr>
<tr>
<td>5.1 Architecture</td>
<td>103</td>
</tr>
<tr>
<td>5.2 Database Structure</td>
<td>106</td>
</tr>
<tr>
<td>5.3 User Activities Support</td>
<td>109</td>
</tr>
<tr>
<td>5.4 Summary</td>
<td>114</td>
</tr>
<tr>
<td><strong>Chapter 6 Team Management Module</strong></td>
<td>116</td>
</tr>
<tr>
<td>6.1 Architecture</td>
<td>117</td>
</tr>
</tbody>
</table>
# Table of Contents

6.2 the Team Database Structure ......................................................... 119  
6.2.1 User Table .............................................................................. 120  
6.2.2 UserType Table ................................................................. 123  
6.2.3 Message Table ........................................................................ 125  
6.3 User Activities Flow ................................................................. 126  
6.4 Communication and Coordination ............................................... 135  
6.4.1 Email ....................................................................................... 136  
6.4.2 Web Forum ............................................................................... 143  
6.4.3 Online List ............................................................................... 150  
6.4.4 Instant Message Unit .............................................................. 155  
6.4.5 Chat Room .............................................................................. 164  
6.5 Summary ...................................................................................... 167  

Chapter 7 Conclusions and Future Development .................................... 168  
7.1 Summary and Conclusions ......................................................... 168  
7.2 Highlight of Contributions ......................................................... 170  
7.3 Research to Future Work ............................................................ 171  

Bibliography ....................................................................................... 174
# Table of Contents

Appendix A CyberEye Setup ........................................... 179  
CyberEye Viewer Setup .............................................. 179  
Setup for the Instant Message Unit ................................ 181  
Appendix B CyberEye Interface ...................................... 183
List of Figures

Figure 1-1. Collaboration time-place matrix ........................................................................ 7
Figure 1-2. Cosmo VRML player......................................................................................... 11
Figure 2-1. The client /server architecture on the Internet ................................................ 32
Figure 2-2. The architecture of CGI ..................................................................................... 39
Figure 2-3. The architecture of ASP.................................................................................... 41
Figure 2-4. The architecture of Java Servlet and JSP ........................................................... 43
Figure 2-5. A Typical Scene Graph in Java 3D .................................................................... 52
Figure 2-6. IDE of Visual InterDev ................................................................................... 59
Figure 3-1. Infrastructure of CyberEye ................................................................................ 65
Figure 3-2. Architecture of CyberEye ................................................................................ 72
Figure 3-3. Visit Log .......................................................................................................... 78
Figure 4-1. Architecture of CyberEye Viewer ..................................................................... 82
Figure 4-2. User Interface of CyberEye Viewer .................................................................. 83
Figure 4-3. The client component of CyberEye Viewer ...................................................... 86
Figure 4-4. Communication between Java applet and servlet ........................................... 88
List of Figures

Figure 4-5. User Interface of CyberEye Viewer ......................................................... 90
Figure 4-6. 3D Manipulation Toolbar ........................................................................ 91
Figure 4-7. Rotate Manipulation ............................................................................. 96
Figure 4-8. Pan Manipulation ................................................................................. 97
Figure 4-9. Zoom Manipulation .............................................................................. 97
Figure 4-10. VRML Converter ................................................................................ 99
Figure 5-1. The architecture of Product Information Management Module .............. 104
Figure 5-2. Models table opened in Access .............................................................. 108
Figure 5-3. Add new Product Model ....................................................................... 110
Figure 5-4. Upload VRML File ................................................................................ 111
Figure 5-5. Update product design ......................................................................... 112
Figure 5-6. Search Product Design ......................................................................... 114
Figure 6-1. Architecture of Team Management Module ......................................... 118
Figure 6-2. The Team Database Structure ............................................................... 119
Figure 6-3. User Table ........................................................................................... 120
Figure 6-4. UserType table .................................................................................... 125
Figure 6-5. Message Table .................................................................................... 126
List of Figures

Figure 6-6. User Activities Flow in CyberEye ........................................................... 127
Figure 6-7. Choose User Type .................................................................................. 128
Figure 6-8. Register Form ....................................................................................... 129
Figure 6-9. Load Photo ............................................................................................ 130
Figure 6-10. Login Page .......................................................................................... 131
Figure 6-11. Update page ....................................................................................... 132
Figure 6-12. Search User Function ......................................................................... 134
Figure 6-13. Logoff from CyberEye ........................................................................ 135
Figure 6-14. Automatic Email Notification Unit ....................................................... 138
Figure 6-15. Send Password Via Email .................................................................. 139
Figure 6-16. Generating Mailing List ..................................................................... 141
Figure 6-17. Mailing List ....................................................................................... 142
Figure 6-18. Architecture of the Web Forum ........................................................... 144
Figure 6-19. Main Screen of Web Forum ................................................................ 145
Figure 6-20. View Individual Message ................................................................... 146
Figure 6-21. Post a New Message .......................................................................... 147
Figure 6-22. Reply to a Message ............................................................................ 148
Figure 6-23. User Information ................................................................. 150

Figure 6-24. A typical HTTP transaction .............................................. 151

Figure 6-25. Online List ...................................................................... 152

Figure 6-26. The Architecture of Online List ....................................... 153

Figure 6-27. Online List on the Web Forum (Sun workstation) .......... 155

Figure 6-28. Architecture of Instant Message Unit ............................. 157

Figure 6-29. Instant Message Receiver .................................................. 158

Figure 6-30. Send instant message by using Online List ....................... 160

Figure 6-31. Send instant message by using user list ............................ 161

Figure 6-32. Architecture of Chat System ............................................. 165

Figure 6-33. Chat Room ..................................................................... 166

Figure A-1. Setup page in CyberEye ...................................................... 180

Figure A-2. Instant Message Client Setup Program ............................. 182

Figure B-1. Cover page of CyberEye ..................................................... 183

Figure B-2. Login page of CyberEye ..................................................... 184

Figure B-3. Web Forum in CyberEye .................................................... 184

Figure B-4. Replying to message ......................................................... 185
List of Figures

Figure B-5. Cover page of CyberEye Viewer .............................................................. 185
Figure B-6. CyberEye Viewer .................................................................................. 186
Figure B-7. Searching user ..................................................................................... 186
Figure B-8. Search result ...................................................................................... 187
Figure B-9. User information ................................................................................ 187
Figure B-10. Chat Room ....................................................................................... 188
Figure B-11. Sending instant message ................................................................. 188
Figure B-12. Instant Message Receiver ................................................................. 189
List of Tables

Table 2-1. Market Share for Top Web Servers in November 1999.......................... 54

Table 5-1. Fields in the Model Table........................................................................ 107

Table 6-1. Personal Information Fields .................................................................. 121

Table 6-2. Technical Information Fields .................................................................. 122

Table 6-3. Administrative Information Fields .......................................................... 123

Table 6-4. User Types in CyberEye .......................................................................... 124

Table 6-5. User Access Rights Table ....................................................................... 124
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>ASP</td>
<td>Active Server Pages</td>
</tr>
<tr>
<td>AWT</td>
<td>Java Abstract Windowing Toolkit</td>
</tr>
<tr>
<td>BBS</td>
<td>Bulletin Board Service</td>
</tr>
<tr>
<td>CGI</td>
<td>Common Gateway Interface</td>
</tr>
<tr>
<td>COM</td>
<td>Common Object Model</td>
</tr>
<tr>
<td>CORBA</td>
<td>Common Object Request Broker Architecture</td>
</tr>
<tr>
<td>CPD</td>
<td>Collaborative Product Development</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascading Style Sheets</td>
</tr>
<tr>
<td>DCOM</td>
<td>Distributed Component Object Model</td>
</tr>
<tr>
<td>DHTML</td>
<td>dynamic HTML</td>
</tr>
<tr>
<td>DUCADE</td>
<td>Domain Unified Computer Aided Design Environment</td>
</tr>
<tr>
<td>ECMA</td>
<td>European Computer Manufacturers Association</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>ICM</td>
<td>Interdisciplinary Communication Medium</td>
</tr>
<tr>
<td>IDE</td>
<td>Integrated Development Environment</td>
</tr>
<tr>
<td>IE</td>
<td>Microsoft Internet Explorer</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
</tr>
<tr>
<td>IGES</td>
<td>Initial Graphics Exchange Specification</td>
</tr>
<tr>
<td>IIS</td>
<td>Internet Information Server</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>JDBC</td>
<td>Java Database Connectivity</td>
</tr>
<tr>
<td>JSP</td>
<td>Java Server Pages</td>
</tr>
<tr>
<td>JSWDK</td>
<td>JavaServer Web Development Kit</td>
</tr>
<tr>
<td>JVM</td>
<td>Java Virtual Machine</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>MATE</td>
<td>Multi-user Architecture for Team Environments</td>
</tr>
<tr>
<td>MTS</td>
<td>Microsoft Transaction Server</td>
</tr>
<tr>
<td>ODBC</td>
<td>Open Database Connectivity</td>
</tr>
<tr>
<td>OpenGL</td>
<td>Open Graphics Library</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>Perl</td>
<td>Practical Extraction and Report Language</td>
</tr>
<tr>
<td>SCD</td>
<td>Synchronous Collaborative Design</td>
</tr>
<tr>
<td>SGML</td>
<td>Standard Generalized Markup Language</td>
</tr>
<tr>
<td>SMTP</td>
<td>Simple Mail Transfer Protocol</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Sockets Layer</td>
</tr>
<tr>
<td>STEP</td>
<td>The Standard for the Exchange of Product Model Data</td>
</tr>
<tr>
<td>TCP/IP</td>
<td>Transmission Control Protocol/Internet Protocol</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>VBScript</td>
<td>Visual Basic Script</td>
</tr>
<tr>
<td>VRML</td>
<td>Virtual Reality Modeling Language</td>
</tr>
<tr>
<td>WAN</td>
<td>Wide Area Network</td>
</tr>
<tr>
<td>WFDM</td>
<td>A Web-based Framework for Design and Manufacturing</td>
</tr>
<tr>
<td>WWW</td>
<td>World Wide Web</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
</tbody>
</table>

xvii
Chapter 1

Introduction

1.1 Motivation and Background

The traditional product design process is largely sequential. In this over the wall approach, team members from different disciplines do their tasks in isolation from each other. They have difficulty in exchanging design intents, decisions, and problems. As a result, conflicts and the problem areas can go unnoticed in the early stage of the design process. This can cause the product development cycle to be unnecessarily extended and will have an impact upon the quality of products.

Concurrent Engineering has widely been regarded as a preferred alternative approach to the traditional sequential engineering process for shorter development time, reduced cost and higher quality of the product. Concurrent Engineering is a systematic approach to integrated product development that emphasizes response to customer
Chapter 1 Introduction

expectations and embodies team values of cooperation, trust, and sharing [Miao and Haake, 1999]. This approach focuses on close cooperation amongst the work group members to accomplish the product development tasks. Concurrent Engineering provides the potential to improve the design efficiency, to detect the defects during earlier phases and thus shorten the product development life-cycle time [Prasad, 1997].

Collaboration and coordination has been the key issue in Concurrent Engineering. Concurrent Engineering teams are usually composed of team members specializing in a variety of disciplines, such as design, manufacture, management, and marketing. In some situations, it is important to even include suppliers and customers. Furthermore, these team members, suppliers and customer are increasingly more likely to be located in different places. As the complexity of a product increases, design activities, scattered across space, time and discipline, are becoming unmanageable. To maintain competitiveness in today’s global market, a collaborative design environment that efficiently coordinates a variety of design activities is crucial to the successful implementation of Concurrent Engineering.

A collaborative design environment can be defined as a computer networked environment, in which geographically distributed team participants work together on a shared basis to achieve a common set of consistent goals. A successful collaborative design environment needs a seamless communication channel, and a mechanism to ensure that team members do cooperate in a heterogeneous and distributed environment.
[Prasad, 1997]. The Internet is one of the best candidates to implement such a collaborative design environment based on cost and global accessibility.

The Internet is changing not only the way people live, but also the way engineering is performed. Although currently, the transmit speed of the Internet is slower than local network or intranet, the next generation of the high performance Internet is expected to have bandwidth over 1 gigabits/s (billion bits per second) [Chung, 1998]. This will make the exchanging of CAD-size software systems and data models on the Internet possible. Moreover, the rapidly developing Internet technologies, such as HTML (Hypertext Markup Language), Java, ASP (Active Server Pages), offer the potential for low cost and efficient communication, cooperation and coordination among geographically dispersed team participants.

While there is a significant increase in the use of Internet in business, there is relatively little work done on how these technologies can be used to support the collaborative design environment. This is primarily as a result of two key issues which distinguish the Internet-enabled collaborative design environment from the general E-commerce application. The first issue is the 3D display and manipulations on the Internet; the second is one of collaboration and coordination on the Internet.
1.1.1 3D Display and Manipulation on the Internet

The key issue for a collaborative design environment is how geographically distributed team participants can view and manipulate shared 3D product models through the Internet. HTML form, static text and images can satisfy the requirement for building the user interfaces of normal E-commerce applications. However, the interaction amongst the team participants in collaborative design environment is much more complicated. The collaborative design environment requires not only simple HTML form interfaces, but also an interactive graphic interface through which the 3D product models can be displayed and manipulated. Team participants should be allowed to view the realistic, rendered product model and to rotate, scale and translate realistic 3D product model using mouse or keyboard functions. The programs that fulfill such tasks are referred to as CAD browsers [Regli, 1997, 1].

Based on various observations, such a CAD browser should have the following features:

- **Internet-enabled**: The CAD browser should support for established Internet standards and protocols, for example, TCP/IP, HTTP and WWW.

- **Platform-independent**: The CAD browser should have the portability across the heterogeneous hardware and software environments present in the overall scope of engineering activities, such as Unix workstations, Windows system, Macintosh, or Linux.
Chapter 1 Introduction

• **Low cost:** Team participants should not be required to purchase expensive CAD systems to run the CAD browser. The CAD browser should be based on existing software, such as Web browsers. This will help to reduce the cost of the CAD browser and create the market.

• **Support of open standards:** The CAD browser should support open standards, because such open standards will allow participation by a wide variety of parties, including suppliers and customers. Examples of such standards include STEP and VRML.

• **Extensible:** The CAD Browser should be extensible so that more functions can be added; it should be easily integrated with other software systems, particularly, the collaborative design environment.

A CAD browser with the above attributes is critical to a successful collaborative design environment. The main objective in this thesis is the development of such a CAD browser.

1.1.2 Collaboration and Coordination on the Internet

The second key issue for collaborative design environment is that how to improve the collaboration and coordination amongst geographically located team participants by applying the newest Internet technologies.
Chapter 1 Introduction

Collaborative design environments involve the interaction among geographically distributed cross-functional teams (design, manufacturing, assembly, marketing and management etc.) as well as suppliers, contractors and customers. The collaboration and coordination among these team members and other participants is critical to a successful design. The nature of the collaboration and coordination among team participants can be divided into four types using the time/space matrix shown in Figure 1-1 [Saad and Maher, 1996] [Peters and Kress, 1997].

The first type of interaction is performed at the same time and at the same place. Presentation supporting software such as Microsoft PowerPoint is representative of the systems supporting this kind of interaction.

Interaction that happens at the same place but at different time is called asynchronous or non-real-time interaction. Examples of system supporting the asynchronous interaction include physical bulletin boards or share files [Saad and Maher, 1996].

When the interaction is performed at different places and at different time, it is called asynchronous distributed interaction. For example, electronic mail (email) and Web forum support asynchronous distributed interaction.

The final type of the interaction happens at the same time but in different places. This kind of interaction is called synchronous distributed interaction. Microsoft NetMeeting is one of the most popular commercial software supporting synchronous
distributed interaction. Users all over the world can use NetMeeting to talk, chat with each other and share drawings with whiteboard accessories (tools). They can even view video of each other. The main feature of such a system is the ability to support synchronous interaction amongst participants despite their geographical whereabouts.

![Figure 1-1. Collaboration time-place matrix](image)

Information sharing and activity synchronization are two key approaches to achieve collaboration and coordination amongst team participants scattered over a wide geographic range [Miao and Haake, 1999]. While information sharing is mainly based on asynchronous distributed interaction, activities synchronization requires synchronous
distributed interaction. Both asynchronous distributed interaction and synchronous distributed interactions are extremely important for a collaborative design environment.

Numerous tools that support asynchronous distributed interactions have been widely used in various Web applications, such as email, newsgroup, and Web forum. However, synchronous distributed interaction is difficult to implement in a Web-based collaborative design environment. Web pages are based on Hypertext Transfer Protocols (HTTP), which is a stateless protocol. When a Web server receives an HTTP request, it responds to this request and terminates the connection between the Web server and the client. The connection between client and server exists only during the request and response process. Although this feature makes HTTP efficient for distributing information, it makes HTTP difficult to track client sessions and to recall information. This is the reason why a Web-based application is not suitable for supporting synchronous distributed interactions.

In this thesis, the focus is dedicated to synchronous distributed interactions with particular concentration on how to apply the newest Internet technology to implement these interactions within a collaborative design environment. To implement information sharing and activity synchronization, a team management module is necessary to organize and control access to the shared information and to coordinate the activities of team participants. How to develop such a user management system and incorporate it into a collaborative design environment is another concern of the thesis.
Chapter 1 Introduction

1.2 Literature Survey

A number of efforts have been conducted in both industry and academia with regard to the development of the collaborative design environment. These are described in the approximate time sequence in which they were developed.

The SHARE Project

The SHARE project [Toye et al., 1993] is among the early efforts to develop a collaborative product development over the Internet. The project seeks to apply information technology in helping design teams gather, organize, re-access, and communicate both informal and formal design information to establish a shared understanding of the design and design process. Because the project was developed at the early stage of the Internet era, the shared information only includes notebooks, handbooks, and requirement documents. The 3D product model is not included. Also, electronic email is the only communication tool available.

MATE

MATE (Multi-user Architecture for Team Environments) is a multi-user architecture for collaborative design, in which existing applications, such as CAD, modeling programs and analysis programs can be shared by more than one designer [Saad and Maher, 1996]. Based on the nature of a shareable workspace for computer
supported collaborative design, two categories of workspaces are considered in the MATE: shared visual representation and shared underlying representation.

In MATE, a CAD system, specifically AutoCAD, is used as the basis for shared visual representation. Team members are able to use and edit the graphical models developed individually in a team design environment. However, the CAD system does not support the distributed 3D display and manipulation.

MATE adopts a video conferencing component to support synchronous interactions between designers. However, the video conferencing component is not integral to the system and runs separately.

**CyberView**

CyberView [Kim, 1998] is a prototype implementation of a distributed Concurrent Engineering system that is concerned with the storage, visualization, and communication of 3D data. The CyberView system is based on a new formalism for mapping STEP data into an Object Oriented Database schema and new algorithms for converting the STEP data into VRML. The system also allows team members to mark-up VRML worlds and examine the mark-up comments of others.

In CyberView, a commercial VRML viewer, Cosmos player, is used to achieve 3D display and manipulation on the Internet. Figure 1-2 shows the interface of Cosmos player.
Chapter 1 Introduction

Figure 1-2. Cosmo VRML player

CPD System

Bharadwaj et al. [Bharadwaj et al., 1996] proposed a two level model framework for collaborative product development environment and implemented in a prototype CPD (Collaborative Product Development) System. The proposed model framework supports CPD through the concept of shared product design web pages, shared geometric models (VRML models) and their associated hyper-linked product (on a shared database) models on the WWW. In addition to this, the architecture for the integration of various
product design services through the co-location of a distributed collection of Web servers has also been proposed.

Like CyberView, the display and manipulation of a 3D product model in the prototype CPD system relies on a third-party VRML player. One shortcoming of the CPD system is that it only provides information sharing facilities; it does not provide facilities to support interactions amongst users.

ICM

ICM (Interdisciplinary Communication Medium) is a prototype environment for interdisciplinary communication of design information in the conceptual design stage [Fruchter et al. 1996]. ICM integrates a shared graphical modeling environment and network-based services.

Like MATE, AutoCAD was used in the ICM as a shared graphic modeling environment. Internet email is used for routing change notifications.

DUCADE

DUCADE (Domain Unified Computer Aided Design Environment) is an experimental prototyping environment for multi-disciplinary collaborative design [Wang, 1997]. This system not only integrates the design tools and information in both mechanical and electrical domains, but also provides a vertical link to the manufacturing facilities.
Chapter 1 Introduction

Commercial CAD systems, such as AutoCAD and Pro/Engineer, have been encapsulated in the DUCADE system. However, this system did not provide an integrated facility to support 3D visualization of product model and communications amongst the team participants. Another limitation is that this system only focuses on a small-scale network environment in a single enterprise.

SCD model

Maher and Rutherford [Maher and Rutherford, 1997] presented a SCD (Synchronous Collaborative Design) model to integrate CAD and database management in a collaborative design session. The implementation of this model also uses AutoCAD as the shared CAD system. Email is used for the exchange of lexical information such as letter of intent, specifications, bills of quantities, etc.

WFDM

WFDM (A Web-based Framework for Design and Manufacturing) [Chung, et al., 1998] is a Web-based engineering framework that integrates various advanced design/manufacturing systems such as CAD, FEM and CAM. It provides relationships among product data, and coordinates with manufacturing process. Designers can access the framework through the Web browser and execute the various tools distributed on the networks using the Common Gateway Interface (CGI).

In the framework, I-DEAS is used to translate solid models into IGES (Initial Graphics Exchange Specification) format. The IGES files are then converted to VRML.
format to allow parts to be moved, rotated, and zoomed using a commercial VRML viewer within a Web browser.

**VisionManager**

VisionManager is a prototype for design evolution capture, visualization, and reuse in support of multi-disciplinary collaborative teamwork (Fruchter et al., 1998). VisionManager is implemented using AutoCAD as the geometric modeling environment, the Illustre Server for storing the product models, and Internet email for routing notifications. AutoCAD is also used to achieve the 3D display and manipulation of product models.

**SCOPE System**

SCOPE (Session-based Collaborative Process-centered Environment) is a prototype system that integrates a process modeling approach into an information sharing environment (Miao and Haake, 1999). However, the SCOPE system did not provide the necessary facility to share 3D product models. Furthermore, details on the implementation of facilities that support the asynchronous and synchronous interaction are not described.

**Workbench**

The Internet Integrated Workbench (Allen, 1999) was designed specifically for platform-independent design and manufacturing collaboration. The workbench consists of two principal parts: an Internet-accessible portion and a platform-specific
collaboration notebook. The Internet accessible portion runs on a local server and consists of a Project Area that contains project-specific information such as drawing, specifications and schedules, and a Document Vault, which stores file of any type that can be uploaded via browsers.

In this workbench, a team member can only view a snapshot of the part model in static image (bmp format). If the team member wishes to display and manipulate the 3D model of the part, he/she has to install expensive Pro/Engineering on his/her workstation and download the large-size .prt file (native format of Pro/Engineering).

The workbench supports asynchronous distributed interaction. For example, automatic email is used to inform project team members of the newly available CAD drawings. However, synchronous distributed interaction is not supported in Workbench. Team members still have to resolve the conflicting issues via the help of the face-to-face meeting.

After reviewing various research efforts, a number of drawbacks have been observed regarding the two key issues for the collaborative design environment: 3D display and manipulation, and collaboration and coordination on the Internet.

- **3D display and manipulation on the Internet**

  The issue of how to enable team participants to view and manipulate shared 3D product models through the Internet is not fully addressed in systems previously outlined. Some design collaborative design environments do not provide 3D display and
manipulation facilities (SHARE project, SCOPE system, DUCADE). Others only allow team participant to view the static images of the product models (Such as Workbench). Obviously, static images cannot satisfy the requirement of a collaborative design environment.

Many collaborative design environments adopt a commercial CAD system (AutoCAD, I-DEAS or Pro/Engineer) to achieve 3D display and manipulation, such as MATE, ICM, SCD model and VisionManager. However, this approach will add the software cost to the system since all team participants will be required to install expensive CAD systems on their computer, although they only use a small part of the functionality provided by the CAD system. Moreover, the CAD system is difficult to integrate within the collaborative design environment.

CyberView, CPD System and WFDM are based on VRML format and use a commercial third-party VRML viewer to enable 3D display and manipulation. Although the adoption of a commercial VRML viewer avoids the task of developing a CAD browser, this approach has the following limitations. First, the main purpose of VRML format is for rendering. It is only an approximation of the underlying geometric and topological data. VRML is sufficient if the product model needs only to be viewed, but it is impossible to conduct design modifications based strictly on VRML. Secondly, most commercial VRML viewers are designed for general-purpose use; they work well with normal E-commerce applications, but are not suitable for engineering applications.
Furthermore, since the commercial VRML viewers are developed by a third party, they cannot be customized or expanded to suit the specific requirements of a collaborative design environment. Moreover, commercial VRML viewers run separately and it would be difficult to integrate these into a collaborative design system.

- **Collaboration and coordination on the Internet**

  The existing collaborative design environments appear to have the following drawbacks with respect to the degree of collaboration and coordination on the Internet.

  First, some of the collaborative design environments were implemented on a small-scale networked environment (LAN, Intranet) rather than the Internet. These systems include MATE, ICM, DUCADE, SCD model, SCOPE system, and VisionManager. This satisfies the requirement for a single enterprise. With the increasing complexity of the designed product, collaborative design will involve many enterprises over a wide geographically range. Internet should be used as the infrastructure of collaborative design environment.

  Most of reviewed collaborative design environments support synchronous distributed interactions. Email is widely used as the communication tool to achieve collaboration and coordination in most of the existing systems. Although email is one of the most convenient and effective ways to transmit information on the Internet, it is not a real time communication tool and works well for asynchronous distributed interactions. But it is not suitable for supporting synchronous distributed interaction.
Synchronous distributed interactions are not fully supported in most systems. Some systems [MATE] adopt video conferencing software to support the synchronous distributed interaction; however, the video conferencing software is not integral to the system and is run separately.

Another drawback in the existing collaborative design environments is the lack of an integrated user management system. Most environments provide facilities for information sharing, but they do not have a team management system to control and coordinate the access to the shared information.

Finally, some of the collaborative design environments are not platform-independent. For example, the mentioned MATE, ICM and SCD only work on Unix workstations. However, not all team participants have access to expensive Unix workstations. Customers, for example, are most likely to use a Window's system on a PC. A collaborative design environment should be low-cost, platform-independent, supporting heterogeneous operation systems and platforms.
Chapter 1 Introduction

1.3 Thesis Overview

1.3.1 Scope and Contributions

This thesis is concerned mainly with two key issues for the development of collaborative design environment: the 3D display and manipulation of product models on the Internet, and the collaboration and coordination through the Internet.

3D Display and Manipulation on the Internet

In order to overcome the above drawbacks in the development of the CAD browser, a novel approach has been proposed in the thesis. In this approach, the cutting-edge 3D technology, Java 3D, is used to achieve the 3D display and manipulation. One of the new generation of server-side technologies, Java servlet, is used to connect with the database at the server. Java applet is used to build the user interface. Java applet and Java servlet communicate with each other to exchange product information. Based on this approach, a Java 3D based CAD browser called CyberEye Viewer has been developed. The CyberEye Viewer is an Internet-enabled, platform-independent, low-cost, open-standard, extensible 3D CAD browser.

Collaboration and Coordination through the Internet

To address the second issue, it was necessary to first investigate the enabling Internet technologies and to apply these technologies to the collaborative design
environment. A framework is proposed for the collaborative design environment that supports both asynchronous and synchronous distributed interactions. Based on this framework, the prototype, referred to as CyberEye, has been developed.

CyberEye is an Internet-enabled, platform-independent collaborative design environment to support the execution of multidisciplinary team engineering on the Internet. It integrates the Product Visualization Module (the CyberEye Viewer), the Product Information Management Module, and the Team Management Module to support the execution of multidisciplinary team design on the Internet.

The Product Information Management Module is used to achieve shared understanding of product data. It provides facilities to control access to shared information and a two-way information exchange between clients and server. The Team Management Module is mainly used to manage the activities of team participants. It also provides communication facilities to support both asynchronous and synchronous distributed interactions amongst team participants. Unlike some existing collaborative design environments, all these communication facilities are integrated into the system.

1.3.2 Thesis Objective

One of the objectives of this thesis is to investigate how cutting-edge 3D and Internet technologies will affect traditional engineering practice, with a particular focus
Chapter 1 Introduction

on how these technologies can be exploited with the implementation of the collaborative design environment.

Based on these investigations, an Internet-enabled, platform-independent collaborative design environment, identified as CyberEye, has been developed. CyberEye deals with the two key issues required for the development of collaborative design environment, namely, the 3D display and manipulation, and the collaboration and coordination on the Internet.

1.3.3 Thesis Organization

The thesis is organized as follows:

Chapter 1 presents the background, motivation and review of past work in the area of collaborative design environments. Chapter 2 introduces and analyzes enabling Internet technologies and tools. Chapter 3 presents an overview of the infrastructure and architecture of CyberEye. Chapter 4 describes the architecture and operation of the CyberEye Viewer and the implementation details. Chapter 5 presents the architecture, implementation and operation of the Product Information Management Module. Chapter 6 presents the architecture, implementation and operation of the Team Management Module. Finally, in Chapter 7, the conclusions, directions and suggestions for future development are outlined.
1.4 Concepts and Terminologies

A number of definitions are presented. Some of these are directly excerpted from Microsoft Press® Computer and Internet Dictionary © 1997, 1998 Microsoft Corporation [All rights reserved].

- **ActiveX**

  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. ActiveX is built on Microsoft's Component Object Model (COM). Currently, ActiveX is used primarily to develop interactive content for the World Wide Web, although it can be used in desktop applications and other programs. ActiveX controls can be embedded in Web pages to produce animation and other multimedia effects, interactive objects, and sophisticated applications.

- **ASP**

  ASP (Active Server Pages) is a Web-oriented technology developed by Microsoft that is designed to enable server-side (as opposed to client-side) scripting. Active Server Pages are text files that can contain not only HTML tags and text, as in standard Web documents, but also commands, written in a scripting language (such as VBScript or JScript), that can be carried out on the server. This server-side work enables a Web author to add interactivity to a document or to customize the viewing
or delivery of information to the client without concern for the platform the client is running. All Active Server Pages are saved with an .asp extension and can be accessed like standard URLs through a Web browser, such as Microsoft Internet Explorer or Netscape Navigator. When an Active Server Page is requested by a browser, the server carries out any script commands embedded in the page, generates an HTML document, and sends the document back to the browser for display on the requesting (client) computer. Active Server Pages can also be enhanced and extended with ActiveX components.

- **COM**

COM (Common Object Model) is a specification developed by Microsoft for building software components that can be assembled into programs or add functionality to existing programs running on Microsoft Windows platforms. COM components can be written in a variety of languages, although most are written in C++, and can be unplugged from a program at runtime without having to recompile the program. COM is the foundation of the OLE (Object Linking and Embedding), ActiveX, and DirectX.

- **Compiled Language**

Compiled language is a language that is translated into machine code prior to any execution, as opposed to an interpreted language, which is translated and executed statement by statement.
Chapter 1 Introduction

- **DHTML**

DHTML (Dynamic Hypertext Markup Language) is a technology designed to add richness, interactivity, and graphical interest to Web pages by providing those pages with the ability to change and update themselves dynamically, that is, in response to user actions, without the need for repeated downloads from a server. Examples of DHTML actions include moving graphics on the page and displaying information in response to mouse movements or clicks. A number of vendors, including Microsoft and Netscape, have developed their own versions of DHTML and have submitted them to the World Wide Web Consortium (W3C) for possible inclusion in the Document Object Model (DOM) specification being developed by W3C.

- **Interpreted Language**

Interpreted language is a language in which programs are translated into executable form and executed one statement at a time rather than being translated completely (compiled) before execution.

- **Java 3D**

Java 3D is a network-centric, scene graph-based API, that has revolutionized 3D graphics application developments. Java 3D combines the advantages of the Java language and OpenGL, making the development of Internet-based 3D software very easy.
• JavaScript

JavaScript is a scripting language developed by Netscape Communications and Sun Microsystems, Inc. that is loosely related to Java. JavaScript, however, is not a true object-oriented language, and it is limited in performance when compared to Java because it is not compiled. Basic online applications and functions can be added to Web pages with JavaScript, but the number and complexity of available API functions are fewer than what is available with Java. JavaScript code, which is included in a Web page along with the HTML code, is generally considered easier to write than Java, especially for novice programmers. A JavaScript-compliant Web browser, such as Netscape Navigator, is necessary to run JavaScript code.

• JDBC

JDBC technology is an API that lets you access virtually any tabular data source from the Java programming language. It provides cross-DBMS connectivity to a wide range of SQL databases, and now, with the new JDBC API, it also provides access to other tabular data sources, such as spreadsheets or flat files. The JDBC API allows developers to take advantage of the Java platform's "Write Once, Run Anywhere" capabilities for industrial strength, cross-platform applications that require access to enterprise data. With a JDBC technology-enabled driver, a developer can easily connect all corporate data even in a heterogeneous environment (Directly excerpted from the Web site of Sun Microsystems Inc.)
• **Plug-in**

Plug-in is a software component that plugs into the Netscape Navigator. Plug-ins permit the Web browser to access and execute files embedded in HTML documents that are in formats, which the browser would normally not recognize, such as animation, video, and audio files.

• **Scripting Language**

Scripting language is a simple programming language designed to perform special or limited tasks, sometimes associated with a particular application or function. An example of a scripting language is Perl.

• **SMTP**

SMTP (Simple Mail Transfer Protocol) is a TCP/IP protocol for sending messages from one computer to another on a network. This protocol is used on the Internet to route e-mail.

• **Winsock**

Winsock is short for Windows Sockets. It is an application programming interface standard for software that provides a TCP/IP interface under Windows. Winsock has gained the general support of software developers, including Microsoft.

• **URL**
URL (Uniform Resource Locator) is an address for a resource on the Internet. URLs are used by Web browsers to locate Internet resources. A URL specifies the protocol to be used in accessing the resource (such as http:// for a World Wide Web page or ftp:// for an FTP site), the name of the server on which the resource resides (such as www.whitehouse.gov), and, optionally, the path to a resource (such as an HTML document or a file on that server).

- **VBScript**

VBScript (Visual Basic Script) is a subset of the Visual Basic for Applications programming language, optimized for Web-related programming. As with JavaScript, code for VBScript is embedded in HTML documents.
Chapter 2

Enabling Technologies and Tools

This chapter describes some of the cutting-edge Internet technologies and tools that will affect traditional engineering practice, with a particular focus on how these technologies can be exploited in the proposed CyberEye, an Internet-enabled, platform-independent collaborative design environment.

2.1 Internet and WWW

Rapid advances in Internet technology are changing not only the way of life for many people, but also the way engineering is performed. Compared with LAN (Local Area Network) and WAN (Wide Area Network), the Internet has the added advantage of low cost and global accessibility. The WWW is one of the services operating on the Internet. The WWW is actually an information infrastructure that consists of interlinked hypertext documents residing on HTTP servers around the world. The Internet and
WWW offer the potential for improved communication and cooperation for engineering practice.

Although the Internet has already begun to affect the engineering practice, it has not as yet revolutionized traditional engineering practice. There are two reasons for this. First, the transmission speed of the Internet is slow compared with LAN and WAN; however, with the next generation high-speed Internet, it will be possible to handle the exchange of CAD-size software systems and data models. Secondly, the WWW was initially designed to deliver static text and graphics. Design and manufacturing applications software require an interactive graphic interface and the capability to display and manipulate 3D product models. With the advent of many new Internet technologies, such as Java applet, VRML, Java 3D and Java Servlet, it is now possible to generate interactive graphic interfaces and display 3D models in the Web browser.

The Internet and WWW will influence the engineering practice in the following ways:

- **Global collaboration and data sharing**

  The Internet makes the global sharing of product data possible by providing an information infrastructure that is available worldwide. Online product catalogs that provide access to all standard components are already available on the Internet. Because of the low cost and global accessibility of the Internet, access to and the sharing of data is not limited to engineers, but is also available to management
teams, purchasing teams, contractors, suppliers and even customers. Additionally, the Internet greatly improves the collaboration amongst engineers by providing numerous multi-media communication tools, including email, Web pages, online chat, Internet phone and video conferencing.

- **Leverage legacy systems**

  Numerous companies still use legacy systems developed many years ago. The introduction of the Internet will not cause these legacy systems to be abandoned. Instead, the Internet will play a critical role in leveraging the legacy software systems. For example, Java can be used to build generic front-ends to formerly proprietary systems. CORBA can wrap legacy code and programs. The Internet will keep thousands of critical legacy software systems alive and useful [Regli, 1997, 2].

- **Breaking apart of large CAD systems**

  Commercial CAD systems (such as Pro/Engineer and I-DEAS) are becoming larger and larger with increased functionality. At the same time, prices are also increasing. Some users only use very a small portion of the functions, but need to acquire and buy the whole system. For example, customers may only need to view 3D product models, which is a basic function of the CAD system.

  The Internet allows large CAD systems to be broken into small components. These components will be integrated with other Internet tools or services such as the Web browser, email and Web authentication. Users can access these components through
the Internet on a pay-per-use basis. In this way, users do not need to buy the entire CAD system, but can pay only for those functions that they use.

2.2 Client-Side Web Technologies

Figure 2-1 illustrates the client /server architecture of the Internet. It consists of the client-side frame and the server-side frame. Web technologies that are applied at the client-side frame are called client-side Web technologies and are mainly used to create the front-ends for Web applications. Currently available client-side Web technologies include HTML/DHTML, client-side scripting, Java applet and ActiveX control. All these technologies are implemented inside the Web browser.
Chapter 2 Enabling Technologies and Tools

Figure 2-1. The client/server architecture on the internet

2.2.1 HTML/DHTML

HTML is a markup language used for documents on the World Wide Web. Most Web browsers, notably Netscape Navigator and Microsoft Internet Explorer, recognize HTML. HTML presents texts and graphics on the Internet. HTML is the foundation of all Web applications. DHTML (Dynamic Hypertext Markup Language) is a technology designed to add richness, interactivity, and graphical interest to Web pages.
2.2.2 Client-Side Scripting

Since HTML is a simple text markup language, it cannot do interactive tasks like responding to the user, making decisions, controlling the Web browser and doing calculations. A more complex language, a scripting language, is required.

Scripting languages are interpreted languages, and therefore different from compiled languages, such as C and Java. Scripting languages do not need be compiled but require script hosts to interpret and execute the code. For client-side scripting, the script host is the Web browser.

There are two main client-side scripting languages: JavaScript and Visual Basic Script (VBScript). JavaScript is the most popular client-side script language, and is supported both by Netscape Navigator and Microsoft Internet Explorer. JavaScript has been standardized by ECMA (European Computer Manufacturers Association), and therefore JavaScript is also called ECMAScript. VBScript is a subset of Visual Basic optimized for Web-related programming. The disadvantage of VBScript is that it only works with Microsoft Internet Explorer. Therefore, only JavaScript is used as client-side scripting languages in CyberEye.

JavaScript adds interactive and dynamic features to static HTML pages. Some of these features find their applications in the collaborative design environment. The following JavaScript applications are implemented in CyberEye:
Chapter 2 Enabling Technologies and Tools

- **Control the Web browser**

  JavaScript can control Web browsers. For example, in CyberEye, JavaScript is used to display help information on the status bar of the Web browser. When a user logs off, JavaScript is used to automatically close the browser window.

- **Detect Web browser type**

  Netscape Navigator and Microsoft Internet Explorer have different supports for DHTML and client-side scripting. To solve the problem of incompatibility, JavaScript can be used to detect the type and the version of the Web browser, and generate the correct DHTML statements and client-side scripting for different browsers.

- **Validate form content**

  JavaScript can be used to validate the content of a form before it is sent to the Web server. Because client-side validations occur on the client-side, the speed is much faster than server-side validation.

  All forms in CyberEye are validated using JavaScript. For example, when a new team participant registers, JavaScript is used to check if the participant filled out the required information in the register form. If not, a dialog is displayed to notify the participant that more information is required.

- **Create animation and the rollover effect**
Chapter 2 Enabling Technologies and Tools

JavaScript can control images in the Web document to create animation or the rollover effect, making the Web page more attractive. Rollover means images will change when you move the mouse pointer over them. Rollover effects are applied on all buttons on the side menu of CyberEye.

2.2.3 Java Applet

In most E-commerce applications, the interaction between the client and the server can be met with several static requests. HTML works well for a user interface of such application. However, the interaction between the client and server in a collaborative design environment is much more complicated. The collaborative design environment requires not only a simple HTML interface, but also an interactive graphic interface in which the 3D product model can be displayed and manipulated. Simple HTML forms are not sufficient. Only Java applets can be used fulfill this task.

Java applets are Java programs that are to be embedded in a Web page. They are different from ordinary stand-alone programs called Java applications, because Java applets run in the Web browser. While HTML presents static text and graphics and client-side scripting adds some dynamic effects, Java applets provide an interactive graphic interface to the Web page. Moreover, because Java applets are programmed using Java, they can accomplish much more complicated functions than HTML can.

Furthermore, Java applets provide the following benefits:
• **Universal access**

Because Java is a platform-independent language, Java applets can run on most platforms, including Unix systems, Windows systems and Linux systems.

• **Low cost**

Java applets are stored on the Web server. Each time a Java applet is requested, it is automatically downloaded from the Web server to the user’s computer and runs inside the Web browser. Users do not need to install the Java applet on their computer. This can reduce the cost.

• **Database connection**

Although Java applets run at the client-side, they can access information in the database by communicating with Java servlets, which connect to the database through JDBC (Java Database Connectivity) interface.

For these reasons, the client component of CyberEye Viewer is implemented as a Java applet. The client component can display not only 3D product models, but also product information stored in the database by communicating with a Java servlet.
2.2.4 ActiveX Control

ActiveX control is a specification developed by Microsoft that allows ordinary Windows programs to run within a Web page. It provides an alternative way to present an interactive graphic interface in the Web browser. ActiveX programs can be written in Visual C++ or Visual Basic.

Like Java applets, ActiveX controls are downloaded and executed by the Web browser. However, unlike Java applets, which need to be downloaded each time, ActiveX controls only need to be downloaded the first time they are used. After that, they are installed on the client computer. As a result, ActiveX controls run faster than Java applets. However, only Microsoft Internet Explorer supports ActiveX controls internally. If ActiveX controls need to run in Netscape Navigator, an ActiveX plug-in has to be installed.

The Chat Room client component in CyberEye is implemented as an ActiveX control. It connects with the Chat Server to provide real time communication amongst team participants.
2.3 Server-Side Web Technologies

A server-side Web program is the same as any other program with a few important exceptions. First, the program should be able to be invoked by the Web server and there must be a way in which the Web server passes data to the program.

The selection of server-side Web technology is extremely important for the Internet-enabled collaborative design environment. A few years back, the only server-side Web technology available was Common Gateway Interface (CGI). In recent years, several new server-side Web technologies have been developed, including Active Server Pages (ASP) and Java Servlets and JSP (Java Server Pages).

2.3.1 Common Gateway Interface (CGI)

CGI is the first generation of the server-side technologies and by far the most common. By acting as a gateway between the user request and the data that it requires, CGI enables the contents of a web page to be generated dynamically by a program typically written in a scripting language such as Perl or a programming language such as C. Perl (Practical Extraction and Report Language) is a scripting language widely used by Web application developers.

The biggest problem with CGI is its poor performance. CGI will create a new process for each HTTP request (Figure 2-2). For a Web site with large number of
requests, the repetition of process creations is inefficient and may exhaust the resources of the server very quickly. Another disadvantage of CGI is its weak security. If a programming language (such as C) is used to implement CGI, visitors must be given execution rights to the directory that contains the program. This can cause serious security problems. For these two reasons, CGI is not adopted in CyberEye.

Figure 2-2. The architecture of CGI
2.3.2 Active Server Pages

Active Server Pages (ASP) is the latest server-side technology from Microsoft for creating dynamic and interactive Web pages. Compared with CGI, ASP is easier to implement and overcomes the performance problem of CGI. Furthermore, the ASP environment is extensible by utilizing COM (Common Object Model) components technology.

ASP acts like glue, by sticking everything in the Web application together. An ASP file contains HTML statements, client-side scripts and server-side scripts. The architecture of ASP is shown in Figure 2-3. Since server-side scripts are executed by the ASP service on the Web server, Server-side scripts can connect to the database through the ODBC interface and implement transaction functions with the help of Microsoft Transaction Server (MTS). The result sent back to the browser only contains HTML statements and client-side scripts.

One disadvantage to the Active Server Pages is that they can only be used on a Microsoft Web server (Internet Information Server, Personal Web Server) on a Microsoft operating system (Windows 95/98, Windows NT).

ASP is used to implement the Product Information Management Module and the Team Management Module of CyberEye.
Figure 2.3: The Architecture of ASP
Chapter 2 Enabling Technologies and Tools

2.3.3 Java Servlet and JSP

A Java servlet is a server-side program that services HTTP requests and returns results as HTTP responses. Servlets provide Web developers with a simple, consistent mechanism for extending the functionality of a web server.

Like Java applets, Java servlets are also implemented with Java programming language. However, there are three basic differences between applets and servlets:

- Applets run at the client-side, whereas servlets run at server-side.

- Applets run in the Web browser, whereas servlets require a Web server that supports servlets.

- Applets have the graphical interface displayed in the Web browser window, whereas servlets do not have any visual interface.

Figure 2-4 depicts the architecture of the Java servlet and JSP. Since Java servlets run at server-side, they can connect to the database through JDBC and utilize Java Bean to extend functionalities. Java Bean is an object technology developed by Sun Microsystems Inc.
Figure 2-4. The architecture of Java Servlet and JSP
Java Server Pages (JSP) are similar to ASP. While an ASP file contains HTML statements and scripting codes, a JSP file contains HTML statement and Java code. When a JSP file is requested for the first time, Java codes in the JSP file will be automatically compiled and a corresponding servlet will be generated. Then the Web server will execute the servlet and return the result to the Web browser. JSP extends the functions of the servlet by providing a powerful and dynamic page assembly mechanism.

Compared the with other server-side technologies, Java servlet and JSP has the following advantages:

- **Platform independence**

  Since Java servlets are implemented using the Java language, Java servlets are platform-independent. While ASP is only supported by Windows platforms, Java servlet can move around in Unix, Windows NT, Windows and Linux platforms.

- **Efficiency**

  Compared with CGI, Java servlets provide a much more efficient method of handling user requests by using multi-threads model.

- **Modularity**

  Modularity makes a large Web application easier to develop and maintain. Java servlet, JSP and Java Bean provide a way to modularize the application. By using
Java servlets, a large Web application can be broken into discrete modules, with each module being responsible for a specific task.

- **Reusability**

 Java is a completely object-oriented language and provides a mechanism to be reused. Java Bean object technology also contributes to the reusability of Java servlets.

 In CyberEye, Java applets are used to build the server-side component for the CyberEye Viewer. The server-side component is responsible for connecting with database and communicating with the client-side component of the CyberEye Viewer.

### 2.4 Internet-based 3D Technologies

In an Internet-enabled collaborative design environment, it is extremely important for team participants to be able to view and manipulate 3D product models on the Internet, inside the Web browser.

This section describes the two key technologies that are applied in CyberEye to achieve the 3D display and manipulation on the Internet: VRML and Java 3D.
2.4.1 VRML

VRML is an acronym for Virtual Reality Modeling Language. It is the international standard file format for describing interactive 3D multimedia on the Internet. Although it is called a language, VRML is not a general purpose programming language like C++, a script language like JavaScript or a page specification language like HTML. It is a scene description language that describes the geometry and behavior of a 3D scene.

VRML has the following features:

- **VRML is an open standard**

  Since its introduction in 1995, VRML has become the most widely accepted 3D Web graphics standard. VRML was recognized as an international standard (ISO/IEC-14772-1: 1997) by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) in December 1997.

- **VRML has broad industry support**

  Long before its official standardization, VRML became the de facto standard for sharing and publishing data between CAD, animation, and 3D modeling programs. Virtually every one of those programs now exports VRML or has a utility or plug-in to convert its native file format to VRML. For example, both I-DEAS and Pro/Engineering can export VRML files. VRML is included or referenced in the upcoming standard, such as Java 3D, and in other developing standards.
Chapter 2 Enabling Technologies and Tools

- **VRML is Internet-enabled**

  Unlike previous 3D applications, using the Internet to share 3D objects and scenes was built into VRML from the very beginning. VRML is the best candidate for Web 3D at this time.

- **VRML is an efficient format**

  Because the size of VRML files is much smaller than CAD model files in the native format of I-DEAS or Pro/E, it takes less time to transmit them on the Internet.

  Since VRML was introduced by Silicon Graphics in 1994, there have been three versions: VRML1.0, VRML2.0 and VRML97. VRML1.0 allows only a static 3D world. With VRML2.0, users can move and interact with 3D models. VRML97 is almost the same as VRML2.0 except that VRML 97 is an international specification approved by ISO.

  VRML files can be created in a text editor, CAD systems, modeling and animation packages and VRML authoring software. Most CAD systems have the capability to generate VRML files from native formats files. For example, current version of I-DEAS can export VRML2.0 files and the current version of Pro/Engineer can generate VRML1.0 files.

  The limitation of VRML is that it is only an approximation of the underlying geometric and topological data. It only supports polygons and several primitives, such
as cube, cylinder and cone. Free-formed surfaces, such as B-spline surfaces, are not supported in the current version of VRML.

To view VRML Web pages, users need a VRML-enabled browser, such as WebSpace from Silicon Graphics, a VRML add-in for Internet Explorer or a VRML plug-in for Netscape Navigator. However, all these VRML-enabled browsers, add-ins or plug-ins are products of third parties; developers cannot customize them to suit their specific needs for a collaborative design environment. This is the reason we developed our own VRML viewer, CyberEye Viewer. CyberEye Viewer is programmed with the Java language. One key technique that makes CyberEye Viewer possible is the Java 3D API (Application Programming Interface).

2.4.2 Java and Java 3D

Java

Java is a next-generation object-oriented programming language. Although similar to C++, Java is smaller, more portable, and easier to use than C++. Java was also designed to be secure and platform-neutral. This makes it a critical tool for distributed Internet programming. It enables the creation of novel platform-independent software tools, systems, and agents.

Previously Java was not suitable for the development of full-scale CAD/CAE applications because there were no native Java libraries or APIs for supporting solid and
Chapter 2 Enabling Technologies and Tools

dependent geometric modeling. However, with the introduction of Java 3D API by Sun Microsystems Inc. in December 1998, Java has overcome this shortcoming and is now the best candidate to program Internet-enabled CAD/CAE applications.

In CyberEye, both the client component (Java applet) and the server component (Java servlet) of CyberEye Viewer are programmed with the Java language.

Java 3D

Graphics APIs have evolved in step with hardware and customer requirements. In the early 1980s, graphics applications were developed in FORTRAN. Developers were limited to primitives (lines, polylines, polygons, etc.) and wire frame surfaces. In the 1990s, C++ programming language and OpenGL became the standard for 3D graphics development. Developers were able to create more powerful 3D application that can implement texture mapping, rendering and lighting.

Now we are entering the Internet age where collaboration is the key. Information, including 3D models, is shared across the Internet. While Java is becoming the most popular programming language because it is platform independent, Java 3D API combines the advantages of the Java language and OpenGL, making the development of Internet-based 3D software very easy.

Java 3D is a network-centric, scene graph based API, that revolutionizes 3D graphics application development. It is a standard extension to the Java2 JDK. The API
Chapter 2 Enabling Technologies and Tools

provides a collection of high-level constructs for creating and manipulating 3D geometry and structures.

The benefits of Java 3D include:

- **Network-centric**

  Java is the language of the network and Java 3D extends that paradigm to 3D graphics, enabling collaboration in 3D across the network.

- **Write once, run anywhere portability**

  As a part of the Java programming language, Java 3D is platform-independent. It scales on various platforms, from a low-end laptop to high-end graphics workstation, taking advantage of the capabilities at each level.

- **Scene graph based**

  Java 3D is a scene graph based API. This high-level graph environment removes much of the complexity of programming in 3D. Java 3D's scene graph architecture greatly simplifies the writing of 3D programs. Developers no longer have to be a graphics expert to write effective 3D applications and applets.

  A Java 3D virtual universe is created from a scene graph. The scene graph is assembled from objects to define the geometry, sound lights, location, orientation, and appearance of visual and audio objects. A graph is a data structure composed of nodes and arcs. A node is a data element and arc is a relation between data elements. Figure
Chapter 2 Enabling Technologies and Tools

2-5 shows a sample scene graph that is used to construct a cone-cylinder object. Each scene graph has a single Virtual Universe object. A Locale object provides a reference point in the virtual universe. A Branch Group object is the root of a subgraph. There are two different categories of scene subgraph: the view branch graph and the content branch graph. The content branch graph is used to specify the contents of the virtual universe. The view branch graph specifies the viewing parameters such as the view location and direction. Transform Group is used to specify the location and orientation of the visual object in the virtual universe. Shape3D node, the Appearance and the Geometry Node Components together specify the appearance and geometry of the visual object.

Although Java 3D and VRML are both used for creating Web 3D graphics, they are fundamentally different approaches that may be compatible. VRML is actually a 3D file format that describes interactive 3D worlds on the Internet. Java3D is a 3D graphics API for drawing 3D graphics using the Java language. VRML is easier to implement because no programming is needed (the VRML viewers are written by third parties). However, in some cases VRML's limitations are difficult or impossible to overcome. For example, it is impossible to use VRML to develop a solid modeler that can be used in engineering.

Java 3D can be used to write a VRML browser as was accomplished in this thesis. VRML could become the file format for Java3D. Furthermore, because Java3D is an
API of the Java language, Java3D can be used for those tasks that go beyond VRML's limits. It is possible to use Java and Java3D to write an Internet-enabled 3D solid modeler that allows users to not only display and manipulate 3D model, but also create new models or modify existing models.

![Diagram of a typical scene graph in Java 3D]

Figure 2-5. A Typical Scene Graph in Java 3D
Chapter 2 Enabling Technologies and Tools

2.5 Supporting Tools

This section describes various tools that are necessary for the development of the Internet-enabled collaborative design environment.

2.5.1 Web Servers

The Web Server (or HTTP Server) is a program running on the server that responds to incoming requests and serves up HTML documents and the associated files and scripts. The Web server is the key component in any Web application, including the Internet-enabled collaborative design environment. It is the Web server that connects other parts of the Web application together. Choosing the correct Web server is crucial to the success of a Web application.

According to the Netcraft Web Server Survey (http://www.netcraft.com/survey), the leading web servers, based on market share, are the Apache Web server and the Microsoft’s Internet Information Server (IIS) (Table 2-1).
Apache Web Server

The Apache Web server is the most popular Web server. It can be downloaded from the Apache web site (http://www.apache.org). Unlike IIS, which runs only on the Windows NT server operating system, the Apache web server supports multiple platforms, including Windows, Unix and Linux. The Apache web server supports CGI internally. After installing JServ (an additional component), The Apache web server supports Java Servlet and JSP. However, the Apache web server does not support ASP.

IIS/PWS

IIS is the leading web server developed by Microsoft. IIS4.0 can only run on the Windows NT Server 4.0 operating system. Personal Web Server (PWS) is simplified version of the IIS, which runs on Windows 95/98 or Windows NT Workstation 4.0 operating system. Although IIS and PWS both support Active Server Pages 2.0, IIS has more features, is more secure and is tightly integrated with the operating system. PWS

<table>
<thead>
<tr>
<th>Server</th>
<th>Market Share Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>58%</td>
</tr>
<tr>
<td>Microsoft - IIS</td>
<td>22%</td>
</tr>
<tr>
<td>Netscape - Enterprise</td>
<td>7%</td>
</tr>
<tr>
<td>Others</td>
<td>13%</td>
</tr>
</tbody>
</table>

Table 2-1. The Market Share for Top Web Servers, November 1999
is usually used for study purposes. The next version of IIS, Internet Information Services 5.0, will be integrated into the next generation of Windows operation system (Window 2000 Server).

IIS has the following features:

- **Internal support for ASP**

IIS is the only Web server that fully supports all functions of ASP. The current version of the IIS (4.0) supports ASP2.0. The new version (5.0) is included in Windows 2000 Server, which supports ASP3.0.

- **Integration with Windows NT server and other server products**

IIS provides the highest level of integration with the Windows NT Server operating system and many Microsoft server products such as Microsoft Transaction Server (MTS) and SQL Server.

- **Easy data access**

IIS uses ActiveX Data Objects (ADO) to provide fast, productive and easy-to-use programmatic interface to a variety of data sources including Access, SQL Server and Oracle Database.

- **Support for COM**
Chapter 2 Enabling Technologies and Tools

IIS supports COM, one of the most popular component-based technologies. COM can greatly expand the capabilities of the Web application and makes distributed computing possible.

- **Strong security**

  Security is one of the most important issues for Web applications. Because IIS is tightly integrated with the Windows NT security system, IIS is more secure than many other Web servers that are separated from the operating system.

  One disadvantage of IIS is that it only runs on the Windows NT Server operation system. However, for Web applications, there are no requirements for the operating system on the computers of the clients. Users can access CyberEye on any platform by using a Web browser with some plug-ins (Java3D plug-in and ActiveX plug-in). Another disadvantage is that IIS does not support Java Servlets internally. There are two ways to solve this problem:

1. **Installation of a Java servlet-enabled component**

   Currently, several software companies provide Java Servlet components, such as JRun. A trial version of the JRun can be downloaded from the Web site (http://www.jrun.com)

2. **Adoption of a second Web server that supports Java servlet**
Several Web servers support Java Servlet internally, including Java Web Server, JSWDK (JavaServer Web Development Kit) Web Server and the Apache Web server with JServ. JWSDK Web Server is included in the JWSDL, which can be downloaded from the Web site of Sun Microsystems. JServ is a add-on component for the Apache Web server.

In the current implementation of CyberEye, IIS has been chosen as the main Web server with the JWSDK Web Server being the assistant Web server. While, IIS provides support for HTTP and ASP, the JWSDK Web Server enables Java servlet.

### 2.5.2 Web Browser

The Web browser is the main client component in a Web application. Currently, most users use one of the two main Web browsers: Netscape Navigator or Microsoft Internet Explorer. Netscape Navigator was the most popular Web browser but has been surpassed by Microsoft Internet Explorer recently, particularly with its inclusion in Windows98 and Window2000.

For the support of client-side scripts, Netscape Navigator supports JavaScript but does not support VBScript. Microsoft Internet Explorer supports both JavaScript and VBScript. Considering that users of CyberEye may use both Web browsers, CyberEye only adopts JavaScript as client-side scripts.
Both browsers support Java applets. However, the current Java Virtual Machines (JVM) included in the current version of Netscape Navigator (4.5) and Microsoft Internet Explorer (5.01) do not support the Java2 platform. For example, the interface of CyberEye Viewer is created using Java Swing, a new Java API which is only included in the Java2 platform. In order to support the newest version of Java platform, Java plug-in provided by Sun Microsystems Inc. must be installed in both browsers.

Netscape Navigator does not support ActiveX control internally. However, the Chat Room client component in CyberEye is implemented as an ActiveX control. If users want to use Netscape Navigator, they need to install an ActiveX plug-in. For this reason, Microsoft Internet Explorer is recommended for the users of CyberEye.

Development Tools

**InterDev**

Visual InterDev is a comprehensive, Web-based application development tool introduced by Microsoft. It provides an integrated environment that supports the use of HTML/DHTML, VBScript/Jscript, ASP, database connection, ActiveX control and Java applet. Although a simple text editor can also be used to build Web applications, Visual InterDev significantly augments the productivity of Web developers.

The features of Visual InterDev 6.0 are described as followings:

* Integrated development environment
Chapter 2 Enabling Technologies and Tools

Visual InterDev 6 provides an enhanced integrated development environment (IDE) that is similar to Visual C++ or Visual Basic (Figure 2-6). The IDE includes many features such as quick view and statement completion.

- **WYSIWYG editing**

  The *What You See is What You Get* editing capabilities in Visual InterDev 6 allows developers to visually create and edit Web pages just like using Word to write a document.
Chapter 2 Enabling Technologies and Tools

- **Step by step debugging**

  Visual InterDev allows step-by-step debugging for both client-side script and server-side script. This is very helpful when developing a complex application.

- **Support for the latest Web technologies**

  Visual InterDev supports many new Web technologies, including cascading style sheets (CSS) and DHTML, ASP, ActiveX control and Java applet.

  With the exception of CyberEye Viewer, all components of CyberEye are developed under Visual InterDev 6 on a Window NT platform.

**Java 2 SDK**

The Java 2 SDK is a development environment for building applications, applets, and components that can be deployed on the Java platform. The Java 2 SDK software includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform. These tools are designed to be used from the command line. The Java2 SDK can be downloaded from the Web site of Sun Microsystems Inc.

CyberEye Viewer (which consists of Java applet and Java servlet) is developed under the Java2 SDK.
2.6 Summary

In this chapter, the underlying supporting technologies and tools for collaborative design environments were described and analyzed. Amongst these enabling technologies, Java 3D API is crucial to 3D visualization of product models over the Internet. Java servlet is essential for the communication between Java applet and database. ASP plays an important role in the implementation of the proposed collaborative design environment.

Amongst the supporting tools, Visual InterDev and Java2 SDK provide the development environment. IIS and JSWDK Web server provide the running environment.
Chapter 3

Overview of CyberEye

CyberEye is an Internet-enabled, platform-independent collaborative design environment to support the execution of multidisciplinary team engineering on the Internet. CyberEye deals with the two key issues required for the development of collaborative design environment, namely, the 3D display and manipulation, and the collaboration and coordination on the Internet. It provides supports for 3-D visualization and synchronous interaction over the Internet. This chapter provides an overview of CyberEye. The infrastructure and architecture of CyberEye will be described. In addition, the security issues of CyberEye will be discussed at the end of this chapter.

3.1 Infrastructure

In its current implementation, the infrastructure of CyberEye is composed of one core and three implementation modules. The core module is the Information Source and
the three implementation modules are the Product Visualization Module, the Product Data Management Module and the Team Management Module. The relationships amongst the core and implementation modules are depicted in Figure 3-1.

Located in the kernel of the infrastructure, the Information Source is used to store product and team information. It includes two databases (the Product Database and the Team Database) and a File Vault. The Product Database stores product information and provides links to product model files. The Team Database keeps all information of team participants and messages in the Web Forum. The File Vault is a Web-shared directory on the server that contains product model files (VRML files and CAD files). The VRML files are used for the 3D display and manipulation on the Internet; the CAD files are the native format files of commercial CAD systems, such as Pro/Engineering and I-DEAS.

Outside the kernel is the Product Visualization Module, which directly accesses the Product Database and VRML files in the File Vault. The Product Visualization Module is actually an Internet-enabled CAD browser called CyberEye Viewer, which enables 3D display and manipulation of product model amongst the team participants.

The Product Information Management Module controls and maintains the Product Visualization Module. The Product Information Management Module not only allows team members to submit new designs and to modify or remove existing designs, but
also provides facilities to support the two-way file transfer between clients and the server.

The outermost module is the Team Management Module, which maintains a user login mechanism and controls access to the Product Information Management Module. Additionally, it provides communication facilities to support both asynchronous and synchronous distributed interactions amongst team participants. These facilities include the email, Web Forum, Online List, Chat Room and Instant Message Unit.

Users of CyberEye are divided into five types: administrator, team leaders, team members, suppliers and customers. In this thesis, users of these five types are called team participants. Team participants have different access permissions in CyberEye. For example, while suppliers and customers only have one-way access permissions (download) to Information Source, other team participant are allowed two-way access permissions (download and upload).
Figure 3-1. Infrastructure of CyberEye
3.1.1 Product Visualization Module

The Product Visualization Module in CyberEye is also called CyberEye Viewer. CyberEye Viewer is an Internet-enabled, platform-independent, low-cost, open-standard, extensible 3D CAD browser based on Java3D, Java applet and Java servlet technologies.

The features of CyberEye Viewer include:

- **Support for 3D display and manipulation**

  CyberEye Viewer supports 3D realistic viewing of the product model. Team participants can rotate, pan, and zoom the 3D model using either mouse or keyboard.

- **Support for multiple formats**

  Unlike most CAD browsers that only support one or two formats, CyberEye Viewer provides the potential to support multiple formats.

- **Extensibility**

  Because CyberEye Viewer is implemented using the Java language, developers can expand its function.
• **Internet enabled**

Team participants can use CyberEye Viewer to display 3D product models not only on a local computer, but also at any location on the Internet through the URL (Uniform Resource Locator).

• **Platform-independent**

The *Write Once, Run Anywhere* portability of the Java language makes CyberEye Viewer platform-independent. Thus, CyberEye Viewer can run on various platforms.

• **Database connectivity**

CyberEye Viewer can connect to the database through its server component (Java servlet). In this way, product information stored in the database can be displayed in CyberEye Viewer along with the 3D product model.

• **Low cost**

The client component (Java applet) of CyberEye Viewer is downloaded automatically from the Web server each time when team participants visit the Web page containing CyberEye Viewer, so users do not need install CyberEye Viewer on their computer. The only software required at the client-side is a Web browser.
3.1.2 Product Information Management Module

The Product Information Management Module manages the proper organization of the design data for easy use by geographically distributed team participants and the efficient transfer of design data between clients and server. It has the following features:

- **Central product information storage**

  In CyberEye, collaboration amongst geographically distributed team participants can be achieved through shared VRML format product models, CAD models and associated product information. All VRML files and CAD files are stored on the File Vault. Product information is saved in the Product Database. Because all team participants share the same product models and product information, they can access the changes made to product models immediately. This avoids the ambiguity and the delay problem experienced with traditional broadcasted product data management.

- **User-friendly Web-based interface**

  A simple and easy-to-learn Web-based user interface provides access to all digital product information. Team participants can perform various operations in the Web browser, such as viewing design information, submitting new design or updating existing designs.
• **Active two-way data management**

The product data management system of CyberEye supports active data management as compared to the passive *released data* management capabilities offered by other systems. This means that team members have the right to submit a new product design by uploading a product model to the server and also adding the product information to the database in the Web browser through the Internet. The product model file is copied to the server using a Web-based upload method. This method is better than the FTP method in that the Web-based method is simpler and the file name and the path of the product model will automatically stored in the Product Database.

### 3.1.3 Team Management Module

Collaborative design involves asynchronous and synchronous interactions amongst geographically distributed design team members as well as suppliers and customers. The effective collaboration and coordination amongst these team members and other participants are critical to a successful design.

To achieve better collaboration and coordination in the collaborative design environment, an ASP-based Team Management Module is integrated into CyberEye. The Team Management Module not only manages the activities of team participants,
but also supports both asynchronous and synchronous distributed interactions amongst them.

The team management and coordination in the Team Management Module are achieved by integrating a user login mechanism, and synchronous and asynchronous communication tools.

The module consists of the following components:

- **Secure user login mechanism**

  The Team Management Module includes a secure login mechanism that is implemented with ASP and a SQL Server database. All team participants are required to provide a login name and password if they want to access CyberEye. This is to prevent unexpected intrusion from outside. The Team Management Module also defines various access levels for different user types to control and coordinate the degree of access to the shared information.

- **Asynchronous distributed communication facilities**

  Asynchronous distributed communication facilities include an Automatic Email Notification Unit, a Mailing List Generating System and a Web Forum. The Automatic Email Notification System can automatically generate and send email to respond to various activities, such as registering or submitting a new product design. Depending on the types of activities, the notification emails may be sent to all team participants or only related participants. The Mailing List Generating Unit is a Web-
based facility that helps team participants to create mailing lists dynamically according to their requirements.

The Web Forum in CyberEye is mainly used for the asynchronous discussion between team participants. Because the Web Forum is Web-based, it is preferable to the use of mailing lists or newsgroup since team participants do not want to receive irrelevant email.

- **Synchronous distributed communication facilities**

The Online List, the Chat Room and the Instant Message Unit make synchronous distributed interactions amongst team participants possible. Online List is a table on the Web pages that contains information of team participants currently logged on. The Chat Room is dedicated to the support of real-time communications among the team participants. Unlike the Web Forum, the Chat Room allows participants to interact on a personal, real-time basis. The Instant Message Unit is a Winsock-based real-time communication tool that runs separately from the Web browser. The Instant Message Unit not only allows team members to send an instant message to each other, but also allows the administrator to monitor and coordinate the online activities in real time.
3.2 Architecture

CyberEye delivers a scalable, easily implemented and technically advanced architecture. With a Web-centric, ASP and Java-based architecture, CyberEye provides authorized participants with access to most features of the system through a Web browser. The architecture of CyberEye leverages current standards and leading-edge technologies such as HTML, ASP, Java and ODBC to build a platform for design collaboration.

Figure 3-2 (Previous page) illustrates the architecture of CyberEye. From the network point of view, the architecture can be divided into client-side frame and server-side frame.

3.2.1 Client-Side Frame

Client-side frame forms the front-end of CyberEye. It consists of two software components: the Web browser and the Instant Message Client.

- Web browser

The Web browser can be one of the two most popular Web browsers: Microsoft Internet Explorer or Netscape Navigator. Inside the Web browser, three client-side
technologies are used to build up the interface, namely, HTML/DHTML, Java Applet and ActiveX control.

HTML/DHTML is used to present text and graphics in the Web browser. In addition, HTML forms are used to send information from clients to the Web server. In CyberEye, team participants use HTML forms to enter their name/password, post messages, send instant messages, and input product information.

Java applets provide the possibility to create an interactive graphic user interface in the Web browser. In CyberEye, a Java applet is used to implement the client component of CyberEye Viewer. Through the applet, team participants can manipulate realistic 3D product model using operations such as rotating, panning and zooming. The applet also lists all product models available in the database and displays the product information of the current model.

ActiveX control is another way to add the graphic user interface in the Web browser. In CyberEye, ActiveX control is used to implement the client component of the Chat Room.

- **Instant Message Client**

Instant Message Client is stand-alone software that runs outside of the Web browser. As the client component of the Instant Message Unit, it is used to receive instant messages automatically generated by CyberEye or sent by other team participants.
3.2.2 Server-Side Frame


- **Web Server**

  The Web server is the key component of CyberEye. It connects other components together. The WWW service in the Web server is responsible for receiving requests and sending responses. ASP service is used to enable Active Server Pages technology.

- **Java Virtual Machine**

  Java Virtual Machine (JVM) is necessary for running Java servlet. The Java servlet in CyberEye serves as the linkage between the client-side Java applet and the Product Database.

- **Chat Server**

  Chat Server is the server component of Chat Room. Chat Room Server and Client together provide the facility to enable real-time communication amongst team participants.
Chapter 3 Overview of CyberEye

- **Instant Message Server**

  Instant Message Server is implemented as a COM component. It is responsible for sending instant messages to the Instant Message Client. The instant messages can be either sent by team participants or automatically generated by CyberEye responding to various user activities (such as login or posting messages).

- **Database Servers**

  The Database servers are commercial available database management systems. Because CyberEye adopts the ODBC interface to connect to databases, it supports most database management systems such as Microsoft Access, Microsoft SQLServer or Oracle. CyberEye stores, tracks and manages product information and team information in two logical databases, the Product Database and the Team Database. The Product Database is implemented as a Microsoft Access database and the Team Database is an SQLServer database.

- **File Vault**

  The File Vault is a Web-shared physical directory on the Web server. The main task of the File Vault is used to transfer VRML files and CAD files between the client-side and the server-side. The transfer is two-way. Team members can upload their own VRML files and CAD files to the server or they can download files from the server. Furthermore, the File Vault is also used to upload the photographs of team members to the server.
3.3 Security

Security plays an extremely important role to guarantee the practicality of any Internet-enabled CAD system. CyberEye provides a strong level of security by implementing ASP technology on the Microsoft Windows NT platform.

As one of the securest operating systems, Windows NT meets and exceeds certifiable security standards (C2 security guidelines, which is required by the U.S. Department of Defense). IIS is built on the security model of Windows NT. This means that CyberEye, which adopts IIS as its main Web Server, is protected by high level of security of Windows NT.

All web pages of CyberEye are protected with two levels of security. The first level is the file access controls, IP access security controls and user authentication provided by the Windows NT security model. The second level is the user login mechanism implemented with the Team Database and ASP technology. The user login mechanism uses session variables to identify each user and prevents unauthorized visit. In addition, CyberEye maintains a log of all visits that can be accessed by an administrator using Web browser (see Figure 3-3).
Chapter 3 Overview of CyberEye

3.4 Summary

CyberEye is an Internet-enabled, platform-independent collaborative design environment. The infrastructure of CyberEye consists of the Information Source, the Product Visualization Module, the Product Data Management Module and the Team Management Module. The Team Management Module is the foundation of the infrastructure. From the analysis of the architecture, we can see that many Internet
technologies are applied in CyberEye, including ASP, Java servlet and Java3D. The security of CyberEye is strong due to the Window NT platform and the secure user login mechanism.
Chapter 4

Product Visualization Module

One of the key issues for distributed collaborative design is how 3D product models can be viewed and manipulated by geographically dispersed team participants that are using heterogeneous platforms. To address this issue, a novel approach is proposed in this thesis. In this approach, a Java applet running in the Web browser is used as the front-end with VRML being used as the data format for presentation purposes. Additionally, one of the newest 3D technologies from Sun Microsystems Inc., Java 3D, is used to achieve the 3D display and manipulation. A Java servlet running on the server is used to connect to the database through JDBC. The Java applet and Java servlet communicate with each other to exchange product information. Based on this approach, a Java 3D CAD viewer referred to as CyberEye Viewer has been developed. CyberEye Viewer is the Product Visualization Module in the infrastructure of CyberEye.
In this chapter, the architecture and implementation details of the CyberEye Viewer will be first presented. Thereafter, the interface and operation will be described, followed by some discussion of the data input of the CyberEye Viewer.

4.1 Architecture

The Internet-based client-server architecture of CyberEye Viewer is illustrated in Figure 4-1. The architecture of CyberEye Viewer can be split up into three tiers: the user interface (Web Browser and Java applet), the business logic (Java servlet), and the data (Database).

The first tier of the architecture consists of a Web browser (either Microsoft Internet Explorer or Netscape Navigator) and a Java applet. HTML is convenient and simple way to create the text-based interface in the Web browser. However, because CyberEye Viewer requires an interactive graphic interface that can display both the 3D model and product information, HTML alone is not sufficient. A Java applet is used in the CyberEye Viewer to build the user interface. The interface of CyberEye is composed of a list that displays all the names of available product models in the Product Database, a 3D canvas window displaying the 3D product model and information panels that display associated product information (Figure 4-2).
Chapter 4 Product Visualization Module

Figure 4-1. Architecture of CyberEye Viewer
The second tier consists of a Java servlet and a Web server. The Java servlet serves as the linkage between the applet and the database. It accepts the request from the applet and queries the database, then sends the query result back to the applet. The Web server can be any one that supports Java servlets. The current implementation uses the JSWDK Web server developed by Sun Microsystems Inc.

The Product Database and VRML Files belong to the third tier of the architecture. The Product Database is a Microsoft Access database. The servlet connects with the Product Database through JDBC. VRML files are stored on the server and can be
directly accessed by the Java applet through URLs. URLs to the VRML files are stored in the Product Database.

4.2 Implementation Details

In this section, we will describe implementation details of CyberEye Viewer, which include the data format, data storage, 3D display and communication.

4.2.1 Data Format

One of the basic requirements for distributed 3D applications is to find a format for the exchange of geometric information. In CyberEye Viewer, VRML (Virtual Reality Modeling Language) is used as the default format.

VRML is a scene description language for creating 3D interactive Web graphics over the Internet. Unlike GIF animation images, which do not support interaction, VRML allows the user to move around within a graphic image and interact with objects inside the Web browser. VRML is one of the best data format options available for low cost, efficient interactive 3D applications.

However, one important limitation of VRML is that it is only an approximated 3D representation of the geometric model. VRML would work well if team participants only needed to view product models, but it is impossible for participants modify designs
Chapter 4 Product Visualization Module

based on VRML files. Therefore, in CyberEye, the CAD files generated by commercial CAD system (such as I-DEAS and Pro/E) are also stored on the server. If team members want to modify the product model, they can download the CAD files to their computer and use I-DEAS or Pro/Engineer to modify the product model.

4.2.2 Data Storage

Product information associated with product models is stored on the server in a Microsoft Access database called the Product Database. VRML files and CAD files are stored in the File Vault on the server that can be accessed by URL. The Model Table in the Product Database is used to store the URLs to these files. The VRMLURL fields contain the URL to VRML files and the ModelURL fields contain the URLs to Model files. CyberEye viewer can load the VRML files directly through their URLs.

4.2.3 3D Display and Manipulation

CyberEye Viewer is based on Java3D API to achieve 3D display and manipulation of product models. As a standard extension to the Java 2 JDK, Java 3D API is an interface for writing programs to display and interact with three-dimensional objects. It provides a collection of high-level constructs for creating and manipulating 3D geometry and structures for rendering that geometry. Java 3D provides the functions for creation of imagery, visualizations, animations, and interactive 3D graphics application.
programs. In Java3D API, an object called 3D Canvas is a visual component that displays 3D objects (Figure 4-3).

4.2.4 Communication

A Java applet is ideal for building a graphic user interface. However, because the applet runs at the client-side, it is difficult to perform database interactions. The appropriate place to handle this is at the server. The best candidate for this task is...
Chamter 4 Product Visualization Module

Servlets. A servlet is a Java program that runs at the server and can handle the interaction with various databases through JDBC.

The final challenge is how the applet, which runs on the client-side, is able to communicate with a servlet running on a Web server. In CyberEye Viewer, a HTTP based method is used to achieve the communication between the applet and the servlet. Figure 4-4 outlines the procedures of the applet/servlet communication.

First, the applet creates an URLConnection object and opens the connection to the URL of the servlet. Then the URLConnection object creates a DataOutputstream object, which sends an HTTP request to the server, invoking the servlet. The content of the HTTP request contains a SQL query. Once the servlet is initialized, the servlet creates a Connection object and connects the Connection object to the database through JdbcOdbcDriver. The servlet then uses a HttpServletrequest object to get the HTTP request sent by the applet and passes the SQL query in the HTTP request to the Connection object. The Connection object uses this SQL query to query the database and return the query result to a ResultSet object. The content in the ResultSet object is then sent back to the applet by HttpServletResponse object.
Figure 4-4. Communication between Java applet and servlet
Chapter 4 Product Visualization Module

Finally, the applet uses a DataInputStream object created by the URLConnection object to receive the formatted results from the servlet. Other tasks include loading the product model according to the URL in the result and displaying product information in the result.

4.3 Interface and Operation

4.3.1 User Interface Description

The interface of CyberEye Viewer is created using Java Swing API (Figure 4-5). Compared with AWT (Java Abstract Windowing Toolkit) components, Swing components have more functionality, and let Java programs keep the same look and feel on different platforms, such as Windows, Unix and Macintosh. The interface consists of the following components: 3D Display Window, 3D Manipulation Toolbar, URL Location Window, Product Model List and Product Information Panel.

3D Display Window

The 3D display window is the central component of CyberEye Viewer. It is used to display an interactive, realistic 3D model. Team participants can use the mouse and keyboard to manipulate the 3D model conveniently. The manipulation operations include rotation, panning and zoom functions.
Chapter 4 Product Visualization Module

Figure 4-5. User Interface of CyberEye Viewer
3D Manipulation Toolbar

The 3D manipulation toolbar contains buttons for manipulating the 3D product models (see Figure 4-6). While the default position of the toolbar is located at the top of the Viewer, it can be placed anywhere by dragging the pull handle of the toolbar.

The functions of buttons in the toolbar are described below:

- **X Rotate CCW Button**
  Rotates the model along the X-axis in a counterclockwise direction.

- **X Rotate CW Button**
  Rotates the model along the X-axis in a clockwise direction.

- **Y Rotate CCW Button**
  Rotates the model along the Y-axis in a counterclockwise direction.

- **Y Rotate CW Button**
  Rotates the model along the Y-axis in a clockwise direction.
Chapter 4 Product Visualization Module

- **Z Rotate CCW Button**
  Rotates the model along the Z-axis in a counterclockwise direction.

- **Z Rotate CW Button**
  Rotates the model along the Z-axis in a clockwise direction.

- **Front View Button**
  Displays the front view of the model.

- **Rear View Button**
  Displays the rear view of the model.

- **Left View Button**
  Displays the left view of the model.

- **Right View Button**
  Displays the right view of the model.

- **Top View Button**
  Displays the top view of the model.
Bottom View Button

Displays the bottom view of the model.

Zoom In Button

Zooms in on the view of the model.

Zoom Out Button

Zooms out from the view of the model.

Product Model List

The Product Model List displays the names of all product models stored in the Product Database. To load a product model into the viewer, team participants need to select the name of the product model in the Product Model List and then click on the "Display" button. The applet will send a message containing the name of the product model to the servlet. The servlet uses this name to query the Product Database through JDBC. The results of the query include the URL to the VRML file and associated product information. After the servlet sends the query result back to the applet, the applet will load the VRML file according to the URL and displays product information in the applet.

The "Refresh" button is used to reload the product names in the Product Model List. When other team members add new product models or delete product models, clicking
the "Refresh" button will send a message to servlet to re-query all product names in the Product Database. Therefore the Product Model List always reflects the most recent changes.

**URL Location Window**

In CyberEye Viewer, team participants can display 3D product models in two ways. The first way is to click on the name of the model in the Product Model List. The second way is to specify the URL to the VRML file directly in the URL Location Window. The latter approach allows team participants to view VRML files that are not stored in the Product Database. As long as VRML files can be accessed using URLs, they can be displayed in CyberEye Viewer.

**Product Information Panel**

The Product Information Panel displays product information associated with the current product model. There are two Product Information Panels. On the left side is the General Information Panel that displays general information about the product model, including product code, designer name, version, date created and last modified date. On the right side is the Engineering Information Panel that displays engineering information, such as material, manufacture, assembly and specification. All product information is queried from the Product Database by the servlet and is transmitted from the servlet to the applet.
Chapter 4 Product Visualization Module

Status Window

The status window is a text area that displays system information, help information or status information. For example, when a team participant clicks one of the buttons in the 3D manipulation toolbar, the status window will display information pertaining the operation.

4.3.2 User Activities Description

In CyberEye Viewer, team participants can do the following operations:

Start CyberEye Viewer

After team participants have logged in, they can start the CyberEye Viewer by choosing the “Viewer” button on the top menu and then clicking the logo of CyberEye. The initialization of CyberEye Viewer will take about twenty seconds. Once CyberEye Viewer is started, it will automatically query the Product Database and display the names of all product models in the Product Model List.

Display Product Model

To display a product model, team participants need to select the name of that product model from the Product Model List and click on the “Display” button. The 3D model of the product will be loaded to the 3D Display Window. At the same time, the product information will also be displayed in the panels on both sides of the 3D Display Window.
Manipulate Product Model

In addition to the 3D Manipulation Toolbar, a mouse can be used to manipulate the product model in the 3D Display Window. Team participants can use the mouse to perform three types of manipulations: rotate, pan, and zoom.

- **Rotate**

  By pressing down the left button of the mouse and dragging the model, team participants will rotate the model in any direction and angle (see Figure 4-7).

---

**Figure 4-7. Rotate Manipulation**
- **Pan**

  Pressing down the right button of the mouse and dragging the model will move the model to any desired position (see Figure 4-8).

![Figure 4-8. Pan Manipulation](image)

![Figure 4-9. Zoom Manipulation](image)
• **Zoom**

  Holding down the Alt key and pushing down the left button of the mouse, then moving the mouse will zoom in or zoom out the model (see Figure 4-9).

### 4.4 Data Input

The current implementation of CyberEye Viewer adopts the VRML Loader developed by the Java-3D and VRML working group to load the VRML file to the viewer. The current implementation of this VRML Loader does not support all specifications of VRML97 standard. The VRML files generated by I-DEAS comply with VRML2.0 standard. Pro/Engineering generates VRML files that comply with the VRML1.0 standard. When the VRML Loader tries to read the VRML files generated by both CAD systems, some errors will occur. Thus a VRML converter is needed to convert the VRML files generated by I-DEAS or Pro/Engineering to the VRML files that can be read by the VRML Loader, as illustrated in Figure 4-10.
4.4.1 I-DEAS

The VRML files generated by I-DEAS can be displayed in the Cosmo player without any problem. However, these VRML files cannot be loaded into CyberEye Viewer. The problem is that the VRML files contain proto nodes that are not supported by the VRML Loader.

There are two basic data structures in VRML: the node and the field. Nodes are types of object. Nodes contain fields, which are properties of the object. A node is the
basic stand-alone structure in a VRML file. VRML contains 54 different nodes. The Proto node is used to create a customized node that can be treated like a black box in the scene. Once declared, a prototype can be used like any of the standard VRML nodes. The proto nodes in the VRML files generated by the I-DEAS are used to perform special functions, which are useless in CyberEye Viewer. The approach to solve this problem is to develop a VRML Converter that can remove these proto nodes in the VRML files or a new VRML Loader which can ignore these proto nodes.

4.4.2 Pro/Engineer

The VRML files generated by Pro/Engineer comply with VRML 1.0 standard. The VRML Loader cannot load VRML 1.0 files directly. A VRML Converter is needed to upgrade the VRML files from VRML 1.0 standard to VRML 97 standard.

The tasks of developing a VRML Converter and new VRML Loader are ongoing, and will be written in the Java language to permit ease of integration within the CyberEye Viewer.

4.5 Summary

In this chapter, we described the Product Visualization Module (CyberEye Viewer) in detail. CyberEye Viewer plays an important role in the proposed collaborative design
environment because it enables the 3-D display and manipulation of product models on the Internet. Java3D API is the key technology that makes 3-D display and manipulation possible. Through the communication between Java applet and servlet, CyberEye Viewer can connect to the Product Database on the server and display product information. Since all components of CyberEye Viewer are written with the Java language, it has the advantages of being platform-independent, internet-enabled and low cost.
Chapter 5

Product Information Management Module

As team members, suppliers and customers become increasingly interlinked in all phases of a product life cycle, a shared product information model is crucial to interoperability. The Product Information Management Module in CyberEye is used for the management of shared product data, including both the geometric models and associated product information. The system also supports the two-way transmission (upload and download) of product files between the server and clients. In this chapter, the architecture of the Product Information Management Module is presented. Thereafter, the structure of the Product Database is analyzed, followed by a description of the user activities in the Product Information Management Module.
5.1 Architecture

The architecture of the Product Information Management Module is illustrated in Figure 5-1. The architecture consists of the Web server, the Upload component, the Product Database and the File Vault.

The Product Database and the File Vault play important roles in the Product Information Management Module. In CyberEye, collaboration amongst geographically distributed team participants is achieved through shared product model files and associated product information. Product models include VRML files and CAD files. The VRML files are used for the 3D display and manipulation on the Internet; the CAD files are the native format files of commercial CAD systems, such as Pro/Engineering and I-DEAS. All VRML files and CAD files are stored in the File Vault. Product information is saved in the Product Database. Because all team participants share the same product models and product information, they can immediately access the changes made to the product design. This avoids the ambiguity and the delay problem in encountered in traditional broadcasted product data management.
Chapter 5 Product Information Management Module

Figure 5-1. The architecture of the Product Information Management Module
Chapter 5 Product Information Management Module

The Model Table in The Product Database stores information pertinent to the product design. Product model files (including VRML files and CAD files) are stored separately in the File Vault on the server. In order to associate the product models with the product information, two fields in the Model Table (VRMLURL and ModelURL) maintain the URL to these model files.

The Product Information Management Module provides a user-friendly Web-based interface to interact with the Product Database and File Vault. All operations on the Product Database, such as querying, updating, inserting and deleting, are automatically taken care of by the ASP code. Through the Web browser, team members can perform various tasks such as submitting new product designs, updating and deleting existing designs.

The Product Information Management Module also provides the facility to enable team participants to download and upload VRML files and CAD files conveniently through a Web browser. The download function is implemented by simply creating a hyperlink to the file that is to be downloaded. When a hyperlink to this file is clicked, the Web browser will open a dialog to allow the team participant to choose a path to save the file.

The upload function is much more difficult to implement, however. In CyberEye, a COM-based approach is adopted for the upload function. In this approach, the ASP code uses a COM component called *Upload component* to achieve the upload function.
This approach is preferred to the conventional FTP approach for the following three reasons.

- The approach is easy to use. Team participants do not need to create the transfer connection; they simply select the file and click a button.
- The approach not only transfers the file, but also automatically saves the URL to the file in the Product Database, and
- Team participants do not need to install additional software. Unlike the FTP approach, which requires FTP client software, a Web browser is the only software required in the COM-based approach.

In the Product Information Management Module, different team participants have different permissions. Suppliers and customers are only allowed to view the 3D product model and product information. Other team participants can add, modify or delete product information and download or upload product model files as required.

### 5.2 Database Structure

In the current implementation, the Product Database is created using Microsoft Access. There is only one table in the Product Database: the Model Table. Fields in the Model Table and their descriptions are depicted in Table 5-1. While the ModelID field is used by the Product Information Management Module to identify each product
design, team participants use the ProductCode field to distinguish product designs. The VRMLURL and ModelURL fields connect model files with associated product information in the Product Database.

<table>
<thead>
<tr>
<th>Fields Name</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ModelID</td>
<td>Yes</td>
<td>Uniquely identify each product design in database</td>
</tr>
<tr>
<td>ProductCode</td>
<td>Yes</td>
<td>Used to identify product design in design team</td>
</tr>
<tr>
<td>ProductName</td>
<td>Yes</td>
<td>The name of the product</td>
</tr>
<tr>
<td>VRMLURL</td>
<td>Yes</td>
<td>URL to the VRML file of the product model</td>
</tr>
<tr>
<td>ModelURL</td>
<td>No</td>
<td>URL to the native file of the product model</td>
</tr>
<tr>
<td>Designer</td>
<td>Yes</td>
<td>The login name of the designer</td>
</tr>
<tr>
<td>CreateDate</td>
<td>No</td>
<td>The date and time when the product model was created</td>
</tr>
<tr>
<td>LastModifyDate</td>
<td>No</td>
<td>The date and time when the product model was last modified</td>
</tr>
<tr>
<td>Version</td>
<td>No</td>
<td>The version of the product model</td>
</tr>
<tr>
<td>ModelTool</td>
<td>No</td>
<td>The name of the software that is used to create the product model (Such as I-DEAS, Pro/E)</td>
</tr>
<tr>
<td>DataFormat</td>
<td>No</td>
<td>The native format for the product model</td>
</tr>
<tr>
<td>Description</td>
<td>No</td>
<td>General description of the product model</td>
</tr>
<tr>
<td>Material</td>
<td>No</td>
<td>Material of the product</td>
</tr>
<tr>
<td>Manufacture</td>
<td>No</td>
<td>Information about the manufacture methods of the product model</td>
</tr>
<tr>
<td>Assembly</td>
<td>No</td>
<td>Assembly information of the product model</td>
</tr>
</tbody>
</table>

**Table 5-1. Fields in the Model Table**

The Model Table opened in Microsoft Access 97 is shown in Figure 5-2. From this figure, it can be seen that the data type for the ModelID field is *AutoNumber*, which is
automatically increased by one for each new record. The data type for the Description, Specification and Assembly fields is *Memo* since it allows for lengthy text.
5.3 Support of User Activities

In the Product Information Management Module, team participants are allowed to perform the following activities through the Web browser:

Submit New Product Design

The administrator, team leaders and team members have the right to submit a new design. After a team participant clicks the "Add New" button on the left side menu, an HTML form will be displayed to allow the team participant to fill in the product information, see Figure 5-3.

The team participant is required to fill in four text fields: Product Code, Product name, Version and Create Date. TheCreateDate text field will automatically be filled with the current date. The login name of the team participant will be used as the name of the designer. If the model file can already be accessed through a URL, the team participant can select the "Specify a URL" radio button and specify the URL in a text field below the radio button.
If the team participant needs to upload a model file, he/she should select the "Upload to Server" radio button. After submitting this form, the team participant will be required to locate the model file and upload the file to the server, see Figure 5-4. The URL to this file will be automatically saved in the VRMLURL or ModelURL fields in the Product Database once the uploading is completed. At the same time, emails containing information about the new design and instant message will be automatically sent to the administrator, team leaders and team members.
Update Product Design

In the Product Information Management Module, a team participant is only allowed to update an existing design submitted by himself/herself. The administrator and team leaders have the right to update all existing designs. Updating operations include modifying the product information and uploading new versions of the product model files. The administrator, team leaders and team members will be notified of the changes made to the designs, both through automatically generated emails and instant messages. The process of the update is shown in Figure 5-5.
Chapter 5 Product Information Management Module

Figure 5-6. Update product design
Chapter 5 Product Information Management Module

Clicking on “Modify” button on the left side menu will display a list of all product designs in the Product Database. To update a product design, a team participant needs to click on the name of the product design. An HTML form will display the current product information of the product design to allow the team participant to modify. Once the modification is completed, the team participant can click on the “Submit” button. Another HTML form will be displayed to allow the team participant to upload new versions of VRML files and Model files. When a product design is updated, the LastModifyDate field of this design will be automatically set to the current date.

Delete Product Design

Deleting a product design will delete its associated product information stored in the Product Database. While the administrator can delete all product designs, team leaders or team members are only allowed to delete their own product designs.

Search Product Design

The Product Information Management Module provides a powerful search engine to enable team participants to search their desired product design with ease and efficiency, see Figure 5-6. The search engine supports four search methods, namely, search according to the Product Code, Product Name, Designer or Create Date.
In this chapter we described the Product Information Management Module, one of the three modules in CyberEye. The architecture of the module is a Web-based system implemented by ASP technology. The Product Database and the File Vault are two key components in the architecture. The Product Database stores product information. File Vault reserves product model files. The Product Information Management Module
5.4 Summary

In this chapter we described the Product Information Management Module, one of the three modules in CyberEye. The architecture of the module is a Web-based system implemented by ASP technology. The Product Database and the File Vault are two key components in the architecture. The Product Database stores product information. File Vault reserves product model files. The Product Information Management Module
provides an easy-to-use Web interface to support various activities of team participants, such as submitting new product designs, updating or deleting existing designs and for searching designs.
Chapter 6

Team Management Module

The Team Management Module is the outmost layer of the CyberEye infrastructure and comprises the Product Information Management Module, which in turn contains the Product Visualization Module (CyberEye Viewer). The Team Management Module not only manages the activities of team participants, but also supports both asynchronous and synchronous distributed interactions amongst them. In this chapter, the architecture of the module and the structure of the Team Database will be introduced. Thereafter the user activities allowed in the module will be described. Lastly, the architectures, operations and implementation details of the five communication facilities included in the module will be presented.
6.1 Architecture

The architecture of the Team Management Module is shown in Figure 6-1. The entire module is based on a secure user login mechanism to control the user activities with increased security. The user login mechanism is implemented using ASP and an SQL Server database. The tasks of the user login mechanism include adding new team participants to the Team Database (Register), authenticating the team participants through login names and passwords (Login), changing the information of team participants in the database (Update), removing team participants from the database (Delete) and looking up team participants in the database (Search). Only after team participants pass the authentication of the user login mechanism, can they access to the Product Visualization Module and the Product Information Management Module.

The Team Management Module provides five communication facilities to improve the cooperation and coordination amongst team participants. Email and the Web Forum support asynchronous distributed interaction. Online List, Instant Message Unit and Chat Room make synchronous distributed interaction possible.

The Team Database is used to store the information that is necessary for the user login mechanism and the communication facilities.
Figure 6-1. Architecture of the Team Management Module
6.2 Team Database Structure

The Team Database is the data source of the Team Management Module. The structure of the Team Database is depicted in Figure 6-2.

Figure 6-2. The Structure of the Team Database
Chapter 6 Team Management Module

The Team Database is used to store information pertinent to all team participants and messages on the Web Forum. Currently, the Team Database is implemented with Microsoft SQL Server. It consists of three tables: the User table, the User Type table and the Message table.

6.2.1 User Table

Detail information about fields in the User table is shown in Figure 6-3.
The User table stores information about all team participants. According to their application, these fields can be categorized into three groups: personal information fields, technical information fields and administrative information fields.

Personal information fields store personal information such as the last name, the first name, and the mail address, see Table 6-1. When team participants register, they are required to enter their personal information. They can update their personal information at any time once logged in.

<table>
<thead>
<tr>
<th>Fields Name</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>Yes</td>
<td>User's last name</td>
</tr>
<tr>
<td>First Name</td>
<td>Yes</td>
<td>User's first name</td>
</tr>
<tr>
<td>Email</td>
<td>Yes</td>
<td>User's Email address</td>
</tr>
<tr>
<td>PhotoPath</td>
<td>No</td>
<td>URL to user's photo image</td>
</tr>
<tr>
<td>HomePhone</td>
<td>No</td>
<td>User's home phone number</td>
</tr>
<tr>
<td>WorkPhone</td>
<td>No</td>
<td>User's work phone number</td>
</tr>
<tr>
<td>FaxNumber</td>
<td>No</td>
<td>User's fax number</td>
</tr>
<tr>
<td>Address</td>
<td>No</td>
<td>User's working address</td>
</tr>
<tr>
<td>Homepage</td>
<td>No</td>
<td>URL to user's Homepage</td>
</tr>
<tr>
<td>Comment</td>
<td>No</td>
<td>User's comments</td>
</tr>
<tr>
<td>DisplayOption</td>
<td>No</td>
<td>Whether or not display user's personal information</td>
</tr>
</tbody>
</table>

Table 6-1. Personal Information Fields

Technical information fields contain information about the role, duty, and position of team participants, see Table 6-2. The administrator, team leaders and team members are required to enter their technical information when they register. Suppliers and customers do not need to fill in technical information.
### Table 6-2. Technical Information Fields

Administrative information fields are used by the administrator for managing the activities of team participants, see Table 6-3. The LoginName and Passwords field are designated by team participants when they register. After registering, team participants cannot change their login name, but they can choose a different password. The UserType field is used to identify the user type of team participants. The user type is determined when team participants register. Only the administrator can change the user type for team participants. Except for the login name, password and user type, all administrative information is automatically generated and managed by the module.
Chapter 6 Team Management Module

<table>
<thead>
<tr>
<th>Fields Name</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserID</td>
<td>Yes</td>
<td>Automatically generated unique number</td>
</tr>
<tr>
<td>LoginName</td>
<td>Yes</td>
<td>Identify user</td>
</tr>
<tr>
<td>Password</td>
<td>Yes</td>
<td>Required in order to login</td>
</tr>
<tr>
<td>UserType</td>
<td>Yes</td>
<td>Identify user's status</td>
</tr>
<tr>
<td>AccessLevel</td>
<td>Yes</td>
<td>Used to define access level for different users</td>
</tr>
<tr>
<td>LastVisit</td>
<td>Yes</td>
<td>Last visited data and time</td>
</tr>
<tr>
<td>LastVisitHostName</td>
<td>Yes</td>
<td>Last visited host name</td>
</tr>
<tr>
<td>LastVisitIPAddress</td>
<td>Yes</td>
<td>Last visited IP address</td>
</tr>
<tr>
<td>LastVisitUserAgent</td>
<td>Yes</td>
<td>Last visited information (including browser name, operation system and language)</td>
</tr>
<tr>
<td>VisitCounter</td>
<td>Yes</td>
<td>Record how many times the user has logged in</td>
</tr>
<tr>
<td>MessagePosted</td>
<td>Yes</td>
<td>Record how many messages the user has posted</td>
</tr>
<tr>
<td>Online</td>
<td>Yes</td>
<td>Indicate if the user is currently online</td>
</tr>
</tbody>
</table>

Table 6-3. Administrative Information Fields

6.2.2 UserType Table

Participants of CyberEye are classified into five types: administrator, team leader, team member, supplier, and customer, see Table 6-4. Different user types have different authorization levels to access the collaborative design environment. For example, the administrator has the highest right to manage the accounts of all team participants and the product model information. The lowest access right is given to customers. Customers can only view messages on the Web Forum but they cannot post new
Chapter 6 Team Management Module

messages. Table 6-5 summarizes the detailed information about access rights for each user type.

<table>
<thead>
<tr>
<th>User Type</th>
<th>Access Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator</td>
<td>1</td>
<td>Manage the collaborative design environment</td>
</tr>
<tr>
<td>Team Leader</td>
<td>2</td>
<td>Manage all team members</td>
</tr>
<tr>
<td>Team Member</td>
<td>3</td>
<td>Submit new design and modify own design</td>
</tr>
<tr>
<td>Supplier</td>
<td>4</td>
<td>View designs in database and post messages</td>
</tr>
<tr>
<td>Customer</td>
<td>5</td>
<td>View designs in database</td>
</tr>
</tbody>
</table>

Table 6-4. User Types in CyberEye

<table>
<thead>
<tr>
<th>Access Level</th>
<th>Administrator</th>
<th>Team Leader</th>
<th>Team Member</th>
<th>Supplier</th>
<th>Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Chang register passwords</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Monitor On-line Activities</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Delete Product Models</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Delete or Modify Messages on the Forum</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Delete or Update Own Product Models</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Modify All Product Model Information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Use Instant Message Unit</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Modify Own Product Model Information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Submit New Product Model</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Search product model information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Search User Information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
</tr>
<tr>
<td>Use CyberEye Viewer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
</tr>
<tr>
<td>View User Information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>-</td>
</tr>
<tr>
<td>View product model information</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Read Messages on the Forum</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 6-5. User Access Rights Table

124
The UserType table is used to store the information of all user types, identified in Figure 6-4. The UserTypeID field uniquely identifies each user type. The UserType field stores the name of the user type. The Password fields contain the register passwords for all user types. The register password is needed when a new team participant registers. It is used for preventing unauthorized registration. Only the administrator has the right to change these passwords.

![UserType table](image)

**Figure 6-4. UserType table**

### 6.2.3 Message Table

The Message table stores all messages posted on the Web forum and contains six fields, as shown in Figure 6-5. The MessageID field identifies each message. It will be automatically increased by one for each new message. The UserID field connects with the UserID field in User table, so the Web forum knows who posted the message. The Parent field is used to track the relationship between messages. The Subject field stores
the subject of the message. The Date field maintains the date and time when the message was posted. The Message field stores the content of the message.

Figure 6-5. Message Table

6.3 User Activities Flow

In this section, we will describe the activities flow of the team participants in the Team Management Module. Figure 6-6 shows the user activities flow diagram.
Figure 6-6. User Activities Flow in CyberEye
Chapter 6 Team Management Module

Register

All participants of CyberEye are required to register before they can log in. To register as one of the user types, participants must provide a register password of this user type, see Figure 6-7. The purpose of register password is to prevent unauthorized persons from registering and gaining the access to CyberEye. The register passwords for all user types are stored in the UserType table of the Team Database.

![Figure 6-7. Choose User Type](image)

After choosing a user type and providing a correct register password, a register form will be displayed to allow participants to enter their information, see Figure 6-8.

For different user types, the content of the HTML form will be different. For example, while a customer only needs to enter personal information (such as last name, first name and email), a team member has to enter technical information (such as discipline, role and contributions) as well as personal information. After clicking on the
Submit button, information in this form will be sent to the Web server and stored in the Team Database.

Figure 6-8. Register Form

If a team participant wants to add his/her personal photos to the Team Database, he/she is required to choose the photos file and upload it to the Web server, see Figure 6-9. The URL that point to the image files will also be saved in the Team Database with other personal information.
For security reason, all team participants must log in with their login name and password in order to use CyberEye, see Figure 6-10. The login page will remember the password of a team participant if the “Remember my password” check box is selected, so the team participant does not need to input his/her password with each subsequent login.
If a team participant forgets his/her password, the Team Management Module will automatically send this participant an email containing the password if the participant fills in the login name and clicks the "Send Password" button.

After a team participant successfully logs in, a new user session will begin and a session variable will be created to save the login name of this participant. This session variable will be used to identify this participant during the whole user session. At the same time, the value of the Online field for this participant will be set to True, indicating the status of the participant is online.
Chapter 6 Team Management Module

During the user session, the participant can perform various activities permitted by the system, such as reading messages, posting messages, submitting a new design, modifying the design information, checking who is on line, meeting with other people in the Chat Room and sending instant messages.

Update

A team participant can update his/her information any time he/she logs in. On the Update page, a HTML form will display the current information as well as a current photo, as presented in Figure 6-11.
Except for their login name and user type, team participants are allowed to change all other personal information and upload new photo files.

Search

The Team Management Module provides a convenient search function to help team participants quickly find a particular participant in a design team. The search can be conducted using multiple criteria including login name, user type, last name, first name, email and homepage. Display check boxes are used to specify what kind of information will be displayed in the search results. By default, the search results will display login name, user type, email, last visit time, total visit time and the number of messages this participant has posted on the Web Forum. Order radio buttons are used to select a listing order for such search results. By default, the search results will be arranged in alphabetic order according to login name. Other order options include register time, last name, visit times, last visit time and the number of messages posted. After submitting the search criteria, the search result will be displayed in a Web page, where a team participant can send email or instant message to the team participants listed in the search result.

The User Search page and a sample search result are shown in Figure 6-12. In this sample, the search criteria include the login name and the user type. The search result only contains one person.
Figure 6-12. Search User Function
**Logoff**

Team participants are recommended to exit CyberEye by clicking the “Logoff” button on the top menu bar, shown in Figure 6-13, instead of directly closing the browser window. By doing this, the user session will expire immediately, and this prevents unrelated persons from gaining access to CyberEye by clicking the “Back” button of the Web browser. At the same time, the Team Management Module will change the status of the participant from online to offline.

![Figure 6-13. Logoff from CyberEye](image)

**6.4 Communication and Coordination**

The Team Management Module provides five integrated communication facilities to support both asynchronous and synchronous distributed interactions amongst team participants. While email and the Web Forum support asynchronous interactions, the
Online List, Instant Message Unit and Chat Room allow synchronous, real time interaction.

6.4.1 Email

Electronic mail (email) is one of the most convenient and effective ways to transmit information on the Internet. Email is not only used to send text information, but also can transfer multimedia information, such as image files, sound files or even video clips. Although email is not a real time communication medium, its explicit asynchrony gives team participants enough time to read and respond, and allows them to feel in full control of their work. The Team Management Module includes two tools to utilize the advantages of email: the Automatic Email Notification Unit and the Mailing List Generating Unit.

Automatic Email Notification Unit

Based on SMTP (Simple Mail Transfer Protocol) service provided by the IIS, the Automatic Email Notification Unit is implemented with ASP technology. This unit can automatically generate and send notification emails without human interaction by responding to various activities of team participants. Depending on the types of the activities, the recipients of the notification emails may be all participants, team members or simply the administrator.
Chapter 6 Team Management Module

The architecture of Automatic Email Notification Unit is shown in Figure 6-14. The core component is the New Mail object, which belongs to the CDONTS (Collaboration Data Objects for NTS Component) library developed by Microsoft. With the help of SMTP services, the New Mail object makes automatic generation of email and its delivery possible.

The procedure for sending emails is described as follows: First, various user activities trigger the ASP scripts through the Web server. The ASP code will query the Team Database to get the email addresses of recipients and other related information. The ASP code will then use this information to automatically generate the email, and deliver the email through the Send method of the New Mail object.
Figure 6-14. Automatic Email Notification Unit
Activities that can trigger notification emails include:

- **Forgetting a password**

  If a team participant forgets his/her password, the Automatic Email Notification Unit can automatically send an email containing the password to the participant. The participant only needs to enter his/her login name and click the "Send Password" button, as shown in Figure 6-15. The unit will look up the corresponding email address in the database according to the login name. If the record is found, an email containing the password will be generated automatically and sent to the email address.

![Image of login page](Image)

*Figure 6-15. Send Password Via Email*

- **Registering**

  When a new team participant registers, an email containing register information of this new participant, such as login name and password, will be sent to him/her for
future reference. At the same time, this email will also be sent to the administrator to notify him/her of this event.

- **Submitting a new design**

  When a team member submits a new design, an email will be sent to all team members, team leaders and the administrator. The email contains brief information about the new design, such as the name of the product design and the name of the designer.

- **Updating a design**

  If a team member updates an existing design (by uploading a new product model file or by modifying product information), this event will be broadcast to all team members, team leaders and administrator by automatically generated emails.

- **Replying to a message**

  If a message on the Web Forum is replied to, an email will be sent to the team participant who posted this message, so the team participant can visit the Web Forum and check the reply in time.

**Mailing List Generating Unit**

The Mailing List Generating Unit is a Web-based facility that helps team participants to create a mailing list dynamically, according to their requirements. The mailing list can be generated using different search criteria in Search User page. The
search criteria include Login Name, User Type, Last Name, First Name, Email and Homepage, see Figure 6-16.

![Search for users](image.jpg)

**Figure 6-16. Generating Mailing List**

For example, to generate a mailing list containing all team members, the team participant only needs to select "Team member" from the User Type drop down list and clicks the "Search Users" button, as shown in Figure 6-16. A list of all team members is displayed in the next Web page, see Figure 6-17. If the participant wants to send email to all persons on the list, he/she just clicks on the "Email to All" button. To send email to only some of the persons on the list, the participant needs to select the check boxes of
the persons whom he/she want to send to, then click on the “Email to Selected” button. The default email software (such as Netscape Messenger or Microsoft Outlook) will be opened automatically and email addresses of these persons will be filled in automatically. The other two buttons, “Select All Users” and “Reverse Selection” help the participant to make the selection. The “Select All” button selects the check boxes of all persons and the “Reverse Selection” will reverse the previous selection made.

Figure 8-17. Mailing List
6.4.2 Web Forum

User forums have existed for a long time. Before the appearance of the Internet, user forums were created on a Bulletin Board Service (BBS). In a BBS, questions can be read, posted and replied to as required. Currently BBS has been replaced almost completely by their Internet equivalents, newsgroups and Web forums. Unlike Newsgroup that requires a newsreader such as Outlook or Netscape Messenger, Web forum is Web-based and only requires a Web browser. Another advantage of Web forum is that it can be tightly integrated into other systems.

The Web Forum in CyberEye is mainly used for the asynchronous discussion between the administrator, team leaders, team members, suppliers and customers. All information of the Web Forum is stored in the Team Database, see Figure 6-18. Two tables are required to store information of the Web forum. The Message table stores message information, see Figure 6-5, including message subject, posting dates, UserID of the team participant who posted the message as well as the content of message. The User table contains the list of team participants, refer to Figure 6-3. These two tables are connected together by the UserID field. Team participants interact with the Web Forum through a Web page dynamically created by ASP code. ASP code is responsible for the interactions between the Web Forum and the Team Database, such as querying information and updating or deleting records.
Figure 6-18. Architecture of the Web Forum
Chapter 6 Team Management Module

Figure 6-19 presents the main screen of the Web Forum. All messages are grouped by conversation and listed in the order of posting date and time. Each message entry contains the subject of the message, the login name of the team participant who posted it as well as posting date. The bulletins of messages posted in last two days are displayed in red and yellow color to attract attention. If messages on the forum cannot be displayed in one page, the forum will automatically divide messages into multiple pages. “Next Page” or “Previous Page” buttons can be used to navigate through all pages.

![Image of Web Forum](image)

Figure 6-19. Main Screen of the Web Forum
In the Web Forum, team participants can do the following activities:

- **Viewing messages**

  Team participants can view a message by clicking the subject of the message on the Web Forum. In addition to the content of the message, the subject and the name of the poster and posting date and time is displayed.

![View Message](image-url)  

*Figure 6-20. View Message*
• **Posting new messages**

   To post a new message, team participants need to click on the “Post” button on the left side menu. A Web page containing a HTML form allows team participants to enter the subject and content of the message, see Figure 6-21.

   The content of the message supports the HTML format, so team participants can use HTML to format messages (i.e., change font and color of the text) or add images or HTTP links.

![Figure 6-21. Post a New Message](image-url)
• Replying to messages

When a message is displayed, team participants can reply to this message by clicking the “Reply” button link at the bottom of the View Message page, as shown in Figure 6-20. A HTML form will be displayed to allow team participants to type in the subject and the content of the message. The content of the original message will be at the top of the form, see Figure 6-22.

Figure 6-22. Reply to a Message
• **Modifying Messages**

After a message has been posted on the Web Forum, the team participant still has a chance to modify it by clicking the “Modify” link at the bottom of the View Message page shown in Figure 6-20. After modification, the posting date of the message will be automatically updated to the new date and time.

• **Deleting Messages**

Only the administrator has the right to delete any messages on the Web Forum; team participants can only delete messages posted that they posted. A message can be deleted by clicking the “Delete” link at the bottom of the View Message page shown in Figure 6-20. Deleted messages cannot be recovered.

• **Viewing User Information**

On the main screen of the Web forum, the information of a team participant can be viewed by clicking on the login name. A Web page will display detailed information about this team participant as well as a personal photo (should there be one). Different user types can view different levels of information. For example, the administrator can view the administrative information such as IP Address, Host Name and User Agent. Team members can view personal information and technical information, such as
Discipline, Role and Duty. Customers can only view personal information, such as Last Name, First Name and Email Address, see Figure 6-23.

Figure 6-23. User Information

6.4.3 Online List

It is difficult for the Web application to track the online activities of users because HTTP is a stateless protocol. Figure 6-24 shows a typical HTTP transaction. When a Web server receives an HTTP request, it responds to this request and terminates the
connection between the Web server and the client. The connection between the client and the server exists only during the request and response process. Although this feature makes HTTP efficient for distributing information, the feature makes it difficult for Web applications to track client sessions and to remember information. The result is that the system has no idea how many users are currently online and who are they.

(1) The client sends a request to the server and establishes an HTTP connection

(2) The Web Server respond to the request and sends result back to Web Client.

(3) The connection is broken after the result is transmitted to the Web client

Figure 6-24. A typical HTTP transaction
In CyberEye, a novel approach is proposed to solve this problem. In this approach, the user login mechanism and the Team Database work together to detect the online status of the team participants. Team participants who are currently logged in are displayed in a list called the Online List shown in Figure 6-25.

<table>
<thead>
<tr>
<th>Login Name</th>
<th>Login Time</th>
<th>Send Email</th>
<th>Send Instant Message</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:bujian@mie.utoronto.ca">bujian@mie.utoronto.ca</a></td>
<td>128.100.49.124</td>
<td></td>
<td></td>
</tr>
<tr>
<td><a href="mailto:jilin@mie.utoronto.ca">jilin@mie.utoronto.ca</a></td>
<td>128.100.49.122</td>
<td></td>
<td></td>
</tr>
<tr>
<td><a href="mailto:kaveh@mie.utoronto.ca">kaveh@mie.utoronto.ca</a></td>
<td>128.100.49.200</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 6-25. Online List

The concept of the Online List is illustrated in Figure 6-26. There is a field named Online in the User table of the Team Database, see Table 6-3. The data type for this field is Boolean. "True" indicates this team participant is currently logging on. "False" indicates this team participant is offline.
Figure 6-26. The Architecture of the Online List
Chapter 6 Team Management Module

When a team participant logs in, the Team Management Module will set the Online field of this team participant to "True". If the team participant logs off by clicking the "Logoff" button on the top menu, shown in Figure 6-13, the Online fields will be set to "False". The Online List is generated by selecting all team participants whose Online fields are equal to "True". To make the Online List more accurate, all participants of CyberEye are required to quit CyberEye by clicking the Logoff button, instead of directly closing the browser window.

The Online List is displayed on both the Web Forum and the Chat page. Figure 6-27 shows the appearance of the Online List on Web Forum page. The Online List displays login names, login times, email addresses and IP addresses of all team participants who are currently online. By clicking on the email address of a participant, email can be sent to the team participant. By clicking on the IP address, an instant message will be sent to the team participant.
6.4.4 Instant Message Unit

Both email and Web forum are not real time communication tools. They only support asynchronous distributed interactions. To support synchronous distributed interactions amongst geographically distributed team participants, a Winsock-based Instant Message Unit was developed in CyberEye.
Compared with email, the Winsock-based instant message unit has the advantage of speed. While email may take minutes to hours to arrive, instant messages usually only take less than one second to transmit. Moreover, unlike Web-based message systems (such as the Web Forum), where a team participant needs to reload or refresh to view the newest update, the instant message displays on a team participant’s computer immediately and automatically, accompanied by a beep to alert the participant. Because of these two advantages, the Instant Message Unit can be used as a communication tool to supplement the Web-based message system.

The architecture of the Instant Message Unit is shown in Figure 6-28. The unit consists of server and client components. The Server component is a COM object called Instant Message Server. It is installed on the server and can be activated by ASP code through the Web server. The Instant Message Server is responsible for sending instant messages to client components according to their IP Addresses.
Chapter 6 Team Management Module

Figure 6-28. Architecture of Instant Message Unit
The client component is stand-alone software called Instant Message Receiver. Developed with Visual Basic, Instant Message Receiver runs as an independent program outside the Web browser. Team participants need to download the setup file from the Setup page and install the Instant Message Receiver on their computer, see Appendix A: CyberEye Viewer Setup). The interface of the Instant Message Receiver is shown in Figure 6-29.
Chapter 6 Team Management Module

The interface consists of a list window, a check box and a “Clear list” button. The list window displays instant messages in the order of the time they were received. The check box is used to turn on or off the beep when a new instant message arrives. Clicking on the “Clear list” button will clear the content in the list window. When the server component sends out an instant message, it usually takes less than one second to transmit to the receiver. Team participants do not need to open a Web browser and log in to CyberEye to receive instant messages. As long as the receiver is running, team participants will receive instant messages immediately.

There are two applications for the Instant Message Unit in CyberEye. First, it provides a real time communication tool. Secondly, it can be used to monitor the online activities of team participants.

Real Time Communication Tool

Team participants can send instant messages to each other just like sending email. However, there are two basic differences between instant messages and email. First, email is much slower than instant messages. Secondly, while email uses an email address to find the recipient, instant message uses the IP address to find the computer of the recipient. The IP address of a team participant is automatically detected when the team participant logs in.

The Online List can be used to send instant messages to online team participants, see Figure 6-30. Clicking on an IP address in the Online List will display the Instant
Message page. On this page, the login name and IP address of that person will be filled in automatically in the Login Name and IP Address text fields. Team participants only need to type the content of the message in the text area and click the "Send Message" button.

![Current Online](image1)

![Instant Message](image2)

Figure 6-30. Send instant message by using Online List
Another way to send instant messages is to use the search function to get a list of team participants. After typing the content of the message, the team participant can choose to send instant messages to all persons in the list by clicking the “Send to All” button, or send to some persons selected (check boxes on the left column of the list are selected) by clicking “Send to Selected” button. In this way, instant messages can be sent to multiple persons at the same time, see Figure 6-31.

![Image of the user interface for sending instant messages](image_url)

**Figure 6-31. Send instant message by using user list**
Online Activities Monitor

Instant messages can also be triggered by user activities. This feature can be used to monitor and coordinate the online activities of team participants. The following activities will trigger instant messages:

- **Login**

  When a team participant logs in, the Instant Message Unit will send an instant message to the administrator, team leaders and team members, as long as their Instant Message Receiver is currently running. The instant message contains information about the login name, the login time and the host name of the computer. Figure 6-29 shows some sample messages displayed in the Instant Message Receiver.

- **Logoff**

  If the a team participant logs off from CyberEye by clicking on the “Logoff” link, refer to Figure 6-13, an instant message will be sent to all team participants except suppliers and customers.

- **Submitting new product designs**

  Each time a team member submits a new product design to CyberEye, an instant message will be sent to team leaders, team members and the administrator.

- **Updating product designs**
When a team member updates the product model, or modifies the associated product information, an instant message will be sent to team leaders, team members and the administrator.

- **Posting messages**

Posting a message to the Web Forum will also activate the Instant Message Unit to notify all team participants that a new message has been just posted on the Web Forum.

- **Visiting the Chat Room**

When a team participant visits the Web page containing the Chat Room, an instant message will notify all other team participants.

- **Other activities**

Some activities only send instant messages to the administrator for administrative purpose. For example, when a team participant reads a message on the Web forum, the instant message sent to administrator contains the subject of the message as well as the login name of this the participant and time. Other activities include clicking links, visiting Web pages, or downloading model files. The administrator can use these instant messages to monitor the online activities of team participants.
6.4.5 Chat Room

The Chat Room is dedicated to the support of real-time communications among the team participants. Unlike the Web Forum, chat rooms allow team participants to interact on a personal, real time basis.

The Chat Room in the Team Management Module consists of server and client components as shown in Figure 6-32. The server component of the Chat Room is called Chat Server, which is used to manage the communications amongst Chat Room client components. The Chat Server is part of Microsoft Exchange Server and runs on the server with IIS. The Chat Room client component is an ActiveX control that runs inside the Web browser. Because only Microsoft Internet Explorer supports ActiveX control internally, Microsoft Internet Explorer is recommended as the Web browser in CyberEye. The Chat Room client components will be automatically installed on the computers of team participants when they visit the Chat Room for the first time. After that, no download and installation are necessary.

The Chat Room includes a button, a messages box, an input text box and a list window. Clicking on the “Join the Chat” button allows the participant to enter the Chat Room. The login name of the team participant will be used in Chat Room. After the team participant enters the Chat Room, the “Join the Chat” button will change to “Leave the Chat” button. Clicking on the “Leave the Chat” button will exit the participant from the Chat Room.
Chapter 6 Team Management Module

All chat messages sent by the users will be displayed in the messages box immediately. The input text box is used to type a new message. The list window lists the login name of all participants that are currently in the Chat Room.

Figure 6-32. Architecture of the Chat Room
Chapter 6 Team Management Module

The Chat Room supports two types of chat modes: announce mode and whisper mode. While announce mode will send chat messages to all persons currently in the chat room, whisper mode allows the team participant just send chat messages to one person.

![Chat Room](image)

**Figure 6-33. Chat Room**

The Chat Room Web page also contains the Online List. Online List displays team participants that are currently logged in. To invite other team participants to enter the
Chapter 6 Team Management Module

Chat Room, the team participant can send them instant messages by clicking on the IP Addresses of them.

6.5 Summary

The Team Management Module is the foundation of CyberEye. It contains the Team Management Module, Product Visualization Module. The Team Management Module provides a secure user login mechanism to protect the whole collaborative design environment. To achieve better collaboration and coordination amongst team participants, the module integrates five communication facilities: the automatic email and the Web Forum support asynchronous distributed interactions. The Online List, the Instant Message Unit and the Chat Room are dedicated to enable synchronous distributed interactions. Of these facilities, the Online List and the Instant Message Unit are two novel approaches developed in this thesis.
Chapter 7
Conclusions and Future Development

7.1 Summary and Conclusions

This thesis presents an Internet-enabled, platform-independent, distributed collaborative design environment by applying cutting-edge 3D technologies and Internet technologies. The proposed environment integrates a Product Visualization Module (CyberEye Viewer), a Product Information Management Module and a Team Management Module to achieve better information sharing, access control, collaboration and coordination through the Internet.

To overcome the drawbacks in existing CAD browser, a novel approach is presented to implement the 3D CAD browser. In this approach, Java 3D API is used to
Chapter 7 Conclusions and Future Development

achieve 3D display and manipulation, a Java applet is used to create front-end user interface and a Java servlet is adopted to connect to a database and exchange information with the Java applet. Based on this approach, a prototype CAD browser called CyberEye Viewer was developed. Compared with existing CAD browsers, CyberEye Viewer has advantages of being Internet-enabled, platform-independent, low cost, extensible and database-bounded.

The Product Information Management Module achieves the shared understanding amongst team participant through shared product models and shared product information saved in database. This reduces the ambiguity associated with traditional broadcasted product data management. The Product Information Management Module also provides facilities for two-way information exchange between clients and the server. Team participants can view product information, download product models, add new product designs and update or remove existing product designs.

To achieve better collaboration and coordination in the collaborative design environment, several cutting-edge Internet technologies are applied to the Team Management Module. While the automatic email and the Web Forum are used to support asynchronous distributed interactions, the Online List, the Chat Room and the Instant Message Unit allow synchronous distributed interactions amongst geographically dispersed team participants. The Team Management Module is also used
to organize and control access of the shared information, and thus increase the security of the environment.

7.2 Highlight of Contributions

There are three major contributions from this thesis.

1. This thesis investigated some of the cutting-edge Internet technologies that will affect traditional engineering practice, with a particular focus on how these technologies can be exploited in an Internet-enabled, platform-independent collaborative design environment.

2. The thesis proposed a novel approach to enable distributed 3D display and manipulation on the Internet and developed a prototype CAD browser called CyberEye Viewer based on this approach. The new 3D technology, Java3D API, and the new generation server-side technology, Java servlet, are applied in the proposed CyberEye Viewer.

3. This thesis presented a new approach to support synchronous distributed interactions in a Web-based application. In this approach, ASP is used to implement the infrastructure, a novel facility called Online List is used to reflect the online status of team participants, and a Winsock-based Instant Message Unit is used to achieve real-time communication and monitor online activities of team participants.
7.3 Future Work

Development of STEP Loader

One important limitation of VRML is that it is only an approximated 3D representation of the geometric model. Therefore, VRML works well if team participants only need to view product models. But it is impossible to do online design or modification, because unlike a solid model, VRML does not contain all geometric information required by an engineering application. The current implementation of CyberEye only supports online viewing. In the future, online design and online modification will be added so that team participants can use a Web browser to design or modify the product model on the Internet. For this reason, another data format that represents complete geometric information will be supported in CyberEye Viewer. This format is STEP.

STEP (The Standard for the Exchange of Product Model Data) is a comprehensive ISO standard (ISO 10303) that describes how to represent and exchange digital product information. STEP provides an unambiguous representation and an exchange mechanism for computer-interpretable product information throughout the life cycle of a product. The standard aims to provide a consistent data exchange format and application among different application systems such as CAD, CAM, and CAE [Gu and Chan, 1995].
Chapter 7 Conclusions and Future Development

Because the structure of CyberEye Viewer is module-based, it is easy to add the support for STEP by adding a STEP loader. The STEP loader can read and write geometric information from STEP files generated by popular commercial CAD software (such as Pro/Engineer and I-DEAS). With the STEP loader, CyberEye Viewer allow team participants to modify existing product model or even design a new product model on the Internet.

Extension to Feature-based Design

In future implementation, CyberEye Viewer should support feature-based product models. Such functionality can be added to CyberEye Viewer: when a team participant clicks on a feature, that feature will change color, indicating the feature is being selected. At the same time, engineering information associated with that feature will be displayed on CyberEye Viewer.

Java3D is a scene graph-based API. Since the concept of scene graph is similar to the concept of a feature-based design, it is feasible to develop a feature-based CAD browser using Java and Java3D.

Animation Support

The current implementation of CyberEye Viewer does not support animation functions. It is possible to add animation to CyberEye Viewer because VRML supports animation internally. Animation can be used to demonstrate the process of assembly.
Team participants should be able to use mouse to manipulate different parts and assemble them together.

**XML-based Product Information**

XML (Extensible Markup Language) is a subset of the Standard Generalized Markup Language (SGML). XML provides an application-independent self-describing format, in which data can be shared. Although XML is still under development and currently supported only by Microsoft Internet Explorer 5.x, XML is the future of all data manipulation and transmission.

The current implementation of CyberEye presents product information using HTML. It is possible to present all product information using XML in the future. XML-based product information can be used as a universal data exchange standard amongst different enterprises. Various XML-enabled applications (not just Web applications) can read and manipulate XML-based product information without prior agreement.


Bibliography


Appendix A

CyberEye Setup

CyberEye Viewer Setup

CyberEye Viewer is a Java applet that uses Java 3D API to display 3D objects. It will be automatically downloaded from the Web server and executed on computers of users. Users do not need to install CyberEye Viewer on their own computer. However, user should install Java plug-in, Java 3D Runtime Environment and VRML Loader in order to run CyberEye Viewer on their computer.

Users can download and install these software programs from the Setup page. Figure A-1 shows the Setup page in CyberEye.
1. **Install Java Plug-in**

The current Web browsers, Netscape Navigator and Microsoft Internet Explorer, were developed and shipped with older version of Java Virtual Machine (JVM). But the Swing and Java3D API adopted in CyberEye viewer require the latest Java 2 version (or later) from Sun. Eventually, browser development will catch up to the current Java version, but until then, a Java plug-in is required to run CyberEye Viewer Java3D applets. The Java plug-in is software that allows the latest Java virtual machine to run
inside Netscape Navigator or Microsoft Internet Explorer. Users can download the Java plug-in free from Web site of Sun (http://java.sun.com.)

2. Install Java3D Runtime Environment

The Java3D Runtime Environment is necessary for running Java3D applet. It can also be download from the Web site of Sun Microsystems Inc. for free. To install it, users need to run the setup file and following the instructions given by the setup program.

3. Install VRML Loader

Vrml97.jar contains a VRML97 Java 3D Loader developed by The Java3D and VRML Working group. To install it, users need to copy the file (vrm197.jar) to the lib directory where the Java plug-in is installed. For example, if the Java plug-in is installed in the default directory, the vrm197.jar should be copied to “C:\Program Files\JavaSoft\VRE\1.2\lib”.

Setup for the Instant Message Unit

The client component of Instant Message Unit is stand-alone software (called Instant Message Receiver) that runs outside of the Web browser. To use the Instant Message Unit in CyberEye, team participants need to install the Instant Message Receiver on their computers.
Appendix A CyberEye Setup

The installation can be finished in three steps. First, download the zip file from the Setup page in CyberEye. Secondly, unzip the zip files to a directory. Finally, run the setup.exe file and following the instructions of the setup program (See Figure A-2). After installation finishes, the folder of CyberEyeIMS will be automatically added to the Start Menu of the Windows. Users can start the Instant Message Receiver by click on the Start Menu.

Figure A-2. Instant Message Client Setup Program
Appendix B CyberEye Interface

Figure B-1 Cover page of CyberEye
Figure B-6 CyberEye Viewer

Figure B-7 Searching user
Appendix B CyberEye Interface

Figure B-8 Search result

Figure B-9 User information
Figure B-12 Instant Message Receiver