

2006

The development of a virtual reality based CAD system for design review

Thomas Anthony Erlemeier
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Mechanical Engineering Commons](#)

Recommended Citation

Erlemeier, Thomas Anthony, "The development of a virtual reality based CAD system for design review " (2006). *Retrospective Theses and Dissertations*. 1376.
<https://lib.dr.iastate.edu/rtd/1376>

This Thesis is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

The development of a virtual reality based CAD system for design review

by

Thomas Anthony Erlemeier

A thesis submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE

Co-majors: Human Computer Interaction; Mechanical Engineering

Program of Study Committee:
James Oliver, Major Professor
Judy Vance
Thomas Rudolphi

Iowa State University

Ames, Iowa

2006

Copyright © Thomas Anthony Erlemeier, 2006. All rights reserved.

UMI Number: 1439839



UMI Microform 1439839

Copyright 2007 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

DEDICATION

This thesis is dedicated to my late mother, Mary Erlemeier, who inspired me to become an engineer.

Table of Contents

Abstract	iv
1. Introduction	1
1.1 Motivation	1
1.2 Design Criteria	3
2. Prior Work	5
2.1 Interaction Methods of Prior Systems	5
2.2 Architecture of Prior Integrated CAD/VR Systems	7
3. Designing a User Interface for an Integrated VR/CAD System	13
3.1 Design Criteria	13
3.2 VR Interaction Methods	13
3.3 User Interface for Tablet PC	19
4. Architecture Overview	27
4.1 Integration Method	27
4.2 Underlying Components	34
5. Results	40
5.1 System User Interface	40
5.2 System Performance	47
5.3 System Reliability	54
6. Conclusions and Future Work	58
6.1 Conclusions	58
6.2 Future Work	61
Bibliography	64
Acknowledgements	67

Abstract

Virtual reality is currently utilized to view and evaluate product designs but is not used to alter designs. The design is typically completed on desktop workstations with CAD software and then evaluated within a virtual environment. Integrating VR and CAD could potentially improve designs by allowing several engineers and managers to critique and modify designs together, enabling concurrent engineering.

This thesis discusses the development of a virtual reality based CAD system for altering designs during a design review. A list of criteria was first created to assess the system's effectiveness in a design review which formed the basis for the development. The system integrates Pro/Engineer using the J-Link interface and VR Juggler. A wireless networked tablet PC along with a wand and tracker were chosen to operate the system. The developed system proved to meet the majority of the criteria and shows potential to meet all of the criteria.

1 INTRODUCTION

1.1 Motivation

Although virtual reality (VR) is becoming a widespread engineering tool in industry, it is not currently integrated into computer aided design (CAD) systems. CAD is primarily restricted to desktop workstations, consisting of the standard monitor, keyboard and mouse. By contrast, integrated CAD and VR would immerse the user in a virtual environment, allowing users to potentially communicate their designs more effectively. Current technology allows CAD models to be viewed in VR but does not permit users to easily modify the design or communicate the changes back to the CAD system [15].

Immersive projection technology (IPT) such as a CAVE shown in Figure 1 would allow several engineers and managers to view and critique designs together improving the overall design by practicing concurrent engineering [2, 3]. A large stereoscopic display would permit more models to be displayed at full scale and in three dimensions, increasing the sense of realism.

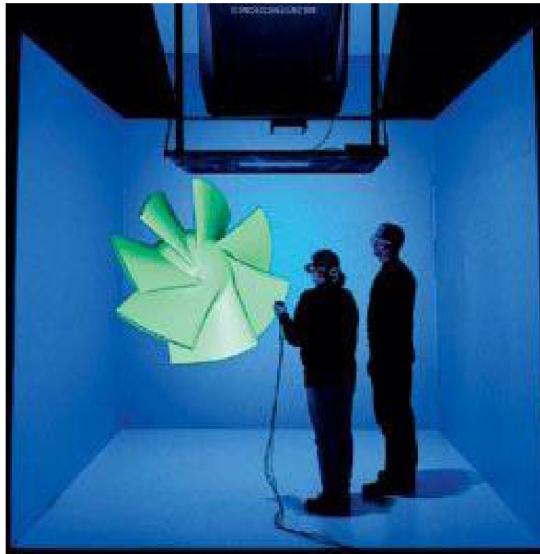


Figure 1. Photo of a CAVE [4]

IPT allows users to interact with the virtual world and also with the real world simultaneously, also known as augmented reality (AR). This unique interaction technology could also be beneficial during the design process. An example of utilizing AR for the design process would be designing the interior of an automobile in a CAVE. A car seat could be placed into the CAVE in the context of a full-size virtual automobile model. The user could then evaluate the design of the interior while actually sitting in the virtual automobile. The stereoscopic display would display the geometry in 3D making it easier for the user to define the relative spatial relationships of different parts [31]. In order for the user to make modifications to the CAD geometry while sitting in the virtual automobile an integrated CAD/VR system would be needed.

An integrated VR/CAD system would not replace the standard method of using desktop workstations but would instead complement their use in the design process. Currently, VR is much too expensive to be used as the primary CAD tool. The cost of a single VR system limits the number that any one company would likely invest in, thereby limiting the number of designers who could work concurrently on a project. As a result, it would be more practical to use an integrated VR/CAD system for design review rather than design creation.

The desire to improve the design process, decrease production time, and decrease development costs by integrating CAD and VR has motivated research in this area. As with many VR applications, the largest challenge has proven to be the design of an interaction method to operate the application [14]. The interaction method must be designed for the ease of use of all available features. CAD systems used on desktop workstations have interface design issues and some of those same problems will carry over to VR, and it will add its own set of challenges. Another large challenge will be designing the system to meet the desired performance with current computer technology.

1.2 Design Criteria

1.2.1 Criteria for an Integrated VR/CAD System

There are several reasons why desktop workstations should still be used for design creation. As a result, an integrated VR/CAD system would be used for design review only. During a design review, a user may want to manipulate constraints on the design to show minor design modifications. Creating new constraints, which is essentially design creation, would not be appropriate in a design review. Creating constraints would slow the review process, and the user interface would be extremely complex.

Since the integrated system would be used for design review only, most of the design work would still be done with commercial CAD systems on desktop workstations. Due to this fact it is important that the modified geometry, done with the integrated system, remains fully compatible with the commercial CAD system that was used initially to create the design. CAD systems are often capable of exporting the geometry to a neutral file format such as IGES or STEP to be imported by other CAD systems, but the ability to modify such non-native geometry is generally limited [18, 19]. Figure 2 below illustrates the criterion of how the integrated VR/CAD system would be used in parallel with a commercial CAD system.

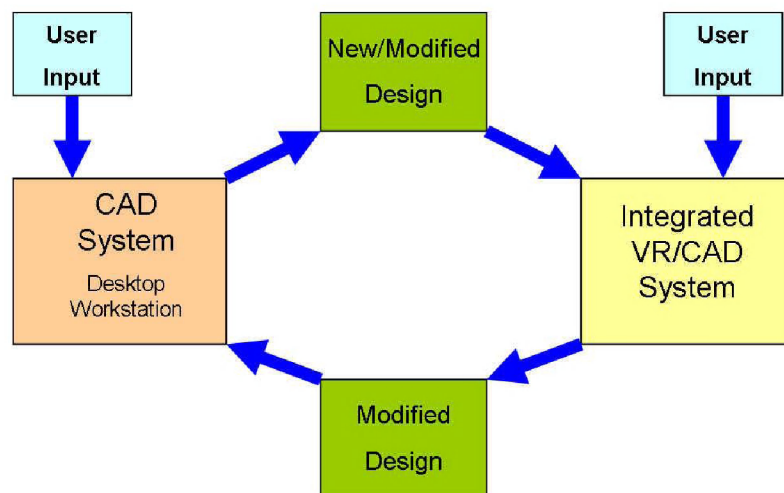


Figure 2. Illustration of how the design data remains compatible with the commercial CAD system and the integrated VR/CAD system

The integrated VR/CAD system must run smoothly and reliably. When a constraint is changed, the system should update the model relatively quick. The system must also be stable so that it does not terminate unexpectedly during a design review.

Navigation throughout the virtual world would be necessary to look at the model from different perspectives. This is important to see all the features of the design and allow for a better critique of the design.

The most important criterion of an integrated VR/CAD system is that it must be profitable for businesses to implement. The costs of the system and the system support must be low enough so that the company using it would still make a profit. A portion of industry is already utilizing VR, making it more economically feasible to add functionality to their existing systems.

1.2.2 Criteria for an Interaction Method

During a design review, the interface of the integrated VR/CAD system must allow the user to make design changes. Below is a list of the criteria the interaction method must satisfy.

- Allow modifications of both part and assembly constraints
- Allow users to input exact numerical constraint values
- Allow the user to modify constraints while involved in a design discussion
- Allow the user to make modifications quickly
- Allow the user to stand
- Allow the user to navigate throughout the virtual world
- Be able to be used by no more than one person
- Be quick to learn
- Be ergonomically sound

2 PRIOR WORK

2.1 Interaction Methods of Prior Systems

2.1.1 Interaction Methods of Prior Systems with Design Creation

An integrated system with design creation is a system that would allow the user to create new features or constraints in the design, not just to modify existing features. There have been many systems in the past that have integrated VR and CAD for surface modeling. Although surface modeling CAD systems are rarely used for mechanical design, the interaction method of previous integrated VR/CAD systems can be studied.

Sculpting of virtual clay is a kind of free-form surface modeling [13]. To sculpt the clay, some type of input device is needed to touch and reshape the clay. A common type of interaction device that is used is a glove with tracking [5]. This gives the user a natural feel while interacting with the clay, but does not provide tactile feedback. Another common interaction method is using a wand with a tracker as a “one fingered data glove” [12]. This method would not have the same natural feel as a data glove but would have buttons directly on the sculpting device for better interaction. A surface modeling program could also allow the user to create their own virtual sculpting tools by shaping them in VR and adding them to a tool chest for re-use [13].

Menus are a vital component of every modeling system [5]. There seems to be a trend in using floating pop-up menus and submenus in the virtual environment (VE). A program called HoloSketch uses pop-up menus in the user’s field of view to change the drawing mode and drawing attributes. The menus pop up when the user presses a button on a wand. The program uses submenus as well for information hiding. When a submenu is selected, the main menu drops back and another menu is drawn in front of it [12].

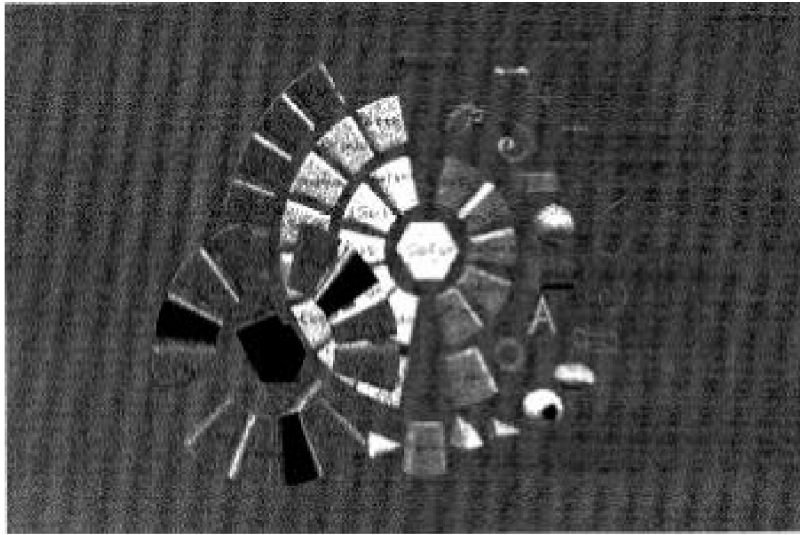


Figure 3. Photo of HoloSketch's menu system [12]

Another program called 3DVIS uses gestures and events to activate and select from various virtual menus. The menus system uses submenus that pop up off an original palette that can be closed by squeezing them with the glove-enabled hand [5].



Figure 4. Photo of 3DVIS's virtual file menu [5]

An application developed by Infiscape uses a tablet or Desktop PC to run the modeling program, 3D Studio Max, while simultaneously viewing the model in a virtual

environment. A tablet PC running the modeling program can be carried into a CAVE, permitting the user to utilize VR during the design creation [26].

Another integrated system is currently being developed by a partnership project with Fakespace and Dassault Systems. The system uses the MCAD software called Catia with multi-pipe rendering to display the 3D models in stereo on a projection system. The CAD system permits input data from tracking devices to change the view frustums on each graphics channel. The primary interaction method of this system is a keyboard, monitor and mouse on a desktop workstation. The workstation would be running Catia while the VR system would be displaying the model in a VE [9].

2.1.2 Interaction Methods of Integrated Systems without Design Creation

There has been very little research performed on integrated VR/CAD systems that only modify existing constraints in a mechanical design model. A system developed by ZGDV uses designs created with the CAD software Mechanical Desktop. The models are tessellated and updated in the virtual environment every time a constraint is modified. The user interacts with the application through a personal interaction panel (PIP) and pen [15]. The PIP is a Plexiglas panel with a tracker attached to give the user a “clipboard feel” of interaction. A graphical user interface (GUI) is drawn on top of the PIP at all times. The user then uses the pen to make selections on the GUI. The PIP and pen provide the user with force feedback when the user presses the pen against the panel.

2.2 Architecture of Prior Integrated CAD/VR Systems

There are several methods to integrate CAD and VR. Before choosing a method, the level of integration must be decided. Figure 5 below is a diagram of four different levels of integration.

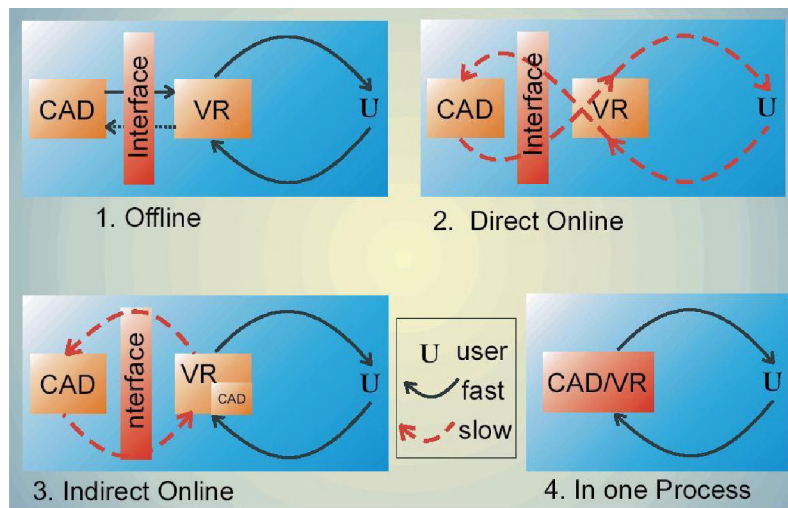


Figure 5. Levels of integration [15]

1. To use an offline system, the user would begin by exporting the CAD geometry into a format more suitable for VR. The user then starts the VR system, and loads the exported CAD geometry. Since the geometry has been translated to a VR friendly format, no changes can be made VR environment. The user must make changes to the design by using the CAD system and re-exporting the geometry to re-load the models into the VR system.
2. A direct online system uses an application programming interface (API) to connect the VR and CAD systems. The user's commands are passed directly to the CAD system, and the VR system automatically updates the modified design.
3. An indirect online system is similar to a direct online system. The difference is that the VR system provides an immediate but inexact presentation of the design changes while the CAD system processes the exact result.

4. “In one process” systems do not use API’s. The VR system is fully integrated into the CAD system. The VR system can directly access the parametric data of the CAD model [15].

To integrate a CAD and VR system, a coupling point must be determined. A CAD system and a VR system both have their own graphics pipeline. The two pipelines must be connected or coupled to integrate the two. An integrated system’s pipeline will begin with the CAD system’s pipeline and then cross-over at the coupling point to the VR system’s pipeline. Figure 6 below displays levels of coupling a CAD and VR system.

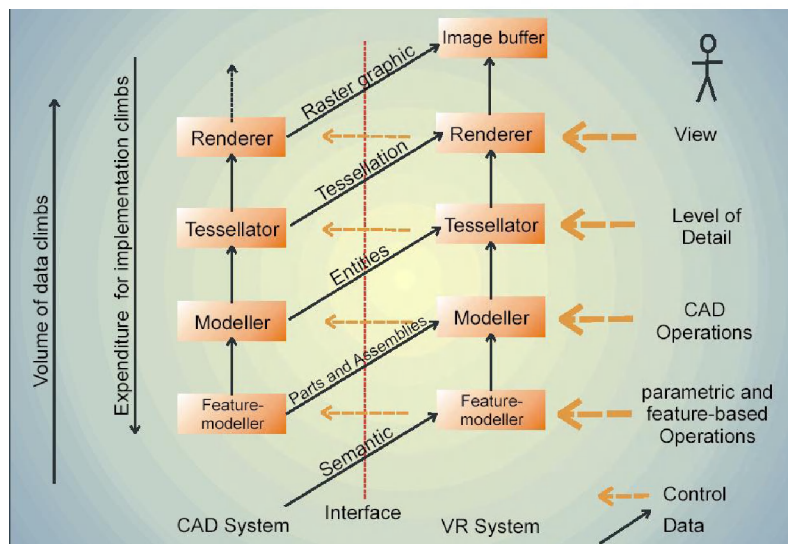


Figure 6. Levels of coupling [15]

2.2.1 Architecture of Prior Systems with Design Creation

An integrated system with design creation is a system that would allow the user to create new features or constraints in the design, not just modify existing features. Most all systems that allow design creation are completely integrated, meaning they would be at type 4 integration as shown in Figure 6 above. A completely integrated system would use the VR system to create and handle the parametric data. Displaying parametric data on a multi-screen system in real-time would require high performance computing [15]. Large models

and assemblies would slow the system down, making it difficult or impossible to interact with.

In order to fully integrate the parametric data from the VR system with the parametric data created by the commercial desktop CAD system, the commercial desktop CAD system would need to be altered to run in VR or vice versa. Currently, there are no commercial CAD systems that will operate on both desktop workstations and on VR systems. Dassault Systemes and Fakespace Systems are currently working collaboratively to make it possible to run the commercial CAD system, Catia, in VR. This would fully integrate CAD and VR as well as the parametric data created between the two.

Since a fully integrated system creates and handles its own parametric data, the models would not be compatible with other CAD systems. The data would have to be exported to a neutral file format such as IGES or STEP. These exported models would be able to be viewed in other CAD systems, but they would generally not be able to be modified. This obstacle would make such a system unable to meet the criteria in chapter 1. Almost all CAD work is currently done on a desktop workstation with a commercial CAD system. Newly created neutral data that was created with the VR system would not fully integrate with the existing commercial CAD data.

Once a CAD system and VR system are fully integrated there is still the challenge of developing a UI to operate such a complex system in a VE. Using the same UI in the VR system as the UI on the desktop system would be extremely difficult for reasons that will be later discussed. The UI of the system could be so complex that it could limit the creation of many designs. The question then arises if design creation is even necessary in an integrated system.

Most every integrated system that is currently capable of design creation uses primitive shapes to create designs such as boxes, cones, spheres, and NURBS surfaces. In some systems, such as HoloSketch and 3DVIS, the user could import a model to use as a

primitive shape [5, 12]. The primitive shapes can then be modified by scaling, rotating, and translating. These simple methods for shape creation are not sufficient for many engineering applications. These systems would also not be able to be effectively used on a standard desktop workstation, so the problem again arises that the parametric data would only be able to be handled in the VR system.

Some integrated systems use ACIS geometry kernel to have full solid modeling capabilities like current commercial CAD software [25]. Many commercial CAD software systems, such as AutoDesk Inventor and Mechanical Desktop use the ACIS geometry kernel in their system. This means that the integrated system would have the capabilities as many commercial systems. The VR system would again be handling the parametric data and would run into performance issues. The VR system would also not be able to share its parametric data with other CAD systems.

2.2.2 Architecture of Prior Systems without Design Creation

There has been very little research performed on integrated VR/CAD systems that modify existing constraints in a mechanical design model. An integrated system that modifies existing constraints means that the design creation is done on a desktop workstation with a commercial CAD system, but can be modified in VR. For this system to be effective, the data modified by the VR system would need to remain compatible with the commercial CAD system as discussed in chapter 1.

A system that modifies constraints could use any of the four types of integration shown in Figure 6. If the system was using type 4 integration it would also have the capability to create designs, not just modify the existing constraints. The system that is currently being developed by Dassault Systemes and Fakespace Systems that was discussed in section 2.2.1 is capable of having the same functionality as Catia [9]. If the system had the same functionality as Catia, the UI could be too difficult to use or the design review could take too long. The UI could be modified to limit the user to only modifying constraints so

that it would be less complex and take less time in a design review.

A system developed at ZGDV used Mechanical Desktop along with an API to communicate with their VR system [15]. This type of system would be type 2 or 3 integration as shown in Figure 6. A PIP and pen was used to interact with Mechanical Desktop while using the VR system. The system's capability depends on the ability of the API and also on how much the user is allowed to do with the UI; however, it appears that the user is allowed to modify constraints in this system.

A virtual assembly design system called VADE was developed at Washington State University [23]. This system uses assemblies created by Pro/Engineer to create "swept volumes" that represent that space required to assemble parts. The system also allows the user to modify parameters in Pro/Engineer resulting in a modified design. The VR system loads the modified design automatically making it an integrated VR/CAD system. An API to Pro/Engineer is used to for this functionality, so this would be type 2 integration. The system is limited to modifying parameters, so the user only has access to certain constraints. Since the primary focus of VADE is assembly simulation, the system requires that parameters affecting assembly operations be established a priori.

3 DESIGNING A USER INTERFACE FOR AN INTEGRATED VR/CAD SYSTEM

3.1 Design Criteria

During a design review, the interface of the integrated VR/CAD system must allow the user to make design changes. Below is a list of the criteria the interaction method needs to satisfy.

- Allow modifications of both part and assembly constraints
- Allow users to input exact numerical constraint values
- Allow the user to modify constraints while involved in a design discussion
- Allow the user to make modifications quickly
- Allow the user to stand
- Allow the user to navigate throughout the virtual world
- Be able to be used by no more than one person
- Be quick to learn
- Be ergonomically sound

3.2 VR Interaction Methods

3.2.1 Interaction Methods with the Wand and Tracker

A wand is an interaction device that generally resembles a remote control for a television. There are several buttons on the wand for input from the user along with a tracker that sends the world coordinates of the wand's position and orientation to the application.



Figure 7. Photo of wand input device

The combined wand and tracker system is used in many VR applications, and can be applied in a variety of ways. Navigation around a virtual world can be accomplished by pointing the wand in the direction the user desires to travel. Buttons on the wand can be used for accelerating and decelerating, switching modes, picking up objects, etc.

The wand can also be used as a finger in the VE, allowing the user to interact with virtual menus and virtual assemblies.

3.2.2 Interaction Methods with the Gloves and Tracker

Gloves allow applications to track the position of a user's hand in a virtual environment. There are two basic types of gloves commonly used in VR, pinch gloves and CyberGloves.

Pinch gloves do not track the position of the fingers on the hand. They generally use one tracker to track the position of the hand as a whole. Sensors in the fingertip of the glove sense when the fingertip touches another part of the user's hand. These devices are very practical for gesture recognition, moving virtual objects, or selecting items off a menu.



Figure 8. Photo of pinch gloves [11]

CyberGloves are another type of glove used in VEs. These gloves track joint locations as well as hand locations. This increases the interactivity of the application by knowing exactly where each finger is, in the VE [8]. Like the pinch gloves, these devices are also very practical for gesture recognition, moving virtual objects, or selecting items from a menu.



Figure 9. Photo of CyberGloves [8]

3.2.3 Interaction Methods with the Microphone

A microphone also makes an effective interaction device. With current speech recognition technology, it takes a short time to teach the system to recognize words

proficiently [16]. This interaction method would be useful for giving commands, selecting items off of a menu, and for inputting exact numerical values in a VE.

3.2.4 Interaction Methods with the PIP and Pen

The personal interaction panel (PIP) and pen is an interaction device that uses an electromagnetic tracker to track the exact location of the panel and the pen in the VE. The PIP is often a Plexiglas panel used to give the user a natural feel like he/she is holding a clipboard or notebook. The VR system uses the tracking data of the PIP to display a virtual GUI on the PIP. The pen is then used to select items on the virtual GUI. The PIP and pen allows the user to have more control of the virtual menu placement and also gives the user a natural feel, as if he/she is actually writing on a notebook. The PIP and pen combination also provides tactile feedback when the user has selected an item.



Figure 10. PIP and Pen with Barco Bench [15]

3.2.5 Interaction Methods with the PHANToM

The PHANToM haptic device is essentially a 3D mouse that provides force feedback to the user. It uses mechanical linkages to give extremely accurate tracking data with little delay. The largest PHANTom available allows a workspace of approximately 900 W x 900

H x 300 D (in millimeters) [24]. This permits the user to move at a 1:1 scale on a standard monitor, but not for large projection systems. The PHANToM could be used in any applications that require selecting, moving, or feeling virtual objects.



Figure 11. Models of the PHANToM haptic device [28]

3.2.6 Interaction Methods with the PDA

The personal digital assistant (PDA) is a lightweight, hand-held, usually pen-based computer. A PDA can be used with wireless technology to interact with a VE utilizing 2D GUIs. Tracking is not necessary with this device but could be utilized for additional interaction. If it were tracked, the PDA could be used to navigate through the VE in ways similar to the wand. Valuable screen real estate would be preserved when using a PDA to display the menus rather than in the large projection system of the VE.



Figure 12. Personal digital assistant (PDA) [17]

3.2.7 Interaction Methods with the Tablet PC

Using a tablet PC with wireless technology for interaction with a VE would be very similar to that of a PDA. A tablet PC would be carried like a clipboard on the user's arm while a PDA is held in the palm of the user's hand. The tablet PC would also utilize 2D GUIs like the PDA and preserve valuable screen real estate in the VE. The tablet PC could also be tracked for additional interaction such as navigation.



Figure 13. Tablet PC [27]

3.2.8 Interaction Methods with the Desktop Workstation

A standard desktop workstation could also be used to interact with a VE. The user could put the workstation in or near the VE and use the standard keyboard, mouse and monitor for interaction. This set up could be feasible with or without wireless networking. This method of interaction could be useful for applications that are VR independent or dependent. An example would be using a CAD system that would run on a desktop workstation the majority of the time but then be used in VR for a design review.

3.3 Optimizing an Interaction Method for an Integrated VR/CAD System

When designing an interaction method for an integrated system, we will have to “mix and match” all available methods to optimize the usability of the system. There are many interaction devices that can be used with many different user interface concepts. Many devices can be combined together with other devices to form a new method. Below is an array of criteria, user interfaces, and interaction devices developed for finding an optimal interaction method for the integrated VR/CAD system.

Criteria	User Interface Concept	Interaction Device
Allow modifications of both part and assembly constraints	Virtual Menus	Wand
Allow users to input exact numerical constraint values	Gesture Recognition	Gloves
Allow the user to modify constraints while involved in a discussion	Speech Recognition	Microphone
Allow the user to make modifications quickly	Touching Virtual World	PIP and Pen
Allow the user to stand	2D GUI	PHANToM

Allow the user to navigate throughout the virtual world	Physical Buttons	PDA
Be able to be used by no more than one person		Tablet PC
Be quick to learn		Desktop Work Station
Be ergonomically sound		

Table 1. Table illustrating criteria, UI concepts, and interaction devices that will be used to optimize the interaction of an integrated VR/CAD system

While optimizing the usability of the integrated VR/CAD system the design criteria have to be considered. The major function of this application is to modify constraints of the model by inputting exact numerical values. Methods based on the idea of reshaping the part by touching it would not be suitable for this system. There are generally too many parts and far too many constraints to be able to select an individual constraint in the model. Some parts and constraints could even be hidden behind other parts, making it difficult for the user to select them. Also during a design review there will be several other people in the VE making it difficult to reach out and touch virtual object if other people are standing in the user's way. Another problem with this concept is that it would be difficult to input exact numerical values without another interaction method.

Since touching and moving parts would not be feasible for a design review, the use of gloves as an interaction device would be limited to gesture recognition and selecting items from a virtual menu. Gesture recognition is limited by the number of commands that can be given by the user, so the user would have to memorize many different gestures. To decrease the number of gestures, the user could use virtual menus displayed in the VE. Yet, even with the menu, it would be difficult to input an exact numerical value for a constraint modification and the user would still be limited by the number gestures they can remember.

Other options that would make it easier for a user to select items off of a virtual menu would be to allow the user to pick the items with a tracked device such as a glove or wand.

Wands are often used as navigation devices in VR, so a wand could serve two functions whereas gloves would only serve one. A wand could work as well as gloves since gloves would not be used for touching the design, gesture recognition, or picking items on a menu.

If the user were to use a wand to pick items off the virtual menu, the user would still have problems inputting exact numerical values. A microphone with speech recognition would be optimal for that particular task but would be difficult to implement since the user might accidentally input values during the course of a design review discussion in the VE. However, a virtual keypad could be displayed in the VE as part of the virtual menu to input the exact numerical values.

Using a wand to select items off a virtual menu including a virtual keypad could satisfy many of the criteria but may be an exceedingly slow method of interaction and would also take up a large amount of real estate in the VE [12]. The slow item selection from a virtual menu is partially caused by Fitts' Law: menu item selection takes more time in three dimensions than in two. In order to meet up with the menu item, the wand tip has to travel farther in the VE than a mouse cursor travels on a conventional computer screen [12]. Also, one would have to move the floating menu frequently so that the design does not get covered up by the menu.

Many of these problems could be solved by using a PIP and pen. This device would give the user complete control of the virtual menu placement and also would have an increased speed of interaction due to the familiar feel of holding and touching a clipboard. Although the PIP and pen appear to satisfy most of the criteria, other problems arise while using the device. For instance, the menu size would have to be reduced so that it would fit on the PIP. Also the limited resolution on a large projection system restricts the size of the menu items; only relatively large items show up clearly. This would result in many more submenus, making navigation through the menus more difficult for the user. Another problem with this interaction method is that it is entirely dependent on tracking. Current

tracking technology is relatively inaccurate and is also subject to latency. These problems make the selection process cumbersome and slow.

A PHANTonM interaction device is extremely accurate in tracking and could be used as a 3D mouse for managing virtual menus. This device does not solve the problems of low screen resolution and the virtual menus occupying valuable real estate. The PHANTonM is also large and unwieldy within the limited space of a virtual environment occupied by other members of the design review team.

Using a PDA for displaying and interacting with a GUI could solve many of the problems that arise in using a menu system in the VE. However, a PDA has a relatively small visual display and low processing capabilities so a tablet PC could be even more effective. A tablet PC has sufficiently high screen resolution to display a GUI without clarity problems and it would also let the user manipulate a GUI in a 2D manner, which would result in much faster menu selection.

Utilizing a tablet PC for an interaction device will allow the user to easily select constraints, modify the constraints with exact numerical values with onscreen keypad, participate in the group discussion, be in the standing position, and operate the application alone. A GUI could be designed on a tablet PC that would be intuitive to any computer user and easy to learn. Utilizing the tablet PC for a design review does not force the user to be in any awkward or uncomfortable positions, making it relatively ergonomically sound, especially if the user is not required to hold it for extended periods of time.

Although the tablet PC allows the user to navigate throughout the VE, an effective method of integrating navigation with the tablet device is necessary. With a GUI to control the speed, the tablet PC can be used to navigate by attaching a tracker to it and pointing the tablet in the direction of motion. Waving the tablet around to navigate while using the display to control the speed would be awkward and inconvenient. Another option would be to create a navigation GUI on the tablet and not use tracking to navigate. This solution would

eliminate the problem of waving the tablet around, but navigating with a 2D GUI in a 3D environment may be difficult. A third option would be to use a wand with tracking to navigate. This solution is simple, and the wand is a standard tool for virtual navigation. Another advantage of using the wand is that it could be easily used as a laser pointer to point at features of the design. If the wand were given a belt clip or hook it could be easily placed on the user's belt when not in use and solve the problem of holding onto three items (the tablet, tablet pen and wand) simultaneously.

Utilizing a desktop workstation as the interaction device would be difficult. A workstation is cumbersome and would not fit well in the limited space of a VE full of users. Because the operator would have to remain at the station, it would be more difficult for him/her to move around the display and participate in the discussion. The tablet PC with a wand appears to be the most effective interaction method.

3.4 Designing a User Interface for a Tablet PC

The tablet PC with a GUI was a large part of the solution to optimizing a user interface for an integrated VR/CAD system. This interaction method has the capability to satisfy all of the criteria, but a GUI design concept is needed to assure all the criteria are met. Any application is limited by the effectiveness of the GUI design [10].

While creating a GUI design concept, it is vital that we consider the intended user. Mechanical engineers will be the most likely user of the UI since they are usually the creators of the mechanical design. Manufacturing engineers and managers will also be involved in the design review which could result with one of them in control of the application as well. It is assumed that mechanical engineers are proficient using a CAD system on a desktop workstation and also competent using Windows applications.

Interface design is constrained by conservative forces, such as the user's expectations created from existing systems and the existence of formal and informal standards [10]. To make the GUI for the tablet PC intuitive and decrease the learning curve, it should have a

design similar to GUIs from CAD systems and other Windows applications.

In a mechanical design there are usually many constraints. If all of them were displayed on the screen simultaneously, the user would have difficulty finding which constraint to modify. The number of constraints could be reduced by displaying only one type of constraint at an instance. There are two types of constraints in most contemporary CAD modeling systems, part constraints and assembly constraints. Displaying only one type of constraint at any one time would still be slow and make it difficult for the user to determine which particular constraint to modify. Constraints can be categorized even further by separating the constraints associated with each part. A user could select both the part and the type of constraint, limiting the number of constraints the application would need to display.

All CAD systems must possess a graphics window to permit users to view and interact with the design. Designers can interact with the design by picking or selecting items and features in the graphics window with the mouse. To stay consistent with existing CAD systems, the GUI on the tablet PC should also contain a graphics window. The user could then easily select the part associated with the constraint that needs to be modified. Utilizing a graphical representation of the assembly would be faster and easier than searching through text references to each part. Additionally, the user would also not be required to remember the names of all the parts in the assembly. Although this method of selecting parts is very intuitive to the user, problems begin to arise when the complexity and size of the design increases. It would be difficult and tedious to select very small parts, and it would be not feasible to select hidden parts in the middle of the assembly.

Since utilizing a graphics window will not allow the user to easily select every part, the GUI will also need to display written text labeling every part. As discussed previously, assemblies could contain many parts, and it would be hard for the user to remember the name of each part and associate the name to the visual representation. Categorization can again be

applied to decrease the number of parts displayed at an instance. CAD models are often comprised of several subassemblies. For example, a car could have a subassembly for the wheel which in turn could have another subassembly for the brakes. The number of parts displayed could then be reduced by using a tree to categorize the parts in subassemblies.

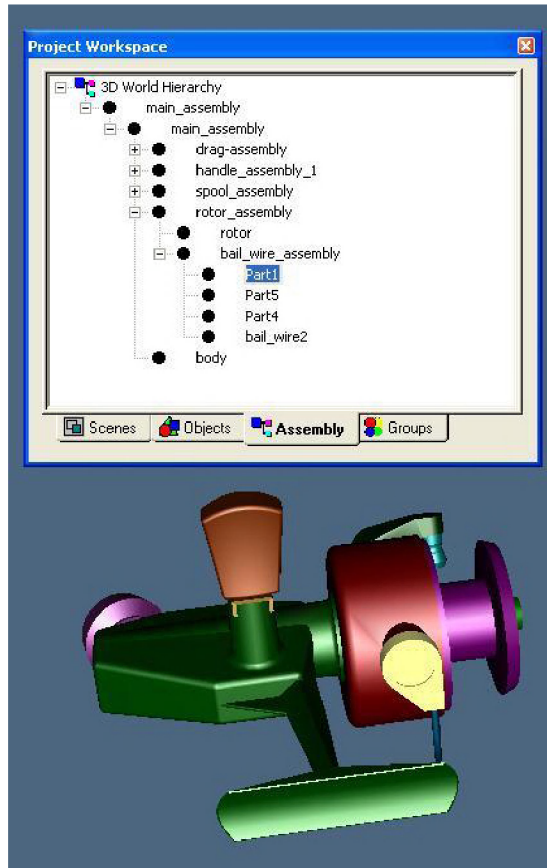


Figure 14. Screenshot of an assembly tree in UGS's VisConcept

Through the use of categorization, the user should be able to easily select which constraint is to be modified by using a tree to find the part associated with the constraint and choosing the type of the constraint to be modified. Toggle buttons should also be placed on the GUI to toggle through each part in case the user can not remember where the part is in the tree. The graphics window also acts as a shortcut for the quick selection of visible parts.

Once the constraint is selected for modification, the user will then need to input an exact numerical value to replace the constraint's existing value. A simple text box with an

onscreen keypad could easily be implemented for this task. After a value is entered, the system would need to know to update the model and check to see if the constraint value is feasible with existing constraints in place. An update button could carry out this task, but we would need an additional feature to check if the value is permitted by the other constraints. Status bars are often used in CAD systems and Windows applications to display similar information. A status bar could be added to display feedback from the system such as invalid constraint values.

The last major feature of the GUI is to add navigation throughout the virtual scene in the graphics window. Buttons for pan, zoom and rotate should be added to change navigation modes. This appears to be a de facto standard among CAD systems. The user would then drag the tablet pen across the window to move the virtual world that is displayed on the tablet.

4 ARCHITECTURE OVERVIEW

4.1 Integration Method

To design the architecture of the integrated VR/CAD system, the level and type of integration were considered. Sections 4.1.1 and 4.1.2 discuss how each integration and vertical coupling level meets the design criteria for the integrated VR/CAD system and will discuss the method used for the system that was developed. Refer back to section 2.2 while reading this section.

4.1.1 Integration Level

In section 2.2 there were four levels of integrating CAD and VR discussed. This section discusses which type was chosen and why. Figure 5 in section 2.2 may provide a helpful reference while reading this section.

An offline system is the simplest and easiest way to integrate VR and CAD. This method is commonly used and is very reliable. Table 2 summarizes the capabilities of an offline system relative to the criteria that were developed for a design review in section 1.2.

Criterion	Yes	No
Can modify the design	X	
Can be used in parallel with a commercial CAD system	X	
Can operate reliably	X	
Can allow the user to navigate in the virtual world	X	
Can allow exact numerical constraint values to be input	X	
Can modify constraints while involved in a design discussion		X
Can make modifications quickly		X
Can allow the user to stand	X	
Can be used by no more than one person	X	
Can be quickly learned to use		X
Can be ergonomically sound	X	

Table 2. Offline system capabilities vs. design review criteria

Clearly, an offline system is not capable of meeting all of the criteria. The user would not be able to carry on a discussion about the design since he/she will be required to restart the system when making a design change, so the design would not be continuously displayed. Making modifications would be extremely slow since the user would be switching back and forth between the CAD system and the VR system. In addition, the complex UI of the CAD system would make the integrated system difficult to learn since the CAD system's UI would drive the system. The user would have to know how to operate the CAD system in order to make modifications.

A direct online system utilizes an API to communicate between the CAD system and VR system. The CAD system can still be utilized independently of the VR system, ensuring that the parametric data remains compatible with both applications. The capabilities of a direct online system relative to the design review criteria are shown in Table 3.

Criterion	Yes	No
Can modify the design	X	
Can be used in parallel with a commercial CAD system	X	
Can operate reliably	X	
Can allow the user to navigate in the virtual world	X	
Can allow exact numerical constraint values to be input	X	
Can modify constraints while involved in a design discussion	X	
Can make modifications quickly	X	
Can allow the user to stand	X	
Can be used by no more than one person	X	
Can be quickly learned to use	X	
Can be ergonomically sound	X	

Table 3. Direct online system capabilities vs. design review criteria

A direct online system has the potential to meet all of the criteria. The design can be easily modified by utilizing an API, but making the modifications quickly would be more challenging. The modified CAD models would have to be translated into a VR-capable

format each time the design is changed. This process could be relatively slow depending on the hardware used and the size and complexity of the model. A direct online system would also be difficult to operate reliably. The system must be able to handle errors generated by the CAD system. A large commercial CAD system typically generates many types of errors and it would be extremely difficult for a direct online system to handle these errors through an API.

An indirect online system and a direct online system would operate very similarly. An indirect online system also utilizes an API to communicate between the CAD system and VR system. The CAD system can be utilized independently of the VR system, ensuring that the parametric data remains compatible with both applications. The major difference between an indirect online system and a direct online system is that an indirect online system will make the modifications to geometry displayed in the VR system as an approximation while the CAD system is making the modifications to the parametric data. The geometry displayed by the VR system will immediately display the modifications as an approximation and then load the geometry with the exact changes after the CAD system has computed the changes. This hybrid system allows the user to see the results from a design modification more quickly. The capabilities of an indirect online system relative to the criteria developed for a design review in section 1.2 are summarized in Table 4 below.

Criterion	Yes	No
Can modify the design	X	
Can be used in parallel with a commercial CAD system	X	
Can operate reliably	X	
Can allow the user to navigate in the virtual world	X	
Can allow exact numerical constraint values to be input	X	
Can modify constraints while involved in a design discussion	X	
Can make modifications quickly	X	
Can allow the user to stand	X	

Can be used by no more than one person	X	
Can be quickly learned to use	X	
Can be ergonomically sound	X	

Table 4. Indirect online system capabilities vs. design review criteria

A direct online system has the potential to meet all of the criteria. An indirect online system would allow the user to visualize the changes as an approximation very quickly, but the amount of time that it would take to reload the geometry created by the CAD system with the exact changes would take just as long as a direct online system. The system would also have to deal with handling errors thrown by the CAD system, as a direct online system does. The CAD system may not be able to make some of the changes requested, so the VR system would need to be intelligent enough not to make those changes as an approximation. Such a complex system could be very difficult to develop to run reliably but is possible.

An “In one process” system or fully integrated system acts as the VR system and the CAD system. The parametric data would be handled by this system and would be displayed in VR. The capabilities of a fully integrated system relative to the criteria that were developed for a design review in section 1.2 are summarized in Table 5.

Criterion	Yes	No
Can modify the design	X	
Can be used in parallel with a commercial CAD system		X
Can operate reliably	X	
Can allow the user to navigate in the virtual world		X
Can allow exact numerical constraint values to be input	X	
Can modify constraints while involved in a design discussion	X	
Can make modifications quickly	X	
Can allow the user to stand	X	
Can be used by no more than one person	X	
Can be quickly learned to use	X	
Can be ergonomically sound	X	

Table 5. Fully integrated system capabilities vs. design review criteria

A fully integrated system is not capable of meeting all of the criteria. A fully integrated system can not be used in parallel with any of today's commercial CAD systems, since it handles its own parametric data. It is very important that the system can be used in parallel since the design creation will all be done with commercial CAD systems. The system would also not be capable of allowing users to navigate in the virtual world. Displaying sizable amounts of parametric data in a VR system would slow the frame rate of the VR system, so that interaction and navigation would not be in real-time. Real-time interaction is a very important aspect of VR and it would not be acceptable to be slower than real-time.

For this project, a direct online system will be used to integrate a CAD and VR system as shown in Figure 15 below. A direct online system is the most capable of meeting the required criteria without undue complexity. The direct online system was chosen over an indirect online system to increase reliability and to decrease complexity.

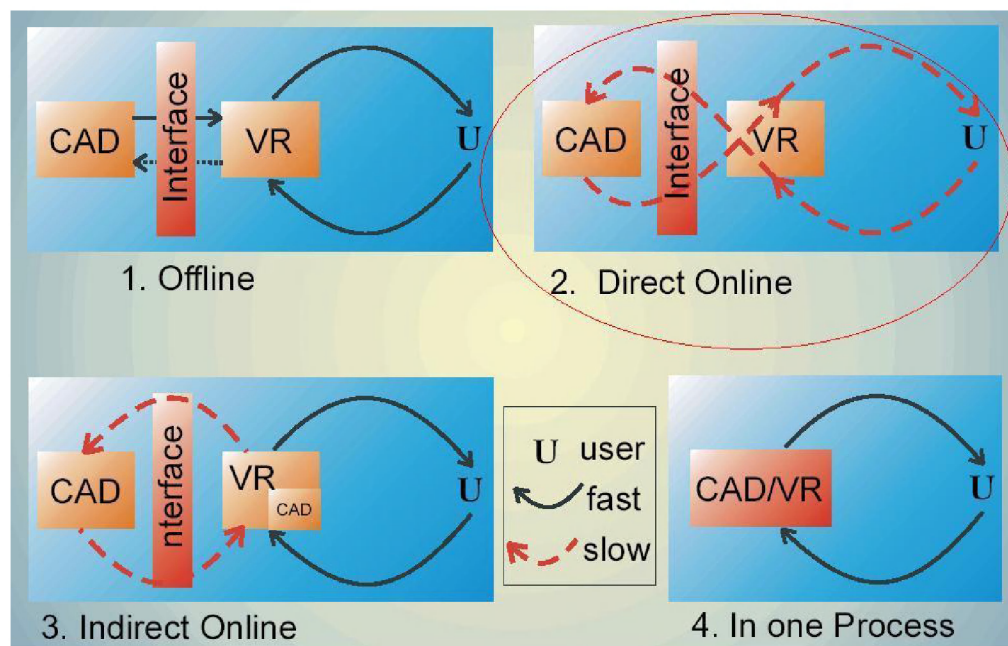


Figure 15. Level of integration that will be used for this project [15]

4.1.2 Vertical Coupling

A CAD system and a VR System both have their own vertical graphics pipeline. Figure 6 in section 2.2 illustrates the two pipelines. The level at which one pipeline is cut to transfer the data to the other system determines the level of coupling of a CAD and VR system. This section discusses which level of coupling was chosen and why it was chosen.

The first level of coupling would be to transfer semantic data from the CAD system to the VR system at the feature level. This would essentially be transferring over the list of features associated with each model along with the required references to the VR system. The VR system would then generate the parametric model with this information. The VR system still handles the parametric data and would thus slow the system down. The VR system would also require a feature modeler that would be complex and difficult to make compatible with the CAD system's feature modeler.

The second level of coupling would be to transfer the parts and assemblies to the VR system in a neutral file format such as VRML or IGES [30]. The VR system would then contain a modeler or loader to load the models and tessellate them. When choosing the neutral file format, it would be preferred to use a format in which the geometry is already tessellated, such as VRML. This would simplify the tessellation process in the VR system and would increase performance. Some file formats such as IGES contain parametric information about the model. The VR system would have to be capable of handling the parametric information and would also be required to tessellate the models. This would result in a more complex VR system with reduced performance.

The third level of coupling would be to transfer geometric primitives such as circles, lines, or boundary representations to the VR system. This method could potentially decrease the amount of data transferred between the two systems since entire parts and assemblies would not be transferred. Only the entities necessary to complete the design change would

need to be transferred. Design changes could be relatively quick; however, the VR system would be required to handle parametric data, reducing system performance.

The fourth level of coupling is to transfer tessellated data or triangles from the CAD system to the VR system. This would significantly reduce the complexity of the VR system since a feature modeler, B-rep modeler, and tessellator would not be required by the VR system. Transferring a VRML file of a part would be one method of transferring tessellated data, but the entire part would be transferred. Transferring only the tessellated data that has changed would be more efficient than transferring an entire part. The CAD system's interface would need to allow access to the tessellator to control the tessellated data in order to achieve this efficiency. Since CAD interfaces do not generally allow low-level access to the tessellator, the entire tessellated part would need to be transferred to the VR system.

The fifth level of coupling would be to transfer image data as a raster graphic to the VR system. This would also significantly reduce the complexity of the VR system. The raster graphic would go straight to the image buffer of the VR system, eliminating many processes in the pipeline. CAD systems are not capable of rendering scenes in real-time, so interaction would be difficult at this level of coupling [15]. Since interaction is very important, transferring the raster image will not be used as the coupling point.

For this project, the CAD system and VR system will be integrated by transferring tessellated parts and assemblies from the CAD system to the VR system as shown in Figure 16. This will allow the VR system to operate in real-time and will also eliminate the need for a feature modeler, B-rep modeler, and tessellator in the VR system. The tessellated data will need to be whole parts, since CAD interfaces do not allow access to the tessellator.

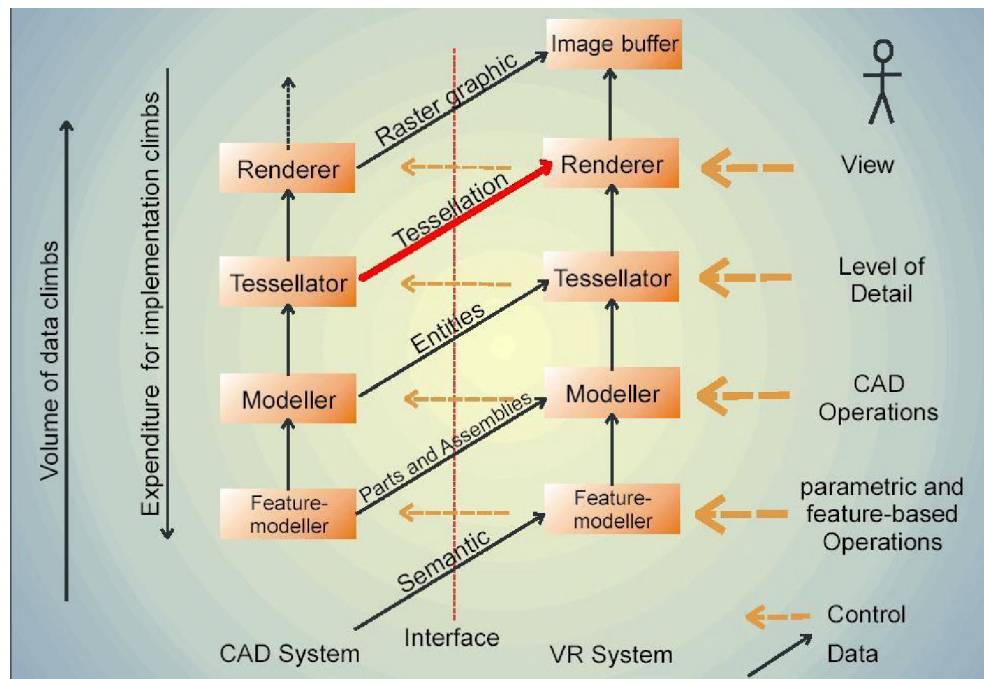


Figure 16. Level of coupling that will be used for this project [15]

4.2 Underlying Components

Several components were used to integrate the CAD system and VR system. Figure 17 illustrates the components and architecture of the integrated system. Each component is discussed in more detail in the following sections.

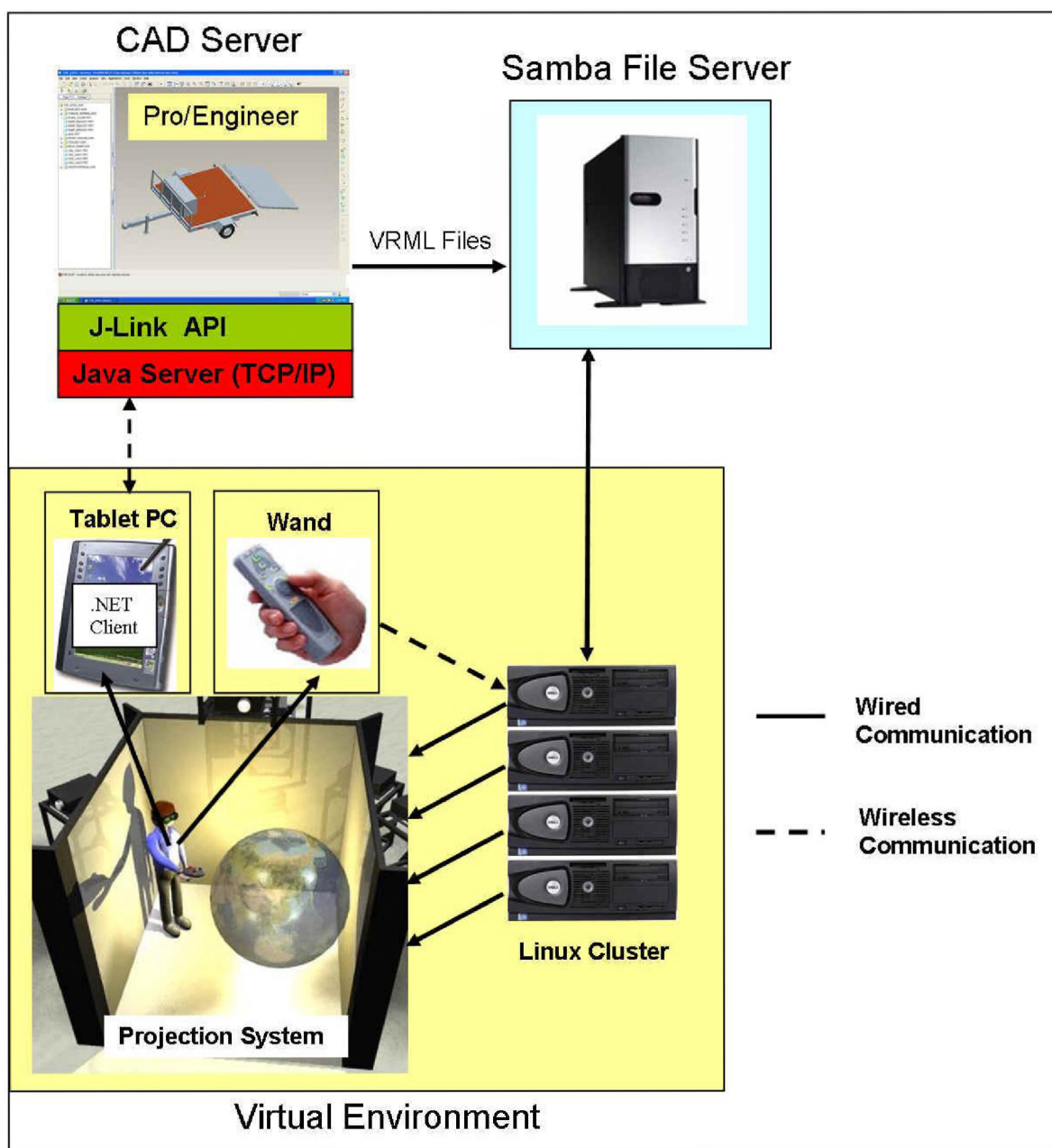


Figure 17. Schematic architecture of the integrated VR/CAD system

4.2.1 CAD Server

The CAD server is a PC running Red Hat Enterprise 4 and Pro/Engineer Wildfire 2.0. The PC runs a java program that acts as a server for other PCs to connect to remotely with a TCP/IP connection. The java program interfaces with the CAD system to transfer information through the CAD system's application programming interface (API).

4.2.1.1 CAD System

The primary component of the integrated system is the CAD system. For this project Pro/Engineer Wildfire 2.0 (Pro/E) by PTC was used. Pro/E is used on a desktop workstation and is traditionally operated with a keyboard and mouse. Pro/E uses a graphical user interface (GUI) to create parametric models and assemblies. Pro/Engineer Wildfire 2.0 can operate on Windows, Linux, HP-UX, Solaris, and Irix operating systems, although for this project the CAD server will be running Linux [21]. Figure 18 below is a screenshot of the Pro/Engineer Wildfire 2.0 GUI and is displaying a solid model of a disc brake assembly.

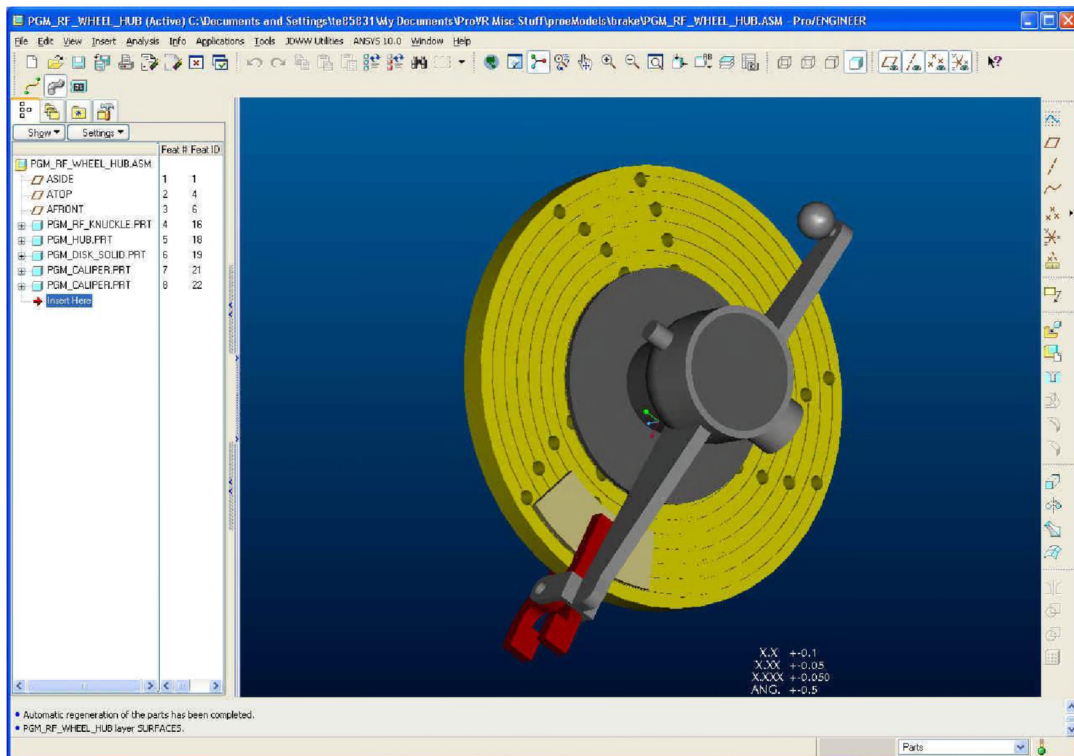


Figure 18. Screenshot of the Pro/Engineer Wildfire 2.0 GUI

4.2.1.2 CAD Interface

To allow communication between the CAD system and the VR system, an interface is required. An application programming interface that uses the java programming language called J-Link can be installed with Pro/Engineer. This powerful interface will allow

manipulation of geometric and parametric constraints externally allowing the design to be modified within the virtual environment.

4.2.1.3 Java Server

The CAD server must allow the user to make changes to the CAD geometry remotely. In order to accomplish this, a server written in the java programming language allows TCP/IP connections. The server sends and receives strings of information to the client. The server also connects to Pro/E through the J-Link API to transfer information to and from the CAD system.

4.2.2 Samba Server

For every change the user makes to the design within the VE, the CAD system exports the CAD geometry to a samba file server in the VRML format. The samba server is a file system shared by both the CAD system and the VR system. The CAD server writes a file to the samba server with the name of the top level assembly as well as the system time. The VR system reads this file and checks the system time periodically. If the system time has changed, it loads in the assembly that is written in this file. A samba server is compatible with both Linux and Windows file systems, so this allows the CAD server to run both Linux and Windows. The samba server also has several high performance hard drives that can read and write data relatively quickly which increases system performance.

4.2.3 Virtual Environment

4.2.3.1 Linux Cluster

A Linux cluster is a cluster of PCs running the Linux operating system. A Windows cluster could also have been used for this project, but a Linux cluster was used to increase system performance. The Linux cluster is used to operate the virtual environment in this application. Each node of the cluster has a graphics card that is stereo capable. Because four projectors and screens were used for this project, four nodes were used in the cluster. Each node runs the Linux operating system as well as the VR platform called VR Juggler [29].

VR Juggler is an open source VR framework that provides the tools necessary for VR application development. For this project, a VR Juggler template application that utilizes OpenSG was modified to work in collaboration with the CAD system [20]. The template application was modified so that it would periodically read an information file from the samba server every 200 frames. The file on the samba server contains the name of the VRML assembly to load along with the system time. When the system time that is listed in the file changes, the cluster loads the assembly whose name is also in the file.

4.2.3.2 Projection System

A CAVE was used for this project although many types of projection systems could be used [2]. The CAVE has three sides and a floor; it is setup similar to the projection system shown in Figure 17. The projection system uses three rear projection screens and a floor to display the graphics of the virtual world. There are four projectors to display the graphics onto these four surfaces, in stereo. The user then stands in the center of the CAVE.

4.2.3.3 Wand and Tracking System

VR Juggler also supports navigation of the virtual environment. VR Juggler is able to run with a wand and a tracking system to allow the user to navigate around the virtual environment. The user also wears a head tracker to increase interaction and make the environment more realistic. The wand and tracking system are both connected wirelessly with the Linux cluster.

4.2.3.4 Tablet PC

The tablet PC runs a client that communicates with the java server and is connected by a wireless TCP/IP connection. The client has a GUI that allows the user to make modifications to the CAD geometry. The client sends and receives information about the geometry to the server. When the client first connects to the server, it loads the entire model tree and all of the dimension names and values associated with the models. All of this information is sent to the client via text strings. The strings are coded in a way so that the

first digits of the string indicate what type of data is to follow. The user interface of the client is discussed in more detail in a later section.

5 RESULTS

5.1 System User Interface

A system architecture and user interface design was developed in chapter 3. This chapter describes the user interface that was implemented for this project. To use the integrated VR/CAD system, the user begins by starting the CAD server application. This is the java program that acts as a server and forwards the requests by the client onto the CAD system. Pro/Engineer is automatically started when the java program begins. The next step when starting up the system is to start up the VR Juggler application on the Linux cluster. This is done at the command line on one of the nodes of the cluster.

With the VR system and CAD system started, the user is ready to use the virtual environment. The user wears 3D glasses and also carries a wand and tablet PC. A tracker is attached to the user's glasses as well as the wand for interaction and navigation. To connect the client on the tablet PC to the CAD server, the user must enter the IP address of the CAD server and select the "Connect" button as shown in Figure 19.

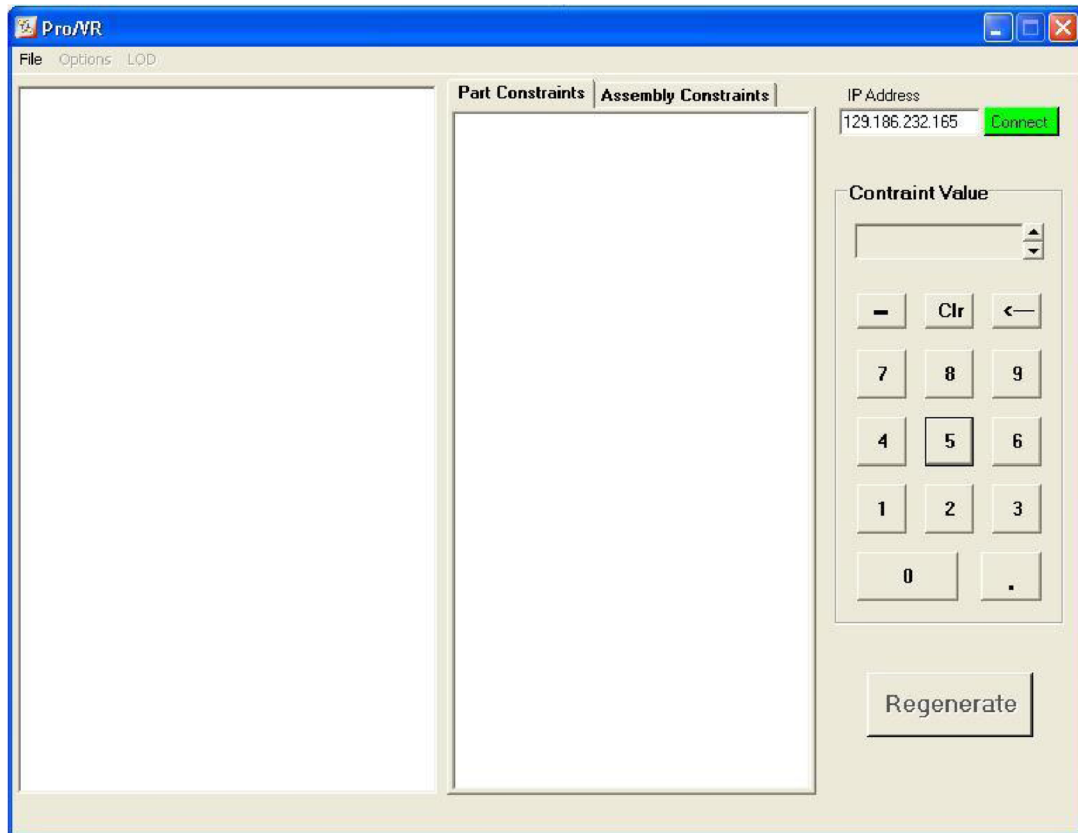


Figure 19. Tablet interface showing client connecting to the server

Notice that all other areas of the user interface are disabled until the client is connected.

Once the client has connected to the server, the user may open a Pro/E model to be evaluated. This can be done by selecting "File" and then "Open" from the pull down menu. The open file dialog will then open, so that the user can browse to the Pro/E assembly that he/she would like to open.

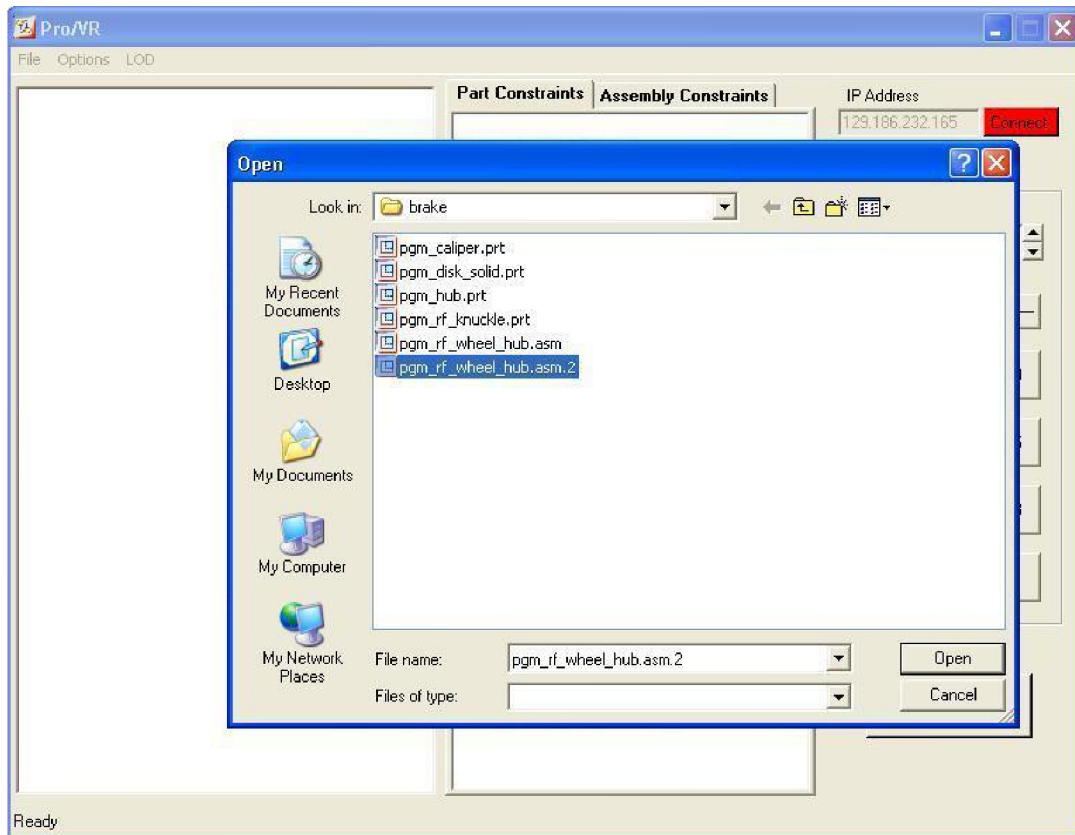


Figure 20. Tablet PC interface showing the Pro/E file open

The system is only capable of loading files that are on the samba server, so the user must ensure that all of the required Pro/E files are on the samba server before he/she begins. After the Pro/E file is selected, the CAD server loads the file into Pro/E and the client will begin populating the model tree. The VR system simultaneously loads and displays the corresponding model geometry after the CAD system has loaded the file and exported the model in the VRML format.

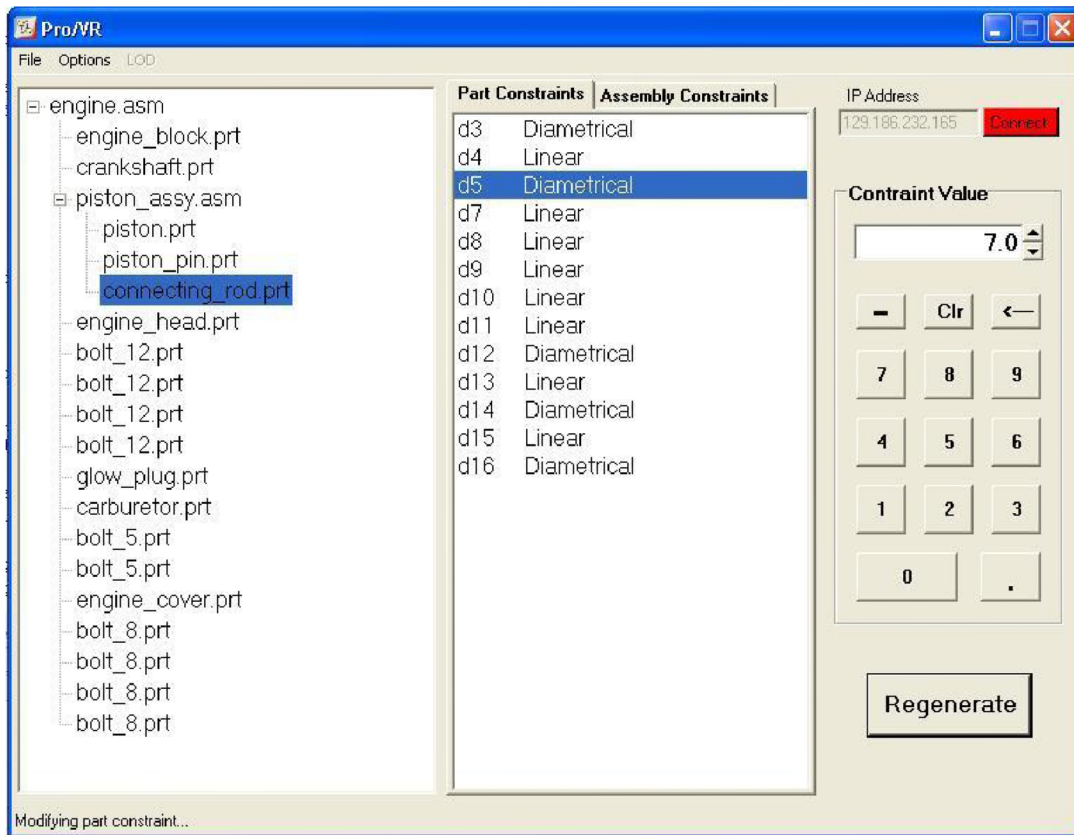


Figure 21. Tablet PC client after the assembly has loaded

The model tree and constraints are loaded into the client. Each part may have part constraints and assembly constraints. The list of constraints for the selected component is listed to the right. The constraint name and type are both listed. If the user selects a different component in the assembly, the list of constraints on the right changes accordingly. The user can also view and modify the assembly constraint associated with each component by selecting the assembly constraints tab. The user modifies the value of a constraint by entering a new value into the text box and selecting the “Regenerate” button. The “Regenerate” button triggers the CAD system to update the model with the changes made and export the modified model to the VRML format. The VR system sequentially reads in the new VRML files and displays the model in the virtual environment. This process allows the user to view the modified design in the VE immediately after the “Regenerate” button is selected.

The “Options” pull-down menu can be used for several purposes. The user can select the “Shrinkwrap” option if the geometry is complex to increase performance in the VR system. Shrinkwrap is a feature that Pro/E uses to simplify geometry. Shrinkwrapping the geometry will eliminate geometry that is not visible in the assembly, such as the internal parts of an engine. If the shrinkwrap option is selected, the user can select if he/she would like to show holes in the geometry. The “level of detail” (LOD) pull-down can also be selected if shrinkwrap is enabled. The user can select a level of detail on a scale of one to ten.

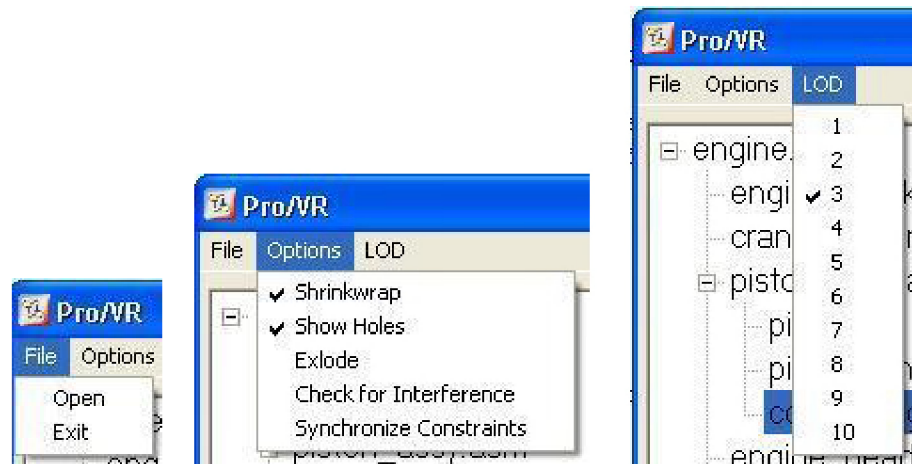


Figure 22. Pull-Down menus

Other selections in the “Options” pull-down menu include explode, check for interference, and synchronize constraints. The “Explode” option will create an exploded scene from the assembly so the parts are more visible. A check mark appears next to “Explode” when it is selected.

In order to update the geometry with shrinkwrapped geometry or an exploded view, the regenerate button must be selected. The regenerate button will change color to a dark grey if changes have been made to parameters on the client and have not been updated on the CAD system. After selecting the regenerate button, the color will turn back to a light grey.

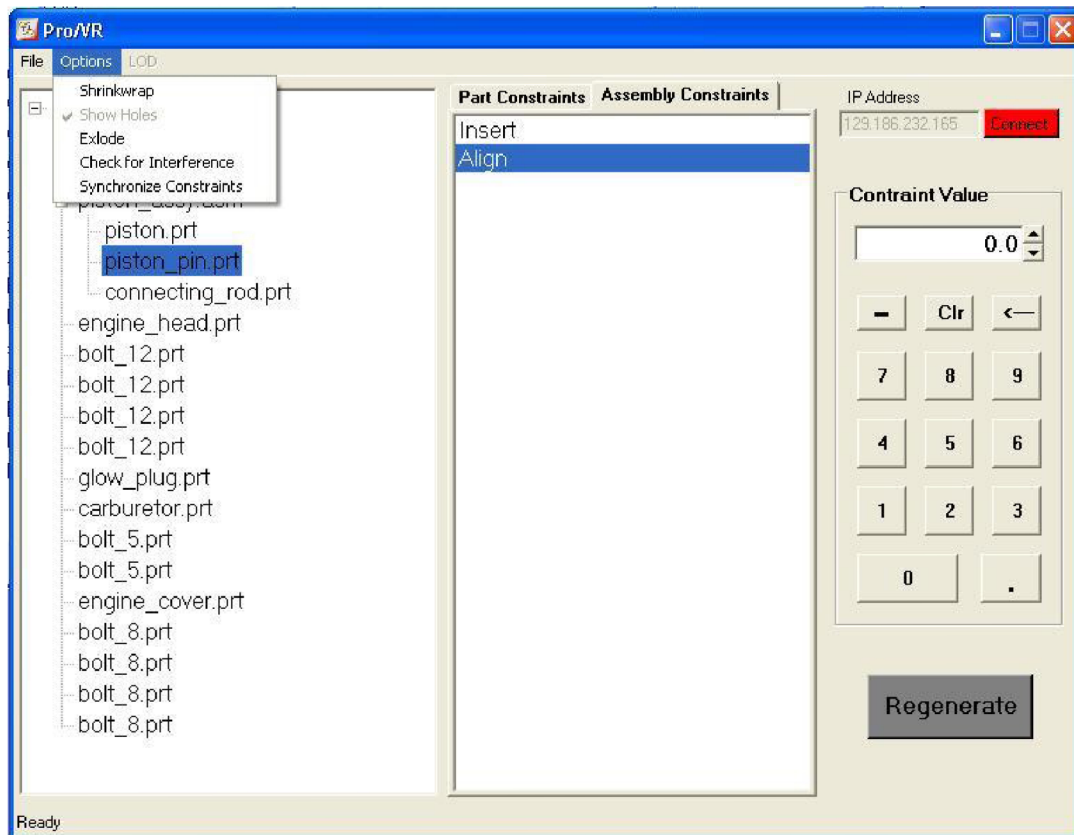


Figure 23. Tablet PC client with dark grey regenerate button

The “Check for Interference” selection in the “Options” pull-down menu can be selected to check for interference in the CAD geometry. A message box will pop-up with the number of interference found. The user can also select the “Synchronize Constraints” option which will re-load all of the constraints and values from the CAD system. This is a safety feature to ensure that the correct constraints and values are displayed on the client. The constraints stored on the client may get out of sync with constraints stored in the CAD assembly if the model is using Pro/Program or contains relations.

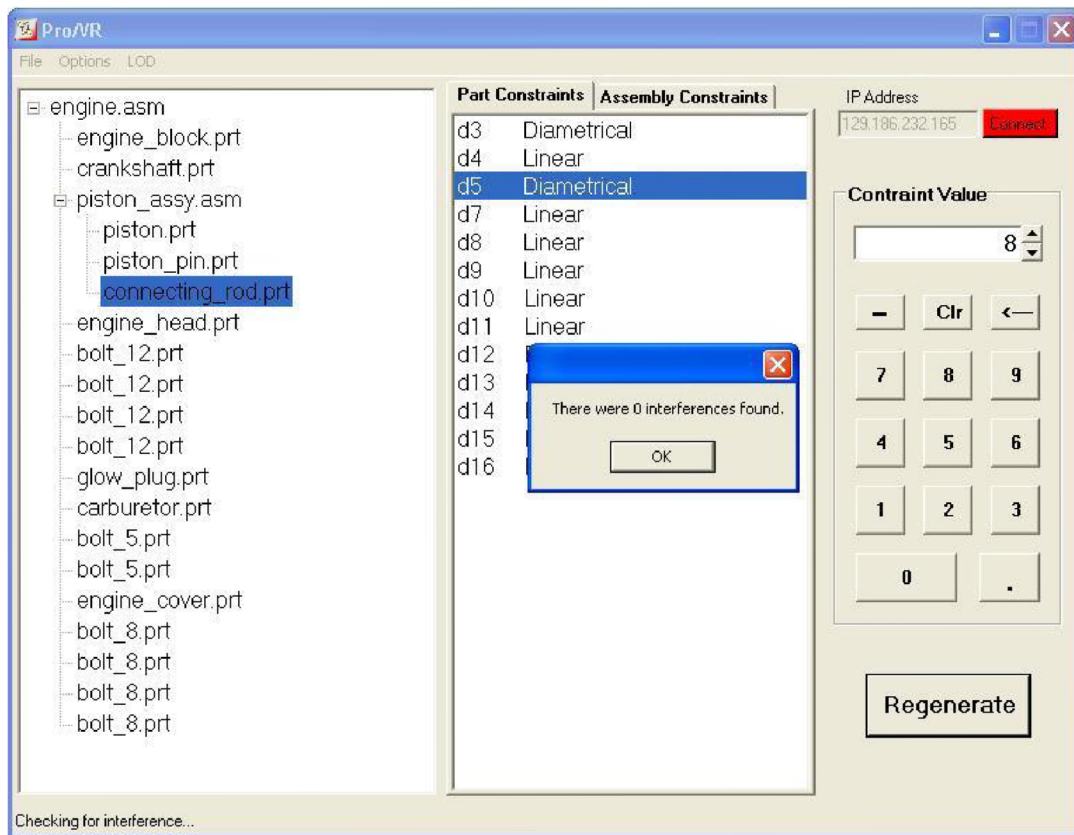


Figure 24. Illustration of the check for interference option

The client also uses a status bar at the bottom left corner of the window. This gives the user feedback on the progress of a task that was selected on the client. Some tasks may take a considerable amount of time when working with large assemblies. Figure 25 below illustrates some of the feedback that is displayed in the status bar.

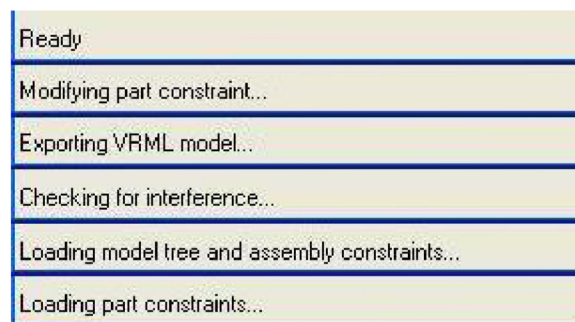


Figure 25. Illustration of some of the feedback that is displayed in the status bar

The tablet PC is used to make modifications to the geometry, and the wand with a tracker is used to navigate around the virtual world. Three buttons on the wand are needed for navigation. The first button increases the velocity by an incremental amount every time it is pressed and the second button does the opposite. Once the user has begun to fly around in the virtual world, the direction of movement is determined by the direction the wand is pointing. The user points the wand in the direction he/she wants to go. The third button is used to rotate the virtual world. A four walled projection system can not display all the virtual space without rotating the virtual world. This is done by holding down the third button and rotating the wand while the button is pressed. This would be similar to dragging and dropping a window on a desktop computer. Figure 7 in chapter 3 is a picture of a wand that is like the one used for this project.

5.2 System Performance

The hardware that is used in the integrated system has a major impact on system speed. The table below lists the hardware that was used when testing the system speed.

Linux Cluster	
Number of Nodes	4
Dell Model	Precision 470
Processors	Dual Xeon 3.0 Ghz 3 mb cache
Graphics Card	Nvidia Quadro fx4400 w/512mb
Memory	2 gb
Network Speed	GigE Intel Pro 1000
Operating System	Red Hat Enterprise 4
CAD Server	
Dell Model	Precision
Processors	Dual 2.8 ghz Xeon
Graphics Card	Nvidia Quadro fx3400 w/521mb
Memory	2 gb
Network Speed	GigE Intel Pro 1000
Operating System	Red Hat Enterprise 4
Tablet PC	

Model	Toshiba Tecra
Processor	2.0 Ghz Intel Centrino
Graphics Card	nVidia GF6600 Le
Memory	1 gb
Network Speed	Wireless G (54 mbps)
Operating System	Windows XP SP2
Samba Server	
HardDrive Interface	Scsi
HardDrive Speed	15,000 RPM
Cached Memory	2x 128mb PERC 3/DC cards
Network Speed	GigE Intel Pro 1000
Processor Speed	Dual 2.2 Xeon 512kb cache

Table 6. Hardware used for each component for testing the system speed

To test the system speed, three Pro/E assemblies were used with the integrated system. The three assemblies were of different sizes with the number of components listed in Table 7.

Model Size	Number of Components
Small	24
Medium	74
Large	264

Table 7. The number of components in the three test assemblies

All three of the assemblies contain many cast and forged parts, so their geometry was complex with many round surfaces.

The first performance measure taken was the speed of opening the model into the integrated system. The timings were taken after both the CAD server and the VR system were started. The start time was taken when the “Open” button was selected on the client, and the end time was taken when the geometry appeared in the virtual environment.

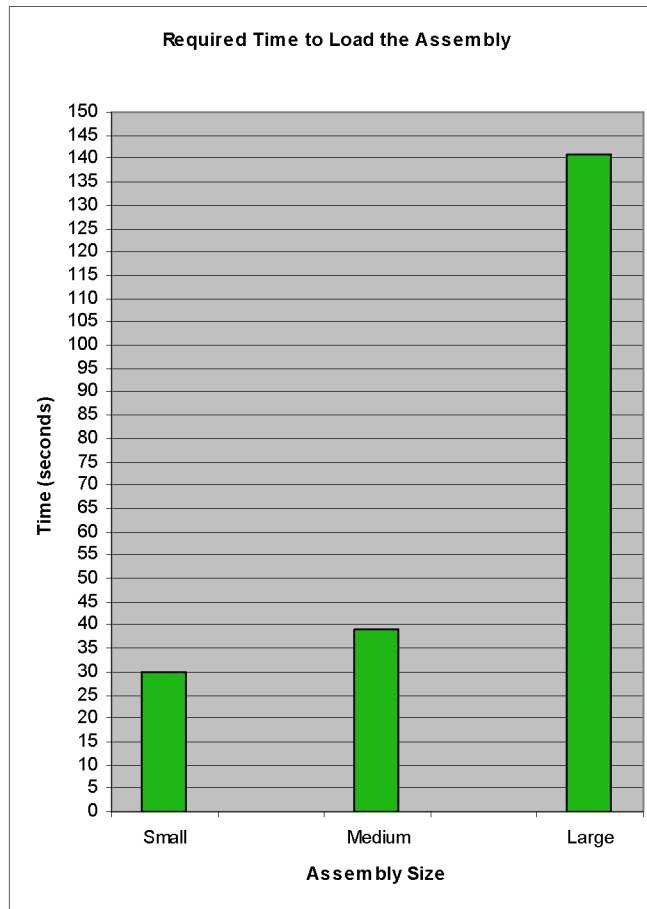


Figure 26. Chart displaying the time required to load the three assemblies

The above results confirm that there is not a linear relationship between time and model size. There are many factors that affect performance in addition to assembly size. The number of dimensions in each part and assembly also greatly impact the system performance. The VR system may have completed loading the models and is displaying the geometry, but the client could still be loading the constraints. The structure and complexity of the assembly tree will have some impact as well as the complexity of the parts within the assembly. The complexity of the part will determine the number of triangles created in the tessellation process and the size of the VRML file. Each node of the four nodes in the Linux cluster reads the VRML model from the samba server, so the time required for the VR system to read in the model is a nonlinear relationship with the VRML file size.

The second timing taken was the time required to modify a part constraint on a component in the assembly. Figure 27 below is a chart displaying the total time required to complete the design change as well as the individual time that was taken by the CAD system and VR system.

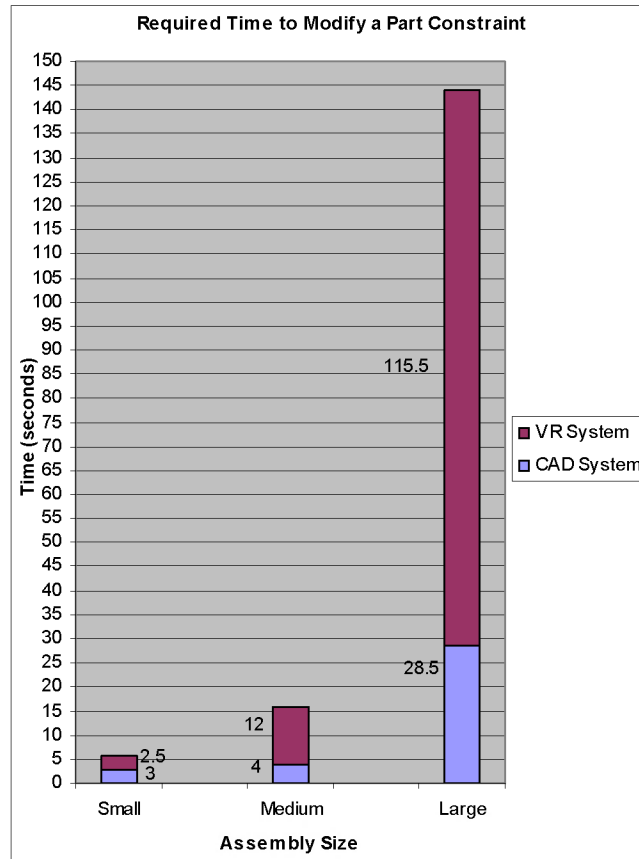


Figure 27. Chart displaying the time required to modify a part constraint

Again there appears to be a nonlinear relationship between time and model size.

When modifying a constraint, the tablet PC does not reload all of the constraint values from the CAD system. This may explain why the time required to modify a constraint is so much less than the time required to load the assemblies with small and medium sized assemblies. Loading constraints into the tablet PC made not have been the limiting factor with the large assembly as the insignificant difference in timings may suggest. Loading such large VRML files into each node of the cluster may be the limiting factor and would explain the similar

times between modifying a constraint and loading the assembly. Another factor that may affect performance is the frame rate of the VR system. The VR system is looking for a modified VRML file every 200 frames, and larger assemblies may reduce the frame rate causing a slower response time to load a modified assembly.

The third timing that was taken was the time required to modify a part constraint on a component in the assembly with the shrinkwrap option turned on. The LOD was set on level two for all of the shrinkwrapped timings. Figure 28 below is a chart displaying the total time required to complete the design change as well as the individual time that was taken by the CAD system and VR system.

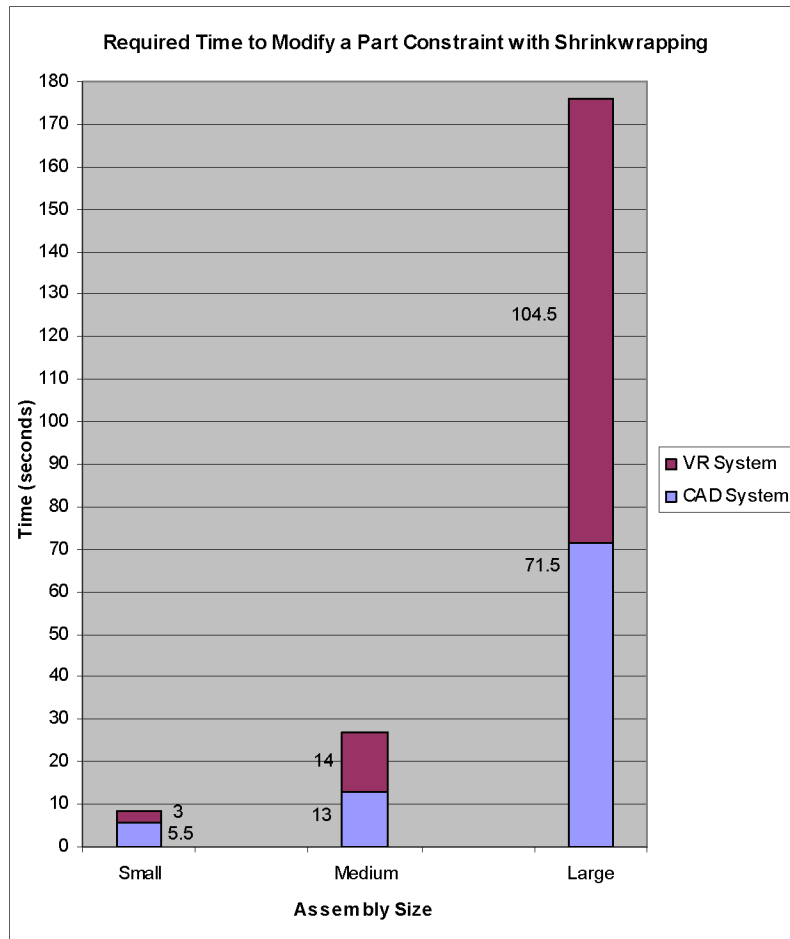


Figure 28. Chart displaying the time required to modify a part constraint with shrinkwrapping

The relationship between time and assembly size with the shrinkwrap feature on while modifying a constraint appears to be very similar to the timings without the shrinkwrap feature on. Shrinkwrapping the models resulted in a longer processing time by the CAD system, and the processing time for the VR system appears to be similar.

The fourth timing that was taken was the time required to explode the assembly. Pro/E is capable of exploding the assembly, so the integrated system utilizes this existing functionality. The time was taken when the “Regenerate” button was selected and again when the exploded assembly appeared in the VE. Figure 29 below is a chart displaying the total time required to explode the assembly by the CAD system and VR system combined.

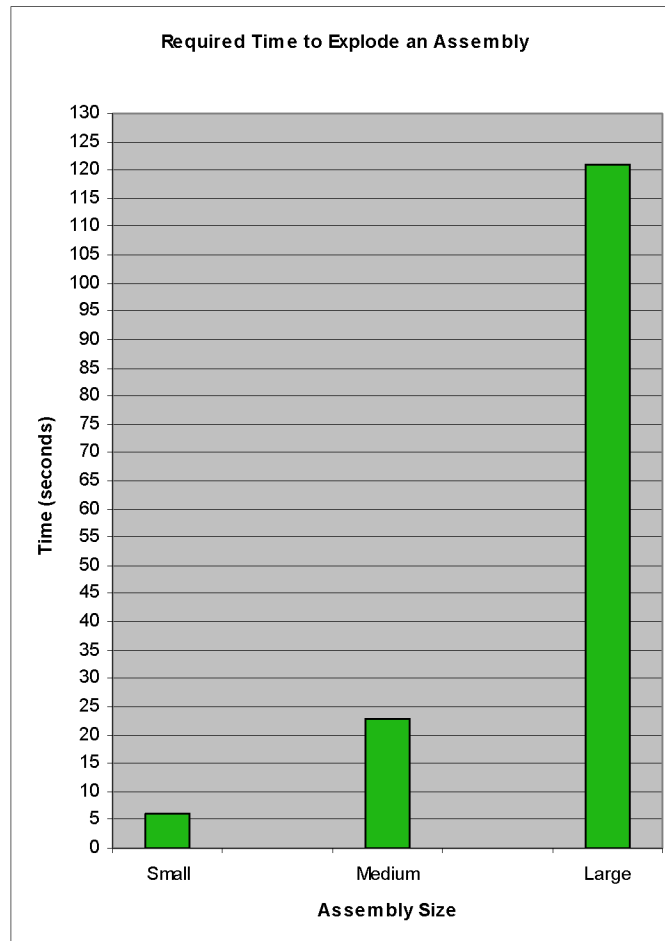


Figure 29. Chart displaying the total time required to explode an assembly with the integrated system

Again, the relationship between time and assembly size appears to be similar to the relationship between time and assembly size when modifying constraints. The difference in time to explode an assembly and time to modify a constraint can be mostly credited to the CAD system's process time. The work done by the VR system is the same for these two functions.

The fifth timing that was taken was the time required to unexplode the assembly. Figure 30 below is a chart displaying the total time required to unexplode the assembly.

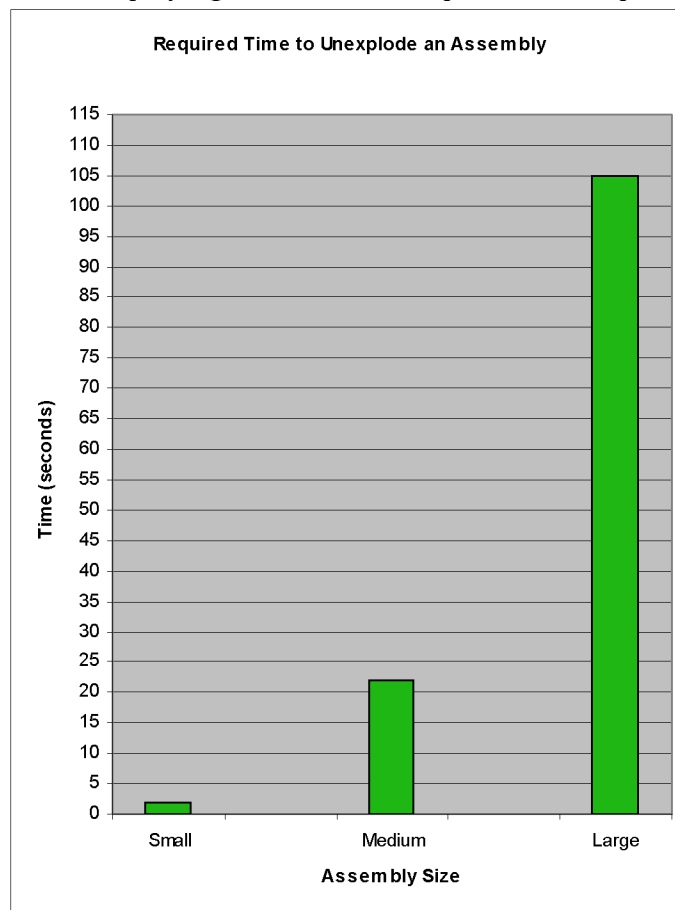


Figure 30. Chart displaying the total time required to unexplode an assembly with the integrated system

The time required to unexplode an assembly is very similar to the time required to explode an assembly. The difference in time between the two can be credited to the CAD

system's process time. The work done by VR system is the same for exploding and unexploding an assembly.

5.3 System Reliability

The reliability of the integrated system is also part of the design requirements. It is very important that the system function properly during a design review. If the system terminates unexpectedly during a design review, it could take several minutes to reload which would hold up the meeting. While running the integrated system, several reliability issues were recorded are discussed in this section.

The largest reliability issue was found to be when models fail to regenerate. After a constraint value is changed, the CAD system must regenerate the model to see the results. If CAD system can not regenerate the model with the new constraint value, it will generate an error and force the user to fix the error to continue. To better understand a failed regeneration, imagine a cylinder with a radius of five and a round on one edge. If the radius of the round is modified to be larger than five, the part will not regenerate.

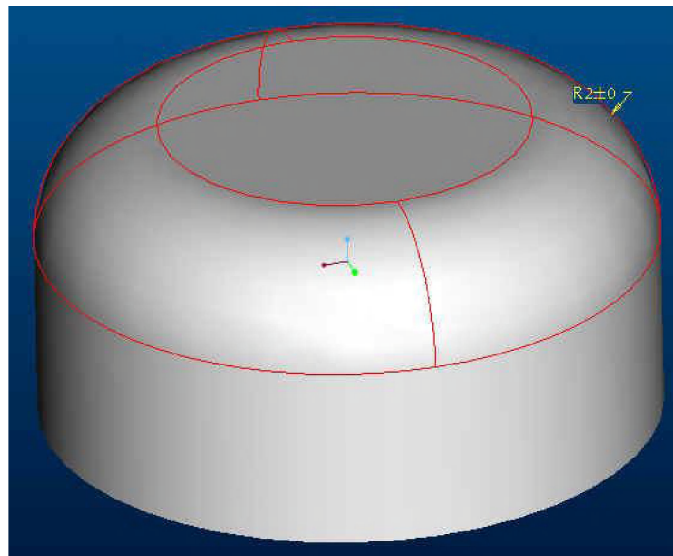


Figure 31. Illustration of a cylinder with a radius of five and a round of radius two

Increasing the radius of the round shown on Figure 31 above to a value that is larger than five would force the round to be self intersecting and the part will not regenerate. If this radius

was changed to a value greater than five by the integrated system, the status bar on the client would continue to read “Modifying part constraint.” The user could assume that the part failed to regenerate if the status bar read the same for more than a few seconds. The user would then need to access the CAD server where Pro/E is running and fix the error with the desktop computer. The modified radius value would then be shown as the wrong value on the client, so it would need to be updated after the error was fixed on the CAD server.

Another reliability issue that was found while testing is that if the system regenerates twice without waiting for the first regeneration to complete, the VR system may terminate unexpectedly. The VR system can begin loading the VRML models as soon as the file with the system time changes. The problem is that not all of the VRML files may be written by the CAD server when the VR system begins loading the assembly. The result is that the system crashes. To ensure that this problem does not happen, the user should not regenerate the assembly until the status bar on the client reads “Ready.”

The time that it takes the system to make a design change and open models is not consistent. As previously discussed, the VR system reads in an information file that contains the time of the last regeneration and the name of the assembly. The VR system reads this file every 200 frames to check if the time in the file has changed. If the time has changed, the VR system will load the assembly whose name written in the file. The inconsistency is caused by the fluctuation in the frame rate. If the VR system has an assembly with a large number of polygons loaded, the frame rate will be slower causing the information file to be read less often. The result is that the VR system does not check if the assembly needs to be reloaded as often. Another factor that may slow the frame rate is the tracking. The user wears trackers on his/her head and wand for interaction. The more movement the tracker needs to detect, the slower the frame rate. The regeneration time could be increased by turning off the tracking while the assembly is regenerating or keep still so that the tracker does not have to detect as much movement.

If the CAD model that is loaded into Pro/Engineer uses relations or uses Pro/Program, the constraints on the client may get out of sync and not read the correct values. A relation is a constraint set up to be a function of other constraints. An example would be the hole diameter of a pipe could be the pipe outer diameter minus a value. If the outer diameter of the pipe is changed, the hole in the pipe is also changed because of the relation. The diameter of the hole on the client in the VR system would not get updated and still read the old value. The client has a feature called “Synchronize Constraints” that will resynchronize the constraints of every component so that they read the same on the client and CAD system. The integrated system does not synchronize constraints after every change because it is a relative slow process and synchronization is rarely necessary. Communication to the CAD system with J-Link API is relatively slow and is often the “bottleneck” of the system when running processes.

The integrated system also has a feature that will check for part interference. The system utilizes this built in feature from Pro/Engineer. When performing an interference check on large assemblies, the function will some times generate an error causing the CAD server to terminate unexpectedly. The user must then restart the CAD server, reconnect to the server with the client, and then reopen the model. It is not recommended using the interference check option on a relatively large assembly or when it is critical that the system does not terminate unexpectedly.

The system can modify part constraints and is also designed to modify assembly constraints. Unfortunately, the system is not able to modify assembly constraints at this time but will available in the future. On the current version of the system, the assembly constraints are displayed on the client and appear to be capable of being modified. If the user attempts to modify an assembly constraint with the client, it will change the value on the client but not on the CAD system.

The reliability of the system is enhanced by not allowing the user to enter constraint values that do not make sense. The client checks what the user has input to see if it is of the proper type, e.g., a decimal number. If the value is not a decimal number, the client will force the user to enter a different value that is. For example the client will not allow the user to enter “5.25.65” or “-2.63-98.”



Figure 32. Message that is displayed when the constraint value entered is not valid

6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

The integration of VR and CAD has potential to improve mechanical designs and decrease the production time through the practice of concurrent engineering. A virtual environment with large stereoscopic displays allows several people to participate in a design review and discuss a realistic looking prototype. The operator of the system may modify constraints of the model to make minor changes to the design during a review.

As a result of this research project, an integrated VR/CAD system was developed and has been presented in this thesis. To begin the development process, criteria were made for the integrated system so that it could be successfully used in a design review. A user interface was then conceptualized to verify much of the criteria before developing the system. Optimization was used to decide which interaction device(s) should be used along with user interface concepts. The interaction devices that would best satisfy this integrated system were found to be a wireless networked tablet PC along with a wand and tracker.

The level of integration and architecture of the system was then developed using the same set of criteria. It was determined that the CAD system (Pro/Engineer) and the VR system would be integrated with the interface called J-Link. The VR system communicates the desired change to the CAD system. The CAD system then makes the changes that were requested and exports the geometry to a Samba file server in the VRML format. The VRML files are then loaded by the VR system from the Samba file server to complete a loop.

A graphical user interface (GUI) is run on the tablet PC to make the design changes was developed using the initial design concept. The actual user interface is different than the conceptual design in that it does not have a graphics window displaying the assembly, but otherwise is the same. The GUI utilizes information hiding to simplify the selection for the

user. The GUI only displays the constraints that belong to the selected component. The only constraint value that is displayed on the GUI is the value of the constraint that is selected from the list.

The speed and reliability of the system were tested using three different sized assemblies. It was found that the smaller two assemblies could be modified much more quickly than the large assembly. The current version of the integrated system would most likely be too slow to make changes to a large assembly during a design review. The large assembly was made up of 264 components and took an average of 143 seconds to modify a part constraint. The medium assembly was made up of 74 components and took an average of 16 seconds to modify a part constraint. The small assembly was made up of 24 components and took an average of 5.5 seconds to modify a part constraint. All three assemblies were made up of cast and forged parts resulting in more polygons and a slower system. The reliability was better than expected. The largest reliability issue occurred when the user entered a constraint value that the CAD system could not regenerate.

The criteria for the integrated system are listed below followed by a list that specifies whether the criteria were met.

- Allow modifications of both part and assembly constraints
- Can be used in parallel with a commercial CAD system
- Allow users to input exact numerical constraint values
- Allow the user to modify constraints while involved in a design discussion
- Allow the user to make modifications quickly
- Allow the user to stand
- Allow the user to navigate throughout the virtual world
- Be able to be used by no more than one person
- Be quick to learn
- Be ergonomically sound

- Operates reliably

The following list summarizes how well the goals of the integrated system were met.

- The current version of the system is only capable of modifying part constraints and not assembly constraints. The interface is setup for modifying assembly constraints, but the system is not able to complete the changes at this point.
- The integrated system can be used in parallel with Pro/Engineer. The modifications that are made with the integrated system are saved in the Pro/Engineer file format. The designs can then continue to be changed on a desktop workstation.
- The interface allows the user to enter exact numerical values and will not accept any value that is not a decimal number.
- A design discussion can easily be performed while modifying the constraint.
- Modifications are relatively quick on small assemblies, but would not meet the criteria on large and complex assemblies.
- The user interface allows the user to stand as well as to sit.
- The user can navigate around the virtual world with the wand similar to how navigation is done in many VR applications.
- The system is able to be operated by only one person; however, it is difficult to run the tablet PC and wand simultaneously. It would be helpful if the wand had a belt clip allowing both hands to operate the tablet PC.
- The system is extremely intuitive to use. Once the system is running, an engineer could easily learn how to operate the system with little or no training in a short period of time.
- The system is ergonomically sound but has room for improvement. The user is managing a tablet PC, pen, wand, and tracking system with a battery pack. There

are many cables and the tablet PC and tracking system can get heavy after long periods of use.

- The current version of the system is more reliable than expected. The system can be run with no problems if the user is aware of the reliability issues. The system has potential to be much more reliable.

6.2 Future Work

After finishing the first version of the integrated VR/CAD system, many ideas for improvements have been considered. Most of the ideas for improvements are to aid the system in better meeting the initial criteria. However, some ideas will be to add new features and will not affect how well the system meets the criteria.

The first improvement is to make the system capable of modifying assembly constraints. The development of this feature was started, but was not completed due to a time constraint. The current version of the system loads all of the assembly constraint names and values into the client and can send modification requests back to the CAD server. The CAD server needs to forward the request onto the CAD system through the API to complete the change. The feature could easily be completed without a lot of additional work.

When making a change to the geometry on the current version of the system, the CAD system exports the entire assembly as VRML files so that the VR system must reload the entire assembly. The system speed could be significantly increased if only the modified parts and assemblies were exported and loaded into the VR system rather than the entire assembly. This could be easily implemented because of the J-Link interface functionality. J-Link has a function call that returns a Boolean value when asking if a component has been modified. The model tree could be traversed, finding the components that were modified and exporting them to the Samba filer server.

The largest reliability issue was found to be regeneration errors that are generated by the CAD server. To improve reliability, the system should be capable of handling these

errors. One option could be to undo the changes and inform the user that the constraint value is not valid on the client. This would be by far the easiest and least complex solution to the problem. This feature was not added to the system due to a time constraint.

With the current version of the system, the CAD server communicates with the VR system by writing files to a server. This is a slow, inefficient method for communication that can be improved. A TCP/IP connection should be made between the VR system and the CAD system to improve performance and reliability. The VR platform that is used for this system will create challenges for creating this connection, but is feasible.

When using the client to choose the component to modify it is sometimes difficult to know which component is actually selected. To clarify which component has been selected, the color of the component could be changed to a unique color and displayed in the VR system when selected. One could also simplify the selection by adding a graphics window to the client and displaying the assembly on the tablet PC. Part picking could then be used in the graphics window to select the desired part.

Selecting the desired constraint to modify is also challenging. All of the constraint names and constraint types are listed for the component that is selected. The constraints have names that generally have no meaning to the user, so it is difficult to know how the constraint correlates to the model. An improvement should be made to inform the user of how the constraint he/she has selected correlates to the model. This could be done on the VR display or a graphics window on the tablet PC.

Other reliability improvements could be made such as requesting the constraint relations from CAD systems, and using the relations to keep all of the constraint values on the client in sync with the CAD system. The system could also be improved by handling the errors generated by the CAD system when performing an interference check on large assemblies.

The tablet PC allows the user to modify constraints, but there may be some cases where the user may want to make minor changes and new constraints would be needed. It could be useful to allow the user to make these changes on the CAD server that is placed just outside the virtual environment. The J-Link interface possesses the capability to develop this functionality. The user could then choose to use the tablet PC or the desktop computer to make modification to the geometry displayed in the virtual environment.

Allowing the user to hide components of the assembly in the VR system would also be a useful feature. A check box could be placed next to the component's name in the model tree, indicating if the component is to be displayed. A cutting plane could also be useful to view cross sections of the assembly. The plane could be attached to the wand in the VR system, so that the user could easily place the plane in the desired location in 3D space.

Bibliography

- [1] C. Boyd, "Human and Machine Dimensions of 3D Interfaces for Virtual Environments," *Conf. companion on Human factors in computing systems*, pp. 41-42, 1995. (Conference proceedings)
- [2] C. Cruz-Neira, D. Sandin, T. DeFanti, R. Kenyon, J. Hart, "The CAVE: audio visual experience automatic virtual environment," *Communications of the ACM*, vol 34, issue 6, pp. 64-72, 1992.
- [3] D. Ullman. "The Mechanical Design Process," The McGraw-Hill Companies, Inc., 2003.
- [4] eSZett GbR. Dec. 2004. Available: <http://eszett-vr.de>.
- [5] F. Kuester, K. Joy, M. Duchaineau, B. Hamann, A. Uva, "3DIVS: 3-Dimensional Immersive Virtual Sculpting," *Proc. workshop on new paradigms in info. visualization and manip. in conj. with the eighth ACM international conf. on info. and knowledge management*, pp. 92-96, 1999. (Conference proceedings)
- [6] F. Purchke, M. Schulze, P. Zimmerman, "Virtual Reality – New methods for improving and accelerating the development process in vehicle styling and design," *Communications of the ACM*, vol. 44, no. 12, pp. 40-43, 2001.
- [7] G. Smets, W. Gaver, C. Overbeeke, P. Stappers, "Designing in Virtual Reality: Perception-action Coupling and Form Semantics," *Proc. INTERACT and CHI*, pp. 11-12, 1993. (Conference proceedings)
- [8] Immersion Corporation. Dec. 2004. Available: <http://www.immersion.com>
- [9] J. Berta, "Integrating VR and CAD," *IEEE Trans. CG&A*, vol. 19, issue 5, pp. 14-19, Sept.-Oct. 1999. (IEEE Transactions)
- [10] J. Gruden, S. Ehrlich, R. Shriner, "Positioning Human Factors in the User Interface Development Chain," *ACM SIGCHI Bulletin, Proc. of the SIGCHI/GI conf. on Human factors in computing systems and graphics interface*, pp. 125-131, 1986. (Conference proceedings)
- [11] Jackson State University. Scientific Visualization Laboratory. Dec. 2004. Available: <http://raven.jsums.edu/-visweb>.
- [12] M. Deering, "HoloSketch: A Virtual Reality Sketching/Animation Tool," *ACM Trans*

- on *Computer-Human Interaction*, vol. 2, no. 3, pp. 220-238, Sept 1995. (ACM Transactions)
- [13] M. Masatoshi, H. Takemura, N. Yokoya, "An Immersive Modeling System for 3D Free-form Design Using Implicit Surfaces," *Proc. of the ACM symposium on VR software and tech.*, pp. 67-74, 2000. (Conference proceedings)
 - [14] M. Mine, F. Brooks Jr., C. Sequin, "Moving Objects in Space: Exploiting Proprioception in Virtual Environment Interaction," *Proc. 24th ann. Conf. on Compo graph. and interactive techniques*, pp. 19-26, 1997. (Conference proceedings)
 - [15] M. Vahl, U. von Lukas, "Integration of Virtual Reality and CAD Based on OMG's CAD Services Interface," http://www.zgdv.de/zgdv/files/zgdv/departments/zr1/Veroeffentlichunge/ecec2003_aper_104/document.pdf, 2003.
 - [16] Microsoft Corporation. Oct. 2006. Available: <http://www.microsoft.com/speech/default.msp>.
 - [17] Nigel Blinman's Personal Website. Dec. 2004. Available: <http://www.blinman.com>.
 - [18] National Aeronautics and Space Administration. Oct. 2006. Available: <http://step.nasa.gov/help/faq.html>.
 - [19] National Institute of Standards and Technology. Oct. 2006. Available: <http://www.nist.gov/iges/>.
 - [20] OpenSG Open Source Software. Aug. 2006. Available: <http://www.opensg.org>.
 - [21] Parametric Technology Corporation. Aug. 2006. Available: <http://www.ptc.com>.
 - [22] R. Rabatje, "Integration of Basic CAD Functions into a VR Environment," *Computer Graphics International 1998*, p.238, 1998.
 - [23] S. Jararam, U. Jayaram, Y. Wand, h. Tirumali, "VADE: A Virtual Assembly Design Environment," *IEEE Trans. CG&A*, vol. 19, no. 6, pp. 44-50, Nov.-Dec. 1999. (IEEE Transactions)
 - [24] SensAble Technologies Inc. Dec. 2004. Available: <http://sensable.com>.
 - [25] Spatial Corporation. Oct. 2006. Available: <http://www.spatial.com/components/acis/>.
 - [26] T. Welsh. VP of Technology. Infiscape. Dec. 2004.
 - [27] Allscripts, LLC. Dec. 2004. Available: <http://www.touchworksemr.com>.

- [28] Univeristy of Hull. Department of Computer Science. Dec. 2004. Available: <http://www2.dcs.hull.ac.uk>.
- [29] VR Juggler Open Source Software. Aug. 2006. Available: <http://www.vrjuggler.org>.
- [30] Web3d Consortium. Oct. 2006. Available: <http://www.web3d.org/>.

Acknowledgements

I would like to thank Dr. James Oliver for giving me the opportunity to be involved at VRAC and for the guidance throughout graduate school. I would also like to thank Dr. Oliver and Dr. Adrian Sannier for their advice and guidance on system architecture design. Special thanks to Jason Schneekloth for teaching me .NET programming and also getting me started with the J-Link API. Finally, I would like to express my gratitude to my family for their lifelong support.