INFORMATION SOCIETY TECHNOLOGIES (IST) PROGRAMME

INTUITION

IST-NMP-1-507248-2

State-of-the-art in VR

Deliverable No.		**D1.B_7**	
Cluster No.	**1**	Cluster Title	**Integrating and Structuring Activities**
Workpackage No.	**WP1.B**	Workpackage Title	**Integration at Resources level**
Authors	Richard Aked (Space Apps) Ioanis Basdekis (ICS-FORTH) Johan Blondelle (Barco) Alessandro Boccalatte (Space Apps) Patrick Bourdot (CNRS/LIMSI) Guillaume BOUYER (CNRS/LIMSI) Annelies Braffort (CNRS/LIMSI) Jessica Cauchard (USAL) Wojciech Cellary (PUE) Bogdan Crenguta (Ovidius) Sergio Di Marca (IDG) Marco Fabbri (Alenia Aeronautica) Marcos Fernandez Marin (IR-UVEG) Terrence Fernando (USAL) Marco Fontana (PERCO) Carmen Garcia-Berdones (UMA) Gilles Gautier (USAL) Philippe Gravez (CEA) Gwenael Guillard (UNIGE) Hilko Hoffmann (FhG-IAO)		Brian Katz (CNRS/LIMSI) Abderrahmane Kheddar (CNRS) Daniel Lichau (MCS) Konstantinos Loupos (ICCS) Jose Louis Los Arcos (Labein) Gonzalo Mendez (UPM) Sas Mihindu (USAL) Alexandros Mourouzis (ICS-FORTH) Norman Murray (USAL) Yoav Nir (Barco) Jerome Peret (Haption) Justo Perez (TECNATOM) Dorin-Mircea Popovici (Ovidius) Arcadio Reyes-Lecuona (UMA) Vaclav Skala (Plzen) Massimo Vota (Alenia Aeronautica) Krzystof Walczak (PUE) Armin Weiss (ART) Kuo-Cheng Wu (USAL)
Status (F: final; D: draft; RD: revised draft):		**D**	
File Name:		**INTUITION-USAL-D-WP1_B_7-SoA**	
Project start date and duration		**01 September 2004, 48 Months**	

Executive Summary

The purpose of this document is to summarise the state-of-the-art in virtual reality technology and its applications. This document will be considered as a “living document” to capture the ongoing advances in virtual reality technology. The content of this document has been structured to capture the advances in Virtual Reality (VR) output devices, VR input devices, virtual reality software platforms and advanced applications.

It is hoped that during the life time of this project, this document will be able to capture the advances made by the key research labs in Europe and worldwide, and will be able to provide invaluable input to the researchers and industrial users. Further input to this document will come from the working groups established within INTUITION.

Contents

1 List of figures

2 List of tables

3 Abbreviation list

AC: Alternating-Current
AOI: Areas Of Interest
API: Application Programming Interface
ATM: Asynchronous Transfer Mode
CAD: Computer-Aided Design
CAVE: Cave AUtomatic Virtual Environment
CCA: Common Component Architecture
CFD: Computer Fluid Dynamics
CGI: Computer Generated Image
CIGI: Common Image Generator Interface
CORBA: Common Object Request Broker Architecture
COTS: Commercial-Off-The-Shelf
COVISE: COllaborative VIsualisation and Simulation Environment
CR: Control Room
CRT: Cathode Ray Tube
CSCW: Computer Supported Cooperative Work
CT: Computer Tomography
CVE: Collaborative Virtual Environments
DBMS: DataBase Management System
DC: Direct Current
DIS: Distributed Interaction Simulation
DIVE: Distributed Interactive Virtual Environment
DLP: Digital Light Processing
DNA: DeoxyriboNucleic Acid
DOF: Degree Of Freedom
DS3D: Direct Sound 3D
DSP: Digital Signal Processing
DVE: Distributed Virtual Environment
EDR: ElectroDermal Response
EEG: ElectroEncephaloGram
EMG: ElectroMyoGram
EOG: ElectroOculoGram
FOM: Federate Object Model
GIS: Geographic Information Systems
HCI: Human Centered Interface
HDTV: High-Definition TeleVision
HLA: High Level Architecture
HMD: Head Mounted Display
HW: Hardware
HW/SW DSP: HardWare/SoftWare Digital Signal Processing
ICT: Information and Communications Technology
IDL: Interface Description Language
IETF: Internet Engineering Task Force
IG: Image Generator
IP: Internet Protocol
IRB: Information Request Broker
ISDN: Integrated Services Digital Network
I/O: Input/Output
LCD: Liquid Crystal Display
LCOS: Liquid Crystal On Silicon
LCP: Linear Complementarity Problem
LED: Light-Emitting Diode
LI: Locomotion Interface
LOS: Line Of Sight
MMI: MultiModal Interfaces
MPI: Message Passing Interface
MRI: Magnetic Resonance Imaging
NDT: Non Destructive Testing
OGC: OpenGIS Consortium
OMG: Object Management Group
OS: Operating System
OSG: Open Scene Graph
PBM: Physically Based Modelling
PC: 1- Personal Computer
2- Phase Coherence
PCB: Printed Circuit Boards
PCE: Path Computation Element
PDA: Personal Digital Assistants
PIP: Personal Interaction Panel
PIU: Protocol Interface Unit
QOS: Quality of Service
QXGA: Quantum eXtended Graphics Array
RPC: Remote Procedure Calls
RTI: Run-time Infrastructure
R&D: Research and Development

SCC: SemiCircular Canal
SDK: Software Development Kit
SOA: State-Of-the-Art
SW: Software
SXGA: Super eXtended Graphics Array
TCP: Transmission Control Protocol
TOF: Time Of Flight
TV: TeleVision
UDP: User Datagram Protocol
USAL: University of SALford
VBAP: Vector Base Amplitude Panning
VE: Virtual Environment
VES: Virtual Education System
VI: Virtual Infrastructure
VLE: Virtual Learning Environment
VM: Virtual Machine
VPS: Voxmap PointShell
VR: Virtual Reality
VRML: Virtual Reality Modelling Language
WSRF: Web Services Resource Framework
VSS: Virtual Sound Server
WWW: World Wide Web
W3C: World Wide Web Consortium

4 Introduction

The main aim of this report is to provide the consortium with a more general overview of Virtual Reality (VR) and Virtual Environments (VEs). It collates together knowledge concerning the current state of the art within existing and emerging commercial VR technology products, concepts and technologies. This will give a clear view of the technological and market status of such a rapidly evolving and diverse field. The characteristics of each of the devices/tools will be defined so that existing VR systems and VE applications can be categorised.

4.1 *What is a VR System?*

In the diagram below, a VR system synthesis has been included, showing the configuration of a basic system, including the connections between various sub-parts of the system. What follows is a brief description of each part of the VR-system and the system as a whole.

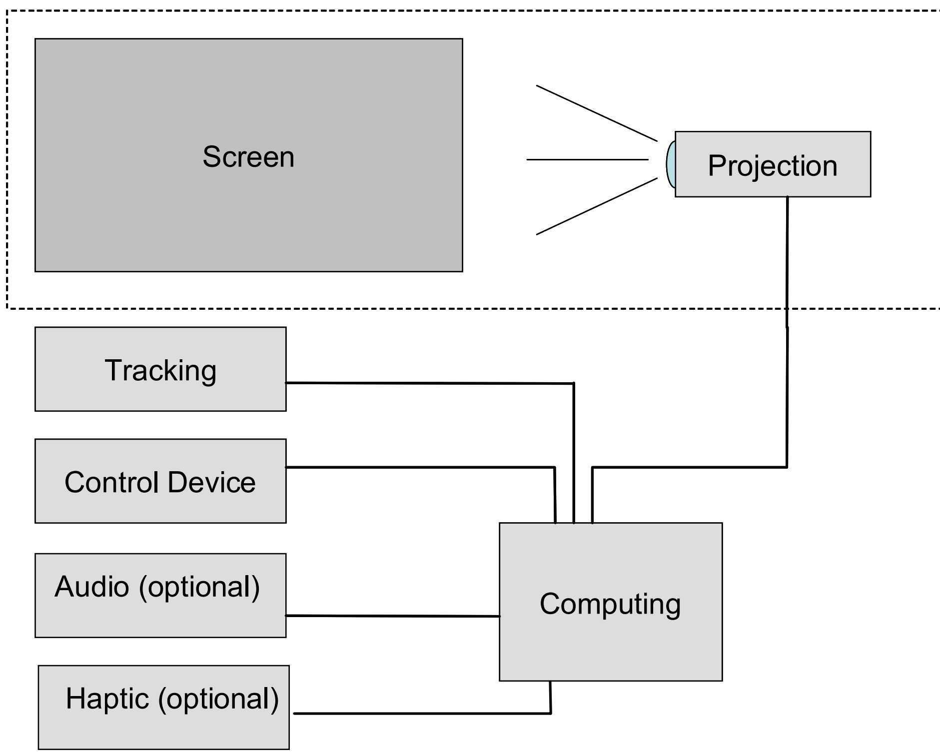


Figure 1 - VR System Synthesis

The "heart" of a VR system, is formed by the computing section that can be a single personal computer or a complete multi-CPU system containing a graphics card or even dual and multi-pipe architectures incorporating local/shared/distributed memory. This is the device that controls the whole VR-system and is the device where the tracking, control and (optionally) audio interfaces are connected while at the same time performs the graphics calculations needed. For the purposes of simplicity this has been shown as one unit, however it can be split into the graphics and the other (tracking, synchronisation etc) processes. The software that runs in this unit contains the operating system of the system and the suitable applications/drivers to incorporate the system as a whole and handle the connected devices, while at the same time perform all the system's calculations regarding moving, object handling as well as graphics processing. This unit as shown provides a suitable RGB-signal to the projection device.

The tracking part of a VR system, can be camera-based, magnetic, ultra-sound, optical (Ribo, 2001), mechanical or inertial (Foxlin et al, 1998), each of them having its advantages and disadvantages depending on the application to be used.

The control device is the system needed for the user to interact with the VR system, and can be a joystick, trackball, wand, glove, 3D mouse, wireless pad/PDA…

The screen of a VR system is the device where the virtual environment is presented. There are various implementations for this to be achieved, ranging from a simple PC monitor to a single/multi-wall, curved screens, front/back projected, active and passive, each of them offering greater or slighter immersion. A description of the categorisation of systems depending on the screen characteristics/specifications has been given in the next section.

The projection unit can also vary between many alternatives. Depending on the type of screen the appropriate projection technology has to be used. This could range between back/front projection, single/multi-projector and active or passive projection techniques.

The audio unit that can be seen in the diagram above is optional therefore there are systems not supporting audio capabilities. It depends on the application to define the need for audio facilities, and whether it could enhance user immersion and realism. The audio interface could be either very simple (headset), simple stereo or multi-channel and 3D sound.

The haptic unit shown in the system can also be optional to provide force feedback to the user when interacting with the virtual objects. The need for haptic feedback also depends on the application. The haptic interface could vary from a simple glove giving simple touch sensation through electrical pulses to full exoskeleton giving force feedback to both arms.

4.1.1 Environmental precautions

There are some precautions that have to be taken regarding the usage of VR systems in non-hostile or harsh environments. These include first of all the presence of dust in the area which when stuck on the screen of the VR system or its mirror (when back-projection is used) is very difficult to remove. Additionally a lot of attention has to be paid on the air-conditioning systems of the area, so that air is circulated, filtered and the temperature is controlled. The temperature should not only be maintained within the systems operating limits but should change witoe-2.7(ed r0t n.2(h)23(u)3(t)-19g.2(h1.1)26. r0t13)-13.2(e-2.7(ed-13.r and)201t)-39.1
l9ly3hen ar u m3h7 sc7m n7m1r3h u7

4.2 Framework for categorisation of VR systems

There are many specifications of VR-systems that define the immersiveness of each system. There are also other characteristics that also categorise each system but mainly in terms of performance.

4.2.1 Immersiveness

In this paragraph a presentation of the three main categories of VR systems based on the immersiveness they provide to the user has been included. The system specifications that characterise them have been presented, categorising the systems into “desktop”, “semi-immersive” and “immersive” as described below.

4.2.1.1 Desktop Systems

Desktop VR systems are special implementations aimed at making the benefits of VR/VE generally accessible for everyday users. The key advantages of desktop VR systems are that they are accessible for everyday users, cheap and un-encumbering. Usually, a simple desktop monitor is used in the desktop VR systems for the presentation of the VE. One of the key limitations of the desktop system is its low degree of immersion due to the use of smaller displays.

Desktop VR can be delivered using various configurations such as a polarised screen and polarised glasses, a stereo screen and shutter glasses (Alpaslan et al., 2005), and a non-stereo screen. Typically, tracking is not deployed in desktop VR systems, though simple tracking is feasible on such systems. For interaction, simple 3D devices such as 3D mouse and joysticks are used in typical desktop VR systems.

4.2.1.2 Semi-immersive Systems

The concept behind the term “semi-immersive” is a fixed, wide angle display system, able to provide a wide angle of view of more than 60 degrees. The most important characteristics of such systems are the quite high graphics performance and computer power. Such systems could incorporate the usage of large screen monitors, large screen projections, or even multiple screen projection systems. For this category low cost LCD projectors could be used but it has to be ensured that the screen is flat to avoid great distortions.

Such systems incorporate the PowerWall™ display systems, typically using several projectors allowing detailed viewing and group interaction with a visual simulation. Alternatively single or multi-wall systems can be used, requiring single or multi-projection systems respectively. Additionally, curved screens can also be used providing the same semi-immersive effects.
(http://www.lcse.umn.edu/research/powerwall/powerwall.html)

Regarding the advantages of such systems, projection based systems provide a much greater sense of presence than the desktop systems since the field of view is much greater. Additionally, the quality of the projected images is very important but for improved resolution, a number of projection systems is needed having the picture divided so that each projection system makes a part of the total composite result. Compared to head mounted display solutions, projection resolutions are by far better, but higher the resolution needed (may need multi-projection systems) higher the cost will be, and it has to be noted that the cost for purchasing and maintaining a projection system is much higher than a desktop system. Moreover,

projection is the ideal solution when a number of people are required to attend the presentation or share the virtual environment.
Concerning the sense of scale, projection-based VR systems can provide a closer to the real visual image (in terms of scale) whereas in desktop VR systems this is not possible.

4.2.1.3 Immersive Systems

In a fully immersive system, the user is supposed to be completely “immersed” within a virtual environment. The sense of full immersion is achieved by providing the user with a visual image wherever he is looking. For this to be done, certain configurations are needed, employing either head-coupled displays which can be head mounted or arranged to move with the head or multi-screen systems that fully surround the user. The idea behind these implementations is that it fills the user’s field of view up to 360° within the environment. It can be noted that again the level of full-immersion depends on the visual representation quality in terms of resolution, field of view and refresh rate. These kinds of systems can provide a higher degree of user immersion since the user’s field of view is taken over by a computer generated view isolating him/her from the real world surroundings.

When talking about immersive systems, the representative systems are the CAVE and the DOME. The former is a system having a surround-screen, surround sound and is projection based. There are several screens needed for a CAVE, surrounding the user and in most cases back-projection is needed. The user is equipped with hand and head tracking devices so that stereo perspective and environment interaction can be achieved. In a CAVE system, multiple users can easily share the virtual environment while discussing and exchanging ideas and discoveries. However, in these systems, only one person is controlling the projection reference point so the other viewers are just passive. The non-tracked users need to stay closer to the tracked user to avoid receiving distorted views of the virtual scene. Typically, the maximum number of users who could share a CAVE at the same time is about five.

The DOME system is a system used for cinematic purposes and is by far larger compared to standard CAVE systems, and they also involve multi-projection technology. DOME systems are large theatre-like environments having a curved screen (dome) above the viewers. Below this screen and around the viewers multiple other (curved) successive screens exist to aid visual performance and user immersion (Refsland et al., 1998).

4.2.1.4 Comparison between Different VR Systems

Concerning comparisons between the above systems, one could focus on various systems characteristics, such as resolution, perception and situation awareness, field of view, immersion and price. Each system has advantages and disadvantages and therefore a good understanding is required of the application that the system will be used for, so that the best system can be selected for the intended purpose.

Main Features	**SYSTEMS**		
	DESKTOP	**SEMI-IMMERSIVE**	**IMMERSIVE**
Resolution	Higher	High	Low-Medium
Perception/Situation Awareness	Low	Mid-High	High
Field Of View	Low (Typ. 50°)	Medium (Typ. 150°)	High (Typ. 360°)
Immersion	Low	Medium	High
Price	Low	Medium	High

Table 1 - VR Systems' Comparison

4.2.1.5 Hybrid Systems

When VR systems were first introduced to the world they were used as VR-only solutions. For example in the automotive industry, only virtual car models were shown on the CADWall, while the oil & gas industry only showed virtual subsurface models. As the use of VR became more and more part of the workflow, the industry demanded an increased flexibility towards the content that can be viewed on VR systems. In many cases the value of the VR content is enhanced by combining it with classical non-VR information. For example office applications such as Microsoft Power Point and Excel allow communication of additional information to the VR content relevant to the subject being discussed.
Hybrid display solutions provide that capability. They can be desktop, semi-immersive or immersive. In addition to a VR-only display, Hybrid displays provide the user the capability to simultaneously visualize VR (stereoscopic 3D or monoscopic 3D) and non-VR (monoscopic) content.

4.2.2 Other characteristics

There are other additional features and characteristics that personify a VR-system. These range from projection techniques, viewing devices to software algorithms. Concerning the viewing devices (glasses) they can be either active or passive. The active glasses are shutter glasses that cut-off viewing from one eye alternatively and at the same time (while synchronised with the projection system) the system projects the suitable image for the “open” eye. This is typically done by using one projector at a high speed so that the user sees the right eye image with the right eye and the left eye image with the left eye. On the other hand the passive glasses are simple polarised glasses and with the aid of two polarised projectors (one for each eye) they provide separate images to each eye.

Concerning projection mechanisms, there are front and back projected systems. These are self-explanatory methods used to provide different types of viewing and interaction scenarios. For instance in CAVE systems or PowerWall systems, back projection is typically used allowing the user to go closer to the screen to freely interact with the virtual objects.

Other characteristics that could describe a VR system in terms of projection are software or hardware edge-blending. These are complicated algorithms that are used in multi-screen (and projector) systems (e.g. double-width screens) and are used to smooth the line created between two images. This can be done in software and hardware each having its advantages and disadvantages depending on the area of usage.

5 Output Devices

5.1 Introduction

The display characteristics are composed of the refresh rate, the resolution, the screen geometry, the luminosity, the field of regard, the field of view, etc…

5.1.1 State of the art projection technology

5.1.1.1 Projector technology

4 mainstream projection technologies are currently available:

5.1.1.1.1 CRT

This is the oldest projection technology, based on three cathode ray tubes (red, green, blue) which are addressed by an electron beam. The image generated by the phosphor surface is projected onto the screen using three separate lenses. The main benefits of CRT projectors are the long lifespan of the components, the photo-realistic representation of the images (no discrete pixels are visible, contrast is excellent) and the flexibility in resolution and refresh rate (120 Hz can easily be achieved, which makes them suitable for active stereo projection). The main downsides are the limited brightness level, which necessitates ambient brightness level control, the large size of the projectors, and the warm-up period and periodic adjustments inherent to the technology.

5.1.1.1.2 DLP

Digital Light Processing is based on Digital Micromirror Devices, developed by Texas Instruments. This technology uses reflective devices, with one mirror per pixel. The micromirror can be deflected either in the “on” or the “off” position to create a bright or dark pixel on screen. Grey levels are obtained by pulse width modulation, vibrating the micromirrors at high frequencies. DLP projectors are available in two main versions: either one DLP chip is used, which creates a full-colour image through sequential display of R, G, B images using a colour wheel, or three DLP chips process the R G B images in parallel.

The main benefits of DLP projectors are the compact size (especially for single-chip), the high brightness levels that can be achieved in conjunction with good contrast, and the overall picture quality and stability. Moreover, the 3-chip DLP projectors are capable of the high refresh rates (96-120 Hz) to display flicker-free active stereo. On ntrpfeon a(ct)-44.(i)16.9(c)2(5(a(ct)p9(D2-20.1(4)8.3(P)-62e)1.5(618.8(l)1 typically limited to 60Hz) and the so-called screen door effect, caused by the transistors and signal wires running between the pixels.

Multi-layer LCD: multi-layer LCD screens enable to create a third dimension thanks to the superposition of the see-through screens. PureDepth offers screens with up to seven layers. One of the drawbacks is the relatively low brightness of the screens that decreases with the number of layers. (http://www.puredepth.com/)

5.1.1.1.4 LCOS

Liquid Crystal on Silicon is a reflective version of LCD, where the LC material is applied onto a silicon substrate that also contains all drive electronics. LCOS is typically used in three-panel configuration, and is available in resolutions up to DHTV and QXGA (4k x 2k are announced for the near future). On the positive side, LCOS offers a good “fill factor” with little space between the pixels, as well as a fast response speed. Devices are small, which enables projectors to be very compact. On the negative side, the maximum brightness levels that can be achieved with LCOS are limited by the complex illumination optics, and some uniformity issues still exist.

5.1.1.2 Stereo principles

Several technologies exist for the projection of stereo imagery. The commonality between all technologies is that two separated images need to be displayed (for left and right eye) with adequate separation in order to avoid crosstalk or “ghosting”. In general, all existing technologies only allow the simultaneous projection of two images, so that only one stereo view point can be displayed. This is especially significant in conjunction with tracking, when the rendered image is adapted to the viewer’s physical position with respect to the screen.

5.1.1.2.1 Active stereo

Active stereo with shutters (or active Infitec) is based on the principle that the left and right eye images are projected sequentially by a single projector, and that the viewers wear shutter glasses to selectively view the left and right eye images. In order to achieve a good, flicker-free image, the refresh rate needs to be at least 96 Hz (which yields 48 Hz per eye). Due to this restriction, only CRT and some versions of 3-chip DLP are natively capable of active stereo projection[1]. The shutter glasses are in fact LCD cells, and are synchronized with the projected images using infrared pulses. On the positive side, the separation between left and right eye image is excellent, and only one projector is required for the projection of both left and right images. Additionally, the effect is independent on the screen type that is used. On the down side, the glasses are heavy and cumbersome, they break, and line-of-sight connection with the infrared emitter needs to be maintained at all times. Some users also report headaches after extended use, probably due to the shuttering effect.

5.1.1.2.2 Active Infinitec™

Active Infitec stereo is based on the principle that the left and right eye images are projected sequentially by a single projector, and that the viewers wear passive Infitec glasses to selectively view the left and right eye images. The selection is created within the projector, and not at the glasses level, as with active stereo above described. This is done by an Infitec wheel that is synchronized with the incoming stereo image. In order to achieve a good, flicker-free image, the refresh rate needs to be at

[1] A shutter system has been developed that allows alternating between two LCD or LCOS projectors at the required frame rate, but this is only gradually finding its way into the market.

least 96 Hz (which yields 48 Hz per eye). This is only available with some three-chip DLP projectors. The separation between left and right eye image is excellent, and only one projector is required for the projection of both left and right images. Additionally, the effect is independent of the screen type that is used. On the down side, the light loss is fairly high.

5.1.1.2.3 Passive stereo

Passive stereo based on polarizers is a very straightforward technology, using the fact that perpendicular polarizers block the light whereas light parallel to the polarizer is transmitted. In its most simple implementation, polarizers are mounted in front of the lenses of about any type of projector. This does cut the brightness to about 40% of the non-polarized value, however. For this reason, LCD projectors are typically preferred for passive stereo applications, as up to 80% efficiency can be achieved.

The glasses are comfortable, lightweight and cheap (even disposable); so that this is the technology of choice whenever large audiences need to be considered. Co-operation, where eye contact with other users is required, is not a problem. However, the screen needs to maintain the polarization, and the separation between left and right eye is good but not excellent. This is most pronounced for linear passive stereo, where the separation is dependent on the head tilt as well. For this reason, circular passive stereo (independent of head tilt) has gained in popularity over the past years. Obviously, two projectors are necessary per channel.

5.1.1.2.4 Infitec™

Passive stereo based in Infitec™ (Interference Filter Technology) is a rather recent development, using spectral filters to separate between the left and right eye images. This combines the benefits of active and passive: a regular screen can be used, the glasses are lightweight and passive, and separation is excellent. The filters do block a significant portion of the light, however, and due to the spectral separation, colorimetric deviations arise. Electronic colour correction is now available for Infitec™, however, eliminating this drawback.

5.1.1.3 Multi user environments

As mentioned above, current stereo projection technology only adequately allows the display of a single eye point position. Therefore, many multi-user environments (e.g. stereo projection in theme parks) are displaying imagery rendered for a central location in the audience. Viewers will thus have more or less distortion in the image, depending on their position relative to the “sweet spot”. In professional environments where multiple users co-operate (e.g. car design, molecular synthesis…), the “key user” is equipped with tracking equipment to determine his position and heading, and the images are rendered for this single view point. The other viewers will share his/her view, with some distortion.

5.1.1.4 Immersive displays

In order to create greater realism in addition to the stereo effect, displays often make use of a certain degree of immersion. The most straightforward way to achieve this effect is to use a large screen that occupies a significant portion of the field of view of the user. This, for example, is the case for so-called CAD walls, where life-size CAD designs are viewed in stereo. In order to avoid shadowing on the screen, these systems are often using rear-projection. The second step in achieving immersion is to use multiple screens that are arranged in a multi-faceted configuration, thereby “folding” the screen around the observers. Various degrees of immersion can be achieved in this way, the pinnacle being the so-called CAVE or I-Space systems that use rear-projected cubes, surrounding the user (including the floor and ceiling) with stereo imagery. The systems normally allow a limited number of users to work at the same time, of which only one will have a correct perspective.

Finally, the screen can also be curved around the viewers in a cylindrical or spherical shape. This has the benefit of offering constant eye relief and avoiding the seams between the channels, but is more complex to integrate.

5.1.1.5 Screen materials

The importance of the screen to the final image quality of the display is often underestimated. It is the last component in the light path that strongly contributes to the final image quality. A matrix of screen materials can be established using the following criteria:

- Front or rear projected
- Passive or active (mono)

Apart from these, one could also make the distinction between rigid and flexible screen, which is significant for cost, mobility and actual functionality. The screen gain is a characteristic that is important for the final brightness performance of the screen. We will provide a qualitative explanation of the gain importance. Last but not least, the screen material and thickness influences the final contrast performance. This is mainly important when using high light output (more than 2000 ANSI lumens) projectors.

5.1.1.5.1 Front passive

These are usually the most difficult screens to work with, as they need a metallic coating to maintain the polarization. Due to the metallic aspect, the viewing angle is severely restricted, creating a very bright image at some locations and a dim one at others. Nevertheless, it is often used for large stereo displays in e.g. theme parks.

5.1.1.5.2 Rear passive

Here too, the polarization needs to be maintained, which typically implies that the screens will have a rather high gain and restricted viewing angle. The problem is less stringent than in front projection, however, and a good trade-off between stereo separation and viewing angle can normally be obtained. The carrier material is usually glass or acrylic.

5.1.1.5.3 Front active

This is the most straightforward screen of all, as essentially a white painted wall is sufficient to achieve good picture quality. A wide diffusion (scattering) angle enhanced visibility from all directions. For enclosed, immersive display environments, integration effects due to multiple reflections between the screens need to be taken into account.

5.1.1.5.4 Rear active

Also a rather easy screen to implement, this is essentially a diffusion layer on a carrier substrate. This makes it possible to optimize for viewing angle, uniformity and contrast, without having to take polarization effects into account. In applications where multiple projectors need to be seamlessly blended onto a single screen, this is the screen of choice.

5.1.1.5.5 Rigid versus flexible

A flexible screen is easy to transport into the final installation site as it can be rolled up. The main drawback of flexible screens is that they are not flat once installed. There is always a light distortion of the screen, which obviously results in a distorted image. In applications where the geometry of the image is key (for example automotive industry) this is not acceptable. Also when multiple projectors are to be edge-blended on a single screen, it is important that the screen is as flat as possible. Any distortion in the screen will make it more difficult to match the multiple projectors to each other.

Semi-rigid screens can also be rolled up for transportation and installation. They have a better flatness level then the flexible screen. Yet, their geometry is not perfect.

The rigid screens made of acrylic material cannot be rolled up. They are more difficult and expensive to transport and install. Acrylic was the most widely used material for CADWalls (Power Walls) in the past. Recent developments in glass material optical characteristics changed that. Glass is the only material today that can guarantee a flat image.

5.1.1.5.6 Screen gain

The screen gain determines the influence of the screen on the light distribution of the projected image. This is a quite complex subject and it is not in our scope to discuss it in detail. Yet, we can provide the following rules of the thumb:

- When blending projectors use a screen with gain 1 or lower
- When the screen gain is higher than 1.5, ensure that the audience is seated in front of the screen and not to the sides of the screen.

5.1.1.5.7 Contrast

The influence of the screen material and gain determines the influence of the screen on the contrast. We mainly want to bring this to the reader's attention so that when they have to make a screen choice, this is a dimension they should verify. In Figure 2, one can see the glowing effect of the raster. This results in a lower contrast of the projected image. In Figure 3, one can see the absence of such a glowing effect and hence a higher contrast screen.

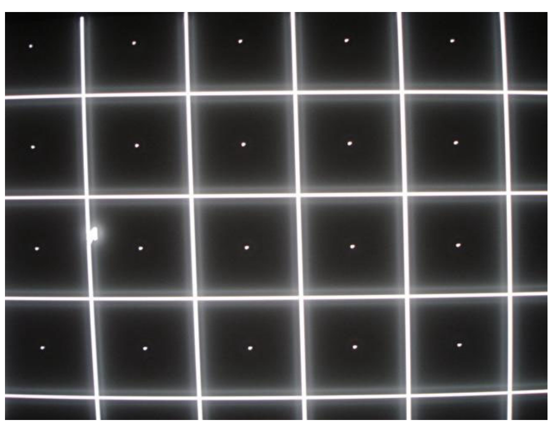


Figure 2 - Regular contrast screen

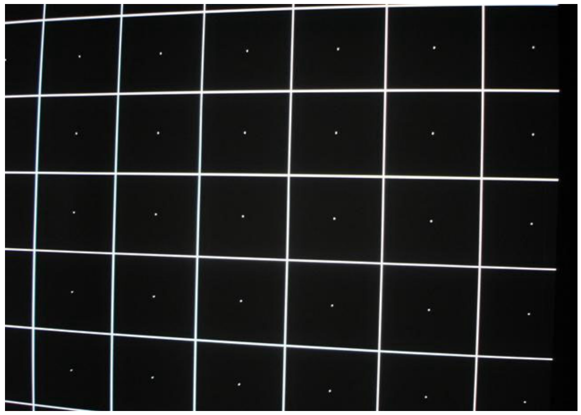


Figure 3 - High contrast screen

5.1.2 State of the Art Display Devices

5.1.2.1 Desktop displays

In terms of true desktop devices, one of the most straightforward implementations of stereo is the combination of a CRT monitor running at active stereo refresh rates, combined with (often wired) shutter glasses. However, as CRT monitors are rapidly losing market share, stereo desktop devices now tend towards autostereoscopic LCD technology (see below).
(Provider: http://www.ps-tech.com/index.php?action=PSS)

5.1.2.2 Multiple Screens Solutions

In many VR setups, such as the CAVE and other (semi)-immersive solutions, the virtual environment is displayed on more than one screen, with each screen typically showing a different portion of the visual scene in order to cover a wider field of view. The image shown on each screen needs to be generated by the software application and sent to a different output device (monitor or projector) for visualizing it.

This can be achieved today in two different ways:

- using a cluster of PCs, each responsible for rendering a different view;
- using a single PC with multiple graphics cards installed or with a single graphics card with multiple outputs (multiple heads)

Of course the two solutions above can potentially be combined together, i.e. using a cluster of PCs, where each computer has multiple video outputs.

5.1.2.2.1 Cluster of PCs

A typical approach consists in using a cluster of PCs, each of which is dedicated to render the scene with a different viewing angle; the computers are connected via a network interface, such as fast Gigabit Ethernet, in order to keep synchronized with each other for rendering the different views consistently; each computer sends out the images to a different display (e.g. projectors). An example is illustrated in Figure 4 for a three-screen configuration.

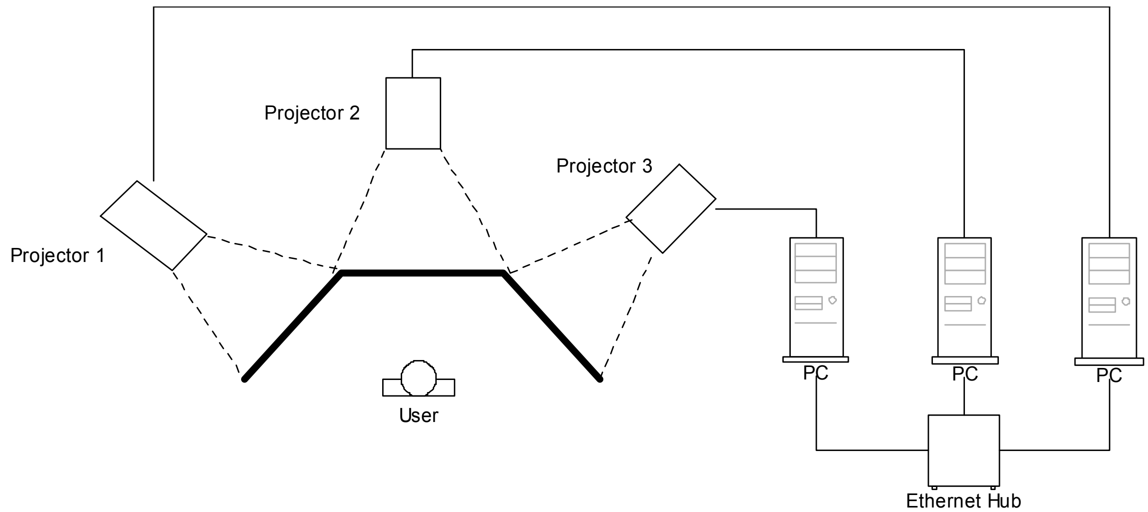


Figure 4 - Multiple Screen Display

In order to achieve a good synchronization between different views (screens), it is important not only to synchronize the software applications running on the different computers (via the network connection), but also to ensure a good synchronization between the video refresh timing of the different screens, i.e. that images are redrawn on each screen at the same time; in other words, when a vertical screen refresh starts on a video output, it should start also on the other video outputs. This implies a synchronization of the so-called vertical retrace signals across all the screens and usually requires the use of Genlock (generator locking device) synchronization.

Genlock can be achieved by professional graphics card (e.g. NVidia Quadro) which allows to be connected to each other using dedicated connectors; a graphics card on one PC is connected to the graphics card on another computer and the refresh signals are sent from one graphics card (acting as master) to another (slave) so that the vertical refresh is forced to start at the same time on all video outputs.

Another possibility consists in using an external Genlock device which can be connected to different graphics cards, ensuring their output synchronization.

5.1.2.2.2 Multiple graphics cards

Modern computers are provided with PCI-express slots, which not only provide a large data throughput, but also allow installing one or more graphics cards on the same machine. Since all modern graphics boards come with (at least) two digital output channels (DVI), the adoption of two graphics card can potentially drive up to 4 different screens, obtaining a quad-view configuration with a single PC.

Alternatively, if the two outputs of each card are used for stereoscopic visualization (one output for the left eye view and one for the right eye view), the adoption of two graphics card can still support two different views on a single machine. Nonetheless, using quad-buffered stereo (for active stereo systems) stereoscopic visualization would be supported again on up to 4 different views.

5.1.2.2.3 Multiple-head graphics cards

Video cards manufacturers have recently released new graphics boards supporting more than two video outputs. For instance, the NVidia Quadro FX4500 X2 supports 4 different digital outputs channels

(Figure 5), allowing to drive four different displays, each one with a maximum resolution of 2560 x 1600 pixels (for a total horizontal resolution of nearly 10,000 pixels).

Figure 5 - NVidia Quadro FX4500

5.1.2.3 HMDs

Head Mounted Displays are a very interesting and low-cost solution to achieve excellent immersion at a moderate price: as the micro-display is worn in front of the viewer's eye, the total field of view can be covered by a relatively compact device. Although a number of semi-transparent version of HMDs are available (often monochrome), allowing simultaneous viewing of the real world and computer-generated imagery, the majority of the devices are occlusive. As left and right eye are covered by separate displays, full stereo viewing is possible. Resolution is limited to SXGA+, driven by micro-display technology.

The use of occlusive HMDs has a number of consequences for the use: low-latency tracking is required to rapidly adapt the displayed imagery to the physical movements of the viewer, so as to avoid the occurrence of motion sickness. Although it is very easy to immerse every user in his/her own virtual space, co-operation is less obvious due to the non-transparent glasses. This implies that avatars or video-captured images of the other users are required. Finally, a major drawback of HMDs is that they are never fully wireless, so that freedom of motion can be compromised.
(List of products: http://www.stereo3d.com/hmd.htm)

5.1.2.4 See-through HMDs

5.1.2.4.1 Video see-through

Video see-through HMDs are based on fully occlusive displays; two little cameras (one for each eye) capture images from the real world and send them to the two displays of the HMD, allowing the user to see the surrounding world through a real-time video stream; computer generated graphics can be superimposed on top of the video images in order to provide the user with a Mixed Reality view.

5.1.2.4.2 Optical see-through

Optical see-through make use of semi-transparent displays, where the user can directly see the surrounding real environment; computer generated images can be superimposed to the real view for AR applications.

The most recent optical see-through devices consist in extremely compact and light-weight goggles that can visualize computer-generated, high resolution, full colour images. Examples of ultra-compact monocular optical see-through visors are represented by Liteye-500 (www.liteye.com) and Shimadzu DataGlass2 (http://www.spi-inc.com).

(List of products: http://www.stereo3d.com/hmd.htm)

5.1.2.5 Projections Systems

5.1.2.5.1 Powerwall (aka CADWall)

The Powerwall was one of the first multi-channel visualization walls, used for collaborative analysis of scientific data. The high resolution and large screen size were achieved through the combination of 2 x 2 projectors in a rear-screen configuration. The Powerwall has led the way to a large family of comparable display systems such as the CAD-Wall (for 1:1 scale visualization of CAD data), the HEyewall, and many others. The CRT projectors used in the Powerwall have now been replaced by bright light valve projectors using either active or passive stereo, and blended into a seamless display using a 1- or 2-dimensional array of channels. As projector sizes have shrunk considerably, it is now possible to create very dense arrays, thereby achieving extremely small pixel sizes. Additionally, computing power has become so affordable that addressing large amounts of pixels is no longer restricted to supercomputers. The fact that one can approach the screen without casting a shadow has made it highly suitable for interactive applications.

(Provider: http://www.fakespace.com/powerWall.htm)

5.1.2.5.2 ImmersaDesk

Developed by the renowned EVL, the ImmersaDesk is in essence a tiltable single-channel display, which can be used as a virtual tabletop. Due to the limited screen size and the easy access to the screen surface, these displays are mostly used for interactive applications with a small number of users. Again, many variations on this theme have found their way into the market, helped by the limited dimensions of the setup and the transportability that is greatly facilitated thereby. One drawback of this system is that the sense of immersion and the depth perception are relatively limited. Due to this restriction, an L-shaped configuration (see Barco Consul) is often used to let the virtual object protrude further from the screen.

(Provider: http://www.fakespace.com/M1Desk.htm)

5.1.2.5.3 Curved

As mentioned earlier, the degree of immersion can be increased significantly by wrapping the screen around the users. For this reason, systems covering 150° or more, to cover the whole field of view, are quite common. In “mono” configuration, this display type is widely used for various simulators (car, cockpit, ship’s bridge…): it is easy to get a mock-up in the centre, and actual stereoscopic cues are often irrelevant due to the large distance to the virtual objects displayed.

The combination of curved screens and stereo is less common: in the case of passive stereo, a large polarization-maintaining screen needs to be used, its metallic reflection characteristics complicating uniform illumination. In the case of active stereo – nowadays usually done with 3-chip DLP projectors – the complication lies in the fact that ultra fast display at double refresh rates needs to be combined with soft edge blending and geometry correction (to compensate for the curved screen shape). This explains why this combination is usually found in the higher segment of the market.

(Provider: http://www.barco.com/VirtualReality/)

5.1.2.5.4 Spherical

One step up from curved screens, spherical displays create an even larger field of view through the use of 2-dimensional tiling of channels. This configuration is most often found in aircraft simulation, as well as in planetariums – usually combined with inclined seating or dome surface, to cover as large a field of view as possible. On the projector side, this type of display is quite challenging due to the multi-sided blending required. Hardly ever is the combination with stereo used: for aircraft simulators, because there is no added value to the use of stereo, for planetariums because of the practical complications associated with the use of stereo. For example, planetariums often use circular seating patterns for the viewers, so that the stereo parallax for viewers on opposite sides of the dome is inverted. It is thus in many cases impossible to use stereo at all, independent of the technology used.

(Provider: http://www.mew.co.jp/e/corp/press/2004/0406-02.htm)

5.1.2.5.5 CAVE

Again developed by the EVL, the CAVE (Cave Automatic Virtual Environment) is a cubic space with projection on all sides. This obviously leads to an astonishing degree of immersion for a limited number of viewers, especially when stereo is used. The use of a cube shape leads to square or rectangular screens, which means that the projectors typically don't require geometry distortion or blending capabilities. Passive stereo is slightly less popular than active, due to the high-gain screens required and the difficulty of establishing a consistent left/right polarization orientation across the whole display. The high gain screens lead to significant brightness non-uniformity, thereby making the seams between the screens more visible.

Tracking is normally foreseen inside a CAVE environment, as movements of the viewer lead to large changes in the aspect ratio and the fields of view covered by the screens. (It is interesting to note that the use of electromagnetic tracking has led to the development of wooden structures for the CAVE, to avoid interference) Due to the restrictions in stereo projection technology, this means that only one viewer will receive a correct stereo perspective; the others will see "through his eyes".

(Provider: http://www.fakespace.com/cave.htm)

5.1.2.5.6 Reconfigurable Systems

As can be deduced from the name, reconfigurable systems form a hybrid between various configurations listed above, using a number of rectangular screen segments of which the position can be changed. A typical setup would allow the users to switch between a PowerWall/CAD Wall configuration, with all screen segments folded open (possibly in conjunction with floor projection to achieve a higher degree of immersion), a 45° angled configuration for e.g. small group presentations, and the CAVE-like configuration for full immersion.

Although reconfigurable systems offer the users excellent flexibility, the degree of complexity to achieve smooth (automated) switching between the various configurations and the associated image generator settings should not be underestimated.

5.1.2.5.7 FogScreen™

The FogScreen™ (Figure 6) is a projection screen formed by a thin curtain (1-3 inch) of "dry" fog. The stability of the image is assured by using only water, by stabilising the airflow thanks to several fans which protect it from turbulences, and by using ultrasonic waves. One particularity of fog screens is the possibility to walk through them.

Figure 6 - Fog Screen™

5.1.2.6 Virtual Retinal Display

More or less comparable to HMDs; virtual retinal displays create an image directly onto the retina of the eye. This is achieved by scanning a laser beam across the retina; there is thus no screen. Light sources other than laser; are less efficient due to optical refraction, which limits the minimum "pixel" size that can be addressed at the retina surface. The obvious benefit of this approach is that very compact deflection optics can be used, so that the field of view of the user is not blocked by the display itself. Virtual retina displays are thus supremely suited to Augmented Reality applications, where computer-generated imagery is superimposed on the real world. Nevertheless, the system has not broken through yet, mainly due to the fear of having a laser shine directly into the eye. For the time being, only low-resolution monochrome devices are on the market, but the advances in semiconductor technology should enable full-colour devices in the years to come.

5.1.2.7 Autostereoscopic Displays

Autostereoscopic displays differ from all of the systems mentioned above in the sense that no special glasses or head-worn devices are required to achieve stereo imagery. This is achieved by means of parallax separation: the image is split into odd and even columns; and optics are put between the display and the viewer so that the left eye only sees e.g. the even columns and vice versa. The most straightforward way to achieve this is to use a barrier (shadow mask), which forms a physical obstruction to block the unwanted column from view. Obviously, one major drawback of this approach is the significant amount of brightness loss. Alternately, cylindrical micro-lenses or prism arrays can be used to deflect the columns' image into separate directions. If the choice is made to deflect into two zones only, the system can only be used by one viewer at a time (there is only a viewing zone for the left and right eye), but the horizontal resolution is only halved. (e.g. a 1280 x 1024 display yields 640 x 1024 for each eye). This can be configured either in a static way, where the lens or prism arrays are preset to cover a fixed "sweet spot" where stereo can be seen, or in a dynamic way, where the array is moved in front of the display in response to data gathered by eye or head tracking.

An alternative approach is to split the image into a large number of viewing zones (e.g. 8), so that multiple viewers can achieve a correct perspective at the same time, or so a viewer moving laterally in front of the screen will see the virtual object change its aspect ratio and viewing angle. The major drawback of this approach is the tremendous loss in horizontal resolution – in the example above, an SXGA display would yield only 160 x 1024 resolution after application of a lens or prism array. A recent development is the switchable LCD panel that can work in full-resolution "mono" mode or in half-resolution stereo mode. This panel has been embedded in PC and laptop monitors, as well as cell phone displays, and is especially attractive thanks to its moderate price point.

The Mixed Reality 3D display proposed by the Henrich Hertz Institute is a 27" screen with a resolution of 3200 by 1200 pixels. This is obtained by used two high resolution video projectors wich project the image on a filter panel. The image appears above the desktop in holographic quality, and the user can interact with it by direct manipulation. It has to be noted that the autostereoscopic displays have recently appeared in mobile devices thanks to the Sharp Actius AL3DU. However, this notebook suffers from the relatively small area which enables the viewing of the 3D, and which requires the user to have a fixed head position.

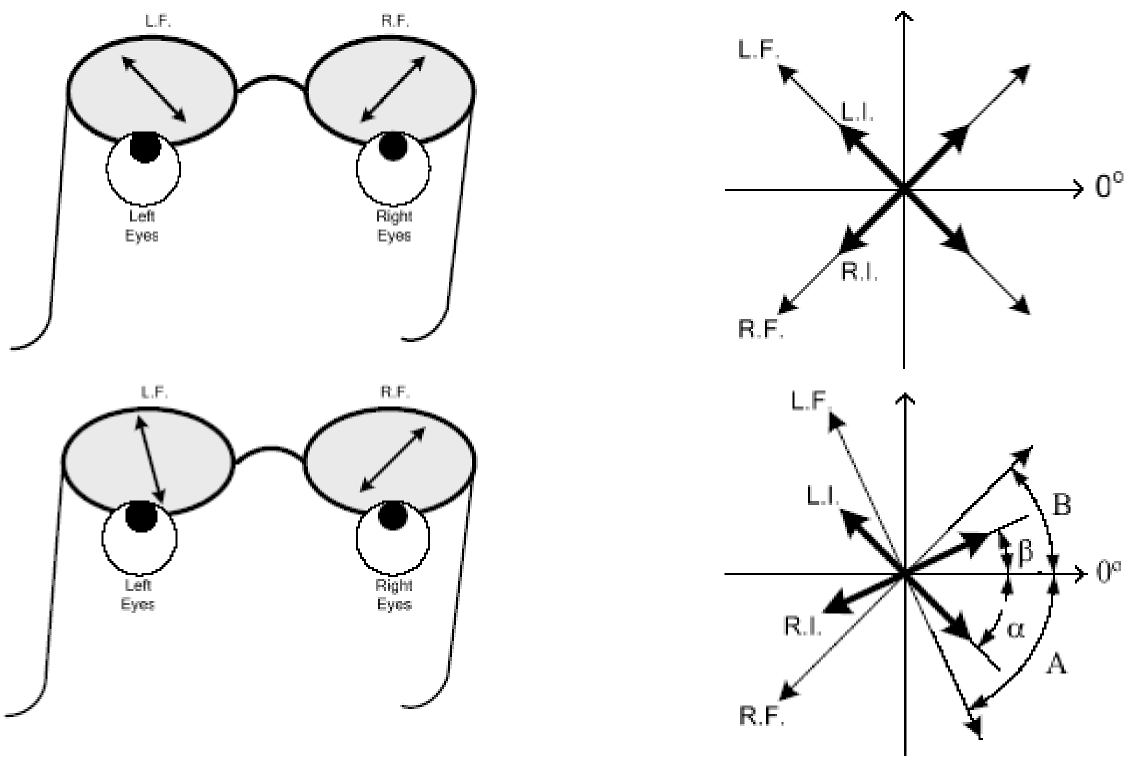
Providers: http://www.3dcgi.com/cooltech/displays/displays.htm)

5.1.2.8 Stereoscopic Display using variable polarized angle

Stereoscopic display systems have been developed without any multiplexing between left and right eyes. Engaging two LCD panels one to control total pixel intensity and other to control left-eye/right-eye distribution ratio could effectively increase the resolution and avoid crosstalk.

Different to the conventional polarized stereoscopic system the oblique system has the following characteristics:

- The two polarized filters are not located at 90º from each other.
- The angle of a polarized image is not the same as a corresponding polarized filter i.e. the left filter is not at the left image angle and vice versa.
- The system of angles is selected is such a way to cancel stereoscopic cross-talk, i.e. leakage from



conventional polarized stereoscopic system (above)	non-orthogonal polarized stereoscopic system (below)

Table 2 - polarized stereoscopic systems

This system has been developed in collaboration with PolarScreens Inc., QC, and Canada, and MacNaughton inc. (NuVision), USA (MacNaughtonet al., 2006).

5.1.2.9 Volumetric Displays

This display type, too, can be viewed without special apparatus. The concept is quite simply that individual points (“voxels”, volume pixels), can be addressed individually inside a given volume. This can be achieved in a number of ways. The first one, addressing a discrete number of Z-planes, uses a projection engine behind a stack of screens with switchable transparency. The 3D image is projected from front to back, first making the screen segment closest to the viewer opaque and projecting the image segments that should be on this layer, next the second layer, and so on to the screen layer furthest away from the viewer. This approach yields reasonably good resolution parallel to the screen layers, but the depth resolution is limited by the number of screen layers.

A better approach is therefore the one adopted by Actuality Systems, in which a very fast projection engine is mounted below a spinning screen, encapsulated inside a Perspex sphere. The principle is comparable to the one described above, with the main difference that the spinning screen allows addressing of much more height levels; multiple millions of individual voxels can be addressed, and the projected image seems to float inside the sphere. The main drawbacks of this approach are the (still) limited resolution, the lack of opacity (only transparent volumes can be generated), and the limited size of the resulting image.

(Provider: http://www.actuality-systems.com/site/content/products.html)

5.1.3 Display management tools

The need for hybrid display solutions was highlighted earlier. The user needs to be able to visualize several sources simultaneously on the display. This could be for example different IGs in combination with PC and video images.

Typically this is done with the help of switchers and splitters that provide the physical routing of the video signals, and a touch panel as a user interface. Typically several user scenarios are pre-defined that can be recalled via the touch panel. This solution has several drawbacks. One is that the touch panel GUI (Graphical User Interface) as simple as it is, will scare off non-trained users. Another drawback is that typically, the number of sources that can be viewed simultaneously is limited. A third drawback is that during the switching from one source to another source, the projection system needs to switch to a different projection mode, which can take up to a few seconds and does not always work.

Another solution that does not have the drawbacks mentioned above is based on a single integrated solution. It is the XDS-1000 designed and manufactured by Barco (http://www.barco.com/VirtualReality). The user interface is on-screen and Windows XP-based, a format that is familiar for all professional users. The number of sources that can be viewed simultaneously is virtually unlimited. The display is driven in a single projection mode independent of the sources that are switched. This contributes significantly to the stability of the system.

5.2 Auditory Channel

5.2.1 Auditory Characteristics

Audio plays a significant role in virtual VEs. The user's sense of presence may be increased considerably by the use of sounds during navigation and interaction. Naturalness of audio can be achieved by both the presented audio spectrum (the number of different kinds of sounds used) and the sound locations. Employing spatial sound allows the participant to become aware of the location and direction of sound. The audio signals continuously change to reflect the participant's movements, creating a feeling of space.
The importance of sound in human perception is emphasized by the fact that total silence is present only in vacuum and in our everyday environments some type of background noise is always present. People suffering from blindness for example are very skilled at listening to audio cues to build a mental model of their environment.

When creating virtual auditory environments, the research problem, on a general level, is to model and simulate sound propagation from sound sources to the ear drums of a listener through the modelled space. Rendering of acoustics, i.e., modelling of sound propagation through the modelled space has been studied quite long time in the field of concert hall acoustics. However, when modelling concert hall acoustics, the goal is to simulate impulse responses in certain positions to be able to compute acoustical attributes, which objectively describe the characteristics of the acoustical space under study. For interactive virtual reality applications simulation methods for static impulse responses cannot be use. Instead, simulation methods that support interactive rendering with 3D sound reproduction capabilities are needed. Currently, two different approaches to model room acoustics in real-time is applied, namely Perceptual modelling and physically-based modelling. Perceptual modelling (Jot, 1999) is an approach where audio rendering is based on statistical models, not the geometry of the modelled room. This technique is well suited for interactive applications since human hearing is not so accurate and efficient rendering methods can be used to achieve real-time response. The biggest application area, computer games, mostly applies perceptual modelling to achieve such real-time response. The other approach is to use physically-based modelling (Savioja et al., 1999; Lokki et al., 2002), where sound paths through the modelled space are traced. Many techniques, such as image-source method (Allen and Berkley, 1979; Borish, 1984) or cone-tracing (Funkhouser et al., 1999) exist for sound path searching. Typically, in real-time rendering, only perceptually the most relevant early reflections are traced and late part of reverberation is modelled with efficient artificial reverberation algorithms (Gardner, 1997).

When designing collaborative working environments, one major challenge is to create correct soundscape to all listeners. In other words: to produce accurate 3D sound field to all listeners. The most accurate 3D sound reproduction method is binaural reproduction with headphones (Begault, 1994). Binaural audio rendering, based on convolving audio signals with Head-Related Transfer Functions (HRTF), enables the listener to accurately perceive the location of the sound source in true 3D space (instead of inside the head as in stereophonic headphone listening) (Blauert, 1997). As an advantage, the sound rendering is direct to the individual and does not interfere with the real acoustic environment. Multiple users require separate renderings. As with stereographic rendering, for interactive systems with head-tracking the sound field has to be calculated and updates for changes in user position and rotation in the virtual environment. Binaural rending is most limited by the need for adaptation to the individual user. Similar to stereographic rendering, where the inter-eye spacing is adjustable for optimal 3D perception, there are a number of parameters which vary from person to person in binaural rendering. Much of current research and development is focused around mastering these parameters. (Listen Project, 2007;,Rio and al., 2003; Katz, 2001) Without proper adjustment of the system to the individual, elevation rendering and accurate localization quickly loose in quality and user satisfaction.

For loudspeaker systems, several techniques are available. For lateral movement of sound sources in the frontal direction (no elevation) a conventional stereophonic loudspeaker arrangement may already be sufficient. Multi-channel sound formats and systems designed for the home theater market and loudspeaker arrangements like AC-3, Dolby Surround and DTS enable the listener to be encircled by sound. These systems, based on the same principles as stereo rely on amplitude panning from speaker to speaker within a pre-defined loudspeaker configuration. While suitable for general sound reproduction in enclosures, source localization accuracy and precise spatial detail are limited (Bergault, 1994)True 3D rendering via loudspeakers can be achieved with an Ambisonic system (Gerzon, 1973), with a Vector Base Amplitude Panning (VBAP) (Pulkki, 1997), which allows arbitrary positioning of loudspeakers These systems allow for large loudspeaker arrays which are inherently pre-defined. Accuracy depends on the number of loudspeaker channels and also on the processing power allocated. For large installations, listener position is also important (Strauss and Buchholz, 1999). Recent studies have a for user tracking. (Corteel, 2004) Requiring large numbers of densely spaced loudspeakers and processing power, this approach provides high quality rendering for numbers of listeners, irrespective of their position. Systems are limited to horizontal synthesis due to complexity and integration with rear project systems is not straight-forward.

Recently, there have been several efforts (Eurographics-HCI-Siggraph) to bring acousticians and virtual reality researchers to promote the development of real-time auralisation systems (virtual acoustic) for immersive virtual environments. Spatial sound plays an important role in virtual reality environments, allowing orientation in space, giving a feeling of space, focusing the user on events in the scene, and substituting missing feedback cues. In order to create a realistic 3D auditory display in the virtual environment, the virtual acoustic system must simulate the acoustic scene, including sound source, listener and environment attributes accurately. The user can affect their listening experience by moving about within the virtual environment. The latency of these systems must be low enough (~50 ms) to prevent a listener from recognising a delay between an action and its corresponding acoustic effect.

In the future, some spatial sound APIs and rendering software should be developed. Currently, no common spatial sound API exists for virtual reality applications. For computer game industry the developers have agreed on rendering guidelines (http://www.iasig.org) which define functionality and features a computer game sound API should implement. Such guidelines that also define multi-channel reproduction devices should be created in the near future for virtual reality applications.

5.2.2 Technical Challenges in 3D-Sound

One major technological limitation is the lack of optimal sound reproduction method. All known methods (both binaural and multi-channel) have some limitations and signal processing needed for auralisation is depending heavily on the applied sound reproduction method. Another future challenge is efficient real-time sound synthesis of ‘everyday sounds’. Such sound signals are needed in virtual reality applications and the control of synthesis models should be interactive.

Making 3D acoustics work without headphones will be a crucial milestone in 3D acoustic technology. When this can be done for multiple users, then it can realistically be implemented in office environments. At present, 3D acoustics is possible for one person in one position. Acoustic shielding and directional sound in the future will allow voice recognition software and audio feedback to be used by multiple users in a shared space without disruption to the surrounding people.

A future workplace will consist of more data being generated and exchanged; therefore using speech (speech recognition, speech synthesis, and speech enabled information search) to facilitate access, content creation, and transactions is the aim. To implement this it is necessary to create a robust speaker-independent natural language (not fixed vocabulary) recognition tool, in noisy environments, and in multiple languages.

In the future, speech technologies should also help to enrich communication between people. For example, automatic summarization and indexing of meetings, lectures, and daily conversations can improve human-to-human speech communications.

5.2.3 Hardware support for 3D sound reproduction

The hardware supporting 3D sound can be divided into two categories: 1) standard PC sound cards, 2) dedicated DSP systems. The API’s for standard PC sound cards are discussed in a separate section, and in this section we concentrate on dedicated DSP systems. Unfortunately, typical dedicated DSP systems are expensive and hard to program. In addition, porting of programs between different systems is very difficult. In the following is a list of main available devices:

- Convolvotron, the first hardware with 3D sound. The system is not available anymore

- Lake Huron, the most promising of current HW/SW DSP environments for audio processing. Existing modules for 3D sound and room simulation available
  (Provider: http://www.dolby.com/professional/Live_Sound/index.html)

- Tucker Davis Technologies RX6, powerful multi-DSP module for 3D sound
  (Provider: http://www.tdt.com/index.htm)

- Lexicon effect processors, efficient general purpose devices, but creation of physically accurate room simulation is impractical
  (Provider: http://www.lexicon.com/)

- Sharc and Tiger Sharc DSP, very powerful DSP boards, no special support for 3D sound. Note that, there are plenty of similar boards available.

5.2.4 3D Audio API's and VR Software

The latest releases of APIs like DirectX or EAX (Environmental Audio Extensions) include functions for the creation of spatially distributed sound sources, however the capabilities of these libraries are rather effect-oriented (as they depend on low-end hardware) rather than precise and accurate 3D audio rendering and therefore do not currently meet the requirements for interactive auditory virtual environments.

For true 3D audio, commercial products exist with dedicated DSP hardware like the Huron workstation (http://www.lakedsp.com/). Other software based products, which make use of continually increasing computational power, exist such as Max/MSP and the Spat library (http://www.cycling74.com/; http://www.ircam.fr/logiciels.html). Recent developments in the MPEG-4 standard (Gallo and Tsingos, 2004) now include the Spat directives and lead the way to a more general format for real 3D audio rendering. For all systems, the synchronization of real-time audio and video is an important issue (Savioja et al., 1999). With optimized instructions (MMX technology), real-time audio signal processing is now feasible. In this case, the system is scalable and will easily benefit from future hardware improvements, while the Intel Signal Processing Library (http://developer.intel.com/vtune/perflibst/spl/index.htm) provides an API for many functions needed for audio signal processing. Research efforts towards optimization of audio rending have even included the development of audio rendering through the GPU, to take advantage of current powerful parallel processing graphics processing units (Gallo and Tsingos, 2004).

Other available APIs include:

- OpenAL, cross-platform 3D audio API. Supports all the major operating systems and platforms including the most important game consoles. (http://www.openal.org/)
- Microsoft Direct Sound 3D (DS3D) provides almost the same functionality as EAX. It runs only on Microsoft Windows.
- Java 3D API has support for perceptual acoustic simulation and it is part of the Java programming platform.

- Virtual Sound Server (VSS) developed at National Center for Supercomputing Applications (NCSA). The main focus of VSS is VR applications. (http://www.isl.uiuc.edu/Software/software.htm)

Examples for research prototypes of interactive multimodal virtual environments are the DIVA system (Savioja et al., 1999) and the SCATIS system (Blauert et al., 2000): both rely on a distributed architecture using several workstations connected by a LAN; the SCATIS system also uses dedicated DSP hardware.

Below we mention few VR systems that support perceptual acoustic simulation. The list is not complete, but contains most important VR systems with some 3D sound functionality:

- Open SceneGraph (OSG) has a sound systems based on OpenAL. (http://www.openscenegraph.org/)
- Open Inventor from Mercury also introduced sound support since its release 4 based on OpenAL. (http://3dviz.mc.com/support/oiv_doc/index.htm)
- VR-Juggler has a sound system called Sonix, but it supports only directional 3D sound. (http://www.vrjuggler.org/sonix/index.php)

5.3 Haptic Channel

5.3.1 Haptic Characteristics

The key characteristics of haptic devices are mechanoreception, spatial resolution, feeling texture, feeling slip, sensing and control bandwidth, thermoreceptors, tactile feedback, kinaesthetic, proprioceptive, force feedback.

5.3.2 Haptic Devices

Haptic Interfaces are force feedback devices enabling human system interaction via the kinaesthetic and/or tactile sense. According to the classical vision, haptic interfaces can be divided into two main categories: kinaesthetic and tactile displays. Kinaesthesia is the sense that detects body position, weight, or movement of the muscles, tendons, and joints but also of the external forces acting on the user. In fact kinaesthetic sense is reached through to proprioceptive sensors. Kinaesthetic devices act on the human body through the kinaesthetic sensorial apparatus. They are able to exert one or more controlled force interactively and to track the position and eventually the orientation of some parts of human body. Tactile displays are related to the tactile sensation, so that haptic information is locally exchanged at the level of the finger pad and is related to a pressure distribution on the human skin.

5.3.2.1 Kinaesthetic Displays:

Nowadays, kinaesthetic haptic interfaces are available, either in laboratories or commercially, for a wide range of application areas. Most of them make use of impedance control paradigm (except the Haptics Master from FCS robotics with is based on the admittance control paradigm). In order to provide maximum performance, they are adapted to the task performed. For general tool based applications there are the PHANToM devices (SensAble Technologies) developed at MIT and the DELTA Haptic Device (FORCE dimension), among many other ones. More generally, they can be basically categorized as a function of force and workspace requirements as well as type of interaction as follows:

- One locus interaction (point interaction for 3DOF interfaces or solid interaction for 6DOF interfaces) :

- **Desktop Interfaces:** Interfaces in this category are developed for desktop application like Virtual Sculpture or Computer Aided Design. They are intended to be used with a workstation and act as a 6 DOF mouse with force feedback. They are used with elbow or wrist support and their workspace is limited (~50 to 100mm), as well as their force capacity (~5 to 10N).\
- **General Haptic Interfaces / Hi Fidelity Teleoperation:** Interfaces in this category are intended for general purpose haptics and teleoperation applications requiring high precision and sensitivity (like tele-surgery). They are intended to be used with elbow support (workspace ~150 to 300mm) and stylus handle to improve precision and their force feedback is limited to finger capacity (~10 to 20 N).
- **Scale One Interfaces / Remote Handling:** These interfaces are used in workbenches or large scale environments to virtually simulate real manipulation as close as possible to scale one (considering displacements and forces). They are intended to be used with one or two hands with large forces (power grasp). Forces are however limited due to technology constraints and for safety purposes (~20 to 60N). They exploit movements of arm and forearm (workspace ~300 to 500mm).

- Limb interaction (hand and/or arm) :
  - **Hand exoskeletons:** They are intended to allow natural finger interaction (workspace ~50 to 150mm and force ~5 to 15N). Force is limited due to technology constraints (compactness is particularly stringent) and for safety purposes. For the sake of simplicity, some of these interfaces like the Cybergrasp™ from Immersion Corporation allow only force on finger closure. More complete interaction can be obtained with CEA-LIST or PERCRO hand interfaces (with limitation to 2 or 3 fingers however). Ease of use and adaptability to general public is of particular importance. They can be mounted on general or large workspace haptic interfaces.
  - **Arm exoskeletons:** They are intended to allow natural hand interaction (workspace ~600 to 900mm and force ~20 to 80N). They can also be used for rehabilitation purposes. Here again, force is limited due to technology constraints and for safety purposes. Higher forces are typically used for rehabilitation. As previously, ease of use and adaptability to general public is of particular importance.

All these interfaces are basically ground based. When considering very large workspace environments (e.g. CAVE) or multi user applications, use of wearable devices can be interesting. Such wearable devices can be in the form of wearable exoskeletons. In this case, lightness and compactness are of particular importance. They can be also in the form of specific devices with fewer DOFs allowing simple and compact design.
This classification is illustrated by Figure 1. Several postures are defined as a function of the workspace (standing, seated on a moving chair, on a fixed chair, elbow support, wrist support). Several handles are defined as well as a function of the force feedback (precision handle, power grasp, and two hands manipulation). Note that the aforementioned zones overlap. Moreover, they are some way arbitrary and their borders are fuzzy.

Most of the existing commercial devices are plotted:

- Haption (France): Virtuose 3D 15-25 (general haptics), Virtuose 6D 35-40 (scale one haptics, rehabilitation), Virtuose 6D 40-40 (remote handling).
- FCS (Netherlands): Haptic Master (scale one haptics, rehabilitation).

- Force Dimension (Switzerland): 3DOF and 6DOF Delta Haptic Device (general haptics), 3DOF Omega Haptic device (desktop).
- NOVINT (USA): Falcon (desktop).
- SensAble (USA): PHANToM OMNI and PHANToM Desktop (desktop), PHANToM Premium 1.0, 1.0 High Force, 1.5, 1.5 High Force and 1.5 6DOF (general haptics), PHANToM Premium 3.0 and 3.0 6DOF (scale one haptics).
- Immersion (USA): Cybergrasp (wearable hand exoskeleton), Cyberforce (haptics workstation (general and scale one haptics).
- MPB (Quebec): Freedom 6S and Cubic 3 (general haptics, precision teleoperation).
- QUANSER (Canada): 3DOF Planar Pantograph (desktop), 5DOF Haptic Wand (general haptics).

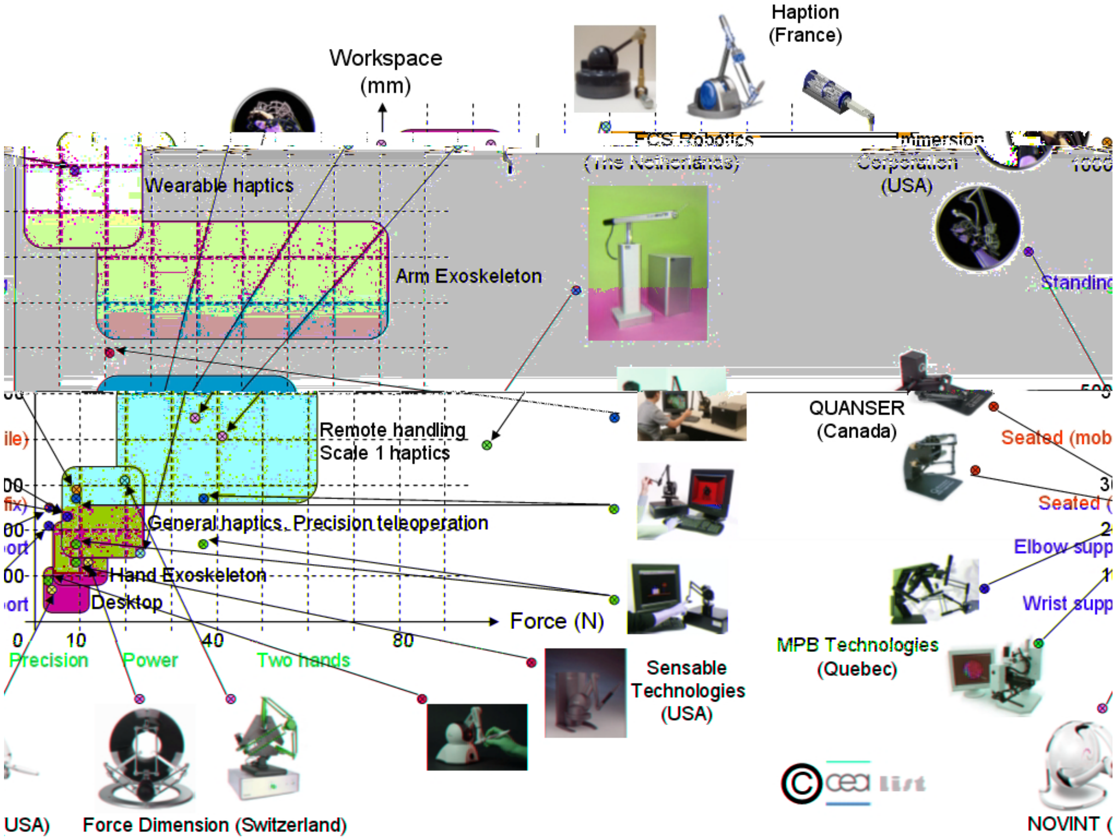


Figure 7 - A classification of existing commercially available haptic interfaces

Figure 7 shows that most companies focus on a particular market and associated applications. In fact, despite all applications share common design drivers (McAffee and Fiorini, 1991; Iwata, 1991; Burdea and Coiffet, 1993; Massie and Salisbury, 1994) whether considering virtual reality or remote handling, a haptic interface must allow free movements in free space and realistic force rendering, which requires on one side a sufficiently large and singularity free workspace, a low friction level (a high back-drivability) and low apparent masses and inertia, and on the other side a sufficient force capacity and a high stiffness), no universal technology allowing to answer all these requirements at the same time exist. Therefore, existing companies develop or use licences and optimize specific technologies adapted to their market.

On the other side, several technologies are developed in laboratories. Figure 2 illustrates as an example state of the art haptic interfaces developed by CEA-LIST (Gosselin et al., 2001; Gosselin et al., 2005) and PERCRO.

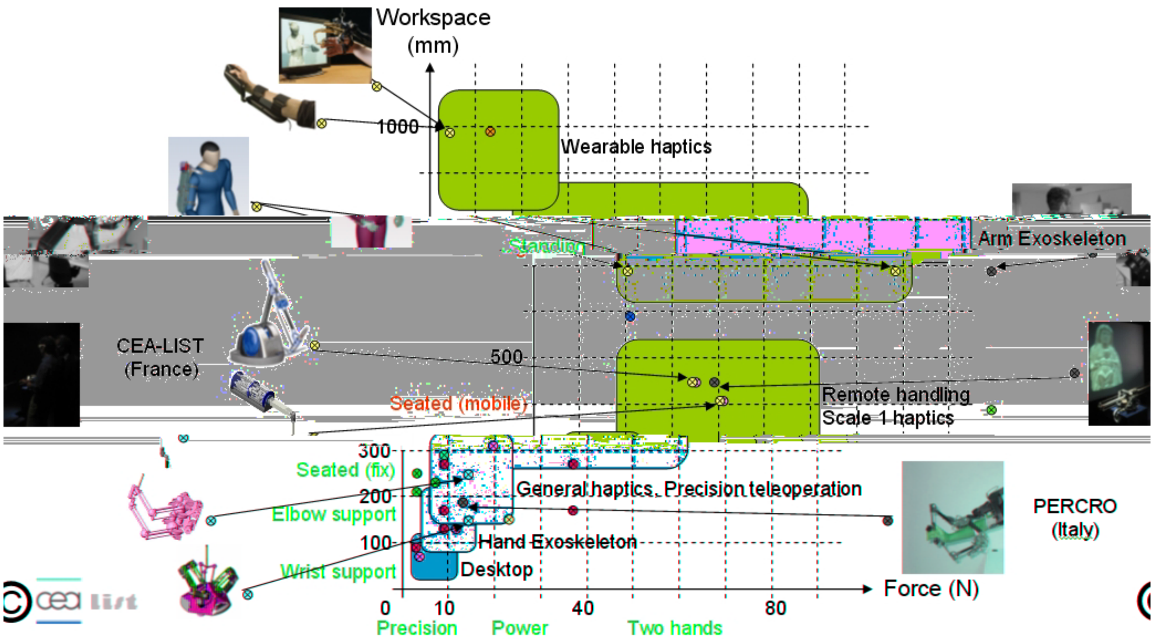


Figure 8 - A classification of some laboratory haptic interfaces

Figure 8 shows that these laboratories have a large experience of haptic systems, comprising serial, hybrid and parallel robots, with restricted or larger workspace, limited or large force capacities, one locus or limb interaction, grounded or wearable devices intended for a large panel of applications. To answer the varied requirements of these applications, several technological solutions have been developed (see Table 3 for a few examples), either relating to the robot architecture and geometry or to the actuation technology or to the interaction paradigms.

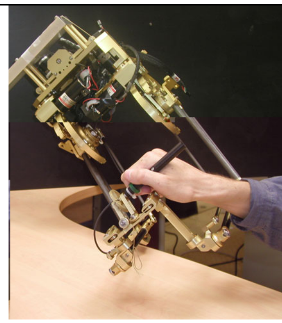
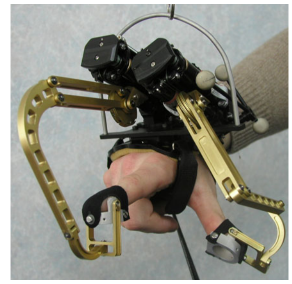
			
Type	One locus Grounded	One locus Grounded	Finger interaction Wearable
Architecture	Hybrid	Parallel	Arborescent
DOFs	Input : 6 Force feedback : 6	Input : 6 Force feedback : 6	Input : 3 (x2) Force : 3 (x2)
Workspace	>200mm and >160°	150mm and ±45°	> $1m^3$ (hand) > finger (robots)
Peak Force	>15N and >0.5N.m	20N and 0.5Nm	>4.74N
Continuous Force	>5N and >0.16N.m	4N and 0.1Nm	>0.8N
Control Stiffness	>5000N/m and >8N.m/rad	-	>5000N/m

Table 3 - Performances of some CEA-List haptic interfaces

Due to this high expertise level, such laboratories are key partners of industry for the improvement of existing devices or the development of new ones for new applications or markets.

5.3.2.2 Tactile Displays

Tactile interfaces are deformable surfaces that communicate information through the sense of touch. Furthermore, a tactile device is a man-machine interface that can reproduce as truly as possible the tactile parameters, such as the texture, roughness, temperature, and shape. Potential applications of tactile interfaces include virtual training for surgeons, remotely touching materials via internet, automotive cockpit applications, active interfaces for the blind and sensory substitution devices.

From the technological point-of-view, tactile stimulation can be accomplished in different ways (Benali Khoudja et al., 2003; Benali Khoudja et al., 2004; Hafez and Benali Khoudja, 2004). The used technologies for virtual environment (VE) systems were inspired from matrix pin-printers technologies and Braille systems for blinds. Due to the strong technological requirement on actuator integration and miniaturisation, these devices are not yet commercially available. Solutions based on mechanical needles actuated by electromagnetic technologies (solenoids, voice coils), piezoelectric crystals, shape memory alloys, pneumatic systems, and heat pump systems based on Peltier modules have been proposed. Other technologies, such as electrorheological fluids which change the viscosity and therefore the rigidity under the application of an electric field, are still under investigations. Technologies dedicated to medical applications such as electro-tactile and neuromuscular stimulators have not yet been used because of their invasive nature. In most of these devices, the complexity of the assembling process increases with the large number of actuators.

As an example, a new concept for such a tactile interface with a high density of micro electromagnetic actuators based on a multi layer approach has been introduced and developed by CEA-List (Hafez and Benali Khoudja, 2004). The system named VITAL (acronym for VIbroTActiLe) is composed of a matrix of flat coils that are independently addressed and which actuate flexible membranes at a specific frequency. This concept allows developing a new generation of tactile matrices with a high density of micro actuators and a quite simple assembling process. The micro-coils can be either machined on a single piece of copper (monolithic structure), or manufactured on a same printed circuit. The micro-coils machined on a single copper piece have the advantage of not being limited in thickness. The coils manufactured in PCB technology have the advantage to reach a great number of turn but can support only small currents. This multilayer design approach allows the complexity of the assembling process to be independent from the number of micro-actuators, allowing developing a new generation of tactile matrices with a high density of micro actuators, a quite simple assembling process and a very competitive price.

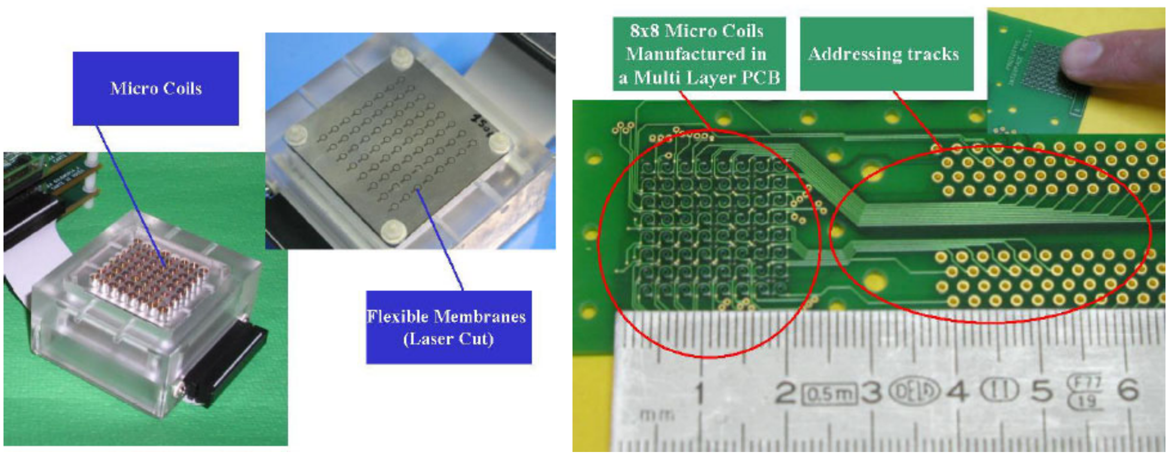


Figure 9 - VITAL interface based on a multi layer approach

5.3.2.3 Applications

5.3.2.3.1 Visuo-Haptic Interface for Hair

Despite significant advances in the domains of real-time animation and photo-realistic rendering of hair, innovative haptic interaction modalities have not been explored extensively by the hair simulation community as a means for intuitive virtual hairstyling. As a contribution to this research area, the team around Nadia Magnenat-Thalmann (2007) has worked on an easy-to-use interactive hair modelling interface. The proposed adaptive visuo-haptic simulation of hair using force feedback haptic devices integrates visual hair simulation and haptic interaction modalities into one multirate-multilayer-multithread application allowing for intuitive interactive hairstyling

5.3.2.3.2 Haptic Simulation, Perception and Manipulation of Deformable Objects

A recent Eurographics Tutorial addressed the challenging research tackled in the domain of haptic simulation, perception and manipulation of complex deformable objects in virtual environments (Magnenat-Thalmann et al. 2007). The tutorial presented state-of-the-art techniques concerning haptic simulation and rendering, ranging from physically based modelling to control issues of tactile arrays and force-feedback devices. Moreover, concrete research results were showed with the presentation of the advanced simulation methods and haptic interfaces developed in the context of HAPTEX, a European research project dealing with haptic sensing of virtual textiles.

5.3.2.3.3 HAPTEX Project

The project HAPTEX combines research in the field of textile simulation and haptic interfaces. Its aims are to provide a VR system allowing for multipoint haptic interaction with a piece of virtual fabric simulated in real-time. HAPTEX has contributed to the achievement of significant advances in the research of textile animation and rendering techniques, as well as in the design and development of novel tactile and force-feedback rendering strategies and interfaces. An overview of these achievements and of future avenues of research is given by (Magnenat-Thalmann et al. 2007).

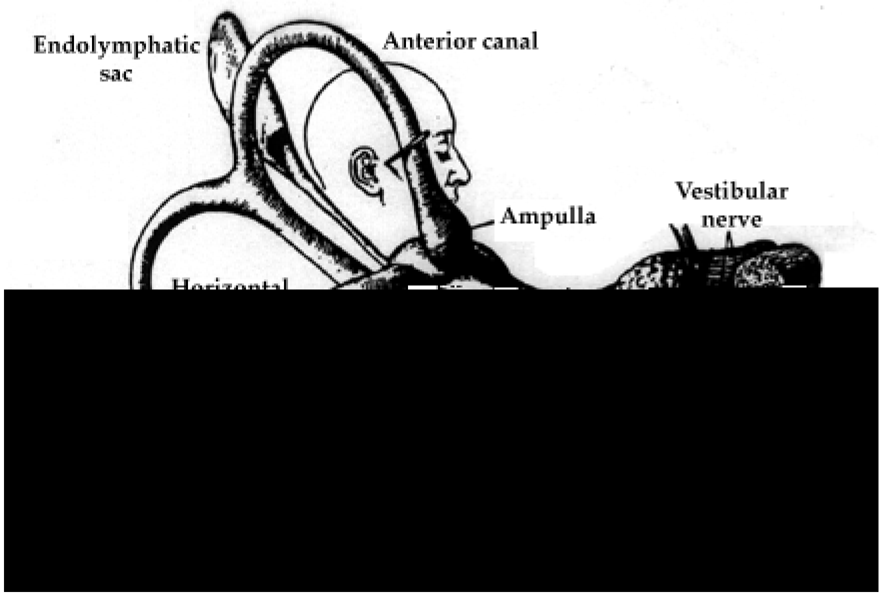
5.4 Vestibular Channel

A vestibular interface is a device able to establish a communication channel between user and computerized environment exploiting the vestibular perceptual apparatus. These devices are able to stimulate the vestibular apparatus of the user reproducing on him the forces generated by his movements or by the movements of the simulated vehicle that he is driving in the Virtual Environment.
A brief description of the vestibular apparatus follows, in order to clarify what is simulated and what is simulable by vestibular interfaces.

5.4.1 Vestibular Characteristics

5.4.1.1 Angular Accelerations

The vestibular apparatus is a small structure that exists in the bony labyrinth of the inner ear (there are two vestibular organs, one in each inner ear) (Howard, 1986). Its function is to sense and generate signal to detect the movements of the head. This function, although basic, is vitally important because it contributes to the coordination of motor responses, eye movements, and posture. In fact, individuals that have had partial or complete loss of vestibular functioning have found it difficult to perform even the most basic of tasks (e.g. standing, walking, or reading). The vestibular organ consists of two principle sets of structures, the semicircular canals and the otolith organs (Figure 10), which work together to provide optimum information on head movement and positioning.



[redacted]

Figure 10 - The Vestibular Apparatus (Howard 1986)

There are three semicircular canals (SCC) (termed the anterior, posterior, and horizontal canals) in each vestibular organ whose function is to detect angular accelerations of the head, acting like biological accelerometers. These canals are bi-directionally sensitive and approximately mutually perpendicular so as to detect angular head movement in any direction. Endolymph fluid fills each SCC and is prevented from passing through the ampula (a widened section of each SCC) by the cupula, a thin flap that stretches across the ampula and acts as a barrier to endolymph flow. When the head is rotated, the force exerted by the inertia of the fluid acts against the cupula of those SCCs that are in the plane of motion; causing it to deflect/bend. (Figure 11)

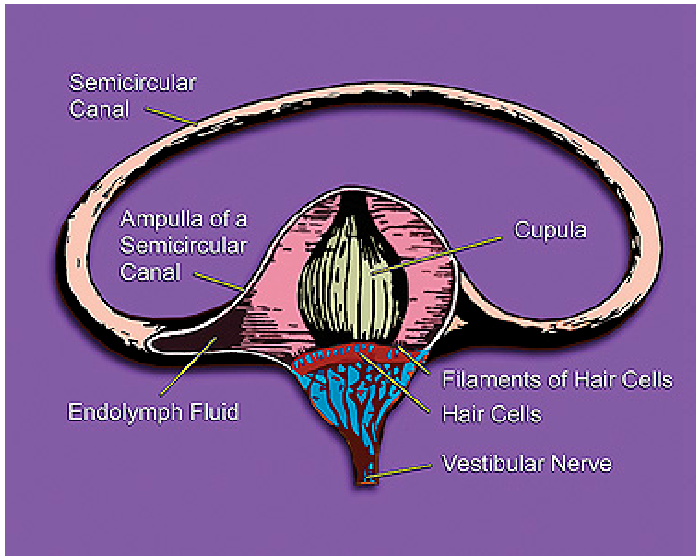


Figure 11 - Semicircular Canal

This deflection causes a displacement of tiny hair cells (located at the base of the cupula in the ampula) which signal this change to the brain via the nerve. For most normal head movements (moderate frequencies), this signal is proportional to head angular acceleration. The receptor system of the SCC can respond to angular accelerations as low as 0.1 deg/sec^2. Also, it is important to note that each SCC transmits a tonic (resting) signal even in absence of motion. This allows the SCC to increase or decrease its response, depending upon whether the head movement is in the direction that the SCC is most sensitive to or in the opposite direction. The brain then integrates information from each SCC pair that occupy the same plane of motion (termed a push-pull pair) to generate an appropriate response for motion in that plane. Between the two vestibular organs, there are a total of three of these push-pull pairs corresponding to three different planes of motion.

5.4.1.2 Linear Accelerations

In addition to the three SCCs, each vestibular apparatus also contains two otolith organs, the utricule and saccule, which sense dynamic changes in linear acceleration (Figure 12) of the head and also provide information on static head position, such as head tilt. These otolith organs are multi-directionally sensitive, as opposed to the bi-directional nature of the SSCs. The receptor portion of these organs contains many hair cells and is called the macula. The macula is covered with a gelatinous substance that contains tiny crystals of calcium carbonite called "otoliths". When the head is tilted or undergoes linear acceleration, the otoliths deform the gelatinous mass, which creates a shear force that excites the receptor cells in the macula. This information is then transmitted via the VIII nerve. The utricule's macula is located in the horizontal plane so as to be sensitive primarily to horizontal linear accelerations, while the saccule's macula is positioned vertically to be maximally sensitive to vertically directed linear accelerations, including gravity.

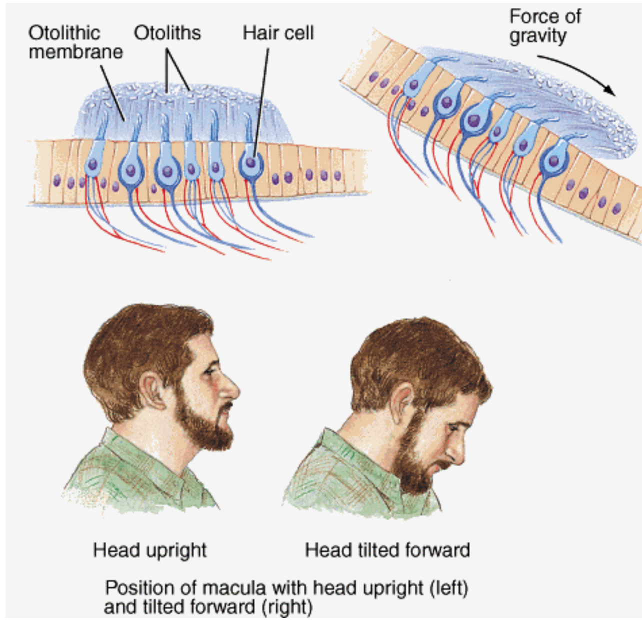


Figure 12 - Working Principle of Linear acceleration sensors

5.4.1.3 Indirect vestibular perception

As described above, the vestibular organs play a major role in the perception of the movements of our body. Anyway the perception of motion is supported also by many other sensorial inputs: sight, audio and also tactile stimulus.
In particular, sight is extremely important in the perception of motion and it is fundamental to perceive the movements at constant velocity or with low values of acceleration. An example of this phenomenon can be seen when we are on a train and have the illusion to move when the train at the side starts.

5.4.2 Working principle

The main issue in the design of a Vestibular Interfaces is to reproduce with high fidelity the forces generated by user movements (accelerations, changing of orientation, etc...) in the Virtual Environment. The interaction takes place thanks to a robotic device able to move, following given trajectories, the body of the user. Since, it is clearly physically impossible to correctly simulate large scale ego-motion in the limited space of a laboratory, there exist some tricks that allow achieving a sufficient level of realism (Von der Heyde and Riecke, 2001).
Most of these tricks make use of some limitation and imperfection that the human sense of motion reveals. One of the most relevant "imperfections", used to cheat the user is the threshold of perceived acceleration, that are the minimum values of acceleration (linear and angular) that the human vestibular system is able to detect. This imperfection can be exploited moving back the robotic structure toward the home position using movements "under the threshold" of perceivable acceleration, gaining additional simulation space. This solution can work until the constant velocity periods exceed respect to the acceleration ones.

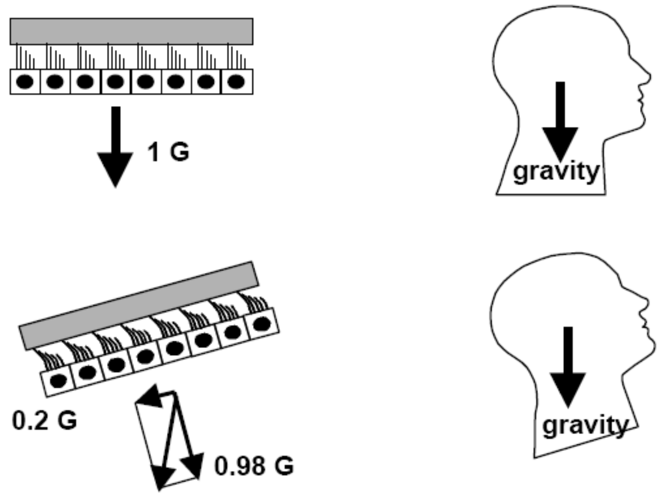


Figure 13 - Continuous Acceleration Simulation

If a continuous acceleration has to be simulated there exists another trick which consists in trading the gravity vector for an acceleration vector (Figure 13). For example, if we want to simulate a linear forward acceleration it is enough to incline the user head backward so that the horizontal (in the user reference system) component of the gravity vector will be perceived as an inertia force.
With a careful mixing of these two tricks it is possible to achieve very interesting results.

5.4.3 Vestibular Devices

5.4.3.1 The Beginning

The first motion base simulator was built in 1931 by Edwin A. Link. It was a very simple flight simulator with just 3 rotational degrees of freedom, clearly completely analogically controlled and actuated by bellows. Despite of its simplicity this system had a very high success and was used widely for pilot training. Since that day Motion Base Simulators have had a great diffusion and, thanks to cost reduction, they have been used in several applications.

5.4.3.2 Morphology

These devices are generally constituted by a real vehicle (or a big part of it) mounted on a robotic platform. The robots used to move the vehicle mock-up can be classified by the number of Degrees of Freedom (DOF). The use of 6DOF Stewart platforms (Boian et al., 2005) is becoming almost a standard because this kind of robotic structure well suits the requirements of high payload, stiffness and fast dynamic responses. Its diffusion lead to the birth of many companies that sell Stewart platform with actuators, sensors and low level control software especially designed for motion simulation.
Stewart platform can be driven by linear electric or hydraulic actuators depending from the required fastness of the dynamic behaviour:

- Hydraulic actuators are stronger and “faster” but on other side they are expensive and require a lot of maintenance.
- On the contrary electrical actuators are “slower” but they are cheaper and require almost no maintenance

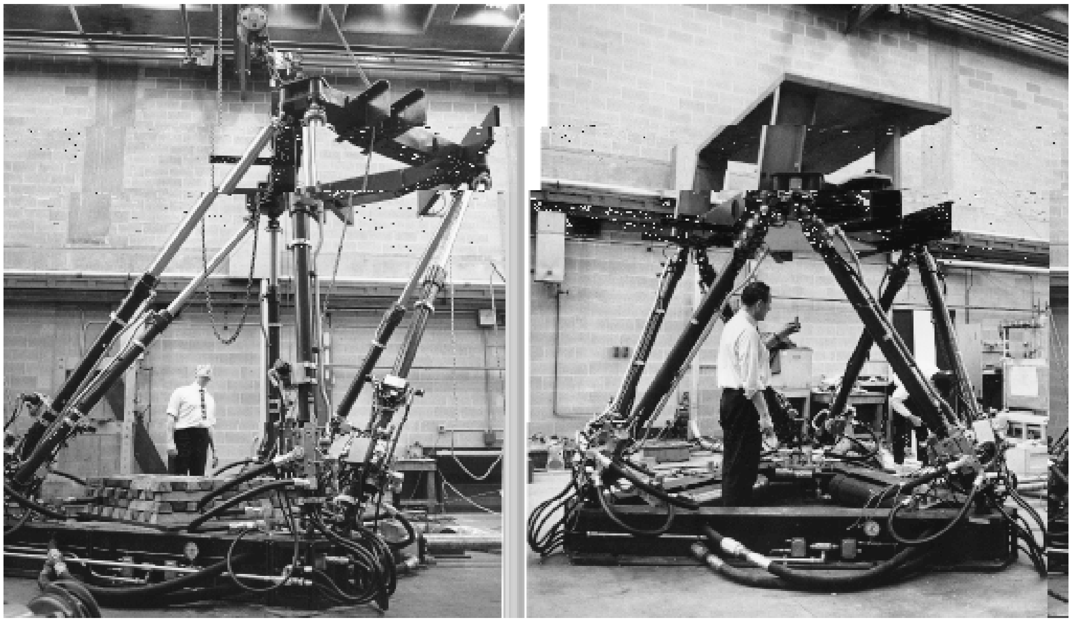


Figure 14 - First hexapod used in a helicopter simulator (mid 60's)

Anyway in many cases there are reduced-complexity Motion Base Simulators that are characterised by simpler custom solution (with 3 or 2 DOF) with fewer DOF.

5.4.4 Overview of Vestibular Systems

5.4.4.1 Grounded vehicle Simulators

As stated above, the origin of the simulators have to be searched among the flight simulators, anyway in the last years the improvement of the control algorithm and the lowering of the cost of electronics and visualisation system lead to the diffusion of motion simulation also in the field of grounded vehicle.
Among the grounded vehicle simulators it's possible to distinguish two categories classified by the application:

- High performance simulators for ergonomic studies and dynamic test;
- Simple and reduced complexity simulators for training and education.

There are a large variety of driving simulators: car, motorcycle, trucks, fork-lifter and many other special vehicles.

5.4.4.1.1 Car

Figure 15 - 6-DOF Spider System for Vehicle Simulator (from FCS Simulator Systems) and Honda Car Driving Simulator

Car simulators didn't achieve a large commercial diffusion. The reason of that stays in the fact that a real car cost only a dozen of thousand Euros so it is not cost-effective to build motion base car simulators for training purpose. The existing simulators are used for ergonomics test by car manufacturer or research laboratories.

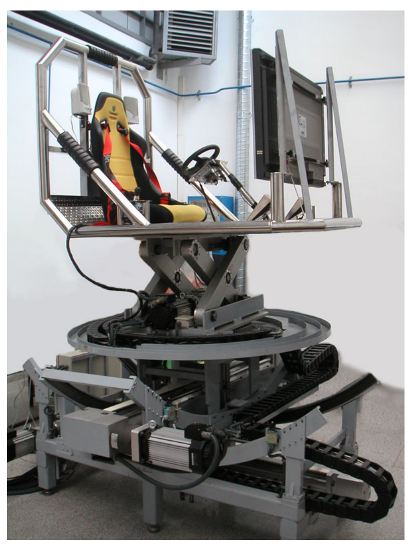


Figure 16 - Reduced DOF example (http://laimuz.unizar.es/simusys/)

In some case there are have been some effort to develop simpler simulators (Figure 16) but their market still remain very restricted .

The project SACARI (Supervision of an Autonomous Car in an Augmented viRtuality Interface) aims at developing an interface to remotely control a car. Two methods have been chosen to drive the car: either the user drives it in real time ("distant driving" mode), either he indicates where the car has to stop, and in which direction and the application orients the car toward the right position ("supervision" mode). A camera is placed in the car to visualise its position and enable the distant driving mode (Tarault et al., 2005) (French - http://www.limsi.fr/Recherche/ActionVenise/Sacari_presentation.html).

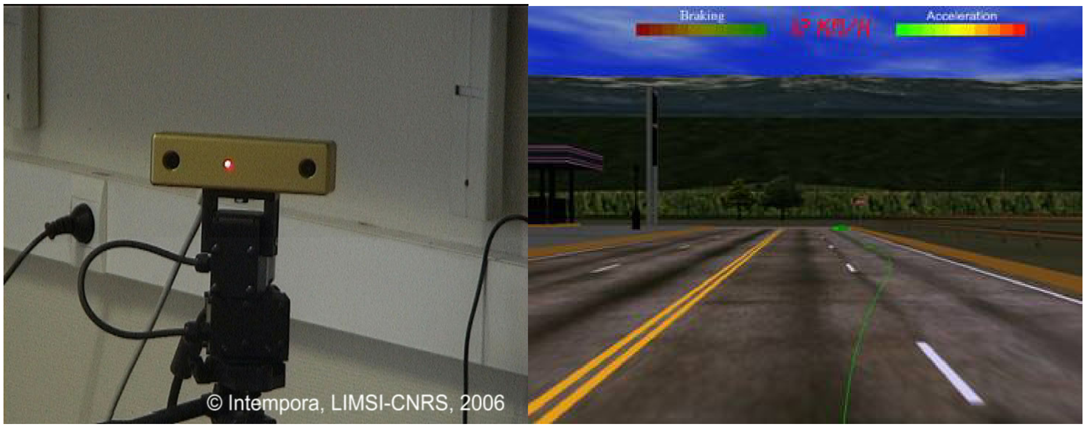


Figure 17 - Exemple of embedded robot and interface for the "supervision" mode.

In an effort to make car simulators more accessible, the SIRCA (SImulador Reactivo de Conduccion de Autómoviles) research project aims at using VR technologies (head-mounted devices) to develop small and medium-size subject-oriented simulators. The project concentrates on the simulation of the surrounding elements, such as the traffic. Another important topic is the synchronisation and proper activation of a group of elements which defines a driving situation; this is called driving scenario. The outcomes of this project include the definition of traffic scenarios data structures and tools to easily define new scenarios, as well as the definition of an architecture that enables the simulator to be run on an average PC. That research led to the development of two products: SIRCA Driving Simulator and TUTOR professional Driver Simulator (S. Bayarri et al., 1996; Coma et al., 2001; Coma et al., 2005).

5.4.4.1.2 Trucks

A larger commercial diffusion have been achieved by trucks MBS. These kinds of devices are mainly used for training drivers in special driving conditions and for psychophysical studies.

Figure 18 - Truck Driving Simulator

5.4.4.1.3 Motorcycle

Figure 19 - Morris simulator by PERCRO

Among the grounded vehicle the motorcycle simulators have quite a lot of diffusion but only in a very simplified version for entertainment. Some experimental setup has been realized especially for studying the ergonomics and equilibrium of the dynamic of motorcycles (Figure 19). The aim of such vestibular interface is to reproduce with high fidelity the dynamic behaviour of the vehicle in order to test its ergonomics.
Other simplified system were proposed by Honda at reasonably costs for training application and to be used a test device for giving driving license to young drivers (Figure 20).

Figure 20 - Reduced DOF Honda Motorcycle Simulators

5.4.4.1.4 Special Vehicle

There exists a large quantity of special designed simulators for particular vehicle such as excavators, crane, shovels etc. The systems are used for personnel training and as a tool for the design of new products by the producer. Mostly these kinds of devices are specially designed for the application and they are able to give only a very simplified vestibular feedback.

(Provider of car, truck, ship and crane simulators: http://www.shipanalytics.com/STS/default.asp)

5.4.4.2 Flight Simulators

A special attention has to be dedicated to flight simulators. The first vestibular interfaces were developed in this field of application. The requirement of cost effectiveness in this case is obviously satisfied. Flight simulators are mainly diffused for test and training of pilots. Many flight simulators receive a special certification that allows using them for testing pilots and assigning them flight licenses. There are many huge systems (Figure 21) that allows simulating with high fidelity most of the standard flight manoeuvres.

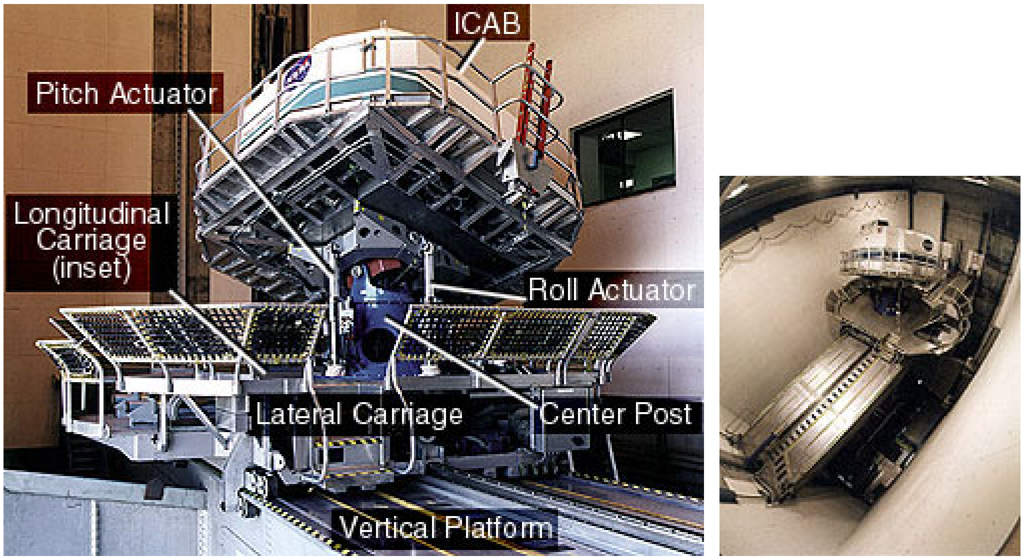


Figure 21 - VMS Motion Base Flight Simulator

(Provider: http://www.cae.com/www2004/Products_and_Services/index.shtml)

5.4.4.3 Locomotion Interfaces

The newer develops in vestibular interfaces comprehend the so called Locomotion Interfaces (LI).
LI were in the past devices able to allow simulating a very large Virtual Environment in the small space of a room. They were constituted by mono or bi-directional treadmill or special robotic arm that moves under the user feet (Figure 22).

Figure 22 - Examples of Locomotion Interfaces without vestibular feedback

The newer devices (Christensen et al., 2000) are able to reproduce also some vestibular cues perceived when moving by foot. The inertial cues are given by a robot arm attached to the user body and a variable slope treadmill. As for the vehicle simulators rapid variations of acceleration are simulated by applying a proper force directly on the user and slow variations or constant accelerations are simulated by tilting the treadmill under the user feet (Figure 23).

Figure 23 - Locomotion System with Vestibular feedback developed by Prof. Hollerback

The Cybersphere (Figure 24) was developed by the University of Warwick and VR Systems UK. It consists in a spherical projection system that comprises a large, hollow, translucent sphere, 3.5 metres in diameter, supported by means of a low pressure cushion of air. Rotational movements of the sphere are measured by means of sensors. An observer makes the sphere rotates by entering in it and walking.
Images are projected upon the surface of the large sphere by means of high power projectors. Four such projectors are mounted on the surrounding walls; a further two are mounted on the front and back walls. A further projector is mounted on the ceiling. Each projector projects an image, generated by a computer, onto the outer surface of the sphere. The surface of this sphere is prepared in a way such that the enclosed observer is able to view the projected images clearly. The combination of the images from each projector provides a fully immersive visual experience for the observer (http://www.warwick.ac.uk/news/pr/230).

Figure 24 - Cybersphere settings

6 Input Channels

6.1 Tracking Systems

6.1.1 State of the Art on Tracking Systems

Tracking systems are used in virtual reality where the orientation and the position of a real physical object are required. Specifying a position in an environment requires the Cartesian coordinates x-y-z with respect to a given reference coordinate system. However, many applications manipulate solid objects and this requires the orientation to be defined by three angles know as pitch (elevation), roll and yaw (azimuth). Thus, 6 DOFs are required to fully describe the position and orientation of an object in an environment. Indeed, VR applications need to know the 3D position and the orientation of users' head, body and limbs in order to enable a realistic interaction between users and the Virtual Environment. For example, the head pose parameters are necessary to compute an exact stereoscopy images (Cruz-Neira et al., 1993) and perform hand-free navigation (Bourdot and Touraine, 2002). Another example concerns arms and hands parameters used in the capture and analysis of sign language (Braffort et al., 1999), as well as in grasping and pointing tasks (Su and Furuta, 1993).

These parameters are measured using various 3D tracking devices sensing electromagnetic, ultrasonic or visible optics waves. These devices are based on technologies that are often intrusive and constrain the user's movements. For a complete survey see (Mulder, 1994).

Recently, 3D tracking devices based on the detection of infrared reflective markers have gained significant interest in VR applications, primarily because they are less intrusive than previous solutions. Indeed, infrared devices do not need to be physically connected with the tracked object. Visible optics trackers have the same characteristics as infrared ones. However, strong constraints on environment lighting intensity and weak acquisition rates considerably restrict their use at present.

Infrared devices are well adapted for the tracking of users in immersive environment. However, in the case of interactive applications, real-time response is essential. For instance, when a user moves his head, stereoscopic image rendering parameters must be adjusted instantaneously. Thus, active research in IR tracking aims to optimise the computations involved in the tracking process (Boulic et al., 2000). For example, Magneau et al. (2004) proposes:

1. A new model for the positioning of markers on the objects to track, which uses relative positions of the markers on the objects ;
2. A new algorithm based on the identification of markers to manage potential occlusions without increasing the number of cameras.

Actually, both positioning model and identification algorithm share a common functional module (Magneau et al., 2002). This module aims to compute the probabilistic matrix that indicates the similarity between two lists of markers. In the positioning model, lists are identical and correspond to the markers on the object. In the identification stage, the first list contains the markers on the object and the second one the markers actually detected in 3d space.

A set of characteristics must be taken into account to evaluate the performance of many tracking systems that are available today (Bhatnagar, 1993; Holloway and Lastra, 1993). These characteristics are listed below.

- Update rate: rate at which position and orientation measurements are reported by the tracker to the computer. Low update rates lead to unconvincing virtual worlds.
- Latency/Lag/Delay: time elapsed from the moment a user makes a movement in the real world until the tracker data are received by the computer. If the lag is significant, this characteristic contribute to cyber-sickness since the computer will not immediately update the data from the head tracker into display.
- Accuracy: amount of error in the measurement. Usually given as a bound on the magnitude of the error or as an average error amount. A tracker that has an accuracy position of 1.0 mm will report positions that are $\pm$ 1.0 mm from the actual position.
- Resolution: smallest change in position and orientation that can be detected by the tracker. A movement smaller than the tracker's resolution will not be reflected in its output.
- Interference/Distortion: All trackers except for inertial systems are subject to either interference or distortions which reduce the accuracy and may produce gross errors.
- Absolute/Relative: Trackers may report about absolute position/orientation respect a unique coordinate system or just send information on changes from the last position (cumulative error).
- Range: Working volume within which the tracker can measure position and orientation with its specified accuracy and resolution.
- Size/Weight: A small size/weight provides major facility and comfort to measure the desired part of body.
- Robustness: The dependence of tracker respect to environmental factors.
- Degrees of freedom (DOFs): Number of independent variables used by the tracker to report to the computer about the type of movement. These movements may be tracked through translation or rotation about the x-y-z axes. Because of these movements are mutually orthogonal, there are six degrees of freedom.
- Safety: provides information about healthy risks when the tracker is used a long-term.
- Wired/Wireless: the use of wired/wireless sensors rebounds to the comfort and freedom of movement.

6.1.1.1 Mechanical Tracking Systems

Mechanical tracking systems measure the position/orientation of an object that is attached to the end of a movable mechanical arm. The arm is joined at a fixed point of reference and is made up of several sections that can rotate and move at the joints. The rotations and movements are measured and used to compute the position and orientation of the object relative to the fixed point of reference. These measurements, coupled with the knowledge of the linkage geometry, allow a very accurate tracking (Bhatnagar, 1993).

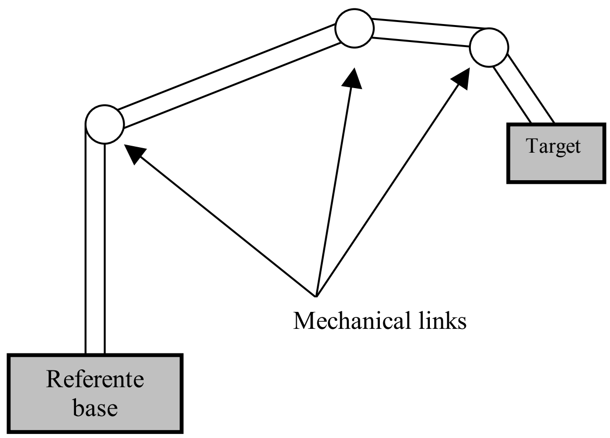


Figure 25 - Mechanical tracking system

- Advantages:
  - High accuracy and resolution
  - Low lag
  - No line of sight (LOS) necessary
  - No magnetic interference problems
  - Good for accurate tracking in small volumes

- Disadvantages:
  - Heavy
  - Calibration necessary
  - Very small working volume

6.1.1.2 Magnetic Tracking Systems

Electromagnetic tracking systems work by measuring the strength of the magnetic fields generated by sending current through three small wire coils, oriented perpendicular to one another. These three coils are embedded in a small unit that is attached to the object which will be tracked. The current has the effect of making each wire into an electromagnet while the current is flowing through it. By sequentially activating each of the wires, and measuring the magnetic fields generated on each of three other perpendicular wire coils, it is possible to determine the position and orientation of the target object (Bhatnagar, 1993; Holloway and Lastra, 1993; Rolland et al., 2000).

6.1.1.2.1 Polhemus

This type of magnetic tracking system operates by alternating-current (AC) magnetic field. Transmitter contains three orthogonal coils that emit a magnetic field when current is passed through them. The receivers also contain three orthogonal coils in which current is induced by the changing magnetic field of the transmitter. Current is supplied to one transmitter coil at a time, and three readings are given by the receiver coils, leading to nine measurements for each measurement cycle. These measurements are then

processed in the control unit to compute the 6-DOF solution. Because the AC field induces eddy currents in conductive metals, tracker should be used in an environment free of metallic objects.
http://www.polhemus.com/

6.1.1.2.2 *Ascension*

This type of magnetic tracking system operates by direct-current magnetic field. Transmitter emits a series of short DC pulses. After eddy currents in surrounding metallic objects have decayed, measurements are taken at receivers. Background magnetism (such as the Earth's magnetic field) is subtracted off of the measurements and transformation is calculated. Advantage to this approach is that it is less sensitive to conductive metals which allow the tracker to be used closer to metal than AC trackers.
http://www.ascension-tech.com/

- Advantages:
  - Inexpensive
  - High accuracy and resolution.
  - Low lag
  - No LOS problems
  - Good noise immunity
  - Receivers are generally small

- Disadvantages:
  - Distortion/Interference due to metallic objects
  - Electromagnetic interference from radiation
  - Accuracy diminishes with distance
  - Size of transmitter
  - Small working volume

6.1.1.3 Acoustic Tracking Systems

Acoustic tracking systems use ultrasonic sound waves ($\geq$ 20Khz) for measuring the position and orientation of the target object (Bhatnagar, 1993). There are two ways of doing this which are shown below.

6.1.1.3.1 *Time-of-flight (TOF) tracking*

It works by measuring the amount of time that it takes for sound emitted by transmitters on the target to reach sensors located at fixed positions in the environment. A typical system might consist of three transmitters and three sensors (receivers). The transmitters emit sounds at known times. By measuring when the sounds arrive at the sensors, the system can determine the amount of time it took for the sound to travel from the target to the sensors, and thereby calculate the distance from the target to each of the sensors. These distances and the known positions of sensors constitute sufficient information for computing the position and orientation of the target object. In order to find position, only one of the transmitters is needed. Orientation is determined by the differences in location indicated by these calculations for each of the three sensors. This method provides a low update rate because of the low speed of sound in air. Furthermore, the speed of sound in air depends by environmental factors as temperature, barometric pressure and humidity.

6.1.1.3.2 Phase coherence (PC) tracking

It works by measuring the difference in phase between sound waves emitted by a transmitter on the target and those emitted by a transmitter at some reference point. The phase of a sound represents the position on the sound wave, and is measured in degrees (360 degrees is equivalent to one wavelength difference). As long as the distance travelled by the target is less than one wavelength between updates, the system can update the position of the target. By using multiple transmitters, orientation can also be determined. This method is subject to error accumulation as it works by periodic updates of position rather than by measuring absolute position at each time step.

- Advantages:
  - Inexpensive
  - High accuracy and resolution
  - No magnetic interference problems
  - Lightweight

- Disadvantages:
  - Low update rates (Time-of-flight tracking)
  - Ultrasonic noise interference
  - Dependence of environmental factors
  - Echoes cause reception of “ghost” pulses
  - LOS necessary
  - Small working volume
  - Cumulative error (Phase coherence tracking)

6.1.1.4 Optical Tracking Systems

Optical tracking systems work by observing marked locations on each tracked object from multiple angles of view, then triangulating the lines of sight to the targets to calculate each target's location. The 3D location of at least three targets is needed to calculate each object's position in space relative to the camera. The configuration of three or more targets attached to an object to allow it to be tracked is called a marker.

In the early years, tracking systems used infrared light emitting diodes as targets, observed by infrared-only light sensors. However, these systems provided the power either via movement-limiting wires or via batteries which require periodic replacement and it represented a major drawback. Next generation of tracking systems now use infrared light from a ring surrounding each infrared camera lens and use balls as targets which are coated with a retro-reflective material in order to mirror the light back to the lenses. These systems eliminate the need for wires to bring power to the targets.

Nowadays, the trackers use available light in the visible spectrum to detect and pinpoint high-contrast targets. The images are captured and processed in the computer (CT, 2005).

Depending on the sensors, optical trackers can be divided into two categories: inside-out and outside-in systems (Bhatnagar, 1993; Holloway and Lastra, 1993; Rolland et al., 2000; Ribo, 2001).

6.1.1.4.1 Outside-In

Sensors are mounted at a fixed location (position and orientation) in the scene. Objects to be tracked are marked with passive or active landmarks. The number and the shape of the landmarks needed on each object depend on the number of degree of freedom (DOF) with which each object is to be tracked. Additional landmarks can be used to provide redundancy in order to improve pose estimation and to overcome occlusion problems.

6.1.1.4.2 Inside-Out

These systems determine position and orientation by means of sensors which are attached directly to the object to be tracked. These sensors observe the scene which is marked with landmarks. The number and the shape of the landmarks can be chosen in the same way as for outside-in systems.

- Advantages:
  - High availability
  - Large working volume
  - Low lag
  - No magnetic interference problems
  - High accuracy and resolution

- Disadvantages:
  - LOS necessary
  - Limited by intensity and coherence of light sources
  - Weight
  - Expensive

6.1.1.5 Inertial Tracking Systems

The principle of inertial sensing is based on the attempt to conserve a rotation in the case of a mechanical gyroscope or a position in the case of an accelerometer.

Orientation of the object is computed by jointly integrating the outputs of the rate gyros whose outputs are proportional to angular velocity about each axis. Changes in position can be computed by double integrating the outputs of the accelerometers using their known orientations (Rolland et al., 2000).

- Advantages:
  - Large working volume
  - Low lag
  - No LOS problems
  - No magnetic interference problems
  - Unnecessary two components: transmitter-receiver
  - Small size
  - Inexpensive

- Disadvantages:
  - Cumulative error
  - Drift
  - Not accurate for slow position changes

6.1.1.6 Commercial Tracking Systems

A classification of commercial tracking systems is shown in the following table. The information about tracking systems has been provided by (Youngblut et al., 1996) and Engineering Systems Technologies (EST, 2005).

Technolo	Product	Vendor	DO	Update ra	Latency	Resolution	Working volum	Price
Mechani	BOOM3C	FakeSpace, Inc.	6	>70 Hz	200 ms	0.1º	6 ft diameter, 2.5 ft high	95000$
	PUSH	FakeSpace, Inc.	6	>70 Hz	200 ms	0.1º	2 ft diameter	45000$
	ADL-1	Shooting Star Technology	6	240 Hz	0.35-1.8 n	0.25 in 0.15º - 0.3º	35 in diameter, 18 in high	1299$
	WrightTrac	Vidtronics, Inc.	6	300 Hz	3.3 ms	0.1º per axis	1/4 sphere, 40 in diameter	795$
Magneti	Fastrak	Polhemus	6	120 Hz / number of receivers	4 -8.5 ms	0.0002 in/in, range 0.025º	10 - 30 ft	6050$
	Isotrak II	Polhemus	6	60 Hz / number of receivers	23-45 ms	0.0015 in/in, range 0.1º	5 ft	2875$
	InsideTrak	Polhemus	6	60 Hz / number of receivers	12 ms	0.0003 in/in, range 0.03º	5 ft	999$
	Ultratrak	Polhemus	6	30 – 60 Hz	12 ms	0.05 in at 5 ft 0.25 ft at 15 ft 0.1º RMS	2 - 15 ft	23250-32250
	Ultratrak 120	Polhemus	6	60 - 120 Hz	12 ms	0.15 in at 5 ft 0.25 ft at 15 ft 0.1º RMS	2 - 15 ft	39500-71500
	Liberty	Polhemus	6	240 Hz	3.5 ms	0.00015 in at 12 in range 0.0012	36 in. at above specifications	6663€
	Patriot	Polhemus	6	60 Hz	17 ms	0.0015 in/in of source and senso separation range 0.1º	5 ft	2083€
	Flock of Birds	Ascension	6	144 Hz	7.5-47 ms	0.5mm at 30.5cm 0.1º at 30.5cm	3-8 ft	2695$ basic system
	3D Navigator	Ascension	6	120 Hz	Not given	0.08 cm at 1.52 m 0.2° at 3.05 m	± 3.05 m in any direction with on transmitter	30417€
	SpacePad	Ascension	6	120 Hz / number of receiver	< 8 ms	Not available	16 x 16 ft	1246€
	microBIRD	Ascension	6	90 Hz	Not given	0.05 mm at 30.5cm 0.1° at 30.5cm	20-71 cm X ± 30 cm Y, Z.	Not availabl
	miniBIRD 800	Ascension	6	120 Hz	Not given	0.5mm 0.1° at 30.5 cm	±76.2cm in any direction	3329€
	MotionStar	Ascension	6	144 Hz	Not given	0.25cm at 3.05m 0.2° at 3.05m	±3.05m in any direction with on transmitter	18842€
	Nest of Birds	Ascension	6	105 Hz	Not given	0.05 mm at 30.5cm 0.1° at 30.5cm	± 91.44cm m in any direction	4913€
	pciBIRD	Ascension	6	105 Hz	Not given	0.05 mm at 30.5cm 0.1° at 30.5cm	±76.2cm in any direction	2829€
	pcBIRD	Ascension	6	144 Hz	10 ms	0.05 mm at 30.5cm 0.1° at 30.5cm	±1.22m in any direction	2063€

Optical	SELSPOT II	Selcom AB	6	10000 Hz	Not giver	0.025% of millirad	Up to 200 m	29980$
	OPTOTRAK 3020	Northern Digita Inc.	6	600 Hz	Not giver	0.01 mm at 2.2 m	Not given	57400$
	MacReflex	Qualisys, Inc.	6	50-200 Hz	Not giver	Not given	0.5 - 30 m (in), 0.5 - 9 m (out)	38500-4850C
	DynaSight	Origin Instruments Corporation	3	64 Hz	16-31 ms	0.1 mm cross range 0.4 mm down range	0.1-1.5m and 1-6 for 7 mm target	2195$
	RK-447 Multiple Target Trackin System	ISCAN, Inc.	6	60 Hz	16 ms	Not given	Not given	36800$
	Laser BIRD 2	Ascension	6	240 Hz	7.2-11.3 n	0.1. mm at 1m 0.05º at 1m	0.15-1.83 m	16454€
	ReActor 2	Ascension	6	900 Hz	Not giver	Not given	3.0 x 4.2 x 2.4 m (LWH)	73333€
	PPT	WorldViz	3	60 Hz	<20 ms	< 1 mm	10 x 10 x 10 m	8250€
	ARTtrack1	A.R.T	6	60 Hz	Not giver	Not given	10 m	Not given
	Polaris	Northern Digita Inc.	6	60 Hz	Not giver	Not given	Not given	Not given
	HiBall-3000	3rdTech, Inc.	6	2000 Hz	<1 ms	0.2mm RMS 0.01° RMS	40' x 40' x 10'	Not availabl
Acoustic	Head/Hand XYZ Tracker	Fifth Dimensior Technologies	3	20 Hz	Not giver	Not given	Up to 3 m	345$
	GP12-3D (Freepoint 3D)	Science Accessories Corp.	3	150 Hz/number of emitters	Not giver	0.002 in	3.25x3.25x3.25 f Up to 16x8x8 ft	4995-6995$
	Logitech 3D Mouse and Hea Tracker	Logitech	6	50 Hz	72 ms	1/250 in (linear) 1/10° angular	5 ft long, 100° cor	Not given
Inertial	MotionPak	Systron-Donner	6	60 Hz	Not giver	0.004°/s	Not given	10000$
	GyroPoint	Gyration, Inc.	3	Not given	Not giver	0.2º	75 ft	299$
	3D BIRD	Ascension	3	160 Hz	15 ms at 360º/s	0.1º at 30.5 cm	±180° Azim-Roll ±90° Elevation	1246€
	InertiaCube 2	InterSense	3	180 Hz	2 ms	0.01º	360º	1413€
	InertiaCube 3	InterSense	3	180 Hz	2 ms	0.03º	360º	1663€
	IS-300 Pro	InterSense	3	500 Hz	Not giver	0.02º	360º	3646€
Acoustic Inertial	IS-900	InterSense	6	180 Hz	4 ms	0.25° - 1.00°	3.0 x 3.0 x 3.0 m	9,950$ (wired) 36,600$ (wireless)

Table 4 - Tracking systems

6.2 Input Devices

VR installations allow or even require in contrast to the classical desktop a big variety of interaction metaphors and devices. The type of interaction metaphor and device is much depending on the concrete application than in the desktop case.

Difference between VEs and the desktop:

- The table is missing. Users do not have a base which guides the hand and arm. Any interaction gets more imprecise than in the desktop case.
- VEs require interaction with 6 degrees of freedom.
- There is typically no standard keyboard and mouse.

Styles of input devices:

- Discrete input devices: Any device which is producing button signals
- Continuous input: Gesture based devices like cyber gloves, joysticks, speech recognition, touch pads, etc.
- Hybrid devices: Combination of both discrete and continuous interaction into a single device. This is the mostly used style.

6.2.1 Text Input

Text input is the most difficult type of interaction in VEs because typically a keyboard is missing. The following devices are used for text input in VEs:

- Standard keyboard
- Chord keyboard
- Contact glove keyboard
- Gesture glove keyboard
- Virtual keyboard operated with a pointer
- PDA based handwriting recognition

6.2.2 Graphic Input

Graphic input is required within VEs for operations such as selection, navigation and menu handling. The available devices are in a broad sense mouse like devices. Interaction is as indirect as in the case of a desktop mouse. The “mouse” pointer is represented as a 3D pointer in the virtual world. The device is used to move the pointer and select virtual objects or 3D menus. The following categories of devices are used for this task:

- Arrow keys: Can be mounted on pointing devices and are then a suitable device for menu handling.
- Joystick: Especially mini joysticks mounted on a pointing device are a suitable device for tracking decoupled menu handling and navigation.
- Trackballs: Popular device for navigation and menu handling. Mostly build in mouse like devices.
- Game pads: Very popular cost efficient devices. Many users are familiar with the operation. Intuitive and fast operation. Some devices are simulating tactile feedback also.
- 6 DOF Devices (Hornet, Dragonfly and bug, Mike OC3, etc.): The classical interaction devices for VEs. These devices are typically combined with buttons, mini joysticks, wheels, etc. Wireless devices designed for use with optical tracking systems are the future. They offer non intrusive and simple 6 DOF interaction methods. If button or joystick signals are required the respective electronics and a battery must be built in.
- Desktop wireless mouse: Useful input device if only the button signals are needed.
- Gyration mouse: Like the desktop mouse but the direction information is produced out of movements. Therefore it is suitable for navigational task as well.
- Trackpads: Very popular device taken over from notebooks. Very efficient for navigation.

- Touch screen: Useful for complex application handling with many menu options and complicated menu structures.
- 6D control - Spacemouse/spaceball: Simulates 6 DOF interaction at the desktop. Very popular devices in the field of CAD systems. Suitable device for non-immersive workplace systems.
- Eye tracking: Interesting technology for navigation and selection tasks. The disadvantage is the need for wearing glasses.
- Bioelectric control: Very experimental.

Graphic input devices are used to select command options from menus and forms presented by a graphic user interface, to point graphic entities presented in 2 or 3 dimensions, and to perform some graphic editing actions on the screen. Graphic input devices allow users to control a cursor or pointer on the screen in 2D areas or 3D environments. With respect to 3D environments, users can manipulate the selected objects changing their position, orientation, or both (6DOF – degrees of freedom).
Various kinds of graphic input devices have been developed over the recent years as a result of the trend in user interfaces evolving from text-based to graphics oriented. Most of the current developments in the area of input devices take advantage of the progress in computing power and miniaturisation. Different types of graphic input devices available on the market are described next.

6.2.2.1 Arrow Keys

On most computers, a keyboard is the primary text input device. The earliest personal computers were almost entirely text-based and had little graphic capability. Arrow keys located on the keyboard were used for pointer control. There are various methods for interaction with arrow keys, the most common of which is step keys. In a graphic environment, step keys move the pointer one or more pixels, the size of the step often changeable by either another set of keys, or as a function of the duration of the key press. Jump keys are designed for a more specific type of environment. Jump keys move the cursor to one of a series of predefined areas on the screen but they are rather commonly used in text-based applications. While arrow keys are neither fast nor efficient for general graphic interaction, they do make a good backup pointer control.

6.2.2.2 Joystick

A joystick is a vertical rod mounted on a base with one or more buttons used for selection. Three types of joysticks can be distinguished:

- An isotonic joystick, often referred to as a displacement joystick, provides an output that is proportional to the displacement of the handle from the normal position. This is translated to a magnitude and direction.
- An isometric joystick – often referred to as a force joystick or a pressure joystick – is a lever that does not move when force is applied to it. The output of the isometric joystick is a function of the amount of force applied to it. A derivation of the isometric joystick is a track-point. This device is a small-sized joystick embedded in a board and operated with the thumb or index finger. Since it takes up only around a square centimetre of workspace, the forces applied to the device need to be amplified by a large amount to be turned into pointer motion, which makes the usage difficult for non-experienced users.
- A switch activated joystick uses a set of switches, evenly spaced around the handle. When the handle is moved, the direction of the movement is determined based on the activated switches. The magnitude of the motion is not gauged.

Joysticks are the best solution for navigation and low-precision pointing applications such as games and visualisation. However, they do not lend themselves to perform higher precision tasks such as drawing.

6.2.2.3 Mouse

A mouse is the most common graphic input device that is dragged along a surface. Moving the mouse causes the pointer on the screen to perform a corresponding motion. Selection is performed by one or more buttons placed on the mouse. There are two basic kinds of mice: optical and mechanical.

A mechanical mouse has a small ball on the underside. Dragging the mouse across a surface rotates the ball, determining the motion of the pointer. An optical mouse is an input device that uses a light-emitting diode (LED), an optical sensor, and digital signal processing in place of the traditional mouse ball and electromechanical transducer. Movement is detected by sensing changes in reflected light, rather than by interpreting the motion of a rolling sphere. Nowadays, the optical mice become more and more popular and gradually supplant the mechanical ones.

6.2.2.4 Trackball

A trackball is a pointing device that is similar to a mouse turned upside down. Normal mice are rolled around on the desktop, while a trackball remains stationary, and instead a user rolls the ball around usually using the thumb, index finger, or the whole hand, depending on the size of the device. Rotating the ball causes the pointer to move in the direction of rotation. Buttons located near the trackball are used for selection.

Trackballs can be made very small, allowing them to be embedded in a handheld computer sized base. One drawback of trackballs is the relation between the selection buttons and the ball, which is especially a problem in click-and-drag tasks.

(Providers: http://www.vrealities.com/main.html)

6.2.2.5 Gyration Mouse

A gyration mouse can be operated in the air instead of a desktop. The gyration cordless mouse uses a dual-axis gyroscope when lifted to detect the motion of the user hand, and relays this information to a computer; eventually leading to movements of the mouse-pointer on the screen. This removes the need for the mouse to be placed on a surface. In Figure 26, the Gyration Ultra GT Cordless Optical Mouse is presented. The mouse is integrated with an optical sensor so it can be also used as a standard mouse on a desk.

Figure 26 - Gyration Ultra GT Cordless Optical Mouse

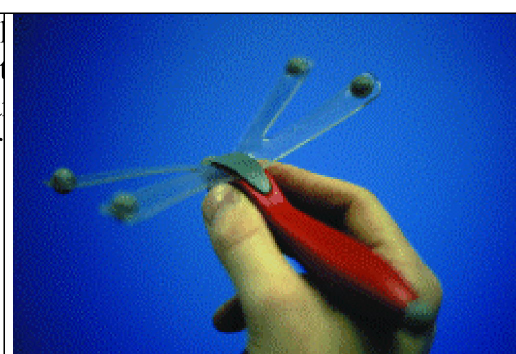
6.2.2.6 6DOF Control

Traditional computer input devices (keyboard, mouse, joystick etc.) are not always appropriate for moving and manipulating 3D environments. They control too few parameters, whereas while placing an object in a 3D virtual space 6 degrees of freedom are required. The 2D pointing devices can only alter 2 degrees of the available 6 within a 3D environment. The operation of the mouse can be extended through the use of modifier keys (on the mouse or keyboard) or through interaction with manipulators placed on the objects within the 3D world allowing them to select the displayed axes on the objects that they wish to manipulate.

One of possible solutions for 6DOF manipulation is the use of pressure sensitive devices such as the spacemouse and spaceball. They offer 6DOF manipulation through the movements of the cap or ball of the device. These devices are classic 6D input devices and have a form of a joystick allowing only small movements, but in all directions, including rotations. They detect the slightest fingertip pressure applied to them and resolve the pressure into X, Y, and Z translations and rotations. This provides intuitive, interactive 6DOF control of 3D graphical images and objects.
These devices require a stable surface for use and as such tend to be used for desktop VR style interaction, rather than within immersive environments where 6DOF tracked devices provide a more intuitive interface. (Provider: http://www.3dconnexion.com/index.php, http://www.inition.co.uk/inition/products.php)

6.2.2.7 6DOF Tracked Devices

Another group of graphic input devices enabling 6DOF manipulation in 3D environments are devices for immersive interaction. The motion of such a device is tracked so a user can manipulate both the position and orientation by moving the device in space. There are a variety of continuous-input devices using different tracking techniques, from which the most popular are briefly described in Table 5.

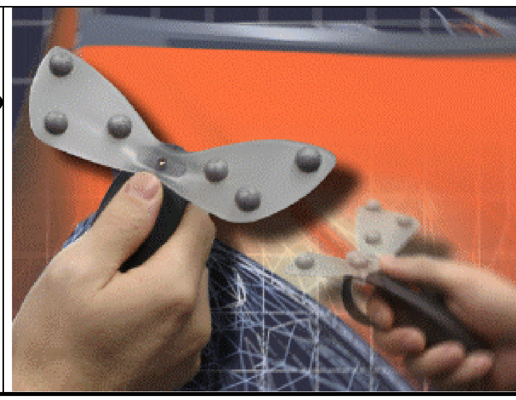
Ascension Wanda	
Wanda is a 3D counterpart of a standard mouse, with an embedded 6DOF sensor that provides the computer with information about the wand's position and orientation. The war consists of three programmable buttons and a joystick. The joystick is used primarily for navigation in combination with th position and orientation information. The buttons are used to se modes and select options. Wanda can be used in free space, unlike a typical mouse or joystick. Wanda is dedicated to be use in CAVE environments to manipulate 3D objects in a virtual environment. (http://www.ascension-tech.com/products/wanda.php, http://www.inition.co.uk/inition/products.php)	
Vircinity Dragonfly	
Dragonfly is a wireless input device with wings used by optical tracking systems to gauge the device movements. Dragonfly wi multiple handholds allows precise handling both for users that a standing or seated. This device without buttons can be used for special interaction concepts or in combination with other interaction devices like e.g. Opti-Hornet (see below).	

Hornet

Hornet is a wired input device designed to house the sensors of electromagnetic tracking systems like, e.g. Polhemus. The devic has two buttons that can be easily reached by thumb and forefinger. Hornet can be additionally equipped with a joystick a trackpoint device. This allows a menu interaction independen of tracking or a redundant possibility of navigation.

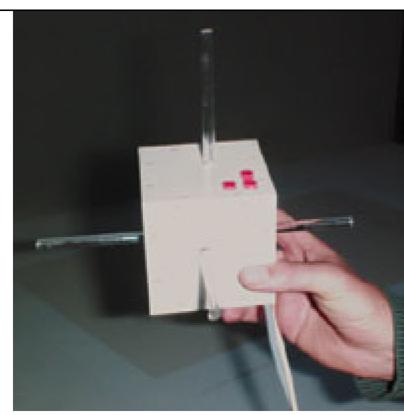
Opti-Hornet

Opti-Hornet is an additional wireless input device with characteristics comparable to Hornet. Attached wings with reflectors for optical tracking systems plus radio communicatio for the buttons supplement the functionality.



Fakespace CubicMouse

CubicMouse is an innovative input device consisting of a cube shaped box and running through it three rods that can be movec and turned – physical representations of the x-, y- and z-axes o the coordinates of the application system. The cube contains an embedded 6DOF tracker inside that allows position and orientation control of a virtual world, whereas the rods are used manipulate the (x, y, z) coordinates of an object in the virtual world.
http://www.inition.co.uk/inition/products.php



NeoWand	
NeoWand combines the functionality of a handheld VR wand c stylus with the familiar interface of a remote control for navigation, manipulation and other programmable functions. Push buttons on the surface provide a method of controlling virtual objects. NeoWand™ is able to support the Polhemus system to allow the device to be tracked. http://www.fakespace.com/	
LaserAid SpaceGrips	
SpaceGrips are handheld input devices providing an alternative to the glove-style input controller commonly used in the VR industry. SpaceGrips may be configured with a variety of electromagnetic tracking systems. http://www.spacegrips.com/SciFi_pale_p1.htm	

Table 5 - 6DOF Tracking devices

6.2.2.8 Trackpad

A trackpad is a stationary pointing device, where a user can operate in 2 dimensions by moving a finger across a small touch-sensitive area. This kind of device is mainly applied in notebook computers and located below the keyboard. There are different technologies for detecting finger strokes. The most common is based on measuring the change in capacitance caused by the user’s finger on a grid of electrodes. Other detection methods utilize conductive, infrared or acoustic phenomena. The former exploits two conductive layers made of electrode grids, which work by touching when pressure is applied. Trackpads have the advantage of having no moving parts. This makes them less prone to collecting dirt, easier to clean, and more adaptable to the hostile environments.

6.2.2.9 Touch Screens

A touch screen is a computer display screen that is sensitive to user touch. A touch screen is essentially a transparent trackpad overlaid on top of a display. Touch screens are used as input of information kiosks, computer-based training devices, and systems designed for users who have difficulty manipulating a mouse or keyboard.

One must note that at the moment of writing (2007), there are a few factors that strongly promote the use of touch screens. Apple has recently introduced the iPhone that is a touch screen-based mobile phone. Instead of the classical pen, the user drives the touch screen commands with two fingers. Microsoft is introducing the Microsoft surface. This touch screen is also driven by the user’s two hands in a more natural way then the classical mouse “drag and click”. What is as important is that the Microsoft OS is now able to work with such touch screen drivers. A third development that is creating a lot of excitement is the Multi-Touch screen by Perceptive Pixel.

There are three types of touch screens depending on the technology applied for detecting finger touch:

- A resistive touch screen panel is coated with a thin metallic electrically conductive and resistive layer that causes a change in the electrical current, which is registered as a touch event and sent to the controller for processing.

- An ultrasonic wave technology uses ultrasonic waves that pass over the touch screen panel. When the panel is touched, a portion of the wave is absorbed. This change in the ultrasonic waves registers the position of the touch event and sends this information to the controller for processing.
- A capacitive touch screen panel is coated with a material that stores electrical charges. When the panel is touched, a small amount of charge is drawn to the point of contact. Circuits located at each corner of the panel measure the charge and send the information to the controller for processing. Capacitive touch screen panels must be touched with a finger unlike resistive and surface wave panels that can use fingers and stylus.

6.2.2.10 Eye Tracking

There are other, less common, graphic input techniques such as gestures, eye tracking and voice. These are not very popular at the moment, usually due to expense or poor usability, but advances in technology may make them cheap and usable enough to provide a viable input method in the future. One of the most intriguing methods of human-computer interaction is the application of eye tracking. The aim of eye tracking is to determine a line radiating forward in space from the eye and indicating what the user is looking at. To do that also head movements should be taken into account because changes of the view line may result from both the head and the eyeball movements. As a result, the head must be held absolutely stationary or more practically the eye tracking should be performed simultaneously with the head tracking.

A variety of technologies have been applied to the tracking of eyeball movements. One of possible solutions is based on recording signals from electrodes placed on the skin around the eye socket. The electrodes measure relative eye movements by detecting the changes in the orientation of the potential difference between the retina and the cornea. Despite the fact that this solution covers a wide range of detected movements, it does not ensure a good accuracy.

A more accurate method of eye tracking relies on a physical attachment to the front of the eye. A contact lens with a special marker is precisely affixed to the cornea. The marker can be either optical or magnetic, and can be tracked reliably. This method is applicable mainly in laboratory conditions, as it is uncomfortable, awkward, and it hinders blinking. This solution can record a limited range of eye movements.

The most practical and most common approach for eye tracking relies on observing the eye with a camera and applying image-processing and pattern recognition techniques to analyse the grabbed picture. In this approach, the position of the eye can be determined by tracking its visible features. A camera can be either set on a desk in front of a user or can be mounted on the user's head. One tracker measures the eye movements with respect to the head, and a separate tracker is responsible for determining the head orientation. When a user wears an HMD, then the head tracking is not necessary because the display moves together with the head. The solution with a camera mounted on the user's head enables a wider range of possible head movements, whereas using a remote camera restricts a user to looking in a direction of the camera. Also, the former approach offers the promise of better stability of eye tracking measurements because the relationship between camera and eye is nearly constant.

Different problems arise in the application of eye movement as graphic input information. Eye movements are often non-intentional or not conscious, so they must be interpreted carefully to avoid annoying the user with unwanted responses to his actions (so-called the "Midas Touch" problem). People are not accustomed to operating devices simply by moving their eyes and they expect to be able to look at an item without triggering an action. Unlike a mouse, it is relatively difficult to control eye position consciously and precisely.

Another problem is that there is no natural counterpart of the mouse button press for the eye tracker. Thus, object selection is performed either by a separate button or using dwell time. In the first solution, the user looks at the desired object that should be selected and then presses a button. In the other one, to select the object a user must gaze at it for a specified period of time.

(Providers: http://www.sr-research.com/index.php, http://www.inition.co.uk/inition/products.php)

6.2.2.11 Bioelectric Control

The rapid increase in the computational performance of modern computers enables handling the computational complexity of analysing bioelectric signals. As a consequence, a growing role of bioelectric measurements in graphic input devices can be observed in recent years. The measurements of the bioelectric signals require the use of electrodes, which can be made small, lightweight, and barely noticeable to the user. A user can operate such an interface using normal body motions and does not need to hold any other device. Different kinds of bioelectric signals can be employed in input devices. The most common methods of bioelectric control are: the electromyogram (EMG), electroencephalogram (EEG), electro-oculogram (EOG), and electrodermal response (EDR). They are briefly described below.

6.2.2.11.1 Electromyography

Electromyography (EMG) is a medical technique for measuring muscle response to nervous stimulation. EMG is performed using an instrument called an electromyograph, which detects the electrical potentials generated by muscle cells when these cells contract. Electromyograms can be read either directly via depth electrodes (invasive) or through the skin via surface electrodes (non-invasive). In the invasive method, needle electrodes are inserted under skin and come into direct contact with the muscle. This technique is considered as more reliable because it gives clear, well defined measurements, requiring less amplification and filtering. However, the use of needle electrodes precludes their use as a casual computer input device due to the health and safety issues.
In the non-invasive method, surface electrodes are stuck on the skin directly above a muscle. In this case, the signal is weaker and noisier than from depth electrodes. Also, using surface electrodes, it is harder to read the potentials from deeper muscles. The problems with surface electrodes are made up for by their ease of application and the possibility of daily use, giving them a fundamental advantage over depth electrodes.

6.2.2.11.2 Electroencephalography

Electroencephalography (EEG) is a technique of exploration of the electrical activity of a brain based on measurements of electric potentials generated by neurons. Measuring is performed by the use of several electrodes placed on the scalp, usually over hairy areas. In order to make a solid contact though the hair, an electrolyte gel is applied, what is a serious disadvantage of this method. To avoid this problem, some EEG systems apply the electrodes only on the forehead, where standard surface electrodes can be used.
The traces resulted from electrodes are known as an electroencephalogram and represent so-called brain waves. Detectable waves can be created by conscious thoughts as well as by sensory input. The low amplitudes of detected signals using EEG require better amplifiers to read in comparison with EMG, and make the system more prone to noise. The low signal frequencies reduce the potential speed of this kind of an input device. The electroencephalograms are widely used in biofeedback systems for clinical purposes, but usage as a computer input devices is limited due to the low accuracy.

6.2.2.11.3 Electro-oculography

Electro-oculography (EOG) is a method of placing electrodes on the skin around the eye to measure the small changes in electric potential between the cornea (front) and retina (back) in the eye. These changes can be used to gauge the intensity of light seen by the eye, or the orientation of the eye. Determining a view direction using EOG is relatively simple in comparison with controlling a pointer using the EEG or EMG technique. To detect the horizontal orientation of the eye the voltage difference from the electrodes placed to the left and right of the eye is measured. Whereas, the difference between the electrodes placed above and below the eye indicates its vertical orientation. Eye trackers based on the EOG are currently viable, and in some circumstances are more convenient than video-based systems.

6.2.2.11.4 Electrodermal response

Electrodermal response (EDR) is a method of measurement of the changes in the resistance of the skin usually resulted from varying emotional states. This technique measures the skin's conductance between two electrodes, which are small metal plates that apply a safe, imperceptibly tiny voltage across the skin. The electrodes are typically attached to the subject's fingers or toes using electrode cuffs or to any part of the body using electrodes such as used in EMG.

The skin conductivity response is the phenomenon that the skin momentarily becomes a better conductor of electricity when either external or internal stimuli occur that are physiologically arousing. The response to stimuli detected by EDR is significantly slower than using either EEG or EMG. The low frequencies and high correlation to emotional stress levels make EDR most appropriate for use for lie detection or biofeedback techniques for stress reduction. A serious disadvantage of EDR is that it is very dependent on the condition of the skin. The signals are harder to detect on areas of the skin with sweat, cuts, or scars as they tend to interfere with the signals.

6.2.3 Gesture Recognition

The interaction with gestures is considered as the most intuitive way of interacting with virtual worlds. Hand tracking has recently received a great attention from the scientific community since it allows developing human computer interfaces which are more natural for the user. Gestures are expressive, meaningful body motions with the intent to convey information or interact with the environment. Hand gestures serve three functional roles: semiotic, ergotic, and epistemic. The semiotic function is to communicate information (e.g. description of scenes or objects), the ergotic function corresponds to the capacity to manipulate objects in the real world (e.g. creation or modification of objects, triggering of commands, navigation), and the epistemic function allows us to learn from the environment through tactile experience. Some of these actions need the creation of a specific vocabulary; others can be performed with natural gestures.

Specific vocabulary

For certain applications in virtual reality, one can need a specific vocabulary where each gesture corresponds to a command that a user want to provide to the application. This kind of gestures needs a learning effort from the user who has to remind all the different commands. These commands can correspond to navigation orders (Mine, 1995), operations to be achieve on objects (Laviola, 1999), etc. Command gestures can lead to a certain tiredness of the user if a command has to be often repeated. Because of these drawbacks, sometimes it can be more profitable to use other modalities (for example commands can often easily be expressed with speech).

Natural gestures

If the user is close to a virtual object, he can manipulate this object in the same way that a real object. He grabs the object, then moves or orientates it, and finally he releases the object. This kind of gesture interaction is called direct manipulation. To interpret manipulation gestures, one can either use a simple recognition system, or built a physical model of the world in order to detect collision between the fingers and the objects (Pappas et al., 2003).

To perform the selection of an object, the user can use the deictic gestures. The most common deictic gesture is the pointing of an object with the extended index. This kind of gesture can be combined with other modalities in order to perform an action on the pointed object. For example, in (Latoschik, 2001) the authors present an application that combines speech and gestures. To grab a distant object, the user has to point the object and then (or simultaneously) say a sentence like "take this object". The process, then, consists to detect the moments of synchronization between the gestures and the voice, and to solve the co-references in the multimodal utterance.

A third kind of gestures, iconic gesture, can be used to perform descriptions, allowing the user to create objects or refer to an object of the scene, by showing some spatial properties on the object. Thus, in (Pratini, 2001), iconic gestures allow the user to create surfaces by extruding a contour, or to create geometric primitives with the shape of the hand. In (Sowa and Wachsmuth, 2001), the user can depict an object of the scene by performing iconic gestures that express some of the dominant features of the object (orthogonality, equal lengths, etc).
Moreover, iconic gestures can be used to manipulate spatial information on the scene graph it self. For example, in (Bossard, 2005), a user can express with gestures that an object is on another one. In this case, gestures are used to modify the scene, but also to add higher level information in the graph scene, as for example spatial relation between entities.

But despite the variety of new devices, human-computer interaction still differs in many ways from human-to-human interaction. Natural interaction between humans does not involve devices because we have the ability to sense our environment with eyes and ears. In principle, the computer should be able to imitate those abilities with cameras and microphones.

In the last ten years, there has been a lot of research on vision based hand gesture recognition and bare hand/finger tracking. Interestingly there are many different approaches to this problem with no single dominating method. Typical hand segmentation techniques are based on stereo information, colour, contour detection, connected component analysis and image differencing. Each technique has its specific drawbacks:

- **Stereo image-based segmentation**: the implementation of stereo algorithm can hardly provide disparity computation in real-time.
- **Colour segmentation** is sensitive to changes in the overall illumination. In addition, it is prone to segmentation errors caused by objects with similar colours in the image.
- **Contour detection** tends to be unreliable for cluttered backgrounds. Much stability is obtained by using a contour model and post-processing with the condensation algorithm, but this restricts the maximum speed of hand movement.
- **Connected component algorithms** tend to be heavy in computational requirements, making it impossible to search through the whole image in real-time. Successful systems employ tracking techniques, which again restrict the maximum speed of movement.
- **Image differencing** generally only works well for moving objects and requires sufficient contrast between foreground and background.

As far as bare hand tracking is concerned, experiences shows that an appropriate mix of some of the above techniques may allow to achieve interesting results.
Literature (Mahmoudi, Parviz 2006) and existing experiences show that it is possible to achieve bare hand tracking and gesture recognition:

- With the hand moving across a non-uniform background, provided that the background colour is different from the object (hand) that is tracked.
- With low illumination conditions, provided that it is possible to perform colour recognition and that the source of illumination is stable (problems have been observed with fluorescent lamps due the features of their power supply).
- With satisfactory update rate (latency lower than 30 ms to achieve a frame rate of 25 Hz) allowing natural hand movement and real-time tracking and recognition.

Up-to-date off-the-shelf bare hand tracking systems are not available; therefore it is difficult to provide a comparison. An evaluation may be performed by considering them with existing systems (e.g. glove based).

6.2.4 Auditory Input

In classical VR systems like CAVEs the users are standing and have no keyboard for command or text input. The operation of more complex VR applications with many commands requires either 3D menus with deep structures or speech recognition for command input. Using complex 3D menus is slow and tiring, but spoken (short cut) commands can help to make interaction more efficient and faster. But it is important not to overload the auditory input channel because users would have to keep in mind many different commands. Another disadvantage is that spontaneous interaction with new users is not easy because the speech recognition system has to be trained first.
Many current VR runtime systems are offering an interface to speech recognition software. Typically any commercially available standard speech recognition software can be used. The translated commands are sent over the network to the VR application and interpreted. Speech recognition is especially useful in VR systems used for virtual training.

7 VE Software Platforms

7.1 Introduction

This section defines and describes the constituent software tools that can be combined to form a VE software platform to develop on. These include for example: scene graphs, collision detection, physically based modelling, distributed and collaborative environments...

7.2 Image Generators (IG)

An Image Generator (IG) is a major component of a VR system that generates a synthetic scene/world that is then fed to an appropriate display and allows the user to receive visual cues.

Technological advances in the field of computation, visual systems, and human-machine interfaces that have taken place since the 70's have enabled the development of system having an ever increasing level of fidelity. From the first analogue calculators to state-of-the-art supercomputers, things have changed quite dramatically, and an ever increasing realism has been brought to more and more affordable systems.

Over the past years a wide variety of visual simulation systems have been conceived and developed, including:

- model board systems using closed circuit television or a laser camera and a laser projector
- the shadowgraph system, using point light sources shining through transparent models
- computer generated image (CGI) systems, using a colour CRT or TV monitor for presentation purposes

Currently the preferred solution comprises of a suitable IG which supplies a number of CRT, LCD or in the future Digital Light Processor (DLP) projectors and Laser projectors.

Requirements for developing IGs refer both to the intended application of the VR system and to the characteristics of human vision. While the scope of application may differ from field to field, the characteristics of user vision are invariable.

The eyes are the most important senses of the human for gaining information about the surrounding world. Over 90% of the information the human being receives during normal daily activities comes through the eyes and this may be even more for particular tasks. The representation of the environment is a significant challenge because of the terrific performance capabilities of the human eye. The central foveal area of the retina, having a resolution better than 1 arc min, answers the question of "what" the pattern vision is. The peripheral retina, having a resolution of about 20 arc min and being highly sensitive to image movement, answers "where" the observer or the environment are moving. The field of vision limits of human vision are:

- Fixed head and eyes: foveal 5° diameter; peripheral 190° horizontally and 113° vertically.
- Moveable head and eyes: foveal 144° horizontally by 170° vertically; peripheral 474° horizontally by 170° vertically.

The following set of basic features should be considered when designing IGs in order to adapt them to the users' characteristics:

- Scene content
- Depth and range effects
- Refresh and update rate
- Polygon capacity

- Pixel rate
- Over-write capacity

Scene content: it is the amount of details which can be included in a generated image. Recent evolutions of graphics hardware in texture support, anti-aliasing and programmable shaders enable a much higher level of rendering quality and realism, cost-effectively, which meets the requirements of VR for immersion and fidelity.

Depth and range effects: the property of human vision that allows the perception of an accurate 3D rather than a flat world is depth perception. Mechanisms associated with this property can be separated in two sets of requirements for IGs: those associated with monocular vision and those associated with binocular vision. Examples of the former are:

- size constancy; based on the constant size of real world objects and so retinal image size relates to object distance
- motion parallax; it describes a change in relative position of objects due to relative motion of the observer with respect to a scene object.
- linear perspective; it is a cue based on the perspective transformation leading to the geometrical aspect of a vanishing point (parallel lines are perceived as convergent with increasing distance).
- object masking (interposition); it happens when a closer object overlaps another being at a greater distance from the observer.
- shadows; provides strong impression of the three dimensionality of objects.

Refresh and update rate: IG must generate and display a new picture at a sufficiently high rate in order to create the illusion of smooth continuous motion. If the refresh rate is below a certain frequency the creation of a perceptible flicker of the image luminance may be possible. The flicker suppression refresh rate is the refresh rate at which the flicker is just imperceptible. Depending on the visualisation hardware, it may vary between 25 and 60 Hz.
The update rate is the generation frame rate or the frequency at which a complete new image is generated. Using a faster update rate results in an improved image quality of dynamic scenes.
Going into a bit more detail, each scene generated by the IG is made up of a set of polygons having an appropriate shape and being textured with appropriate patterns. The following features refer to the IG capacity of handling polygons.

Polygon capacity. The polygon capacity of an image generator is often expressed in polygons per second. Generally, the polygons per second number can be divided by any frame rate to yield the polygons that might be in each image. It should be noted that the polygon rate measures the capacity of the machine to perform transformations from a three-dimensional database into two dimensional screen coordinates. No IG will accommodate 100 vertex polygons at the specified rate. Usually, the specified rate is for triangles, and occasionally for quads (having four vertices). In addition, polygons making up an image generally share vertices in a mesh, each transformation being shared by two or more triangles. Therefore, to avoid the ambiguity in the mesh size, it is conventional to specify the polygon rate for unmeshed triangles.
Depending on the machine architecture, the polygon capacity may vary depending on whether the polygon is to be shaded, smooth shaded, or textured. Each of these attributes adds calculations to the transformation process, so unless the calculations are done in parallel performance will suffer.

Pixel rate. The specified polygon rates will not be achieved if the polygons are too large on the screen. If the polygons have too many pixels, the machine will bog down in making the pixels. The pixel rate is the

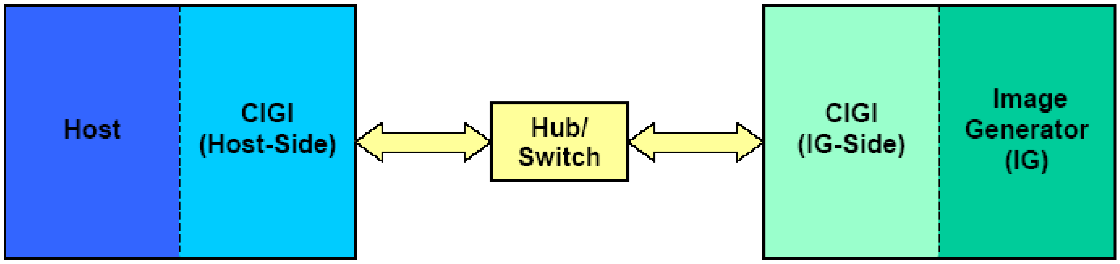
second key IG specification. Consequently, the polygon rate will be specified for triangles so small that the pixel rate limitations do not come into play.
Usually the pixel capacity is specified in terms of generated pixels per second. It is to be noted that because some generated pixels are covered (i.e. occluded) by other pixels, the image generator must generate more pixels than are eventually displayed. Pixels from a nearer object occlude the pixels of the more distant objects. A scene that ends up with a million pixels per output frame of video takes many more than a million generated pixels per frame to render.

Overwrite capacity. The ratio of the image generator capacity to generate pixels to the number of displayed pixels is called the overwrite capacity of the machine. If the overwrite capacity is one, there can be no occlusion whatsoever in the scene, which is rarely the case in practice.
In addition to generating occluded pixels, the image generator must generate extra pixels for anti-aliased edges. If a pixel is half covered by one polygon and half covered by another polygon with which the first shares an edge, then the pixel will have to be visited twice by the image generator even though there is no occlusion. Because fractional pixels take as long to generate as full pixels, the pixel demand is increased by the number of edge pixels. The number of edge pixels in a scene depends upon many factors, including the average size of a polygon.
As an example, an aircraft at altitude mainly looks down at the terrain, so there is little occlusion. Landing areas are flat, and hills and buildings are usually distant. Adding a layer of semi transparent clouds will add additional overwrites. A more severe case is for land vehicle simulation, where nearby vehicles, layers of trees, smoke and fire effects, nearby buildings, and rows or hills all come into play.
In order to cope with the overwrite requirement a number of methods exists, having an influence on the hardware solution:

- Z buffered machines: these are the easiest to understand. Every pixel of every object takes the same time to generate, and every pixel is generated whether or not it ultimately ends up being occluded.
- use of a dedicated algorithm: this solution, so-called list priority, identifies the correct order to write the objects in the frame buffer that is building up the picture. Once the pixels of an object are written, the algorithm guarantees that no other pixels will ever occlude them. Consequently, any portion of the screen that has been written can subsequently be skipped over.
- hybrid architectures: these solutions, which are the most common, use a list priority algorithm for fixed objects and Z buffering for moving objects.

Parallelisation. VR applications often require to aggregate hardware resources in order to fulfil demanding requirements of immersive VR visualisation - e.g. number of displays, resolution, stereoscopy, scene complexity… This remains true for a number of applications despite dramatic enhancements in hardware. Therefore multiple engines may be required in conjunction, either in a shared memory system embedding multiple graphics engines (“multi-pipe”), or a set of interconnected computers with graphics (rendering cluster). This impacts design of VR software to support effectively such parallelisation, for which a number of techniques and tools have been developed.

CIGI. The Common Image Generator Interface (CIGI) is a standardized interface between a real-time simulator host and an image generator. CIGI is an open interface offered to promote commonality in the visual simulation industry. It is also a data packaging protocol and thus does not depend upon a specific physical communications medium or transport protocol. Any suitable physical medium may be used, including Ethernet, Token Ring, optical fibre, shared memory... The transport protocol(s) used should depend upon performance and what is appropriate to the communications hardware.



The goal is to produce a universal plug-and-play type interface to reduce integration costs.
The Interface Control Document (ICD) describes Version x of the open-source CIGI and contains:

- CIGI theory of operation
- Start-up sequence
- Data packaging
- Message synchronization and frequency
- Coordinate system definitions
- Display definitions
- Complete definition of all data packets
- Weather and surface conditions definitions
- Motion trackers
- Sensors
- Position and state queries
- Mission functions

The Software Development Kit (SDK) is an ANSI C API core and it contains an addition of an object oriented class library in C++ for CIGI x. The host emulator fully supports all CIGI data packets and provides test scripting for fast, easy regression testing. It contains record and playback functions and timing analysis functions.

History of CIGI:

- Initial discussion started in mid 2000
- Official release of CIGI 1.0 March 2001
  - Posted on Boeing and Sourceforge web sites
  - Incorporated and tested with CG2 Mantis run-time
- Official release of CIGI 2.0 March 2002
  - CIGI stated as requirement in three Navy RFPs and Naval Aviation Simulation Master Plan
- Official release of CIGI 3.0 Fall 2003
  - Last official release of CIGI 3.2 April 2006

All data and documentation are available on http://cigi.sourceforge.net, includes ICD, SDK and CIGI Host Emulator.

7.3 Scene graphs

Scene graphs are middleware, which are built on top of low-level APIs to provide spatial organization capabilities and other features typically required by high-performance 3D applications. Figure below illustrates a typical OSG application stack.

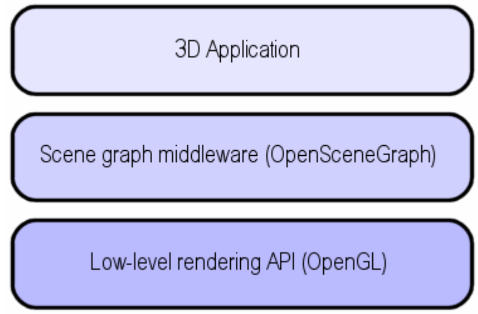


Figure 27 - The 3D Application Stack

Scene graphs expose the geometry and state management functionality found in low-level rendering APIs, and also provide additional features and capabilities, such as the following:

- Spatial organization – The scene graph tree structure lends itself naturally to intuitive spatial organization.
- Culling – View frustum and occlusion culling on the host CPU typically reduces overall system load by not processing geometry that doesn't appear in the final rendered image.
- Level of Detail (LOD) – Viewer-object distance computation on bounding geometry allows objects to be efficiently rendered at varying LODs. Furthermore, portions of a scene can load from disk when they are within a specified viewer distance range, and deleted from memory when they are beyond that distance.
- Translucency – Correct and efficient rendering of translucent geometry requires all translucent geometry to render after all opaque geometry. Furthermore, translucent geometry should be sorted by depth and rendered in back-to-front order. These operations are commonly supported by scene graphs.
- State change minimization – To maximize application performance, redundant and unnecessary state changes should be avoided. Scene graphs commonly sort geometry by state to minimize state changes, and OpenSceneGraph's state management facilities eliminate redundant state changes.
- File I/O – Scene graphs are an effective tool for reading and writing 3D data from disk. Once loaded into memory, the internal scene graph data structure allows the application to easily manipulate dynamic 3D data. Scene graphs can be an effective intermediary for converting from one file format to another.
- Additional high-level functionality – Scene graph libraries commonly provide high-level functionality beyond what is typically found in low-level APIs; such as: full-featured text support, support for rendering effects (such as particle effects and shadows), rendering optimizations, 3D model file I/O support, and cross platform access to input devices and render surfaces.

Nearly all 3D applications require some of these features. As a result, developers who build their application directly on low-level APIs typically end up implementing many of these features in their application, which increases development costs. Using an off-the-shelf scene graph that already fully supports such features enables rapid application development.

A scene graph is a hierarchical tree data structure that organizes spatial data for efficient rendering. The scene graph tree is headed by a top-level root node. Beneath the root node, group nodes organize geometry and the rendering state that controls their appearance. Root nodes and group nodes can have zero or more children. (However, group nodes with zero children are essentially no-ops.) At the bottom of the scene graph, leaf nodes contain the actual geometry that makes up the objects in the scene.

Applications use group nodes to organize and arrange geometry in a scene. Imagine a 3D database containing a room with a table and two identical chairs. You could organize a scene graph for this database in many ways. Figure 28 shows one example of the organization. The root node has four group node's children: one for the room geometry, one for the table, and one for each chair. The chair group nodes are colour-coded red to indicate that they transform their children. There is only one chair leaf node because the two chairs are identical – their parent group nodes transform the chair into two different locations to produce the appearance of two chairs. The table group node has a single child, the table leaf node. The room leaf node contains the geometry for the floor, walls, and ceiling.

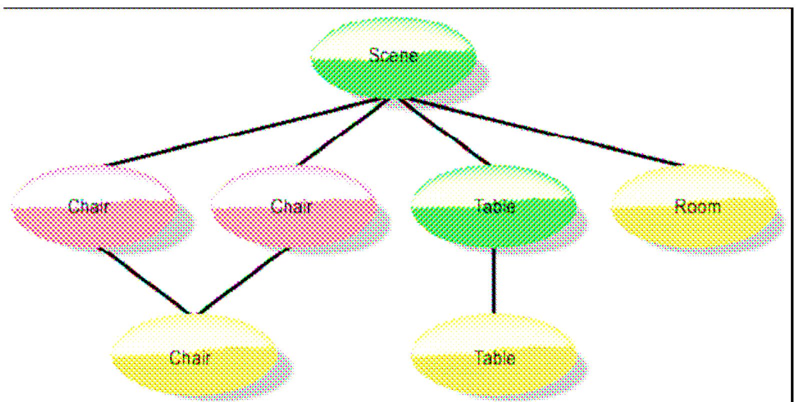


Figure 28: A typical scene graph

Scene graphs usually offer a variety of different node types that offer a wide range of functionality, such as switch nodes that enable or disable their children, LOD nodes that select children based on distance from the viewer and transform nodes that modify the state of child geometry. Object-oriented scene graphs provide this variety using inheritance; all nodes share a common base class with specialized functionality defined in the derived classes.

The large variety of node types and their implicit spatial organization ability provide data storage features that are unavailable in traditional low-level rendering APIs. OpenGL and Direct3D focus primarily on abstracting features found in graphics hardware. Although graphics hardware allows storage of geometric and state data for later execution (such as display lists or buffer objects), low-level API features for spatial organization of that data are generally minimal and primitive in nature, and inadequate for the vast majority of 3D applications.

In a complementary approach on support for multi-thread systems, distributed applications and advanced rendering effects, scene graph languages are also evolving in order to offer more expressivity as any other interface generation API does. The purpose is to offer various renderings that depend on the context of visualisation (as in mobile Augmented Reality) or on the profile of the user (as for users with various skills and abilities) (Reitmayr and Schmalstieg, 2005).

7.3.1 Audio

In the W3C community, some effort has been devoted to multimodality. For instance, CONTIGRA Audio3D offers complex sonic descriptions that can be used together with other XML-formats for 3D graphics such as X3D (Hoffmann et al., 2003). It is based on the concept of a hierarchical scene graph. The approach at the LIMSI lab is different since they consider audio objects as sonified geometrical objects. As such, the sonic objects can be embedded inside the scene graph as any other scene component, thus facilitating the interpolation of geometrical and sonic properties. In a recent experiment on auditory perception, Virtual Choreographer has been used to define an audio-augmented reality environment in which virtual sounds and virtual geometrical objects (corresponding to real objects) were embedded in the same scene description and could interact with one another (Afonso et al. 2005). The future for such scene descriptions is to combine sound spatialisation with acoustic properties of geometrical objects (Tsingos et al., 2004) in an efficient manner at the perceptual and implementation levels. In the future, the audio rendering pipeline will be integrated into the geometric scene graph-based VR toolkits and not considered as an additional component (Naef et al., 2002).

7.3.2 Automatic large scene graph generation and customising

Another issue in the design of scene graph is to cope with large scene descriptions such as the one encountered in the domain of massive data visualisation. For this purpose, it can be desirable to design programs for the automatic generation of scene graphs and associated behaviour. The direct generation of scene graphs from data description has the double drawback of hiding the genericity of the data generation process and preventing the possibility of customising the scene to specific user's needs. Chi's framework for information visualization organises typical information processing for visualisation in three layers (Chi, 2002). The intermediary layer is an abstract data representation. It contains meta-data that can be eventually transformed into visualisation data. The approach of Jacquemin et al. (2005; 2006) to transform abstract representations into visualisation data relies on XSL stylesheets that pave the way to visual rendering and interaction customising. The same abstract data representation can be used to generate a wide variety of interface layouts and behaviours, therefore adapting the visualisation and interaction to different user profiles (interests, abilities, experience...).

7.4 VR runtime systems

There are various VR runtime systems that operate in software and give the user the ability to design and operate integrated 3D applications for specific domains in real-time. Such applications include development libraries that offer useful tools for the virtual environment creation and management. In this chapter a short description of the most important ones are given.

7.4.1 Worldtoolkit

The WorldToolKit® is a multi-platform software development system for building high-performance, real-time, integrated 3D applications for scientific and commercial use. It incorporates a set of function libraries and tools that the user needs to create, manage and commercialise applications. With the aid of an application programmers' interface (API) the user can easily and quickly prototype, develop and configure his applications to best fit his implementation. The WorldToolKit® provides support for networked distributed simulations while at the same time provides options for immersive displays, and interface devices (HMDs, trackers, etc).

Concerning programming features of the WorldToolKit® package, it integrates an object-oriented library in C with high-level functions for configurations, interactions and real-time simulations controlling management. The code has been organised in a number of classes containing the geometric objects, viewpoints, sensors, paths, lighting and other features (rendering etc).

WorldToolKit is portable across various platforms meaning that it is compatible with most operating systems such as Windows, Linux and Sun.

7.4.2 Lightning

The Lightning VR system is another application development framework for virtual reality applications and can be used either as a development system or as a runtime module basically under Unix-based workstations and Linux. Similarly to the aforementioned applications, Lightning includes sets of functions for 3D models, inputs, outputs and communications so that the user can use and produce the application that fits his needs. Its programming environment allows the user to make adjustments using either the incorporated language or using C/C++ modules even at run-time.

7.4.3 CAVElib

CAVElib (http://www.iri-vr.ncl.ac.uk/software/cavelib.html) is a C programming interface that can be combined with Performer, Open Inventor or other OpenGL renderers to manage a number of aspects of VR systems and help building virtual reality applications. These aspects include some display, multi-pipe parallel rendering and tracking configurations. Originally designed and developed at EVL, University of Illinois at Chicago since 1992, CAVElib has been used in a number of VR applications and is available in a commercial version from VRCO.

7.4.4 Input device interface libraries

A number of libraries are available to simplify application developer task to manage the various input devices encountered in VR systems, either commercial such as Trackd (VRCO - http://www.vrco.com/trackd/Overviewtrackd.html) or open source such as VRPN (University of North Carolina Chapel Hill), OpenTracker (University of Vienna) and VRJuggler (see below, includes built-in support for device interface).

7.4.5 VR Juggler

VR Juggler is an open-source virtual reality application development framework developed by the Iowa State University's Virtual Reality Applications Center.
VR Juggler provides a set of application programming interfaces (APIs) that helps the programmer on interface aspects, display surfaces, tracking, navigation, graphics and rendering techniques. The versatility of the application developed allows it to operate under any circumstances independently of the I/O devices connected, computer platform and VR system.

The VR Juggler Suite includes a virtual application, a device management system (for local or remote access to I/O devices), a standalone generic math template library, a portable runtime (providing cross-platform thread, socket and serial port primitives), a simple sound abstraction, a distributed model view-controller implementation and an XML-based configuration system with multivariate types.

7.4.6 OpenGL Performer

OpenGL Performer™ (http://www.sgi.com/products/software/performer/) is a programming interface that helps developers to create real-time simulation and other performance-oriented 3D graphics applications. It incorporates capabilities that could help in complex applications such as visual simulation, manufacturing, virtual reality systems, scientific visualisation, interactivity and computer-aided design. OpenGL performer incorporates capabilities of using multi-processor and multi-graphics pipelines when big amounts of processing power are needed.
OpenGL Performer™ has been built on top of the OpenGL® standard graphics library and combines ANSI C and C++. The suite can be used for applications' development on a variety of operating environments such as SGI IRIX, Linux and MS Windows (2000 and XP).

7.4.7 OpenSG

OpenSG (http://www.opensg.org/) is a portable SceneGraph system that supports the creation of real time graphics programs for Virtual Reality applications. It is developed following Open Source principles and can be used freely. It runs on IRIX, Windows and Linux and is based on OpenGL.
OpenSG is not supposed to be a complete VR system. It is the rendering basis on top of which VR systems can be built. Based on OpenGL, it handles all their primitives and enables multithreaded asynchronous scenegraph manipulations.

7.4.8 OpenSceneGraph

The OpenSceneGraph (http://www.openscenegraph.org/) is an open source high performance 3D graphics toolkit, used by application developers in fields such as: visual simulation, games, virtual reality, scientific visualization, modelling... Written entirely in Standard C++ and OpenGL, it runs on all Windows platforms, OSX, GNU/Linux, IRIX, Solaris and FreeBSD operating systems.

The OpenSceneGraph (OSG) has been developed as a standalone open source scene graph project, thanks to an increasingly wider audience providing both support and applications. From year 2000 on; this middleware encountered an increasing interest among a user community that built thanks to a yearly meeting (OpenSceneGraph birds-of-a-feather: BOF) and a very active osg-users mailing list.
OSG features and add-on libraries were developed at a rapid pace. In 2003, the OSG companion library, Producer, was created to provide a multi-pipe rendering capability for Magic Earth. In 2004, large database paging, terrain support, and shader support were added. During 2006, the OpenFlight plugin was completely revamped; the osgViewer has been created, and an integrated library for managing and rendering views of a scene was included.

Today, several high-performance applications use OSG to manage the rendering of complex 2D and 3D scenes. Though most OSG-based applications are in the visualization and simulation industries, OSG is found in nearly every field that employs 3D graphics, including geographic information systems, computer-aided design, modelling and digital content creation, database development, virtual reality, animation, gaming, entertainment...

OSG's open source nature has many benefits:

- Improved quality – OSG is reviewed, tested, and improved by many members of the OSG community. Over 200 developers contributed to OSG v1.2.
- Improved application quality – To produce quality applications, developers need intimate knowledge of the underlying middleware. If the middleware is closed source, this information is effectively blocked and limited to vendor documentation and customer support. Open source

allows application developers to review and debug middleware source code, which allows free access to code internals

- Reduced cost – Open source is free, eliminating the up-front purchase price.
- No intellectual property issues – Software patent violations cannot be hidden in open source code, it is then easily readable by all.

OpenSceneGraph is also supported by OSGEdit, a tool to edit 3D scenes for the OSG library. OSGEdit is only a composer and not a modeller. It is used to compose complex scenes based on individual models using the OSG library. It is focused on making scenes for use in OSG-based programs.

OSG support is easy to find by subscribing to the osg-users email list or by contracting with professional support.

7.4.9 Open Inventor

Open Inventor (Strauss and Carey 1992) was originally developed by Silicon Graphics as a high level object oriented scene graph API aiming at simplifying and accelerating the development of 3D applications base on OpenGL rendering. The implementation of Open Inventor from Mercury made it available cross-platform and cross-language (available on Windows, Linux, Mac, UNIX, and programmable with C++, .NET or Java API). Open Inventor is now an extensive suite with a number of key features for VR applications: event and interaction model extended for 6DOF tracked input devices, stereo viewing and head tracking, multithreading and multi-pipe rendering, distributed rendering and compositing on cluster with scene graph-based synchronization, built-in support for multi-screen displays, remote rendering, 3D user interface components (immersed dialogs/menus), NURBS, volume rendering, large model visualization, collision detection, VRML compatibility, special graphical effects libraries and high level support for GPU shader programs.

Amira (http://3dviz.mc.com/products/amira.asp) is a 3D data visualization and analysis framework on top of Open Inventor.

The cooperative research project Immersive Inventor focused on optimized utilisation of high performance parallel immersive displays and on defining an extensible and portable programming interface integrating necessary elements to quickly develop immersed user interfaces.
(www.rntl.org/projet/resume2000/Inventor_Immersif.htm - French).

The cooperative research project GeoBench adressed distributed immersive visualisation and haptic interaction applied to (geo) scientific data, based on Amira for the data visualisation and NetJuggler for distributed synchronization through a PC cluster.
(www.rntl.org/projet/Posters-PDF/RNTL-Poster-GEOBENCH.pdf - French)

7.4.10 Skeletal Animation

Serón et al. (2002) extends scene graphs with two nodes that permit the creation of 3D virtual characters. Characters are defined by the degree of freedom to move each of their joints. Therefore, their movements are not limited to predefined behaviours or to the copy of the motions of real actors.

7.5 Real Time Physics Engines (collision detection, rigid body dynamics)

In this section we present algorithms and techniques for real time rigid-body dynamics and collision detection. Real time physical simulation is a synthesis of the most modern approaches to rigid-body simulation, and provides collision detection, contact generation and dynamic response. This synthesis has led to the creation of a highly flexible and extensible framework that manages modules implementing the most up-to-date and efficient algorithms available.

Today, there are a large number of libraries available, a few commercial and a number of non-commercial ones. In this survey we will summarize their main characteristics. We will focus on commercial and open source ones. A detailed survey of academic research was produced by Lin and Gottschalk (1998). The main commercial libraries are Vortex™ from CM-Labs, Novodex™ from Novodex, Havok ™ from Telekinesys and VPS-PBM ™ from Boeing. The main open source engine is ODE (Open Dynamic Engine). These libraries provide robust, ready-to-use libraries for rigid-body dynamics and collision detection.

7.5.1 Collision Detection and Contact Generation

In all these libraries real time physical simulation's collision detection works in a hierarchical fashion allowing for crude, but fast, detection of non-collision, followed by slower, but accurate, detection of collision between objects that are close. For example in Vortex™ a proprietary implementation of a far field processor accepts hints as the distribution of objects in space and will automatically divide objects into sets upon which collision detection must be performed. For detecting collision overlap a number of collision types have been implemented: primitives (sphere, box, cylinder), convex mesh, mesh, triangle list and others. In all these libraries primitives represent the fastest collision types and create very stable contacts because they are smooth surfaces. In general, users are advised to use primitives wherever possible in simulations to obtain these benefits.

In Vortex™ and ODE™ real time physical simulation contain a very fast and very optimised convex mesh solution that implements a GJK (Gilbert-Johnson-Keerthi) algorithm (Canny, 1984; S. Gottschalk et al., 1996; Ong and Gilbert, 1997; Kim et al., 2002). The essential idea is to exploit the coherence of spatial relations between convex objects to allow for fast elimination of non-intersecting objects. Utility functions are provided to create a convex collision object from a mesh by building its convex hull.

The mesh-mesh algorithm implemented in real time physical simulation is proprietary in all commercial libraries. Creating good contacts between two meshes is very complex, and it can be difficult to determine the difference between sharp edges actually present in an object (the corner of a box, for example) and the piece-wise linear approximations to smooth surfaces typical of meshes. In the first case, there is a natural and required discontinuity of contact, while, in the second one, contacts must be smoothed to avoid jitter in a simulation. Research teams are currently working on improving this implementation by using topological information to determine the best choice of contacts.

VPS relies on voxels, both as geometric modelling elements and as bounding volumes for collision detection. Since voxel-based collision detection is volumetric in nature, it can detect when an object is wholly embedded in other object. VPS has been carefully optimised for both speed and memory efficiency. PBM is used here to mean real-time motion simulation including time critical applications such as haptics (force feedback). These are key capabilities in VPS, because VPS was originally designed for 6DOF haptic rendering. PBM differs from VPS collision and proximity detection. Both involve detecting the collisions of virtual objects, but only PBM generates a collision response and calculates subsequent motion. However, PBM does not return highly detailed collision information, such as which parts or triangles were involved in the collision, because that would consume too much time in the target applications. Such detailed information is left to VPS collision and proximity detection. For the same reason, PBM is limited to voxel accuracy, and exact-surface accuracy is left to VPS collision and proximity detection.

In Novodex ™ and Havok ™ a number of collision types have been implemented: primitives (sphere, box, cylinder), convex meshes. Primitives represent the fastest collision types and create very stable contacts. For the mesh-mesh case Novodex™ uses voxels, similar to the approach proposed by VPS. For the mesh case Havok™ uses an approach based on continuous collision detection. A common problem in many applications that include collision detection is that of temporal aliasing. If objects are moving too fast between collision detection calls, many techniques fail to report a collision. Continuous methods proposed by Stephane Redon (Snyder, 1992; Redon et al., 2000) offer a solution to this problem. In addition to being more robust, they have the ability to provide very accurate contact information, which is essential to many simulation applications.

A trend in academic research is to use GPU (Govindaraju et al., 2003).

7.5.2 Dynamic Response

All these libraries implement a consistent framework for both kinematics and dynamics constraints. These may be divided into contact and joint constraints, which are handled in a consistent fashion. Basic Constraint Library is used to take a list of joints defined by the user, contacts generated at run-time and user-defined parameters, and to create a matrix representing the differential equations for the system to be solved.

In all these libraries, except PBM™, the core solver uses an LCP (Linear complementarity problem) solution. This type of solutions for dynamics problems was introduced in Dave Baraff's papers (Baraff, 1990; Baraff, 1994). However, his formulation wasn't completely consistent, and the software has created a unique formulation of the friction and contact engine. For example ideas from Uri Ascher and Dave Trinkle have been used to improve methods of returning to a correct solution from an approximation in Vortex ™ and ODE (Lotstedt, 1984; Stewart and Trinkle, 1996; Anitescu and Potra, 1997). PBM™ uses a simple approach with is based on penalty.

7.5.3 Integration with 3rd Party Applications

All these libraries are supplied with utilities to allow easy integration with 3rd party applications. This is done by exchanging the positions and orientations of objects between the two APIs and synchronising the current time step. Additionally, many utilities are provided to create collision objects from specific graphics objects. "Bridges" are provided between Real time physical simulation and many scene graphs such as Performer, and Virtools. Also, objects in flt or obj format may be loaded from disk and used to build a collision object.

VPS-PBM™ was originally designed for 6DOF haptic rendering with the Phantom from Sensable. Haption & CEA/LIST have developed 6DOFhaptic plug-in for Vortex ™, ODE, Havok™ and VPS-PBM™.

7.6 Distributed Virtual Environments

The widespread use of both fast Internet connections and also high performance graphic cards has made possible the current growth of Distributed Virtual Environment (DVE) systems. These systems allow multiple users, working on different computers that are interconnected through different networks (and even through the Internet) to interact in a shared virtual world (Singhal and Zyda, 1999). This is achieved by rendering images of the environment as the user would perceive them if he was located at that point of the virtual environment. Each user is represented in the shared virtual environment by an entity called avatar, whose state is controlled by the user. Since DVE systems support visual interactions between multiple avatars, every change in each avatar must be notified to the neighbouring avatars in the shared virtual environment. DVE systems are currently used in different applications, such as collaborative design (Salles et al., 1997), civil and military distributed training (Miller and Torpe, 1995), e-learning (Bouras et al., 1998) or multi-player games (Lewis and Jacobson, 2002).

Designing an efficient DVE system is a complex task, since these systems show an inherent heterogeneousness. This heterogeneousness appears in several elements (Morillo et al., 2003):

1. **Hardware.** Each client computer controlling an avatar may have installed different hardware: A very different range of resources like processor speed, memory size, and graphic card technology can be specified for different client computer.
2. **Connection.** Different connections can be found in a single system. From shared medium topologies like Ethernet or Fast-Ethernet to other network connections like ISDN, fiber-optic or ATM can be simultaneously found in some DVE's.
3. **Communication rate of avatars**. Depending on the application, different communication rates of avatars can be found. For example, the communication rate of avatars in a collaborative 3D environment may greatly differ from the communication rate of avatars in a 3D virtual military battle.

Additionally, other factors help to increase the complexity of designing an efficient DVE system. Each of them has become nowadays an open research field:

1. **Data Model.** This concept describes some conceivable ways of distributing persistent or semi-persistent data in a DVE (Macedonia and Zyda, 1996). Data can be managed in a replicated, shared or distributed methodology.
2. **Communication Model.** Network bandwidth determines the size and performance of a DVE. The system behaviour is related to the way that all the scene clients are connected. Broadcast, peer-to-peer or unicast schemes define different network latency values for exchanging information between avatars.
3. **View Consistency.** This problem has been already defined in other computer science fields such as database management. In DVE systems, this problem consists of ensuring that all avatars sharing a virtual space with common objects have the same local vision of them.
4. **Time Consistency.** The notion of time must be the same for all the users in the DVE. Real-time systems mainly include multi-player games and concurrent designing tools.
5. **Message Traffic Reduction.** Keeping a low amount of messages allows DVE systems to efficiently scale with the number of avatars in the system. Traditionally, techniques as dead-reckoning offered some level of independence to the avatars (Singhal and Zyda, 1999). With network support, broadcast or multicast solutions decrease the number of messages used to keep a consistent state of the system (Lee et al., 2002).

Several architectures have been traditionally used for simulating a large set of avatars sharing the same virtual world. Internet multi-player games as Quake and Kali or educational systems as Virtual Education System (VES) are examples of client-server systems (Figure 29c). In these applications, each client computer has a single connection to the only existing server in the system. This server maintains the global state of the simulation, but it becomes a single point of failure in the system. Instead of sending messages to a central server, in peer-to-peer architectures (Figure 29a) avatars exchange messages directly. Several systems have been developed with this architecture, such as NPSNET (Falby et al., 1993).

Although these systems obtain low latencies, they do not properly scale. When the number of avatars greatly increases, clients are not able to handle the amount of messages from other avatars and they simultaneously offer an interactive 3D virtual world to the user. In order to improve scalability, peer-server and server-network architectures group sets of avatars (Figure 29b). Following a peer-server scheme (Figure 29d), systems like ATLAS reduces the volume of information using multicast messaging (Lee et al., 2002). However, this architecture will be useful only when multicast protocols are fully available on the Internet.

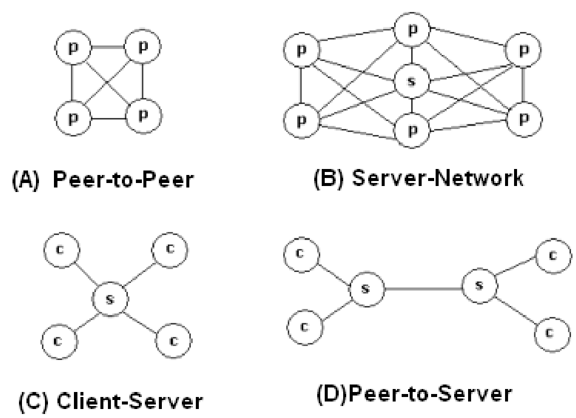


Figure 29 - Architectures

Architectures based on networked servers are becoming a de-facto standard for DVE systems. In these architectures, the control of the simulation relies on several interconnected servers. Figure 29 shows how multi-platform client computers are attached to only one of the servers of the simulation.

In this architecture, when a client modifies an avatar, it also sends an updating message to its server, which in turn must propagate this message to other servers and clients. Servers must render different 3D models, perform positional updates of avatars and transfer control information among different clients. Thus, each new avatar represents an increasing in both the computational requirements of the application and also in the amount of network traffic. When the number of connected clients increases, the number of updating messages must be limited in order to avoid a message outburst. In this sense, concepts like areas of interest (AOI) (Singhal and Zyda, 1999), locales (Anderson et al., 1995) or auras (Greenhalgh, 1997) have been proposed for limiting the number of neighbouring avatars that a given avatar must communicate with. All these concepts define a neighbourhood area for avatars, in such a way that a given avatar must notify his movements (by sending an updating message) only to those avatars located in that neighbourhood. Depending on their origin and destination avatars, messages in a DVE system can be classified as intra-server or inter-server messages (Figure 30).

Intra-server messages are messages exchanged between avatars whose client computers are attached to the same server. On the contrary, inter-server messages are sent when both avatars are assigned to different servers. In order to design a scalable DVE system, the number of intra-server messages must be maximised. Effectively, when clients send intra-server messages they only concern a single server. Therefore, they are minimising the computing, storage and communication requirements for maintaining a consistent state of the avatars in a DVE system.

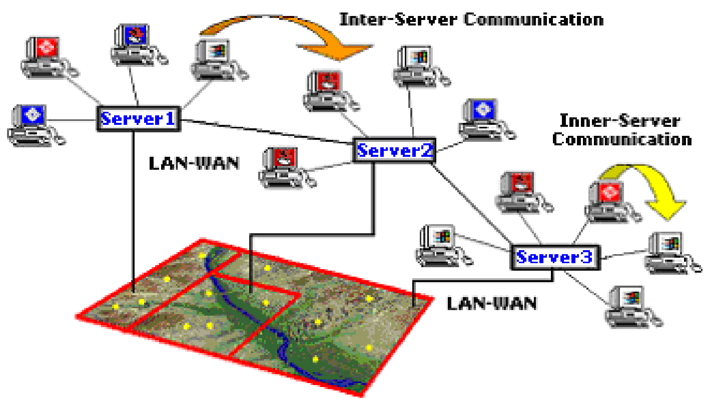


Figure 30 - A multi-server architecture for a basic DVE

In order to design scalable distributed Virtual Environment systems, recent research has led to the definition of different approaches to maintain the system working under its saturation point and maximising system throughput. Moreover, in order to provide quality of service to avatars in a distributed virtual environment systems, avatars should be assigned to servers taking into account, among other factors, system throughput and system latency. This highly complex problem is called quality of service problem in distributed virtual environment systems. Several approaches for solving this problem have been proposed, including the ones from Morillo (http://informatica.uv.es/~pmorillo; P. Morillo et al. 2005) based on modern heuristics (simulated annealing and GRASP). The obtained results from performance evaluation show that the proposed strategies are not only able to provide quality of service to avatars in a DVE system, but also to keep the system away from the saturation point.

7.7 Collaborative Virtual Environments

Collaboration in or with VEs is an emerging and very interesting field of application. VEs can be used for two types of collaboration: collaboration at one VR installation and distributed collaboration at completely separated VR installations.

7.7.1 Collaboration at one VR installation

Currently bigger VR installations like Powerwalls or Caves are used for collaboration in industrial environments. In this context users are performing evaluation sessions mainly to review constructions, design studies, etc. During the sessions one person is tracked (master user) and others are acting as passive observers. Because the perspective is computed correctly for the master user, the perspectives of the other observers are more or less distorted. This can lead to headache, nausea or even sickness. This problem becomes severe if the master user is moving intensively in front of the screen and/or rotating the head very often. As a consequence the master user must be much disciplined and look at the screen instead of looking in the eyes of the conversational partners. This type of collaboration is not suitable for long time usage.

To avoid distorted perspectives each user must be tracked and have its own personal perspective. The respective images should be projected by one pair of projectors per user and all at the same screen.
The difficulty in providing such a solution is the need for correct separation of the projected stereo image pairs. This can be done by a combination of polarised filters for the separation of right and left eye and with active shutter glasses which are separating the correct stereo pair for the user out of the displayed images.
The limiting factors of this approach are:

- The required space and the costs of the projectors. For example, this solution would need 8 projectors for 4 users.
- The perceived brightness is reduced since more users are using the system simultaneously. This is because the length of the time slot available for each user is reduced providing a darker image.
- The quality and speed of the shutter glasses and filters must be very high. Otherwise, the image separation creates disturbing ghosting effects and flickering images.

Currently, prototypes for 4 users are being tested.

Advantages:

- Several users can enjoy an undisturbed view at the virtual world. They can move freely and can look at the conversational partner without destroying the perspective of the other users. Nausea, sickness and headache are not longer a problem.
- This new technology opens a wide field of applications. Psychological research is required to find out the advantages and disadvantages of this type of collaboration.

Drawbacks:

- The reliability of the shutter glasses and filters is still not sufficient.
- The required equipments are expensive.

7.7.2 Collaboration at distributed VR-systems

Stand alone VR systems with one user each can be connected over the network to perform collaborative session. Each user sees the same virtual world but can choose its own viewing perspective and viewing point. The other users might be represented as avatars indicating their positions within the virtual world and there viewing directions. If the representation of their interaction devices is transferred also, the collaboration partners can see where the other users are interacting with the virtual world.

Advantages:
Users can collaborate even over continents. There are many potential applications for that.
Nearly each VR system can be used and connected. There is no special and expensive hardware necessary.
Users have their own perspective and can move freely within the virtual world. They can interact where and with what they want. This can lead to fruitful discussions and efficient work.

Drawbacks:
The greatest disadvantage of this type of collaborations is the lack of visible information about the conversational partner. Users get grip on verbal emotions only. They don't see the partners, only their avatars. This might disturb and can lead to misunderstandings.
More information is transferred and therefore the network should provide sufficient bandwidth for real-time multi-user collaboration. Otherwise the feeling of interaction is disturbed or even destroyed.

Several studies demonstrate how the construction of knowledge (learning) is an inherently social process in which the humans actively construct meaning through a process of information exchange and social interaction with other people. The Collaborative Virtual Environments have the potential of supporting such processes (Churchill et al., 2001).

Collaborative Virtual Environments (CVE) are a subset of Distributed Virtual Environments (DVE) and belong to the field of Computer Supported Cooperative Work. Whereas system throughput (maximum number of the avatars that the system can simulate below the saturation point) and round trip delay of the messages in the network are constantly controlled in common DVE systems, CVE systems focus mainly on new possibilities of interaction, especially concurrent interactions of several users with each others and with one or more objects located in the 3D virtual scene. Therefore, these systems focus on how all actions, movements and expressions are transmitted to all the participants of the simulation.

Nowadays, there are two important challenges in CVE systems. The first is how users perceive each others in the virtual scene. In order to handle this issue, four different approaches have been proposed: *primitive cube-like appearances* (Greenhalgh, 1997), *non-articulated human like* or *cartoon-like avatars* (Benford et al., 1995), *articulated body representations* using rigid body segments (Hagsand, 1996) and *human representations* with animated bodies and faces (Ohya et al., 1994). The second aspect is concerned with how communications (in terms of relations) among the avatars of the CVE system are performed. The fact that the users share the same virtual environment and can interact with it simultaneously enhances their ability to communicate with each other. However, the means of communication that are taken for granted in real life (speech, facial expressions, and gestures) are not necessarily supported in CVE systems due to technical difficulties. Most systems support audio communications and very often a text-based chat capability. Some systems include a means of gestural communications by choosing some predefined gestures (Wolff et al., 2005).

Facial expressions, lip movement, body postures and gestures all play important roles in our everyday communication. Ideally, all these means of communication should be incorporated seamlessly in the virtual environment, preferably in a non-intrusive way. Some approaches recognize this need and present systems where facial expressions are tracked using tape markers while body and hands carry magnetic trackers, allowing both face and body to be synthesised. Figure 31 shows an example of this feature.

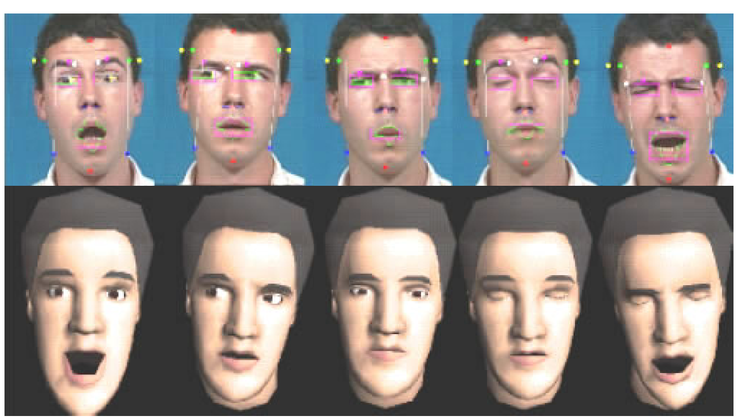


Figure 31 - An example of partial facial communication

7.7.3 Middleware for Developing Distributed Environments

7.7.3.1 MPI

The goal of the Message Passing Interface simply stated is to develop a widely used standard for writing message-passing programs. As such the interface should establish a practical, portable, efficient, and flexible standard for message passing. The main advantages of establishing a message-passing standard are the portability and ease-of-use. In a distributed memory communication environment in which the higher level routines and/or abstractions are built upon lower level message passing routines the benefits of standardisation are particularly apparent. Message passing is a paradigm used widely on certain classes of parallel machines, especially those with distributed memory. Although there are many variations, the basic concept of processes communicating through messages is well understood. Over the last ten years, substantial progress has been made in casting significant applications in this paradigm. Each vendor has implemented its own variant. More recently, several systems have demonstrated that a portable message passing system can be efficiently implemented. It is thus an appropriate time to try to define both the syntax and semantics of a core of library routines that will be useful to a wide range of users and efficiently implementable on a wide range of computers.

MPI-1 (Dongarra et al., 1993) was completed in 1993. It focussed mainly on point-to-point communications. It did not include any collective communication routines and was not thread safe. MPI-2 added some extensions to the basic standard. MPI/RT is the latest instantiation. It incorporates numerous features, such as an API, support for heterogeneous environments, C and Fortran bindings, point to point and collective channels, a reliable communication interface, and thread safety.

7.7.3.2 CORBA and TAO

The Common Object Request Broker Architecture (CORBA) (OMG, 1995) is an open standard for communicating between local and remote processes. On top of packet transmissions, CORBA offers a number of "standard" services which automate many common network programming tasks, such as object registration, location, and activation, request demultiplexing and operation dispatching. At the base level communications are defined by Remote Procedure Calls (RPCs) between processes. The developer defines the interface to their processes using the Interface Description Language (IDL). The IDL is compiled into an interface that exists between clients and servers. Thereafter, the communication details are handled by CORBA. The services mentioned above are then layered over this communication framework.

Initial implementations of CORBA did not deal with real-time processing issues overtly (although this did not preclude a small number of CORBA based VR applications, e.g. COVRA-CAD (Junghyun et al., 1998)). Doug Schmidt and colleagues at the University of Washington State developed The ACE ORB (TAO) (Schmidt, 2006), a CORBA compliant middleware framework, which addresses some of the real-time challenges of distributed processing. Schmidt and colleagues identified a set of patterns and framework components that can be applied systematically to eliminate many tedious, error-prone, and non-portable aspects of developing and maintaining distributed applications.

7.7.3.3 Globus and Grid Computing

There is a number of software infrastructures for Grid computing, e.g. Globus (Foster and Kesselman, 1997); http://www.globus.org/), Legion (Grimshaw and Wulf, 1997) and SNIPE (Fagg et al., 1997)). Services such as authentication, program start-up and data transfer mechanisms are all included in the infrastructures. Globus has been implemented as the distribution mechanism in a number of VR frameworks (including CAVERNSoft (Park et al., 2000)).

7.7.3.4 Common Component Architecture

The Common Component Architecture (CCA) (http://www.acl.lanl.gov/cca-forum) defines a component based communication framework that conceptually sits above middleware such as CORBA or Globus, or lower level IP. There are numerous advantages to component programming when implementing high-performance visualisation tools. This can be realised when considering that the development of virtual reality applications is increasingly a multidisciplinary undertaking. By providing flexible core frameworks, the broader development effort can incorporate research teams with a significant range of skills. Component programming supports the different approaches and requirements inevitable in a multidisciplinary development effort. However, possibly the most interesting facet of component programming to this document is that components can be configured to be executed locally or at remote locations.

CCA constitutes a transparent mean of defining and managing software components by addressing components level interoperability. CCA is the flexible connection to an architecture by a compliant component through a well defined interface. There is a number of CCA compliant frameworks currently in use, including CCAFFEINE (Armstrong et al., 1999) and CCAT (www.extreme.indiana.edu/ccat/). The dataflow model in SCIRUN (Parker, 1999) also incorporates a CCA compliant framework.

7.7.3.5 CACTUS

Cactus (Allen et al., 2000; http://www.cactuscode.org/) is an open source problem-solving environment. It is modular and designed to be implemented across a distributed architecture. Cactus has a central core (or flesh), which connects to application modules (or thorns). User-defined applications form the thorns. Thorns can be combined to provide an application solution which inherently incorporates visualisation. Cactus can be configured to run across a Grid or through a more conventional MPI communication mechanism.

7.7.4 Distributed Virtual Environments

This section summarises the key distributed virtual environments developed by various research teams.

7.7.4.1 SIMNET, DIS, HLA and NPSNET

In 1983, the Defense Advanced Research Projects Agency (DARPA) sponsored the SIMNET (SIMulation NETworking) program to create a new technology to expand the current single task trainers into networked team trainers. SIMNET was tremendously successful, producing over 300 networked simulators with the technology that was to develop into Distributed Interactive Simulation (DIS).

DIS and its predecessor SIMNET have set standards for distributed interactive simulations. This work has been the base for the development of the High Level Architecture (HLA) for distributed simulations.

DIS has radically changed the simulation-based training process. By connecting together many types of simulations into a shared virtual world, DIS dramatically increases the training benefits from simulation. Using DIS, trainees (e.g. soldiers) now train like they operate (i.e. in teams).

The foundation of DIS is a standard set of messages and rules, called Protocol Data Units (PDUs), used for sending and receiving information across a computer network. The most common message is the Entity State PDU which represents all of the state information about a simulated entity that another simulator needs to know.

For example, an Entity State PDU contains data about an entity's position and velocity. By using the position, velocity, acceleration, and rotational velocity data, a receiver is able to extrapolate, or dead reckon, a vehicles' position before the arrival of the next PDU, thereby reducing consumption of network bandwidth. Using this technique, DIS is able to limit the amount of data an average simulator transmits to approximately 250 bytes per second. Optimizations, such as dead reckoning, permit very large virtual battles to take place. The largest DIS exercise, part of DARPA's Warbreaker program, had 5,400 simulated entities interacting in a single DIS virtual world.

The fact that there is no central server is perhaps the most surprising DIS characteristic. DIS is strictly a peer-to-peer architecture, in which all data is transmitted to all simulators where it can be either rejected or accepted depending on the receivers' needs. By eliminating a central server through which all messages pass, DIS dramatically reduces the time lag needed for a simulator to send important information to another simulator. This time lag, known as latency, can seriously reduce the realism, and therefore the effectiveness, of a networked simulator. For example, it is vital that when one simulator fires at another simulator the target is made aware of the incoming ammunition as soon as possible to allow it to take the appropriate defensive action. Any delay introduced by the training device results in negative reinforcement to the trainee.

As DIS matures, the US Department of Defense (DoD) looked ahead to the next generation of modeling and simulation software that will support a wider range of applications with more functionality. The US DoD's Defense Modeling and Simulation Office (DMSO) has led a DoD-wide effort to establish a common technical framework to facilitate both the interoperability, between the wide spectrum of modeling and simulation applications, and the re-use of the modeling and simulation components. This common technical framework includes the High Level Architecture (HLA) and is considered the highest priority effort within the DoD modeling and simulation community.

The HLA effort (http://www.alibre.com; http://www.groove.net) was first initiated to facilitate the interoperability of broadest range of component-based simulations. Although HLA did not originate as an open standard, it has now been recognized and adopted by both the Object Management Group (OMG) and IEEE.

The HLA can be considered as an object-oriented net-VE design. Each simulator, known as a *federate*, is a component which represents a collection of objects, each having a set of attributes and capable of initiating and receiving events. The federate registers each of its objects with a piece of middleware called the Run-Time Infrastructure (RTI). The RTI collaborates with RTI instances on other hosts to learn about remote participants (objects) and delivers information about those participants to the local federate. The local federate, in turn, typically instantiates local objects representing those remote participants. Attribute updates and events are also exchanged through the RTI, which is responsible for handling area of interest management, time synchronization, and other low-level net-VE services on behalf of the application. The collection of federates, along with their associated RTI instances, is termed a *federation*. All the simulators in the federation send and receive state information via calls to and from the RTI (http://www.ptc.com/products/division/MockUp.htm).

More specifically, the HLA defines a set of rules governing how federates, interact with one another. The federates communicate via a data distribution mechanism called the Runtime Infrastructure (RTI) and use an Object Model Template (OMT) which describes the format of the data. The HLA does not specify what constitutes an object (objects are the physical things that are going to be simulated, such as tanks and missiles), nor the rules of how objects interact. This is a key difference between DIS and the HLA.

The RTI lets different types of systems interact. These systems can include simulations which run faster than real-time and simulate objects which are hierarchical aggregates of individual entities (platoons, companies, or battalions) all the way to high-fidelity engineering models which run much slower than real time and simulate individual subsystems with very high accuracy. In the DIS paradigm these two applications would not be able to interact.

However, the strength and flexibility of HLA is also a potential weakness – unless all the HLA simulators in an exercise agree on a single Federate Object Model (FOM) they will not be able to interoperate even though they are HLA compliant. The FOM describes the objects and interactions involved in the federation execution.

Besides facilitating interoperability between simulations, the HLA provides federates with a more flexible simulation framework. Unlike DIS where all simulations receive every piece of data broadcast, federates now have the ability to specify:

- What information they will be producing
- What information they would like to receive
- The data's transportation service, e.g. reliable, best effort
- Whether or not the federation's timing mechanism is synchronous or asynchronous.

The above points make it possible to have more simulations on a network at one time because the amount of data being sent is reduced. The simulation software is also simplified because it does not need to process extraneous information.

The two major components of HLA are the RTI and the OMT. The OMT provides a standard format for describing a simulation in terms of its objects and the interaction between objects.

Again, objects are the physical things that are simulated, and interactions are the events that occur in simulations, such as detonations and collisions.

As previously stated, the RTI's primary function is that of a data distribution mechanism. Federates send information through the RTI which distributes the information to the appropriate parties. The RTI does not maintain information about the state of the federation; nor does it handle any semantics associated with the interaction between the federates, such as what coordinate system to use, what happens during a collision, or how to dead-reckon remote vehicles.

Moreover, the RTI does not specify the exact byte layout of data sent across the network. The RTI provides a common set of services to the federates. They can be divided into six categories:

1. Federation Management: Handles the creation, dynamic control, modification, and deletion of a federation execution.
2. Declaration Management: Enables federates to declare to the RTI their desire to generate (publish) and receive (subscribe/reflect) object state and interaction information. Federates can subscribe to only the objects they want (or have the capability) to receive, e.g. tanks might need only data pertaining to ground movement, or airplanes might only need data pertaining to flight activities
3. Object Management: Enables the creation, modification, and deletion of objects and interactions. These services comprise most of the network traffic during runtime.

4. Ownership Management: Allows federates to transfer ownership of object attributes to other participants in the simulation.
5. Time Management: Provides useful services for setting, synchronizing, and modifying simulation clocks. Time Management services are tightly coupled with the Object Management services so that state updates and interactions are distributed in a timely and ordered fashion.
6. Data Distribution Management: Federates can provide conditions governing when to start or stop transmitting and receiving certain pieces of data.

As the training and simulation industry moves toward HLA compliance, existing DIS applications will need to be updated in order not to become obsolete. There are four techniques to make the transition from DIS to HLA: translator, wrapper, native, and protocol interface unit (PIU). Some of these techniques are easier and more cost-effective than others, and each has its own advantages and drawbacks. Table 6 illustrates the benefits of the four approaches.

	Translator	Wrapper	Native	PIU
Forward Compatibility	X	X	X	X
Backward Compatibility	X			X
Ease of use	X			X
Low Latency		X	X	X
Scalability		X	X	X
Takes full advan-tage of HLA			X	X

Forward compatibility: Technique's ability to be upgraded to newer versions of HLA
Backward compatibility: Technique's ability to switch between HLA and DIS
Easy of use: Requires only limited modifications to existing simulation software
Low latency: Technique does not cause a delay in between sender and receiver
Scalability: Technique's ability to interface with a large number of simulations

Table 6 – 4 Approches for DIS to HLA transition

The techniques are discussed in detail below:

7.7.4.1.1 Translator

Using this technique, a separate application (often another computer) is placed on the network to translate network traffic between the different protocols (see Figure 32). The Translator technique requires a separate application or hardware device to manage communications between applications using different protocols.

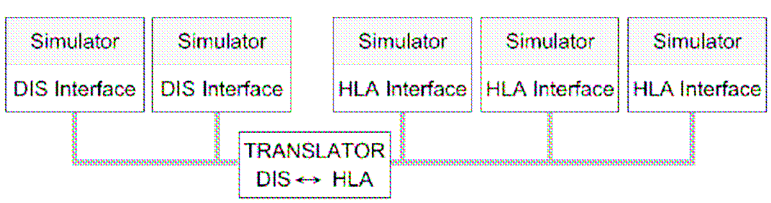


Figure 32 - DIS-HLA Translator

A translator requires no software modification to the simulator, but because data must travel though this extra piece of hardware, the simulator's latency increases by roughly a factor of ten. Having all traffic passing through one computer is risky since it puts a single point of failure into an otherwise distributed system. The translator permits limited forward and backward compatibility, but limits the scalability and flexibility of the simulator. Also, when using a translator, the simulator cannot take advantage of future HLA features.

7.7.4.1.2 Wrapper

With a wrapper approach, software is added underneath the simulation's DIS interface to translate the data from the old protocol (DIS) to the new protocol (HLA) just before it is sent and to translate the data from HLA to DIS just after it is received. The Wrapper technique links additional code to a DIS application to provide interoperability with HLA applications (see Figure 33).

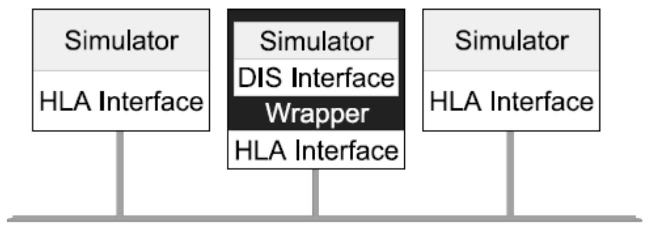


Figure 33 - Wrapper

Unlike the translator, a wrapper does not require additional hardware. All the changes are made via limited modifications to the simulator's software. However, forward and backward compatibility requires further software changes, and as the translator, the wrapper does not allow the simulator to benefit from HLA specific features.

7.7.4.1.3 Native

Creating a native HLA simulation implies that all interfaces to the network are contained within the simulation software. The native technique (Figure 34) requires that the simulation software contain all necessary interfaces to the network.

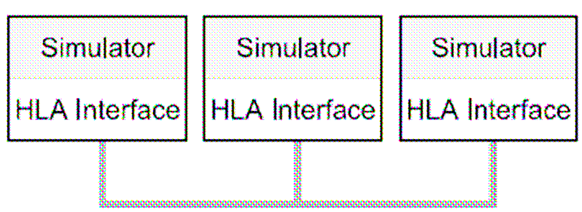


Figure 34 - Native HLA Simulation

A native HLA simulator can take full advantage of all HLA features. However, these advantages come at the expense of huge software modifications at the initial transition and then additional modifications for any future protocol changes. Besides, there is no backward compatibility.

7.7.4.1.4 Protocol Interface Unit (PIU)

The simulation interfaces with the network via a software system known as a PIU. A good PIU, such as MÄK Technologies' VR-Link, will have one API which supports all the features in both protocols, DIS and HLA. A less capable PIU will limit functionality to the lowest common denominator, and the simulator will be unable to take advantage of any features unique to either protocol.

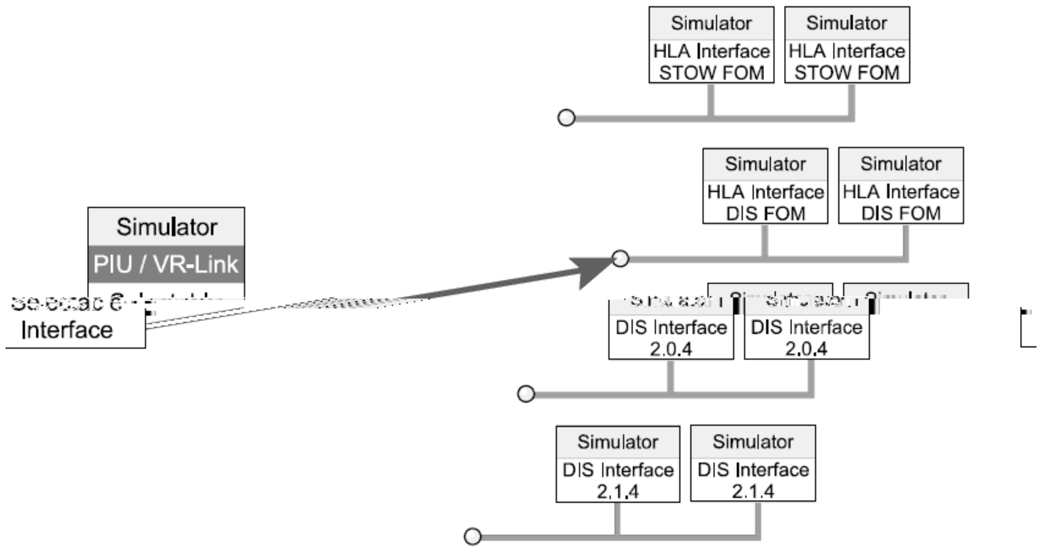


Figure 35 - Protocol Interface Unit

The Protocol Interface Unit (Figure 35) technique supports all features of both protocols, and allows switching among different FOMs within HLA.

The PIU approach may be the best technique to update a DIS simulator to HLA. It provides an easy upgrade path to HLA, while maintaining backward compatibility with DIS. Using a PIU also permits a simulator to switch among different FOMs within HLA and even different versions of the DIS protocol. A PIU requires only minimal modifications to the simulation software and provides the most flexibility when designing a new simulation. On the downside PIUs can be complex and expensive to write and to maintain.

7.7.4.2 DIVE

DIVE (Distributed Interactive Virtual Environment) (Duda, 2000; http://www.sics.se/dive/) was developed by the Swedish Institute of Computer Science. The development effort concentrated primarily on the distribution component of the Virtual Environment. Certainly, in its early days DIVE emerged as an important example of how to develop distributed Virtual Reality. DIVE implements a distributed dynamic database. It has the capability for adding new objects and modifying the existing databases in a reliable and consistent fashion. Reliable multicast protocols and concurrency control are employed through a distributed locking mechanism to facilitate database updates.

7.7.4.3 MASSIVE

MASSIVE (Faugeras, 1993; http://www.crg.cs.nott.ac.uk/research/systems/MASSIVE/) is an object oriented system developed at the University of Nottingham to support multi-user collaborative virtual environments. MASSIVE is based on a 'spatial model' of interaction (Foley et al., 1996). It is mainly developed as an academic research tool, for researchers investigating online collaboration in virtual environments. The work is focussed on promoting awareness of others in the virtual environment. The system supports interaction with the virtual environments via text, 2D and 3D graphical interfaces. It also supports real-time audio, allowing users to communicate with one another using speech. The emphasis is on user-to-user interaction, rather than a user's interaction with objects in the environment. MASSIVE uses a combination of peer-to-peer and client/server processes. It employs multicasting as a collaboration baseline.

7.7.4.4 AVANGO

Avango is developed by the German National Research Center for Information Technology (GMD) (Goldberg, 1989; Gonzalez and Woods, 2002; http://www.avango.com/). It provides an extensible object oriented framework for constructing networked Virtual Environments. Avango is built on a dataflow model using fields within the objects to support a generic streaming interface. The dataflow graph defines the behavior of the objects in the world. It provides a replicated scene graph across the network. The objects can be instantiated as either local or distributed (public or private). It defines two categories of objects: Nodes (scene graph elements for rendering) and Sensors (which import external device data into the application).

Avango has a C++ API and a binding to an interpreted language called Scheme (Haykin, 1998). Typically complex and performance critical functionalities are implemented in C++. From then on the application is implemented using Scheme scripts.

Avango uses a process group model. Each group member is guaranteed to receive the delivery of network messages in exactly the same order. In addition, when a new member joins the group, communications are suspended until the new member's state is updated. This guarantees state consistency.

7.7.4.5 DEVA/MAVERIK

DEVA (Jain et al., 1995) is a distributed Virtual Reality kernel developed by the Advanced Interfaces Group at the Manchester University. DEVA decouples a virtual object's representation into client side and server side objects. The server side part represents the virtual object and defines its behaviour, part of which may be generic and part may be specific to the application. The client side part contains interpretations of this behaviour, and only responds to instructions from the server. The messages from the server to the client are similar to a high-level vocabulary used to describe the effect of the behaviour. DEVA establishes a locus of control by managing state in one place. This means that the state of an entity can be managed locally rather than in the server if this is appropriate. Rather than using dead-reckoning to combat lag and jitter, DEVA uses "twines" (Kanade et al., 1995), which smooth over the irregular updates.

DEVA is tightly coupled to a VR kernel called MAVERIK (Kanade et al., 1997) which provides rendering, input, output and spatial management capabilities. The main advantage this gives is that the application data does not need to be duplicated and stored in a specific data structure such as a scene graph. Instead, MAVERIK promotes the use of an application's own data structures. This avoids the need to conform to a rigid and potentially inappropriate data structure.

7.7.4.6 COVISE

COVISE (Collaborative Visualisation and Simulation Environment) exploits high-speed network infrastructures in distributed computing and collaborative engineering (Kang, 1999; http://www.hlrs.de/organization/vis/covise/). COVISE is predominantly focussed on visualising high end supercomputing problems. The database is fully replicated across nodes at the start. All of the user inputs, system outputs and system administration tasks are handled at a single point so reducing the complexity of keeping all users synchronised.

7.7.4.7 CAVERNsoft

CAVERNsoft (Torguet et al., 2000; http://www.cavernsoft.org/), based on a client server model, combines a networking library and a database. It facilitates the development of collaborative interactive Virtual Environments. Applications use the Information Request Broker (IRB) to mediate communication between network instantiations. Users can request what type of network connection they wish to use: CAVERNsoft supplies the option of TCP/IP, UDP or IP/Multicast. The user defines the desired bandwidth and acceptable network attributes (latency and jitter) and the remote IRB attempts to comply with these requests. In this way the user is able to specify the desired quality of service expected from the distributed environment.

7.7.4.8 VRJuggler

VR Juggler (Ma et al., 2000; http://www.vrjuggler.org/) is developed at Iowa State University's Virtual Reality Applications Center. It is an object-oriented framework written in C++. It is designed to enable the developer to put VR applications together quickly with elements of reusable code by abstracting the system details to plug-in modules.

Performance is a central concern of Juggler. To this end the Environment Manager provides the developer with the ability to monitor performance data. It is also possible to reconfigure the application at run-time to adjust performance-related properties. VRJuggler is designed for cross platform development and supports the SGI, NT and HP platforms. Networking support is provided through an abstract Network manager. The network manager provides an indirection to a network implementation, which is pluggable at runtime.

7.7.4.9 BAMBOO

Kent Watsen's Bamboo (Matthews et al., 2002; http://watsen.net/Bamboo/) is a VR framework based heavily on software components which can be dynamically loaded at run-time. It is the combination of the modules that defines the application. Generic modules can be used to support I/O and other common tasks. The application developer implements application specific modules which refer to other modules known to the system.

Bamboo is built on top of the ADAPTIVE Communication Environment (ACE) (McLachlan, 1992), which provides the system with networking, concurrency (threading) and synchronisation functions. The Bamboo rendering classes are built on top of the scene-graph based OpenGL++ (Michalewicz, 1996).

7.7.4.10 ViRAL

ViRAL (Virtual Reality Abstraction Layer) (Parker, 1997) is a graphical component-based framework for VR applications development. Its most important features are its simplicity, a component-based design and integrated graphical user interfaces. The ViRAL frameworks offer reusable designs and constructors for the development of VR applications.

7.7.4.11 Amira/Avizo Teamwork

Support for collaboration was recently introduced in Amira and derived Avizo - 3D data visualization and analysis software (http://3dviz.mc.com/products/amira_VR_overview.asp) : work sessions can be shared remotely across multiple systems – whether desktop or VR. Each collaborator represented by an avatar can alternatively become operator or spectator, either with synchronized or independent view.

7.7.5 Relevant Standards in Distributed Virtual Environments

This section presents the relevant activities which can contribute towards establishing standards for developing distributed virtual environments in the future.

7.7.5.1 HLA

Distributed simulation is an application of distributed systems technology that enables simulators to be linked together over networks such as the Internet so that they work together (or interoperate) during a simulation run. The HLA (High Level Architecture) is a standard that defines the distributed system technology to make this interoperability possible. Rather than a networking protocol (wire standard) like DIS, HLA defines an architecture with a set of API (Application Programming Interface) Standards. Simulation applications (known as federates in HLA) communicate by making calls to the HLA APIs. A piece of software known as the RTI (Run-time Infrastructure) implements the HLA API, and is responsible for transporting data from one federate to another. Like DIS, the HLA Standards are owned by IEEE. There are three documents that comprise the HLA Standard, all available from IEEE (Pedrycz and Gomide , 1998):

- IEEE 1516-2000 - IEEE Standard for Modelling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules: This provides the rules and definitions for implementing and using HLA (IEEE product code SH94882).
- IEEE 1516.1-2000 - IEEE Standard for Modelling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification: This defines the various services provided by an HLA RTI, and contains the APIs (IEEE product code SH94883).
- IEEE 1516.2-2000 - IEEE Standard for Modelling and Simulation (M&S) High Level Architecture (HLA) - Object Model Template (OMT) Specification: This defines the format used for describing object models in HLA. An object model dictates what kinds of data a particular set of HLA federates will exchange (IEEE product code SH94884).

There is a fourth document that is not technically part of the definition of HLA, but that defines some of the recommended practices for using HLA. It is called IEEE 1516.3-2003 - IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP) (IEEE product code SH95088).

While the IEEE 1516 series of standards represents the "current" version of HLA, many HLA simulations are still using an earlier version of HLA known as HLA 1.3. This version was maintained by the U.S. Defense Modeling and Simulation Office and the U.S. Department of Defense prior to IEEE Standardization.

A Commercial-off-the-shelf (COTS) simulation package is a term used to refer to software used by many simulationists to build and experiment with models. There have been various attempts to support interoperability between COTS simulation packages. Currently these approaches are not compatible. The HLA – COTS Simulation Package Interoperability Forum (HLA-CSPIF) has been created (Schalkoff, 1989) in an attempt to support interoperability and to unify research and development activities in this area.

7.7.5.2 W3C

The World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. W3C is a forum for information, commerce, communication, and collective understanding (Scharstein and Szeliski, 2002). In its document "Architecture of the World Wide Web", the core design components, constraints and good practices to the principles and properties of the World Wide Web are described as relatively simple technologies with sufficient scalability, efficiency and utility. The core design components of the architecture are the identification of resources, representation of resource state, and the protocols that support the interaction between agents and resources in the information space.

7.7.5.3 IETF

The Internet Engineering Task Force (IETF) (Sonka et al., 1998; http://www.ietf.org/overview.html) is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. It is open to any interested individual. The actual technical work of the IETF is done in its working groups, which are organised by topic into several areas, such as routing, transport, security, etc. One of their main goals is to define standards for these areas facilitating the compatibility, the interoperability and the exchange of information. Many architectures are being defined by the groups, such as for IPv6 addressing, TRILL, IPFIX, network protection, Path Computation Element (PCE), RSerPool, Secure Mobile and XCON.

7.7.5.4 WSRF

Web services must often provide their users with the ability to access and manipulate state, i.e. data values that persist across and evolve as a result of Web services interactions. While Web services successfully implement applications that manage state today, we need to define conventions for managing state so that applications discover, inspect, and interact with stateful resources in standard and interoperable ways. The WS-Resource Framework (Sturman, 1992) defines these conventions and does so within the context of established Web services standards.
Initial work on the WS-Resource Framework has been performed by the Globus Alliance and IBM, who released initial architecture and specification documents with co-authors from HP, SAP, Akamai, TIBCO and Sonic for public comment and review on January 20, 2004. These documents were submitted to the OASIS standards group in March 2004. The WSRF Technical Committee has been formed to work on WS-ResourceProperties, WS-ResourceLifetime, WS-ServiceGroup, and WS-BaseFaults specifications. The WSN Technical Committee has been formed to work on WS-BaseNotification, WS-Topics, and WS-BrokeredNotification specifications.
The WS-Resource Framework is inspired by the work of the Global Grid Forum's Open Grid Services Infrastructure (OGSI) Working Group. Indeed, it can be viewed as a straightforward refactoring of the concepts and interfaces developed in the OGSI V1.0 specification in a manner that exploits recent developments in Web services architecture (e.g. WS-Addressing).

7.8 Virtual Machine Technology for VR, VE and Future Workspaces

Virtualisation facilitates heterogeneous operating systems to operate on virtual machines (VMs) installed on the same physical computer. As shown on Figure 36, each VM consists of an operating system (OS), virtual hardware and application programs (E.g. Collaboration system). The operating system of VM is called the guest OS that works on a consistent, normalised set of virtual hardware devices and components (VMware, Inc, 2006; http://www.vmware.com/).

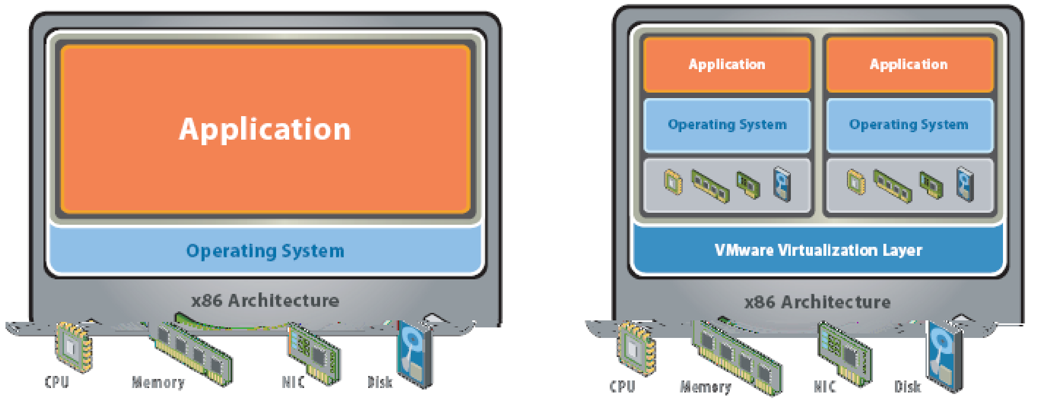


Figure 36 - Before and after Virtualisation of x86 Architecture with the VMWare Virtualisation technology

VMs are highly portable and deployable in seconds to many distributed sites as a condensed file of any filing system format. The application systems configuration and management, networking, data backup, etc. become streamlined and do not require constant attention. Also, server upgrades and maintenance, server moving, and replication could be performed with zero downtime. Hence, virtualisation provides the most efficient and effective deployment strategy for setting up infrastructures for VR, VE and Future Workspaces. The users could avoid the inefficient, time consuming and costly routines for obtaining, managing and maintaining new resources by utilisation of VMs.

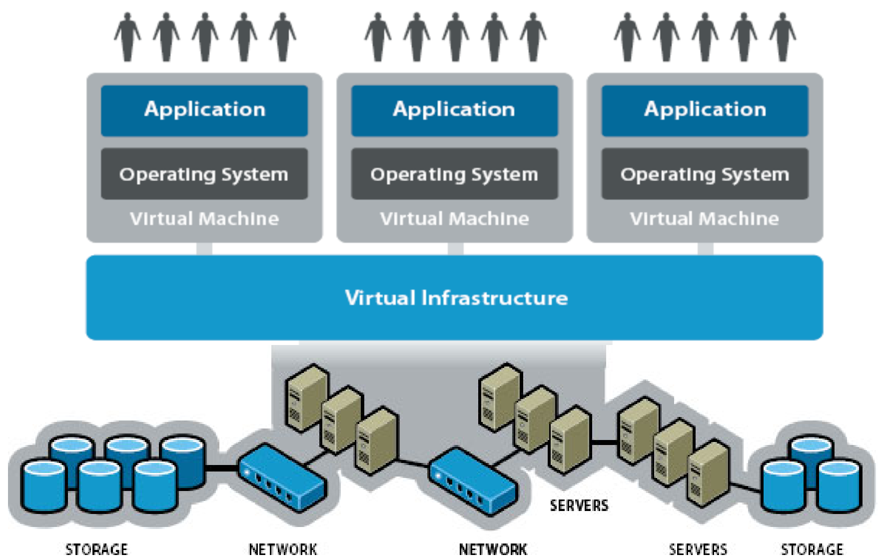


Figure 37 - VMware Virtual Infrastructure

Existing computer infrastructure within an organisation could be converted into Virtual Infrastructures (VIs) as figured below. This type of extension of the technology facilitates transforming farms of individual servers, storage and networks into a pool of resources that could be shared among many VMs to serve different needs of users. The VMs could be networked into virtual networks, similarly to physical networks. Typical VI that consists of many servers could also be geographically dispersed.
The new technology processors such as Intel VT have extended hardware virtualisation capabilities further so that the processor architecture could be able to overcome the OSs inability of supporting resource sharing features. The technologies such as Intel VT and VMware Visualisation provide

enterprise grade visualisation mechanisms that are highly suitable for designing virtual infrastructures for VR, VE and future workspaces.

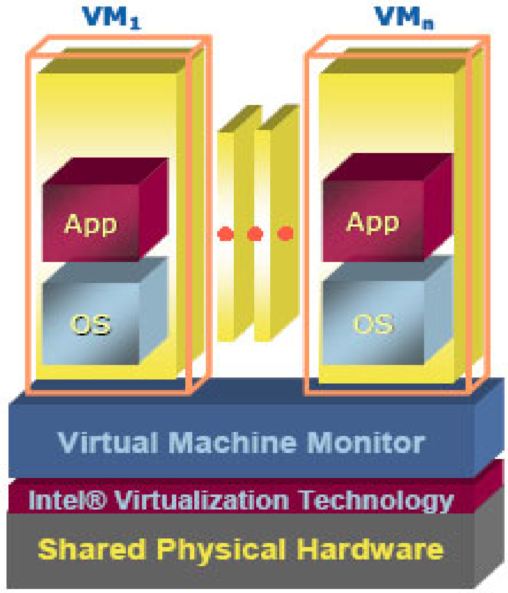


Figure 38 -Virtualisation Technology: Intel Approach

Figure 38 describes Intel VT and VMM architecture. Utilisation of multiple virtual machines (VM_1...VM_n) could enhance the efficient use of computer resources.
The development and testing of applications under different platforms (Hardware environments and various OS) could be facilitated by this technology within a single physical computer. This also provides the ability to test and deploy 32bit and 64bit applications on the same computer with the assumption that the physical hardware and base OS could support 64bit processing.

8 VE Systems for Industrial Applications

8.1 Introduction

Virtual Reality (VR) technologies offer a good opportunity to improve and optimise typical processes in the Industry. A vast amount of potential applications can be envisaged. While some of them may still fall into the R&D side (due to technological or cost barriers) many others are now showing cost benefits. The continuous decrease of the associated costs and the increasing capabilities of the VR technology have slowly increased the technology penetration. This is especially true for the so-called desktop VR.

8.2 Application Areas

8.2.1 Scientific and Information Visualisation

The main goal of scientific visualisation is to present abstract sets of data in such a manner that information carried by these data may be easily perceived and understood by humans. Such presentation needs to support knowledge recovery. Therefore, the cognitive aspects of presentation metaphors and methods need to be carefully investigated and incorporated. The use of virtual reality visualisation technologies in scientific visualisation is natural. In most cases, scientific applications use 3D diagrams and plots displayed on a standard 2D monitor with the use of common (e.g., VRML) or specialised browsers (fish tank technology). However, a number of visualisation systems use more immersive techniques like CAVE or HMDs. Selection of the visualisation environment and technology depends mostly on the area of application and on the targeted audience. Examples of projects and products related to scientific visualisation are described below, categorised into a number of application domains:

- Biology and medicine
- Mathematics
- Aerospace
- Geosciences, meteorology and oceanography
- Chemical engineering
- Large datasets
- Physics

8.2.1.1 Biology and medicine

Research in molecular biology and Bioinformatics developments focused on the architecture of the genomes and interactions in the living organisms cells, take up a major challenge for the dynamic comprehension of the organisation of cellular activity on various scales. The spatio-temporal representation, exploration and visualisation then become key stages in order to help the biologists for better targeting their research and their studies. Virtual Reality and Data Integration should offer user friendly and powerful interfaces.

As an example of VR researches in this field, the transversal action on Virtual and Augmented Reality "VENISE" (http://www.limsi.fr/venise) is developing a Genomic VR platform based on two software tools: ADN-Viewer (Gherbi and Hérisson, 2001; Hérisson et al., 2004) and Genome3Dexplorer (Férey et al., 2004b). One of the major scientific stakes of this Genomic VR platform is the immersive visualisation and exploration by the contents of genomic and biological information. With this intention, it was necessary to undertake research in modelling and structuring and scene management of these huge

3D data. ADN-Viewer is an interactive software environment focused on 3D representation and immersive visualisation of the DNA sequences. This software offers a representation of the 3D structure that the double-helix would take apart from any interaction with other macromolecules. It can be observed in an immersive way on all integration levels, from gene to chromosome. This software was augmented by a work aiming at the representation of the "biological" contents of DNA sequences. This representation helps users to study the spatial organisation of these contents and their interactions. That was made possible by the development of a middleware called GenoMEDIA (Féreyet al., 2004), which integrates ADN-Viewer, genomic databases and a multimedia server. On the other hand, Genome3Dexplorer software addresses the representation, visualisation and exploration of huge genomic and biological databanks. The approach is based on a dynamic graph model and exploiting XML concept. The main interest of this other software is to explore in a homogeneous way huge masses of complex and heterogeneous data. It exploits for the visualisation a federator format to take into account factual as well as textual data.

Concerning the particular field of docking, the problems, methods and algorithms are well reviewed (Halperin et al., 2002; Morris et al., 1998; Camacho and Vajda, 2002; Fernandez-Recio et al., 2002). Researches on the link between VR and molecular modelling or docking exist, but only for the visualisation part of the problem (Stevens and Judson, 1997). Nagata et al. (2000) are first suggesting the use of haptic feedback but without any immersion or multimodality, and specifically for molecular modelling. Trindade et al. (2001) have studied the immersive simulation of simple molecules (H20…) but not the docking problem in itself. Koutek et al. (2002) have built a workbench for small molecules (2000 atoms) with a LaserBeam device but again with no docking.

8.2.1.1.1 Crumbs

Crumbs is a virtual environment tracking tool for biological imaging. Crumbs (http://csdl.computer.org/comp/proceedings/biomedvis/1995/7198/00/71980018abs.htm) is designed for "inside-to-outside" viewing, allowing a user to be fully immersed in the data. Crumbs uses different modalities, such as audio and video. Audio is used as a natural command interface to the Crumbs system. In addition, the developers of Crumbs have invested significant effort to accelerate volume rendering.

8.2.1.1.2 The Proteus System

The Proteus system (http://www.cwi.nl/~robertl/papers/2002/vr/paper.pdf) is a virtual environment for the exploration of multi-channel, time-dependent, 3D images obtained by confocal microscopy. The goal of Proteus is to provide an environment in which a biologist is seated in front of a dataset describing a three-dimensional cellular structure which can be scaled, probed and manipulated as if it were a real cell. Proteus provides a number of visualisation techniques, which are useful for the display of structural characteristics of 4D biological data. These techniques include the interactive control over volume and iso-surface rendering parameters (e.g., transfer functions and iso-values), data selection tools to cut and paste data regions, free hand drawing tools for path tracing, distance measuring tools, and tools for tracking features in the data. Proteus has been written using PVR, an in-house system for portable virtual reality applications, and may be used in various environments, such as desktop workstations, large fish tank displays and CAVE systems.

8.2.1.1.3 3DIVE

3DIVE is an immersive system for interactive volume data visualisation and exploration inside the CAVE virtual environment. Combining interactive volume rendering and virtual reality provides a natural immersive environment for volumetric data visualisation. More advanced data exploration operations, such

as object level data manipulation, simulation and analysis, are supported in 3DIVE by several new techniques. In particular, volume primitives and texture regions are used for the rendering, manipulation, and collision detection of volumetric objects; and the region based rendering pipeline is integrated with 3D image filters to provide an image-based mechanism for interactive transfer function design. The system has been released as public domain software for CAVE/ImmersaDesk users, and is currently being used by various scientific and biomedical visualisation projects.
(http://www.cs.iupui.edu/~sfang/cadcg.pdf)

8.2.1.1.4 3D Visible Human Project

The 3D Visible Human Project (http://www.nlm.nih.gov/research/visible/visible_human.html) is a 3D reconstruction of a human body from a digital image dataset of a complete human male and female cadaver, with digitised anatomical photographs, as well as magnetic resonance imaging (MRI) and computer tomography (CT) data.

8.2.1.1.5 The Vesalius Project

The Vesalius Project (http://cpmcnet.columbia.edu/vesalius/), at the Columbia University, focuses on three main goals:

1. Generating a gallery of 3D photographic quality colour models of anatomical (sub)-structures, obtained from the Visible Human Data sets, which can be manipulated in real-time on any platform and transmitted efficiently. In addition, creating interactive 3D models of anatomical structures from the photographs (e.g., skull).
2. Developing an anatomical knowledge base containing a "nucleus" of anatomical information specifically designed to support a wide spectrum of curriculum applications (ontology).
3. Developing applications, in particular 3D interactive anatomy lessons for the anatomy curriculum, using the models and other visual material from the 3D gallery and content from the ontology.

8.2.1.1.6 BioCoRE and interactive molecular dynamics (IMD)

BioCoRE (http://www.ks.uiuc.edu/Research/biocore/), which stands for Biological Collaborative Research Environment, is a network-centred meta-application that has four components:

- a workbench that provides analysis tools, data sharing, resource allocation, simulation control, and interactive molecular dynamics,
- conferencing that includes audio and visual communication, visualisation, and training,
- a notebook for record keeping,
- a documentation capability.

In the BioCoRE environment, researchers use its IMD molecular visualisation program and simulation engine to display 3D models of molecular systems such as proteins, nucleic acids, and lipid bi-layer assemblies. In addition, a force feedback tool enables researchers to, in effect, poke at a system or change its shape to see how it responds.

8.2.1.1.7 Telescience for advanced tomography applications

This project (https://telescience.ucsd.edu/) is sponsored by the National Institute of Health and the National Center for Research and Resources. It builds Web-based collaboration tools providing remote access to high resolution, 3D microscopy in biomedical and neuroscience research. Using these tools, scientists are able to access and work with data from specialized imaging instruments such as high and ultra-high voltage electron microscopes at remote facilities. The telescience tools provide transparent access to supercomputing resources required to produce, refine, and analyse complex 3D images of cellular and among archived research results, the collaboration tools can help molecular biologists nationwide study the basic structures of cellular organelles like mitochondria and debilitating disorders such as Parkinson's and Alzheimer's disease.

8.2.1.2 Mathematics

8.2.1.2.1 Construct3D: A Virtual Reality Application for Mathematics and Geometry Education.

Construct3D (http://www.ims.tuwien.ac.at/research/construct3d/) is a three-dimensional geometric construction tool based on the collaborative augmented reality system called Studierstube (Schmalstieg et al., 2002). It uses a stereoscopic head mounted display (HMD) and the Personal Interaction Panel (PIP) – a two-handed 3D interaction tool that simplifies 3D model interaction. The application is used for mathematics and geometry education at high school and at university level.

8.2.1.2.2 The PORTAL: Parallel Optical Research Testbed And Laboratory.

The PORTAL (http://www.math.tu-berlin.de/geometrie/f5/portal.shtml) is a virtual reality theatre aimed at visualisation of complicated mathematical structures. For visualisations of complicated structures in three dimensions, immersive virtual reality as in the PORTAL, which gives the user the feeling of moving around the objects being viewed, is vastly superior to the two-dimensional view presented on a workstation screen.

8.2.1.3 Aerospace

Concerning Computer Fluid Dynamics (CFD), the idea that immersive visualisation of fluids can improve data understanding is no longer a new one. For example, a collective work published in 1994 explicitly mention the challenge of applying VR to aeronautic flow visualisation: "*The virtual wind tunnel* (Bryson and Levit, 1992; http://www.cavs.msstate.edu/research/windtunnel.html) *uses virtual reality to facilitate the understanding of precomputed simulated flow fields resulting from computational fluid dynamics calculations. The visualization of these computations may be useful to the designers of modern high-performance aircraft. The virtual wind tunnel is expected to be used by aircraft researchers in 1994 and provides a variety of visualization techniques in both single-user and remotely located multiple-user environments*" (Durlach and Mavor, 1994). Indeed, several publications have presented significant advances in stereoscopic and sometimes immersive flow visualization, in the mixing domain (Mann et al., 2001), in aerodynamics (Wesche, 1999), and more generally in post-computing of numerical simulations (Belleman et al., 1998; Wasfy and Noor, 2001). Concerning post-computing tools, there exists a number of services and industrial code development companies like CHAM (http://www.cham.co.uk/phoenics/d_polis/d_enc/enc_vr.html) with its Virtual-Reality user interface (PHOENICS-VR) or POWERFLOW (http://www.exa.com/pdf/PowerVIZscreen.pdf) with the PowerZIZscreen. Commercial standard CFD visualisation software tools also propose VR solutions (e.g. AMIRA VR or ENSIGHT GOLD). In the academic field, key actors are NASA (http://www.nas.nasa.gov/About/Gridpoints/PDF/nasnews_V02_N02_1994.pdf), the Department of Mechanical Engineering of the John Hopkins University in Baltimore (http://www.wse.jhu.edu/oc/02_pdf/MECH.pdf), the Montréal University (http://www.polymtl.ca/rv/en/index.html), as well as several institutes linked to the GMD National Research Center for Information Technology in Germany.

From this quick overview emerges the fact that even if the "Virtual WindTunnel" has been envisaged for quite some time, actually interacting with simulation and visualization input parameters for the analysis of instationary 3D phenomena is a goal that is still to be reached. From the VR point of view, CFD is also an interesting application field for the development of multimodal interfaces, so as to present a multimodal perception of flows based on a combination of visual, audio and haptic feedbacks. CFD typically deal with vectorial, highly complex 3D data fields that could highly benefit from a distribution of rendering on the available human perception arrays, a trademark of modern VR systems.

8.2.1.4 Geosciences, meteorology, and oceanography

8.2.1.4.1 Computer Fluid Dynamics

A very innovative application of VR is the simulation of otherwise inaccessible physical phenomena. An example is "virtual relativity", consisting in the depiction of the real world, as seen by an observer moving at the speed of light. Recently, Weiskopf (2000) modelled the propagation of light in a relativistic setting to simulate, using an adaptation of the polygon rendering technique, the subjective view of an observer with a speed close to the critical limit. On simple polyhedral scenes, rendering can be achieved in real-time, leading to the first stage of a VR simulation of a relativistic universe. Effects that were predicted by theory (contraction of length, time dilation, space-time curvature, colour shift) are clearly observable in a VE, by arbitrarily setting the speed of light to a convenient value (for example, to 5 km/h!).

8.2.1.4.2 NOAA High Performance Computing and Communications

(National Oceanic Atmospheric Administration HPCC) Office: A numerous works on visualisation of oceanographic data. Examples include:

- El Niño and La Niña Tao Array Demo: 3D animations of El Niño and La Niña temperatures from the TAO network of moored buoys in the Tropical Pacific Ocean;
- An animation of fish recruitment and dispersal in Shelikof Strait, and a world of temperature and dynamic height in the Bering Sea;
- PMEL visualizations of global gridded dataset from the Ferret archives;
- Hydrothermal Vents Demo: 3D visualizations and animations of hydrothermal vent plume formation and development from the VENTS program;
- Temperature contours from CTD casts in the Bering Sea from the ICE-90 program;
- Global Carbon Cycle Project: Surface CO2 measurements from the tropical Pacific by the GCCP group, showing how CO2 is related to the thermocline.

(http://www.pmel.noaa.gov/vrml/3DViz.html)

8.2.1.4.3 Marine Virtual Explorer

Marine Virtual Explorer (MARVE) at Stanford University is an interactive 3D simulation that provides the opportunity to explore a hydrothermal vent field on the East Pacific Rise. Through the creation of an authentic environment, MARVE (http://ldt.stanford.edu/~lmalcolm/marvesite/) allows students to play the role of an oceanographer on a mission to map and sample the underwater world.

8.2.1.4.4 USGS Tsunami Research

Animations and virtual reality models of tsunami waves (http://walrus.wr.usgs.gov/tsunami/): One may find general information on how local tsunamis are generated by earthquakes as well as animations, virtual reality models of tsunamis and summaries of past research studies.

8.2.1.4.5 Scientific Visualization Studio

SVS of the NASA Computational & Information Sciences and Technology Office (CISTO): The mission of the Scientific Visualization Studio is to facilitate scientific inquiry and outreach within NASA programs through visualization. Projects regarding 3D visualization of scientific data include:

- The NASA Seasonal-to-Interann`ual Prediction Project: The SVS is developing methods for visualising the multi-variable three-dimensional data that results from the NSIPP coupled land-ocean-atmosphere model.
- The Tropical Rainfall Measuring Mission: The SVS produces precipitation maps and three-dimensional storm flybys from TRMM data.

8.2.1.5 Chemical Engineering

8.2.1.5.1 Virtual Reality in Chemical Engineering Laboratory (VRiCHel)

The three most significant projects of the VRiChEL lab are Vicher 1, Vicher 2, and Safety (http://www.vrupl.evl.uic.edu/vrichel/). Vicher 1 is a virtual simulation of a modern chemical plant, focusing on catalyst decay and different methods of handling this problem on an industrial scale. Students start out in the welcome centre, and then move on to explore more significant areas. Vicher 2 is another interactive virtual chemical plant simulation, focusing on non-isothermal effects in chemical kinetics and reactor design. Safety is another virtual chemical plant that differs from the Vicher simulations in two important aspects. It is a non-functional world, in that there are no operational control panels, and it contains a much higher level of realistic detail, since it is based on photographic data taken at an actual chemical production facility. There is an extensive help facility in both the Vicher applications and Safety. In the later case, the help window includes photographic details that would not otherwise be possible in a Virtual Reality simulation.

8.2.1.5.2 3D crystal structures with VRML

3D crystal structures(http://www.ill.fr/dif/3D_crystals.html) in VRML (Virtual Reality Modelling Language) may be generate0 -2.52(er)-8.8po m avbai(t)-44(y)18.2(o -2.52(( )-20(s)7.4(t)-29(m)18.2d(e)
L-i
o mtin

on the documents concept profiles. One may inspect individual documents and clusters and also suppress viewing of selected documents and clusters so as to concentrate on items of greater interest.

8.2.1.6.3 VR-VIBE

VR-VIBE (http://www.crg.cs.nott.ac.uk/research/technologies/visualisation/vrvibe/) is intended to support the co-operative browsing and filtering of large document stores. The essence of VR-VIBE is that multiple users can explore the results of applying several simultaneous queries to a corpus of documents. By arranging the queries into a spatial framework, the system shows the relative attraction of each document to each query by its spatial position and also shows the absolute relevance of each document to all of the queries. Users may then navigate the space, select individual documents, control the display according to a dynamic relevance threshold and dynamically drag the queries to new positions to see the effect on the document space.

8.2.1.6.4 Periscope system

The Periscope system (http://periscope.kti.ae.poznan.pl/) is an intermediary system in between a user searching documents on the Web and indexing search engines. Its aims are to provide a user with 3D visualisation of search results. The visualisation environment is automatically selected according to a novel method called AVE. In the Periscope system a user submits queries, which may be expressed by the use of textual data (e.g. keywords) and/or by interaction with visual objects composing a virtual scene that represents a web search result. Queries are translated by the Periscope system to conform the search engine query language and sent to a cooperating and indexing search engine. In response, the search engine sends back the search result to the Periscope system, which visualises it as a 3D virtual scene. The visualisation engine of the Periscope system uses a set of interfaces to automatically generate virtual scenes and to create necessary user interface elements.

8.2.1.7 Physics

The study of the irradiation-induced defect formation in various metals and metal alloys is of interest in order to understand the degradation of the physical properties of the materials used in pressure vessels in nuclear power plants as well as in metal coatings for fusion-based alternative energy sources (Nordlund et al., 1998). The theoretical description of radiation effects in materials requires modelling and simulation of processes that occur over widely disparate length and time scales (Averback et al., 1998; Tarus et al., 2003). Since most damage produced in materials during ion irradiation derives from a complex process occurring in collision cascades, much research has been devoted to studying these events (Nordlund et al. 1999).

The positions and kinetic energies of each particle of the system were displayed using the open-source software RasMol, which allows 3D visualization from various angles. It has the advantage that it can easily display large systems, the colour coding (or the greyscale) indicating qualitatively, the kinetic energy of each particle. However it does not allow the representation of different transparencies of various particles to better visualize particular parts of the systems (for instance the hot regions with high energy particles displaced from their equilibrium positions).

To obtain a more flexible visualization software than RasMol, a VR-based software has been created: ScientView, based on the AReVi API developed by CERV (Reignier et al., 1998). AReVi is a C++ and OpenGL based source, and is adaptive to very different configurations: from desktop systems to 3D stereoscopic immersion systems. ScientView allows the immersion within the simulated virtual environment leading to an interactive 3D visualisation of the experiment. Like RasMol, it allows various visualization perspectives, from different angles; but it also permits the navigation through the simulated environment; for instance following the impact particle (or any other particle of the system). Another special feature is the capability to modify the level of transparency of various particles which gives a clearer picture of the shock wave and the molten regions. Moreover, other options are related to sequences of images, the user being able to switch between no animation, step by step and continuous interpolation-based animation modes. The software also allows for reverse display of the time evolution as well as for choosing and visualising particles in any section plane parallel to the walls of the simulating cell.

The virtual environment has been considered as a space of human experience, and Popovici D.M. (2004) proposes a reactive agent-based model that permits the user's setting in the situation, the perception of space by its user, as well as the user's evolution in this space. In other words, everything inside the virtual space is an agent, able to perceive, decide, and react based on its profile, internal structure, and tasks, to the environment evolution, so to the user actions also; as section plane movement and transparency filter selection. The material cells' behaviour consists in following the given path and evolving according to the given temperature and position information set.

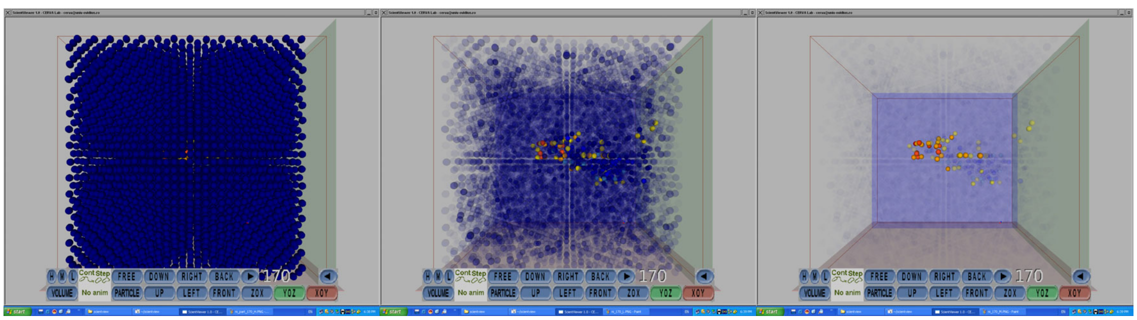


Figure 39 - ScientView's Screenshot

a. b. c.

On Figure 39, one can see the lateral views of the shock wave following a typical collision cascade in a 13500 atom Al target irradiated with a 500 eV particle, displayed using ScientView.

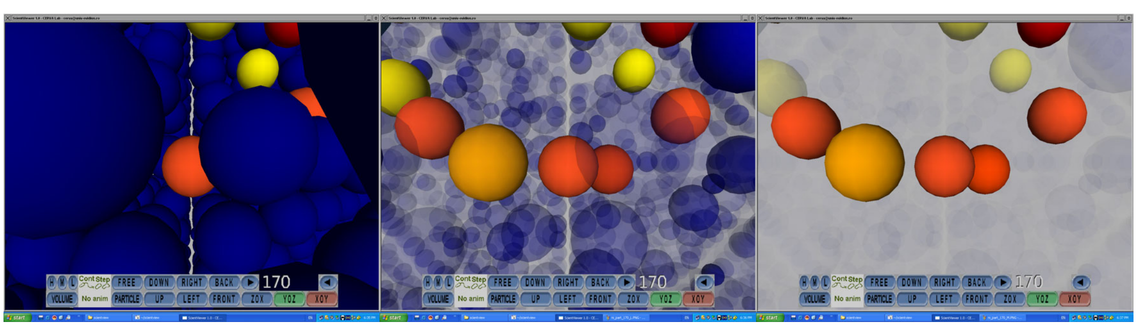


Figure 40 - ScientView's Screenshot

d. e. f.

On Figure 40, one can see impact particle's corresponding views in the same configuration, i.e. a typical collision cascade in a 13500 atom Al target irradiated with a 500 eV particle, displayed using ScientView.

The transparency of the Al atoms was set to:

- a.,d.: Low (transparency equal to 0 for all particles, like in RasMol)
- b.,e.: Medium (transparency 0 for particles with Ek < 0.2 eV and 1-Ek/200 for the rest)
- c.,d.: High (transparency 0 for particles with Ek < 0.16 eV and equal to 1-Ek/2000 for the rest).

8.2.2 Maintenance Systems

Virtual Reality is a powerful tool with a high potential for training in the maintenance area. Its use on simple platforms with low hardware and software requirements make it possible to develop applications with flexibility and reconfiguration capabilities. There is a great variety of VR applications for maintenance systems. Amongst others, the following can be cited.

8.2.2.1.1 Virtual Manuals

VR can be used for the development of virtual maintenance manuals, virtual mock-ups, illustrated parts catalogues, etc. These consist in 3D modelling of virtual animated and interactive images representing the general layout of equipment, rooms, and buildings, as well as assembling, disassembling and movement of specific equipments. Electronic technical manuals use 3D simulation as an illustration of maintenance procedures. To play a particular step of the procedure, there are controls for procedure playback (Play, Pause, Stop, Next, and Previous) and zoom sliders for changing the zoom in 3D window. Navigation commands allow the view of an object from various sides.

An electronic documents search system, based on a 3D model of the equipment, and a 3D Navigator permits to locate documents related to various parts of the equipment (maintenance zones).The 3D Navigator displays an image of an equipment (in 3D window) and the list of Maintenance Zones, which shows their Numbers and descriptions. By placing the mouse pointer over various parts of the equipment in 3D window, it is possible to identify the number for the corresponding zone. In addition, this will highlight the appropriate item in the list of Maintenance Zones.

Electronic part catalogues contain built-in 3D models of assemblies. These catalogues display in 3D windows an image of a given part, which is chosen in the list of the parts. In addition, the name of a part, over which the mouse pointer is positioned in the 3D window, will be highlighted in the list of the parts. Removal procedures, used to separate the desired assembly from other parts and assemblies, will start automatically after the catalogue is launched. Sequences and actions are presented to the trainee in terms of concrete maintenance procedures or tasks to be performed. The animation of the virtual images supported by photographs, videos or texts messages allows the development of high added value didactic applications easy to update.

These types of applications can be developed, to achieve different performance modes as:

- Automatic mode where the system shows the trainee the sequence and the steps of the procedure to carry out, with explanations and help functions on the specific maintenance tasks.

- Training mode where the trainee is trained in the execution of a task by carrying out the different procedure steps. The trainee can navigate and interact with the scenario. The system warns on the errors committed by the trainee. Help functions to guide him in the operation to execute as well as buttons to consult complementary information (in any format: text, voice, graph, video, etc) can be additionally provided.
- Evaluation mode where the trainee interacts with the Virtual Environment in the same way as in the training mode, but without any help. Trainee actions are either monitored by an intelligent tutoring system or recorded for further debriefing to evaluate the achieved skills.

8.2.2.1.2 Maintenance Systems based on Augmented Reality

Augmented Reality is another important research field for application of VR to Maintenance Systems. Use of AR is basically oriented to computer guided maintenance of complex mechanical elements with two complementary functionalities:

- user assistance for achieving assembly/de-assembly and maintenance procedures,
- training to assembly/de-assembly and maintenance procedures.

AR provides more flexibility in working methods while preserving user mobility in a context where the access to conventional documentation is cumbersome. It improves the user friendliness of the working environments. In this approach, reality is augmented with additional information. The trainee or maintenance technician sees the real objects and interacts with them, but at the same time, the system provides him with helpful additional information (virtual) which enlarges (augments) the reality easing the execution or learning of a determined task.

A typical AR system comprises a micro-camera, special see-through goggles, positioning and pointing devices, microphones and earphones. The goggles allow seeing the real environment while working as a computer screen where help and guidance in terms of documents, texts, graphics, audio or video files are dumped as virtual images to carry out the task.

The “augmentations”, the help and guidance being provided to the trainee, can be classified as:

- Text messages
- Images/Graphics
- Videos
- 3D models of equipment pieces.
- Audio: Codified sounds, instructions or explanations.
- etc

The visual augmentations are registered in the see-through goggles in the following ways:

- **Equipment referenced**. The image is shown as any other part of the equipment in a fixed location and it does not track the movement of the trainee.
- **Trainee referenced**. The image is displaced according to the movement of the user head. The images follow up his glance.
- **World referenced**. These images behave as any other object of the environment where the trainee is.

The user can access to full documentation and manuals directly registered to his working environment. Visual and audio augmentations are used to guide the user through the right procedure to apply. The system is controlled through speech, positioning and a pointing device system. Integration of speech interpretation and 3D user position tracking techniques are coupled to visual augmentation elements being registered, in real time, to the scene observed in see-through goggles.

The positioning system allows to know the coordinates of each part of the equipment and the trainee position and to represent virtual images in the real scenario pointing out the different parts and helping the user in the task execution.

A 3D navigation menu is shown in the user goggles letting him know the steps or the actions of the process he is following. The interactions between the system and the user are made by speech recognition to allow hands-off.

8.2.3 CAD/CAM and Engineering Simulation

A priori, it is not difficult to accept that VR can provide obvious advantages along the whole life cycle of a product in those cases where a shared perception of a complex scenario (mainly visual scenarios) is a key factor of success. VR can be used for the design, construction and maintenance of complex products.

During the design phase, it is clear that VR is a tool that provides an ideal representation of the product to be developed that can be explored and modified in a very intuitive way so that the interaction between the customer and the developer (or among the developers) can be fully supported. This has a positive impact on the development/construction phase as the number of the late modifications is considerably lower, reducing costs and shortening times. For the same reasons, many manufacturing/development/assembling problems can be anticipated during the design phase, reducing cost, time and improving the quality of the product.

Visualisation of critical scenarios and paths during the specification or design phases provides therefore a unique way to reduce development time and costs. Virtual Reality can be applied in engineering designs and in the modification of existing ones allowing:

- Design optimisation from performance point of view.
- Implementation of concurrent engineering philosophy to assess the design from different perspectives involving multi-functional teams
- Presentation to the client with improved customer satisfaction
- Rehearsal of construction tasks
- Operation and maintenance training

VR based user interfaces can also be used to simulate and visualise the movements of robotic devices, for instance those used for the in-service (or pre-service) inspection of the main components of a Power Plant. The user can see the real robot movements on line (while performing the inspection) and navigate in the related 3D environment. He can also run an off-line simulation to plan the intervention procedures.

VR can be applied as well to the design and validation of procedures (operating procedures, emergency procedures, etc) and to the design of Control Room Modifications. Interactive VR models of Power Plant Control Rooms are used to start making modifications in the CR (physical layout, panels, etc). Both the customer and the designer can test in the virtual model alternative solutions before freezing the design and manufacturing the components. Human-machine interface considerations can be taken into account more easily during the design phase and the reporting of the modifications can be made easily.

During the product lifecycle, Virtual and Augmented Reality can be successfully used for virtual prototyping, aesthetic evaluation, maintenance or formation (Gomes de Sà and Zachmann, 1999) (Biermann et al., 2002) (Lee et al., 2004). Nevertheless, in these cases, the objects are created with traditional CAD software and are then exported into standard graphic object formats, such as VRML or STL, in order to be used in VE.

However, some studies have been done to bring shape design in VE. Some results (Chu et al., 1997; Kiyokawa et al., 1997) focused on immersive interaction for a design task, since accurate interaction is one of the problems of VE for geometric modeling.

Most of the immersive approaches to geometric design are based on object representations that are favorable with a 3D interaction. They use models that allow object creation and modification with direct manipulation (Deisinger et al., 2000): voxels grid (Lu et al., 2002), function representation (Savchenko, 2000), freeform surfaces drawing (Usoh et al., 1996), simple polyhedral primitives (Butterworth et al., 1992), physical-based deformation (Llamas et al., 2003) or complex multi-resolution mesh (Gregory et al., 2000). However, all these applications can only be used for early design stage of product lifecycle, such as sketching. Only few attempts have been done in the field of part design (Zhong et al., 2002), and in these studies objects can't be modified in VEs.

Therefore, in the previous VE-CAD systems, the creation of parts with immersive interaction was possible. However, modification of these objects, still with an immersive interaction, was not feasible. A recent approach (Convard and Bourdot, 2004) consists in replacing editing of the history tree by direct interaction on the shape of the solid. These reactive objects in VE allow better use of the 3D interaction and can improve the overall system ergonomics during the design task. These techniques can be combined to a multimodal user interface in order to provide natural interaction (Convard and Bourdot, 2005).

8.2.4 CAD Virtual Workspace

8.2.4.1 Overview

Future Workspaces Research Centre at the University of Salford has initiated a research programme to explore an interface and workspace issues, with the view to contributing to novel interfaces for future CAD environments. The main aim of this research is to develop an novel interactive space (physical and virtual) and an intuitive modelling framework, closer to the designer's thinking, so that user actions can be expressed as higher-level languages and the meaning of that expression can be interpreted directly into a virtual form (Norman 1986).

8.2.4.2 Design Workspace Environment

Figure 41 below shows the proposed CAD workspace. Several touch-enabled displays with a pen device allow the user to work on the screen itself providing better hand-eye coordination. At present, the tablet display is used for drawing various 2D model profiles and the tablet PC is used for selecting sculpting tools and changing model attributes such as colours, line width, transparency and stereo effect. A TRACE system, from BARCO, is used for 3D stereo visualisation of the 3D Models. A Vicon optical tracking system, attached on the 3D display, is used to track the physical tool to provide direct sculpting facilities on the stereoscopic objects. This sculpting interface provides the user with a library of virtual sculpting tools to perform direct sculpting operations on the 3D model.

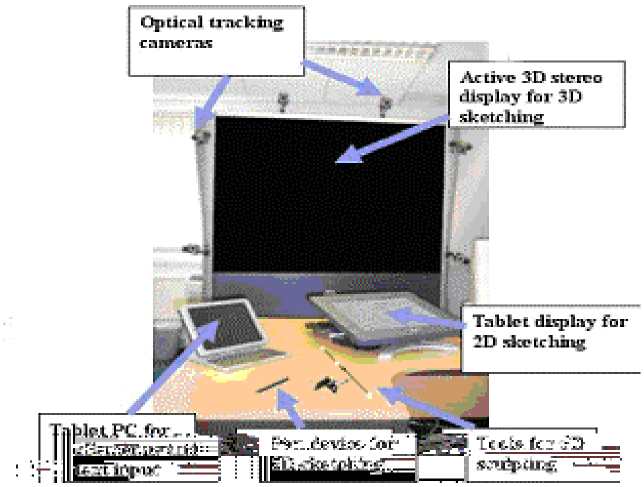


Figure 41 - CAD design workspace

The diagram in Figure 42 demonstrates various graphical contents for different display devices. The 2D space showing the 2D model profile is visualised on the tablet display, and the 3D space showing the 3D stereo model is displayed on the TRACE system. The control panel for application settings is displayed on the tablet PC. These three different display devices are linked together as one single station and operated by a single designer.

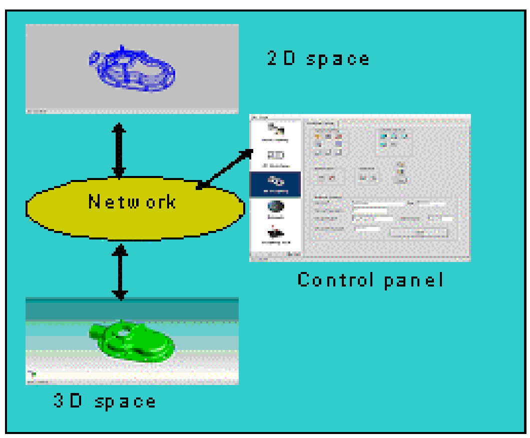


Figure 42 - Graphical contents in different display devices

The characteristics of this working environment are that the multiple user interfaces are seamlessly glued together to enable high-quality and productive person-centric workspace and the design is evolved through the use of the intuitive/familiar tools and multi-displays. More importantly, with the properties of flexible, scalable and adaptable the workspace can be easily configured from individualisation to interoperability. This improves designer abilities to work collaboratively, therefore increasing creativity which, in turn, will boost innovation and productivity as well as support new value creation form.

8.2.4.3 User Interfaces in Design Workspace

The user interface is the most critical concern in the design process. This research adopts two high-level metaphors (sketching and sculpting) as the prominent user interfaces to reduce the distance of semantic and support designers' creativity.

8.2.4.3.1 *Sketching task*

In sketching interaction, the designer performs the "pen and paper" sketching metaphor on a touch-flat display. This provides a precise hand-eye-coordination for faster navigation and more precise drawing. It allows designers to dynamically draw 2D technical shapes such as points, lines, arcs, circles and freehand splines, with automatic constraint mechanisms to define relationships with these geometric entities.

Figure 43 shows a sketching scenario to illustrate the process of freehand sketch. First the designer draws a freehand sketch. This sketch is then immediately converted into a b-spline curve with the continuity healing. The designer can continue drawing another object such as a circle and the tangent constraint between these two objects is generated automatically if this circle is close enough to the freehand spline curve. Finally, this 2D profile can be dynamic manipulated by dragging one of the objects while maintaining the constraint imposed.

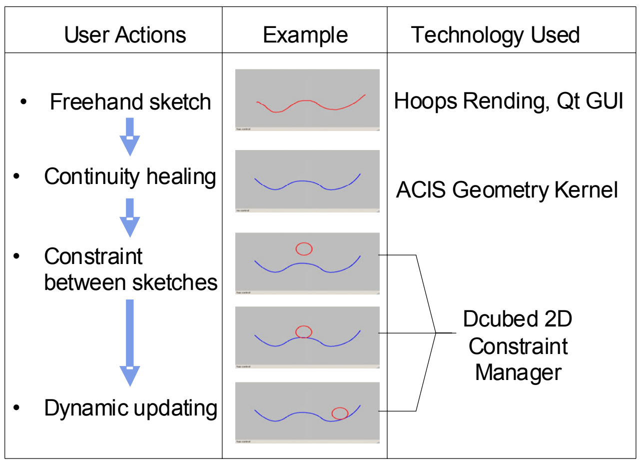


Figure 43 - The sketching interaction processes

8.2.4.3.2 *Sculpting task*

The sculpting interaction provides a set of 3D virtual tools to perform sculpting tasks in a predictable way. This challenge is achieved by incorporating the following characteristics:

- Non-intrusive direct manipulation.
- Deformable modelling.
- Sculpting metaphors - Virtual sculpting tools

The virtual tools explored in this research can be categorised into three types:

- The Cutting Tools (Figure 44 a): They perform operations such as slicing, cutting and trimming on the model (e.g. knifes);
- The Force Applying Tools (Figure 44 b): They provide forces that pull or push the target model once touched (e.g. hammers, attractors);
- The Surface Manipulation Tools (Figure 44 c): They provide a set of deformable constraints to perform deformation on a given surface. This transforms the model into the clay-like material for the user to manipulate the model directly.

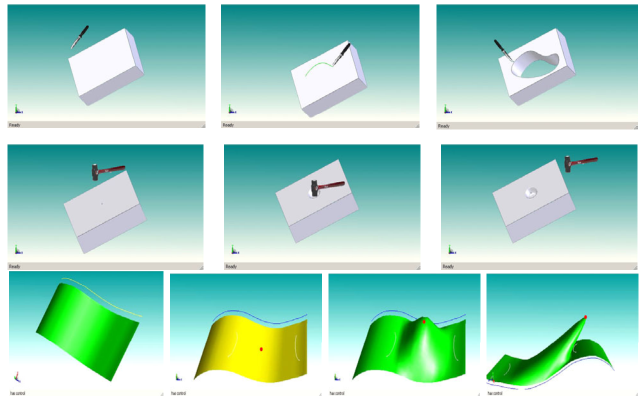


Figure 44 - (a) top: The cutting tool; (b) middle: The force applying tool; (c) bottom: The surface manipulation tool

8.2.4.3.3 The communication between the sketching and sculpting tasks

The communication between the sketching and the sculpting tasks needs to be processed in a synchronised way in order to give users the feeling of real-time update. Making the distribution of the design model transparent to the user was one of the difficulties in the implementation stage. This process is described as followed: in the sketching activity, once the satisfactory 2D profile has been conducted, it can be converted into a 3D solid model through modelling operations such as spinning, sweeping, lofting and extrusion. Figure 45 shows that the 2D curve profile generated in 2D environment is swept into a surface in the 3D environment, and the 2D environment in turn gets updated with the surface profile automatically.

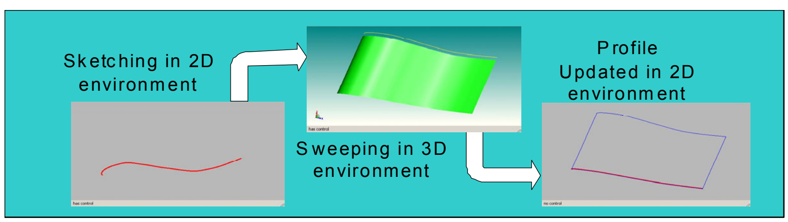


Figure 45 - The sketching interaction processes

During the modelling operation, the environment (2D sketching or 3D sculpting) used by the designer is called the active environment, while the other environment is considered to be the passive environment. The passive environment gets updated according to the user actions in the active environment. Figure 46 demonstrates the use of the surface manipulation tools in two different environments. Here, initially a surface is presented in both environments. The designer firstly focuses on the 2D environment as the active space to define the point/curve deformable constraints. The passive environment (3D space) gets updated by showing these constraints. The designer finally switches his/her attention to the 3D sculpting environment (active) and manipulates the tools to create a complex shape, making the 2D sketching environment to be the passive environment. Any update made on the 3D sculpting environment is reflected on 2D sketching environment in real-time.

This research is exploring the use of 2D sketching and 3D sculpting paradigms to implement a highly interactive design paradigm. This direct and familiar user interface can reduce the burdens of technological devices and therefore the designers can place the efforts on their creative tasks. In particular, the technology evolves continuously, but if the system is closer to what users expect then it could be easier to learn and use.

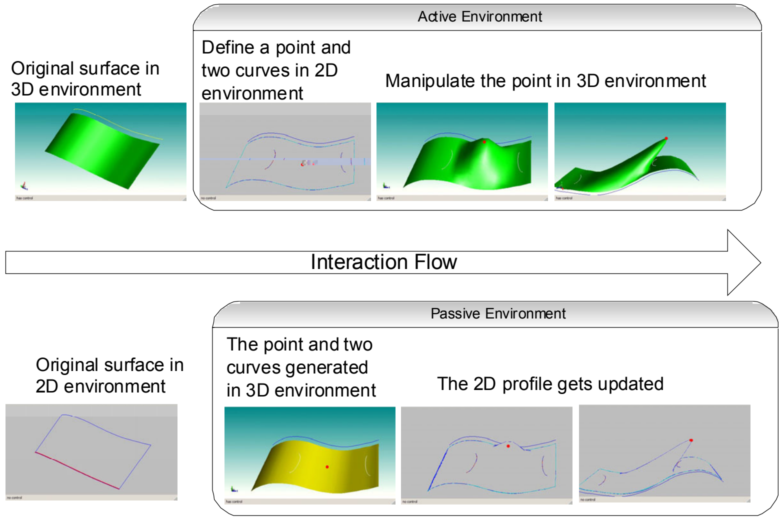


Figure 46 - The sculpting interaction process

8.2.5 Immersive Modelling

The immersive environment enables a user to visualise three-dimensional spaces more easily than a two-dimensional interface. Until recently more and more research has adopted immersive/semi-immersive environments for CAD modeling applications. Three examples are summarized below.

SketchAR (Fiorentino Michele et al., 2002; Figure 47; Figure 48) is one of the first systems to support three-dimensional modelling and the integration of a modelling kernel that distinguishes SketchAR from traditional VR systems. SketchAR provides the user with a semi-immersive environment. Using his two hands and hand movements directly in 3D space he/she can create free form models in real-time. The principal approach of the system to model curves and surfaces is by creating 3D strokes in free space using a 3D input device. This way the user can create different types of curves and surfaces, e.g. coons patches, skin surfaces and net surfaces.

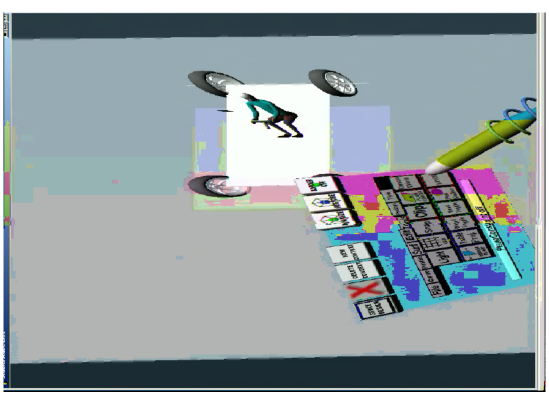


Figure 47 - SketchAR user interface

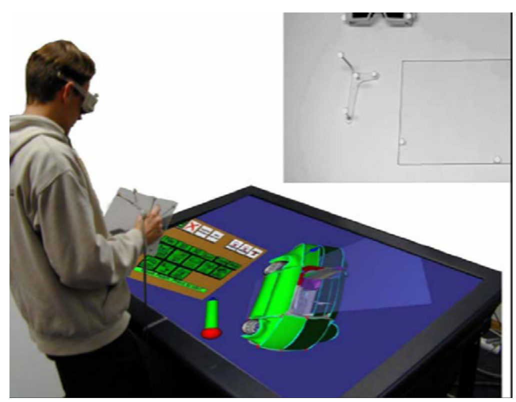


Figure 48 - Direct 3D interaction and visualization in SketchAR

The DesignersWorkbench (Kuester et al., 2000) is developed for a semi-immersive virtual environment with two-handed modelling, sculpting and analysis tasks (Figure 49; Figure 50). This project emphasises two key components from the classical product design cycle: freeform modelling and analysis. In the freeform modelling stage, content creation by two-handed sculpting of arbitrary objects is emphasised. It uses polygonal, volumetric or mathematically defined primitives. The analysis component provides the tools required for pre- and post-processing steps for finite element analysis applied to the created models.

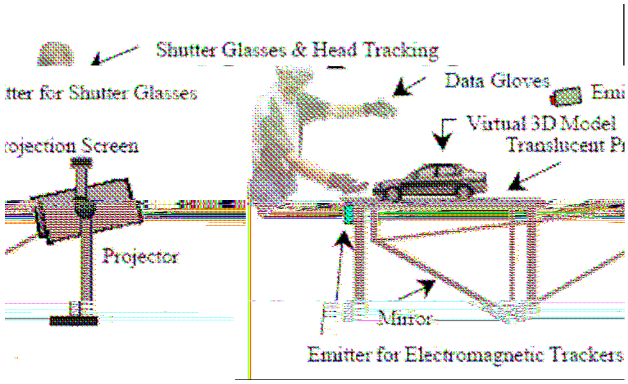


Figure 49 - DesignersWorkbench hardware setup

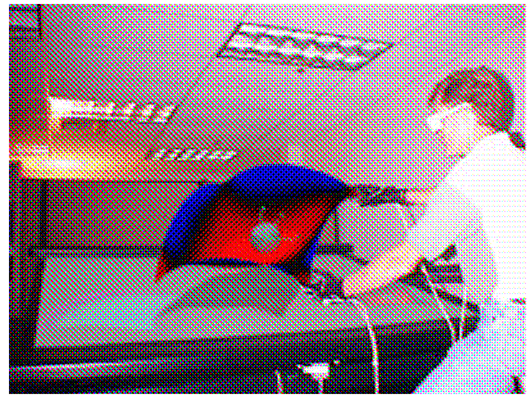


Figure 50 - DesignersWorkbench manipulation

Another example using gesture-based interaction to generate 3D sketches in a semi-environment workbench is presented by Diehl et al. (2004; Figure 51). The goal of this system is to support the designer from the early stages of sketching up to the modelling of exact product representations. This is achieved by a concretisation assistant that generates vector-based 3D-geometries out of voxel-based 3D-sketches.

Figure 51 - Gesture-based modelling in immersive environment

8.2.6 Emergency Planning

Virtual reality software offer the chance to simulate the most hazardous, even life-threatening situations realistically, so that emergency services are fully trained and better equipped to save lives or prevent serious injury.

Emergency management is characterised by involving a broad set of teams working to resolve the emergency situation in a collaborative way. Different actors and communication channels are involved in one of the most complex accident scenarios. Once the emergency is detected, they interact dynamically, with high levels of interdependence, receiving information from multiple sources and under pressure of time. In order to handle and be prepared for such a situation, prior planning is essential. An important part of an emergency plan is devoted to training. Training is mainly accomplished by the use of emergency drills based on scenarios that simulate different accidents.

Virtual Reality environments can simulate the behaviour of the agents directly involved in the specific training. It can support:

- Preparation of emergency scenarios
- Decisions that have to be taken for all the organisations that may be linked to each scenario.
- Simulation of decision implementation
- Evaluation of the decision taken
- Evaluation of the consequences of the decision
- Feedback to the user on the decision taken
- Learning lessons concerning the whole process

Emergency response training is expensive for a number of reasons: The number of people involved in an exercise can be high, the disruption caused may be expensive and the equipment involved may be costly to operate. It may also be infrequent for similar reasons, which may mean that opportunities for learning are sparse. Virtual Reality technology can solve these problems and be used for training on emergency management in a collaborative way by means of intelligent agents techniques and virtual world modelling allowing:

- Intelligent agents to stand in for some of the emergency response teams.
- The emergency response team to act in a virtual world.
- Increase the frequency and quality of Learning opportunities for teams, individuals in teams and organisations

8.2.7 Training

8.2.7.1 Overview

The Virtual Reality technology is a powerful tool for training programmes. This type of technology is able to provide convincing virtual environments which are an important characteristic for simulating real-life tasks.

Training through VR technology is recommended in the following situations

- Real-life operations too dangerous
- Real-life operations too expensive
- Real-life operations not possible
- Real-life operations with high difficulty to learn

The main benefits of using virtual reality over traditional training methods are listed below:

- Reduction of training costs
- Increase of safety
- Reduction of downtime

The training through Virtual Environments is based on the idea that the user learns and retains better the acquired knowledge with practice. Simulated scenarios help user learn faster and remember procedures better. These scenarios force users to put into practice what they have learnt and confronting them with the consequences of their decisions.

In a training process, the main goal is to acquire or improve knowledge and skills for the tasks performance due to provided information or experience. According to “power law of practice” (Farmer et al., 1999), the time required to perform a task is directly proportional to the performance time on the first trial and inversely proportional to the trial number. Clearly, it is observed that the process of task repetition will achieve better results and it is not a problem in a virtual environment because it can be repeated indefinitely.

In the context of training, Farmer et al. (1999) talk about the main skills related with a user and about the kind of instructional activities. They classify skills in this way:

- Perceptual-motor skills: It involves coordination between stimulus or perceptual inputs received from the environment and the motor responses that respects the received inputs.
- Procedural skills: It involves the execution of an algorithmic sequence of discrete actions for the performed task. An important issue is the determination of the kind of information incorporated within the mental model, how this information can be acquired most efficiently and how it affects transfer and retention.
- Cognitive skills: It is the synthesis and integration of different types of information which have been acquired through human senses and that is made understandable to the user.

The natural way of performing the tasks influences the perceptual-motor skills whereas intuition doesn't influence the procedural and cognitive skills.

Furthermore, instructional activities can be distinguished in terms of whether they occur prior to, during or after training sessions. These activities can be labelled briefing, tutoring and debriefing. The major distinction between tutoring and briefing/debriefing is that tutoring is a process in real-time whereas briefing/debriefing is a process that proceeds off-line. Furthermore the type of interactions (navigation/selection/manipulation) with the objects which compose the virtual training world can influence the training sessions.

Navigation is a basic mechanism of interaction that is required in many immersive virtual applications and based on the control of the user's viewpoint motion in the three-dimensional environment. In most cases, navigation is not an end unto itself. Rather it is simply used to move the user into a position where he can perform some other, more important tasks. Because of this, the navigation technique should be easy to use, easy to learn and cognitively simple (the mechanism of navigation should be transparent for the user).

Respecting selection process, it involves the choice of one or more virtual objects by the user for some purpose. Often, selection is performed to set up manipulation, which sets the position and/or orientation of a virtual object. Obviously, unless the user is constantly manipulating a single object, he must first select the object he wants to manipulate. Some virtual objects can be reached directly by the user (user's hand maps to the virtual hand) and for some others, the user must extend his virtual hand much further than his physical hand.

In order to perform a useful training, the interaction mechanisms must have the following characteristics:

- Transparency for the user.
- Ease of learning.
- Efficiency in the task performed (more accuracy and velocity).
- Comfort in simulations of large periods of time.

8.2.7.2 VR-based prototype for engineer training

8.2.7.2.1 Non Destructive Testing

Training engineer students to become experts in Non Destructive Testing (NDT), usually hire experts in specific line of business, like maintenance of engines, ships, aircrafts or maintenance of pipelines in oil or nuclear industry, need to invest funding for increase performance and quality in education for NDT courses. NDT methods and equipment require more IT as technology evolves. Nevertheless, education and training processes starts to use new technologies as VR and AR. This why, each NDT research and development centre should require IT experts and VR/AR-based training environments to install basic comprehensive knowledge in NDT.

EngView is an example of successful integration of VR-based NDT equipment in standard education at OVIDIUS University of Constanta. With EngView, new technical engineers with strong understanding of NDT are educated in more efficient way. NDT as a profession today still does not have the broad interception in general technical education; however some examples show that it is possible to integrate virtual NDT equipment in regular study program.

8.2.7.2.2 Overview

The Virtual and Augmented Reality Research lab (CERVA) together with NDT department of OVIDIUS University of Constanta are currently using methods that improve the learning/training activities and permit the maintaining of close collaboration with local companies and other research institutes. In this direction, a modular three components-based NDT 3D system is being developed: the real NDT equipment (Figure 52), its virtual replica, and the student's training software that integrates practical work with theory.

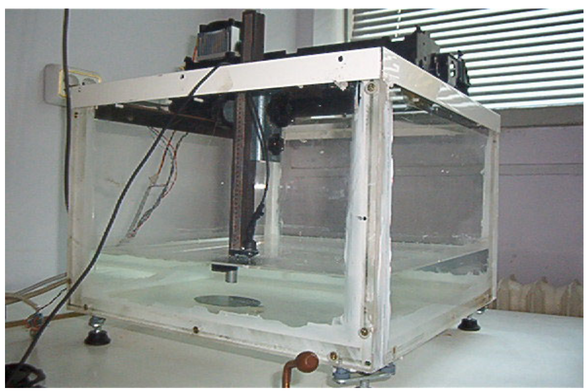


Figure 52 - EngView - real scanning setup

The main pedagogical objective is to assure students a rapid and successful integration. But the difficulty of this task rise due to different factors; such as: the different levels of knowledge that the students attain during their studies or the student's interest in the presented information and, with the same importance factor, the student's motivation to learn.

The learning speed varies from a person to another. Often, theory is easier to grasp than to translate into practice. Or vice-versa, practical skills are quickly achieved, even without any basic understanding of the theory. Despite this situation, NDT supposes both theoretical and practical skills to be well achieved.

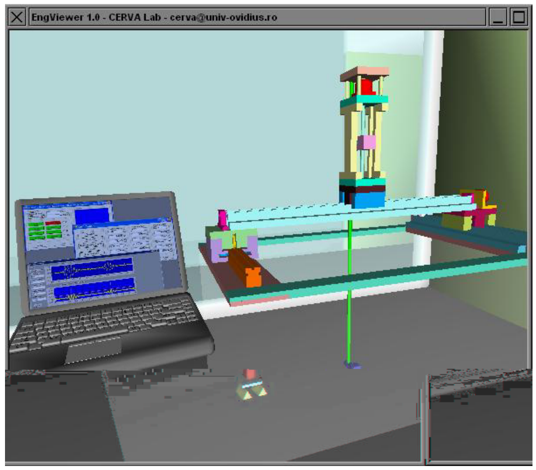


Figure 53 - virtual replica of the real configuration

8.2.7.2.3 Architecture

The used architecture in NDT training process supposes the existence of the real NDT scanning device as well as the control (trainer's) computer (Figure 52, Figure 53) completed with an experimental database (DB) used as tested experiments repository.

In the real environment, the trainer/user may setup the experiment parameters and control the real scanning device on the three axes through the SMMC interface (Figure 54). Based on three levels, SMMC module permits the control of the crane-like mechanical device based on step-by-step engines. The most visible layer represents a configurable user interface which detects the user commands and transmits them to the intermediary level. At the intermediary level, the user commands are coded according to the adopted system solution. Using a specific driver, the intermediary level is linked to the low-level layer. This level contains the specific hardware drivers (parallel and USB ports, and acquisition card SMC4 – Physical Acoustic Corporation). According to the scanning device movements and the experiment conditions, it obtains the ultrasound signal for analysis and characterization. The real experimental measurements were obtained by the immersion testing method, where the transducer is placed in the water, above the test object, and a beam of sound is projected (Zagan et al., 2003).

The tutor is working with real architecture system to collect the signals received by the transducer (echo waveforms) from different samples. They are then sent to an oscilloscope, where their amplitude and velocity are read directly from the sampler which performs the sampling of the signals. The current experiments' parameters, material characterization and analysis results obtained using the real setup are all stored for later use in (self) training sessions.

The real experimental setup is extended with virtual copies of the real scanning device, copies that implements the full functionalities of the real ones, together with correspondent trainee's computers. This way, the trainer actions within real environment may be broadcasted in real-time at all active 3D virtual copies of the scanning configuration, through EngView. This actually represents the basis in the educational and training process.

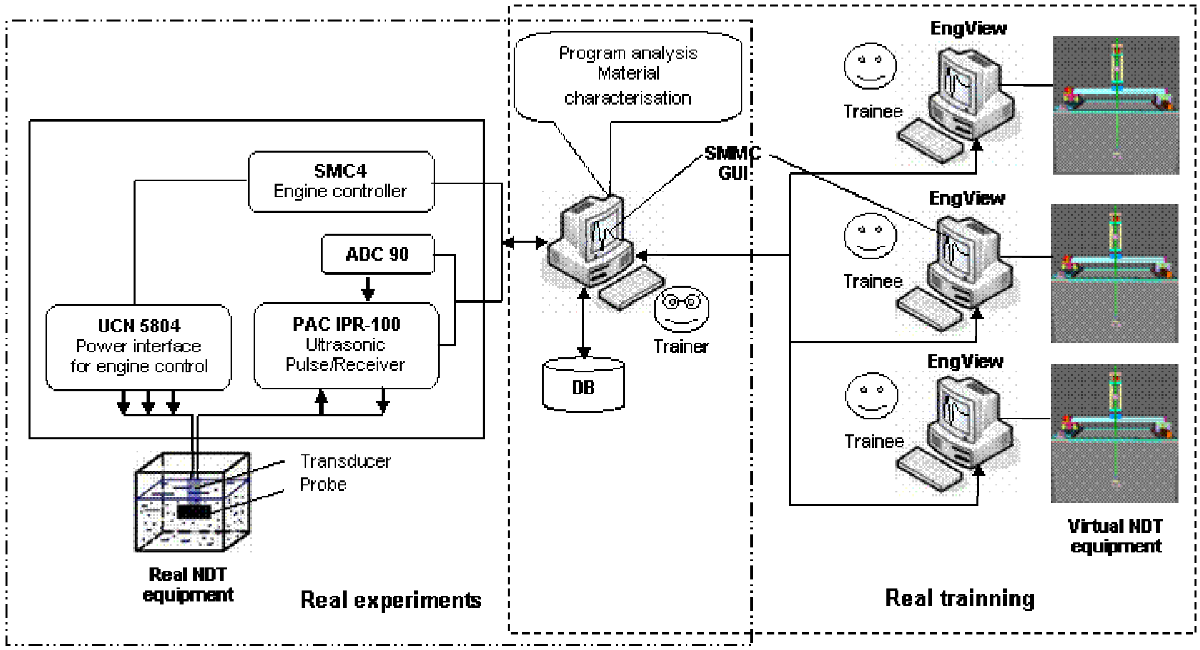


Figure 54 - EngView architecture

EngView proposes a friendly user interface which accomplishes the level of NDT equipment networking with latest IT/VR/AR technology, and enables subjects to achieve practical and theoretical skills needed by NDT to be integrated in standard program. As shown in Figure 54, EngView integrates SMMC module at each trainee working station. Using this interface, the students may control the movements of the virtual 3D scanning installation and may repeat the shown experiments. When the real configuration is available, they may test it in real conditions (but not concurrently).

Moreover, the trainee obtained experimental parameters are compared and analyzed at the trainer site in order to identify, characterize and publish the ultrasound signals. This way, the trainer is able to evaluate the trainee achievement of knowledge/skills level concerning individual NDT operation methodology.

We have considered the virtual environment as a space of human experience, as proposed by Popovici, D.M. (2004); a reactive agent-based model that permits the user's setting in the situation, the perception of space by its user, as well as the user's evolution in this space. In other words, everything inside the virtual space is an agent, able to perceive, decide, and react based on its profile, internal structure, and tasks, to the environment evolution, to the user actions also; as virtual scanning installation movement components.

8.2.7.2.4 Implementation

The EngView prototype is mainly based on the ARéVi API, developed by CERV (European Virtual Reality Centre, Brest, France), for the user immersion within the simulated virtual environment, as well as on C++ software components (as SMMC) that connects the virtual environment to the scanning machine control software (Reignier P. et al., 1998). ARéVi API has the advantage to be open-source, C++ and OpenGL based, and adaptive to very different configurations from desktop systems to 3D stereoscopic immersion systems.

8.2.8 GIS Systems

Geographic Information Systems (GIS) enable maintaining information about spatial phenomena and provide users with means to analyse them. Typical functions of GIS systems are: capturing, structuring, manipulation, analysis, and presentation of spatial data. Traditional GIS systems typically offer only

It enables 3D visualisation and analysis of surface data. Using *ArcGIS 3D Analyst*, it is possible to view a surface from multiple viewpoints, perform advanced surface analysis, determine what is visible from a chosen location on a surface, create a realistic perspective image that drapes raster and vector data over a surface, and record or perform three-dimensional navigation. The *ArcGlobe* application in *ArcGIS 3D Analyst* enables managing and visualisation, from a local or a global perspective, large sets of three-dimensional geographic data. *ArcGlobe* provides the capability to interact with geographic information presented as data layers on a three-dimensional globe model.

Traditionally, real-time 3D terrain simulations have been performed using specially designed and expensive hardware and software. Recently, with the advent of powerful, low-cost PC computer technology, procedures for generating real-time terrain simulations using inexpensive commercial-off-the-shelf hardware and software have been designed. Typically, for simulation purposes, the image data (e.g. aerial photography), elevation data, feature data and 3D models that were instanced in the GIS database are transformed into an integrated data structure called a run-time database. A run-time database is designed for optimal display performance in a real-time image generation environment. Several commercial products for transforming GIS databases into real-time databases and for viewing their contents are currently available.

- *MultiGen-Paradigm* (http://www.multigen.com/), Inc. markets an end-to-end, integrated solution for real-time 3D applications. It consists of 3D entity modelling, terrain and environment authoring tools with connection to GIS databases, players and standards.
- *TerraSim Inc.* (http://www.terrasim.com/) has developed and markets terrain and feature generation tools called *TerraTools* (http://www.terrasim.com/products.html). They enable production of accurate 3D models for urban, suburban, and natural environments. *TerraTools* supports integration of custom GIS source data, CAD models, and design and architectural models.
- *MetaVR Inc.* (http://www.metavr.com/) produces a low-cost PC visualisation tool: the *Virtual Reality Scene Generator* (*VRSG*) (http://www.metavr.com/products/vrsg/vrsgoverview.html). The *VRSG* application provides textured, real-time visualisation of dense multi-user virtual worlds. Advanced terrain and texture paging technology enables VRSG to depict realistic virtual worlds derived from imagery, feature and elevation databases.
- *Terrain Experts Inc. (TERREX)* (http://www.terrex.com/www/index.htm) produces a high-end real-time simulation database development tool: *TerraVista* which enables automatic creation of 3D terrain real-time databases for use with real-time visualisation systems. *Terrex* also provides the *TerraViz* – a high-performance 2D/3D visualisation tool. *TerraViz* supports 2D/3D measure functions, scenario creation and playback, and the ability to create AVI movies from either saved scenarios or interactive fly-through.
- GeoIndex (http://robotica.uv.es/grupos/artec/English/proyectgeoindex.htm) is an ongoing demonstration project that aims to conduct a feasibility study for the development of a PC-based system (Windows NT) for visualising and accessing information in existing databases that can be geo-referenced. The geographical information is composed of 3 fundamental layers: digital terrain models (DTM), over which satellite images (textures) are draped, and (optionally) vectorial cartographic information such as communication networks, rivers, lakes, etc., and 3D icons which represent points of interest (POI) like hotels, tourist areas, etc. The combination of these layers of geoinformation form what has been called a virtual landscape (Poveda and Perez, 1998; Perez et al., 2002).

  The system is optimised to allow the user to 'fly' freely through the virtual landscape. Users can fly to an area of interest and click on a 3D icon found on the virtual landscape to ask for information relating to it. This information is presented in multimedia formats using an ordinary web browser window.

To achieve the previously pointed out purposes the research has been conducted to the development of new data structures to dynamically represent 3D terrain data and procedures to database segmentation and compression to be able to sent it via internet (Perez et al., 2004).

8.2.8.2 AR Applications

8.2.8.2.1 UbiCom

The *UbiCom* (http://www.ubicom.tudelft.nl/) project developed a system that enables AR visualisation of GIS related information (e.g., position of underground cables, owner of a building). Within this project, GIS technology is used for three purposes:

1. calculating the exact positions and orientations of mobile units
2. identifying object occlusions
3. providing geographic information requested by users.

The exact pose estimation in the *UbiCom* system is based on computer-vision technique. GPS units included in the mobile equipment provide initial approximate unit positions. The exact positions are calculated by matching lines extracted from video streams (obtained from the video camera of the mobile unit) and lines retrieved from the 3D GIS. In addition, the rendering subsystem retrieves specific data about the positions and the shapes of physical objects in the field of view, i.e. those objects that can occlude the virtual objects. Finally, virtual objects presenting GIS data are superimposed on the video.

8.2.8.2.2 VRGIS

VRGIS Corporation (http://www.vrgis.com/) provides a real-time environment navigation and description system for visually impaired. The system is designed to give the assistance in physical environment through real-time, wearable technology. It delivers critical information at the 'point of need,' providing instant description and navigational data about a location. Such data includes key building information, entrances and exits, number of floors, stairs, and location of pertinent features like elevators, doors, restrooms, reception areas, and telephones. Through lightweight wearable computing technologies, environmental data intelligently finds and adapts itself to the user, instead of the user having to adapt and find obscurely located information.

8.2.8.3 Urban Planning

There have been many attempts to develop urban planning environments based on virtual environment technologies. This section summarises several key developments in this area.

Coors et al. (1999) present a collaborative urban planning system using a Responsive Workbench (Kruger and Frohlich, 1995) and a tangible interface. In this system, 3D-GIS is used to create urban models. A set of physical tools, each associated with certain functions, are tracked simultaneously facilitating group interaction with the planning and decision support system. However, using physical artefacts can only achieve simple functions such as information query because no more information, such as gesture and actions (e.g. confirmation) can be captured from this environment.

Ishii et al. (2002) use physical models and a luminous table with two projectors to create an augmented urban planning workbench. 2D drawings, 3D physical models and digital simulation are overlaid into a single information space to support the urban design process. This collaborative environment allows communication to take place between the stakeholders and the resulting design ideas to be visualised. But the workspace setting and the involvement of physical building modelling constrained that urban environment representation to being flat and be projected to a proper scale with physical building models which limits its ability for presenting a large urban area.

Hopkins et al. (1999) use a SMART board as a collaborative urban planning tool. It offers the user the ability to draw sketch-plans directly onto the virtual model. One of the key characteristics of this platform is the use of a touch-sensitive screen; this provides the user with the ability to draw a sketch-plan into computers. However this environment is limited to one user-view of the design result and 2D visualisation which does not give the user a comprehensive view of the scenario generated.

Although the methodologies and applications described above present valuable contributions to collaborative urban planning and design, the workspace setting limits the forms of representation of urban models, the scale of urban area and the form and function of interaction. Therefore there is a need to introduce a more comprehensive environment for supporting collaborative urban planning.

8.2.8.4 Standards

Several standards related to GIS, VR and the Internet were also developed. GeoVRML (http://www.geovrml.org/), an extension of VRML, has been proposed and implemented to provide geoscientists with a suite of enabling functions for the representation of large volume and high precision georeferenced data. Recent advances in GeoVRML include a number of new nodes that enable the transparent and accurate representation of geographical data, support for scalability and increased level of details.

8.2.8.4.1 OpenFlight

OpenFlight (http://www.multigen.com/products/standards/openflight/index.shtml), MultiGen-Paradigm's native real-time database file format, has become the de facto standard in the visual simulation industry. The standard describes logical and hierarchical scene structure that informs the real-time image generator what, when, and how to render, resulting in real-time 3D scenes with high precision and reliability. It is designed to support advanced real-time functions that include: levels of detail, culling volumes, switch nodes, drawing priority and binary separating planes (the hidden surface removal technique).

8.2.8.4.2 MetaFlight

MetaFlight (http://www.multigen.com/products/standards/metaflight/index.shtml) is a formal real-time database description specification. Contrasted with OpenFlight, which contains the geometry and its description within a single file, MetaFlight describes the structure, organisation, file naming and coordinate systems of all the datasets that comprise a more complex database. MetaFlight provides the bridge between real-time database generation tools and runtime applications. MetaFlight simplifies data integration and optimises real-time efficiency by communicating the metadata that enables runtime applications to take advantage of the database structure.

8.2.8.4.3 OpenGIS Consortium

OpenGIS Consortium (OGC) is an international voluntary consensus standards organisation. In the OGC, more than 280 commercial, governmental, non-profit and research organisations worldwide collaborate in an open consensus process encouraging development and implementation of standards for geospatial content and services, GIS data processing and exchange. A couple of standards have been recommended by OGC, known as OGC specifications, which are used for data model representation and interoperability. The most important specifications are Geography Markup Language (GML), Web Feature Service (WFS), Web Map Service (WMS), Web Coverage Service (WCS), Catalog Service Web (CS-W) and Simple Features – SQL (SFS).

8.2.8.4.4 OS MasterMap

OS MasterMap is an intelligent digital map designed by Ordnance Survey in the UK, based on OGC's GML 2, for use with geographical information systems and databases. By adopting GML, Ordnance Survey has given himself an opportunity to replace the NTF OS Land-Line standard, and give users much better base data. MasterMap currently contains four layers: topography, address, imagery and ITN (Integrated Transport Network) layer. Each layer carries millions of features, such as buildings, roads, phone boxes, post boxes, and landmarks, to present real world information digitally. The MasterMap is only supplied as GML files.

8.2.9 Educational Systems

The VR technology enables creation of virtual learning environments (VLE), where students can learn by interacting with virtual objects similarly to the ways they would interact with real objects. The VR systems can be used for teaching a variety of subjects such as history, culture, arts, geography, astronomy, chemistry, and physics. The potential of the application of VR systems in the education domain seems to be great since teaching material can be presented in meaningful and intuitive ways, and practical skills may be trained and observed from different perspectives. The experience gained by students during the learning process can be the basis for a group discussion in a classroom environment. Other attributes of VR systems that appeal to education are self-directed and natural learning, as well as increased motivation. Most people learn faster by doing, and VR systems provide a much higher level of interactivity than other computer-based systems.

8.2.9.1 Advantages of educational VR systems

There are many examples of where VR has been used for educational purposes but the level of success to a large extent depends on the flexibility offered by the Virtual Environment. Usually, students who use a VR based system enjoy themselves, and this factor contributes to the sense of their satisfaction from learning. The characteristics in favour of VR systems to be used in educational applications are the following:

- Flexibility
- Inherent safety
- Wide application
- Rich interactivity
- Intuitive interaction
- Motivation to learn
- Better learning results (doing is more effective than reading)
- Learning from mistakes
- Enhanced sense of presence
- Better understanding of processes

8.2.9.2 Measures of VR learning systems

The usefulness of a VR learning system for teaching in a particular case can be estimated by taking into account certain properties of the system. Important measures that should be considered in addition to the visual quality and complexity of the environment are the following:

- Autonomy – determines to what extent a VE is capable of performing its own actions independent of user intervention. An autonomous VE follows its own path to goals and may or may not change its course in response to user actions.
- Presence – describes how profound the experience of being in an actual place is. For presence to be high, the user must be allowed to interact with the VE both naturally and intuitively. When presence is high, the computer interface becomes imperceptible.
- Interaction – denotes the ability of the user to perform actions in the VE according to a logical rationale. The laws that govern the VE should become apparent over time, allowing for a meaningful interactive experience.

8.2.9.3 Constructivist approach

To improve the efficacy of learning using VR systems, users should be given the opportunity to become actively involved in constructing the knowledge through a coherent, direct interaction with knowledge domain representations. Direct interaction within a VE enables users to construct knowledge on the basis of interaction with their environment. This method of gaining knowledge is referred to as constructivism. The constructivist learning has a great advantage over other learning methods because the learners shape the learning experience by themselves. In consequence, the constructivist approach involves designing for learning rather than planning for teaching. The constructivist approach can be adapted to any subject area or curriculum by involving students as active participants immersed within VLEs. The students can directly experience and interact with the concepts, principles, rules, and procedures found in the domain, instead of being passive recipients of information given to them by a third-person instructor.

8.2.9.4 VR simulation for teaching

Through immersion in a VLE, students become a part of the phenomena that surround them. Consequently, the learning process becomes more efficient because the students can acquire knowledge through their direct experience of the environment. Virtual environments allow students to actually participate in a large number of experiences, to see representations of scenarios familiar or unfamiliar to them. They can experience interactive simulations of situations that they might not be able to encounter in the real world, for example due to lack of access or because these experiences are impossible in the real world. A VR simulation can be a representative of some real environment (e.g. a realistic model of real building interiors) or an abstract system (e.g. a theoretical 3D visualisation of molecules and their associated forces). The simulation comprises appropriate data for visualisation and optionally data for auditory and haptic representation of the system.

From an educational point of view, VR simulation seems to be particularly useful in the following situations:

- situations that would otherwise be dangerous (e.g. chemical or radioactivity experiments);
- situations where observation of internal structure is important to aid understanding (e.g., inner workings of machines);
- situations where interaction is important to aid understanding;
- situations that are too complex for conventional teaching methods;
- phenomena not normally visible to the naked eye:
    - macroscopic and microscopic (e.g., astronomical events and molecular movements)
    - very fast and very slow (e.g., explosions and continental drifts);

- demonstration of complex abstract concepts (e.g., magnetic fields) or as a support to explain concepts to people with health problems. For example, the INMER project aims at providing tools for persons with autism and persons with Down Syndrome (http://autismo.uv.es/; Herrera et al., 2004).

8.2.9.5 Distance learning

Advances in the ICT technologies have made it feasible to employ distance-learning systems in support of the growing demands for remote educational services, which can be accessible to people from their homes. Theoretical courses can be successfully provided by the use of standard computer-based technologies such as WWW with textual and multimedia contents. Nevertheless, a much more challenging task is to provide practical courses. This can be achieved by using autonomous Virtual Environments that provide users with high degrees of presence and interaction. One of the most essential benefits is the reduction of the costs associated with teaching. This is due to replacing real expensive resources such as buildings and teaching equipment with their virtual counterparts and reduction of required human resources. Important benefits are offered to the students including flexibility of time and place of learning, travel cost reduction, and accessibility for disabled people.

8.2.9.6 Collaboration

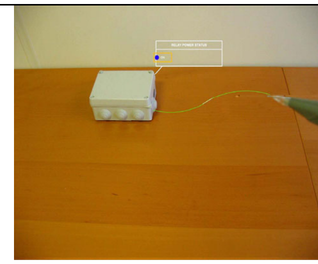
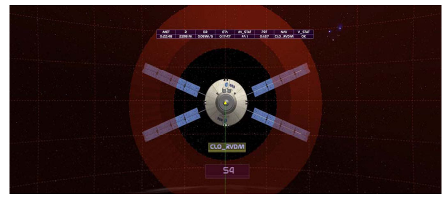
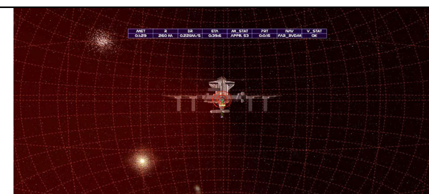
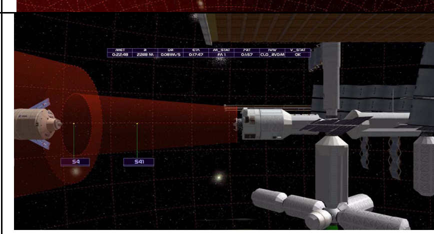
In the past, most educational applications of VR have involved a single student interacting with objects within a virtual environment (VE). More recent advances in VR technologies enabled the proliferation of collaborative virtual environments (CVE), in which more than one student can be placed simultaneously. Nowadays, an increasing interest in collaborative learning can be observed due to the fact that it offers important cognitive benefits. Collaboration is considered to be a critical component of the study since learning is often described in terms of being largely social in nature. In CVEs students and teachers can be remotely located but still share a virtual space, in which learning can occur. The students can learn not only via interactions with objects in the environment but also with other participants. It introduces an important social dimension to the experience, which helps the students learn how to clarify their ideas and interpretations through processes of articulation and discussion. Also, they can be taught how to solve possible conflicts engendered by collaboration and co-construct knowledge along with other participants.
Comparing face-to-face collaboration with computer-based conferencing, the former is better for creative problem exploration and idea generation, while the latter is more suitable for linking ideas, interpretation, and problem integration. Virtual environments with virtual face-to-face collaboration seem to be an ideal solution for distant collaboration because they combine the best of both face-to-face and computer-based collaboration.

8.2.10 Space applications

8.2.10.1 MMIT Mixed Reality System

The MMIT Mixed Reality System (MMI MRS) is an example of the use of VR in the space application domain. Currently, it has been used to explore *Augmented Reality On-board Procedure System* and *Immersive Rendez-vous.* This work is focusing on two types of users:

- Operations Engineers, to showcase new technologies to inspire new MMI concepts
- Astronauts and ground controllers, to try out and evaluate new MMI paradigms

The AR Onboard Procedure environment showcases Augmented Reality paradigms with a very tangible example. The system loads at run-time a procedure implementing an operational scenario. The system then guides the astronaut through the whole process leveraging sensor inputs representing position and orientation of the astronaut's head and arm to overlay relevant information in the natural field of view of the astronaut.	
The "rendez-vous" part illustrates how immersive VR can be used to dramatically increase the situational awareness during mission critical phases. This environment extends and builds on the concept of the god's eye view display, which is used for instance during ATV astronaut training by the instructor to monitor progress of training scenarios. Until now, god's eye view type systems were strictly used on the ground. The purpose of this environment is to prove the concept of using such systems onboard.	
The Keep Out Zone is represented as an orange animated dotted grid around the ISS. The picture below shows you how the Keep out Zone has been rendered in the VE.	
The nominal ATV approach cone is represented as a transparent red cone.	

8.2.10.2 Neuro-cognitive investigations in Space

Virtual Reality has been used in the context of human space flight to perform cognitive neurophysiology experiments in orbital space flight (e.g. onboard the Space Shuttle and the International Space Station (ISS)), in microgravity conditions. More experiments in this field are also foreseen for the near future.

VR allows for the study of the human brain in space and in particular of sensory-motor and cognitive functions in a realistic three-dimensional environment. The combination of electro-encephalograms (EEG) and Virtual Reality generator in the ISS may offer a unique opportunity to better understand cognitive brain functions in the presence of a virtual situation mimicking real environmental on Earth and in microgravity.

Other kinds of electrophysiology signals, such as EOG and EMG, can provide important information on the subjects' perception and processing of 3D space in micro-gravity conditions. This scientific approach may also provide new countermeasures that may be relevant not only for human space exploration but also in clinical treatments.

8.2.10.2.1 Previous In-Orbit Experiments

In this section we give a brief overview of some of the main experiments in the field of cognitive neurophysiology that have been conducted in orbit.

8.2.10.2.2 NeuroCog-I Experiment

The NeuroCog-I was part of the Odissea Project scientific program. The experiment aimed at understanding how humans perceive space, the role that the sensory information of sight, balance, motion and position play in this, and how their perception is affected by weightlessness.

In this experiment, each cosmonaut was tested on the ground before flight, during space flight aboard the ISS, and on the ground after their return to Earth. Prior to flight, cosmonauts were tested in 2 pairs of sessions over the 2 months preceding lift-off. Sessions within each pair were separated by at least one day. In flight, subjects were tested on two days over the course of their space flight, with at least one day between sessions. Subjects were tested on at least two days during the week immediately following the landing and two more times within one to three weeks later.

8.2.10.2.3 Stimulation with 3D virtual tunnel

Subjects looked straight ahead through a form-fitting facemask and a barrel frame to the laptop screen. It was viewed through a tunnel, thus removing any external visual references. In subsequent experiments, the navigation of the subject in a virtual tunnel generated by the computer was studied. On Earth, subjects performed the experiment while seated upright in front of the computer. During space flight, they performed the experiment in two conditions. In the attached condition, the cosmonauts used belts, foot straps and a tabletop to reproduce essentially the same-seated posture as that used on Earth. In the free floating condition, subjects held the experimental apparatus (laptop computer and tunnel) in their hands with an elastic band used to hold the mask against the face. An assisting cosmonaut then positioned the subject in the centre of the free working volume within one of the space station modules. The subject was then released and both subject and apparatus floated free from any contact with the station. The assisting cosmonaut ensured that no contact with the walls of the station occurred. To accomplish this, the assistant applied short tugs on the clothing of the subject to adjust the position without giving strong directional cues.

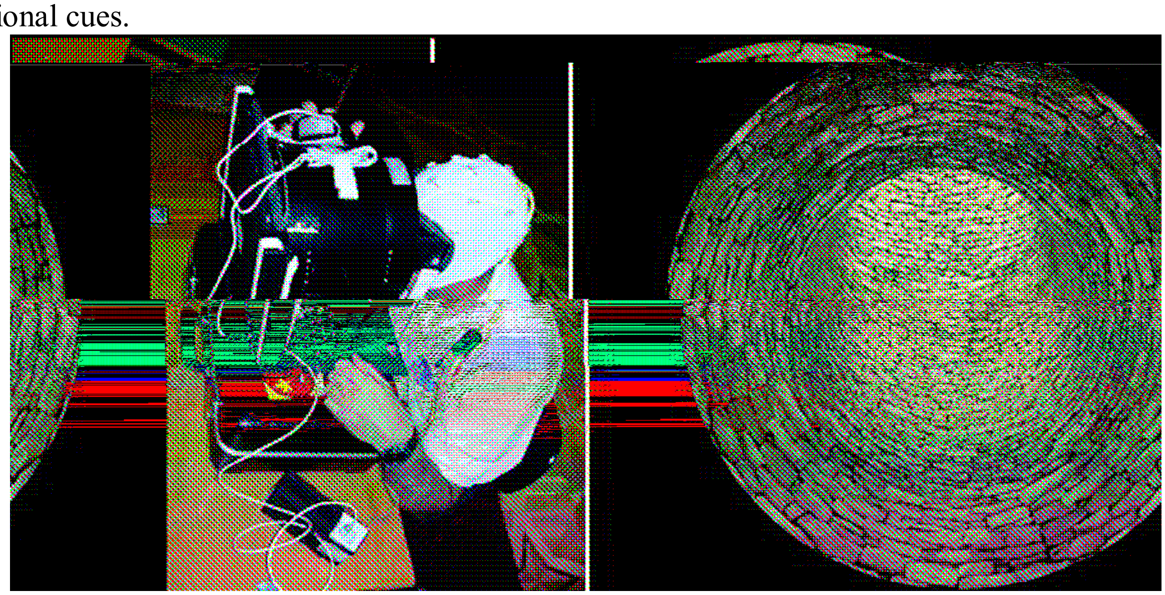


Figure 55 - The NeuroCog equipment (laptop + mask) and the 3D tunnel displayed on the screen

8.2.10.2.4 EEG/EOG recordings

The electroencephalogram (EEG) of the cosmonauts was measured using an electro cap adapted for the HALLEY recorder on board of the ISS.
Blinks and horizontal eye movements were monitored with electrodes at the outer canthi of the eyes (horizontal EOG) and above and below the right eye (vertical EOG).

8.2.10.2.5 COGNI Experiment

The COGNI experiment focused on the investigation of neurological functions related to the perception of motion and orientation. It was performed in orbit in October 2001, as part of the CNES French mission called ANDROMEDA, and had the following two specific objectives:

1. Understand how the human brain represents motion
2. Verify whether the brain gives priority to the gravitational axis in perceiving visual scenes

For the first objective, the experiment was designed to understand how the brain represents the three dimensions of space, in particular whether it treats the horizontal level in a different way with respect to the vertical dimension. The experimental setup in this case consisted of a laptop computer displaying a sequence of virtual tunnels in 3D. The astronaut was required to look at the display through a mask and a circular tube eliminating any visual cue of the surrounding real environment, and then to run along the sequence of tunnels in the virtual world.

For the second objective, the same equipment was used, but in this case the visual stimuli simply consisted of an oriented line, displayed on the screen and disappearing after a while, followed by a second line that the astronaut had to try to re-orient in the same direction as the first one.

9 Conclusion

This document has managed to define the main components of a VR system and summarise their state-of-the-art as of today. Current document provides a reasonable exposure to various advances made by the partners of this consortium and other international players.

However, due to the large scope of the field of VR, this work needs to be continued over the life time of the project to generate a more complete report on state-of-the-art in VR. Therefore, this document will be considered as a "living document" to capture current and future advances in the area of virtual reality.

10 References

Abe, K., Asano, F., Suzuki, Y., and Sone, T. (1995). *Sound Pressure Control at Multiple Points for Sound Reproduction*. Proc. Int. Congress on Acoust., Trondheim, Norway.

Afonso, A., Katz, B.F.G., Blum, A., Jacquemin, C., and Denis, M. (2005). *A Study of Spatial Cognition in an Immersive Virtual Audio Environment: Comparing Blind and Blindfolded Individuals*. In Proceedings of ICAD 05-Eleventh Meeting of the International Conference on Auditory Display, Limerick, Ireland, July 6-9.

Allen, G., B.W., Goodale, T., Hege, H., Lanfermann, G., Merzky, A., Radke, T., and Seidel, E. (2000). *The Cactus Code: A Problem Solving Environment for the Grid*. In Proc. High Performance Distributed Computing (HPDC-2000). IEEE Computer Society.

Allen, J.B., and Berkley, D.A. (1979). *Image method for efficiently simulating small-room acoustics*. Journal of Acoustical Society of America 65(4): 943-950.

Alpaslan, Z.Y., Yeh, S.C., Rizzo, A.A. and Sawchuk, A.A. (2005). *Quantitative Comparison of Two Stereoscopic Three-Dimensional Computer Interaction Methods*. Published in Proc. Stereoscopic Displays and Virtual Reality Systems XII Symposium, Proc. SPIE, Vol. 5664, San Jose, CA.

Anderson, D.B., Barrus, J.W., and Howard, J.H. (1995). *Building multi-user interactive multimedia environments at MERL*. IEEE Multimedia, (4):77-82.

Anitescu, M., and Potra, F.A. (1997). Formulating Dynamic Multi-Rigid-Body Contact Problems with Friction as Solvable Linear Complementarity Problems. Nonlinear Dynam 14(3): 231-247.

Armstrong, R., Dennis, G., Geist, A., Keahey, K., Kohn, S., McInnes, L., Parker, S., and Smolinski, B. (1999). *Towards a Common Component Architecture for High-Performance Scientific Computing*. Eight IEEE International Symposium on High Perfomance Distributed Computing.

Averback, R. S., Diaz de la Rubia, T. (1998) in *Solid State Physics* **51** (Eds Spaepen F. *et al.*) 281–402 (Academic, New York)

Baraff, D. (1990). *Curved Surfaces and Coherence for Non-penetrating Rigid Body Simulations*. Computer Graphics (Proc. SIGGRAPH), ACM.

Baraff, D. (1994). *Fast Contact Force Computation for Nonpenetrating Rigid Bodies*. SIGGRAPH 94 Conference Proceedings, Annual Conference Series, Addison Wesley, ACM SIGGRAPH.

Bayarri, S., Fernández, M., and Pérez, M. (1996). *Virtual Reality For Driving Simulation*. Communications of ACM, Vol 39, Nº 5, Pag:72-76.

Belleman, R.G., Kaandorp J.A., and Sloot, P.M.A. (1998). *Virtual environment for the exploration of diffusion and flow phenomena in complex geometries*. Future Generation Computer Systems. 14 (3-4). 209-214.

Benali Khoudja, M., Hafez, M., Alexandre, J.M., and Kheddar, A. (2003). *Electromagnetically Driven High-Density Tactile Interface Based on a Multi-Layer Approach*. The Best Paper in International Symposium on Micromechatronics and Human Science, IEEE Conference, Nagoya Japan.

Benali Khoudja, M., Hafez, M. (2004). *VITAL: A VibroTActiLe Interface with Thermal Feedback*. Journée Scientifique Internationale IRCICA, Lille.

Benford, S., Bowers, J., Fahlén, L., Greenhalgh, C., Mariani, J., amd Rodden, T. (1995). *Networked Virtual Reality and Cooperative Work*. Presence: Teleoperators and Virtual Environments 4(4): 364-386.

Bergault, D.R. (1994). *3-D Sound for Virtual Reality and Multimedia*. Academic Press, Cambridge MA.

Berkhout, A.J. (1988). *A Holographic Approach to Acoustic Control*. Journal of the Audio Engineering Society 36: 977-995.

Bhatnagar, D.K. (1993). *Position Trackers for Head Mounted Display Systems: A Survey*. UNC Technical Report No. TR93-010, Chapel Hill, NC: Dept. of Computer Science,, University of North Carolina.

Biermann, P., Jung, B., Latoschik, M., and Wachsmuth, I. (2002). *Virtuelle Werkstatt: A Platform for Multimodal Assembly in VR*. In Proceedings VRIC'02. Laval, France.

Blauert, J., (1997), *Spatial hearing - the psychacoustics of human sound localization* , The MIT Press, Cambridge, MA

Blauert, J., Lehnert, H., Sahrhage, J., and Strauss, H. (2000). *An Interactive Virtual-Environment Generator for*

Borish, J. (1984). *Extension of the image model to arbitrary polyhedra*. Journal of Acoustical Society of America 75(6): 1827-1836.

Bossard, B. (2005). *Gestural Interaction for Virtual Scene Description*. Proceedings of Gesture Workshop.

Boulic, R., Silaghi, M.-C. and Thalmann, D. (2000). *Visualization of local movements for optimal marker positioning*. In AMDO. pp 133-144.

Bouras, C., Fotakis, D., and Philopoulos, A. (1998). *A distributed virtual learning centre in cyberspace*. Proceedings of the 4th International Conference on Virtual Systems and Multimedia (VSMM'98).

Bourdot, P. and Touraine, D. (2002). *Polyvalent display framework to control virtual navigations by 6DOF tracking*. In IEEE Virtual Reality Conference 2002, IEEE VR 2002, Orlando, Florida (US).

Braffort, A., Gherbi, R., Gibet, S., Richardson, J., and Teil, D. (1999). *Gesture-Based Communication in Human-Computer Interaction*. International Gesture Workshop, GW'99, Gif sur Yvette, France. Lecture Notes in Artificial Intelligence n° 1739.

Brown, J .P., Matthewman, J .C. *The Cambridge Crystallograhic Subroutine Library.* RL-81-063.

Bryson, S., and Levit, C. (1992). *The Virtual Wind Tunnel*. IEEE Computer Graphics and Applications. Future Generation Computer Systems. 12 (4). 25-34.

Burdea, G. and Coiffet, P. (1993). *La réalite virtuelle*. Hermes Publishing. Paris.

Butterworth, J., Davidson, A., Hench, S., and Olano, T.M. (1992). *3DM: A Three Dimensional Modeler Using a Head-Mounted Display*. In Proceedings ACM Symposium on Interactive 3D Graphics. pp 134-138.

Camacho, C.J. and Vajda, S. (2002) *Protein-protein association kinetics and protein docking*. Current opinion in structural Biology. 12. 36-40. Elseiver Science Ltd.

Canny, J.F. (1984). *Collision detection for moving polyhedra*. IEEE Trans. Patt. Anal. Mach. Intell. 8(2): 200-209.

Chi, E.H. (2002). *A Framework for Visualizing Information*. Netherlands: Kluwer Academic Publishers.

Christensen, R., Hollerbach, J.M., Xu, Y. and Meek, S. (2000). *Inertial force feedback for the Treadport locomotion interface*. Presence: Teleoperators and Virtual Environments, 9 no. 1.

Chu, C.C.P., Dani, T.H., and Gadh, R. (1997). *Multi-Sensory User Interface for a Virtual-Reality-Based Computer Aided Design System*. Computer Aided-Design. 29 (10), pp 709-725.

Churchill, E.F., Snowdon, D.N., and Munro, A.J. (2001). *Collaborative Virtual Environments*. Springer-Verlag.

Coma, I., Fernández, M. et al. (2001). *An integrated system for the whole design process of driving simulation experiments*. Driving Simulation Conference (DSC'01). Sophia-Antipolis. France.

Coma, I., Fernández, M., Vera, L., and Olanda, R. (2005). *Dynamic driving scenario design*. IADAT Journal of Advanced Technology. ISSN1885-6411.

Convard, T., and Bourdot, P. (2004) *History Based Reactive Objects for Immersive CAD*. In the Proceedings of ACM Solid Modeling, SM'04. pp 291-296, Genova, Italy.

Convard, T., and Bourdot, P. (2005) *A Multimodal Immersive Solid Modeler with Reactive Objects*. To appear in the proceedings of the 11th International Conference on Human-Computer Interaction, HCII2005, Las Vegas, Nevada USA.

Coors, V., Jasnoch, U., and Jung, V. (1999). *Using the Virtual Table as an Interaction Platform for Collaborative Urban Planning*. Computers & Graphics 23(4): 487-496.

Corteel, E. (2004). *Création et manipulation de scènes sonores pour la Wave Field Synthesis*. Les cahiers de Louis Lumière, n° 2, Octobre.(in French)

Cruz-Neira, C., Gandin, D.J., and DeFanti, T.A. (1993). *Surround-screen projection-based virtual reality: The design and implementation of the CAVE*. In the proceedings of ACM SIGGRAPH 93. pp 135-142.

CT. (2005). *Claron Technology Inc*. http://www.clarontech.com/3rd_generation.htm.

Deisinger, J., Blach, R., Wesche, G., Breining, R., and Simon, A. (2000). *Towards Immersive Modeling - Challenges and Recommendations: A Workshop Analyzing the Needs of Designers*. In the proceedings of Eurographics Workshop on Virtual Environments (EGVE00). Amsterdam, The Netherlands.

Diehl, H., Muller, F., and Lindemann, U. (2004). *From raw 3D-Sketches to exact CAD product models?Concept for an assistant-system*. EUROGRAPHICS Workshop on Sketch-Based Interfaces andModeling.

Dongarra J.J., Hempel, R., Hey, A.J.G., and Walker, D.W. (1993). *A proposal for a user-level, message passing interface in a distributed memory environment*. Technical Report TM-12231, Oak Ridge National Laboratory.

Duda, R.O., Hart P.E., and Stork, D.G. (2000). *Pattern Classification*. Wiley Interscience.

Durlach, N.I., and Mavor, A. S. (1994). *Virtual Reality: Scientific and Technological Challenges*. Editors: Committee on Virtual Reality Research and Development, National Research Council.

EST. (2005). *Engineering System Technologies Gmbh & Co*. http://www.est-kl.com.

Fagg, G.E., Moore, K., Dongarra, J.J., and Geist, A. (1997). *Scalable Network Information Processing Environment (SNIPE)*. SC'97.

Falby, J., Zyda, M., Pratt, D., and Mackey, R. (1993). *Npsnet: Hierarchical data structures for real-time three-dimensional visual simulation*. Computers & Graphics 17(1): 6569.

Farmer, E., Van Rooij, J., Riemersma, J., Jorna, P., and Moraal, J. (1999). *Handbook of Simulator-Based Training*. Ashgate Pub. Ltd., Hants.

Faugeras, O. (1993). *Three-dimensional computer vision: a geometric Viewpoint*. MIT Press.

Férey, N., Gros, P.-E., Hérisson, J., and Gherbi, R. (2004) *Exploration by visualization of numerical and textual genomic data*. Journal of Biological Physics and Chemistry 4. 102-110.

Férey, N., Gros, P.-E., Hérisson, J., and Gherbi, R. (2004b). *A Distributed Multimedia Database Visualization within an Immersive Environment for Bioinformatics*. IEEE Sixth International Symposium on Multimedia Software Engineering, December 13-15 2004, Florida International University, Miami (FL). USA.

Fernandez-Recio, J., Totrov, M., and Abagyan, R. (2002) Soft protein-protein docking in internal coordinates. Protein Science, 11 :280-291, Cold Spring Harbor laboratory Press.

Fiorentino Michele, D.A.R., Stork, A., and Monno, G. (2002). *Spacedesign: A Mixed Reality Workspace for Aesthetic Industrial Design*. ISMAR 2002 IEEE and ACM international synopsium on mixed and augmented reality, Darmstadt, Germany.

Foley, J.D., van Dam, A., Feiner, S.K., and Hughes, J.F. (1996). *Computer Graphics - principles and practice*. Addison-Wesley.

Foster I., and Kesselman, C. (1997). *Globus: A Metacomputing Infrastructure Toolkit. International Journal of Supercomputer Applications*. 11(2): 115-128.

Foxlin, E., Harrington, M. and Altsuler, Y. (1998) *Miniature 6-DOF Inertial System for Tracking HMDs*. SPIE vol. 3362 Helmet and HeadMounted Display III, Orlando, FL, April.

Funkhouser, T., Min, P., and Carlbom, I. (1999). *Real-time Acoustic Modeling for Distributed Virtual Environments*. ACM Computer Graphics, SIGGRAPH'99 Proceedings.

Gallo, E., and Tsingos, N. (2004). *Efficient 3D-audio processing with the GPU*. GP2, ACM Workshop on General Purpose Computing on Graphics Processors , Los Angeles

Gardner, W. (1997). *Reverberation Algorithms*. Applications of Digital Signal Processing to Audio and Acoustics. Kluwer Academic Press.

Gherbi, R. and Hérisson, J. (2001) *3d Modelling tools for spatial-based in silico analysis of DNA*. In International electronic journal on Computer Graphics and Geometry. Volume 3, N° 1.

Gerzon, M. (1973). *Periphony: With-height sound reproduction*. Journal of the Audio Engineering Society 21(1/2): 2-10.

Goldberg, D.E. (1989). *Genetic algorithms: in search optimisation and machine learning*. Addison-Wesley.

Gomes de Sà, A., and Zachmann, G. (1999). *Virtual Reality as a Tool for Verification of Assembly and Maintenance Processes*. Computers and Graphics. 23 (3), p 389-403.

Gonzalez, R., and Woods, R. (2002). *Digital Image Processing*. Addison-Wesley.

Gosselin, F., A. R., Coulon-Lauture, F., and Rougeux, D. (2001). *VIRTUOSE 3D: a new input device for virtual reality*. VRIC 2001 Proceedings - Laval Virtual, Virtual Reality International Conference, Laval, France.

Gosselin, F., Bidard, C., and Brisset, J. (2005). *Design of a high fidelity haptic device for telesurgery*. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA 2005, Barcelona, Spain.

Gottschalk, S., Lin, M.C., and Manocha, D. (1996). *OBB-Tree: A Hierarchical Structure for Rapid Interference Detection*. SIGGRAPH 96 Conference Proceedings, Annual Conference Series, ACM SIGGRAPH, Addison Wesley.

Govindaraju, N.K., Redon, S., Lin, M.C., and Manocha, D. (2003). *CULLIDE: Interactive Collision Detection between Complex Models in Large Environments using Graphics Hardware*. ACM SIGGRAPH/Eurographics Graphics Hardware Proceedings.

Greenhalgh, F.C. (1997). *Analysing movement and world transitions in virtual reality teleconferencing*. Proceedings of 5th European Conference on Computer Supported Cooperative Work (ECSCW'97).

Gregory, A., Ehmann, S., and Lin, M.C. (2000). *inTouch: Interactive Multiresolution Modeling and 3D Painting with a Haptic Interface*. In Proceedings IEEE Virtual Reality Conference (VR'00). New Brunswick, New Jersey. p 45-52.

Grimshaw A. S., and Wulf, W.A. (1997). *The Legion Vision of a Worldwide Virtual Computer*. Commun. ACM 40(1).

Guastavino, C., and Katz, B.F.G. (2004) *Perceptual evaluation of multi-dimensional spatial audio reproduction*. The Journal of the Acoustical Society of America, Vol.116, n°2.

Hafez, M., and Benali-Khoudja, M. (2004). *3D Tactile Rendering Based on Bi(multi) stable SMA Monolithic Systems*. IEEE Conference, Nagoya Japan.

Hagsand, O. (1996). *Interactive MultiUser VEs in the DIVE System*. IEEE Multimedia Magazine 3(1).

Halperin, I., Ma, B., Wolfson, H. and Nussinov, R. (2002). *Principles of docking: An overview of search algorithms and a guide to scoring functions*. Proteins 2002 Jun 1, 47(4). 409-430.

Haykin, S.S. (1998). *Neural networks - a comprehensive foundation*. Prentice Hall.

Hérisson, J., Gros, P.-E., Férey, N., Magneau, O., and Gherbi, R. (2004). *DNA in Virtuo: Visualization and Exploration of 3D Genomic Structures. 3rd ACM-SIGRAPH*. International Conference on Virtual Reality, Computer Graphics, Visualization and Interaction, Stellenbosch (Cap town), South Africa

Herrera, G., Plasencia, A., Labajo, G., Jordan, R., and de Pablo, C. (2004). *Using'Ambient Intelligence' for Compensating Intellectual Difficulties of People with Sever Learning Difficulties and/or Austistic Spectrum Disorders*. Lecture Notes in Computer Science. Vol. 3118-pages 969-975.

Hoffmann, H., Dachselt, R., and Meissner, K. (2003). *An Independent Declarative 3d Audio Format On The Basis Of XML*. In Proceedings of the 2003 International Conference on Auditory Display, Boston (MA). USA.

Holloway, R., and Lastra, A. (1993). Virtual *Environments: A Survey of the Technology*. UNC Technical Report No. TR93-033, Chapel Hill, NC: Dept. of Computer Science, University of North Carolina.

Hopkins, L.D., Johnston, D.M., and George, R.V. (1999). *Computer Support for Sketch Planning*. Computers in Urban Planning and Urban Management On the Edge of the Millenium, Venice.

Howard, I.P. (1986). *The vestibular system*. Handbook of Perception and Human Performance.

Ishii, H., Ben-Joseph, E., Underkoffler, J., Yeung, L., Chak, D., Kanji, Z., and Piper, B. (2002). *Augmented Urban Planning Workbench: Overlaying Drawings*. Physical Models and Digital Simulation. IEEE and ACM International Symposium on Mixed and Augmented Reality, Darmstadt, Germany.

Iwata, H. (1991). *Force Display for Virtual Worlds*. Proceedings of the International Conference on Artificial Reality and Tele-existence, Tokyo, Japan.

Jacquemin, C., Folch, H., and Nugier, S. (2005). *Exploration d'analyse de données textuelles et navigation contrôlée dans OCEAN*. In Proceedings, Conférence Francophone sur l'Interaction Homme-Machine IHM'05, Toulouse, France.

Jacquemin C., Folch H., and Nugier S. (2006). *OCEAN: 2 1/2D Interactive Visual Data Mining of Text Documents*. In Proceedings, 10th International Conference on Information Visualisation, IV 06, London, UK.

Jain, R., Kasturi, R., and Schunck, B.G. (1995). *Machine Vision*. McGraw-Hill.

Jot, J.-M. (1999). *Real-time spatial processing of sounds for music, multimedia and interactive human-computer interfaces*. Multimedia Systems, Special Issue on Audio and Multimedia.

Junghyun, K.S.K., Ungyeon, Y., Namgyu, K. (1998). *COVRA-CAD: A CORBA based Virtual Reality Architecture for CAD*. Proc. of Int. Conference on Virtual Systems and Multimedia.

Kanade, T., Narayanan, P.J., and Rander, P.W. (1995). *Virtualised Reality: concepts and early results*. IEEE Workshop on the representation of visual scenes.

Kanade, T., Rander, P.W., and Narayanan, P.J. (1997). *Virtualised reality: constructing virtual worlds from real scenes*. IEEE Multimedia Magazine 1(1): 34-47.

Kang, S.B. (1999). *A survey of image based rendering techniques*. SPIE 3641: 2-16.

Katz, B.F. (2001) *Boundary element method calculation of individual head-related transfer function. I. Rigid model calculation*. Journal of the Acoustical Society of America, vol. 110 (5).

Kim, Y.J., Otaduy, M.A., Lin, M.C., and Manocha, D. (2002). *Fast Penetration Depth Computation for Physically-based Animation*. ACM Symposium on Computer Animation.

Kiyokawa, K., Takemura, H., Katayma, Y., Iawasa, H., and Yokoya, N. (1997). VLEGO*: A Simple Two-Handed Modelling Environment Based on Toy Blocks*. Proceedings VRST'97. pp 27-34.

Koutek, M., van Hees, J., Bakker, A.F., and Post, F.H. (2002). *Virtual Spring Manipulators for Particle Steering in Molecular Dynamics on the Responsive Workbench.* In proceedings of the EUROGRAPHICS Workshop on Virtual Environment (EGVE 2002). Barcelona, Spain.

Kruger, W., and Frohlich, B. (1995). *The Responsive Workbench: A Virtual Work Environment*. IEEE Computer Graphics 28(7): 42-48.

Kuester, F., Duchaineau, M.A., Hamann, B., Joy, K.I., and Ma, K.L. (2000). *The DesignersWorkbench: Towards Real-Time Immersive Modeling*. IS&T/SPIE Electronic Imaging, San Jose, California.

Latoschik, M.E. (2001). *A General Framework for Multimodal Interaction in {VR} : Pr{OSA}*. Proceedings of IEEE Virtual Reality.

Laviola, J.J. (1999). *MSVT: A Virtual Reality-Based Multimodal Scientific Vizualisation Tool*. Proceedings of 2nd International Conference on Computer Graphics and Imaging, pages 221-225.

Lee, D., Lim, M., and Han, S. (2002). *Atlas - a scalable network framework for distributed virtual environments*. Proceedings of ACM Collaborative Virtual Environments (CVE 2002): 47-54.

Lee, S., Chen, T., Kim, J., Kim, G.J., Han, S., and Pan, Z. (2004). *Affective Property Evaluation of Virtual Product Designs*. In Proceedings IEEE Virtual Reality Conference (VR'04). Chicago, IL, USA.

Lewis, M., and Jacobson, J. (2002). *Game engines in scientific research*. Communications of the ACM 45(17-31).

Lin, M.C., and Gottschalk, S. (1998). *Collision detection between geometric models: a survey*. IMA Conference on Mathematics of Surfaces. San Diego (CA).

Listen Project. (2007). Information Society Technologies Program - IST-1999-20646: http://listen.gmd.de/

Llamas, I., Kim, B., Gargus, J., Rossignac, J., and Shaw, C.D. (2003). *Twister: A space-warp operator for the two-handed editing of 3D shapes*. ACM Transactions on Graphics (TOG). 22 (3), pp 663-668.

Lokki, T., Savioja, L., Vaananen, R., Huopaniemi, J., and Takala, T. (2002). *Creating Interactive Virtual Auditory Environments*. IEEE Computer Graphics and Applications 22(4): 49-57.

Lotstedt, P. (1984). *Numerical simulation of time-dependent contact friction problems in rigid body mechanics*. SIAM Journal of Scientific Statistical Computing 5(2): 370- 393.

Lu, Y.H., Wang, W.T., Liang, R.H., and Ouhyoung, M. (2002). *Virtual Sculptor: A Feature Preserving Haptic Modeling System*. In Proceedings ACM International Workshop on Immersive Telepresence (ITP2002). Juan Les Pins, France.

Ma, J., Gao, W., and Wang, R. (2000). *A parallel multistream model for integration of sign language recognition and lip motion*. Lecture Notes Computer Science 1948: 582-589.

Macedonia, M. and Zyda, M. (1996). *A taxonomy for networked virtual environments*. IEEE Multimedia Magazine 4(1): 48-56.

MacNaughton, B., Gaudreau, J.E., Bechamp, M., and Power V.S. (2006). *Stereoscopic Displays and Virtual Reality Systems XIII.* Proc. SPIE Vol. 6055, 605518, Stereoscopic Displays and Virtual Reality Systems XIII; Andrew J. Woods, Neil A. Dodgson, John O. Merritt, Mark T. Bolas, Ian E. McDowall; Eds.

Magnenat-Thalmann, N., Montagnol, M., Bonanni, U., Gupta. R. (2007) Visuo-Haptic Interface for Hair. Proc. Of CyberWorlds'07

Magnenat-Thalmann, N., Volino, P., Bonanni, U., Summers, I.R., Bergamasco, M. (2007) Haptic Simulation, Perception and Manipulation of Deformable Objects, Tutorial Notes, Proc. of EUROGRAPHICS '07,Computer Graphics Forum, vol. 26

Magnenat-Thalmann, N., Volino, P., Bonanni, U., Summers, I. R., Bergamasco, M., Salsedo, F., Wolter, F. E. (2007). From physics-based simulation to the touching of textiles: The HAPTEX Project. Proc. of ICEC '07

Magneau, O., Bourdot, P., and Gherbi, R. (2002). *3D tracking based on infrared cameras*. In International Conference on Computer Vision and Graphics, Zakopane, Poland.

Magneau, O., Bourdot, P., and Gherbi, R. (2004). *Positioning and identification of markers for 3D tracking*. Mécanique & Industries 5, p 221-227.

Mahmoudi, F., Parviz, M. (2006). *Visual Hand Tracking Algorithms*. Geometric Modeling and Imaging-New Trends, 2006: 228-232.

Mann, R., Stanley, S., Vlaev, D., Wabo, E., and Primrose, K. (2001). *Augmented-reality visualization of fluid mixing in stirred chemical reactors using electrical resistance tomography*. Journal of Electronic Imaging. 10 (3). 620-629.

Massie, T.H. and Salisbury, J.K. (1994). The PHANToM haptic interface : a device for probing virtual objects. Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Chicago.

Matthews, I., Cootes, T.F., Bangham, J.A., Cox, S., and Harvey, R. (2002). *Extraction of visual features for lipreading*. IEEE Trans. PAM 24(2): p 198-213.

McAffee, D.A., and Fiorini, P. (1991). *Hand Controller Design Requirements and Performance Issues inTelerobotics*. ICAR 91, Robots in Unstructured Environments, Pisa, Italy.

McLachlan, G.J. (1992). *Discriminant analysis and statistical pattern recognition*. Wiley-Interscience press.

Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs*. Springer-Verlag.

Miller, D.C., and Torpe, J.A. (1995). *Simnet: The advent of simulator networking*. IEEE 83: p. 1114-1123.

Mine, M.R. (1995). *Virtual Environment Interaction Techniques*. Technical report, University of North Carolina.

Morillo, P., Orduña, J.M., and Fernández, M. (2003). *A Comparison Study of evolutive algorithms for Solving the Partitioning Problem in Distributed Virtual Environment Systems*. In Parallel Computing, Special Issue of Parallel and Nature-inspired Computational Paradigms and Applications , Vol. 30/5-6. p 585-610, Springer-Verlag, New York.

Morris, G.M., Goodsell, D.S., Halliday, R.S., Huey, R., Hart, W.E., Belew, R.K., and Olson, A. J. (1998). *Automated Docking Using a Lamarckian Genetic Algorithm and and Empirical Binding Free Energy Function*. Journal of Computational Chemistry. 19. 1639-1662.

Mulder, A. (1994). *Human movement tracking technology*. Technical report 94-1, School of Kinesiology, Simon Fraser University.

Naef, Staadt, O., and Gross, M. (2002). *Spatialized audio rendering for immersive virtual environments*. In Proceedings of the ACM symposium on Virtual reality software and technology (VRST 2002).

Nagata, H., Tanaka, E., Hatsuta, M., Mizushima, H., and Tanaka H. (2000) Molecular Virtual Reality System with Force Feedback Device. ICAT Conference.

Nelson, P.A., Orduna-Bustamante, F., and Hamada, H. (1995). *Inverse Filter Design and Equalization Zones in Multichannel Sound Reproduction*. IEEE Trans. Speech and Audio Processing 3(3).

Nordlund, K., Ghaly, M., Averback, R.S., Caturla, M., Diaz de la Rubia, T., Tarus, J. (1998) Phys. Rev. B, **57**, 7556

Nordlund, K., Keinonen, J., Ghaly, M., Averback, R. S. (1999) Nature **49**, 398.

Norman, S.a.D., S. (1986). *User Centered System Design: New Perpectives on Human-Computer Interaction*. London: LEA.

Ohya, J., Kitamura, Y., Takemura, H., Kishino, F., and Terashima, N. (1994). *Real-time Reproduction of 3D Human Images in Virtual Space Teleconferencing*. VR 1993.

OMG. (1995). *The Common Object Request Broker: Architecture and Specification*. Revision 2.0. OMG Document.

Ong, C.J., and Gilbert, E.G. (1997). The Gilbert-Johnson-Keerthi distance algorithm: A fast version for incremental motions. Proceedings of the IEEE International Conference on Robotics and Automation.

Pappas, M., Fragos, D., Alexopoulos, K., Karabatsou, V. (2003). *Development of a Three-Finger Grasping Technique on a VR Glove*. Proceedings of Virtual Concept.

Park, K.S., Cho, Y.J., Krishnaprasad, N.K., Scharver, C., Lewis, M.J., Leigh, J., and Johnson, A.E. (2000). *CAVERNSoft G2: A toolkit for High Performance Tele-Immersive Collaboration*. In proceedings of the ACM Symposium on Virtual Reality Software and Technology.

Parker, J. (1997). *Algorithms for image processing and computer vision*. John Wiley and Sons.

Parker, S.G. (1999). *The SCIRun Problem Solving Environment and Computational Steering Software System*. PhD Thesis.

Pedrycz, W., and Gomide, F. (1998). *An introduction to fuzzy sets: analysis and design*. MIT Press.

Pérez, M., Fernández, M., and Lozano, M. (2002). *Adding Synthetic Detail To Natural Terrain Using a Wavelet Approach*. Lecture Notes in Computer Science. Vol. 2330. p 22-31. ISSN: 0302-9743.

Pérez, M., Fernández, M., Morillo, P., and Coma, I. (2004). *Locally Constrained Synthetic LoDs Generation for Natural Terrain Meshes*. Future Generation Computer Systems on Computer Volume 20, Issue 8, p 1375-1387, Computer Graphics and Geometric Modeling. Elsevier Science Journal.

Popovici, D.M. (2004). *Modeling the space in virtual universes*, PhD Thesis: Politechnica University of Bucharest.

Poveda, S.J., and Perez, M. (1998). *GEOINDEX: A virtual window to greographical information systems*. Complex Systems, Intelligen Systems & Interfaces. Nimes. France.

Pratini, E. (2001). *New Approaches to 3D Gestural Modeling - the 3D SketchMaker Project*. Proceedings of ECAADE.

Pulkki, V. (1997). Virtual sound source positioning using vector base amplitude panning. Journal of the Audio Engineering Society 45(6): p 456-466.

Redon, S., Kheddar, A., and Coquillart, S. (2000). *An Algebraic Solution to the Problem of collision Detection for Rigid Polyhedral Objects*. In proceedings of IEEE International Conference on Robotics and Automation.

Refsland, S.T. et al. (1998). *Virtual Great Barrier Reef: A theoretical Approach towards and Evolving, Interactive VR Environment Using a Distributed DOME and CAVE System*. Distinguished Paper Award, Virtual Worlds: Lecture Notes in Artificial Intelligence, First International Conference, VW'98, Paris, France.

Reignier, P., Harrouet, F., Morvan, S., Tisseau, J., Duval, T. (1998). *AReVi: A Virtual Reality Multi-agent Platform*, Lecture Notes in Computer Science, Vol. 1434. p 229-240.

Reitmayr, G., and Schmalstieg, D. (2005). *Flexible Parametrization of Scene Graphs*. In Proceedings of VR 2005.

Ribo, M. (2001). *State of the Art Report on Optical Tracking*. Technical Report. VRVis 2001.

Rio, E., Vandernoot, G., and Warusfel, O. (2003). *Perceptual Evaluation of Weighted Multi-Channel Binaural Format*. Proc. of 6th Int. Conference on Digital Audio Effects (DAFx-03), London, UK

Rolland, J.P., Davis, L.D., and Baillot, Y. (2000). *A survey of tracking technology for virtual environments*. In Fundamentals of Wearable Computers and Augmented Reality. Ed. Barfield and Caudell.

Salles, J.M., Galli, R., Almeida, A.C., Belo, C.A.C., and Rebordao, J.M. (1997). *Mworld: A multiuser 3D virtual environment*. IEEE Computer Graphics 17(2): p 55-65.

Savchenko, V. (2000). *3-D Geometric Modeller with Haptic Feedback: engraving simulation*. In Proceedings Phantom Users Research Symposium.

Savioja, L., H., J., Lokki, T., and Vaananen, R. (1999). *Creating Interactive Virtual Acoustic Environments*. Journal of the Audio Engineering Society 47(9): p 675-705.

Schalkoff, R. (1989). *Digital image processing and computer vision*. John Wiley & Sons Inc.

Scharstein, D., and Szeliski, R. (2002). *A taxonomy and evaluation of dense two-frame stereo correspondence algorithms*. IJCV 47(1-3): 7-42.

Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavári, Z., Encarnação, M., Gervautz, M., and Purgathofer, W. (2002). *The Studierstube Augmented Reality Project*. PRESENCE - Teleoperators and Virtual Environments, MIT Press.

Schmidt, D.C. (2006) *Our research on high-performance and real-time corba*. Available at: http://www.cs.wustl.edu/~schmidt/corba-research-overview.html

Serón, F.J., Rodriguez, R., Cerezo, E., and Pina A. (2002). *Adding Support for High-Level Skeletal Animation*. IEEE Transactions on Visualization and Computer Graphics.

Singhal, S., and Zyda, M. (1999). Networked Virtual Environments: Design and Implementation. New York, ACM Press.

Snyder, J. (1992). *Interval analysis for Computer Graphics*. Computer Graphics 26(2): p 121-130.

Sonka, M., Hlavac, V., and Boyle, R. (1998). *Image processing analysis and machine vision*. Chapman & Hall Computing.

ŞOPU, D., POPOVICI, D.M., GÎRŢU, M.A. (2007). *Molecular dynamics simulation of defect formation in irradiated face centered cubic materials*, Optoelectron. Adv. Mater. 9. p 799-809

Sowa, T., and Wachsmuth, I. (2001). *Interpretation of Shape-Related Iconic Gestures in Virtual Environments*. In Gesture and Sign Language in Human-Computer Interaction, p 21-33, LNAI 2298, Springer-Verlag.

Stevens, R. and Judson, I.R. (1997). *Using Immersive Virtual Reality for Visualizing and Modeling of Molecular Nanosystems*. Fifth Foresight Conference on Molecular Nanotechnology. Palo Alto (CA).

Stewart, D.E., and Trinkle, J.C. (1996). *An Implicit Time-Stepping Scheme for Rigid Body Dynamics with Inelastic collisions and Coulomb Friction*. International Journal of Numerical Methods in Engineering.

Strauss, H., and Buchholz, J. (1999). *Comparison of Virtual Sound Source Positioning with Amplitude Panning and Ambisonic Reproduction*. In: Collected Papers from the Joint ASA, EAA and DAGA Meeting "Berlin 99", DEGA Oldenburg.

Strauss, P. and Carey, R. (1992). *An Object Oriented 3D Graphics Toolkit*. In Proc. SIGGRAPH'92.

Sturman, D.J. (1992). *Whole-hand input*. PhD thesis, Media, Arts & Science.

Su, S.A. and Furuta, R. (1993). A specification of 3d manipulation in virtual environments. In 1993 IEEE Annual Virtual Reality International Symposium. p 3837-393.

Tarault, A., Bourdot, P., and Vézien, J.-M. (2005). *SACARI : An Immersive Remote Driving Interface for Autonomous Vehicles.* Workshop on Computer Graphics and Geometric Modelling (CGGM' 05), in 5th International Conference on Computational Science (ICCS' 05). Atlanta (GA), USA.

Tarus J., Nordlund, K. (2003). Nucl. Instr. Meth. Phys. Res. B **212**, 281

Torguet, P., Balet, O., Gobbetti, E., Jessel, J.P., Duchon, J., and Bouvier, E. (2000). *CAVALCADE: A system for collaborative prototyping*. International Journal of Design and Innovation Research 2(1): p 76-89.

Trindade, J., Paiva, J.C., and Fiolhais, C. (2001) *Visualising molecules: on-line simulations and virtual reality*. Europhysics News. Vol. 32 No. 1.

Tsingos, N., Gallo, E., and Drettakis, G. (2004) *Perceptual Audio Rendering of Complex Virtual Environments*. In proceedings of SIGGRAPH 2004.

Usoh, M., Slater, M., and Vassilev, T.I. (1996). *Collaborative Geometrical Modelling in Immersive Virtual Environments*. In Proceedings of the Eurographics workshop on Virtual environments and scientific visualization '96. Monte Carlo, Monac. pp 111-120.

Von Der Heyde, M., and Riecke, B.E. (2001). *How to cheat in motion simulation approach to motion cueing*. Technical Report No. 089.

Wasfy, T.M., and Noor, A.K. (2001) *Visualization of CFD results in immersive virtual environments*. Advances in Engineering Software. 32 (9). p 717-730.

Wesche, G. (1999) *Three-dimensional visualization of fluid dynamics on the Responsive Workbench*. Future-Generation-Computer-Systems. 15 (4). p 469-475.

Voss, G., Behr, G., Reiners, D., and Roth, M. (2002). *A multi-thread safe foundation for scene graphs and its extension to clusters*. In Proceedings of the Fourth Eurographics Workshop on Parallel Graphics and Visualization, p 33-37.

Weiskopf, D. (2000). *An Immersive Virtual Environment for Special Relativity*. WSCG Conference Proceedings, V. Skala (ed.), University of West Bohemia, Pilsen, 337-344.

Wolff, R., Roberts, D.J., Steed, A., and Otto, O. (2006). *A Review of Tele-collaboration Technologies with Respect to Closely Coupled Collaboration*. International Journal of Computer Applications in Technology (IJCAT), Special Issue on: Collaborative Multimedia Applications in Technology.

Youngblut, C., Johnson, R., Nash, S., Wienclaw, R. and Will, C. (1996). *Review of Virtual Environment Interface Technology*. IDA Paper P-3186, Institute for Defense Analyses, 1801 N. Beauregard Street, Alexandria, VA 22311-1772.

Zagan, R., Petculescu, P., Prodan, G., Peride, N., (2003). *Comparison between ultrasonic and wavelets analysis for characterization stainless steel alloys*, World Congress on Ultrasonic, Paris, ISSN 1312-1669, p 62-64.

Zhong, Y., Mueller-Wittig, W., and Ma, W. (2002). *A Model Representation for Solid Modelling in a Virtual Reality Environment*. In Proceedings of the Shape Modeling international 2002 (Smi'02). SMI. IEEE Computer Society, Washington, DC, 183.

Important Web References:

http://3dviz.mc.com

http://imagis.cirad.fr/imagis/imagis_an.html

http://karma.geo.tudelft.nl

http://laimuz.unizar.es/simusys

http://www.acl.lanl.gov/cca-forum

http://www.alibre.com

http://www.ambisonic.net

http://www.barco.com/VirtualReality/en/products/product.asp?element=1970

http://www.creativelabs.com/eaudio/resource/eax3.html

http://www.enovia.com

http://www.esri.com

http://www.esri.com/software/arcgis/extensions/3danalyst/index.html

http://www.extreme.indiana.edu/ccat

http://www.fogscreen.com

http://www.geovrml.org

http://www.groove.net

http://www.iasig.org

http://www.intel.com/business/bss/products/server/idc_virtualization_wp.pdf

http://www.intel.com/business/bss/products/server/virtualization.htm

http://www.k2vi.com

http://www.metavr.com

http://www.metavr.com/products/vrsg/vrsgoverview.html

http://www.microsoft.com/DirectX/

http://www.microsoft.com/windows/netmeeting

http://www.multigen.com

http://www.multigen.com/products/standards/metaflight/index.shtml

http://www.multigen.com/products/standards/openflight/index.shtml

http://www.openrasmol.org

http://www.ptc.com/products/division/MockUp.htm

http://sourceforge.net/projects/arevi

http://www.terrasim.com

http://www.terrasim.com/products.html

http://www.terrex.com/www/index.htm

http://www.terrex.com/www/TerraVista.htm

http://www.tgs.com/

http://www.ubicom.tudelft.nl

http://www.vrgis.com

http://www.vmware.com/pdf/virtualization.pdf

http://wscg.zcu.cz

http://www.wacom-europe.com/uk/products/cintiq/cintiq18.asp