3. *VC++, VIRTUAL REALITY AND 3DS MAX*

3.1 INTRODUCTION

Structural aspects of the most devastating disasters have been studied in current work. Software implementation of analysis, design and retrofit for each disaster, has been accomplished through separate modules. Combination of various softwares has been used to accomplish a combination of end results. VC++ has primarily been used to develop the main program in each module for finding forces arising in a building, owing to each disaster. Analysis has been carried out either through VC++ program or through SAP2000. The concept of virtual reality has been built into each module to study the parameters of damage, displacement and forces for each disaster using 3DStudio Max and VRML (Virtual Reality Modeling Language) plug-in modules. All three softwares of VC++, SAP and 3DS Max have been interfaced in each module so that user does not have any difficulty in transiting from one software to other.

The present chapter deals with the basics of various softwares used in this research work. Each is a stand alone platform with powerful tools to serve the purpose of programming, modeling, analysis, design, 3D animation and web-enabling. Brief description of each has been given for an overview of the programming paradigm in the current work.

3.2 VISUAL C++ PROGRAMMING ENVIRONMENT

C and C++ languages have played a vital role in the programming paradigm. C++ has introduced an extraordinary package of prewritten, ready-to-use code of Microsoft Foundation Classes (MFC), thus making Windows programming a lot easier task and relieves programmers from long monolithic coding. The standard C++ techniques, such as inheritance and polymorphism make it possible to customize the MFC packages.

Visual C++ not only makes use of the MFC, but makes Windows programming accessible to the novice programmer, by introducing programming tools, such as the menu editor for designing menus, the dialog editor for designing dialog boxes, image editor to create toolbars etc. Visual C++ provides an integrated design environment in which one can write programs and run them. In addition, it has capabilities to organise many files that a Windows program needs for the projects. It is a package that provides a comprehensive, up-to-date production-level development environment for developing all Windows applications. Visual C++, [76] provides a powerful 32-bit compiler to develop Win32 application for Windows operating system.

3.3 VISUAL C++ COMPONENTS

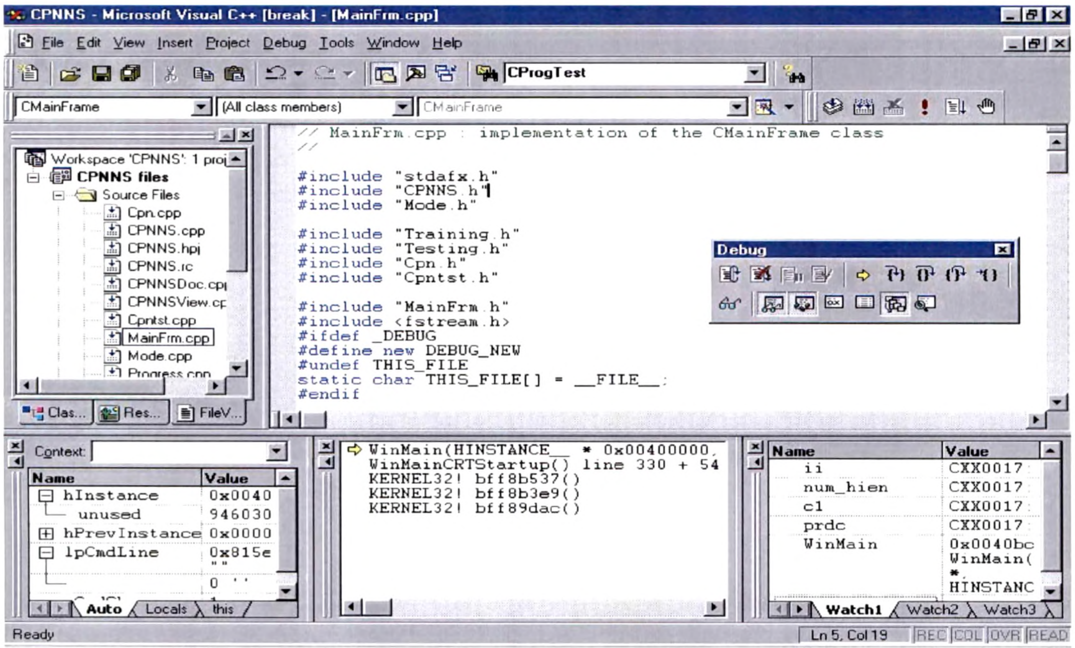
As mentioned earlier the main features of VC++ are available through its elegant components as given in the following paragraphs.

3.3.1 Microsoft Developer Studio and the Build Process

The Developer Studio is a Windows-hosted integrated development environment (IDE) that's shared by Visual C++, Visual Basic, Visual J++, Microsoft FORTRAN and some other products. This IDE has come a long way from the original Visual Workbench, which was based on QuickC for Windows. Docking windows and configurable toolbars, plus an editor that can be customized and runs macros, are included in the Developer Studio97 version. The on-line help system works like a Web browser. **Figure 3.1** shows Developer Studio in action.

To work with DevStudio IDE, one needs to known the concept of project, work space and make file. A project is a collection of interrelated source files that are compiled and linked to make up an executable Windows-based program or a DLL. Project source files are generally stored in a separate subdirectory. A project depends on many files outside the project subdirectory too, such as include files and library files. A Workspace Window is a docking window containing a set of tabbed panes that display information and different views of the projects. For VC++, the Workspace Window shows Class View, File View, Resource View and InfoView. A "make file" expresses all the interrelationships among source files (A source code file needs specific "include" files, an executable file requires certain object modules and libraries and so forth). A "make" program reads the make file and then invokes the compiler, assembler, linker and resource compiler to produce the final output, which is generally an executable file.

Fig. 3.1 VC++ Workspace, Editor & Dialog Window



3.3.2 The Resource Editors –Workspace Resource View

By simply clicking on the Resource View tab in the Visual C++ Workspace windows, a resource can be selected for editing. The main window hosts a resource editor appropriate for the resource type. The VC++ also hosts a wysiwyg (What You See Is What You Get) editor for menus, a powerful graphical editor for dialog boxes and tools for editing icons, bitmaps and strings. The dialog editor allows user to insert Active X controls in addition to standard Window controls and the new Windows common controls.

3.3.3 The C/C++ Compiler

The Visual C++ compiler can process both C source code and C++ source code It determines the language by looking at the source code's filename extension. A C extension indicates C source code, and CPP or CXX indicates C++ sources code. The compiler is compliant with all ANSI standards, including the latest recommendations of a working group on C++ libraries, and has additional Microsoft extensions. Templates, exceptions, and runtime type identification (RTTI) are fully supported in Visual C++ version 5.0. The new C++ standard Template Library (STL) is also included, although it is not integrated in to the MFC library.

3.3.4 The Resource Compiler

The visual C++ resource compiler reads an ASCII resource script (Rs) file from the resource editors and writes a binary RES file for the linker.

3.3.5 The Linker

The Linker reads the OBJ and RES files produced by the C/C++ compiler and the resource compiler, and it accesses LIB files for MFC code, runtime library code and windows code. It then writes the project EXE file. An incremental link option minimizes the execution time when only minor changes have been made to the source files. The MFC header files contain #pragma statements (special compiler directives) that specify the required library files, so one need not to tell the linker explicitly which libraries to read.

3.3.6 The Debugger

Good Debugging facility is the essential requirement for any compiler package. The visual C++ debugger has been steadily improving, but it doesn't actually fix the bugs yet. The debugger works closely with Developer Studio to ensure that breakpoints are saved on disk. Toolbar buttons insert and remove breakpoints and control single-step execution. **Figure 3.1** illustrates the visual C++ debugger in action. It can be noticed that the variable and watch windows can expand an object pointer to show all data members of the derived class and base classes. If the cursor is positioned on a simple variable, the debugger shows its value in a little window. To debug a program, user must build the program with the compiler and linker options set to generate debugging information.

3.3.7 AppWizard

AppWizard is a code generator that creates the working skeleton of a windows application with features, class names and source code filenames that you specify through dialog boxes. AppWizard code is minimalist code; the functionality is inside the application framework base classes. AppWizard gets programmer started quickly with a new application.

3.3.8 Class Wizard

ClassWizard is a program (implemented as a DLL) that's accessible from Developer Studio's View Menu. ClassWizard takes the drudgery out of maintaining Visual C++ class code. Class Wizard writes the prototypes, the function bodies and (if necessary) the code to link the Windows message to the function. Class Wizard can update class code that user writes, to avoid the maintenance problems common to ordinary code generators.

3.3.9 The Source Browser

The Visual C++ Source Browser lets user examine and edit an application from the class of function viewpoint instead of from the file viewpoint. It's little like the "inspector" tools available with object-oriented libraries such as Smalltalk.

3.3.10 Microsoft Foundation Classes

Microsoft's Visual C++ compiler provides a Foundation Class library, containing a new set of tools for the development of C++ and C++ Windows applications. The Foundation Class library comes in a 16-bit and 32-bit version. The 32-bit version is used for Windows 95, Windows 98 and Windows NT applications. The 32-bit version is an expanded library offering class support for control bars, property sheets, OLE and so on.

3.3.11 Microsoft Foundation Class Library Design Consideration

The foundation Class library design team set rigorous design principles that had to be followed in the implementation of the Foundation Class library. These principles and guidelines include the following:

- ◈ Utilise the power of C++ without overwhelming the programmer.

- Make the transition from standard API function calls to the use of class libraries as simple as possible.
- Allow the mixing of traditional function calls with the use of new class libraries.
- Balance power and efficiency in the design of class libraries.
- Produce a class Library that can migrate easily to new platforms.

These classes were designed in a fashion that requires minimal relearning of function names for seasoned windows programmers. This feature was achieved by carefully naming and designing class features. The Foundation Class library team also designed the Class library to allow a "mixed-mode" operation. That is, classes and traditional function calls can be intermixed in the same source code.

3.4 OVERVIEW OF VIRTUAL REALITY

The two approaches for interacting with a virtual world are Immersed and Non-immersed [37]. In the immersed approach, the user is immersed in the virtual world through large curved screens, body suits or head mounted device (HMD) in which the audio and visual perceptions of the user are as close to the real world, while isolating him/her from the external inputs. Head and body tracking devices move as the user moves and interaction with objects that exist in the virtual world occurs through sensory devices such as data gloves.

The non-immersed approach, also known as desktop virtual reality, enables users to interact with the virtual world with more conventional devices such as a keyboard, mouse, space ball, or joystick and a monitor. Although this doesn't give the same level of spatial awareness as in the immersed approach, it does provide users with a low cost solution, which seems to be an attractive compromise by many users who are uncertain about spending long hours in a helmet or heavy body suit.

3.4.1 Immersed Approach

Virtual Worlds can be created with the help of devices which enable the user to be completely immersed in them. The various head and body devices along with add on features of sound, touch and smell give artificial inputs (virtual) to the user who is completely cutoff from the external sensory inputs of the real world.

3.4.1.1 Head mounted display

Figure 3.2 shows a common setup for a Virtual Reality World. The subject wears a Head Mounted Display commonly known as HMD. The HMD is used to project a computer-generated world in front of the subject. On one of his hands, he wears a glove used to transmit information about the movement of the hand and its fingers. Above the subject hangs a transmitter. Receivers are placed on both the HMD and the glove. The transmitter and receiver together make up the trackers. These are used to get information about the place the subject occupies in the real world.

Fig. 3.2 Common Setup of Virtual Reality

An HMD (**Fig.3.3**) is a helmet with two little TV screens inside. Using optics, the picture on this miniature TV can be watched by the user. The optics is used to watch from close up or to enlarge the image. Using this optics, the image takes up the whole view of the user.

Fig. 3.3 Head Mounted Display and Data Glove

The computer-generated world consists of colored polygons. These polygons can also be colored using an image. The image is then put on the polygon and stretched until it fits the polygon. This technique is called Texture Mapping.

When there are light sources in the computer-generated world, for instance the sun, the technique of shading can be used to change the colors of the polygons according to the position of the light source. This helps to make the world look more real.

The Virtual World must be such that it appears in three-dimensional images, otherwise the display is not convincing. This requires binocular stereoscopic visual display, which is currently being done by using a bulky headset, rather like a motorcycle helmet.

The display must be wide angled, so that people can see things out of the corner of their eye, even when looking straight ahead. The headset uses a tracking device to detect the head movement with a scope to detect eye movements whilst the person stays still. Such systems are already being experimented for people with disabilities.

The latest generation of headsets is less bulky than the old motorcycle helmets, and looks more like a set of heavy-duty industrial goggles (**Fig. 3.4**). In future one can expect Virtual Reality Spectacles, and later on VR Contact Lenses, technologies for both of which are already in post-R&D phase of development. These systems can provide all the visual clues without the cumbersome weight. Experimentation is also already underway on projecting images directly into the retina of the eye using very low power lasers.

Fig. 3.4 Advanced Head Mounted Display

Another fundamental problem with the current generation of headsets is that communication with the VR computer is achieved through direct cable links between the headset and the computer. This limits how far the user can walk. Remote control scanners can replace the need for cabling. Data from the user as to where he is, and data from the computer as to what he should be seeing and feeling, will be delivered using broadband communications networks.

3.4.1.2 Data glove

Trackers are used to find out where the subject is in the real world as shown in Fig. 3.3. This information about the real whereabouts is then used in the virtual world. A receiver is often attached to the HMD. This way, whenever the subject moves his head in the real world the virtual world will change accordingly. A second receiver attached to the Cyber Glove (TM) (**Fig. 3.5**), is used to get the bend of the individual fingers. Inside the gloves there are small pieces of metal that have sensors to notice when the fingers are bent. Through the sensors it is possible to know the angle of each individual joint of the hand. Knowing this information about the different joints makes it possible to recognize gestures made with the hand. One of the most common gestures is the grab gesture. Another common gesture is the fly gesture, a finger pointing in the direction where the user wants to fly.

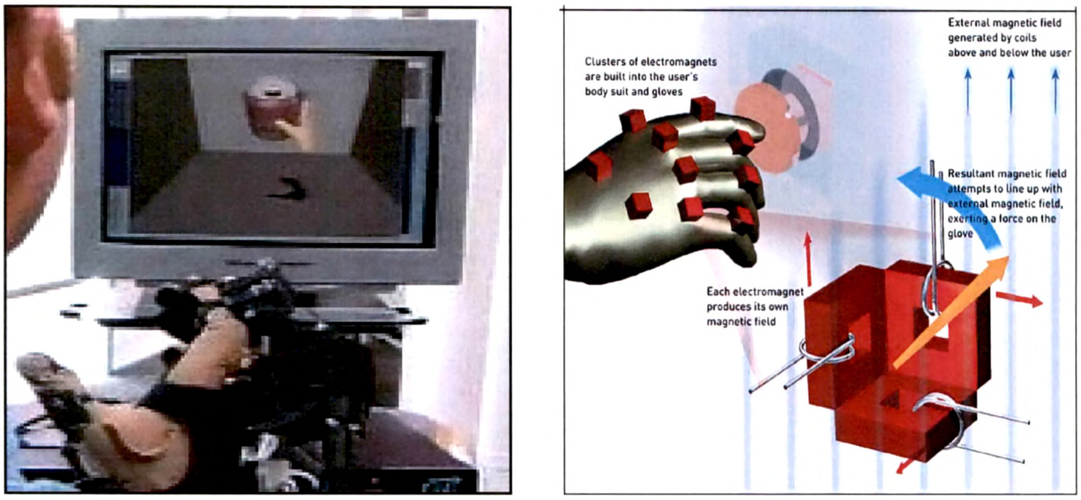


Fig. 3.5 Data Glove and Its Structure

The data glove is put on the hand, and can then be 'seen' as a floating hand in the Virtual Space. It can be used to initiate commands. For example, in Virtual Spaces where gravity does not exist, pointing the glove upwards makes the person appear to fly. Pointing downwards brings him safely back to the ground. Thus the virtual hand acts like a cursor on a standard PC and is able to execute commands by pointing at a particular icon and clicking. The gloves rely on optical fibers to convert the hand movements into signals to the computer. When a human reaches out to grip a virtual object, although the virtual image shows he is gripping an object, the human cannot feel any resistance to the hand tightening movement.The gloves have their limitations as they can be tiring and feel artificial. The future of the data glove as an input device is questionable, though there seems little alternative for sensory output.

3.4.1.3 Body suits

A feel of the virtual reality as close as it is possible, is through a lightweight body suit. This suit has fiber optic cables or motion sensors at the major joints allowing

the VR computers to track the user's movements precisely. The more advanced suits as those used in arcade style VR games, can cost up to $20,000.
In time, the density of sensors, which can be placed about the body, will increase, enabling more accurate portrayals of movement. As with other input devices, enhancements will be made to incorporate output as well. As 'haptic' displays which transmit the sense of touch and force feedback become more prevalent, other physical stimuli may be applicable via such a suit.

3.4.1.4 Sound generation

Virtual worlds are not necessarily silent, so the person can be surrounded by 3-D sounds (**Fig. 3.6**), making the experience of the virtual world all the more convincing. With the fast developments in digital signal processing and sound generation systems in the computers, research is underway to create convincing and realistic three-dimensional sounds at exactly the right moment, for instance when the data glove hits a VR wall.

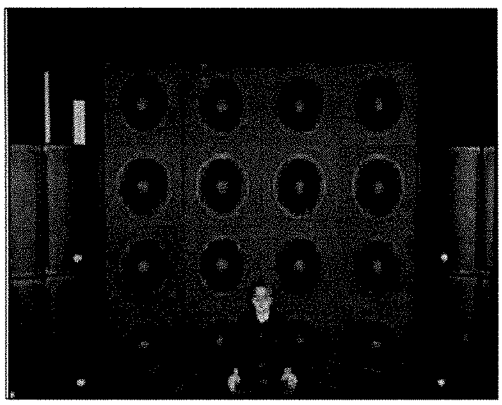


Fig. 3.6 3D Sound Generators

Fig. 3.7 Momi Chair

3.4.1.5 Haptic devices

Virtual reality comprises not only of audio and visual feedback, but it is also possible to push the user whenever he hits an object. This is called force feedback. Using this technique the user can feel the objects in the virtual world and can have an experience as if the object is present in the real world. There are a variety of force feedback devices such as Momi Chair in **Fig. 3.7** that are available in the market.

3.4.2 Non-Immersed Approach

With time, the meaning of VR broadened and, as of today, VR is also being used for semi-immersive systems like large screen projections with or without stereo or table projection systems like the Computer-Aided Virtual Environment (CAVE, **Fig.3.8**) or the Immersadesk and similar devices. Even non-immersive systems, like monitor-based viewing of three-dimensional objects, are called VR systems.

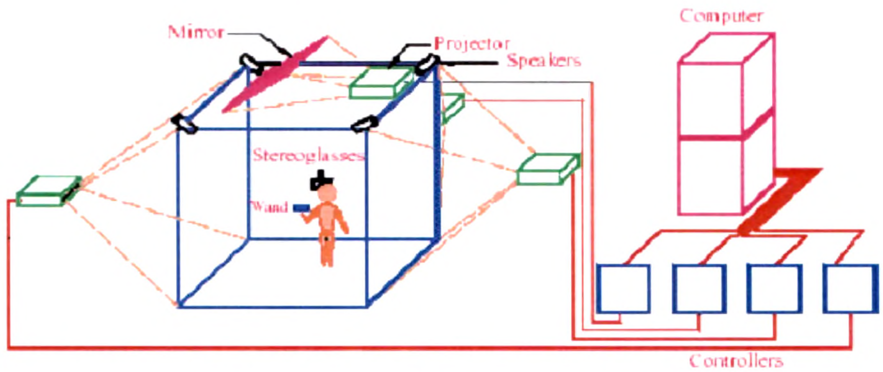


Fig. 3.8 CAVE Projection System

The rapid development of the World Wide Web in the recent past has created additional versions of virtual reality as, for example, in Apple's QuickTimeVR or, more importantly, with VRML, the Virtual Reality Modeling Language for the World Wide Web. The boundaries are becoming blurred, but all variations of VR have gained importance and are slowly merging into a very broad spectrum of technologies used to view and interact with three-dimensional worlds in real-time.

Any design project consists of two basic stages of analysis and synthesis of components. These stages are typical of Computer-Aided-Design process of any structural system. But, when we add a Virtual Reality component as well, the designer becomes an integral part of the CAD process and the design loop. Today's familiar interfaces: the keyboard, mouse, monitor, and GUI, force us to adapt to working within tight, unnatural, two-dimensional constraints. However, VR technologies let the engineer interact with real-time 3D graphics in a more intuitive, natural manner. This approach enhances his ability to understand, analyze, create and design complex dynamical systems. A VR system lets the engineer also experience data directly through the advanced interfaces. He can look and move inside a virtual model or environment, drive through it, lift items, hear things, feel things, and in other ways experience graphical objects and scenes much as can be experienced in the physical world. As a result, VR serves as a problem-solving tool in real time. It is also a cost-effective means for virtually navigating through and designing prototypes of expensive and complex structural systems without actually building them.

The non-immersed approach, enables users to interact with the virtual world with more conventional devices such as a PC and its keyboard, mouse, space ball, or joystick. The absence of haptic devices renders this method devoid of any spatial awareness or sensory inputs as in the real world. But the trade off on the cost, resources and absence of heavy and cumbersome HMD or the body suit is welcomed by most users.

Multimedia, CAD, ICAD, MAYA, 3D Home Architect and 3D Studio Max (**Fig. 3.9**) are some common software for producing the Non-immersed Virtual Reality on the Desktop Applications. A number of softwares are available which provide the facility of "WALK THROUGH" and "WALK EDIT" environments.

Fig. 3.9: Building Details in 3DS Max

3.5 APPLICATIONS OF VR IN CIVIL ENGINEERING

The flexibility offered by Virtual Reality to visualize and interact with the virtual world, provided that these technologies are available at a reasonable cost, will enable designers, clients, and contractors to use the Virtual Environments (VE) to rapidly construct and test their prototypes before starting on the actual project. The application of Virtual Environments offers a high potential to enhance design solutions [77].

3.5.1 Architecture and Construction Industry

By combining 3-D photo realistic images with a navigation mechanism Virtual Environments on the desktop PC can now enable the engineer, to simulate "Walk Through" and "Walk Edit" in the Virtual Space (**Fig. 3.10**) before expensive construction of the physical structure begins. Walking through the Virtual Environment enables designers and/or clients to visualize the design solution, modify it if required, experiment with new ideas, and resolve design or construction related problems at an early stage.

Fig. 3.10 Creating Buildings

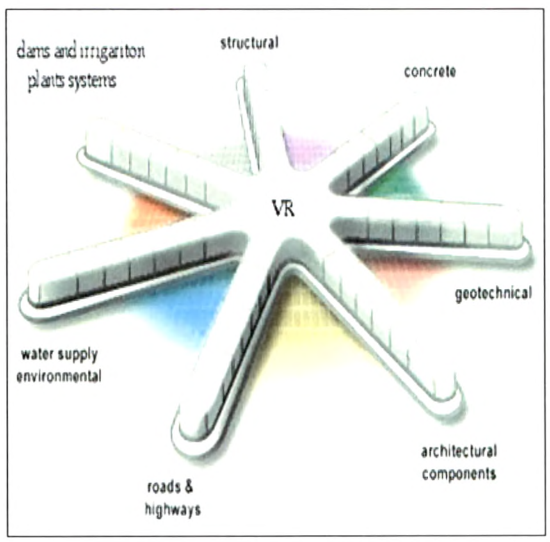


Fig. 3.11 VR Applications

The ability to review the design and rehearse the construction of the facility in 3D interactive and immersive environment can increase the understanding of the design intent, improve the constructability of the project, and minimize changes and abortive work that can be detected prior to the start of construction. Unlimited virtual walkthroughs of the facility can be performed to allow for experiencing, in near-reality sense, what to expect when construction is complete.

The Construction Industry, on the other hand, is an intensive information processing industry and the majority of that information is centered on the project (**Fig. 3.11**). The nature and complexity of construction applications, such as planning, estimating, tendering, ordering etc., make the task of providing and sharing common information very difficult. However, it is widely accepted that vast amounts of project information can be extracted from the drawings using CAD. In order to maximize benefits of CAD, 3D component based systems are required to present information in a form consistent with the needs of users, and must be supported by adequate database and knowledge based systems. In order to present 3D images in a near-to-reality format and supported by databases and knowledge based systems, at real time, powerful graphical techniques are required.

Current CAD packages have improved presentation of the design but their usage is still limited to few images of the designed product. However, they are not interactive, i.e. users are unable to interact with the presented images, and are difficult and expensive to generate. These shortcomings can be easily fulfilled in the interactive 3D Virtual Reality tools and Virtual Environments.

Virtual Reality (VR) and its associated techniques provide an excellent solution to the above problems. Besides its powerful features in presenting the photo-realistic images, it can provide users with an interactive platform whereby 3D images can be linked to and supported by databases, knowledge based systems and design/simulation programs. Texture mapping and dynamic lighting are provided to create more realistic images for enhancing 3-D objects. Moreover, objects can be supported by simulation and design programs, which enable designers to move, rotate, and alter such objects to achieve optimal design.

The emerging VR technology can alleviate the misinterpretation between designers and clients as a complete design solution can be simulated in a VE. There is a whole new opportunity of improving the design solutions by allowing designers to explore alternative designs quickly and efficiently by interacting with design elements in Virtual Space.

Since Civil Engineering systems are always designed to be human and social centric systems, this technology provides an implementation stage of civil engineering projects even before the real system is fully realized. This has tremendous implications in planning, training and project completion cycles of civil engineering. These applications are typically civic related activities and need large investments for effective implementation. Since these are governmental projects, usually based on taxpayer's collections, developing and under-developed countries have not adopted such expensive technologies. Desktop and PC based tools and software are now available at extremely low costs so that building and construction projects are more amenable for VR based applications

3.5.2 Iterative Design

At present, virtual building models are often produced as end products. They are produced to visualize final schemes at a similar point in the process in which rendered architectural walk-throughs would be produced. Good virtual models from CAD data can take many hours of work, and the models produced are not automatically updated when changes are made to the CAD data. Design decisions made after viewing the building in the virtual world should be easily made in both the CAD and VR models, without time-consuming repetitive work.

3.5.3 Site Layout Models

A VE with an expert system can assist project managers in laying out facilities on construction sites. A preliminary layout plan describes the site layout when the project is started with all necessary facilities located, and takes into account the space allocation for major activities that will take place in the near and long term. Construction site layouts generally receive little advanced planning, and almost no planning during construction. The inefficiencies and costs of poorly laid out jobsites, can be averted by user intervention at intermediated stages in a virtual world before commencing the project. The completed expert system will then design a preliminary layout plan, and will provide plans at discrete time intervals. This will allow site management to continuously monitor and update the plan as site activities proceed.

3.6 3DSTUDIO MAX

Integrated Development Environment (IDE) offers a complete graphical environment to develop applications. MAX Studio is an Integrated Development Environment and a product of AutoDesk for 3D MAX commonly known as 3DS MAX [78]. It is optimized for 3D modeling, modifiers, animations and rendering. Max enhances changes and improvements into a formidable interactive 3D graphics and animation field as it has the advanced features incorporated in its organization with many unique features such as MAX Script. MAX allows creation of live 3D worlds with the use of its powerful features like materials, modifiers, lights, cameras and animation.

Its interface offers a simplicity that often belies its powerful capabilities. These capabilities include fast raster graphics technology, excellent 3D modeling tools, standard modifiers, rich material and map library, flexible and easy access of sub-object levels, variety of indispensable objects like cameras, lights, particle objects, space warps, helpers, NURBS, patches, meshes and so on, outstanding rendering techniques, macro recorder and mini-listener and programming control through visual MAX Scripts. Basically MAX is a parametric language which enables change of parameters of any object at any time. MAX is created with 3D designers and animation artists in mind but it is also very user friendly.

3.7 IMPLEMENTATION OF APPLICATIONS IN 3DS MAX

A project in MAX interface, can be created as follows. First user creates the geometry objects by selecting the required category and type in Create Panel, which may be 2D, 3D or any compound object. Geometry can be created by clicking or dragging in the required direction in any of the Top, Left, Front or Perspective viewports. Modify panel is helpful in modifying the assigned dimensions. User can also use various modifiers to obtain desired patterns in the geometry.

If required, sub-object methods are used, which include change of object geometry from within its finite structure like nodes, segments, faces, sides and polygons. Smart tools like array, mirror, quick selection, spacing tool, filter, align, configure, units, snap, pattern selection and the rest make it very competent. Suitable material is selected from the material library and is assigned to selected object's surface. Material is like a sticker on the object's surface which gives "as real" effect in the presence of lights. Lights are set up at appropriate places in the space to obtain the required lighting. The angle of view from which the scene needs to be viewed is created by putting cameras at respective locations. The animation sequence is determined after the above operations are completed. Animation setup is prepared following the predetermined animation sequence. Various special effects like particle features, space warps and environment effects can be supplemented in the scene. User can also add various sounds at his desired locations. The animation output can be obtained as a movie (.AVI, .MOV) by rendering process. Completed MAX file can also be exported to Virtual Reality Machine Language (VRML) environments, .3DS format, ASCII format, .DWG format or PRJ format which makes it even more flexible to compose in the post production stage. The 3DS MAX user interface is shown in **Fig. 3.12**.

Components in 3DS MAX User Interface shown in Fig. 3.12

1	Menu Bar	9	Animattion Playback Controls
2	Window/ crossing Selection Toggle	10	Animate Keying Controls
3	Snap Tools	11	Absolute/ Relative Coordinate Toggle and Coordinate Display
4	Command Panels	12	Prompt Line and Status Bar
5	Object Categories	13	Max Script Mini-Listener
6	Rollout	14	Track Bar
7	Active Viewport	15	Time Slider
8	Viewport Navigation Controls	16	Main Toolbar

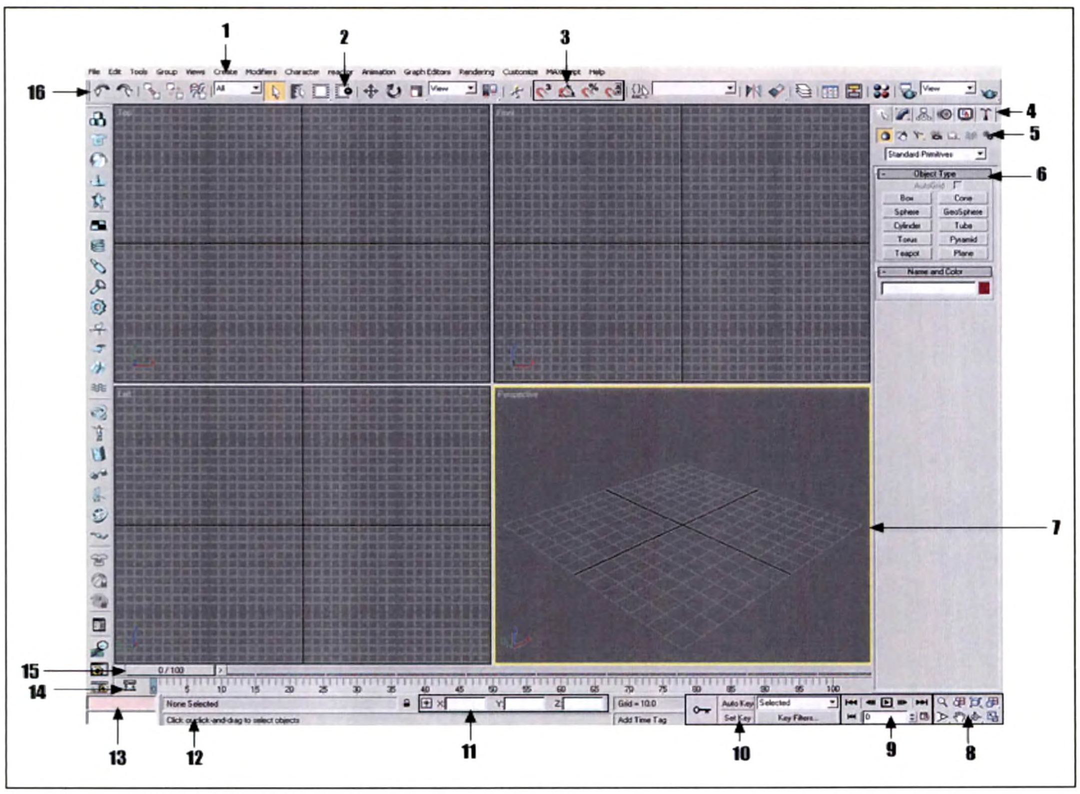


Fig. 3.12 3DS MAX User Interface

3.8 ALTERNATE OPTIONS OF 3DSTUDIO MAX

3.8.1 Flash

FLASH is a product of Macromedia and is an excellent tool for creating 2D animations, web developments and business presentations. Flash includes basic drawing tools, Active Script programming language, timeline and layers. The latest version includes capabilities to include video, sound, complex programming effects as well as database interactivity. User can create simple graphics illustrations using Flash's drawing tools. Flash saves graphics as vector images, as opposed to the typical bitmap type of image.

The advantage of vector-based images is that they are much smaller than bitmaps. This is important in keeping animation files to a reasonable size. Flash can be used to make quick drawings to be saved as GIF files. This can be easier than using a more sophisticated drawing application. Complex animations can be programmed through the use of the Action Script programming language. User can also add sound to the Flash animations or movies. User interactions can be added to control the animations or Flash movies. Buttons can be clicked to perform some task like starting or stopping a movie. Buttons can also be animated on the interaction. Menus are forms of buttons used for Web navigation. Macromedia Flash has a good set of vector-based drawing tools, as well as animation tools. The Flash plug-in is required for users to see the Flash animations.

3.8.2 MAYA

MAYA is a product of Alias for 3D modeling, rendering, animation and paint package that delivers all the tools and features 3D artists need to produce animation and visual effects on the computer desk. It is one of the most powerful and advanced tool for the creation of computer-generated animation and special effects. MAYA is for the advanced digital content creator who wants to work with the ultimate creative tools for maximum creative possibilities. It has unlimited features such as 3D modeler which can create models using polygons, NURBS and other advanced features. The ability to modify and edit objects to the minutest possible detail makes this software every artist's choice. The unlimited bundle of this software has features for character design and powerful animation creation, where one could make the created objects virtually dance and look like real ones. The ability to animate or move individual areas of an object makes MAYA very special. To add visual realism, MAYA also has volumetric lighting and real world atmospheric effects like Fog, Sun, Clouds, Ground, and Sea. The motion builder is one of the important components of MAYA and is used for 3D games development.

3.8 VRML FOR WEB IMPLEMENTATION

Virtual Reality Modeling Language (VRML) is a tool with an enormous potential for building a webspace, just as HTML is used to create a webpage. VRML describes scenes in three dimensions as well as on the time scale. This next generation language is composed of VRML modeler that is optimized to be a sketching tool (not a CAD tool) where user can model his ideas in 4D. The most attractive feature of this tool is that on completion of the work it can be shared with a vast group over the Internet [79].

As web enabled systems come into vogue, it is important to focus on how to shorten the modeling process and, at the same time, achieve the necessary level of interaction. One solution is to combine the sophisticated tools, designed to support the creation of high quality 3D models, with the techniques necessary for achieving interactivity. A combination of 3D Studio MAX and VRML can create web enabled 3D virtual spaces where users sitting in different parts of the world can interact.

VRML is an ASCII-based open, non-proprietary language, which can be used without licensing. International Standardization Organization (ISO) has officially adopted it. The current version of this language supports animation, spatial sound, collision detection and scripting. Virtual Reality is defined by a VRML file, which consists of a multi-tree of nodes. Each node is of a predefined type but new types of nodes can also be created.

When creating VRML worlds, the programmer has three choices to create a model: 1.Visually create and edit the virtual world using specialized tools such as 3DS MAX; 2. Create and edit VRML code with the help of a line editor (such as Notepad); or, 3. A hybrid of methods 1 and 2 which means switching back and forth from visually oriented tools to line editors. The advantage in using the first approach is to minimize modeling time by using the visual environment for creating models. This approach would reduce the time spent in the coding phase of the projects. As a bonus, debugging time would be reduced to a minimum. Files created in 3DS Max can be extended to be used in VRML wherein 3-dimensional space can be created on the web. This space defines the way a user can move in, perceive it and interact with it along with all the add-on features such as lights, camera and sounds.

VRML describes the process of making a virtual world while 3DS MAX is an integrated environment for creating professional quality 3D models. Photo-realistic still images and film-quality animations can be created on a PC using these models. In a 3DS project, the geometry is created as a first step and then texture and materials are mapped on to this geometry, thus giving the model a realistic look. Finally, lights and cameras are added to construct a full scene. Each of these objects can be arranged in various settings and environments. These environments form the basis of the scenes. The characters and objects can be animated and set into motion. This virtual world can then be captured as a film sequence. A scene created in this manner can be used in a variety of ways in animations or as a new Virtual World.

Later, these models, which will constitute a Virtual World, are translated to VRML and viewed using a VRML browser. The browser renders the VRML in real time and turns the code into perceivable space with which we can interact. 3DS MAX supports the modeling process in two ways: 1. With a modeling environment capable of producing high quality 3D models and 2. With VRML helper objects. Without helper objects interaction is limited to looking at objects from different angles. Integration of sound, touch and other spatial perceptions can be accomplished only by using helper objects, which significantly extend interactivity in the VRML environment.

3.9 INTERACTIVITY IN A VIRTUAL WORLD

Interactivity in Virtual Reality worlds traditionally means employment of three of our five senses: sight, hearing and touch. Combining targeted impressions on these senses produce perceptions of space and interaction with objects located in that space. For example, incremental enlargement of an object produces a perception of movement towards that object. What a user expects to do in a virtual world is to move freely about, manipulate objects as one does in the real world and experience a spatial sense of sound. A spatial sense of sound means that the sound has a source fixed at a single point (a node) in the virtual environment. Moving towards that source increases the volume of sound and moving away decreases the volume. Also, the perception of the source of the sound is experienced relative to the user's position in the world. By shifting position in the virtual environment, the user's perception of the direction of the sound source changes. User interaction in VRML world is gained by using standard predefined VRML sensors. Sensors can be considered as special kind of nodes designed to react when properties of the Virtual Environment change or when a user operates a sensor in a predefined way.

3.10 VRML HELPER OBJECTS IN 3DS MAX

Interaction in a VRML world using 3DS MAX, is achieved by adding VRML helper objects. During export, the helper objects become VRML nodes. The following helper objects are predefined in 3DS MAX (**Fig. 3.13**):

- Anchor : Creates a link in a VRML file. The link is embedded in an object;
- TouchSensor : (Touch Sensor) Represents an area of sensitive space that, when touched by user, triggers event.
- Sound and Audioclip : These two helpers allow the insertion of spatial or ambient sounds in a scene. Sound helper is always used in combination with Audio clip, while Audio clip can be combined with TouchSensor.
- ProxSensor : (Proximity Sensor) Represents an area of sensitive space that, when activated by the approach of user, triggers an event.
- TimeSensor : It is used for adding time-based animation controls - such as the start and end frames for the animation of a particular object.

- NavInfo : (Navigational Information) Provides navigational information of virtual space. This helper directly influences the way a user moves around the virtual world. Every scene should have this helper included.
- Background : The Background helper allows the creation of a Sky and/or Ground backdrop for the virtual world, which produces very simple and plain "Earth and sky" perception. Use of different types of objects and modifiers can produce the effect of ground elevation.

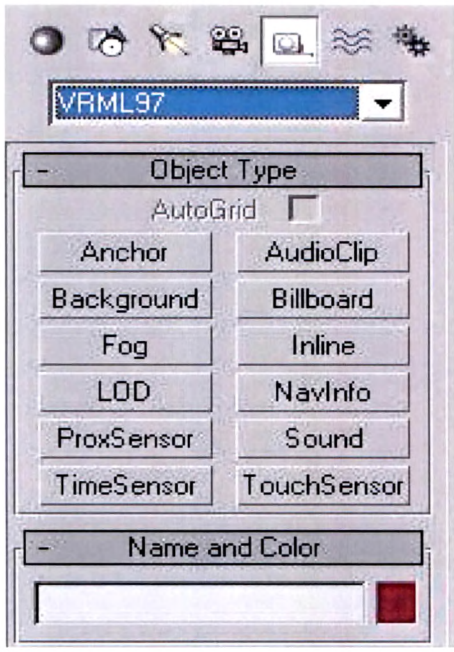


Fig. 3.13 VRML97 Helpers

3.11 EXPORTING 3D SCENES FROM 3DS MAX TO VRML

Once a scene is created, it is ready to be exported to VRML environment. 3DS MAX has a built-in VRML exporter. In version 5, 3DS MAX creates VRML files (.wrl) that are compatible with the VRML97 standard. As explained above, a 3DS MAX file can be exported to VRML environment with the use of VRML helper objects and MAX export command options. The detailing file prepared in MAX is exported to VRML environment without beams, columns and infill walls so as to get complete idea of the detailing (**Fig. 3.14**) on this platform. As the number of objects in the detailing file is very large they are exported with the optimum details. User can operate the file in a platform independent interactive 3D environment, which is the best advantage of using VRML files.

3.12 VISUALIZING VRML FILE IN COSMO PLAYER

The Cosmo Player plug-in is a popular VRML browser, which allows application developers to add depth, lighting and motion to web content based entirely on cross platform and open standards. With the use of VRML language and Cosmo Player it is relatively easy to visualize data, navigate databases, move inside structures and share 3D information. It uses OpenGL or Direct3D to take advantage of any hardware graphics acceleration. The screenshot in **Fig. 3.15** shows how a model appears within a VRML viewer (Cosmo Player). The viewer renders a three-dimensional representation of the model (as defined within the VRML file) and provides tools at the bottom for moving about this scene. One can move in or out and around and through the objects.

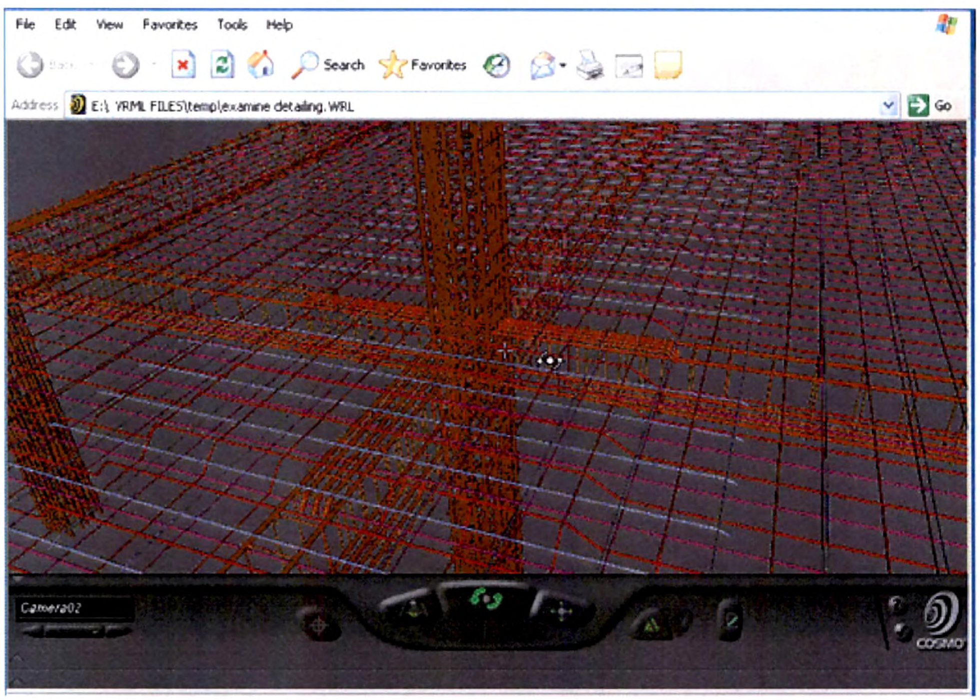


Fig. 3.14 Application on VRML Platform of COSMO

VRML gives output as .wrl files which are opened in the Cosmo Player. After installing the Cosmo Player in the system, user can open the .wrl files in the Internet explorer where Cosmo Player opens by default. Dragging the mouse across the globe will make the molecule rotate. The thumbwheel moves the molecule closer or farther away. The diamond box to the right allows for translation in the horizontal and vertical directions. Clicking the target button on any object makes that part of the object to move to the center of the view screen. User can change the views of the world space depending on the number of cameras arranged in the scene and displayed on the left hand button.

3.13 INTERFACING OF VC++, SAP2000, 3DS MAX AND VRML

It is clear from the previous paragraphs that VC++, 3DS Max and VR are powerful tools, each of which enables a professional level look for an application program. In the current work each of these has been used to its capacity to create a software that delivers a powerful tool to the structural engineer who had been relegated to using commercial software with either limited choices or at times lengthy outputs leaving the user with so much data and information that it takes an equal amount of time to conclude and discriminate which part of the output is relevant and needs to be retained for the particular job and which may be discarded. Besides the conventional approach, the software prepared here enables access to 3D environment in the structural domain itself, thus opening new vistas of interacting with a structure for various alternatives.

In summary, by interfacing the three powerful softwares of VC++, SAP2000 and 3DS Max in their own domains, the limited options in structural engineering can be vastly expanded and put to good use, especially in understanding the complex nature of forces due to hazards which can turn into disasters purely due to structural failures. Finally the 3DS Max version can be transformed onto a VRML platform to bring in the Virtual Reality aspect and can be simultaneously web-enabled.