The integration of immersive Virtual Reality applications into Catia V5

D. Rantzau*, F. Maurer**, C. Mayer, R. Löffler, O. Riedel, H. Scharm, D. Banek

SGI Professional Services, Practice Immersive Solutions, Germany
[*dirkrt@sgi.com, **franzm@sgi.com]

Abstract. During the last few years a wide range of stand-alone virtual reality (VR) applications have been developed. The typical data source for most of these applications are 3D styling tools or 3D-CAD systems from which data has to be converted into a representation useful in VR software systems. The drawbacks of the stand-alone approach are well known: data conversion, loss of semantics and the problems of establishing a feedback path for updating data in the underlying CAD model from within the VR software. In this paper we will introduce our solution to this problem by proposing an integrated VR/CAD system approach based on the CATIA 3D-CAD software, more precisely the CATIA V5 application architecture [1]. We will discuss the requirements and implications from the CAD data perspective and show how the VR integration inside the CAD system can be realized. Derived from a specific end user scenario we introduce a sample application for virtual prototype reviews.

1 Introduction

Immersive Projection Technology (IPT) is getting more and more widespread in industry. Multi-wall stereo projection systems (well known as CAVE^{TM}) recently became manageable because new and powerful computer hardware technology is available and companies bringing the technology to the market are more focusing on end user solutions than simply providing raw technology.

On the software side the term Digital Mock-Up (DMU) has appeared recently, targeting at a new and innovative product development. DMU, in a broader sense, is a complete digital product description for development, design and manufacturing. One of the major data sources for DMU is 3D-CAD systems. The integration of VR technology inside existing CAD systems is desirable especially from the end user perspective: less software packages pose fewer problems with data handling and mastering different software systems. Moreover, as additional DMU functionality is becoming more and more available in standard CAD software packages, this functionality can be accessed from within the integrated “VR front-end” also.

In the following we will describe our approach taken towards the realization of an integrated VR/CAD system based CATIA V5. In the first section we begin to describe the current situation of VR software technology for engineering from the data perspective. From this we derive the requirements for an integrated system approach. We then shortly describe the component based architecture of CATIA V5. After this the integration steps of VR modules into the CATIA CAA framework is described. We then introduce a virtual prototype review application, which has been derived from a specific end user scenario.

2 State of VR Technology for CAD/DMU

Today users can choose from a lot of VR software systems exploiting different concepts such as the toolkit or framework approach found in Lightning [2], VirtualDesign2 [3], VR Juggler [4], Avango [5] or application focused systems such as EAI VisConcept [6] or Opus Realizer [7]. An interesting example of a collaborative VR system with CAD functionality based on scene graph sharing has been described by Smith et al. in [8].

However, from recent user studies [9] it is known that a number of problems still lie in the immaturity of the available software. We believe that one of the main reasons for this discovered "immaturity" lies in the stand-alone approach which is directly related to the problems of data transfer and data management. The data interfaces used today are primarily based on reading files (neutral or native formats) coming from the engineering applications, i.e. the 3D-CAD system or a styling tool.

3D-CAD systems for example generate continuous surface and volume descriptions often referred to as free form surfaces and solids. VR rendering systems deal with polygons or occasionally with voxels. The process of converting the CAD data to a polygonal representation is called tessellation. For all major CAD formats including the neutral formats IGES, VDAFS and STEP tessellators exist. Unfortunately, different problems related to this conversion process occur, which pose a burden to end users wanting to adopt VR technology in their daily productive work:

- Insufficient model quality (inconsistent normal directions, cracks by not considering topology, level of detail (LOD's) too coarse),
- unneeded model complexity (wrong SAG values, improper algorithms, tessellation of invisible details, LOD's too fine),
- poor render performance (poor culling structure, missing LOD's, unneeded model complexity),
- missing data structure (loss of logical structure, trade-off to culling structure),
- typical CAD conversion problems with neutral formats like IGES or VDAFS (least common denominator),
- poor interoperability with legacy systems, e.g. integration between VR and deployed product data management systems,
- the need for re-invention of functionality in VR software systems which is actually already available in CAD systems.

The following figure shows an abstract view of a CAD system (left) consisting of an application layer (top), and an API layer for accessing base functionality (middle) found in the graphics and kernel layers (bottom). When looking at the VR system on the right, it is interesting to note that similar layers can be discovered there. In most systems application functionality is encapsulated in one layer e.g. as modules (compiled or scriptable) which access other components through the API layer. In both cases there is a graphics component, often based on a scene graph API such as OpenGL Performer. The CAD and VR kernel are the glue between all the components. Any CAD like functionality such as measurement of distances between two points have to be re-implemented in the VR software, often with the drawback of losing a certain degree of precision due to the different data representations in each system.

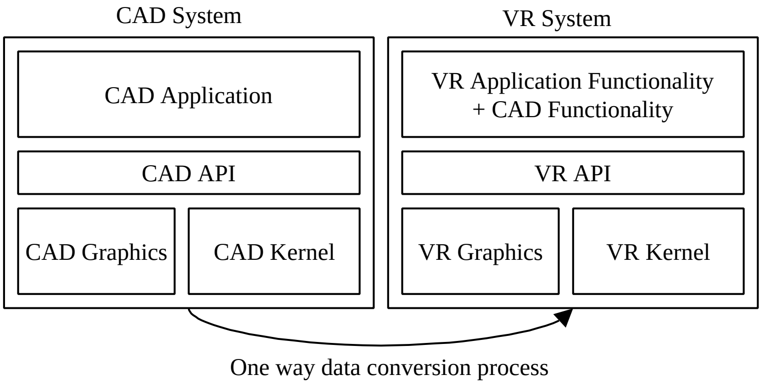

Fig. 1. Layers in CAD and VR systems

From this data perspective the idea of melting both systems into one is self evident, however several requirements have to be satisfied to provide the correct infrastructure for realizing immersive applications:

- The CAD graphics system has to be tuned for high performance rendering and should support multiple synchronized graphics pipes/channels for driving IPT systems such as CAVE's and Powerwalls.
- Support for 3D input devices needs to be added to the CAD system.
- A 3D user interface should replace or augment the traditional 2D menu driven approach when switching to the "immersive mode" using IPT.
- Effective user interaction metaphors have to be implemented to augment or extend standard CAD functionality depending on the specific end-user application needs.

While the broadly deployed CATIA CAD software in the version 4.x isn't able to meet the requirements above, the CATIA V5 software by Dassault Systèmes fulfills some of the needs already or at least provides a framework in which missing VR extensions can be implemented efficiently.

Preliminary CATIA V5 work describing the integration of a generic VR device support targeting a glove based input with BOOM based output have been proposed by Fakespace in [10] but no productive application had been realized at that time. In the following we describe our integration approach and show how this can be leveraged from applications targeting actual end user scenarios respectively specific market segments.

3 The Catia CAA architecture

Unlike traditional CAD legacy systems, which have FORTRAN based kernels reaching back 20 years ago, the CATIA V5 CAA (Component Application Architecture) has been designed using latest software design methodology. The final CAD product consists of a number of cooperating component frameworks. They can easily be extended or completely new frameworks can be added to the system. The key benefits for VR applications on top of CAA V5 are a completely object oriented C++ API and a high performance OpenGL rendering kernel.

Two kinds of CAA applications can be created: stand-alone programs which use only the CAA functionality, or CAA commands which are plugged in the runtime environment of Catia. CAA commands are encapsulated as shared objects with well known interfaces. During the program start-up phase, Catia looks in so called dictionary files for the commands that are available. This dictionary provides the link between the command and the corresponding shared object. In order to provide a technique for grouping commands of an application, Catia has the concept of Workbenches. Workbenches in the CAA context are assemblies of menu items, toolbars and commands which all belong to a certain application. After the Catia start-up the user can activate the Workbench he needs and then has access to all functions provided by the application. These concepts allow a seamless integration of immersive applications from the end user point of view.

4 Bringing VR inside the CAD system

One of the main prerequisites for integrating VR applications into a CAD software system is the capability of the rendering engine to perform interactive “realtime” visualization. One of the first steps of this work was to examine the underlying CATIA V5 system architecture in terms of event latency and renderer performance. In contrast to its predecessor CATIA V4, the renderer of the CATIA V5 CAD software is completely based on the platform independent graphics API OpenGL. Most of the graphical capabilities of V5 is located in the visualization framework which comprises the main part of the visualization model of the CATIA system architecture. Dassault Systèmes has developed a high-level graphics API, which uses a small but very efficient scene graph. Application developers can model a visualization database by composing elements such as group nodes, representation nodes (Rep), graphical primitives (GP), graphical attribute sets and so on (see also fig. 2). The scene graph can then be added to an active 3D viewer class, which is solely responsible for the actual rendering process. The viewer also manages viewpoints, lights, background colors and so forth. Furthermore the CATIA visualization provides important VR features such as:

- culling algorithms (including back face as well as occlusion)
- level-of-detail (LOD) algorithms (depth and motion)
- tessellation algorithms (the tessellation quality is adjustable by the user)

After testing the rendering capabilities of the CATIA software we came to the conclusion that V5 offers all the functionality necessary for immersive visualization. Especially the integration of the OpenGL Multipipe™ SDK [11] by SGI will enable the rendering system to generate multi-channel, multi-pipe display output. Based on these features it is possible to support every current known display configuration, including CAVE™s.

CATIA is using the Model-View-Controller (MVC) paradigm and distinguishes between the application and the visualization model. Both models are loosely coupled. The application model contains and manages all the CAD data base specific entities like the parametric information of the geometric objects or the information on how a certain part has been designed (e.g. by using sketching, filleting, chamfering or other topological operations like the subtraction of two geometries). Each visible element of the application model has its surrogate in the visualization model. Thus, based on the user settings, each visible geometry of the application model has its tessellated representation in the visualization model. The following fig. 2 sketches a rough scheme of the CATIA V5 visualization process.

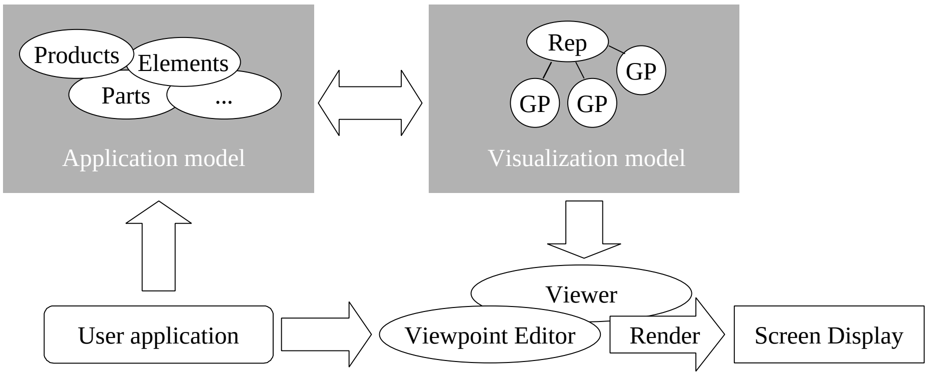

Fig. 2. The simplified visualization process in CATIA V5 using the application model and visualization model

From the VR application programmer’s point of view, the separation of the application and visualization model creates the freedom to make a distinction between mechanisms that require persistent storage and operations that are only necessary during the lifetime of an “immersive session”. Programmers are free to use both of the model contexts from above. In case of our first application described later, the so called “Immersive Solutions Review on CAA V5”, all the user interface elements such as the 3D menu and 3D pointers are entities that reside in the visualization model only –they do not have any meaning outside the VR session. On the other hand, the annotations made during a collaborative immersive session are stored persistently in the application model and they also do have a graphical representation to indicate their position in the design model. After the immersive session has ended the user can decide whether to store the changes and additions in the same CATIA document or to use the “save as” operation of the desktop view to generate a new one.

To be able to re-use as many common VR functionality as possible, we have decided to use a layered approach in which several VR applications encapsulated as modules and sit on top of a common VR API layer. Each of these application modules is targeted towards a specific end user task or market segment like e.g. a virtual prototype review (fig. 3).

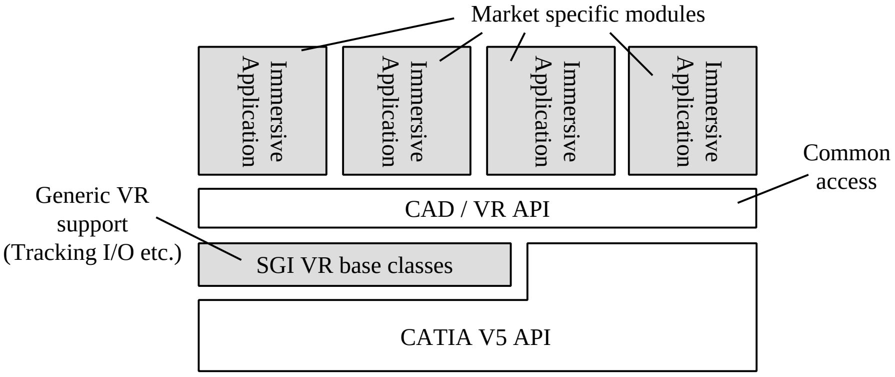

Fig. 3. Immersive applications on top of the Catia V5 CAA API

A framework with several VR base classes has been added to the system to enable easy VR application development on top of CATIA V5. These base classes offer a set of functionality that is needed in nearly every immersive application (though are not mandatory to use) such as the support for the most common 3D devices, well known 3D navigation modes like fly and examine mode as well as 3D selection. 3D selection is implemented with a newly designed 3D menu called Immunipa (fig. 4), which is very intuitively accessible and configurable from within any immersive application.

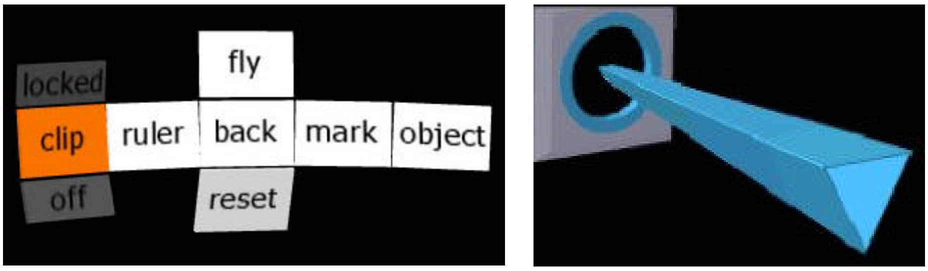

Fig. 4. Immunipa 3D menu (left), 3D selection pointer (right)

The interaction behavior inside the application is implemented as a state machine that reacts to the incoming events from the 3D-tracking or button devices. A default behavior is already realized in the base framework. This behavior can be extended or overwritten by the application.

Finally the application consists of two parts: the device daemons and the application module that is derived from the classes of the base framework. The daemons read the data of the external devices and send them as events to the VR application. An event dispatcher gets this data in the form of a small number of integer and float values and calculates the new state of interaction or triggers some action (see fig. 5). Actions can access any available function callable inside the CAD core system. This of course paves the way to implement those features that would otherwise hard or impossible to implement in a stand-alone VR system.

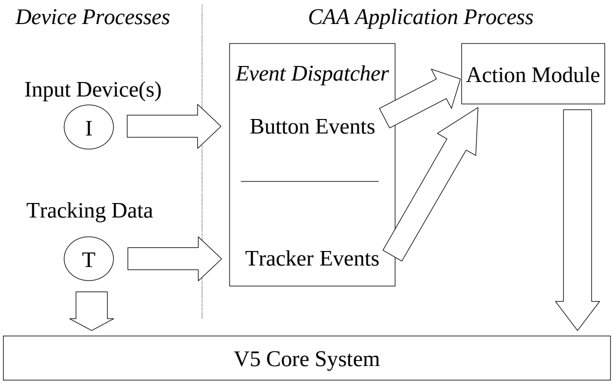

Fig. 5. Device communication between deamon and modules

From the end user perspective, the available immersive applications are accessible by the "add-in" toolbar in the DMU Navigator package. This add-in provides some commands to configure the VR session and of course one to launch the immersive mode. It is important to note that when the VR application is launched all the properties which were setup on the desktop are still valid. Vice versa, all the changes made in the VR context (like annotations) are still visible on the desktop and can also be stored as a part of the CATIA product structure for later analysis. This is what can be called the two-way communication between the DMU view of the model and the underlying CAD representation.

5 Sample application: Immersive Solutions Review on CAA V5

One of the most important lessons we have learned in the past as an immersive solutions provider is that the investment in IPT installation have the highest return-on-investment when it is possible to map existing work processes into the immersive environment as efficiently as possible. We have therefore chosen to support in the first application a widespread and proven work process in design departments, the so called DMU or virtual prototype review. The advantages of VR combined with IPT as a communication medium and tool matches perfectly with the collaborative DMU review work process.

A typically scenario involving an immersive DMU review process with the Immersive Solutions Review application could be as follows:

> “A development team of different engineers is working on the design of a functional unit of a new car. Most of the team members have to interface with each other and therefore it is necessary to have regular team meetings to discuss the state of the work. Such team sessions typically include reviews based on a 3D visualization of the assembled digital mock-up consisting of the different parts provided by the team members. The team leader is therefore querying the product data management system (or the file server) and merges the latest design parts into an assembly. Since the new assembly gets to big in data size to render, he decides to reduce the tesselation quality using the CATIA product settings and to turn the material display off. One of the tasks for the session includes a comparison of two sub units, and therefore he puts one of the variants into “no show”.
>
> The meeting starts by using the immersive CAD option that is initialized by clicking on a button in the CATIA workbench which switches between the team leader’s desktop console to the real-time visualization using a Powerwall in 1:1 scale mode. Soon after the group discussion begins, several design defects are encountered and improvements are discussed at the Powerwall. With the 3D-annotation tool the session moderator marks critical regions of interest with his 3D-joystick. Later on, group members will find the annotations as a part of the product structure so that they are able to work through the annotation list to improve the design for the next development cycle. An agreement within the team has been found, that the second discussed variant fits better to the functional requirements but a problem for the assembly process is discovered. The necessary space to house the units during production won’t be sufficient. With the 3D-measurement tool the moderator is able to exactly ascertain the space between the critical units rapidly and it soon becomes obvious that another development team is concerned with the effects of the necessary changes. As a result of the session, team members of the other development team are informed and invited as well as member of the product management team to join the next visualization session in order to find a common decision how to circumvent the discovered problem.”

Although the scenario described above is an abbreviated description of the much more complex complete DMU work process, it should become clear, how the immersive CAD application can support a simultaneous engineering approach in the car design departments. The following figures show two examples of the functionality available to the end user.

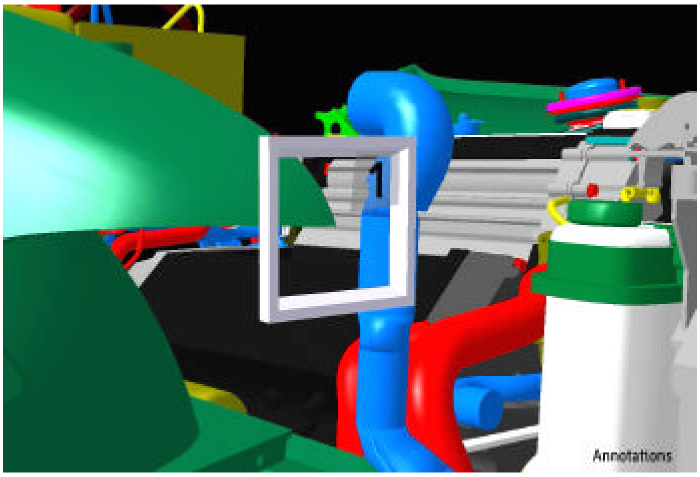

Fig. 6. Annotation feature

Fig. 6 shows the annotation feature as (numbered) markers which can be placed anywhere in the virtual model to mark the regions of interest.

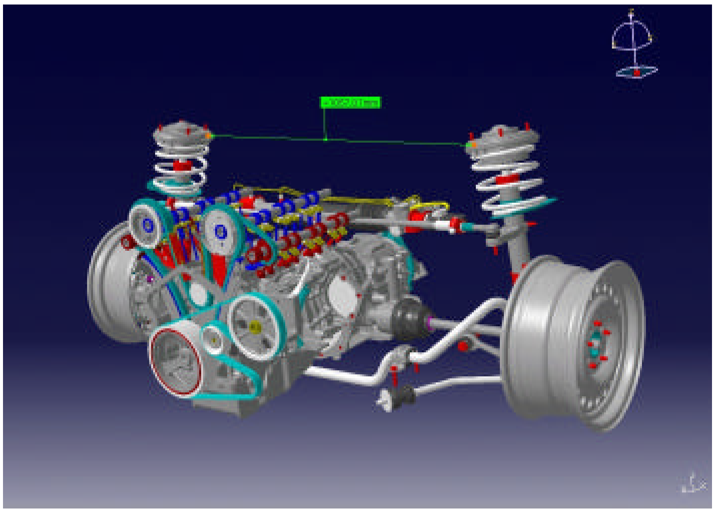

Fig. 7. Distance measurement feature

In fig. 7 the straight line represents the distance between two points on the outer car body which is shown numerically above the line.

The Immersive Solutions Review application is a good example of how a simple VR interface allowing a lot of user control can expand the scope of a traditional CAD system towards a group reviewing tool. When using a 1:1 scale visualization facility like the Powerwall, 3D navigation and annotation features can be effectively used to review the DMU models and virtual prototypes facilitating decision processes across development teams. From the functionality side we are still in an “alpha” phase where we gather valuable input from end users from the manufacturing industry to adjust the user interface to the actual work processes.

6 Conclusion

In this paper we have introduced an example of an immersive application inside a standard 3D-CAD package. Our approach of focusing on specific end user scenarios instead of developing “the” VR front-end of a CAD system reflects the results of what we have learned during the last years as an immersive solutions provider. Immersive projection technology will continue to play a significant role as the visualization technology of complex and design oriented products for the years to come. We believe that the trend towards the better integration of VR functionality into existing CAD/CAE packages is an excellent mechanism for bringing VR and IPT to a wider audience.

Acknowledgements

We’d like to thank all our partners and colleagues at Dassault Systèmes and SGI for their enthusiasm and effort bringing VR to life inside the CAD system CATIA V5, especially our friend Bob Kuehne for his valuable input.

References

1. URL: http://www.catia.com
2. Blach R., Landauer J., Simon A., Rösch A.: “A highly flexible virtual reality system”, in: Future Generation Computer Systems 14 (1998), Elsevier Science, pp. 167-177
3. URL: http://www.vrcom.de
4. Just C., Bierbaum A., Baker A., Cruz-Neira C.: “VR Juggler: A Framework for Virtual Reality Development”, Proc. of the IPT 98 Workshop, Ames, Iowa, 1998
5. Tramberend, H.: “Avocado: A Distributed Virtual Reality Framework”, Proc. of the IEEE Virtual Reality Conference 99, Houston, Texas, March 13-17, 1999
6. URL: http://www.eai.com
7. URL: http://www.opticore.com
8. Smith, R. et al: “Collaborative Visual Eyes”, Proc. of the IPT 2000 Workshop, Ames Iowa, 2000
9. Delainey, B.: “The Market for Visual Simulation/Virtual Reality Systems”, CyberEdge Information Services Study, Spring 1999
10. Berta, J: “Integrating VR and CAD”, IEEE Computer Graphics and Applications, Vol. 19, 1999
11. OpenGL Multipipe SDK. URL: http://www.sgi.com/software/multipipe/sdk/index.html