

# Virtual reality approaches for immersive design

D. Weidlich · L. Cser · T. Polzin · D. Cristiano ·  
H. Zickner

Received: 23 February 2009 / Accepted: 6 March 2009 / Published online: 26 March 2009  
© Springer-Verlag 2009

**Abstract** The current application of virtual reality (VR) systems in the design process is limited mostly to design review. The reason for this limitation is the different data formats used for CAD and VR visualization. To use the benefits of VR during the design process, solutions for immersive design, the model manipulation inside the VE based on CAD data, are required. There are different approaches allowing VR systems to work as an active development platform. Three examples introduce the realization of the integration of CAD and VR software at different levels by the online coupling of complete applications or by integration of CAD core functionalities in VR systems.

**Keywords** Virtual reality · 3D interaction · Design · Product development

## 1 Introduction

The increasing use of digital product design methods, especially of computer-aided design (CAD) and computer-aided engineering (CAE), have reduced design times. The user interface is usually desktop-based and therefore models are not commonly represented in their real size. A large amount of modeling time is used for navigation and object selection.

This particularly applies to today's 3D CAD systems. While these systems have a great number of component and module modeling and assembling functions, much time in addition to navigation and selection is required to draft and parameterize geometry models.

Virtual reality (VR) applications have been in active use in the automobile and aerospace industries as well as in production engineering since the 1990s. Today's VR technologies offer enormous potential for improving the comprehensibility of design and simulation data, which implies declining error rates [1]. Potential sources of errors, especially during the outline and detailing phases, can be targeted and excluded by the immersive experience of geometry and topology data. Therefore this article focuses on integrating VR as a user interface into the process of geometric modeling and detailing. It shows three paths towards a solution: VRax<sup>®</sup>, navigation interface for modeling (NavIMode), and Construct|Tools.

## 2 State of the art

Researchers have implemented three prototypical ways of immersive modeling: linking a VR system and a CAD core, linking a VR system and a CAD system, and using voxel models for geometry description.

Examples of linking of VR-CAD core are the advanced realism CAD environment (ARCADE) [2] and Constructed [3]. Both systems are based on the VR system "Studierstube", and both systems use the ACIS core modeler for describing geometries. ARCADE is a cooperative, directly manipulative 3D modeling system that takes a user-centered approach. An important interaction feature of ARCADE is the topological context based limited modification technique for transforming objects. The construct3D framework uses augmented

---

D. Weidlich · D. Cristiano · H. Zickner (✉)  
Institute for Machine Tools and Production Processes,  
Chemnitz University of Technology, Chemnitz, Germany  
e-mail: holger.zickner@mb.tu-chemnitz.de

D. Weidlich · T. Polzin  
Fraunhofer Institute for Machine Tools and Forming Technology  
IWU, Chemnitz, Germany

L. Cser  
Corvinus University of Budapest, Budapest, Hungary

reality as a user interface and provides functions for creating volume primitives and for modification using Boolean operations. The Spacedesign system [4] presented in 2002 is an example of a system for immersive curve and surface modeling. Spacedesign enables the creation and modification of 3D curves and surfaces in VR or AR environments.

Another approach is voxel-based modeling as applied, for example, in virtual clay modeling (VCM). VCM with haptic feedback is presented in [5]. An interaction is implemented using tool-like interactive devices. They allow virtual modeling by adding and removing. The virtual assembly design environment (VADE) [6] system was the first that linked a complete CAD system with a VR environment. The VR environment used was an head mounted display (HMD). VADE was developed for assembly simulation and planning and facilitates the creation of alternative assembly strategies for components. Other relevant works on multimodal input using additional 2D input devices are described in [7, 8]. Sketches can be created and functions activated in the VR using a personal data assistant (PDA).

### 3 VRAX<sup>®</sup>: VR based development platform

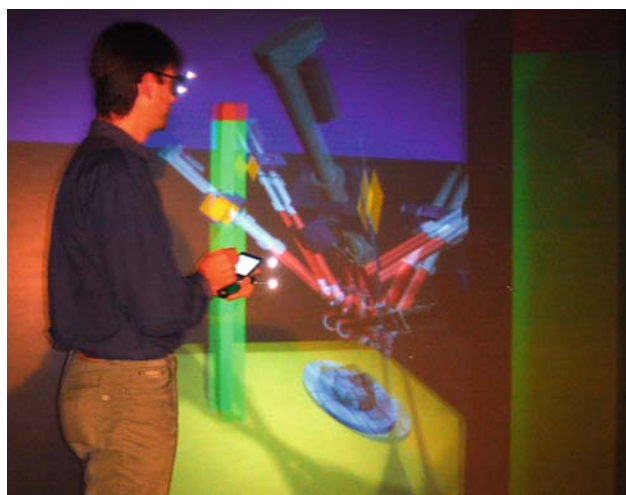
#### 3.1 The VRAX<sup>®</sup> method

VRAX<sup>®</sup> [9] denotes a platform for designing machine tools that are analogous to existing CAX tools based on VR technologies. VR technology is used as an active development and design medium in this process, and data created in VR are recirculated into the overall development process. Variants of machine tools with parallel kinematics can readily be created by combining a modular approach with free modeling functionality.

The basis is a workflow for technology-optimized machine tool design. Templates are provided for each phase of the workflow. A special template called the machine template provides the basis for the parallel kinematic structure to be developed. It contains all required information in a rough plan structure, e.g., the number and arrangement of joints. Generic algorithms determine the kinematic dimensions such as the positions of joints, strut lengths, etc. from the stiffness and working space requirements. These parameters subsequently initialize the machine template. Further assembly is supported by interaction metaphors such as snap-in for positioning. The machine elements are automatically positioned and aligned according to their function within the kinematic chain.

#### 3.2 VRAX<sup>®</sup>: immersive modeling

The AVALON VR system was linked to the OpenCascade free CAD core to implement immersive modeling within the



**Fig. 1** Using a PDA for sketches

workflow. This allows the implementation of special modeling functions such as the extrusion of cross sections along B splines. The course of the extrusion path can be changed interactively by manipulating specific check points. The extrusion target can be defined in advance using area selection. These areas are decisive interfaces between the joints and the frame as well as the frame and the working platform. An additional 2D interactive element in the form of a PDA is used in the VRAX<sup>®</sup> system to create the cross section and enter discrete parameters (Fig. 1).

Volume primitives are provided for the rough design of the frame and working platform, which are frequently welded. The geometries generated are transferred to, and used in, conventional CAD systems using the STEP standard. This neutral data format guarantees compatibility with most CAD systems.

#### 3.3 System architecture

The system architecture of the VRAX<sup>®</sup> system was designed according to a three-layer model [10]. A VR system forms the visualization and interaction layer. The second layer is the application core which contains the machine tool model and the processing control unit. The machine tool model is responsible for synthesizing all geometry (polygonal models, CAD volume models, partial models) and metadata. The processing control unit is the computer-internal representation of the workflow. Based on a digital product data model for machine tools with parallel kinematics, the design, simulation and calculation systems required during the development process (CAD, FEM, MBS, calculation algorithms for structure optimization, databases, ...) are integrated in the third tier via a product data management system (PDM system). Abstract model descriptions and their parameters are stored in form of a model matrix rather than storing concrete

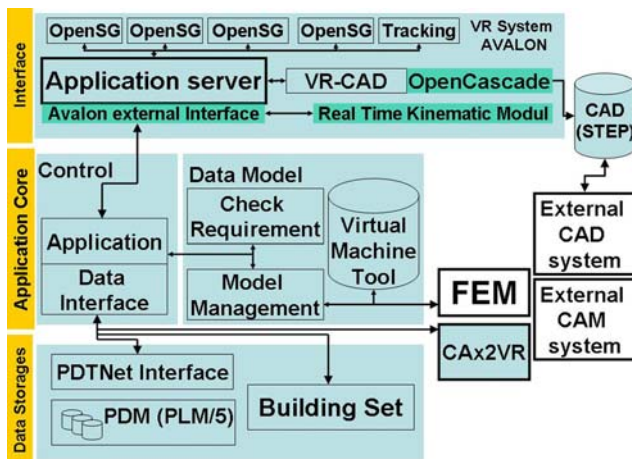


Fig. 2 System architecture of the VRax® system

models. A VR system that is linked with the given design and simulation systems operates as an interface between the users and the digital data model. The modular design makes it possible to make interactive changes in the VR system and feed them back into the 3D CAD system (Fig. 2).

#### 4 NavIMode: integrated VR-CAD environment

##### 4.1 Motivation

An integrated VR-CAD environment for designing complex modules is created by linking a VR system and a CAD system. The selection, navigation, and transformation tasks are strictly separated from the modeling functions. The goal is to make the exploration of complex modules simpler and more intuitive. The designer keeps all CAD functions.

##### 4.2 NavIMode: new approaches to interaction

NavIMode is an interface between CAD systems and VR systems. A new interactive device is configured using a freely movable tablet PC. In addition to the usual functions of CAD systems, the user can find his or her way visually in the draft design. A special function, the virtual loupe, is provided for this purpose (Fig. 3).

The display visualizes the virtual model according to its position and orientation, and the user can look into the virtual world as if looking through a physical frame. This function is implemented by the application server by means of analyzing tracking data (Fig. 4).

The challenge is to assign projection parameters of the VR (camera position, angle of aperture) to the respective CAD system parameters such as zoom factor, translation, and orientation in real time.



Fig. 3 Virtual loupe

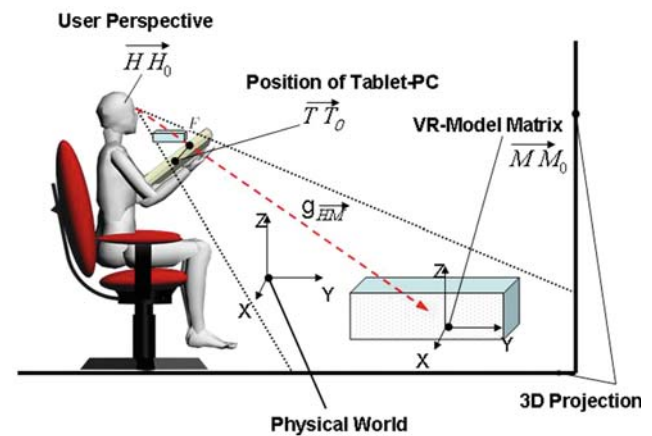


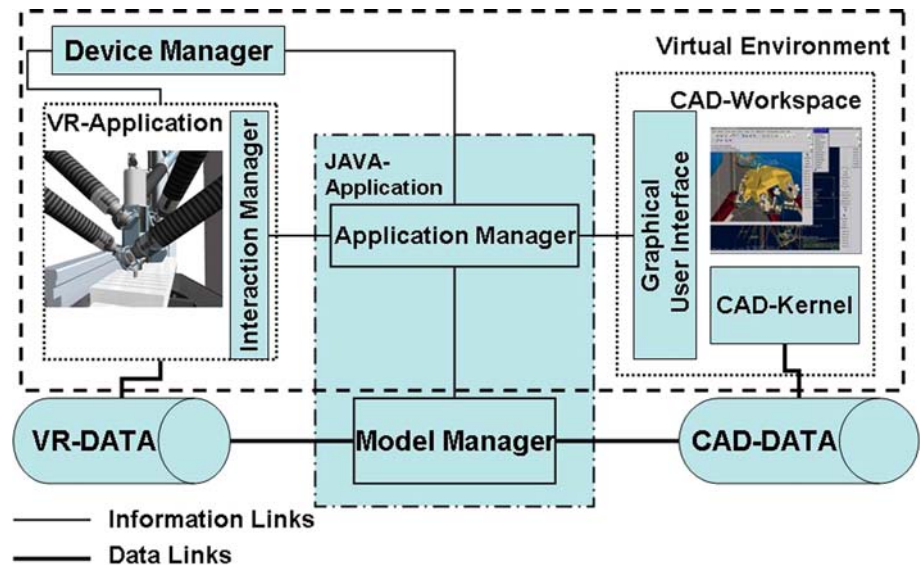
Fig. 4 Implementation of the virtual loupe

The ChangeView() method was implemented for this purpose. It changes the zoom factor, view orientation, and view translation of the active CAD model in real time. In addition to the active CAD window, parameters transferred include the position  $\vec{H}$  and orientation  $H_0$  (head, pitch and roll) of the visor or head, position  $\vec{T}$  and orientation  $T_0$  (head, pitch and roll) of the tablet PCs, and offset values for position  $\vec{M}$  and orientation  $M_0$  (head, pitch and roll) of the model in the cave. The starting point for calculating the zoom factor is the axis of view in the form of the straight line

$$g_{HM} : \vec{H} + t \cdot \vec{HM} : t \in \mathbb{R}^+ \quad (1)$$

It is used to calculate the distance of the head from the model center and the distance of the head from the tablet PC. The parameters for the orientation of the CAD view are derived from the view orientation matrix. It is calculated from the

**Fig. 5** NavIMode system architecture



matrix product  $O_{HM}^{-1} \cdot O_M$ , where:

$$O_{HM}^{-1} = R_y \left( -HM_o^{(1)} \right) \cdot R_x \left( -HM_o^{(2)} \right) \cdot R_z \left( -HM_o^{(3)} \right) \quad (2)$$

$HM_0$  describes the angle of orientation of the view axis and is calculated from:

$$HM_O = \left( -\arctan \left( \frac{\vec{HM}^{(1)}}{\vec{HM}^{(2)}} \right), \arctan \left( \frac{\vec{HM}^{(3)}}{\sqrt{\left( \vec{HM}^{(1)} \right)^2 + \left( \vec{HM}^{(2)} \right)^2}} \right), T_O^{(3)} \right) \quad (3)$$

The vector  $\vec{TF}$  is of great significance for the view translation. It points away from the model center to the view axis to which it is perpendicular. Point  $F$  is the perpendicular point from point  $\vec{T}$  to the straight line  $g_{HM}$ .

$$\vec{TF} = \vec{TH} - \frac{\left\langle \vec{HM}, \vec{TH} \right\rangle}{\left\langle \vec{HM}, \vec{HM} \right\rangle} \vec{HM} \quad (4)$$

In addition, the roll angle  $T_O^{(3)}$  of the tablet PC is decisive for the pan parameter. This is why

$$\vec{TF} := R_y \left( T_O^{(3)} \right) \cdot \vec{TF} \quad (5)$$

is set. The model is now moved in  $x$  direction by the value of vectors  $\vec{TF}^{(1)}$  and in  $y$  direction by  $\vec{TF}^{(3)}$  (*SetPanY*). It is planned to supplement the basis transformations described

with an adjustment of perspective and correct clipping of the CAD model according to the user's position.

#### 4.3 Architecture

NavIMode is an interface between a VR system and a CAD system. The architecture of the system is strictly object-oriented, and its structure is modular. The communication of each module is organized via events. The modules currently communicate via a wireless local area network (WLAN) using an Ethernet connection. This allows wireless working but it also creates some problems due to the higher latency times of WLAN infrastructures (Fig. 5).

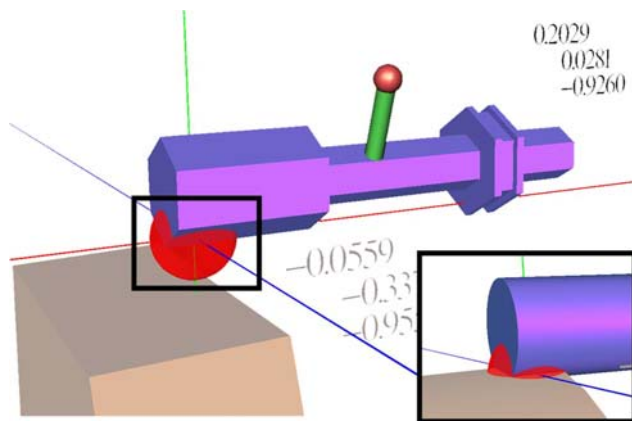
The CAD system offers JAVA libraries for controlling the graphical user interface (GUI) and for accessing the model data C++ libraries as application programming interfaces (APIs). A TKL/TK and C++ classes are available on the side of the VR system for accessing objects of the scene graph and for implementing interaction metaphors. The two systems are linked using the following modules: *application manager*: initializes the asynchronous connection between the VR and CAD systems, *device manager*: collects data on the position of the visor and input device, *model manager*: keeps the CAD model and the VR model consistent and maps CAD elements and VR objects and *interaction manager*: coordinates all activities of the system and provides navigation, selection, and modeling functionality.

### 5 Construct|Tool: immersive modeling environment

#### 5.1 Motivation

A Construct|Tool was developed with the goal to implement an immersive modeling environment which works on





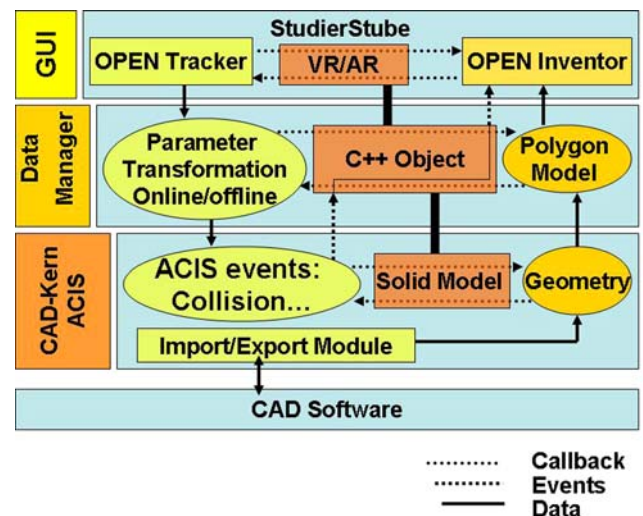
**Fig. 6** Collision detection based on polygons (*left*) and based on a B-Rep model (*right*)

the basis of exact geometries. In addition to modeling functions such as interactive Boolean subtraction, generation of freeform surfaces, creation of primitives, kinematic analysis functions such as real time collision calculation are available. The environment is particularly suitable for designing and verifying forming tools as it includes the option to perform collision calculations based on exact geometries. The required modeling of volume elements, freeform areas, and simultaneous verification of mechanical functions of sliders, ejectors, and handling devices can be performed as an integral part of the modeling process. In addition, the problem of installing the tool in an existing press can be considered as early as in the design process.

## 5.2 Functionality of Construct|Tool

The user interface is the 3D interface of the “StudierStube” AR software consisting of pen and PIP (personal interaction panel) [11]. The pen of Construct|Tool provides two buttons that allow fast access to frequently used functions. Exceeding the functionality of Construct3D, a data interface with CAD systems provides the options of transforming objects directly and discretely and of real time collision calculations. There is a visual and acoustic feedback for collision warning. In addition, the objects can be prevented from entering other geometries while being interactively transformed (Fig. 6).

Functions such as real time and Boolean subtraction are available for modeling specific geometries (such as engraving). This function enables a user to skim material off a geometry to model freeform areas. The intersection volume of the virtual modeling tool and the object is removed in real time. The motion path of the tool is approximated along a B spline to create two continuous area transitions. Like all objects in Construct|Tool, the modeling tool can be moved directly using the control console. Collision visualization



**Fig. 7** Construct|Tool system architecture

tools display the position and size of the removed volume during the modeling process.

## 5.3 Architecture of Construct|Tool

Construct|Tool is based on the “StudierStube” AR/VR system [11] and on the “Construct3d” framework [3]. The core of the software is the data manager that initializes the link between interaction data and geometrical data. The collision detection function of Construct|Tool is based on the methods of the ACIS API. It enables very accurate calculations of geometry penetration (e.g., for examining tolerance pairs). A specific feature is the option to prevent data transfer to ACIS to keep interaction performant during large-scale operations (Fig. 7).

## 6 Conclusion and outlook

The three systems presented mark current approaches to implementing immersive design. The VRax<sup>®</sup> solution is a consistent design platform, NavIMode is an integrated VR-CAD environment, and Construct|Tool is a combined modeling and verification tool.

By combining construction kits and the possibility of immersive modeling, VRax<sup>®</sup> is suitable for the fast design of products with a high component repetition rate and multiplicity of variants.

For designing products with a complex design topology and a large number of components, NavIMode offers reduced times of conversion processes, transparent design processes and reduced selection, navigation, and positioning times.

Construct|Tool makes the geometry description immediately available as B-rep model (boundary representation).

This provides a precise testing environment for installation and kinematic simulations.

The future use of immersive design offers a consistent design process that is transparent to the customer. The approaches presented allow a significant increase in efficiency in the draft and design process.

## References

1. Neugebauer, R., Weidlich, D.: Zickner: virtual reality solutions for the design of machine tools in practice. In: Proceedings of Intelligent Computation in Manufacturing Engineering, ICME, Italy (2006)
2. Stork, A.: Effiziente 3D-Interaktions- und Visualisierungstechniken für benutzer-zentrierte Modellierungssysteme. PhD Thesis, Sept. 2000, pp. 116 (2000)
3. Kaufmann, H.: Geometry Education with Augmented Reality. PhD Thesis (2004)
4. Fiorentino, M., de Amicis, R., Monno, G., Stork, A.: Spacedesign: a mixed reality workspace for aesthetic industrial design. In: Stefan, M. (ed.) u.a., Institute of Electrical and Electronics Engineers (IEEE) u.a. (2002)
5. Krause, F.-L., Göbel, M., Wesche, G., Biahmou, T.: A three-stage conceptual design process using virtual environments. In: Proceedings of WSCG 2004, Plzen (2004)
6. Jayaram, S., Wang, Y., Jayaram, U., Lyons, K., Hart, P.: A virtual assembly design environment. In: Proceedings of the IEEE VR'99, IEEE (1999)
7. Conrad, S.: System control and hybrid 3D/2D manipulation in virtual environments using tracked handheld computers, 2. In: Workshop der GI-Fachgruppe VR/AR, Aachen (2005)
8. Conrad, S.: Immersive interactive authoring of virtual environments based on state charts and event flows. In: Proceedings of the Central European Multimedia and Virtual Reality Conference
9. CEMVRC: Eurographics Association, Prague, pp. 57–62 (2005)
10. Neugebauer, R., Weidlich, D., Zickner, H., Hensel, S., Ihlenfeldt, S., Polzin, T.: A virtual reality based engineering tool for fast configuration of machine tools with parallel kinematics, VRax®. In: Proceedings of the 5th Chemnitz Parallel Kinematics Seminar PKS 2006, Chemnitz, pp. 39–62 (2006)
11. Hasselbring, W.: Modeling software architectures. In: Paech, B., Desel, J. (Hrsg.) Tagungsband zur Modellierung 2005, Heidelberg, März 2005. IEEE Recommended Practice for Architectural Description of Software-Intensive Systems, 2000. IEEE Standard, pp. 1471–2000 (2005)
12. Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavári, Z.S., Encarnacao, L.M., Gervautz, M., Purgathofer, W.: The Studierstube Augmented Reality Project, Presence-Teleoperators and Virtual Environments, vol. 11, pp. 33–54 (2002)