

# Building Immersive VR Experiences for the General Public: Technical, Economic and Ergonomic Choices

Notes for VR FOR PUBLIC CONSUMPTION Workshop at VR 2004 - 27 March 200

David Nahon

Virtools

93, rue vieille du temple

75003 Paris - France

<http://www.virtools.com/> - [nahon@virtools.com](mailto:nahon@virtools.com)

<http://david.cyber-nahon.net> - [david@cyber-nahon.net](mailto:david@cyber-nahon.net)

Successfully designing immersive VR experiences for the general public is a challenging task. The parameters to take into account can be classified in 3 major categories:

- **Ergonomics** of the application itself,
- **Economic** constraints,
- **Technical** excellence.

## Attributes of the Target Audience

The following specificities must be considered when designing applications for the general public:

*Note: by "general public" we are referring to edutainment and similar contexts, such as museum and gallery exhibits, public performance, etc.*

- Spectators come from a wide range of background knowledge, skills and "comfort level" (children being "better" users than adults).
- Spectators may show up (or leave) at any time.
- Users may be "technologically inhibited" or afraid of interacting, may not even have wanted to attend.
- Users may be impatient or easily frustrated.
- Users may not do what is expected, may interact in unpredictable ways.
- Entertainment value must be sustained.
- Hardware potentially at risk (theft or damage).

## Technical Perfection: The Key to Museums is Easy Maintainability

- Hardware must have "military-like" robustness:
  - o Management of the cables, plugs and other equipment,
  - o No active device without info-trainer.
- Maintenance Document must provide all salient information:
  - o Morning startup,
  - o Evening shutdown,
  - o Manual checking to verify that all operations are functioning correctly.
- Simple operating modes (one click!).
- Automatic failure detection and recovery/alert (message on console, sms, e-mail, etc.).
- Spare hardware available and ready for use (simple video routing).

## Display Options

- HMD (head-mounted tracker)
  - o Disney approach (maintains frame attached to head with HMD clipped on).
  - o Handheld is more accessible (nVision VB 30, custom, etc.), cf. *Art Impact* and *SoSoSo* by Z-A/Maurice Benayoun.
- CAVEs (AEC, ICC, etc.)
  - o Possible only with "info-trainers" present.
  - o Very few people/hour can view the application.
  - o Frustrating for viewers not wearing the tracker and/or interacting (for a good example cf. *World Skin* by Z-A/Maurice Benayoun).
- Large Screens (domes, Reality Center, I-cone, etc.)
  - o Requires hardware edge blending for high quality.

- Difficult to maintain (setup is always unsatisfactory unless daily adjustments are made).
  - Advantageous since many people simultaneously have a good view of the application, but expensive
- Small Tracked Stereo Walls (VixBox, GeoWall, etc.)
  - Performs fine with passive stereo.
  - Uses passive tracking (by video analysis).
  - Can they be used without any info-trainer present ?

## Tools for Developing VR Applications

### Virtools is an Intuitive Interactive 3D Authoring Tool Based on Graphical Scripting

- Ideal for technical artists.
- Accommodates collaborative work between technical and creative production team members.
- Fast, iterative platform: no compilation, modifications are instantly displayed.
- Compatible with most DCC (Max, Maya, etc.) and CAD formats (Catia, Pro/e, Solidworks, Unigraphics, etc.).

### Virtools VR Pack extends the standard Virtools Dev platform to provide:

- Excellent stereo capabilities, even with consumer graphics cards.
- Support for most VR Peripherals via VRPN.
- Support for multiple displays and PC clusters (genlock, Framelock, etc.), scalable solutions.
- Easy to start/stop from a remote control room.

## Ergonomics Issues: Typical Public Kiosk "State Machine"

- Engaging standalone mode with presence detection.
- Credits and documentation when presence is detected.
- Validation of user interaction and understanding through a small tutorial.
- Interaction with possible session time control (adjustable).
- Inactivity timeout if no user interaction detected, return to standalone mode.

## Strategic Choices

- Cheap hardware doesn't necessarily mean economic savings.
  - Think of total cost of ownership (TCO):
    - How long will the projector lamp last, how much does it cost?
    - How often will I need to realign the image, how long does it take?
  - Include thorough testing of low-end consumer hardware to validate robustness, aging parameters.
- Content and hardware are extremely interdependent.
- Designing a VR Experience doesn't mean simply designing what's on the screen – the overall physical environment is crucial.
- Think of public flow/viewer behavior:
  - Base parameters on length of time for individual viewer experience, time spent waiting in line, etc.
  - Consider spectators who are not interacting and their experience as viewers.
  - Include "profitability" issues.
- Evaluate the need for an "info-trainer": Is it affordable? Is it necessary? (Info-trainers should be actively involved as part of the show, not replacements for bored security guards.)

## David Nahon

[david@cyber-nahon.net](mailto:david@cyber-nahon.net) (perso)  
[nahon@virtools.com](mailto:nahon@virtools.com) (pro)

<http://david.cyber-nahon.net>



## Biography

David Nahon was born in Paris in 1968. He studied Electrical Engineering at SUPELEC, one of the major French "Grande Ecole", and obtained a Master of Science in "Communications and Signal Processing" from the Imperial College of Science and Technology in London, in 1991.

Following his strong interest in visual arts he began studying 3D computer graphics and esthetics at the MSc of University Paris 8 "Art and Technology of Image", and started a Ph.D. in "Esthetics Science and Technology of Arts".

In 1994 he entered Z-A Production as a research and development engineer and developed PERSONA, a visual programming toolbox for procedural and interactive animation in real-time, which was used for many 3D computer animation productions in Z-A.

At the same time, he got involved in the development and direction of various interactive artwork from Maurice Benayoun, using network, real-time 3D animation, VR immersive environments, interactive sound, multimedia, internet ... and became technical and R&D director of Z-A

After 8 1/2 years at Z-A, David Nahon joined Virtools as the "VR and Simulations Solutions Manager"

## Principal References

### Is the Devil Curved

<http://www.z-a.net/diable/index.en.html>

Interactive sound and image installation (for Imagina 1995)  
Tech: 3D real-time animation, behavioral dynamic system, genetic algorithm, sound client/server with 4channel spatialization (ZAsoundserver)

### The Tunnel under the Atlantic

<http://www.z-a.net/tunnelatlantique/index.en.html>

### The Paris/New Delhi

<http://www.z-a.net/tunnelpnd/index.en.html>

Networked Virtual spaces

Tech: Visio-conferencing over IP, site sync, beginning of ZAprfiler

### World Skin

<http://www.z-a.net/worldskin/index.en.html>

CAVE installation created for Ars Electronica Center.

Tech: Behavioral sound and space.

### And what about me (1) ?

<http://www.benayoun.com/EtMoi/gb.html>

Collaborative visual java applet

Tech: Dedicated server, applet, image processing

### The Idea Machine

Roulette-like game for an e-commerce site ([www.clust.com](http://www.clust.com)).

Tech: Text generation (Using Jean-Pierre Balpe's engine), Flash, cgi (perl, servlets)

### Crossing Talks (Communication rafting)

<http://www.benayoun.com/crosstalking/crosstalking.html>

CAVE interactive environment for ICC, Tokyo

Tech: Infrared video user tracking, QuickTime in texture, visio-conference in texture, sound behavior, space behavior.

## Art Impact (Collective Retinal Memory)

<http://www.benayoun.com/artimpact>

3D and Web collaborative installation for Centre Pompidou, Paris  
Tech: Client/server architecture, java applet, real-time 3D, dedicated server, real-time image processing, panorama real-time viewing.

## Panoramic Tables

<http://www.z-a.net/tablepano>

Augmented reality installation for Hanover Expo 2000  
Tech: Virtual/real matching, dynamic behavioral animation.

## Oiron à la Conciergerie

<http://www.z-a.net/oiron>

Subpart of Art Impact. Shown at La Conciergerie de Paris

## Labylogue

<http://www.z-a.net/labylogue>

Networked virtual environment for exhibition "Tu parles ! Le français dans tous ses états", Dakar, Brussels, Lyon.  
Tech: Voice recognition, text to speech, text generation (Jean-Pierre Balpe's Engine), OpenGL in java, video-conferencing.

## The interactive Membrane

<http://www.z-a.net/membrane>

Central part of the 14 months lasting exhibition "Transformed Man" at Cité des Sciences et de l'industrie, Paris  
Tech: 7 synchronized PCs with animated textures and pixel shaders on GeForce3. Newer version of the ZAPProfiler

## The Waves

Interactive video artwork - Thierry Kuntzel, created for LisboaPhoto03 at Centro Cultural do Belem, Lisboa  
Tech: Realtime control of the playback of a quicktime video with OpenGL. Users positions tracking with IR Laser

## Principal Software:

### PERSONA

<http://www.z-a.net/persona>

A visual programming toolbox for procedural and interactive animation in real-time.

### ZAsoundserver

<http://www.z-a.net/zasoundserver>

Client/server software for playing, spatializing and modulating sounds in real time.

### The Travel Tunnel / ZAPProfiler

<http://www.z-a.net/tunvoy>

A dynamic and intelligent visual database viewer

### Moving Words

(like on <http://www.benayoun.com/Acentre.html>)

A java applet for presenting images and text links in a dynamic way

### SAS<sup>Cube</sup> / SASlib / Virtools VRPack

<http://www.z-a.net/sascube>

The SAS Cube is the first French *Immersive Cubic Room* to run in France, and the first running on a cluster of PCs,

The SASLib is an API similar to the CAVELib, but targetted to PC clusters. It allows to write SAS Cube applications in C/C++.

The [Virtools VR Pack](#) is a complement to Virtools Dev that permits the creations of highly interactive distributed VR content with a graphical programming language (for the SAS Cube or any other VR platform)

## DEVELOP ONLINE AND OFFLINE APPLICATIONS WITH RICH INTERACTIVITY

### VIRTOOLS DEV: A CUTTING-EDGE COMPREHENSIVE TECHNOLOGY AT YOUR FINGERTIPS

Virtools' unique and proven development paradigm considers 3D objects as individual components, independent of the data associated with them. The resulting architecture is extremely flexible, allowing developers to attribute modular behaviors to objects and manage them easily and efficiently.

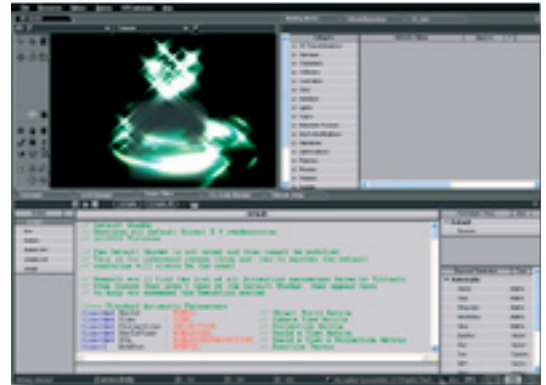
Developers drag and drop behaviors in an intuitive graphical user interface to create complex applications with the high-quality graphics and interactivity found in top-selling games. The new Action Manager is an important productivity enhancement for authoring in Virtools. Developers are able to create scripts for frequently used functions which performs a predefined task on a selection or parameter and access them in just a few keystrokes. For high-end developers, the Software Development Kit (SDK) available with Dev allows them to create custom behaviors and access the system API's.

Once the application has been created and tested, it is easily deployed offline, over the Internet or on immersive displays.

Virtools Dev includes a complete project management environment that integrates seamlessly with other solutions such as NxN's alienbrain Asset Management software (accessed directly through the Virtools Dev interface) or other data management softwares, and facilitates version control and files and media management. Virtools is the perfect toolset for effective communication and workflow. Manage and supervise projects with Virtools, while making adjustments in real-time that accelerate corrections and tuning.

Designers no longer have to manage the «boring» details of low-level programming, but can concentrate on content.

With Virtools, a single graphical user interface opens the door to specific advanced add-on modules such as sophisticated Physics, Artificial Intelligence, and advanced Multiuser capacities, not to mention integration and use of 3<sup>rd</sup> party technologies. Virtools Dev let you equally deploy behavior-enriched, adaptative interactivity for dynamic 3D applications thanks to the Web Building Blocks packaged into Dev.



### ACCELERATE TIME-TO-MARKET, REDUCE DEVELOPMENT COSTS AND ELIMINATE DEVELOPMENT RISKS

By separating objects from the data and behaviors applied to them, studios and integrators using Dev can more economically develop various application modules simultaneously, further shortening time to market. Development time is significantly reduced thanks to Dev's library of over 450 reusable behavior building blocks.

Virtools Dev's graphical user interface enables both non-technical designers and high-end programmers to assemble the sophisticated behaviors that create rich interactivity. Applications grow more complex while scripts remain easy to handle.

Applications requiring the use of off-the-shelf behaviors and system features can be developed faster. What previously required over a year of development time can now be accomplished in a matter of months. On average, our clients have reduced their product timelines by 50%.

By using Virtools Dev, companies can now bring together a balanced team of designers and hard-core programmers, all working with Dev as a single hub to create high-powered technical applications that deliver compelling interactive content.

By dramatically optimizing the development of interactive 3D applications with Dev, our clients minimize production costs. Budgets can buy the most versatile development team with the widest range of expertise – from designers to developers – and with Dev's smooth workflow, time to market is further reduced. All factors considered, product schedules are accelerated and development costs can be reduced by as much as 35%.

Corporate developers, game studios, web agencies and system integrators can't afford to take risks when they embark on developing highly interactive applications. The industry is full of ambitious projects that have never seen the light of day due to development obstacles, delays or out of control costs.

By using Virtools Dev, companies reduce development risks associated with creating web and stand-alone applications.

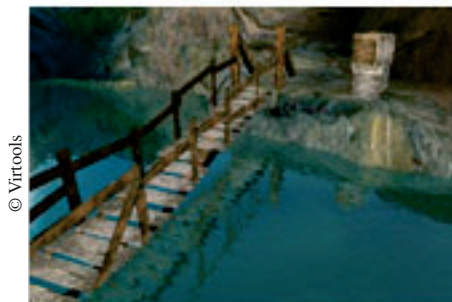
Dev's iterative development process lets production teams move forward together. Gameplay can be tested throughout the development process and the reusability of Dev building blocks optimizes workflow for future projects. With Virtools Dev, small budgets buy solid production teams, and effective workflow greatly reduces production time.

#### FEATURES OVERVIEW

- **New paradigm based on separation of objects, data and behaviors, and applications created by assembling objects to behaviors.**
- **Intuitive user interface with real time visualization window and graphical programming.**
- **Web Building Blocks for dynamic media and data download and user-customized content delivery.**
- **Use the Virtools Scripting Language and its intelligent coloring system, context-sensitive completion and function arguments display.**
- **Complete project management capabilities (compatibility with NxN's alienbrain in the Dev interface).**
- **Cutting-edge rendering with Programmable Vertex and Pixel Shaders.**

## VIRTOOLS™ DEV

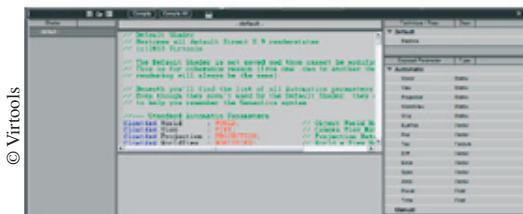
Virtools™ Dev includes 5 key components: **the Graphical User Interface** to develop sophisticated applications by visually assembling objects and behaviors, **the Behavior Engine** to run interactive applications, **the Render Engine** to render graphics in real-time, **the Virtools Scripting Language** to create low level specific functions without any C++ line and **the SDK** to create custom behaviors.



## THE USER INTERFACE

The Virtools Dev graphical user interface is used throughout every stage of development. It includes:

- A 3D Layout to display content in a real-time environment.
- Graphical tools for navigating, creating, editing, selecting and manipulating 3D objects, lights, cameras and curves.
- Creating and editing lights, cameras, materials, textures, grids and paths.
- Translation, rotation, scaling of 3D entities and navigation within the virtual environment.
- Drag-and-drop behaviors onto 2D and 3D objects.
- Creation of new reusable behaviors by graphically combining existing ones.
- A Schematic View to graphically assemble and fine-tune behavior building blocks for creation of interactive content.
- A Script Debugger to fine-tune the application.
- Entity Setup Tools to edit the parameters of any object that has associated behaviors.
- An Attribute Manager for quick visualization and modification of attribute values for multiple objects.
- An Action Manager to create scripts for frequently used functions which performs a predefined task on a selection or parameter and access them in just a few keystrokes.
- A Hierarchy Manager to display a tree view of all the objects present in any level.
- A Parameters Debugger to inspect and edit data values.
- A Path Manager to define paths to data sources (image, audio or other data files).
- A Profiler to probe how much computing time is devoted to particular tasks.
- A Shader Editor to create programmable vertex and pixel Shaders in DX9 (HLSL).



## THE VIRTOOLS SCRIPTING LANGUAGE

The Virtools Scripting Language is a powerful scripting language that complements the Virtools Dev Schematic editor and the Virtools SDK with an intelligent coloring system, context-sensitive completion and function arguments display. VSL scripts can be processed at run-time or in Author mode.

VSL offers full debugging mode with breakpoint support, watchable variables with value editing, and step by step debugging (also step into/out support):

- For programmers, VSL is an **interface to the SDK from within Dev**. With no need to create a C++ project, you can easily and quickly test new ideas and implement custom code, without custom DLLs.
- For scripters, VSL is ideal for **replacing complex parameter operations** (math calculations, string operations) and creating advanced behavioral scripts.

## THE BEHAVIOR ENGINE

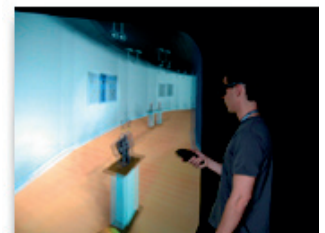
The behavior engine runs both custom and out-of-the-box behaviors. Dev includes standard behaviors from the following categories: Cameras, Characters, Collisions, Controllers, Grids, Interface, Lights, Logics, Materials-Textures, Mesh modifications, Narratives, Optimizations, Particles, Sounds, Shaders, Visuals, Web, World Environments, and more...

The Virtools Behavior Library can be extended with custom behaviors developed with the SDK, with the Behavior Pack or with third-party behaviors created by the Virtools user community.

## THE RENDER ENGINE

The render engine provides high-quality, real-time rendering of 3D images and animations in real time. It includes the following features:

- Support for key industry standards: DirectX and OpenGL.
- Supports programmable Vertex and Pixel Shaders (DX9, HLSL).
- Support for 3D modeling objects and animation from 3ds max, Maya and Lightwave.
- Optional access to render engine source code.



## THE SOFTWARE DEVELOPMENT KIT (SDK)

The Virtools SDK is a suite of development tools (libraries, DLLs, header files) that provide access to all the low-level functionality used by Virtools software. Developers can write the following application components:

- Custom application executables using the Virtools engines as underlying technology.
- Extensions to the Virtools engines, such as Behaviors, Media Importer, Manager, Render Engines Plugins, Rasterizers and Extension Plugins (specific Parameter Types).

## OPEN ARCHITECTURE

Virtools offers an open and flexible architecture that is compatible with the following standard technology format:

- 3D files: 3DS Max, Maya, 3DS, X., Lightwave
- Images: JPG, PNG, TIFF, TGA, BMP, PCX.
- Sounds: MP3, WMA, WAV, MIDI.
- Video: AVI, Active Movie supported formats.

## TECHNICAL REQUIREMENTS

- Pentium 500 or faster PC (Pentium III recommended).
- Windows NT 4.0 (with Service Pack 6) or Windows 95/98/98SE/2000/ME/XP.
- 128 MB of RAM (256 MB recommended).
- Direct3D or OpenGL compatible 2D/3D accelerator card with 32 MB of RAM (3D card supporting DX9 for shaders).
- Sound Card (not required but recommended).
- Internet Explorer 4.0+.

### Virtools SA

93, rue Vieille du Temple  
75003 Paris - FRANCE  
Ph: (+33) 1 42 71 46 86 / Fax : (+33) 1 42 71 86 53

### Virtools Canada Inc.

1250, Bvd Rene Levesque Ouest/Suite 2200  
Montreal QC H3B 4W8 - CANADA  
Ph: (+1) 514 989 3126 / Fax : (+1) 514 989 3127

# Virtools™ VR Pack

## VIRTOOLS SOLUTION FOR TOTALLY IMMERSIVE PC CLUSTER

Companies increasingly recognize the benefits of placing users directly within the data space for certain immersive applications - using VR (virtual reality) technology. Users can directly navigate through the environment and interact with it, much like experiencing the real world. However, VR has been prohibitively expensive, both in terms of development and equipment (specialized rendering equipment and advanced VR peripherals). Companies require a powerful complete solution that allows them to concentrate on creating the ultimate immersive VR experience.

*Virtools VR Pack is an add-on library for Virtools Dev that allows developers to create totally immersive, full-life experiences using industry standard VR peripherals and/or PC based distributed computing (clusters).*

*“With Virtools VR Pack, creating contents for immersive spaces such as the SAS Cube® is made very easy. The power of Virtools Dev, combined with the VR Pack’s extensive support for VR peripherals and distributed computing( through PC clusters), positions this VR solution as a serious alternative to big monolithic configurations. It’s a revolution for virtual reality”, said David Nahon, Virtools VR and Simulation Solutions Manager*

### FULL IMMERSION

Based on the open source standard VRPN (more info at <http://www.cs.unc.edu/Research/vrpn>), the Virtools VR Pack provides an easy out-of-the-box solution to manage and get the most out of industry standard VR trackers. A VRPN server is installed on the PC managing the input devices. Device-state information is specified in a configuration file (that includes required axis modifications and tracker/antenna offsets) and is retrieved in a Virtools scene via Virtools Behavior Building Blocks, familiar to all Dev users.

With head-tracking Once correct user-centered perspective is setup with the head-tracker and full immersion is achieved through the use of stereoscopy, whether using active or passive equipment (shutter glasses, shutter screens, polarizing filters or glasses).

### DISTRIBUTED COMPUTING ON PC CLUSTERS

Virtools VR Pack enables distributed computing for PC clusters, which can radically lower the cost of VR projects while offering very high performance. Scenes are synchronized by reading parameters in the master scene and passing them to client scenes. Relevant parameters for synchronization are determined by the developer authoring the application, and are easily accessible in the Schematic editor of Dev.

PC cluster distributed computing enables rendering from different viewpoints, a requirement for high performance passive stereoscopy or multi-screen displays.

Note that PCs with two graphic cards or a dual-head graphic card can render two viewpoints simultaneously.

Virtools VR Pack can also take advantage of scalable cluster platforms like Orad DVG: Add more rendering nodes per screen to increase polygon or pixel performance and/or antialiasing quality.

### GENERAL FEATURES

- Designed for PC clusters
- Support for many industry standard VR peripherals
- Multi-Display synchronization with FrameLock & Genlock capabilities

- Simulation mode in the absence specific hardware
- Active or passive stereoscopy

### SUPPORTED DISPLAYS ( PARTIAL LIST)

- Virtual reality cubic rooms (e.g. SAS Cube®, Cave®, VR Cube®, I-Space®)
- Panoramic rooms (e.g. Reality Center®)
- Immersive tables/desks (e.g. Baron®, ImmersaDesk®, Workbench®)
- High-resolution image walls (e.g CAD wall)
- Desktop multi-monitors
- Any head mounted displays or VR goggles

### SUPPORTED INPUT DEVICES (PARTIAL LIST)

- Polhemus Fastrak
- Ascension Technology Flock of Birds, including standalone mode
- InterSense IS600, IS900, Inertiacube, Intertrax
- ARTtracking optical trackers and flysticks
- Murray Consulting Inc Wanda
- Logica/Magellan SpaceBall and SpaceMouse
- All windows compatible game controller

### TECHNICAL REQUIREMENTS

- Virtools Dev
- DirectX/OpenGL compatible 3D graphics accelerator card
- Genlockable graphics card for multi-screen active stereoscopy (e.g Wildcat 7210, Nvidia quadro fx 2000-G)
- Scalable cluster hardware for ultimate performance (e.g. Orad DVG)

### KEY BENEFITS

- Greater flexibility and reduced costs with standard PC hardware clusters
- VR peripheral device drivers ready to use out of the box
- More realistic experiences with advanced display synchronization
- Rapid, low cost content development using simulation mode (single PC)
- Faster, more effective development using Dev’s intuitive interface (GUI)