

# AN UNCONVENTIONAL SOFTWARE ENVIRONMENT FOR FACTORY LAYOUT DESIGN AND AUTOMATED CONVERTER

Petr Horejsi, Jiri Polcar

Department of Industrial Engineering and Management  
Faculty of Mechanical Engineering  
University of West Bohemia  
Pilsen, Czech Republic

e-mail: tucnak@kpv.zcu.cz, jpolcar@students.zcu.cz

The development of a company layout is a crucial issue during Product Lifecycle Management. When introducing a new product or innovating the current one whilst respecting the digital factory concept there is a need for stepping through more phases from a technical point of view: construction, technology and digital manufacturing design. This paper deals with the digital manufacturing design stage only – which consists mainly of the development of the manufacturing layout. The manufacturing design stage can be divided into: Rough planning, Fine planning, Ergonomic Analyses and Discrete Simulation Validation [Kurkin 2010a][1]. All these stages can be supported by Virtual Reality hardware and software tools. This paper presents a method for producing simulations of factory design with a new self-developed procedure. The first version of an automatic convertor was also developed. This convertor can convert a layout designed with the aid of commercial off the shelf software visTableto a particular environment.

## Keywords

Virtual Reality, visualization, Source Engine, visTable manufacturing, factory layout

## 1. Introduction

Virtual Reality is a medium defined by two main genres: virtual tour and virtual training. The best way to validate a brand new design is to be immersed by it. This paper will show how we used tools which were made by developers of computer games to create a new environment which can be used for development of a virtual layout. Developed virtual tours can be performed not only using conventional keyboard/mouse control but can also be presented in CAVE (Computer Aided Virtual Environment) – Virtual reality laboratory with special controllers.

There are several software tools for modeling digital factories with the possibility of adding models from a 3D library to user composition. Two of today's major solutions are the Dassault Systeme DELMIA and Siemens Tecnomatix. These can be used to perform a large number of analyses [Kurkin 2010b]. From these two packs the full virtual tour in CAVE can be provided by Teamcenter VisMockUp software but with limited interaction. The other suitable tool is visTable which natively supports a virtual tour but without the possibility of interacting with the virtual world. There was chosen the visTable software for automatic conversion to developed environment. More case studies (layout optimization) for practical applications were performed. It was also used for educational purposes because of its user friendliness.

Source Engine was chosen as the new "sandbox" environment (the engine is the graphical and mathematical core of an application). This engine was originally designed for gaming. This is not the first use of originally gaming engine for serious applications. Source Engine was for example used as a restaurant hygiene simulator [Namee 2006].

There is currently a big boom in serious games (training) with the implementation of the multiplatform Unity3D Engine. Unity3D has been used for Serious Games, Task Management Training, Medical Visualization, Real time CAD Rendering, etc. [Unity 2013]. We can also mention for instance Cry Engine which is used as a surgical simulator [slinky 2006].

## 2. Source Engine and Source SDK

As mentioned before –the implementation described here uses **Source Engine**, which is a modular game engine for PC, Linux, Mac OS, PS3 and Xbox. Source Engine was released in 2004 for the computer game Counter-Strike: Source, and then for Half-Life 2 one year later. Source is a high-quality graphics core with simulated physical systems support using the Havok engine. Source engine is based on the DirectX architecture with the possibility of High Dynamic Range. More features such as multicore rendering support, Hardware Facial Animation, "Soft" particles, etc. have recently been added [Wiki 2013].

The virtual environment is built by a **map** (or level). The virtual world can be comprised of more interconnected maps. All particular environments are limited to a user-defined enclosed volume which is composed of **brushes**. Brushes are basic 3D objects that represent walls, floors, ceilings, cliffs, terrain, etc. The world details: e.g. furniture, humans, trees or sometimes even whole buildings (with esthetic functionality only) are represented by **3D models** (static, dynamic and physical props). Brushes are objects of lower detail than 3D models and are modeled and edited directly in Source SDK enclosed tools. On the other hand, 3D models are modeled using complex 3D editing tools like 3DS MAX, Blender or XSI Soft Image and then imported to a Source based map. All items and NPCs (Non-Player Characters) are represented by detailed 3D models. Functional parts of maps are implemented by so-called **entities** (doors, lifts, switches, light control, any mathematical or physical logic, etc.).

Every virtual environment developed in Source Engine is physically stored in a specific directory structure which includes for instance these components (directories):

- Maps – all maps are stored here (BSP – Binary Space Partitioning format)
- Materials – contains all textures (VTF – Valve Texture File) including a material description file in text format (VMF – Valve Material File)
- Models – all the detailed 3D models (MDL format). Only special textures from Materials directory can be used for models.
- Scenes – scripted scenes data (choreography) created by the FacePoser tool which is included in the Source SDK
- Sound – sounds in WAV format

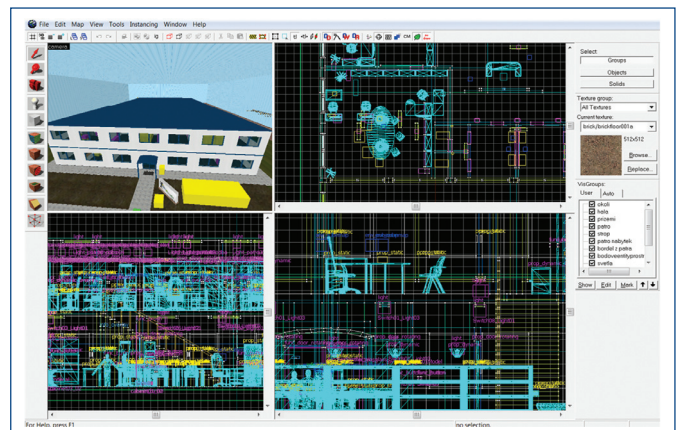


Figure 1. Editing in Valve Hammer Editor

It is possible to inherit data from other modifications (adapted Source Engine packs) or Source powered games. In other words, it is possible to use all previously designed models, textures, NPCs, etc.

Components of the virtual environments in Source Engine are created and edited using the Source SDK which is a powerful development package tool which can be downloaded via the Steam distribution system by the owner of a valid Source engine game (like Half Life 2). This environment includes a possibility of new modification (so called 'mods') development. Within or without this modification a new map can be developed. There is also a tool for new map creation called **Hammer Editor** (see Fig. 1). Software tool FacePoser can be used for creating choreographed scenes including face mimics and lips to sound synchronization. There are a lot of other tools included, usually file format converters, such as image to VTF (Valve Texture File) and many more. The source code of the engine is included, so the possibilities are virtually unlimited.

### 3. DIGITOV package

First a reference model was created in order to validate the possibility of developing such complex models in the given engine. There was used a case study: manufacturing of a remote-controlled car. The evaluation of the development, implementation and validation process of more referential models raised the need for new content. So as to be able to effectively create new virtual models of manufacturing factories, it was necessary to produce a brand new library comprising 3D models, textures, interactive features and what are known in Source SDK as **prefabs** (logically grouped brushes, entities and 3D models). Some of these were created while working on the reference model. In order to maximize the modularity of the "building set", a lot more components like these needed to be added (see Fig. 2). The DIGITOV modification for the computer game Half-Life 2: Episode Two started to take form. This is not a standard game modification, like a new story, but a collection of components for developers of virtual enterprise environments. The package includes an automatic installer developed in Delphi. There is just one requirement for the DIGITOV package to work: to own a user license for the Half-Life 2: Episode Two computer game, preferably in the Orange Box edition.

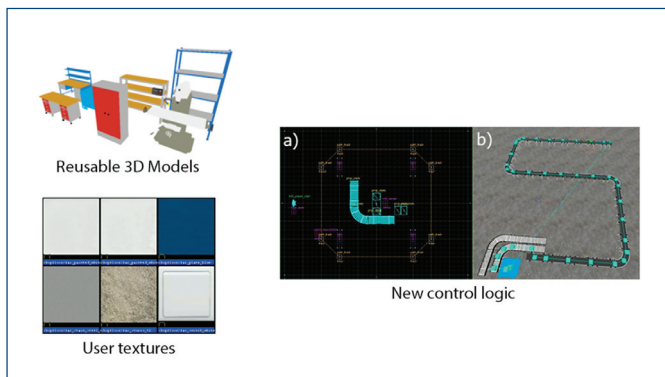


Figure 2. Content of DIGITOV PACKAGE, a) conveyor belt curve prefab, b) conveyor belt constructed from prefabs in Hammer Editor

DIGITOV currently contains approximately 150 new 3D models, more than 50 textures for models, more than 150 textures for maps and more than 25 static and dynamic prefabs. Static prefabs usually represent concrete or brick blocks with/without windows/doors, from which a raw factory layout could be composed. Dynamic prefabs are sets of prepared interactive aggregations. One example of dynamic prefabs from the DIGITOV package are particular active parts of a conveyor belt (start/end, straight, curve, T-shape part, etc.), where besides the model and track path are also defined points where the product model is transformed (assembled), where the belt can be

connected to another part of the belt, where the operator performs interactions, etc. (see Figure 2a), b))

The package also contains various predefined choreographed scenes with virtual employees such as various types of greetings, mentoring, presentations and more. These can be assembled into arbitrary sequences in the level editor. The original virtual actors' clothes were "virtually cleaned and ironed" (see Figure 3-a). All these choreographic scenes were prepared using the FacePoser tool, where the particular face mimics and speech is synchronized and composed into replicas in a timeline (see Figure 3-b). For English language Microsoft Speech SDK can be used for automatic mimic and voice synchronization. For other languages time consuming manual synchronization has to be done.

The DIGITOV package is installed in the modifications folder. DIGITOV supports a new Source Engine 2013 environment.



Figure 3. a) Re-skinned NPC employees in Face Poser, b) Action editor

### 4. An example of DIGITOV package implementation for training

Here is described one of the virtual enterprise models which was designed with the aid of the DIGITOV package from a user point of view.

The first person camera perspective view of the model is shown just after the virtual environment has been loaded. The control system help is displayed immediately after loading the model. The user has a complete mouse and keyboard controlled freedom of movement (user can also use a tracking device). The right mouse key serves for displaying labels or hints – mainly for interactive elements (like offices, persons, products, etc.). This feature is very useful for instance for key elements of conveyor belt inspection.

While taking a tour in one of the models the user follows a storyboard (see Fig. 4):

- External view of the manufacturing enterprise. It is possible to examine the parking lot, climb up on to the roof. Let us take a look inside,
- where the user is welcomed. A virtual lady will take us to the presentation room,
- where the lecturer will show us some basics of the production in this factory (this presentation is more than 10 minutes long, but can be skipped)
- then the user is brought to the manufacturing section. Each particular section of the manufacturing process can be inspected.
- The user can also be familiarized with the functions of every single administrative office in the administrative part of building.



Figure 4. Storyboard

- f) The manufacturing section can be observed from a bird's-eye view.

## 5. Validation and Conclusion – DIGITOV

The developer of the brand new virtual layout now has a tool for exporting his vision of the layout to virtual reality. The developed package is an alternative to other enterprise design software. It is a convenient tool not only for students, but can also be used for commercial use (a valid commercial license needs to be purchased by an enterprise). In order to make maps in DIGITOV environment, only a gaming license (10 Euro) is required.

In comparison with large commercial tools for factory design like Delmia, Tecnomatix or Autodesk Factory CAD the proposed package does not offer the option to perform analyses of material flow, capacity tasks, etc. but the DIGITOV package has advantages in high fidelity visualization, validation of perspective or ergonomics, solution of training tasks and user interactivity implementation with low-cost. It can be used for instance to show the factory to new employees, who are recruited in another part of the world or for visualization of not yet existing company buildings. It can be used as a marketing tool.

Industrial engineers who were approached reacted very positively to the presentation power and high fidelity of simulation. DIGITOV offers an assets library which can compete with factory design software packages. It is also an open environment – so many features can be added or modified by the user.

Right now with the DIGITOV package and a little experience a fully interactive factory design can be produced in a few hours. For example, in comparison with DELMIA, the same factory layout was developed in half the time. The layouts can be easily presented in virtual reality laboratories. Similar high-end virtual reality commercial solutions like Team Center VisMockUp or ICIDO (used mainly in the automotive industry) does not provide many interactivity options. DIGITOV can be successfully used for virtual training and tour not only locally, but it also supports remote access. Virtual meetings can be held in this environment. With properly prepared maps managers can literally change the layout in real-time during virtual layout-optimization brainstormings.

The environments can be viewed stereoscopically on a 3D monitor or in CAVE. The virtual world can be controlled using a keyboard and a mouse or with a special Razor Hydra controller, where a sensor monitors the position of the special controllers in both hands via a magnetic field. The right location has to be found while using this controller in CAVE because of possible interference issues.

The DIGITOV package successfully supports the lessons of the “Digital enterprise and virtual reality” subject at the University of West Bohemia (so far for two years). The students are shown the basics of developing virtual worlds. The package is also being used for lessons at the University’s “Summer school of virtual reality” for high school students (twice during summer vacations). A manual and an education e-book (with more than 200 pages) were written alongside the development.

2 main authors and 12 support students spent 3 years developing DIGITOV and the e-book.

## 6. An Automated Converter

Right now with the DIGITOV package and a little experience a fully virtual reality ready and interactive factory design can be produced in a few hours from scratch. Because of the absence of powerful analyses tools it cannot be expected that all designers will use only the developed DIGITOV package for factory design. But it is a very smart way to make smooth visualizations with the aid of virtual reality techniques (stereoscopy and tracking controller is supported). Interactivity can also be added. DIGITOV would be a very good visualization add-on.

There was missing a “bridge” between developed environment and conventional tools. There was developed a first trial prototype version of an automated converter called VIS2VMF, which can use layouts in the visTable software and convert them to VMF format, which can be read by Hammer Editor. The vision is to make a fully-automatic conversion. The latest version of VIS2VMF software cannot convert all objects from visTable. It can perfectly import 2D layout to Hammer Editor, but some objects and textures are missing (see Fig. 5). The developer needs to fine-tune the missing content manually. But this new approach at least halves the time needed in comparison with developing from scratch.

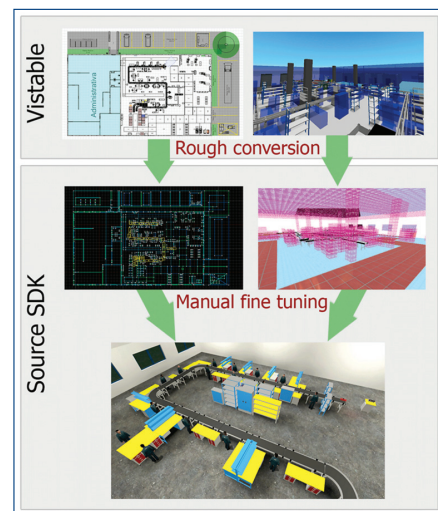


Figure 5. VIS2VMF flow diagram

VMF (Valve Map File) is a source file with a map for Hammer Editor. It is a simple text file describing the 3D world (like VRML). A CSV format (Comma Separated Values – readable by Excel) could be generated from visTable- which is also a text file describing the 3D world. There is need to adjust the format of CSV in accordance with VMF. This task is not as simple as it looks. It is need to detect:

- a) **closed geometric simple objects** in visTable, which can be transformed to simple primitives – brushes (like walls, crates, etc.) in VMF. In visTable these components are called segments. For instance a single wall is made of more segments than needed. There is a need for uniting these segments in one bigger brush in further optimization.

- b) **complex object** which could be represented only by 3D models. The same models have to be on both sides. All visTable models in VRML format have to be manually converted to a native MDL format of Source SDK. This model is saved to VMF map as prop\_static entity.
- c) **worker, person** – these types are also detected in order to ensure animations and interactivity of workers. There can be many scripts (dialogs, interactivity, animations sequence, etc.) added to the worker in the fine-tuning phase.

All these objects are converted according to their types to VMF format.

## 7. Conclusion – converter

This paper presented DIGITOV framework which is right now ready-to-use and also an automated converter where the main core (developed in C++) is almost complete but there is still a lot of work to do. Right now comes the phase of conversion of more than 200 3D models in order to be compatible. This conversion includes the transposition adjustment as well as texturing and creation of description files for all models. Then there is an expectation to place a skybox (fictive hollow cube which defines the finite boundaries of 3D space) automatically. The next goal is optimization of conversion. This includes mainly the visibility issues. The VMF map is compiled into BSP (Binary Space Partitioning) structure which ensures (among other things) the rendering of a visible volume of space. Our converter should divide the 3D world logically in order to save compiling and real-time rendering time.

## Acknowledgements

This paper was prepared with support of the Internal Science Foundation of the University of West Bohemia SGS–2012-063.

## References

- [Kurkin 2010a] Kurkin, O., Simon, M., Optimization of Layout Using Discrete Event Simulation, In: Business transformation through innovation and knowledge management: An academic perspective, vols 1–4, 14<sup>th</sup> International-Business-Information-Management-Association Conference, Istanbul, Turkey, Jun 23–24, 2010, ISBN: 978-0-9821489-3-8
- [Kurkin 2010b] Kurkin, O., Januska, M., Product Life Cycle in Digital factory, In: Knowledge management and innovation: A business competitive edge perspective, vols 1–3, 15<sup>th</sup> International-Business-Information-Management-Association, Cairo, Egypt, Nov 06-07, 2010, ISBN: 978-0-9821489-4-5 (2010) 1881-1886
- [Namee 2006] Mac Namee, Brian et al.: Serious Gordon: using serious games to teach food safety in the kitchen. 9<sup>th</sup> International Conference on Computer Games: AI, Animation, Mobile, Educational and Serious Games CGAMES06 (2006)
- [slinky 2006] Virtual Surgery Far Cry Demo: <http://www.youtube.com/watch?v=QF0yyfhchvg> [10/2013]
- [Source 2013] Source Engine games: <http://www.giantbomb.com/source-engine/3015-751/games/> [10/2013]
- [Unity 2013] Unity Serious Games Reel: <http://video.unity3d.com/video/7402585/unity-serious-games-reel> [10/2013]
- [Wiki 2013] Wikipedia: Source(game engine): [http://en.wikipedia.org/wiki/Source\\_\(game\\_engine\)](http://en.wikipedia.org/wiki/Source_(game_engine)) [10/2013]

## Contacts

Ing. Petr Horejsi, Ph.D., Bc. Jiří Polcar  
 University of West Bohemia  
 Department of Industrial Engineering and Management  
 Univerzitni 8, Plzen, 306 14, Czech Republic  
 tel.: + 420 377 638 495  
 e-mail: tucnak@kp.v.zcu.cz, e-mail: jpolcar@students.zcu.cz