Simulation Training in Oil Platforms

Ismael Santos
CENPES Petrobras Research
Rio de Janeiro, Brazil
ismaelh@petrobras.com.br

Peter Dam, Pedro Arantes, Alberto Raposo
Tecgraf PUC-Rio
Rio de Janeiro, Brazil
{peter,parantes,abraposo}@tecgraf.puc-rio.br

Luciano Soares
Insper Instituto de Ensino e Pesquisa
Rio de Janeiro, Brazil
lpsoares@insper.edu.br

***Abstract*— Training people for working in an oil platform involves teaching operators to locate hundreds of devices in a hazardous environment and to correctly operate them. One small mistake can seriously hurt or kill somebody and compromise the integrity of the plant. On the other hand, access to real platforms is restricted and trainees need to relay on photos and diagrams in order to learn, which does not provide a realistic environment to completely allow them to understand their job. Realistic simulated training using Virtual Reality in areas such as aviation has shown that workers can better understand the physical environment and tasks they have to accomplish, assuring a better understanding of the target system. This paper presents techniques used for a simulation system for an oil platform operation. Several interaction techniques were adapted and implemented to achieve a realistic operation scenario.**

Keywords-virtual reality; CAD models, walk-througt

I. Introduction

Simulations are frequently used in industry to create and verify procedures. Virtual Reality (VR) can be applied in this context allowing realistic visualization of any environment, besides the fact that the user can naturally interact on it in real-time. But one of the challenges of engineering simulations in VR is to deal with massive 3D models [1][2], which without a proper treatment may not furnish a real-time interactive experience. At the same time, the execution of complex simulations aimed to mimic a realistic behavior of the virtual model upon user interaction with virtual platform devices (valves, etc.), should be properly coordinated otherwise it will impair realism perception.

A current trend in virtual reality systems is the use of game development tools (game engines) to produce applications for industrial usage with high degree of interaction and realism. Thus enhancing the engagement of users using similar techniques and principles that are used to gaze the attention of game users. The segment of serious games is one of the main supporters of this type of technology, fostering the development of better tools.

The solutions developed in this research project are based on optimized CAD models viewed in real time in immersive environments. The optimized CAD geometry data is loaded on the Unity3D game engine[3], allowing browsing and interaction on the scene. A sample of a CAD model with millions of polygons is shown in Figure 1. It is possible to see that the large amount of geometric structures makes the interaction with this type of data very complex in virtual reality environments.

This document describes the challenges involved in the production of an optimized training system for an oil platform operation. Although, in real life, most of the tasks in an oil platform are remotely controlled, some tasks are not automated because are rarely executed and therefore need a person in loco to execute those tasks, operating certain devices. Therefore, a realistic navigation though the simulated environment and possibility of manual interaction were developed as part of the integration of the entire simulated training.

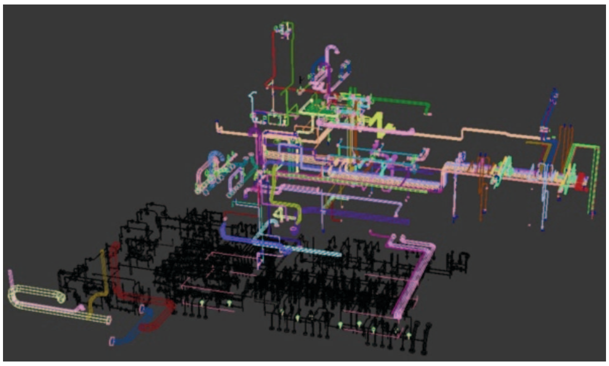

Figure 1: Piece of CAD model with 6,354,218 polygons.

Section 2 presents some related work with this research. Software architecture is explained in section 3. Section 4 presents topics about model treatment. Section 5 discusses about optimization with the help of textures. The way the user interacts in the immersive virtual environment is covered in section 6. Since the main objective is to use this training system in immersive environments, section 7 covers some details to achieve this goal. Section 8 presents an important topic to achieve the physical realism in an oil platform, which is the process plant's fluid simulation, as it describes the rules and dynamic behavior of the oil platform process plant. Conclusions are presented in section 9.

II. Related Work

Realistic simulated training using Virtual Reality in aviation area has shown that workers can better understand the place and the tasks they have to accomplish [4]. Although the topic of training people in operating oil platforms is important, and many centers in the world try to improve employees efficiency with specialized training [5], very little equivalent research was found using virtual reality to make employees more adapted for the complex procedures in an oil platform. Most of the related work found is oriented to safety and evacuation, and not to standard procedures of an oil platform [6][7]. Companies like EON [8] and laboratories like LITE [9] are developing commercial solutions for training, but unfortunately not enough details are published in order to understand and reproduce the simulation.

978-1-5090-4149-7/16 $31.00 © 2016 IEEE
DOI 10.1109/SVR.2016.18

The project presented by Brasil et al. [10] shows a simulation of workers in a drilling rig, training operational procedures in order to reduce the chances of accidents and failures through a continued training program offered to workers. For that, a game, which simulates a drilling rig daily operation, was implemented. Players are exposed to various circumstances in which they can exercise their knowledge about their work and learn what to do in troublesome situations, also allowing instant interactions among several players in the virtual 3D environment. One limitation of this tool is that the simulation is web-based, which can limit the use of immersive environments.

Pérez et al. [11] presented a virtual reality simulation of an oil platform. The system is oriented for desktop environments, lacking more information for an immersive environment. Actually, the authors presented a mixed training environment, using virtual and real scenarios, but the usage of an immersive visualization could provide a deeply and thoroughly understood of the simulation. Besides that, all the procedures can be trained in advance using the virtual reality environment. Although the limited coverage of possible usages, the work presents several implementation details.

Some works at the Hangzhou Dianzi University [12][13] are being developed for a more complete training in oil platforms, but these systems are not virtual reality oriented. However, all the numerical simulations can be used in an interactive virtual environment.

Rondomedia [14] team developed a software with a complete overview of an oil and gas operation platform. It is a game with several missions that put the user in control of operations, like locating your drilling site, controlling specialist floating cranes, flowing the oil from the platform to refinery. But since it is oriented for entertainment several realistic features are missing.

One of the reference systems used to understand the needs for operator trainer systems is the Ambtrei [15], which aims to improve operator performance by simulating scenarios such as emergency conditions, thus reducing accidents and increasing processes economical results. This system mimics the actual Control Room of a semi-submersible platform, but does not support virtual reality simulations in the automatic system.

III. ARCHITECTURE

Due to the complexity of the project, a special care must be taken since the beginning for accommodating expansion and connection with other services or solutions, which can be also external software developed by third party. The design decisions chosen for the software implementation are:

- Component based software development was used to easily change or incorporate new features;
- Access by web to the Graphical User Interface and ECOS (Supervision and Operation Control Station) interface;
- Multi-user support to allow interaction among trainees (operators) and instructor (or supervisor).

Figure 2 shows a diagram with architecture's main components and their interconnection. The three main components are:

- State Manager, responsible for keeping the notion of a consistent global state among all participants. It synchronizes all events and controls the distribution of the processes to all users. It can also store all the events and actions in the collaboration session and reuse it when necessary, for example in a late join in the session of a new user;
- Training Module, it traverses the graph that contains the training procedure, and controls a score for the user with correct and incorrect actions;
- Process Simulator Proxy, creates a bridge to an external process' plant fluid simulator such as Hysys [22].

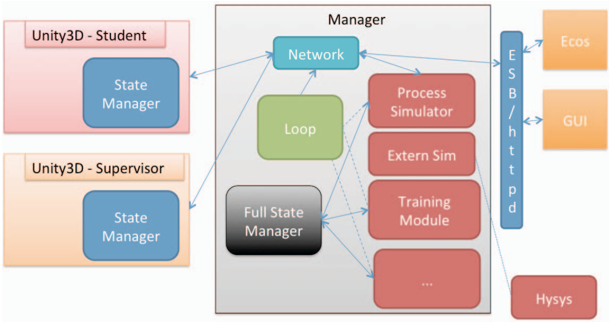

Figure 2: Software architecture.

The modules communicate among each other in real time, using a message passing mechanism. A modular architecture is important in the development process since it allows the replacement of old modules almost instantaneously.

IV. MODEL TREATMENT

In order to maintain higher frame rates for realistic visualization of complex 3D models, the geometry was carefully pre-processed for speeding up the render process. Modifications in the model are processed to create enough detailed models without sacrificing the interactivity of the simulation.

Engineering CAD models usually need some pre-processing to reduce the number of polygons. CAD tools usually create models of industrial plants with a very high density of polygons, that when imported into virtual reality tools, present serious problems and inconsistences, such as duplicate surfaces, degenerated polygons and incomplete mesh representation of objects [16].

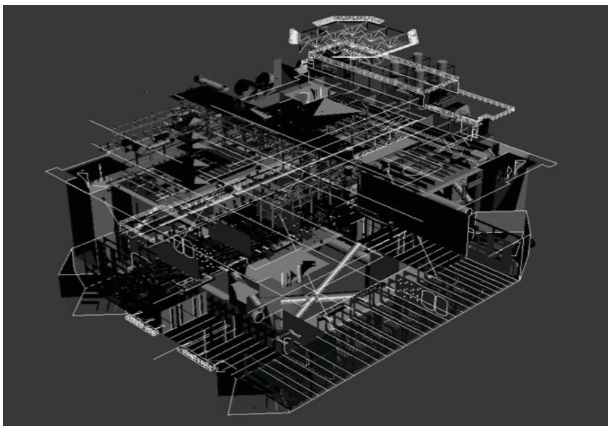

Figure 3: Original CAD model with 23,065,2620 polygons.

The oil platform used for tests in this research was originally modeled in PDS (Plant Design System) as presented in Figure 3, with some hidden parts of the model to better show its complexity.

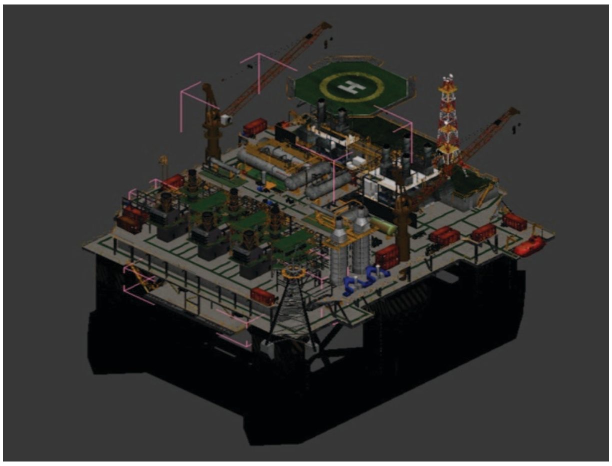

Figure 4: Complete VR model with 448,462 polygons.

After several graphics optimizations in the original model to make it usable in a VR environment, its complexity was reduced more than 50 times with significant improvement in realism, as shown in the model of Figure 4. Changing geometry meshes with specially tailored texture maps, which take into account information of the surrogate environment in a technique known as light mapping [17], produced a 3D model with a considerable improved realism. Nevertheless geometry optimizations were the most important optimization technique used so far. The main optimizations were, automatic decimation of complex and small parts, collapsing edges of the same objects, reorganizing geometry to minimize state changes during the render of polygons.

Improvements in this model were achieved by dividing the original model into smaller parts in order to perform the optimization process individually, once working with the entire model at once was very slow. After all the optimizations, the models are reintegrated to reestablish the rendering and verify internal collisions. For instance, during the optimization, two pipes can intersect each other due to some refinements in the flanges.

An important factor in modeling with few polygons, known as "lowpoly", is to replace some parts of the object with simple polygon with texture. The number of polygons allowed in each model will vary according to their visibility, interaction, number of polygons the graphics engine supports and the final platform being used.

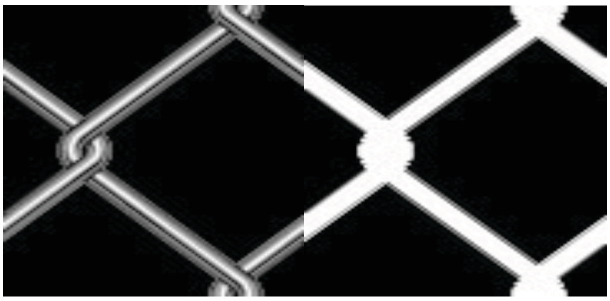

Figure 5: Texture of the wire fence and its opacity map.

One important example where optimization techniques using textures was used is for the construction of the wire fences on the platform (Figure 5). Wire fences are used very often in several areas of an oil platform, and modeling each wire with an associated geometry would severely compromise the performance of the entire model. For that, a texture map with color and transparency is used, being repeated side-by-side and top-bottom in a planar geometry. This process avoids the usage of a detailed geometry thus reducing the load time of the model with a more realistic model.

V. Optimizations

One of the major drawbacks of CAD models is the fact that engineers use only a mock object of the real one and not a more realistic representation of what is actually being used as an object. This is due to the fact that the engineers used to model the blue prints with simple representations of the devices, like valves, tanks, etc. But even for an assembling plan this kind of mock object is not good, because problems like the interference of different devices or ergonomic issues for accessing a specific location are not covered, reducing the potential of the 3D CAD tools. For example, Figure 6 shows on the left, a representation coming from the CAD model, on the right is a more realistic model of what is found in the installations indeed.

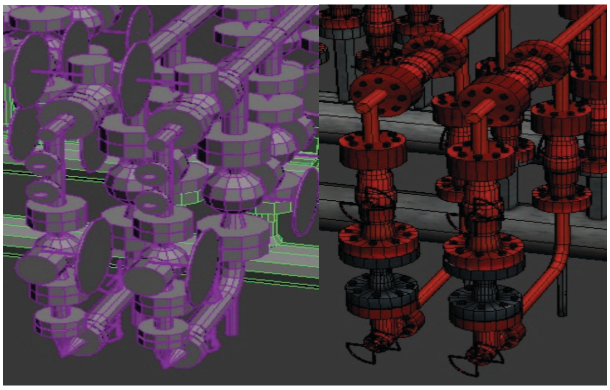

Figure 6: Original and optimized CAD model.

Besides the much more realistic visual appearance, the amount of polygons was reduced considerably in structures that repeat very often. For example, the original objects in the scene shown in Figure 7 have 253,340 polygons.

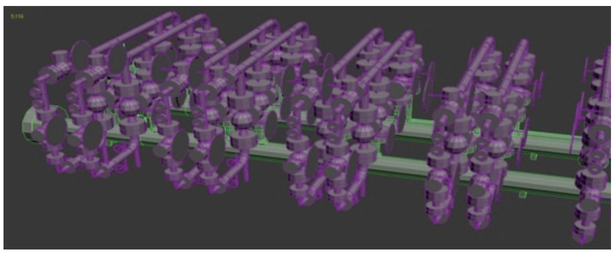

Figure 7: Valves at CAD model.

The same set of optimized objects, as shown in Figure 8, has only 110,023 polygons. More than half of the polygonal load in this scene was cut and better realism was achieved.

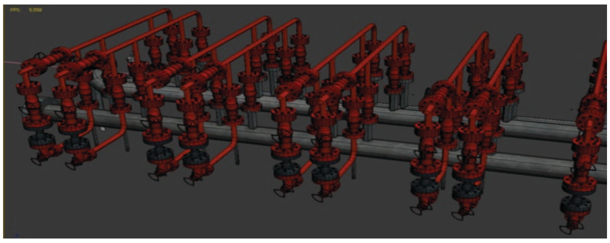

Figure 8: Valves optimized for VR.

Other point that is important for the simulation and interaction is to keep the correct representation of the elements in the CAD model. The elements in the CAD model have the same tag code, allowing the engineers to easily locate each element in blue prints and virtual world. This information must be replicated in the VR model.

A. Models Texture Mapping

After cleaning up the model, the next step is to map textures. For that, it is necessary to open the 3D mesh using the UVW coordinate system, used to define how textures are mapped on the surfaces by their vertices.

In order to make arrangements and mapping of the textures, a split mesh must be created. Although some tools allow splitting them automatically, a manual process appears to improve the final results. Nevertheless, much care should be taken because those split meshes usually compromise the quality of the maps, since there is no control of the mapping generated automatically.

Figure 9 shows several textures produced for some parts of the oil platform. In order to simplify the process, the UVW mapping is always the same for the same object to all the textures (color, alpha, bump, normal, specular).

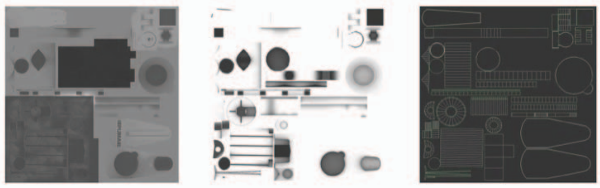

Figure 9: Different channels for texture effects.

A technique that uses texture mapping and produces better visual results with little impact on geometric processing is the use of Normal Mapping. A normal map acts as a relief map that adds surface detail in a lower polygon count. The maps are formed by a multi-channel data, then the color channels represent the X, Y and Z coordinate system. Each color represents a coordinate being displaced. This means that you can create a model with millions of polygons and high geometric resolution is saved in the format of a normal map that retains the spatial data that make up this model. This process leads to a much lower polygon count, with a very low processing overhead.

The texture maps used are no larger than 4096x4096 pixels, usually enough to several objects fitting on it. This minimizes the number of maps, saving the texture memory performance.

B. Global Illumination

An offline global illumination rendering is applied and saved on the textures. This process is known as baking, and it is used to improve the final rendering quality. Lighting effects are applied on the textures, like shadows, caustics and color blend, leaving the virtual environment much more realistic and detailed.

Other filter that improves realism is the Ambient Occlusion. All these effects are based on ambient light. Figure 10 presents an example of mapping textures to a CAD model. It is possible to see that a model with a much-reduced number of polygons may generate a representation rich in details.

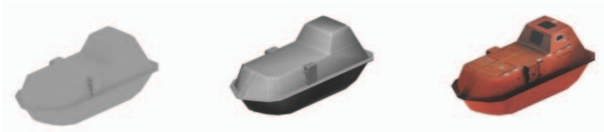

Figure 10: Lifeboat with 187 polygons.

VI. Immersive Environment Interaction

During a training section, the user has several tasks necessary to perform in order to accomplish a particular procedure. Although most of the oil platform devices are remotely controlled, some of them need a local intervention by the operator.

Besides moving inside the platform, like crouching and climbing ladders, frequent tasks in an oil platform simulation are opening and closing valves (and this involves rotating the valve in a specific direction), pushing buttons and grabbing different HSE (Health Safety and Environment) equipment such as velvet and lights, etc.

A. Object Selection

Bowman et al. [24] define that most interactions in virtual environments fit into three categories: navigation, selection and manipulation. Selection, or target acquisition task [25], refers to the task to acquire or identify a particular object of the whole set of available objects.

In this kind of simulation, the selection of objects is performed in environments with a large amount of details. Those scenarios are composed of hundreds of static objects and some other objects available to interact with. Since there are objects of various sizes, which can hinder the acquisition and identification of objects to be manipulated, a technique was implemented to assist the user in the selection task: drawing silhouette on the objects being interacted (Figure 11). The silhouette is drawn when the selection tool (e.g., mouse, wand) is placed over the object and the user is close to it.

The design of silhouettes is a method of non-photorealistic rendering [26], which aims to emphasize the details of the model. The same technique is used in cartoon rendering algorithms or edge detection techniques on several non-photorealistic rendering tools.

Figure 11: Silhouette for selected and manipulated objects.

B. Mini-Map

Navigation on oil platforms is a complex task not only due to their size but also due to the fact that each region is very similar to others, due to the replication process adopted during construction, which makes different places appear to be the same. The user can get lost very easily. The video-game technique of using a mini-map was applied in this simulation. In this case a second camera is placed just over the user's avatar and a second pass render with just the floor were rendered, giving a good indication about the user location as shown in Figure 12.

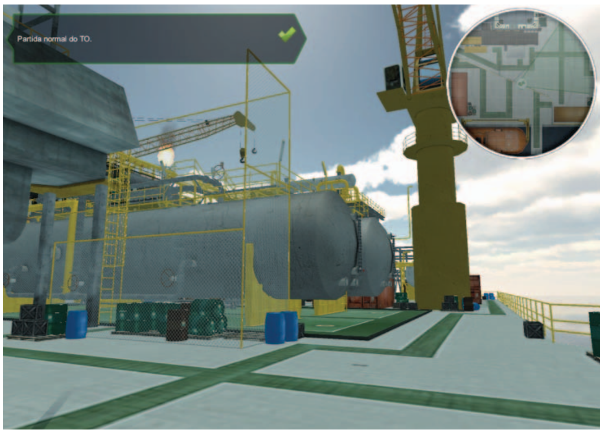

Figure 12: Mini-map showing the user's location.

VII. Immersive Visualization

One goal of the project is to conduct training in immersive environments with stereoscopy resources. After evaluation of many different alternatives for the projection system, a CAVE (Cave Automatic Virtual Environment) [19] was chosen because it is a very appropriate immersive environment for this kind of training systems.

In order to reduce the costs of the projection system, it was decided to use home projectors (nonprofessional), leading to different strategies for stereoscopy generation. One of the formats used for stereoscopy is the side-by-side [18]. In this format the stereoscopic images are formed by each eye's images placed side-by-side. By convention, the image is positioned to the left intended for the left eye, while the right image is intended for the right eye. This format is used in HDMI 1.4 systems, which is a recent format used for most modern projectors. The HDMI 1.4 is characterized by two viewing modes: unsqueezed and squeezed. In the squeezed mode, the images of the left and right eyes are squeezed horizontally and placed next to each other in a single frame, each image with half of the horizontal resolution. On the other hand, the unsqueezed mode keep their original size to be placed side by side in the final image, doubling the frame size to be transmitted.

The disadvantage the squeezed mode is the 50% loss of horizontal resolution while squeezing the pictures. The advantage lies in the transmission and handling of the final image, since the frame has the same resolution, making it easier to transmit.

Figure 13 shows the difference between the two squeezed modes. Two images of resolution 1920 × 1080 were squeezed to 50% of its original width. As a result, each image has a resolution 960 x 1080, resulting in a final frame of 1920 x 1080 when the images are added side-by-side. When projected the projector shows an unsqueezed image, where the image of both left and right eyes keeps on native resolution of 1920 x 1080.

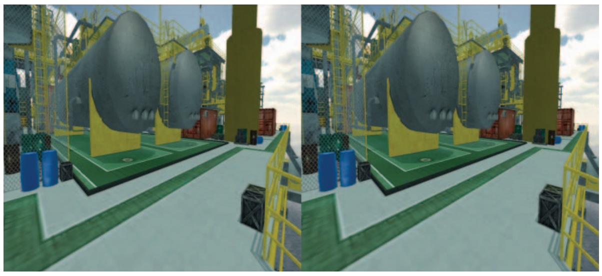

Figure 13: Render in a side-by-side.

Similar to side-by-side, the so called above/below format presents the same problem when squeezing the images, however this occurs vertically in this mode. For example, two images of 1920x1080 resolution are resized to 1920x540 to make up the final frame size of 1920x1080.

This squeezed format was used in the multi-display system presented on the next sub-section. Since the projectors only support this format, all the frames delivered must be prepared for that set of devices.

Immersive displays consist of systems that allow the user to visualize images with a high field of view. Usually these solutions are based on a small number of monitors or projectors that are placed side-by-side improving the feeling of presence.

The CAVE consists of a visual output device that has three or more projection screens, usually around 3 meters in width and height, surrounding the user. The main advantage of the CAVE is the fact that immersion feeling is increased by high visual stimulus caused from large projection surfaces. The challenge for executing VR applications in a CAVE is because it is a device consisting of multiple projection screens. Once the CAVE projection screens cover the users' field of view, it is necessary to generate an individual image for each direction.

Figure 14 presents the simulation running in a CAVE. In this image, three faces of the CAVE appear with the respective image. Since the picture was head-tracked when shooting this photo, it is possible to see a continuity of the image among the screens, the lines are basically straight not folding in the screen corners.

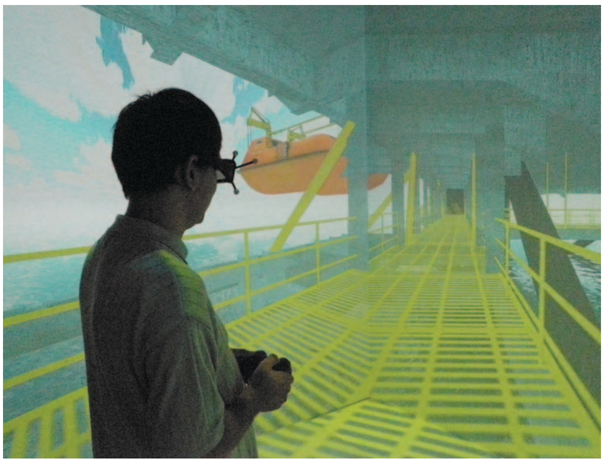

Figure 14: Oil platform simulated in a CAVE.

The CAVE is a very appropriate environment for this kind of training simulation. The user can navigate through the internal elements using the joystick integrated on the wand, and see the entire surrounding environment due the high field-of-view. The user can interact with a device and at the same time check what is happening around.

In order to help this development, some programing tools enable the port of desktop applications to immersive environments [20]. Since these multi-display applications usually run on a distributed system, it is also important to control the synchronization to keep the images coherent.

A. Audio

In order to improve realism, recorded audio from some machinery is spatially reproduced on the simulation. This audio can be positioned in different areas of the platform and this will reproduce the associated sound when the user gets closer. This audio simulation can be extended for immersive environments with several speakers allowing the user to have a correct idea where the sound is coming from [21].

VIII. PROCESS' PLANT FLUID SIMULATION

The Process' plant flow simulator is the component responsible for controlling the state of each device in the oil & gas process plant. Through the valves, remotely or locally operated in the virtual scene the user can interact with the simulated virtual plant, obtaining feedback of its interaction through the result of some measurement equipment, such as manometers, present in the virtual environment. Those virtual measurement devices have almost the same behavior as the real ones. This allows the trainees to verify if they are pursuing the operational procedure accordingly, once they are expected to keep the platform working properly.

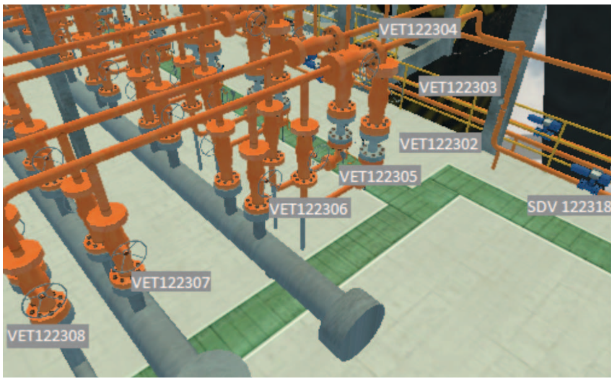

Figure 15: Valves and pipes available to interact.

In order to calculate the process' plant state, external software can be used for that, like the Hysys [22], and for that an OPC [23] bridge was ported. But for regular applications a simplified Fluid Simulation solution was adopted. From the topology of the platform process, the fluid simulator builds a directed acyclic graph (DAG), whose vertices and edges are the valves and pipes respectively. The edges are considered multivalued containing information of pressure, temperature and oil level. Figure 15 shows the valves and pipes available to interact with.

The fluid simulator first determines the source nodes, because it is through them that the simulation starts. Traversing the DAG, from the sources, the system simulates the passage of oil through pipes, giving each of them a specific value. Currently there are three pressure values: "no flow," "flow" and "blocked flow", these values were normalized to 0, 12 and 30 (bar); temperature for 0, 75, 150°C; 0%, 50% and 100% for oil level, respectively. For instance, a pipe with flow blocked (because a closed valve) will have pressure level equal to 30 bars.

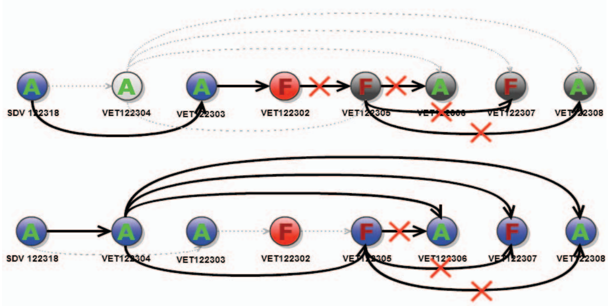

Figure 16: Graph simulated flow.

Figure 16 shows two ways a fluid can flow in a simple pipeline topology. When the valve is open (A) the oil goes to the next valve, otherwise when the valve is blocked (F) the flow stops in that point. A new flow simulation is triggered every interaction with a valve. Since this simulation is usually heavy and by consequence slow, there is a tolerance of half a second (0.5s) for an interaction to trigger a new simulation. This means that the position of the valves is computed every 0.5s. Then, if the user turns a valve for 3 seconds, 6 simulation steps are going to be computed, reducing an overload over the fluid simulation. This is a very simple algorithm that basically traverses the graph.

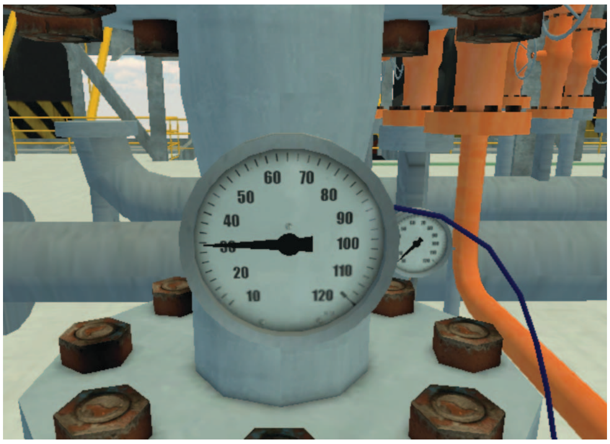

Figure 17: Simulated manometer.

The pressure levels of a pipe can be read through the valves that connect the pipes. The pressure levels can be read visually in the virtual scene, as presented in Figure 17, and even fluctuations can be analyzed since the pointer will swing.

IX. CONCLUSIONS

The techniques, presented in this document, allow the construction of a virtual reality training environment of an oil platform which is more interactive than the current training procedure used nowadays, which is based on photos and videos. Visual realism is one of the great challenges of this kind of system, since it is strongly demanded by oil&gas engineers, who needs several details in order to really have a training procedure that is not showing something wrong. This demand for realism and the necessity of real time visualization of large CAD models are challenges that had to be overcome. Therefore, an important part of the work in the system is devoted to the treatment of the geometric models, as shown in sections 3 and 4.

One of the main challenges was to create realistic ways to navigate in the virtual environment and to interact with the models, in order to reproduce the real procedures with high fidelity. Some aspects of this navigation were discussed in section 6. Regarding user output, multi-projected stereoscopic displays pose many implementation difficulties, as explained in section 7.

Finally, one of most complex challenges was the integration of the several pieces of the simulation, considering that the interactions on the simulated objects must provide input for the simulators (fluid for instance), and the results of the simulation provide input for graphics objects (values on changing manometers, for example).

A complete real oil platform with a reasonable level of details was modeled, running on immersive systems. Real procedures are being constantly tested on the system, allowing the software developers to continuously improve each part of the process. The software is scaling with the complexity increase in the geometry and the fluid simulation.

As a future work, because of the heavy processing requirements of the fluid simulations, the simulator will run on a dedicated computer, communicating through the OPC protocol[27]. Besides that, there are efforts on integrating the Microsoft Kinect on the current system, allowing a more natural interaction with the simulated platform devices, but one problem that needs to be addressed is how to use the Microsoft Kinect in large screens, since it is not planned for large immersive environment.

REFERENCES

[1] Raposo, A. B., Santos, I. H. F., Soares, L. P., Wagner, G., Corseuil, E., Gattass, M.. “Environ: Integrating VR and CAD in Engineering Projects”. IEEE Computer Graphics & Applications, v.29, n.6.

[2] S. Yoon, E. Gobbetti, D. Kasik, and D. Manocha, “Real-Time Massive Model Rendering,” Synthesis Lectures on Computer Graphics and Animation, Morgan & Claypool Pubs., 2008.

[3] M. Menard, “Game Development with Unity”, Course Technology PTR, ISBN: 1435456580, 2011.

[4] Stone, R.J., Panfilov, P.B., and Shukshunov, V.E, Evolution of aerospace simulation: From immersive Virtual Reality to serious games, Recent Advances in Space Technologies (RAST), 2011.

[5] P. O'Connor, and R. Flin, “Crew Resource Management training for offshore oil production teams,” Safety Science, Volume 41, Issue 7, August 2003, pp. 591-609, ISSN 0925-7535.

[6] R. Hubbold, and M. Keates, “Real-Time Simulation of a Stretcher Evacuation in a Large-Scale Virtual Environment,” Computer Graphics Forum, 2000, pp. 123–134. doi: 10.1111/1467-8659.00449

[7] W. Chun, L. Lei, Y. Jiwu, Z. Liangyun, and L. Gang, “Development of Emergency Drills System for Petrochemical Plants Based on WebVR,” Procedia Environmental Sciences, 2011, pp. 313-318.

[8] VIirtualMeeting, Case Study #1 Simulation Based Training, http://eonrealityblog.wordpress.com/2010/10/19/case-study-1-simulation-based-training/ , 2010.

[9] Louisiana Immersive Technologies Enterprise, Immersive Virtual Learning Environment (IVLE), http://www.lite3d.com/ivle/Oil-Rig-IVLE, 2011.

[10] I.S. Brasil, F.M.M. Neto, J.F.S. Chagas, R.M. de Lima, D.F.L. Souza, M.F. Bonates, and A. Dantas, "An Inteligent Agent-Based Virtual Game for Oil Drilling Operators Training," Virtual Reality (SVR), 2011 XIII Symposium on , vol., no., pp.9-17, 23-26 May 2011.

[11] B. Zayas Perez, M.M. Marin, and E.I. Perez, , "Developing a Virtual Environment for Safety Training," Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007.

[12] W. Feng, and H. Ping-zhi, “Design and Implementation of Simulation Training System for Oil Extraction,” Journal of Liaoning University of Petroleum & Chemical Technology, doi: CNKI:SUN:FSSX.0.2008-04-019, 2008.

[13] H. Hai-liang, X. An-ke, and Kong Ya-guang, “Design of simulation training system for oil extraction,” Mechanical & Electrical Engineering Magazine, doi: CNKI:SUN:JDGC.0.2008-01-008, 2008.

[14] rondomedia, "Oil Platform Simulator", Excalibur Publishing Limited, http://www.excalibur-publishing.co.uk/oilplatform.htm, 2011

[15] A.C. Pereiraa, A. Rierab, G. Padillab, E. Musulinb and N.J. Nakamurac, Operator Trainer System for the Petrobras P-26 Semi-Submersible Oil and Gas Production Unit, Computer Aided Chemical Engineering, Volume 27, 2009, Pages 1959–1964.

[16] A. Raposo, I. Santos, L. Soares, G. Wagner, E. Corseuil, and M. Gattass, “Environ: Integrating VR and CAD in Engineering Projects,” Computer Graphics and Applications, IEEE, Vol:29, Issue:6, 2009.

[17] Keshav Channa: Light Mapping - Theory and Implementation

[18] Lipton, L., “Stereo-vision formats for video and computer graphics”, in Proc. SPIE 3012, 239, 1997.

[19] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. Kenyon, and J. C. Hart, “The CAVE, Audio Visual Experience Automatic Virtual Environment” in Communications of the ACM, pp. 64-72, June 1992.

[20] L. Teixeira, D. Trindade, M. Loaiza, F. G. de Carvalho, I. Santos, and A. Raposo, “LVRL: Um Framework com Recursos RV para Aplicações Desktop.” XIV SVR, May 2012

[21] R. R. A. Faria, and L. F. Thomaz, L. Soares, B. T. Santos, and M.K. Zuffo, and J.A. Zuffo, “AUDIENCE–Audio Immersion Experiences in the CAVERNA Digital,” Proceedings of the 10th Brazilian Symposium on Computer Music, Belo Horizonte, Brazil, 2005.

[22] Hysys, http://www.aspentech.com/hysys/

[23] Object Linking and Embedding for Process Control, http://www.opcfoundation.org/

[24] D. A. Bowman, E. Kruijff, J.J. Laviola, and I. Poupyrev, “3D User Interfaces: Theory and Practice”, 2004.

[25] S. Zhai, P. Milgram, “Human Performance Evaluation of Manipulation Schemes in Virtual Environments”, in Proceedings of the 1993 IEEE Virtual Reality Annual International Symposium, IEEE Press, pp. 155-161, 1993.

[26] T. Akenine-Möller, E. Haines, and N. Hoffman, “Real-Time Rendering, 3rd edn”, AK Peters, Natick, 2008.

[27] L. Zheng, and H. Nakagawa, "OPC (OLE for process control) specification and its developments," SICE 2002. Proceedings of the 41st SICE Annual Conference, vol. 2, 2002, pp. 917-920.