



**HAL**  
open science

# Design of an annotation system for taking notes in virtual reality

Damien Clergeaud, Pascal Guitton

► **To cite this version:**

Damien Clergeaud, Pascal Guitton. Design of an annotation system for taking notes in virtual reality. 3DTV-CON 2017: 3DTV Conference 2017: Research and Applications in Future 3D Media , Jun 2017, Copenhagen, Denmark. pp.4. hal-01536680

**HAL Id: hal-01536680**

**<https://hal.science/hal-01536680>**

Submitted on 12 Jun 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# DESIGN OF AN ANNOTATION SYSTEM FOR TAKING NOTES IN VIRTUAL REALITY

*Damien Clergeaud, Pascal Guitton*

Inria & Université de Bordeaux, France

## ABSTRACT

The industry uses immersive virtual environments for testing engineering solutions. Annotation systems allow capturing the insights that arise during those virtual reality sessions. However, those annotations remain in the virtual environment. Users are required to return to virtual reality to access it.

We propose a new annotation system for VR. The design of this system contains two important aspects. First, the digital representation of the annotations enables to access the annotation in both virtual and physical world. Secondly, the interaction technique for taking notes in VR is designed to enhance the feeling of bringing the annotations from the physical world to the virtual and vice versa. We also propose the first implementation of this design.

**Index Terms** — Virtual Reality, Annotation System, Interaction Technique, Design, Tangible User Interface

## 1. INTRODUCTION

The industry uses Virtual Reality (VR) for a long time (CAD reviews, ergonomic studies or assembly task design). VR enables to iterate on a project before releasing the first physical prototype of a system. Moreover, VR can be used collaboratively for gathering coworkers from various remote sites in the same virtual environment (VE) in order to solve problems. When such a collaborative VR session happens, experts get a new point of view on their product. New insights may emerge which need to be saved for reviewing the design later. Users can directly save those insights in the VE. Thereby, they can use various kind of media for describing their ideas (text, image, video, etc.). For instance, users can take a picture of an object and then draw on it to show which part need to be manipulated after.

A lot of techniques in the literature enable to save information in VR. Thanks to the digital properties of this space, users can directly annotate the VE by using any media (text, image, video, etc.). But none of those solutions take into account the need to easily export those annotations outside the VE. In industry, VR sessions take part of a more global process. The VE need to be designed for easily access what was decided before the session and for easily save and export what will be solved or decided during the session.

Until now, users need to use VR application or a light client (3D Desktop application) to access the VE in order to extract annotations. This is the main drawback of those techniques.

In this paper, we propose a new approach to design annotation systems (AS) used in collaborative VR sessions in order to better include them in a global industrial process. In the next section, we recall some related work that inspire us. Then, we present the design our annotation system and the implementation of our first prototype. Finally, we conclude and present future work.

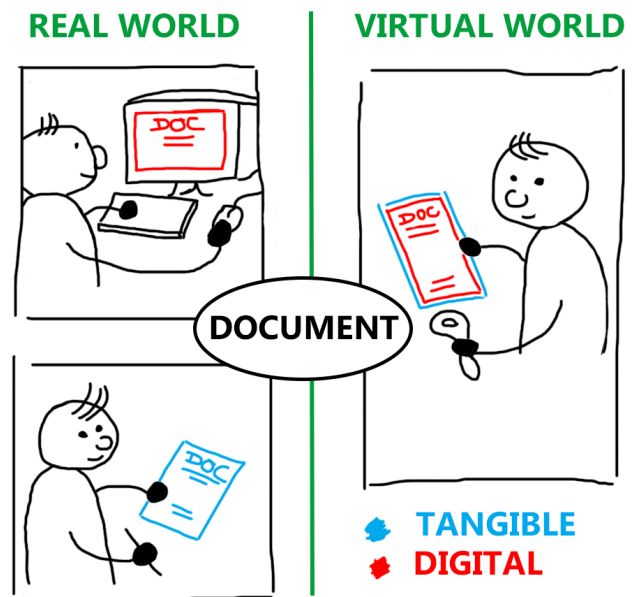


Figure 1. We propose to provide a new annotation tool in order to be able to access/edit usual digital documents in VR.

## 2. RELATED WORK

The virtual world is dynamic and enables to link abstract information to any virtual objects. Bowman and al. defined those VE as Information-Rich Virtual Environments [1]. They noticed that it could be a good way to share/teach knowledge about something by immersing a student. For instance, they developed a simulation where students were able to learn about gorillas habitat at the Atlanta Zoo. Students could access information related to the design of the exhibit and related to gorillas.

Poupyrev and al. proposed the Virtual Notepad [2]. The Virtual Notepad is a tablet and a pen synchronized with a VE. Users could directly write and draw in VR. This interaction technique is directly inspired by real notepad.

A lot of annotations systems in the literature enable to edit/create textual [3, 4], vocal [5, 6], drawing [2, 7] or even photo and video annotations [4, 8]. Some AS provide structured annotations in order to build and keep track of the decision process [4]. Each annotation is structured as a thread on a forum. All those AS, the VE is the central piece of the collaboration. People discuss and make decisions through the VE representing what gather them. Pick al. [8] designed a collaborative VE that integrate immersive VR for factory planning processes. The system allows multiple users using an interactive table to design the layout of a factory. Users could also use VR for getting a more immersive point of view of the layout. Thus, users could access and edit the annotations created/edited on the interactive table directly from

VR. All virtual objects of the VE were stored in a database. The table client and VR client were just synchronized views of this database. Other systems use the same solution to enable different modality of access [4]. The main drawback of this kind of solution is that users need a 3D client to reach the information. So, they have to come back in the VE for recovering the information. The DIVE system [3] proposed to save notes by writing it on a "notebook". Notes were directly stored in a text file. Therefore, users were able to keep those notes after the session. On the other side, they were limited to use textual annotations in this "notebook".

### 3. DESIGN OF OUR ANNOTATION SYSTEM

#### 3.1. Go on Expedition

The starting point of our design was to find how did industrial experts work before using VR. After creating solutions during a meeting, experts went to the assembly line to visit the product and to test their solutions. Then, they were meeting again to discuss the several insights that had emerged during this test.

Finally, VR sessions have just replaced those field trips. Access to VR sessions during an industrial process is just going outside the meeting room to go somewhere else. Thus, in order to design our annotation system, we needed to understand how people take notes during an expedition.

There are two kinds of annotation that people use when they do an expedition:

- Annotation in the environment: Annotation in the environment are generally for navigation purpose. If you let a road sign during your expedition, it is essentially for helping the next person that come after you at this same place. You could also let information to explain some properties of an object in the surrounding.
- Annotation on a notepad: People take notes on a notepad to bring it back with them. Those annotations are observations, drawing, pictures, insights, etc.

To conclude, annotation in the environment are made for people that will come after you to this environment. On the other hand, annotations on a notepad are made for keeping this information to exploit it in the future.

Then, if we want that industrial experts take benefits of VR sessions, we have to provide them a tool that they can use both in VR and at their office in order to exploit them.

To design this AS, we divided the task in two parts. First, the information must be accessible from VR and the physical world. Thus, the way to store information should enable multiplatform accessibility. Then, we focused on designing a tool for accessing the information. For design purpose, we decided that the interaction technique must be close to the interaction technique in the physical world.

#### 3.2. Information Management

The better way for making information accessible from various places or tools is to separate the internal representation of information from the ways to present it. Model-View-Controller (MVC) is a software architectural pattern for that. Pick and al. used this model for implementing several clients to access to the same VE [8]. In this case, the VE is at the center of the collaboration. In our case, the information must be at the center.

During projects, coworkers communicate with documents. Those documents are usually PDF or Word-like document. So, we decided to use the XML-based document for taking notes in VR.

The benefit to using this kind of document is that users can read and edit it with usual word processor tools. Users can even print it to read it in a comfortable way. We propose to design a tool that provides a new way to interact with those usual documents in VR.

In this case, the document is used and prepared before going to VR sessions. In the industry, the document is usually made of technical specifications, some procedures for assembling the system, etc. Then, the description of the task procedures for the expedition in VR can be added to this document. This procedure guide the users in the VE.

The document can also be used as a backup. It can store information about the VR session without being displayed. The creation time or the position of an annotation in the VE are useful information. It can be used to replay the annotation process of a user in order to understand the decision process of a solution.

Since we know how to manage the information in our AS, we present the interaction technique that enables to access the information in VR in the next section.

#### 3.3. Virtual Notepad

To use documents in VR, we need to design an interaction technique in order to allow users to access or edit it. We decided to extend the Virtual Notepad [2] proposed by Poupyrev and al. as mentioned in the Related Work section.

The main benefit of this interaction technique is the tangible property which enables users to hold it physically in their hand. Thus, the feeling of bringing the document in VR environment is reinforced. It also makes the interaction technique easier to learn.

Although, holding the notepad constrained users if they want to use their both hands freely. In this situation, they can not use their controllers, that are the most common way to interact with the VE. Putting on the floor the notepad or the controllers depending on the situation is not a good solution. It asks useless effort to the users.

In order to solve this design problem, we took inspiration from the virtual world. In VR, virtual objects can be used without gravity. Users can let floating a virtual object in the air in order to manipulate another object and go finally back to the first one after. This property of VE is comfortable for users. It would be great to be able to do the same with the tangible Virtual Notepad. Our assumption was that users should not be confused about a floating tangible object. For instance, in the International Space Station (ISS), astronauts are used to interacting with objects by letting them float in the air.

At this time of the design process, it remains a problem. Even if the Virtual Notepad can float, it could constrain the user's movement. Since it is tangible, users can hit by realizing some manipulation tasks. Always inspired by the properties of virtual objects, this problem is solved if the Virtual Notepad can be displayed or not according to our needs (as a virtual menu that we active or not).

So, to conclude this design section, we decided to extend the Virtual Notepad with properties inspired by virtual objects. We assume that giving virtual properties to a tangible object used in VR can reinforce the feeling of presence of the users.

## 4. IMPLEMENTATION

#### 4.1. Apparatus

For the first prototype of this system we used a HTC Vive HMD with an Optitrack optical tracking system. The optical tracking system are used to tracked the position of the tangible notepad.

The HMD and the controllers are tracked by the HTC tracking system. The simulation is developed with Unity 5.5<sup>1</sup>. We used the SteamVR<sup>2</sup> plugin and the VRToolkit framework<sup>3</sup> to implement the interaction technique. We used a gaming performance computer (Intel Core I7 @ 4.0 Ghz, 16Gb RAM, 6Gb Nvidia GTX 1060).

## 5. IMPLEMENTATION

In this section, we present the implementation of our AS. We developed it for an industrial context. Thus, the basic scenario was:

- Bringing the document in VR. The document contains the task procedure to follow in VR. This content has been added by using a computer.
- Filling the document with annotations. Users must be able to express their insights in the document thanks to the AS.
- Bringing back the document at the office. The document with the annotations is accessible by using a computer.

Now, we present the apparatus and then the implementation of the XML document.

### 5.1. Implementation of the document

For implementing the design of the document, we decided to create an XML format according to the scenario. An XML format is easily transformable by using an XSLT file. Basically, an XSLT file enables to translate an XML format to another XML format. For instance, we can transform a specific XML format to a PDF format by using a specific XSLT.

So, we design the format of our XML document to easily describe an industrial procedure. A procedure is a set of tasks that can be nested or juxtaposed. In this first design, a *Task* is an XML node. At least a *Task* has a *Title* and a text *Description*. A task could be divided to several task. Thus, each *Task* node could have child *task* nodes.

We also created an XML node called *Note*. Each task could have many *Note* nodes. Notes are added to a task in the order of creation. Each *Note* has a *Type*, an *Author*, a *Creation Date* and *Content*. The *Content* depends on the *Type* of the *Note*. An example of a *Task* node with a child task and a note is shown below.

```
<task>
  <title> ... </title>
  <description> ... </description>
  <task>
    ...
  </task>
  <note>
    <type> ... </type>
    <author> ... </author>
    <creation-date>...</creation-date>
    <content> ... </content>
  </note>
</task>
```

In VR, the document is displayed by using UI interaction elements in Unity. Basically, each *Task* and each *Note* nodes are a UI Button.

<sup>1</sup><https://unity3d.com>

<sup>2</sup><https://www.assetstore.unity3d.com/en/#!/content/32647>

<sup>3</sup><https://vrtoolkit.readme.io/>

### 5.2. Adding annotation to the document

The document is displayed in the VE. In order to add a note to a particular task, users have just to directly select with the virtual ray or their virtual hand the task on the notepad. In this way, they are setting the context to this task. Every note that users will create will be attached to this task until they select another task on the notepad.

The annotation is contextual. Users can add notes to the document by selecting a task and then create a note. In this way, the note is directly linked to the currently selected task. For selecting a task, the document must be displayed.

For this first prototype, we focused on some annotation techniques that seems necessary. We implemented various ways to create contents:

- Text : Text is the most basic way to share content on a document. In order to provide text support to immersed users, we integrated a virtual keyboard. This technique is not efficient for writing text, but it enables users to write some words in order to remember an idea quickly.
- PostIt: PostIt is a surface artifact that you can put where you want and draw/write on it what you want. They can draw directly on it with the virtual hand. They can manipulate it for positioning it in the VE. A copy of the PostIt is directly added to the document, updated each time users modify it. The position and orientation of the PostIt are saved in the document but is hidden to the user.
- Picture/Photo: Users can take a snapshot of the surrounding by using a virtual camera. The picture is directly added to the contextual task. If users want to draw on this picture to highlight something particular, they can open it by selecting it in the document. Then, they can directly draw on it with the virtual hand. The picture on the document is directly updated after any modification by the users.
- Video: Users can take a video snapshot of the VE by using a virtual camera. The video is directly added to the document.

We also added techniques to directly annotates in the VE. Since users are immersed in a digital world, it is easy to modify the surrounding without destroying it instantly. And some basic techniques are necessary in order to provide more expressiveness to users for sharing or saving their insights. We enable users to draw directly:

- on surfaces: any surfaces of virtual objects are drawable to highlight some part of virtual objects in the VE.
- in 3D space: Users can directly draw in space (inspired from Google Tilt Brush). So, they can sketch 3D models by directly draw the outline of a shape.

This kind of annotations are added directly to the document without being visible.

Drawing annotation belongs directly to the surrounding space. In that case, it does not make sense to make it visible in the document. So, Users need to take a picture or a video of the annotation in the surrounding if they want to add it to the document. Although, the drawing annotation is stored in the document to save it.

### 5.3. Implementation of the Virtual Notepad

For implementing the design described in the design section, we used a plexiglass board for playing the role of the Virtual Notepad.



Figure 2. Picture of an user using the Virtual Notepad.

An optical tracking system tracks it. In the simulation, the document is attached to the position of the board.

For the floating and disappearing properties, we took inspiration from the live show. In a theater play, the stage design is usually adjustable. One of the techniques used is to attach some piece of the stage with black rope. It can give the illusion of an object flying, or it can hide an object by pulling it against the ceiling.

So we decided to suspend the Virtual Notepad with a rope. We attached the tangible interface to the ceiling with a pulley. The pulley is used to move the tangible virtual notepad to make it reachable or not by the user. When users want to make the notepad disappear, we just need to pull on the rope for making the notepad unreachable.

For the controllers, we implemented a similar system to enable users to take the Virtual Notepad. We attached the controllers to the belt of the user with a retractable link. This system is inspired from the tool belt of an electrician. A belt is a good place for sorting tools. Since users can't see their body, it was necessary to attach the controllers with something retractable in order to avoid making them fall.

## 6. DISCUSSION AND FUTURE WORKS

In this paper, we propose an interaction technique that enables to access and edit common digital document in VR. Using a document based on XML enables to make the information easily transit between the virtual world and the meeting sessions. Inspired from the Virtual Notepad [2], our interaction technique unifies the notepad metaphor (tangible user interface in VR) and the nature of how the information is stored (XML document).

We designed an interface close to reality, but the way to take notes is based on virtual tools. Currently, the document is like a dynamic menu in the VE. Paper or Word-like document are not just dynamic menus. A simple way to improve this technique can be to add a tangible virtual pen to the notepad. This pen could be stuck to the notepad with a magnet. Then, users could use this pen in order to draw or write on the document.

Until now, the suspension of the Virtual Notepad is controlled by an operator (Wizard-of-Oz method). But, we can imagine replacing this implementation by an automatic electronic system controlled by the simulation (Arduino and serial communication).

The current design of the document can be extended to support collaboration scenarios (like Google Doc for instance). Until now, since the author is stored in each modification, it supports asynchronous collaboration. For supporting synchronous collabora-

tion, we must provide tools that enable to edit the document at the same time.

Furthermore, The design is not attached to a particular implementation. We can imagine developing the same design in a CAVE. In a CAVE, the Virtual Notepad could be a tablet. For suspending the tablet, it will be difficult to hide the rope in a CAVE.

## 7. CONCLUSION

In this paper, we proposed a new approach to design an AS. The primary purpose of the design was to unify how information is used in an industrial context and how users interact for taking notes. We decided to use XML format to manage the information between the virtual and the physical world. For the interaction technique, we extended the Virtual Notepad technique by adding two new properties: floatable or not and displayable or not. We also presented our first implementation of this design.

We are currently working on enabling synchronous collaboration on the document. It can become a powerful tool for facilitating collaboration. Usually, industrial experts work as a team.

Finally, we also think that applying the design approach described in this paper could benefit to create or extend other interaction techniques in VR.

## 8. REFERENCES

- [1] Doug A Bowman, Chris North, Jian Chen, Nicholas F Polys, Pardha S Pyla, and Umur Yilmaz, "Information-rich virtual environments: theory, tools, and research agenda," in *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, 2003, pp. 81–90.
- [2] Ivan Poupyrev, Numada Tomokazu, and Suzanne Weghorst, "Virtual notepad: handwriting in immersive vr," in *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*. IEEE, 1998, pp. 126–132.
- [3] Emmanuel Frécon and Anneli Avatare Nöu, "Building distributed virtual environments to support collaborative work," in *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, 1998, pp. 105–113.
- [4] Tiago João Guerreiro, Daniel Medeiros, Daniel Mendes, Maurício Sousa, Joaquim A Jorge, Alberto Raposo, and Ismael HF dos Santos, "Beyond post-it: Structured multimedia annotations for collaborative ves.," in *ICAT-EGVE, 2014*, pp. 55–62.
- [5] Reid Harmon, Walter Patterson, William Ribarsky, and Jay Bolter, "The virtual annotation system," in *Virtual Reality Annual International Symposium, 1996., Proceedings of the IEEE 1996*. IEEE, 1996, pp. 239–245.
- [6] J Verlinden, J Bolter, and Charles van der Mast, "Voice annotation: Adding verbal information to virtual environments," in *Proceedings of the European Simulation Symposium, 1993*, pp. 60–69.
- [7] Thomas Jung, Mark D Gross, and Ellen Yi-Luen Do, "Annotating and sketching on 3d web models," in *Proceedings of the 7th international conference on Intelligent user interfaces*. ACM, 2002, pp. 95–102.
- [8] Sebastian Pick, Sascha Gebhardt, Benjamin Weyers, Bernd Hentschel, and Torsten Kuhlen, "A 3d collaborative virtual environment to integrate immersive virtual reality into factory planning processes," in *Collaborative Virtual Environments (3DCVE), 2014 International Workshop on*. IEEE, 2014, pp. 1–6.