Narcís Parés
npares@iua.upf.es

Roc Parés
rpares@iua.upf.es
Audiovisual Institute, Universitat Pompeu Fabra,
Pg. Circumval. lació, 8
08003 Barcelona, Spain
www.iua.upf.es/∼gvirtual

Interaction-Driven Virtual Reality Application Design

A Particular Case: El Ball del Fanalet or Lightpools

Abstract

Virtual reality application design is usually guided by a content-driven strategy, which gives priority to the application's content and context. In this paper, we shall describe and study a novel strategy in VR application design that is centered on the design of the user interaction, regardless of the specific content of the application. This is especially useful in creative/artistic applications of VR. We shall present the specific case of an artistic VR application from which this strategy has emerged. This VR experience, *El Ball del Fanalet or Lightpools* has been successfully presented at the Miró Foundation in Barcelona (Spain).

1 Introduction

The development of VR applications has undoubtedly evolved from the development strategies of computer applications in general. The phases of the development seem clear when the content or the topic on which the application is based drives the whole development process (Kalawsky, 1993; Waxelblat, 1993; Loeffler & Anderson, 1994; Bryson, 1995; Earnshaw, Vince, & Jones, 1995). The topic defines a context, and the context is determinant in the choice of the metaphor used in the interaction elements and interface (Erickson, 1990). These strategies, which we call *content-driven* strategies, are naturally suited to the development of VR applications from a wide range of ambits: from scientific simulations to architectural designs.

Nevertheless, the content-driven approach exhibits certain limitations when experimenting with new approaches to interface design and exploring specific qualities of VR as an art production medium or an audiovisual communication medium (Parés & Parés, 1993, 1994). In this paper, we will define the development phases of a general VR application as we understand them and then go on to state our definition for content-driven strategy as the commonly used strategy. We will then define the *interaction-driven* strategy and compare it to the content-driven strategy, followed by a real example of interaction-driven development in an artistic VR experience. Finally, we will extract some conclusions from the differences between the two strategies.

Presence, *Vol. 10, No. 2, April 2001*, 236–245
© *2001 by the Massachusetts Institute of Technology*

2 Development Phases of a VR Application

We now will describe what we consider to be the most important phases in the development of a VR application regardless of the development strategy, in order to study later how these phases are organized differently depending on the application strategy that is chosen. These are the phases involved in simulation loop design, the interfaces (physical, logic, and mapping), object modeling, object behavior design, and stimuli design.

2.1 The Simulation Loop

The simulation loop is the structure of a VR application that repeats a series of actions for the duration of the application's life. This repetition of action may determine the time measure of the application by defining cycles of a certain length, or it may be adapted to a specific internal clock that determines the time evolution of the application. These actions manage the control of the application at every cycle, that is, at every instant in time. Its essential parts are

- **Sensor management:** read the input data from sensors detecting user movements and decisions.
- **Renew user state:** modify the general properties, situation, and status of the virtual subject, according to the data captured by the sensors.
- **Renew object state:** modify the general properties, situation, and status of the virtual objects, from their behavior algorithms and the decisions that these entail.
- **Determine interactions:** detect whether the modifications of the virtual subject and the virtual objects lead to any interactions that can produce a change in state of the application and its elements.
- **Management of the sensorial outputs:** generate all the stimuli that the experience is composed of (such as images, sounds, movement, and touch) in order to give coherent feedback to the user according to the instantaneous context.

The simulation loop is structured around a set of important development phases that include identifying the types and sources of data used and the processes involved in the application.

2.2 The Interfaces

Interface design is extremely important because it defines a two-way communication between the user and the experience, allowing the user to perceive and modify the experience.

The stages of interface design can be divided into three essential parts:

- deciding which external channels will communicate with which internal channels and how (mapping),
- determining which elements will be the external application links (physical interfaces), and
- determining which elements will be the internal links (logic or software interfaces).

A very simple example is the combination mouse-cursor in a standard Windows environment. The physical interface is the mouse itself; the software interface is the cursor; and the mappings are the relation between mouse-movement units and cursor-movement units, the relation between the pressing of a mouse button and the cursor's reaction selecting a certain object, and so forth.

2.2.1 Mapping. *Mapping* refers to the way in which the user's sensorial channels are linked with the actions that s/he controls and the stimuli s/he receives in the VR experience. These mathematical links let us define what will alter what and how much. This not only changes how the environment reacts to the user, but also modifies his/her perception of it. For instance, in the mouse-cursor example, the mapping between mouse displacement and cursor displacement is crucial for practical reasons. That is, if a large displacement of the mouse is needed to move the cursor one screen unit, it might make the task of moving screen objects too slow. On the other hand, if a small displacement of the mouse makes the cursor move a large screen distance, then the task of moving screen objects might be impossible because of a lack of control. These two extreme cases not only describe practical interaction issues but also affect how the user perceives the screen space (world). In the first case, the world seems very large in relation to the user, because it takes a great effort to move from one place to another, whereas, in the second case, the world seems extremely small, because the user moves very quickly from one end to the other.

2.2.2 Physical Interfaces. Mapping requires data input and output through a set of devices that physically connect the user's output channels to a software interface inside the application, and data output from the application to the input sensorial channels of the user through a set of displays (not only visual displays). These are the physical interfaces.

2.2.3 Software (or Logic) Interfaces. In simulation applications in general, the software interface is associated with the metaphor chosen in the application context and is a virtual representation of the physical interface (for example, a virtual hand when a dataglove is used or a virtual steering wheel in a vehicle simulator). An interesting exception to keep in mind is the mouse-cursor example, in which the cursor has no relation whatsoever either with the mouse or with the desktop metaphor. As we shall see, this idea of a software interface that is not a direct simulation is enhanced by the new application-development strategy that we are presenting, and opens a wide range of possibilities for audiovisual communication and artistic VR applications.

2.3 Object Modeling

Object modeling is an important phase in the development of a VR application because objects must be carefully tuned (polygon count, texture size, and so on) to meet the computing-power constraints, the visual requirements, and the defined interaction. In simulation applications, one may choose to use highly tuned objects from object libraries, but, in artistic applications, one may not be able to use predefined objects.

2.4 Object Behavior

Object behavior is a complex task both because of its inherent difficulties (such as intelligence, reaction, and spontaneous actions) and because of the lack of general and homogeneous tools to develop the behaviors. A similar issue is found here as in object modeling: that is, simulation applications may use behavior libraries, whereas artistic applications probably have to develop specific behaviors.

2.5 Stimuli

Both generated and captured stimuli must meet certain coherence conditions with respect to the human user. Undesired interactions between stimuli may cause instability and discomfort in the user (McCauley & Sharkey, 1992)—for example, if the application does not offer kinaesthetic feedback in accordance to the visual feedback. This makes it an especially delicate phase in the VR application-development process. Artistic applications might ignore these coherence conditions in their search for new relations with users' mental and physical models.

3 Content-Driven Strategy

Now that we have introduced the development phases for VR applications, let us turn to analyze how these phases are arranged when the content drives the development.

Content-driven strategy may also be referred to as a *top-down* strategy, because it starts at a high abstraction level, analyzing the content and context of the application, and works down to the implementation of it. The sequence of stages that we define in this strategy—including implementation, problem evaluation, and user requirement stages—are the following.

- **Define the application topic or theme:** such as aircraft pilot trainer, laparoscopic surgery trainer, aerodynamics analyser.
- **Define the type of application:** explorative, ma-

nipulative, or contributive interaction (Parés, 1999).

- **Identify the type of user:** expert, certain knowledge of the topic, general public, and so on.
- **Identify the necessary virtual objects:** landscape objects (vegetation, buildings, etc.), internal organs, wing model, etc.
- **Identify the data involved:** such as input data, partial results, final results, and data types.
- **Identify processes:** such as algorithms, computations, behaviors, and subjacent laws.
- **Identify input interfaces:** such as type of sensors needed, user interfaces, links, and mappings.
- **Identify output interfaces:** such as results presentation and type of display peripherals.
- **Identify object-modeling tools needed:** geometric modeling, generative algorithms, CAD tools, scanner inputs, etc.
- **Identify application development tools:** such as programming libraries, developing environments with GUI, and sensor drivers.

This type of development strategy is naturally suited to the analysis of almost any computer application that is based on a certain content (theme or topic). Note that some of these stages are not necessarily sequential, but they may be analyzed in parallel with others next to them in the list.

4 Interaction-Driven Strategy

Although most VR applications would follow the content-driven strategy, there are certain cases in which it can be very useful to develop an application by concentrating on how the user is to interact with the application. In other words, analyzing the interfaces, interaction with the elements, and participation/manipulation/contribution of the user, in such a way that the obtained results allow a spontaneous emergence of, for example, the specific topic, content, "aroma," and tone, of the application.

Such may be the case when the developers are interested in testing certain interaction theories through a series of experiments. The resulting experiences are contextualized only when the application has its final format to be presented to the users. Another case in which this might be useful is artistic approaches in which the artist is trying to convey concepts and relations through the interaction proposed, as opposed to trying to propose a certain topic or aesthetic value.

This interaction-driven strategy may also be referred to as *bottom-up* strategy, not because it is the exact opposite of the content-driven (or top-down) strategy, but because the development begins with identifying the interfaces, sensors, and mappings, and, at the same time, processes and behaviors, instead of starting at the definition of the application topic.

Therefore, the development stages for the interaction-driven strategy are rearranged in the following order.

- **Identify input interfaces:** such as type of sensors needed, user interfaces, links, and mappings.
- **Identify output interfaces:** such as results presentation, type of display peripherals, and mappings.
- **Identify the type of user:** such as expert, certain knowledge of the topic, and general public.
- **Define the type of application:** explorative, manipulative, or contributive interaction.
- **Define the application topic or theme:** find a related metaphor by analyzing the type of interaction.
- **Identify processes:** such as algorithms, computations, behaviors, and subjacent laws.
- **Identify the necessary virtual objects:** objects related to the interaction and the metaphor, although they may be completely abstract.
- **Identify the data involved:** such as input data, partial results, final results, and data types.
- **Identify object-modeling tools needed:** geometric modeling, generative algorithms, etc.
- **Identify application development tools:** such as programming libraries, developing environments with GUI, and sensor drivers.

This strategy facilitates the design of software interfaces that do not necessarily represent physical interfaces because there is no a priori content that determines them. This also opens the possibility of defining mappings that do not respond to models extracted from our physical surroundings.

The freedom in the design of both these elements of a VR application enhances the development of innovative interaction, even to the extent of designing experiences in which the stimuli, which are fed to the user, do not conform to coherent systems in the sense that a content-driven application would force.

Here, the metaphor and the application context for the interface should emerge from a close study of the proposed interaction, which sketches analogies traced from this interaction to the context from which the metaphor is extracted.

5 El Ball del Fanalet or Lightpools

El Ball del Fanalet or Lightpools (Hoberman, Parés, & Parés, 1999), an artwork developed by Perry Hoberman and Galeria Virtual (Parés & Parés, 1995), follows the interaction-driven strategy. In fact, the development of this piece made us aware of the existence of this new strategy. We will first explain what the piece is about in its final format. Although this might mislead the reader in thinking that it is a content-driven application, we will later describe its development stages, which have led to the structuring of the interaction-driven VR application-development strategy.

5.1 Description of the Piece

El Ball del Fanalet or Lightpools is a multiuser experience that uses virtual reality technology. It takes place in a circular arena approximately six meters in diameter, onto which a real-time, computer-generated image is projected from above (figure 1). Each user is given one of four *fanalets* (a paper lantern typical of a Catalan popular dance) (figures 2 and 3) as they enter the arena, and each contains a colored light with a battery pack and a position sensor. The sensor reports its position to a host computer through ultrasound pulses, allowing each fanalet's position to be tracked in three-dimensional space. The two horizontal dimensions are used to position a colored circle of light projected onto the floor directly below each fanalet. The third dimension (height) is used to determine the size and brightness of the lightpool, so that its behavior mimics the effect of a light source emanating from the fanalet: as the fanalet is lowered, the pool becomes smaller and more intense; as it is raised, the pool becomes larger and dimmer. The users thus have the impression that the lightpool is projected directly from their fanalet, and gives them an immediate, intuitive sense of how to interact with the work.

Each lightpool is a kind of window onto a virtual ground plane that otherwise remains shrouded in darkness. Small, colored tetrahedrons (proto-objects) are spontaneously and randomly generated throughout the arena, glowing briefly like embers before disappearing. Each proto is matched in color to one of the four fanalets (and lightpools). If a user manages to illuminate a proto (during its brief life) with the appropriate lightpool at a sufficient intensity, the proto grows and metamorphoses into an articulated object. In effect, the object appears to feed on light. The objects range from mechanical to biomorphic, abstract to ornamented. When the object has reached sufficient size, it remains stable, and is thereafter under the control of the user (figures 4 and 5). At this point, if the user fails to decrease the intensity of the light (by raising his or her fanalet), the object grows until it bursts, scattering a new crop of colored protos onto the floor. Other users can then nurture these protos, and the cycle starts again.

Once an object is grown and has stabilized, it becomes the user's "partner" and can be trained to dance. Any sufficiently rapid sequence of movement of the fanalet is interpreted as a start of this training, which makes the partner learn every new movement of the fanalet. The partner follows and memorizes this movement until the fanalet remains still, and then starts to repeat it continuously on its own (until it is taught a different sequence). Slower movements of the fanalet are interpreted as leading movements, which the trained partner follows as it dances around the lightpool. By alternating rapid and slow movements of the fanalet, users can teach, dance, and interact with their partner. Although the projected image is two-dimensional, the partners themselves are represented in three dimensions,

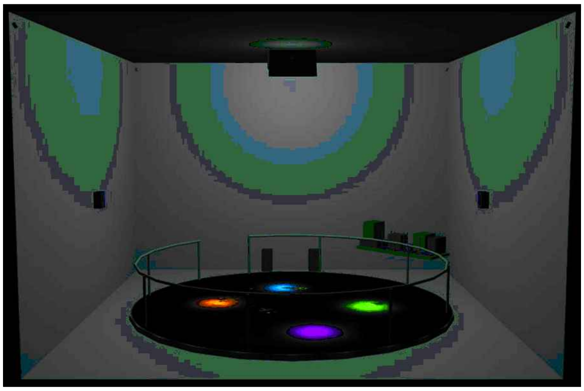

Figure 1. *Setup of El Ball del Fanalet or Lightpools.*

Figure 2. *"Technified" Fanalet* (paper lantern).

Figure 3. *The typical Catalan popular dance: El Ball del Fanalet.*

Figure 4. *A user growing an object to obtain a dance partner.*

Figure 5. *A user interacting with his/her new partner.*

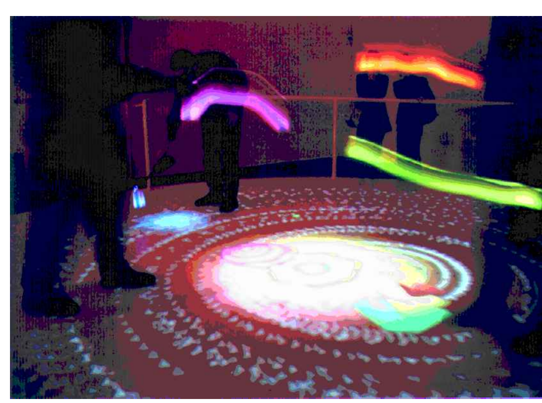

Figure 6. *The fanfare and shower of lights that precede the autonomous dance of the objects and lightpools.*

so that upward and downward motions of the fanalet affect the partner's orientation and scale. Each partner is composed of three distinct parts (subobjects) that follow the motion of the fanalet at slightly different rates; this gives the partners a somewhat fluid appearance; certain details are revealed only when the partner is in motion (a bit like a peacock showing off its feathers). If the user trains a partner for too long, the partner becomes bored and leaves the arena through the exit in the railing, leaving the user with an empty lightpool. The user must then nurture a new proto to start anew.

From these interactions emerges a complex, ever-changing dance of participants and virtual objects. New users can enter at any time (as long as there is an available fanalet), and users can stay as long as they like. Additional spectators can comfortably watch from behind the circular railing that encloses the arena. Several users can share a single fanalet, passing it back and forth as they wish. The atmosphere is one of a casual but concentrated chaos, as users observe and interact with their objects and each other.

Each event in the piece has an audio component as well: a delicate glassy tinkle as protos appear, the whoosh of a pneumatic pump as objects grow, whipping air currents as objects are trained, and so on. A second, independent, stereo audio track—consisting of a variety of ballroom dance music (processed with huge amounts of reverberation)—plays continuously and softly in the background, as though it is being heard from a great distance.

After two or more users have grown and trained stable objects, they may, at any point, bring their fanalets together in a ritual gesture that indicates that they want their objects to join together in a choreographed object-dance. After a brief fanfare and a shower of light (figure 6), the lightpools and objects leave the user behind and perform a group dance to a driving drumbeat, using the movements that they have been taught as they follow a choreographed path. The lightpools then exit the arena leaving the objects outside, and come back to the users to begin the cycle again.

El Ball del Fanalet or Lightpools is, in part, an investigation of the proposition that, given the current state of available technology, it makes some sense to project an image onto the floor, instead of an upright wall. Of the three dimensions (height, width, depth), we are suggesting that height might be the most easily expendable. This strategy allows immersion or entrance into a two-dimensional image. Projecting the image onto the floor also forces new strategies of spatial representation, because it effectively makes perspective space incomprehensible. The project also suggests a number of possible relationships that can be experienced within a public space in an atmosphere that is reminiscent of a ballroom or skating rink. We are interested in the area of what might be called *casual* or *open-ended* interactivity, an area that remains underexplored. Most media installations are still designed for a single user or for remotely networked users, and most still require users to remain immobile. We are trying to create the conditions under which an interactive, immersive image can be experienced as a casual, open, social space.

After its installation at the Miró Foundation in Barcelona, this piece has proven to be very successful in breaking the fear or apprehension that is often found in users who have no previous experience in VR installations. Although there was no formal psychological study of the users' reactions or appreciation of the piece, after 7,000 users from very different origins and cultures[1] had passed through it in fifteen days, we observed that very few had troubles in their interaction and that most of them had a successful experience with the piece. We believe this is due to the fact that the user is led into the experience through a very natural and clear metaphor that has smoothly emerged from the designed interface and which is well suited to the proposed interaction. We conclude this has been favored by the interaction-driven strategy as will be seen in section 5.2.

5.2 The Interaction-Driven Development of El Ball del Fanalet or Lightpools

Now we will describe the steps taken in developing this piece. Although this should be read as a continuous description, we have tagged the sections corresponding to each stage in the interaction-driven strategy to guide the reader in the mapping of the structure of the interaction-driven strategy onto the design process.

5.2.1 Identify Input and Output Interfaces. Our main goals were to design a multiuser interface that would allow the users to engage in a social interaction among themselves and would not alienate them as multiuser VR applications usually do.

1. The Miró Foundation, being a world-renowned museum, is one of the most visited museums in Barcelona and its visitors range from school trips to tourists from around the world.

In this manner, we developed the idea of an interface in which the environment would be projected onto the floor and would be large enough to hold up to four simultaneous users. We did not want the entire environment to be visible at once within the interaction area. We decided the users should be able to see only a delimited area around them, forcing them to move about in order to understand the whole environment. Hence, some sort of wireless position sensor would be necessary to track the users' position, allowing the system to reveal the areas of the environment through which they pass.

It was important to us that the users would have to collaborate with each other to be able to acquire complete understanding of the experience, by sharing their limited personal view with the others.

5.2.2 Identify the Type of User. At the same time, we wanted the piece to be accessible to a very wide range of people. Therefore, the experience needed to provide simple interaction mechanisms, although the relations derived from their interaction could be rich and diverse.

5.2.3 Define of the Type of Application. We also wanted the application to be an explorative and manipulative experience, with a certain degree of contribution on the users' part. These contributions would be channeled through the change in behavior of the objects in the environment.

5.2.4 Define of the Application Topic or Theme. At this point, we still did not know what the actual appearance of the experience would be like or which type of objects we would design. Based on the aforementioned premises, we thought an appropriate metaphor for such a multiuser experience would be that of a dance floor. We chose a typical Catalan couples dance known as *El Ball del Fanalet,* which seemed particularly appropriate for the following reasons:

- It is a social dance, hence promoting user interrelations and not only user-environment interaction.
- The couple in the dance holds a paper lantern (a *fanalet*) with a lit candle inside, which must be carefully moved while dancing smoothly so as to avoid having the whole lantern catch fire. This fanalet was very appropriate as a metaphor for the exploration of the environment through the limited visible area that we wished to impose on the users. This area evolved into the virtual light spot, or lightpool, which would follow the user around the dance floor.
- The light projected by the fanalet could be used as energy to unleash changes in the environment as an interaction element.

With this frame of reference, we decided to search for all possible dance relations:

- **A single user:** The sole fact of having the user exploring the environment would already sketch a dance scenario.
- **User with virtual object:** We decided that the defined objects should function as dance partners for the user.
- **User with user and virtual object with virtual object:** Once each user has a virtual dance partner, a user might approach another user to dance with him/her and at the same time make their virtual objects dance with each other.
- **Virtual object with virtual object:** We decided that there should be a mechanism through which virtual objects could start a dance between them, becoming completely autonomous from their owner/user.

All these decisions led to the final design of *El Ball del Fanalet or Lightpools* and its components.

5.2.5 Identify Processes. The main algorithms developed were the behavior algorithms for the lightpools and the objects (from protos to partners). These were developed through the implementation of finite automata, but we will not describe them here.

5.2.6 Identify the Necessary Virtual Objects. We wanted the virtual objects to meet certain requirements:

- They should not represent anything in particular, because there was no specific topic to which they would be related. The dance floor metaphor had set a context, but we did not want to represent any particular dance floor nor define any virtual human dancers.

- They should be able to evolve in size and shape.
- They should exhibit interesting movements that could be appropriate to the notion of dancing in abstract terms.

With these properties in mind, small objects were designed as particles, so that they would start as proto-objects with the potential of being nurtured to grow to a more defined shape. These "mature" objects would become the dance partners for the users. The partners would follow the lightpool that made them grow with an elastic link, which would cause a delay in their response to the lightpool's movements. Each partner would in fact be composed of three subobjects, each of which would show a progressively larger delay such that the elastic movement would be exaggerated and the partner would break up into its parts while moving and recompose while at rest. This behavior would create a train of movement that would accentuate the sensation of dancing and choreography.

5.2.7 Identify the Data Involved. The input data was identified as the position changes of the wireless sensors in the fanalets, because these changes were to unleash the different reactions and states of the lightpools and consequently of the objects (both protos and partners).

5.2.8 Identify Object-Modeling Tools Needed. We decided a standard modeling tool was to be used.

5.2.9 Identify Application-Development Tools. Because great flexibility and algorithmic power were required, programming with a standard programming language such as C, enhanced by high-level VR libraries (WorldToolKit from Sense8), was found to be the best approach.

As can be seen from the above description, the development of the experience (from the conceptual stages to the actual implementation) did not take into account the content of the final design until late in the process. This method radically changes the way developers think of the application because the user interaction is the central issue.

To summarize, in this piece we started by defining the users' interaction (such as relations between users (social aspects), relations between users and environment, and view of a limited area of the environment) and the interface (the limited area that the user controlled). This led us to find appropriate metaphors that would enhance the interaction: a dance floor and light spots. Finally, the metaphors set the context for the content of the experience, that is, the final aspect, feel, and setup.

6 Conclusions

We have presented a novel strategy for developing VR applications that proves to be very powerful in experimental, creative, and artistic VR experiences. The idea behind this new strategy is to concentrate on user interaction rather than concentrating on the content specified by the application's topic; hence, we have named it *interaction-driven* strategy as opposed to *content-driven* strategy. In fact, there is no a priori application's topic when approaching development in this manner. This provides great design freedom to the developers in the definition of new interfaces and new interaction relationships that allow for experimentation in user approach and reaction to VR applications. This freedom is due to the fact that this strategy does not fix the application topic and its related metaphor, but rather lets the metaphor emerge naturally from the defined interaction and sets the topic through the metaphor.

The interaction-driven strategy was found experimentally while developing an artistic multiuser VR application, for which we had imposed upon ourselves the restriction of designing first how we wanted the user to interact. We worked under this restriction to concentrate on the social aspects of multiuser VR interaction and interfaces. We have described both the piece and its development process and have shown how the interaction-driven strategy has enhanced experimentation with interaction and interface aspects in the final result.

It is clear that one is working under different frames of reference when developing a VR application with the content-driven strategy than with the interaction-driven strategy. The whole background positioning is turned upside-down, giving a predominant position to the content and topic of the experience, or to the ways in which the users approach the experience and interact, respectively.

On evaluating the difference between the two, it is clear that the use of one or the other depends upon the desired final result. In interaction-driven development, taking advantage of the high level of freedom in the interaction model of the experience, one may be much more creative and innovative. One may define types of users that would not be possible if the theme and context of the application were fixed from the beginning. On the other hand, in content-driven development, one may be much more rigorous when defining requirements, and the lack of flexibility in the interaction model makes everything more controllable under standard and known frames.

The content-driven model is used in all applications in the simulation and scientific ambit, whereas the ambit of artistic production may choose to use a content-driven strategy for applications with a specific topic, or an interaction-driven strategy for applications with an important experimental use of interaction strategies. It is important, at this point, that many more applications be developed with this new strategy to be able to evaluate it in greater detail.

The two strategies that have been studied here are not the only two that can be obtained from the analysis of the development stages of a VR application. Other strategies may be defined and studied in the future, but we have especially described the interaction-driven strategy because we have obtained it experimentally from our artistic application.

Acknowledgements

We would like to thank Perry Hoberman for sharing his experience throughout the whole production process of *El Ball del Fanalet or Lightpools*. Also, we thank the Miró Foundation for its support and contribution to the production and first installation of the piece. We would also like to thank X. Artigas and R. Franch for their excellent work in developing the piece. Finally, we want to thank especially Orit Kruglanski for her excellent constructive criticism to both the style and content of this article.

References

Bryson, S. (1995). Approaches to the successful design and implementation of VR applications. In R. A. Earnshaw, J. A. Vince, & H. Jones (Eds.). *Virtual reality applications* (pp. 3–15). London: Academic Press.

Earnshaw, R. A., Vince, J. A., & Jones, H. (Eds.). (1995). *Virtual reality applications*. London: Academic Press.

Erickson, T. D. (1990). Working with interface metaphors. In B. Laurel (Ed.), *The art of human computer interface design* (pp. 65–73). London: Addison-Wesley.

Hoberman, P., Parés, N., & Parés, R. (1999). El Ball del Fanalet or Lightpools. *Proceedings of International Conference on Virtual Systems and Multimedia* (pp. 270–276). Dundee, U.K.: VSMM'99.

Kalawsky, R. S. (1993). *The science of virtual reality and virtual environments*. Reading, MA: Addison-Wesley.

Loeffler, C. E., & Anderson, T. (Eds.). (1994). *The virtual reality casebook*. New York: Van Nostrand Reinhold.

McCauley, M. E., & Sharkey, T. J. (1992). Cybersickness: perception of self-motion in virtual environments. *Presence: Teleoperators and Virtual Environments, 1*(3), 311–318.

Parés, N., & Parés, R. (1993). Galeria virtual. *Proceedings First Eurographics Workshop on Virtual Environments. Eurographics Technical Reports Series* (pp. 127–134).

———. (1994). Galeria virtual: A platform for non-limitation in contemporary art? *Proceedings Virtual Reality and Applications*. Leeds, U.K.: British Computer Society.

———. (1995). Galeria virtual: A platform for non-limitation in contemporary art? In R. Earnshaw, J. Vince, & H. Jones (Eds.), *Virtual reality applications* (pp. 186–207). London: Academic Press.

Parés, N. (1999). Characterization, issues and properties of artistic VR applications. VR as an art production medium. Pre-thesis research work in the audio-visual communication doctoral program. Universitat Pompeu Fabra, Barcelona, Spain (in Catalan only).

Waxelblat, A. (Ed.). (1993). *Virtual reality applications and explorations*. London: Academic Press.