INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team siames

Image Synthesis, Animation, Modeling and Simulation

Rennes

THEME COG

Activity Report

2004

Table of contents

1. Team

Head of project-team
Bruno Arnaldi [Professor (Insa)]

Administrative Assistant
Catherine Godest [CNRS]
Mathilde Briard [INRIA since november 2004]

Staff member Inria
Alain Chauffaut [Research Engineer]
Georges Dumont [Research Scientist until September 2004]
Anatole Lécuyer [Research Scientist]

Staff member CNRS
Stéphane Donikian [Research scientist]

Staff member University of Rennes 1
Kadi Bouatouch [Professor]
Yves Bekkers [Professor]
Thierry Duval [Assistant Professor]
Fabrice Lamarche [Assistant Professor since September 2004]

Staff member ENS Cachan
Georges Dumont [Assistant Professor since September 2004]

Staff member University of Rennes 2
Romain Thomas [Teaching Assistant until August 2004]

Project technical staff
Guillermo Andrade [Inria until march 2004]
Christian Le Tenier [Inria until march 2004]
Frédéric Devillers [Inria]
Stéphane Ménardais [Inria until June 2004]
Tangi Meyer [Inria since July 2004]

Ph. D. Student
Jean-Eudes Marvie [INRIA Grant until October 2004]
Caroline Larboulette [MENRT Grant]
Nicolas Mollet [INRIA Grant]
Chadi El Zammar [INRIA Grant]
Julien Perret [MENRT Grant]
Marwan Badawi [Lebanon Grant]
Olivier Martineau [Cifre Grant]
Sébastien Paris [Cifre Grant]
Nicolas Pronost [MENRT Grant]
Jean-Marie Souffez [INRIA Grant]
Richard Kulpa [Grant]
Yari Rad [Sfere Grant]
Javorslav Krivanek [INRIA Grant]
Pascal Gautron [MENRT Grant]

Post-doctoral fellow
Marco Congedo [DGA Grant until July 2004]
Mathieu Renouf [INRIA Grant since October 2004]

Software development staff

Jean-Marie Houssay [Inria]
Vincent Quester-Séméon [Inria since December 2004]

Collaborator

Gérard Hégron [Head of CERMA, Nantes]
Franck Multon [Laboratoire de physiologie et de biomécanique de l'exercice musculaire Rennes II]

2. Overall Objectives

The main context of our research activities concerns the **simulation of complex systems**. Indeed, our research topics deal with lighting simulation, mechanical simulation, control of dynamic systems, behavioral simulation, real time simulation and modeling of virtual environments.

Our studies are focusing on the following topics:

- **Computer Graphics**: our main works concern the design and integration of *models*, the design of new *algorithms* and of the *complexity* of the proposed solutions.
- **Simulation**: our main goal is to be able to compare the results produced by our simulation algorithms with real data in order to experimentally *validate* our approaches.
- **Systemic approach**: in order to validate the two previous points, we have to be able to treat *real industrial test cases* through the use of realistic implementation of our solutions.

More precisely, our studies deal with three complementary research themes:

- **lighting simulation**: realistic image synthesis algorithms give high quality results by the use of physical based illumination models in order to evaluate the light / material complex interactions.
- **physical system simulation**: first, our approach concerns the computation schemes needed to produce the state equations of the system (symbolic and/or numeric computation). Second, we are concerned by the control of these physical systems (virtual characters,...). In this field, we focus our attention on computer animation and simulation.
- **behavioral modeling and simulation**: in order to simulate the behavior of living beings in specific tasks, we design tools dedicated to the specification and simulation of dynamics entities (autonomous or semi-autonomous). Our behavioral models integrate continuous and discrete aspects, in one hand to be able to control motor capabilities of the entity and, in other hand, to take into account cognitive capabilities. We also focus our research activity on the virtual environment modeling process. In this field, we integrate, in the modeling process, geometrical information as well as topological and semantic information.

Two transverse topics are also very active:

- **Virtual Reality**: this field deals with some of our research topics such as lighting simulation, animation or simulation. Our approach adresses real industrial problems and proposes new solutions using our research results. The objective is to adapt the simulation of complex systems to the haptic constraints induced by the interaction by human beings.
- **OpenMASK software simulation platform**: the need of integration of our different research activities has produced a real time and distributed Virtual Reality and Simulation environment. This software is distributed according to an Open Source model (see http://www.openMASK.org).

3. Scientific Foundations

3.1. Panorama

The Siames team works on the simulation of complex dynamic systems and the need of 3D visual restitution of the results. These results could be producted in real time or in batch, depending on the nature of the simulated phenomena. Our scientific activity concerns the following points:

- motion of dynamic models for animation and simulation: in this field, our work deals with the modeling of physical systems, the control of these systems and all kind of interaction that could occur during the simulation. Special attention is given to contact and collision algorithms.
- behavioral simulation of autonomous entities: this topic concerns both the interaction between entities and the perception, by the entity, of the surrounding environment. The geometrical information is too poor to take into account the potential relationships between a behavioral entity and its static and dynamic environment. In order to provide high level interaction, topological information on space organization and on the objects of the environment is added to data structures.
- lighting simulation: in complex architectural environments, light propagation and interaction with object material generate a big amount of computation using a lot of memory. Our work on this subject concerns the use of a standard workstation or a network of workstations in order to provide the simulation results. This simulation also has to provide tools for the visual characterization of the quality of the results from the human perception point of view.

3.2. Dynamic models of motion

Keywords: *animation*, *hybrid systems*, *identification*, *levels of detail*, *movement*, *simulation*.

Animation: Models and algorithms that produce motion accordingly to the animator specification.

Physically Based Animation: Animation models which take into account the physical laws in order to produce motion

Hybrid System: dynamic system resulting of the composition of a part which is differential and continuous and a part which is a discrete event system.

State Vector: data vector representing the system at time t, example: position and velocity.

As for realistic image synthesis, the physically based animation introduces physical laws in algorithms. Furthermore, natural motion synthesis (living beings) needs to take into account complex phenomena such as mechanics, biomechanics or neurophysiology in order to treat aspects like planning and neuro-musculo activation.

The generation of motion for 3D objects or virtual characters needs to implement dedicated dynamic models depending on different application contexts: natural motion simulation, animation for multimedia production or interactive animation.

The mathematical model of the motion equations and the algorithmic implementation are based on the theory of dynamic systems and use tools coming from mechanics, control and signal analysis. The general structure of the dynamic model of the motion is a hybrid one, where two parts interact. The first one is a differential part while the second one is a discrete event system:

$$\begin{array}{c} \frac{dx}{dt} = f(x(t), u(t), t) \\ x_{n+1} = g(x_n, u_n, n) \end{array}$$

In this equation, the state vector x is the concatenation of discrete and continuous state parameters, u is the command vector and t the time.

For example, the contact and collision mechanical computation is performed using an hybrid system. Physically, a collision is a discontinuity in the state vector space (impulse = velocity discontinuity).

In the context, some emerging topics appear:

Automatic model generation: using a high level specification language, the challenge consists in producing both the hybrid dynamic model and the control algorithm.

Identification : a synthetic model is always difficult to produce off-hand. A new method consists in observing real systems using structural and parametric identification tools in order to determine it.

Level of detail : this tendency is essential in order to treat complex models and can be applied to solve geometric complexity but also mechanical complexity.

3.3. Lighting Simulation and Rendering

Keywords: *lighting simulation, partitioning, rendering, visibility*.

Global illumination : direct and indirect illumination computation.

Rendering: computation of an image of a virtual world as seen from a camera.

Partitioning : subdivision of a 3D model into cells.

Client-server : a server contains complex 3D scenes, a client sends requests for objects to the server.

Level of detail : an object is represented with a mesh at different resolutions.

A global illumination model describes the light transport mechanism between surfaces, that is, the way each surface interacts with the others. Therefore, the global illumination model is a key problem when accuracy is needed in the rendering process (photorealism or photosimulation). As global illumination is a computation intensive process, our research consists in making it tractable even for large and complex environments.

Another objective is to propose a new navigation system built upon our client-server framework named *Magellan*. With this system one can navigate through 3D models or city models (represented with procedural models) transmitted to clients over a network. Regarding procedural models, their geometry is generated on the fly and in real time on the client side. These procedural models are described using an enhanced and open version of the L-system language we have developed. The navigation system relies on different kinds of preprocessing such as space subdivision, visibility computation as well as a method for computing some parameters used to efficiently select the appropriate level of detail of objects.

To attain realism in computer graphics, two main attempts have been adopted. The first one makes use of empirical and ad-hoc illumination models. The second one makes use of the fundamental physical laws governing the interaction of light with materials and participating media. It integrates characteristics of the human visual system, in order to produce images which are exact representations of the real world. Our work follows this second approach and relies on the real aspects of materials and on the real simulation of global lighting using physics-based reflection and transmission models as well as a spectral represention of the emitted, reflected and refracted light powers. Unfortunately, global illumination is still a demanding process in terms of memory storage and computation time. Our objective is to rely on the radiance caching mechanism and on the performance of the new graphics cards to make interactive global illumination possible even for complex scenes.

In case of real-time remote navigation, transmission and real-time visualization of massive 3D models are constrained by the networks bandwidth and the graphics hardware performances. These constraints have led to two research directions that are progressive 3D models transmission over Internet or local area network and real-time rendering of massive 3D models.

In regard to progressive 3D models transmission, one can suggest the use of geometric levels of detail (LODs). Indeed, as soon as one LOD is selected according to its distance from the viewpoint, the finer LOD is prefetched over the network. In the same spirit, one can select the LOD of 3D objects to be transmitted based on the available bandwidth, the client's computational power and its graphics capabilities. Our work makes use of these two approaches.

As for real time rendering of massive 3D models on a single computer, one can find many solutions in the literature. The most commonly used solution consists in subdividing the scene into cells and computing a potentially visible set (PVS) of objects for each view cell. During walkthrough, only the PVS of the cell containing the current viewpoint is used for rendering. Our system for interactive building walkthrough follows this approach.

4. Application Domains

4.1. Panorama

Application fields of our research mainly concern the activities where intensive relationships exist between the simulation of physical systems and 3D visualization of the results.The concerned application fields are:

- architectural and urban environments
- energy propagation
- virtual actors and biomechanics
- virtual reality and augmented reality

4.2. Virtual Reality

Keywords: *Augmented Reality*, *Virtual Reality*.

Our activity in this field mainly concerns the multi-modality of human interaction. We focus our works on hap-tic and pseudo-haptic interaction, on local or distant cooperative work in the context of industrial application. We are also concerned by the production of innovative software solutions.

4.3. Virtual actors and biomechanics

Keywords: *Virtual actor*.

Human motion is a very challenging field. We try to increase the knowledge by producing parametric models of human movements. Indeed, by the use of motion capture systems and simulation of our models we can access to internal state of parameters. We could not access to them on real human. Consequently, we are able to produce virtual experiment in order to validate scientific hypothesis on natural motion. We also work on the analysis-synthesis loop in order to produce very efficient motion models with motion blending, real time constraint management, etc.

4.4. Virtual prototyping and physical models

Virtual prototyping deals with the use of simulation results in order to validate specific functional features during the design process. In the field, we use an optimization technique based on evolutionary algorithms and results coming from CAD process.

5. Software

5.1. Panorama

In order to validate our scientific results, we develop prototypic softwares with the capacity to treat industrial problems. The softwares presented in this section are all used in industrial cooperations.

5.2. OpenMASK: Open-Source platform for Virtual Reality

Keywords: *distributed simulator*, *interactivity*, *middleware*, *modularity*, *real-time simulator*, *software platform*, *virtual reality*.

Participants: Alain Chauffaut [contact], Jean-Marie Houssais.

OPENMASK (Open Modular Animation and Simulation Kit) is the federative platform for research developments in the Siames team. It is also recommended by PERF-RV (French National RNTL project on Virtual Reality). Technology transfer is a significant goal of our team.

OpenMASK is a software platform for the development and execution of modular applications in the fields of animation, simulation and virtual reality. The unit of modularity is the simulated object. It can be used to describe the behavior or motion control of a virtual object as well as input devices control like haptic interfaces. Building a virtual environment with OpenMASK consists of selecting and configuring the appropriate simulated objects, and choosing an execution kernel fulfilling the application needs. Of course, new classes of simulated objects have to be built first if they do not exist. But they can be reused in other applications.

OpenMASK comes with multi-site (for distributed applications : distributed virtual reality, distributed simulation ...) and/or multi-threaded (for parallel computations) kernels. These kernels enable off-line simulation as well as interactive animation. Visualization can be powered by Performer (Sgi) or by OpenSG (Fraunhofer Institute).

OpenMASK provides an Open C++ API dedicated to simulated object development and execution kernel tailoring.

Main features offered by the execution kernels:

- Hosting: creation and destruction of simulated objects.
- Naming: simulated objects, classes and attributes are named.
- Activating: regular activation (each object can have its own frequency) and/or occasional (on event reception) for simulated objects.
- Communicating:
  * using data flows between simulated objects
  * using signal diffusion in the environment
  * using events between objects
  * thanks to the provided data-types or specialized data-types created for the application
  * with adaptation to the different activation models using interpolation and extrapolation
- Time management: automatic data dating and unique time-stamp during computation.
- Distributing: presently powered by Parallel Virtual Machine (PVM). Distribution is transparent to the programmer but could be controlled by the operator.

Main features offered by the visualizing object:

- Mono or multi-pipes visualization, adapted for reality centers and workbenches. Multiple views and stereo-vision.
- Support of all geometrical file formats supported by Performer or by OpenSG.
- Component extensibility to take new animation primitives into account (available : quaternions, rotations, translations, matrices).
- X11 event or GLUT events captures and owner forwards.
- 2D or 3D picking and subscribers forwards.

Technology transfer: Our technology transfer initiative is based on industrials partners and supported by Open-Source distribution. We are supported by INRIA with dedicated resources (ODL 2001/02 and ODL 2003/04). First, we provided the platform which is of general interest. Now, we are delivering simulated objects dedicated to Virtual Reality, most of them with an Opensource licence: interactors, virtual human, force feedback processor, collisions manager, VRPN peripherical abstractions. OpenMASK is available on Irix, Linux and Windows systems.

5.3. MKM : Manageable Kinematic Motion

Participants: Stéphane Ménardais, Richard Kulpa, Franck Multon [contact], Bruno Arnaldi.

We have developed a framework for animating human avatars in real-time, based on captured motions. The first part of this work deals with the reconstruction of captured motion files. It is done offline with a software that imports motions in most usual formats like C3D (Vicon) or BVH (BioVision) and exports them in a morphology-independent file format which allows to replay the same motion on any avatar in a scene.

This format is based on a simplified skeleton which normalizes the global postural informations. This new formalism allows the motion to be adapted automatically to a new morphology in real-time. This is done by taking kinematic constraints into account.This approach dramatically reduces the post production and allows the animators to handle a general motion library instead of one library per avatar.

The second part of the framework provides an animation library which blends several kinematic parametrized models and adapts them to the environment and the avatar's morphology. This work deals with motion models obtained from biomechanical and statistical studies, motion synchronization (using biped footsteps), motion blending (real-time priorities, local skeleton blending, levels of details) and retargetting (interactive kinematics constraints solvers, filtering).

This tool has been used in several applications, for example in a virtual museum or a presentation for imagina 2002. It has been improved in the RIAM project "AVA-Motion", which ended in june 2004, to become a complete, "ready to use", library for industrial companies. It currently runs on Windows and Linux with different viewers and it has been also integrated in two different software architectures: AVA from the Daesign company and OpenMASK, our own platform.

5.4. HPTS++ : Hierarchical Parallel Transition System ++

Participants: Fabrice Lamarche, Stéphane Donikian [contact].

HPTS++ is a platform independent toolkit to describe and handle the execution of multi-agent systems. It provides a specific object oriented language encapsulating C++ code for interfacing facilities and a runtime kernel providing automatic synchronization and adaptation facilities.

HPTS++ is the last evolution of the HPTS model. Firstly designed for behavioral animation, it provides a generic and platform independent framework to describe multi-agent systems. It is composed of a language allowing agent description through finite state machines and a runtime environment handling parallel state machine execution and offering synchronization facilities.

The language provides functionalities to describe state machines (states and transitions) and to inform them with user specific C++ code to call at a given point of during execution. It is object oriented: state machines can inherit of other state machines and/or C++ classes to provide easy interfacing facilities. States and transition can be redefined in the inheritance hierarchy and the state machines can be augmented with new states and transitions. Moreover, state machines are objects that can provide a C++ interface (constructor/destructor/methods) for external calls. The compilation phase translates a state machine in a C++ class that can be compiled separately and linked through static or dynamic libraries. The runtime kernel handles parallel state machine execution and provides synchronization facilities. It includes a recent research work on automatic behavior synchronization. Each state of a state machine is informed with a set of resources (or semaphores) to specify mutual exclusions between state machines. Each state machine is informed with a priority function specifying its importance at each simulation time step. Each transition is informed with a degree of preference allowing to describe possible adaptations in regard with resource availability or need. Those three properties are combined by a scheduling algorithm in order to automatically and consistently adapt state machines execution with respect to their respective priorities and resource conflicts. Moreover, this algorithm provides an automatic dead lock avoidance mechanism. This property enables independent state machine description and ensures consistent execution without knowledge of their description and without explicit hand coded synchronization. Moreover, the kernel supports dynamic state machine construction and dynamic resource declaration.

This toolkit runs under Windows (Visual C++ 6.0 et .NET), Linux (g++ 2.96 - 3.2) and IRIX systems (CC). It has been used in different research fields such as behavioral animation, scenario description and automatic cinematography. Its scheduling system provides new paradigms for multi-agent systems description while ensuring the overall consistency of the execution.

5.5. Magellan: a Framework for Remote Real-Time Interactive Visualization

Keywords: *partitioning*, *real-time rendering*, *streaming*, *visibility*.

Participants: Jean-Eudes Marvie, Kadi Bouatouch [contact].

This software is devoted to real-time navigation through complex environments transmitted across low-bandwidth networks.

We have developed a framework, named *Magellan* which allows real-time walk-through of 3D models located on a remote machine and transmitted over a low bandwidth network, using the TCP/IP protocol. The global architecture of our system is illustrated by Figure 1. The server provides access to several city models, each one being represented by one database. Each database is a set of VRML97 files describing the 3D model. Each remote client machine can connect to the server to walk through a 3D model using its associated database. There is right now no interaction between clients, and each client renders its own representation of the city model.

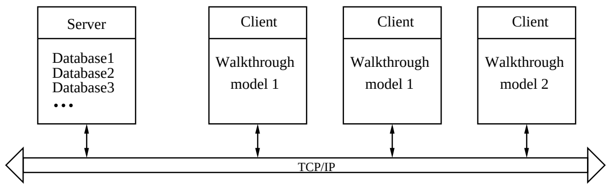

Figure 1. Global system architecture.

One of the main mechanisms in our system is to transmit and visualize only the subset of the model (geometry and texture maps) that is potentially visible from the current viewpoint. With this aim in view, the scene is subdivided into cells and a PVS of objects is computed for each cell. In addition, we determine the adjacency relationship between cells. During walk-through the cells adjacent to the current one, as well as their PVSs are prefetched over the network. A more pertinent prefetching technique is to determine the next visited adjacent cell using motion prediction. In this way, the database is progressively transmitted to the client. Furthermore the geometry used for the rendering process is the PVS of the visited cell. This PVS is frustum culled using the bounding box of each of its objects, and the obtained result is sent to the OpenGL hardware renderer.

5.6. CityZen: Automatic City Model Generation

Keywords: *City model generation, L-systems*.

Participants: Jean-Eudes Marvie, Julien Perret, Kadi Bouatouch [contact].

This software is devoted to the generation of city models in which buildings are described with a modified version of L-systems.

The new VRML97 scripting language we have developed aims at describing procedural models for representing cities. As the VRML97 ECMAScript language provides a few facilities for generating 3D models and partial scene graph structures, we propose a solution which consists of an extension of the L-system language that allows such operations. An L-system is a context-sensitive parallel grammar that performs rewriting operations on symbols controlling a graphic turtle. In our language we keep the grammar functionality but we do not make use of the turtle paradigm. This language is well adapted to progressive transmission of data over a network. Indeed, only the procedural models (described by grammar rules) are transmitted along with some parameters, which drastically reduces the amount of data to be transmitted. The geometry of the procedural models is reconstructed on the fly on the client side in the context of a client-server architecture.

6. New Results

6.1. Exploitation of highly complex CAD models in VR

Keywords: *CAD, Multiresolution data structures, Virtual Prototyping*.

Participants: Georges Dumont, Jean-Marie Souffez.

We work on the importation of complex CAD models in Virtual Reality applications. This work is the subject of a PhD thesis directed by Georges Dumont, and part of the RNTL SALOME 2 project. It is based on OpenMASK for the Virtual Reality applications, and SALOME for the production of CAD models. The aim is to handle these models interactively in a VR scene, to allow virtual prototyping of the models.

6.1.1. Results for virtual exploitation of highly complex CAD models

A bridge has been implemented between SALOME and OpenMASK, that takes care of importatiing of CAD models from SALOME to OpenMASK, and of exportating from the VR session to the SALOME platform. The classical use of this bridge is represented in figure 2:

1. SALOME produces a CAD model;
2. The import module handles the information provided by SALOME to prepare the VR session;
3. The model is exploited during the VR session, and eventually information is returned to the SALOME platform.

The information concerning the CAD model consists of two parts:

- Geometric information: the geometry of the model (vertices and faces);
- High-level information concerning the model, such as topological and semantic information.

The import module handles this information. Since the geometry of the model is too precise, it is simplified. The import module can also use the high-level information to build the VR scene. For example, one can think of the specification of the constraints existing in an articulated robot.

As for the simplificating of the geometry of the model, we have to keep the exploitation of these models as relevant as before the decimation, so we need to be able to locally recover the original precision of the model. We have therefore developed multiresolution algorithms, which are composed of two parts:

- As a preprocess, we decimate the model and store the history of the decimation in a data structure;
- At run-time, we traverse the data structure to locally refine the model when needed.

The multiresolution data structure is based on face clusters. This face cluster hierarchy makes the multiresolution cheaper to traverse for the CPU, while allowing high-rate rendering by the GPU. This multiresolution data structure allows real-time inspection of the model, without any loss in the relevance of the original CAD model.

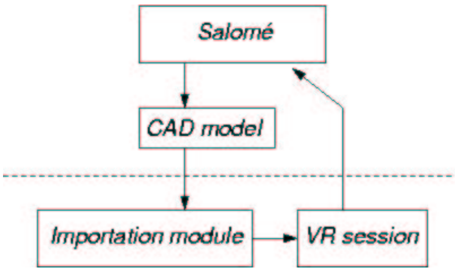

Figure 2. Coupling SALOME and VR

In figure 3, we present two levels in the multiresolution hierarchy.

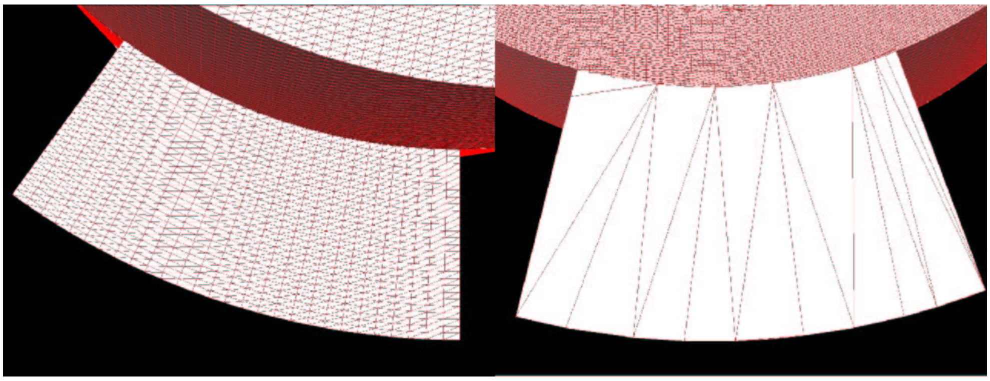

Figure 3. Two levels of details for the same model.

6.2. Real-time navigation through complex 3D models across low-bandwidth networks

Keywords: *Navigation, client-server, level of detail, real-time.*

Participants: Kadi Bouatouch, Jean-Eudes Marvie.

We are interested in interactive realistic navigation through complex scenes across low bandwidth networks. The problem to solve consists in determining, at each time of the navigation, which representation of the scene to transmit from the server to the client so as to reduce the latency time of rendering entailed by the geometric and photometric complexity of the scene. To solve this problem, we developed a distributed navigation framework relying on a client-server architecture. This framework allows to use different rendering techniques associated with different kinds of 3D models (points, surface, volume) and to meet constraints such as frame-rates fixed by the user.

For a purpose of remote navigation through 3D models, we developed a framework, named Magellan, which is outlined in Figure 4. This framework allows the automatic generation of *client*, *server* and *builder* applications for Linux, Win32 and SunOS platforms. Client/Server applications manage automatically multi-server/multi-client connections.

In Magellan architecture, Client/Server communication uses the TCP/IP protocol. The server side assigns a thread to each connected client. Each thread marshalls and sends data nodes on demand. The requested data can be complete or partial files to allow the transmission of progressive data. A client is composed of two main threads and any other user threads. The loading thread asks for nodes, unmarshalls nodes and adds nodes to the scene graph through the scene graph handler. The main thread performs three tasks consisting in generating the viewpoint motion, computing collisions and gravity, and updating the scene graph depending on the viewpoint position. The prefetching module uses motion predictions to down-load future visible cells. The rendering module selects the visible cells or objects and invokes their display methods. Finally, the added user module, named rewriting thread, performs the rewriting of parallel L-system scripts. All these tasks can access and modify the generic scene graph if needed through the scene graph handler.

Using C++ inheritance, these applications automatically manage *generic visual objects*. They are responsible for read and write from/to files, transmission across the network and display. The framework provides a C++ message class for marshalling/unmarshalling the data transmitted over the network. These objects use OpenGL calls to generate their visual representation. In this way we can easily implement new visual objects that will be used by these applications. Furthermore, making the objects responsible for their own display allows for hybrid rendering such as merging polygon-based and point-based renderings.

Besides, the framework allows to develop and use any new *rendering*, *motion* and *prefetching* modules. In addition, the user can integrate new modules into the framework such as the *rewriting* thread used for generating geometric models associated with procedural models. Each of these modules can be implemented with threads if needed. They can all access the scene graph handler to manipulate the generic scene graph encapsulated by this handler.

The generic scene graph data structure is composed of one root node and any needed external root nodes. Each root node is associated with a file. A node can be either a *VisualNode*, a *TextureMapNode* or a *ConvexCellNode*. The *ConvexCellNode* nodes are used if the scene is subdivided into cells. With this kind of node, the *rendering module* takes advantage of cell-to-cell or/and cell-to-geometry visibility information. In addition, the *prefetching module* can prefetch cells using motion prediction, and the *motion module* can use the result of space subdivision to speed up collision detection. The framework provides several other classes of nodes that are beyond the scope of this report.

6.3. Generation of city models

Keywords: *City model generation*, *L-systems*, *rewriting systems*.

Participants: Kadi Bouatouch, Julien Perret, Jean-Eudes Marvie.

We designed and implemented a software for creating cities using procedural models based on our new scripting language.

We developed a new VRML97 scripting language whose aim is to model cities. As the VRML97 ECMAScript language provides a few facilities for generating 3D models and partial scene graph structures, we propose a solution which consists of an extension of the L-system language that allows such operations. An L-system is a context-sensitive parallel grammar that performs rewriting operations on symbols controlling a graphic turtle. In our language we keep the grammar functionality but we do not make use of the turtle paradigm. Nevertheless, we easily emulate a turtle to describe plant models. In addition to the classical L-system functionalities such as the stochastic or condition guided rules choices, we introduce the possibility to cut a parallel rewriting process by placing a “!” character before the rule to be rewritten at first. This point is

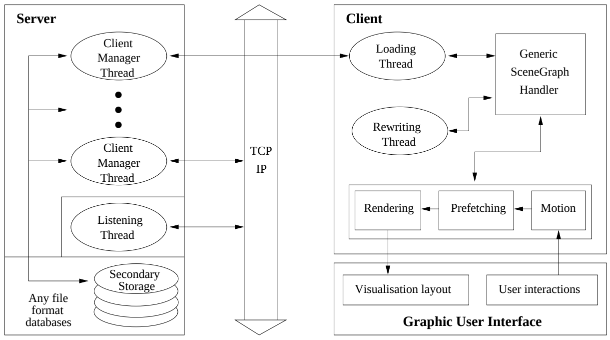

Figure 4. Magellan architecture.

very important since it allows for the instantiation of VRML97 nodes. As an example, consider the case of a natural tree. An L-system rule associated with a branch of this tree may generate two other rules, one for each of its leaves. Let us call leaf rule these two new rules. The two leaf rules can be rewritten in parallel since they do not share any information. Now, if we consider a rule describing a building frontage that generates multiple window rules, the first one generates the window geometric model while all the subsequent ones instantiate this window model to repeat the window pattern lying on a frontage model. We must ensure that the generation of the first window model ends before other rules use it by placing the character "!" just before the first rule. In this way, once the first rule has been rewritten, all the other rules can get rewritten in turn and in parallel.

In addition to this mechanism we have implemented built-in rules that allow dynamic allocation of VRML97 built-in or prototyped nodes through parameter passing and fields access. In order to allow geometry generation, we make use of mathematical operators using floats and integers, list manipulations as well as trigonometric functions. Using these mechanisms we can easily generate meshes and node hierarchies for pattern based geometric models using only one rewriting call.

6.4. Interactive Global Illumination

Keywords: *GPU*, *Global Illumination*, *Interactivity*, *Irradiance and Radiance Caching*.

Participants: Kadi Bouatouch, Javorslav Krivanek, Pascal Gautron.

We designed and implemented a software for interactive global illumination using programmable graphics hardware.

Computing global illumination amounts to solve the rendering equation which is an integral equation. Unfortunately, this equation does not have an analytic solution in general. Consequently, Monte Carlo integration is the method of choice for solving it. However, Monte Carlo integration requires the computation of many samples, which makes it demanding in terms of computation time. Our objective is to propose an algorithm which allows interactive global illumination.

Our approach makes use of ray tracing, Monte Carlo integration and caching. It aims at extending the "irradiance caching" algorithm. Note that this algorithm is based on the observation that the diffuse component of radiance, reflected on a diffuse surface and due to indirect illumination, changes very slowly on this surface. This allows to sparsely sample and cache the incoming radiance, then reuse the cached samples to estimate the incoming radiance at nearby points. This method is computationally efficient since the sampling is sparse. However this method is limited to indirect diffuse lighting computation.

We focus on extending the irradiance caching approach to indirect glossy global illumination. Our algorithm relies on "radiance caching". It is based on the caching of directional incoming radiances. We first designed a new set of basis functions defined on the hemisphere to represent directional incoming radiance and BRDFs.

This representation along with a new gradient-based interpolation method are the bases of our radiance caching-based algorithm.

For now, only CPU-based algorithms have been developed. We are now aiming at designing a radiance caching algorithm which takes advantage of programmable GPUs, to achieve interactive global illumination computation.

6.5. Interactions within 3D Virtual Universes

Keywords: *Collaborative Virtual Reality*, *Immersive Interactions*, *Multimodal Interactions*.

Participants: Thierry Duval, Alain Chauffaut, Chadi Zammar.

Our goal is to offer better interaction possibilities to end-users of 3D virtual environments. We first explore the different interaction possibilities in the fields of multi-users collaboration and multi-modal interactions, then try to provide generic tools to enable interactivity with virtual objects: to make virtual objects interactive ; to encapsulate physical virtual reality device drivers in an homogeneous way.

This work uses the OpenMASK environment to validate the concepts, to create demonstrators, and to offer interaction solutions for all the researchers of our team who use OpenMASK.

Interaction distribution between several sites relies upon the distributed mechanisms offered by OpenMASK: referentials and mirrors.

Multi-users and multi-modal interactions use the data-flow communication paradigm supported by OpenMASK, allowing data transfer from outputs towards inputs, and facilitate the fusion of the inputs coming concurrently from several outputs. They also use the sending event communication paradigm of OpenMASK that allows to send events even to objects that are located on distant sites.

During this year, we worked upon:

The finalization of our architecture for objects interactivity: we provide adapters to make simulated objects interactive. These adapters are divided into several classes to realize three tasks:

- the first task is to teach a simulated object the communication protocol useful to talk with an interactor.
- the second task is to dynamically create new inputs in order to use the interaction data provided by an interaction tool.
- the third task is to provide a way to connect an interactive object to an interaction tool, in order to be able to dynamically change the interaction behavior of an interactive object during a simulation.

It is possible to combine all these tasks in a modular way to obtain a great number of interaction possibilities.

Interactors for new interaction tools: We also worked on a generic way to combine drivers for physical devices, in order to implement new interaction tools. We call these generic modules interactors. The parts of these tools can be combined in a generic way to create several kinds of interaction tools. These interactors are based on several classes, once again to achieve several tasks:

- the first task is to teach an interaction tool the communication protocol useful to talk with an interactive object, to provide inputs to manage user events, and to create new outputs useful to provide new data to interactive objects, thanks to the control parameters of an interaction tool. This way it is possible to provide any number of control parameters to control an interactive object.

– the second task is to calculate new values of control parameters that could be useful for interacting with a simulated object, accordingly to some physical or virtual input devices, such as a virtual ray, a virtual hand, or a virtual 3D cursor. We do this in a generic way, to allow different behaviors for these interaction tools. We also provide an interactor based on keyboard and mouse, with the same architecture as the other virtual interactors.

Interactions with haptic feedback: We also provide new interactors that can work with physical objects, by enabling a virtual coupling between physical objects and kinematical objects (the objects we use to interact with). So we can now create virtual universes containing both kinds of virtual objects : physical and kinematical, and these objects can interact together, thanks to this virtual coupling, as illustrated in figure 5.

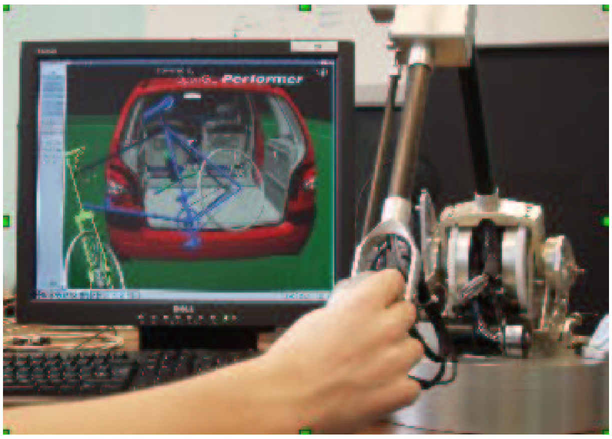

Figure 5. Interactions betwen physical and kinematical objects.

The software architecture of these new tools is similar to our standard architecture for interaction, and we now are working on the possibility to be able to use a same interactor to interact with several kinds of interactive objects (physical and kinematical ones).

Interactions to deform objects: We studied how to dynamically change the shape of a virtual object. We associate a virtual object (a small handle box) to each vertex of the geometrical mesh, then we ask the mesh to use these handle boxes as vertice to draw itself. Changing the position of any of these handle boxes will change the shape of the deformable object. These deformations can be made using animation techniques (by programming the positions of the handle boxes) or by interaction, as we can choose to make these handle boxes interactive, like any virtual object. An interactive deformation is illustrated in figures 6 and 7. The object on the left of these two figures is the initial shape, the object on the right of Figure 6 is the deformable object before deformation (we can see its associated handle boxes, visualized as small blue cubes), and the object on the right of Figure 7 is the deformable object after deformation (after the user has interacted with the handles boxes to change the shape of this object).
This first approach will not be scalable for shapes with hundreds of thousand of vertices, so for this kind of objects we will have to use another approach, either by using techniques like parametrical surfaces, or by allowing to deform only a small area of interest of the global shape.

Migration of virtual objects during collaboration: We studied the possibility to make the virtual objects migrate from one proces to another during a collaborative simulation.

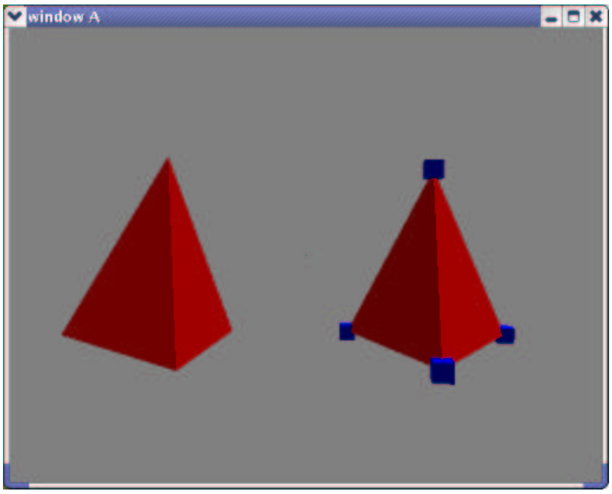

Figure 6. The object before the interactive deformation.

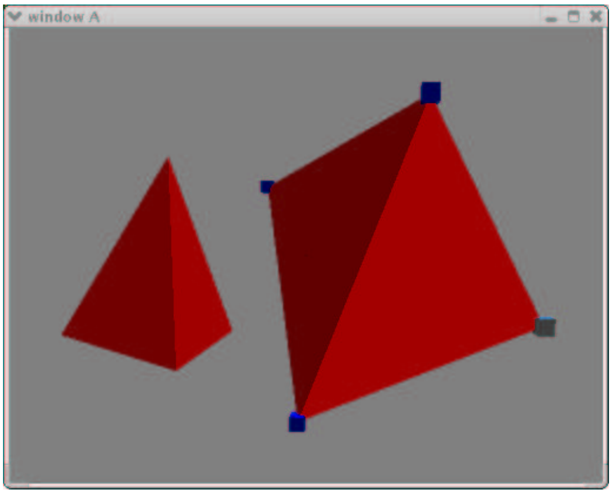

Figure 7. The object after the interactive deformation.

It can be useful when there are network problems, when we want to interact as efficiently as possible with an interactive object, or simply when we want to withdraw a process without losing the referentials it handles.
Distant interactions between a local interactor and a distant interactive object (located on another process on another site) can seem strange to a user because there will allways be a small lag between the evolution of the interactor and the evolution of the object in interaction with it. This strangeness will increase with network latency, which can even lead to unusable interactions if the network latency is too high or not stable enough. So it can be useful to be able to make an interactive object migrate (maybe temporarily) to the same process than the interactor that is controlling it. It is also very useful to make some objects migrate to the process of an interactor, when we know that this interactor will have to interact with them and if we can predict that there will be some network problems during the time of these interactions. Of course, it does not solve network problems during simultaneous multi-users interactions with a shared object.
During a collaborative simulation, when we want to withdraw a process on a distant site, we need to be able to make some objects migrate if we want to allow the persistency of the virtual universe.
A first version of migration of virtual objects has been implemented within the OpenMASK kernel, but we are still working to improve it.

6.6. Mechanical models and virtual prototyping.

Keywords: *mechanical models, optimization, simulation, virtual prototyping*.

Participants: Georges Dumont, Christofer Kühl.

The design of an active multi-link micro-catheter actuated by Shape Memory Alloy (SMA) micro actuator is adressed in this part. This may be a response to one major medical demand on such devices, which will be useful for surgical exploration and interventions. We propose a training and design simulator dedicated to such catheters. This simulator is based on the OpenMASK simulation platform. Since the catheter is a robotic system, it is evaluated by a dynamical simulation addressing a navigation task in its environment. The mechanical model of the prototype is presented. We have to develop an interaction model for contact. This model uses a real medical database.

The foreseen endoscope has to be agile enough to crawl its way inside complex environments like human ducts. So, the real prototype is a modular stiff poly-articulated structure.

The interests for a design and training simulator, presented in figure 8 are triple:

- Teaching young surgeons: the training on simulator is obviously more accessible, less expensive and less risky and should allow to experiment pathological cases;
- Preoperative training: the simulator will allow the surgeon to virtually repeat the operation on the patient, before the real operation;
- Virtual prototyping: the simulator allows to virtually test the tool quality with respect to a given operation task. So, by optimization techniques based on genetic algorithms, we can test which is the best candidate to achieve this task, and propose an adapted endoscope.

A mechanical model of the real prototype allows us to better understand its capabilities. The model is a kinematical open chain described by a Denavit and Hartenberg representation. The obtained dynamic equations are classically written as: $C \cdot \dot{q} = G(q,t) + K \cdot (q - q_0) + F(q, \dot{q}, t)$

In this equation, q is the vector of the parameters describing the configuration, G represents the gravity effect, K represents the elasticity effect of the SMA actuators in uncontroled configuration and F represents the forces exerted by the SMA actuators in the controlled configuration and the contact forces. These equations are resolved by using C++ libraries proposed by Barenburg (2000). The approach is based on six degrees of freedom (DOF) object models and uses an iterative constraint correction method. We solve the equations by using a Runge-Kutta (2 or 4) algorithm, according to the Newton-Euler laws of motion. Then, as the joints are

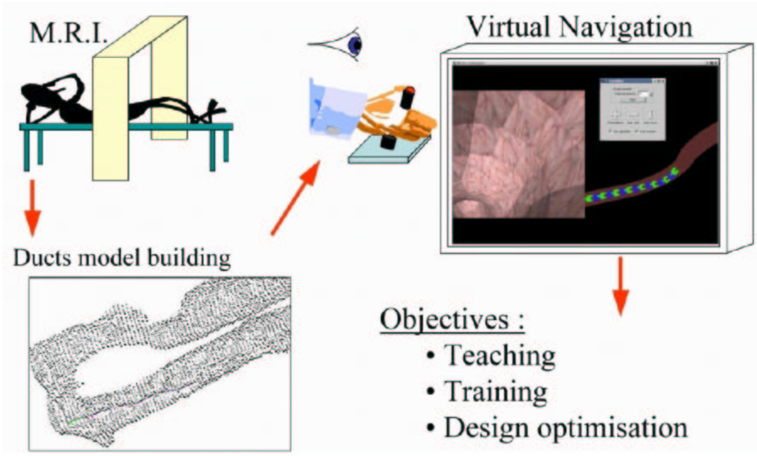

Figure 8. Overview of the simulator.

translated into terms of geometrical constraints, an iterative constraint correction method is used by means of a Newton Raphson algorithm.

Various models have been proposed to describe the behaviour of human organs or tissues, including non-linear behaviour and relaxation. Because the interaction occurs between a stiff body imposing the motion (the links of the endoscope on one hand) and a soft body (human tissues on the other hand), we have to compute the reaction force by a compliance method. This method is applicable with respect to the dynamical model and is a rapid computation method, which is in the scope of our objective to produce "real-time" simulations.

To optimize the design of the endoscope, we have proposed to use a genetic algorithm, as presented on figure 9. The algorithm analyzes the results for the simulated prototypes and evolves to choose the best one.

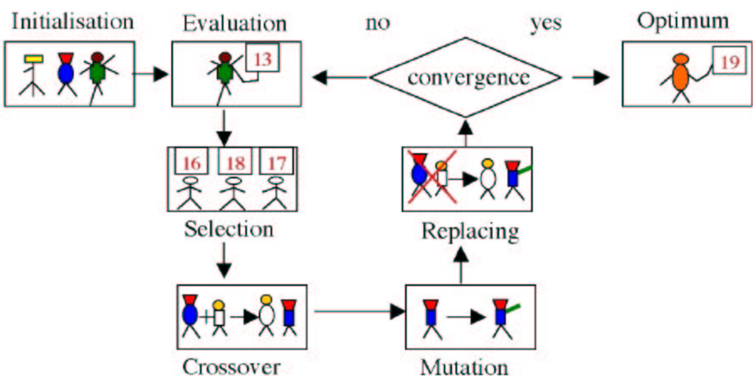

Figure 9. Genetic algorithm used for optimization design.

6.7. Morphological and stance interpolations in database for simulation of locomotion.

Keywords: *Bipedalism and Locomotion, Motion retargeting, Simulation and Biomechanics, Virtual Human.*

Participants: Georges Dumont, Nicolas Pronost.

We are working on morphological and stance adaptation of locomotion models for virtual humans in the scope of an ATIP CNRS project. We have developed a computer tool for testing hypotheses and generating a plausible walk according to anatomical knowledge. To do so, we introduce an interpolation method based on morphological data, and both stance and footprint hypotheses.

The study of the articulations leads us to propose a model of the articulated chain, including links and joints. We have developed a motion retargeting method based on morphological and stance parameters. This allows to develop the bipedal gait comprehension and to propose to the paleoanthropologists a tool for testing locomotion hypotheses. The method relies on real gait acquisition. The classical retargeting methods deal with geometrical gait adaptation. Here, the data are processed by two successive algorithms:

- The first is a morphological adaptation, based on the skeleton dimensions and on the articulations configuration;
- The second is a stance adaptation, based on the footprints and on the natural posture.

In figure 10, we present a schematic view of this adaptation method.

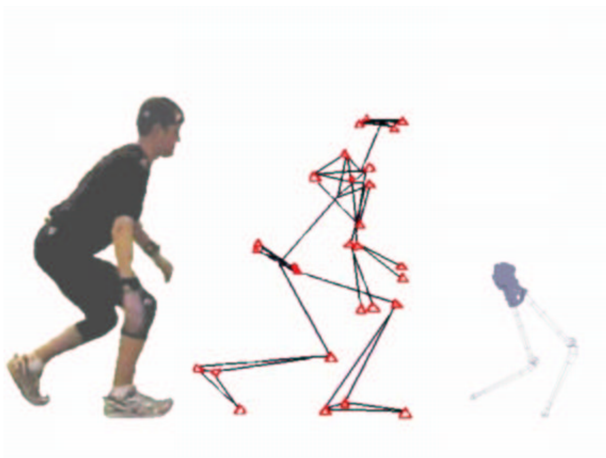

Figure 10. Locomotion adaptation from human to extinct hominid

We interpolate on a specific representation of the movement, called *poulaine*, which is the Cartesian trajectory of the ankle in the reference frame centered at the middle of the pelvis. We use a motion capture system to acquire movements of a walk cycle. We then model these movements with a generic parametric representation and define a database of movements.

The interpolation process on this database produces a retargeted motion adapted to the morphology of the considered skeleton. This interpolation is done according to three main hypotheses. The first concerns the reference stance, the second is the lateral spacing of feet, and the third is the length of the step. The fundamental idea of our method is to separate the $\overrightarrow{x}$, $\overrightarrow{y}$ and $\overrightarrow{z}$ interpolations and this is why we captured the walks according to the most influential parameters on each axis. Then the algorithm allows the choice of the best curves with respect to morphology of the goal skeleton and with respect to the stance hypothesis.

The calculated *poulaine* constitutes the trajectory of the effector X of the kinematical chain. This trajectory is imposed in the traditional inverse kinematics equation. To solve this inverse kinematics problem, we use a solver where the primary task is to follow the *poulaine*. It also supplies secondary tasks which ensure the respect of joint limits, the minimisation of the rotational kinetic energy and the respect of the posture of reference.

The multidimensional method finds an application in the field of anthropology: it contributes to draw a plausible walk for early hominids using their anatomical and osteological data. Lucy (A.L. 288-1) is a very well preserved specimen of the *Australopithecus Afarensis* species, early hominid almost 3,3 Millions years old. The pelvic girdle and lower limb skeleton (pelvis, femur and tibia) were reconstructed virtually and articulated. The calculated trajectory combines three different curves, and corresponds geometrically to an intermediate stature, between an upright human and a “bent-hip, bent-knee” human. Figure 11 shows angular trajectories obtained for the pelvis (3 DOF), hips (3 DOF) and knees (1 DOF).

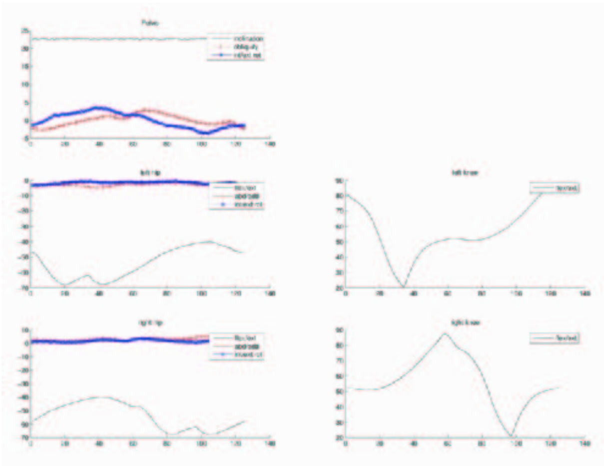

Figure 11. Angular trajectories of pelvis, hips and knees for Lucy’s skeleton

6.8. Haptic Interaction in Virtual Reality

Keywords: *Force-Feedback*, *Haptic*, *Milling-Machines*, *Navigation*, *Pseudo-Haptic*, *Textures*, *Training*.

Participants: Anatole Lécuyer, Bruno Arnaldi.

Haptic interaction consists of providing the user of a Virtual Reality system with the sensations involved by touch (i.e. tactile and force feedback) during the manipulation of virtual objects. We describe hereafter our recent results in the field of haptic interaction in virtual reality: first the pseudo-haptic simulation of textures, second and third the development of a virtual environment based on haptic and pseudo-haptic feedback dedicated to the technical training of milling machines in Virtual Reality.

Haptic interaction consists in providing the user of a Virtual Reality system with the sensations involved by touch (i.e. tactile and force feedback), mainly during the manipulation of virtual objects. Historically, the development of haptic interfaces originates from tele-operation. Indeed, the first force-feedback interfaces were developed for tele-operations within hazardous environments. But nowadays, a larger number of applications has been foreseen for haptic interaction in Virtual Reality. These applications belong to various fields: Medicine (surgical simulators, rehabilitation), Education (display of physical or mathematical phenomena), Industry (virtual prototyping, training, maintenance simulations), Entertainment (video games, theme parks), Arts and Creation (virtual sculpture, virtual instruments), etc. Thus, the field of "haptics" concerns an increasing number of researchers and companies specialized in Virtual Reality.

The integration of haptic feedback within a virtual environment raises many problems at different levels - including the hardware and software issues. Furthermore, a current major limitation for the design of haptic interfaces is our poor knowledge concerning human haptic perception. It is indeed fundamental to take into account the psychological and physiological issues of haptic perception when designing the technology and the use of virtual environments based on haptics. We therefore concentrated our work on both the perception issues and the implementation issues. We present hereafter our recent results in the field of haptic interaction in virtual reality:

1. the simulation of pseudo-haptic textures,
2. the influence of haptic feedback on the perception of self-motion in VR,
3. the Virtual Technical Trainer : a virtual environment dedicated to the technical training of milling machines in VR.

6.8.1. Pseudo-Haptic Feedback : the Simulation of Pseudo-Haptic Textures

This work studied the simulation of textures with a pseudo-haptic feedback. We proposed a new interaction technique to simulate textures in desktop applications without using a haptic interface. The proposed technique consists in modifying the motion of the cursor on the screen, when an input device is manipulated by the user, i.e. the Control/Display ratio.

Assuming that the image displayed corresponds to a top view of the texture, an acceleration (or deceleration) of the cursor indicates a negative (or positive) slope of the texture. Figure 12 illustrates this technique and displays the modification of the motion of the cursor during the simulation of a circular bump. The bump is displayed on the screen in top-view, i.e. as a disk. When climbing the bump, the speed of the cursor decreases. Once the center of the bump is reached, the speed of the cursor increases. The simulation of a hole is achieved inversely.

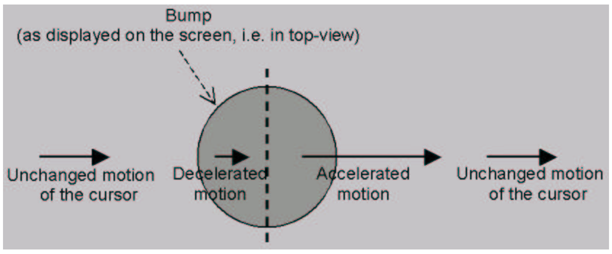

Figure 12. Pseudo-Haptic Textures : Modification of the speed of the cursor when passing over a bump.

Experimental evaluations showed that participants could successfully identify macroscopic textures such as bumps and holes, by simply using the variations of the motion of the cursor. Furthermore, the participants were able to draw the different profiles of bumps and holes which were simulated, correctly. These results suggest that our technique enabled the participants to successfully conjure up a mental image of the topography of the macroscopic textures.

This work lead to a publication at ACM CHI'04 [28] and to the deposition of a patent titled "Modulation of the cursors position in video data for the computer screen" [44]. The French patent was deposited on October 26th 2003 (patent number: 0311302). The extension of the patent to the European Community was proposed on October 2004. Many applications are foreseen, such as: the feeling of images (pictures, drawings) or GUI components (windows edges, buttons), or the improvement of navigation and visualization of scientific data.

6.8.2. Influence of Haptic Feedback on Perception of Self-Motion in Virtual Reality

This work was achieved as an external collaboration supervised by College de France (LPPA of Pr. Alain Berthoz) and CEA LIST (French Commission for Atomic Energy).

In Virtual Environments (VE), navigation is simulated almost exclusively with visual stimulation. Most of the other sensory stimulations which are usually present when navigating in real life are generally absent in VR. The missing information concerns mainly three different types of sensory feedback: the proprioceptive feedback, the vestibular feedback, and the copy of the corollar discharge. The main objective of this study was thus to study the possibility to substitute or simulate the missing information (mainly proprioceptive and vestibular sensations) with another sensory feedback: a haptic feedback sent in the user's dominant hand. Two theoretical questions were raised:

1. Could haptic feedback substitute for the missing sensations of navigation in VR (i.e. proprioceptive and vestibular information)?
2. Could haptic feedback improve the perception of self-motion, the memorisation of a trajectory, and the creation of cognitive maps?

Figure 13. Passive Navigation inside a Virtual Tunnel.

An experiment was thus conducted to evaluate the influence of haptic feedback on the perception of self-motion in virtual reality. Participants were asked to estimate the angles of turns made during a passive visual navigation (see Figure 13). Sometimes, during a turn, a haptic feedback was sent to the dominant hand of the participants. This haptic feedback consisted in rotating the participants fist by the same angular value as the visual turn (see Figure 14).

The presence of haptic feedback globally influenced the performances of the participants. On average, with haptic feedback, the participants less frequently under-estimated the angles the turns made in the visual navigation. These results suggest that the perception of self-motion could be improved in virtual reality by using a appropriate haptic feedback. Haptic stimulation during navigation could partially substitute for the missing information provided by proprioception and vestibular system. These results were published at Haptics'04 [29].

Figure 14. Experimental Set-Up.

6.8.3. The Virtual Technical Trainer : a Vocational Training Tool in Virtual Reality

This work was achieved as an external collaboration with a consortium of industrial and academic partners: CLARTE (Centre Lavallois de Ressources Technologies), AFPA (Association Nationale pour la Formation Professionnelle des Adultes), and University of Paris 5. It is also related to the RNTL french platform for Virtual Reality "PERF-RV".

Hundreds of people are trained to the use of milling machines in AFPA centers (AFPA : French National Association for Vocational Training of Adults) each year. Learning with a milling machine is a long and complex process. It is expensive since it requires a large amount of materials and it implies maintenance costs.

Therefore, we proposed a new system called the Virtual Technical Trainer (VTT). This system is dedicated to the technical training of milling in Virtual Reality. More specifically, VTT simulates the milling activity. VTT provides milling trainees with a real haptic feedback using a PHANToM force-feedback arm (see Figure 15). This force feedback is used to simulate resistance, when the tool mills the material.

Figure 15. VTT : (left) PHANToM-Based Solution, (right) Pseudo-Haptic Solution with a SpaceMouse.

We also investigated the use of pseudo-haptic feedback to simulate force feedback within VTT. A pseudo-haptic feedback was incorporated in the VTT environment by using a passive input device (a SpaceMouse) which was associated with the visual motion of the tool on the screen. It is possible to simulate different sensations of resistance, by appropriately modifying the visual feedback of the tools motion.

A preliminary and informal evaluation of VTT was made with a group of 28 AFPA trainees at the beginning of their training period. It was made with the PHANToM device since it was the only solution technically available at this time. Trainers were present and lead the session. Trainees were asked to mill a virtual work-piece and to make oral comments about the prototype. The results were globally positive. Trainees were enthusiastic about the technology. However, this informal evaluation also showed the current drawbacks of the PHANToM solution. These drawbacks should be avoided by using the pseudo-haptic solution.

This work was published at VRIC'04 [37] and EuroHaptics'04 [20].

6.9. Brain-Computer Interaction in Virtual Reality

Keywords: *Brain-Computer Interface, EEG, LORETA, Neurofeedback, Real-Time, Visualisation*.

Participants: Anatole Lécuyer, Marco Congedo, Fabrice Lamarche, Jean-Eudes Marvie, Bruno Arnaldi.

Brain-Computer Interaction consists in using the cerebral activity of a person to directly control a machine (e.g. a robot, a computer, or a Virtual Reality simulation). We describe hereafter our recent results in this field: a virtual environment called OpenViBE (Open source Virtual Brain Environment) for the 3D visualisation - in virtual reality - of the whole brain activity in real time, using an EEG (Electro-Encephalo-Graphy) acquisition machine.

When the physiological activity of the brain (e. g., electroencephalogram, functional magnetic resonance imaging, etc.) is monitored in real-time, feedback can be returned to the subject and she can try to exercise some control over it. This idea is at the base of research on Neurofeedback and Brain-Computer Interfaces. Current advances in the speed of microprocessors, graphics cards and digital signal processing algorithms allow significant improvements of these methods. More meaningful features from the continuous flow of brain activation can be extracted and feedback can be more informative.

Borrowing technology so far employed only in Virtual Reality, we have created Open-ViBE. Open-ViBE is a general purpose platform for the development of 3D real-time virtual representation of brain physiological and anatomical data. Open-ViBE is a flexible and modular platform that integrates modules for brain physiological data acquisition, processing, and volumetric rendering.

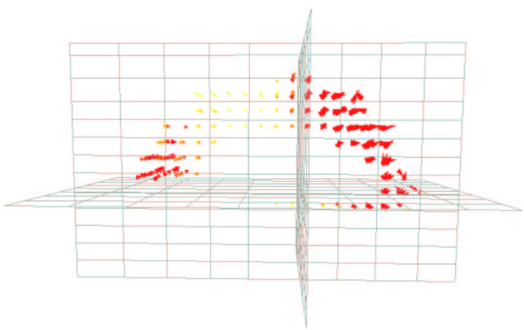

Figure 16. Brain activity visualization with a solution space restricted to the cingulate gyrus.

When input data is the electroencephalogram, Open-ViBE uses the estimation of intra-cranial current density to represent brain activation as a regular grid of 3D graphical objects. The color and size of these objects co-vary with the amplitude and/or direction of the electrical current. This representation can be superimposed onto a volumetric rendering of the subject's MRI data to form the anatomical background of the scene. The user can navigate in this virtual brain and visualize it as a whole or only some of its parts (see Figures 16 and 17). This allows the user to experience the sense of presence ("being there") in the scene and to observe the dynamics of brain current activity in its original spatio-temporal relations.

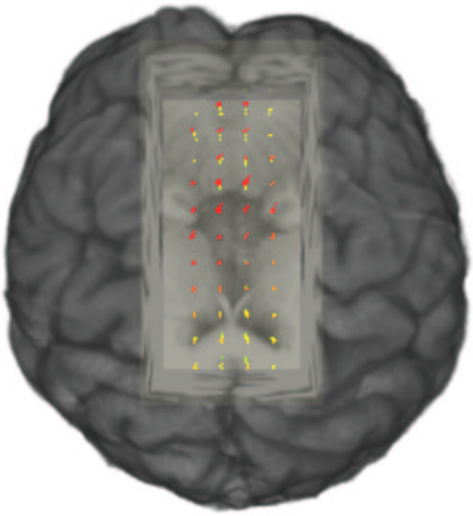

Figure 17. 3D texture rendering of individual MRI (T1) slices. Part of the brain is clipped by a parallelepipedic transparent object allowing the user to visualise the cingulate gyrus. The brain is seen from the top.

The platform is based on publicly available frameworks such as OpenMASK and OpenSG and is open source itself. In this way we aim to enhance the cooperation of researchers and to promote the use of the platform on a large scale.

This work was accepted for publication in Journal of Neurotherapy [11].

6.10. Virtual reality to analyze interaction between humans

Keywords: *Human Motion, Motion Understanding*.

Participants: Franck Multon, Richard Kulpa, Stephane Ménardais, Bruno Arnaldi.

Virtual reality was previously used in several domains to train people performing a costly and complex task (such as repairing complex structures or driving vehicles). In such applications, metaphors are generally used to interact with virtual objects and the subjects consequently do not react exactly as in the real world. In these applications, the feeling of being there (called presence) that ensures realism can thus only be analyzed through questionnaires. In sports, realism and presence can also be evaluated through the gestures performed by the subjects. Let us consider the thrower and goal-keeper hand-ball duel. Previous results [19] demonstrated that real goalkeepers react realistically to virtual opponents. We also verified that a small modification in the opponents' gestures engendered modifications in the goalkeepers' parry whereas no modification was found for a same throw repeated two times. In neuroscience and sports science, anticipation is a control skill involved in duels between two players. According to elements considered in an opponent's gestures, people are able to predict events that will occur in a near future. In [13], we demonstrated that this phenomenon is also recovered in a virtual environment. In this environment, the opponents' gestures are animated through a kinematic model that could engender unrealistic trajectories. Nevertheles, the animation module, even if it is based on simplifications, seems to reproduce the visual elements considered by goalkeepers. Future works will focus on two main directions. The first one deals with considering the minimum animaiton level of detail required to recover this natural anticipation skill. The second one deals with capturing the goalkeepers' motions in real-time and making the synthetic opponent react as a consequence. Hence, in this approach, the virtual opponent would be capable of getting elements in the real goalkeeper's motions.

6.11. Real-time animation of virtual humans with motion capture

Keywords: *Human Motion*, *Motion Capture*.

Participants: Franck Multon, Stephane Ménardais, Bruno Arnaldi.

We developed a new process to animate virtual humans thanks to motion capture. A simplified skeleton was proposed to store motion capture trajectories. This skeleton deals with the positions of the skeleton extremities with respect to a proximal point (such as the wrist trajectory in the shoulder reference frame). This data is normalized by the segment length in order to obtain a-dimensional values. The intermediate articulations that are not directly taken into account can be retrieved by inverse kinematics. To help the inverse kinematics algorithm and avoid redundancy, the plane containing the segment is also stored (such as the plane containing the wrist, the elbow and the shoulder). Hence, adapting the movement to a new character involves to scale the a-dimensional data to fit the given anthropometric data. A second step deals with the adaptation to the environment by displacing the extremities in order to ensure spacetime constraints. Those constraints are specified by the user for each movement. For example, in human locomotion, a user can specify the contact phases and associate for each of them a constraint that ensures foot contact and avoids sliding. The constraints are taken into account in real-time. Finally, the remainder of the skeleton is retrieved by using inverse kinematics. To blend several motions with each other, we also developed a new method based on the definition of elementary motions. Those elementary motions are composed with the a-dimensional kinematic data (as described above), a state (activated or not) and a priority. The priority is associated to a set of joints. For example, locomotion has high priorities on the legs but slightly influences the upper-body. On the opposite, grasping involves the upper-body but does not influence the lower-body. Each priority and state change continuously depending on time. Hence, the final motion, blended with several elementary motions, is a weighted sum of all the trajectories, joint per joint. Nevertheless, synchronization must be ensured in order to avoid unrealistic movements: blending a movements involving a left foot-contact and another involving right foot-contact. The results show that this system enables to animate a human-like figure with several pre-recorded motions that can be adapted to the skeleton and the environment in real-time.

Thanks to this approach, it is possible to control captured motions in order to deal with high-level orders such as going through a direction, increasing or decreasing speed, and making body parts reach some specific points in the Cartesian reference frame. Hence, we developped an intermediate interface that makes it possible to specify such high-level orders and to calculate the required motion adaptations. To this end, constraints are designed on one or more joints: changing the root global displacements, ensuring foot contact on footprints. We can control every kind of motion that involves foot-contacts in order to make a virtual human move in a dynamic environment. For example, a drunk person can adapt her perturbated gait in order to follow an interactively driven target (see figure 18). It is also possible to make a karateka follow a given trajectory on the ground thanks to a sequence of kicks. As a future work, we wish to improve the inverse kinematics solver to deal with constraints on every part of the skeleton instead on only joints.

6.12. A model of hierarchical spatial cognitive map and human memory model dedicated to realistic human navigation

Participants: Romain Thomas, Stéphane Donikian.

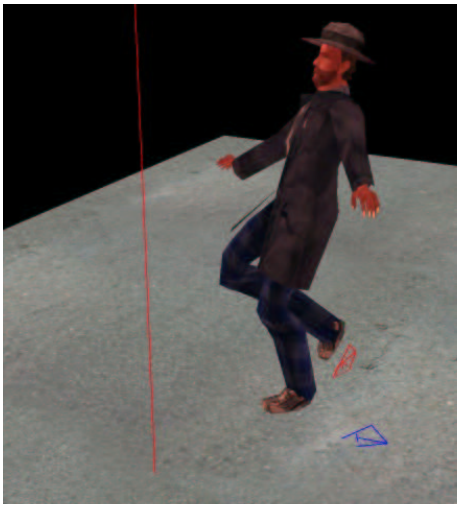

Figure 18. Screenshot of a drunk person following a vertical red target by adapting motion capture in real-time.

In the behavioral animation field of research, the simulation of populated virtual cities requires that agents are able to navigate autonomously through their environment. It is of interest to tend to the most realistic human-like planning and navigation. In order to do so, we have designed a navigation system for autonomous agents, which implements theoretical views from the field of human behavior in urban environments. Concerning the perception of the environment, models used in behavioral animation has mainly focused on the visual field to filter what is viewed inside a global geometric database. Information used to navigate has been considered as identical for all autonomous characters and as corresponding to an exact topographic representation of the environment. However, in reality each person has a unique representation of a city map depending on her past experience and on her knowledge of the city. Moreover this cognitive map will evolve with time.

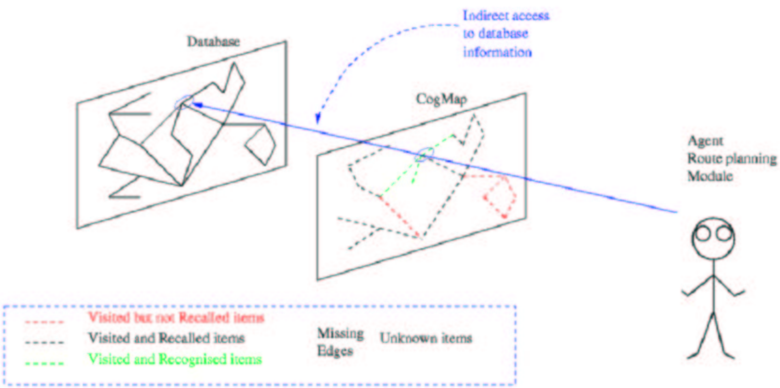

Figure 19. Database/Cognitive Map relation.

We proposed a new model which allows to represent, for each autonomous agent, an individual hierarchical cognitive map merged with a simple human-like memory model for navigation simulation in an urban environment. It allows to implement navigation as a planned and reactive navigation loop to be computed alternatively.

As shown in Figure20, the system is compounded of 5 different modules:

The database which represents the environment and stores all the data related to it. We modeled the urban environment as a database via an a semantically and geometrically Informed Hierarchical Topological Graph (IHT-Graph).

The cognitive map which "filters" the information of the environment. Our model of cognitive map has a topological and hierarchical graph structure which partially maps the regions of the environment the agent has explored during the simulation. This map can be seen as a filter on the environment. It does not contain geometrical nor semantic information about the urban objects encountered, but only controls the partial access to the database when the agent recalls or perceives the urban objects.

The memory controller which manages the memory in the cognitive map. As a simplified model of human memory, we use the recall and recognition attributes, and their respective thresholds of activation to parameterize the cognitive map in two different ways. The memory model is merged with the cognitive map under two forms: a **long-term store** mechanism and a **short-term** one. Concerning the long-term memory, each object of the map is endowed with **Recall** and **Recognition** parameters in order to manage the retrieval of information. Links between objects are parameterized through the **graph of landmarks** which guarantees an associative memory mechanism to the system. The short-term memory respects the Milner rules on its capacity (7 ± 2 elements) and stores subgraphs of the cognitive map, linked together in a mereo-topological way.

The route planning module which implements the planning and navigation algorithms. The agent plans its route using its cognitive maps, knowing that most of the time, the navigation plan is not complete enough to reach its destination, it has to switch between two different types of navigation: **planned navigation** and **reactive navigation**.

The navigation module based on the HPTS decisional system which manages the behaviors of the agents in the environment, which underlies the composition of the **planned and reactive navigation algorithms**. The first one uses the plan generated using the information gathered in the cognitive map, while the second one, holds for the navigation when the agent is lost (i.e. the agents fell into the critical decision cases ensuring a local loss in the environment). The reactive navigation mode, following first main axes, allows the agent to plan a new route taking into account the new information gathered. During the navigation, the pedestrian meets relevant urban objects not recalled but recognized, which trigger the recognition of a region of the map.

6.13. Microscopic crowd simulation

Participants: Fabrice Lamarche [contact], Stéphane Donikian [contact].

We propose a system able to simulate several hundreds of autonomous pedestrians in indoor and/or outdoor environments. This system is based on a spatial subdivision algorithm extracting topological information, a fast hierarchical path planning algorithm, an optimized structure for collision prediction and a local optimization algorithm which can be configured to translate different navigation rules inspired by pedestrian behaviour studies.

Virtual human navigation inside virtual environments has a key role in behavioral animation. This process is continuously used for several sorts of interactions (moving to take something, to watch something...). Based on the analysis of studies on pedestrian behavior, we propose a generic and real time model able to simulate several hundreds of autonomous agents navigating in indoor and/or outdoor environments. This model is based on four sub models: a spatial subdivision model, a fast hierarchical path planning algorithm, a neighborhood graph and a reactive navigation controller [16][10].

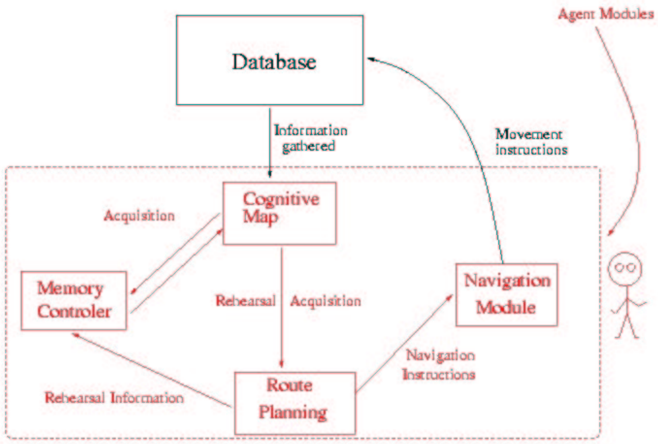

Figure 20. Architecture of the cognitive agent.

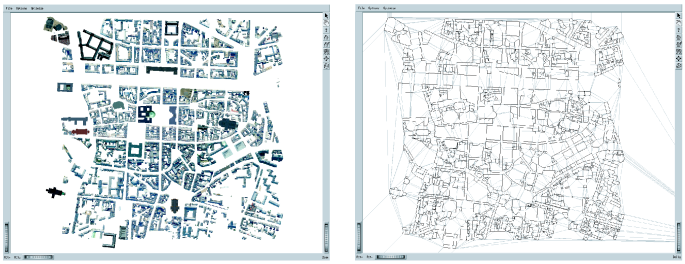

Figure 21. Spatial subdivision of the center of Rennes.

The spatial subdivision process extracts convex cells from the geometric database of the environment (Cf. fig. 21). During the subdivision, a specific algorithm computes all bottlenecks to identify critical regions in the map. The generated subdivision is automatically informed with topological properties. This information is used to generate a hierarchical topological map that provides a suitable structure to create a hierarchical path-planning algorithm. This algorithm is able to compute, in real time, paths inside very large environments. It is used to generate paths to follow for navigating agents. A second aspect of navigation is collision prediction. In order to limit complexity, we globally compute a neighborhood graph filtered with visibility (in real time thanks to the spatial subdivision). This graph is computed with a complexity of O(n ln n) for n entities and contains O(n) neighborhood relations. Those relations, expressing the proximity and the visibility of agents, are only dependent on the density of the crowd but not on a maximum prediction distance. This property allows the automatic adaptation of the prediction distance: near prediction in dense crowds and far prediction in sparse crowds. Neighborhood relations are then used to predict future collisions. When a collision is predicted, a local optimization algorithm tries to find the best direction and speed to adopt to follow the path while avoiding collisions. This algorithm is configured with modules responsible of the computation of a solution avoiding a certain type of collision. By interchanging the modules, it is possible to create different styles of navigation and map them on real studies on pedestrian behaviour.

This system is able to handle the realistic simulation of 2400 autonomous pedestrians (Cf. fig. 22) planning their path and avoiding each other and navigating in the center of Rennes city (1.3km x 1.3km). This simulation runs in real time on an AMD Athlon XP 1800+ with a RADEON 8500 graphics card.

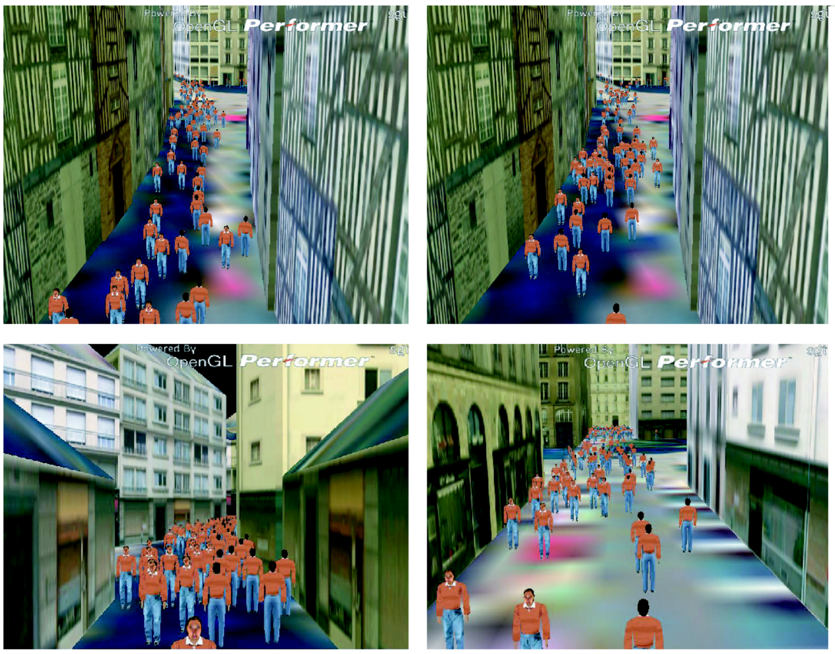

Figure 22. Examples of populated streets in Rennes.

6.14. Crowd simulation inside exchange areas, with levels of services characterisation.

Participants: Sébastien Paris, Stéphane Donikian [contact].

Crowd simulation is an emergent problem nowadays, because of a growing interest on behalf of industries and certain organisations. Many studies have been performed, mostly using macroscopic models, or *light* microscopic ones (like particle based). We propose, in our model, to simulate crowd motion by association of a multitude of individual behaviours, and to analyze results in order to extrapolate a levels of services classification.

This study is carried out within the framework of an industrial thesis in collaboration with AREP, *Aménagement Recherche pour les Pôles d'Echange*, which is a subsidiary company of the SNCF, *Société Nationale de Chemins de Fer*.

The goal of this study is to validate train station architectural plans with respect to the movements of people inside the station. That is made possible thanks to a tool for crowd simulation allowing data extrapolation which will then be synthesized in levels of services.

6.14.1. Levels of services

The concept of "Levels of services" is not a recent concept. Firstly defined by *J.J. Fruin* in 1971, it was reused by many researchers. But all of them made their classification only with two discriminating factors: density and flow of people. Such a classification seems well suited for security studies, but suffers a lack of information for thorough ones.

What we take into account with our levels of services, in addition to classical ones, is:

- Information accessibility to a pedestrian inside a given place, to help her find her way for example;
- Distance to travel to carry out a certain number of necessary tasks, shortest being best;
- More specific quality factors, like average waiting time.

The last point to be approached is the fact that a level of service is not evaluated on an overall basis for the studied place, but locally at each zone of interest.

6.14.2. Crowd simulation

Our model is based on a microscopic simulation of each actor, taking into account social configurations like family. These actors, while interacting, will produce, starting from a certain density, the observable macroscopic phenomenon called *crowd*.

So, one actor behavior should not be chaotic nor random, but governed by a final need. Such a behavior is simulated by a *goal oriented* model, managing interactions at the same time with other actors and objects of the environment.

Interaction with objects

We have chosen a representation close to Gibson's affordances with regard to the objects. Indeed, each object *contains* all the interaction possibilities it offers, which we call *relations*.

A relation is made up of three parts: *duration*, *pre conditions*, and *post conditions*. We will not insist on the duration which only represents the necessary time to carry out the relation.

The pre conditions are Boolean expressions to satisfy in order to process the relation, and post conditions are the results obtained afterward. Therefore, we have schematically an expression for test and an expression for result.

Variables inside these expressions are called *conditions*. They can be of two types: local to the actor, like her money, or global to the system, like the current time.

So, if we take a relation, we should be able to link it to other ones by its pre and post conditions. But we can notice that there can be several ways to obtain a relation, like several needs for the same one. This observation leads to the use of three operators:

- Succession, $A \longleftarrow B$ meaning A obtained by B;
- Disjunction, $A \mid B$ meaning either A or B is necessary;
- Conjunction, $A \& B$ meaning both A and B are necessary.

Finally, by giving a post condition for goal, one can build a logical tree of the relations to be achieved to obtain it. The leaves of this tree are base relations that cannot be linked, in other words the *entry of the system*.

Note that contrary to the affordances, the behavior associated with each object is not used individually during simulation, but only at initialization to build a goal oriented global procedure including all necessary objects.

This stage leads to the construction of a macro hierarchical automaton managing the procedural aspect of the behavior of a typology of actor.

Interaction with other actors

The first basic interaction between two actors, *collision avoidance*, results directly from *F. Lamarche* work on microscopic crowd simulation.

With regard to other interactions, like *social structures*, *imitation* (leading to crowd phenomenon), or others, they are part of our work in progress.

6.15. BCOOL : Behavioral and Cognitive Object Oriented Language

Participants: Fabrice Lamarche [contact], Stéphane Donikian [contact].

BCOOL stands for Behavioral and cognitive Object Oriented Language. This language is dedicated to the description of the cognitive part of an autonomous agent. It provides object oriented paradigms in order to describe reusable cognitive components. It focuses the description on the world and the interactions it provides to agents. A stable action selection mechanism uses this representation of the world to select, in real time, a consistent action in order to fulfill a given goal.

In the field of behavioral animation, the simulation of virtual humans is a center of interest. Usually, the architecture is separated in three layers: the movement layer (motion capture and inverse kinematics), the reactive layer (behaviors) and the cognitive layer. The role of the cognitive layer is to manipulate an abstraction of the world in order to automatically select appropriated actions to achieve a given goal. BCOOL is dedicated to the description of the cognitive world of the agents [10]. Inspired by Gibson's theory of affordances, it focuses on the description of the environment and the opportunities it provides, under a form allowing goal oriented action selection. A stable action selection algorithm uses this description to generate actions in order to achieve a given goal.

The language provides object oriented paradigms to describe the cognitive representation of objects populating an environment. The notion of class is used to describe different typologies of objects and the notion of inheritance allows specializing the description. Objects are described through properties and interactions similar to methods in object-oriented languages. Notions of polymorphism are exploited to redefine interactions and specialize them through the inheritance hierarchy. Notions of relations between object instances are also provided. Relations and properties are boolean facts describing the world. A specific operator enabling incomplete knowledge management has been added. It enables reasoning on the knowledge of the truth value of a fact in a similar way as the fact itself. The description of actions uses preconditions and effects to allow planning and is also informed with C++ code describing an effective action called once the action is selected. Thanks to this property, the cognitive process can be easily connected to the reactive model in charge of the realization of selected actions. Thanks to knowledge operator, perceptive and effective actions are described in the same way. Thus, perceptive actions can be selected to acquire some necessary information during the planning process. Once the abstract world is described, a second language is used to describe a world populated of instances of cognitive objects. This description is used to generate a database describing the world, the relations and the actions that can be performed by agents. This database is then exploited by the action selection mechanism to select, in real time, actions in order to fulfill a given goal. The mechanism is able to handle three types of goal:

- Avoidance goal: those goals are used to specify facts that should never become true as a consequence of an agent action.
- Realization goal: those goals are used to specify facts that should become true.

- Maintain goal: those goals allow specifying facts that should always stay true inside the environment.

Once the goals are provided, the action selection mechanism select actions and calls their associated C++ code to run associated reactive behaviors. Actions are selected incrementally in order to take into account all perceived modifications of the world in the next selection. This way, the mechanism is goal oriented and reactive.

BCOOL provides a high level framework, focusing on the description of reusable cognitive components while providing easy connection with the reactive model. The incremental generation of actions allows to handle the dynamics of the world by taking into account all perceived modifications during the action selection phase. Its aim is to provide a generic and real time framework taking into account dynamic constraints imposed by behavioral animation.

6.16. Synoptic objects allowing autonomous agents to interact with a virtual environment

Participants: Marwan Badawi, Stéphane Donikian [contact].

We propose the STARFISH (Synoptic-objects for Tracking Actions Received From Interactive Surfaces and Humanoids) architecture that uses Synoptic Objects to allow real-time object manipulation by autonomous agents in an informed environment. We define a minimal set of primitive Basic Actions which are used to build Complex Actions. We then assign these actions to Interactive Surfaces which are the parts of an object's geometry that are concerned by the action. The agent then uses these Interactive Surfaces to get the data specific to the object when it wants to manipulate it and to adapt its behavior accordingly.

STARFISH stands for Synoptic-objects for Tracking Actions Received From Interactive Surfaces and Humanoids. It a new system which allows the easy definition of interactions between autonomous agents and the Synoptic Objects in the environment.

Synpotic Objects are objects designed to offer to the autonomous agent a summary, or synopsis, of what interactions they afford. When an agent queries such an object, it knows what actions it can perform, where it should position itself, where it should place its hands, what state the object is in, whether it is allowed to perform the action, etc... All these indications, are given through the use of STARFISH Actions and Interactive Surfaces.

STARFISH Actions consist of a group of simple atomic actions, the Basic Actions which are the building blocks used to create Complex Actions. Through the Basic Actions described in Figure 23, we can build more complex and varied actions. Note that for the remainder of this section, whenever we refer to a Basic Action, its name will be typeset in **bold face** to differentiate it from the corresponding verb.

Transfer	Transfer location of an object (go, carry)
Move	Move own body part (kick, reach)
Grasp	Grab an object (grasp)
Ingest	Take an object into own body (eat)
Speak	Produce sound (say, sing)
Attend	Focus sense organ (listen, look at)

Figure 23. The complete set of Basic Actions.

For example, the *Open Door* Complex Action (illustrated in figure 24) can be easily decomposed into its Basic Actions:

- **Transfer** self to the door.
- **Move** arm towards knob.
- **Grasp** knob.
- **Move** hand to turn knob.

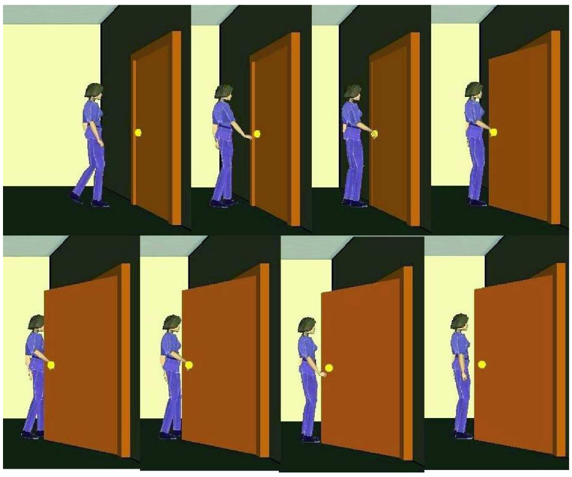

Figure 24. An agent pulling open a door.

- **Displace** door into the open position.
- un-**Grasp** to let go of the knob.

Interactive Surfaces are then used to model and describe the affordances of an object. They are generally parts of the object's geometry that act as hotspots when a certain action is to be accomplished, such as where to place the agent's hand when **Grasp**ing the doorknob or where to stand in order to open that door(Figure 25). Each Synpotic Object can have as many Interactive Surfaces as needed, depending on how much interaction it offers.

Seeing how the Interactive Sufaces and STARFISH Actions are two completely independent concepts, we do not have to go through with them both when creating a new Synpotic Object. When an object with a new shape but with the same functionality has to be introduced into the simulation, it is only necessary to model its appropriate Interactive Surfaces without having to modify their associated behaviors.

6.17. Scripted interactions in a virtual educational environment

Participants: Bruno Arnaldi, Nicolas Mollet, Frédéric Devillers.

We designed a part of a virtual formation environment. One can find the context of our project in the section "industrial contracts", and "GVT". Firstly, we created a general pattern to obtain interactive objects and a generic interaction process. Secondly, we focused on the definition of what has to be done in this virtual environment, by using our interaction mechanism. Our objective was to create generic designs, reusable objects and behaviors, and complex scenarios.

6.17.1. A general interaction model between behavioral objects

We wanted to obtain a generic treatment of interactions between behavioral objects. A classical problem of interaction between objects is the question *where should we put the definition of the interaction?* One solution is to put the definition of the interaction in the object itself. We can mention the works of Kallmann, with what he called "smart objects", or the work on Synoptic Objects developed by Badawi in our team. We thought that it would be interesting to have information about interaction in the objects, but not have the definition of interaction totally described in a particular object. This definition has to be located somewhere between the objects, with parts of definitions distributed between the objects concerned by the interaction. These parts

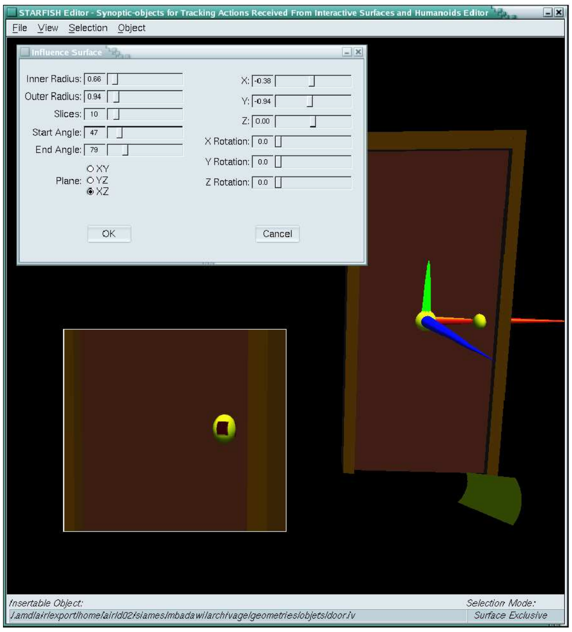

Figure 25. Interactive Surfaces for the door modeled in the STARFISH Editor.

are named “capacities”. The definition of interaction located between the objects is named “relation”. The “capacity” gives a set of interaction possibilites between the objects which have it. As for the “relation”, it contains definitions of interactions, based on sets of capacities. This work lead to the deposition of a patent. For example, let us consider only two objects, a screw and a nut. We want to create a screwing interaction between those objects. According to our model, the screw will have male-screwing capacity (length, thread pitch, the male state, etc.) and the nut will have female-screwing capacity (thread pitch, a boolean which indicates if there is already a screw on it, the female state, etc.). We will also have a “screwing relation”. This relation contains the definitions of the screwing interaction: a male-screw can be screwed in a female-screw, when the size is good, when the thread pitches match, etc. This relation offers the possibility to screw the screw in the nut, and gives a certain state to this set of two objects: when it is screwed we can not manipulate the screw independently, and when we move the nut, it moves the screw. We can notice that the “screwing relation”, the “female-screwing capacity” and the “male-screwing capacity” can be re-used whenever we need to define screwing interactions between two objects.

6.17.2. A graphical scenario language for complex interactions

Based on the precedent model of interactions, we wanted to have the ability to express a complex sequence of interactions. Thus our goal was to define the referential sequence of actions for a student in the formation environment. So we created a new scenario language. This language has two main aspects: it can be written directly, as we assume when we talk about a language. But it is also a graphical language. This language inherits from different graphical languages such as grafcets and state-charts, and also languages like HPTS for the hierarchical state machine aspect. Our language consists of steps, and links between steps. Each step describes an action which can be: an atomic action (in the previous example, we can select “screwing”), or a calculation action which uses internal variables, or a conditional action (the result of its evaluation has two possibles ways when exiting) or a hierarchical action-step (a scenario can have steps which contain a sub-scenario. Those steps describe a global action locally). This language is interpreted and dynamic. Files

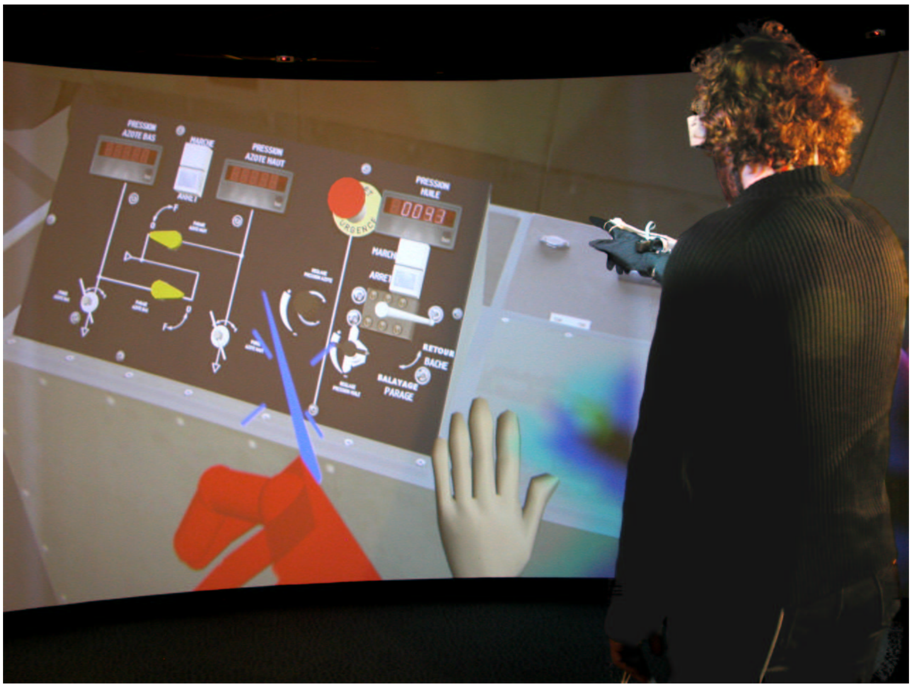

Figure 26. An application based on the “relation between objects” model

are loaded and represented in memory in a virtual machine for scenarios. The memory of this machine can be modified dynamically: the memory representation of the scenario is interpreted with the evolution of the scene. We can edit the language whenever we want in the virtual machine, anywhere in the scenario except for the representation of the current steps. After actions are executed in the environment, the virtual machine interprets next steps in its own memory. This work lead to the deposition of a patent.

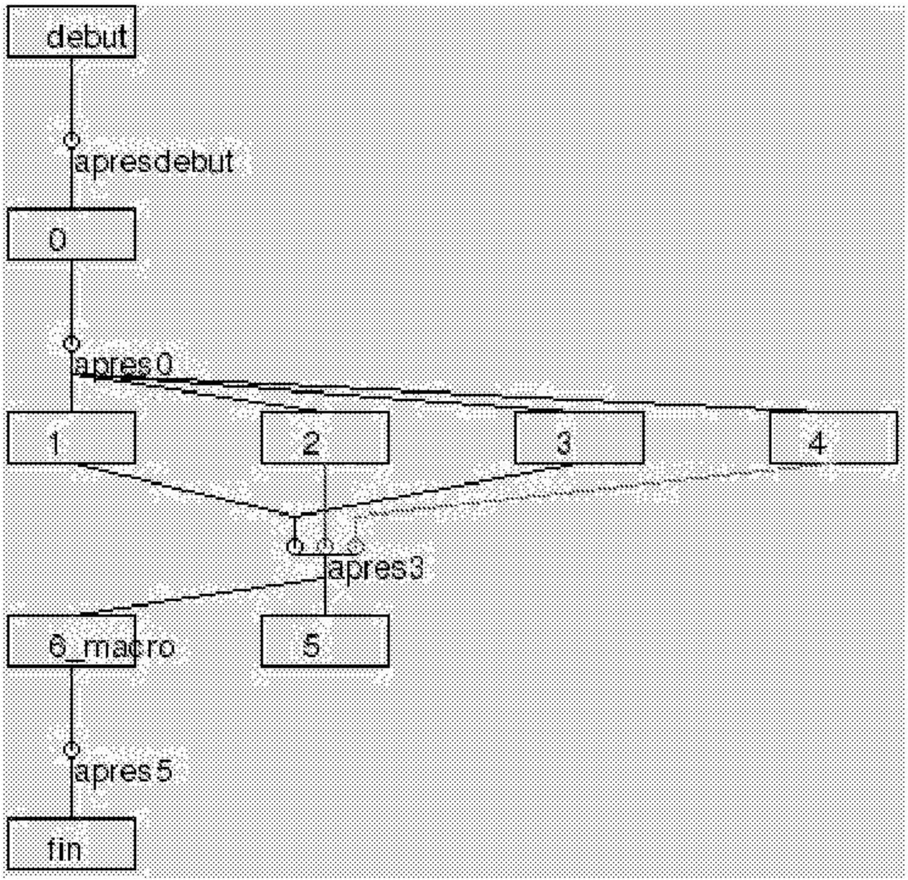

Figure 27. Graphical vision of our scenarii langage

7. Contracts and Grants with Industry

7.1. RNTL PERF-RV : French platform on Virtual Reality

Keywords: *Cooperative Work*, *Haptic Feedback*, *Immersion*, *Virtual Reality*.

Participants: Bruno Arnaldi, Alain Chauffaut, Thierry Duval, Tangi Meyer, Guillermo Andrade, Christian Le Tenier.

The main goals of the PERF-RV platform is, on one hand, to share Virtual Reality dedicated equipments (Reality Center, Workbench, haptic devices, ...) and, on other hand, to factorize knowledge on interdisciplinary industrial activities such as in automotive, aeronautic or energies industry.

The PERF-RV consortium is composed of INRIA, CEA, Ecole des Mines de Paris, ENSAM Chalon sur Saône, Labri, Laboratoire de Robotique de Paris, LIMSI, ADEPA, EADS CCR, IFP, Clarté, Dassault Aviation, Giat-Industrie, PSA and Renault.

Concerning PERF-RV, our activities are focussed on three main topics:

coordination of PERF-RV activities: B. Arnaldi is the chairman of PERF-RV. His activities concern:

- Management of the meetings and working groups,
- management of the deliverables for the Research Ministry,
- organization of the project final review for the Research Ministry and also for external researchers and industrials.

Multimodal and collaborative work: Here we go on with the integration within OpenMASK of all our software tools about navigation and interaction within 3D Collaborative Virtual Environments.
The results shown at the final project review (CEA Fontenay aux roses, the 14^{th} and 15^{th} of October) merged collaborative interactions and haptic feedback to allow users to share collaborative interactions with haptic feedback upon technical parts of a virtual mockup of a car (a Renault Mégane Scenic). This demonstration was done with three users and three computers, two of them were using haptic devices: a Virtuose and a Spidar.
This demonstration incorporates the last results about adapters (constrained and multi-users simultaneously), interactors and virtual coupling between kinematical and physical objects, and allows several users to interact with physical objects to put them in the car.

Immersion and haptic feedback: in collaboration with Renault and CEA, we have defined some relevant scenarios concerning mount and unmount manipulation in the automotive industry.
In order to propose a flexible software solution, we have designed and implemented a modular software distributed architecture dedicated to virtual reality constraints including haptic feedback. This solution is integrated in the last version of our software platform OpenMASK. In this context, the main features of OpenMASK include the creation of a shared communication environment for the different software components used in VR (collision detection, mechanical simulation, device integration, ...).
We have demonstrated this approach by some relevant experiments. For example, we can integrate quite different simulators such as *CONTACTTolkit*, *SMR* and *DYNAMO*, in order to allow them to interoperate. *CONTACTTolkit* contains a 4D continuous collision detection algorithm and contact solver (Inria Rocquencourt, I3D team). *SMR* is a hierarchical rigid multi-body system simulator(LAAS/CNRS and LRP). *DYNAMO* is a non-hierarchical mechanical constraints solver(Eindhoven University). This feature is fundamental in order to access high performance in industrial test-case (automotive or aeronautic industry).
We have also demonstrated the capacity to integrate haptic devices *Virtuose 6D* (Haption Company, see figure 28) and Spidar (Tokyo University) in our software architecture.

These demonstrations have been shown at Imagina 2004, Linux Expo 2004, and at the final PerfRV review.

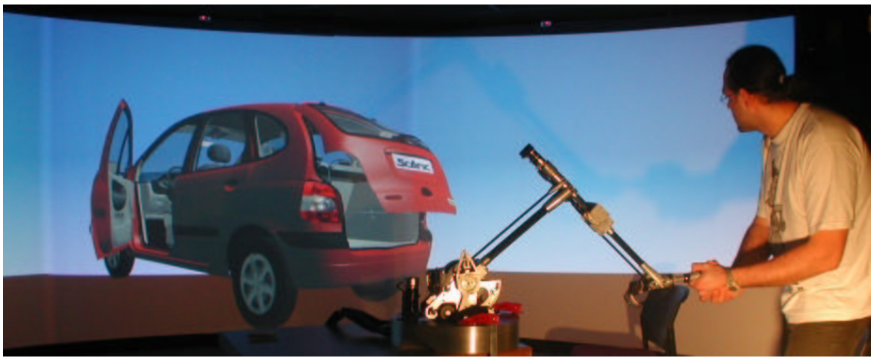

Figure 28. Direct immersive manipulation with haptic device Virtuose 6D.

7.2. RIAM AVA Motion.

Keywords: *Behavior Coordination*, *Motion Control*, *Real-time Animation*, *Virtual Human*.

Participants: Stéphane Ménardais, Richard Kulpa, Franck Multon, Stéphane Donikian [contact].

The AVA Motion program is a partnership with the Laboratory of Biomechanics of the University of Rennes 2 and with the Dæsign and Kineo Cam companies. It was supported by the RIAM (French National Network for Research and Innovation on Multimedia) until june 2004. The main objective of this program is to develop a middleware dedicated to real-time virtual humans (cf. figure 29). It is composed of three layers:

- Path-planning: it determines the best path for a character to go to a specific place. It is based on robotics algorithms. This part is handled by Kineo Cam company;
- Motion control: it handles the motion of the avatar in order to follow the desired path and to make some actions. This is the main part of our contribution to the project;
- Reactive behaviors: it manages the behavior of the actor. It is the top-most layer of this project. It makes the decisions and consequently sets the parameters of the two previous modules as well as their activation and deactivation. This part is made by the end-user partner: the Dæsign company. In order to help them handling several resources shared by many actions and characters, we have integrated HPTS++ inside their behavioral module.

During this project, the MKM (Manageable Kinematics Motions) library was created (cf. 5.3). In addition to this animation engine, we have developped two models: a walking model (cf. figure 30) and a grasping model. They use a reference motion and adapt it according to high level parameters.

MKM uses motions for the human animation. These motions can come from captured or synthetized motions. For the motions created by the infographists, we have developped a 3DSMax plugin (cf. figure 31). It automatically saves a motion to the s4d file format. This is the internal format of the motions. It allows an automatic adaptation to a different morphology and to external constraints such as the environment. MKM can also directly read BVH files, a widely used file format.

Finally, in order to use MKM in a context of productivity, we have created the s4DMaker software (cf. figure 32). It allows to create the s4d motion files from captured or synthetized motions. It also handles additional constraints such as the sequence of support phases which is used during the synchronization process.

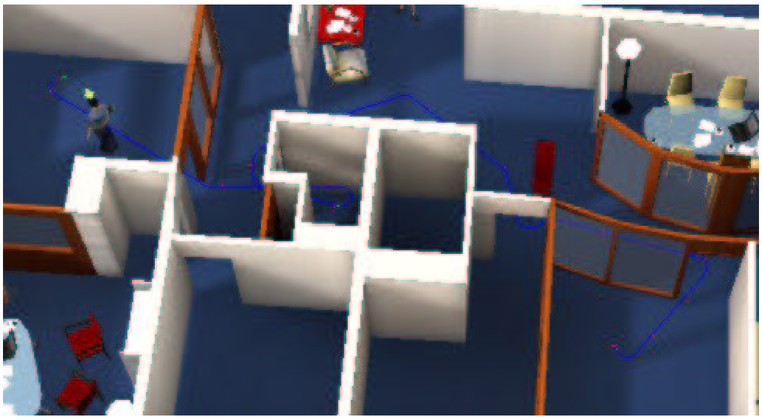

Figure 29. Application of the project AVA-Motion: Edward walking in an apartment.

Figure 30. The walking model used in a dynamic environment.

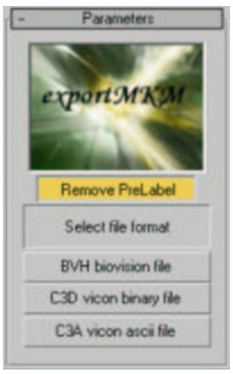

Figure 31. Plugin 3DSMax to export s4d motion files.

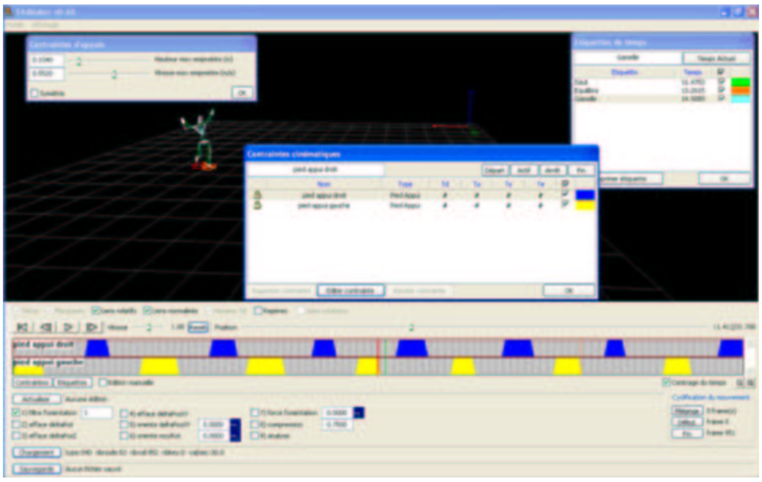

Figure 32. S4dMaker software: creation of s4d motions. It allows automatic morphological and environmental adaptations.

7.3. Giat-Industrie: Virtual Training.

Keywords: *Interaction, Scenario, Training, Virtual Reality*.

Participants: Bruno Arnaldi, Frédéric Devillers, Nicolas Mollet.

The Giat Virtual Training project (INRIA, Giat-Industries and ENIB) is a very challenging one. Indeed, in this project, we introduce very advanced VR technology in order to produce very customizable VR applications dedicated to industrial training. Our work is based on OpenMASK, the VR-platform of the Siames team (INRIA), and AReVi (ENIB-Li2 VR-platform). All our software is totally re-usable in industrial training, and is not dedicated to a particular industrial equipment or procedure. We focus our activity into the following points:

- design of true reactive 3D objects with embedded behaviors.
- design of a high level specification language in order to describe complex human activity (virtual activity in relationship with the real activity).

For those two points, you can find more informations in the "new results" section. Our partner ENIB is concerned by the pedagogic point of view of the training. The main goal of this overall project is to produce a real application in order to validate all the new concepts we introduce. This application has still been shown at two meetings: Eurosatory and Perf-RV, and is now in a validation process. The GVT project leads to the depot of 5 patents.

7.4. RNRT VTHD++

Keywords: *3D interactions, Awareness of the network latency within CVEs, Collaborative Virtual Environments, Collaborative interactions*.

Participants: Thierry Duval, Chadi Zammar.

Within this contract that is following the VTHD contract, we work to make our OpenMASK distributed kernel more tolerant to network problems. Thanks to this new kernel, we can enable a weak synchronization between the different processes involved in a collaborative simulation. Then, we can visualize the differences

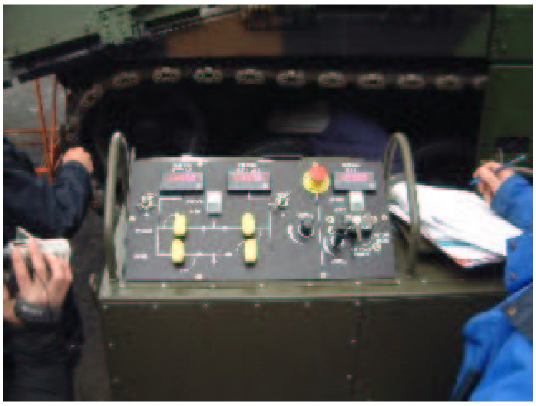

Figure 33. Maintenance training (real).

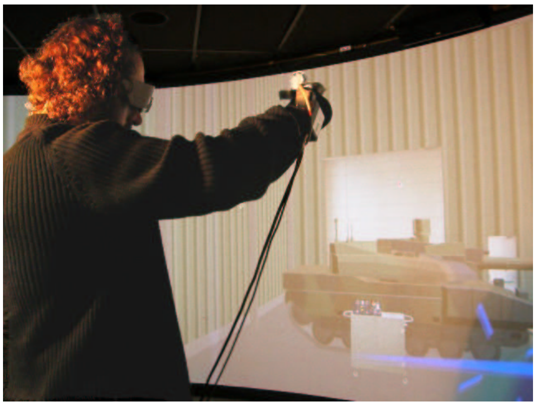

Figure 34. Maintenance training (virtual).

between the simulated objects located on different sites, making the end-users aware of the network problems. Our aim is to provide tools that would allow to evaluate the capabilities of the VTHD++ network for rapid rerouting and dynamic provisioning.

Here our aim is to use our collaborative OpenMASK kernel to create multi-sites and multi-users 3D collaborative applications upon the VTHD++ network.

The end-users of the CVEs are to be aware of the problems due to the network. These problems such as latency or temporary breakdowns can make their view of the world inconsistent with the views of the other end-users.

It is the reason why we offer the possibility to visualize the differences between referentials and mirrors: to make a user, located on the same network node than a simulated object, aware of the fact that the other users may perceive this object in a different way (at a different location for example), because of the latency introduced when the system updates the mirrors of a referential. This kind of awareness must allow an end-user to perceive fluctuations of the network latency, and it should allow to validate the QoS obtained with the dynamic provisioning service, because it could show the instantaneous QoS provided by the network.

We also want to allow the use of CVEs with OpenMASK even during network breakdowns, thanks to our new modified kernel. Then, we want to make the users aware of that kind of problem, and inform them that collaborative work is still possible, within some limitations, waiting for the network to come up again. This should allow to validate the correct behavior of the rapid rerouting service offered by VTHD++.

We also have improved the distributed OpenMASK kernel to take into account the particularities of the collaboration between very distant sites linked with high-speed networks. Our software tools have been tested on a local network; the new distributed kernel and some of these new tools have already been deployed on VTHD++, between Rennes and Grenoble, allowing us to make collaborative interactions with haptic feedback between two remote haptic devices: a Virtose in Rennes and a Spidar in Grenoble. Such a demonstration has been shown at the last VTHD++ meeting in Rennes, and a video of this demonstration will be used for the final VTHD++ review.

7.5. ROBEA ECOVIA : Study of Visuo-Haptic Integration

Keywords: *Haptic*, *Perception*, *Robotics*, *Sensory Integration*, *Virtual Reality*, *Vision*.

Participants: Anatole Lécuyer, Marco Congedo.

The aim of the ROBEA Project "ECOVIA" is to study human perception and integration of visual and haptic information. Expected results should improve computer-human interaction in robotic systems - during tele-operations, or in virtual reality systems.

The aim of the ROBEA Project "ECOVIA" is to study human perception and the integration of visual and haptic information. Expected results should improve computer-human interaction in robotic systems, notably during tele-operations or in virtual reality systems.

ECOVIA is planned for 2 years and began on October 2003. It is a collaboration between 5 partners : 2 INRIA projects (i3D and SIAMES), CEA LIST (French Commission for Atomic Energy), Collège de France (LPPA), and University of Paris 5 (LCD).

This research is part of a complex project for the simulation and implementation of fulfilling Virtual Environments (VE) and for the application of sensory integration in a robotic context. In this framework, it is of great interest to study the perception of integrated visual and haptic information.

The comprehension and modeling of multimodal - and more specifically - visuo-haptic integration have been debated for ages. Ernst and Banks have recently proposed a statistical model for the visuo-haptic integration based on the maximum-likelihood estimator (MLE) of the environmental properties. They proposed a model in which each sensory channel contributes to the perception in an inversely proportional fashion to its variance (noise). Within ECOVIA, we first plan to test this model and second to study and if possible to model other aspects of visuo-haptic integration.

7.6. ROBEA Bayesian Models for Motion Generation

Keywords: *Bayesian Model*, *Behavior Modeling and Learning*.

Participant: Stéphane Donikian.

The ROBEA (CNRS Interdisciplinary Research Program) Project entitled "Bayesian Models for Motion Generation" is a partnership with the Cybermove and Evasion Research Projects of the Gravir Lab in Grenoble. The aim of this program is to study how bayesian models can be used to teach an autonomous agent its behaviors, instead of specifying all the probability distributions by hand. It requires to be able to measure at each instant sensory and motor variables of the controled character. The first year has been mainly devoted to the integration of the bayesian programming and interactive natural sceneries modules developed by our partners inside OpenMASK. In the second year, we have developed the urban application that will be used to study the learning by example of a pedestrian navigating in a virtual city. In the third year, we will study how bayesian programming can be used inside HPTS++ to learn behaviors by example and we will make some navigation experiments in our reality center.

7.7. RNTL SALOME2

Keywords: *Simulation Platform*, *Virtual Reality*.

Participants: Georges Dumont, Jean-Marie Souffez, Bruno Arnaldi.

Salome2, RNTL project : The Siames project is involved in this RNTL project with twenty other partners (Open Cascade, Esi Software, Cedrat, Principia R&D, Mensi, Goset, Eads CCR, Renault, Bureau Veritas, CEA/Den, EDF R&D, CEA/List, Cstb, Brgm, Ifp, L3S, Inpg/Leg, Armines, Lan Paris6, Lma Pau). The SALOME 2 project aims to improve the distribution of digital simulation software developed in France which are considered references in their application domains. It provides them with a generic, user-friendly, and efficient user interface which facilitates cost reduction and reduces the delays in performing studies. It is a means of facilitating the linkage of code, the reuse of portions of code, the inter-operability between simulation code and the CAD modeling software. Our interest in this project is the coupling of the advanced Virtual Reality researches with the CAD environment.

8. Other Grants and Activities

8.1. National Actions

- ODL OpenMASK: OPENMASK (http://www.openmask.org) (Open Modular Animation and Simulation Kit) is an experimental VR software platform (see section 5.2) always under development. In order to promote the Open Source deployment of this solution, we pay attention to the manpower dedicated to the evolution of the software platform. Granted by INRIA (under the INRIA ODL programm), a software developper is in charge of the future evolutions of this software.
- CNRS-ATIP project : Locomotion of extinct hominids. This collaboration implies four laboratories :
  - The laboratory IRISA UMR 6074, Siames project
  - The laboratory of Physiology and Biomechanics of Muscular Exercise (Rennes 2 university)
  - The laboratory of Dynamic of human evolution, UPR 2174
  - The laboratory of anthropology of populations of the past, UMR 5809

The interest of this project lies in the involved pluridisciplinarity. We want to understand the bipedal locomotion and to be able to model and simulate this locomotion for extinct hominids. The skeleton is in a pretty good condition. Its study allowed us to understand the functional surfaces involved in the gait motion. A three dimensional model of this skeleton is now available. The study of the articulations lead us to propose a model of the articulated chain, including links and joints. We have developed a motion retargeting method based on morphological and postural parameters. This allows to develop the bipedal gait comprehension and to propose to the paleoanthropologists a tool for testing locomotion hypothesis.

- We are involved in a national project, called V2NET, funded by the french ministeries of of research and teecommunications. The different partners are: THOMSON multimedia, LABRI (research computer science laboratory depending on the university of Bordeaux 1, ENSERB and CNRS), the RDC GROUP (computer science company), ARCHIVIDEO (company specialized in 3D modeling), the laboratory " Hyperlanguages et Multimédia dialogs " of Francetelecom R&D and finally IRISA. The objective of this project is to develop and integrate new tools for an efficient navigation in 3D real or virtual environments in the context of client-server applications.

8.2. European Actions

NoE: Intuition

We are member of the core group of INTUITION : VIRTUAL REALITY AND VIRTUAL ENVIRONMENTS APPLICATIONS FOR FUTURE WORKSPACES which is a Network of Excellence involving more than 68 european partners form 15 different countries. This project belongs to the joint call IST-NMP of the FP6 program.

INTUITION's major objective is to bring together leading experts and key actors across all major areas of VR understanding, development, testing and application in Europe, including industrial representatives and key research institutes and universities in order to overcome fragmentation and promote VE establishment within product and process design. To perform this, a number of activities will be carried out in order to establish a common view of VR technology current status, open issues and future trends.

Thus the INTUITION Network aims are:

- Systematically acquire and cluster knowledge on VR concepts, methodologies and guidelines, to provide a thorough picture of the state of the art and provide a reference point for future project development;
- Perform an initial review of existing and emerging VR systems and VE applications, and establish a framework of relevant problems and limitations to be overcome;
- Identify user requirements and wishes and also new promising application fields for VR technologies.

To overcome fragmentation caused by high technology costs and stand-alone research effort and to structure the relevant research area, INTUITION Consortium will:

- Integrate VR resources, equipment, tools and platforms and work towards their networking and common use;
- Create a Joint Program of Research on VR/VE technologies and their application to Industry and Society;
- Design and perform training courses both for researchers but also for key personnel on the use and development of VR technologies, application and tools;
- Structure and integrate the research work on VR technologies by creating virtual complementary teams all over Europe avoiding overlaps and internal competition;

- Work on interoperable solutions and suggest the ways of their possible implementations in a variety of industrial and societal applications;
- Identify best practices for knowledge management in the relative scientific and research field.

Last but not least INTUITION activities are planned towards a longer-term perspective. It aims at establishing VR in the product and process design and serving as a roadmap for future evolutions and research activities in this area. This will be served by:

- Designing new initiatives for future research projects;
- Working towards the penetration of VR systems in the Industry, including SMEs;
- Widely disseminating VR/VE to the Industry, research community and to the general public both within EU and internationally;
- Supporting standardization;
- Working towards the creation of a permanent, self sustaining network integrating all relevant national networks like PERF-RV, MIMOS, CONSTRUCT IT and other national research efforts;
- Building synergies with partners coming from non-European countries to assign a worldwide sense to its efforts;
- Working towards establishing a pan-European Association, to live on after the end of the project, thus establishing and safeguarding a lasting integration and structuring effect within this field.

INTUITION's consortium members are: ICCS, Alenia, CEA, CERTH, COAT-Basel, CRF, INRIA Rhône-Alpes, FhG-IAO, UNOT, USAL, VTEC, VTT, ART, ALA, ARMINES, BARCO, CSR SRL, CLARTE, CNRS (5 laboratories), CS, Dassault Aviation, EADS, ENIB, EPFL, EDAG, EDF, ESA, ICS-FORTH, FHW, FTRD, FhG-IPK, FhG-FIT, LABEIN, HUT, ICIDO, INRS, IDG, MPITuebingen, UTBv, NNC, ONDIM, OVIDIUS, PERCRO, PUE, RTT, SNCF, SpaceApps, ETH, TUM, TECNATOM, TILS, TVP - S.A., TGS Europe, TNO, UPM, UCY, UNIGE, UMA, UniPatras, UoS, Twente, IR-UVEG, UoW, Plzen.

IP: Enthrone

We are involved in an IP european project called Enthrone bringing together several industrial and academic european partners.

The ENTHRONE project proposes to develop an integrated management solution which covers the whole audio-visual services distribution chain, including content generation, distribution across networks and the user terminals reception. The aim is not to unify or impose a strategy on each individual entity of the chain, but to harmonize their functionality, in order to support an end-to-end QoS architecture over heterogeneous networks, applied to a variety of audio-visual services, which are delivered at various user terminals. To meet its objectives, the project will rely on an efficient, distributed and open management architecture for the end-to-end delivery chain. The availability and access to resources will be clearly identified, described and controled all the way along the content distribution chain. The MPEG-21 data model will be used to provide the common support for implementing and managing the resources functionalities. ENTHRONE aims to define open interfaces based on MPEG-21 standard and XML scripts, which will enable the interoperation of systems and equipment, developed by different manufacturers, and will help developers to adapt their products to be compatible with the end-to-end QoS concept. The project's approach is not based on a centralized model but on a distributed model, where diversified QoS policy based management functions are performed in many geographically distributed environments (content generation, networks, terminals).

9. Dissemination

9.1. Scientific Community Animation

- S. Donikian has been member this year of the Program Committee of the following international conferences: Eurographics'04, ACM AAMAS'04, ACM/Eurographics SCA'04, CASA'04, VRIC'04, AFRIGRAPH'04.
- Member of the core group of the RNTL Salome2 project, to develop relations between Computer Aided Design and Virtual Reality. G. Dumont.
- Participation to the thematic pluri-disciplinary network micro-robotics, RTP number 44 of CNRS STIC department G. Dumont
- Participation to the specific action AS151 (RTP 7) of CNRS STIC : "Virtual Human: towards a very realistic real time synthetic human". G Dumont
- Member of the national comity of RTP 15 of CNRS : "Interfaces and Virtual Reality" : Anatole Lécuyer
- Member of the national core group of AS 131 (RTP 15) of CNRS : "Haptic Interface and Haptic Information" : Anatole Lécuyer
- Member of the CCSTIC (National committee for STIC) : B. Arnaldi.
- Member of “Comité d'Orientation du RNTL”: B. Arnaldi.
- Active member of AFIG (French Association for Computer Graphics - treasurer of the association: S. Donikian.
- Member of “conseil scientifique du GDR ALP”: Stéphane Donikian.
- Co-animator of the french working group on Animation and Simulation : S. Donikian.
- Member of “comité scientifique du Programme Interdisciplinaire de Recherche du CNRS, ROBEA” (Robotique et Entités Artificielles): S. Donikian.
- Member of the national core group of the thematic pluri-disciplinary network Virtual Reality and Computer Graphics, RTP number 7 of CNRS STIC department:B. Arnaldi.
- Member of the national core group of the thematic pluri-disciplinary network Virtual Reality and Interfaces, RTP number 15 of CNRS STIC department: A. Lécuyer.
- Member of the national core group of AS 30 (RTP 7) of CNRS: "Virtual Reality and Cognition": B. Arnaldi, S. Donikian and T. Duval
- Member of the national core group of AS 44 (RTP 7) of CNRS: "Simulation and Visualisation of Natural Phenomenon": K. Bouatouch
- Co-chair of the specific action AS 151 (RTP 7) of CNRS: "Virtual Human: towards a very realistic real time synthetic human": S. Donikian
- Member of the national core group of the thematic pluri-disciplinary network Methods and Tools for Computer-Human Interaction, RTP number 16 of CNRS STIC department: T. Duval.
- Leader of the Working Group on "Haptic Interaction", within the INTUITION European Network of Excellence. A. Lécuyer.

9.2. Courses in Universities

- DEA RESIN ENS Cachan, University of Paris 6 : Virtual Prototyping (G. Dumont).
- MASTER IT-IHM-RV (Master of Technological Innovation Computer-Human Interface and Virtual Reality, University of Maine, Laval, France) : Physical models for virtual reality (G. Dumont).
- Mechanical Agregation course: mechanical science, plasticity, finite element method... ENS Cachan (G. Dumont).
- MASTER IT-IHM-RV (Master of Technological Innovation Computer-Human Interface and Virtual Reality, University of Maine, Laval, France) : Haptic Perception and Computer Haptics (A. Lécuyer).
- Chairman of *Image* Computer Science DEA (K. Bouatouch).
- Computer Science DEA Ifsic: Ray Tracing and Volumic Visualization (K. Bouatouch and B. Arnaldi).
- Computer Science DEA Ifsic: Real-Time Motion (B. Arnaldi).
- Computer Science DESS MITIC (IFSIC), Virtual Reality and Interaction (T. Duval and S. Donikian).
- DEA Calais, Computer Animation (Stéphane Donikian).
- DIIC LSI & INC, DESS ISA Ifsic : Man-Machine Interfaces and Design of Interactive Applications (T. Duval).
- Computer Science DEA : Spatial Reasoning (S. Donikian).
- DESS CCI Ifsic : Computer Graphics(T. Duval).

10. Bibliography

Major publications by the team in recent years

[1] B. ARNALDI. *Animation de systèmes physiques*, mémoire d'habilitation à diriger les recherches, université de Rennes I, février 1994.

[2] B. ARNALDI, R. COZOT, G. DUMONT. *A unified Model for Physically based Animation and Simulation*, in "Applied Modelization, Simulation and Optimization, Cancun, Mexique", juin 1995.

[3] B. ARNALDI, T. PRIOL, L. RENAMBOT, X. PUEYO. *Visibility Masks for Solving Complex Radiosity Computations on Multiprocessors*, in "Parallel Computing (Elsevier)", vol. 23, n° 7, July 1997, p. 887-897.

[4] D. BADOUEL, K. BOUATOUCH, T. PRIOL. *Distributed data and control for ray tracing in parallel*, in "IEEE Computer Graphics and Applications", vol. 14, n° 4, juillet 1994, p. 69-77.

[5] K. BOUATOUCH, P. GUITTON, B. PÉROCHE, F. SILLION. *Simulation de la lumière en synthèse d'images : aspects algorithmiques*, in "TSI", vol. 14, n° 10, novembre 1996.

[6] S. DONIKIAN. *Les modèles comportementaux pour la génération du mouvement d'objets dans une scène*, in "Revue Internationale de CFAO et d'Infographie, numéro spécial AFIG-GROPLAN", vol. 9, n° 6, 1994.

[7] G. HÉGRON, B. ARNALDI, C. LECERF. *Computer Animation*, chap. Dynamic Simulation and Animation, Prentice Hall, juillet 1995.

[8] J. NOUGARET, B. ARNALDI. *Spécification cinématique et simulation dynamique: quelle combinaison optimale pour l'animation?*, in "Numéro Spécial "Journées d'informatique graphique", Revue Internationale de CFAO et d'Informatique Graphique", vol. 10, n° 4, octobre 1995.

Doctoral dissertations and Habilitation theses

[9] S. DONIKIAN. *Modélisation, contrôle et animation d'agents virtuels autonomes évoluant dans des environnements structurés et informés*, Ph. D. Thesis, HDR Thesis, University of Rennes I, Aug 2004.

[10] F. LAMARCHE. *Humanoïdes virtuels, réaction et cognition : une architecture pour leur autonomie*, Ph. D. Thesis, University of Rennes I, dec 2003.

Articles in referred journals and book chapters

[11] C. ARROUET, M. CONGEDO, J. MARVIE, F. LAMARCHE, A. LÉCUYER, B. ARNALDI. *Open-ViBE: a 3D Platform for Real-Time Neuroscience*, in "Journal of Neurotherapy (accepted for publication)", august 2004.

[12] B. BIDEAU, L. FRADET, F. MULTON, S. MÉNARDAIS, R. KULPA, B. ARNALDI. *Validation d'un processus de traitement allant de la capture du mouvement à l'immersion de sujets en réalité virtuelle : application au tir au handball*, in "Revue Internationale de CFAO", 2004.

[13] B. BIDEAU, F. MULTON, R. KULPA, L. FRADET, B. ARNALDI, P. DELAMARCHE. *Virtual reality, a new tool to investigate anticipation skills: application to the goalkeeper and handball thrower duel*, in "Neuroscience letters", vol. 372(1-2), 2004, p. 119-122.

[14] G. DUMONT, C. KÜHL. *A Dynamical Training and Design Simulator for Active Catheters*, in "International Journal of Advanced Robotic Systems", vol. 1, n° 4, december 2004.

[15] C. KÜHL, G. DUMONT. *Coloscopy Simulation: Towards Eendoscopes Improvement*, in "International Journal of Computer Methods in Biomechanics and biomechanical Engineering, Special issue on "Computer-aided medicine and surgery"", to appear 2004.

[16] F. LAMARCHE, S. DONIKIAN. *Crowds of Virtual Humans : a New Approach for Real Time Navigation in Complex and Structured Environments*, in "Computer Graphics Forum (Proc. of Eurographics 2004)", vol. 3, n° 23, 2004.

[17] T. DUVAL, C. LE TENIER. *Interactions 3D coopératives sur des objets techniques avec OpenMASK*, in "Mécaniques et Industries", vol. 5, n° 2, 2004, p. 129–137.

Publications in Conferences and Workshops

[18] M. BADAWI, S. DONIKIAN. *Autonomous Agents Interacting With Their Virtual Environment Through Synoptic Objects*, in "CASA 2004", 2004.

[19] B. BIDEAU, F. MULTON, R. KULPA, L. FRADET, B. ARNALDI. *Virtual reality applied to sports: do handball goalkeepers react realistically to simulated synthetic opponents?*, in "ACM, SIGGRAPH VRCAI (to be published in journal), Singapor", ACM Press, june 2004, p. 210-216.

[20] F. CRISON, A. LÉCUYER, A. SAVARY, D. M. DHUART, J. BURKHARDT, J. DAUTIN. *The Use of Haptic and Pseudo-Haptic Feedback for the Technical Training of Milling*, in "Proceedings of the Eurohaptics Conference, Munich, Germany", 2004.

[21] S. DONIKIAN, J. PORTUGAL. *Writing Interactive Fiction Scenarii with DraMachina*, in "TIDSE'04, 2nd International Conference on Technologies for Interactive Digital Storytelling and Entertainment", LNCS, Springer-Verlag, June 2004.

[22] G. DUMONT, C. KÜHL. *Mixed Beam Model to Calculate the Behaviour of Shape Memory Alloy Spring Actuators*, in "Proceedings of 5th International Conference On Integrated Design And Manufacturing In Mechanical Engineering, Bath, G.B.", 5-7 april 2004, AIP-Primeca, April 2004.

[23] F. DEGRIGNY, T. DUVAL. *Utilisation du modèle PAC-Amodeus pour une réutilisation optimale de code dans le développement de plusieurs versions d'un logiciel commercial*, in "Proceedings of IHM'2004, Namur, Belgique", 2004, p. 149–156.

[24] G. B. G. NICOLAS. *Inverse kinematics for the calculation of plausible bipedal locomotion using anthropological knowledge*, in "Computer Animation and Social Agents, Geneva", july 2004, p. 103-110.

[25] P. GAUTRON, J. KŘIVÁNEK, S. N. PATTANAIK, K. BOUATOUCH. *A Novel Hemispherical Basis for Accurate and Efficient Rendering*, in "Rendering Techniques 2004, Eurographics Symposium on Rendering", June 2004, p. 321–330.

[26] C. LARBOULETTE, M.-P. CANI, B. ARNALDI. *Ajout de détails à une animation de personnage existante : cas des plis dynamiques*, in "Journées Francophones d'Informatique Graphique (AFIG'2004), Poitiers, France", AFIG / EG-France, nov 2004, http://www-evasion.imag.fr/Publications/2004/LCA04.

[27] C. LARBOULETTE, M.-P. CANI. *Real-Time Dynamic Wrinkles*, in "Computer Graphics International", Greece, IEEE Computer Society Press, 2004, http://www-evasion.imag.fr/Publications/2004/LC04.

[28] A. LÉCUYER, J. BURKHARDT, L. ETIENNE. *Feeling Bumps and Holes without a Haptic Interface: the Perception of Pseudo-Haptic Textures*, in "Proceedings of ACM International Conference on Computer-Human Interaction, Vienna, Austria", 2004.

[29] A. LÉCUYER, M. VIDAL, O. JOLY, C. MEGARD, A. BERTHOZ. *Can Haptic Feedback Improve the Perception of Self-Motion in Virtual Reality?*, in "Proceedings of the 12th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Chicago, US", 2004.

[30] J. E. MARVIE, K. BOUATOUCH. *A Vrml97-X3D Extension for Massive Scenery Management in Virtual Worlds*, in "Web3D Symposium", ACM SIGGRAPH, April 2004.

[31] S. MÉNARDAIS, R. KULPA, F. MULTON. *Synchronization of interactively adapted motions*, in "ACM, SIGGRAPH/EUROGRAPHICS Symposium of Computer Animation, Grenoble, France", ACM Press, august

2004, p. 325-335.

[32] S. MÉNARDAIS, F. MULTON, R. KULPA, B. ARNALDI. *Motion blending for real-time animation while accounting for the environment*, in "IEEE Computer Graphics International, Crete, Greece", june 2004, p. 156-159.

[33] R. NAMANE, F. BOUMGHAR, K. BOUATOUCH, S. MIGUET.. *QSplat Compression*, in "AFRIGRAPH'2004", 2004.

[34] N. PRONOST, G. NICOLAS, G. DUMONT, F. MULTON, G. BERILLON. *Simulation d'une locomotion bipède par adaptation de mouvement et cinématique inverse*, in "11èmes Journées du GTAS", 17-18 June 2004.

[35] N. PRONOST. *Interpolation morphologique et posturale pour la simulation de mouvement d'un humanoïde virtuel*, in "11èmes Journées des utilisateurs Vicon", 19 March 2004.

[36] N. PRONOST. *Modélisation et calcul de mouvement pour l'humanoïde virtuel - Une application paléoanthropologique*, in "20èmes Congrès Industriel de Belfort", 23-24 March 2004.

[37] D. M. DHUART, G. MICHEL, J. BURKHARDT, A. LÉCUYER, F. CRISON, J. DAUTIN. *An Application to Training in the Field of Metal Machining as a Result of Research-Industry Collaboration*, in "Proceedings of the Virtual Reality International Conference (VRIC), Laval, France", 2004.

[38] D. NIEL THALMANN, N. MAGNENAT-THALMANN, S. DONIKIAN. *Half day tutorial : Automatic generation of Animated Population in Virtual Environments*, in "Eurographics 2004", September 2004.

Internal Reports

[39] J. KŘIVÁNEK, P. GAUTRON, S. PATTANAIK, K. BOUATOUCH. *Radiance Caching for Efficient Global Illumination Computation*, Technical report, n° 1623, IRISA, Rennes, France, June 2004.

Miscellaneous

[40] B. ARNALDI, F. DEVILLERS, N. MOLLET, J. TISSEAU, E. CAZEAUX, E. MAFFRE. *Patent: Méthode de modélisation graphique et comportementale tridimentionnelle, réactive et interactive d'au moins deux objets*, decembre 2003, Number : FR0314480.

[41] B. ARNALDI, F. DEVILLERS, N. MOLLET, J. TISSEAU, E. CAZEAUX, E. MAFFRE. *Patent: Méthode de scénarisation de session de formation*, juin 2004, Number : FR0406229.

[42] B. ARNALDI, F. DEVILLERS, N. MOLLET, J. TISSEAU, E. CAZEAUX, E. MAFFRE. *Patent: Méthode pédagogique détectant l'intention de l'apprenant*, juin 2004, Number : FR0406228.

[43] B. ARNALDI, F. DEVILLERS, N. MOLLET, J. TISSEAU, E. CAZEAUX, E. MAFFRE. *Patent: Système de formation à l'exploitation, à l'utilisation ou à la maintenance d'un cadre de travail*, juin 2004, Number : FR0406231.

[44] A. LÉCUYER, L. ETIENNE, B. ARNALDI. *Patent: Modulation de position de curseur dans les données vidéo pour écran d'ordinateur*, november 2003, Number : 0311302.