Collaborative Development of an Open Framework for Medical Simulation

S.M. Cotin[1], P.F. Neumann[1], X. Wu[1], S. Fonteneau[2], P.J. Bensoussan[2], J. Dequidt[2], D. Marchal[2], L. Grisoni[2], S. Karpf[2]

[1]The Sim Group, CIMIT/Harvard University, Massachusetts, USA
Cotin.Stephane, Neumann.Paul, Wu.Xunlei@mgh.harvard.edu
[2]Alcove Team, INRIA, Lille, France
Sylvere.Fonteneau, Pierre-jean.Bensoussan,Jeremie.Dequidt, Damien.Marchal, Laurent.Grisoni, Sylvain.Karpf@lifl.fr

Abstract. This paper investigates an Open Framework for Medical Simulation (named SOFA for *Simulation Open Framework Architecture*), and proposes ideas for structuring and supporting its expansion. Lately, through a joint effort between the Sim Group at CIMIT and the Alcove group at INRIA, we have investigated the foundations for a more extensive, flexible, and comprehensive framework. In addition, five internationally renowned research groups have contributed to this initiative by converting their algorithms into compatible modules. Although still in its very early developmental phase, the current project illustrates some of the key concepts we believe will enable collaboration and interoperability in Medical Simulation. The main objective of SOFA research is to foster collaboration among research groups. It is our hope that SOFA will simplify the development cycles, reduce production costs, and provide a means to share components through a common interface.

1 Introduction

Increasing computational power, as well as current achievements in the field of interactive computer graphics and virtual reality, has already led to the rapid development of more or less sophisticated medical simulation systems during the past ten years. These systems offer an appealing way to provide adequate training without any risks of direct patient involvement. However, in spite of the impressive development on the field, some fundamental problems still hinder the field's development:

- For a successful simulator all of these components have to be built with reasonable quality making system development expensive and slow in research as well as industrial environment. This also leads in many cases to the selection of a sub-optimal solution, compromising the quality of the resulting training system.
- Isolated research and development has led to unjustified duplication of effort which is an enormous waste of intellectual and economical resources. The coordination of methodological and software development would not only help to substantially reduce the development cost, but also concentrate resources for rapidly advancing in novel areas instead of duplicating previous efforts.
- A large variety of anatomical digital models has been created independently. Unfortunately, in contrary to the coordinating effort of the National Library of

Medicine in the preparation and maintenance of raw cryosection images and radiological data sets, no serious effort has been made up to now to collect, organize and distribute the resulting derived models and the coupled auxiliary dataset.

2. Methodology

2.1. Mapping and multi-representation core

Although deformation, collision detection (CD), collision response (CR), visual feedback and haptic feedback may not be aspects of every simulator, they are the current foci of the medical simulation research. Therefore, we believe that they should be an integral part of a generic framework. Other aspects – like physiology – will be considered later. The approach we propose is illustrated in the left of Figure 1. It includes a *Behavior Model* that can describe either a deformable structure, or a rigid object, or an articulated model, thus representing a soft organ, a bony structure, a thread, or a medical device. Each model is composed of different views of the same entity, with their corresponding algorithms and internal representations. To abstract and isolate each representation from another, a Mapping mechanism is introduced.

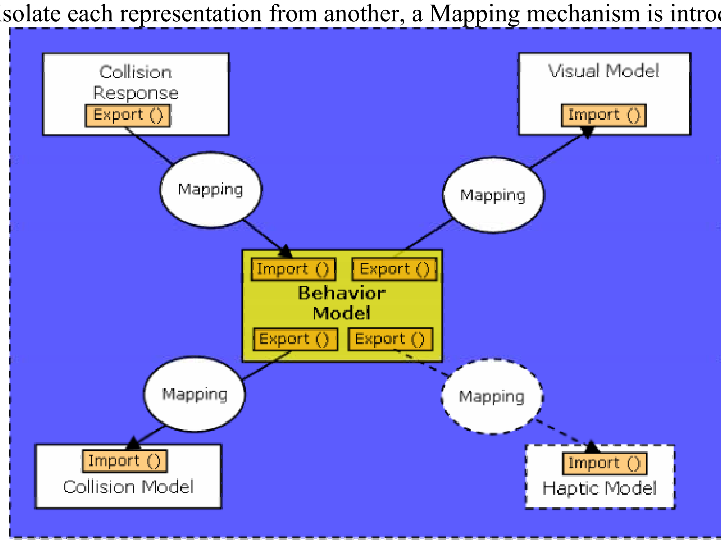

Fig. 1. Multi-representation and mapping concepts.

The diagram in Figure 1 illustrates two essential concepts of the framework: *mapping* and *multi-representation*. Each representation is optimized for a particular "task" – e.g. collision detection, visualization – while at the same time improving interoperability by creating a clear separation between each functional aspect of a "model". Examples of representations include, but are not limited to, a Behavior Model, a Collision Model, a Visual Model, a CR Model, and a Haptic Model. The Behavior Model is a generalization of a Deformable Model since it can represent any structure that can be interacted with in the simulation: instrument, rigid structure, and of course soft tissue model. The next step is to define a mechanism to "map" or “translate” each representation into another. This, of course, requires each module to “export” some information about its internal representation. Initially done through a set of methods like Export() and Import(), as described below, this is now implemented as a programming interface (API) between two modules. The role of the mapping mechanism is not very different from a software driver that fits between a hardware component and an application.

To reduce the overhead and to avoid penalizing computation speed, it might not be necessary to compute the mapping at every “time step”. Update() method is implemented to update one representation according to the changes in the other one(s). The choice of the *update rate*, as well as the model from which the deformation will be derived, is left to the developer/end user, but the SOFA core will provide the key building blocks allowing for a large number of combinations, depending on the requirements of the application.

2.2. Collision detection and scheduling design

CD and scheduling require a different approach due to their global aspect. CD cannot be performed at the level of an object/model or it would be highly inefficient (if N objects, it’s a N^2 problem). The solution we propose is to embed CD in the Environment, whose role includes rendering, synchronization, and etc. Each Behavior Model defines a Collision Model which is a particular representation of the geometry. At each “time step”, all the models (Visual, Contact…) are updated. The Environment, during the CD phase, processes all Contact Model representations. The potential contacts and penetration information is sent back to each model through another mapping -- Response Model. Its role is to separate the specific requirements for a Behavior Model to respond to a contact from the choice of the CD.

The problem of scheduling (i.e. dynamic allocation of CPU cycles to each component and sub-component of the system) is also a complex aspect of SOFA. Similarly to collision detection, it involves a global level which role is to optimize the use of the CPU(s) depending on specific requirements for a given module. The scheduler will work in concert with the collision manager, since both of them operate at the highest level of the simulation.

3. Proof of concept

Two meetings were held in 2004 with representatives from five international research groups in Medical Simulation community. A proof of concept has been created to demonstrate the feasibility of such collaborative project, which entailed:

1. Define and implement a set of generic concepts for the framework core:
    a. Multi-representation of a simulated object (independent modeling and computation of different representations of the same object, such as visual model, behavior model, collision model, haptic model, …)
    b. Implementation of mapping interfaces, i.e. specific APIs that allow each representation (visual, behavior, collision…) to “communicate” with another representation of the same object.

c. Implementation of several real-time deformation algorithms including a Finite Element Model, a Spring-Mass model, and a rigid body model.
d. Implementation of a real-time collision detection algorithm compatible with a common collision model for each deformation method
e. Implementation of a simple collision response scheme able to determine contact points and forces when objects collide

2. Integrate basic concepts in a demonstration prototype.

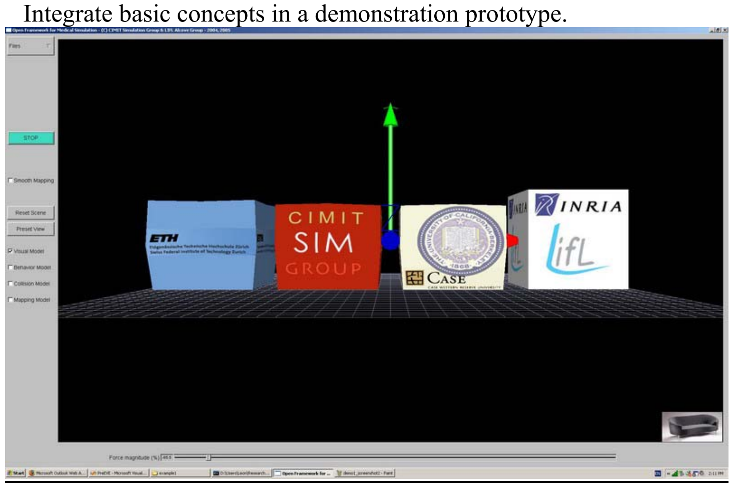

Fig. 2. Four behavior models, i.e. (from the left) linear FEM, Mass-Spring1, Mass-Spring2 , Rigid, mapped to the same geometry. Different textures distinguish modules contributed by corresponding institutions. Same force is loaded on the top face.

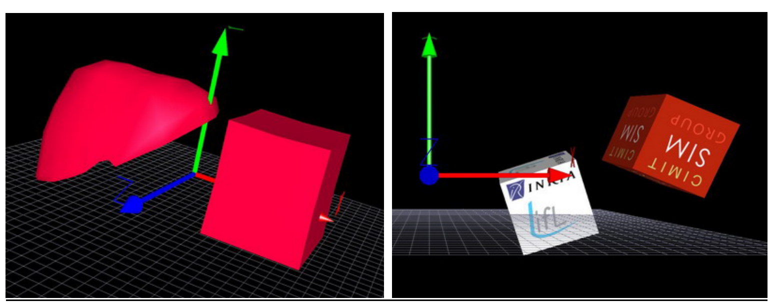

Fig. 3. *Left:* Another illustration of the mapping concept: the same behavior is applied to different geometries. Even with a large discrepancy between the internal deformable model representation (a cube) and the external visual model (a liver), reasonable behavior can be observed. *Right:* Two objects with different behavior (rigid for the white cube, soft for the red one) collided. A collision model, mapped onto the visual model but update according to the internal behavior model, was used to compute the contact points and reaction forces.

Each team involved has contributed by submitting source code or a library implementing a deformation method, or by implementing parts of the core framework. The basis of the proof-of-concept was to have a simple geometric shape (a cube) controlled by different behavior methods. Each algorithm was provided by a different group, while the framework was developed by the Sim Group and Alcove team.

- Behavior Models: linear FEM, Mass-Spring, and Rigid Body; Visual Models: triangulation in OBJ format; Collision Model: bounding spheres; Response Model: penalty (forces)
- Requirements / Specs: Input Device: Graphical User Interface + mouse; Geometry: cube (8x8x8 cm^3) with different mesh densities; Scene: 2 cubes sliding on a rigid and fixed planar surface

From Figure 2 and 3, this proof-of-concept showed that it is entirely feasible to set up collaborative international effort to establish all the key concepts of SOFA, while maintaining the collaborative nature of the research and development. From the preliminary results we have identified several areas of research that need to be investigated further.

4. Conclusion

We have demonstrated the feasibility of a generic software framework for medical simulation by developing an initial prototype with several deformation algorithms from multiple research laboratories. The long-term benefits of a generic framework will decrease research and development time, and eventually production costs. It also will help promote collaborations, accelerate prototyping, and provide upgradeability.

Currently, we are preparing for an alpha release in January, 2006 at the Medical Meets Virtual Reality (MMVR) conference. We are still looking for collaborators to contribute modules on haptics, collision detection, collision response, and more deformation algorithms. Please find more technical details and videos from the website, http://www.sofa-framework.org/.

References

1. S. Cotin, D. Shaffer, D. Meglan, M. Ottensmeyer, S. Berry, S. Dawson - CAML: A general framework for the development of medical simulations. In Pien, H, ed. Proceedings of SPIE Vol. 4037: Battlefield Biomedical Technologies II, 2000.
2. M. Chabanas and E. Promayon - Physical Model Language: towards a Unified Representation for Continuous and Discrete Models, International Symposium on Medical Simulation, Cambridge, 2004.
3. P. Meseure, J. Davanne, L. Hilde, J. Lenoir, L. France, F. Triquet, C. Chaillou. A Physically-Based Virtual Environment dedicated to Surgical Simulation, International Symposium on Surgery Simulation and Soft Tissue Modeling, p. 38-47, 2003.
4. T. Goktekin, M. Cenk Cavusoglu, F. Tendick - GiPSi: An Open Source/Open Architecture Software Development Framework for Surgical Simulation, International Symposium on Medical Simulation, Cambridge, 2004.

5. Dawson SL, Cotin S, Meglan D, Shaffer DW, Ferrell MA. Designing a computer-based simulator for interventional cardiology training, (with editorial comment). Catheterization and Cardiovascular Interventions, 2000; 51:522-528.