Making Sense of Virtual Environments: Action Representation, Grounding and Common Sense

Jean-Luc Lugrin and Marc Cavazza
School of Computing, University of Teesside
Middlesbrough, TS1 3BA United Kingdom
{j-l.lugrin, m.o.cavazza}@tees.ac.uk

ABSTRACT
The development of complex interactive 3D systems raises the need for representations supporting more abstract descriptions of world objects, their behaviour and the world dynamics. The inclusion of Artificial Intelligence representations and their use within 3D graphic worlds face both fundamental and technical issues due to the difference in representational logic between computer graphics and knowledge-based systems. We present a framework for such an integration illustrated by a first prototype.

ACM Classification: H5.1 [Multimedia Information Systems] Artificial, Augmented and Virtual Reality -

General terms: Theory, Algorithms, Design and Experimentation

Keywords: Knowledge Representation, Qualitative Reasoning, Intelligent Virtual Environments.

INTRODUCTION

With the development of Virtual Reality (VR) and Interactive 3D (I3D) environments, there has been a growing interest in describing virtual world structure and dynamics at a more abstract level [5] [19] [18]. A number of researchers have proposed to integrate Artificial Intelligence representations "on top" of virtual worlds to facilitate a conceptual description of scenes and their evolution, thus introducing the concept of Intelligent Virtual Environments [2]. Typical applications include: world creations from ontological descriptions [18] or from Natural Language descriptions [8], multimodal interaction [20], and behaviour simulation and interpretation [6] [26].

Physical simulations of I3D systems still lack the ability to perform accurate real-time simulations [15]. They rely on various types of physical event discretisation based on the graphic primitives for interaction, essentially contact and collision events. This discretisation, upon which most 3D platforms are still based, is actually inefficient to solve many problems of common sense physics. The reason is that the implementation of such effects is entirely procedural and not based on first principles, nor even on generic concepts or re-usable categories. However, this discretisation of behaviour which takes place in most 3D graphics engines also supports the integration of symbolic reasoning for behavioural simulation [7], as well as providing a general principle for the integration of knowledge-based systems in VR.

In this paper, we introduce a framework for a consistent integration of semantic representations in VR supporting the interleaving of simulation and interpretation. Our main objective is to articulate object and action representations into the cycle of transformations affecting the virtual world, and to investigate the specific representational problems faced when relating the virtual world dynamics to knowledge structures.

PREVIOUS AND RELATED WORK

There has been a growing interest in high-level representations of virtual world simulations in recent years, coming from various perspectives, from virtual world design to the implementation of intelligent agents.

Badler's group [4] pioneered the introduction of explicit action representations in virtual environments, initially to support the execution of action variants under the influence of natural language instructions. This was the first time in VR that actions where conceptualised in some form of ontology, which was termed an "actionary". Kallmann and Thalmann [17] introduced the notion of smart objects in virtual simulations to associate typical behaviours following real-time interactions for instance with virtual agents. Their approach has been a first step towards the introduction of a more generic and abstract behaviour representation associated to the objects (as opposed to all-out scripting), as well as a preoccupation with functional aspects, although not recurring to AI techniques strictly speaking.

IUI'07, January 28–31, 2007, Honolulu, Hawaii, USA.

Several research groups have recently explored semantic representations for virtual environments: Muller-Tomfelde et al. [28] used knowledge structures to facilitate the exploration of virtual environments. Latoschik et al. [20] have developed symbolic representations for virtual environments, initially as part of multimodal interfaces to VR. Kleinermann et al. [19] have introduced semantic representations for virtual environments to facilitate the design of virtual worlds. Kalogerakis et al. [18] proposed the use of ontologies to structure the contents of virtual worlds, using OWL graphs to represent 3D objects and scenes.

From a different perspective, Lieberman et al. [25] have advocated the introduction of Common Sense representations in interactive systems. While Minsky [27] made a strong case for this, they have given several practical examples of the role of CS in supporting interactive systems [24], including an Augmented Reality kitchen [21] using a subset of Open Mind [30].

In the field of AI-based behavioural simulation, Erignac [9] was the first author to introduce the use of Qualitative Simulation in virtual environments to simulate the behaviour of complex devices with which virtual humans had to interact. In previous work [7], we have explored the use of Qualitative Process Theory [12] to support Qualitative Simulation in virtual environments, mostly for reasoning with liquids and thermal exchanges. More recently, Zhou and Ting [36] have also adopted Qualitative Physics for object behaviour in tactical simulation.

REPRESENTATIONAL ISSUES

In essence, the introduction of knowledge layers in virtual environment constitutive of Intelligent Virtual Environments [2] can be characterised by the conflict between two representational logics. One corresponds to the graphical data structures representing the virtual objects, and the other to the knowledge representation for the same objects as well as the actions occurring in the virtual world.

Graphical representations have evolved out of concerns for efficient display, rendering and the control of physical interaction (mostly collision between objects and between objects and volumes, from which most graphical events can be defined). The graphical data structures for a typical object (a glass) are represented on Figure 1. This representation comprises graphical data structures such as meshes (the core object which is displayed with a position and an orientation) and textures (elements of visual appearance that can also be dynamically modified to figure changes in state) grouped into "objects". The relationships between various objects or between components of a complex object are actually graphic primitives such as `base`, which will determine the joint motion of an object in contact with the base and the base itself. A significant proportion of the representation is dedicated to physical parameters which are also used by the physical simulation system to compute object dynamics. The elements of visualisation are derived from meshes and textures both for static images and real-time animations. It should be noted that such graphical representations are not structured and do not support spatial or part-whole inferences, which have to be procedurally encoded in an object instance's behaviour.

Functional Representations and the integration between Objects and Actions Representations

Our working hypothesis is that a successful representation should be able to articulate object representations with the actions and processes they are most likely to take part in, while preserving the possibility of generic inference.

The description of object properties from the perspective of their use constitutes a specific research topic known as Functional Reasoning [11]. Bicici and Saint-Amant [3] have provided a valuable classification of the various approaches to functional reasoning by analysing the role assigned to shape, causality and physics in various approaches. Of particular interest is the fact that, while discussing Winston et al.'s work [35], they emphasised how shape information can be enriched with semantic properties, for instance the fact that a cup should also be described as "lift-able". This is important in defining relevant levels of granularity for the object descriptions. Vaina and Jaulent [31] have introduced a representation scheme to support functional recognition, which they have referred to as a "compatibility model". They consider that the functional categorisation of objects should make use of criteria which are specific to actions. Finally, Van Leeuwen et al. [32] have investigated from a psychological perspective the relations between affordances and the perception of tool function. All of this previous research has encouraged us to adopt a representation inspired from the object function, in which for instance physical states could be interpreted in functional terms. This is also in line with much of the contents of the "ontology for liquids" proposed by Hayes [13], in which much of the basic properties of liquid containers have been described originally.

This symbolic representation (Figure 1), taking the form of a small semantic network, attempts at relating structure to function, on the basis of actions likely to affect the object. On the structural side, it describes the part-whole structure of the object and the elementary physical properties of its components. They are not all shown on the Figure 1 as they can be derived from the property of substances of which the object parts are made, etc. Structural properties such as part-whole relations support common sense inferences through the semantic graph, which can be triggered by the manipulation of object parts, for instance the fact that if the bottom of the glass is tilted, all parts supported by the base will also be tilted (including its `internal_volume` and `opening`). On the functional side, it attaches functional states to the object, more specifically object parts such as its `internal_volume`, which takes states such as `FULL` or `EMPTY` depending on the quantity of substance it contains. The specification of the type of content is also part of the functional description (OpenFluidContainer), although its does not preclude the possibility for the glass to contain other objects, such as small solids. `Filling` and `Emptying` processes are not part of the object representation, but can be triggered by the existence of connecting paths between the object's `internal volume` and the `outside environment`. These paths are established from the object structure and the physical states of each of its constituent parts. The container's `opening` for instance enables only inwards path when `VERTICAL` but enables outwards paths when `TILTED` (this is a functional version of the notion "right way up" in Hayes' ontology of liquids [14]). An important aspect of this representation is its connection to the graphical object of which it constitutes a semantic description. Elements in bold whose label bears an asterisk are actually grounded in the 3D world, as they correspond to graphical objects taking part in physical interactions. Certain parts are linked to "virtual" (in the strongest sense of the word) objects, which correspond to certain functional parts, such as the opening of a container; these have an existence but are not strictly speaking physical objects. We have chosen to represent them explicitly in the virtual world to be able to capture certain physical interactions (although these are "invisible" objects) For instance, `Opening#1` (Figure 1) is implemented in the graphical world as an invisible object attached to the top of the glass, whose position and collision information will promptly detect if any object has landed on itself, and is so blocking the glass's opening. In our semantic representation, such an action will be interpreted by removing its `Connected-To` link to the `outside`. They are the points of entry into the semantic representation, those likely to be affected by transformations of the virtual world, and updated by the post-conditions of actions and processes. On the other hand, properties attached to those nodes can be propagated within the object representation through common sense inferences.

Object representations also serve as a basis for scene representations. Several objects can be connected through semantic relations expressing their spatial arrangement enabling inference over the aggregate semantic network.

SYSTEM OVERVIEW AND ARCHITECTURE

The overall system architecture is designed to integrate knowledge representation and reasoning with real-time interaction and simulation. The former maintains a semantic interpretation of the world, while the latter updates the reference world (object creation, modification and destruction) and visualises such dynamic modifications. From an implementation perspective, it

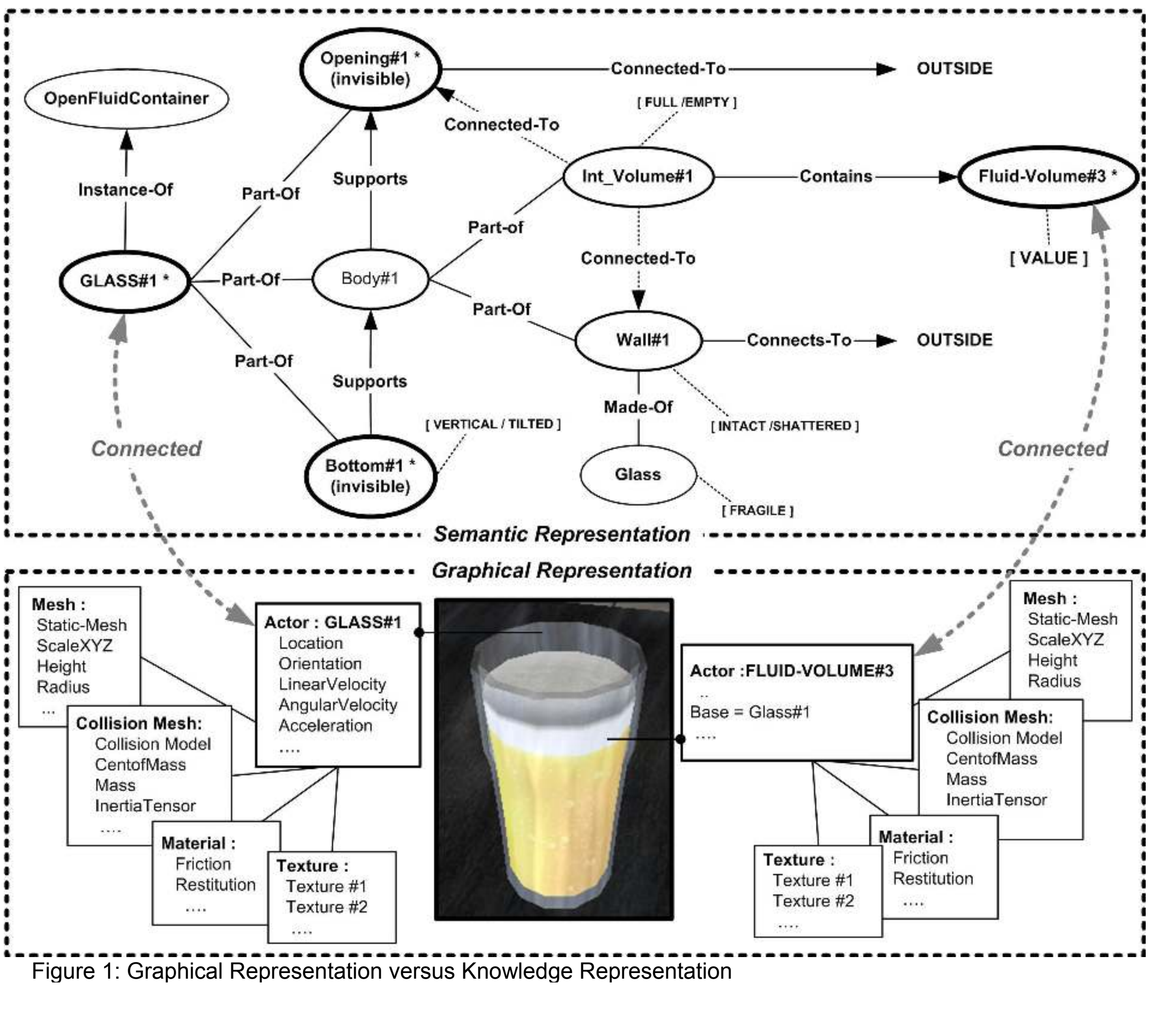

Figure 1: Graphical Representation versus Knowledge Representation

involves developing external modules on top of a commercial game engine (Unreal Tournament™) [23]. The only caveat is to avoid the idiosyncrasies deriving from the implementation of a given graphic environment: however the notion of event systems on which our approach relies is common in VR [16] and has also been used in previous work with a custom built 3D graphics system [6].

The architecture re-uses the event-based approach we have developed previously to integrate AI-based behaviour in 3D worlds for qualitative and causal simulation [7] [26]. The system architecture is represented on Figure 2. The UT engine supports world visualisation and physical interaction with the world objects. Embedded in the UT engine is the Karma™ Physics engine which supports real-time simulation for all simple object dynamics and plays a particular role in the grounding of certain representations in the virtual world.

On top of the UT engine, we have developed an Event Interception System (henceforth EIS), which is in charge of all discretisation and constitutes the main interface between the VR world and the semantic layer. It contains: i) an event interpretation module, analysing low-level events produced by the engine (for instance the `bump(obj1, obj2))` for the collision between graphical objects) to trigger the recognition of actions and processes ii) an effect visualisation module, which executes the procedural consequences of high-level actions and processes "physically" triggering their effects in the virtual world.

All software components of the top (semantic) layer are developed outside the engine as C++ modules communicating through sockets via the UT socket interface. Central to the semantic layer is the scene representation, which is composed of a set of object representations formalised as semantic networks. These objects representations and their relations will be updated as the world is modified and will also serve to carry out common sense inference about the situation. A separate module for common sense knowledge contains an inference engine that will update the object representations each time part of them has been altered by the post-conditions of an action or process. This updating will be based both on semantic propagation within the object representation and on the use of standalone common sense rules under declarative form. The behavioural engine is in charge of semantic action recognition, i.e. the instantiation of action

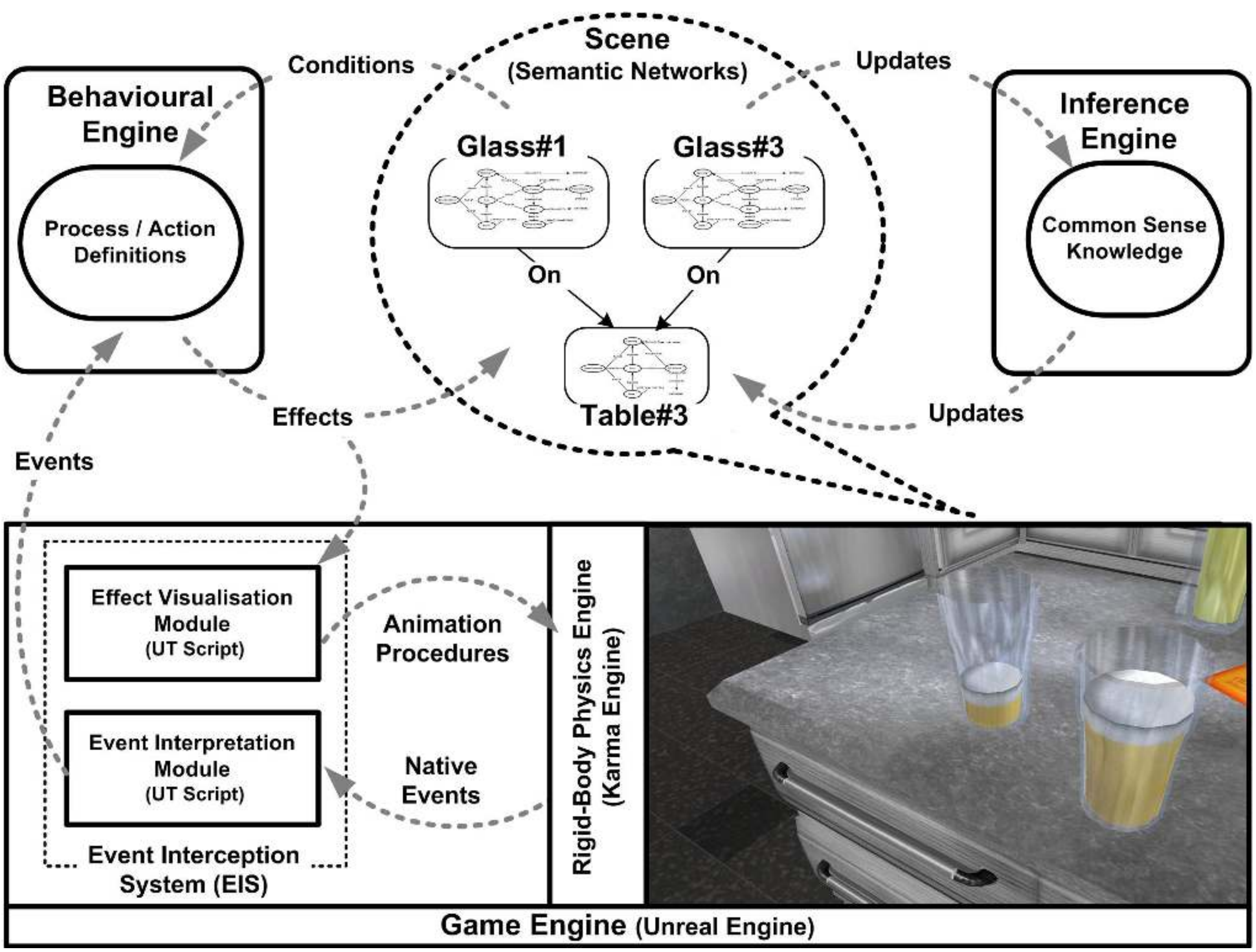

Figure 2: System Architecture for the Integration of a Knowledge Layer in an Interactive 3D Environment

representations from low-level events and the interpretation of object changes in conceptual terms. It also co-ordinates the physical transformations of graphical objects with the modification of their semantic representations.

ACTION REPRESENTATION, SIMULATION AND GROUNDING

In I3D systems, there is a need to make sense of the environment and recognise actions as they occur, and also to apply specific actions to the world objects (which can for instance be planned by an autonomous agent). The relation between symbolic representations and the objects and events taking place in the virtual world can be seen as a particular case of grounding [34] [39]. Ideally, grounding is a circular process, coupling real-time physical simulation to the updating of symbolic representations and executing in the virtual world the physical transformation corresponding to the actions post-conditions.

Action recognition and execution are not strictly symmetrical processes in terms of representation. Let us consider an action whose effects can be carried out entirely under the control of a Physics engine. An example of such action would be the `TILTING` of a glass (Figure 5). In the virtual world, a `TILTING` action can be triggered by a high-momentum impact on specific locations of the standing glass. The triggering events as well as the following motion of tilting can be entirely determined by the Physics engine. The subsequent physical simulation will provide all required state changes in the grounding world as well as display them in real-time. In the resulting state, the glass will be laying horizontal and still on the table surface.

The recognition of this type of grounded actions (i.e. actions that can take place on physical grounds without being enacted by an agent) can be achieved through the use of state-based action recognition described initially by Andre [1] and implemented in VR by Cavazza and Palmer [6]. This method of action recognition often relies on an underspecified representation of the action itself, where only certain instantaneous events are represented. The sequence of events is defined to be characteristic of the specific action to be recognised: parsing such a sequence of events enables to instantiate all the relevant elements of an action representation. For instance, to detect that an object has landed on a surface, it is sufficient to recognise its impact on the surface followed by a standing position on that surface (in practice this can be achieved by the recognition of successive impacts of decreasing momentum.

For reasons of computational complexity which are easily understood, the virtual world cannot rely on an accurate and comprehensive simulation of all physical phenomena. Physical simulation is actually limited to solid dynamics, but does not include accurate simulation for continuum mechanics, fluid dynamics or thermodynamics.

For those actions that require some level of discretisation because they cannot be entirely simulated at the physical level, the relationship between action application and recognition is somehow different. One such action is `BREAKING`, occurring for instance when a glass collides with a hard object (whether the glass itself or the object is moving). Its representation associates *causes*, in the form of pre-conditions and triggers and *effects*, in the form of procedures and post-conditions (Figure 3). The detection of the action actually corresponds to a semantic interpretation of the low-level physical events, in which causes and effects are associated within a semantic representation. In the instantiated action representation of Figure 3, the trigger is an impact event involving the glass, represented by a dynamic, time-stamped, predicate `%Impact(Glass#1,Table#1 t0)`. Once our system has generated this impact event, the action recognition system verifies the validity of its pre-conditions, namely that the impact took place on a hard object (accessing physical properties of the object from its representation, such as `Hard(Table#1)`. Upon successful instantiation of the action representation, its effects can be activated in the virtual world (with the corresponding updating of the knowledge representation). In our example, the post-condition `~Shattered(Glass#1)` will both trigger an animation and update the glass's semantic representation, by propagating the `shattered` state to its Wall part (see Figure 1). In our formalism, such predicates that also activate an animation visualising objects alterations, are prefixed with a tilde (~). Here the activation of the `~Shattered` predicate triggers a sequence of animations that generates the glass fragments and particles.

To summarise, we can say that fully grounded physical actions have different representations for execution and recognition, while discretised actions use a unique semantic representation which detects the physical conditions triggering the action (e.g. an impact in the case of a Breaking action) and executes its effects on the virtual objects.

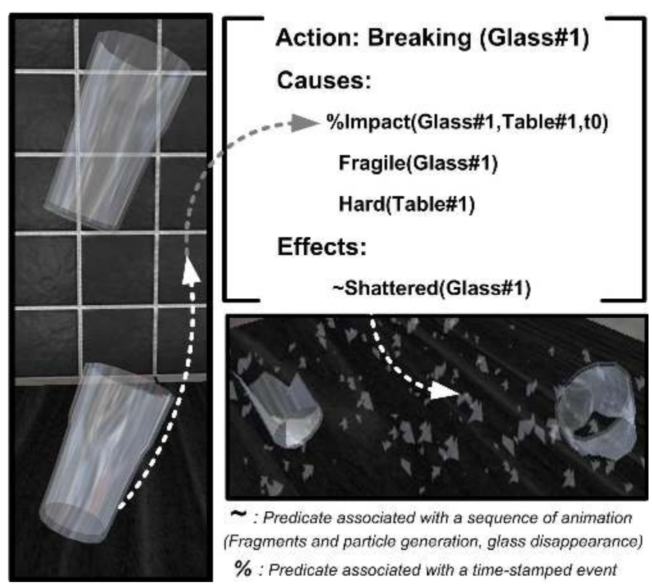

Figure 3: The Action Representation Formalism

COMMON SENSE REASONING

Much of the semantics of VR is about scene structure, spatial relationships, possible actions and their consequences, and making sense of the world's dynamic evolution. A proper interpretation of structure and transformations often requires common sense reasoning. There have been many different approaches to common sense reasoning in AI since Pat Hayes' Naïve Physics Manifesto [13]. The current day situation can be summarised into two main approaches. One is the very large knowledge base approach, which has now evolved into ontological work, such as Cyc [22] and OpenMind [30]. The other could be termed a "deep knowledge" approach and is epitomised by Qualitative Reasoning [33] [12]. One major issue has always been the level of granularity and modularity of such knowledge units. Common sense knowledge can actually be embedded in action or process representations (for instance as pre-conditions) or exist as independent knowledge (often presented as rules). While elements of commonsense are included in our action and process representations, we have chosen to restrict common sense inference *sui generis* to "rules" of generic nature that could be applied throughout object representation, possibly across object categories. Such rules would include the transitivity of inclusion along part-whole relations, the fact that an object is not free if another object is on top of it, that a glass object is "breakable" depending on its size and shape (a window or glass wall but not a marble), the fact that a wet object may not be flammable, etc. In other words "primitive" common sense is best kept separate under declarative form. In a further step, we could try to connect our functional representations to an external large-scale common sense ontology from which to automate the acquisition of such knowledge (as well as testing the compatibility of its granularity with our representation).

EXAMPLE RESULTS

The knowledge layer can be used to support common sense inferences in conjunction with qualitative simulation. In the configuration of Figure 4, a cardboard menu sits on top of the glass. Let us place a bottle cap on top of the menu. This cap is actually supported by the menu. The scene representation has been updated to reflect this (the actions

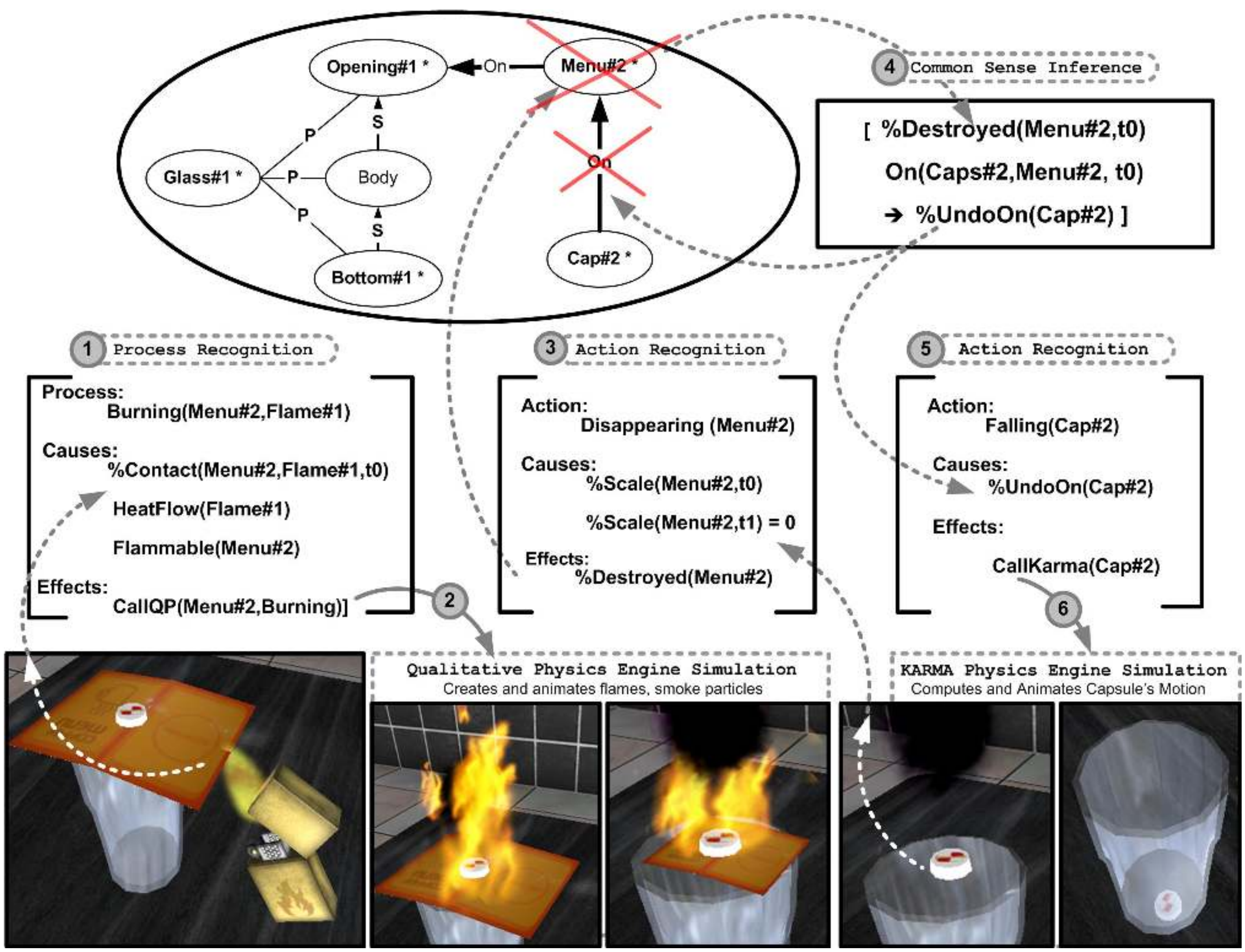

Figure 4: Integrating Physical Simulation, Common Sense Reasoning and Qualitative Simulation (see text for explanations)

detected when the cap was put on the menu are not represented on the Figure). Let us set the cardboard menu alight by approaching a flame: this will trigger a `Burning` QP. The combustion process starts when a flammable object gets in contact with a Heat-Flow (e.g. a naked flame) generated by a Heat-Source (a lighter) itself manipulated interactively by a user (Figure 4-1). In essence, the combustion process, simulated by our QP Engine (Figure 4-2), decreases the `amount-of-substance` in the burning object until it disappears (when the corresponding qualitative variable reaches zero). The real-time visualisation of this process is the regular reduction of the object *scale* (a graphical parameter modifiable in real-time affecting an object's size), while simultaneously generating and controlling two particle emitters, one simulating the flame and one the combustion smoke.

The process termination is associated with a specific event confirming the disappearance of the burning cardboard piece: `%Destroyed(Menu#2)` triggered by the recognition of a `DISAPPEARING` action, as shown in Figure 4-3 (The `DISAPPEARING` action is a generic action recognising the consequences of processes altering an object's existence, such as melting, burning, evaporating, which is kept external to the process simulation itself). In our formalism, predicates prefixed with % correspond to dynamic predicates (these are used for recognition of changing states, in particular object destruction or removal; they are not unlike a suppression demon in data-driven programming). The disappearance of the menu in the object representation leads to the update that the relation `on (menu#2, caps#2)` no longer holds.

This inference is obtained through the common sense rule operating on the object representation (Figure 4-4), according to which if a supporting object is destroyed the object standing on top of it is no longer supported. The latter is represented through a `%UndoOn(cap#2)` predicate. This predicate can in turn activate the recognition of a `FALLING` action (Figure 4-5), being part of its pre-conditions. The effects of `FALLING` include calling the Karma™ physics engine to simulate the actual fall of the

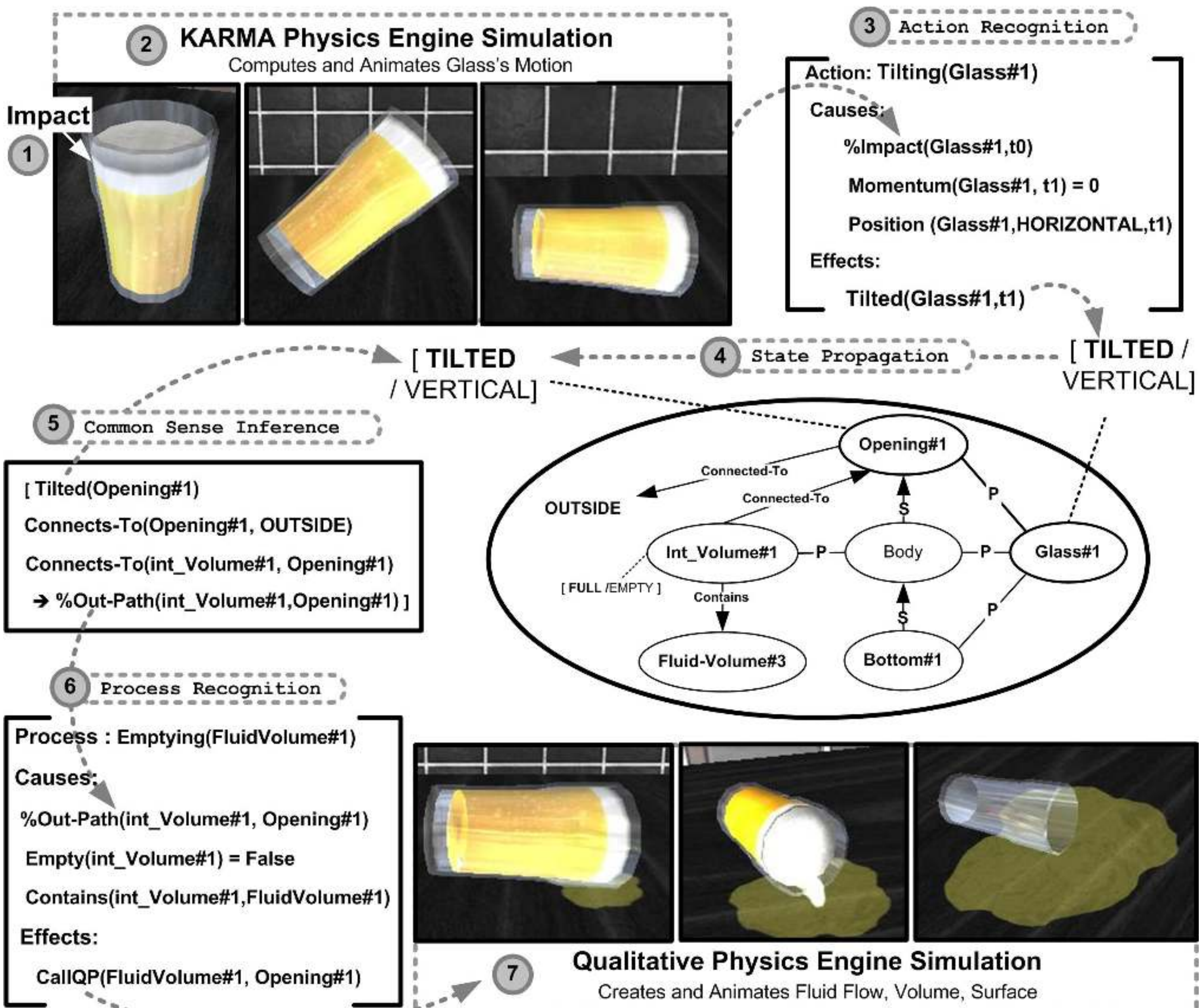

Figure 5: Qualitative Simulation and Common Sense Reasoning (see text for explanations)

cap (Figure 4-6) (using reference resolution to pass the internal identifier of the reference graphic object cap#2). In turn, the Physics engine will simulate the fall into the glass during which more low-level events are generated upon collision with the glass's mesh. These events, named Kimpact(…), can be parsed into a LANDING action, which is defined through the recognition of a couple of impacts, followed by a still position. A further common sense inference, not represented in the figure, yields that the cap is eventually *inside* the glass, as it faced its opening and its size is smaller than it. This example illustrates again the interleaving between the various aspects of the simulation (Physical, Qualitative) and the inferences derived from the scene representation. One limitation here is that, unlike in our previous implementation, we are using slightly simplified QP, which are simply executed as durative actions (i.e. cannot be dynamically modified once they have been started).

Another example, the tumbling glass (Figure 5), illustrated the integration between various levels of simulation (physical, qualitative) mediated by the semantic representation, as well as the overall system dynamics. This example illustrates the integration of physical interaction, action recognition, common sense inference and qualitative simulation. It shows how a simulation which is currently largely scripted in most I3D systems can be based on first principles, which will also support appropriate high-level descriptions of the events.

As described in Figure 5-1, a glass containing a certain amount of liquid will tilt when being hit by another moving object (irrespective of the origin of that other object's motion). The glass's movement is directly managed by the Physics engine, which will calculate the glass' trajectory, while displaying a real-time animation of its motion (Figure 5-2). It can be recognised as a tilting action through the action recognition procedure, which instantiates the formal definition of a TILTING action from the low-level physical events captured by the EIS (translated into the dynamic predicate %Impact(Glass#1, t0)). As a result of a TILTING action recognition, the glass representation is finally updated according to the action's post-conditions (Figure 5-3). The final position of the glass is passed to the object representations from which it can be inferred that 1) when the glass is tilted, its opening is tilted as well (state propagation through part-whole relationships in Figure 5-4) 2) Because TILTED corresponds to a functional position of the Glass#1 as an OpenFluidContainer, this establishes a new outwards path between the glass' internal_volume and its opening. As a consequence, the dynamic predicate %out-path is instantiated (Figure 5-5). This type of inference is a direct consequence of our functional approach to object representation.

Our representation gives a rationale to this, based on the notion of "paths" which determine how fluid flows can be established. The possibility of an out-path activates an Emptying QP [7], whose pre-condition detects such paths (Figure 5-6). The simulation of that QP creates a flow of liquid (Figure 5-7); during the simulation the amount of liquid in the glass is decreased (this being updated in the object representation until the glass' internal volume reaches the EMPTY value, feeding back the simulation results into the object representation).

CONCLUSIONS

We have presented a first prototype exploring the implementation of Intelligent Virtual Environments in some depth, trying to address all relevant aspects within a single consistent framework. In this prototype, we have integrated work from several areas of Artificial Intelligence supporting Common Sense reasoning (mostly Qualitative Reasoning and Knowledge Representation), and have proposed an architecture for their real-time integration into VR.

The emphasis has been on deep representations rather than large-scale ontologies and the prototype under development remains of moderate complexity with a total of 25 object categories and 40 actions and processes. In that sense, if metrics were to be used to measure its scale or complexity, these should rather be inspired from Faulkenheimer and Forbus [12] who reported that their few dozens of qualitative processes were formally equivalent to thousands of Horn clauses. This is still modest if compared with large Common Sense knowledge bases, yet allowed us to explore problems of deep representation in the context of dynamic, interactive systems. It is probably too early to assess the real-time performance of the system considering its scale. However recent results obtained with parts of this architecture have suggested that the use of AI techniques was still compatible with the response times of interactive systems [26].

The real-time integration of knowledge representations in VR opens multiple applications in virtual world design, interpretation and user interaction. At the same time it constitutes a good experimental setting in which to address some traditional AI problems such as grounding, with the potential of supporting more fundamental work as well.

ACKNOWLEDGMENTS

Sean Crooks developed the virtual environments including additional animations, used by our current prototype.

REFERENCES

1. André, E., Herzog, G and Rist, T. On the Simultaneous Interpretation of Real World Image Sequences and their Natural Language Description: The System SOCCER. *Proceedings of the 8th European Conference on Artificial Intelligence*, Munich, 1988, pp. 449-454.

2. Aylett, R. and Luck, M. Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments, *Applied Artificial Intelligence*, 14(1), 2000, pp. 3-23.

3. Bicici, E. and St. Amant, R. *Reasoning about the functionality of tools and physical artifacts*. Technical Report TR-2003-22, Department of Computer Science, North Carolina State University, April, 2003.

4. Bindiganavale, R. Schuler, W. Allbeck, J. Badler, N., Joshi, A. and Palmer, M. Dynamically Altering Agent Behaviors Using Natural Language Instructions. *Autonomous Agents 2000*, ACM Press, pp. 293-300.

5. Campos, J., Hornsby, K., and Egenhofer, M. A Model for Exploring Virtual Reality Environments. *Journal of Visual Languages and Computing 14 (5)*, 2003, pp. 469-492.

6. Cavazza, M. and Palmer, I. High-Level Interpretation in Virtual Environments. *Applied Artificial Intelligence*, *14:1*, 1999, pp. 125-144.

7. Cavazza, M., Hartley, S., Lugrin, J.L., and Le Bras, M. Qualitative Physics in Virtual Environments. *Proceedings of the 9th international conference on Intelligent User Interfaces,* Funchal, Madeira, Portugal, ACM Press, 2004, pp. 54 – 61.

8. Clay S.R. and Wilhelms, J. Language-Based Interactive Manipulation of Objects, *IEEE Computer Graphics and Applications,* Vol. 16, Number 2, March 1996, pp. 31-39.

9. Erignac, C., *Interactive Semi-Qualitative Simulation in Virtual Environments*. PhD Thesis, University of Pennsylvania, 2001.

10. Falkenhainer, B., and Forbus, K. Setting up large scale qualitative models. *In Proceedings of the Seventh National Conference on Artificial Intelligence,* Menlo Park, Calif.: AAAI Press, 1988.

11. Far, B.H. *Functional Reasoning, Explanation and Analysis,* Technical Report JAERI-M 91-225, Japan Atomic Energy Research Institute, Jan. 1992.

12. Forbus, K.D., Qualitative Process Theory, *Artificial Intelligence*, *24*, 1984, pp. 85-168.

13. Hayes, P. Naive Physics I: Ontology for Liquids. In: J. Hobbs and It. Moore (Eds.), *Formal Theories of the Commonsense World.* Norwood NJ: ABLEX Publishing, 1985.

14. Hayes, P. The naive physics manifesto. In D. Mitche, Editor, *Expert Systems in the Micro-Electronic Age*. Edinburg University Press, 1978, pp. 242-270.

15. Hecker, C. Physics in computer games. *Communications of the ACM*, *43(7)*, 2000, pp. 34-39.

16. Jiang, H., Kessler, G.D. and Nonnemaker, J. DEMIS: A Dynamic Event Model for Interactive Systems, *Proceedings of the ACM VRST 2002 Conference*, Hong Kong, China, 2002. pp. 97-104.

17. Kallmann, M. and Thalmann, D. Modeling Behaviors of Interactive Objects for Virtual Reality Applications, *Journal of Visual Languages and Computing*, *13(2)*, 2002, pp. 177-195.

18. Kalogerakis V., Christodoulakis S., Moumoutzis N., 2006. Coupling Ontologies with Graphics Content for knowledge driven visualization, *Proceedings of the IEEE Virtual Reality Conference 2006*.

19. Kleinermann, F., De Troyer, O., Mansouri, H., Romero, R., Pellens, B., Bille, W. Designing Semantic Virtual Reality Applications, *Proceedings of the 2nd INTUITION International Workshop*, Senlis, France, 2005.

20. Latoschik, M.E., Biermann P., and Wachsmuth, I. Knowledge in the Loop: Semantics Representation for Multimodal Simulative Environments. *Proceedings of the 5th International Symposium on Smart Graphics* 2005 Frauenwoerth Cloister, Germany, pp. 25-39.

21. Lee, C.H.J, Bonanni, L., Espinosa, J.H., Lieberman, H. and Selker, T. Augmenting kitchen appliances with a shared context using knowledge about daily events. *Proceedings of the 11th international conference on Intelligent user interfaces*, Sydney, Australia, 2006, pp. 348-350.

22. Lenat, D.B. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM*, *38(11)*, 1995, pp. 33-38.

23. Lewis, M. and Jacobson, J. Games Engines in Scientific Research. *Communications of ACM, Vol. 45*, No. I, 2002. pp 27-31.

24. Lieberman, H. and Espinosa, J. A goal-oriented interface to consumer electronics using planning and commonsense reasoning, *Proceedings of the 11th international conference on Intelligent User Interfaces*, Sydney, Australia, ACM Press, 2006, pp. 226 – 233.

25. Lieberman, H., Liu, H., Singh, P., and Barry, B. Beating Common Sense into Interactive Applications. *AI Magazine 25*(4), 2004, pp. 63-76.

26. Lugrin, J-L. and Cavazza, M. AI-based World Behaviour for Emergent Narratives. *Third ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, Hollywood, California, USA, 2006.

27. Minsky, M. Commonsense-based interfaces. *Communications of the ACM*, *43(8)*, 2000, pp. 66–73.

28. Müller-Tomfelde, C., Paris, C. and Stevenson, D. Interactive Landmarks: Linking Virtual Environments with Knowledge-Based Systems. *In: Proceedings of the OZCHI*, Wollongong, Australia, November 2004.

29. Roy, D. Semiotic Schemas: A Framework for Grounding Language in the Action and Perception. *Artificial Intelligence, 167(1-2)*, 2005, pp. 170-205.

30. Singh, P. and Barry, B. Collecting commonsense experiences. *Proceedings of the Second International Conference on Knowledge Capture (K-CAP 2003)*, Florida, USA, 2003.

31. Vaina, L. and Jaulent, M.-C. Object structure and action requirements: A compatibility model for functional recognition. *International Journal of Intelligent Systems*, *6*, 1991, pp. 313–336.

32. Van Leeuwen, L., Smitsman, A., & Van Leeuwen, C.Affordances, perceptual complexity, and the development of tool use. *Journal of Experimental Psychology: Human Perception and Performance, 20,* 1994. pp. 174-191

33. Weld, D., and De Kleer, J., *Reading in Qualitative Reasoning about Physical Systems*, Morgan Kaufmann, 1990.

34. Winograd, T. *Understanding Natural Language*, Academic Press, New York, 1972.

35. Winston, P.H., Katz, B., Binford, T.O. and Lowry, M.R. Learning Physical Descriptions From Functional Definitions, Examples, and Precedents. *Proceedings of the American Association for Artificial Intelligence Conference* 1983, pp. 433-439.

36. Zhou, S. and Ting, S.P. Qualitative Physics for Movable Objects in MOUT. *Annual Simulation Symposium*, IEEE Press, 2006, pp. 320-3