Visualisation of Qualitative Processes

Simon Hartley, Marc Cavazza, Jean-Luc Lugrin and Mikael Le Bras
({m.o.cavazza, s.hartley, j-l.lugrin, m.lebras}@tees.ac.uk)
School of Computing, University of Teesside
Middlesbrough TS1 3BA United Kingdom

Abstract

In this paper, we describe our approach to the visualisation of qualitative physics, which uses a 3D graphical engine to present the results of a qualitative simulation. The qualitative formalism we use for the simulation is Qualitative Process Theory. The choice of this qualitative reasoning formalism was due to the representation that exists within the formalism, which enables us to add interactions via the process' preconditions. The visualization system we describe is developed using a game engine and takes advantage of its advanced graphical rendering system and its event-based system to provide interaction and a number of 3D graphical elements, which are used as the visualisation primitives. We use a virtual kitchen as a test environment. In this virtual world, we have implemented a library of physical processes for object behaviour and complex device behaviour all of which are visualised in user real-time (where appropriate). After a presentation of the system architecture and its implementation, we discuss example results from the system. This approach has potential applications in virtual laboratories and for virtual prototyping.

Introduction

Qualitative Physics has been especially adequate for the modelling of complex devices and multi-component systems. However, it would be appropriate to offer a user-friendly interface for experimenting with qualitative reasoning, both in terms of visualisation and interaction. In this paper, we argue that a representation of the objects and processes involved through an interactive Virtual Environment constitutes such an appropriate interface.

The advancement of qualitative physics into a virtual environment has two distinct advantages. The first being that a virtual environment can be used as an interface to facilitate the assembly of qualitative simulation experiments from qualitative physics libraries and objects. However, The most significant advantage is its use as a visualisation tool used to observe simulations as they progress. A visualisation of the evolution of a process has applications within qualitative reasoning (especially in development and debugging) and for applications that explore common sense reasoning. To achieve visualisation, a significant level of interaction must exist between the qualitative system and the virtual environment.

This has led to the development of a method for interaction with objects that directly relates them to the qualitative system. These interactions not only allow the user to interact with objects, but the user and the objects to interact with the environment as well.

It is common for virtual worlds to develop at least the basics of physical behaviour, i.e. gravity and kinematics (for collision). In interactive virtual environments, the implementation of these basic physical behaviour is according to the constraints of user interaction in order to maintain acceptable response rates. This constraint has led to the rationale, that in order to maintain this interaction rate, the physical simulation is discretised following interaction events. Kinematical aspects within these systems tend to be simulated through traditional numerical approaches, while more complex mechanical events (objects breaking or exploding) are pre-calculated. The transition between the two approaches is managed by event systems allowing high-level interpretations to be made about actions that occur in the system such as hit, touch and collide. These interactive systems are therefore potentially extensible via their event model, giving an avenue to incorporate an additional physics library into the system, which extends the original without replacing it. These event-based architectures support the integration of QR systems into these environments. The integrated QR will benefit from all interactive features in terms of object manipulation, agency and causality. The integration process will have to devise the visualisation primitives that should be associated to certain events, allowing the representation of both process activity and object state. The reason being that qualitative simulations are naturally discretised: this central property makes them an instinctive choice [Cavazza *et al*, 2002, 2003].

In addition to the user centred interaction, we believe that in order to create and maintain complex virtual environments, a high-level representation is required from the implemented qualitative physics library. This high-level representation is required in a library of object behaviours, not only as the library needs to be reusable between environments, but to support collaborative work. This desire for a high-level representation has led to the use of qualitative process theory as the basis for the qualitative library. In addition, the representation of behaviours as a process (or processes) enables the use of high-level descriptions for the concepts they convey which are easier too communicate.

In the remainder of this paper, we will present the results from our ongoing research into the use of a virtual environment to visualise a library of qualitative processes. We start by describing the system architecture, developed using a game engine, Unreal® Tournament 2003 (UT 2003), and take advantage of its event-based system to integrate qualitative process theory in an interactive fashion. Following this, we present case studies for the visualisation of some of the processes within the system. In particular, we show how, due to the library of processes, we are able to instantiate multiple model fragments in different environments and have the user interact with them. For a demonstration, we present an implementation of the library within a virtual cafe as a test environment. In this virtual world, we have implemented various behaviours: for physical object behaviour and for complex device behaviour (appliances), all simulated in user real-time. We conclude by discussion of the results achieved so far and present our plans for future work.

Discussion

In the system we have implemented, we have utilised a highly interactive environment to visualise a library of qualitative processes, currently consisting of 32 processes. The demonstration environments focused upon the following:

- Visualisation: the visualisation of the processes performed in real-time. Changes in the qualitative variables enable a graphical response. The scope of a process, (the visualisation) is not limited by the current implementation, as the granularity of the qualitative system is separate from the granularity of the graphical representation. By the judicious use of the event system, we may visualise the model in different ways.
- Interaction: in the demonstration environment, we have introduced multiple objects and multiple process instances to enhance interactions. The simple behaviours modelled by the model fragments allow the system to maintain an interaction rate, which responds to the user input. The causality embedded in the processes causes the cognitive interpretation of the interaction by the user and gives the environment a truly interactive feel.
- Physics Library: the library is independent of the virtual environment and scenario, as a model fragment, or series thereof defines each scenario. The independence is due to the common event based system that enables the scenario to activate and deactivate any process either in initialisation or during runtime. We are able to use the qualitative library in several environments
- Real-Time Simulation: models for complex devices, such as a fridge, provide the user unique interaction experiences within the system. The simulation of devices run in real time and the effects produced are interacted with and perceived by the user through the visualisation engine, which reacts in real-time to the changes in the data.

The visualisation of a process has two challenges, the first is how to represent process activity the second is how to represent object states. We believe our event-based system has the advantage of being able to solve both of these challenges and remain more than a system simply with animation triggers. To achieve this we present our mapping of the following qualitative data:

Table 1: Visualisations Primitives.

Qualitative Variable	3D Visual Primitives
Heat	Agitation of Fluid Surface
Combustion Rate	Height of Flame
Combustion Rate	Light Intensity
Amount of Substance	Height of Candle
Type of Substance	Rate of Particles(Smoke)
Generation Rate	Rate of Particles(Steam)
Fluid Flow Rate	Rate of Particles(Splash)
Fluid Flow Rate	Column Diameter
Object State	Texture State
Amount of Substance	Height of Fluid Surface

The visualisation of qualitative physics within interactive virtual environments has a significant potential for applications in several different contexts most notably: virtual environments for virtual prototyping, virtual laboratories, simulation, and training, virtual reality art.

Implementation

Qualitative simulations are built from a detailed study of a system, which is then translated into a qualitative formalism. Hence, the decision upon which formalism should be implemented for the qualitative simulations to successfully translate the models behaviour should be made by a careful analysis of the overall system requirements. The requirements of our system are:

i) The physical simulation maintains a rate that is compatible with an interaction rate in the system.

ii) We should be able to build reusable model fragments that can be combined and made to represent new objects and behaviours.

iii) Causality within the system should be closely associated with user interaction.

Within the qualitative reasoning community the predominant ontologies are *device*, *process* and *constraint*; all of these ontologies support a high-level of representation for physical systems in order to produce a model for the qualitative predictions [Bailey-Kellogg and Zhao, 2001]. Process ontologies model physical behaviours as processes and qualitative process theory formalism expands upon this to generate possible evolutions of processes by detailing the interactions between the process' individuals. If we can apply user input to prime these interactions, we will be able to utilise this formalism. Another factor for consideration is a subtle limitation of device-based ontologies. Namely, these theories do not formalise the critical stage of moving from the description and relationships of objects to an appropriate abstract description for the model. The qualitative process theory formalism provides this vital stage in the modelling, by producing an appropriate abstract description for the model of the objects and relationships [Forbus, 1989].

From these considerations, the choice of qualitative process theory is due to the representational properties in the formalism, which enables us to add interactions via the processes preconditions. If we ensure that the representation of the behaviours by qualitative processes does not consume too many resources and respond in an appropriate time limit; we ensure that the first requirement of the system is met.

Fulfilment of the second requirement occurs if we use the formalism to produce model fragments that are simple encapsulations of the behaviours. Since, qualitative process theory formalism provides a stage for the abstract description for the model within the system; we will be able to achieve this requirement. Having analysed the overall system requirements we have found an easy and effective method for the integration of qualitative process theory into a virtual environment.

The next stage is to envision the different behaviours as processes. To achieve this, each object represented in the system has to embody the concept, both qualitatively and graphically, of the represented behaviours. The qualitative system represents the objects physical properties in terms of quantities, which have a distinctive qualitative space. However, the integration of the qualitative system into a system with an advanced graphical capability directly implies that the original qualitative representation is no longer sufficient for the complete description of the objects behaviour. Changes are required in order to benefit from this new system.

As the visual system has become the primary element for user reasoning of system behaviour, we need a high level of integration between the two systems in order to represent such behaviour. The qualities that are required for the visual component of the system is it must be controllable by the qualitative variables and it must adequately represent (to the user) an interpretation of the expected behaviour. To achieve this goal of giving the user the best possible interpretation of the behaviour within the system the best system design architecture would have to attain a close integration for the representations within the graphical and qualitative systems.

Integration of qualitative physics relies on UT 2003´s native event system that we have extended with the development of specific “QP events” that activate the QP simulations from the interaction with virtual world objects. The UT2003 engine has a set of basic events generated by the graphics engine from collision detection primitives. For example, the QP event *QP_start_heat_flow* is primed by the touch basic event as shown in figure 3.

These basic events are; bump, touch, un-touch, enter volume, exit volume, hit, and landed. All of the basic events relate to an object’s position in the environment. The events come from the collision of or interaction with objects and provide a wealth of data about spatial configurations, which are useful in determining object relations and/or device status and assembly. The UT2003 Engine supports the extension of these basic events via an embedded scripting language, unreal script. This scripting language manages the control of every aspect of the environment from the animation and creation of objects to the camera view of the user. Using unreal script, we have derived a new set of events for the system, which we call Qualitative Physics events (QP events). These events perform an evaluation of whether a process preconditions are satisfied. The generation of QP events uses the basic events and object properties. For example, moving an object to the inside of the oven, generates an enter volume event. If the object is of type QP object, then the heat path between the oven and the object is set to true, which satisfies the preconditions for the heat flow starting the process. The qualitative system responds to the changes in the qualitative variables by generating events, which we call Qualitative Physics effects (QP effects) and are used to trigger corresponding changes in the graphical system. For example, a direct change in volume, level, or indirect changes such as boiling or evaporation upon exceeding a certain threshold landmark or limit point. The generation of QP effects uses the discretisation of the qualitative variables to pass the evolution of a quantity to the virtual environment. The QP effects alter the variables within the system. QP effects are also generated when the processes become active or inactive [Cavazza at al, 2003] and are used to produce a variety of visualisations, such as particle systems for steam or colour changes for concentration. This implementation allows us to generate events for the change in a quantity whenever that quantity changes or whenever its passes a discrete limit point. The Qualitative System achieves this by testing the relevant quantities when applying the process’ influences after the resolution of influences. During this testing, the qualitative system can generate an event each time the quantity changes, by generating an event immediately, or can generate an event only when the landmark is passed by testing the quantity space for the variable. This makes the system flexible in terms of the graphical representations used. We have developed this into an approach, which more closely matches the idea of the visual system not being simply an animation, but a logical extension to the qualitative reasoning for the visualisation of processes in an interactive virtual environment. This is due to the systems, ability to specify how the generation of events occurs, and how the graphical engine responds to the transmission of this data. As in the system, implementation uses a combination of flexibility in representation for the qualitative results and their intertwining of visual primitives. The approach in conventional visualisation begins with the analysis of the quantities to be represented. The concept behind our visualisation approach alters this to analyse how the qualitative system can be used to best describe the behaviour. Thus, we have produced a model of the physical behaviour, which gives a list of available processes for the system. As the behaviour we wish to model within the system is encapsulated by the possible states and in QPT, these are represented as processes. During this process prototyping stage, the preconditions for the processes are generated and user interaction is made a major factor within a processes preconditions. This implies that integration of user interaction, as a critical element within process activation, has been successful. Since, user interaction now creates the situations in which the preconditions can become satisfied (e.g. the alignment of a container and a liquid flow). Therefore, for a process to become active, the user must satisfy its preconditions before it can check any other condition.

A list of all available processes is used by the system to generate a model of all of the processes that can occur within a given virtual world configuration. This means that all the objects, which are represented in the system, can be interacted with and activate or deactivate the processes associated with them (i.e. processes in which they can take part, such as fluid flow for containers).

The first design decision for the visualisation within the system is the level of detail that we wish to provide for the visualisation. The decision to visualise the system at the highest level of abstraction could provide too few visual indicators for the process and yet a decision to visualise at lower levels could provide too many visual indicators. At best, this would confuse the user and at worst, it would slow the system down and interfere with the interaction rate. Since one of the aims of the system is to promote the interactive element, this would be extremely counter productive. These considerations have led to the increased use of multiple small models of the behaviour that we are able to visualise in detail. The concept of user interpretation of visual effects has led to the implementation decision, to give a higher visualisation priority to a process' direct influence than to the indirect influences of process. This allows the graphical system to resolve cases where the simulation presents too many visualisation effects to the user.

The qualitative system relates the changes in these qualitative variables to variables in the graphical system. For instance, the boiling effect contains a qualitative variable used to represent the rate of boiling (generation-rate), which affects the amount of a boiling substance in a container by converting it to a gas. The effects of boiling upon the substance that we wish to visualise are: the disturbance of the liquid, loss of liquid and the effects of the production of the gaseous substance achieved by:

- A vertex fluid surface animation upon the surface of the liquid that shows the process activity by controlling its agitation parameter with the process' generation rate.
- Controlling the lowering of the height of a volume actor to represent the qualitative amount of substance.
- Using a diverser Particle emitter, controlled by the process generation-rate data, to specify the amount of particles produced and the size of the particles to represent the production of the gaseous substance.

The application of these various effects seen in Figure 1.

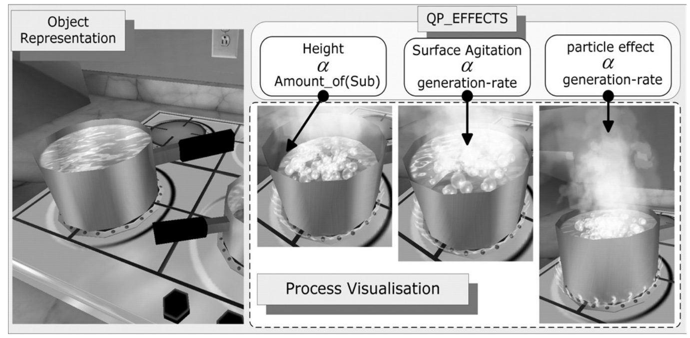

Figure 1: Relating Changes in Qualitative Variables to Visual Effects.

The native Physics engine included in the visualisation system has its own representations and variables, which can be utilised by the qualitative system. In addition to these variables, the graphical system has a range of variables that perform graphical manipulations for the internal physical behaviours. The access to these variables by the event-based system allows the extension of the range of behaviours. For instance, the representation of a pump model in the graphical system uses an oscillating system in which the oscillation rate depends upon the pump rate. So, when the pump is under normal operation it is in its working state. If the pump enters another state (i.e. a losing state) this will be reflected by the change in the oscillation behaviour.

The Process Library and System Architecture

Qualitative systems can be made to capture all significant behaviours and essential object behaviour of the modelled system. To achieve this, Qualitative Process Theory defines a process, which is an approximation for the modelled behaviour. This identification of processes for the behaviours within the system is an essential part of its development. However, we are trying to provide a library of processes that can be utilised within different virtual environments, without having to remodel the system each time. The processes described should explain the general process behaviour and allow for extension through further envisionment using existing processes as a starting point. This is, at present, implemented envisioning combinatorial scenarios with true compositional scenarios as a definite possibility. The initial basis for the implementation of processes was for thermodynamic processes [Collins and Forbus, 1989]. As these fit into description of processes that occur in many environments and have a associated simple model fragment that can be used as a basis for the creation of more detailed models. In an experiment into the potential of qualitative process theory within the system, we have implemented mechanical processes for the system. The initial library of processes implemented for the system is shown in figure 2,which also describes some of the QP objects for the system.

To make an object a QP object involves the assignation of a descriptor to the object that allows it to participate in

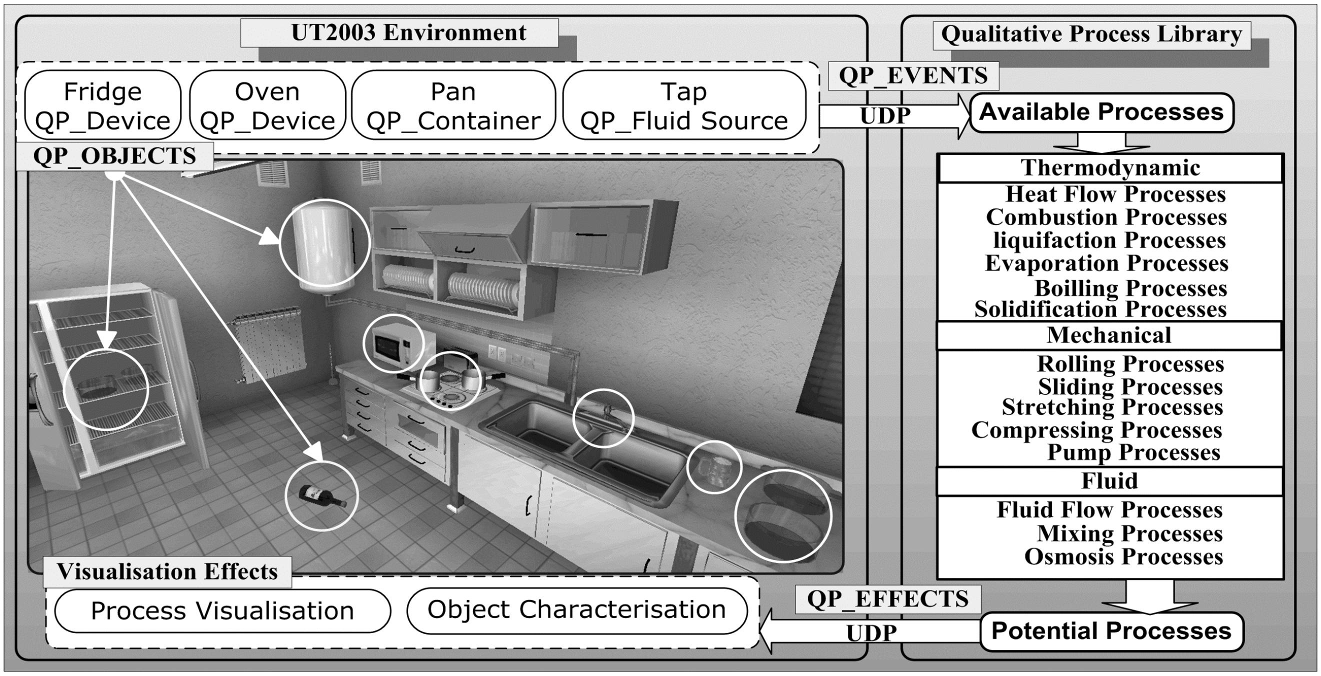

Figure 2: System Overview

various processes via the QP Events and QP Effects. Communication if the events and effects to and from the graphical system are via UDP protocol. For instance, the pan is a QP container and so it has the event QP fluid path aligned allowing us to fill it with water. The qualitative system, upon receiving this event, activates the fluid process and generates the QP Effects that describe the graphical effects (see figure 6).

However, this does not describe the individual view for the objects, which are modelled using qualitative process theory. Enabling a process for an object, is as simple as assigning a specific parameter to the object property list. The initialisation of the system communicates the assigned property lists for all of the objects. The qualitative system uses this property list to generate a list of potential processes by checking the objects for the properties required to take part in a process. For example, to participate in heat flows, an object needs the property heat, or for solid materials, a combustion or melting point allows the participation in the combustion and liquefaction processes respectively. By using this technique, the qualitative system identifies all the potential processes for the environment that by user manipulation and interaction with objects can be potentially active. For example, the heat process has a heat path connected precondition that is activated by the basic events of "touch" or "enter volume". The "touch" event occurs when two QP objects are moved into contact. The "enter volume" basic event occurs when an object is placed inside another. For heat flow putting a QP object in the fridge/oven or putting a pan upon the kitchen hob that generates the QP Event that triggers a heat flow to the pan as in figure 3.

The advantages of this design are the minimization of model restructuring during runtime and the concentration on communicating the object responses to the processes. For instance, an empty container will not participate with the boiling process. However, if we fill the container with water, this gives the object a substance to which the boiling process applies. Eventually, this water would evaporate and the boiling process would stop. The system is simple in its conceptualisation of the object behaviour, in the current implementation, at the expense of structural interaction. We cannot combine model fragments to create different models or fragments during the simulation. For example, we have not conceived a model where we can combine a series of objects and have the combination display a different behaviour than its constituents, like the construction kit suggested in [Erignac, 2000].

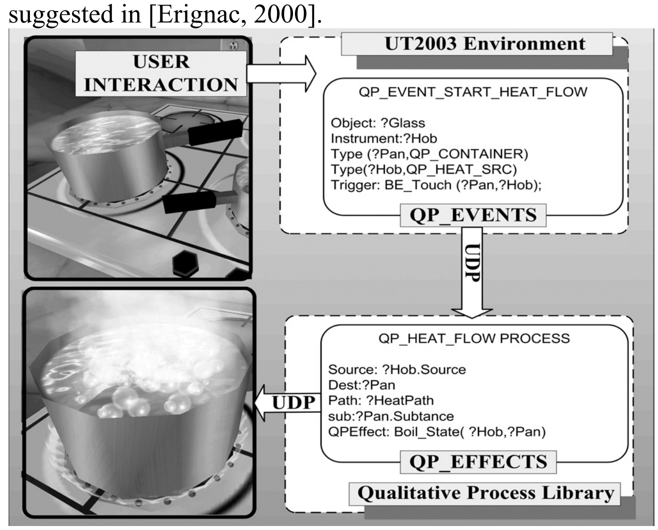

Figure 3: Starting a Heat Flow Process

Results

We have developed a library of processes, including many common thermodynamic, mechanical, and fluid flow processes. (The number of processes in the library currently stands at 32.) These new processes for object interactions and behaviours provide opportunities to develop diverse visualisations for the interactive environments that capitalise upon them for object interactions and behaviours [Forbus, 1984]. Since these are commonplace behaviours and familiar physical laws, we have applied these processes to everyday environments (kitchen, bathroom, and café) and benefited from the clear affordances these associations imply. For instance, turning on a water flow from a tap results from a simple interaction and grabbing an empty container (glass) to place under a liquid flow can start a "filling" process.

The representation of an object's state, by the library of 3D graphical primitives, centers on the use of textures, colours, and particle systems. For instance, colours and textures have uses for the representation of concentration or changes in state; where as particle systems have uses when trying to represent the production of a gas or a splash effect upon a water level. Changing object textures can visually represent object states, which correspond to landmark values reached by underlying qualitative variables. The implementation of this is by the use of dynamic texture systems making associations between the finite set of textures to an object and its state. During a simulation, landmark values of qualitative variables map directly to the corresponding textures by QP effect events. Further, it is possible when necessary to blend textures for a smoother transition in the objects appearance. An example of this representation detailed in our demonstration environment by the change in texture of water to represent the object state changing to frozen. The fridge model uses the Qualitative System defined by using the thermodynamic processes in Collins and Forbus (1989) (figure 4).

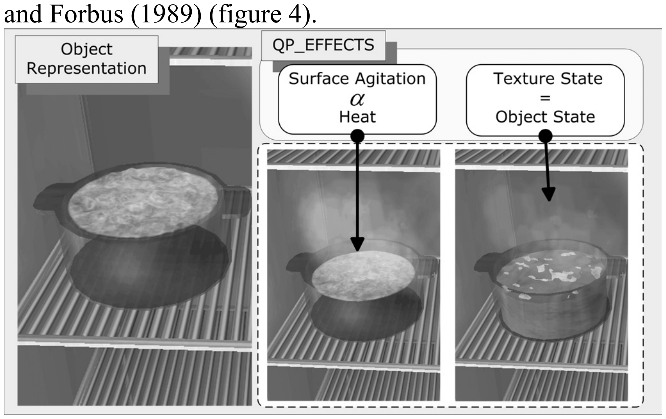

Figure 4: Reflecting State Changes by Altering Textures

The characterisation of an object and its various states is the first stage in the object's visualisation. This characterisation represents an objects initial (unaltered) state and modified in various ways by the different visualisation effects. The visualisation of the processes has to represent not only the level of process activity, but also the changes it triggers within the qualitative state variables. The blending of 3D graphical primitives from a library of existing animations achieves this. Such a library of animations represents process activity (boiling, evaporating) and the variations of qualitative variables (decreasing amount of liquid), which are displayed in conjunction with object appearance. Figure 1 shows boiling effects where water level, water activity and steam are all shown via the use of a volume actor (with mesh and texture), a fluid surface and a diverser particle emitter respectively.

This style of visualisation can be evolved by a combination of blending a representation of the object with the changing qualitative variables. For example, in one of the environments we have implemented the combustion of a candle. The modification to the height of the candle is by the rate of combustion. The activation of the combustion of the candle causes the creation of a new light source within the environment and a particle effect for the soot. The amount of light and the height of flame are dependent upon the qualitative rate of combustion, whereas the amount of soot is dependant upon the substance. The change in object representation, (i.e. its height) is blended with the change in texture for the candle to produce the burning effect (figure 5).

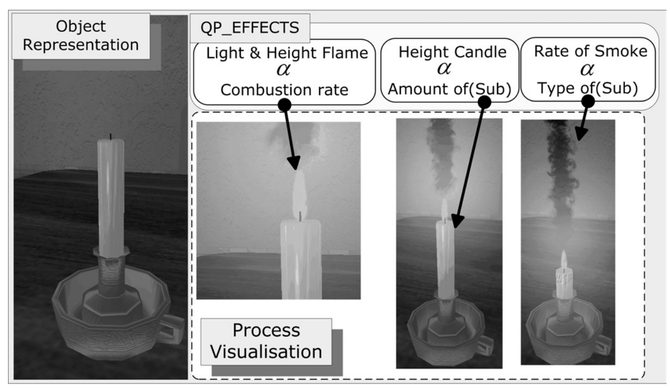

Figure 5: Blending Object Representation with Evolution of Variables

The operational devices within the system can provide many forms of interaction. The most common of these operational interactions being switches and taps for the kitchen environment. The decision to produce a visualisation for the filling effect within the system has the advantage of user familiarity. The process visualisation is achieved by using particle effects and by creating a new object for the flow representation. Thus, the creation of the visual effect for the flow directly comes from the user interaction with the tap. The created object represents the rate of flow into the container and its radius depends on its rate. These QP effects for the "fluid flow" process, relate the qualitative variable data for the rate of fluid flow (flow-rate) to the radius of cylinder for the flow object (a representation of the

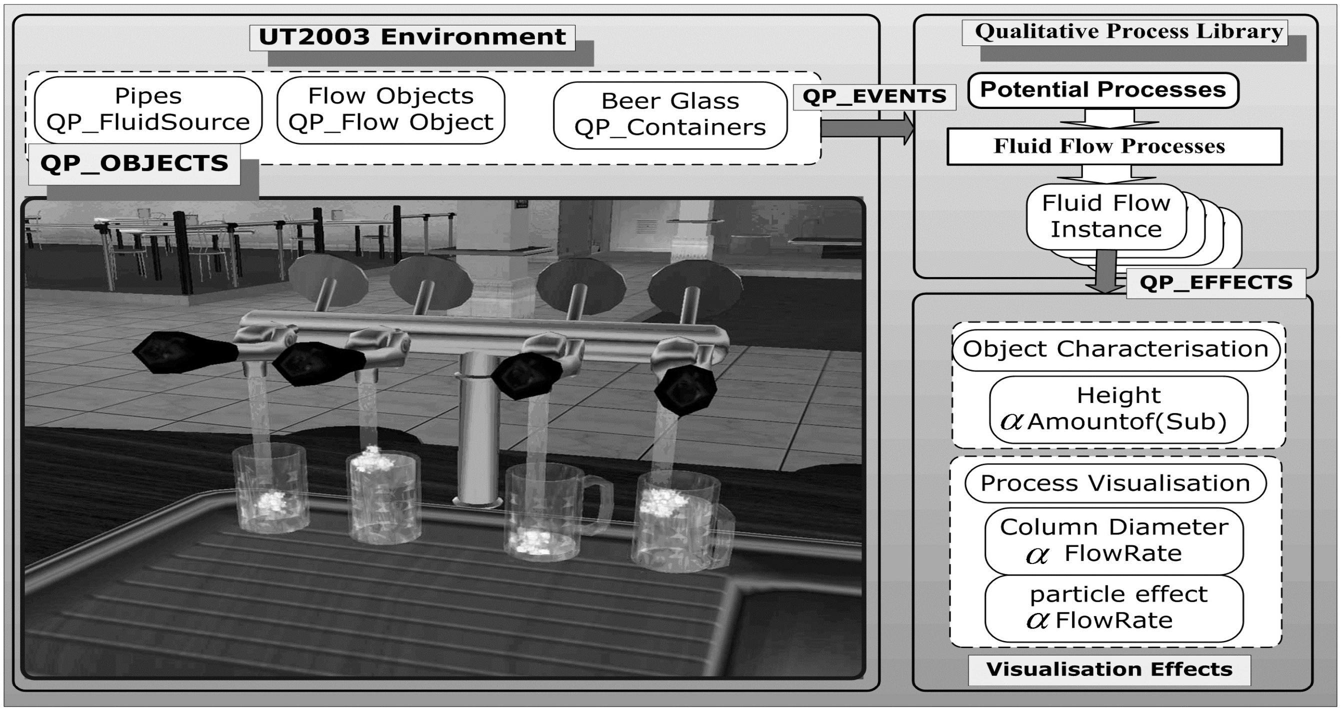

Figure 6: Multiple Fluid Flow Processes

column of fluid from a source), which provides a visual representation for the flow process. Enhancement of this effect is achieved by animating the texture to produce the appearance of a moving liquid and by the use of a particle effect at the surface of the substance, which represents the rate of fluid flow. The creation and destruction of these instances of these graphical primitives occurs when the fluid flow process becomes active or inactive. These graphical primitives are easy to render by the system, meaning many such examples can be present in the system simultaneously. This is an example of a simple qualitative behaviour encapsulation that can be visualised by the basic elements and controlled by the operation of a switch, an operational interaction of the user.

We have developed an ontology for the representation of the objects within an environment, that can be capitalised upon by the QP system during initialisation when selecting potential processes. This ontology uses descriptors such as heat source, fluid source, fixed object, movable object, un-sealable /sealable container, substance, flow. Adding these descriptors to an object makes it a QP object. For example, a QP object placed within the oven will activate "enter volume" that gives the data of the object that entered the oven. Since both objects have properties, which describe them as QP objects and one is a QP heat source. This generates the QP event "heat flow aligned", which is sent to the qualitative system. The simulation finds the processes between the oven (an immovable heat source) and the object (a movable QP object) and sets the process instances "heat aligned" precondition to true. In the qualitative system, two heat flow processes should exist, due to heat being able to flow from/to either object. However, one of the heat objects is a heat source and so the qualitative system does not instantiate this potential process. Now that the process has passed its preconditions, the second step in the process activation consists in testing the quantity conditions between the processes instances individuals' quantity variables. The process will become active starts if these conditions are satisfied. If the process becomes active, then a QP effect, *QP_start_heat_flow_process*, is generated and relayed to the visualisation engine. Since, the effect includes the process data for the rate of heat flow this allows the visualisation engine to give an indication that a process is active and a representation of how active it is.

In the discussions, we have focused on the representation and visualisation of the system purely within the 3D graphical environment. In the architecture of the implemented system we presented we describe the qualitative system as having being implemented as a separate module, which affords us with another opportunity to visualise the system. Visualisation of the system performed at the qualitative engine stage benefits from not having the usual restriction of having only the information that is contained in the event system. Instead, we have complete access to all the data. The availability of all the data within the qualitative system allows the normal representations such as the plotting of graphs and representations of relationships.[Bouwer,2001] For example, we are changing the amount of water in the glass by using a fluid flow process. However, by using the ontology we can specify the glass to hold an infinite amount so the water will not overflow, but the glass will get heavier (due to the mass of water) so the user will eventually be unable to move the glass. As the glass fills, it eventually reaches the limit point for the glass/container mass, which generates a QP Effect sent from the qualitative physics

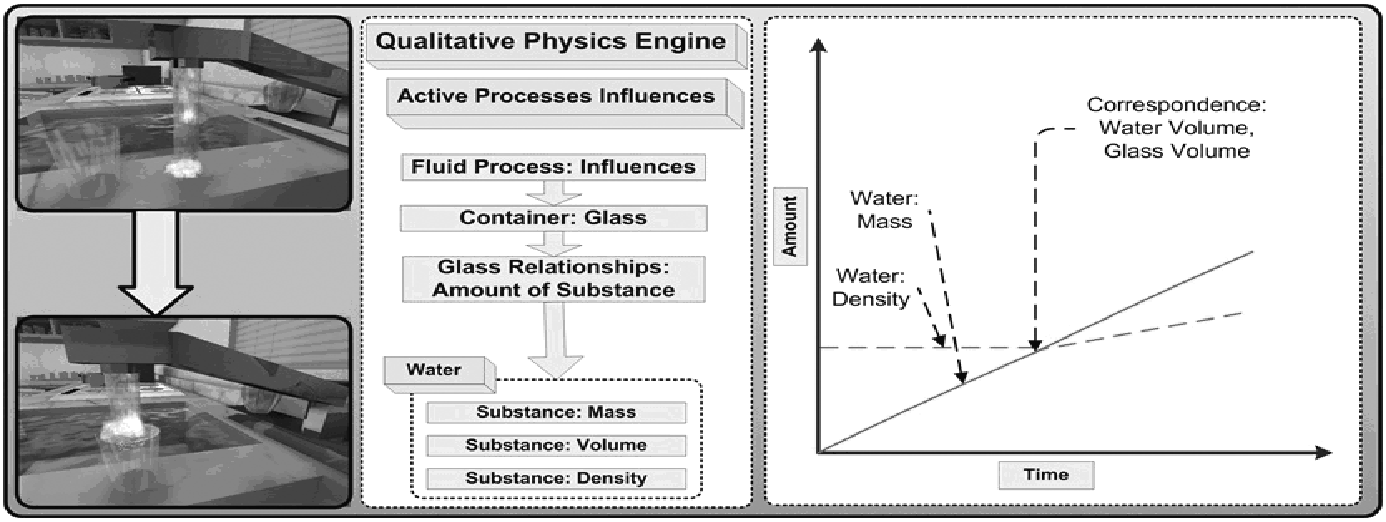

Figure 7: Graph for Fluid Flow Processes.

engine to the UT 3D virtual environment. This event informs the virtual environment that the mass has passed a certain value and makes the object immovable. The Qualitative system then enforces a correspondence between the water volume and the glass volume. Since, we have specified relations between mass, volume, and density within the qualitative system and now fixed the volume of water. This means the density will change, shown in figure 7.

Conclusion

We have presented our implementation of a visualisation system for Qualitative Physics within Virtual Environments, based on Qualitative Process Theory. The path we chose for implementation uses the formalism of QPT, due to its high-level representation of the physical behaviours. To implement the qualitative system, we use the discretisation of its domains to generate events used to control the visualisations within the Virtual Environments.

The implementation challenges we faced were how to integrate a qualitative system and how to retain user interactivity. The event-based system provides a sound basis for the integration of the library of processes as it enabled us to solve these initial implementation problems.

We had a problem deciding upon how to best perform the envisionment for the visualisation of a qualitative system. However, within the library of qualitative processes we formalised, we did not require complex envisionment procedures, in part due to the granularity of the model and in part, a wish to make as many objects and processes available for interaction.

The most important conclusion lies in the diversity of behaviours that are visualised in the virtual environments. In future developments we aim to expand the qualitative library and explore methods for model fragment interactions

Acknowledgments

Work funded in part by the European Commission through the ALTERNE project (IST-38575).

References

Bouwer, A. and Bredeweg, B. (2001). *VisiGarp: Graphical Representation of Qualitative Simulation Models* Gautam Biswas (ed). Proceedings of the International workshop on Qualitative Reasoning, QR'01, pages 142-149, San Antonio, Texas, USA, May 17-19.

Bailey-Kellogg, C. and F. Zhao, (2001) "Influence-based model decomposition," *Artificial Intelligence*, 2001, 130:pp. 125-166

Collins, J. and K. Forbus. (1989). *Building qualitative models of thermodynamic processes.* Proceedings of the 3rd international workshop on Qualitative Reasoning 1989, Stanford, CA, May.

Cavazza, M., Hartley, S., Lugrin, J.-L. and Le Bras, M., (2002). *Alternative Reality: Qualitative Physics for Digital Arts*, Proceedings of the 17th international workshop on Qualitative Reasoning 2003, Brasilia, Brasil.

Cavazza, M., Hartley, S., Lugrin, J.-L. and Le Bras, M., (2003) *Alternative Reality: A New Platform for Digital Arts*, ACM Symposium on Virtual Reality Software and Technology(VRST2003), pp. 100-108, Osaka, Japan, October.

Forbus, K.D.(1984), *Qualitative Process Theory*, Artificial Intelligence, 24, 1-3, pp. 85-168, 1984.

Forbus, K. D. (1989) *The Qualitative Process Engine* D. Weld, J. de Kleer (Eds.), Readings in Qualitative Reasoning About Physical Systems, Morgan Kaufmann, 1990.

Erignac, C. (2000) *Interactive semi-qualitative simulation*. Proceedings of the 14th international workshop on qualitative reasoning (QR2000), Morelia, Mexico. June, 2000.