DESIGN AGENTS IN VIRTUAL WORLDS

A User-centred Virtual Architecture Agent

MARY LOU MAHER, NING GU
Key Centre of Design Computing and Cognition
Department of Architectural and Design Science
University of Sydney, 2006 Australia

Abstract. This paper presents a User-centred Virtual Architecture (UcVA) Agent, a kind of rational agent capable of representing a person in virtual worlds and designing virtual worlds based on current needs. The two major structural aspects of this agent are Avatar Agent and Design Agent. The UcVA agent in this paper is presented from various perspectives: structure, process, and use scenarios. A proposed application of these agent models and an analysis of User-centred Virtual Architecture designed by this agent illustrates and evaluates this approach to designing virtual architecture.

1. Introduction

The design and development of virtual worlds has focused on the implementation and rendering of the 3D models that provide the place infrastructure. Our research involves the development of user centric models for Virtual Architecture that builds on implementations of virtual worlds, but also includes design agents. While Virtual Architecture can be understood as having its roots in physical architecture by providing 3D infrastructure, the software available to build virtual worlds makes it possible for these worlds to be highly interactive and dynamic. From this perspective, designing Virtual Architecture can go beyond the conventional 3D modelling and converting processes of traditional Architecture to apply agent models as the basis for its representation and implementation. This leads to the development of virtual places that are dynamically designed as needed, without the legacy of the persistent infrastructure of physical architecture.

This paper presents a so-called "user-centred" approach to Virtual Architecture and the UcVA Agent that designs Virtual Architecture. A User-centred Virtual Architecture (UcVA) Agent is a computing agent, which on one hand represents a person from the real world as an Avatar

Agent in the virtual world capable of providing a kind of agency for collaborating with other agents and interacting with the virtual world, and on the other hand acts as a Design Agent capable of designing and constructing dynamic virtual places for the Avatar Agent as needed. Hence virtual architecture becomes user-centred which means the virtual places will be created dynamically based on the current needs of the users. The UcVA Agent model is illustrated with various UML diagrams. The application of this model is introduced using ActiveWorlds bots.

2. Rational Agents

A rational agent is one that is able to reason about itself within an environment and then act based on its beliefs and desires (Wooldridge, 2000). This distinguishes a rational agent from the computational agents that perform actions based on predefined events, such as search agents on the web. There are several models of rational agents, some of which we consider here as the basis for developing our agent model.

Russell and Norvig's Reflex Agent diagram (1995) illustrates a rational agent that reasons about what the world is like and what the agent should do (see Figure 1). The Utility Agent (Russell and Norvig, 1995) illustrates a rational agent with additional beliefs and desires about the world and its own actions (see Figure 2).

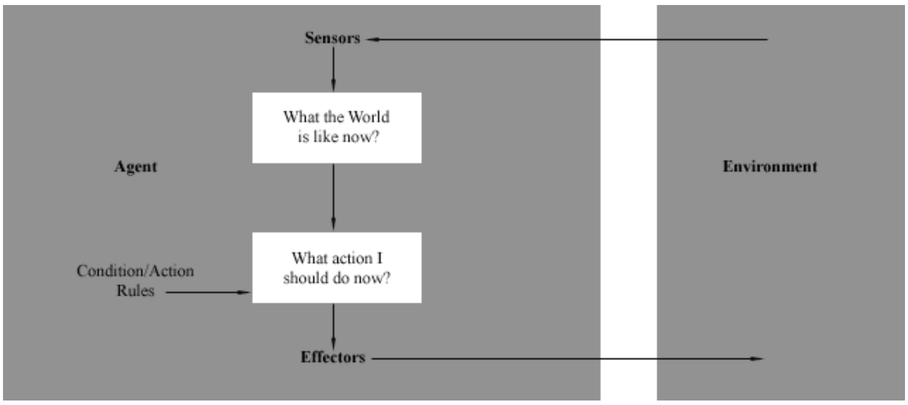


Figure 1. Reflexive Agent (from Russell and Norvig, 1995)

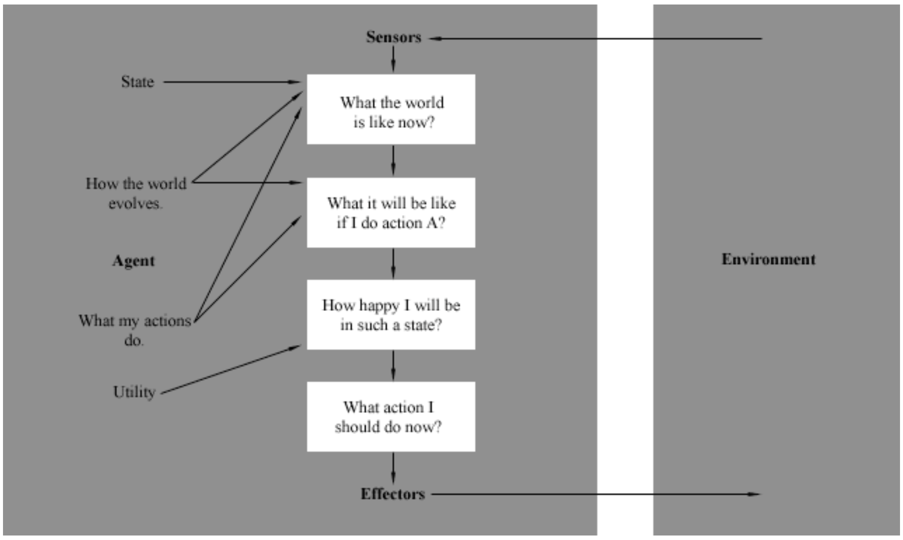


Figure 2. Utility Agent (from Russell Norvig, 1995)

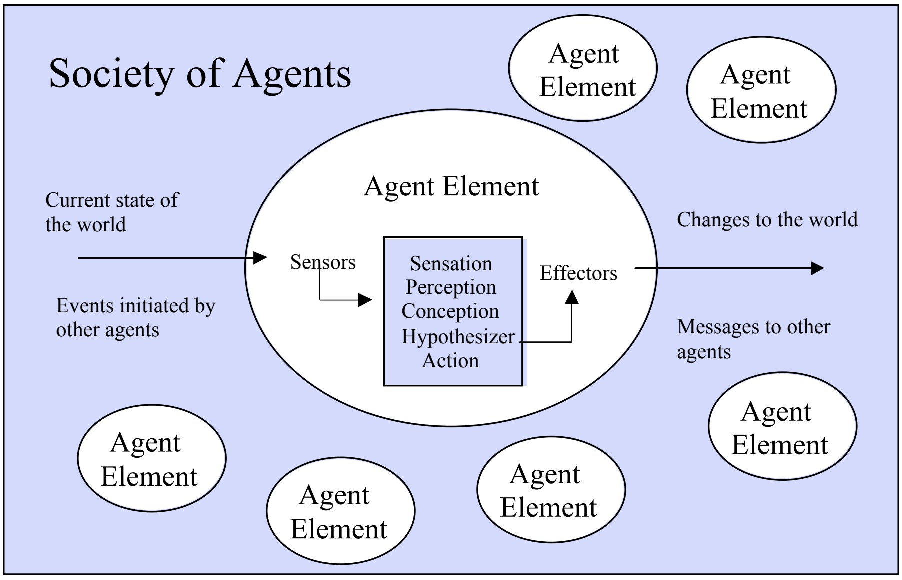


Figure 3. A Society of Agents for 3D Multi-user Virtual Worlds (Maher and Gero, 2002)

Maher and Gero (2002) propose a multi-agent system as the core of a 3D multi-user virtual world. Each object in the world is an agent in a multi-agent system. The agent model provides a common vocabulary for describing, representing, and implementing agent knowledge and communication. The agent can sense its own environment and can modify the spatial infrastructure needed for a specific collaborative or communication need of the users of the world. Their society of agents model is illustrated in Figure 3.

3. User-centred Virtual Architectural Agent Model

The Russel and Norvig agents presented above look at a single agent and its reasoning processes. The society of agents from Maher and Gero look at a single agent as a component of a virtual world with an existing infrastructure. Here, we look at an agent that has two main components: an avatar and a designer. The main components of the UcVA agent are considered as the basis for developing the overall structure of the agent model. The UcVA Agent then is further expanded from a process perspective. From a process perspective, we look at how to implement those characteristics as computational processes. And last one is a scenario perspective from which we use a scenario to study the information-flow among different components of the UcVA Agents, other agents or objects, users and actors within this scenario. The UcVA Agent model is illustrated using UML class diagrams.

3.1. UCVA AGENT MODEL: A STRUCTURE PERSPECTIVE

The aim for developing the UcVA Agent is to go beyond the conventional 3D modelling and converting processes, and to develop virtual places that are dynamically constructed as needed, without the legacy of the persistent infrastructure of Physical Architecture. This requires the UcVA Agent represent not only a user but also parts of the virtual place where the UcVA Agent resides. Figure 4 is a UML class diagram of a UcVA Agent. The red lines indicate the whole-part class relationships. The diagram shows a UcVA Agent as a whole class, which has two part classes, an Avatar Agent and a Design Agent. An Avatar Agent represents a user. It provides a kind of agency in the virtual place to communicate and collaborate with other agents and interact with the virtual place. A Design Agent develops virtual places for Avatar Agents by applying design rules/principles based on the agreements reached by the Avatar Agents. Hence, a part of the UcVA agent is a design component.

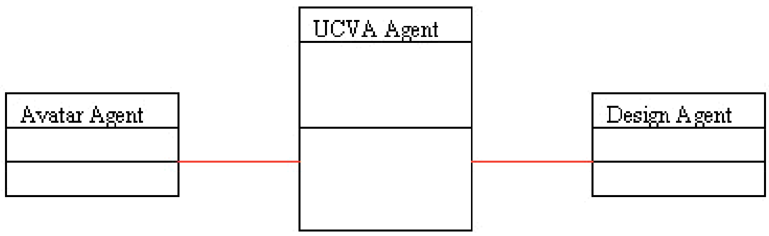


Figure 4. Agent Structure with Attributes and Operations are Omitted

With the definition of these two major aspects of a UcVA Agent, the representation of place as the design result of the Design Agent is included as one of the aspects when representing a person. Hence a virtual world can be considered as a society of UcVA agents. In this society, there is only one generic model for all the agents. External communication among different UcVA agents is largely simplified, they will be transformed as internal communication between Avatar Agent and Design Agent within each agent unit. The dependency between avatar (representing a person) and place (as design result by Design Agent) becomes a major factor for the evolution of virtual worlds. Places are generated and changed according to the needs of people who are using them. People are interrelated, as they are the units to not only construct the virtual community (via representing each person as Avatar Agents) but also design, implement and maintain their virtual places (via including Design Agent as part of the UCVA Agent).

3.2 UCVA AGENT MODEL: A PROCESS PERSPECTIVE

An agent is a system that operates independently and rationally, seeking to achieve its goal by interacting with its environment. As a rational agent (Wooldridge, 2000), it has goals and beliefs and executes actions based on those goals and beliefs. The process perspective considers the reasoning processes within the agent to identify how it reasons about itself and the virtual world. The main computational processes in an agent model for virtual worlds are sensation, perception, conception, hypothesiser, and action activation, as outlined in (Maher and Gero, 2002), and illustrated in Figure 5:

1. Sensation transforms raw inputs into something more appropriate for reasoning and learning.

2. Perception finds patterns in the sense data that are used in developing the agent's concepts of the world.
3. Conception assigns meaning to the patterns that situates these patterns in the context of virtual world.
4. Hypothesiser identifies the relevant goals for the agent based on comparing its expectations with the current state of the world.
5. Action activation reasons about how to achieve the goal or goals by defining a set of actions to be activated.

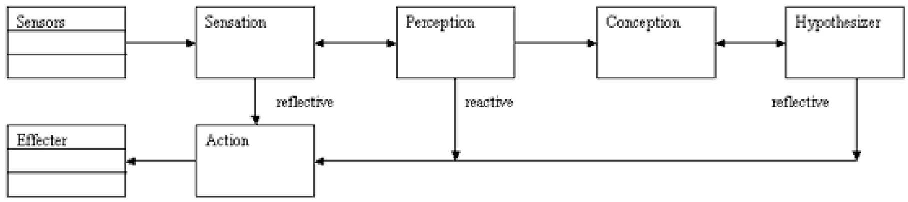


Figure 5. Agent model for 3D virtual worlds, from (Maher and Gero 2002)

The dynamic characteristic of an agent system relies on the hypothesiser and action activation, which determine how the agent responds to different situations. The situation is defined by the data sensed from the sensation process. Percepts and concepts can be understood as the knowledge for reasoning, realising and learning. The perception and conception processes define what type of response the agent will have to the data the agent senses, that is, they define agent behaviours. Maher and Gero (2002) consider the following three levels of agents for 3D collaborative virtual worlds:

1. Reflexive: this agent type is characterised by a fixed response to a fixed event triggered by the mouse or keyboard.
2. Reactive: this agent type is characterised by a reasoned response to an expected event in the virtual world.
3. Reflective: this agent type is similar to the agent behaviour of the reactive type above, except the object is able to consider alternative hypothesis based on learned concepts about the world.

We modify the process model of Maher and Gero for the UcVA Agent to accommodate the two separate functions: the first one is to provide a kind of agency for collaborating with other agents and interacting with the virtual world, and the second one is to design and construct dynamic virtual places as needed. These two functions are realised by two separate parts: Avatar Agents and Design Agents. The kinds of activities each agent part performed can be seen as follows:

1. Avatar Agent: negotiates, decides to meet with someone, talks, interacts with its own world of information, …
2. Design Agent: designs, builds, and deconstructs places for the Avatar Agent to carry out certain activities.

In the followings models, each agent part is illustrated separately. The main computational processes for the Avatar Agent (see Figure 6) are defined as sensation, perception, conception and action. In addition to the processes, there are two interfaces with the world: the sensors and the effectors. A sensor takes the information from the world and sends it to a process. An effecter takes information from an agent process and effects a change in the world. The red lines indicate the whole-part class relationships. These reasoning processes could be realised by applying different methods such as rule-driven, goal-driven, case-based reasoning (i.e. adapting agent memory) and so on. The knowledge associated with each process is generalised as a Rulebase in the model. Learning as one kind of actions expands Rulebase based on past experiences.

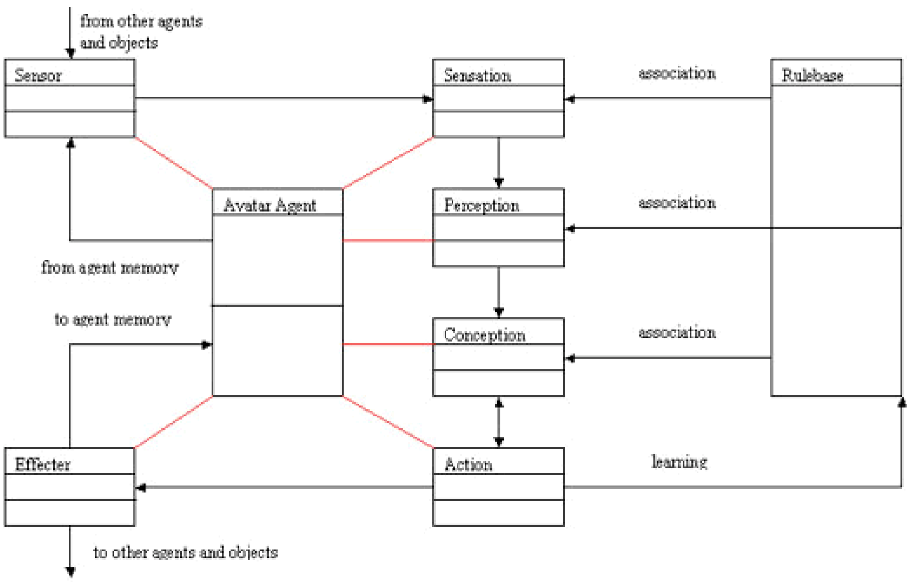


Figure 6. Avatar Agent Processes as Classes with Attributes and Operations are Omitted

One of the actions performed by the Avatar Agent is to provide information to the Design Agent, so the Design Agent would know how/what to design, build and deconstruct. The main computational processes for the Design Agent have sensation, perception, conception in common with the Avatar Agent, but the agent also has a process called design (Figure 7). Similarly, the agent has sensors and effectors in common with the Avatar Agent. For the Design Agent, the major actions are to design, build and deconstruct virtual places. The Design Agent is defined as a whole class. The red lines indicate the whole-part class relationships. Similar to the Avatar Agent, there is also a Rulebase defined in the model. For the design agent the Rulebase would include the design rules or principles for designing virtual places. This Rulebase can be expanded through past design experiences and cases.

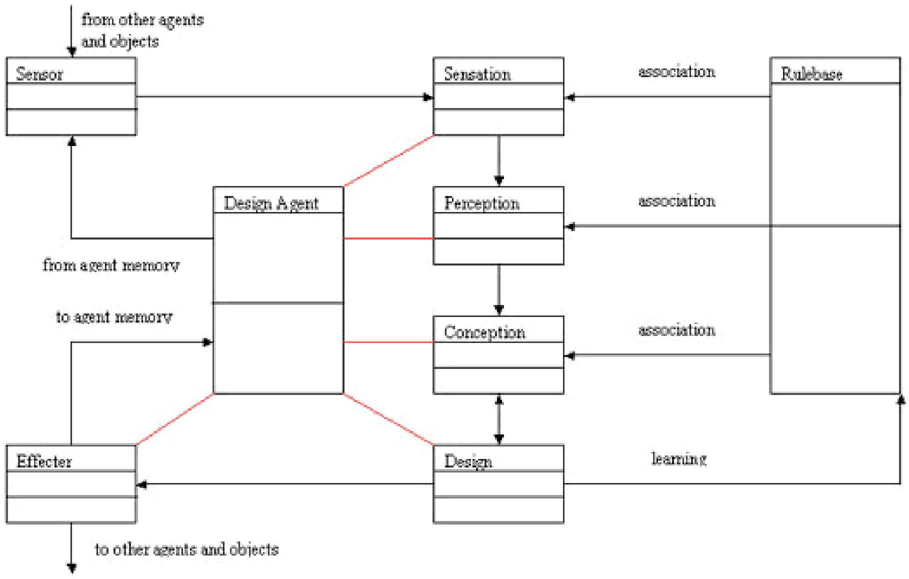


Figure 7. Design Agent Processes as Classes with Attributes and Operations are Omitted

The aim of defining the UcVA Agent model is for designing virtual worlds. Maher and Gero's model (Figure 5) have a similar purpose of modifying the virtual world based on the current needs of the users. In terms of representation, the virtual worlds resultant from both models are a society of agents. However, there is a significant difference. The virtual world resultant from Maher and Gero's model is a predesigned infrastructure, where each element of the world is an agent: each avatar, and each place component, such as a wall. In a virtual world resultant from UcVA Agent, practically, there is only one kind of agent in the world. A UcVA Agent represents a person and has the ability to build virtual place for itself and other agents as needed. It is not necessary to define virtual places separately. Hence, there is an important synthesis process of combing the Avatar Agent model and the Design Agent model for the definition of the UcVA Agent.

Based on the Avatar Agent model and the Design Agent model, a complete UcVA agent model from a process perspective is developed using Sensors and Effectors as the interface between the UcVA agent and the virtual world. The main computational processes are defined as sensation, perception, conception and the combination of action (from Avatar Agent) and design (from Design Agent). Together with Sensor and Effecter, the UcVA Agent whole class is defined (see Figure 8), the red lines indicate the whole-part class relationships. In addition to the four computational processes sensation, perception, conception and action, the UcVA designs. It also enables learning, which expands the Rulebase based on past experiences. The designing process reflects the design component of the UcVA agent. The design process is based on the information received from conception. It applies relevant design rules and principles from the Rulebase for designing virtual places. The learning process is a self-improvement process for the UcVA Agent. Through learning, UcVA Agent constantly update design rules and principles in Rulebase as the design cases are generated.

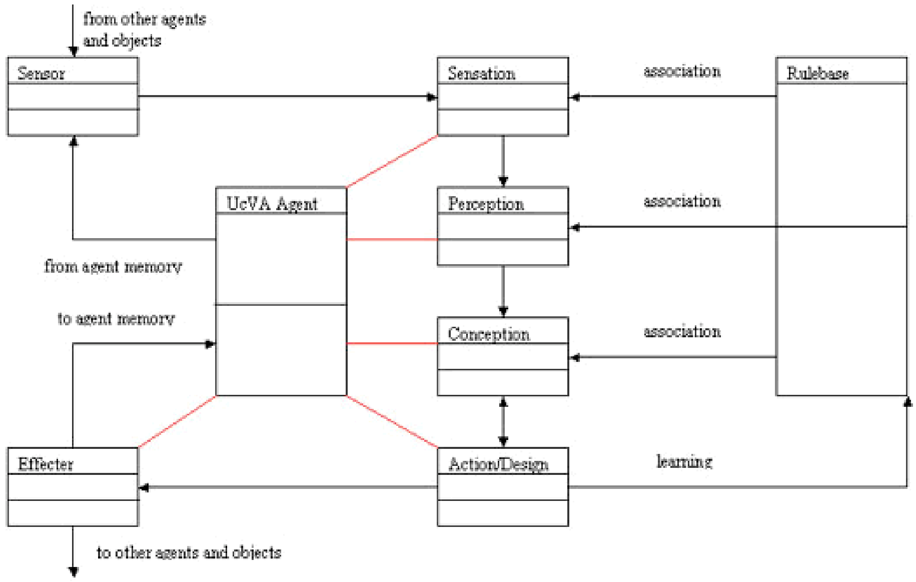


Figure 8. UcVA Agent Processes as Classes with Attributes and Operations are Omitted

3.3 UCVA AGENT MODEL: A SCENARIO PERSPECTIVE

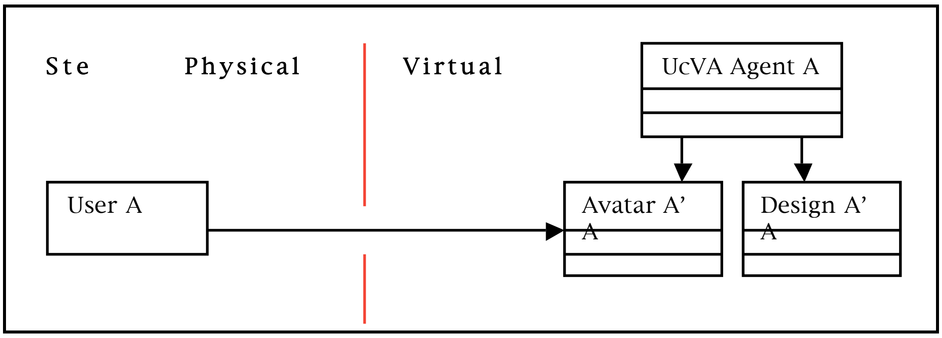
In some cases, an agent has the ability to operate usefully by itself, however the increasing interconnection and networking of computers is making this situation rare. In the usual states of affairs the agent interacts with other agents. In an multi-agent system for developing User-centric Virtual Architecture, there are interactions among Avatar Agents, Design Agents, various objects, users, actors like computer bots and so on. We use a simple scenario to develop a series of UML sequence diagrams to study the relationships among different scenarios related to the UcVA agent.

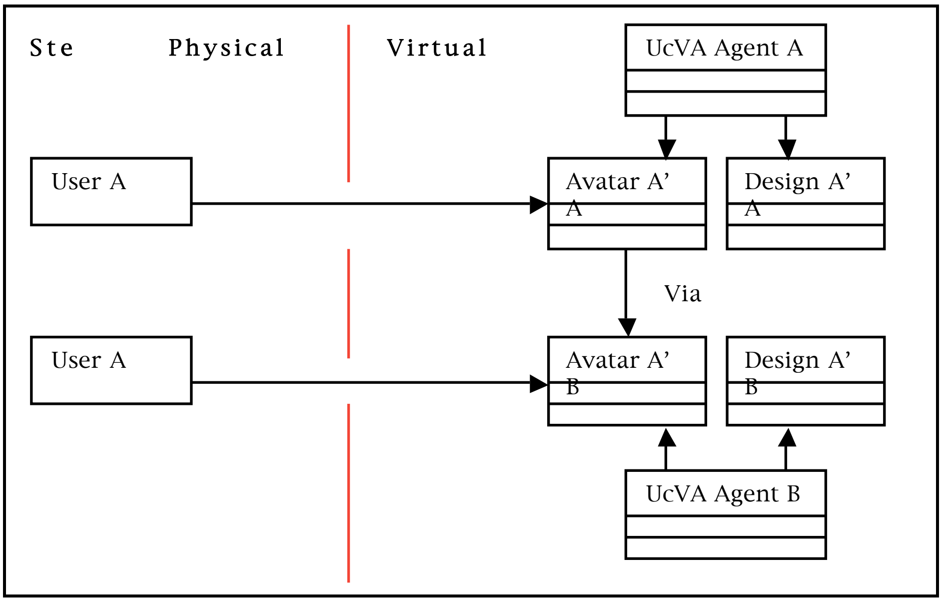
The scenario is developed as the following steps:

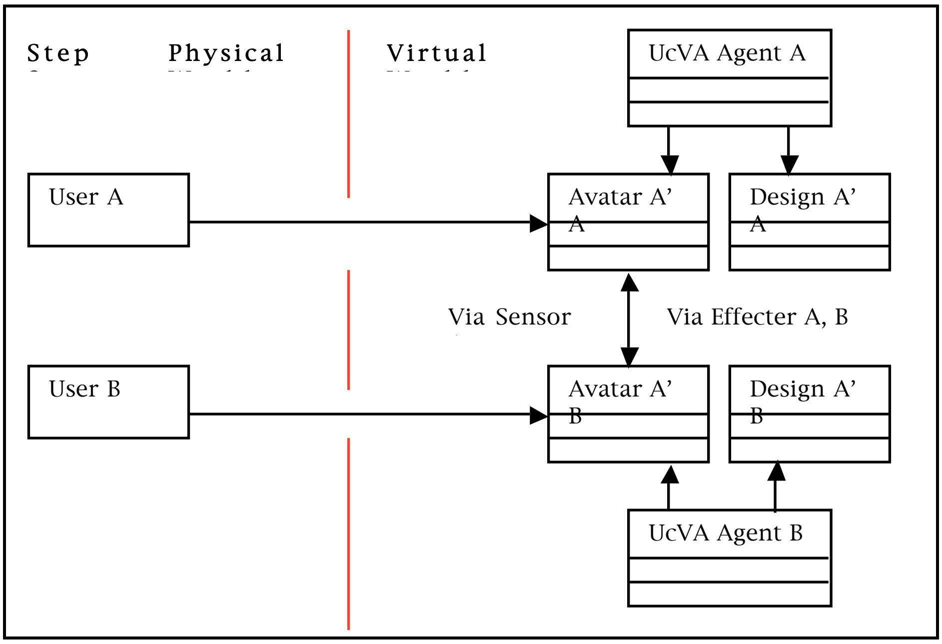
1. User A login to the UcVA world, he/she is visualised with the visualisation of his/her UcVA agent's Avatar Agent Part.
2. His/Her UcVA Agent (A) discovers another user B is also in the world through the UcVA Agent's Avatar Agent part.
3. The Avatar Agent part (A) negotiates with the other Avatar Agent part (B).
4. These two Avatar Agent parts agree to use the Design Agent part (A) to build a public place for both UcVA Agents.

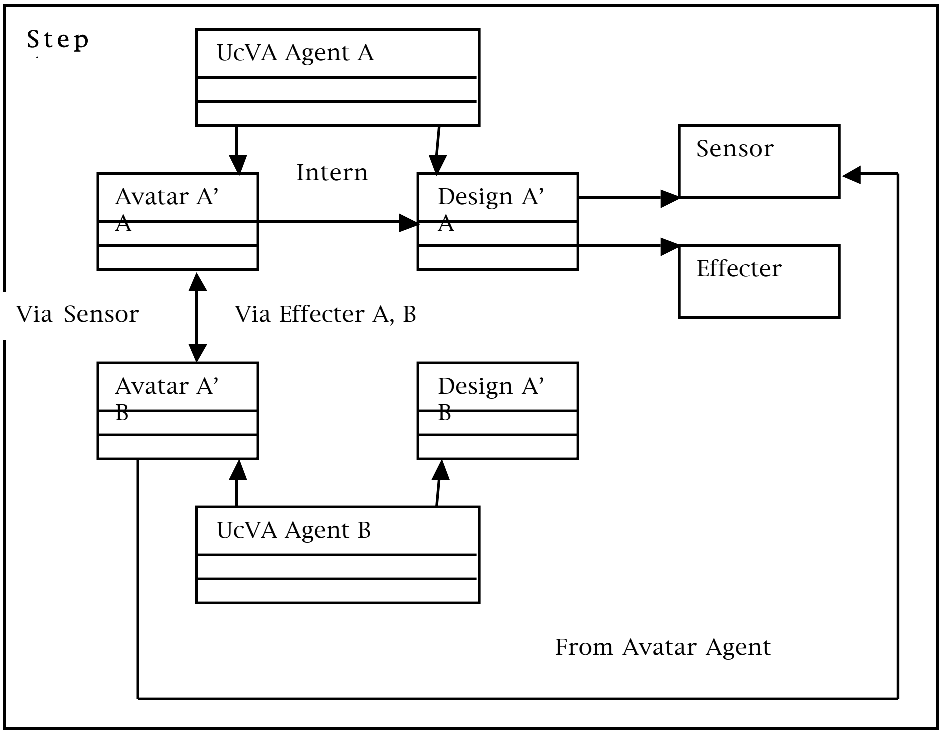
5. The Design Agent part (A) applies its design rules/principles to design a virtual place for both UcVA Agents' based on their needs.

To realise this scenario, we have developed flow diagrams to simulate the situations the UcVA Agents would encounter, and developed a series of UML sequence diagrams to lead to the ending. Figure 9 illustrates these steps showing the interaction between the users and the agents in the virtual world.






Step
Physical
Virtual
UcVA Agent A
User A
Avatar A'
A
Design A'
A
Via Sensor
Via Effecter A, B
User B
Avatar A'
B
Design A'
B
UcVA Agent B


Step
UcVA Agent A
Intern
Sensor
Avatar A'
A
Design A'
A
Effecter
Via Sensor
Via Effecter A, B
Avatar A'
B
Design A'
B
UcVA Agent B
From Avatar Agent

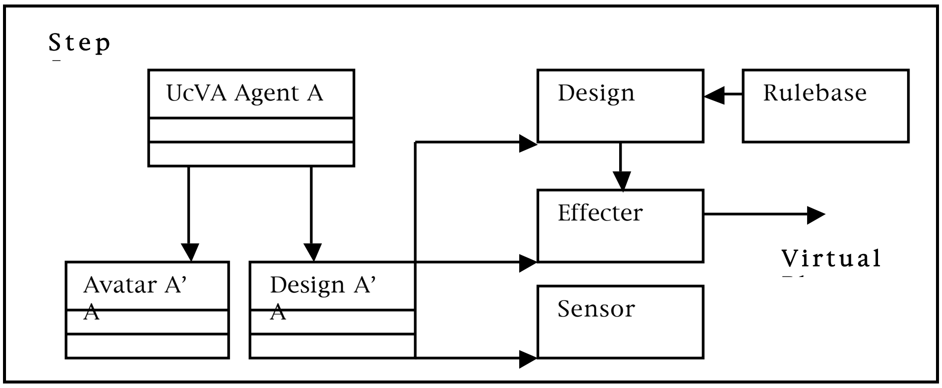


Figure 9. Illustration of Scenarios

4. Illustration of UcVA Agent Model Using AW Bots

The illustration of the UcVA Agent Model requires an implementation of a UcVA Agent within a 3D multi-user virtual world. A trial application is being developed based on Active Worlds[1] bots. AW is a multi-user 3D immersive virtual world in which users in the world can create and build new parts of the world. Each user is represented as a 3D animated character. Various AW bots developed by Hamfon[2] can retrieve information about the world from the server and interact with the building objects and avatars within AW in different ways, such as chatting (ChatBot), data or statistical analysis, building and modifying virtual places (BuildBot). Each bot is a separated computing entity running outside the AW server. Currently, each bot has a predefined set of actions that are triggered by specific user initiated events. Our approach is to make a bot an agent that is capable of reasoning about the events in the world and its own communication and design actions.

Figure 10 is an example of a virtual world designed by an Avatar Agent and a Design Agent. Although our model combines these two as part of one agent, in the current implementation they have separated visual representations. The character on the right hand side is the visual representation of the Avatar Agent representing an AW user called "Ning" and the character on the left hand side is the visual representation of the Design Agent. This design case shows the UcVA agent in its preliminary stage. The Avatar Agent here represents a designer from the physical world. The Design Agent is a computing entity owned by the Avatar Agent in the virtual world. The designer generates an existing

---

[1] http://www.activeworlds.com

[2] http://home.sprintmail.com/~ncrowle/hamfon/hamfon.htm

design of a virtual place and hard-codes the design data into the design database of the Design Agent. The Design Agent is controlled by the designer from the physical world via the Avatar Agent, and generates a virtual place based on the hard-coded design data for the Avatar Agent.

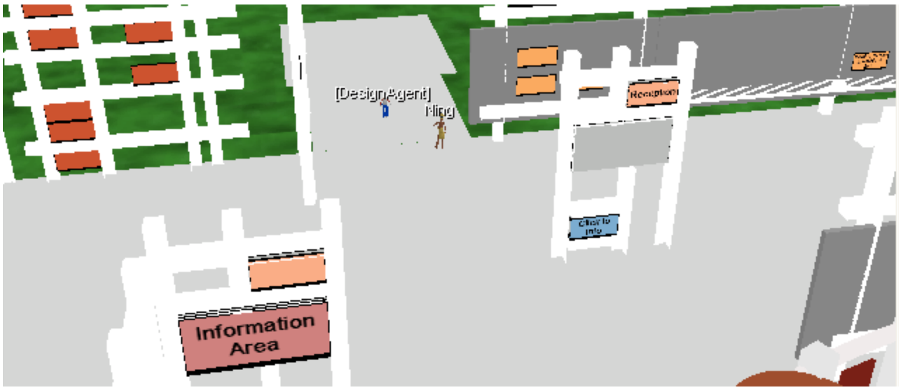


Figure 10. An Example of Virtual World Designed by an Avatar Agent and A Design Agent

There are obvious differences from the agent being demonstrated above using AW avatars and bots to UcVA agents. It is noticed that in the above design case: the Avatar Agent part and Design Agent part are represented separately. The rational agency has not yet been fully applied to either the Avatar Agent or the Design Agent. For example, the Avatar Agent does not have any actions on its own, it is controlled by the user, and the Design Agent does not reason about the needs of the user but simply builds a new virtual place when instructed based on hard coded design data. Hence in order to develop a UcVA Agent based on AW and AW bots, we need to consider the following issues:

1. Use a AW 3D animated character as a basis for developing the Avatar Agent part, this requires integrating various computational processes of the UcVA Agent: sensation, perception, conception, action activation into the currently vision-only AW 3D character.
2. Further develop AW BuildBot into the rational Design Agent part, this requires changing currently hard-coded design instructions into dynamically goal-driven design rules/principles database.
3. Combine the Avatar Agent part and Design Agent part as a whole UcVA Agent.

5. Analysis of User-centred Virtual Architecture

The motivation of developing a model such as the UcVA Agent is to change the current status in designing virtual architecture. Currently, in terms of interactions, online activities supported by virtual architecture have been largely expanded. Interactions within these environments are more than shooting and chatting. Other kinds of intelligence are required to cope with these changes. Programmers have added many new interactive entities into these environments. However they have become part of the preprogrammed infrastructure. There is a need for revising representation methods for designing virtual architecture so that we can go beyond the preprogrammed infrastructure that is our example in physical architecture.

In terms of representation, virtual architecture is currently designed and implemented in a so call place-centred manner, which means the resultant virtual architecture is structured as static components in the virtual domain. Similar to physical architecture, this type of virtual architecture exists whether people use it or not. The design and existence of place-centred virtual architecture is separated from users. There are obvious disadvantages of place-centred virtual architecture. Firstly, a lot of indirect communications between the users and place-centred architecture are needed to be defined in order to establish the interactions between/among them. Secondly, at the implementation level, the components of virtual architecture need to be predefined, programmed and stored online statically. That’s why people will see they may be walking alone in one of the well-rendered Internet cities as if they have survived from neutron bombs. These place-centred virtual cities are designed, implemented and stored permanently in particular servers. They do not reflect the actual needs and situations of current users. The cities are silent, pointless and dead. If an agent-based approach is taken, the static nature of the place infrastructure can be seen in Figure 11.

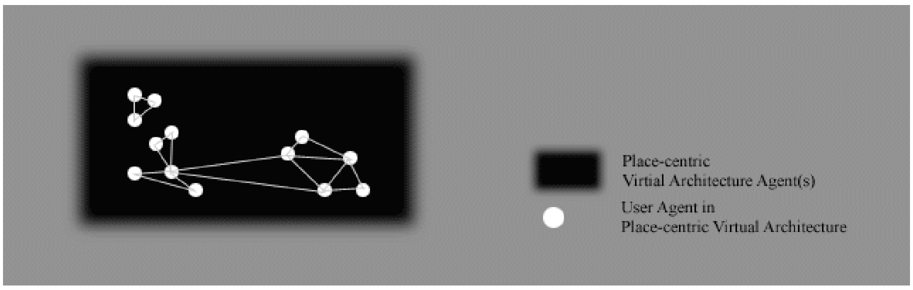


Figure 11. Place-centred Virtual Architecture

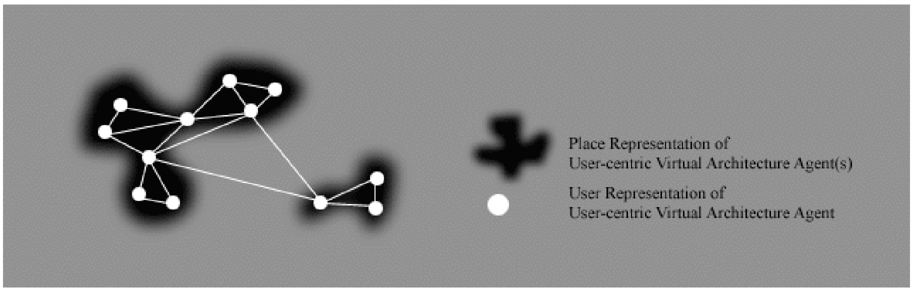


Figure 12. User-centric Virtual Architecture

Current examples of virtual architecture are adapted from a conventional representation method of virtual architecture, a so-called "place-centred" approach, from which virtual architecture is understood as a place or an assembly of places with permanent structure. These places are user-independent. Users have separate representations within these places. An ideal type of virtual architecture should be user-centred. User-centred virtual architecture is associated with the representation of users. Hence the interactions between them become internal and direct. The existence and the states of user-centred virtual architecture are situated based on users' needs. User-centred virtual architecture is stored as dynamic design elements and sets of rules (i.e. for designs and interactions) rather than static metaphorically designed architectural components. User-centred Virtual Architecture is the design result of the UcVA Agent, which is presented in this paper. This kind of virtual architecture is illustrated in Figure 12.

6. Summary

In summary, this paper presents an agent approach to solve design problems in virtual worlds. It presents an agent approach for representing, designing and implementing Virtual Architecture, which leads to virtual worlds that are dynamically designed and constructed as needed, without the legacy of the persistent infrastructure of physical architecture. This User-centred Virtual Architectural agent model has been illustrated from three different perspectives: structure, process and scenario, which provide a series of blueprints for theoretical understanding and technical implementation of UcVA Agent and User-centred Virtual Architecture. The paper also presents an application of this agent using AW avatars and bots, which identifies related issues for future study.

References

Maher, ML, and Gero, JS: 2002, Agent Models of 3D Virtual Worlds, *Proceedings of ACADIA 2002*, Pamona, California.

Maher, ML, and Gu, N: 2002, Designing Virtual Architecture: From Place to User Centred Design, *International Journal of Design Computing* **4**: http://www.arch.usyd.edu.au/kcdc/journal.

Russell, S and Novig, P: 1995, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, NJ.

Wooldridge, M: 2000, *Reasoning About Rational Agents*, MIT Press, Cambridge, MA.