SITUATED DESIGN OF VIRTUAL WORLDS USING RATIONAL AGENTS

MARY LOU MAHER AND NING GU
Key Centre of Design Computing and Cognition
University of Sydney, Australia 2006
Email address: mary@arch.usyd.edu.au

Abstract. The design and implementation of virtual worlds is separated in time from the use of the virtual world, and therefore is not directly related to an individual's experience of the world. We propose an approach to virtual worlds platforms in which the avatar, a visual representation of a user, has agency. As a rational agent, the avatar can respond to events in the world either through the human control of the avatar or via the avatar's agent. This kind of agency interacts with the world on the user's behalf and directly in response to the user's activities in the world. We use rational agents to reason about the use and the design of the virtual world. Our agent model is an adaptation of a design agent and includes a design grammar for automatically designing and building a portion of the world.

1. Introduction: Internet Gaming and Virtual Architecture

Internet games enable real-time interactions among players regardless of their geographical locations. With the development of various approaches to fast and efficient network applications, one has many reasons to believe that Internet games could potentially elevate the degree of interactivity far beyond those that other gaming forms could offer. One of these potential sources for generating interactivities comes from the reformation of online gaming environment design. Currently, the design and development of virtual worlds as gaming environments has focused on the implementation and rendering of 3D models that provide the place infrastructure as scenes. Most existing online gaming environments are pre-programmed by game designers or programmers defining different levels as 3D scenes and by associating various behaviors to the objects in the scene.

We have developed several virtual worlds and considered the combination of 3D objects with associated programmed behaviors to create interactive places for groups in collaborative situations [7, 8]. While we recognize that virtual worlds have their 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 worlds can go beyond the conventional 3D modelling and construction process. We propose the use of rational agents to design and build the virtual world in response to the users and events in the world. This research involves the development of situated design agents that extend existing platforms for virtual worlds [6, 5].

2. A Situated Agent Approach to Situated Design of Virtual Worlds

Agent-based computing started from 1970s. An increased interest of computer agents has been seen recently in the development of Internet applications influenced by the concepts of artificial intelligence and artificial life. Agents as designers need to account for a changing world and therefore their reasoning should be situated. Situatedness holds that "where you are when you do what you do matters" [1, 2]. The stance of situatedness distinguishes itself from other views that consider knowledge as being independent of its locus or application. The situated design agent defines situations by locating relevant elements in a context so that the decision and action taken reflects both the situations and the construction of the situations [2].

2.1. Situated Design Agent Models For Desiging Virtual Worlds

In a situated design agent, the central component is a Rational Agent. A rational agent is a system that operates independently and rationally, seeking to achieve its goals by interacting with its environment [9, 11]. It has goals and beliefs and executes actions based on those goals and beliefs. This distinguishes a rational agent from the computational agents that perform actions based on predefined events, such as search agents on the web. A situated design agent is proposed as a rational agent for dynamically designing and implementing virtual architecture. This agent, on one hand represents a person from the real world as an avatar in the virtual world. This agent-based avatar monitors the communication among the avatars and reasons about the activities in the world. The design component of the agent reasons about how to design and build portions of the world in response the users' needs and activities. We refer to these two components of the situated agent as the Communication Agent and the Design Agent.

Maher and Gero [4] study a society of agents, in which a single agent is represented as a component of a virtual world with an existing infrastructure. Based on the five main computational processes in an agent model for 3D virtual worlds outlined by Maher and Gero [4]: Sensation, Perception, Conception, Hypothesizer and Action Activation, the following six computational processes have been adapted for the development of our situated design agent.

1. Sensation: transforms raw sensor data from the virtual world into something more appropriate for reasoning.
2. Perception: finds patterns in the sense data that are used in developing the agent's concepts of the virtual world.
3. Conception: assigns meaning to the patterns that situates these patterns in the context of virtual world.
4. Hypothesising: identifies the design goals for the agent based on comparing its expectations with the current state of the world.
5. Design: reasons about how to achieve the design goals by selecting, combining, and locating elements that comprise the design.
6. Action Activation: identifies the actions needed to build the design given a design description.

2.2. Design Reasoning Using A Design Grammar

The design reasoning component of the agent uses the shape grammar formalism [10, 3]. Knight [3] summarises the definition and the components of a shape grammar: a shape grammar is a set of shape rules that apply in step-by-step way to generate a set, or language of designs. A shape grammar is both descriptive and generative. The shape rules are the description of the spatial forms of the designs. The basic components of the shape rules are shapes. Shapes can be understood as points, lines, planes or volumes. The designs are generated using shape operations. They are addition, subtraction and spatial transformation such as rotating, shifting and mirroring.

We consider the representation of the design of a virtual world as having a functional core and a visual shell [8], which could be described as the following two layers using a design grammar:

1. The descriptions of spatial forms of the design in the virtual world.
2. The descriptions of functions that support certain online activities.

The first layer relates to the generation of forms, the second layer attaches certain functional and behavioral aspects to these forms. We illustrate the generation of the example design rules with an analysis of an existing virtual world design, the CRC World. The CRC world is a virtual environment for collaborative research. It provides networked places for meetings, conference, exhibition and storage. The CRC world was not designed with shape grammars. However, a set of shape rules could be generated to reproduce designs that share a similar style with the CRC world. This purpose of the analysis is to:

1. Identify the shape rules and their components that could generate the forms of the CRC World, to be adapted to the design of our shape grammars for generating forms in the agent/meeting scenario.
2. Identify the functional aspects of the meeting/seminar rooms in the CRC World in order to prepare for representing the functional aspect of a meeting room in the scenario.

In the CRC World floor plan, the basic construction modules are square, rectangle, circle and ellipse. The floor plan can be analysed as explicit shapes of the above modules (Figure 1).

Architectural spaces are designed in 3D. The basic components of the shape rule for this purpose are usually perceived as volumes. However, the CRC world is a single-storey building block. There is also no complicated spatial intersection in the design. Hence the design of the CRC world can start from the generation of a 2D plan. The 2D modules will be replaced with 3D units in the end. One possible way to reproduce this plan is by generating shape rules (Figure 2, 3, and 4). These rules then can be applied step by step to generate the CRC World floor plan. Figure 5 illustrates this process, from a chosen initial shape to the final generated floor plan.

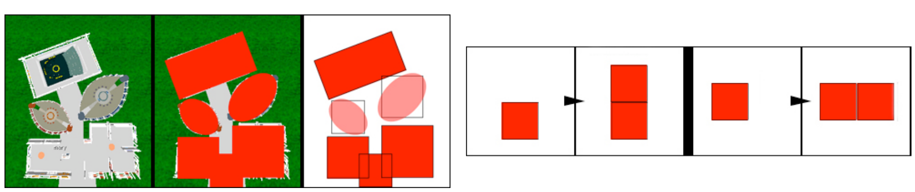

Figure 1 (left). The floor plan (left) is analysed as explicit 2D shapes (right).

Figure 2 (right) (from left to right). Rule 1: vertical addition; rule 2: horizontal addition.

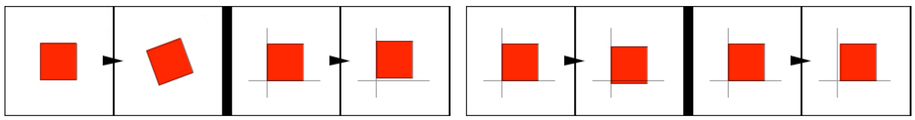

Figure 3 (from left to right). Rule 3: rotation; rule 4: vertical transformation 1; rule 5: vertical transformation 2; rule 6: horizontal transformation 1.

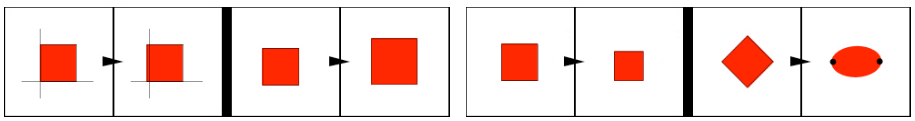

Figure 4 (from left to right). Rule 7: horizontal transformation 2; rule 8: scaling-up; rule 9: scaling-down; rule 10: replacement using oval shape.

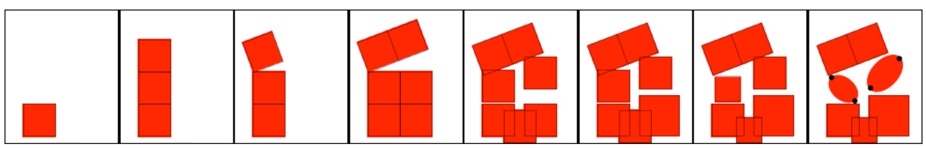

Figure 5 (from left to right). Initial shape: a square; step 1: to apply rule 1 twice; step 2: to apply rule 3 once; step 3: to apply rule 2 four times (two shapes overlap with each other at the bottom-right corner); step 4: to apply rule 4, 5, 6 and 7 multiple times; step 5: to apply rule 8 once; step 6: to apply rule 9 twice; step 7: to apply rule 10 twice.

The next stage is to replace the 2D shapes from the above floor plan with 3D units (Figure 6). The similar style of the frame-like walls used in the CRC World will be used to shape the vertical volumes. The final stage is to refine the design details, mostly to delete the repeated wall partitions in the overlapped units using the replacement rules (Figure 7).

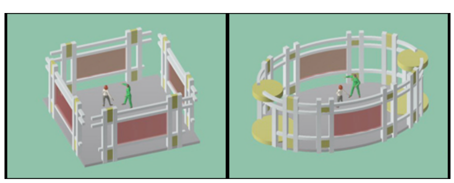
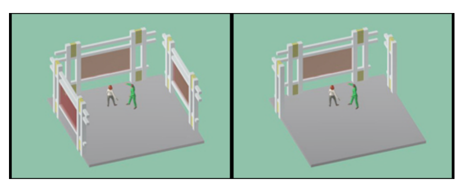

Figure 6 (left). A Standard Cubical unit: to replace a square shape; a standard oval unit: to replace aellipse shape.

Figure 7 (right). An cubical unit variation 1: to replace a standard cubical unit; An cubical unit variation 2: to replace a standard cubical unit.

The CRC World (Figure 8) is implemented in AW[1], a three dimensional object-oriented platform. Each object in the world is not only a model for visualisation, but also a media for supporting online actions. These actions provide interactivities for supporting various kinds of online activities. The key function of the meeting/seminar room of the CRC World is to host small meetings and seminars. In order to support these two activities, the meeting/seminar room is featured with two "screen" objects for posting slides during a presentation, and various "sign" objects on the frame-like walls for attaching web contents related to the meeting/seminar. There is also a floating ring placed in the middle of the room to focus the avatars' attention. During the meeting/seminar session, avatars would gather around the ring. The above three objects could be perceived as the functional and behavioral aspects that support the needs of having an online meeting/seminar in the CRC World. Hence we conclude the following three rules (Figure 9) as the descriptions of functions that support online meetings/seminars.

1. If a 3D unit is a meeting area, a floating ring will be placed at the center for gathering purposes, to focus people's attention.
2. If a 3D unit is a meeting area, two screens will be placed in the room for online presentation.
3. If a 3D unit is a meeting area, signs will be placed around the presentation screen for posting online documents.

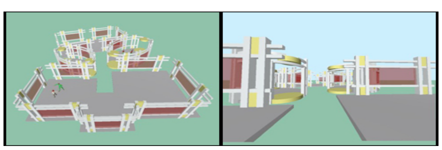
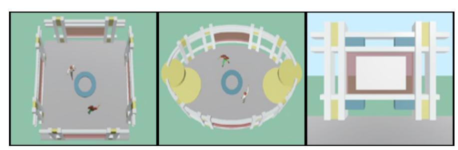

Figure 8 (left). Two views of the re-designed CRC World using a shape grammar.
Figure 9 (right) (from left to right). A standard cubical unit with a ring placed at the center; a standard oval unit with a ring placed at the center; a view of a side wall showing a presentation screen attached at the middle with two signs placed both above and below.

From the analysis of the CRC World design using a shape grammar, we have derived a set of design rules for generating CRC World-like forms and a set of rules for providing functional components for supporting online meetings. To control the size of the design, a set of constraints is needed to reflect the number of the persons who would be using the meeting room for example:

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

1. The initial shape is chosen to be a 8m X 8m square.
2. For less than 10 people, the addition rules are applied twice. For 10 to 20 people, the addition rules are applied three times.

The above rules for generating forms, and representing functional aspects of meetings are the major levels of reasoning included in the situated design agent. Each agent/user can have its own rules for designing different types of virtual worlds in various styles, or their rules can be merged to produce a hybrid design style.

3. Situated Design Agent for Internet Gaming Environments

In an Internet gaming environment designed using agents, each player can be represented by a situated design agent. The Communication Agent part provides a kind of agency in the gaming environment for the player to interact with other player and the world. The Design Agent part develops and modify the environment as needed according to the current situation sensed by the agent. The six computational process of the situated design agent presented earlier can be re-configured for a gaming environment.

The design grammar presented above is for a meeting room scenario. It is developed based on an existing design and intends to design a meeting room having a CRC-World-like style and providing functions for supporting online meetings. For the meeting room scenario, the shape rules are developed to satisfy the following requirements, or the requirements could be simply understood as the constraints for forming rules. They are: the number of persons that participate in the online meeting, a visual style that evolve from an existing design and some functional aspects that support a particular kind of activity: online meeting.

Design problems share some similarities. It is possible to develop shape rules for designing Internet gaming environments by outlining the requirements of these environments. Generally speaking, the constraints for forming the design rules are: the number of players in the game, types of scene (battle field, castle and so on) and some functional aspects that support particular kinds of activities (shooting, combating, treasure hunting, scoring and so on).

A starting point for generating design grammars for designing Internet gaming environments could come from an analysis on some existing Internet gaming environment:

1. Identify the shape rules and their components that could generate the scenes for a particular Internet game.
2. Define the functional aspects of the gaming environment, which refer to the interactivities/actions in the game.

4. Conclusion and Future Research

A situated design agent model has been proposed and implemented using a meeting room scenario. As discussed in section 3, the agent model could also be adapted to designing Internet gaming environments. Combining the design knowledge of a shape grammar and the reasoning algorithm of the Jess production system language, the situated design agent model provides a dynamic approach for designing and implementing networked environments for supporting various online activities. An Internet gaming environment developed using the agent model has much potential:

1. The design agent approach enables the Internet games to be user-centred, as the gaming environment would be automatically generated and modified based on the current situation of the game and the player rather than pre-programmed by designers or programmers.
2. The mechanism of the shape grammar can produce numerous variations that share the same style. This aspect allows the scene design in a game to be truly dynamic.
3. The shape grammar is also able to describe functional aspects such as actions and interactions, which could increase the degree of interactivity in an Internet gaming environment.

The focus of this paper is on the development of the design grammars as the generative part of the rational agent. The first stage of our future research will apply a more complete scenario to further study the whole reasoning process of the agent. The second stage will extend and enrich the six computational processes beyond the limit of specific design scenario, to support the design and implementation of virtual environments in general.

References

[1] Clancey, W. J.: 1997, *Situated Cognition*, Cambridge University Press, Cambridge.

[2] Gero, J. S.: 1999 Constructive memory in design thinking, *in* G. Goldschmidt and W. Porter (eds), *Design Thinking Research Symposium: Design Representation,* MIT, Cambridge, pp. I.29-35.

[3] Knight, T.: 2000, Shape Grammars in Education and Practice: History and Prospects, *International Journal of Design Computing*, Vol 2, http://www.arch.usyd.edu.au/kcdc/journal/vol2.

[4] Maher, M.L., and Gero, J.S.: 2002, Agent Models of 3D Virtual Worlds, *Proceedings of ACADIA 2002*, Pamona, California.

[5] Maher, M.L. and Gu, N.: 2002, Design Agents In Virtual Worlds: A User-centred Virtual Architecture Agent, in John S Gero and Frances MT Brazier (eds), *Agents in Design 2002*, MIT, Cambridge, MA, pp 23-38.

[6] Maher, M.L., 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.

[7] Maher, M.L. and Gu, N.: 2002, Virtual Worlds = Architectural Design + Computational Elements, Proceedings of ANZAScA, Deakin University, Australia.

[8] Maher, M. L., Gu, N. and Li, F. (2001),Visualisation and object design in virtual architecture *in* J. S. Gero, S. Chase and M. Rosenman (eds), *CAADRIA2001*, Key Centre of Design Computing and Cognition, University of Sydney, pp. 39-50.

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

[10] Stiny G. and Gips J.: 1972, “Shape Grammars and the Generative Specification of Painting and Sculpture,” in C. V. Freiman, ed., *Information Processing 71*, North Holland, Amsterdam, pp. 1460-1465.

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