Ning Gu, Mary Lou Maher
Designing Adaptive Virtual Worlds

Ning Gu, Mary Lou Maher

Designing Adaptive Virtual Worlds

Managing Editor: Monika Michałowicz

Associate Editor: Davina Jackson

Language Editor: Andrew Kerber

DE GRUYTER
OPEN

Published by De Gruyter Open Ltd, Warsaw/Berlin

ISBN: 978-3-11-036765-2
e-ISBN: 978-3-11-036766-9

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.dnb.de.

Managing Editor: Monika Michałowicz
Associate Editor: Davina Jackson
Language Editor: Andrew Kerber

www.degruyteropen.com

Cover illustration: © Ning Gu and Noel Yaxley

Contents

Part I: **Introduction**

Part II: **Rule-based Place Design in 3D Virtual Worlds**

Part III: **An Adaptive Virtual Gallery**

Part IV: **The Future of Adaptive Virtual Worlds**

Part I: **Introduction**

1 Introduction

The boundary between our physical environment and the digital world is becoming increasingly blurred as we move more and more of our personal communication, health records, bank details, professional interactions, educational materials and records, and entertainment online. On one extreme, we have a physical world void of any digital content, an increasingly small part of the developed world, but still present. On the other extreme we have digital personas that exist only in virtual worlds such as Second Life[1]. These two extremes hardly characterize the complexity of the boundary between the two, but they allow us to think about what may be common in the extremes.

The boundary can be described as a sense of immersion or a focus of attention that is often crossed: for example, we may be interacting with a friend or colleague on Skype[2] or World of Warcraft[3] when someone walks into the physical room we are in and starts talking to us. The boundary can also be described in technical terms that distinguish the physical from the digital, however, these distinguishing features become less defining as new products are available that merge the two worlds, such as Google Glass[4] where the digital is superimposed on the physical world and Sifteo Cubes[5], where physical and gestural interaction affect digital content. This book explores one extreme, the intentional design of virtual worlds to accommodate our increasing presence in digital environments that extend our sense of place. This focus on place provides a basis for integrating our digital lives into our physical lives by building on our existence as physical beings. By focusing on the design of place in virtual worlds, we can better accommodate the myriad of technologies that lie between the extreme of a physical place void of digital content and a purely digital world void of physical existence.

1.1 Why Adaptive Virtual Worlds?

Virtual worlds provide immersive experiences in another world that is inhabited by other people represented as avatars. While this has been a popular arena for massive multiplayer online role-playing games[6] such as World of Warcraft, the various ventures into developing virtual worlds for education, commerce, collaboration

1 http://secondlife.com
2 http://skype.com
3 http://us.battle.net/wow
4 http://www.google.com/glass/start
5 http://sifteo.com
6 http://en.wikipedia.org/wiki/Massively_multiplayer_online_role-playing_game

and entertainment have had a difficult time achieving recognition as a widely-used technology. We do not attempt to address the issue of the viability of virtual worlds, but instead present a focus on the design of adaptive virtual worlds as a basis for the future of virtual worlds. We present design principles that are based on how virtual worlds are similar yet distinct from the physical world. A virtual world can be familiar in its reference to the physical world, but can be different because it is not restricted by the physics of the physical world. In particular, we focus on how virtual worlds can be designed to adapt to the needs of their inhabitants.

In light of the rapid proliferation of technologies that enhance our physical existence with smart phones, smart clothing, smart homes, smart buildings, we see an opportunity to extend the design of physical environments beyond the digital enhancement of physical devices and places, to the design of virtual worlds that can extend our physical world using the metaphor of place. The early history of virtual worlds has shown that these new places can provide many of the affordances of the physical world that may be too expensive or not accessible to a large percentage of the population. As computing devices, including desktop computers, laptops, tablets, and smart phones, are becoming available to a significant percentage of the population in both first- and third-world countries, this emphasis on designing adaptive virtual worlds provides a forum for extending the ability for people to have meaningful experiences such as online education. Some online education models include MOOCs[7], where the number of students in a single class be in the hundreds of thousands, and crowdsourcing initiatives, such as citizen science, where the number of participants can increase to millions. More generally, we are seeing the development of communities whose focus and interests are not based on their physical location, yet their personal existence and experiences are physical. By pursuing the design of adaptive virtual worlds, we hope to raise an awareness of the potential and importance of both the adaptive aspects and the need for design principles.

1.2 Why Design Places?

Designing places has a history significantly longer than the history of virtual worlds. We focus on the design of place in the context of virtual worlds in order to distinguish the content of this book from other books that focus on the design of the 3D models or the technology to support a virtual world. With a focus on place, we can merge the design principles that have evolved over a long history of architecture with the principles of new technology that make virtual worlds possible. Architectural design has often been considered to be essentially about place making for built environments.

7 http://en.wikipedia.org/wiki/Massive_open_online_course

Through designing places, this book aims to build on the rich knowledge base of architectural design and extend and adapt it for virtual worlds.

We take into consideration the differences between designing with physical materials and designing with digital content. An example of a design domain that has principles similar to designing virtual worlds is the design of web sites. We separate the representation of the content of the virtual world from its dynamic, adaptive, and interactive characteristics, allowing us to represent place as something that changes over time as needed. In the design of a web site, the content is specified within the HTML tags, where the tags define the categories of content. The layout and design style of the web site is described by the CSS rules, where categories of content are given specific rules for their appearance and location on the web page. The interactivity and dynamics of the web site are specified in JavaScript, Ajax, JQuery or other scripting languages for client-side processing and in PHP, Ruby, Python or other languages for server-side processing. These distinctions are similar to the design of virtual worlds, where the content, style, and interaction/dynamics are defined as the materials of design. The difference is that web sites use the metaphor of documents as the basis for design, and we use the metaphor of place in designing virtual worlds. In this book, we define the materials for designing virtual worlds in terms of the content or objects in the world as 3D models; the layout of a 3D place as layout rules; the design and arrangement of the objects in the place as object design rules; the circulation and accessibility of the place as navigation rules; the dynamics of the objects and place as interaction rules in a grammar; and the ability to be adaptive by wrapping an agent model around the grammar that senses and responds to the needs of the users in the virtual world.

1.3 Overview of This Book

This book is divided into four parts. Chapter 2 (the remainder of Part I) provides the background for the book, which briefly discusses the evolution of virtual worlds and virtual world research. This chapter also highlights the concepts of metaphor and place making for virtual worlds that are fundamental for our approach to designing and representing virtual worlds.

Part II of the book presents the research and theoretical foundation for designing adaptive virtual worlds. Chapter 3 illustrates the generative design grammar framework that provides organizing principles for developing grammars that addresses both the syntactic (visualization: layout and object design), and semantic (navigation and interaction) considerations when designing places in 3D virtual worlds. Chapter 4 describes the Generative Design Agent (GDA) model that is wrapped around the generative design grammars, providing the mechanisms to sense and change in 3D virtual worlds. Designing adaptive virtual places is then realized through the GDA's application of generative design grammars, adapting to the changing needs during real-time interactions in 3D virtual worlds.

Part III demonstrates the use and effectiveness of the generative design grammar framework and the GDA model for the design and realization of adaptive virtual worlds through the development of a specific generative design grammar (Chapter 5) in an adaptive virtual gallery scenario (Chapter 6). This demonstration also provides a context for exploring and discussing various practical issues in designing adaptive virtual worlds.

Part IV concludes the book by highlighting the future development of adaptive virtual worlds. This final chapter discusses various considerations for designing adaptive virtual places using design rules and agents in terms of technical environments, design generation and controls, design styles, and the impact on the built environments.

2 Virtual Worlds

Virtual worlds are places that exist entirely in networked environments in which people co-exist, communicate and interact through their avatars. These worlds are dynamic and interactive environments that support a broad range of social, entertainment, educational, and productive activities that are loosely based on activities in the physical world. A common metaphor for the design of these worlds is the concept of place. Most designs use the concept of place as the basis for designing the ambient environment and objects in the virtual world. Virtual worlds are designed for specific purposes and can support various activities online.

The design of places in virtual worlds draws on our experience and knowledge of architectural design in the physical world. The metaphor of place and reference to concepts from architectural design provide a consistent and familiar base for the people in the virtual world and also for the designers of virtual worlds. This metaphor and reference facilitates the interaction of virtual world users with the designed environments and with each other. Effectively, the physical world provides the inspiration for the design of virtual worlds. We are beginning to see the opposite effect as well, that virtual world designs are providing the inspiration for the design of physical places. The characteristics of virtual worlds that make them dynamic, adaptive, and interactive are especially relevant to contemporary and future place design in the physical world, serving as an ideal test bed for developing and maturing design fields such as intelligent environments, ubiquitous computing, pervasive computing, and interaction design.

2.1 Evolution of Virtual Worlds

Virtual worlds emerged from the 1990s as an environment that allows multiple users access to a single application in real time on the internet. Singhal and Zyda (1999) describe virtual worlds as software systems where multiple users connect from different geographical locations and interact with each other in real time. They also characterize virtual worlds with five common features: (1) a shared sense of location; (2) a shared sense of presence; (3) a shared sense of time; (4) online communication; and (5) interaction with the virtual environments. Highlighted by these features, virtual worlds are capable of providing multiple users with the ability to interact with each other, to share information, and to interact with the virtual environment by manipulating virtual world objects in the environment through immersive computer graphics. To further articulate the definition of virtual worlds, Singhal and Zyda also draw the differences between virtual worlds and other networked environments or tools emerged at the time: the ability to support multiple users differentiates virtual worlds from early virtual reality applications and game environments[8]. The ability to share and manipulate virtual world objects differentiates virtual worlds from traditional online communication tools such as online chat rooms and video conferencing systems. The ability to support real-time interactions differentiates virtual worlds from email and traditional web browsing. However, despite these definitions and characterizations, there is not yet a universal definition or standard form of virtual worlds. Virtual worlds as a topic for design and research is relatively new and still evolving. Different designs of virtual worlds are currently perceived as integral parts of the social media that support online communication and interaction.

The brief history of virtual worlds can be traced back to about the 1980s. The term "cyberspace" was first introduced in the science fiction novel *Neuromancer* (Gibson, 1984). In this novel, Gibson (1984) describes cyberspace as "a consensual hallucination experienced daily by billions of legitimate operators, in every nation… a graphical representation of data abstracted from the banks of every computer in the human system… lines of light ranged in the nonspace of the mind, clusters and constellations of data. Like city lights…". The main contribution of Gibson's novel, as described by Whittaker (2004), was that it gave "expression to the emerging technologies" at the time, such as personal computers, the internet, computer graphics, virtual reality and other digital forms of information. Thereafter, especially in the 1990s, the term "cyberspace" attracted the attention of the general public, and became extremely popular and influential among design practitioners and researchers in different areas. To a certain extent, cyberspace seems to have become an express tunnel that links together the present and future and which provides a new horizon for opportunities, innovation and imagination. It has also had a continuing impact on a wide range of domains such as architecture and design, computer science, education, e-commerce, social science and so on. This newly emergent area does not develop alone. It has since been extended to include many sub-domains, such as communication and collaboration, design representation, human computer interaction, artificial intelligence, computer graphics, virtual and augmented reality and other networked technologies.

The early developments in the design of virtual worlds can be categorized to have taken two major paths: (1) the conceptual development of the design and purpose of virtual worlds; and (2) the technical development of the design and implementation of virtual worlds. In the conceptual development of virtual worlds, design practitioners and researchers explore the possibilities of virtual worlds, illustrate the future of virtual worlds, and study the impact of virtual worlds on existing design theories and practices. For example, Benedikt (1991) collects a series of influential writings by designers, artists, novelists, engineers, businessmen and academics to predict and illustrate the future

8 New generations of virtual reality applications and game environments can now support multiple users.

of cyberspace from different perspectives where everything seems possible. These writings also outline the dramatic changes in the physical world and our future daily life due to the influence of cyberspace. Woolley (1993) discusses the emergence of virtual worlds, which changes public reality through virtual reality and artificial reality. Anders (1998) presents theories and examples that use space as a cognitive tool for managing our daily activities in the physical world, and how these concepts may be extended to cyberspace. Wertheim (1999) follows the history of the western conception of space from the middle ages to the information ages, and critically accesses the cyberspace and cyber culture. Now in the 21st Century, when we look back and re-examine some of the concepts and predictions, virtual worlds are indeed challenging and gradually changing the traditional forms of communication, education, entertainment and business. The physical world, however, has not been radically overtaken by the emergence of virtual worlds as some have predicted. Virtual worlds co-exist with the physical world and supplement the physical world. One current example aiming for the seamless blend of the virtual and the physical is the concept of ubiquitous computing initiated by Weiser (1991) and researched by many others nowadays (Abowd & Mynatt, 2000; Dourish, 2004). This concept has also been applied to the urban scale to consider the development of ubiquitous cities in the real context (Sang et al, 2008).

The second category of early development is the technical advances for the design and implementation, in which design practitioners and researchers work on the technical realization and building of virtual worlds. The early forms of virtual worlds emerged in the early 1990s, when the internet became more accessible. The origin of virtual worlds has served two purposes: (1) military simulation; for example, SIMNET (simulator networking) developed by the US Department of Defense that simulates battlefields for military training purposes; and (2) networked games; for example, Doom and Quake, 3D networked games first released in the 1990s by id Software[9], downloaded and shared by millions since their release. Beyond its origins of military simulation and networked games, virtual world design and implementation have been enriched and diverted to take on various forms for different purposes. The internet has accommodated many different technologies supporting the early text-based virtual worlds, graphical virtual worlds and 3D virtual worlds. The purposes that virtual worlds serve have expanded to include social communication, education, design collaboration, e-commerce, and others. The main literature on virtual world design and implementation, as suggested by Maher (1999), can address one or more of the following issues: (1) implementation – technologies for realizing virtual worlds; (2) representation – a consideration of the digital representation and management of various components of virtual worlds; (3) interface – the types of interface provided to people for accessing virtual worlds, interacting with the environments, and interacting with each other.

9 http://www.idsoftware.com

At the implementation level, 3D models have become the dominant form for visualizing virtual worlds following some early attempts at using text-based and graphical representations. The introduction of the Virtual Reality Markup Language (VRML) in 1994 led to the first 3D web browser. Virtual Reality (VR) technologies, which had been used for simulation, then became the main means for realizing virtual worlds (Morgan & Zampi, 1995). This marked the beginning of the 3D era for virtual worlds.

At the representation level, the main interest concerns the use of metaphors; for example, the study on how text-based virtual worlds, mainly MOOs, are represented and designed (Cicognani & Maher, 1998); and its extensions from the linguistic characterizations to include graphical and spatial characterizations (Maher et al, 1999; 2000; 2001). Using the place metaphor, the latter studies draw an analogy to the built environment, and a coherent hierarchy of architectural elements, such as buildings, rooms and objects are developed for representing object-oriented virtual worlds. More recently, computational models have been used for representing virtual worlds in order to integrate artificial intelligence to virtual worlds. These models include examples such as cognitive agent models (Maher & Gero, 2002; Smith et al, 2003; Maher et al, 2004) and generative design algorithms (Gu & Maher, 2003; Muller et al, 2006). At the interface level, the input and output devices of virtual worlds have been explored. Research on human computer interaction and ubiquitous computing has provided new technologies for enhancement in these areas, for example, various mobile and handheld devices for more flexible access to virtual worlds, and more affordable augmented and mixed reality solutions such as Google Glass for better immersive display. However, the most common interface for accessing virtual worlds is still a desktop or laptop computer.

2.2 Design Metaphors

Through the use of metaphor, concepts in one domain can be understood, expressed and experienced in terms of another (Lakoff & Johnson, 1980). The early use of metaphors can be seen in ancient literatures, and have been widely applied and recognized in poetry and novels. Metaphors are not simply about language; in fact, the human conceptual system is largely metaphorical (Lakoff & Johnson, 1980; MacCormac, 1985; Erickson, 1990). In other words, the use of metaphors shapes our human understanding, thus greatly affects how we think and what we do. The essential aspects of metaphor are very well highlighted by MacCormac (1995): "Without metaphors, to describe and explain the unknown would become impossible". This is especially evident in contemporary societies with the rapid development of new technologies, which appear to have been unfamiliar to us initially. Therefore, we extend the familiar concepts to understand, apply and further develop these new technologies. There are many examples of such in our daily life; for example, telegraph, television, digital television, email, e-commerce, and e-learning. Lakoff and Johnson (1980) highlight the power of metaphors. Metaphors pervade the human conceptual system. The use of metaphors enables us to grasp many concepts that are either abstract or not clearly delineated in our experiences. It is by means of other concepts that we better understand in terms of familiar structure, spatial orientation, objects, and so on. Metaphors define reality, especially social realities, by providing a coherent network of entailments that highlight certain aspects of our experiences but hide others. The acceptance of the metaphor forces us to focus only on those highlighted aspects of our experiences, and leads us to believe that the entailments of the metaphor are true. The created reality can then be used as guidance for future actions, and these actions, in turn, reinforce the power of metaphors to make experiences more coherent. Some new metaphors can also provide new meanings. Unlike conventional metaphors that structure the general conceptual system of our culture, these new metaphors are imaginative and creative and exist outside the conventional conceptual system. Such metaphors are capable of giving us a new understanding of our experience.

In the design domain, the use of metaphors allows practitioners and researchers to adopt familiar design concepts to make references to other design knowledge, or to simply be inspired. Different metaphors have been applied to compose music, create artworks, design artefacts and generate systems in areas such as music and creative art, architectural design, engineering design and design computing. For example, in photography, Radice (1998) documented a series of pictures by the Italian designer and photographer Ettore Sottsass taken between 1972 and 1973. These pictures are largely composed of artificial objects with the natural desert surroundings of Spain. Each artificial object was carefully selected and placed in the scene. These objects went beyond their original meaning in industrial civilization, and were used as metaphors by the artist which he illustrated through the interplay with the natural landscape. For example, a door does not simply define a gateway to exit one place and to enter another place; instead, through the door we "enter into darkness" or "are meeting our love".

In computer science, the use of metaphors is also very common. Since the nature of our conceptual system is metaphorical, it is not difficult to understand that the use of appropriate metaphors plays an important part in assisting our interactions with the environments we inhabit. This is also the case when we interact with computer environments (Erickson, 1990; Marcus, 1998; Hsu & Schwen, 2003). Stefik (1996) provides a historical review of the metaphors we use to describe and design computers and computer networks. They are "giant brains", "information superhighway" and "multiple metaphors". The main metaphor applied to computers in the 1950s was the giant brain, however, this metaphor did not predict the broader future of computers. Computers did not become bigger in order to become more powerful. In the 1980s, the information superhighway metaphor was applied to describe the high-speed connectivity of the computer networks. This metaphor is useful for thinking about the flow of digital information, but does not provide adequate insight for illustrating how our future lives would be affected by computers. Stefik presents multiple metaphors to extend and enrich the concept of the information superhighway by thinking about emerging information infrastructure and exploring new possibilities. Stefik explores four of these multiple metaphors: digital library (for thinking about information storage and shared knowledge), electronic mail (for thinking about digital communication), electronic marketplace (for thinking about digital properties and e-commerce), and digital worlds (for thinking about virtual community and its supporting infrastructure such as virtual reality, telepresence and so on). Each metaphor addresses one particular aspect, which together provide a richer range of meanings and possibilities for the information superhighway.

From these multiple metaphors, it is evident that new technologies and experiences make reference to various familiar concepts in the physical world. Similarly, to extend these ideas further, we can consider designing virtual worlds in terms of designing built environments in the physical world (which is relatively more familiar to us). This leads to the use of the place metaphor for designing virtual worlds.

2.3 Place Making in Virtual Worlds

The image of architecture has always been defined and perceived as the art of places. In the physical world, places differ from spaces by including social and cultural values, in addition to spatial configurations (Kalay, 2004). This distinction of place nicely highlights the key elements in the formation of places. They are the spatial environment, the people who inhabit the environment, and the interactions between these two. This understanding is echoed by many other scholars. Norberg-Schulz (2000) uses the word "totality" to describe the art of places, which by principle deals with "the experience of living". Steele (1981) illustrates a diagram to define the concept of a sense of place: "Setting + Persons = Sense of Place". The term "setting" in the diagram refers to the "surrounding" (spatial setting) and "context" (social setting). The persons are psychological factors. Relph (1976) claims that places at all scales are "whole entities" that synthesize natural and artificial objects, functions and activities, and meanings given by intentions.

By using the metaphor of place we can design virtual worlds by place making rather than by page, document or database making. The underlying rationale of using the place metaphor is based on an assumption that, because to a large extent our social and cultural behaviors are organized around spatial elements of the physical world, we can carry over these patterns of behaviors to virtual worlds by designing them to have the same potentials for conception and interaction that the physical world exhibits (Kalay & Marx, 2001; Champion & Dave, 2002; Kalay, 2004). The patterns of behavior we learn in the physical world therefore become useful in virtual worlds. By structuring virtual worlds in a way that allows us to apply these learned traits from the physical world, we can reduce the cognitive effort needed to inhabit the worlds. By adopting the metaphor of place, designing virtual worlds as a relatively new area is able to make analogical references to place design that has been developed for centuries, rich with its own theories and practice. The analogy provides a base to understand and further extend the use of these networked environments. On the other hand, the emergence of virtual worlds also has had a reverse impact and created new dimensions for exploring place design. Such examples include liquid architecture (Novak, 1991), hyper architecture (Puglisi, 1999), information architecture (Schmitt, 1999) and interactive architecture (Fox & Kemp, 2009). Virtual worlds made up of computing entities that can be directly scripted and programmed can also inform the latest digital design development such as the scripting cultures (Burry, 2011).

However, place making in virtual worlds is, after all, different from place making in the physical world. Firstly, virtual worlds are networked digital environments. Virtual worlds can be experienced only via computers. Secondly, designing virtual worlds can go beyond the principles of physical place design to further explore the potentials specific to virtual worlds, as these worlds do not need to obey the law of physics and other than input and display devices, comprise pure computing entities. With these considerations in mind, designing virtual worlds, therefore, is not authentic place making; rather, it is a loose sense of place making that adopts certain relevant design concepts and knowledge from the built environments for the virtual context. Based on Kalay's (2004) criteria for virtual place making, we summarize the following characteristics of place making in virtual worlds.

- Functional virtual places – places provide ambient environments for certain intended activities online, which offers the reason or purpose for being there;
- A sense of location – places provide relative locations, and locations create a context for the intended activities to occur. A sense of location helps to recall our traces in the virtual worlds, and these traces help us to differentiate one place from another;
- A sense of presence – places involve some kind of engagement with objects and people. Through these interactions, a sense of presence is provided; and
- Uniqueness of virtual places – virtual worlds afford a variety of experiences different from our physical experiences; for example, in virtual worlds we have unique ways of navigation from place to place using hyperlinks, and the ambient environment and objects can be interactive and even proactive.

2.4 Design Platforms and Examples

There are far fewer examples of designed virtual worlds compared to designed places in the physical world. This is reasonable considering the relatively short history of virtual worlds. During the past two decades, technologies and tools for designing virtual worlds have undergone a series of evolutionary changes from the early text-based virtual worlds such as MUDs (Multi-User Domains) and MOOs (MUD, Object-Oriented) to the current 3D virtual worlds.

In the text-based virtual worlds, words are the only matter because the creation of the worlds depends almost exclusively on the use of words. The virtual worlds and their components are described using only text. People connect to a shared networked environment to interact with the environment and each other by using textual commands. MUDs are widely recognized as the first generation of shared networked environments. This technology was originally developed as a networked place for the role-playing game Dungeons and Dragons, which was then extended into MOOs to support other online games, and to further service the virtual communities in general activities, such as social communication and online learning.

In 1990, Stephen White developed the first MOO server. MOOs enable an easier creation of virtual worlds through object-oriented programming[10]. The use of object-oriented languages for designing virtual worlds is influential, as many of the current virtual worlds are object-oriented. Subsequent development of MOOs adds a graphical dimension by using a web browser window for visualizing the virtual world using digital images – each MOO object is often visualized with a graphical icon and some MOOs also provide a 2D layout plan to organize these icons and a 2D map to direct users.

In text-based virtual worlds, users rely largely on spatial reasoning and mental models to perceive and process spatial information of the virtual worlds. In 2D graphical virtual worlds, the textual descriptions are enhanced by the use of digital images which provide virtual worlds with a new graphical dimension. However, 2D images do not allow us to fully exploit the spatial organization potential of the virtual worlds. After these early attempts and the development of personal computers that could process and display 3D models, the next step in the implementation of virtual worlds was to search for 3D alternatives that could mimic the built environments and support 3D spatial experiences.

Nowadays, most virtual worlds are visualized using 3D models. Platforms for designing 3D virtual worlds include Active Worlds[11], Second Life, OpenSim[12], Open Cobalt[13], 3DVIA Studio[14], and others that have been developed from or inspired by gaming engines such as Quake[15] and World of Warcraft. The development of 3D models as virtual worlds is a major focus for most of these design platforms. This focus leads to a strong emphasis on the visual aspect of the designs. In these virtual worlds, virtual world objects often appear as place-element-like models. Selected

10 http://www.moo.mud.org/moo-faq
11 http://www.activeworlds.com
12 http://opensimulator.org
13 http://www.opencobalt.org/
14 http://www.3dvia.com/studio/
15 http://www.idsoftware.com/games/quake

objects can have scripted behaviors to support predefined actions for interactions in the virtual world. Virtual world users are represented as avatars, the animated characters. Avatars communicate with each other via in-world communication and interact with the environments by activating behaviors of various objects.

We briefly review several platforms for designing 3D virtual worlds that support these typical features and show a typical place design for each platform.

Active Worlds (AW) is one of the earliest platforms for designing and operating 3D virtual worlds. Using AW, virtual worlds are designed and implemented based on the AW object library. The object library provides a list of 3D models that simulate place elements. Users are able to build virtual places using these models. The object library can be expanded by using external 3D modeling and translation tools. Virtual world objects can have behaviors by using AW triggers and commands, a simplified scripting language. The interactions in AW can also be further extended using the AW Software Development Kit (SDK). Figure 2.1 shows a virtual cinema designed and implemented using AW.

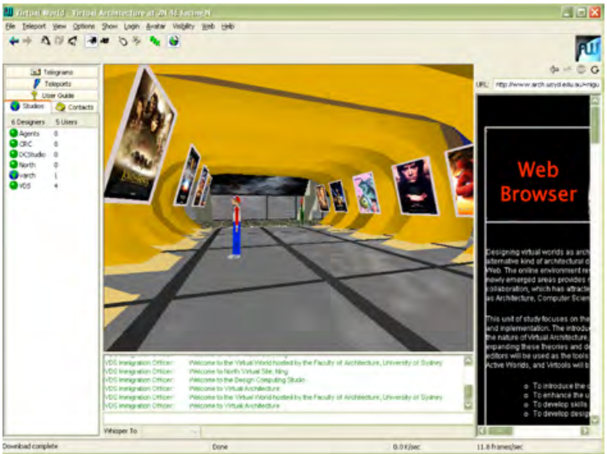


Figure 2.1: A virtual cinema designed in the Active Worlds University of Sydney universe, by the students.

Second Life (SL), as one of the most popular 3D virtual world platforms, is developed with a strong focus on virtual communities in supporting a diverse range of online activities. Besides gaming, social communication and e-learning activities, SL is particularly known for its e-commerce activities including trading for virtual estates and properties. SL provides basic in-world modelling tools for designing and implementing virtual worlds by direct manipulation of geometric primitives. Behaviors of virtual world objects and avatars can be controlled and customized using Linden Scripting Language (LSL). Figure 2.2 shows a public place in SL.

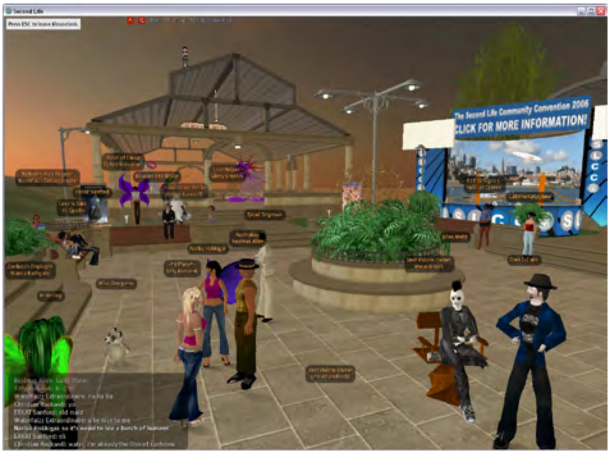


Figure 2.2: A public place in Second Life.

Open Cobalt emphasizes the use of virtual worlds as workplaces for research and other professional collaboration. Open Cobalt enables users to access and share not only objects within the virtual world but also other remote software applications, through the virtual world. As shown in Figure 2.3, although Open Cobalt supports the creation of virtual places, the focus of Open Cobalt is more on its capabilities for supporting real-time online collaboration via shared objects and applications.

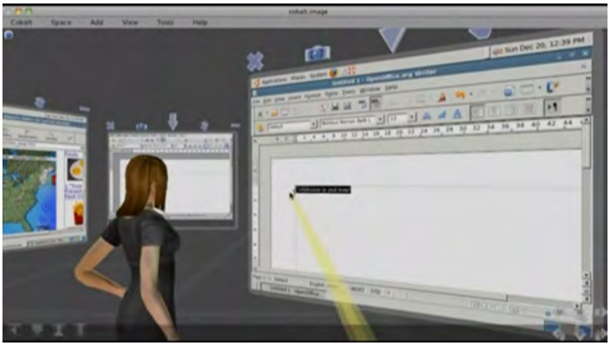


Figure 2.3: Collaborative document sharing and editing in a virtual world created using Open Cobalt (image taken from Open Cobalt web site: http://www.opencobalt.org).

3DVIA Studio (3S) develops 3D virtual worlds as different virtual and augmented reality applications for different purposes, ranging from online gaming to real-time product and building visualization. 3S provides 3D model editing features and can support the import of large industry data sets. It also serves as a graphical programming system for designing interactions in 3D virtual worlds. Figure 2.4 shows a popular social game played on Facebook and developed using the platform. 3S is robust for designing highly interactive and customized 3D virtual worlds, but can be technically demanding for general designers as many virtual world features need to be developed from scratch.

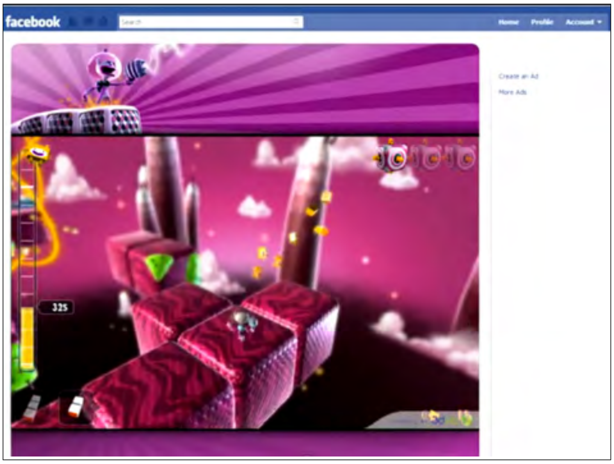


Figure 2.4: Billions Save Them All, a 3D social game played on Facebook and developed using 3DVIA Studio (http:// www.3dvia.com/studio/gallery/billions).

While we are seeing a rise in the number of virtual world platforms and opportunities to build and experience places in virtual worlds, most 3D virtual worlds share the following two common characteristics: (1) there is a general lack of formal approaches to understanding and designing 3D virtual worlds as places; and (2) the current process of virtual world design and implementation can be cumbersome, and relies heavily on the designers' ability to not only conceive of the place as a set of 3D geometric models, but also program and script the behaviors of each object. There are numerous books that explain how to build and program in virtual worlds (eg. Weber et al, 2007), but few describe the design of places as an intentional design activity.

This book contributes to the theoretical foundation of designing virtual worlds by introducing a formal computational approach to designing places in virtual worlds based on design grammars and computational design agents. Virtual world designers, researchers and online communities will be guided to understand the design principles of virtual worlds as places, examples of rules for designing places for specific purposes in virtual worlds, and principles for making virtual worlds adaptive through the use of computational agents. The book also explores the impact and future of these dynamic, adaptive, and interactive virtual worlds for networked games, social media places, collective design intelligence and interactive architecture.

References

Abowd, G.D. and Mynatt, E.D. (2000), Charting Past, Present, and Future Research in Ubiquitous Computing, *ACM Transactions on Computer Human Interaction*, 7(1), pp. 29-58.

Anders, P. (1998), *Envisioning Cyberspace: Designing 3D Electronic Spaces*, McGraw-Hill, New York.

Benedikt, M. (ed) (1991), *Cyberspace: First Steps*, MIT Press, Cambridge, MA.

Burry, M. (2011), *Scripting Cultures: Architectural Design and Programming*, Wiley, London.

Champion, E. and Dave, B. (2002), Where Is This Place?, *Proceedings of ACADIA 2002*, California State Polytechnic University, pp. 87-97.

Cicognani, A. and Maher, M.L. (1998), Two Approaches to Designing Virtual Worlds, *Proceedings of Design Computing on the Net 98*, *International Journal of Design Computing*, 1 (online).

Dourish, P. (2004), *Where the Action Is: The Foundations of Embodied Interaction*, MIT Press, Cambridge, MA.

Erickson, T.D. (1990), Working with Interface Metaphors, in B. Laurel (ed), *The Art of Human-Computer Interface Design*, Addison-Wesley, Reading, MA, pp. 65-73.

Fox, M. and Kemp, M. (2009), *Interactive Architecture*, Princeton Architectural Press, New York.

Gibson, W. (1984), *Neuromancer*, Ace Books, New York.

Gu, N. and Maher, M.L. (2003), A Grammar for the Dynamic Design of Virtual Architecture Using Rational Agents, *International Journal of Architectural Computing*, 4(1), pp. 489-501.

Hsu Y-C, and Schwen, T.M. (2003), The Effects of Structural Cues from Multiple Metaphors on Computer Users' Information Search Performance, *International Journal of Human-Computer Studies*, 58, pp. 39-55.

Kalay, Y.E. (2004), *Architecture's New Media: Principles, Theories, and Methods of Computer-Aided Design*, MIT Press, MA.

Kalay, Y.E. and Marx, J. (2001), Architecture and the Internet: Designing Places in Cyberspace, *Proceedings of ACADIA 2001*, Buffalo, New York, pp. 230-240.

Lakoff, G. and Johnson, M. (1980), *Metaphors We Live By*, University of Chicago Press, Chicago.

MacCormac, E.R. (1985), *A Cognitive Theory of Metaphor*, MIT Press, Cambridge, MA.

Maher, M.L. (1999), Designing the Virtual Campus as a Virtual World, *Proceedings of Computer Supported Collaborative Learning 99*, Stanford University, pp. 376-382.

Maher, M.L. and Gero J.S. (2002), Agent Models of Virtual Worlds, *Proceedings of ACADIA 2002*, California State Polytechnic University, pp. 127-138.

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), *Proceedings of CAADRIA 2001*, University of Sydney, Australia, pp. 39-50.

Maher, M.L., Liew, P-S, Gu, N. and Ding, L. (2004), An Agent Approach to Supporting Collaborative Design in 3D Virtual Worlds, *Automation in Construction*, 14(2), pp. 189-195.

Maher, M.L., Simoff, S., Gu, N. and Lau, K.H. (2000), Designing Virtual Architecture, *Proceedings of CAADRIA 2000*, Singapore, pp. 481-490.

Maher, M.L., Simoff, S., Gu, N., Lau, K.H. (1999), Two Approaches to a Virtual Design Office, *Proceedings of Design Computing on the Net 99*, *International Journal of Design Computing*, 2 (online).

Marcus, A. (1998), Metaphor Designs in User Interfaces, *Journal of Computer Documentation*, 22(2), pp. 43-57.

Morgan, C.L. and Zampi, G. (1995), *Virtual Architecture*, B.T. Batsford, London.

Muller, P., Wonka, P., Haegler, S., Ulmer, A. and Van Gool, L. (2006), Procedural Modeling of Buildings, *ACM Transactions on Graphics*, 25(3), pp. 614-623.

Norberg-Schulz, C. (2000), *Architecture: Presence, Language, Place*, Skira Editore, Milan.

Novak, M. (1991), Liquid Architecture in Cyberspace, in M. Benedikt (ed), *Cyberspace: First Steps*, MIT Press, Cambridge, MA, pp. 225-254.

Puglisi, L.P. (1999), *Hyper Architecture: Spaces in the Electronic Age*, Birkhäuser, Basel, Switzerland.

Radice, B. (1988), *Ettore Sottsass: Design Metaphors*, Rizzoli International Publications. New York.

Relph, E. (1976), *Place and Placelessness*, Pion Limited, London.

Sang, H.L., Jung, H.H., Yoon, T.L. and Tan, Y (2008), Towards Ubiquitous City: Concept, Planning, and Experiences in the Republic of Korea, in Tan, Y., Koray, V. and Scott, B. (eds.), *Knowledge-Based Urban Development: Planning and Applications in the Information Era*, IGI Global, Hershey, PA, pp. 148-169.

Schmitt, G. (1999), *Information Architecture: Basis and Future of CAAD*, Birkhäuser, Basel, Switzerland.

Singhal, S. and Zyda, M. (1999), Networked Virtual Environments: Design and Implementation, ACM Press, New York.

Smith, G., Maher, M.L. and Gero J.S. (2003), Designing 3D Virtual Worlds as a Society of Agents, *Proceedings of CAAD Futures 2003*, Kluwer Academic Publishers, the Netherlands, pp. 105-114.

Steele, F. (1981), *The Sense of Place*, CBI Publishing Company, US.

Stefik, M. (1996), *Internet Dream: Archetypes, Myths, and Metaphors*, MIT Press, Cambridge, MA.

Weber, A, Rufer-Bach, K. and Platel, R. (2007), *Creating Your World: The Official Guide to Advanced Content Creation for Second Life*, Wiley, London.

Weiser, M. (1991), The Computer for the 21st Century, *Scientific American*, 265(3), pp. 94-104.

Wertheim, M. (1999), *The Pearly Gates of Cyberspace: A History of Space from Dante to the Internet*, Doubleday, Sydney, Australia.

Whittaker, J. (2004), *The Cyberspace Handbook*, Routledge, London.

Woolley, B. (1993), *Virtual Worlds: A Journey in Hype and Hyperreality*, Penguin Books, London.

Part II: **Rule-based Place Design in 3D Virtual Worlds**

3 Generative Design Grammars

A generative design grammar is a set of rules that describe a design style. In this book we show how such a grammar can be developed to design places in 3D virtual worlds. By encoding the design style as a grammar, the rules can be executed to generate many different individual designs sharing the style, in contrast to the current practice of handcrafting each individual design. This chapter presents a framework that provides organizing principles for developing a grammar that addresses the essential considerations in designing places in virtual worlds. The framework specifies the general structure of the design grammar and its basic components: the design rules. Designers can develop their own grammars for designing places in virtual worlds by using our framework to develop rules that describe their own design style. The design and development of specific places in 3D virtual worlds are then realized through the application of the grammar at first to generate the initial design, and next to adapt the design during real-time interactions in virtual worlds, as described in the next chapter.

3.1 Grammars for Rule-based Design

Our concept and development of generative design grammars are inspired by the notion of shape grammars (Stiny, 2008, Stiny & Gips, 1972) – a classic example of how a grammar can capture the style of a design and be used to generate multiple designs within that style. The inspiration comes directly from shape grammars' suitability as a sound formalism for describing and generating designs in general. Over the last three decades, the theories and applications of shape grammars have been refined and tested in a wide range of design disciplines; for example, paintings and visual arts (Knight, 1994a), architectural design (Stiny & Mitchell, 1978; Duarte, 2005; Eloy & Duarte, 2011), product (furniture) design (Knight, 1994a), engineering design (Shea & Cagan, 1997), and design computing (Gu & Maher, 2003).

3.1.1 Notions of Shape Grammars

Knight (2000) summarizes that a shape grammar is a set of shape rules that can be applied in a step-by-step manner to generate a set or language of designs. A shape grammar is both descriptive and generative. The application of the shape rules generates designs, and the rules themselves are the descriptions of the forms of the generated designs. In general, applications of shape grammars have two main purposes. On one hand, shape grammars can be used as design tools to generate vast varieties of design languages. On the other hand, shape grammars as design analysis tools can be used both to analyze existing designs in order to better understand these designs, and to generate shape rules that produce these designs and other similar designs.

Stiny (1980) defines four basic components of a shape grammar:

– S: a finite set of shapes.
– L: a finite set of symbols.
– R: a finite set of shape rules.
– I: initial shape.

Each shape rule follows the form of $S_a \rightarrow S_b$. S_a and S_b are two different labeled shapes.

Shape grammar theories have developed over the years to include many extensions, such as parametric grammars (Tapia, 1992), color grammars (Knight, 1994b), description grammars (Stiny, 1981), structure grammars (Carlson et al, 1991), parallel grammars (Knight, 1999), and so on, to address different aspects of design.

3.1.2 Shape Grammars and Design Style

Shape grammars are able to apply simple shape rules to produce designs with rich descriptions. Shape grammars enable different designs that share a similar style to emerge by alternating the sequence of shape rule application. By analyzing existing designs of a specific style, shape grammars can formally describe this style and generate other designs that also share the style. Further, by incorporating additional devices, shape grammars are able to describe and generate new design languages in an extension to the original languages.

To develop a shape grammar to describe and generate a specific style involves the following steps (Knight 1994a):

– To define a vocabulary of shapes and a set of spatial relations that are common to the design instances of the style.
– To define shape rules that fix the occurrences of the spatial relations.
– To provide an initial shape to start the design process.

Following the above steps, a shape grammar can be developed to produce a specific design language. Further, through transformations using rule addition, rule deletion or rule change, this shape grammar can be transformed into a new shape grammar. In this way, a new design language emerges.

3.1.3 Design Constraints in Shape Grammar Application

One critical issue in the application of shape grammars for design is to develop a grammar that produces designs that meet the design goals or constraints. According to Knight (1999), there are two different approaches to achieve that objective. The first approach is to incorporate the constraints into the shape rules so that the generated designs meet the given goals. Knight (1998a) defines different types of shape grammars ranging from unrestricted, standard shape grammars with the least predictable outcomes, to simple, restricted ones with the most predictable outcomes. Producing a shape grammar with more predictable outcomes requires knowledge about constraints and goals while developing the shape grammar. Alternatively, an unconstrained shape grammar generates designs, after which a person or automated process evaluates the designs and selects the ones that meet the specific goals and satisfy the given constraints.

3.2 Generative Design Grammars for Rule-based Place Design in 3D Virtual Worlds

The design and implementation of a generative design grammar is the development of a generative design system – a computational approach for describing, generating and automating design in general. Besides design grammars, generative design systems have also been developed in other forms such as genetic algorithms (Holland, 1992; Ding & Gero, 2001), cellular automata (von Neumann, 1951; Wolfram, 2002) and swarm intelligence (Deneubourg, 1977; Payman, 2004). However, the nature of designing places in 3D virtual worlds as compositions of virtual world objects has made grammars an ideal design formalism for 3D virtual worlds.

We describe generative design grammars for virtual world designs by making a direct comparison to shape grammars for design in general. A shape grammar is a set of *shape rules*, which can be applied in a step-by-step manner to generate a set or language of designs (Knight, 2000). The nature of shape grammars is both descriptive and generative:

- *Shapes* as the basic components of shape rules, which can be in the form of points, lines, planes, volumes or any combination of the above, are elements of the designs that a shape grammar generate.
- *Rules* as the basic mechanisms for generating designs via shape operations and spatial transformations.

Similarly, a generative design grammar is a set of *design rules* that can be applied in a step-by-step manner to generate a set or language of place designs specifically for places in 3D virtual worlds. Inheriting the descriptive and generative nature of shape grammars:

- A generative design grammar describes place designs in 3D virtual worlds where the basic components of its design rules are *virtual world objects*.
- Place designs in 3D virtual worlds are generated by *operations and transformations* on virtual world objects that result in compositions that make sense in a virtual world.

These descriptive and generative qualities of design grammars serve the purposes of rule-based place design in 3D virtual worlds by formally specifying the components and processes relevant to designing in virtual worlds.

3.2.1 A Place Design in 3D Virtual Worlds as "Objects in Relations"

A design generated by a shape grammar is viewed as "elements in relations" (Stiny, 1999; 1990). To apply a shape grammar for design generation is basically to identify the "elements" of design, and define and alter (add, subtract or replace) the "relations" among the "elements" via shape rule applications. In this manner, shape grammars can generate complex designs based on simple design elements. This view of design is consistent with the object-oriented nature of virtual worlds. A place design in 3D virtual worlds can be viewed as "objects in relations".

3D virtual worlds are typical object-oriented systems. Examples of commercial platforms for 3D virtual worlds where the basic components for design and building are 3D objects include the majority of platforms such as Active Worlds, Second Life, and There[1]. In these virtual worlds, a design is constructed through the placement and configuration of virtual world objects. Depending on the object definition and classification of the virtual world, an object in a 3D virtual world may refer to a whole virtual place – a "container" object; for example, a virtual gallery or a virtual meeting room. A virtual world object may also refer to a component or an entity in a virtual place, or form a part of the place; for example, a digital picture in a virtual gallery or a wall of the virtual meeting room. Each virtual world object can have an appearance of a 3D geometric model in the virtual world, and together these models provide the visualization of the ambient environment. The virtual world objects then can be configured via scripts or codes to have certain behaviors that enable people to interact with the place and with each other. Therefore, a place design in 3D virtual worlds essentially comprises various virtual world objects that visually and functionally support intended human activities (Gu & Maher, 2003):

1 http://www.there.com

- Visually/spatially, via the use of the place metaphor, the 3D models are composed to form an ambient environment where people can inhabit and the intended activities can take place online.
- Functionally, selected virtual world objects are ascribed with behaviors accordingly to support the intended activities online. Therefore, people's interactions with the virtual world and among each other become possible.

Similar to the way shape grammars describe and generate designs in general, when designing places in 3D virtual worlds, generative design grammars describe virtual places in terms of virtual world objects and their relations, in the forms of design rules. The rules are applied to generate place designs in 3D virtual worlds in terms of the compositions of virtual world objects. This compositional characteristic of design generation makes generative design grammars an ideal design formalism for place design in 3D virtual worlds.

3.2.2 Design Phases of 3D Virtual Worlds

The view of virtual worlds as functional places that support an extended range of activities online provides a common ground for designing virtual worlds. This common ground highlights two key issues: activities and metaphor. Firstly, virtual worlds exist for certain purposes supporting various professional or social activities. Secondly, virtual world designs apply the metaphor of place. Based on this understanding, designing virtual worlds can be divided into the following four phases:

- Layout: the layout of the virtual place defines how areas are related to each other spatially in a way that defines and accommodates specific categories of intended activities in the 3D virtual world.
- Objects: each place is then configured with a number of specific virtual world objects, such as walls and floors that provide visual boundaries of the different areas in the place and more generally, objects such as the information desk or paintings that provide visual cues for supporting the intended activities.
- Navigation: navigation in virtual worlds can be facilitated with way finding aids and hyperlinks for assisting the users' movements from one place to another.
- Interactions: this part of the design process ascribes certain scripted behaviors to selected virtual world objects so that the users can interact with the objects and with each other by triggering those behaviors.

Through these four design phases, places can be generated in 3D virtual worlds in terms of visualization design (place layout and object design), navigation design and interaction design. These are the four inseparable design phases to provide an integral structure of a virtual place. Due to the use of the place metaphor, some design characterizations of the generated virtual places can find similarities to places in the physical world, for example, different layouts of virtual places and different forms of virtual world objects. However, virtual world designs also have many characteristics that are different from their physical counterparts, especially in terms of navigation and interaction. After all, virtual worlds are networked environments that do not need to be strictly constrained by the metaphor. Designing virtual worlds can go beyond the principles of place design in the physical world.

3.2.3 Addressing Design Requirements of 3D Virtual Worlds

Generative design grammars are able to address both visual/spatial requirements as well as non-visual/spatial requirements for designing places in 3D virtual worlds. Virtual world objects are visualized using 3D models. Because of the use of the place metaphor, these 3D models often depict place or place-like forms. The design requirements that are related to the visual/spatial aspect of virtual worlds are analogous to those of architectural design. There are many successful shape grammar examples of places and designs in the physical world; for example, the Palladian grammar (Stiny & Mitchell, 1978), the Mughul Gardens grammar (Stiny & Mitchell, 1980), the Prairie Houses grammar (Koning & Eizenberg, 1981) and the Siza Houses grammar (Duarte, 1999) that have been developed to generate and/or analyze different architectural examples. Inheriting the capabilities of design description and design generation from shape grammars, generative design grammars are capable of addressing similar kinds of visual/spatial requirements in designing places for 3D virtual worlds.

The design requirements that are related to the functional aspect of virtual worlds are often non-visual/spatial requirements. They are about ascribing scripted behaviors to selected virtual world objects to support the intended activities. For example, a simple behavior might be to display a digital image in a virtual gallery when a virtual canvas is “touched” by a visitor. Unlike physical artifacts, the visualization of a virtual world object and the behaviors of the object in the virtual world are artificially ascribed and associated rather than causally or physically related. The coupling of the two can be determined purely based on the designer’s preferences, although they often make a reference to the adopted place metaphor to maintain a sense of design consistency. Similar kinds of functional problems have been addressed using shape grammars in design. Although shape grammars are spatial, Stiny (1981) demonstrates the use of *description functions* to address the composition of designs in other terms that are non-spatial; for example, those related to the functions, purposes, uses and meanings of the designs. Knight (1999) further suggests the linkage of an original shape grammar with a *parallel description grammar* to generate designs (in terms of the composition of shapes) and other non-spatial descriptions in parallel. More recent developments such as discursive grammar (Duarte, 2005) and transformation grammar (Eloy & Duarte, 2011) also aim to address the semantic and syntactic issues in design. Similarly, generative design grammars are more than visual/spatial grammars. We formalize the relevant non-visual/spatial considerations of place design for virtual worlds in our framework by grouping the rules into four different categories to be applied in the four design phases of 3D virtual worlds accordingly, in order to explicitly address both visual/spatial considerations as well as non-visual/spatial considerations.

3.2.4 Capturing Stylistic Characterizations of Virtual Places

A specific style is exemplified when several designs "each create a similar impression" (Stiny & Mitchell, 1978). As discussed above, place design in 3D virtual worlds can be considered in terms of visualization design (place layout and object design), navigation design and interaction design. The results of these four design phases provide an integral set of stylistic characterizations for distinguishing places in 3D virtual worlds. Compared to many novice designs, virtual worlds designed with a specific style in mind will achieve better consistency, making it easier for users to get oriented and interact in the virtual world.

A shape grammar is capable of generating design instances that belong to an existing language of design as well as defining a new language of design. One might not instantly recognize the new design styles defined by shape grammars like the kindergarten grammar (Stiny, 1980); however, the applications of shape grammars have provided examples that analyze many well-known styles of painting, furniture design and architectural design, as well as generate design instances from these well-known styles, or even extend the styles. March and Stiny (1985) use "syntax" and "semantics" as the two major factors to distinguish designs from one another. "Syntax" determines how the shapes are composed to represent the designs. "Semantics" describe the designs in terms of functions, purposes, uses or meanings other than shapes. Design styles that are described and generated using shape grammars are considered primarily in terms of these two factors. The studies of shape grammars address these two factors through the original shape grammar formalism and the use of *description functions* (Stiny, 1981).

In summary, generative design grammars describe and generate languages of design that capture specific stylistic characterizations. A generative design grammar has different categories of design rules and the rules are applied at different stages to address both visual/spatial and non-visual/spatial considerations. The design rules that address visual/spatial problems are syntactic rules, and the design rules that address non-visual/spatial problems are semantic rules. We describe categories of design rules to specify place design in 3D virtual worlds in terms of syntax or visualization: place layout and object design, and semantics: navigation and interaction.

3.3 Generative Design Grammar Framework

Our generative design grammar framework provides guidelines and strategies for developing generative design grammars. The framework outlines the general structure of a generative design grammar and the general structure of its basic components: design rules. Following the structures suggested in the framework and integrating with different design and domain knowledge, designers will be able to develop their own design grammars for designing places in 3D virtual worlds.

A generative design grammar G is comprised of design rules R, an initial design D_i, and a final state of the design D_f.

$$G = \{R, D_i, D_f\} \tag{3.1}$$

The basic components of a generative design grammar are design rules R. The general structure of our grammar for virtual place design comprises four sets of design rules: layout rules R_a, object design rules R_b, navigation rules R_c, and interaction rules R_d.

$$R = \{R_a, R_b, R_c, R_d\} \tag{3.2}$$

This general structure is determined by four design phases of place design in virtual worlds:
- To develop the *layout* of the place in the virtual world where each sub-area of the place has a purpose that accommodates certain intended activities online.
- To configure the virtual place with designed virtual world *objects* that provide visual boundaries of the place and visual cues for supporting the intended activities.
- To specify *navigation* methods that use way finding aids and hyperlinks for assisting the movements of the users' avatars between different places in 3D virtual worlds.
- To design and activate *interactions* by ascribing scripted behaviors to selected virtual world objects in the virtual place so that people can interact with the virtual world and with each other.

The four set of design rules: layout rules, object design rules, navigation rules and interaction rules directly correspond to the above four design phases of virtual worlds. The generative design grammar framework is illustrated in Figure 3.1. The sequence of applying the design rules follows the order of layout rules, object design rules, navigation rules and finally interaction rules.

Different generative design grammars can be developed by following this general structure to design places for 3D virtual worlds when defining specific rules that encode the design and domain knowledge for different styles and purposes. The stylistic characterizations of virtual world designs, in terms of the syntax and the semantics, are defined accordingly in these four sets of design rules.

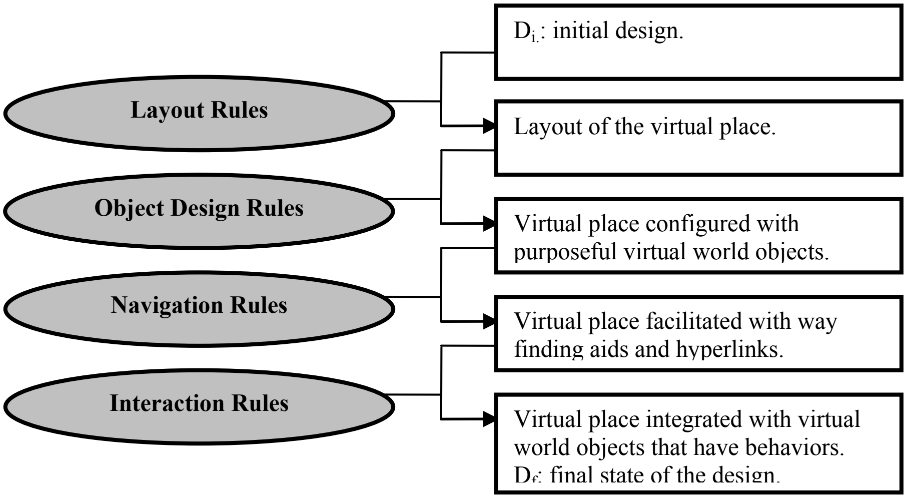


Figure 3.1: The generative design grammar framework.

3.4 General Structure of Design Rules

The basic components of a generative design grammar are design rules. The general structure of design rules is similar to the general structure of shape rules. In shape grammars, a shape rule can be defined as:

$$LHS \rightarrow RHS \tag{3.3}$$

which specifies that when the left-hand-side shape (LHS) is recognized in the design, it will be replaced by the right-hand-side shape (RHS). The replacement of shapes is usually applied under a set of shape operations or spatial transformations. The shapes are labelled (the use of spatial labels and state labels) for controlling the shape rule applications.

Similar to formula 3.3, a design rule of a generative design grammar is defined as:

$$LHO + sL \rightarrow RHO \tag{3.4}$$

which specifies that when the left-hand-side object (LHO) is recognized in the 3D virtual world, and the state labels sL are matched, the LHO will be replaced by the right-hand-side object (RHO). The term “object” used here can refer to a virtual world object, a set of virtual world objects or virtual world object properties.

The general structure of design rules implies the following two aspects:

- State labels are singled out and expressed explicitly as sL in the rule structure. The use of state labels is essential to the application of generative design grammars as they direct the application to ensure that the generated virtual world design satisfies the given design requirements. Each design rule is associated with certain state labels representing specific design contexts. In order for a design rule to be applied, a virtual world object, a set of virtual world objects or virtual world object properties need to be recognized in the 3D virtual world that match the LHO of the design rule, and the design context as represented by the sL of the design rule needs to be relevant to the current design needs, as interpreted by the designers or computational agents.

– The basic components of design rules are virtual world objects and their properties, not shapes. Therefore, they are not entirely visual/spatial. As a result, for the interaction rules and parts of the navigation rules, the replacement of LHO with RHO may not change the way an object looks or is placed, but may add scripted behaviors to an existing object.

3.4.1 Layout Rules

Layout rules are the first set of design rules to be applied in the application of a generative design grammar. They are visual/spatial rules that generate the layout of the place according to the kinds of intended activities to be supported in the 3D virtual world. The use of divided virtual areas for different activities provides a way of organizing and allocating activities in the 3D virtual world, and creates a sense of movement for people when changing from one activity to another. Because of the use of the place metaphor, layout problems in 3D virtual worlds can have similar solutions to the ones in the physical world. However, designers have more freedom in making design decisions since virtual worlds do not have to obey physical constraints. In virtual worlds, many layout-related issues, for example, adjacency, do not need to strictly follow their physical counterparts because virtual places can also be hyper-linked.

Figure 3.2 illustrates an example layout rule taken from a generative design grammar for virtual gallery design. In this design rule, the LHO of the design rule is a reception area of the virtual gallery. The rule shows that after it is applied the LHO will be replaced by the RHO: the same reception area with a gallery area added spatially adjacent to the reception.

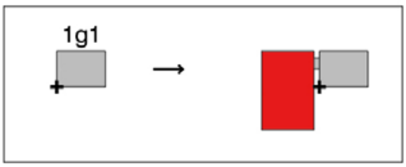


Figure 3.2: An example layout rule for designing a virtual gallery.

Symbol  represents a reception area of the virtual gallery. Symbol  represents a gallery area in the virtual gallery for displaying exhibitions. State label sL=1 indicates that layout rules will always be the first category of design rules to be applied in a generative design grammar. The meaning of state label sL=g1 is to generate a gallery area in the 3D virtual world according to certain specifications (g1). In order for this design rule to be applied the following two conditions need to be met:

- A reception area  is recognized in the 3D virtual world.
- The design contexts represented by sL=g1 (to generate a gallery area in the 3D virtual world according to certain specifications) are related to the current design needs (such as a given design brief or emerging design requirements like the sudden increase of galley visitors), as interpreted by the designers or computational agents.

3.4.2 Object Design Rules

Object design rules are applied after layout rules, they are also visual/spatial rules. After a layout of the virtual place is produced, object design rules further configure the place to provide visual boundaries of the place and visual cues for supporting the intended activities through virtual world object design and placement. Because of the use of the place metaphor, the development of object design rules can also refer to many principles and examples in built environments. Depending on the designers' preferences, they may choose to simulate a place from the physical world, or to experiment with other alternatives since virtual worlds are in fact free from any physical constraints. However, 3D virtual worlds designed with a consistent use of forms, colors and other visual/spatial elements are arguably more effective in assisting users' orientation and interaction.

Figure 3.3 shows an example object design rule that generates the visual boundaries for a gallery area in the virtual gallery, and Figure 3.4 shows an example object design rule that arranges the interior of the gallery area with virtual canvas objects for displaying digital images.

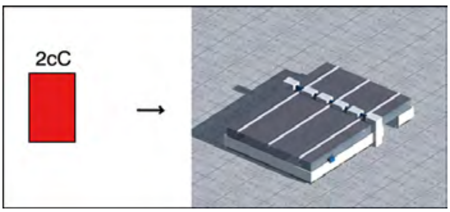


Figure 3.3: An example object design rule that generates visual boundaries for a gallery area in the virtual gallery.

In Figure 3.3, state label sL=2 indicates that object design rules will be the second category of design rules to be applied in a generative design grammar, after the layout rules. State label sL=cC means to apply a cold-color scheme (cC) for the interior of the generated virtual place. In order for this object design rule to be applied, the following two conditions need to be met:

- The layout generated for the virtual gallery contains a gallery area.
- The design contexts represented by sL=cC (to apply a cold-color scheme for the generated virtual place) are related to the current design needs (in terms of design preferences), as interpreted by the designers or computational agents.

In Figure 3.4, state label sL=gIM1 means to arrange a gallery area for displaying digital images using certain configuration settings (gIM1). The application of this object design rule also requires two conditions:

- The visual boundaries of a gallery area have been defined in the virtual gallery.
- The design contexts represented by sL=gIM1 (to arrange the gallery area for displaying digital images using certain configuration settings) are related to the current design needs (in terms of the exhibition requirements such as the type and quantity of virtual exhibition items), as interpreted by the designers or computational agents.

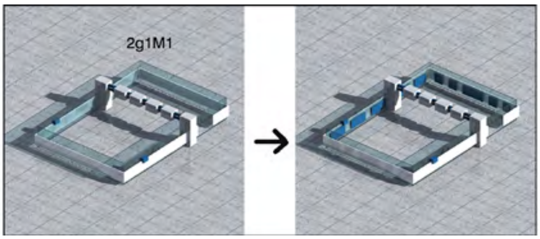


Figure 3.4: An example object design rule that arranges the interior of a gallery area for displaying digital images in the virtual gallery.

3.4.3 Navigation Rules

Navigation rules are applied next in a generative design grammar, after layout rules and object design rules. Navigation rules provide way finding aids and hyperlinks in the generated virtual place to assist users' navigation. Way finding aids in virtual worlds have been studied with direct references to those in built environments (Vinson, 1999; Darken & Sibert, 1996; 1993). There are at least two types of way finding aids that can be integrated into virtual worlds from the built environments:

- The use of spatial elements; for example, paths, openings, hallways, stairs, intersections, landmarks, maps, signs and so on.
- The use of social elements; for example, the assistance gained from designated guides (i.e. a conversational bot) or other virtual world users.

Besides these way-finding aids originating from built environments, virtual worlds also have their own unique forms of navigation since virtual places can be hyper-linked. Most virtual worlds allow users' avatars to move directly between any two locations using hyperlinks. The origin of hyperlinks used in virtual worlds can be traced back to the navigation in hypertext systems such as in a web page (Dourish, 1999, Ruddle et al, 1997). Current commercial virtual worlds such as Active Worlds, Second Life and other game engines all support different forms of hyperlinks for connecting virtual places.

Navigation rules are not entirely visual/spatial. The application of these rules involves virtual world object design and placement for defining way-finding aids and hyperlinks in the generated place, which is related to the visual/spatial aspect of designing places in 3D virtual worlds. However, before these object design and placement are conducted, navigation rules are mainly about recognizing the connectivity within the generated place and with other places as well as finding appropriate navigation methods for users to access these places.

Figure 3.5 is an example navigation rule. The LHO of this rule shows that two gallery areas are generated for the virtual gallery, separated without direct access to each other. The RHO of this rule shows that a pair of hyperlinks is created inside these two gallery areas, which allows visitors to travel freely between the two areas. State label sL=3 indicates that navigation rules are the third set of rules to be applied in a generative design grammar, after layout rules and object placement rules.

Figure 3.5: An example navigation rule for connecting two separated gallery areas in the virtual gallery using hyperlinks.

Figure 3.6 shows the effect of this navigation rule as illustrated in Figure 3.5. The left-hand-side image captures the interior of one of the gallery areas. The right-hand-side image shows that a hyperlink portal is created. In this particular case, the hyperlink portal appears as a color stone on the floor. After appropriate scripted behaviors are ascribed, the portal will take the visitor directly to the other gallery area when it is "stepped" on by the visitors' avatars.

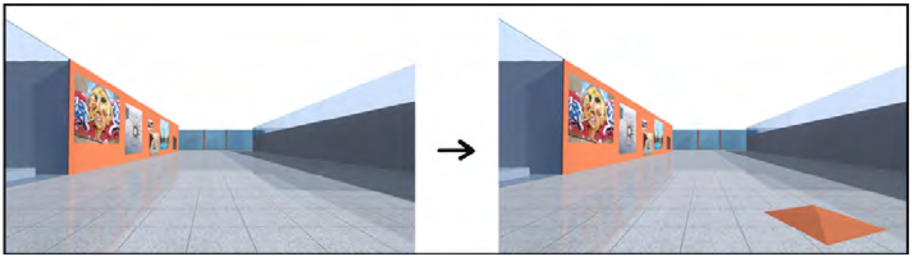


Figure 3.6: The effect of the example navigation rule as shown in Figure 3.5.

3.4.4 Interaction Rules

Interaction rules are the final set of design rules to be applied in a generative design grammar. The application of interaction rules ascribes scripted behaviors to selected virtual world objects in the generated place. People can interact with the virtual place and with each other by triggering these behaviors.

Interactions in virtual worlds in general are not well developed. There are examples that derive from various actions in computer games. There are also more advanced attempts; for example, the integration of artificial intelligence. The interaction rules of generative design grammars do not intend to develop new interactions for 3D virtual worlds. The types of interactions generated by these rules will be mainly supplied by the existing virtual world design platforms realized in the forms of different scripted behaviors for virtual world objects. However, by developing the interaction rules, the conventions of triggering these behaviors in virtual worlds can be defined and categorized, for designers to use selectively for different purposes.

Interaction rules are non-visual/spatial rules that recognize selected virtual world objects in the 3D virtual world and ascribe appropriate behaviors to these objects. There can be at least two different types of interaction rules. One supplements object design rules and the other supplements navigation rules. Object design rules define visual boundaries for the generated place as well as design and place purposeful virtual worlds objects in the generated place. The first type of the interaction rule ascribes scripted behaviors to relevant virtual world objects in order to support the intended activities in the generated virtual place. The other type of interaction rule looks for way finding aids and hyperlinks generated by navigation rules and ascribe scripted behaviors to activate them.

Because interaction rules do not operate on a visual/spatial level, they are not appropriate to be expressed using illustrations. Examples of interactions rules are provided in Chapter 5 of this book. They are expressed in the form of “IF... THEN...”, taken from a generative design grammar developed for virtual gallery design. Without getting into the technical details of ascribing behaviors to virtual world objects through scripting, Figure 3.7 shows the effect of an example interaction rule of the first type for supplementing object design rules. The left-hand-side image is the exterior of a virtual building with an empty virtual advertisement board. The right-hand-side image shows the same advertisement board displaying an animation, after the interaction rule is applied, which configures the properties of the virtual advertisement board object using a scripting language to enable the animation to be shown.

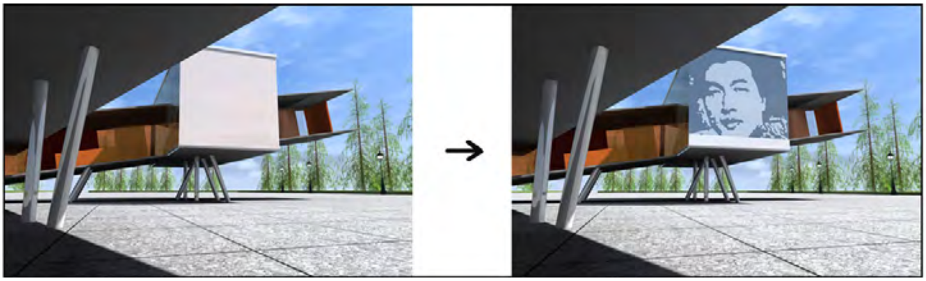


Figure 3.7: The effect of an example interaction rule for displaying animation.

3.5 Characteristics of Generative Design Grammars

Generative design grammars adopt the descriptive and generative nature of shape grammars, but modify some of the original shape grammar properties to suit the purpose of designing places in 3D virtual worlds. As design formalisms, they have similarities as well as differences. The characteristics of generative design grammars can be highlighted based on a comparison to shape grammars in general. The comparison is drawn in terms of the following three aspects: grammar components, meeting design requirements, and computation.

Grammar components: in shape grammars, a set of shape rules, an initial shape and a final state constitute a shape grammar (Knight, 1994a). Shape rules are applied recursively in a step-by-step manner to generate designs. Spatial and state labels are used to control the application of shape rules.

– In generative design grammars, similarly, a grammar comprises design rules, an initial design and a final state of the design. More specifically, a generative design grammar has four sets of design rules: layout rules, object design rules, navigation rules and interaction rules. They can be categorized as syntactic and semantic rules. Each set of the design rules corresponds to a phase of place design in 3D virtual worlds.
– Generative design grammars also use spatial labels and state labels to control the application of design rules. The original use of state labels in a shape grammar is to control the sequence of shape rule applications. In our generative design

grammar, this original purpose is maintained so that the design rules can be applied in the sequence of layout rules, object design rules, navigation rules and interaction rules. In addition, a special set of state labels are developed to represent a set of design contexts that relate the current design needs to the application of a design rule. Using these special state labels, the application of the generative design grammar is also able to adapt an existing design to the changing needs of the user inhabiting the virtual place.

Meeting design requirements: as discussed earlier, Knight (1999) points out that there are generally two different ways to connect shape grammars with design goals. The first approach is to integrate the constraints into the shape rules so that the generated designs will meet the given design requirements. The key process here is to add constraints to the original, unrestricted shape grammars to increase the predictability of the designs they generate. The second approach is to develop the original, unrestricted shape grammars and enable the grammar application to be executed without constraints but apply a manual or automated search and test strategy to the generated designs in the end to select the designs that satisfy the given design requirements.

– Generative design grammars are restricted. They are similar to set grammars (Stiny, 1982). Set grammars are restricted kinds of shape grammars. According to Knight (1998b), the main difference between set grammars and the original shape grammars is the way designs are decomposed. In a set grammar, each generated design can only be viewed and decomposed in one definite way. In an original shape grammar, each generate design can be viewed and decomposed in unlimited ways. This liberty of the original shape grammars on one hand supports two very important aspects of shape grammars' generative power: emergence and ambiguity. On the other hand, it also makes it very difficult to predict the behaviors and design outcomes of grammar applications. Similar to designs generated by set grammars, a place in 3D virtual world designed and composed using generative design grammars can be decomposed uniquely into a set of purposeful virtual world objects and their properties. The application of a generative design grammar also needs to be controllable and predictable to a certain extent so that the generated places can be useful in supporting the intended activities, since virtual worlds in the context of this book are essentially functional networked places. The purpose of generating virtual world designs that meet current design needs is given higher priority than supporting emergence and ambiguity in design. However, the level of control should certainly be flexible and adjustable to support different design purposes and tasks.
– As discussed earlier, to direct the application of a generative design grammar relies on the use of a special set of state labels. Each design rule in a generative design grammar is associated with at least one of these state labels, which represent a specific design context of designing places in 3D virtual worlds. In order for a design rule to be applied, the LHO of the design rule needs to be matched in the

virtual world, and the design context represented by the state label of the rule also needs to match the current design needs. In this manner, the application of the generative design grammar is directed to generate virtual world designs that meet the given design requirements.

- Generative design grammar applications can be carried out manually by designers or be automated by computational agents, during real-time interactions in a 3D virtual world. The generated places can be implemented and put into use immediately. These require each generated design to meet the current design needs, and to be implemented with efficiency in order to keep up with the fast pace of real-time interactions in the 3D virtual world. The search and test approach is impractical in this respect, and therefore was not considered further in the context of this book.

Computation: in the initial studies of shape grammars, the design generation is performed by hand. To manually perform the step-by-step application and examine and present different generated design instances, is very time-consuming. Automating this process has many advantages. According to Gips (1999), the automation of shape grammars has at least four different purposes. The most common purpose is to assist the generation of designs from a shape grammar. The other three purposes are to analyze if a given design belongs to the language generated by a certain grammar, to generate a shape grammar based on a corpus of given designs, and to assist designers in designing shape grammars. Gips further points out that the challenge in the automation of shape grammars lies in the tension between the spatial nature of shape grammars and the symbolic nature of the underlying computer representations and processing. In order to implement shape grammars as a computational process or system, it requires a different kind of thinking and representation (Stiny, 2008). Although generative design grammars can be either applied manually by designers or automated by computational agents, they are intended for computer implementations.

In summary, following the above framework, generative design grammars can be developed for designing specific places in 3D virtual worlds. The application of a generative design grammar begins when the current design needs are identified and the initial design D_i is recognized in the 3D virtual world. The appropriate design rules are subsequently selected and applied. The application is terminated when there are no matching rules and the final state of the place design D_i is then generated for the 3D virtual world. Different generative design grammars are able to generate places in 3D virtual worlds with different stylistic characterizations. Different designers and users may adopt them to reflect their own styles and identities in virtual worlds.

References

Carlson C., McKelvey R. and Woodbury R. (1991), An Introduction to Structures and Structure Grammars, *Environment and Planning B: Planning and Design*, 18, pp. 417-426.

Darken, R.P. and Sibert J.L. (1996), Way Finding Strategies and Behaviours in Large Virtual Worlds, *Proceedings of ACM SIGCHI'96*, ACM, New York, pp. 142-149.

Darken, R.P. and Sibert, J.L. (1993), A Toolset for Navigation in Virtual Environments, *Proceedings of the 6th Annual ACM Symposium on User Interface Software and Technology*, Atlanta, pp 157-165.

Deneubourg, J.L. (1977), Application de l'ordre par fluctuations a la description de certaines etapes de la construction du nid chez les termites, *Insect Soc*, 24, pp. 117-130.

Ding, L. and Gero, J.S. (2001), The Emergence of the Representation of Style in Design, *Environment and Planning B: Planning and Design*, 28(5), pp. 707-731.

Dourish, P. (1999), Following Where the Footprints Lead: Tracking Down New Roles for Social Navigation, *in* A. J. Munro, K. Hook, and B. D. Benyon (eds), *Social Navigation of Information Space*, Springer, London, pp. 15-34.

Duarte, J. (1999), Democratized Architecture: Grammars and Computers for Siza's Mass Housing, *Proceedings of the International Conference on Enhancement and Promotion of Computational Methods in Engineering and Science*, Elsevier Press, Macau, China.

Duarte, J. (2005), A Discursive Grammar for Customizing Mass Housing: the Case of Siza's Houses at Malagueira, *Automation in Construction*, 14(2), pp. 265-275.

Eloy, S and Duarte, J. (2011), Transformation Grammar for Housing Rehabilitation, *Nexus Network Journal*, 13(1), pp. 49-71.

Gips, J. (1999), Computer Implementation of Shape Grammars, *NSF/MIT Workshop on Shape Computation*, http://www.shapegrammar.org/implement.pdf

Gu, N. and Maher, M.L. (2003), A Grammar for the Dynamic Design of Virtual Architecture Using Rational Agents, *International Journal of Architectural Computing*, 4(1), pp. 489-501.

Holland, J. H. (1992), *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA.

Knight, T.W. (2000), Shape Grammars in Education and Practice: History and Prospects, *International Journal of Design Computing*, 2 (online).

Knight, T.W. (1999), Applications in Architectural Design, and Education and Practice: Report for NSF/MIT Workshop on Shape Computation, *NSF/MIT Workshop on Shape Computation*, http://www.shapegrammar.org/education.pdf

Knight, T.W. (1998a), Designing a Shape Grammar, *in* J.S. Gero and F. Sudweeks (eds), *Proceedings of Artificial Intelligence in Design '98*, Kluwer Academic Publishers, The Netherlands, pp. 499-516.

Knight, T.W. (1998b), Shape Grammars, *Environment and Planning B: Planning and Design*, Anniversary Issue, pp. 86-91.

Knight, T.W. (1994a), *Transformations in Design: a Formal Approach to Stylistic Change and Innovation in the Visual Arts*, Cambridge University Press, England.

Knight T.W. (1994b), Shape Grammars and Color Grammars in Design, *Environment and Planning B: Planning and Design*, 21, pp. 705-735.

Koning H. and Eizenberg J. (1981), The Language of the Prairie: Frank Lloyd Wright's Prairie Houses, *Environment and Planning B: Planning and Design*, 8, pp. 295-323.

March, W.J. and Stiny, G. (1985), Spatial Systems in Architecture and Design: Some History and Logic, *Environment and Planning B: Planning and Design*, 12, pp. 31-53.

Payman, A. (2004), *Swarm Intelligence*, Jet Propulsion Laboratory, Pasadena, CA.

Ruddle, R.A., Payne, S.J., and Jones, D.M. (1997), Navigating Buildings in "Desk-top" Virtual Environments: Experimental Investigations Using Extended Navigational Experience, *Journal of Experimental Psychology: Applied*, 3(2), pp. 143-159.

Shea, K. and Cagan, J. (1996), Innovative Dome Design: Applying Geodesic Patterns with Shape Annealing, *Artificial Intelligence for Engineering Design Analysis, and Manufacturing*, 11(5), pp. 379-394.

Stiny, G. (2008). *Shape: Talking about Seeing and Doing*, MIT Press.

Stiny G. (1999), Commentary: Shape, *Environment and Planning B: Planning and Design*, 26, pp. 7-14.

Stiny G. (1990), What Is a Design?, *Environment and Planning B: Planning and Design*, 17, pp. 97-103.

Stiny G. (1982), Spatial Relations and Grammars, *Environment and Planning B: Planning and Design*, 9, pp. 113-114.

Stiny G. (1981), A Note on the Description of Designs, *Environment and Planning B: Planning and Design*, 8, pp. 257-267.

Stiny G. (1980), Introduction to Shape Grammars, *Environment and Planning B: Planning and Design*, 7, pp. 343-351.

Stiny G. and Gips J. (1972), Shape Grammars and the Generative Specification of Painting and Sculpture, *in* C.V. Freiman (ed), *Proceedings of Information Processing 71*, North Holland, Amsterdam, pp. 1460-1465.

Stiny, G. and Mitchell, W.J. (1980), The Grammar of Paradise: on the Generation of Mughul Gardens, *Environment and Planning B: Planning and Design*, 7, pp. 209-226.

Stiny, G. and Mitchell, W.J. (1978), The Palladian Grammar, *Environment and Planning B: Planning and Design*, 5, pp. 5-18.

Tapia M.A. (1992), Chinese Lattice Designs and Parametric Shape Grammars, *The Visual Computer*, 9, pp. 47-56.

Vinson, N.G. (1999), Design Guidelines for Landmarks to Support Navigation in Virtual Environments, *Proceedings of CHI'99*, Pittsburgh, pp. 278-285.

von Neumann, J. (1951), The General and Logical Theory of Automata, *in* L.A. Jeffress (ed.), *Cerebral Mechanisms in Behavior - the Hixon Symposium*, John Wiley, New York, NY, pp. 1-41.

Wolfram, S. (2002), *A New Kind of Science*, Wolfram Media, Champaign, IL.

4 Generative Design Agents

This chapter describes a design agent model, called Generative Design Agent (GDA), whose computational processes are specifically developed for reasoning and designing in 3D virtual worlds. The agent model is wrapped around the design grammar, providing the mechanisms to sense and change the virtual world. Each GDA can be associated with a user of 3D virtual worlds and serves as his/her personal design agent to automatically and dynamically design places in virtual worlds, satisfying the person's changing needs and capturing his/her style. By automating the generative design grammar application using GDAs, adaptive place design in 3D virtual worlds is realized.

4.1 Computational Agents

While agent-based computing started in the 1970s, recently the concept of agents has become important for internet applications, drawing on ideas from artificial intelligence and artificial life.

4.1.1 Notions of Agents

There is no universal definition for the term agent. However, in the context of computer science, agents as intentional systems operate independently and rationally, seeking to achieve goals by interacting with their environment (Wooldridge & Jennings, 1995).

An agent is always located in an environment; it receives inputs from the environment, and acts autonomously upon the environment. According to Russell and Norvig (2009), agents are distinguished from objects and other computer programs, by a constraint of rationales on the agent's beliefs, goals and actions. As shown in Figure 4.1, Russell and Norvig's reflex agent diagram (2009) illustrates the basic interaction between an agent and its environment. The agent reasons about the condition of its environment and acts based on condition-action rules. Their utility agent diagram (2009) shown in Figure 4.2, illustrates a rational agent with additional beliefs and more complex reasoning processes.

We build on the definition of an agent as situated in an environment, capable of reasoning about the environment, and acting upon its beliefs and desires (Wooldridge, 2000). This definition distinguishes agents from other computational applications that perform actions based on predefined events, such as search engines on the web.

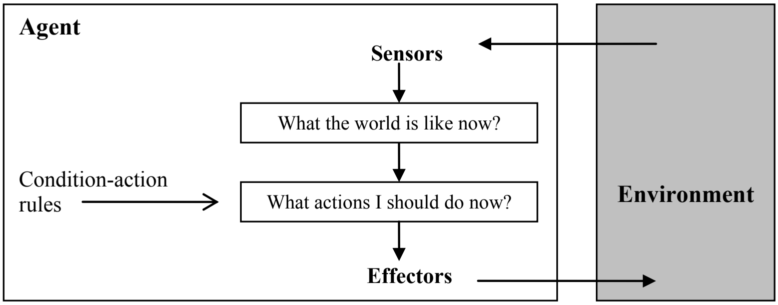


Figure 4.1: Reflex agent (after Russell and Norvig, 2009).

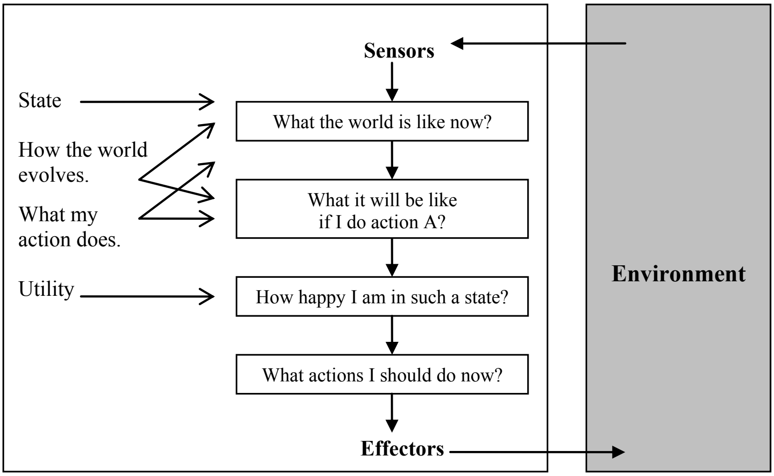


Figure 4.2: Utility agent (after Russell and Norvig, 2009).

An agent has the ability to operate usefully by itself. However, the increasing interconnection and networking of computers make this situation rare. Most problems require the agent to interact with other agents (Huhns & Stephens, 1999). Hence the concept of the multi-agent system is introduced and associated with distributed artificial intelligence. Ferber (1999) defines a multi-agent system to include the following elements:

- An assembly of agents.
- An environment where agents are located.
- A set of objects that are located in the same environment. These objects are passive, which means that they can be perceived, created and manipulated by the agents.
- An assembly of relationships.
- An assembly of operations for agents to perceive, create and manipulate.
- Operators for pursuing the operations.

4.1.2 A Common Agent Model for 3D Virtual Worlds

Maher and Gero (2002; 2003) develop a common agent model for 3D virtual worlds for increasing the level of interactivity of virtual worlds. The agent model is applied so that virtual world objects can respond more generally to their uses. Such responses can result in a dynamic world that configures and reconfigures itself as needed.

As shown in Figure 4.3, using this model each virtual world component can be represented as an agent element, and a virtual world can be represented by a society of agents; for example, wall agents, door agents, room agents or building agents. Each agent is able to sense and respond to its use in the virtual world by reasoning about the current state of the virtual world. The agent model provides a common vocabulary for describing, representing, and implementing agent knowledge and communication. In this common agent model, key agent components and computational processes are outlined for supporting interactions between the agents and the virtual world, and interactions among different agents in the virtual world. An agent developed using this common model interacts with the virtual world via sensors and effectors, and has five kinds of reasoning: sensation, perception, conception, hypothesizer and action.

- Sensation transforms raw inputs from the sensors into data that are more appropriate for agent reasoning and learning.
- Perception is a process that finds grounded patterns of invariance in the agent's representation of the sense data for constructing concepts.
- Conception learns and uses concepts to reinforce or modify the agent's beliefs and goals.

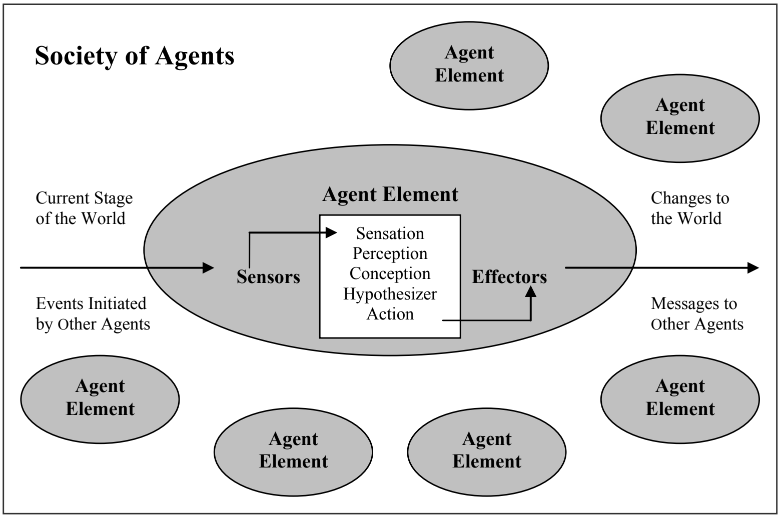


Figure 4.3: A virtual world as a society of agents (after Maher and Gero, 2002; 2003).

- Hypothesizer identifies mismatches between the current and desired states of the virtual world and hypothesizes goals in order to reduce or eliminate mismatches.
- Action reasons about the sequence of operations on the virtual world, when executed, can achieve the agent's goals.

This agent approach to virtual worlds provides new kinds of interactions not only among the virtual world elements, but between these elements and the virtual world users. Using such a reasoning mechanism, agents can function in three different modes: reflexive, reactive, and reflective (Figure 4.4). Each successive mode requires a more sophisticated reasoning mechanism.

- Reflexive mode: the agent responds to sense data from the environment with pre-programmed responses. This mode is equivalent to the kinds of behaviors that are available in most virtual worlds. Reflexive mode is the simplest among the three.
- Reactive mode: the agent's reasoning involves both the sense data, the perception processes that manipulate and operate on these data, and knowledge about the processes.
- Reflective mode: the agent filters its sense data depending on its current goals and beliefs. The agent also adjusts its perception processes depending on its current goals and beliefs, which may lead to changes in its concepts. The reflective mode includes all these dynamics to allow the agent to adjust its interests accordingly.

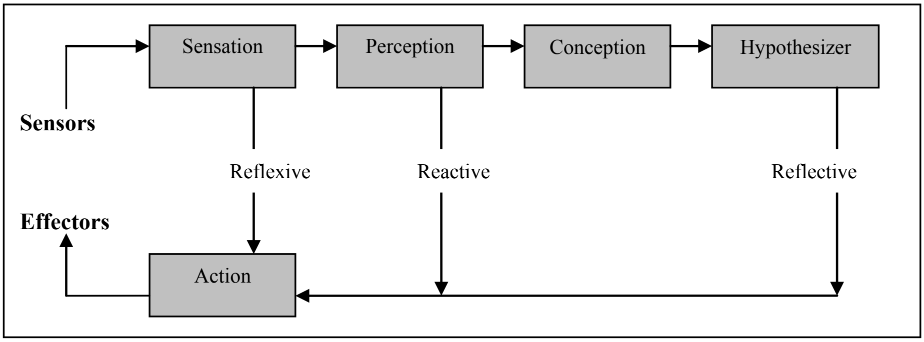


Figure 4.4: A common agent model for 3D virtual worlds (after Maher and Gero, 2002; 2003).

Taking together the advantages of agents being capable of operating independently and rationally, and who seek to achieve goals by interacting with their environment, we propose that agents serve as design assistants for virtual world users in order to automatically and dynamically design, implement and manipulate virtual worlds as needed through real-time interactions.

4.2 Generative Design Agent Model

We consider some of the existing agent models as the basis for developing the Generative Design Agent (GDA) model. Russell and Norvig's reflex agent diagram (2009) illustrates the basic concepts of an agent (Figure 4.1). The agent reasons about the condition of its environment and acts based on condition-action rules. Russell and Norvig's utility agent diagram (2009) illustrates an agent with additional beliefs and more complex reasoning processes (Figure 4.2). For the development of the GDA model, we draw attention to the way Russell and Norvig's models define basic relations between an agent and its environment and provide mechanisms for establishing such relations.

With the common agent model for 3D virtual worlds (Figure 4.4) by Maher and Gero (2002; 2003), each virtual world component can be represented as an agent element, and a virtual world can be represented by a society of agents; for example, wall agents, door agents, room agents or building agents. Each agent is able to sense and respond to its use in the virtual world by reasoning about the current state of the virtual world.

In this common agent model, key agent components and computational processes are outlined, for supporting interactions between the agents and the virtual world, and interactions among different agents in the virtual world. As shown in Figure 4.4, an agent developed using this common model has five kinds of reasoning: sensation, perception, conception, hypothesizer and action. Using such a reasoning mechanism, agents are able to behave in three different modes: reflexive, reactive and reflective, with each mode requiring more complex reasoning.

In contrast to this common agent model for 3D virtual worlds, the GDA model takes a different approach by giving the agency to virtual world users, rather than to the virtual world components that make up the environments. Using the GDA model, a virtual world can also be represented by a society of GDAs. Each GDA can be associated with a user's avatar in the virtual world and serves as the user's personal design agent in the virtual world. Because the agencies are given to the users, rather than to existing virtual world components, the virtual world therefore does not need to be pre-defined. GDAs reason, design and act on behalf of the virtual world users, and automatically and dynamically design places in the virtual worlds, satisfying the people's changing needs and capturing their own design styles and preferences.

The reasoning mechanism of the GDA model is developed based on the framework shown in Figure 4.4, where sensors and effectors act as the interface between a GDA and the virtual world. The five computational processes of the GDA model are sensation, interpretation, hypothesizing, designing and action. They are illustrated in Figure 4.5. Compared to the common agent model for 3D virtual worlds, the differences of the GDA model may be briefly outlined thus:

- Sensation: similarly, a GDA uses its sensors to retrieve relevant raw data from the virtual world to prepare for the process of interpretation.

- Interpretation: in the common agent model, an agent interprets the sense data internally in two different stages via the processes of perception and conception. In the GDA model, these two processes are integrated into the general process of interpretation. The GDA conducts interpretation in three different stages. The ultimate outcomes of the processes are the GDA's interpretations of the current design needs in the virtual world and the current state of the virtual world.
- Hypothesizing: in the process of hypothesizing, the GDA sets up design goals that aim to eliminate or reduce the mismatches between the current design needs in the virtual world and the current state of the virtual world.
- Designing: GDAs are developed specifically for the purpose of designing places in 3D virtual worlds. To address this design aspect, one of the computational processes of the GDA model is singled out as designing. In the process of designing, the GDA provides a virtual place design in order to satisfy its current design goals. The design component of the GDA is supported by the application of a generative design grammar.
- Action: the final processes of action include action planning and action activation. The GDA plans actions for implementing the generated place design in the virtual world and realizing other initiated changes, as well as activates these actions in the virtual world. The results of action lead to changes in the virtual world, which trigger the GDA to start a new cycle of reasoning and designing. In this manner, the virtual world is able to be automatically and dynamically designed, implemented and manipulated, adapting to its use.

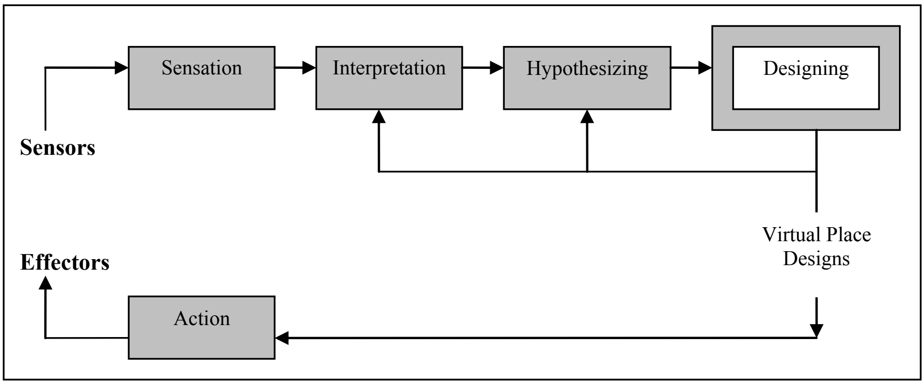


Figure 4.5: Five computational processes of the GDA model.

4.3 Representations of a 3D Virtual World

A situated view of design (Gero & Kannengiesser, 2004) assumes a non-static environment where the act of designing takes place. Adopting such a view, the process of designing in the GDA model can be modelled as the interactions of three representation layers of the virtual world: the external world, the interpreted world and the expected world. The external world comprises representations outside a GDA. The interpreted world is the internal representation of the external world that exists inside the GDA, in terms of its knowledge and experience. The expected world is a part of the interpreted world in which the results of designing are predicted based on the GDA's current design goals and its interpretations of the current state of the world. Figure 4.6 illustrates the interactions among these three representation layers.

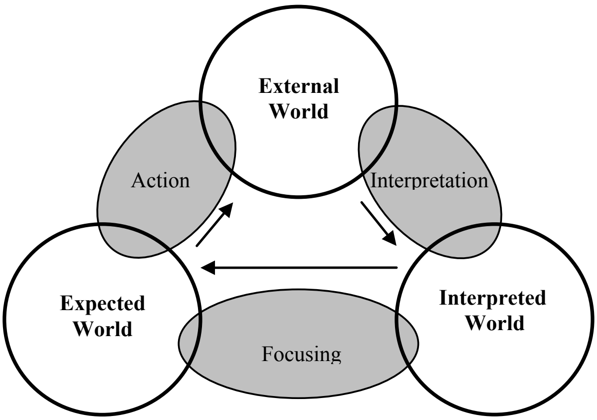


Figure 4.6: Interactions among the external world, the interpreted world and the expected world (after Gero and Kannengiesser, 2004).

4.3.1 Three Representation Layers of a 3D Virtual World

The development of the GDA model adopts the three representation layers for the context of designing virtual worlds in order to capture the characteristics of virtual worlds being adaptive and non-static. For GDAs, a virtual world W can be viewed as the union of the external world W_{ext}, the interpreted world W_{int}, and the expected world W_{exp}.

$$W = W_{ext} \cup W_{int} \cup W_{exp} \qquad (4.1)$$

At any given moment, a virtual world is comprised of various components that construct the environments, the representations of its users (avatars), and events that occurred in the world. These entities exist outside a GDA. Depending on the GDA's capability and focus, the external world W_{ext} of the GDA can include the whole or parts of these entities. The interpreted world W_{int} is the GDA's internal representation of W_{ext}, which reflects on the GDA's interpretations of W_{ext}. The current design needs in the virtual world and the current state of the virtual world are interpreted in W_{int}. The expected world W_{exp} is where the GDA hypothesizes its design goals and generates virtual world designs, or initiates other changes in the virtual world for satisfying the goals. The GDA's design goals are hypothesized by matching against its interpretations of the current design needs and the current state of the virtual world. A virtual place design is then generated to satisfy the current design goals by applying the GDA's generative design grammar.

The external world W_{ext} comprises A_{ext}, E_{ext}, wA_{ext} and O_{ext}.

$$W_{ext} = A_{ext} \cup E_{ext} \cup wA_{ext} \cup O_{ext} \tag{4.2}$$

where A_{ext}, E_{ext} and O_{ext} are each represented by a set of elements of the same kind, and wA_{ext} is represented by an ordered list of properties.

$$A_{ext} = \{avatar_{ext\_1}, avatar_{ext\_2}, \ldots, avatar_{ext\_n}\}$$

$$E_{ext} = \{event_{ext\_1}, event_{ext\_2}, \ldots, event_{ext\_n}\}$$

$$wA_{ext} = (size_{ext}, capacity_{ext}, owner_{ext}, server_{ext}, system\_time_{ext},)$$

$$O_{ext} = \{O_{ext\_1}, O_{ext\_2}, \ldots, O_{ext\_n}\}$$

Symbol A_{ext} is the representation of the users' avatars in the virtual place, that is, GDAs, and other avatars (with no agency in the virtual world).

Symbol E_{ext} represents various events in the virtual world.

Symbol wA_{ext} represents the attributes of the virtual world; for example, size, capacity and ownership of the virtual place, server location, system time in the virtual world and so on.

Symbol O_{ext} represents objects in the virtual world. In an object-oriented virtual world, an object may refer to a virtual place; for example, a virtual gallery or a virtual meeting room. An object may also refer to an entity in a virtual place, or it can form a part of the place; for example, a digital picture in a virtual gallery, or a wall of the virtual meeting room.

The interpreted world W_{int} comprises A_{int}, E_{int}, wA_{int} and O_{int}.

$$W_{int} = A_{int} \cup E_{int} \cup wA_{int} \cup O_{int} \tag{4.3}$$

where A_{int}, E_{int}, wA_{int} and O_{int} are the GDA's internal transformations of A_{ext}, E_{ext}, wA_{ext} and O_{ext}:

$$A_{int} = \tau(A_{ext})$$

$$E_{int} = \tau(E_{ext})$$

$$wA_{int} = \tau(wA_{ext})$$

$$O_{int} = \tau(O_{ext})$$

Symbol A_{int}, E_{int}, wA_{int} and O_{int} represents the GDA's interpretations of A_{ext}, E_{ext}, wA_{ext} and O_{ext}. Further, the GDA interprets the current design needs in the virtual world N and the current state of the virtual world sT:

$$N = \tau(W_{int})$$

$$sT \in W_{int}$$

The expected world W_{exp} comprises A_{exp}, E_{exp} and O_{exp}

$$W_{exp} = A_{exp} \cup E_{exp} \cup O_{exp} \quad (4.4)$$

where A_{exp}, E_{exp} and O_{exp} are used to represent design goals hypothesized by the GDA that aim at eliminating or reducing the mismatches between N and sT:

$$A_{exp} \in \tau_A(N, sT)$$

$$E_{exp} \in \tau_E(N, sT)$$

$$O_{exp} \in \tau_O(N, sT)$$

Symbol A_{exp} represents the expected attributes of the GDAs and other avatars; for example, the expected location of a GDA or an avatar.

Symbol E_{exp} represents the expected events in the virtual world; for example, to send a message to users who connect to the virtual world.

Symbol O_{exp} represents the expected objects and their attributes in the virtual world.

The GDA's expected world does not include wA_{exp}, as the attributes of the virtual world are generally not changeable.

4.3.2 Representations of Virtual World Objects

We adopt the Function-Behavior-Structure (F-B-S) framework (Gero, 1990) to the context of designing virtual worlds for representing virtual world objects. In general:

$$O_i = O_i^F \cup O_i^B \cup O_i^S \quad (4.5)$$

where O_i represents any virtual world object.

For the GDA model, the representation of a virtual world object in the external world includes structures $O_{ext\_i}^S$. Functions $O_{ext\_i}^F$ and behaviors $O_{ext\_i}^B$ of the object are assumed nil, and are addressed in the GDA's interpreted world.

$$O_{ext\_i}^S = \{structure_{ext\_1}, structure_{ext\_2}, \ldots, structure_{ext\_n}\}$$

$$O_{ext\_i}^B = \emptyset$$

$$O_{ext\_i}^F = \emptyset$$

Every object in the interpreted world has a counterpart in the external world. The interpreted functions $O_{int\_i}^{F}$ and behaviors $O_{int\_i}^{B}$ are derived from the interpreted structures $O_{int\_i}^{S}$.

$$O_{int\_i}^{S} = \tau(O_{ext\_i}^{S})$$

$$O_{int\_i}^{B} = \tau(O_{int\_i}^{S})$$

$$O_{int\_i}^{F} = \tau(O_{int\_i}^{B})$$

An object in the GDA's expected world may or may not have a counterpart in the interpreted world, since a place design in the virtual world can either be generated based on existing structures or by new creations. The GDA's design goals are hypothesized in terms of the expected functions $O_{exp\_i}^{F}$ and the expected behaviors $O_{exp\_i}^{B}$, both according to the current design needs N and to the current state of the virtual world sT.

$$O_{exp\_i}^{F} \in \tau(N, sT)$$

$$O_{exp\_i}^{B} = \tau(O_{exp\_i}^{F})$$

Once a design goal is hypothesized, the GDA applies its generative design grammar to generate structures represented by $O_{exp\_i}^{S}$, to be realized in the virtual world, in order to satisfy the design goal or parts of the design goal.

$$O_{exp\_i}^{S} = \tau(O_{exp\_i}^{F}, O_{exp\_i}^{B}) \tag{4.6}$$

In summary:

- The external world $W_{ext} = A_{ext} \cup E_{ext} \cup wA_{ext} \cup O_{ext}$. The representation of each virtual world object in the external world includes structures $O_{ext\_i}^{S}$. Functions $O_{ext\_i}^{F}$ and behaviors $O_{ext\_i}^{B}$ of the object are assumed nil, and are addressed in the GDA's interpreted world.
- The GDA's interpreted world $W_{int} = A_{int} \cup E_{int} \cup wA_{int} \cup O_{int}$. Each object in the interpreted world has a counterpart in the external world. $O_{int\_i}^{S}$ can be the same or different from $O_{ext\_i}^{S}$ depending on the individual GDAs. $O_{int\_i}^{F}$ and $O_{int\_i}^{B}$ are derived from $O_{int\_i}^{S}$.
- The GDA's expected world $W_{exp} = A_{exp} \cup E_{exp} \cup O_{exp}$. An object in the expected world may or may not have a counterpart in the interpreted world. $O_{exp\_i}^{F}$ and $O_{exp\_i}^{B}$ represent a GDA's design goals. To satisfy these design goals or parts of these goals, the GDA applies its generative design grammar to generate a place design or parts of the design represented by $O_{exp\_i}^{S}$ for the moment to be implemented in the virtual world.

4.4 Computational Processes of a Generative Design Agent

Using the above representations, the five computational processes of a GDA can be defined as below. Figure 4.7 illustrates these processes.

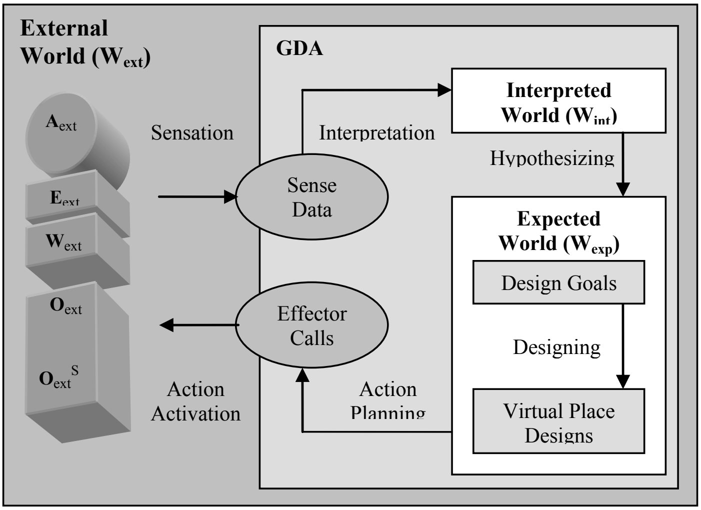


Figure 4.7: Five computational processes of a GDA.

In the process of sensation, a GDA retrieves relevant raw data from the external world to prepare for the processes of interpretation.

In the process of interpretation, the raw sense data are filtered, focused and transformed to construct the GDA's interpreted world W_{int}:

$$W_{ext} \rightarrow W_{int} \quad (4.7)$$

$$W_{ext} = A_{ext} \cup E_{ext} \cup wA_{ext} \cup O_{ext} \quad (4.2)$$

$$W_{int} = A_{int} \cup E_{int} \cup wA_{int} \cup O_{int} \quad (4.3)$$

More importantly, the GDA further interprets the current design needs in the virtual world N and the current state of the virtual world sT.

$$N = \tau(W_{int})$$

$$sT \in W_{int}$$

In hypothesizing, the GDA sets up goals in its expected world W_{exp} to eliminate or reduce the mismatches between N and sT:

$$W_{int} \rightarrow W_{exp} \quad (4.8)$$

$$W_{exp} = A_{exp} \cup E_{exp} \cup O_{exp} \quad (4.4)$$

There can be at least three different types of goals hypothesized by the GDA. The first type of goals are design goals - goals that are related to designing in virtual worlds, which are the focus of this book. These design goals are represented by expected functions O_{exp}^F and expected behaviors O_{exp}^B:

$$O_{exp}^{F} = \{O_{exp\_1}^{F}, O_{exp\_2}^{F}, \dots, O_{exp\_n}^{F}\} \quad (4.9)$$

$$O_{exp}^{B} = \{O_{exp\_1}^{B}, O_{exp\_2}^{B}, \dots, O_{exp\_n}^{B}\} \quad (4.10)$$

for any expected function $O_{exp\_i}^{F}$ and expected behavior $O_{exp\_i}^{B}$:

$$O_{exp\_i}^{F} \in \tau(N, sT)$$

$$O_{exp\_i}^{B} = \tau(O_{exp\_i}^{F})$$

Sometimes, to eliminate or reduce the mismatches between N and sT requires not only virtual place designs, but also other changes in the virtual world. The second type of goals involve initiating changes of attributes to the GDAs or other avatars in the virtual world: A_{exp}. The third type of goals involve initiating events that occur in the virtual world: E_{exp}.

$$A_{exp} \in \tau_{A}(N, sT)$$

$$E_{exp} \in \tau_{E}(N, sT)$$

In designing, the GDA applies its generative design grammar to generate a place design in the virtual world for the moment in order to satisfy the current design goals. The design is represented by expected structures O_{exp}^{S}:

$$O_{exp}^{S} = \{O_{exp\_1}^{S}, O_{exp\_2}^{S}, \dots, O_{exp\_n}^{S}\} \quad (4.11)$$

where any expected structure $O_{exp\_i}^{S} = \tau(O_{exp\_i}^{F}, O_{exp\_i}^{B})$.

The processes of action include action planning and action activation. The GDA first plans actions for implementing the generated virtual place design: O_{exp}^{S}, and for realizing other initiated changes: A_{exp} and E_{exp}. Next, these planned actions are activated via the GDA's effectors in the virtual world:

$$O_{exp}^{S} \rightarrow O_{ext}^{S} \quad (4.12)$$

$$A_{exp} \rightarrow A_{ext} \quad (4.13)$$

$$E_{exp} \rightarrow E_{ext} \quad (4.14)$$

The process of sensation is straight forward. Interpretation, hypothesizing, designing and action are further elaborated and illustrated below. These five computational processes form a recursive loop. New creations and changes in the virtual world will trigger the GDAs to start a new cycle of reasoning and designing. In this manner, virtual worlds are able to be automatically and dynamically designed, implemented and manipulated as needed, adapting to their uses.

4.4.1 Interpretation

At any given moment, a virtual world comprises various components that construct the environments, the representations of its users, and events occurred in the world. Interpretation has three separated processes. They are marked as 1, 2 and 3 in Figure 4.8. The GDA's interpreted world W_{int} is constructed based on the external world W_{ext}:

$$W_{ext} \rightarrow W_{int} \qquad (4.7)$$

$$W_{ext} = A_{ext}\ U\ E_{ext}\ U\ wA_{ext}\ U\ O_{ext} \qquad (4.2)$$

$$W_{int} = A_{int}\ U\ E_{int}\ U\ wA_{int}\ U\ O_{int} \qquad (4.3)$$

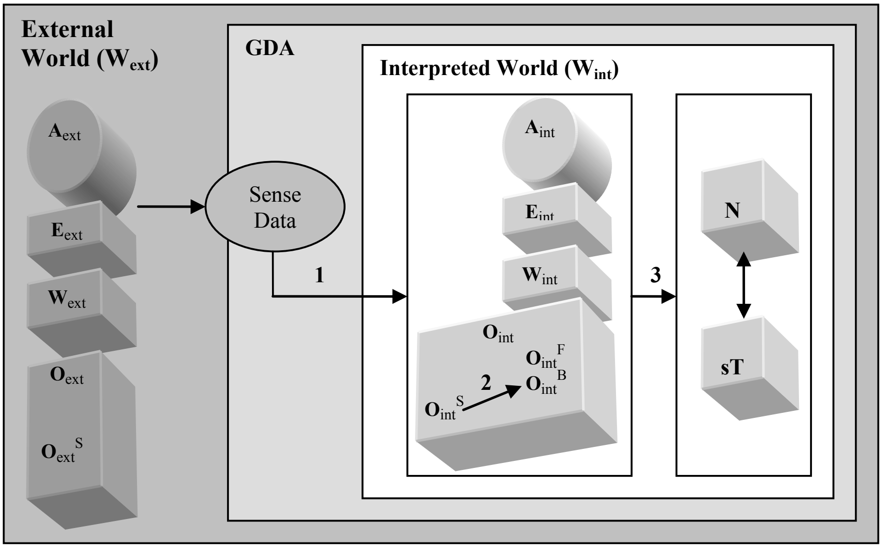


Figure 4.8: Three processes of interpretation.

In the first process of interpretation, the sense data from the external world W_{ext} are filtered, focused and transformed into information that are relevant and can be understood by the GDA:

$$A_{int} = \tau(A_{ext})$$

$$E_{int} = \tau(E_{ext})$$

$$wA_{int} = \tau(wA_{ext})$$

$$O_{int\_i}{}^{S} = \tau(O_{ext\_i}{}^{S})$$

In the second process of interpretation, for any virtual world object, the interpreted functions $O_{int\_i}^{F}$ and interpreted behaviors $O_{int\_i}^{B}$ are derived from the interpreted structures $O_{int\_i}^{S}$.

$$O_{int\_i}^{B} = \tau(O_{int\_i}^{S})$$

$$O_{int\_i}^{F} = \tau(O_{int\_i}^{B})$$

In the final process of interpretation, the current design needs in the virtual world N, and the current state of the virtual world sT, are interpreted based on the information gained from the first two processes:

$$N = \tau(W_{int})$$

$$sT \in W_{int}$$

To support each computational process, a GDA's reasoning mechanism is developed as a general rule base, where each rule can be expressed using the formula of "IF… THEN…". In the processes of interpretation, the following formulas are applied to develop rules for interpreting A_{int}, E_{int}, wA_{int}, $O_{int\_i}^{S}$, O_{int}^{B}, O_{int}^{F}, N and sT. Symbol C used in the formulas represents certain general conditions. To demonstrate the uses of these formulas, each formula is presented with an example rule.

Formula 1:	**Example rule:**
IF A_{ext} AND C ($C \in W_{ext}$) THEN A_{int}	IF GDA A is sensed in the virtual world. AND User A is set as the artist character in the virtual gallery. THEN It is interpreted that the artist is present in the virtual world.
Formula 2:	**Example rule:**
IF E_{ext} AND C ($C \in W_{ext}$) THEN E_{int}	IF An object is mouse-clicked by the artist. THEN It is interpreted that the artist is examining the object.
Formula 3:	**Example rule:**
IF wA_{ext} AND C ($C \in W_{ext}$) THEN wA_{int}	IF The system time of the virtual world is between 6pm to 6am. THEN It is interpreted as "night" time in the virtual world.
Formula 4:	**Example rule:**
IF $O_{ext\_i}^{S}$ AND C ($C \in W_{ext}$) THEN $O_{int\_i}^{S}$	IF An object is configured with a script for displaying digital images. THEN The structure of the object is interpreted as having a programming component for displaying digital images.

Formula 5:	Example rule:
IF $O_{int\_i}^{S}$ AND C ($C \in W_{ext}$) THEN $O_{int\_i}^{B}$	IF The structure of an object is interpreted as having: A visual component of a 3D model with a flat surface. A programming component for displaying digital images. THEN The behavior of the object is interpreted so as to display digital images.

Formula 6:	Example rule:
IF $O_{int\_i}^{B}$ AND C ($C \in W_{ext}$) THEN $O_{int\_i}^{F}$	IF The behavior of the object is interpreted so as to display digital images. THEN The function of the object is interpreted as a digital picture frame.

Formula 7:	Example rule:
IF W_{int} THEN N	IF The artist is present in the virtual world. AND The artist has one exhibition to display. AND Visitors are present in the virtual world. THEN A gallery area is needed for displaying the exhibition.

Formula 8:	Example rule:
IF W_{int} THEN sT	IF No gallery area is recognized in the virtual world. THEN The current state of the virtual world is interpreted as having no exhibition place available.

4.4.2 Hypothesizing

In the process of hypothesizing, the GDA establishes design goals in its expected world W_{exp} in order to eliminate or reduce the mismatches between the current design needs (N) and the current state of the virtual world (sT). The process of hypothesizing is illustrated in Figure 4.9.

$$W_{int} \rightarrow W_{exp} \tag{3.8}$$

$$W_{exp} = A_{exp} \cup E_{exp} \cup O_{exp} \tag{3.4}$$

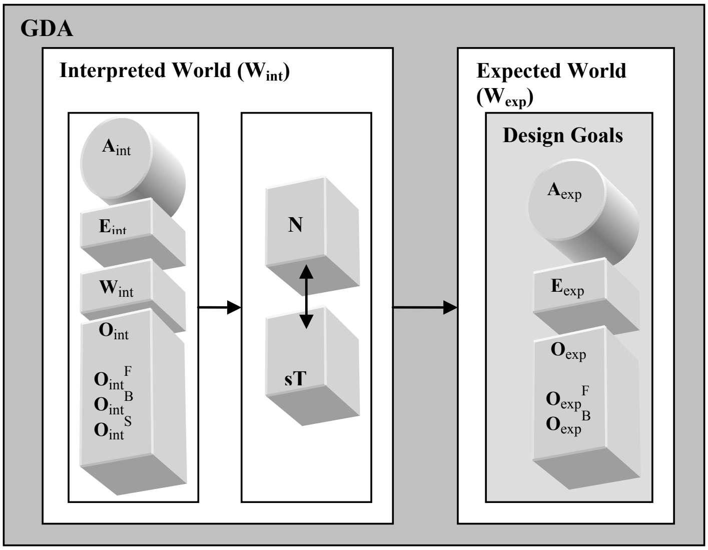


Figure 4.9: The process of hypothesizing.

An object in the GDA's expected world may or may not have a counterpart in the interpreted world, since a virtual place design can be generated based either on existing structures or by new creations. There can be at least three different types of goals hypothesized by the GDA. The first type of goals are design goals for designing virtual worlds. These goals are represented by expected functions O_{exp}^F and expected behaviors O_{exp}^B:

$$O_{exp}^F = \{O_{exp\_1}^F, O_{exp\_2}^F, \dots, O_{exp\_n}^F\} \quad (4.9)$$

$$O_{exp}^B = \{O_{exp\_1}^B, O_{exp\_2}^B, \dots, O_{exp\_n}^B\} \quad (4.10)$$

for any expected function $O_{exp\_i}^F$ and expected behavior $O_{exp\_i}^B$:

$$O_{exp\_i}^F \in \tau(N, sT)$$

$$O_{exp\_i}^B = \tau(O_{exp\_i}^F)$$

The second type of goals are used for initiating changes of attributes to the GDAs or other avatars in the virtual world: A_{exp}; for example, to change the location of a GDA or an avatar through hyperlinks. The third type of goals are used for initiating events in the virtual world: E_{exp}; for example, to send a message to users who connect to the virtual world.

$$A_{exp} \in \tau_A(N, sT)$$

$$E_{exp} \in \tau_E(N, sT)$$

In the processes of hypothesizing, the following formulas are applied to develop rules for hypothesizing $O_{exp\_i}^{F}$, O_{exp}^{B}, A_{exp}, and E_{exp}. Symbol C used in the formulas represents certain general conditions. To demonstrate the uses of these formulas, each formula is presented with an example rule.

Formula 9:	**Example rule:**
IF N AND sT THEN O_{exp}^{F}	IF A gallery area is needed for displaying an exhibition. AND The current state of the virtual world is interpreted as having no exhibition place available. THEN The expected function is a gallery area in the virtual world for displaying the exhibition.

Formula 10:	**Example rule:**
IF O_{exp}^{F} AND C ($C \in W_{ext}$) THEN O_{exp}^{B}	IF The expected function is to provide a gallery area in the virtual world for displaying an exhibition. AND The exhibition contains X number of digital images. THEN The expected behavior of the gallery area (to be generated) is to display X number of digital images.

Formula 11:	**Example rule:**
IF N AND sT THEN A_{exp}	IF A visitor is invited to an exhibition. AND The current state of the virtual world is interpreted as: The exhibition is displayed in a gallery area located at (x, y, z). The visitor is outside the visual boundary of the gallery area. THEN The expected location of the visitor is set to (x, y, z).

Formula 12:	**Example rule:**
IF N AND sT THEN E_{exp}	IF A gallery area is needed for displaying an exhibition. AND The current state of the virtual world is interpreted as that a gallery area is available at location (x, y, z). THEN Send a message to inform the artist about the gallery area.

4.4.3 Designing and Action

In the process of designing, a GDA generates a virtual place design for the moment to satisfy its current design goals. The virtual world design is represented as O_{exp}^{S}:

$$O_{exp}^{S} = \tau(O_{exp}^{F}, O_{exp}^{B}) \tag{4.6}$$

The process of designing requires the GDA to have generative power for providing virtual place designs as needed. The design component of the GDA is supported by the application of a generative design grammar. Driven by the GDA's design goals, the GDA applies its generative design grammar to generate a place design in the virtual world, in order to satisfy the relevant design goals or parts of the goals.

The process of designing is followed by the process of action, which includes action planning and action activation. The GDA first plans actions for implementing the generated design, O_{exp}^{S}, and for realizing other initiated changes, A_{exp} and E_{exp}. Next, these planned actions are activated via the GDA's effectors in the virtual world. Figure 4.10 illustrates the process of designing and the process of action.

We have discussed that the basic components of a generative design grammar are design rules. Each design rule can be expressed in the following form:

$$LHO + sL \rightarrow RHO$$

where LHO refers to left-hand-side object, and RHS refers to right-hand-side object. The term "object" used here can mean a virtual world object, a set of virtual world objects or virtual world object properties. Each design rule is associated with certain state labels sL, which represent specific design contexts that are related to the GDA's design goals. In order for a design rule to be applied the following conditions need to be met:

– The LHO of the design rule is recognized by the GDA in the virtual world, and
– The design contexts represented by the state labels sL of the design rule are related to the GDA's current design goals.

To demonstrate how a generative design grammar is applied by the GDA, we use a similar rule example shown in Section 3.4.1. A design rule for generating layouts of a virtual gallery can be illustrated as shown in Figure 4.11.

In Figure 4.11, the LHO of the design rule is a reception area, and in the RHO, a virtual gallery area is added and placed spatially adjacent to the reception area.

Symbol  represents a gallery area in the virtual gallery for displaying exhibitions.

Symbol  represents the reception area of the virtual gallery.

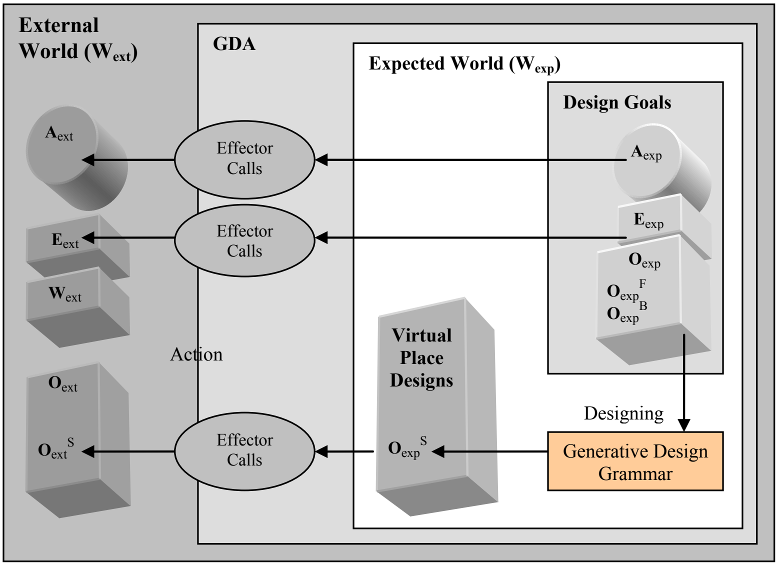


Figure 4.10: The process of designing and the process of action.

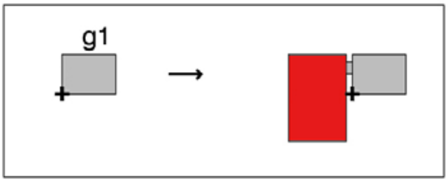


Figure 4.11: An example design rule for generating layouts of a virtual gallery.

The meaning of state label sL=g1 is to generate a certain gallery area in the virtual world. In order for this design rule to be applied, the following two conditions need to be met:

– A reception area  is recognized in the virtual world by the GDA.
– The design context represented by sL=g1 (to generate a certain gallery area in the virtual world) is related to the current design goals of the GDA, represented as O_{exp}^F and O_{exp}^B.

The application of each design rule replaces the LHO with the RHO, which forms the O_{exp}^S or parts of the O_{exp}^S. An example grammar for dynamic design of a virtual gallery is presented in Chapter 5, with its application by a GDA demonstrated in Chapter 6 using a design scenario.

4.5 Generative Design Agents for Designing in 3D Virtual Worlds

A GDA is a computational agent, which has five computational processes defined specifically for reasoning and designing in 3D virtual worlds. The design component of a GDA is supported by the application of a generative design grammar. Applying the GDA model, rule-based place design in 3D virtual worlds can be automated for the current needs. Virtual worlds can be distinguished from the physical world and other static virtual worlds that are pre-defined prior to their uses, and become dynamic as the GDA model enables virtual places to be designed, implemented and manipulated as needed, adapting to their uses.

4.5.1 Generative Design Agent Model as a Design Model

The GDA model is a design agent model. Each GDA is capable of reasoning, designing and manipulating the virtual world as needed. The Function-Behavior-Structure (F-B-S) framework (Gero, 1990) uses F, B and S as variables to describe designing in general. Eight processes are depicted from the framework. They are marked as 1 to 8 in Figure 4.12 and named formulation (1), synthesis (2), analysis (3), evaluation (4), documentation (5) and three different kinds of reformulation (6, 7 and 8) accordingly.

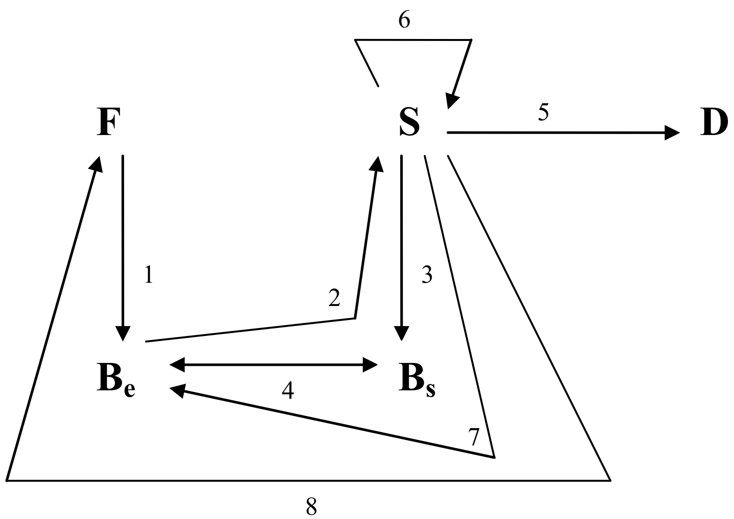


Figure 4.12: The F-B-S framework for describing designing (after Gero, 1990).

In Figure 4.12, symbol F represents functions, B_e represents expected behaviors, S represents structures, B_s represents actual behaviors that derive from the structures, and D represents design descriptions. The eight processes proceed in the following order:

- Designing starts with formulation, where the design requirements (understood in terms of functions) is transformed into the expected behaviors.
- In synthesis, the design solution is provided in the form of structures, which intend to support the expected behaviors.
- In analysis, the actual behaviors are derived from the structures.
- In evaluation, the actual behaviors are compared with the expected behaviors.
- If the actual behaviors are evaluated to be satisfactory, documentation will proceed to produce design descriptions.
- If the actual behaviors are evaluated to be unsatisfactory, appropriate reformulation will proceed to adjust relevant elements and restart the process.

The GDA model as a design model is developed based on the above understandings, but with adaptation for designing in 3D virtual worlds. As discussed earlier, at any moment a GDA views a virtual world W as the union of the external world W_{ext}, the interpreted world W_{int}, and the expected world W_{exp}:

$$W = W_{ext} \cup W_{int} \cup W_{exp} \quad (4.1)$$

Place design in virtual worlds can be represented as interactions among W_{ext}, W_{int} and W_{exp}. There are four processes depicted from the framework of designing virtual worlds. They are marked as 1 to 4 in Figure 4.13: hypothesizing (1), designing (2), action (3), and (re)interpretation (4).

Firstly, in hypothesizing, the GDA sets up design goals in terms of the expected functions O_{exp}^F and the expected behaviors O_{exp}^B, in order to eliminate or reduce the mismatches between the current design needs in the virtual world N, and the current state of the virtual world sT, interpreted by the GDA.

$$O_{exp}^F = \{O_{exp\_1}^F, O_{exp\_2}^F, \dots, O_{exp\_n}^F\} \quad (4.9)$$

$$O_{exp}^B = \{O_{exp\_1}^B, O_{exp\_2}^B, \dots, O_{exp\_n}^B\} \quad (4.10)$$

for any expected function $O_{exp\_i}^F$ and expected behavior $O_{exp\_i}^B$:

$$O_{exp\_i}^F \in \tau(N, sT)$$

$$O_{exp\_i}^B = \tau(O_{exp\_i}^F)$$

Secondly, in designing, the GDA applies its generative design grammar to provide a virtual place design for the moment to satisfy its current design goals or parts of the goals.

$$O_{exp}^S = \tau(O_{exp}^F, O_{exp}^B)$$

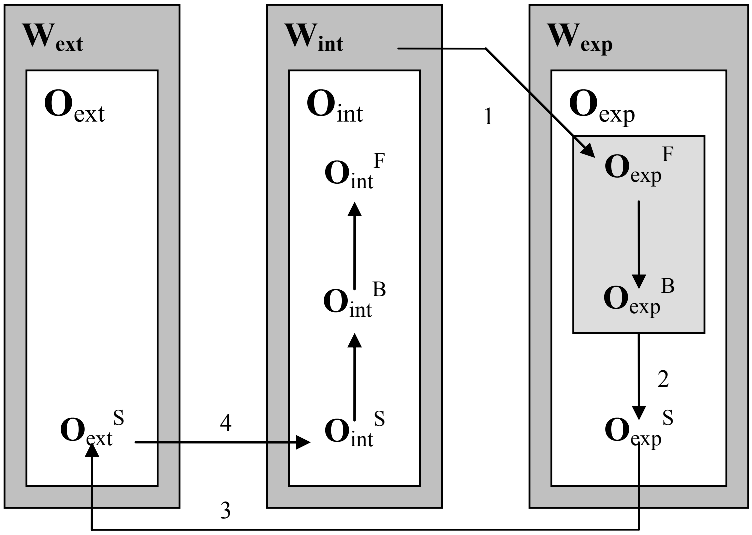


Figure 4.13: The F-B-S framework adapted for representing designing in 3D virtual worlds.

Thirdly, in action, the actions for implementing the generated virtual world design are planned, and the planned actions are activated in the virtual world via the GDA's effectors.

$$O_{exp}^{S} \rightarrow O_{ext}^{S} \tag{4.12}$$

The implementation of the generated place design in the virtual worlds causes further changes to the virtual world, which triggers (re-)interpretation.

$$O_{int} = \tau(O_{ext})$$

For any interpreted virtual world object O_{int_i}:

$$O_{int\_i}^{S} = \tau(O_{ext\_i}^{S})$$

$$O_{int\_i}^{B} = \tau(O_{int\_i}^{S})$$

$$O_{int\_i}^{F} = \tau(O_{int\_i}^{B})$$

The GDA interprets the new sense data from the virtual world. If the newly interpreted N and sT has no mismatch, designing in the virtual world will terminate for the moment. Otherwise, a new cycle of hypothesizing, designing and action will start.

Comparing Figure 4.12 with Figure 4.13, in place design for virtual worlds, by applying the GDA model, the design life-cycle is shortened. This highlights the characteristics of such a kind of virtual world as a different kind of architecture that exists for the moment. They are responsive and proactive to changes in the environments.

- Using the GDA model, designing in virtual worlds becomes inseparable from its implementations. Design descriptions generated by the grammar application are only stored temporarily in the GDA's expected world before being implemented in the virtual worlds.
- Compared to designing in general, designing in virtual worlds using the GDA model does not explicitly specify design evaluation and reformulation. The original processes of evaluation and reformulation are integrated into (re) interpretation. After a virtual place design is implemented, the GDA interprets the actual implemented design, and starts a new cycle of hypothesizing, designing and action to make design adjustments, if necessary.

4.5.2 Generative Design Agents for Designing in 3D Virtual Worlds

The characteristics that make the GDA approach to designing in virtual worlds distinctive are summarized below.

In terms of representation:

- The GDA model is especially suitable for reasoning about and designing dynamic, non-static virtual worlds. For each GDA, the representation of a virtual world is constructed in terms of three different views: $W = W_{ext} \cup W_{int} \cup W_{exp}$.

In terms of designing:

- Different from the conventional approach to designing in virtual worlds, and using the GDA model, firstly the design and implementation processes are automated. Secondly, place design in virtual worlds becomes an internal process that occurs inside the worlds and adapts to their uses. Previously, these processes were mainly controlled by human designers outside the worlds.
- In contrast to other agent-based approaches where the agency is provided to existing virtual world components like walls, doors, rooms, buildings and so on, the GDA approach provides the agency to the users of the virtual world. Shifting the agency from existing virtual world components to people frees virtual worlds from being pre-defined. In the GDA approach, rational design agents are applied to automatically and dynamically design places in virtual worlds for different purposes in different styles during their uses, rather than being limited to reasoning about and modifying existing places in the virtual worlds.

In terms of the roles of human designers:

- In most virtual worlds that currently exist, the design tasks rely heavily on human designers. In general, the designers produce virtual place designs and implement them to support different activities. In later stages, the implemented designs may be reused, modified or demolished by the designers for other purposes. However, in these virtual worlds the general virtual community, who inhabit the environments can rarely influence the design process and make changes to the worlds during their uses. In some virtual worlds where the general community members are given permission to freely design and construct virtual places, due to the lack of design knowledge and design assistance, chaos can often be created in the virtual worlds in terms of design planning, unity and style.

– Using the GDA model, human designers define generative design grammars that produce different design languages for the virtual worlds, rather than pre-define every detail of all possible places in the virtual world. The actual design tasks are carried out by the GDAs during real-time interactions in the worlds. GDAs can apply different generative design grammars to generate designs for different purposes and in different style.

The GDA model presented in this book focuses on its support for rule-based place design in virtual worlds. This book hence focuses on detailing the five computational processes of the model, which are specifically developed for a GDA to reason, design and act in the virtual worlds. Designing in virtual worlds has so far been explored in terms of individual processes by each GDA; for example, when a design problem arises in a multi-GDA virtual world it is assumed that a dominant GDA will reason, design and act on behalf of all users in that virtual place.

However, the reasoning mechanism of the GDA model enables each GDA to reason about the virtual world as well as other GDAs in the world. The GDA model is applicable to collaborative design or collective design (Maher et al, 2010) scenarios in virtual worlds if agent communication is established. Although agent communication is beyond the scope of this book, it is one of the major future development areas for designing in virtual worlds. These will be discussed further in the end of the book.

References

Ferber, J. (1999), *Multi-agent Systems: An Introduction to Distributed Artificial Intelligence*, Addison-Wesley, England.

Gero J.S. (1990), Design Prototypes: a Knowledge Representation Schema for Design, *AI Magazine*, (11)4, pp. 26-36.

Gero J.S. and Kannengiesser U. (2004), The Situated Function-Behaviour-Structure Framework, *Design Studies*, 25, pp. 343-351.

Huhns, M.N. and Stephen, L. (1999), Multiagent Systems and Society of Agents, in G. Weiss (ed), *Multiagent Systems, a Modern Approach to Distributed Artificial Intelligence*, MIT Press, Cambridge MA, pp.79-120.

Maher, M.L. and Gero J.S. (2002), Agent Models of Virtual Worlds, *Proceedings of ACADIA 2002*, California State Polytechnic University, pp. 127-138.

Maher, M.L., Gero, J.S., Smith, G. and Gu, N. (2003), Cognitive Agents in 3D Virtual Worlds, *International Journal of Design Computing*, 6 (online).
Maher, M.L., Paulini, M. and Murty, P. (2010), Scaling Up: From Individual Design to Collaborative Design to Collective Design, *Proceedings of DCC 10*, Stuttgart, Germany, pp. 581-600.
Russell, S. and Norvig, P. (2009), *Artificial Intelligence: A Modern Approach* (Third Edition), Prentice Hall, Englewood Cliffs, NJ.
Wooldridge, M. (2000), *Reasoning About Rational Agents*, MIT Press, Cambridge, MA.
Wooldridge, M. and Jennings, N.R. (1995), Intelligent Agents: Theory and Practice, *Knowledge Engineering Review*, 10(2), pp. 115-152.

Part III: **An Adaptive Virtual Gallery**

5 A Generative Design Grammar for a Virtual Gallery

This chapter presents a generative design grammar for an adaptive virtual gallery. Developed using the generative design grammar framework, this example grammar has four sets of design rules to address different design phases of the virtual gallery: layout rules, object design rules, navigation rules and interaction rules. The grammar is intended to be the design component of a Generative Design Agent (GDA) representing an artist in a 3D virtual world.

5.1 An Adaptive Virtual Gallery

A virtual gallery is a typical place in virtual worlds that should be designed dynamically to adapt to its use, instead of being pre-defined; changes in the requirements of the design occur often, for example, changes in the types of activities, changes in exhibition requirements, changes in the number of visitors, and so on. Designing a virtual gallery using a generative design grammar enables the gallery to be dynamic and adaptive – different designs of the gallery being generated as needed for different moments, accommodating various changes in real time.

5.1.1 Galleries: Blending the Physical and the Virtual

Historically speaking, galleries and museums were not initially made for the public. Nowadays, they form central parts of public places in society. Public galleries and museums have gradually gone beyond the original purposes of artwork display and appreciation to provide public places supporting cultural exchange, art education and social gathering. To design a gallery or a museum, therefore, is not only to design exhibition spaces, it is also to enable the formation of community, the establishment of social values and the interactions among artists, visitors, and artworks.

The development of galleries and museums is always closely influenced by the evolution of our society, such as the advancement of economics, politics, science and technology. For example, the majority of galleries and museums are currently open to the public and serve their interests. However, most major galleries and museums are based on private collections, which were established long before the formation of these public institutions (Newhouse, 1998). This clear shift from private places to public places was mainly due to the modern movements of economics and politics in our society. The emergence of new technologies also had a great impact on the development of galleries and museums:

– New technologies provoke new art forms. They inspire artists, offer new topics and perspectives for art thinking, and provide new design tools and media for art making. For example, the invention of moving images at the beginning of the

20th Century was enriched by the development of different media throughout the century. This eventually led to the wide recognition of the idea of media art since the 1990s (Schwarz, 1997).
- New technologies enable new designs of exhibition places. There are at least two purposes for developing new designs of exhibition places. Firstly, advanced building materials and techniques can improve the quality of exhibition places by providing, for example, better lighting and ventilation. Secondly, new designs of exhibition places are needed in order to accommodate emergent art forms, for example, venues such as the Centre for Art and Media Karlsruhe[1] in Germany is specifically designed for displaying media art.

More recently, with the advancement of digital and networked technologies, galleries and museums have become ready for the next reformation blending the physical and virtual presence. Responding to the development and application of these technologies, galleries and museums have started to use internet environments to display digital replicas from their collections, attracting a much wider range of visitors.
- A virtual gallery provides easy access and flexible viewing. People can visit the gallery without concern for time and geographical distance.
- Exhibitions in a virtual gallery are not constrained by the gallery's physical capacity. For example, the virtual Getty Museum[2] provides access to all its collections, which is rarely the case for most galleries in the physical world.
- Some interactive installations enable visitors to participate in the art creation process and interact with the installations from the internet interfaces. This is a new form of blending the physical and virtual exhibition places for supporting emergent art forms.

However, the majority of virtual galleries rely heavily on their physical counterparts. Some serve as electronic databases of digital replicas based on existing collections from the physical world. Others use digital media to simulate galleries from the physical world in order to provide virtual tours. These virtual galleries serve the purpose of supplementing existing galleries in the physical world. However, they do not reflect the full potential of virtual galleries. The future of virtual galleries should not be limited to the expansion of the electronic databases or the refinement and optimization of the 3D simulations.
- A virtual gallery can provide exhibition places for digital and network-based interactive artworks that cannot be accommodated by a gallery in the physical world.

---

1 http://on1.zkm.de/zkm/e
2 http://www.getty.edu

- A virtual gallery can provide a different kind of experience. Unlike conventional galleries, which provide largely passive viewing, a virtual gallery as a networked environment can be flexibly programmed to be reactive, and even proactive. Virtual galleries can provide highly personalized experiences for visitors in terms of artwork selection and viewing, as well as the design of the exhibition place and format. The traditional relationships among the artists, the artworks, the visitors and the exhibition places can be further challenged.
- Similar ideas have been presented in many conceptual designs; for example, the Guggenheim Virtual Museum by Asymptote Architects[3] and various entries for the New Italian Blood Virtual Museum Competition[4]. However, in order to realize these ideas, many design and technical issues need to be resolved.

More recently with the increasing popularity of social networks and social computing, we are witnessing an increasing use of a wide variety of social media including Facebook[5], Instagram[6], Pinterest[7], Youtube[8], blogs and wikis etc. for both art organizations and individual artists to express and promote their artistic vision and interest. These examples include Facebook pages such as Callum Docherty/Art[9] used by individual artists as their personal virtual studio and gallery, blogs created and maintained by well-established art cooperative agencies such as Magnum Photos[10], and open-source virtual community for museum design[11] set up by professional organizations such as the Tech Museum of Innovation. Virtual galleries using 3D virtual worlds as the base technology have the potential to integrate these different media and simultaneously explore the quality of virtual gallery as place through the use of the place metaphor.

5.1.2 A Virtual Gallery Adapted for Changing Needs

The virtual gallery grammar described in Chapter 3 is further developed here as a component of a design agent to illustrate an adaptive virtual gallery that demonstrates the potential of a dynamic and proactive networked environment, rather than the simulation of a gallery from the physical world.

3 http://www.asymptote.net
4 http://www.newitalianblood.com/virtualmuseum
5 https://www.facebook.com
6 http://instagram.com
7 https://pinterest.com
8 http://www.youtube.com
9 https://www.facebook.com/pages/Callum-DochertyArt/185636154945505
10 http://www.magnumphotos.com
11 http://thetechopensource.thetech.org

Our virtual gallery is designed for an aspiring artist as personal studio and an exhibition space. The first purpose of the gallery is to provide online places to display the artist's work and to serve as the main portal to access the artist's online presence similar to the purpose that social media or a web site now provides. The second purpose is to provide alternative places supporting other activities of the artist in the virtual gallery; for example, art creation, collaboration, meeting, lecture and other public functions. The artist is provided with the agency in the virtual world, represented as a Generative Design Agent (GDA). The GDA assists the artist by reasoning, designing and acting on behalf of the artist in the virtual world. The design component of the artist's GDA is realized by the application of this example grammar.

A conventional virtual gallery may provide similar kinds of exhibitions and alternative places, supporting similar kinds of activities, but the difference is that our virtual gallery is adaptive – dynamically designed as needed. Similar to galleries in the physical world, the design of a conventional virtual gallery and the arrangement of its exhibitions are pre-defined prior to its use. The resultant virtual gallery serves certain specific purposes, but does not take into consideration possible changes to the purposes during its use, changes which often occur when the artist, visitors and their avatars interact with each other and with the gallery. In our virtual gallery, different designs of the gallery are dynamically generated, and different arrangements of the exhibitions are dynamically provided to respond to various changes that occurr during the use of the gallery, to better server the needs and to optimize the individual experiences of different parties. Such changes can be initiated from the following sources:

- From the artist: changes of exhibition requirements, changes of activities and changes of design preferences.
- From the visitors: changes of activities and changes of group dynamics.

The visual forms of the virtual gallery provide an awareness of locations by defining a 3D ambient environment. In addition to providing access to the artist's online presence on various social media and other networked environments, more importantly the virtual gallery displays both digital replicas of the artist's artworks from the physical world, and digital or network-based exhibitions that cannot be displayed in galleries from the physical world. The exhibitions are arranged and allocated in different gallery areas for viewing and interaction rather than being integrated into an electronic database or a set of web pages for browsing. Visitors are represented as avatars in the virtual gallery, providing an awareness of self and others. Through these avatars, visitors are able to explore the virtual gallery and move from area to area for different activities online. With the assistance of the GDA, the dynamic design of the gallery places and the arrangement of exhibitions are automated, adapting to their uses.

5.2 Overview of an Example Grammar

Applying the generative design grammar framework, our example grammar is developed to have four sets of design rules applied in the sequence of layout rules, object design rules, navigation rules and interaction rules. Each set of these design rules corresponds to a design phase of the virtual gallery.

- The application of layout rules provides layouts of the virtual gallery by allocating different areas. Each area has a purpose for accommodating certain intended activities in the virtual gallery.
- The application of object design rules configures each area of the gallery with certain purposeful objects that provide visual boundaries of the area and visual cues for supporting the intended activities.
- The application of navigation rules specifies navigation methods in the gallery by using way-finding aids and hyperlinks for assisting the visitors' navigation among different areas, and their visits in the exhibitions.
- The application of interaction rules ascribes scripted behaviors to selected objects in each area of the gallery. Therefore, exhibitions and other intended activities can function.

Each design rule is developed by following the general structure of a design rule as illustrated in (3.4).

$$\text{LHO} + \text{sL} \rightarrow \text{RHO} \tag{3.4}$$

By applying the example grammar under different contexts, different designs of the gallery can be dynamically generated as needed adapting to its use. The corpus of designs generated by the grammar share similar stylistic characterizations in terms of visualization, navigation and interaction determined by the application of layout rules, object design rules, navigation rules and interaction rules.

5.2.1 Design and Composition of a Virtual Gallery

A virtual world design can be viewed as "objects in relations". A design of the virtual gallery, generated by the example grammar, can be analyzed as a composition of virtual world objects and their properties. Using the objects and their properties as design elements, the four sets of design rules compose different aspects of the design. One aspect is in terms of the purposes of the virtual gallery: the application of layout rules allocates various areas of the virtual gallery. Each area has a purpose for accommodating certain intended activities in the gallery. Changes to purposes may require the creation of new areas, adjustment and replacement of existing areas, or removal of existing areas. For the example grammar, the virtual gallery can have five different kinds of purposeful areas. They are combined and allocated as needed. These purposeful areas are introduced in Section 5.3.

Further, each purposeful area of the virtual gallery is supplemented with visual boundaries, visual cues, way finding aids and hyperlinks, and object behaviors for supporting the intended activities through the application of object design rules, navigation rules and interaction rules. These design elements are introduced accordingly in Sections 5.4 to 5.6.

In summary, the GDA, combined with the virtual gallery grammar, dynamically composes designs of a virtual gallery by adding/subtracting different virtual world objects and their properties, adapting to its use. Figure 5.1 is the visualization of one such virtual gallery design.

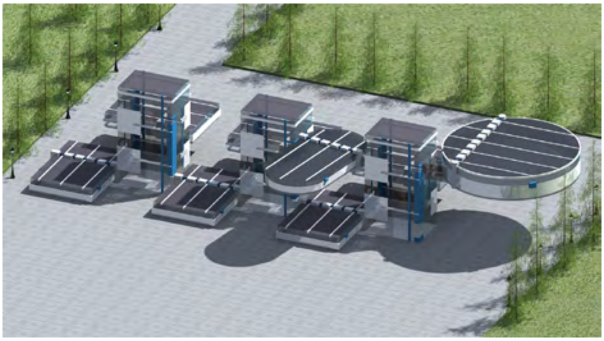


Figure 5.1: The visualization of a virtual gallery design generated by the example grammar.

Later in this chapter, we describe each set of design rules grouped into additive rules and subtractive rules. These two groups of design rules accommodate the two kinds of design operations: addition and subtraction. However, designers should not be limited by these two kinds of operations. They are for demonstration purposes only. Designers can explore and apply other design operations such as replacement and transformation to suit their design preferences and needs.

5.2.2 Design Goals

The virtual gallery is designed for specific purposes. The design generated by the example grammar needs to meet the GDA's current design goals in order to support the intended activities in the virtual gallery for the moment. Generative design grammars connect to the GDAs' design goals via the use of state labels. State labels used in the example grammar are developed to represent a set of typical design contexts, constrained by a virtual gallery scenario to be demonstrated later in the book.

The state labels used in each set of the design rules and the design goals hypothesized by the GDA are discussed in each of the following sections respectively.

5.3 Layout Rules

Layout rules are the first set of design rules to be applied in the application of the example grammar. In our virtual gallery grammar, there are 15 additive layout rules and 17 subtractive layout rules in total. Layout rules generate layouts of the virtual gallery by identifying and allocating different areas. The application of interaction rules will then ascribe scripted behaviors to selected objects in each area of the gallery enabling interactive exhibitions and interactive design infrastructure.

The virtual gallery can have five different areas: the reception area, the gallery 1 area (standard or expanded), the gallery 2 area (standard or expanded), the artist's personal studio area, and the multi-function area. The shapes below are symbolic and indicative, rather than specific geometric models for the objects that will comprise the 3D design.

 a reception area

a corridor

 a standard gallery 1 area

 an expanded gallery 1 area

 a standard gallery 2 area

 an expanded gallery 2 area

 the artist's personal studio area

 the multi-function area

Other symbols used in layout rules are:

a spatial label

the registration mark

The initial design is: 

A reception area provides visitors with information regarding the virtual gallery and the exhibitions. It also provides access to other areas of the virtual gallery. The virtual gallery can have multiple reception areas. They play a vital role for supporting the circulation of this particular design of the virtual gallery. Any other area of the virtual gallery always connects to a reception area. In our example gallery style, a reception area has three levels: it connects to a gallery 1 area from floor 1, to a gallery 2 area from floor 2, and to the artist's personal studio area and the multi-function area from floor 3.

In the example grammar, the artist has a maximum of two exhibitions at a time. A gallery 1 area is designed for exhibition 1, and a gallery 2 area is designed for exhibition 2. The exhibitions may display digital replicas of the artist's artworks from the physical world, as well as digital or network-based artworks that cannot be displayed in galleries in the physical world. In this example grammar, these two kinds of exhibition items are called digital images and interactive installations. A gallery 1 area and a gallery 2 area each can have two sizes: a standard area and an expanded area for displaying exhibitions with different scales. The virtual gallery can have multiple gallery 1 areas and multiple gallery 2 areas as each area is designed for hosting a certain number of avatars. The maximum number of visitors in a gallery area is set to be X for the comfort of the visitors' viewing and the ease of the avatars' movements in the area. These constraints are set for this particular virtual gallery due to our preferences. In practice, they can also be addressed with the use of a parametric grammar to adjust the dimensions of different gallery areas, rather than adding, subtracting and replacing modules of these gallery areas as demonstrated here.

The artist's personal studio area is private; it is for the artist and invited guests only. The artist uses the area to create artworks, to collaborate with other artists, and to meet with invited guests.

The multi-function area can be used for at least two different purposes. The first purpose is to serve as a conference venue for events such as exhibition opening, media conferences and public lectures. The second purpose is to serve as an additional gallery area for visually large-scale installations.

The spatial labels are used to control the application of layout rules. For example,  is the layout of a standard gallery 1 area, and is marked with a spatial label on its left edge. This indicates that a layout rule can be applied once to this symbol to compose the layout of the virtual gallery from this left edge.

5.3.1 State Labels

For the example grammar, the state labels used in layout rules can be roughly divided into three groups: state label 1, state labels used in additive layout rules and state labels used in subtractive layout rules.

sL=1 is used in all layout rules indicating they are the first set of design rules to be applied.

States labels used in additive layout rules are:

sL=S: the personal studio area for the artist is needed in the virtual gallery.

sL=g1: the initial standard gallery 1 area is needed in the virtual gallery.

sL=g2: the initial standard gallery 2 area is needed in the virtual gallery.

sL=g1+: an additional standard gallery 1 area is needed in the virtual gallery.

sL=g2+: an additional standard gallery 2 area is needed in the virtual gallery.

sL=r+: an additional reception area is needed in the virtual gallery.
sL=gE1: a standard gallery 1 area needs to be expanded.
sL=gE2: a standard gallery 2 area needs to be expanded.
sL=mC: the multi-function area is needed as a conference venue.
sL=mI: the multi-function area is needed for exhibiting visually large-scale installations.

State labels used in subtractive layout rules are:
sL=g1-: a standard/expanded gallery 1 area is redundant.
sL=g2-: a standard/expanded gallery 2 area is redundant.
sL=r-: a reception area is redundant.
sL=gR1: an expanded gallery 1 area needs to be reduced.
sL=gR2: an expanded gallery 2 area needs to be reduced.
sL=g-: the initial standard gallery area is not needed.
sL=m-: the multi-function area is not needed.
sL=cS: the current design of the virtual gallery is to be used as a static design the virtual gallery stops being dynamically designed to adapt to its use).

A state label represents a specific design context. The label is matched if the design context it represents is related to the GDA's current design goals in terms of designing the virtual gallery. As discussed earlier in the book, the GDA's design goals can be represented by the expected function O_{exp}^{F} and the expected behaviors O_{exp}^{B}.

For example, to match sL=S, firstly in the process of interpretation, the artist's GDA interprets that:

– The artist is present in the virtual gallery (A_{int}).
– There is no studio space available in the virtual gallery for the artist (O_{int}).

Next in the process of hypothesizing, the GDA hypothesizes the following design goal based on the above interpretations:

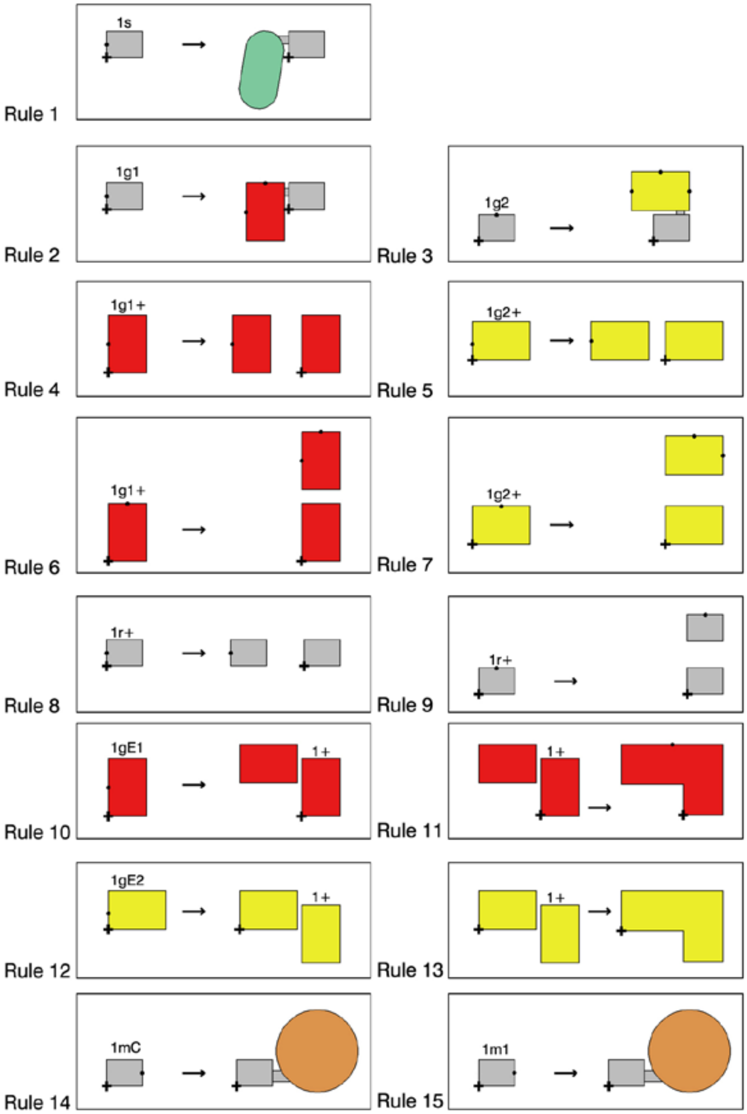
– O_{exp}^{F}=S (a personal studio area for the artist is needed in the virtual gallery).

With the design goal O_{exp}^{F}=S being hypothesized, sL=S is matched. A design rule is selected for application only if the LHO of the rule is found, and the state labels of the rule are matched.

5.3.2 Additive Layout Rules

The application of additive layout rules adds different areas for generating the layout of the virtual gallery. The 15 additive layout rules developed for the example grammar are listed below. The application of an additive layout rule implies that the LHO of the rule can be replaced by the RHO of the rule, where an additional area is added, when the following conditions are met.

– LHO of the rule is recognized in the virtual gallery, and
– sL of the rule is matched.



For example, additive layout rule 1 shows that the artist's personal studio area can be added spatially adjacent to a reception area if:

– The reception area is found in the virtual gallery, and
– sL=S is matched.

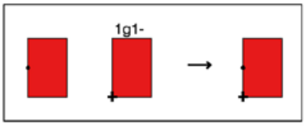
Among these 15 additive layout rules:
- Rules 1, 2, 3, 14 and 15 are applied to add the artist's personal studio area, the initial standard gallery areas and the multi-function area accordingly, adjacent to a reception area.
- Rules 4 to 7 are applied to add additional gallery areas. As mentioned earlier, for the example grammar, each gallery area is designed for hosting a certain number of avatars. Once the limit is reached, an additional gallery area is added for future visitors.
- For the example grammar, a standard gallery area can also be expanded to accommodate more exhibition items. Rules 10 to 13 are developed for this purpose.

5.3.3 Subtractive Layout Rules

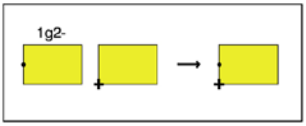
The application of subtractive layout rules removes or reduces different areas from an existing layout of the virtual gallery. The 17 subtractive layout rules developed for the example grammar are listed below. The application of a subtractive layout rule is similar to an additive layout rule except that its purpose is to subtract one or more areas.

For example, subtractive layout rule 16 shows that the artist's personal studio area can be removed if:
- The artist's personal studio area is adjacent to a reception area in the current layout, and
- sL=cS is matched.

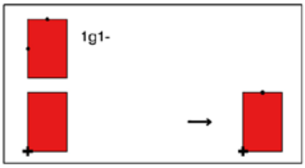
Among these 17 subtractive layout rules:
- Subtractive layout rules 13 to 16 are applied to remove the initial standard gallery areas, the multi-function area and the artist's personal studio area accordingly.
- As shown in additive layout rules, additional gallery areas can be added to accommodate more visitors. Once these areas become redundant, subtractive layout rules 1 to 6 can be applied to remove these additional gallery areas.
- Additive layout rules 10 to 13 enable standard gallery areas to be expanded to accommodate more exhibition items. On the contrary, subtractive layout rules 9 to 12 reduce expanded gallery areas back to the size of standard gallery areas.
- Subtractive layout rule 17 is a termination rule. Spatial labels can be removed by applying this rule so that the application is terminated.



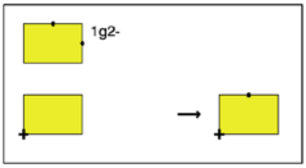
Rule 1



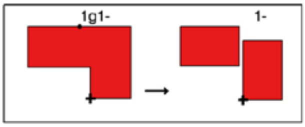
Rule 2



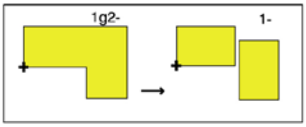
Rule 3



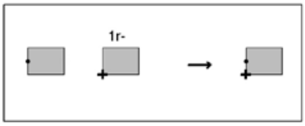
Rule 4



Rule 5

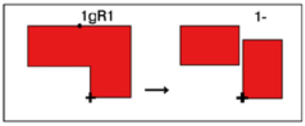


Rule 6

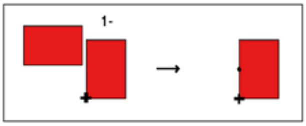


Rule 7

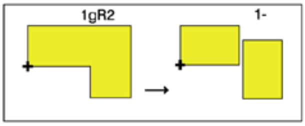
Rule 8



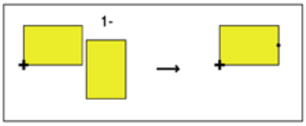
Rule 9



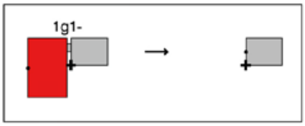
Rule 10



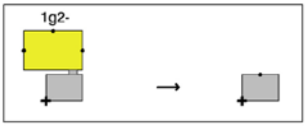
Rule 11



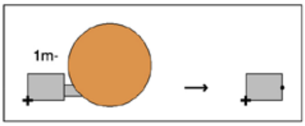
Rule 12



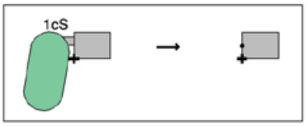
Rule 13



Rule 14



Rule 15



Rule 16

Rule 17

5.4 Object Design Rules

Object design rules are the second set of design rules to be applied in the application of the example grammar. There are 24 additive object design rules and 9 subtractive object design rules in total. After a layout of the virtual gallery is produced, object design rules further configure each area to provide visual boundaries of the area and visual cues for supporting various intended activities through object design and placement. The objects appear in the virtual gallery as various 3D models. Besides the symbols introduced earlier, additional symbols and illustrations used in object design rules are illustrated in Figures 5.2 to 5.10.

Figures 5.2 to 5.5 illustrate the visual boundaries of different areas in the virtual gallery. For the example grammar, the artist prefers a cold-color scheme for the design of the virtual gallery. The following visual boundaries are illustrated using a cold-color scheme.

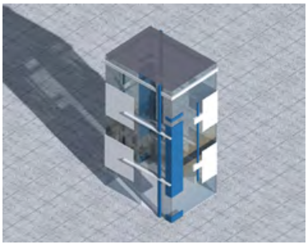


Figure 5.2: The visual boundaries of a reception area.

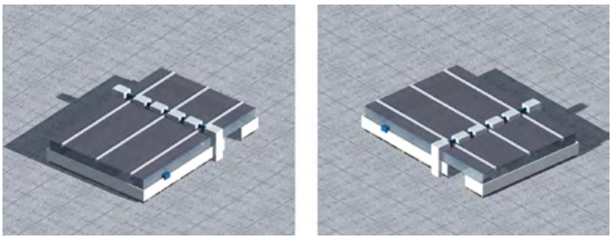


Figure 5.3: The visual boundaries of a standard gallery 1 area (left) and a standard gallery 2 area (right).

Figures 5.6 to 5.10 show various purposeful objects that are designed for the virtual gallery. As shown in Figure 5.6, a helpdesk is placed in a reception area to provide information about the virtual gallery and the exhibitions. The information is visualized by the warm-color cubes to provide a contrast to the ambient environment and they are placed on the frame-like partition behind the desk. Another frame-like partition is

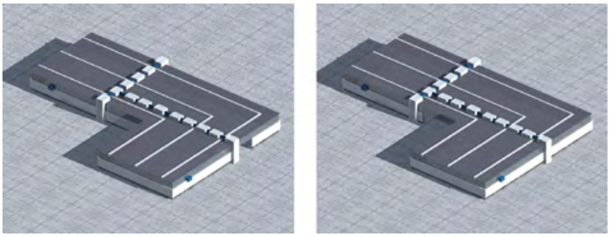


Figure 5.4: The visual boundaries of an expanded gallery 1 area (left) and an expanded gallery 2 area (right).

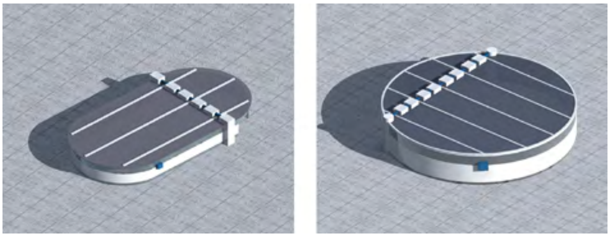


Figure 5.5: The visual boundaries of the artist's personal studio area (left) and the multi-function area (right).

also placed in a reception area for storing hyperlinks for assisting navigation within the virtual world and for providing access to other media and networked environments. They are also visualized by the warm-color cubes. With these hyperlinks, the visitors will be transported to different designated locations in the virtual gallery, or to be re-directed to the desired media and networked environments.

Figure 5.7 shows different objects that are used to arrange the gallery areas and the multi-function area for exhibition purposes.

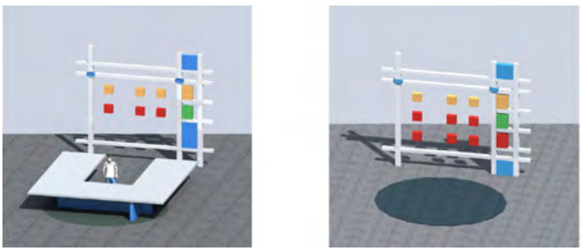


Figure 5.6: A helpdesk (left) and a frame-like partition with hyperlinks (right) in a reception area.

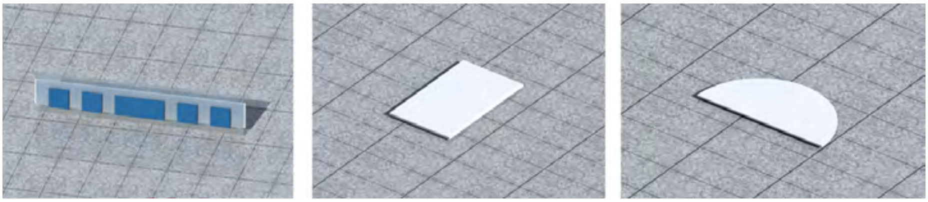


Figure 5.7: A partition for displaying digital images (left), two different stands for displaying interactive installations (middle and right).

In the artist's personal studio area, digital tools and information visualized by warm-color cubes are stored on the frame-like partitions. Figure 5.8 illustrates three different kinds of partitions used for such purposes. The rest of the symbols and illustrations used in object design rules are illustrated in Figures 5.9 and 5.10.

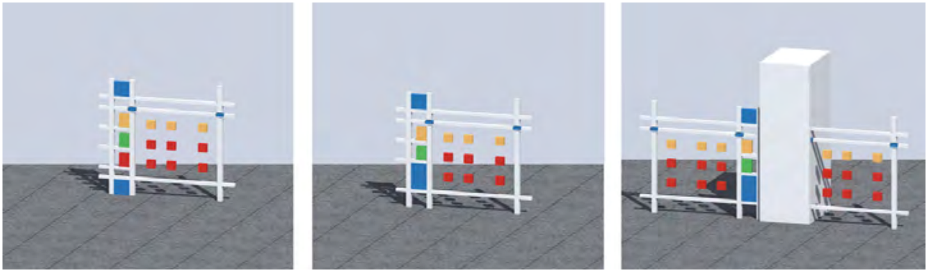


Figure 5.8: Frame-like partitions used in the artist's personal studio area for storage purposes.

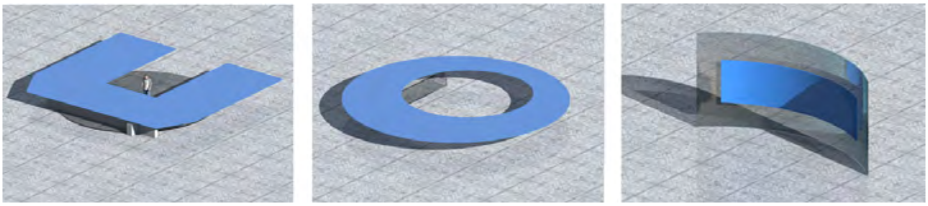


Figure 5.9: A working desk for the artist (left), a meeting table (middle) and a presentation screen (right).

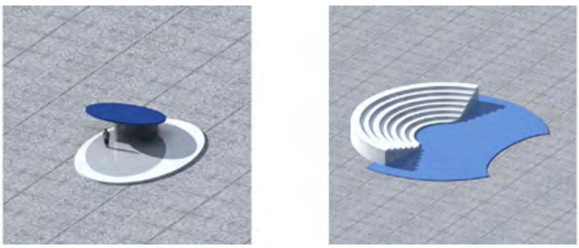


Figure 5.10: Conference facilities: a podium (left) and an auditorium (right).

5.4.1 State Labels

For the example grammar, the state labels developed for object design rules can be roughly divided into two groups: state label 2 and state labels used in additive object design rules. Subtractive object design rules share the same group of state labels, defined earlier in Section 5.3.1, for subtractive layout rules.

sL=2 is used in all object design rules indicating that they are the second set of design rules to be applied.

State labels used in additive object design rules are:
sL=cC: to apply a cold-color scheme for the interior of the virtual gallery.
sL=gIM1: to arrange a gallery area for displaying digital images using configuration 1.
sL=gIM2: to arrange a gallery area for displaying digital images using configuration 2.
sL=gIM3: to arrange a gallery area for displaying digital images using configuration 3.
sL=gIM4: to arrange a gallery area for displaying digital images using configuration 4.
sL=gIS1: to arrange a gallery area for displaying interactive installations using configuration 1.
sL=gIS2: to arrange a gallery area for displaying interactive installations using configuration 2.
sL=gIMS: to arrange a gallery area for displaying both digital images and interactive installations.
sL=mS: to configure a meeting area in the artist's personal studio area.
sL= mMc: to arrange the multi-function area as a conference venue.
sL=mMi: to arrange the multi-function area for exhibiting visually large-scale installations.

For example, to match sL=mS, firstly in the process of interpretation, the artist's GDA interprets that:

- The artist requests a meeting venue for a small group of participants (A_{int} or E_{int}).
- The virtual gallery has no meeting facility (O_{int}).

Next in the process of hypothesizing, the GDA hypothesizes the following design goal based on the above interpretations:

- $O_{exp}{}^{F}$=mS (to configure a meeting area in the artist's personal studio area).

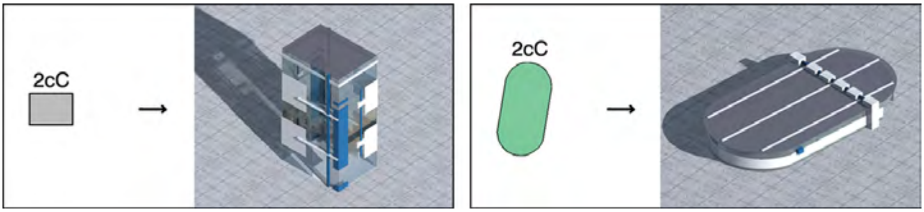
With the design goal $O_{exp}{}^{F}$=mS being hypothesized, sL=mS is matched.

5.4.2 Additive Object Design Rules

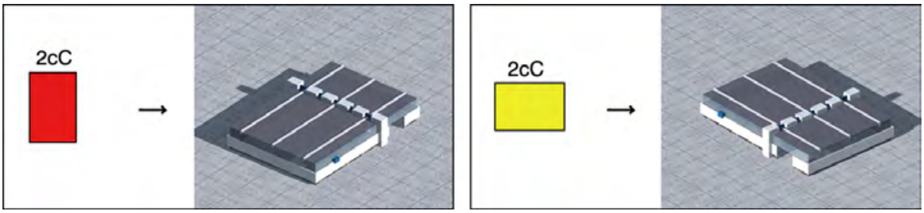
Two kinds of additive object design rules are developed for the example grammar: transforming 2D layout into a 3D environment and arranging 3D objects for specific purposes. To transform the 2D layout of the virtual gallery into 3D environments there are seven additive object design rules. In these rules the LHO represents the layout of an area to be replaced by the RHO representing relevant 3D virtual world objects that define visual boundaries and specify purposeful objects for the area in the virtual gallery.

For example, additive object design rule 3 shows that a standard gallery 1 area can be provided with relevant 3D virtual world objects, to define its visual boundaries, and specify purposeful objects for the area if:

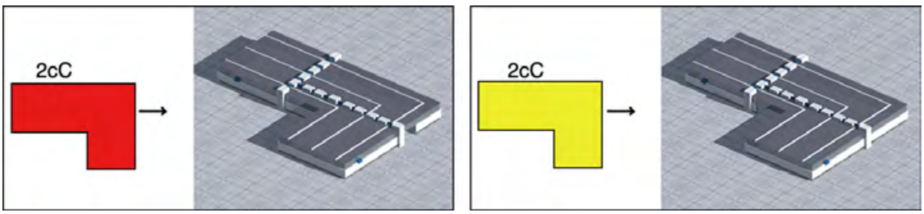
- The layout of a standard gallery 1 area has been generated, and
- sL=cC is matched.
-



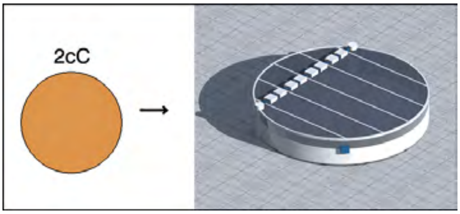
Additive object design rule 1 (left) and 2 (right).



Additive object design rule 3 (left) and 4 (right).



Additive object design rule 5 (left) and 6 (right).



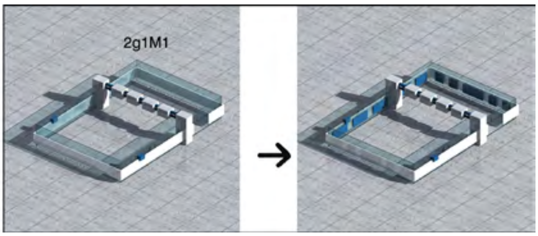
Additive object design rule 7.

The second kind of additive object design rules further arrange each area for different purposes. They are illustrated below from rules 8 to 24. In these illustrations, the roof of each area is removed in order to show the interior of each area.

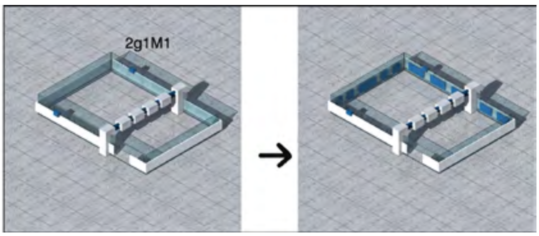
Rules 8 to 21 are applied to arrange the gallery areas for different exhibitions. Our example grammar considers two kinds of exhibition items - digital images and interactive installations - for the virtual gallery. Digital images are rendered on the surfaces of digital picture frame objects. Interactive installations are assemblies of objects with scripted behaviors that make them interactive to the visitors in the virtual gallery.

The following rules provide seven different configurations to arrange gallery areas for exhibitions containing different items and in different scales. For example, additive object design rule 8 shows that a standard gallery 1 area can be arranged for displaying digital images using configuration 1 if:

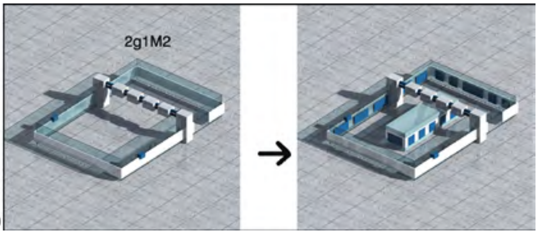
– A standard gallery 1 area is found in the virtual gallery, and
– sL=gIM1 is matched.



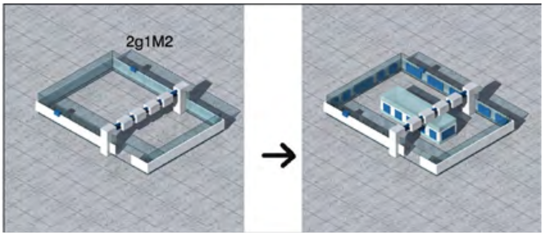
Rule 8



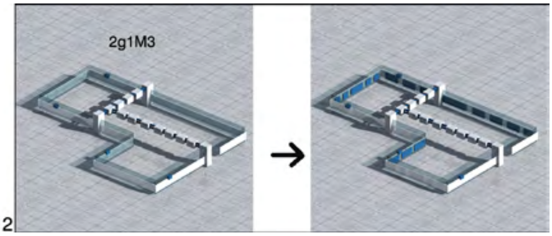
Rule 9



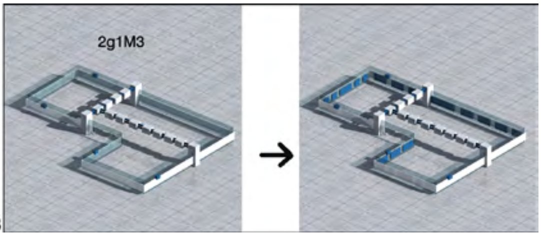
Rule 10



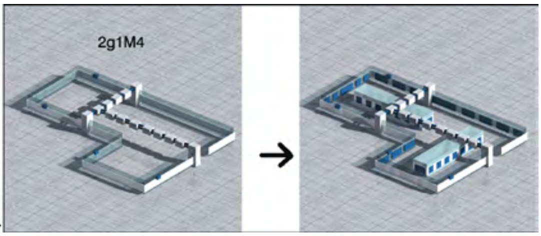
Rule 11



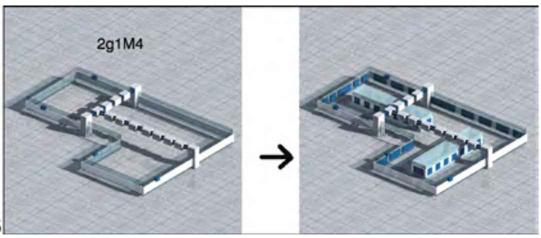
Rule 12



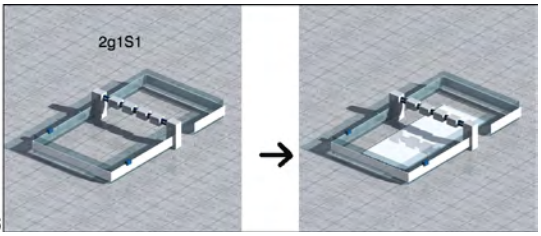
Rule 13



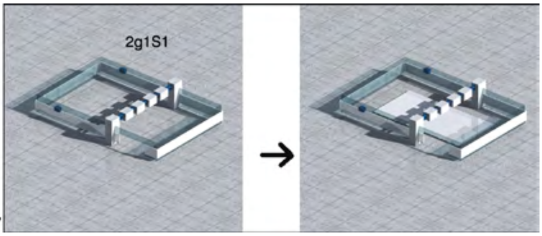
Rule 14



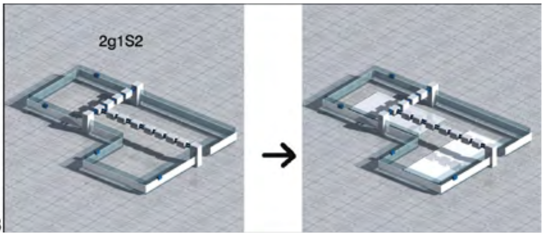
Rule 15



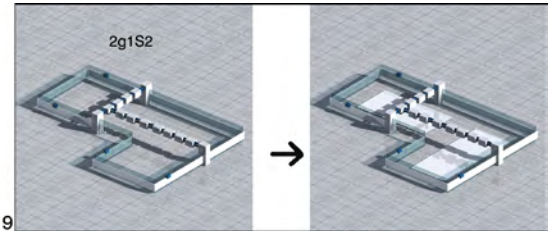
Rule 16



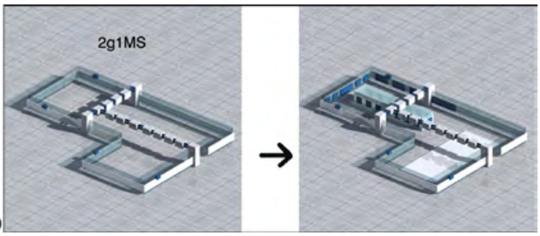
Rule 17



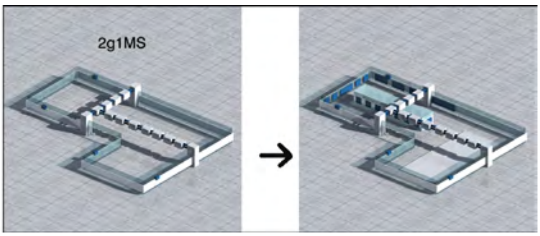
Rule 18



Rule 19

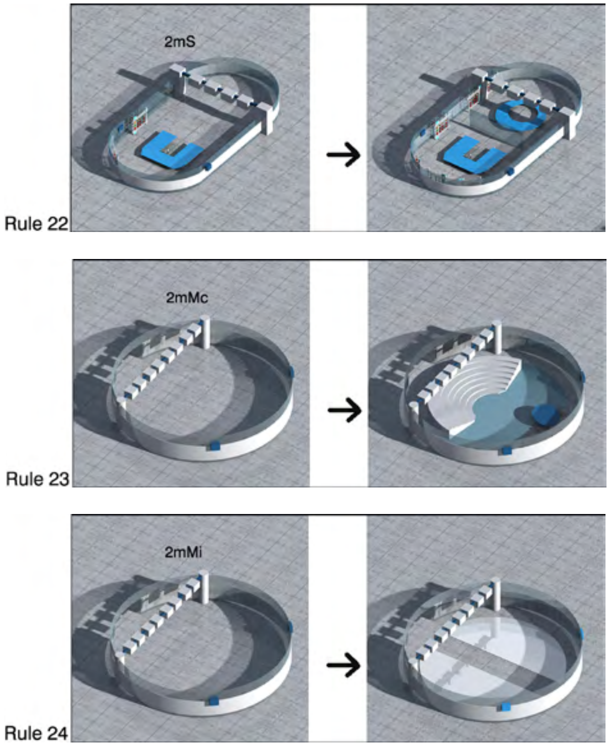


Rule 20



Rule 21

Additive object design rule 22 is applied to arrange a meeting area inside the artist's personal studio area. Additive object design rules 23 and 24 are applied to arrange the multi-function area. Rule 23 arranges the area as a conference venue for various public events serving a larger crowd. Rule 24 arranges the area for exhibiting visually large-scale installations.

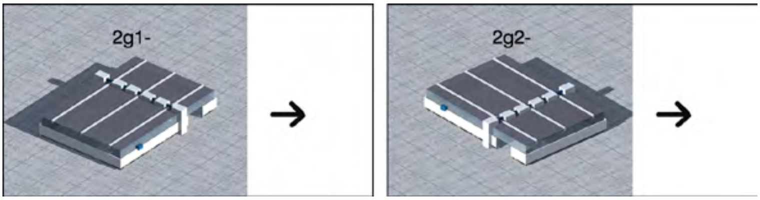


5.4.3 Subtractive Object Design Rules

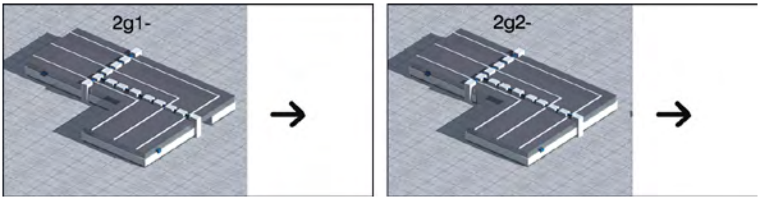
Subtractive object design rules are closely related to the subtractive layout rules. They are directed by the same group of state labels. Subtractive object design rules remove visual boundaries and purposeful objects from each area of the virtual gallery. The nine subtractive object design rules developed for the example grammar are listed below.

For example, subtractive object design rule 8 shows that the artist's personal studio area can be removed if:

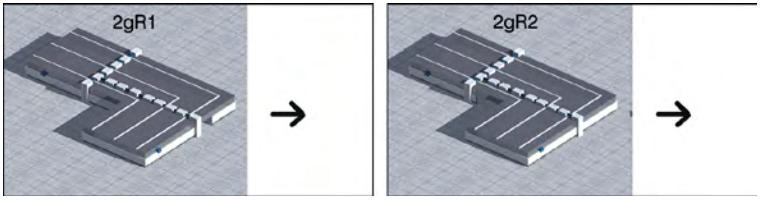
- The artist's personal studio area is adjacent to a reception area in the gallery, and
- sL=cS is matched.



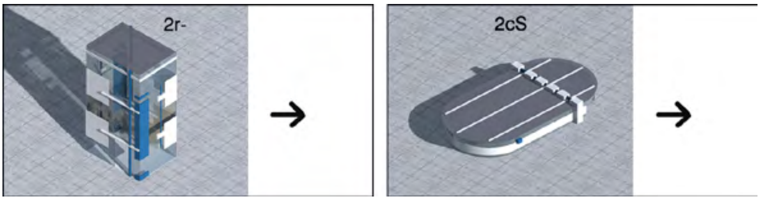
Subtractive object design rule 1 (left) and 2 (right).



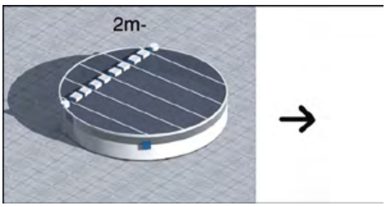
Subtractive object design rule 3 (left) and 4 (right).



Subtractive object design rule 5 (left) and 6 (right).



Subtractive object design rule 7 (left) and 8 (right).



Subtractive object design rule 9.

5.5 Navigation Rules

Navigation rules are the third set of design rules to be applied after layout rules and object design rules. There are 23 additive navigation rules and 6 subtractive navigation rules in total. Navigation rules provide way finding aids and hyperlinks in the virtual gallery to assist the visitors' navigation among different areas and their visits in the exhibitions.

For the example grammar, the main navigation method for short-distance travel in the virtual gallery is through "walking" by the avatars. The main navigation method for long-distance travel in the virtual gallery is through the use of hyperlinks. These two main methods are enriched to include the following guidelines.

- Each gallery area, the artist's personal studio, or the multi-function area is always spatially adjacent to a reception area, connected via openings.
- The virtual gallery can have multiple reception areas in order to connect with all other areas of the gallery. Any two reception areas are accessible to each other via hyperlinks.
- A reception area has three levels: it connects to a gallery 1 area from floor 1, to a gallery 2 area from floor 2, and to the artist's personal studio area and the multi-function area from floor 3. Any two floors of a reception area are accessible to each other via hyperlinks.
- A path is laid between two spatially adjacent areas connected via openings for directing visitors.
- Paths are laid in each gallery area for guiding visitors in the exhibition.

Based on these guidelines, it is noted that the reception areas serve as nodes for the circulation in the virtual gallery. For example, a design of the virtual gallery may comprise a standard gallery 1 area, a standard gallery 2 area and the artist's personal studio area, all connected to a reception area. If a visitor intends to reach gallery 2 from gallery 1, his/her avatar first walks to floor 1 of the reception area by following the path laid between gallery 1 and the reception area. From floor 1 of the reception area and through hyperlinks, the avatar reaches floor 2 of the reception area where gallery 2 is spatially connected. The avatar then follows the path to reach gallery 2.

Besides the symbols that have been introduced in the earlier sections, additional symbols are used in navigation rules.

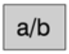
⧫ or ⧫: an opening to connect two spatially adjacent areas.

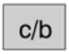
●- - - -●: a pair of hyperlinks that transport GDAs/avatars between any two locations in the virtual gallery. In the example grammar, this kind of hyperlink is used in different reception areas of the virtual gallery.

●——●: a pair of hyperlinks that transport GDAs/avatars between any two locations in the virtual gallery. In this example grammar, this kind of hyperlink is used in different floors of a reception area.

: a path laid between two spatially adjacent areas for directing visitors from one area to the other.

: a path laid in a gallery area for guiding visitors in an exhibition.



and 

: any two different floors of a reception area.

5.5.1 State Labels

Navigation rules share some of the state labels, defined earlier in Section 5.4.1, for object design rules. The only states label specifically developed for navigation rules is state label 3.

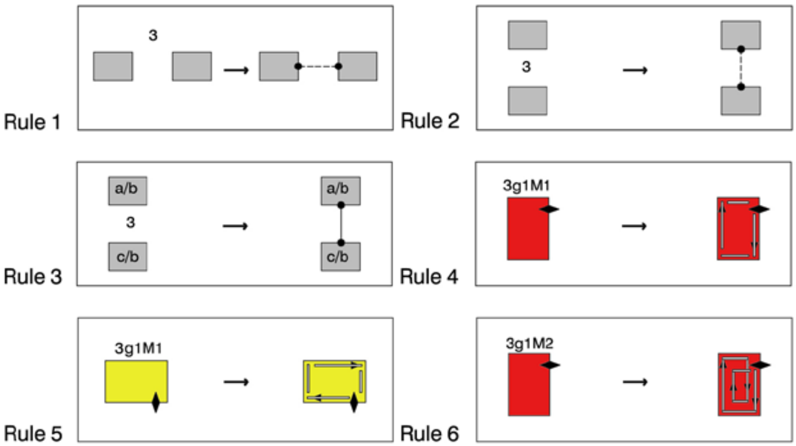
sL=3 is used in all navigation rules indicating they are the third set of design rules to be applied.

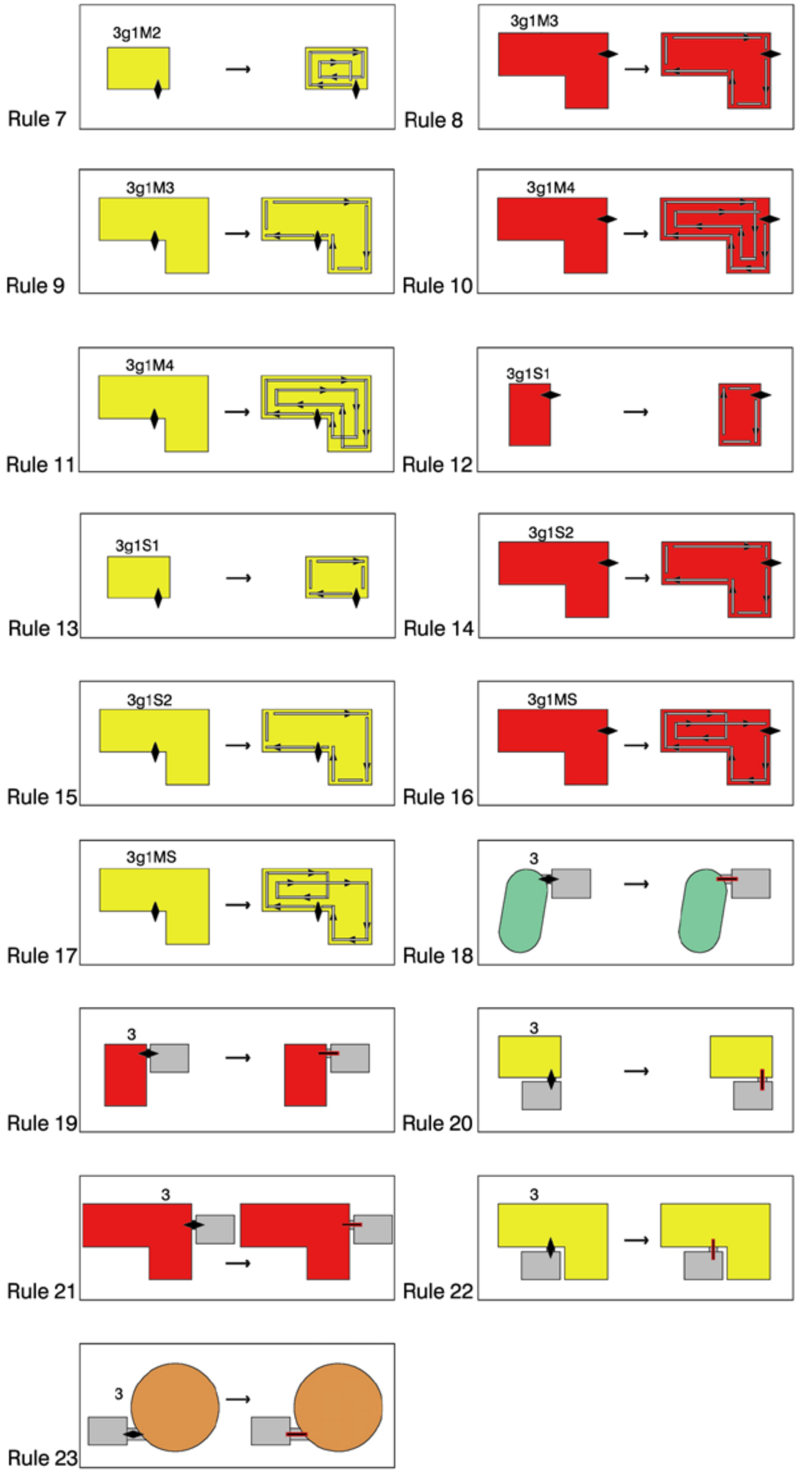
5.5.2 Additive Navigation Rules

The application of additive navigation rules adds way finding aids and hyperlinks to the generated design of the virtual gallery. Like most virtual world objects, way finding aids and hyperlinks appear as 3D models in the virtual gallery. However, for the ease of presentation, navigation rules are illustrated in 2D plan view in this book. The 23 additive navigation rules developed for the example grammar are listed below.

For example, additive navigation rule 4 shows that paths can be laid for a standard gallery 1 area to guide visitors through the exhibition if:

- The standard gallery 1 area is found in the virtual gallery, and
- sL=gIM1 is matched.




Rule 7
3g1M2
Rule 8
3g1M3
Rule 9
3g1M3
Rule 10
3g1M4
Rule 11
3g1M4
Rule 12
3g1S1
Rule 13
3g1S1
Rule 14
3g1S2
Rule 15
3g1S2
Rule 16
3g1MS
Rule 17
3g1MS
Rule 18
3
Rule 19
3
Rule 20
3
Rule 21
3
Rule 22
3
Rule 23
3

Among these 23 additive navigation rules:
- Rules 1 to 3 are applied to add hyperlinks to reception areas of the virtual gallery.
- Rules 4 to 17 are applied to lay paths in gallery areas to guide the visitors through different exhibitions.
- Rules 18 to 23 are applied to lay paths between different areas of the virtual gallery for directing the visitors when the areas are connected via openings.

5.5.3 Subtractive Navigation Rules

Subtractive navigation rules are not visual/spatial. They are mainly about recognizing the connections among different areas in the virtual gallery and removing way finding aids and hyperlinks once the connections are lost or change. The six subtractive navigation rules developed for the example grammar do not operate at a visual/spatial level. They are described in the form of “IF… THEN…”.

Rule 1.

sL=3
IF: A pair of hyperlinks connects reception area a with reception area b.
AND
Reception area a is not sensed in the virtual gallery.
AND/OR
Reception area b is not sensed in the virtual gallery.
THEN: Remove the hyperlinks.

Rule 2.

sL=3
IF: A pair of hyperlinks connects floor a with floor b in a reception area.
AND
Floor a is not sensed in the reception area.
AND/OR
Floor b is not sensed in the reception area.
THEN: Remove the hyperlinks.

Rule 3.

sL=3
IF: A path is laid in a gallery area for guiding visitors in an exhibition.
AND
The layout of the gallery area changes.
AND/OR
The visual boundaries of the gallery area change.
THEN: Remove the path.

Rule 4.
sL=3 IF: A path connects a reception area with the artist's personal studio area. AND The reception area is not sensed in the virtual gallery. AND/OR The artist's personal studio area is not sensed in the virtual gallery. THEN: Remove the path.

Rule 5.
sL=3 IF: A path connects a reception area with a gallery area. AND The reception area is not sensed in the virtual gallery. AND/OR The gallery area is not sensed in the virtual gallery. THEN: Remove the path.

Rule 6.
sL=3 IF: A path connects a reception area with the multi-function area. AND The reception area is not sensed in the virtual gallery. AND/OR The multi-function area is not sensed in the virtual gallery. THEN: Remove the path.

5.6 Interaction Rules

Interaction rules are the final set of design rules to be applied in the application of the example grammar. There are six additive interaction rules in total. Interaction rules ascribe scripted behaviors to selected objects in the virtual gallery. Therefore, visitors can interact with the environment by triggering these behaviors to participate in various intended activities.

Similar to subtractive navigation rules, interaction rules do not operate at a visual/spatial level. Interaction rules developed for the example grammar are also described in the form of "IF… THEN…".

5.6.1 State Labels

sL=4 is the only state label used in interaction rules indicating they are the final set of design rules to be applied.

5.6.2 Additive Interaction Rules

The example grammar has two different kinds of additive interaction rules. One supplements object design rules and the other supplements navigation rules. Object design rules define visual boundaries for each generated area of the virtual gallery and place purposeful objects in the areas. The first kind of the interaction rules (rules 1 to 4) ascribes scripted behaviors to selected objects so that exhibitions and other intended activities can function. Other interaction rules (rules 5 and 6) look for way finding aids and hyperlinks generated by navigation rules and ascribe appropriate behaviors to activate them.

Rule 1.

sL=4
IF: The 3D model of a digital picture frame object is recognized within a gallery area.
AND
The digital picture frame object is currently not configured.
THEN: Render the appropriate digital image onto the surface of the 3D model from the artist's exhibition.
AND
Enable the digital image to be enlarged and accessed from the web browser.

Rule 2.

sL=4
IF: The 3D model of an object that forms a part of an interactive installation is recognized within a gallery area or the multi-function area.
AND
The object is currently not configured.
THEN: Ascribe appropriate behaviors to the object according to the artist's exhibition requirements.

Rule 3.

sL=4
IF: The 3D model of a digital document object is recognized within a reception area, the artist's personal studio area, or the multi-function area.
AND
The digital document object is currently not configured.
THEN: Attach the relevant digital information to the object.
AND
Enable the detail of the information to be accessed from the web browser.

Rule 4.

sL=4
IF: A 3D model of a digital projector object is recognized in the multi-function area or the artist's personal studio area.
AND
The digital projector object is currently not configured.
THEN: Load the conference/meeting materials to the object for presentation.

Rule 5.

sL=4
IF: The 3D models of a pair of hyperlinks are recognized connecting two reception areas.
AND
The hyperlinks are currently not configured.
THEN: Detect the coordinates of the hyperlinks.
AND
Activate the hyperlinks using the detected coordinates.

Rule 6.

sL=4
IF: The 3D models of a pair of hyperlinks are recognized connecting two different floors of a reception area.
AND
The hyperlinks are currently not configured.
THEN: Detect the coordinates of the hyperlinks.
AND
Detect any obstacle between these two locations.
IF: No obstacle exists.
THEN: Activate the hyperlinks using the detected coordinates.
IF: Any obstacle exists.
THEN: Change the hyperlinks.
AND
Activate the hyperlinks using the detected coordinates.

5.6.3 Subtractive Interaction Rules

For the example grammar, subtractive interaction rules do not need to be explicitly defined. The application of additive interaction rules ascribes scripted behaviors to selected objects in the virtual gallery. The programming scripts that support the behaviors become attributes of the objects. Therefore, when other design rules of the grammar are applied to remove or make changes to the objects any attributes previously associated with the objects will be automatically removed.

5.7 Limitations

Generative design grammars draw inspiration from the shape grammar formalism and are adapted for rule-based place design in 3D virtual worlds.

There are a number of limitations in the current configuration of the example grammar. However, as the example grammar illustrated in this chapter is only a simplified example of a possible kind of generative design grammars for demonstration purposes, and therefore, the limitations listed below do not necessarily apply to other generative design grammars and can be optimized.

Firstly, each set of design rules in the example grammar is grouped into additive rules and subtractive rules. The design generation of the virtual gallery is by adding/subtracting different virtual world objects and their properties. This method of design generation is simple and clear for demonstrating the concept of rule-based place designs in 3D virtual worlds. Using this example grammar the dynamic design of the adaptive virtual gallery clearly reflects on the addition/subtraction of different areas and various purposeful objects and properties as needed. However, this is not the only way to generate a virtual gallery; for example, to implement the example grammar as a parametric grammar would enable the virtual gallery to be generated through parametric design to accommodate more complex and dynamic forms and spatial relations. This would also free the grammar from current constraints regarding the sizes and capacities of various gallery areas.

Secondly, object design rules in the example grammar are rigid. The varieties of the generated outcomes are limited and these rules should be enriched in the future. Although the current object design rules provide visual boundaries to each area of the virtual gallery and arrange the interior of the area in a very limited way, they are efficient in demonstrating the use of object design rules and their roles in a generative design grammar. However, in order to produce more interesting designs, object design rules will need to enable wider varieties of design considerations and styles.

Finally, designers currently strictly control the possible layouts and configurations of the generated gallery designs. Nevertheless, design rules can be developed to allow suitable levels of ambiguity or even randomness to be integrated into the designing process, to enable the generation of novel designs that are less predictable. For example, instead of allowing object design rules to choose from a fixed set of layout plans that configure the gallery interior for displaying exhibitions in a limited way, object design rules can be developed from scratch to address gallery configurations by directly dealing with the design and spatial arrangement of individual objects, in order to produce more dynamic and less predictable outcomes.

References

Newhouse, V. (1998), *Towards a New Museum*, Monacelli Press, New York.
Schwarz, H.P. (1997), *Media-Art-History*, Prestel-Verlag, Munich.

6 An Adaptive Virtual Gallery

In this chapter we present a design scenario that applies the generative design grammar described in Chapter 5 within a Generative Design Agent (GDA). The design scenario consists of eight different stages. These different stages present various situations where changes may occur in a virtual gallery during its use; for example, changes of activities, changes of exhibition requirements, changes of visitors, and so on. The design scenario demonstrates that the artist's GDA reasons about these changes and dynamically generates and modifies the design of the virtual gallery to accommodate the changes. Although the design scenario is constructed with a specific kind of virtual gallery in mind, it demonstrates the effective use of GDAs and generative design grammars for rule-based place design in 3D virtual worlds.

6.1 Design Scenario

The characters in the design scenario include the artist, the guest and various visitors. The artist is represented as a Generative Design Agent (GDA) in the virtual gallery. Other characters are represented as general avatars, which only have the visual representation as the 3D animated characters but have no agency – the ability to reason, design and act in the virtual gallery. The GDA on the other hand can reason, design and act on behalf of the artist in the virtual gallery. The design component of the GDA is supported by the application of the example generative design grammar. The eight stages of the design scenario present situations where various changes occur in the virtual gallery; for example, changes of the artist's and other visitors' activities, changes of the artist's exhibitions, increase and decrease of visitor numbers, and so on. The GDA senses the changes and hypothesizes design goals in order to accommodate these changes. To meet these design goals, the GDA applies the example grammar to generate different designs of the virtual gallery as needed.

6.1.1 Purpose of the Design Scenario

The purpose of the scenario is to demonstrate the application of the GDA and the example generative design grammar for rule-based design of a virtual gallery that adapts to its use. The design scenario shows how a GDA reasons, designs and acts in the virtual gallery for some typical situations where changes occur in the gallery during its use. The scenario does not intend to address all possible situations and changes, although they can be considered and addressed in future development.

An ideal agent-based virtual world should be represented by a society of GDAs where the designs of the world are dynamically generated by the multi-agent society, so that the designs meet the common goals of the virtual community and reflect their common interests. To do so, the communication among GDAs needs to be addressed. Although the GDA model is applicable to multi-agent virtual worlds, since the reasoning mechanism of the GDA model enables each GDA to reason about the virtual world as well as other GDAs in the virtual world, agent communication is however beyond the scope of this book. To control the complexity of the scenario, as mentioned earlier, only the artist is represented with a GDA in the virtual gallery.

- When the artist is present in the virtual gallery, the virtual gallery will be dynamically designed, implemented and manipulated by the GDA, on behalf of the artist. To a certain extent, the visitors' needs and interests in the virtual gallery are also reflected in the generated designs, as the design goals of the GDA are hypothesized based on its interpretations of the virtual gallery and different users' activities in the gallery.
- When the artist is not present, the virtual gallery will be replaced by a static design. Static and adaptive virtual galleries are further discussed below.

6.1.2 Generative Design Grammar Application

The application of the example generative design grammar is directed by a set of special state labels in order to generate designs that meet the GDA's current design goals. Each design rule of the example grammar is associated with one of these state labels. A design rule is applied only when the following conditions are met.

- The LHO of the rule is recognized in the virtual gallery, and
- The design context represented by the state label is related to the GDA's current design goals.

However, in cases where there is more than one design rule that meets the above conditions a control mechanism is needed to resolve the conflict. In general, there are three main methods for controlling the generative design grammar application. They are random selection, human designer or user intervention and agent learning mechanism.

- The random selection method allows the system to randomly select one design rule from the set of rules that meet the conditions.
- The human designer or user intervention method allows the system to turn to human designers or users for instructions once such a conflict occurs.
- The agent learning mechanism provides a more autonomous but more complex approach to allow the system to resolve such a conflict based on the GDA's previous design experiences and user feedback.

The example grammar applies the human designer or user intervention method. When the artist's GDA finds more than one design rule that can be applied, the GDA informs the artist and waits for further instructions. In the design scenario, such a case occurs twice, one at Stage 2 and the other at Stage 5.

6.1.3 Static and Adaptive Virtual Gallery

In the design scenario, the virtual gallery has static and adaptive designs. The scenario starts with the login of the artist to the virtual world, where the artist's virtual gallery is located. The artist is represented by a GDA in the virtual world. This marks the beginning of the virtual gallery being dynamically designed, implemented and manipulated as needed by the GDA, adapting to its use. As the artist is the only character of the scenario given the agency in the virtual gallery, adaptive virtual gallery therefore is not possible without their presence. Prior to the start of the scenario, a static design of the virtual gallery is used in the virtual world. The design is the result of the artist's previous visit. As shown in Figure 6.1, this static design comprises one gallery area for displaying one of the exhibitions. The gallery area connects to a reception area from floor 1 of the reception area. Without the GDA, the virtual gallery is static because the reasoning and designing activities are supported through the GDA.

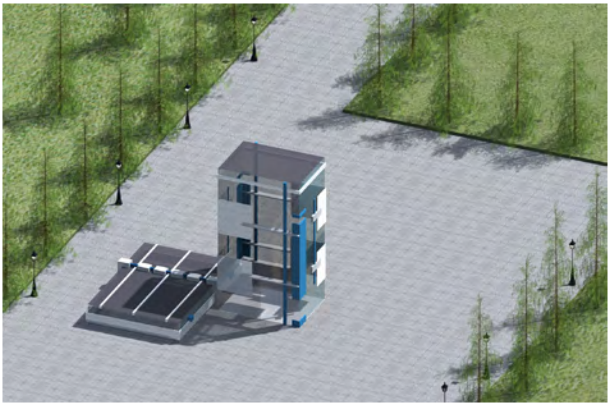


Figure 6.1: The visualization of the static virtual gallery design used prior to the design scenario.

At the end of the design scenario, the artist disconnects from the virtual world. The GDA applies the termination rule to terminate the grammar application before terminating its own agent program. As a result, a static design of the virtual gallery is generated once again. The dynamic designing process will re-start when the artist returns to the virtual gallery.

6.2 Stage 1: The Artist Enters the Virtual Gallery

At Stage 1, the artist connects to the virtual world, where the artist's virtual gallery is located. The artist is represented by a GDA in the virtual world. At the moment, he intends to display two new exhibitions: exhibitions 1 and 2 in the virtual gallery. In the initial static virtual gallery, the GDA senses a number of visitors, each of whom are represented by an avatar. On behalf of the artist, the GDA welcomes the visitors and notifies them that the virtual gallery will be temporarily under construction for arranging new exhibitions. The visitors are reminded that in a very short moment they will be invited and transported to the new exhibitions.

The GDA, meanwhile, demolishes the initial static virtual gallery and at the same location places the initial design of the example grammar: the layout of a reception area. With the initial design being recognized, the virtual gallery's dynamic designing process starts.

6.2.1 Layout Rule Application

The application of the design rules in a generative design grammar follows the sequence of layout rules, object design rules, navigation rules and interaction rules.

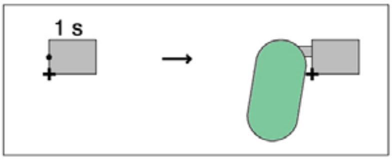
To prepare for the application of layout rules at Stage 1, The GDA performs the following reasoning:

- In the process of interpretation, the GDA interprets (1) the presence of the artist, and that (2) the artist currently has no studio space in the virtual gallery. The GDA also interprets that (3) the artist intends to display two exhibitions: exhibitions 1 and 2, and (4) currently the virtual gallery has no gallery space available.
- In the process of hypothesizing, based on interpretation (1) and (2), the GDA hypothesizes a design goal O_{exp}^{F}=S (the personal studio area for the artist is needed in the virtual gallery). Based on interpretation (3) and (4), the GDA hypothesizes the next two design goals O_{exp}^{F}=g1 (the initial standard gallery 1 area is needed in the virtual gallery) and O_{exp}^{F}=g2 (the initial standard gallery 2 area is needed in the virtual gallery).

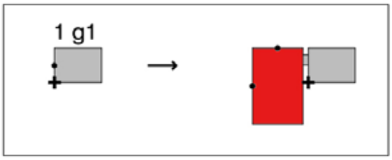
With these design goals being hypothesized, the GDA starts the rule matching process in order to apply the example grammar:

- For matching the LHOs of the design rules: the initial design  is recognized.
- For matching the state labels of the design rules: because O_{exp}^{F}=S, O_{exp}^{F}=g1 and O_{exp}^{F}=g2 are hypothesized, therefore sL=S, sL=g1 and sL=g2 are matched.

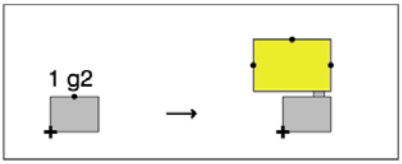
Based on the above criteria, the GDA searches the design rules of the example grammar for eligible rules to apply, starting from layout rules. As each layout rule is associated with state label sL=1 that indicates the stage of the grammar application, the matched state labels become sL=1 S, sL=1 g1 and sL=1 g2. The matched layout rules are:



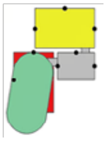
Match 1: additive layout rule 1.



Match 2: additive layout rule 2.



Match 3: additive layout rule 3.

The application of the above layout rules generates a layout  for the virtual gallery, which comprises a reception area, a personal studio area for the artist, a standard gallery 1 area for displaying exhibition 1, and a standard gallery 2 area for displaying exhibition 2.

6.2.2 Object Design Rule Application

Object design rules are the second set of design rules to be applied.

Now that the layout of the virtual gallery is generated, additional reasoning is required by the GDA to continue with the application of object design rules. In order to generate visual boundaries and visual cues for each area, the GDA needs to be provided with the artist's design preferences for the interior of the virtual gallery and the specifications of the exhibitions.

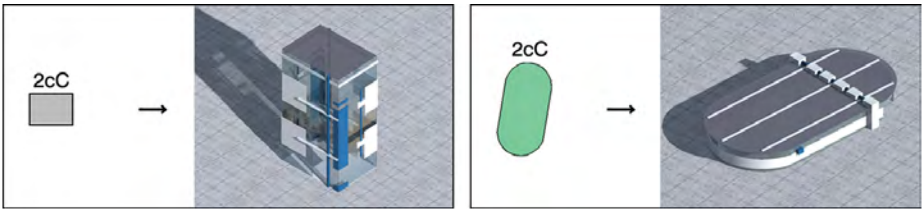
– In the process of interpretation, based on the instructions given by the artist, the GDA interprets that (1) the artist prefers the use of a cold-color scheme for the interior of the virtual gallery, and (2) exhibitions 1 and 2 each contains various digital images in various sizes which can be arranged for displaying in the virtual gallery using configuration 1.

– In the process of hypothesizing, based on interpretation (1), the GDA hypothesizes a design goal O_{exp}^{F}=cC (to apply a cold-color scheme for the interior of the virtual gallery). Based on interpretation (2), the GDA hypothesizes the next design goal O_{exp}^{F}=gIM1 (to arrange the two gallery areas for displaying digital images using configuration 1).

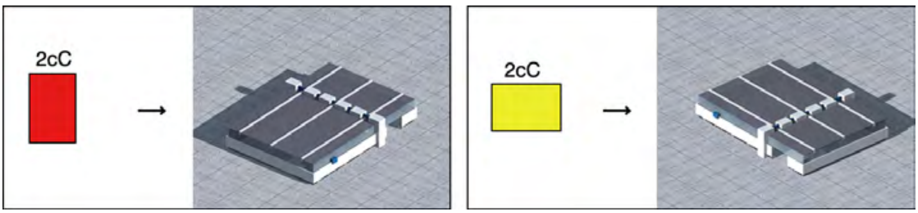
With these design goals being hypothesized, the GDA starts the matching process:

– For matching the LHOs of the design rules:  ,  ,  and , the layouts of the four areas are recognized.
– For matching the state labels of the design rules: because O_{exp}^{F}=cC and O_{exp}^{F}=gIM1 are hypothesized, therefore sL=cC and sL=gIM1 are matched.

Based on the above criteria, the GDA continues the search in object design rules for eligible rules to apply. As each object design rule is associated with state label sL=2 that indicates the stage of the grammar application, the matched state labels become sL=2 cC and sL=2 gIM1. The matched object design rules are:

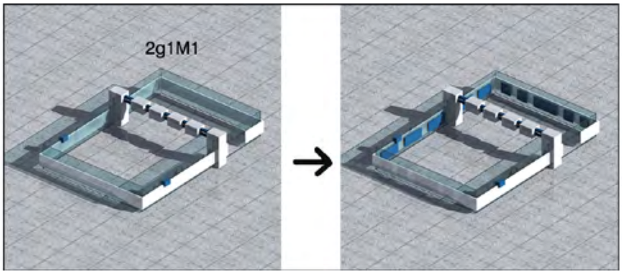


Matches 1 and 2: additive object design rules 1 (left) and 2 (right).

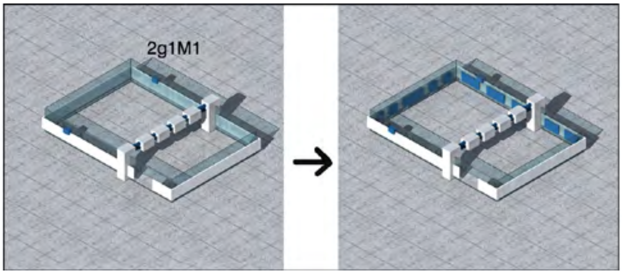


Matches 3 and 4: additive object design rules 3 (left) and 4 (right).

The application of these four object design rules provides the generated 2D layout of the virtual gallery with purposeful 3D objects to define visual boundaries and provide visual cues for each area. Subsequently, for the two gallery areas, the following two object design rules are also matched to further arrange the gallery areas for displaying the two exhibitions:



Match 5: additive object design rule 8.



Match 6: additive object design rule 9.

The result of the object design rule application is shown in Figure 6.2. The virtual gallery has four areas: the reception area, the artist's personal studio area, a standard gallery 1 area and a standard gallery 2 area. The reception area has three floors and connects to gallery 1 from floor 1, gallery 2 from floor 2, and the studio area from floor 3.

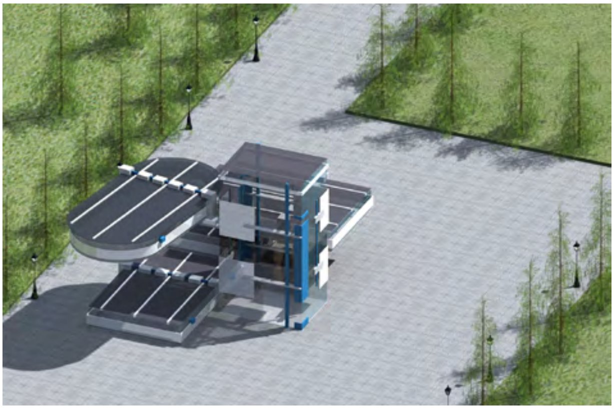


Figure 6.2: The visualization of the virtual gallery design for Stage 1.

6.2.3 Navigation Rule Application

Next, the GDA applies navigation rules to provide way finding aids and hyperlinks[12] to the generated design of the virtual gallery. Navigation rules are the third set of design rules to be applied after layout rules and object design rules. In the matching process:

- For matching the LHOs of the design rules: the four areas of the virtual gallery generated at Stage 1 are recognized. They are shown in Figure 6.3 from left to right: the reception area, the artist's personal area, the standard gallery 1 area and the standard gallery 2 area.
- For matching the state labels of the design rules: the design goal $O_{exp}{}^{F}$=gIM1 is related to not only the application of object design rules, but also the application of navigation rules. Therefore, because $O_{exp}{}^{F}$=gIM1 is hypothesized, sL=gIM1 is hence matched.

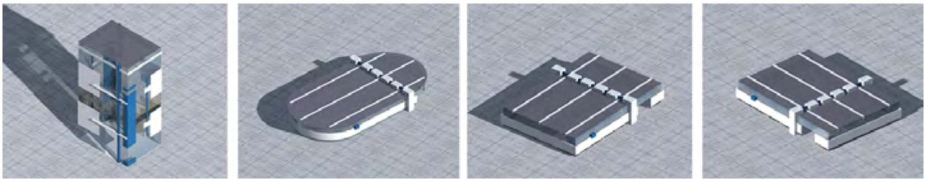
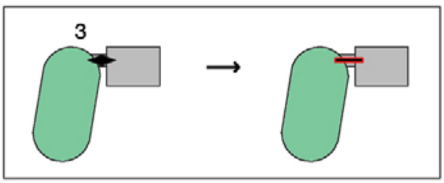


Figure 6.3: The visualizations of the four areas of the virtual gallery generated at Stage 1.

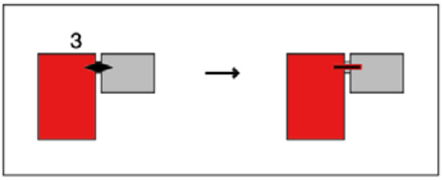
Based on the above criteria, the GDA searches navigation rules of the example grammar, and the matched navigation rules are:

Match 1: additive navigation rule 3.



Match 2: additive navigation rule 18.

12 The guidelines for using way finding aids and hyperlinks in this virtual gallery are presented in Chapter 5.



Match 3: additive navigation rule 19.

Match 4: additive navigation rule 20.

Match 5: additive navigation rule 4.

Match 6: additive navigation rule 5.

Additive navigation rule 3 is applied multiple times to connect the three floors of the reception area together with each other using hyperlinks. Additive navigation rules 4 and 5 are applied to lay paths in the two gallery areas for guiding visitors through the exhibitions. Finally, additive navigation rules 18, 19 and 20 are applied to lay paths to connect the reception area with the artist's personal studio area, and the two gallery areas, for directing visitors.

6.2.4 Interaction Rule Application

To complete the design for Stage 1, the GDA applies interaction rules to ascribe scripted behaviors to selected objects in the virtual gallery. Therefore, visitors can interact with the environment by triggering these behaviors and to participate in various intended activities. Interaction rules are non-visual/spatial rules, they are about recognizing selected objects in the virtual gallery and ascribing appropriate behaviors to these objects. Therefore, in the matching process, the main concern is to match the LHOs of the design rules. The only state label used in all interaction rules is sL=4 indicating they are the fourth, the final set of design rules to be applied.

The GDA recognizes various digital picture frame objects in the two gallery areas for displaying digital images of the exhibitions, various digital document objects in the reception area and the artist's personal studio area for storing and redirecting digital information, and various hyperlinks on different floors of the reception area. Based on these criteria, the GDA searches interaction rules of the example grammar, and the matched interaction rules are:

Match 1: additive interaction rule 1:

sL=4
IF: The 3D model of a digital picture frame object is recognized within a gallery area.
AND
The digital picture frame object is currently not configured.
THEN: Render the appropriate digital image onto the surface of the 3D model from the artist's exhibition.
AND
Enable the digital image to be enlarged and accessed from the web browser.

Match 2: additive interaction rule 3:

sL=4
IF: The 3D model of a digital document object is recognized within a reception area, the artist's personal studio area, or the multi-function area.
AND
The digital document object is currently not configured.
THEN: Attach the relevant digital information to the object.
AND
Enable the detail of the information to be accessed from the web browser.

Match 3: additive interaction rule 6:

sL=4
IF: The 3D models of a pair of hyperlinks are recognized connecting two different floors of a reception area.
AND
The hyperlinks are currently not configured.
THEN: Detect the coordinates of the hyperlinks.
AND
Detect any obstacle between these two locations.
IF: No obstacle exists.
THEN: Activate the hyperlinks using the detected coordinates.
IF: Any obstacle exists.
THEN: Change the hyperlinks.
AND
Activate the hyperlinks using the detected coordinates.

Additive interaction rule 1 is applied multiple times to display digital images that form the two exhibitions using various digital picture frame objects. Additive interaction rule 3 is applied multiple times to provide information about the virtual gallery and the exhibitions in the reception area, and to store the artist's digital tools and data in his personal studio area and also to redirect to other media, using various digital document objects. Additive interaction rule 6 is applied multiple times to activate the hyperlinks so that the visitors can navigate through the three floors of the reception area to access different parts of the virtual gallery.

The final design of the virtual gallery for Stage 1 is now generated with four areas: the reception area that provides information regarding the virtual gallery and the exhibitions, the personal studio area equipped with various digital tools and data for the artist, and two gallery areas that are configured for displaying the artist's two exhibitions. In the processes of action, the GDA plans actions for implementing this generated design of the virtual gallery, and activates the planned actions via its effectors in the virtual world. Once the design for Stage 1 is implemented, the GDA notifies the artist and the visitors before transporting the artist to the personal studio area and the visitors to the gallery areas.

For the remaining stages of the scenario, the execution of each stage follows similar procedures shown above at Stage 1. In the following sections, these procedures are described in a simpler version.

6.3 Stage 2: Exhibition 1 Receives More Visitors

At Stage 2, more visitors connect to the virtual world and visit the virtual gallery. At one point, the number of visitors in the standard gallery 1 reaches the maximum capacity of a standard gallery area[13]. The artist's GDA senses this change and applies the example grammar to add an additional gallery area for displaying exhibition 1. Any future visitors who wish to visit exhibition 1 will be automatically transported to this newly generated gallery area, until the number of visitors in the original gallery area drops below the maximum number.

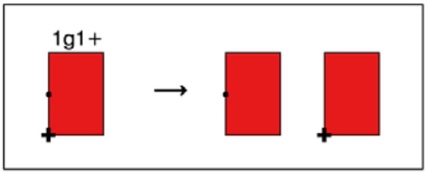
6.3.1 Execution of the Design Scenario

The execution of Stage 2 follows similar procedures shown at Stage 1. In the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the initial change at Stage 2 is the increase of visitors in the standard gallery 1 area. Based on its current interpretations,

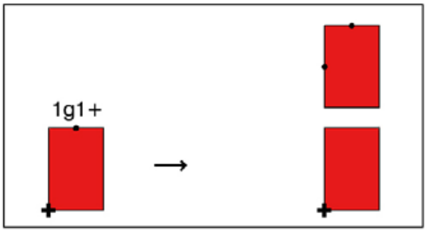
13 For this example grammar, a gallery area has two different sizes and each gallery area is designed for hosting a certain number of avatars for the comfort of the visitors' viewing and the ease of their avatars' movements in the area.

in the process of hypothesizing, the GDA hypothesizes design goals. For example, one of the design goals hypothesized by the GDA at Stage 2 is O_{exp}^{F}=g1+ (an additional standard gallery 1 area is needed in the virtual gallery). With new design goals being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

Firstly, in the application of layout rules the newly matched state label is sL=1 g1+. The following two layout rules both satisfy the criteria:

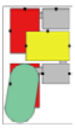


Additive layout rule 4.

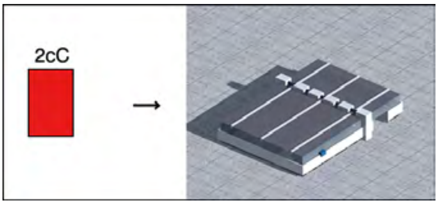


Additive layout rule 6.

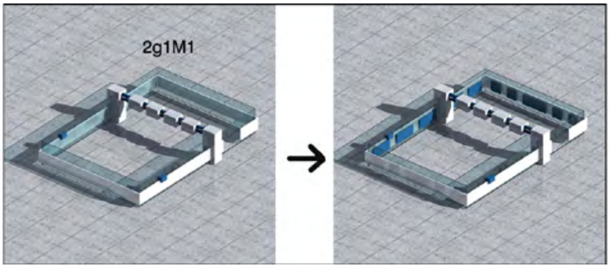
Both layout rules add an additional gallery area in the virtual gallery. Additive layout rule 4 expands the layout of the virtual gallery along the X axis (local to the design), and additive layout rule 6 expands the layout along the Y axis (local to the design). The GDA turns to the artist for further instructions. In the design scenario, the artist prefers to expand the layout of the virtual gallery along the Y axis, therefore the GDA applies additive layout rule 6 for design generation.

With the application of other relevant layout rules, a layout  of the virtual gallery is generated for Stage 2 of the design scenario.

Secondly, in the application of object design rules the newly matched state labels are sL=2 cC and sL=2 gIM1. The matched object design rules are:



Additive object design rule 3.



Additive object design rule 8.

Additive object design rule 3 is applied to define visual boundaries and provide visual cues for the newly generated gallery area, and this gallery area is further arranged for displaying exhibition 1 by applying additive object design rule 8. The result of the object design rule application is shown in Figure 6.4.

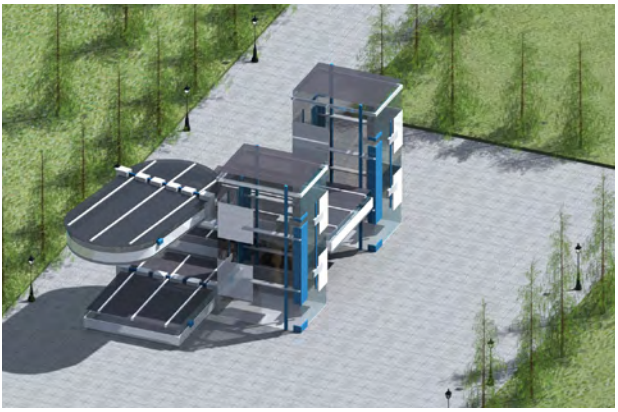


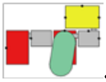
Figure 6.4: The visualization of the virtual gallery design for Stage 2.

To complete the design, the GDA searches and applies appropriate navigation rules to provide way-finding aids and hyperlinks to connect with the newly generated areas. Finally, the GDA searches and applies appropriate interaction rules to ascribe scripted behaviors to selected objects in the newly generated areas when it is necessary.

Compared to the design generated for Stage 1, the current design has an additional gallery area (a standard gallery 1 area) for displaying exhibition 1. An additional reception area is also generated to connect the newly generated gallery area with the rest of the virtual gallery.

6.3.2 An Alternative Design of the Virtual Gallery for Stage 2

As mentioned earlier, the application of layout rules for Stage 2 can have two different ways to expand the virtual gallery. An alternative design could have been generated if the artist had preferred to expand the virtual gallery along the X axis.

If this had been the case, additive layout rule 4 (see page 107) would be applied, and the layout generated for Stage 2 would be . As shown in Figure 6.5, this would change the design of the virtual gallery for Stage 2, and subsequently change all the designs generated for the rest of the design scenario.

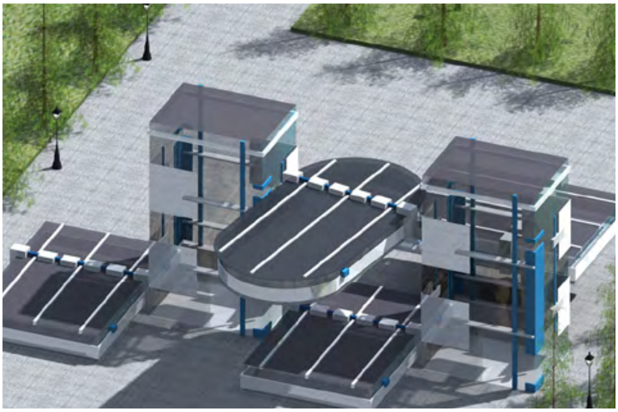


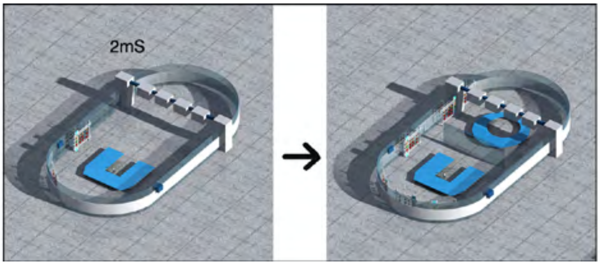
Figure 6.5: The visualization of the alternative design at Stage 2.

6.4 Stage 3: An Invited Guest Enters the Virtual Gallery

At Stage 3, the artist is browsing some digital information in his personal studio area. He instructs his GDA to organize a meeting venue for meeting his guest B, who will be arriving in the virtual gallery soon. The GDA applies the example grammar to arrange a meeting area inside the artist's personal studio area. When user B connects to the virtual world, the GDA welcomes B on behalf of the artist and transports B's avatar directly to the newly-generated meeting area.

For the execution of Stage 3, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the changing need of the artist at Stage 3 reflects the request for a meeting venue. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes a design goal O_{exp}^{F}=mS (to configure a meeting area in the artist's personal studio area). With the new design goal being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

To configure a meeting area in the artist's personal studio area does not require the application of layout rules since the personal studio area has been generated at Stage 1. In the application of object design rules, the newly matched state label is sL=2 mS. The matched object design rule is:



Additive object design rule 22.

As the meeting area is arranged inside the artist's personal studio area it is not necessary to provide way finding aids and hyperlinks separately for this area. Therefore, navigation rules are also not applied at Stage 3. To complete the design, the GDA searches and applies appropriate interaction rules to ascribe scripted behaviors to activate various meeting facilities in the newly arranged meeting area.

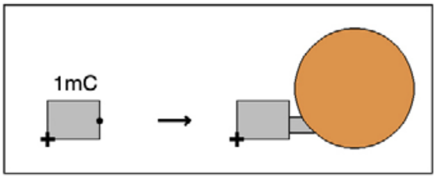
The current design of the virtual gallery is the same as the design generated for Stage 2, except that a meeting area is now arranged inside the artist's personal studio area.

6.5 Stage 4: The Artist Prepares for an Opening Address

At Stage 4, the artist decides to give an opening address to highlight the newly opened exhibitions after the meeting with the guest. The artist instructs the GDA to organize a venue for the public function. The GDA applies the example grammar to generate the multi-function area in the virtual gallery and arranges the area as a conference venue. After the artist finishes the preparation for the talk, the GDA sends an invitation to the visitors on behalf of the artist and transports those who accept the invitation to the multi-function area.

For the execution of Stage 4, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery. For example, the changing need of the artist at Stage 4 reflects the request for a venue to host a public talk. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes design goals; for example, one of the design goals hypothesized by the GDA at Stage 4 is O_{exp}^{F}=mC (the multi-function area is needed as a conference venue). With new design goals being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

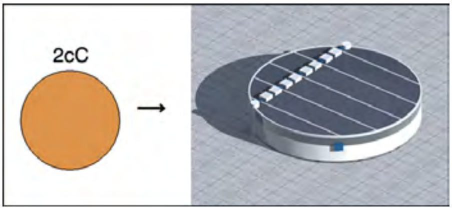
Firstly, in the application of layout rules, the newly matched state label is sL=1 mC. The matched layout rule is:



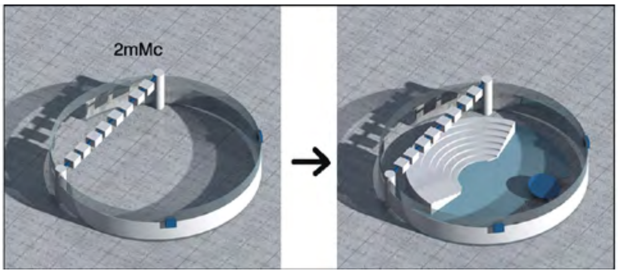
Additive layout rule 14.

By applying this layout rule, a layout  of the virtual gallery is generated for Stage 4. The multi-function area is added.

Secondly, in the application of object design rules the newly matched state labels are sL=2 cC and sL=2 mMc. The matched object design rules are:



Additive object design rule 7.



Additive object design rule 23.

Additive object design rule 7 is applied to define visual boundaries and provide visual cues for the newly generated multi-function area, and this area is further arranged as a conference venue by applying additive object design rule 23. The result of the object design rule application is shown in Figure 6.6.

To complete the design, the GDA searches and applies appropriate navigation rules to provide way finding aids and hyperlinks to connect with the multi-function area, and, finally, the GDA searches and applies appropriate interaction rules to ascribe scripted behaviors to activate various conference facilities in the multi-function area.

Compared to the design generated for Stage 3, the current design of the virtual gallery has a venue for public functions, that is, the multi-function area is arranged as a conference venue for the artist's opening address.

Figure 6.6: The visualization of the virtual gallery design for Stage 4.

6.6 Stage 5: Exhibition 2 Receives More Visitors

After the talk some visitors remain in the multi-function area for discussion with the artist. The rest return to the exhibitions. At one point, the number of visitors in the standard gallery 2 area also reaches the maximum capacity of a standard gallery area. The artist's GDA senses this change and applies the example grammar to add an additional gallery area for displaying exhibition 2. Any future visitors who wish to visit exhibition 2 will be automatically transported to this newly generated gallery area until the number of visitors in the original gallery area drops below the maximum number.

6.6.1 Execution of the Design Scenario

For the execution of Stage 5, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the initial change at Stage 5 is the increase of visitors in the standard gallery 2 area. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes design goals; for example, one of the design goals hypothesized by the GDA at Stage 5 is $O_{exp}{}^{F}$=g2+ (an additional standard gallery 2 area is needed in the virtual gallery). With new design goals being hypothesized, new state labels can be matched for the search of eligible design rules for application.

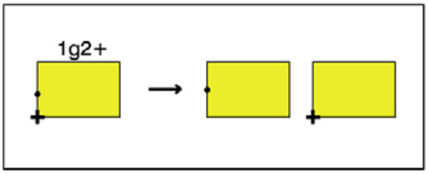
Stage 5 is very similar to Stage 2, where a design goal $O_{exp}{}^{F}$=g1+ (an additional standard gallery 1 area is needed in the virtual gallery) was hypothesized. Following the same procedures shown at Stage 2, the GDA matches new states labels, searches and applies appropriate layout rules, object design rules, navigation rules and interaction rules to generate an additional gallery area to accommodate more visitors for exhibition 2.

The application of layout rules generates a layout  of the virtual gallery for Stage 5. The design of the virtual gallery for this stage is shown in Figure 6.7. Compared to the design generated for Stage 4, the current design of the virtual gallery has an additional gallery area (a standard gallery 2 area) for displaying exhibition 2.

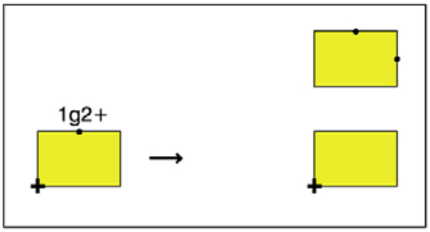
Figure 6.7: The visualization of the virtual gallery design for Stage 5.

6.6.2 Alternative Designs of the Virtual Gallery for Stage 5

Similar to Stage 2, the application of layout rules for Stage 5 can also have two different ways to expand the virtual gallery by applying the following two rules:



Additive layout rule 5



Additive layout rule 7

Additive layout rule 5 expands the layout of the virtual gallery along the X axis (local to the design), while additive layout rule 7 expands the layout along the Y axis (local to the design). In the scenario, the artist once again prefers to expand the layout of the virtual gallery along the Y axis, therefore the GDA applies additive layout rule 7 for design generation.

For both Stages 2 and 5, as discussed, there are two different ways to expand the layout of the virtual gallery. Figure 6.8 shows the two different layouts of the virtual gallery that can be generated for Stage 2. Based on these two layouts for Stage 2 the subsequent stages can each have two different layouts generated for the virtual gallery. The two layouts of the virtual gallery that can be generated for Stage 4 are illustrated in Figure 6.9.

Figure 6.8: The two layouts that can be generated for Stage 2.

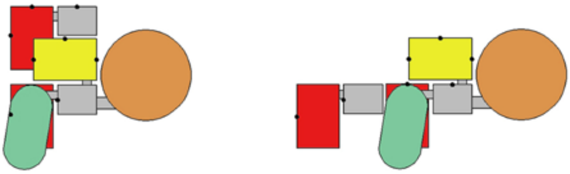


Figure 6.9: The two layouts that can be generated for Stage 4.

Based on the two layouts shown in Figure 6.9, by alternating the application of additive layout rules 5 and 7 three alternative layouts of the virtual gallery can be generated for Stage 5, besides the one generated for this design scenario. The alternative layouts are illustrated in Figure 6.10.

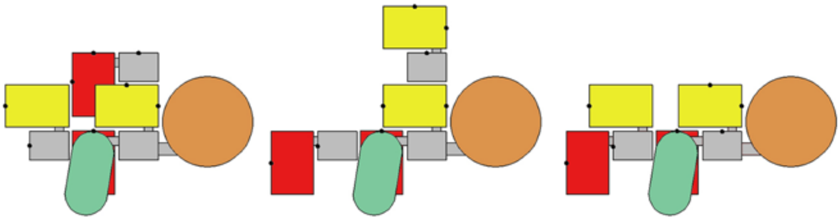


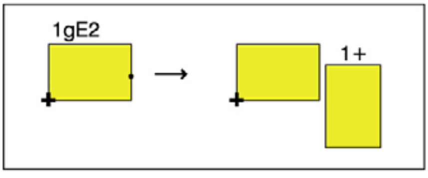
Figure 6.10: The three alternative layouts that can be generated for Stage 5.

6.7 Stage 6: The Artist Changes Exhibition Requirements

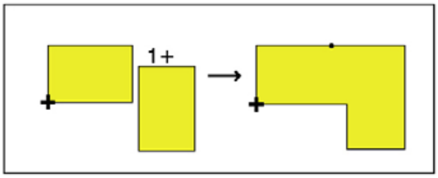
At Stage 6, the artist returns to his personal studio area and decides to modify exhibition 2 by adding more exhibition items. The artist instructs the GDA to accommodate the changes. On behalf of the artist, the GDA informs the visitors about the changes and applies the example grammar to expand and rearrange both gallery 2 areas in order to display the new exhibition 2.

For the execution of Stage 6, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the changing need of the artist at Stage 6 reflects the changes of exhibition 2. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes design goals; for example, one of the design goals hypothesized by the GDA at Stage 6 is O_{exp}^{F}=gE2 (the two standard gallery 2 areas need to be expanded). With new design goals being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

Firstly, in the application of layout rules, the newly matched state label is sL=1 gE2. The matched layout rules are:



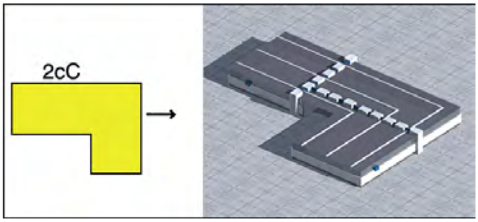
Additive layout rule 12.



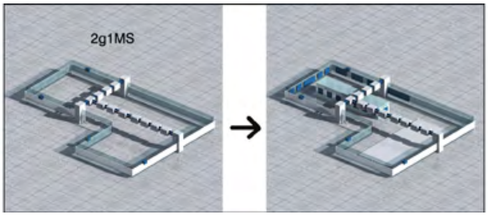
Additive layout rule 13.

By applying the above rules, a layout  of the virtual gallery is generated for Stage 6. The two standard gallery 2 areas are expanded.

Secondly, in the application of object design rules, the newly matched state labels are sL=2 cC and sL=2 gIMS. The matched object design rules are:



Additive object design rule 6.



Additive object design rule 21.

Additive object design rule 6 is applied to define visual boundaries and provide visual cues for the newly generated expanded gallery 2 areas[14], and the areas are further arranged for displaying the new exhibition 2 by applying additive object design rule 21. The result of the object design rule application is shown in Figure 6.11.

To complete the design, the GDA searches and applies appropriate navigation rules to provide way finding aids and hyperlinks to connect with the expanded gallery 2 areas, and, finally, the GDA searches and applies appropriate interaction rules to

14 At Stage 6, before the application of additive object design rules, various subtractive object design rules are applied to remove excessive objects from the previous design.

ascribe scripted behaviors to selected objects in these newly generated areas, when it is necessary.

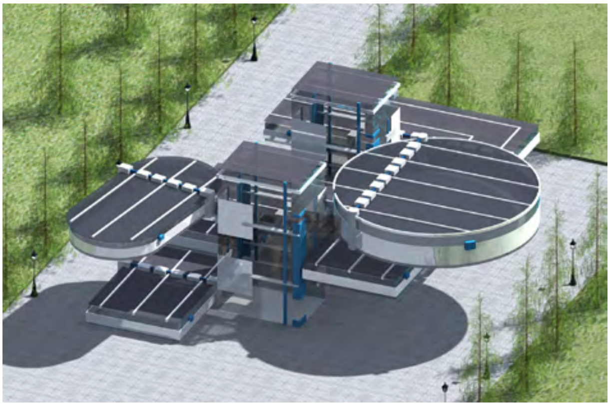


Figure 6.11: The visualization of the virtual gallery design for Stage 6.

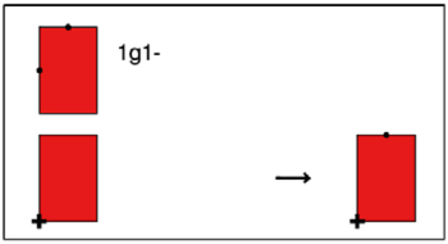
Compared to the design generated for Stage 5, the current design of the virtual gallery has expanded the two gallery 2 areas for displaying the new exhibition 2 which has more exhibition items.

6.8 Stage 7: Some Visitors Leave the Virtual Gallery

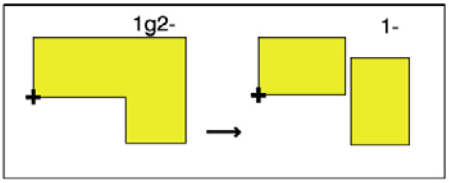
At Stage 7, more and more visitors disconnect from the virtual world. At one point, one of the gallery 1 areas has no visitor. The artist' GDA senses this change and applies its generative design grammar to remove this gallery area. Similar situations soon occur in one of the gallery 2 areas and in the multi-function area.

For the execution of Stage 7, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the main change at Stage 7 is the decrease of visitors in the virtual gallery. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes design goals. For example, one of the design goals hypothesized by the GDA at Stage 7 is O_{exp}^{F}=g1- (a standard gallery 1 area is redundant). With new design goals being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

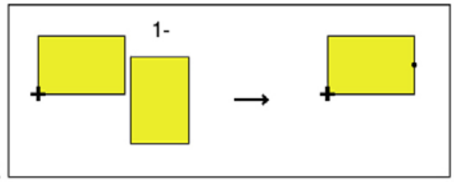
Firstly, in the application of layout rules the newly matched state labels are sL=1 g1-, sL=1 g2- and sL=1 m-. The matched layout rules are:



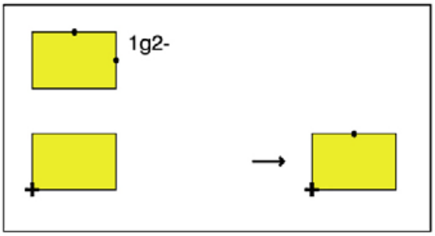
Subtractive layout rule 3.



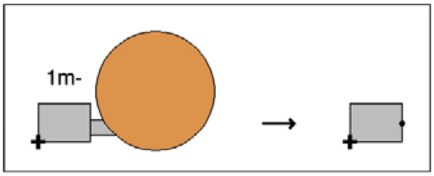
Subtractive layout rule 6.



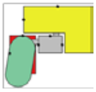
Subtractive layout rule 12.



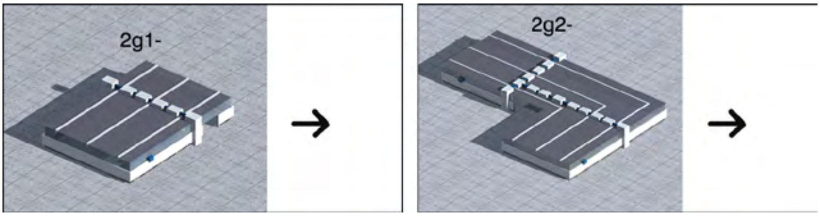
Subtractive layout rule 4.



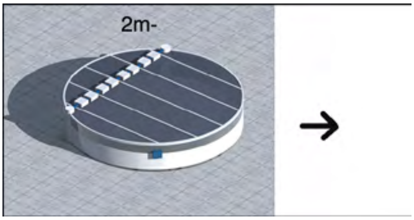
Subtractive layout rule 15.

By applying the above rules, a layout  of the virtual gallery is generated for Stage 7. Relevant gallery areas and the multi-function area are removed.

Secondly, in the application of object design rules the newly matched state labels are sL=2 g1-, sL=2 g2- and sL=2 m-. The matched object design rules are:



Subtractive object design rules 1 (left) and 4 (right).



Subtractive object design rule 9.

The above subtractive object design rules are applied to demolish the visual boundaries and visual cues of the relevant gallery areas and the multi-function area. To complete the design, the GDA searches and applies appropriate navigation rules to demolish any excessive way finding aids and hyperlinks due to the removal of the areas. No interaction rule is applied at Stage 7 because scripted behaviors are automatically removed once the objects to which they are ascribed are removed.

Figure 6.12 shows the visualization of the virtual gallery design generated for Stage 7. Compared to the design generated for Stage 6, two gallery areas (a standard gallery 1 area and an expanded gallery 2 area) and the multi-function area are removed in the current design. A reception area is also removed due to the removal of the above areas.

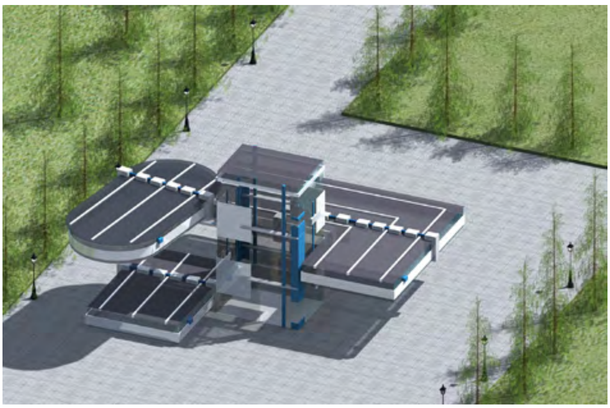


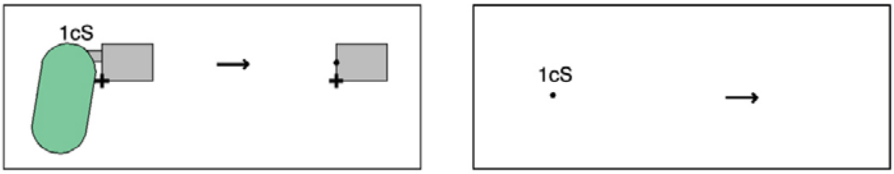
Figure 6.12: The visualization of the virtual gallery design for Stage 7.

6.9 Stage 8: The Artist Logs Off

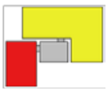
At the final stage of the design scenario the artist disconnects from the virtual world. The GDA records the current design of the virtual gallery for future references. The GDA then applies the example grammar to remove the artist' personal studio area and removes all spatial labels of the design to terminate the example grammar application, before terminating its own agent program. This generates a static design of the virtual gallery. Visitors can continue their visits to the exhibitions. The dynamic design process will re-start when the artist returns to the virtual gallery.

For the execution of the final stage, in the process of interpretation, the GDA interprets the changing needs of the artist and the visitors and changes in the virtual gallery; for example, the key change at this final stage is the artist' disconnection from the virtual world. Based on its current interpretations, in the process of hypothesizing, the GDA hypothesizes a design goal O_{exp}^{F}=cS (the current design of the virtual gallery is to be used as a static design and the gallery stops being dynamically designed). With the new design goal being hypothesized, new state labels can be matched for the search of eligible design rules in the example grammar for application.

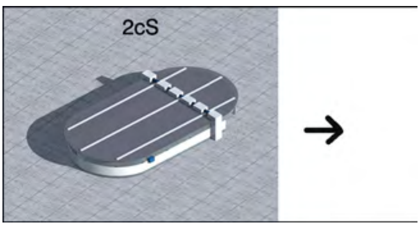
Firstly, in the application of layout rules the newly matched state label is sL=1 cS. The matched layout rules are:



Subtractive layout rules 16 (left) and 17 (right).

The application generates the layout  of a static design for the virtual gallery. Subtractive layout rule 16 is applied to remove the layout of the artist' personal studio area, and subtractive layout rule 17 is applied multiple times to remove all spatial labels from the design so that the application of the example grammar can be terminated.

Secondly, in the application of object design rules the newly matched state label is sL=2 cS. The matched object design rule is:



Subtractive object design rule 8.

Subtractive object design rule 8 is applied to demolish visual boundaries and visual cues of the studio area. To complete the design, the GDA searches and applies appropriate navigation rules to remove any excessive way finding aids and hyperlinks due to the removal of the studio area. Similar to Stage 7, there is no need for interaction rules to be applied when removing scripted behaviors in the final stage.

The generated static design of the virtual gallery is visualized in Figure 6.13. The static design comprises a standard gallery 1 area, an expanded gallery 2 area, and a reception area that connects to the two gallery areas.

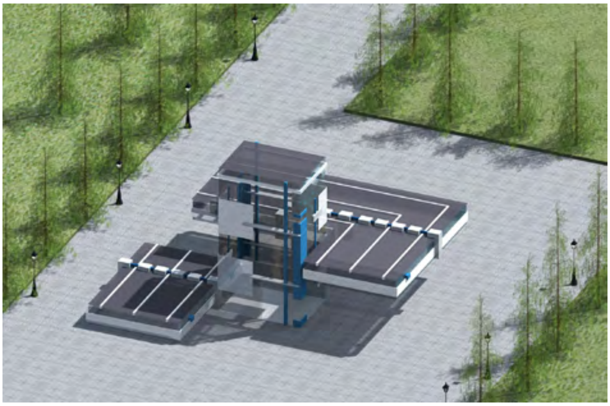


Figure 6.13: The visualization of the static virtual gallery design generated in the end of the design scenario.

6.10 Discussion

As demonstrated in the design scenario, the artist's GDA has dynamically generated eight designs of the virtual gallery for the eight stages of the design scenario. The virtual gallery adapts to its use through this dynamic design generation. Figure 6.14 shows how the virtual gallery changes from Stage 1 to Stage 8. The designs of the virtual gallery generated for the scenario are visualized in sequence from the top to the bottom in the left-hand-side column of the figure. The layouts of these designs are illustrated in the right-hand-side column of the figure. As discussed above, alternative designs could have been generated if the artist had preferred to expand the virtual gallery in a different dimension at Stages 2 and 5. These alternative designs are presented in the forms of their layouts, also in the right-hand-side column. They are marked with a darker background color to contrast the ones that are generated for the scenario.

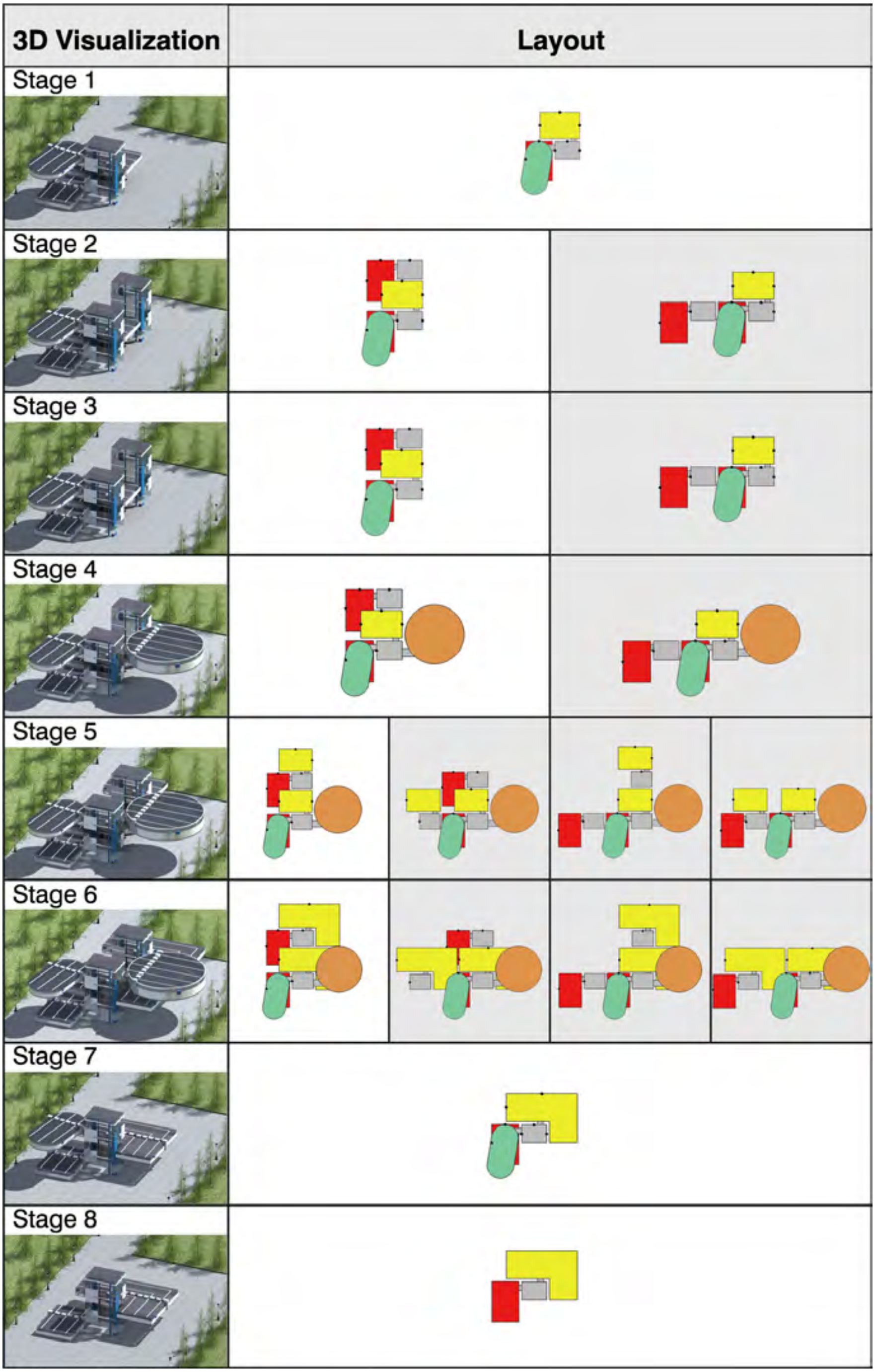


Figure 6.14: The corpus of virtual gallery designs generated for the design scenario.

6.10.1 Summary

Although the design scenario is constructed in the context of a specific kind of virtual gallery for an artist and it is developed with a set of limitations as described at the beginning of the chapter, nevertheless, the scenario demonstrates the effective use of GDAs and generative design grammars both for rule-based place design in 3D virtual worlds in general.

The eight stages of the design scenario present the following changes that may occur in the virtual gallery during its use:

- Stage 1: the artist becomes present in the virtual gallery.
- Stage 2: the number of visitors increases.
- Stage 3: the artist and an invited guest plan to have a meeting.
- Stage 4: the artist plans to give an opening address.
- Stage 5: the visitors are distributed differently in the virtual gallery.
- Stage 6: the artist decides to change one of the exhibitions.
- Stage 7: the number of visitors decreases.
- Stage 8: the artist disconnects from the virtual world.

The execution of the design scenario shows that the artist's GDA reasons about the above changes and applies the example generative design grammar illustrated in Chapter 5 to dynamically design and implement the virtual gallery adapting to its use. Aiming to satisfy the GDA's current design goals, the application of the example grammar is directed by a special set of state labels so that each generated design of the virtual gallery provides appropriate areas equipped with visual boundaries, visual cues, way finding aids and hyperlinks, and other purposeful objects to support the intended activities of the virtual gallery for each stage. The comparison of the eight different virtual gallery designs generated for the eight stages of the design scenario is shown in Figure 6.14.

6.10.2 Stylistic Characterizations of the Virtual Gallery Designs

The design scenario demonstrates the capability of generative design grammars as a design formalism for virtual worlds. A generative design grammar is able to describe and generate a design language for virtual worlds that captures certain stylistic characterizations shared by all design instances. The corpus of virtual gallery designs generated for the design scenario presents a small set of samples from the design language defined by the example grammar. Although these virtual gallery designs are generated for different purposes, at different moments during the use of the virtual gallery, they provide a similar impression by sharing a sense of design coherency. The stylistic characterizations shared by these generated virtual gallery designs can be outlined from the following three aspects.

- Visualization: the virtual gallery designs generated by the example grammar are coherent in terms of the spatial relations defined for different areas of the virtual gallery, and the use of forms and color schemes for visualizing the virtual world objects.
- Navigation: the generated virtual gallery designs follow the same guidelines for providing way finding aids and hyperlinks to assist the visitors' navigation in the virtual gallery.
- Interaction: the generated virtual gallery designs also have similarities in activating object behaviors in virtual worlds. Therefore, similar experiences can be gained and applied to assist the visitors participating in activities in the virtual gallery.

6.10.3 Technical Implementation

In terms of the technical implementation of the design scenario, each stage of the scenario is implemented in a virtual world developed using virtual world platform Active Worlds. The GDA described in the scenario is implemented on the base of the AW agent package[15]. The design rules of the example grammar for supporting GDA's designing, and a general rule base for supporting the GDA's reasoning, are written using Jess[16], a rule-based scripting language (Friedman-Hill, 2003). A similar implementation can also be achieved in other virtual world platforms, for example, through Linden Scripting Language (LSL) in Second Life.

References

Friedman-Hill, E. (2003), *Jess in Action*, Manning Publications, Greenwich, CT.

15 The AW agent package was developed by Greg Smith and Mary Lou Maher at the University of Sydney through Active Worlds Software Development Kit (SDK).

16 http://herzberg.ca.sandia.gov/jess

Part IV: **The Future of Adaptive Virtual Worlds**

7 The Future and Impact of Adaptive Places

This book has presented an approach to designing adaptive places in 3D virtual worlds using grammars and computational agents. Our approach to adaptive virtual worlds is distinctive, and this reflects on the following two aspects.

Our approach associates computational agents with virtual world occupants, rather than virtual world objects. Generative Design Agents (GDAs) serve as personal design agents to users of virtual worlds that can reason, design and act on their behalf in virtual worlds. Shifting the agency from existing virtual world objects to virtual world occupants frees virtual worlds from being static by providing a reasoning process for the world to adapt to users' needs. GDAs perceive a virtual world as a dynamic and evolving entity in three different perspectives: external, internal, and expected ($W = W_{ext} \cup W_{int} \cup W_{exp}$). By applying generative design grammars, these computational agents dynamically design virtual worlds in response to their use.

Our approach changes the roles of human designers in designing virtual worlds. Designers develop generative design grammars that describe and generate different design languages with different purposes and stylistic characterizations for different virtual places, rather than pre-defining every detail of all possible designs. The GDAs can automate the actual design tasks by applying generative design grammars within virtual worlds in real time. This approach turns designing virtual worlds into a unique process integrated into the virtual world that is highly adaptive and responsive to the changing needs in the virtual worlds.

In this chapter, we explore several considerations for the development and implementation of adaptive virtual places using design rules and agents, and discuss the future and impact of adaptive virtual worlds.

7.1 Technical Environments for Adaptive Virtual Places

Throughout the book, we have illustrated our approach in the context of 3D virtual worlds. However, our approach can be applied more generally in light of the many technical environments for designing virtual places. While we focus on 3D virtual worlds, the approach can generalize to any online environment: a web environment such as a networked game, a mobile device for accessing social media places, a mixed reality environment, and a large public interactive display. For example, if the virtual place is expressed on a web site, the layout, object placement, interaction and navigation can be achieved with a design grammar and agent implemented using:

- HTML to specify the categories of content is equivalent to types of objects in a 3D virtual world such as walls, doors, floors, picture frames, etc.;
- XML and files of different media such as images, videos, sound, and 3D models to specify specific content items is equivalent to the 3D models and media in a virtual world;

- CSS to specify the style rules for each content element or class of elements is equivalent to the layout rules in the virtual world grammar;
- JavaScript to define the local client interactivity is equivalent to the interaction rules that are specific to one user in the virtual world grammar; and
- PHP to describe the shared interactivity shared with other users is equivalent to the navigation and interaction rules that are common to all users.

So, while our development of the design grammar and agent model for adaptive virtual worlds was inspired by place design in 3D virtual worlds, the models are independent of the technology associated with 3D virtual worlds.

7.2 Functional Designs vs. Creative Designs

Generative design grammars provide a computational approach that can be adopted to start formally defining design languages for virtual worlds. In our demonstration of an adaptive virtual galley, the example grammar describes and generates a specific language of virtual gallery designs by capturing specific stylistic characterizations through layout rules, object design rules, navigation rules and interaction rules. These four sets of design rules are developed to define a way virtual gallery designs are composed, according to specific design considerations, in terms of both syntax (visualization: layout and object design), and semantics (navigation and interaction). However, the structure of the grammars, the design operations adopted in the rules and the constraints for directing their applications can be flexibly adjusted to address different design considerations. One example consideration is to prioritize the functional aspect and/or the creative aspect of the generated designs, to suit specific purposes for networked gaming, social networking, online learning, e-commerce, collaborative design and so on.

Generative design grammars both adopt the descriptive and generative nature of shape grammars, and modify some of the original shape grammar properties to suit the purpose of designing 3D virtual worlds. As discussed in Chapter 3, currently the application of a generative design grammar is controllable and predictable to a certain extent so that the designed virtual places will be functional and capable of supporting the intended activities. In other words, currently the purpose of generating virtual world designs that meet the GDAs' design goals has been given higher priority over retaining design ambiguity and emergence during the grammar application, which is essential for enabling novel and unexpected designs. However, for virtual worlds that have less functional demand, generative design grammars can be less restricted to enable more creative and diverse virtual world designs or better balance of both the functional and creative aspects.

Adaptive virtual worlds can be highly functional, as they can be perceived as a kind of architecture customized to every "moment", because the virtual worlds are dynamically designed and re-designed as needed during use. To more precisely and appropriately meet the actual needs, the knowledge required for the GDA's reasoning process particularly interpretation and hypothesizing should be carefully defined and integrated. The application of a generative design grammar is always directed to meet the current design goals hypothesized by the GDA that reflect its interpretation regarding the current needs of the virtual world users and the current state of the virtual world. It is especially subtle for the GDA to interpret the current needs of the virtual world users. A series of future research issues may arise from here; for example, how should a GDA interpret? By receiving instructions from different virtual world occupants explicitly, by guessing the occupants' intentions based on their activities or from the agent's own learning experience? How precise can and should the interpretation be? What kind of interpretation should lead to changes of virtual world designs? The list of questions can continue.

Further, the design rules in current generative design grammars and the constraints for directing their applications are pre-defined by human designers. If a design grammar needs to address different design considerations or to capture different design characterizations, currently these changes will need to be made by designers through changing the components in the grammar. In future studies, more dynamic alternatives can be explored. For example, machine learning or case-based reasoning can be integrated into the approach to enable GDAs to learn from their past design experiences, and use these experiences to enrich and revise generative design grammars automatically, in order to better address different functional and/or creative design considerations.

7.3 Individual, Shared and Collective Design Styles

A generative design grammar describes and generates virtual world designs through the four sets of design rules, therefore the generated designs share a certain design style in terms of visualization, navigation and interaction that are defined by these rules. The current categorization of the design rules provides a base to further detail and expand these stylistic characterizations for defining other virtual world design styles.

In our approach, the agent model currently focuses on defining the five internal computational processes that are specifically developed for a GDA to reason, design and act in 3D virtual worlds. Designing virtual worlds has so far been explored as individual activities carried out by a single GDA. Currently in a collaborative virtual world with the presence of multiple GDAs, when a design problem arises it is assumed that a dominant GDA will reason, design and act on behalf of all occupants. However, the reasoning mechanism of our agent model enables each GDA to reason about the virtual worlds as well as different elements in the virtual worlds, including other GDAs. This means that the GDA model is able to accommodate agent communication. With agent communication, virtual worlds can be designed collaboratively by multiple GDAs. Agents are especially suitable for engineering complex, distributed systems. (Jennings, 2000). By providing GDAs with the capability of collaborative designing, a virtual world becomes a distributed system designed by multiple GDAs. Each GDA contributes to the design of the virtual place supporting their shared activities. During collaborative designing, participating GDAs may each have a different generative design grammar that can produce virtual place designs in a unique design style. This leads to the study of shared design styles for virtual worlds. Various research questions can arise from this future direction. For example, how should individual design styles influence each other? How should different design styles transform and morph to accommodate shared interests and preferences? Should there be a common design style shared by a virtual community, and what is the relation and effect of this shared style for individual styles and identities?

In more recent years the term "collective intelligence" (Lévy, 1997) and its potential application in design i.e. "collective design" (Maher et al, 2010) has appeared widely over a broad spectrum of scientific, sociological and design studies, due to the popularity and adoption of social media and social network amongst a unprecedented large number of participants. Collective design is an emerging application of collective intelligence that utilizes very large networked communities in the design process (that has been traditionally reserved for exclusive domain experts) for solving complex problems in potentially innovative ways. 3D virtual worlds as a social medium have been regarded as a possible open-source design environment that can support these collective design processes with massive participation. In the context of collective design, the issues discussed above regarding shared design style will need to be further scaled up to consider collective design style.

7.4 Extending Adaptive Virtual Places into the Built Environments

The place metaphor is an important concept in our approach to designing 3D virtual worlds. With the place metaphor, we relate designing these networked environments to designing places in the physical world, considering particularly their layout, object design, navigation and interaction. Although designing virtual places has been heavily influenced by designing physical places, the characteristics of virtual worlds being dynamic, adaptive, and interactive can be equally influential to current and future place design in the physical world, providing an ideal test bed for exploring new and innovative place design such as liquid architecture (Novak, 1991), hyper architecture (Puglisi, 1999), and information architecture (Schmitt, 1999), minimising the risk of safety and cost before being implemented as built environments. With the advancement and increased adoption of digital and networked technologies, we are also seeing the gradual blending of both physical and virtual places such as those examples of galleries and museums as discussed in Chapter 5 that have presence in both the physical and virtual worlds.

As a test bed for exploring new and innovative place design in the physical world, adaptive virtual places are especially relevant to interactive architecture (Fox & Kemp, 2009). Interactive architecture is concerned with the built environments we inhabit as the interactive interfaces between us and the embedded digital information and computational intelligence. "Sensitivity", "smartness", and "responsiveness" are the main characteristics of interactive architecture that support "smart and sustainable living" (Jeng, 2012). The design agent model of this book will provide a formal structure to explore and test the reasoning mechanism required for the place to sense and respond. The generative design grammar framework will provide a rule-based design approach to create and/or modify the components of the place as needed.

References

Fox, M. and Kemp, M. (2009), *Interactive Architecture*, Princeton Architectural Press, New York.

Jeng, T. (2012), Interactive Architecture: Spaces that Sense, Think, and Respond to Change, in N. Gu and X. Wang (eds), *Computational Design Methods and Technologies: Applications in CAD, CAM and CAE Education*, IGI-Global, Hershey, PA, pp. 257-273.

Jennings, N.R. (2000), On Agent-based Software Engineering, *Artificial Intelligence*, 117 (2), pp. 277-296.

Lévy P. (1997), *Collective Intelligence: Mankind's Emerging World in Cyberspace*, Perseus Books, Cambridge, MA.

Maher, M.L., Paulini, M. and Murty, P. (2010), Scaling Up: From Individual Design to Collaborative Design to Collective Design, *Proceedings of DCC 10*, Stuttgart, Germany, pp. 581-600.

Novak, M. (1991), Liquid Architecture in Cyberspace, in M. Benedikt (ed), *Cyberspace: First Steps*, MIT Press, Cambridge, MA, pp. 225-254.

Puglisi, L.P. (1999), *Hyper Architecture: Spaces in the Electronic Age*, Birkhäuser, Basel, Switzerland.

Schmitt, G. (1999), *Information Architecture: Basis and Future of CAAD*, Birkhäuser, Basel, Switzerland.

List of Figures

Illustrations by Ning Gu and Rongrong Yu unless stated otherwise.

Index