Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

SUPPORTING PARAMETRIZATION OF BUSINESS GAMES FOR MULTIPLE EDUCATIONAL SETTINGS

Stijn-Pieter van Houten
Alexander Verbraeck

Section of Systems Engineering
Delft University of Technology
Jaffalaan 5, 2628 BX, Delft, THE NETHERLANDS

ABSTRACT

The parametrization of business games benefits from the usage of a multi-tier architecture and software services. This paper shows that the multi-tier concept supports parametrization of business games for multiple educational settings, where for example the number of players, the scenario to use, and the underlying simulation model may differ each time. Using the concept of multi-tier architectures, we implemented a strict decoupling between the graphical user interface layer, the business logic layer, and the resource management layer. This decoupling is supported by using interfaces between the different services. The services have been successfully used to develop, adapt, and use the Distributor Game for different educational settings. Further research will focus on extending the set of services to better support changes in graphical user interfaces that players use and the specification of the services' parameters themselves using a web-based user interface.

1 INTRODUCTION

Managing todays multi-actor systems, such as supply chains, is becoming an increasingly challenging task given market developments such as globalization, and technology developments such as the usage of the Internet. Supply chains are flexible, dynamic and complex networks (Simchi-Levi, Kaminsky, and Simchi-Levi 2003). Networks are the complex, multifaced organizational structures that result from strategic alliances (Webster 1992). Todays managers are faced with increasing competition and globalization in an ever increasingly complex and dynamic system. Multiple tools exist to support managers in gaining more insight into these kinds of systems, such as simulation models or case-studies. Business games are another tool that can be used to support managers. Business games may be used for a variety of purposes, such as creating awareness of the complexity and dynamics of the systems that the business games players are part of, learning and training and to support the exploration of a variety of strategies before choosing one. Business games can be used to prepare managers for the challenges they will face in the decision making arena. Summers (2004) states that new business practices increase the demand for new types of business games.

With respect to the development of business games, trends can be observed that lead to more complex and dynamic games with the goal to reflect the properties of real-world systems better (Chiesl 1990, Gold and Pray 2001, Lainema and Makkonen 2003). We are challenged to take advantage of state of the art technologies in the delivery of business games. Increasingly games are computer supported, using the Internet and the World Wide Web, to provide multi-media content and/or simply using the computer for crunching numbers. In addition, given the challenge of providing support for managers dealing with todays management challenges, we choose to focus on computer-supported business games.

In spite of the many possibilities for computer supported and Internet-mediated business games, there are many problems to overcome. A study by Chang et al. (2003) on the use of business games in Hong Kong found that the long preparation time for facilitators and high start-up costs are a major issue. A survey by Faria and Wellington (2004) yields a number of reasons as to why users of computer supported business games have stopped using them. Some of the most important reasons are: the underlying simulation model is unsatisfactory, the software is too complex, the players did not like the model, and administrative problems when preparing a game session or playing the game. Thavikulwat (1999) observed that the business games used today do not differ much from earlier, non-computer supported games. The cost of creating such computer supported and Internet-mediated business games is still high. Hence, given the issues presented above, providing ways to parameterize business games to support multiple educational

1-4244-1306-0/07/$25.00 ©2007 IEEE

settings while lowering costs when developing and using business games is the logical way to go.

In this paper we present that the concept of multi-tier architectures supported us in developing software services that support parametrization of business games. In section 2 we present background information with regards to multi-tier architectures and the structure of computer-supported business games. Section 3 introduces the different levels of parametrization, the related challenges and how we address these challenges. We describe and illustrate our proof of concept in section 4. We conclude this paper with conclusions and directions for further research in section 5.

2 MULTI-TIER ARCHITECTURES AND THE STRUCTURE OF BUSINESS GAMES

The type of computer-supported business games that we use to address today's management challenges are examples of distributed information systems as described by Alonso et al. (2004). Typically, such a distributed system consists of three layers (Alonso et al. 2004):

1. *a presentation layer.* This layer contains e.g. a player graphical user interface (GUI) or a facilitator GUI to manage a business game that runs on a remote server.
2. *an application layer.* This layer contains services that together form the model that reflects the real-world system we are simulating using the business game.
3. *a resource management layer.* This layer contains e.g. a database to store data and to make it accessible to the services of the application logic layer.

Based on Thavikulwat (2004), we distinguish four services that constitute a software architecture for a computer-supported business game: (1) a representational service, (2) a timing service, (3) a hosting service and (4) a scoring service. A service is a self-describing open component that supports, if designed well, the rapid, low-cost composition of a distributed information system, e.g. a business game (Papazoglou and Geagakopoulos 2003). A representational service supports the representation of a part of the real-world system a game developer would like to abstract into a model as part of a business game (Thavikulwat 2004). This service contains the logic to simulate the effects of decisions made, i.e. it should support players to see cause-and-effect relationships (Thavikulwat 2004). For example, it contains the actors, business logic and products of a supply chain and it defines the relations between them. A timing service can support the progress of time in a business game and a hosting service determines the way users interact with the model of a business game running on a server, e.g. by means of a GUI that can be downloaded, by using web-pages, or another way. Given that a business game can be used for assessment, a scoring service may be needed (Thavikulwat 2004). We present an overview of a multi-tier architecture combined with the services of a computer-supported business game in Figure 1.

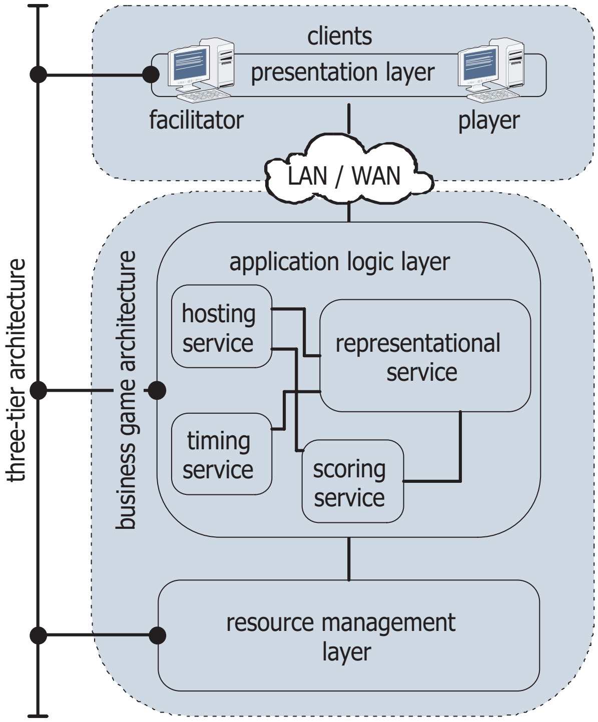

Figure 1: Example of a three-tier architecture combined with business games' services.

3 SUPPORTING PARAMETRIZATION OF BUSINESS GAMES

Parametrization of a business game takes place during two activities when developing and using a business game: during the specification of the business game's model and during finalizing the business game. Specification is executed by a game developer and this activity is about specifying the parameters of the conceptual model, e.g. the actors, their relations, handlers for decision making (Verbraeck and van Houten 2005), etc., and the scenario to support the educational goal of a business game. The finalizing activity is executed by a facilitator and includes preparing guides for users, preparing user names and passwords, etc.. Further, this activity includes several parameters that need to be specified in the case of a business game such as the Distributor Game (van Houten et al. 2005):

- the specification of the scenario to use for a specific educational setting at hand.

- the specification of whether an actor in the business game is computer- or player-controlled, depending on the number of available players.
- the specification of what decisions need to be made by players during game play.
- the specification of the statistics to use and present to players during game play.

The challenge now becomes how to support the parametrization during the two mentioned activities for multiple educational settings. Following the concept of a multi-tier architecture, see section 2, we implemented a strict decoupling between GUIs, e.g. the player application, the application logic layer and the resource management layer. Hence, there are no hard-coded links between the different tiers. In the field of software engineering the term hard-coded refers to unchangeable code. Hard-coded features are built into the software in such a way that they cannot be easily parameterized differently. As a result, changes that are made on one layer do not necessarily impact the other layers. The player GUI in itself is flexible with regards to the specified set of decisions that a player needs to make, the specified set of products for the business game and the selected set of statistics. Depending on the specified parameters' values, the GUI adjusts the several components of the player application automatically (an illustration of the player application can be found in van Houten and Jacobs 2004).

Within the application logic layer, we implemented a same type of decoupling between the different services present at this layer. The parametrization of for example the representational service does not affect the other services, i.e. they do not need to be recoded for each model that is specified for a business game. Parametrization of the representational service is supported by using a library that is designed specifically to be used in business games for the application domain of supply chain management (Verbraeck and van Houten 2005). This library supports the parametrization of a variety of supply chain models, with different products, handlers, number of actors and their relations, etc.. The decoupling between the different services is implemented in such a way that for example depending on the number of specified players, the scoring service automatically generates the data needed to rank them.

The resource management layer keeps track of the transactions that are exchanged between the actors of the model during game play. Again, the decoupling between this layer and the application logic layer is such that regardless how the model for a business game is parameterized, using the earlier mentioned library, transactions are stored and made available to the services of the application logic layer.

The concept that supports the implementation of the decoupling between the different tiers is that of interfaces. An interface describes the syntax of a service description; it specifies the signatures of the operations to be implemented by classes implementing the interface. If we recall the concept of a service as defined by Papazoglou and Geagakopoulos (2003), we may conclude that an interface embodies the concept of a service description. Using interfaces, we implemented for example that a player GUI invokes, during its instantiation, a method called `getProducts()`. The return of this method results in the set of specified products. How this method is implemented in the model that returns this set is not of interest to the player GUI. Hence, the usage of interfaces enabled us to implement the various types of decoupling we mentioned above.

We note that specification of the parameters during the development and finalizing activities and the instantiation of a business game is implemented using input files that contain the using parameters' values. We illustrate the instantiation of a business game in Figure 2. The service that instantiates a business game is part of the application logic layer, though we did not illustrate this in Figure 1.

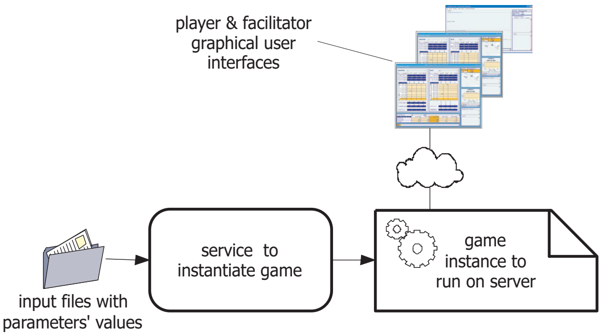

Figure 2: Instantiation of a business game.

4 PROOF OF CONCEPT

The value of the decoupling between the different tiers and within the tiers to support parametrization of business games for multiple educational settings is best illustrated using a case as an example. The Distributor Game (van Houten et al. 2005) is a business game that is used to illustrate the complexity and dynamics of a global supply chain for consumer electronics to players on a MBA level. After a demonstration, a request was made by two teachers from the Erasmus University Rotterdam to modify the Distributor Game to make it usable for students on an undergraduate level. They requested to lower the number of products used in the Distributor Game from 4 to 3, to automate the decisions made by players involving payments and, given the educational setting at hand, to decrease the number of players in the game from 18 to 15. We present a number of representative business games for the application domain of supply chain management in Table 1. None of these business

Table 1: Overview of the number of players for representative business games.

Business game	*Number of players per game instance*	*More information*
Beer Game	4	<web.mit.edu/jsterman/www/sdg/beergame.html>
LOGA Logistics Game	4	Bowersox, Closs, and Helferich (1986)
Littlefield Technologies	class size	<responsive.net/littlefield.html>
Supply Chain Game	class size	<responsive.net/scgame.html>
TAC - Supply Chain Mgmt. Scenario	6 (agents)	<www.sics.se/tac/page.php?id=13>
Harvard Global Supply Chain Management Simulation	class size	<harvardbusinessonline.hbsp.harvard.edu>

games support making comparable changes as requested in case for the Distributor Game.

By modifying the input files that contain the parameters' values, we accomplished the requested changes within 2 hours (!). No changes needed to be made to the player or the facilitator GUIs, to the business logic or to the database that keeps track of the transactions during game play. We do note however that the GUIs are only flexible to a certain extent. Specification of parameters other than the products, decisions made by players or the statistics that a player can consult during game play requires changes in the coding. Thus, parametrization of the GUI used in this research is limited. The business game that was the outcome after the changes was dubbed the Erasmus Distributor Game. Further, we tested the Erasmus Distributor Game by specifying the 15 players to be computer-controlled. By executing the game in this setting, we verified and validated the game.

We used the Erasmus Distributor Game in two sessions with 27 graduate and 13 undergraduate students. To support the dynamics during game play for the second session we specified that in each region one distributor was computer-controlled. During the first session, facilitation was taken care of by us, during the second session the teacher facilitated the complete session and we were only present to provide some technical support, e.g. we instantiated a game instance. The results of the sessions were evaluated by the teacher and taken into account when determining the overall grades for the courses of which the game was part. We evaluated the game on its effectiveness using a questionnaire, of which an overview can be found in Corsi et al. (2006). Based on the results obtained for the questionnaire we conclude that the Erasmus Distributor Game is useful and effective for learning.

5 CONCLUSIONS

Business games are getting more complex and dynamic to better mimic todays systems, such as global supply chains. Hence, the development, usage and management of these games is becoming more challenging and difficult. In this paper we demonstrated that by closely following the multi-tier architecture concept, the parametrization of a business game for multiple educational settings can be supported.

Following the concept of a multi-tier architecture, our architecture consists of a presentation layer, an application logic layer and a resource management layer. The decoupling between these layers, and between the multiple services that are part of the application logic layers, is achieved using interfaces. These interfaces embody the concept of a service description.

The flexibility that is part of our business games' architecture for supply chain management supported us in parameterizing a business game that was modeled after an existing version of a business game. We easily changed the number of products used in the game, the decisions a player needs to make, and the number of players.

Several topics can be addressed in future research. The first topic concerns the player graphical user interface (GUI). Though this GUI is somewhat flexible and can be parameterized in different ways, further research for increasing this flexibility is needed. A second topic is the development of wizard-like functionality that supports for example a teacher instead of a game developer to parameterize a business game. Such a wizard should support a teacher in specifying the type and number of products to use and present available knowledge in an understandable way.

REFERENCES

Alonso, G., F. Casati, H. Kuno, and V. Machiraju. 2004. *Web services. concepts, architectures and applications*. Data-Centric Systems and Applications. Berlin: Springer.

Bowersox, D., D. Closs, and O. Helferich. 1986. *Logistical management*. New York: McGraw-Hill.

Chang, J., M. Lee, K. Ng, and K. Moon. 2003. Business simulation games: the hong kong experience. *Simulation & Gaming: An Interdisciplinary Journal* 34 (3): 367–376.

Chiesl, N. 1990. Interactive real time simulation. In *Guide to business gaming and experiential learning*, ed. J. Gentry, 141–158. East Brunswick:NJ: Nichols/GP Publishing.

Corsi, T., S. Boyson, A. Verbraeck, S. van Houten, C. Han, and J. MacDonald. 2006. The real-time global supply chain game: New educational tool for developing supply chain management professionals. *Transportation Journal* 45 (3): 61–73.

Faria, A., and J. Wellington. 2004, June. A survey of simulation game users, former users, and never users. *Simulation & Gaming: An Interdisciplinary Journal* 35 (2): 178–207.

Gold, S., and T. Pray. 2001. Historical review of algorithm development for computerized business simulations. *Simulation & Gaming: An Interdisciplinary Journal* 32 (1): 66–84.

Lainema, T., and P. Makkonen. 2003, March. Applying constructivist approach to educational business games: Case realgame. *Simulation & Gaming: An Interdisciplinary Journal* 34 (1): 131–149.

Papazoglou, M., and D. Geagakopoulos. 2003. Service oriented computing. *Communications of the ACM*. 46 (10): 25–28.

Simchi-Levi, D., P. Kaminsky, and E. Simchi-Levi. 2003. *Designing & managing the supply chain*. 2nd ed. New York, NY: McGraw-Hill.

Summers, G. 2004. Today's business simulation industry. *Simulation & Gaming: An Interdisciplinary Journal* 35 (2): 208–241.

Thavikulwat, P. 1999, September. Developing computerized business gaming simulations. *Simulation & Gaming: An Interdisciplinary Journal* 30 (3): 361–366.

Thavikulwat, P. 2004. The architecture of computerized business gaming simulations. *Simulation & Gaming: An Interdisciplinary Journal* 35 (2): 242–269.

van Houten, S., and P. Jacobs. 2004. An architecture for distributed simulation games. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R. Ingalls, M. Rosettig, J. Smith, and B. Peters.

van Houten, S.-P., A. Verbraeck, S. Boyson, and T. Corsi. 2005. Training for today's supply chains: An introduction to the distributor game. In *Proceedings of the Winter Simulation Conference*, ed. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, 2338–2345. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Verbraeck, A., and S. van Houten. 2005. From simulation to gaming: an object-oriented supply chain library. In *Proceedings of the 2005 Winter Simulation Conference*, ed. M. Kuhl, N. Steiger, F. Armstrong, and J. Joines, 2346–2354.

Webster, F. 1992, October. The changing role of marketing in the corporation. *Journal of Marketing* 56:1–17.

AUTHOR BIOGRAPHIES

STIJN-PIETER VAN HOUTEN is a Ph.D. student at Delft University of Technology. His research is focused on services for decision support environments, specializing in interactive distributed simulation. His e-mail address is <s.p.a.vanhouten@tudelft.nl> and his web page is <www.tbm.tudelft.nl/webstaf/stijnh>.

ALEXANDER VERBRAECK is a full professor in the Systems Engineering Group of the Faculty of Technology, Policy and Management of Delft University of Technology, and a part-time full professor in supply chain management at the R.H. Smith School of Business of the University of Maryland. He is a specialist in discrete event simulation for real-time control of complex transportation systems and for modeling business systems. His current research focus is on development of generic libraries of object oriented simulation building blocks in C++ and Java. His e-mail address is <a.verbraeck@tudelft.nl>, and his web page is <www.tbm.tudelft.nl/webstaf/alexandv>.