Experiencing the Challenges of Data Warehouse Development: Implementation of a Serious Game

Emergent Research Forum

Mohammed Khojah
University of Colorado-Denver
Mohammed.Khojah@ucdenver.edu

Michael Mannino
University of Colorado-Denver
Michael.Mannino@ucdenver.edu

Abstract

Serious games have been used in education for decades. Playing increases motivation and engagement, fundamental components of the learning process. This paper presents the design and prototype implementation of an information technology strategy game called Emerge2Maturity. The game is about data warehouse development challenges. The game framework combines selection of a data warehouse architecture and assessment of related capabilities. We developed a prototype implementation using a storyboard concept and embedded Microsoft Excel sheet. Although data warehouse development is complex, the game is designed to capture important tradeoffs. The paper describes the details of the game design, show examples of the implementation and explain how the game is going to be evaluated.

Keywords

Game Based Learning, Serious Game, Simulation, Data warehouse, Architecture, Strategy

Introduction

Using games to facilitate learning through simulating real life events is a fundamental goal of design science in the information systems discipline (Kankanhalli et al. 2012; Zichermann et al. 2011). Moreover, attention to computer-based game research is growing. Games have a long history in business education. Computer games facilitate learning about strategy, collaboration, integration, and development maturity (Leger, 2006); processes difficult to grasp using traditional learning methods without practice and experience.

With complexity and ambiguity in designing and implementing data warehouses, students and IT professionals struggle to understand relationships between architecture selection and capability assessment in an organizational setting. Researchers have acknowledged the gap between knowledge learned in school and skills gained through experience (Boyle & Strong, 2006; Kim, Hsu, & Stern, 2006; Mackrell, 2009). Clearly, there is a strong demand for innovative learning approaches to help students experience complex relationships involving technology and organizational structures.

In this paper, we present the design and prototype implementation of a serious game named Emerge2Maturity. The game facilitates learning by IT students and professionals about data warehouse architecture selection and capability assessment. The game design can be applied to other areas of IT strategy selection and capability assessment.

Related Work

The Game-Based Learning paradigm focuses on “achieving the particular objectives of given educational content through game play” (Kim, Park and Baek, 2009: p. 801). Serious games developed through this paradigm provide benefits other than entertainment (Michael & Chen, 2005). In higher education, serious games support interactive learning and engagement through entertainment (Prensky, 2007) with advantages over traditional teaching approaches (Pivec, 2004). Serious games should include elements like fantasy, narrative, and role-playing (Squire et al, 2003). Armstron (2001) suggested some
guidelines for a successful serious game: (1) Predict reactions in realistic simulation. (2) Should be as close as possible to the real world situation. (3) Instruct participant to act like if they are in real situation. (4) The description of the situation must be clear and unambiguous. (5) Give several options to select from.

Three main data warehouse architectures, Independent Data Mart (IDM), Data Mart Bus Architecture (DBA) and Enterprise Data Warehouse (EDW) are used in the game. These architectures comprise the majority in a survey by Alsqour et al. (2012). Research in architecture selection defined factors that influence the selection of a data warehouse architecture (Choudhary, 2010 and Ariyachandra and Watson, 2010). Seven factors are found to have direct effect on architecture selection. However, only three factors were found to have indirect effect through the strategic view construct: information independence, task routineness, and level of sponsorship. The strategic view indicates the perception of an organization on its IT and data sources. Since the game involves the link between strategy and architecture, we used these three factors.

The other approach, maturity model, assumes that the development of a data warehouse is a continuous process evolving over time. The maturity model for data warehouse development (Sen et al. 2012), consists of five stages: initial, repeatable, defined, managed, and optimized. Levels are evaluated through six main features. In the game implementation, we use three features, data size, data quality, and level of integration because of their importance and ease of measurement.

Game Design

This paper presents the development of a role-play game named Emerge2Maturity. Role-playing involves participants making decisions in a realistic simulation.

Learning Difficulties and Objectives

Data warehouse development is a complex process involving several related factors and extended time periods to reach a stable solution. Learners need to align capabilities with architecture selection and balance benefits and risks as organizations acquire capabilities to operate data warehouses.

In response to these difficulties, a serious game is designed to decompose the complexity of data warehouse development process into simple steps occurring in a specific order using common factors across organizations. The game depict tradeoffs between the costs of developing data warehouse architectures and benefits to organizations.

Framework

The game framework combines both aspects of data warehouse deployment in an organization as shown in Figure 1. Organizations should align between their strategic view of a data warehouse and increase their capabilities on operating a data warehouse. In some situations, events occur where organizations have to adjust their strategy or capability based on other internal or external factors.

Figure 1: Interaction between Strategy and Capability
Phases

Emerge2Maturity game has three main phases: initial, progress, and mature. Each phase has three main parts: strategy assessment, architecture matching, and capability manipulation. In the strategy assessment part, players determine the best strategy based on some given information and then match the strategy with the best architecture. In the capability manipulation part, players try to build an architecture using limited resources by maximizing the profits while keeping the risks at a manageable level. The players then take the next phase where inputs are different and some other effects like internal or external events may occur (Figure 2).

![Figure 2: Game Flow](image)

There are five main dimensions in the game: Strategy, Architecture, Capability, Internal Events, and External Events. **Strategy** is assessed by three elements: Information Independence, Task Routineness, and Level of Sponsorship. Information regarding these elements is provided at the beginning of each phase. The player then has to determine the best strategy based on these given data. Players will choose between three main strategies: initial level, progress level, and mature level. The decision should reflect the future goal of the data warehouse architecture design.

**Architecture** reflects the level of integration among the data sources in the organization. Three main types of architectures are selected for the game: independent data mart, data mart bus, and enterprise data warehouse. Each type corresponds with one of the previous strategies. The layout of the architecture as well as the description of the components will be provided. Players should select the architecture that matches with the current strategy plan.

**Capability** is manipulated by data size, data quality, and level of integration. **Data size** is measured by the number of data sources. **Data quality** is measured by the level of controls and transformations for standardization. **Level of integration** is measured by matching and reconciliation efforts among overlapping data sources. There are different data source categories, each with values for features of technology, complexity, and size. These features determine the production capabilities, costs, benefits, and risks associated with each category. Two key performance indicators (KPIs) are used: profits and risks. The players should balance benefits with revenues and the risk with costs. Each architecture requires a level of capabilities. For example, enterprise data warehouse requires all the data sources in the organization to be eventually integrated. Also, the data quality should be at the highest level.

Events are occurrences of temporal or long-term actions that require organizations to modify their strategy and/or capabilities. **Internal event** is an occurrence of actions within the organization such as mergers, acquisitions, and new business initiatives. **External event** is an occurrence of actions that organizations have no control such as a recession, government regulation, and litigation.

**Capability Models**

We have created a prototype of the Emerge2Maturity game using storyboard concept for the strategy assessment and architecture matching parts and using Microsoft Excel software for the capability
In the capability manipulation part, the three phases in the game represent three different projects. For each project, a player decides on the number of data sources ($X$) from each category $i$ ($X_i$). This variable represents the capability of the architecture to produce amounts of queries from the data warehouse. The number of data sources involves a cost function ($C$) with fixed cost ($F$) and variable cost ($V$) for each category $i$.

$$C(X_i) = F_i + V_i X_i$$

Total costs ($TC$) for the number of data sources is a summation of costs for each data source category.

$$TC = \sum_i C(X_i)$$

Revenue ($R$) involves a stochastic demand and production function for each data source category. Demand ($D$) represents the estimated queries that the organization would use. Production ($P$) is the number of queries that each data source can support.

$$R = \sum (\min(P_i, D_i) \times B_i) ; \text{ where } B_i: \text{ benefit per query; } P_i: \text{ production, } D_i: \text{ demand}$$

Finally, the profit ($Pr$) is calculated by the revenue minus total costs:

$$Pr = R – TC$$

The optimization model is profit maximization for each data source category subject to constraints on the budget ($Bd$) for total costs and minimum number of data sources for each category ($MDS_i$).

$$\arg\max_X Pr , \text{ subject to } TC \leq Bd; X_i \geq MDS_i \forall i$$

In the simulation, demand combines expected production and a forecasting error term from a Normal distribution with a mean of 0 and standard deviation depending on data source category. The average and the standard deviation of profits for all categories are calculated. The standard deviation is used to determine risk. To meet demand, players must select the number of data sources from each category that produces the maximum profit. The game is designed so that profit increases when more productions are added until it reaches the point where adding more productions will result in marginal revenue equal to marginal costs. The game uses graphics, sound tracks, and personal characters to enrich the virtual experience. Dialogs to communicate messages between the player and the game are also used.

In this example, players have to determine the number of data sources to include in the architecture (Figure 3). The actual needs for the data sources are hidden from the players. So, they have to find out what is the best combination of data sources to use in the initial phase.

**Future work**

We are working on a complete implementation of Emerge2Maturity using HTML5 and Javascript. Incentives features like ranking, badges and scores will be added in the complete
implementation. We plan to evaluate the fully implemented game on engagement with a questionnaire and learning outcome using a quasi-experiment. In the quasi-experiment, participants from both a conventional graduate course and a Massive Open Online Course (MOOC) specialized in DW will be invited to join a learning session. They will be randomly assigned to one of two groups: the control group and the treatment group. The control group will be taught using a conventional learning course that includes lectures and notes. The treatment group will be given access to play the game. We will devise an assessment instrument that covers important learning processes and knowledge from the Gagne’s instructional events (Gagne, 1970). The assessment instrument will have items involving learning perception, engagement, and evaluations for the game. We anticipate that students who use the game are going to perform better in the test and their enthusiasm and interests toward the course material will exceed the control group.

References


