Chapter XIII
Modeling with System Archetypes: A Case Study

Mahendran Maliapen
IGate Consulting, Canada

ABSTRACT

This chapter examines the application of system archetypes as a systems development methodology to create simulation models. Rapid organizational change and need to adapt to new business models limits the lifespan of both the databases and software applications. With the information representation permitted by archetypes, diagnostic analysis and can help to evolve generic classes and models for representing the real world. Archetypes do not describe any one problem specifically. They describe families of problems generically. Their value comes from the insights they offer into the dynamic interaction of complex systems. The case of a healthcare system is discussed here. As part of a suite of tools, they are extremely valuable in developing broad understandings about the hospital and its environment, and contribute more effectively to understanding problems. They are highly effective tools for gaining insight into patterns of strategic behavior, themselves reflective of the underlying structure of the system being studied. Diagnostically, archetypes help hospital managers recognize patterns of behavior that are already present in their organizations. They serve as the means for gaining insight into the underlying systems structures from which the archetypal behavior emerges. In the casemix model of the hospital, the investigation team discovered that some of the phenomena as described by these generic archetypes could be represented. The application of system archetypes to the strategic business analysis of the hospital case reveals that it is possible to identify loop holes in management's strategic thinking processes and it is possible to defy these fallacies during policy implementation as illustrated by the results of the archetype simulation model. In this research study, hospital executives found that policy modification with slight variable changes helps to avoid such pitfalls in systems thinking and avoid potentially cost prohibitive learning had these policies been implemented in real life.

INTRODUCTION

The system development methodologies applied to information systems are based on the premise that the domain concepts with the use of information engineering tools represent business processes and data directly into software and database models (Jacobsen, 1992; Martin & Odell, 1992; Rumbaugh et al., 1991; Walden & Nerson, 1995). The approach allows rapid prototyping and quick testing.

However, rapid organizational change and need to adapt to new business models limits the lifespan of both the databases and software applications. The lifecycles of change to update these systems with the organic growth of organizations is costly and often requires rearchitecting the very foundational concepts these software systems are based on. In UML, the foundation model and classes are clear (Booch, 1994). The time lag between model concept, use cases, and skeleton software code by applying forward and reverse software engineering to incorporate base changes forces continual rebuilding, testing, and redeployment of systems. These methodologies do not support the representation of information in such a way that would enable system diagnosis, analysis, and simulation to assist and to gauge the "appropriateness" of its real world representation. If these diagnostic changes are not made, the systems suffer creeping obsolescence, and as a result, diminishing utility over time.

The term *archetype* is used to denote knowledge level models which define valid information structures. A system dynamics archetype can be defined as a molecular building block of stocks and flows for a model structure. An archetype is therefore an abstraction of a feedback structure that is known to generate a particular type of behavior. It describes a representative unique behavior, which is the characteristic for that thumbnail part of the greater structure within the model. It can be considered a "plug and play" module into the structure of any other model where such characteristic behavior is required. With the publication of *The Fifth Discipline* (Senge, 1990), there was dearth of interest in using the science of archetypes; explicit system modeling of complex issues can be achieved by examining the whole system. The goal is to understand how the feedback structure of a system contributes to its dynamic behavior. The stocks and flows, the polarities of feedback loops interconnecting them, and shifts in the significance or dominance of various loops in the structure help contribute to this understanding.

At a molecular level, there are two distinct types of causal loop structures, namely balancing and reinforcing loops. The behavior mode of a simulation at any given time is determined by the strongest feedback loop(s). The overall pattern of behavior over time can be related to changing relative strengths of feedback loops. A system with one balancing and one reinforcing loop produces S-shaped development if the reinforcing loop dominates in the first phase, and the balancing loop dominates in the second phase. This form of information representation is consistent and universal.

System archetypes are patterns in corporate structure or behavior that recur again and again. They are that part of an organization, which represents keys to "pattern recognition" activities, incorporated in the discipline of system thinking. Nine archetypes are generally acknowledged as forming the set of tools that reveal patterns of behavior in systems (Senge, 1990).

- Limits to growth (a.k.a. limits to success)
- Shifting the burden
- Eroding goals
- Escalation
- Success to the successful
- Tragedy of the commons
- Fixes that fail
- Growth and under investment
- Accidental adversaries