# Providing knowledge support to scenario development in eLearning Simulations

**Chen Xi**

B.S IN MANAGEMENT INFORMATION SYSTEM, FUDAN UNIVERSITY

**A THESIS SUBMITTED**

FOR THE DEGREE OF MASTER OF SCIENCE
SCHOOL OF COMPUTING
NATIONAL UNIVERSITY OF SINGAPORE
2004

# Acknowledgements

I would like to express my deepest gratitude and thanks to the following:

- My supervisor, A/P Yeo Gee Kin, for her invaluable guidance and patience throughout the duration of the project.
- My parents, for their understanding, love and support.
- My friends, for their constant encouragement.

# Abstract

Scenarios are used in various fields, such as Human-Computer Interaction (HCI), Requirement Engineering (RE) and Strategy Management. They are useful to fill the gap between the users' knowledge of a system and the system's current status. They portray the key situations critical to the problem being investigated in the context. Scenario development in such applications has, however, not been very much discussed, let alone machine-supported.

Ontology is a tool for knowledge sharing by providing a set of commonly acceptable vocabulary. It is also used to facilitate machine processing by providing a certain uniform structured way of representing knowledge.

In an eLearning environment, traditional teachers are replaced by facilitators that mainly streamline the computerized lessons and provide constant guidance and feedback. eLearning simulations are developed to impart various kinds of domain knowledge to the learners or trainees, e.g. business and economics knowledge in a Business Simulation Game (BSG). eLearning simulations take trainees through the simulated processes they will face in specific real-life social or work environment. These processes range from routine procedures to crisis development that usually involves making decision in response to the events that turn up in the process. To provide best learning experience in restricted time-frame, some key learning situations are to be presented to the learners to respond to. Scenarios are used here to represent these situations. A scenario-based learning process takes after the experiential learning model which is an iterative process.

Learners can learn from the process of solving problem and from reflection upon the feedback from the facilitators using such simulations. The use of scenarios is important to the learning process. Thus, scenarios have to be carefully crafted with clear purpose. Usually, scenario crafting requires understanding of simulation internals, in particular, the administrative parameters in BSGs that determine the simulation runs. The steps involved can be knowledge-intensive and time-consuming.

In this thesis, I propose a knowledge-based framework to better support the scenario development process. In an eLearning simulation, the facilitators could create scenarios based on a domain knowledge ontology. The ontology provides support to specific software to help map a facilitators' scenario description to administrative parameters in a BSG. A domain knowledge ontology for a particular BSG is built for a particular BSG as an illustration. Its relation to a scenario management system developed for the same BSG is discussed. The same knowledge-based framework can be used to extend the channels of communication between a facilitator and trainees to enhance the learning process. Further work can be done in extending the framework to support scenario developments in the other fields such as those mentioned in the opening paragraph.

**Keywords**: simulations; scenarios, knowledge management, problem-solving, decision-making

# Table of Content

## List of Figure and Table

# Chapter 1 Introduction

The purpose in eLearning simulations, which include business simulations among others, is to teach trainees the processes they will go through in specific real-life social or work environment. The motivation is that an educator may know well about various rules of thumb or generalizations about what to do in different situations but may not be able to say what action to take clearly in words. There are also situations that decisions are made based on the combination of many basic rules of thumb because the real world is very complicated. How to combine these rules is a kind of art which can only be learned from real experience. The educators may know the art but cannot express it clearly without reference to specific real situations. Thus, there may be difficulties for them to transfer their knowledge to the learners.

eLearning simulation could solve such problems by letting the learners gain experience in a simulated virtual environment. But a tool is still needed to describe or represent specific key situations in this environment. These situations are important in transferring knowledge, which are known by the educators. For example, in an airplane simulation machine, learners can learn how to operate the equipment on the airplane. But the knowledge like when to put down the undercarriage correctly, how to land safely when there is strong wind and so on could only be learned in specific situations. The educators in this case know that in the flying process, there are situations like landing, taking off and engine being down and they will train the learners in these situations to obtain knowledge.

Scenarios are used as a tool to describe specific key situations in eLearning simulation. But few work has been done to systematically analyze the use of scenario in eLearning simulation. I studied in this thesis the use of scenarios in fields like Human-Computer

Interaction (HCI), Requirement Engineering (RE) and Strategy Management in order to find some references for the use of scenarios in eLearning simulation.

Scenarios used in HCI, RE and Strategy Management are used to facilitate the cognition process and help sharing of knowledge between stakeholders.

In human computer interaction (Young et al., 1986, Carroll et al., 1992 and 1999), scenarios are used to portray key instances of users' possible activities or behaviors when they use the system being developed. They help to evoke reflection in the content of design work, helping developers coordinate design action and reflection (Carroll, 1999). In requirement engineering, scenarios are defined as a sequence of actions, which is decided by the capabilities of the artifacts in the target system, showing how a transition from one state to another might occur (Anderson et al., 1992). Scenario based analysis are believed to help requirement analysis by describing external system behavior directly from the user's point of view, supporting early and continued user involvement and interaction, providing guidelines to build a cost-effective prototype, helping validate the requirements specification, and providing acceptance criteria for requirement-based testing (Hsia et al, 1994 & Potts, 1994). In strategy management (Ringland, 1999), managers think through assumptions and hypotheses about processes and actions that are likely to bring up a certain future situation of an organization. They postulate models and procedures that may determine the components of the scenario. The usage of scenarios in strategy management focuses more on the future state as an aid in defining and implementing change, which can help bring to light the direction where one would like to go when making decisions.

Via scenarios, information-rich storylines, which are in natural language in most cases, are constructed to provide vivid descriptions of targeted worlds, such as the expected system in human computer interaction and requirement engineering and the expected decision outcome in strategy management. The benefits that scenarios bring to these fields are explained by principles from cognitive psychology (Carroll, 1992). Because learning also involves cognitive process, the use of scenarios may also help learning.

Because various entities and their actions or capabilities are included in the storylines of a scenario, the contextual knowledge which defines the concept of these entities is incorporated, although it is not always stated explicitly. The sharing of contextual knowledge between educators and learners is the base of their communication by scenario. Knowledge ontology is used there to modeling the context knowledge of scenario in order to facilitate the sharing of knowledge. An ontology can enable knowledge sharing by using a commonly accepted vocabulary specialized to some domain to define its facts. The terms in the vocabulary are actually a collection of conceptualizations of the facts which include objects, concepts, and other entities that are assumed to exist in some area of interest and relationships that hold among them. An ontology usually has a layered structure in order to achieve knowledge reusability and scalability. I developed a business and economics domain ontology in this thesis in order to provide a set of commonly acceptable items which could be used to describe the real world of business and economics. Some implemented ontologies are studied as reference to my building of the business and economics domain knowledge ontology.

To use scenarios in eLearning simulation to provide the best learning experience in a restricted time-frame, scenarios have to be carefully crafted with clear purpose. Scenario crafting requires understanding of simulation internals, in particular, the administrative parameters that determine the simulation runs. The testing of scenarios should consider the user's acceptance to the scenario generation and representation tools and coherence between the purpose and the actual result.

In all, this thesis looks at the detailed steps involved and outlines a knowledge-based approach to support the work. I report my work in the proposed framework using a business simulation that has been used for many years in a course on decision making.

The framework consists of three parts structured in top-down fashion: the domain knowledge ontology, the scenario knowledge base and the specific application which utilizes the knowledge modeled in the previous two parts. The domain knowledge ontology is used to provide a single set of terms and definitions which adequately and accurately covers the relevant concepts in the business and economic world. This can be used to resolve any misunderstandings where terms are used differently. Besides, it is also used to provide a uniform way to represent the business and economics knowledge, which can help computers capture key information in a natural language description and organize the knowledge base in a structured way. In the scenario knowledge base, I also propose a formal way to present the scenarios using XML comparing to the mostly used informal way of natural language.

The contributions of this thesis are as follows. There is a systematic study on scenarios in other fields to get deep understanding of scenarios. There is an analysis of scenario-based learning process. Finally, a business and economics domain knowledge ontology is developed to use as the knowledge support for the use of scenarios in an eLearning simulation.

The thesis is organized as follows. Chapter 2 talks about eLearning simulation and reviews the existing work on scenario development and in eLearning simulations. Deficiencies are found and the use of knowledge ontology is proposed here. Chapter 3 discusses the use of ontology to model the domain knowledge in various fields and present some implemented ontologies. Chapter 4 reviews the use of scenarios in HCI, RE and Strategy Management. Chapter 5 proposes at first the use of scenarios in eLearning simulations. Then the learning process via scenarios is analyzed. Details of the steps involved in scenario development and features of scenarios in eLearning Simulation are discussed based on the study of scenarios in Chapter 4.  Chapter 6 describes a knowledge-based approach to scenario management system in MAGNUS – a Business Simulation Game. The domain knowledge ontology and the scenario knowledge base of MAGNUS are described and I discuss knowledge modeling and engineering of business and economics knowledge. Chapter 6 details the design of the proposed system. The last chapter concludes the thesis.

The content of Chapter 5 is written up in two conference paper “Manage Knowledge in Business Simulation Game, accepted by PACIS 2004, and “Developing Scenarios in eLearning Simulations", accepted by ICCE 2004. Chapter 3 is also written up in the PACIS paper. The content of Chapter 6 is written up in the conference paper "Using Scenarios in Business Simulation Game", accepted by ISAGA 2004.

# Chapter 2 eLearning simulation and scenarios

## 2.1. eLearning Simulations

Simulations are in widespread use by leading instructors in the eLearning society. Simulations in eLearning provide the trainees media which are very close to how people often learn from actual world experiences. The purpose of eLearning simulations is to teach trainees knowledge or skills they will need to go through a certain process in specific real-life work or social environment. As R W Kindley (2002) said, "Simulation brings key experiential learning moments to you, usually by allowing you to fail fast, fail often, but fail safely." By going through simulations, learners have the opportunity to understand and experience the different processes that take place and better prepare themselves for the real thing. The progressive 'virtual environment' that is presented in the simulations portrays the scenarios that offer the opportunities for learning.

There are many different eLearning simulations: situational simulations, technical simulations, procedural simulations, and business simulations. Situational simulations teach interpersonal skills, soft skills and conversational skills, among others. Technical simulations simulate physical systems such as a piece of equipment, which can train people the skill of using such equipment in a virtual environment. One example of such kind of system is the aviation simulation system used to train airline pilots, such as a Boeing 737-800 full-flight simulator. System/Process simulation is a model of a computer system that leads the learner, step-by-step, through a series of end-to-end system processes, which allows the learner to experience how a system functions by interacting with a simulated model. Business simulations teach business management skills and train players in running mock companies and accounting practices.

Business Simulation Games (BSG) have been extensively used to re-create the business environment using computer systems for training or teaching purposes. By engaging themselves within the game, players can increase their understanding of the business and improve the ability in decision making and problem solving. One distinctive feature of a BSG is that it allows learners to test their decisions without having the risk of actually doing it. This makes BSG very desirable for use in business organizations or education organizations like business school, as this will help the trainee get almost-real experience without running a real company and bearing the risks of financial loss. The following are some existing BSGs.

- BusSim (http://www.bussim-ed.com/)

  This game has 7 modules: manufacturing, finance, marketing, operation, strategy, human resources and inventory. Each focuses on different aspects of the management of an enterprise. Some common items which will influence the player's decision are: labor cost/unit, material cost/unit, utility cost, interest rate, retail price (of products), and stock price.
- Total Enterprise Simulation (http://www.eskimo.com/~fritzsch/)

  This game has four dimensions: finance, marketing, production and operation. Human resource management is integrated in all four dimensions. Players in this game should make decisions about loans, product price, market research, production scheduling, etc.

- MAGNUS (http://magnus.comp.nus.edu.sg/)

  MAGNUS is a business simulation game (BSG) developed and maintained by the National University of Singapore. This game has 3 modules: production, finance and marketing with two types of market: consumer market and industrial market. The entities identified are: consumer products, industrial products, plant, labor, raw materials, consultancy fees, R&D fees, economic factors, consumer market, industrial market, transportation cost, warehouse, and cash.

- Capitalism II is published by Ubi Soft Inc. It presents a very complete simulation of a capitalistic economic world. The algorithms and logic in this game are mostly learned form marketing textbooks. Player can do their business in following industries: retail, mining, farming, stock breeding, estate, media, manufacturing, finance and transportation. There is also no existence of government. Players do not need to care about taxing, industry policies and the difference between different countries. Consumers' behavior is influenced by three factors: price, quality and brand image in this game. The players should make decision on price, location, recruitment, and brand policy etc. Research and development is also very important in this game which influences the technical skills producing a certain product and influences, finally, the quality of the products.

## 2.2 Related work of Using Scenarios in eLearning Simulations

### *2.2.1 Flight Mission Scenario Generator*

Flight Mission SCENario GENerator (ScenGen) (Raman, 1988) is a prototype knowledge-based system developed for the Flight Crews Development Group to help the Flight Deck

Engineers generate flight mission scenarios utilizing the tools and problem solving heuristics.

The flight scenarios are developed by experienced flight deck engineers on the basis of their knowledge and flight parameters for different situations to help the workload analysis of the flight crew. The domain knowledge is represented by structured rules and the inference is accomplished through backward chaining as well as forward chaining. The knowledge base uses frames, which are defined as a real or abstract area in the knowledge base, parameters and rules structure with each frame addressing a sub-problem of the main problem. These frames are stored in a hierarchical tree structure. Each of these frames store information on the rules and parameters involved in its area.

### *2.2.2 Scenario Management System of MAGNUS*

Lua and Yeo (2003) proposed a scenario management system for MAGNUS. The system consists of 3 parts: the scenario identification part, the game parameter selection part and the knowledge part. The purpose of developing this system is to help the game administrators create, manage, and manipulate scenarios to pose various business situations to influence the decision making of the players. The scenario identification part of the system is used to parse the free text scenarios input to find out some key information. Then the key information is mapped to the parameter list to select relevant game parameters. At last the values of these parameters are modified as the output. The whole process is supported by the game knowledge.

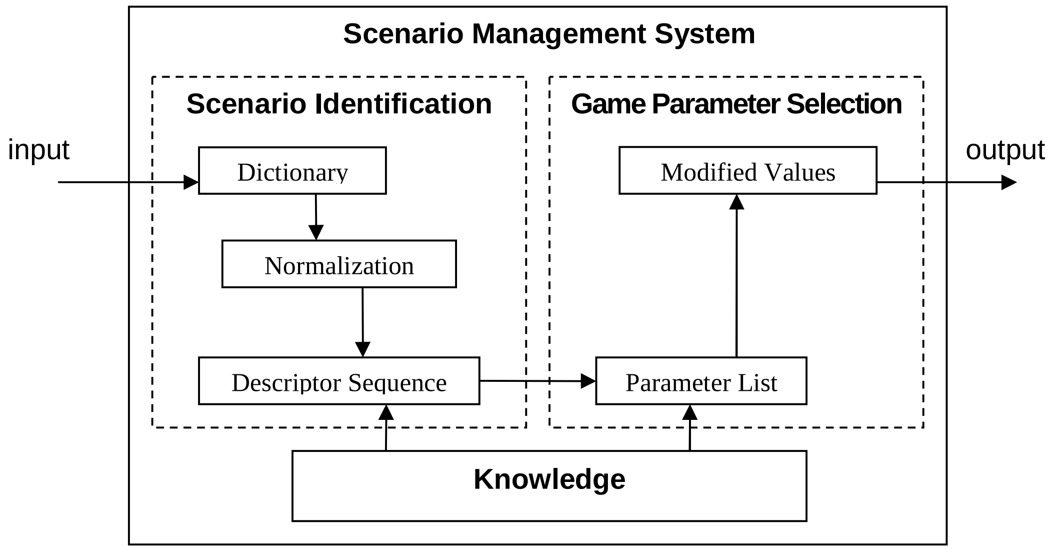


Figure 1: Overview of the Scenario Management System

### *2.2.3 The characteristic and deficiency of present use of scenario in eLearning Simulation*

Both the scenario management system of MAGNUS and ScenGen are knowledge based systems. They are supported by domain knowledge. They have knowledge base which contains rules and sets of vocabularies for catching key information. But in these works, problems like how the domain knowledge is represented and support of developing knowledge base are not stated clearly.

On the other hand, these two systems involve the creating and managing of scenarios. Scenarios are created based on the domain knowledge and they will be processed according to the knowledge base. But how the domain knowledge supporting the generation of scenarios are not discussed. For example, in Lua and Yeo's work (2003), a dictionary, which contains key word, is used to capture key items in free text scenario description. But what kinds of words should be included in this dictionary and how to organize the words according to the specific domain are not discussed?

Another problem is how to choose and organize the eLearing simulation system's parameters according to the domain knowledge which may facilitate the mapping from scenarios to parameters.

Thus, a tool is needed to solve the above problems. I propose a knowledge ontology to be this tool. Related works of ontology will be discussed in next chapter. Before going to the discussion of building knowledge ontology for eLearning Simulation, I discuss about scenarios in Chapter 4 and scenario-based eLearning process in Chapter 5 to review characteristics of scenarios and the learning process as preparation for the developing of knowledge ontology.

# Chapter 3 Using ontology to model the context knowledge

## 3.1 What is ontology?

The word ontology comes at first from Philosophy. It is then used widely in the field of artificial intelligence, knowledge-based system, knowledge management and so on. The definition of ontology is expressed in various ways. Here, I use Asuncion and Benjamins's (1999) definition which based their summarization of Chandraskearan et al.'s work (1999): "Ontologies aim at capturing domain knowledge in a generic way and provide a commonly agreed understanding of a domain, which may be reused and shared across application and groups."

From this definition, one characteristic of ontology can be inferred: it is a model of knowledge in order to enable knowledge sharing. To share the knowledge, an ontology uses a commonly accepted vocabulary specialized to some domain or subject matter to define the facts in this domain. The terms in the vocabulary are actually a collection of conceptualizations of the facts. The facts here include objects, concepts, and other entities that are assumed to exist in some area of interest and relationships that hold among them (Genesereth and Nilsson, 1987).

Because the process of modeling knowledge is very laborious, it would be economical if the knowledge model could be used for not only one application and if it could be modified easily. This problem leads to the consideration of ontologies' reusability and scalability (Jarrar and Meersman, 2002). Reusability requires the ontology to fit the needs of different kind of tasks. The higher the reusability, the more tasks the ontology can serve (Russ et al., 1999). Scalability implies that the ontology should be able to manage growth and evolution. The process of building up a ontology involves generating consensus on conceptualizations. Generating such consensus is a mental process and implemented by exemplifying, testifying, investigating, etc (Guarino and Giaretta, 1995). Difficulties and disagreements normally appear at a "deeper" level of abstraction as (Gangemi, 1998) indicate that "disagreement persists at a deep, 'ontological' level'…" For example, people may believe the fact that economic environment will influence the economic growth of a country but how the influence works will bring disagreement. To weaken the impact the "disagreement" will cause to reusability and scalability, Jarrar and Meersman (2002) advice to separate conceptually and architecturally, the relatively stable fact of ontology from its rule. His idea is supported by many other researchers. For example, Researchers from problem solving area (Chandrasekaran and Johnson, 1993, Clancey, 1992, and Swartout and Moore, 1993) have proposed the idea of structuring the knowledge into different layers of abstractions, while Steels (1993) proposed a componential framework that decomposes knowledge into reusable components. In most case, a knowledge could have two part: domain factual knowledge and reasoning knowledge which is about how to achieve various goals (Chandrasekaran et al. 1999, and Gruber, 1995).

### 3.2 Methods of building ontology

Several formal methods of building ontology have been proposed, such as Gruinger and Fox's (1995a) and Uschold and Gruninger's (1996). But few are domain-independent. However, we still can find something common. Firstly, these methods all start from the identification of the ontology's purpose and the need for domain knowledge acquisition.

Secondly, they also identified the need for ontology evaluation. Thirdly, these methods model the world using concepts, instances, relations, functions and axioms.

- Concepts are used in a broad sense. A concept can be anything about which something is said and therefore, could also be the description of a task, function, action, strategy, reasoning process, etc. Concepts are usually organized in taxonomies.
- Relations represent a type of interaction between concepts of the domain, such as subclass-of and component-of.
- Functions are a special case of relations in which the one set of elements map to another set of elements uniquely. An example of functions is as follow, with input of material, products could be produced.
- Axioms are used to model statements that are always true.
- Instances are used to represent elements.

### 3.3 Criteria and principles for building ontology

Besides the principle of reusability and scalability, Gruber (1995) has proposed some criteria for building the ontology:

- Clarity and objectivity means that the ontology should provide the meaning of defined terms by providing objective definitions and also natural language documentation.
- Completeness means that a definition expressed in terms of necessary and sufficient conditions is preferred over a partial definition.
- Coherence permits inferences that are consistent with the definitions.

- Minimal ontological commitments means to make as few claims as possible about the world being modeled, giving the parties committed to the ontology freedom to specialize and instantiate the ontology as required.

Asuncion and Benjamins (1999) have done a survey which summarizes a set of principles that have been proved useful in the development of ontologies to meet the above mentioned criteria.

- Ontological Distinction Principle (Borgo et al 1996), means that classes in an ontology should be disjoint.
- Diversification of hierarchies means to increase the power provided by multiple inheritance mechanisms (Arpirez et al, 1998).
- Modularity is (Bernaras et al, 1996) is to minimize coupling between modules.
- Minimization of semantic distance between sibling concepts (Arpirez et al, 1998), means that similar concepts are grouped and represented using the same primitives.
- Standardization of names whenever possible (Arpirez et al, 1998).

Some implemented ontologies are reviewed in the following section. Their strength and weakness are discussed in section 3.4.4.

## 3.4 Implemented Ontologies

### *3.4.1 Enterprise (Fraser et al, 1995)*

The Enterprise Project is designed to improve or, if necessary, replace the existing modeling methods with a framework for integrating methods and tools which are appropriate to enterprise modeling and the management of change. The Enterprise

Ontology which is the base of the framework is conceptually divided into several major sections. These are listed below, along with a few of the most important concepts for each.

*Meta-Ontology*: Entity, Relationship, Role, Actor, State of Affairs;

*Activities and Processes*: Activity, Resource, Plan, Capability;

*Organization*: Organizational Unit, Legal Entity, Manage, Ownership;

*Strategy*: Purpose, Strategy, Help Achieve, Assumption;

*Marketing*: Sale, Product, Vendor, Customer, Market;

### *3.4.2 TOVE (Gruninger et al, 1995b)*

The goal of the TOVE (Toronto Virtual Enterprise) project is to create an enterprise ontology for the following purpose: (1) provides a shared terminology for the enterprise that every application can jointly understand and use, (2) defines the meaning (semantics) of each term in a precise and as unambiguous manner as possible using First Order Logic, (3) implements the semantics in a set of Prolog axioms that enable TOVE to automatically deduce the answer to many "common sense" questions about the enterprise, and (4) defines a symbology for depicting a term or the concept constructed thereof in a graphical context. The TOVE ontologies constitute an integrated enterprise model, providing support for more powerful reasoning in problems that require the interaction of the following ontologies:

- Activities, states, and time
- Organization
- Resources
- Products

- Services
- Manufacturing
- Cost
- Quality

### *3.4.3 Metamodel of UML (Mullar, P-A, 1997)*

Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system. UML gives people a standard way to write a system's blueprints, covering conceptual things such as business processes and system functions, as well as concrete things such as classes written in a specific programming language, data base schemas, and reusable software components. UML is a widely adopted standard that represents best practices and lessons learned from well over a decade of experience of modeling complex software-intensive systems. Elements are the building blocks of UML, and comprise model elements and visual elements. Model elements represent abstractions of the system being modeled. The elements used in the Metamodel of UML present “type/instance” and “type/class” dichotomies. The various interactions between elements are presented using relationships.

A *type* specifies a domain of values and a set of operations applicable to those values. A class implements a type: it provides the representation of attributes and the implementation of operations (methods).

UML defines five kinds of *relationships*: dependency, association, generalization, transitions and links. The first three are used more commonly than the last two.

An *association* specifies a bi-directional, semantic connection between types.

A *generalization* specifies a classification relationship in which an instance of a subtype may be substituted for an instance of a supertype.

A *dependency* is a unidirectional usage relationship between elements (both model and visual). The dependency relationship connects elements within the same model.

UML uses state machine to describe the behavior of objects (instance of a certain class). The key conceptions of state machine are state, event, and transition, etc. state describes the condition of an object at a given point in time. An event corresponds to the occurrence of a given situation in the problem domain. In contrast to lasting states, an event is by nature an information snapshot that must be treated immediately. An event is used as a trigger to go from one state to another – a transition.

### *3.4.4 Features and deficiencies of the implemented ontologies*

TOVE and Enterprise are previous attempts to build ontology for enterprise. These two ontologies provided a very complete set of vocabularies which could be used for building other ontologies involving the domain knowledge of enterprises. But they focused more on the internal of enterprises and did not pay much attention to the outside world of enterprises. Another problem is that, in these two ontologies, the items in the vocabularies were not organized in a uniform way. As a result, they could not support computer processes very well. UML is widely adopted for modeling conceptual or concrete things. The Metamodel of UML provides a uniform way to describe entities. But it has not been elaborated to work for modeling a specific domain, for example, the business and economics world. Nonetheless, it is a good base for developing knowledge ontology in eLearning Simulation.

# Chapter 4 Scenario in Context

The origin of the term "scenario" is from theatrical studies, and then the concept was brought into research by military and strategic gaming (Jarke et al, 1998). Now it is widely used in various fields. Relatively little work has been done to analyze the use of scenario in eLearning Simulation. But it is also common to see this term being brought up in other fields like human computer interaction, requirement engineering, and strategy management. Thus, I tried to analyze the use of scenarios in these fields to find some useful features, such as the content elements of scenario, the developing methods of scenarios, and the main function of the scenarios.

### 4.1 Scenario in Strategic Management

Blanning (1995) defines scenario in strategic management from both a narrow sense and a broader sense. From the narrow sense, a scenario is a description of future situations of an organization. From the broader sense, it consists of (1) assumptions and hypotheses about processes and actions, (2) models and procedures used to determine the components of the scenario, (3) quantitative and qualitative measures, and (4) decisions, situations and interpretations.

Case-based reasoning from artificial intelligence is considered a promising strategy for linking decision theory to scenarios based on similar situations analyzed (Klein et al, 1993 and Tsatsoulis et al, 1997), which is based on psychological theories of human cognition. It rests on the intuition that human expertise does not depend on rules or other formalized structures, but no experiences. (Kolodner, 1993). Case as mentioned here is actually what we mean by scenario, a description of a closed-world assumption.

Decision theory primarily concerns the problem of making choice among a set of alternatives. The decision problems are usually presented as a decision tree which typically illustrates the options and choices a decision problem will face, starting from leaves – outcomes and moving toward the root – decisions (Pomerol, 1997). These options and choices are actually the actions reacting to expected events while these events will also impact the outcomes of actions (Raiffa, 1968). Based on this analysis, a scenario is then simply a set of conditional actions consistent with the decision tree.

Case-based reasoning can help decision theory to establish decision variables and predict outcomes, and to assign subjective probabilities and utilities on the basis of the historical information and domain knowledge. Meanwhile, with the help of decision theory, case-based design now could have an automated way to evaluate alternatives and to explain the rationale and intent behind design decisions.

The usage of scenarios in strategic management clearly focuses on future state scenarios as an aid in defining and implementing change, which can help bring to light the direction where one would like to go when making decisions. On the other hand, scenarios that are considered plausible, but are not selected as most-likely-to-happen, are not discarded. Instead, they provide an element of flexibility to the chosen plan (Jarke et al., 1998).

For the generation of scenario in strategic management, researchers have proposed some methods (Brown 1968, Wack 1985a and 1985b), among which the most clearly structured one is intuitive logic method (Bui et al., 1996). Intuitive logic method involves five steps to generate scenarios: (1) analyse the organisation decisions, (2) identify key decision factors, (3) analyse environmental forces, (4) define scenario logic, and (5) analyse implications for decisions and strategies.

Due to the nature of strategic management, there are still problems in the developing of scenarios (Jarke et al., 1998): (1) lack of well-defined objectives, (2) lack of sound assumption, (3) lack of structure, (4) lack of definite level of detail when describing scenarios, (5) gap between scenarios' discreteness and realities' continuousness, (6) exhaustivity of scenarios, and (7) scenarios as processes and not outcomes. I will take these problems into consideration when developing scenarios in eLearning Simulation.

### 4.2 Scenario in Requirement Engineering (RE)

The main task of requirement analysis is to generate specifications that describe system behavior unambiguously, consistently, and completely (Hsia et al., 1994). It is a process to determine what set of capabilities of a system best satisfies the clients' needs (Anderson and Durney, 1993). Requirement can be presented in terms of state transitions that an artifact in the system is intended or expected to affect, both positive and negative (Anderson and Durney, 1993).

Scenarios are believed to be able to help requirement analysis by describing external system behavior directly from the user's point of view, supporting early and continued user involvement and interaction (Potts et al., 1994), and providing guidelines to build a cost-effective prototype, helping validate the requirements specification, and providing acceptance criteria for requirement-based testing (Hsia et al., 1994). Scenarios are defined as a sequence of actions, which is decided by the capabilities of the artifacts in the target system, showing how a transition from one state to another might occur (Anderson and Durney, 1993). Scenario analysis is the process of understanding, analyzing, and describing system behavior in terms of particular ways the system is expected to be used (Hsia et al, 1994). The product of it could be a document that consists of sets of correct complete, consistent, and validated scenarios which represents capabilities of the target artifact that will be needed if the desired behaviors are to be realized.

In the field of RE, the method for generating scenarios is also studied by some researchers. In (Hsia et al., 1994), the authors also propose a systematic, formal method for scenario analysis, which consists of several steps: scenario elicitation, scenario formalization, scenario verification, scenario generation, prototype generation and scenario validation. Scenario elicitation is to describe and classify required scenarios from the user's view. Scenario formalization is to present scenarios in a regular grammar – in a formal or semi-formal model. Scenario verification is to verify the formal model to uncover inconsistencies, redundancies, and internal incompleteness in the scenarios. Scenario generation is to generalize scenarios in the verified formal model. Prototype generation is to construct the expected system on the basis of the scenarios generated. Scenario validation is to validate the prototype to uncover invalid scenarios.

From another point of view, Rolland et al. (1998) proposed a classification framework of scenario based approaches in requirement engineering. They also stated that loosely speaking, use of examples, scenes, and narrative descriptions of contexts, mock-ups and prototypes can all be called as scenario based approach.

The proposed scenario framework suggests considering scenarios along four different views: form view, content view, purpose view, and lifecycle view, each view allowing capturing a particular relevant aspect of scenarios. Each view is also depicted using a facet way which is measured by a set of relevant attributes. These views can also be regarded as some key contents one should consider when generate a scenario.

*The form view* deals with the expression mode of a scenario. Are scenarios formally or informally described, in a static, animated or interactive form? *Form view* has two facets: description facet and presentation facet. The former one deal with what is used to describe a scenario while the latter one deals with in which form it is described.

*The contents view* concerns the kind of knowledge which is expressed in a scenario. Scenarios can, for instance, focus on the description of system functionality or they can describe a broader view in which the functionality is embedded into a larger business process with various stakeholders and resources bound to it. The content view has four facts: abstraction facet which concerns about the abstraction level of a scenario, context facet which aims at classifying scenario approaches according to the amount of contextual information they capture, argumentation facet which is the explanation of a scenario, and coverage facet which aims to classify the scenarios based approach according to the kind of information they capture.

*The purpose view* is used to capture the role that a scenario is aiming to play in the requirements engineering process. Describing the functionality of a system, exploring design alternatives or explaining drawbacks or inefficiencies of a system are examples of roles that can be assigned to a scenario.

*The lifecycle view* suggests considering scenarios as artifacts existing and evolving in time through the execution of operations during the requirements engineering process. Creation, refinement or deletion are examples of such operations. The question of persistency versus transience is also addressed in this view. Lifecycle view has two facets: lifespan facet and operation facet. The lifespan facet distinguishes scenarios between transient scenarios and persistent scenarios. Transient scenario has short lifespan while persistent ones exist as long as the documentation of the project they belong to. The operation facet describe about what kind of operations a scenario brings out.

### 4.3 Scenario in Human Computer Interaction

In the field HCI, scenarios are used to portray users' possible activities or behaviors when they use the system being developed. It is a detailed description of a specific instance of human-computer interaction (Young et al. 1987, Carroll et al. 1992 and Carroll, 1999). The use of scenario in HCI can help to overcome the difficulties caused by the narrow scope of theories derived from work in the cognitive sciences by using scenario to supplement their reliance on empirical data (Young et al. 1987). More specifically, during the system developing process, "(1) scenarios can help evoke reflection in the content of design work, helping developers coordinate design action and reflection; (2) scenarios are at once concrete and flexible, helping developers manage the fluid of design situations; (3) scenarios afford multiple views of an interaction, diverse kinds and amounts of detailing, helping developers manage the many consequences entailed by any given design move; (4) scenarios can also be abstracted and categorized, helping designers to recognize, capture and reuse generalizations; (5) scenarios promote work-oriented communication among stakeholders, helping to make design activities more accessible to the great variety of expertise" (Carroll, 1999).

### 4.4 Benefits of using scenarios

Carroll suggested that the benefit of scenario in RE and HCI can be explained by principles from cognitive psychology:

- "Concrete material is cognitively accessed and interpreted more easily and more thoroughly. For example, people can remember a prototypical instance far better than they can remember the definition of the category to which that instance belongs" (Medin et al., 1978, and Rosch et al., 1976).
- "Incomplete material is elaborated with respect to one's own knowledge when it is encountered. This process of elaboration creates more robust and accessible memories, relative to memories for more complete material" (Wertheimer, 1938).
- "When people communicate, they first summarize or allude to relevant background information, and then present what is novel. This structure cues the listener or reader as to what the speaker or writer considered to be novel information, easing

comprehension and analysis" (Haviland et al., 1974). Narratives, including scenarios, tend to follow this structure.

- "People tend to overestimate the relevance of things that are familiar to them (Kahneman et al., 1972 and Tversky et al., 1974). This tendency is extremely difficult to mitigate, but can be managed by making exceptional patterns vivid. Narratives structures like scenario represent an excellent vehicle for managing this phenomenon."

It is reasonable to believe these benefits of scenarios would work in other cognitive fields such as learning. More specifically, it is possible to use scenarios to provide a base for creating more robust knowledge.

### 4.5 Scenario Generation

I found, from works in Requirement Engineering, some reference to the way of scenario generation. The scenario generating method proposed by Hsia et al (1994) and the classification framework proposed by Rolland et al (1998) are considered from two orthogonal dimension of generating scenario. Method in (Hsia et al, 1994) is much more from a lifecycle view of generating process. Although the framework in (Rolland et al, 1998) is proposed to classify the research works in scenario-based analysis, we can regard it as a set of considerations to apply during the process of generating scenarios. The combination of these two methods could show a more complete framework of scenario generation. The scenario generating lifecycle is decided at the very first. Then, at the elicitation step, the purpose and content of scenarios should be decided in an informal way – describing the scenario in natural language. And problem of the formal presentation of scenario is going to be considered in the formalization way. These three problems: form, content and purpose will be checked again and again in the later steps.

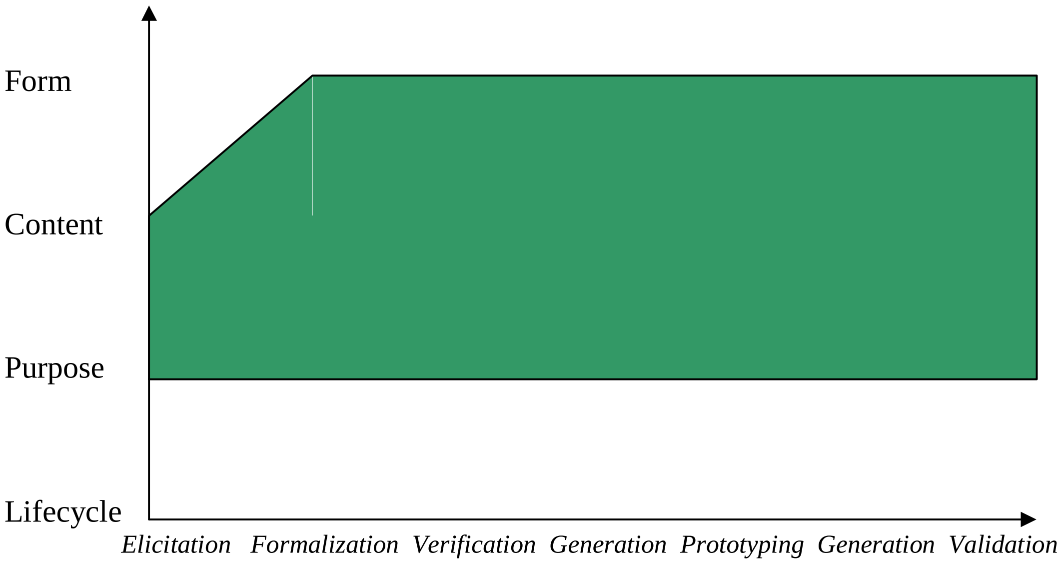


Figure 2: Scenario Generation Framework

Although this framework is generated mainly based on research works in HCI and RE, it can also be referenced by other fields. The representation mode, the knowledge included, the purpose of using scenario are some common concerns when using scenario. The steps in the lifecycle could also be a guide line for other scenario-related work. The process of a specific work may not be the same as the one in the framework, but, should be an iterative process of generation-modification-validation.

### *4.5.1 Elements of scenarios from the content view*

Scenarios can be used for different purpose and represented in different ways but there are some common elements in the content of scenario which could be highlighted as shown in the following definition of scenario.

Firstly, scenario is a description of situations.

- “The consequence of possible decisions, measures and events are called a scenario.” (Mesarovic et al, 1974)
- “A description of the development of the object of the analysis in alternative framework conditions.” (Hansmann et al, 1983)
- “A holistic script about the future, which defines working environment of a company based on different assumptions and describes the paths from present to the future.” and ”Possible, but not necessarily probable views of the future.” (Meristö, 1991)
- “At the most general level, scenario refers to a situation or more precisely (since it has a temporal component), an episode.” (Wright, 1992)
- “A description of a possible future situation relevant for specific LCA applications, based on specific assumptions about the future, and (when relevant) also including the presentation of the development from the present to the future.” (Pesonen et al, 1998)

In the first three definitions, scenario as a path from present to future is also indicated. As path is mentioned here, a measure used to scale the time length of this path is implied.

Secondly, beside the above mentioned characteristics, scenario involves uncertainty.

- “Scenarios are used to describe that part of the organizations’ environment for which projections are difficult or even impossible. **Scenarios give the possibility to prepare for alternative and uncertain future options** without knowing anything about the probability of the possible outcomes. This makes the scenarios different from forecasts. Effective scenarios are distinct, logical and they are different enough from each other so that they are able to describe the central changing factors of the future and place questions on existing assumptions.” (Vartia, 1994)

- “A description of a complex future situation that occurrence **cannot be predicted for sure** as well as the presentation of the development that could lead from the present to the future.” (Gausemeier et al,1995)
- “In contrary to prognoses, scenarios do not try to predict the future. Scenarios do more try to "throw light on" **thinkable future possibilities**.” (Scholz et al, 1996)
- “One **possible picture of future** conditions of the object and its environment; above mentioned conditions are described by characteristics of the results of given sequences of events (situations) and factors which disturb the natural run (evolution) of these sequences.” (Bartusik et al, 1997).

Thirdly, scenario is always under a certain context which defines the knowledge of the specific domain.

- “An important feature of a scenario is that it depicts activities in a **full context**, describing the social settings, resources, and goals of users.” (Nardi, 1992)
- “Scenarios embody information about **the environment**, person, and details of screen and input devices as well as other objects and activities happening.” (Verplank, 1993)
- “A scenario is a description of the world, **in a context** and for a purpose, focusing on task interaction. It is intended as a means of communication among stakeholders, and to constrain requirements engineering from one or more view points (usually not complete, not consistent, and not formal)”. (Jarke et al, 1998)

The “working environment” in Meristö’s (1991) definition and “organization environment” in Vartia’s (1994) definition are also concerns about such a knowledge context.

## 4.6 Findings about scenarios from context

Synthetically, via scenarios, information-rich storylines, which are in natural language in most cases, are constructed to provide vivid descriptions of targeted worlds, such as the expected system in human computer interaction and requirement engineering and the expected decision outcome in strategy management. Because of scenarios' ability to give concrete narration, cognitive process like learning could also benefit from scenario according to Carroll's analysis as discussed in section 4.

Besides, because various entities and their actions or capabilities are included in the storylines of a scenario, the domain knowledge which defines the concept of these entities is incorporated. Many other authors have pointed out the context dependence of a scenario, but the bigger context which relates to the domain knowledge is only implicit in their discussion. An explicit model of the domain knowledge should be developed to support the use of scenarios.

When developing scenarios, several aspects are considered like the form, the purpose, the lifecycle, and the content. Besides descriptions of actions and capabilities of entities, the content of scenario could also include descriptions of uncertainty factors and time factors. The lifecycle of developing scenarios could be an iterative process involving generation and verification. Detail of these problems in the field of eLearning Simulation will be examined after studying the features of scenario based learning process which will be discussed in Chapter 5.

# Chapter 5 Learning through scenario based eLearning simulations

To have a clear idea of the details about scenarios mentioned in the Chapter 4, under the context of eLearning Simulation, the process of scenario-based learning should be examined.

## 5.1 Scenario-based learning

### *5.1.1 The experiential model of learning*

Scenario-based learning takes after the experiential model of learning (Kolb, 1984). Kolb's model of experiential model of learning consists of four elements: concrete experience, observation and reflection, the formation of abstract concepts and testing in new situations. The model is represented as **Figure 3**:

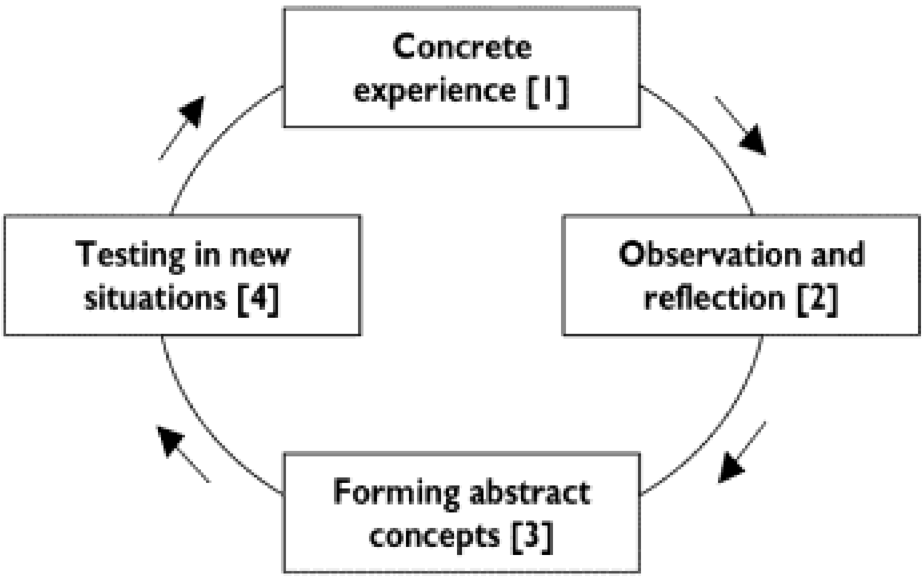


Figure 3: Experiential Model of Learning

Kolb and Fry (1975) argue that the learning process could begin at any one of the four steps - and that "it should really be approached as a continuous spiral". However, "the learning process often begins with a person carrying out a particular action and then seeing the effect of the action in this situation" – Concrete experience in **Figure 3**. Following this, "the second step is to understand these effects in the particular instance so that if the same action was taken in the same circumstances it would be possible to anticipate what would follow from the action" – Observation and reflection. The third step "would understand the general principle under which the particular instance falls" – Forming abstract concepts. "Generalizing may involve actions over a range of circumstances to gain experience beyond the particular instance and suggest the general principle. Understanding the general principle does not imply, in this sequence, an ability to express the principle in a symbolic medium, that is, the ability to put it into words. It implies only the ability to see a connection between the actions and effects over a range of circumstances."

"An educator may well have various rules of thumb or generalizations about what to do in different situations. They will be able to say what action to take when say, there is need to sell more product but they will not be able to verbalize their actions in commonly understandable words. There may thus be difficulties about the transferability of their learning to other settings and situations."

"When the general principle is understood, the last step, according to David Kolb is its application through action in a new circumstance within the range of generalization –

testing in new situation. In some representations of experiential learning these steps, (or ones like them), are sometimes represented as a circular movement. In reality, if learning has taken place the process could be seen as a spiral. The action is taking place in a different set of circumstances and the learner is now able to anticipate the possible effects of the action."

### *5.1.2 Scenario-based learning in eLearning Simulation*

According to the experiential model of learning, I propose the scenario-based learning model in this section.

In an eLearning simulation environment, there are mainly two kinds of participants: the facilitator or administrator and the learners. The role of administrator is usually taken by the educator. The administrator sets up the simulation environment and the learners are trained, mostly in the form of solving problems, in such an environment. Administrators have the knowledge of the context knowledge of the target learning field, such as what kind of entities exist, what attributes entities have, and what actions entities could perform etc. They have also the knowledge of reasoning of the consequences and trends of events, which are the depiction of entities' attributes or actions. In other words, they have in their mind, an image of the future wheel (Glenn, 1994) of the target learning field as shown in **Figure 4**. There are three platforms in the figure, namely the historical forces, the current impacts and states and the future consequences. On each platform, there are groups of circles which stand for events. The smaller circle stands for the consequence of the bigger circle linked to it. The line connected between circles in two platforms means the upper one is the trend of the lower one. But, the fact is that they may have difficulties to tell the learners how to reason without reference to specific social situation. Scenarios are used to help them at this point. Added in **Figure 4** are two rounded rectangles, representing two scenarios which is the description of states which may be true in reality and possible events which may cause changes to the states. The enclosed circles in this rectangle are the events included in these scenarios. By presenting these scenarios to the learners, administrators situate the learners in a simulated reality which is new compared to the situation described in class and text books.

The learners then test the relatively abstract concepts they learnt by solving problems in these new situations. Their problem-solving processes are similar to the processes of decision making. The decision problem can be presented as a decision tree which consists of nodes standing for options of possible events. These events are indeed the part of the circles connected with each other in **Figure 4**. The events included in the scenarios may exist in the decision tree in learners' mind as shown in **Figure 5**. As a result, the learners can then deduce the possible outcome according to the decision tree with reference to the scenario presented. By such a process, the learners gain concrete experience.

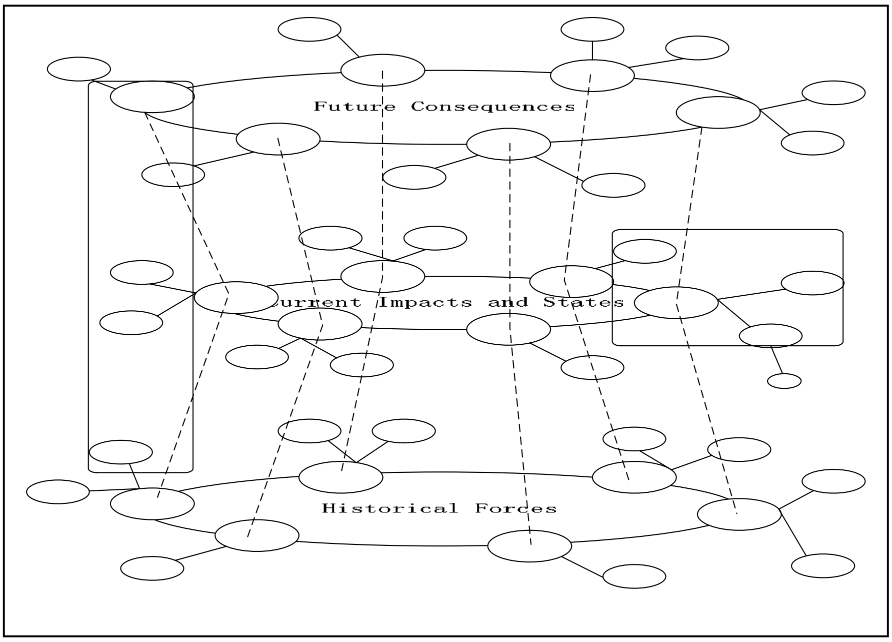


Figure 4: Future Wheel in Administrators' Mind

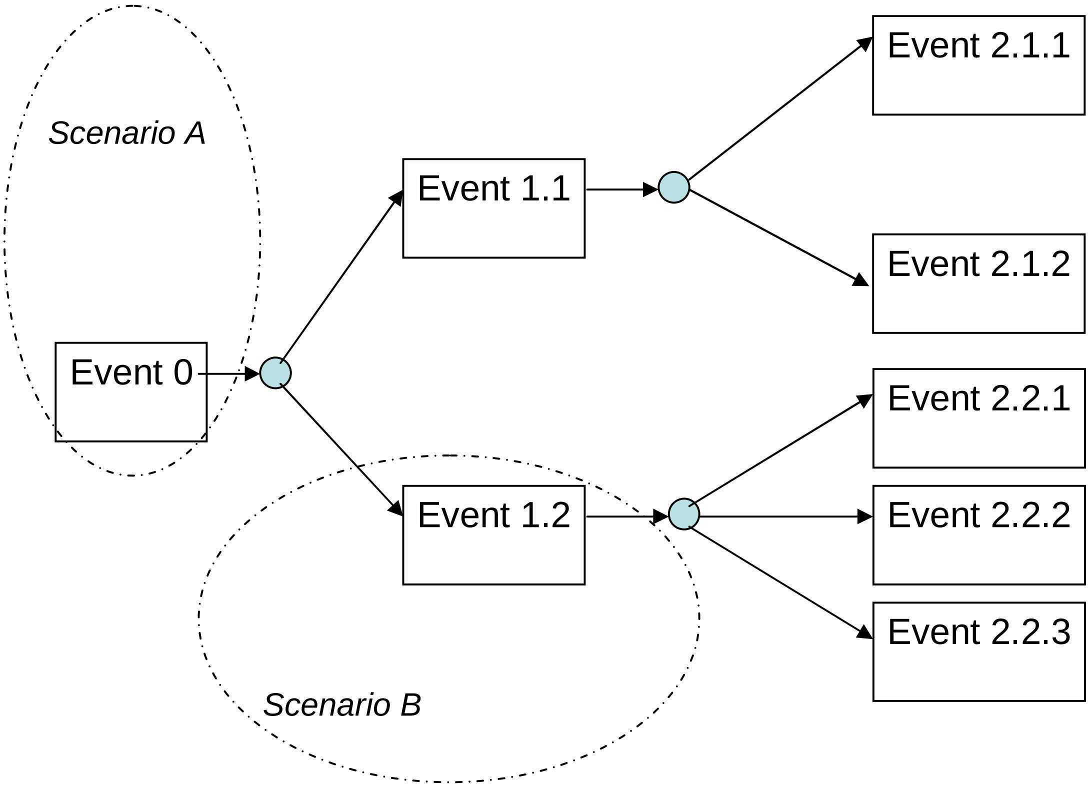


Figure 5: Scenario and Players' decision tree

After that, the administrators can inspect whether the decision trees in the learners' minds are identical with the knowledge being transferred to them by examining whether the outputs of the learners' decisions fall in the possible range within the expected future wheel. The performance of learners' decision will also be evaluated either by the game administrator or eLearning system as part of the feedback to the learners. Learners gain knowledge – abstract experience or adjust their understanding from the process of problem solving – concrete experience by reading feedback from the administrators on the effectiveness of their solution and reflecting upon their decision performance. It is believed that the transfer of knowledge could be ensured by immediate and tangible rewards received for successful behavior.

Compared to the experiential learning model, scenario-based learning always begins at the Testing step (step 4 in **Figure 3**) while the experiential learning model could begin at any of the four steps. The abstract knowledge from textbook may be a prerequisite but not necessary involved in the process of scenario-based learning. Instead of testing in real situation, scenarios are used to provide vivid descriptions to simulate the real situation. Learners gain concrete experience by making decisions under the situation described vividly by the scenarios. What's more, the testing in the same situation could be repeated several times in order to consolidate the knowledge learnt. The whole process is shown in **Figure 6.**

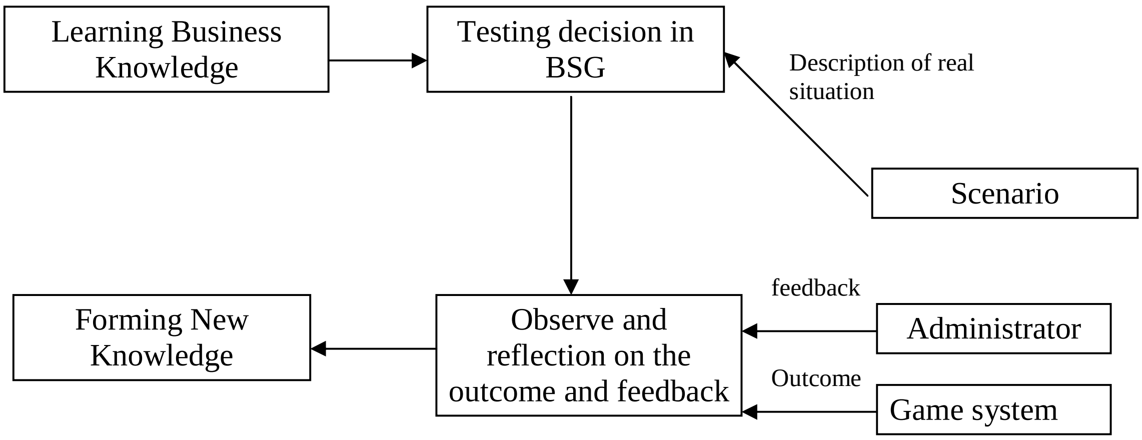


Figure 6: scenario based learning process

## 5.2 Scenario Development in eLearning simulation

### *5.2.1 From the purpose view*

The purpose of using scenario in eLearning simulation is to situate the learner in a virtual social situation to influence their decision making. Scenarios should entail key uncertainties, important predetermined trends, the behavior of the learners, and the behavior of objects of lessons who are part of the current impacts and who may also have a stake in the future consequences in their specification.

### *5.2.2 From the content view*

The content of scenario is the description of the simulation world. In the eLearning simulation environment, there are two levels of world divided according to the accessibility to the learner: macro world and micro world. The macro world is the external environment not supposed to be under the control of the learners.  The micro world is where the learners have certain influence of changes by his chosen course of action. Administrator, he can access to both macro and micro world. Usually, a scenario can relate to both macro and micro worlds. A simulation will take specification input from the environment adjusted by the administrators. It will also take specification from the action input from the learners because the simulated environment also interacts with learners and, as a result, will be changed somewhat. Consequently, from an initial scenario, both the administrators' input and learners' action will contribute to the generation of a new resulting scenario. Thus, in a broad sense, a scenario also covers that parts of the environment that is under the influence of the learners. In this thesis, we study scenarios which are used to administrate the eLearning system and to give learners information. A scenario often incorporates temporal effectiveness, either explicitly or implicitly. The storylines in a scenario may also depict events that are causal-consequential and therefore implicitly and automatically time-tagged. It is also noted that changes of events may take effect gradually or discretely. In summary, a scenario is a partial description of the environment in context with a time qualifier. A scenario can apply to a specific time instance, e.g., the present, or over a time span. A scenario may refer to events that are exogenous, which is about the macro world, or endogenous, which is about the micro world.

The context knowledge, which will be represented in the knowledge ontology of the simulation world is important to the content of the scenario because it is the base of the communication between administrators and learners; the modeling of the knowledge requires reaching agreement upon the terms or vocabularies. The higher the degree of agreement, the easier is the communication.

### *5.2.3 From the form view*

The representation form of scenarios has two fold. Firstly, a scenario is also usually formulated and ultimately expressed in a concrete mode, such as a video, a graph, or narrative free text which could be understand by common people. Secondly, in eLearning simulations, simulation entities, which are objects that interact with each other in the eLearning simulation, and are open to administrators, do not necessarily map directly to the events depicted in the narrated scenarios. Therefore, there is a requirement of modeling the context knowledge of the learning domain, both macro and micro, and translation of the planned scenario into behavior of the simulation entities according to the knowledge model. Thus scenarios should have another representation form which is structured and using vocabularies which are commonly acceptable. The way of structured representation and the vocabularies should be defined in the knowledge ontology discussed in Chapter 6.

### *5.2.4 The testing of scenario in eLearning simulation*

The usage of scenario is actually an interactive process between administrators and learners. As a result, the testing of scenario in eLearning simulation should contain three phases.

Firstly, whether the scenario tool in the system could facilitate the work of administrator should be evaluated. This kind of tools should be able to let the administrators create scenarios correctly and easily as they wish, which means the tools could generate all possible scenarios and the generating process require as few exogenous knowledge as possible.

Secondly, whether the learner can perceive the meanings of scenario should be evaluated. The representation of scenario should be vivid. For example, if the scenario is described using natural language, vocabulary used should be commonly accepted. If jargon is to be used, it would be beyond the knowledge of learner. Their way of expression should have as few interpretations as possible.

Thirdly, the effectiveness of scenario for the learning process should be evaluated. The modeling entities open to administrators, or administrative parameters, combined with the learners' inputs, determines the result of simulation. Administrators would expect the simulation result to fall into specific patterns, one way or another with different learners' inputs. But they would have to make sure that the mapping of the antecedent scenario to effect the simulation is done correctly. In order to test the scenarios, or more precisely, to test if the setting of administrative parameters is done correctly, the learners' inputs would have to be simulated to complete the necessary specification of the simulation entities involved. Thus, testing scenarios can also be an iterative, laborious and time-consuming process.

Besides, the testing of scenario also is the testing of the knowledge ontology. Problems like whether the vocabulary and the uniform way for describing the domain world is enough to cover all possible scenarios should be considered.

# Chapter 6 A knowledge-based approach to a scenario management system

I will propose a knowledge-base approach to scenario management system in the remaining part of this thesis by describing such a system for MAGNUS. The main task is to build a business and economic domain knowledge ontology. The primary purposes of building this ontology are: to enable knowledge reusing and sharing, to help monitor all key information in the expression of scenarios, and to help sharing the scenario between different scenario management systems and the extensions of such systems. The method and criteria for building ontologies discussed in Chapter 3 will be used as reference here. The feature of scenarios in eLearning Simulation discussed in Chapter 5 will be considered here to design a uniform way of representation to facilitate computer processing.

The system can be divided into three parts structured in top-down fashion: the factual part of the ontology, the problem-solving part of the ontology -- the scenario knowledge base and the specific application which utilize the knowledge modeled in the previous two parts.

## 6.1 Factual Part of Business and Economics Domain Knowledge Ontology

### *6.1.1 The purpose of developing the ontology*

I have developed a business and economics domain knowledge ontology for this purpose; it includes a wide variety of carefully defined terms which are widely used for describing business and economics world in general.

The major role of an ontology is to act as a communication medium as shown in **Figure 7**. More particular in our work, we use the ontology as communication medium between:

- Different people: the administrators and the learners in the BSG;
- Different implemented applications (such as different BSGs, different models of a BSG, and the scenario generation tool and the BSG database etc.)

Also, and very importantly, the ontology is intended to assist:

- Acquisition, representation, and manipulation of business and economics knowledge; such assistance is via the provision of a consistent core of basic concepts and the uniform way to describe the concepts.
- Structuring and organizing knowledge base.
- The explanation of the rationale, inputs and outputs of future scenario creation tools.

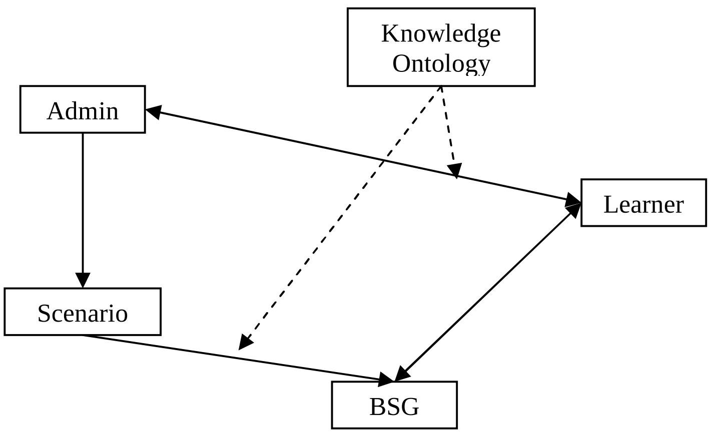

Figure 7: Using Scenario in BSG

### *6.1.2 Meta Layer*

In our BSG, the business and economics domain knowledge ontology also has a layered structure. The first layer is Meta layer, which is the most abstract conceptualization of the knowledge. We developed our Meta layer according to the Metamodel in UML (as introduced in Chapter 3). UML is widely used and accepted. It is very powerful when being used to model not only the entities but also other things like states and states transition etc. which are very useful in the world of business and economics. The concepts used in the Meta layer are:

- *Entities* are the general designation to all the objects which exist in the business and economics world which provides the representation of attributes and the actions an entity is capable to perform.
- A*ttributes* are properties of the entities. Attributes are used to portray the entities. An attribute exists if and only if there is at least one entity which has this attribute.
- *Actions* are operations entities can perform. An action exists if and only if there is at least one entity which can perform this action.
- *States* are the description of the conditions of entities, their attributes and actions, at a given time point. A state can transfer to another state.
- *Events* are the occurrence of given situations, such as the implementation of entities' action and the changes to the value of entities' attribute, in the business and economics world. Events are the trigger of the transferring from one state to another state.

- *Relations* mean the connections between *entities*, *events*, and *states*. In our knowledge ontology, we concerns about relations of *association, generalization, dependency,* and *transition.* Their meanings are as defined in UML.
- *Time* is used in our ontology but we do not plan to re-define it, rather, we will import the terms and definitions from KRSL (Lehrer, 1993) which in turn was based on Allen's work (Allen, 1984).

  *Time Line*: an ordered, continuous, infinite sequence of Time Points.

  *Time Point*: a particular, instantaneous point in time;

  *Time Interval*: an interval of time specified as two Time Points and bounds on the distance between the two time points.

#### *6.1.3 Semantic Layer*

The second layer is the semantic layer which defines the abstract concepts in Business and Economics knowledge, which are actually the types of *entities*, their *attributes* and *actions*, and type of *relations*. These concepts in the ontology are developed after the study of basic economics knowledge from economics textbook and the aforementioned business simulation games, such as BusSim, Total Enterprise Simulation, Capitalism II and MAGNUS, in order to gain consensus. The concepts in this layer are the reification of the Meta layer. These concepts' reification are in the lower layer such as instance layer. Examples of the types of entities identified are *government, enterprise, people, resources, products, market, consumers* and *economic environment*. Entities will interact with each other either implicitly or explicitly which are depicted by relations. The main types of relations take effect in this layer are dependency and association. The entity types and relations defined in this layer are listed as follow:

**Relations:**

Component-of: a kind of *dependency* relation. E.g. A is a component-of B means A is a part of B.

Used-by: a kind of *dependency* relation. E.g. A is used by B means B called the usage of A when perform a certain action.

Partnership: a kind of *association* relation. A group of entities carry on business in common.

Ownership: a kind of *association* relation. E.g. A has the ownership of B means A has the property or right of use of B.

Effect-of: a kind of *dependency* relation. It means the causal-consequence or trend relation between two *events*.

**People:** natural person**,** the concept of people will not appear alone in our ontology. It'll work as individual component of a union. The attributes and actions of people will be described when we define a union, such as consumers.

**Resources**: entities used by other entities' actions. A resource may be shared by more than one other entity.

*Attributes of a resource*

- Price: the monetary value must be paid to obtain the ownership or right of use of the resources, to use the resources and to get rid of the resources.

- Current Value: the current monetary value of the resources.
- Amount: quantifiable measures denoting how much is available for use.
- Productivity: the ability to produce products when being used.
- Space taken up: if not space resources, the space resources needed to store the resources; if space resources, the amount of space unit it takes up.
- Owner: the entity who has the property or right of use of this resource. This attribute builds up ownership relation between the resource and its owner entity.

*Actions of resource*: influence the productivity and profit of enterprises.

**Enterprise**

*Attributes of an enterprise*

- Productivity: the ability to produce products.
- Fixed assets: quantifiable measures denoting how much fixed assets the enterprise has
- Liabilities: quantifiable measures denoting how much liabilities the enterprise has
- Owner Equity: quantifiable measures denoting how much owner equity the enterprise has
- Industry: what kind of industry the enterprises belong to.
- Confidence: the willingness to function properly.
- Nationality: local / foreign.
- Profit: quantifiable measure denoting how much the enterprise earns.

*Actions of an enterprise*:

- Manage: the arrangement of various actions of the *enterprise*.
- Exchange: using *resources* or *products* to trade *resources*.

- Produce: input resources, output products. This action builds up used-by relation between the input resources and the enterprise.

**Enterprise Unit**

*Attributes of an enterprise unit*

- Owner: the enterprise this enterprise unit belongs to. This attribute builds up component-of relation between the unit and the enterprise.

*Actions of enterprise unit*

- Operation: using *resources* to change the attributes of *enterprise* or perform *enterprise's* actions.

**Government**: it can adjust the attributes of resources and establish rules for the competition between enterprises.

*Attributes of a government:*

- Political situation: qualifiable measures denoting the stableness of the government.
- Support: qualifiable measures denoting the support from people

*Actions of government*:

- Taxation: levy a tax, in forms of certain *resources*, upon specific products.
- Subsidy: give subsidy, in form of certain *resources*, to specific products.
- Legislation and regulation:
- Government spending: using *resources* to trade other *resources.*
- Political actions: like war, coup.

**Government Unit**

*Attributes of a government unit*

- Owner: the *government* this *government unit* belongs to. This attribute builds up component-of relation between the unit and the *government*.

*Actions of government unit*

- Operation: perform actions of *government*.

**Consumer**: Consumers could be a union of people, enterprises or the government.

*Attributes of a consumer:*

- Age: quantifiable measures denoting the age of the people belong to this union.
- Gender: male or female
- Occupation: types of job the consumers have.
- Income level: quantifiable measure denoting the how much consumers can earn over a period of time.
- Purchasing power: quantifiable measure denoting the consumers' ability to buy.
- Confidence: quantifiable or qualifiable measure denoting the consumer's willingness to buy.
- Actual status: Boolean measure. If the value is true, the consumer agrees to buy a product; otherwise he may buy but hasn't bought yet.

*Actions of consumer:*

- Buy: using *resources* to trade for products

**Market:**

*Attributes of a Market:*

- Product: the *product* this market needs. This attribute builds up an association relation between the market and the product.
- Consumers: the union of consumers the market has. This attribute builds up an association relation between the market and the consumers.
- Need: a physical, psychological or sociological requirement of the product needed by this market.
- Super Market: a larger market in one of whose segmentation this market locates.

**Market segment:** this concept is actually an instance of the market. Market segment has component-of relation with its Super Market indicated by attribute "Super Market".

**Product**: the output of enterprise's produce action.

*Attributes of a product*

- Producer: the enterprise that produced this product.
- Materials: quantifiable measure denoting how much a set of *resources* is used to produce this product.
- Cost: the monetary value of all resources involved in the building of the products, transporting cost.
- Price: the monetary value others must pay for obtaining the products.
- Status: finished/work-in-process
- Life cycle: new, old etc.
- Nationality: imported / domestic

- Amount: quantifiable measure denoting how much are there the product.
- Brand: a name identifiable by consumers associated with one or more products of a enterprise.
- Image: what consumers believe to be true of a brand?
- Features: a set of characteristics which may satisfy the need of the *market*.

**Economic environment**

*Attributes of an economic environment*

- Economic Condition: overview description of a country's economy situation, good / bad etc.
- Economic Index: various index indicating the condition of a certain aspect of economy environment.
- Range: the range of economy environment, local, regional, or world wide.

### *6.1.4 Instance Layer*

The third layer is the instance level, the reification of the concepts defined in the second layer. There are too many instance of the concepts defined in the semantic layer. Thus I only illustrate.

**Reification of *resource***

Space resource:

Cost: 1) The price per area unit or the rental per area unit per time unit.

2) The fee used to maintain the ordinary function of the land.

Amount: size of land measured by area unit.

Labor resource:

Cost: hiring cost, firing cost, orientation cost, training cost, professional cost, salary.

Amount: number of people

Productivity:

Current value:

Different types of labor resources: worker, manager.

Technology resource:

Cost: R&D fee.

Money resources

Cost: rates

Current value: relative value compared to other currencies. It is measured by exchange rate.

Different types of money resources: money from t-bill, money from short-term-loan, money from long-term-loan, money from stock.

Information resources: the value of other entities properties.

Cost: the price of information.

Material resource

Cost: price, transporting cost.

Amount

Equipment resource

Cost: price or rental, maintain cost.

Current value

Amount: number of unit.

***Reification of enterprises:***

Production: the process to use raw materials to produce product. Input: labor, money, material. Output: products.

Recruiting: to obtain more labor resources. Input: money, current labor in HR department. Output: new labor forces.

Education and training: to raise labor's productivity. The input resources include the current labor forces in HR department. Output: higher productivity of labor.

Research and develop: to discover new technology to improve productivity. The input resources include money, labor. Output: new technology resources.

Investment: to gain profit. Input: money, output money resources, capital resources.

Building: to construct new factory or office. Input: money, labor. Output: space resources.

Equipment purchasing: buying new equipment. Input: money. Output: equipment resources and improved productivity.

Process re-engineering: to improve the product efficiency by adjust the system structure of the enterprise. Input: labor, money. Output: higher productivity

Marketing: to research the market to obtain demand and supply information.  Input: labor, money. Output: information of consumer needs

Advertising: increase consumers' needs of a certain products. Input: money, labor. Output: increase the consumer's needs.

Trading: sign contract with other enterprises or government. Input: labor, information. Output: the function of production.

Retrenchments: reduce the current labor force. Input: HR department labor force, money resources. Output: decrease the current labor force.

Selling: sell products to consumers. Input: products, labor. Output: more money resources.

We intended that the core terms would remain substantially unchanged. However, the business and economics domain knowledge ontology as a whole is expected to evolve during the use of the scenario management and the development of the BSG, and will be refined and extended as required.

### 6.2 Scenario knowledge base

The scenario knowledge base consists of two parts: an event dictionary and a relation dictionary from which terms are extracted to form a scenario. The event dictionary stores possible events which may occur in the business and economic environment. These events will be ready for retrieval by game administrators to construct scenarios. Relation dictionary stores effect of relations between events for the use in reasoning about event happenings. Events dictionary is the set of circles in **Figure 4** while relation dictionary are the set of lines connecting circles.

The relation dictionary is dependent on the event dictionary, which is dependent on the ontology. This structure is also based on the consideration of dividing stable knowledge and relatively unstable knowledge.

## 6.3 The Scenario Management System

### *6.3.1 Definition of Scenario in BSG*

Before giving the definition of scenarios in BSG, we introduce several concepts. These concepts are derived from the analysis of the definition of scenarios in different disciplines presented in the section 2:

- Event: as the definition in the Knowledge Ontology.
- Life: this term is used to refer to the duration of an event's effectiveness which is the time length between the time point an event begins to take effect and the time point an event ends to take effect.
- Uncertainty: In the context of business and economics, descriptions are usually assumptions and hypotheses about processes and actions, i.e., the events. Most of the descriptions express conditions of uncertainty and ambiguity, as claimed (Wack, 1985) by an economist "*The future is no longer stable; it has become a moving target*". Thus qualitative or quantitative factors should be used to measure the uncertainty. This concept will also be used in the reasoning of implied scenario which will be mentioned in later.

Under the aforementioned configuration of knowledge context and the concept defined above, we define a scenario of BSG as the specification of a set of events and the state as defined in section 6.1.2. Sometimes, a scenario could involve only one event. These events begin to take effect concurrently at the time point scenarios are used in the business world simulated. The event will trigger the transition of states. There are two kinds of scenario: direct scenario and implied scenario. A direct scenario is only the specification of stable states which will not transit to another state. If the scenario has specification about events, a new scenario contains the specification of new state may be created implicitly, which is called implied scenario. In this system, it is required that after all the reasoning process, there should be a final product of direct scenario. If not, there should be something wrong in the knowledge base.

### *6.3.2 The representation of the ontology and Scenarios*

The design of the ontology uses XML and XML Schema language to create tags to represent entities, events, scenarios and other concepts, which can support the scalability to ease the exploration of new scenario and new game knowledge. Part of the tags and their meaning are listed in **Table 1**:

| | |
|---|---|
| <Scenario>…</Scenario> | A scenario |
| <Event>…</Event> | An *event*. Event has an attribute id which is used for the indexing of event. |
| <Entity>…</Entity> | An *entity* |
| <Attribute>…</Attribute> | An attribute of an *entity* |

| | |
|---|---|
| <Action>…</Action> | An *action* |
| <Relation>…</Relation> | An interactive relation between events |
| <Uncertainty>…</Uncertainty> | The uncertainty factor in the description of an event and relation |
| <Life>…</Life> | The duration of an event or relation's effectiveness |
| <Predication>…</Predication> | The depiction of attributes |

Table 1: Tags and their meaning

```
<Event id>
        <Entity>
                <Attribute/> (<Action/>)
        </Entity>
        <Predication/> (optional)
        <Uncertainty/> (optional)
        <Life/> (optional)
</Event>
```

Figure 8: Event Representation

```
<Scenario>
        <Event/>
        <Event/>
        ……
        <Event/>
</Scenario>
```

Figure 9: Scenario Representation

As shown in **Figure 8**, each event is assigned an attribute ID for the use of indexing. Entities here are described by their attributes' value. Predication is the description of the attributes' value. Items "Uncertainty" and "Life" are optional.

A scenario is a set of events, as shown in **Figure 9**.

In the relation dictionary, rules for the reasoning of implied scenario are listed. The rules are represented as in **Figure 10**.

```
<Relation causeID=** resultID=**>
        <Uncertainty/> (optional)
        <Life/> (optional)
</Relation>
```

Figure 10: Relation Representation

Each relation represents for an implied relationship of an antecedent event and its resulted event. Tags “Uncertainty” and “Life” have the same meaning as the ones in event representation.

The values of attributes “causeID” and “resultID” refer to the value of attribute “ID” in an event included in a scenario representation. The attribute “causeID” referring to is an event in antecedent event and the one “resultID” referring to the resulted event.

### *6.3.3 The use of the system*

The game administrators can input more detailed description of possible events to compose a scenario in this system. These descriptions of events are at first mapped to the events dictionary. Matched events will then be used to find implied scenarios by querying the relation dictionary.

> *The economy is still sluggish. The political situation is rather unstable. The government is fast losing support due to its unpopular policies.*

This is a scenario described in natural language, there are three events describing economy, political situation and government respectively. These free form expressions are then transformed into structured representation form in order to facilitate machine process. The representation of this scenario is like in **Figure 11**:

```
<Scenario>
    <Event>
        <Entity>
            <EconomicEnvironment>
                <Attribute><EconomicCondition/></Attribute>
            </Economic Environment>
        </Entity>
        <Predication>still sluggish</Predication>
    </Event>
    <Event>
        <Entity>
            <Government>
                <Attribute><PoliticalSituation/></Attribute>
            </Government>
        </Entity>
        <Predication>rather unstable</ Predication>
    </Event>
    <Event>
        <Entity>
            <Government>
                <Attribute><Support/></Attribute>
            </Government>
        </Entity>
        <Predication>fast losing</Predication>
    </Event>
</Scenario>
```

Figure 11: Scenario Representation Example

The “Sluggish” is the description of the economic condition which is an attribute of entity *economy environment*. . The “political situation” and “support” are all attributes of entity *government.*

In the events dictionary, these three events will be given an id for the use of indexing:

<Event id=”1”>…<EconomicEnvironment>...<EconomicCondition/>…

<Event id=”5”>…<Government>…<PoliticalSituation/>…

<Event id=”10”>…<Government>…<Support/>…

In the relation dictionary, relations involving these events will use id as reference to a certain event. For example, for the relation “sluggish economic condition may block the fund investment from foreign countries which may take a few years time to recover.” The representation in the relation dictionary is as in **Figure 12**:

```
<Relation causeID=”1” resultID=”4”>
        <Uncertainty>highly possible</Uncertainty>
        <Life>years</Life>
</Relation>
```

Figure 12: Relation in the relation dictionary

Thus, event with id 4 is the consequence of event with id 1:

```
<Event id=”4”>
        <Entity>
                <Resource type=”Investment”>
                        <Attribute><Amount/></Attribute>
                </Resource>
        </Entity>
        <Predication>less</Predication>
</Event>
```

Figure 13: Event 4 in the events dictionary

In the tag “<Resource type=“Investment”>”, “type” is an attribute of entity *resource* which is used to identify what kind of resource it is. At last, the application will modify the parameters in the database of MAGNUS according to the events involved, which are either directly included in the scenario or implied by the relation.

I have developed a tool with Graphic User Interface to let the administrators choose from several options to create scenarios. This tool can also detect implicated event according to the relation dictionary as shown in **Figure 14 (a), 14(b) and 14 (c)**.

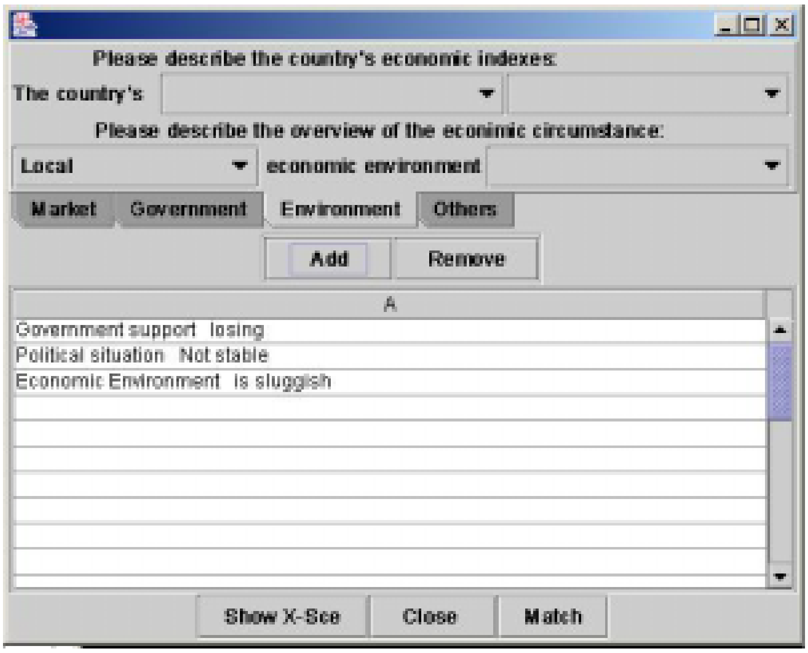


Figure 14 (a): The GUI interface


Figure 14 (b): a Generated Scenario

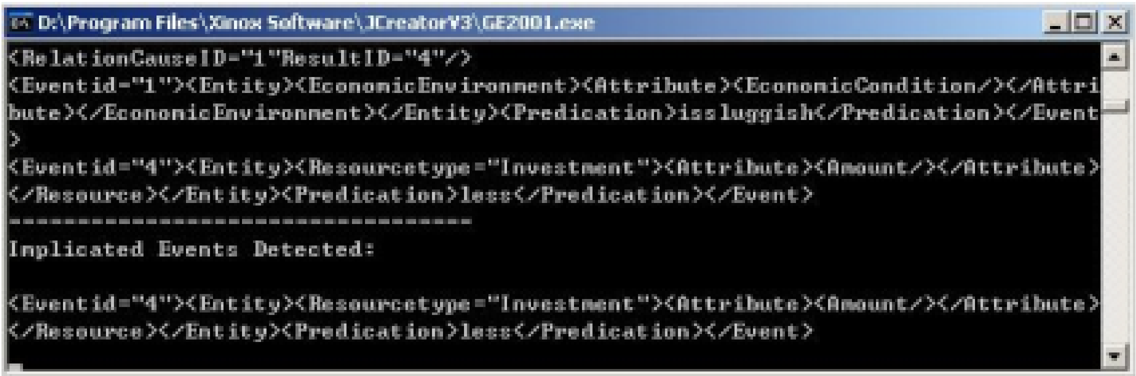


Figure 14 (c): Detecting Implicated events

The business and economics domain knowledge ontology is the base of the whole scenario management system. The event dictionary is built upon the ontology and provides the base of the relation directory. This infrastructure can greatly ease the use of a scenario management system because the more stable a fact is, the less modification to it is needed. Thus, users do not always need to concern about the upper layers' issues; they can mainly focus on the bottom levels such as instance layer or layers below instance layer. For example, when the users want to add new events, they do not need to modify the semantic layer entities. The only exception occurs when the events being added contains entities or attributes the upper layers have not defined but this situation is rare because of the relatively more stable nature of facts in the upper layers.

### *6.3.4 The support from the business and economics domain knowledge ontology*

This three-tiered infrastructure provides the base of the scenario management system. To generate scenarios, a game knowledge-base specific to the eLearning simulation will also have to be built. In (Lua &Yeo, 2003), they describe some possible ways of knowledge modeling of such a game knowledge-base. In my work, a specific way of modeling the knowledge to support the game knowledge-base is discussed. With the domain knowledge ontology, a set of items are provided for the creation of scenarios and rules in scenario knowledge base which could be the support of the communication between administrators and the scenario management system as well as between the scenario management system and simulation system's parameter list.

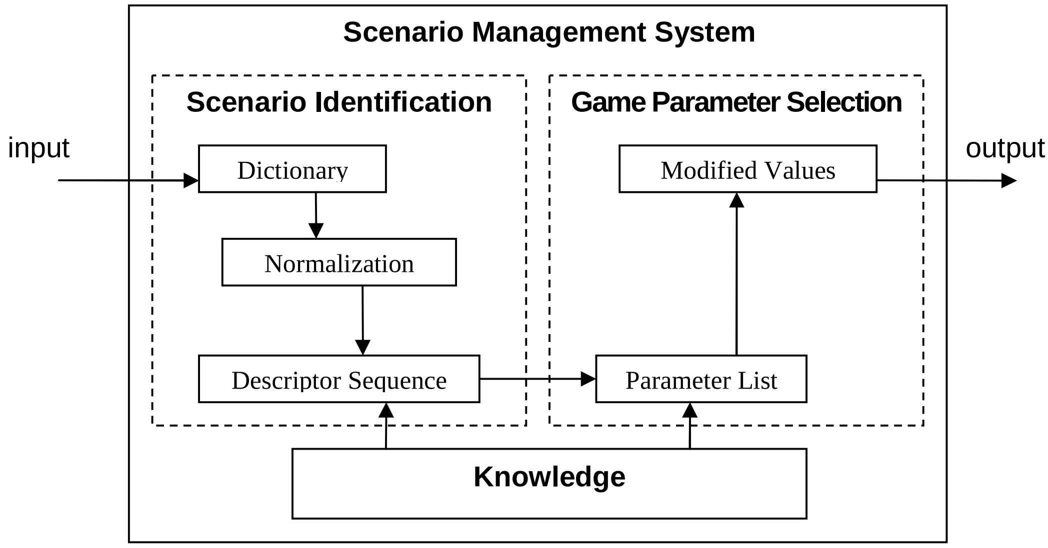


Figure 1

With the support from the domain knowledge ontology, I propose a new scenario management system, compared to the one presented by Lua and Yeo (2003) (**Figure 1**), as shown in **Figure 15**. For the scenario identification part, a dictionary could be organized according to the factual part of the ontology or more specifically the instance layer. This new dictionary groups the words according to the concepts defined in the ontology and the relationship between them. For example, "*political situation*" will be linked with "*government*" (please refer to section 6.1.4, page 47) and "*economic condition*" will be linked with "*economic environment*" (please refer to section 6.1.4, page 50). On the other hand, besides applying normalization, synonyms under the same concept defined in the ontology would be replaced by the terms used in the ontology.

With such key words identified from the free text scenarios and replaced by concept terms in the ontology with semantic relations, the next step is to generate scenarios in a structured form to facilitate computer processing. The ontology has defined structured representation form of events (examples could be found in the section 6.3.3) to support this step.

As I have defined in section 6.3.1, a scenario is the specification of a set of events. Administrators would describe the same event in various ways. If these descriptions are all put in the scenario knowledge base, it will be very large which are not suitable for computer processing because many inferences would have to be used and the search within the scenario knowledge base would be costly. As a result, for the ease of computer processing, standard items and uniform structured representation form is needed. For example, there are many ways to say “the investment will be less”, but all these ways will be translated into a structured form of events as shown in **Figure 16**, and then these structured events will be filled into the semantic structure of scenario defined in the knowledge base as shown in **Figure 9** to form the structured representation forms of scenarios.

By providing the items and the structure representation form of events, the ontology helps the scenario management system to generate structured representation form of scenarios, which supports the communication between administrators and the scenario management system. In the field of natural language processing, similar works of domain knowledge ontology like WordNet (Fellbaum, 1998) have been done to improve processing.

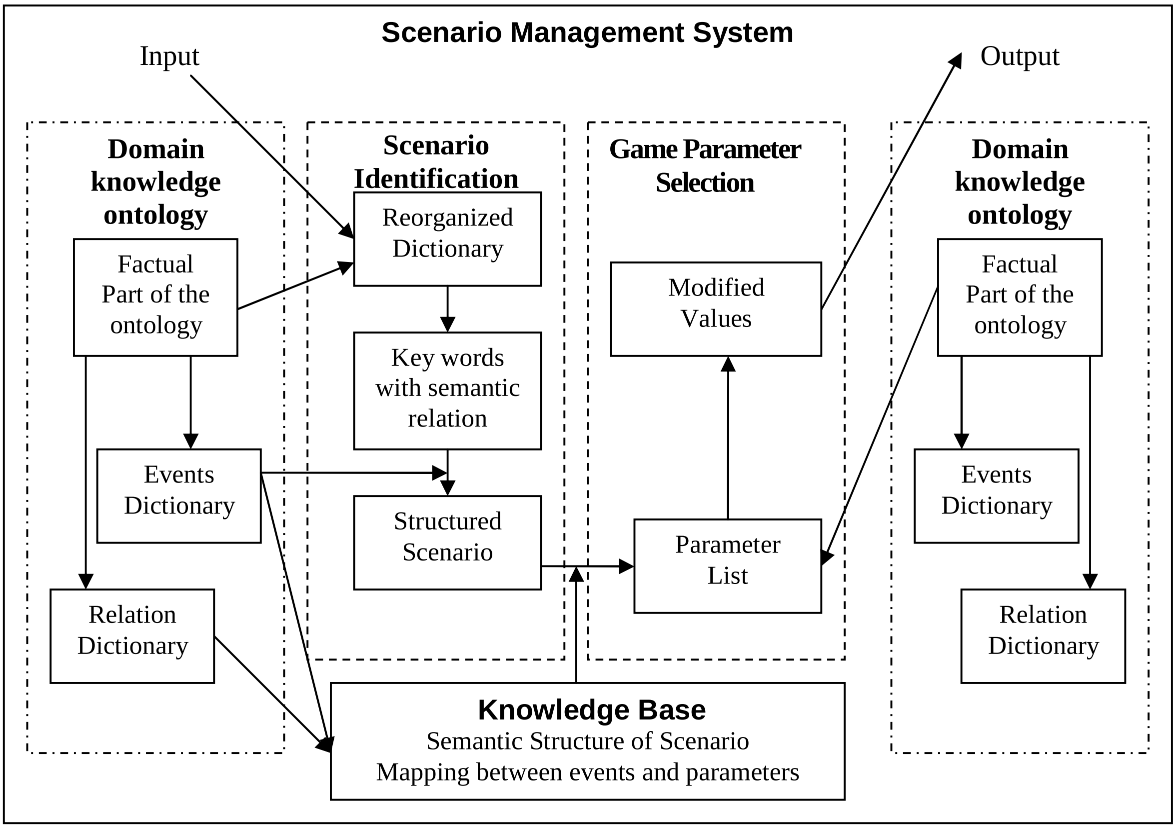


Figure 15: the New Scenario Management System

The relation dictionary and the events dictionary of the ontology will do the reasoning to find any implicated scenario. The knowledge base of the scenario management system now serves to map the events to parameters. This is in contrast to Lua's (2003) work where the mapping is between one whole scenario and parameters. Once an event is recognized, the values of a set of parameters are changed. The administrators could modify the mapping between events and parameters without touching the ontology which could save a lot effort and provide more flexibility.

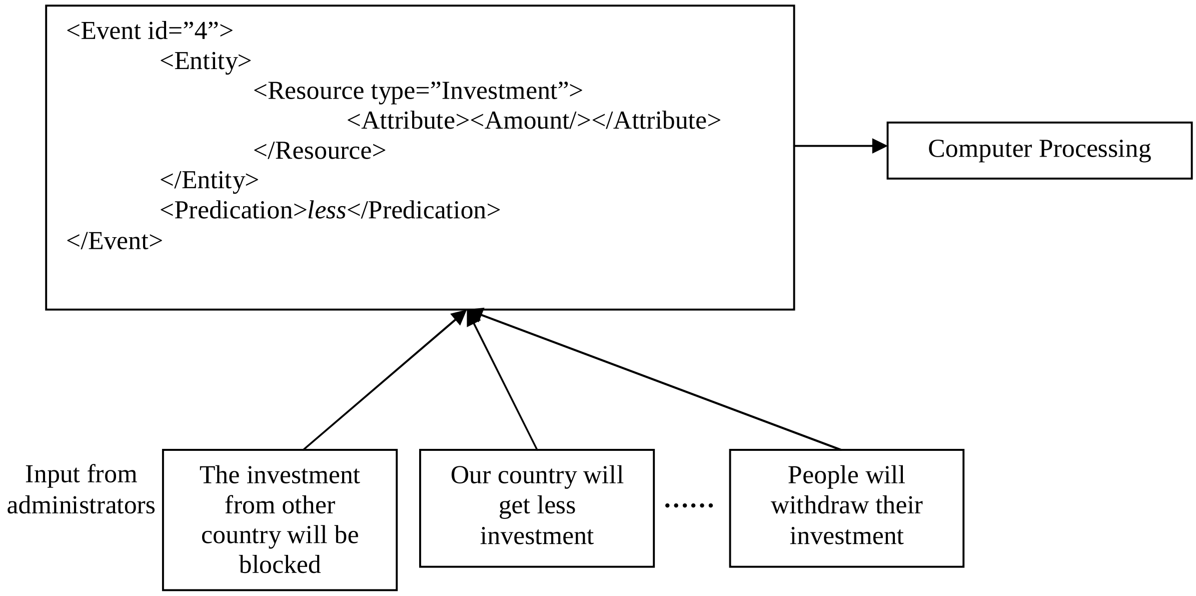


Figure 16: To facilitate computer processing

The domain knowledge ontology could also support the developing and organizing of BSG's parameters. For example, in the domain knowledge ontology, the entity *Labor resource* has some attributes like hiring cost, firing cost, productivity. With this support, in the BSG, the parameters like unit-hiring-cost, unit-firing-cost, and unit-labor-per-unit-product are developed and linked to labor parameters. Because the developing of parameters is supported by the domain knowledge ontology, the mapping between the knowledge base and parameters would have less difficulty. In other words, the communication between the game parameters and scenario management system is supported again by the ontology.

# Chapter 7 Conclusion and future works

### 7.1 Summary

This thesis discusses my work in the design of a knowledge-base framework to support scenario development in eLearning simulations. At first, existing works about scenario management in eLearning Simulation are studied. Deficiencies of these systems are discussed. The use of knowledge ontology is proposed to help overcome these problems. Related works about scenarios in fields like HCI, RE and Strategic Management are studied to understand what a scenario is and what the benefits of using scenarios are. I explain the scenario-based learning process according to the empirical learning model. Scenarios generated from the simulations would have to be true representation of the scenarios facilitators or administrators devised to achieve certain learning objective. Each facilitator has in his mind a future wheel (**Figure 4**) showing the possible events that may occur and the possible routes of transition from present states to future states. Scenarios are used here to describe a part of the future wheel to test learners understanding of the relevant concepts. Thus, from the concrete experience of solving problem in simulated real situation and from their own reflection on the feedback from the facilitators about their performance in the simulation, the learners could learn valuable knowledge.

Based on the discussion of Scenario in other fields and the scenario-based learning process, features of scenarios in the context of eLearning simulations are analyzed and they are taken into consideration when developing the ontology.

I also discuss domain knowledge ontology and tried to construct the business and economics knowledge ontology for business simulation. The purposes of using ontology are at first to define the concepts which will be used in the simulation system in order to reach an agreement and also to find a uniform way to represent the knowledge. I developed my system based on the Metamodel in UML. The reason I use UML is that it is widely used and accepted by many people. UML can model not only the attributes and behavior of entities and can also model the states and transition between states.

At last, I propose a knowledge-base approach to scenario management system in the remaining part of this thesis by describing such a system for MAGNUS. The infrastructure of the system can be divided into three parts from top down: the domain knowledge ontology, the scenario knowledge base and the specific application which utilize the knowledge modeled in the previous two parts. The knowledge ontology is the base of the whole scenario management system. The knowledge base consists of a relation dictionary and event dictionary. The event dictionary is built on the knowledge ontology and provides the base of relation directory. For the consideration of scalability and ease of use, we use XML to write both the knowledge base and knowledge ontology because XML is characterized by its strong extendibility.

The contributions of this work are as follow. First, I presented a systematical analysis of the use of scenario in HCI, RE and Strategy Management. Although the description of scenario is indispensable of context knowledge, few existing works on scenarios discuss context knowledge modeling. Second, I propose in my work the scenario-based learning model. Third, I analyzed the features and developin methods of the domain knowledge ontology and propose a new scenario management system with support from domain knowledge ontology.

Based on the analysis of the related work on scenarios, we identified some common elements in the statement of scenario. I thus proposed a structure way of scenario representation while, in most other works, scenarios are represented in free forms like descriptions in natural language. Such way of scenario representation could greatly help computers find key information.

### 7.2 Future Work

Future work related to this these could be done in two aspects: the extension of the use of the domain knowledge ontology and improvement to the present work.

One extended use of the domain knowledge ontology to is to provide support to the other parties involved in the eLearning simulation, namely, the students and the simulation developer. For the students, the domain knowledge ontology could add learning support to them in the similar way as in many of the discussions in incorporating ontology in education design, e.g., (Mizoguchi, 1999). For example, the taxonomy of the domain knowledge ontology could be developed according to the learning process: one layer of the domain knowledge could be the knowledge that has learnt by the student while another layer could be the knowledge that is to be learnt by the students. Another extended use of the domain knowledge ontology is to help simulation developer in further improving the models used in the simulation (Yeo, 1991). For example, in a BSG, according to the original simulation model, potential sale of a company is influenced by the price, marketing effort spent, quality of her product and her reputation. The domain knowledge ontology would be able to tell the game developer additional factors that may be considered, e.g., seasonal fluctuation.

One possible improvement to the current work could be the completing and classification of the scenario knowledge base. As many events, which may occur in the simulation world, should be found in order to cover all possible input of scenario. The same work should be done to the relations. In my work, the events and relations are listed in the scenario knowledge base with no structure. Classification of the relations and events according to a certain criteria is also valuable for the indexing and searching work. For example, events related to "*economic environment*" could be put together. Further more, events which will bring positive influence to "*economic environment*" could also be put together. So do events with negative influence. A hierarchical structure could also be introduced to the organizing of the scenario knowledge base.

# Reference List

Allen, J.F. (1984), Towards a general theory of action and time. Artifcial Intelligence, 23, pages 123–154

Anderson, J.S. & Durney, B. (1993), Using Scenarios in Deficiency-driven Requirements Engineering, Proceedings of IEEE International Symposium on Requirements Engineering, San Diego, CA, USA, pages 134-141, New York, NY, USA, ACM Press.

Arpirez, J., Gomez-Perez, D., Lozano, A., and Pinto, S. (1998), An Ontology-Based WWW Broker to Select Ontologies. Proceedings of the Workshop on Applications of Ontologies and Problem-Solving Methods, ECAI, pages 16-24, Aug.

Asuncion, G-P. and Benjamins, V.R. (1999), Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods, in Proceedings of the LJCAI-99 workshop on Ontologies and Problem-Solving Methods(KRR5), Stockholm, Sweden, Aug. 2.

Bartusik, K., and Cabala, P. (1997), Metoda scenariuszy w planowaniu strategicznym. Przeglad Organizacji No. 2, pages 20-25

Bernaras, A., Laresgoiti, I., and Corera, j. (1996), Building and Reusing Ontologies for Electrical Network Applications. Proceedings of the 12th ECAI, pages 298-302.

Blanning, R.W. (1995), a Decision Support Framework for Scenario Management, Proceedings of the International Symposium on Decision Support Systems, Vol. 2, Hong Kong, pages 657-660

Borgo, S., Guarino, N., and Masolo, C. (1996), Stratified Ontologies: the Case of Physical Objects. Proceedings of the Workshop on Ontological Engineering, held in conjunction with ECAI 1996, Aug. pages 5-15.

Brown, S. (1968), Scenarios in System Analysis. In: Quade ES, System Analysis and Policy Planning: Applications in defense. Elsevier, New York, pages 298-309

Bui, X.T., Kersten, G. and Ma, P-C, (1996), Supporting Negotiation with Scenario Management. Proceedings of the 29th Hawaii International Conference on System Sciences, Vol 3, pages 209-219.

Carroll, J.M. and Rosson, M.B. (1992), Getting around the task-artifact cycle: how to make claims and design by scenario, ACM Transactions on Information Systems, Vol 10, Issue 2, pages 181-212.

Carroll, J.M. (1999), Five Reasons for Scenario-Based Design, Proceedings of the 32nd Hawaii International Conference on System Sciences, IEEE CS Press.

Chandrasekaran, B., Josephson, J.R., and Smith, J.W. (1999), Ontologies: What are they? Why do we need them? IEEE Intelligent Systems and Their Applications, 14(1): pages 20-26, 1999. Special Issue on Ontologies.

Chandrasekaran, B., and Johnson, T.R. (1993), Generic Tasks and Task Structures: History, Critique and New Directions, in Second Generation Expert Systems, David J.M., Krivine J.P., and Simmons R. (Eds.), Springer-Verlag, London, pages 233-272.

Clancey, W.J. (1992), Model Construction Operators, *Artificial Intelligence*, 53(1): pages 1-115.

Fellbaum, C. (1998), WordNet – *An electronic lexical database*, MIT Press, Cambridge, M.A.

Fraser, J., Tate, A., and Uschold, M. (1995), The Enterprise Toolset – an Open Enterprise Architecture. The Impact of Ontologies on Reuse, Interoperability and Distributed Processing, pages 42-50. Unicom Seminars, London, 1995.

Gangemi A., Pisanelli D., and Steve G. (1998), Ontology Integration: Experiences with Medical Terminologies, Formal Ontology in Information Systems, N. Guarino(ed.), IOS Press.

Gausemeier, J., Fink, A., and Schlake, O. (1995), Szenario-Management, Planen und Führen mit Szenarien. Carl Hanser Verlag, München, Germany

Genesereth, M.R. and Nilsson, N.J. (1987), *Logical Foundations of Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann.

Glenn, J.C. (1994), The Futures Wheel, United Nations University, Washington D.C., USA

Gruber, R.T. (1995), Towards Principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies, 43, pages 907-928

Gruninger, M. and Fox, M.S. (1995a), Methodology for the Design and Evaluation of Ontologies. Proceedings of International Conference on AI Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.

Gruninger, M. and Fox, M.S. (1995b), The logic of enterprise modeling. In J. Brown and D. O'Sullivan, editors, Reengineering the Enterprise, pages 83-98, Chapman and Hall, 1995.

Guarino, N. and Giaretta, P. (1995), Ontologies and Knowledge Bases: Towards a Terminological Clarification, Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, N. Mars(ed.), pages 25-32, IOS Press, Amsterdam.

Hansmann, K.W. (1983), Kurzlehrbuch Prognosenverfahren. Mit Aufgaben und Lösungen. Wiesbaden, 1983, Germany

Haviland, S.E. and Clark, H.H. (1974), What's New? Acquiring New Information as a Process in Comprehension. Verbal Learning and Verbal Behavior, 1974 (13), pages 512-521.

Hsia, P., Samuel, J., Gao, J., & Kung. D. (1994), Formal Approach to Scenario Analysis. *IEEE Software*, Vol. 11, Issue 2, March 1994, pages 33-41.

Jarke, M., Bui, X. T. & Carroll, J.M. (1998), Scenario Management: An Interdisciplinary Approach. *Requirements Engineering Journal*, Vol. 3, No. 3-4.

Jarrar M and Meersman R, (2002), Scalability and Knowledge Reusability in Ontology Modeling, In Proceedings of the International conference on Infrastructure for e-Business, e-Education, e-Science, and e-Medicine, SSGRR2002s.

Kahneman, D. and Tversky, A. (1972), Subjective Probability: a Judgement of Representativeness. Cogn Psychol, 1972(3), pages 430-454.

Kindley, R. W. (2002), The Power of Simulation-based e-Learning, *The eLearning Developers' Journal, September* 17, 2002.

Klein, G.A., Orasanu, J., Calderwook, R. (1993), *Decision Making in action: models and methods*. Ablex, Norwood, NJ.

Kolb, D. A. (1984). Experiential learning: Experience as the source of learning and development. New Jersey: Prentice-Hall.

Kolb. D. A. and Fry, R. (1975), Toward an applied theory of experiential learning, in C. Cooper (ed.) *Theories of Group Process*, London: John Wiley.

Kolodner, J. L. (1993), *Case-BasedReasoning*, Morgan Kaufmann, San Mateo.

Lehrer, N. (1993), Knowledge representation speci.cation language. Technical report, DARPA/Rome, Laboratory Planning and Scheduling Initiative, Reference Manual.

Lua, T.M. and Yeo, G.K. (2003), Scenario in Simulation Games, Proceedings of ISAGA, Kazusa Japan, pages 753-760.

Medin, D.L. and Schaffer M.M. (1978), A Context Theory of Classification Learning. Psychol Rev. (85), pages 207-238.

Meristö, T. (1991), Skenaariotyöskentely yrityksen johtamisessa. VAPK-Kustannus, Helsinki, Finland.

Mesarovic, M., & Pestel, E. (1974), Mankind at the Turning Point: The Second Report to the Club of Rome. New York: E. P. Dutton & Co., Inc.

Mizoguchi, R. (1999) Ontology-Awareness for Intelligent Instructional Systems, in Advanced Research in Computers and Communications in Education, G.Cumming et al, (Eds), IOS Press, pages 45-49

Mullar, P-A. (1997), Instant UML, Wrox Press Ltd, Birmingham.

Nardi, B.A. (1992), The use of scenarios in design. SIGCHI Bulletin, 1992, 24(4), pages 13-14.

Pesonen, H-L, Ekvall, T., Fleischer, G., Huppes, G., Jahn, C., Klos, Z.S, Rebitzer, G. Sonnemann, G.W., Tintinelli, A., Weidema, B.P. and Wenzel, H. (1998), Framework for Scenario Development in LCA, School of Business and Economics, Jyväskylä, Finland.

Pomerol J-C. (1997), Artificial Intelligence and Human Decision Making, *European Journal of Operation Research*, Vol. 99, Issue 1, pages 3-25.

Potts, C., Takahashi, K., and Anton, A.I. (1994), Inquiry-Based Requirements Analysis, IEEE Software, Vol 11, Issue 2, pages 21-32.

Raiffa, H. (1968), Decision analysis, McGraw-Hill, New York

Raman, S. (1988), Flight Mission Scenario Generation with Knowledge-Based system, ACM.

Ringland, G. (1999), Using Scenarios in ICL to Develop and Implement Strategy, *ICL Systems Journal*, Autumn.

Rolland, C., Ben, A.C., Cauvet, C., Ralyte, J., Sutcliffe, A., Maiden, M., Jarke, M., Haumer, P., Pohl, K., Dubois, E., and Heymans, P.A. (1998), A Proposal for a Scenario Classification Framework. Requirement Engineering, 3(1), pages 23-47.

Rosch, E. Mervis, C.B., Gray, W., Johnson, D., and Boyes-Braem, P. (1976), Basic Objects in Natural Categories. Cogn Psychol. (7), pages 573-605.

Russ, T., Valente, A., MacGregor, R. and Swartout, W. (1999), Practical Experiences in Trading Off Ontology Usability and Reusability, In 12$^{th}$ Workshop on Knowledge Acquisition, Modeling and Management.

Scholz, R.W., and Tietje, O. (1996), Methoden der Fallstudie, in Industrieareal Sulzer-Escher Wyss. vdfHochschulverlag, Zürich, Switzerland

Steels, L. (1998), The componential framework and its role in reusability", *Second Generation Expert Systems*, J.-M. David, J.-P. Krivine & R.Simmons, (eds.), pages 273-298. Berlin:Springer-Verlag.

Swartout, W.R. and Moore, J.D. (1993), Explanation in Second Generation Expert Systems, Second Generation Expert Systems, J.-M. David, J.-P. Krivine and R.Simmons(eds),. Berlin: Springer-Verlag.

Tsatsoulis, C., Cheng, Q. & Wie, H-Y. (1997), Integrating Case-Based Reasoning and Decision Theory. IEEE Expert, Vol. 12 Issue 4, pages 46-55.

Tversky, A. and Kahneman, D. (1974), Under Uncertainty: heuristics and biases. Science (185), pages 1124-1131.

Uschold, M. and Gruninger, M. (1996), Ontologies: Principles, Methods and Applications, Knowledge Engineering Rev. Vol. 11, No. 2.

Vartia, P. (1994), Talouden ennustamisen vaikeus. The Research Institute of the Finnish Economy, Helsinki, Finland

Verplank, B., Fulton, J., Black, A. and Moggridge, B. (1993), Observation and Invention – Use of Scenarios in Interaction Design, Tutorial notes, Intechi

Wack, P. (1985a), Scenarios: unchartered waters ahead. Harvard Business Rev, Nov/Dec.

Wack, P. (1985b), Scenarios: shooting the rapids. Harvard Business Rev, Nov/Dec.

Wertheimer, M. (1938), Laws of Organization in Perceptual Forms, A Sourcebook of Gestalt Psychology. Paul, Trench, Trubner, London.

Wright, P. (1992), What's in a scenario? SIGCHI Bulletin, 24(4),11

Yeo, G.K. (1991), A Framework for Developing Simulation Game Systems, Simulation and Gaming, Vol. 22, No. 3, Sept. pages 308-327

Young, R. M. & Barnard, P. (1986), The Use of Scenarios in Human-Computer Interaction Research: Turbocharging the Tortoise of Cumulative Science, Proceedings of the SIGCHI/GI conference on Human factors in computing systems and graphics interface, Toronto, Ontario, Canada, pages 291 – 296, New York, NY, USA, ACM Press.