

A Scenarios Mediated Approach for Tacit Knowledge Acquisition and Crystallisation: Towards Higher Return-On-Knowledge and Experience

Cheah Yu-N
School of Computer Sciences
Universiti Sains Malaysia
11800 Penang, Malaysia
yncheah@cs.usm.my

Syed Sibte Raza Abidi
School of Computer Sciences
Universiti Sains Malaysia
11800 Penang, Malaysia
sraza@cs.usm.my

Abstract

The 'Knowledge Age' has fuelled the need to capitalise on organisation-wide Intellectual Capital with the aim of gaining competitive advantage vis-à-vis a higher return-on-knowledge and experience. In this paper, we propose a novel tacit knowledge acquisition and representation strategy using Scenarios based on the assumption that tacit knowledge can best be explicated through controlled challenge situations. We also describe in detail, a knowledge crystallisation algorithm that is used to refine the acquired tacit knowledge by modelling the natural mechanics of crystallisation and annealing. We conclude by asserting that our approach would further enhance organisation-wide performance through its superior quality and value-added delivery of organisational services.

1 Introduction

In today's 'Knowledge Age', there is an ever increasing demand for more sophisticated Knowledge Management (KM) methodologies and techniques to provide the so-called 'next-generation business solutions'—solutions that endeavour to provide a higher return-on-knowledge to the parent enterprise [Lan99]. It is widely acknowledged that an organisation's competitive advantage and its capacity to achieve higher return-on-knowledge is derived from the effective operationalisation and management of its *Intellectual Capital*. With definitions abound, Intellectual Capital is generally described as comprising the *Human Capital*, *Structural Capital*, and *Relational (or Social)*

Capital of an organisation. Put simply, Human Capital is the knowledge, skills, experiences and intuitions possessed by individuals in an organisation; Structural Capital refers to knowledge that is embedded within an organisation operational system, i.e. an organisation's processes, workflows, systems, policies and procedures; and finally Social Capital refers to the organisation's relationships with its network of customers as well as its network of strategic partners and stakeholders. Of the three facets of Intellectual Capital, in this paper we will focus on Human Capital.

Vis-à-vis Human Capital, the knowledge possessed by an individual can be broadly differentiated between *Explicit Knowledge* and *Tacit Knowledge*. Explicit Knowledge can best be described as canonical knowledge, i.e. knowledge formalised within databases, business rules, manuals, protocols and procedures and so on. Tacit Knowledge is non-articulated or non-canonical knowledge, i.e. knowledge that does not manifest as rules. Rather it exists as the domain experts' skills, common sense and intuitive judgement whilst solving problems [Che00]. The problem in many organisations today, we believe, is that explicit knowledge has been given more prominence and that experience- and skill-rich tacit knowledge is ineffectively or even hardly captured and utilised. This reliance on explicit knowledge may also result in rigid policies and thinking patterns that would hinder an organisation's ability to gain competitive advantage and, thus, resulting in low return-on-knowledge and experience.

In this paper, we will focus on knowledge creation strategies [Non94], in particular the acquisition, representation and crystallisation of tacit knowledge with the aim of allowing tacit knowledge to be utilised effectively to gain competitive advantage with higher return-on-knowledge and experience. By using healthcare as an exemplar domain, we present a novel tacit knowledge explication approach that purports the presentation of *Scenarios* pertaining to hypothetical problem situations to healthcare experts and in turn to record their 'tacit' problem-solving methodology and knowledge in solving the given problems. The specialised knowledge extracting 'scenarios' are custom designed to reflect atypical problems – i.e. not the kind of problems that can be solved by routine procedures. Rather, these are

The copyright of this paper belongs to the paper's authors. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage.

Proc. of the Third Int. Conf. on Practical Aspects of Knowledge Management (PAKM2000)
Basel, Switzerland, 30-31 Oct. 2000, (U. Reimer, ed.)

<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-34/>

problems whose solutions may demand an interplay of informal and ad hoc intuitive (or based on experience) judgements with formal problem-solving strategy.

The forthcoming discussion spans across the lifecycle of knowledge creation; initiating with a discussion on the proposed specialised knowledge extracting *Scenarios* vis-à-vis mental models (as suggested in cognitive science literature). We will discuss in detail, a scenario structure and representation scheme that makes reference to the notion of *Meta-Scenario* and its components. Next, we will proceed to discuss scenario acquisition mechanisms. Finally, we will present the latest extension of our work, i.e. a technique for incremental knowledge refinement and categorisation through knowledge crystallisation which is based on the novel notion of *Knowledge Nucleation and Growth* [Che00] – the formation of knowledge crystals by the amalgamation of multiple contextually/structurally similar scenario items. The work reported here purports a synergy between artificial intelligence techniques (for representation, reasoning and learning purposes) with existing concepts and practices in knowledge management. In conclusion, we assert that our approach would further enhance organisation-wide performance through its superior quality and value-added delivery of organisational services.

2 Tacit Knowledge Acquisition and Representation Using Scenarios

Generally, traditional approaches, such as the interviewing of domain experts, and subsequently observing and analysing their problem solving methodologies [Mor91], obtaining knowledge from reference materials and databases, and other techniques such as role playing, talkback, 20 questions, repertory grid, etc. only work well to procure explicit knowledge. With this in mind, it is widely contended that traditional strategies are ineffective when it comes to acquiring an expert's tacit knowledge. This is because traditional knowledge acquisition techniques do not take into account the intrinsic origin and composition of tacit knowledge during its acquisition.

In view that tacit knowledge is highly intangible, abstract and hidden, one can attribute its origin to be seemingly incorporated, embedded or interleaved with certain innate and essential skills – problem-solving skills, analytical skills and generalisation or abstraction skills. We argue that it is the selective and systematic manipulation of these innate skills, subject to the nature and specification of the problem to be solved, that brings into relief the so-called tacit knowledge that we seek to capture. Furthermore, with regards to representation schemes, the mention of schemata and mental models vis-à-vis tacit knowledge representation is of relevance here as it posits that mental models are perceptions of the world. Thereby, an insight into tacit knowledge representation (within our minds) can be achieved by understanding the cognitive make-up of such mental models.

In order to acquire tacit knowledge from domain experts, we propose a strategy that is grounded in the assumption that tacit knowledge can best be explicated by

provoking domain experts to act and apply their knowledge and skills to solve novel or atypical problems. Such a provocation is to be achieved by repetitively presenting domain experts 'hypothetical' *Scenarios* [Che00] pertaining to novel or atypical problems and then observe and analyse the domain expert's tacit knowledge-based problem-solving methodology and procedures. In this context, the proposed problem-specific scenario presents domain experts the implicit opportunity to introspect their expertise and knowledge in order to address the given problem, to explore their 'mental models' pertaining to the problem situation and solution, and finally to apply their skills and intuitive decision making capabilities. This sequence, allows tacit knowledge to be 'challenged', explicated, captured and finally to be stored. This strategy is in line with the notion of 'contrived' knowledge acquisition techniques [Spe99, Sha90].

The novelty of our approach draws from the way 'hypothetical' atypical problem situation (a problem-rich scenario that characterises the various elements and peculiarities of a possible real-life situation) is derived. Another novelty stems from the goal of the our strategy – i.e. to explicate tacit knowledge vis-à-vis charting out the domain expert's thought processes, or more appropriately, mental models (as in cognitive science) for a set of well-defined problems. Having introduced and discussed the rationale behind the use of scenarios, we will now provide a description of scenarios.

2.1 Description of Scenarios

A scenario, by itself, is a customised, goal-oriented narration or description of a situation, with a mention of actors, events, outputs and environmental parameters. Put simply, a scenario (a) depicts a sequence of distinct actions that might be taken to accomplish a particular task; and (b) details the sequence of interactions – comprising exchange of messages and responses to intermediate outcomes – performed or experienced by entities to fulfil the goal [Bee98]. In essence, a scenario is a collection or sequence of *hypothetical* (but mimicking real) situations encountered by a domain expert, together with the intermediate responses/actions by the domain expert and henceforth describes one or more episodes or events of the scenario.

From a cognitive science perspective, a scenario can be deemed as a means to explicate the domain expert's *mental model* of the problem and its solution. Mental models are not explicit and need to be inferred from by putting them into action. This description of mental models therefore relates closely to the qualities of tacit knowledge and its acquisition through the use of scenarios [Far88].

From an artificial intelligence perspective, a scenario is very similar to a *Case*. However, the major distinction between the two is that a case is a real-life situation-action structure, whereas a scenario represents a sequence of hypothetical situations carefully designed to draw out tacit knowledge. Furthermore, cases are merely 'frozen' snapshots of an episode with no apparent attempt to

capture significant temporal or sequential elements. On the contrary, as per our specifications, scenarios can manifest a temporal nature whereby they can capture the sequence of events as they may have occurred during a particular episode. In summary, we posit that a scenario is seemingly a more apt representation of an episode or situation, as compared to the traditional case-based representation.

Functionally, the proposed scenarios is akin to the *Scripts* knowledge representation construct [Sch77], However, we would like to point out that scripts are somewhat limited in representing the context of the situation in its entirety. On the contrary, our proposed scenarios cater for the clear representation of context vis-à-vis the provision for the linking of contextual documents to the sequence of events (described within the scenario). We believe that the explicit characterisation of the full context – i.e. a description of the social settings, resources and goals of the users [Nar92] – in describing a particular situation is an important consideration with regards to

situational descriptions. Since tacit knowledge is both rich in content and is also ‘deep rooted’, such contextual information is imperative for the successful illustration of the bigger picture, i.e. the sequence of episodes and events. To support our claim pertaining to the efficacy of a scenario-based representation, we will now discuss in detail the proposed generic structure of a scenario.

3 Generic Scenario Structure and Representation: An Overview

Scenarios may be composed of four main components [Sch97, Pot94]: *Meta-Scenario*, *Scenario-Construct*, *Episode* and *Event*. For pragmatic reasons, scenarios are represented by a four-tier scheme where Meta-Scenarios are placed at the top level followed by Scenario-Constructs, Episodes and Events at the bottom level (see Figure 1).

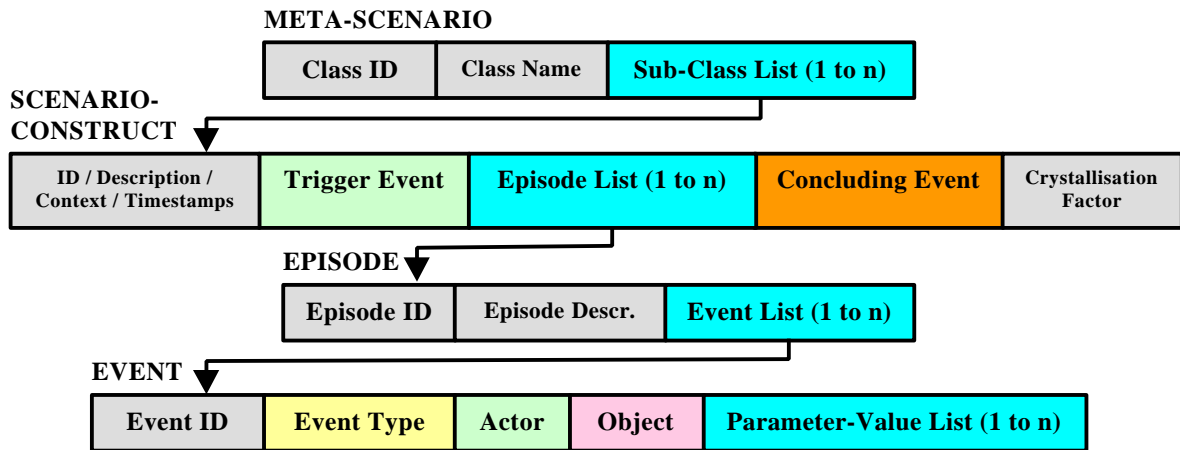


Figure 1: The Scenario Structure outline.

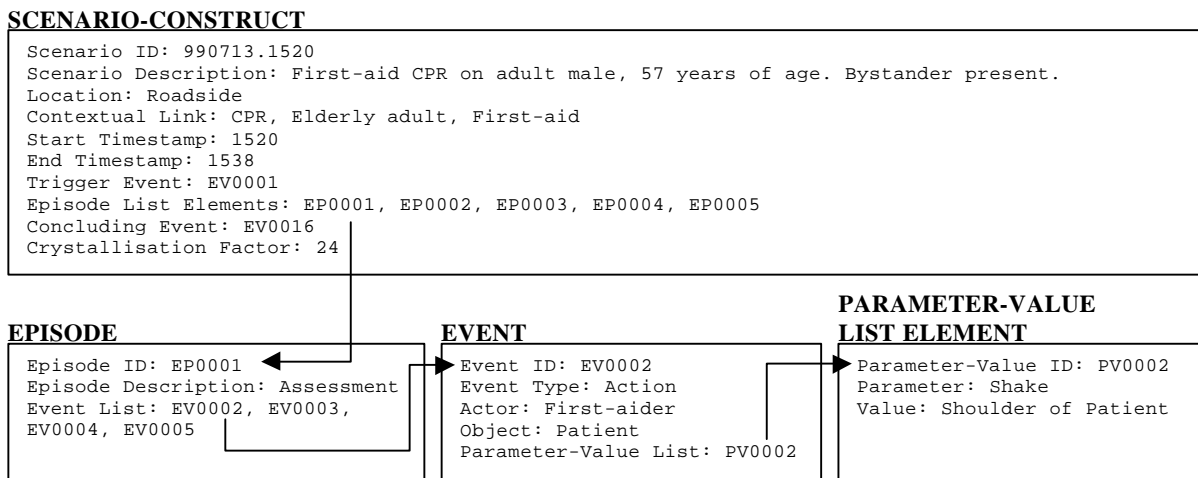


Figure 2: Sample Scenario-Construct, Episode, Event and Parameter-Value List Element.

The scenario storage medium – i.e. a *Scenario Base* – adheres to the same scheme such that the various components of a scenario are stored in distinct repositories. An exemplar Scenario-Construct with a sample Episode, Event and Parameter-Value List Element from a cardiopulmonary resuscitation scenario base are shown in Figure 2. We discuss, below, the four scenario components.

3.1 The Meta-Scenario Component

The Meta-Scenario component serves to implement a two-level (class and sub-class) categorisation of scenarios. Each category is called a class of scenarios and would have a series of Sub-Class List Element (one for each sub-class). Each meta-scenario could have the representation shown in Figure 3.

	Example
Class ID	CL0001
Class Name	CPR
Scenario Sub-Class	CPR for Adult
Scenario List	990713.1520, 990726.2053
⋮	
Scenario Sub-Class	CPR for Infant
Scenario List	990804.1037

Figure 3: Representation of a Class in a Meta-Scenario. Shaded rows indicate a Sub-Class List Element.

3.2 The Scenario-Construct Component

The Scenario-Construct – a constituent of a scenario – stores the description of individual scenarios. The representation for the Scenario-Construct is shown in Figure 4. Scenario-Constructs comprise a sequence of episodes that are arranged in chronological order to mimic the temporal characteristics of the scenario. Such a representation scheme ensures tractability in terms of the sequencing (or chaining) of multiple episodes within a scenario.

	Example
Scenario ID	990713.1520
Scenario Description	First-aid CPR on adult male, 57 years of age. Bystander present. Location: Roadside.
Contextual Link	CPR, Elderly adult, First-aid.
Start Timestamp	1520
End Timestamp	1538
Trigger Event	EV0001
Episode List	EP0001, EP0002, EP0003, EP0004, EP0005
Concluding Event	EV0016
Crystallisation Factor	24

Figure 4: Representation of a Scenario-Construct.

A unique feature of the Scenario-Construct is the Contextual Link field, which stores keywords to help locate (through a search on specific document bases) formal or informal documents containing contextual information pertaining to the episodes and events of a particular scenario.

The Scenario-Construct also has a Crystallisation Factor field that indicates how often the scenario was accessed and judge as useful. The role of this field will be further discussed in Section 5.2.

3.3 The Episode Component

The Episode component stores details of individual episodes of a scenario (see Figure 5). Each Episode comprises an Event List that stores the sequence of events that make up an episode in a scenario.

	Example
Episode ID	EP0001
Episode Description	Assessment
Event List	EV0002, EV0003, EV0004, EV0005

Figure 5: Representation of an Episode.

3.4 The Event Component

The Event component stores details about individual events. The representation for an Event is shown in Figure 6. There are three Event Types: *Normative* – events that are expected to occur on a normal basis, *Obstacle* – events that hinder the progress of the task, and *Action* – events that define the course of action undertaken by an actor.

	Example
Event ID	EV0002
Event Type	Action
(Actor)	First-aider
(Object)	Patient
Parameter-Value List	PV0002

Figure 6: Representation of an Event.

The IDs of parameters and values of an event (in the form of Parameter-Value List Elements) are stored in the Parameter-Value List.

In our scenario structure, episodes and events are generic in nature until they are bound at the Scenario-Construct (scenario base) level through the order in which they are arranged and also through the effect of the Start and End Timestamps. Having formalised the scenario representation, we would now discuss the scenario-based (tacit) knowledge acquisition methodology.

4 Tacit Knowledge Acquisition Methodology: The Use of Scenarios

To facilitate organisational knowledge acquisition activities, we have developed a tool called the *Scenario Composer* [Che00] that facilitates domain experts to

respond to a given scenario through the use of a series of electronic forms whose attributes correspond to a particular component of a scenario, i.e. Meta-Scenarios, Scenario-Constructs, Episodes and Events. They prompt domain experts to provide information or suggest values to the various scenario-defining attributes presented in the electronic form. Figures 7 and 8 are screenshots of the Scenario Composer

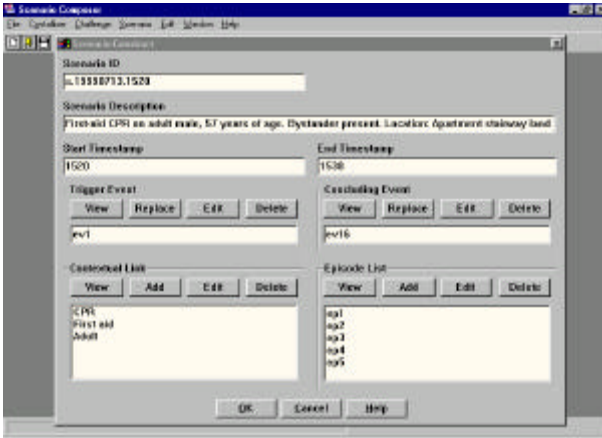


Figure 7: Scenario-Construct screenshot.

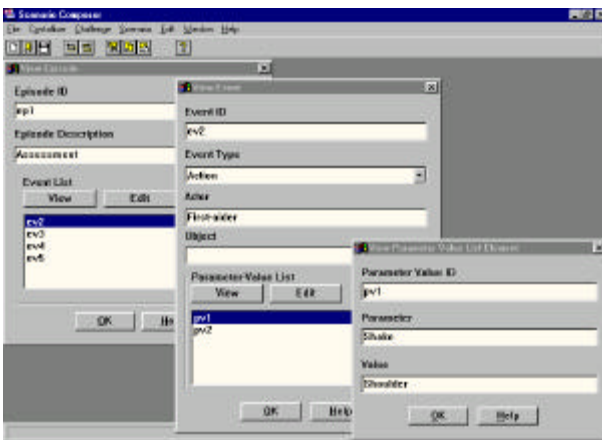


Figure 8: Episode, Event and Parameter-Value List Element screenshot.

Our novel scenario acquisition exercise distinguishes between two types of scenarios:

1. *Solved Scenarios*: Scenarios that define actual situations/problems that have already been encountered and solved/addressed by domain experts. They are akin to traditional form-based cases that are acquired through traditional knowledge acquisition techniques. Scenario bases start out as having only solved scenarios.
2. *Challenge Scenarios*: Scenarios that represent atypical situations and are posed to domain experts as a challenge to their expertise. Challenge scenarios are in line with our contention that tacit knowledge is best explicated when experts are required to solve atypical problems. In most instances, challenge scenarios are

derived from existing solved scenarios by way of selecting a *Point of Interrogation (POI)* – a distinct point in the scenario between two events of type *Obstacle* or *Normative* and an event of type *Action*. The result is a challenge scenario that is then presented to the domain expert for the explication of his/her tacit knowledge (see Figure 9). The construct following the *Challenge* and *POI* captures the domain expert’s response, i.e. the explicated tacit knowledge.

For practical purposes, once a scenario base is sufficiently populated with knowledge – derived from both solved and challenge scenarios – it can then be used for an assortment of knowledge-driven activities. In the healthcare domain, for instance, it can be used for *Healthcare Enterprise Modelling* [CA99, Che99b].

As it turns out, there is no restriction on the terms used by the experts. Therefore, there is a need to integrate domain ontologies with the Scenario Composer to enforce a certain degree of standardisation. Without doubt, domain ontologies hold an important place in our scenario-based tacit knowledge acquisition mechanism. This is because experts tend to use different, though similar, terms in expressing themselves. The inconsistent use of terms could cause serious problems especially when scenarios are eventually used for inferencing purposes.

Ontologies, in our scenario context, could be integrated or implemented in a pre- or post-input fashion. We can also view it as being proactive or reactive. This means that we could perform standardisation either before or after a scenario is accepted into the scenario base. Ideally, this should be done before a scenario is accepted to avoid ambiguity in the event the expert is no longer unavailable to provide clarification. Upon detecting a potentially ambiguous terminology (such as those that are applicable in more than one context, or those that have more than one meaning), this pre-input mechanism would suggest standardised terms for the expert to choose. In the post-input version, the expert has little or even no control of how this mechanism would standardise the terms used.

The process of acquiring tacit knowledge and making it explicit is only a small step in the ongoing efforts to ‘create’ knowledge [Non94]. Moreover, the scenario base in its ‘natural’ state is deemed inefficient in view that the scenario items are not categorised in such a way that would guide potential inferences by limiting an inference engine’s search scope. Therefore, in the following sections, we would explore further on the modes and phases of knowledge creation leading to the refinement and categorisation of the scenario base through the crystallisation of the explicated tacit knowledge.

5 Knowledge Crystallisation

Nonaka proposes four processes or modes that effectuate the transformation between tacit knowledge and explicit knowledge [Non94]. These transformations span across five distinct phases, as shown in Figure 10. In a KM parlance, crystallisation is an integral process in the *creating concepts* phase—it refers to the process where

Scenario	Trigger Event	Event ID	Event Type	Event Description		
First-aid CPR on adult male, 57 years of age. Bystander Present	Episode	EV0001	Obstacle	Patient has pain at centre of chest, lasting more than a few minutes, radiating to shoulders, neck and arms.	Challenge	
		EV0002	Action	First-aider shakes shoulder of patient gently and shout to ask if patient is alright.		
		EV0003	Obstacle	Patient's state of consciousness is unresponsive.		
	Concluding Event			Action	First-aider calls for help.	Expert's Response + 'Tacit
				Action	First-aider requests bystander to telephone Emergency Medical Services.	
				Action	Place patient in a comfortable position.	
			Normative	Patient's pulse is 83 beats per minute and breathing at 15 breaths per minutes. Emergency Medical Service arrives 23 minutes after call made by bystander.		

Figure 9: A portion of a Challenge Scenario showing the Challenge, Point of Interrogation (POI) and the Domain Expert's Response.

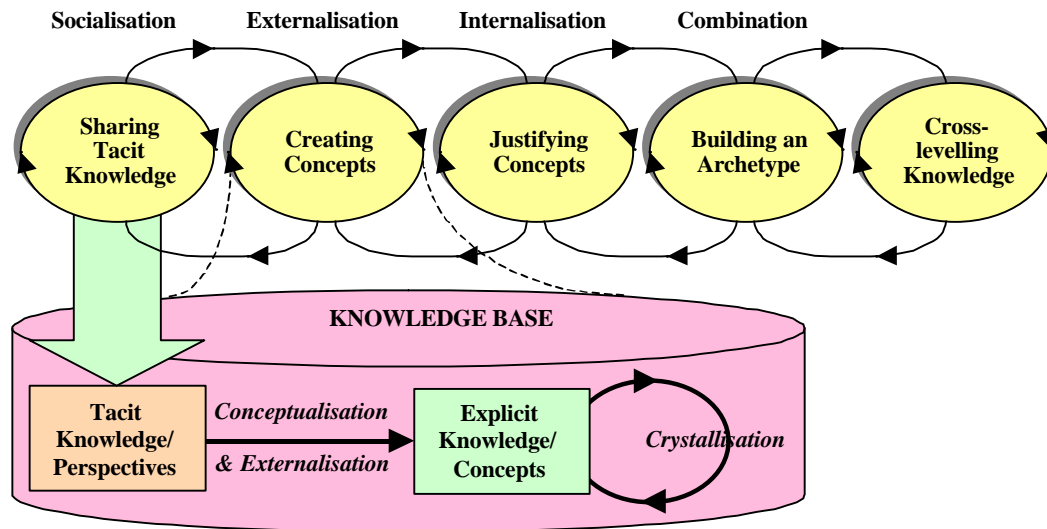


Figure 10: The Five Phase Model of the Organisational Knowledge Creation Process [Non95].

the various departments in an organisation, test the reality and applicability of the created concepts. As a consequence, knowledge that is proven effective, useful and objective is maintained and perpetuated. Now, from a chemical parlance, crystallisation is interpreted to mean to 'solidify and internally arrange'.

In our work we intend to map the notion of chemical crystallisation to a knowledge creation context, whereby: (a) knowledge items (scenario items in our case) mimic molecules, ions or atoms in a supersaturated chemical solution; and (b) the creation of concepts is achieved by

the satisfaction of mutual constraints and unification amongst an ensemble of knowledge items, akin to the process of arrangement of ions, as per chemical rules, to form a crystal. We call the knowledge unit created by our crystallisation process as a *Knowledge Crystal*.

A knowledge crystal can be viewed as a structure with a repetitive arrangement of scenario items in various perspectives, with the objective refine the scenario base at the Scenario-Construct level. Basically, there can be two stages in knowledge crystal formation:

1. *Nucleation*: The formation of a new child knowledge base in a heterogeneous (complex) knowledge base.
 2. *Growth*: The repetitive addition of free scenario items.
- A prerequisite for both processes is *supersaturation* of scenario items. This means that crystallisation could only occur when the *knowledge density* of a knowledge base exceeds a predefined level [Bun96].

5.1 Nucleation

Nucleation begins with the presence of *Knowledge Seeds* in the knowledge base and they are analogous to impurities in a chemical solution. In the state of supersaturation, these knowledge seeds form the nuclei for the growth of knowledge crystals. Conceptually, knowledge seeds serve as a platform on which scenario items grow on.

There are three types of knowledge seeds:

1. *Structural*: The structural knowledge seed ensures that only scenario items of the same structure may interlink to it. The scenario base could potentially contain Scenarios, Episodes and Events that have different structural composition.
2. *Contextual*: The contextual knowledge seed ensures that only scenario items of the same context may interlink to it. It forms a knowledge crystal that is contextually uniform.
3. *Combination (Structural and Contextual)*: This knowledge seed ensures that only scenario items of the same structure and context may interlink to it.

The structure of a knowledge seed may consist of the following items (see Figure 11):

1. *Seed ID*: States the identification number of the knowledge seed.
2. *Seed Type*: Determines if the seed is a Contextual or Structural knowledge seed or a Combination.
3. *Context*: States the context of the knowledge seed. This will be used as the basis to attract scenario items and ensures that the knowledge crystal is contextually uniform.
4. *Scenario Structure*: This field lists down the attributes (fields) of scenario items that can be bound to the Knowledge Seed. This ensures that only scenario items that are structurally the same can link to the seed (as per Structural Knowledge Seeds).
5. *Scenario Item List*: This field lists down the Scenario IDs of all Scenario-Constructs that is linked to the seed, i.e. crystallised.

	Example
Seed ID	KS0001
Seed Type	Contextual
Context	First-aid
Scenario Structure	["Scenario ID", "Scenario Description", "Contextual Link", "Start Timestamp", "End Timestamp", "Trigger Event", "Episode List", "Concluding Event"]
Scenario Item List	["s.19990713.1520", ...]

Figure 11: Structure of a Knowledge Seed.

Nucleation could be initiated when needed by the scenario base administrator or through an automated seeding process. This automated process follows an analysis of the knowledge base's content and nucleation occurs as often as needed depending on the requirements in creating child knowledge bases.

5.2 Growth

Growth occurs after nucleation is established. It is a repetitive addition of scenario items to the nuclei (knowledge seed) through the formation of links to the nuclei. Depending on the knowledge seed, these scenario items can be of the same structure and/or the same context.

Ideally, these links should be established between scenario items of similar structure and context, i.e. originating from a combination knowledge seed. Not only would this fit closely to the original definition of chemical crystallisation but it would also facilitate the implementation and operationalisation of emergent child knowledge bases. Child knowledge bases, comprising scenario items of similar structure and context, can therefore be envisaged as being more subject-focused in terms of their application domain and are also easier to manipulate in terms of query construction.

Besides the knowledge seeds' structural and contextual properties, we posit that the scenario items' 'temperature' or energy level (akin to that in simulated annealing [Tay99, Kir83]) is an important factor in the growth stage. The notion of the energy of a scenario item could be translated into the amount of activity surrounding a particular scenario item, i.e. the number of times it was referred or used by the system. We propose that when a scenario item is referred to and is frequently judged as useful, its accessibility is said to increase. Thus, its level of energy (or activity) decreases, i.e. for a scenario item to be easily accessed, it should not 'move' too much. When the energy level of the scenario item decreases, its ability to crystallise or link to other scenario item increases (in line with thermodynamic principles). Conversely, when a scenario item is referred to and is frequently judge as not useful, its accessibility decreases and energy increases. When this occur, it is less likely to bind with other scenario items. These relations are summarised in Table 1.

It should also be noted that a scenario item can only crystallise when its energy level drops below a certain level, i.e. it is accessed and judged as useful frequently enough. Therefore, a scenario item's frequency of 'useful' access would need to exceed a predefined threshold before it can crystallise.

In order to facilitate the crystallisation process, a *Crystallisation Factor (CF)* field is added to the Scenario-Construct representation (as mentioned in Section 3.2) to store the number of times the scenario item was accessed and judged as useful. The CF would be decremented when it is not judged as useful after a specified period of time. It is possible for the CF to have a negative value in the event that it is not useful for a long time.

Upon completing the crystallisation process, the resultant scenario base may contain (free) scenario items as well as knowledge crystals. Free scenario items are the

Frequency of Access	Accessibility	Energy Level	Crystallisation
↑	↑	↓	↑
↓	↓	↑	↓

Table 1: The Relation between Frequency of Access, Accessibility and Energy Level in the process of Crystallisation.

items of the original scenario base that did not bind to any of the formed crystals. In fact, the resultant scenario base changes from time to time depending on whether new scenario items are added to the scenario base. Figure 12 illustrates the states before and after (or during) crystallisation.

In the Figure 12, two crystals are formed with two different types of knowledge seeds. Scenario Item 'E' is shown linking with the crystal on the left as they are of similar context and/or structure. Scenario Item X (a free scenario item) is on its own, as there are no crystals with a structure or context similar to its own.

With reference to Figure 12, crystallisation is shown as an on-going repetitive process, executed in parallel even while more knowledge is accessed, shared and added into the scenario base. Functionally, crystallisation is terminated momentarily when the density of scenario items falls below the predetermined threshold.

5.3 The Crystallisation Algorithm

In an attempt to crystallise the scenario base, we have incorporated into the Scenario Composer, a knowledge crystallisation component. The steps executed by this component in the crystallisation process are as follows (A flowchart of the algorithm is shown in Figure 13):

- Step 1: *Create Knowledge Seed*: A knowledge seed is created by the domain expert or scenario base administrator using the representation detailed in Section 5.1. For a newly created knowledge seed, the Scenario Item List is an empty list.
- Step 2: *Embed Knowledge Seed*: The knowledge seed is 'placed' into a scenario base to initiate the crystallisation process.
- Step 3: *Attract Similar Knowledge Units*: Search the scenario base for scenario items that have a similar context and/or structure as the seed (depending on the Seed Type field).

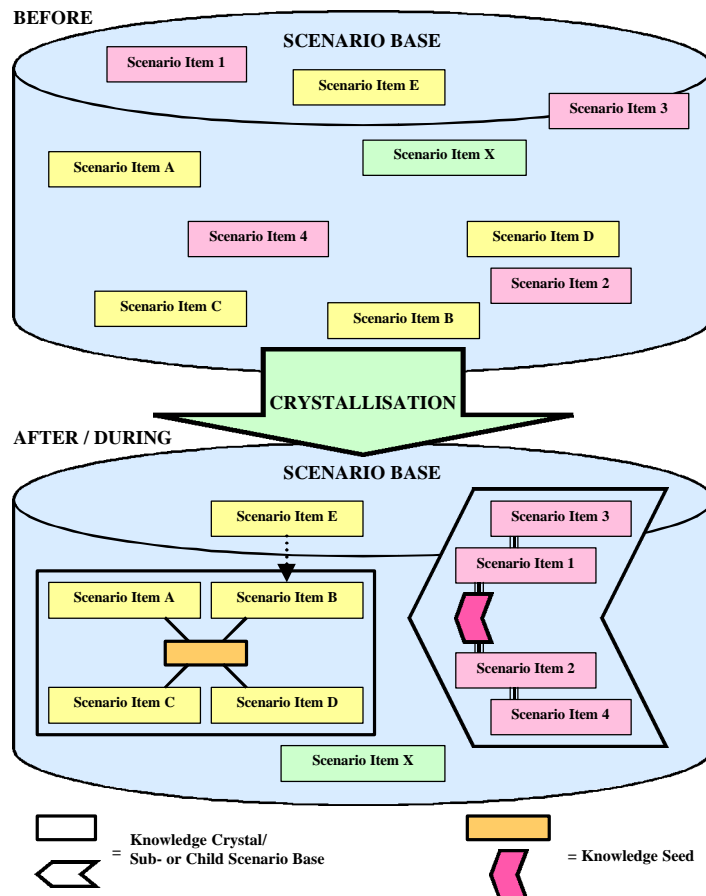


Figure 12: The Scenario Base Before and After (or During) Crystallisation.

Step 4: *Select Most Similar Knowledge Unit*: Among the matched items, select the scenario item that has been frequently accessed and judged as being the most useful, i.e. having the highest CF.

Step 5: *Associate Knowledge Unit to Knowledge Seed*: Link the selected scenario item to the knowledge seed by way of adding the scenario item's Scenario ID to the seed's Scenario Item List.

Step 6: *Prioritise Selected Knowledge Units*: Rearrange the Scenario Item List (of the seed) so that the CF of the scenario items is in descending order.

Note that crystallisation begins after Step 2, provided the scenario base is sufficiently populated by free scenario item. A user-defined threshold determines the number of free scenario items that are needed to initiate the crystallisation process.

In Step 4, as mentioned before, the selected scenario item must have a CF that exceeds a predefined threshold before it can be linked to the knowledge seed in Step 5.

In Steps 4 and 5, the matched items could have simply been sorted according to their CF in descending order and

then added en bloc to the seed's Scenario Item List. However, we choose not to do this (at least for the time being) to mimic more closely the crystallisation process in chemistry (which is incremental in nature). The incremental approach is also more time consuming. Nevertheless, from a chemistry point of view, the slower the crystallisation process, the better the quality of the crystals formed. However, at the moment we are not able to demonstrate any clear advantage for our choice.

Step 6 allows the Scenario-Construct with the highest CF to be at the head of the Scenario Item List. This increases the efficiency of potential inferencing strategies as the most reliable or useful scenarios are considered first. In the previous section, we proposed that the CF of the Scenario-Construct will be decremented, after a predetermined period of time, if it is deemed not useful. This strategy ensures that outdated or less useful scenario items would eventually be among the last to be considered during inferencing.

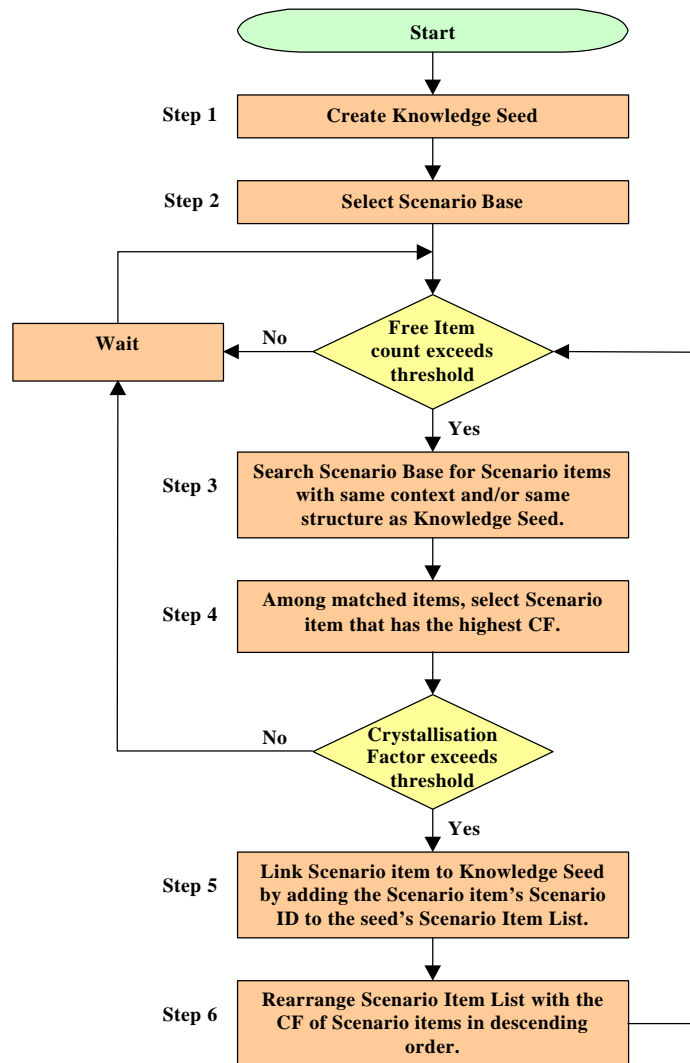


Figure 13: The Knowledge Crystallisation Flowchart.

After Step 6, if the number of free scenario items still exceeds the threshold, the process continues from Step 3. The crystallisation process stops momentarily when the number of free scenario items drops below the threshold only to resume when the threshold is exceeded.

We argue that the manner in which crystallisation is taking place by modelling chemical crystallisation does in fact conform to Nonaka's original view where the explicated tacit concepts are tested for reliability and applicability [Non94]. When a user accesses and evaluates a particular scenario item, he/she is actually testing and affirming its applicability and usefulness. Therefore, we argue that the more a scenario item is accessed and judged as useful, the more applicable the scenario item is deemed to be and more crystallised the concepts or scenarios.

The crystallisation algorithm can be adapted to perform garbage collecting functions, i.e. the removal of scenario items that prove to be very isolated cases and are

not useful. Garbage collecting can potentially be executed in parallel with the existing crystallisation algorithm after Step 3 where instead of selecting the most popular and useful scenario item for crystallisation, we select the one that is least accessed and hence is removed from the scenario base. A detailed discussion of scenario-based garbage collecting is beyond the scope of this paper. Nevertheless, we take this opportunity to highlight its practicality and possibility.

In general, we will like to point out that our approach for knowledge crystallisation renders a more physical and autonomous connotation such that the knowledge itself undergoes a proactive automatic categorisation as compared to the original, more social and reactive approach. Note that our approach to crystallisation aims to link scenario items to a knowledge seed in a crystal-forming paradigm that ranks the scenario items in terms of their applicability and usefulness.

```

<!DOCTYPE SCENARIO_CONSTRUCT [
  <!ELEMENT SCENARIO_CONSTRUCT (SCENARIO_DESC,
    CONTEXTUAL_LINK+, START_TIMESTAMP,
    END_TIMESTAMP, TRIGGER, EPISODES+,
    CONCLUDING, CRY_S_FAC)>
  <!ELEMENT SCENARIO_DESC (#PCDATA)>
  <!ELEMENT CONTEXTUAL_LINK (#PCDATA)>
  <!ELEMENT START_TIMESTAMP(#PCDATA)>
  <!ELEMENT END_TIMESTAMP (#PCDATA)>
  <!ELEMENT TRIGGER EMPTY>
  <!ELEMENT EPISODES EMPTY>
  <!ELEMENT CONCLUDING EMPTY>
  <!ELEMENT CRY_S_FAC (#PCDATA)>
  <!ATTLIST SCENARIO_CONSTRUCT SCENARIO_ID ID
    #REQUIRED>
  <!ATTLIST TRIGGER EVENT_ID IDREF #REQUIRED>
  <!ATTLIST EPISODES EPISODE_ID IDREF
    #REQUIRED>
  <!ATTLIST CONCLUDING EVENT_ID IDREF
    #REQUIRED>
]>

<!DOCTYPE EPISODE [
  <!ELEMENT EPISODE (EPISODE_DESC, EVENTS+)>
  <!ELEMENT EPISODE_DESC (#PCDATA)>
  <!ELEMENT EVENTS EMPTY>
  <!ATTLIST EPISODE EPISODE_ID ID #REQUIRED>
  <!ATTLIST EVENTS EVENT_ID IDREF #REQUIRED>
]>

<!DOCTYPE EVENT [
  <!ELEMENT EVENT (EVENT_TYPE, ACTOR?,
    OBJECT?, PARAMS_VALUES+)>
  <!ELEMENT EVENT_TYPE EMPTY>
  <!ELEMENT ACTOR (#PCDATA)>
  <!ELEMENT OBJECT (#PCDATA)>
  <!ELEMENT PARAMS_VALUES EMPTY>
  <!ATTLIST EVENT EVENT_ID ID #REQUIRED>
  <!ATTLIST EVENT_TYPE EV_TYPE
    (normative|obstacle|action) #REQUIRED>
  <!ATTLIST PARAMS_VALUES PARAM_VALUE_ID
    IDREF #REQUIRED>
]>

```

Figure 14: Exemplar Scenario-Construct DTD.

```

<SCENARIO_CONSTRUCT SCENARIO_ID="990713.1520">
  <SCENARIO_DESC>First-aid CPR on adult male,
    57 years of age. Bystander present.
    Location: Roadside.</SCENARIO_DESC>
  <CONTEXTUAL_LINK>CPR</CONTEXTUAL_LINK>
  <CONTEXTUAL_LINK>Elderly
    adult</CONTEXTUAL_LINK>
  <CONTEXTUAL_LINK>First-aid</CONTEXTUAL_LINK>
  <START_TIMESTAMP>1520</START_TIMESTAMP>
  <END_TIMESTAMP>1538</END_TIMESTAMP>
  <TRIGGER EVENT_ID="EV0001"/>
  <EPISODES EPISODE_ID="EP0001"/>
  <EPISODES EPISODE_ID="EP0002"/>
  <EPISODES EPISODE_ID="EP0003"/>
  <EPISODES EPISODE_ID="EP0004"/>
  <EPISODES EPISODE_ID="EP0005"/>
  <CONCLUDING EVENT_ID="EV0016"/>
  <CRY_S_FAC>24</CRY_S_FAC>
</SCENARIO_CONSTRUCT>

<EPISODE EPISODE_ID="EP0001">
  <EPISODE_DESC>Assessment</EPISODE_DESC>
  <EVENTS EVENT_ID="EV0002"/>
  <EVENTS EVENT_ID="EV0003"/>
  <EVENTS EVENT_ID="EV0004"/>
  <EVENTS EVENT_ID="EV0005"/>
</EPISODE>

<EVENT EVENT_ID="EV0002">
  <EVENT_TYPE EV_TYPE="action"/>
  <ACTOR>First-aid</ACTOR>
  <OBJECT>Patient</OBJECT>
  <PARAMS_VALUES PARAM_VALUE_ID="PV0002"/>
</EVENT>

```

Figure 15: Exemplar Scenario-Construct instance in XML.

5.4 The Final Representation of Scenarios

Ideally, the explicated and crystallised tacit knowledge would be stored in a representation language that is portable, flexible and expressive. Currently, a likely choice of representation language is the Extensible Markup Language (XML). The choice of XML is justified further by the fact that it is easily parsed and thus, facilitates the translation of the XML document into other representational language. Exemplar Document Type Definition (DTD) fragments for the Scenario-Construct, Episode and Event representations and their XML instances are shown in Figures 14 and 15 respectively.

6 Concluding Remarks

We believe that we have initiated a novel scenario-mediated approach to facilitate the capture and utilisation of the domain experts' tacit knowledge, thereby leading to the crystallisation of the acquired knowledge. Here we will like to point out that we do understand that the basis of our tacit knowledge acquisition approach is to some extent akin to traditional knowledge acquisition strategies, such as critiquing, role playing and simulation. However, we believe that the novelty of our approach derives from the following facts: (1) tacit knowledge is 'invoked' by not mere interviewing the domain expert or document analysis, rather by subjecting the domain expert to solve 'controlled' challenges (hypothetical atypical problem situation) that itself are derived from existing solved real-life scenarios; (2) the scenario knowledge construct represents the hierarchical make-up of tacit knowledge in terms of an ensemble of distinct knowledge units; (3) the acquisition of tacit knowledge follows the formal structural specification of the scenario, thereby ensuring a mapping of user-mediated knowledge items (with varying degrees of specificity and context) to a scenario structure; and (4) the crystallisation of acquired tacit knowledge in terms of the chemical crystallisation process.

In this paper, we have discussed the said tacit knowledge acquisition and representation approach that attempts to capture the essence of expert-quality problem-solving using scenarios. We have also seen how natural phenomena, such as crystallisation and annealing, can be effectively adapted into Knowledge Management efforts to refine and categorise knowledge and to allow it to dynamically evolve into scenario or knowledge bases that are capable of providing relevant and up-to-date knowledge on-demand. The definition and development of scenarios, the crystallisation algorithm and the Scenario Composer are ongoing. Nevertheless, it is hoped that our methodology would lead to the improvement of organisation-wide return-on-knowledge and experience resulting in the superior quality and value-added delivery of organisational services.

References

- [Bee98] G.W. Beeler, W. Rishel, A.M. Shakir and M. Walker. Message Development Framework Version 3.1. *Health Level Seven, Inc.*, 1998.
- [Bun96] H.R. Bungay III. Crystallization. <http://www.esb.ucp.pt/~bungah/coag/crys.htm>, 1996.
- [Che00] Y.-N. Cheah and S.S.R. Abidi. A Strategy for Knowledge Acquisition and Representation: A Case for Scenarios. *3rd Int. Conf. on The Practical Application of Knowledge Management (PAKeM 2000)*, Manchester, 2000.
- [Che99a] Y.-N. Cheah and S.S.R. Abidi. Healthcare Knowledge Management Through Building and Operationalising Healthcare Enterprise Memory. In P. Kokol, B. Zupan, J. Stare, M. Premik and R. Engelbrecht (Eds.), *Medical Informatics Europe '99*, Ljubljana, Slovenia, Amsterdam: IOS Press, 1999.
- [Che99b] Y.-N. Cheah and S.S.R. Abidi. Evaluating the Efficacy of Knowledge Management and Organisational Memory Toward Healthcare Enterprise Modelling. *Workshop on Knowledge Management and Organizational Memories, 16th Int. Joint Conf. on Artificial Intelligence (IJCAI '99)*, Stockholm, 1999.
- [Far88] M.U. Farooq and W.D. Dominick. A Survey of Formal Tools and Models for Developing User Interfaces. *Int. Journal of Man-Machine Studies*, 29:479-496, 1988.
- [Kir83] S. Kirkpatrick, C.D. Gerlatt Jr. and M.P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671-680, 1983.
- [Lan99] A. Lancini and E. Mercier-Laurent. Traffic Road Accidents Return on Experience. Discovering, Organizing and Sharing Knowledge to Minimize Risks and Costs in a Mutual Insurance Company. *6th Int. Symp. on the Management of Industrial and Corporate Knowledge (ISMICK '99)*, The Netherlands, 1999.
- [Mor91] I. Morrison, B.A. Schaefer and B. Smith. Knowledge Acquisition: The Acquire® Approach. *1st Semi-Annual Conf. in Policy Making and Knowledge Systems*, 1991.
- [Nar92] B.A. Nardi. The Use of Scenarios in Design. *SIGCHI Bulletin*, October, 1992.
- [Non94] I. Nonaka. A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*, 5(1):14-37, 1994.
- [Non95] I. Nonaka and H. Takeuchi. *The Knowledge-Creating Company*. New York: Oxford University Press, 1995.
- [Pot94] C. Potts, K. Takahashi and A. Anton. Inquiry-Based Scenario Analysis of System Requirements. *Int. Conf. on Requirements*

- Engineering (ICRE '94)*, Colorado Springs, 1994.
- [Sch77] R.C. Schank and R.P. Abelson. Scripts, Plans, and Knowledge. In P.N. Johnson-Laird and P.C. Wason (Eds.), *Thinking: Readings in Cognitive Science*. Cambridge: Cambridge University Press, 1977.
- [Sch97] A.C. Schultz, J.J. Grefenstette and G.A. De Jong. Learning to Break Things: Adaptive Testing of Intelligent Controllers. In T. Baeck, D.B. Fogel and Z. Michalewicz (Eds.), *Handbook of Evolutionary Computation*, Bristol: IOP Publishing Ltd., 1997.
- [Sha90] N.R. Shadbolt and A.M. Burton. Knowledge Elicitation. In J.R. Wilson and E.N. Corlett (Eds.), *Evaluation of Human Work: A Practical Ergonomics Methodology*, 321-345, London: Taylor and Francis, 1990.
- [Spe99] P.-H. Speel, N. Shadbolt, W. de Vries, P. H. van Dam and K. O'Hara. Knowledge Mapping for Industrial Purposes. *12th Workshop on Knowledge Acquisition, Modeling and Management (KAW '99)*, Banff, 1999.
- [Tay99] T. Tay. Groping the Elephant. <http://www.comp.nus.edu.sg/~tayhanbo/cs1305/index.html>, 1999.