Chapter 4

Empirical Modelling Technology For The Wider Agenda For Computing

4.0 Overview

Chapter 3 presented an overview of the characteristics and properties of model building in EM. This chapter considers the implications of EM on the technical and strategic demands for the wider agenda for computing in finance. Section 4.1 discusses how EM tackles the technical demands for this agenda. It argues that EM introduces a paradigm shift in the software system development activity by taking into account the experiential, the situated, and the human centred aspect of SSD. This section presents also the efforts of EM in researching a suitable framework for deploying prevalent technologies to support the paradigm shift in SSD. Section 4.2 attempts at illustrating, with diverse case studies, how the strategic demands for the wider agenda for computing in finance can be addressed by adopting the technical solutions provided by EM. The case studies provide a proof of concept of the potential success of EM in establishing a closer integration between the computing activity - the software system development activity - and various activities in the real world domain, that can be better mediated with computer-based technology. These activities include: communication, learning, business process modelling, decision support, product design, and visual exploration. The relevant case studies considered are: the railway accident, the virtual electronic laboratory, the warehouse, the timetable, and the attribute explorer. Section 4.3 concludes with potential prospects for the success of EM in tackling the wider agenda for computing in finance at the institutional, market, and investment levels and constitutes a background for case studies tackled in subsequent chapter 5, 6, 7, and 8.

4.1 Meeting the Technical Demands of the Wider Agenda for Computing in Finance

The technical demands for the wider agenda for computing in finance imply the need for greater integration between the human and automatic activities (semi-automation), the account for the situated and experiential aspects of the SSD activity, and the novel use of computer-based technology with other technologies (cf. electronic devices). This motivates a paradigm shift in the software system development activity and its corresponding computer-based support.

4.1.2 The Paradigm Shift in SSD

This section argues that the distinctive qualities of model building in EM introduce a paradigm shift in the software system development activity and its computer-based support: i. using an EM approach, modelling evolves as a central activity throughout the entire SSD activity; ii. the SSD activity in EM is characterised by its openness, situatedness, and the absence of the adoption of circumscribed methodologies that reduce the flexibility of the SSD activity; iii. users-developers-designers interact collaboratively in a situated manner to cultivate requirement engineering in the SSD activity.

i. ***Empirical Modelling Technology motivates a shift in emphasis in the software system development activity from programming to pervasive agent-oriented explanatory modelling.***

Modelling vs programming in traditional SSD

Software is the means that enables a computer to become a tool for solving problems. The traditional software development cycle involves five stages: requirement analysis, specification, design, implementation, and verification. Modelling activity[1] takes place in the requirement engineering, analysis and design phases, whereas the programming activity is involved mainly in the implementation stage. These two activities are fundamentally different in character. This stems from the fact that the modelling activity is more closely related to the real world domain and involves more human insight and input, whereas the programming

[1] Using specification languages such as the Unified Modelling Language UML, and cooperating sequential processes CSP.

activity assumes a detachment from the real world and a reliance on abstraction and circumscription (cf. Figure 4.1).

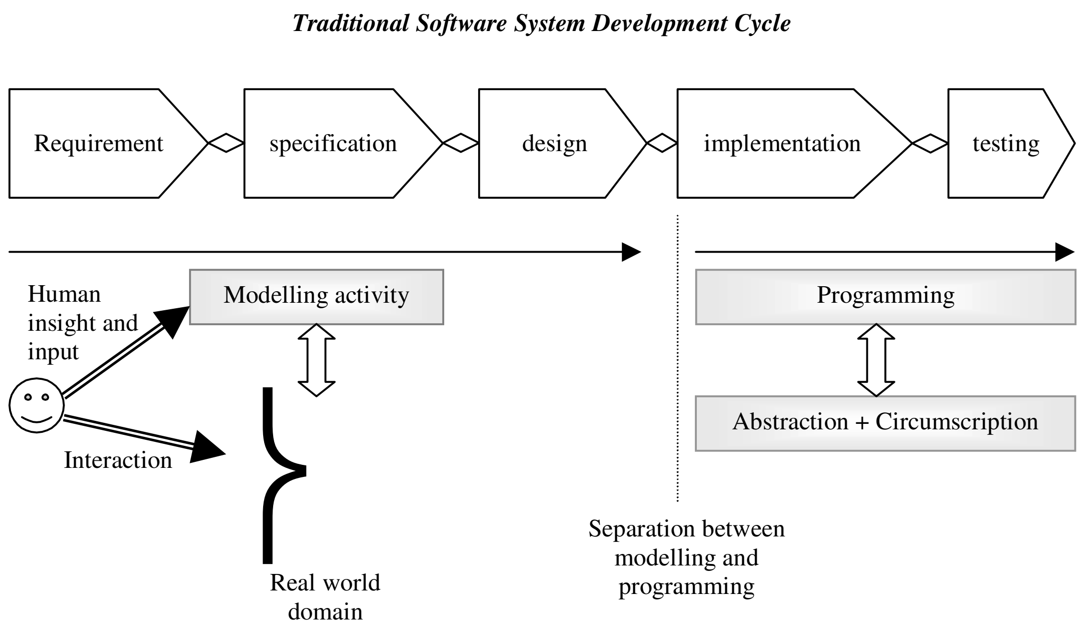


Figure 4.1 Modelling vs programming in traditional SSD

As software systems have become more complex, there has been greater emphasis on the role of modelling in SSD supported by new modelling approaches[2]. There has also been a complementary evolution in programming languages[3], whereby the implementation phase can

[2] Such as Agile Modeling AM (www.agilemodeling.com) that offers a practice-based methodology for effective modelling and documentation of software based systems [Amb02]. AM is a collection of values, principles, and practices for modelling software that can be applied on a software development project in an effective and light-weight manner. An agile modelling approach can be adopted in requirement, analysis, architecture, and design. However, AM is not a complete software process. AM focuses mainly on effective modelling and documentation. It does not include programming activities although it recommends to prove the model with code. AM addresses the issue of how to apply modelling techniques on a software project taking an agile approach such as extreme programming (www.extremeprogramming.org), Dynamic Systems, Development Methods (DSDM) (www.dsdm.org), or SCRUM (www.controlchaos.com).

[3] Programming languages can be classified by generations: Machine codes formed the first generation; auto-codes and assemblers formed the second generation; high-level languages such as Fortran, Cobol, and Pascal formed the third generation; languages for non-professional programmers, such as spreadsheets, are considered as fourth generation. Fourth generation languages include decision support languages. Programming languages can be also classified by their purposes: general multi-purpose, interactive, special purpose. Another classification would be by operation: the fundamental operation of functional languages (e.g. LISP) is the evaluation of expression; the fundamental operation of imperative languages is the use of sequence of commands (e.g. Fortran, Cobol, Pascal, Basic, C); declarative languages attempt to combine the operation of functional and imperative languages [WC88].

use higher level abstractions better suited to the application, thus offering greater support to SSD[4]. Neither of these developments has had a significant impact on the nature of the relationship between the modelling and the programming activities. Programming still presumes a preconceived and abstractly specified functionality and modelling is still used as a means to identify this preconceived functionality. To reach the programming stage the product of the modelling activity has to be rich enough to deliver an appropriate circumscription of the real world associated with a reliable pattern of behaviour and a specified functionality. This implies that modelling takes precedence over programming and that the starting point for the programming activity is determined by the outcome of the modelling.

EM for SSD

EM aims at reconciling the imbalance between theoretical (abstract) and empirical (real world) ingredients in current SSD practices [Hon99]. Empirical Modelling technology has evolved as a means for modelling application domains by focusing on the representation of the state of the real world mediated through metaphorical representations that are developed in a situated manner. The impact of this evolution has been to show that EM activity can lead to the construction of computer-based artefacts whose relationship to their real world context is very similar to that reached in traditional modelling for SSD at the point of circumscription[5]. There are then two significant differences between Empirical Modelling and traditional modelling for SSD:

(1) In EM, the computer-based artefact exists prior to what would serve in traditional modelling for SSD as a point of circumscription.

(2) In EM, the modelling activity can carry on beyond what would serve in traditional modelling for SSD as a point of circumscription, in effect taking the place of the programming activity.

In an EM approach to SSD, the artefact reflects the distinctive qualities identified in section 3.2. These give the artefact an explanatory and agent-oriented character that enables it to be integrated closely with activities external to the system.

---

[4] Extreme Programming (www.extremeprogramming.com) is an example of a disciplined approach to software development that witnessed success in the industry because it stresses testing, customer satisfaction and team work, while implementing simple ways to enable groupware style development [Bec00]. XP aims at improving a software project in four essential ways: communication, simplicity, feedback, and courage.

[5] In particular, see the doctoral theses of Yung (1992), Ness (1997), and Sun (1999).

ii. EM technology proposes an open, situated, and amethodological approach to Software System Development (SSD).

SSD in theory and practice

Studies in the area of software system development focus on the process initiated when capturing the statement of the problem of a client and terminating at the delivery of the software product that will potentially solve the problem. Britton (1993) views system development as a gradual progression from the client's initial vague ideas about the problem, via a series of transitional stages to a complete formal statement, expressed in a programming language, which can be executed on a machine. A typical life cycle of a software system may be divided into seven stages: problem definition, feasibility study, analysis, system design, detailed design, implementation, and maintenance. The early phase of the software system development cycle is commonly referred to as software requirement analysis or software requirement specification [Dav93]. Different approaches, techniques, and notations are used to model a system. Some approaches involve constructing many different models of the system - each leading to a partial description of the system. On account of this, understanding the whole system necessitates an understanding of the relationship between the different partial models and their complementary role. In current software system development practices, there is a clear separation between the representations used in specifying and implementing the developed product. The diversity of applications and the rapidly changing environment pose great challenges to the software system development process. Adopting a general, context independent, methodology to structure the software development process and all its phases is a challenging task because different applications require different approaches. Britton (1993) argues that it is impossible for a single development methodology to prescribe how to tackle the great variety of tasks and situations encountered, and proposes the development of a toolbox of techniques and skills, modelling tools, and approaches to support the development process. Adapting the software development process to a rapidly changing environment is a difficult task especially for ill-defined and volatile software systems [Flo87].

SSD in EM

EM technology proposes an amethodological approach to Software System Development [Sun99]. The proposed software system development process consists of a collection of situated activities that arise in the construction and use of the required system in the real world. This puts a greater focus on the interaction between human agents involved in the software development process and the product-under-development. The product under development is represented by an interactive artefact (ISM) that reflects the evolving software system. The proposed amethodological approach must take both technical and social aspects into consideration in the software development process. In EM, Software System Development begins with an open ended analysis of a domain [RCR00] that informs the development of a system with dynamic rather than rigid boundaries. Deriving useful systems from an EM model necessitates an appropriate circumscription of the model once a desired functionality can be relied upon. However, the boundary of the system need not be preconceived, but can grow as the modellers' understanding develops [RCR00].

iii. An EM approach considers SSD activity as a human centred activity involving a social network of users-developers-designers interacting collaboratively to cultivate the requirements for the system under development

Users and developers role in SSD

Historically, the view of SSD has been transformed from a technical to a social one [HKL95], as technical processes are unable to cope with all the issues[6] surrounding the SSD activity [HKL95, CS90, Mum95]. Many approaches that regard SSD as a social activity have been introduced [CS90, AF95, Mum95]. Their key concern is to facilitate the interaction between the users and developers in the SSD activity. The professional work practices (PWP) approach [Kyn91] aims at improving developer's professional work practices. The co-operative design approach to SSD puts the emphasis on users' involvement as developers. It argues that technology should encourage reciprocal learning between users and developers [CWG93]. Neither PWP nor co-operative design is a method-based approach; they provide principles, concepts and techniques to support a social perspective in SSD.

User-developer-designer collaboration in an EM approach to SSD

SSD in EM involves building a computer-based artefact reflecting the views of users and developers (designers) [Sun99, BRSW98]. This shifts the emphasis from a developer centred approach to SSD that abstracts user requirements rigidly and detaches these requirements from their real world context. In contrast to existing SSD practices that eliminate the human factor from the SSD process by adopting rigid algorithms and preconceived mechanisms, EM regards SSD as a social process that needs technical support. This technical support goes beyond the use of computer for knowledge representation to knowledge construction and knowledge sharing. Distributed Empirical Modelling provides a framework for the collaboration of the users and developers (designers) to cultivate the requirements[7] of the

[6] These include job satisfaction, user resistance, learning and interpersonal interaction.

[7] The early phase of the conventional software system development life cycle is commonly called software requirements analysis or software requirements specification [Dav93]. It involves analysing the software problem and concluding with a specification of the desired external behaviour of the

system under development. In the Distributed Empirical Modelling framework (DEM), the interpersonal interaction between users and developers is supported at two levels: modellers, as external observers, can intervene to shape agency in a concurrent environment; or modellers, as internal obervers, act as agents to carry out interaction through pretend play.

Requirements engineering in an EM approach to SSD

In EM, the situated and context-dependent character of the requirements development phase is emphasized. A framework entitled Situated Process for Requirements Engineering [Sun99] is established, where human agents interact within a collaborative environment, on the basis of their current context and resources, to identify requirements [CRB00]. From an EM perspective it is difficult to limit the requirement engineering activity to a single stage in software system development. This activity is conducted gradually and pervades the whole of the software system development. This is a promising but technically challenging approach to SSD. The ' opportunity cost' of meeting evolving user needs is not being able to come up eventually with a finished product.

4.1.2 The Broad Foundation For Computing

It is unrealistic to expect Empirical Modelling technology alone to lead to a paradigm shift in SSD. The support of other current technologies (artificial intelligence, databases, and virtual reality) is needed. EM technology aims at identifying an appropriate framework for deploying these technologies to effect a paradigm shift in SSD. This has been reflected in EM research directed at: i. new foundations for AI; ii. a new perspective on database technology; and iii. principles for wider use of VR technology in modelling social domains.

i. EM technology proposes new foundations for artificial intelligence that motivate a shift from a logicist AI approach to an experience based AI approach where the activity of developing construals to represent experiential knowledge of the real world domain is central.

The experiential dimension in AI

Broadly speaking, AI is concerned with creating computer programs that can emulate human decision making and behaviour. In approaching AI, Empirical Modelling attempts to find a better ontology for the terms intelligence, learning, artificial intelligence, metaphors, and agent. It addresses the criticism faced by the logicist approach to AI, and investigates the

---

software system to be built. This analysis is referred to as functional description, or functional requirements. Problem analysis and product description are two types of activities conducted in the conventional software requirements phase.

principles and foundations of a computational framework for artificial intelligence that account for private experience in the real world. It also establishes a relationship between language and learning [Bey99].

Private and public knowledge in AI

Empirical Modelling Technology relates the concept of learning and intelligence to the migration from private experience to public knowledge and from empirical interaction in the world to theoretical formulation. This migration involves a series of sequential activities: the interaction with artefacts, the construction of practical knowledge, the identification of agency and dependency, the identification of generic patterns of interaction, non-verbal communication, the identification of common experience and subjective knowledge, and the symbolic representation and use of formal language to describe the private experience. An analogy can be drawn with the migration from private experience to the linguistic description of this experience. Human intelligence is related to the empirical mechanism that leads to the formulation of theories that formulate and circumscribe a reliably occurring pattern of experience. A logicist approach to AI [Kir91, LF91, LI95, Der87, CW96, Bro91, Bro91-2] attempts to frame human activity around objective knowledge and skills. It typically requires all significant operations to be based on highly reliable expectations, and assumes that practice is context independent. In contrast, EM technology accounts for subjective experience and knowledge and advocates an empirical approach to construing a system. A construal in EM is empirically established and experientially mediated. The choice of agents is pragmatic and the changes in the state of the system are not preconceived.

The EM frameworrk for deploying AI

The basic principles and foundation of the Empirical Modelling framework for AI are concerned with construing phenomena in terms of observable, dependency and agency, and constructing physical artefacts, mainly computer-based, to represent them. A virtual correspondence is established between the real world referent and the artefact through continuous interaction with the artefact and its real world referent. The EM approach to AI emphasizes the importance of accounting for subjective and private knowledge (first person concerns), provisional and unreliable insight, and the particular context under consideration. Unlike a closed world simulation, an EM artefact is open and subject to modification in the light of future experience.

In thinking about AI in EM, three perspectives are identified according to the roles for agency in the corresponding computer-based model and the extent to which its behaviour is circumscribed: a first person perspective (the corresponding computer-based artefact represents subjective private and personal knowledge); a second person perspective (the role of agency played in the artefact is projected from a first person to a second person); and a third person perspective (the computer-based artefact resembles a computer program, where agents are predefined and their role is circumscribed and formally specified).

ii. In EM, a database is regarded as a computer model that enables its user to simulate different modes of interaction with a real world system.

The principles of agent oriented databases are elaborated in [BCY95]. Tools to support these principles are under development. From an EM perspective, a database can be regarded as a computer model that enables its user to simulate different modes of interaction with a real world system. Agent oriented modelling is adopted to develop the database computer model. A system of definitions (definitive scripts) is used to record the current values of real world observables, and generalise the principles adopted in the relational language ISBL[8]. EM thinking about agent oriented databases aims to confront the challenge of the changing nature of data and agency in modern database applications and the manner in which data is presented to and manipulated by the user. An EM model that illustrates data manipulation subsuming many traditional database functions is the timetable model discussed in [BWMRR00].

iii. Virtual Reality technology can potentially benefit from adopting EM principles and techniques in modelling complex domains. An EM - VR merge is proposed.

Ontological issues bearing on the questions: “What is reality? What is virtual reality? What is the correspondence between the two?” have been considered in EM [Bey99]. The major contribution of EM technology to VR technology is the proposal for a new technical approach to shape visualization embedded in internet web-browsers alternative to that provided by VRML. This approach, pioneered by Richard Cartwright [Car00, Car99], is based on the transmission of recipes for building shape models at the internet browser. It provides a high degree of open-ended interaction flexibility for users, compact transmission of models over networks, support for distributed multi-user shape modelling, and the representation of data dependencies between application domain data and shape models. Cartwright’s concept of *Empirical Worlds* emerged in the course of applying EM principles to computer-based geometric modelling. The users of Empirical Worlds can generate new geometry on the fly without the need for the shape to have been preconceived and constructed as a set of polygons during the creation of a VRML world. A related proposal for exploiting EM technology in

[8] The relational database query language ISBL was an early example of a definitive notation with relational algebra as the underlying algebra.

deploying VR technology in a social application domain is introduced in [MBG01]. This proposal will be considered in more detail in chapter 6.

In seeking a suitable framework for deploying VR technology in social application domains, EM principles for human computer interaction HCI[9] are potentially of interest. EM research in open-ended HCI, as presented in [BRSW98], advocates a shift from user-centred design as introduced in [PRSBHC94] to the development of explanatory models where both the user and the designer operate within a computer-based environment whose state metaphorically represents the experiential context from which the domain knowledge is derived. Conventional HCI design detaches the designer from the real world environment, whereas HCI design in EM advocates user and designer collaboration and their interaction with the real world environment to develop an explanatory model that serves the development of the computing system.

4.2 Meeting the Strategic Demands of the Wider Agenda for Computing

4.2.1 Closer integration between the SSD activity and diverse activities in real world domains

The paradigm shift in SSD motivated by EM enables a closer integration between the SSD activity and activities undertaken in different application domains, with greater advantage and

[9] Human computer interaction (HCI) is inherently a multidisciplinary subject. It involves theories of human behaviour as well as the principles of computer systems design [PRSBHC94]. Exploration of the social and organizational context of the user is also important. Research in the area of HCI aims at developing the right system to fit a specific purpose, it involves studying users and their tasks and relating this information to design styles, human factors theories, guidelines and standards in order to select an appropriate form of interaction. Computer supported cooperative working (CSCW), hyper and multimedia and virtual reality are state of the art in applications in the area of HCI. It is commonly argued that future human-computer interaction will come to resemble our everyday interaction with the world [BRSW98]. Attempts are made to frame HCI in terms of three interdependent abstractions: methods, tools, users. This tends to promote the view of the computer as a general purpose tool capable of playing the role of many pre-existing tools such as calculator, folders, and desktop utilities. This is challenged by novel uses of computers (beyond conventional uses of existing tools) emerging with the advance of computing and communication. Preece (et al, 1994) believe that there is no single solution to the problem of how to do interaction design. Beynon (et al, 1998) argues that everyday experience of our environment is characterised by an openness to interact, and an ignorance of the consequences of our interaction, that has no counterpart in interaction with closed-world computer models.

potential for success in social application domains. Interpersonal communication, learning, participative business process modelling, strategic decision support, product design and visual exploration are all examples of activities in different application domains where computer-based support has failed so far to address its intended objective. The following points summarize the broad implications of EM technology on these activities.

- ***In computer-based interpersonal communication:*** *EM technology promises richer support for computer mediated interpersonal communication by accommodating various modes of interaction, providing distributed communication of definitive scripts, enabling collaborative communication, and attributing various agent-based roles and views to participants.*

- ***In computer-based group learning:*** *EM technology promises greater support for experiential group learning activity.* *Empirical Modelling Technology is concerned with representing the processes that lead to the discovery of concepts [Bey99].*

- ***In strategic decision support[10]:*** *EM technology proposes an experience based approach to strategic decision support where human and computer activities integrate more coherently in exploring a decision ([RR00], [BRR00-1], [BRR00-2], [RCR00]). EM technology can play three roles in SDSS: it helps in coping with imprecise, qualitative problems; it offers end-user involvement, and distributed access to users. The spreadsheet has been a primary tool for decision support, especially in conducting what-if scenarios. Empirical Models generalise the spreadsheet use in decision support in three respects: presentation, underlying algebra, and agency.*

[10] Decision making is a central human activity. Simon (1960) identifies three phases of decision making: intelligence, design, and choice. Mintzberg et al. (1976) conceives the process of decision making in three major components: problem identification, development of alternative solutions and selection among the alternatives. The computational paradigm that best matches this characterisation of decision making is that of logic programming [Kow79]. Decision making is classified in three levels [Sut89]: strategic, tactical, and operational. Strategic decision support is the most challenging as it involves qualitative description of the system, and decisions either to be taken by top management (end-users), and/or likely to be shared among managers (Group Decision Support Systems). Davis (1996) argues that decision support is more a service than a product. Decision support systems (DSS) are computer-based systems that bring together information from a variety of sources, assist in the organization and analysis of information, and facilitate the evaluation of assumptions underlying the use of specific models [Sau97].

- ***In visual exploration:*** *EM technology proposes an amethodological approach to data analysis and exploration that helps in revealing qualitative and hidden aspects of relationships between data.*

- ***In participative business process modelling (BPM)[11]:*** *EM technology proposes a framework that complements a structured approach[12] to business process modelling with an experience based approach to participative business process modelling where the role of the human in partaking in the modelling process is preserved. A shift in perspective from the paradigm of method-tool-user to a paradigm where the computer and human integrate more coherently to achieve the intended business objective is proposed. EM technology advocates a shift in perspective from specifying a business process by an abstract pattern to specifying a process as developed from a situated[13] activity. In seeking a reliable pattern in the business process, a semi-automated, situated style of modelling is proposed.*

- ***In participative business process re-engineering (BPR)[14]:*** *Recognizing the challenges faced by BPR in attempting to move from the problem domain (real world application) to the solution domain (world of programs and systems) [CRB00] and the difficulty in communicating between the business and IT specialist, Empirical Modelling Technology emphasises the importance of business process comprehension to enable process renewal. The business context and domain are important factors to be considered in BPR. EM*

[11] A business process is a series of activities to achieve a business goal. Modelling a business process means expressing the flow and the dependencies of steps in the respective processes in order to make the dynamic behaviour explicit, to be able to communicate it, to analyse it with respect to possibilities of improvement, and to use it for simulations as well as for controlling automated workflow [Sko98]. This definition embraces the meaning, the objective, and the use of BPM.

[12] Business management aspires to establish environments and routines that operate according to conventions and protocols as reliable as scientific laws [BRR00-2]. This aim might be difficult to achieve because a business process cannot be detached from its real world context as in the case of scientific and engineering applications. A business process is shaped by its environment. Many external, internal, and social factors affect the business process. Rules of the business process might be difficult to observe or adapt to.

[13] Situated models of business processes are motivated by the instability in the business environment.

[14] Re-engineering a process means discovering how a process currently operates, redesigning that process to improve efficiency and remove wastage, and finally implementing the new process using whatever technology is deemed appropriate. BPR seeks to devise new ways of oranizing tasks, organizing people, and making use of IT systems so that the resulting processes better meet the goals of the organization [VRWH94]. Re-engineering entails a radical change not only an improvement [Dav96, Dav96-2]. BPR activity embraces a wide range of disciplines [FRKBCJ00].

technology proposes a SPORE (Situated Process For Requirement Engineering) framework[15], encompassing participative BPR (i.e. supporting many users in a distributed environment) [CRB00].

- ***In product design[16]:*** *Empirical Modelling technology proposes a modelling approach for product design based on observation and experimentation and lays a foundation for a computational environment for modelling a product in its situated context. A high level of interaction, openness, and good presentation media are major requirements identified for this computational environment.*

4.2.2 Illustrative Case Studies

The following review of case studies undertaken throughout the history of the Empirical Modelling projects reveals broad implications of EM on the integration of computer-based technology with real world activities that are inherently not computer-based.

The Railway Accident
Illustrating the EM implications on Computer-based Interpersonal Communication

Motivation

Human interaction within a social network is rich, context dependent, and situated [Son93], [SBK00], [Son99]. The relationship between communications media technology and human communication is key in determining the extent to which human communication is enhanced by computer mediation. Moreover group activity is essential to redress individual bias in subjective perception (cf. Gruber's and Sehl's Shadow Box experiment [Goo90]) and to

[15] The SPORE framework was proposed by Sun (1999) in his doctoral thesis [Sun99]. Within this framework, requirements are viewed as a solution to the problems identified in the application domain and are developed in an open-ended and situated manner. People participating in the requirements engineering process are able to cultivate requirements through collaborative interaction with each other. This collaboration aims at solving problems rather than formal specification of requirements. The input to the SPORE situated model are: central problems of the domain which are subjectively determined by participants; relevant contexts such as organizational goals and policy and the relationship between participants, and available resources such as documents, technology and past experience. The outputs of the model are provisional solutions, new context, new resources and new problems feeding the model. The model is iterative and incremental. This means that it is built from a sequence of structured development cycles.

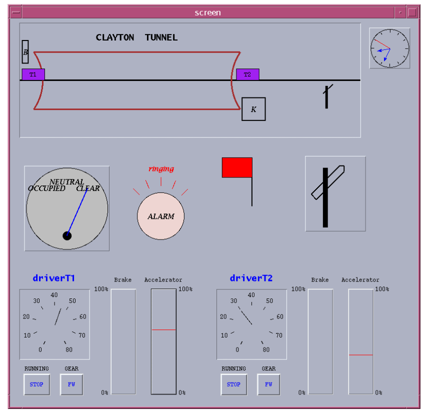
[16] Product design is the activity of generating a solution that meets the requirements imposed on a product [ESc93]. Fischer (et al, 2001) emphasize the importance of the designer creativity during the design process. This creativity is stimulated by the designer observation and knowledge about the product or its requirements. This knowledge is continuously growing as the design progresses.

share insight in cognitive activities including design [Son93], software system development [Sun99], decision support, learning, etc.. This motivates the consideration of a case study that reveals the complexity of interpersonal communication, its impact when mediated via communication technology, the various objectives that can be met through distributing the computer-based activity, and the practical application of emerging technologies for distributed computing.

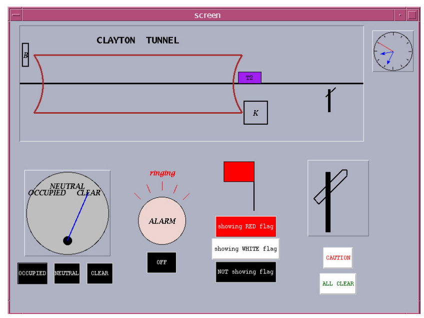
About the model

The railway accident model was developed by P. Sun as a practical illustration of the Distributed Empirical Modelling (DEM) Framework. It is regarded as one of the basic models to demonstrate the practical application of EM principles to various areas in computing. These include:

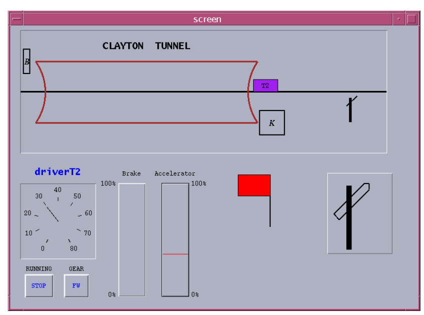
- user-developer-designer collaboration in the course of requirements engineering in software system development [SB99];
- understanding the impact of communication technology on human communication and proposing principles for computer mediated interpersonal interaction that support the sharing of cognitive models [BS99].



God's View (External Observer)



Killick's View (Internal Observer)



Driver's View (Internal Observer)

Figure 4.2 Snapshots of the railway accident model

The model aims at reconstructing a railway accident that occurred in the Clayton Tunnel near Brighton in 1861 [Rolt82]. The modelling has involved constructing computer-based artefacts to represent the views of the human agents involved in the accident (the drivers and signalmen as internal agents) and the global view of an external observer with exceptional state changing privileges (cf. the snapshots of the various views in the model). The model supports sharing insight into the individual understanding of the practices of the signalmen and drivers and how these contributed to the accident.

The personal agents' perspectives are specified by definitions that shape their agency of the use of different observables in the model. Attached to this definitive script is a visualization that represents the modeller's view and the roles of different agents in shaping the state of the distributed interaction.

Conclusions drawn from the model

The case study highlights several advantages that distributed modelling has over centralised modelling in general, as well as others that are specific to the use of EM. An open development paradigm, enabling various modes of interaction and giving various agency roles for participants in the modelling activity, supports a better understanding of the features of the model and the prospect for its use in conjunction with diverse activities in the real world domain. Identifying observables, agents roles, and the view of each participant in the modelling activity is essential for establishing the context of interaction among modellers.

In particular, the distributed railway model reveals the contribution of EM technology to distributed modelling in computer science in three respects:

1) *supporting agent roles and diverse modes and context of interaction:* In DEM, participants acts as internal agents in their own views and through their own privileges of action. A participant' s view in DEM is different in character from an interface in conventional programming. It represents his / her provisional[17] working space including the metaphorical representation of observables that can be seen (**`oracles`**) and manipulated (**`handles`**) at any one moment in time.

2) *distributing the centralised spreadsheet modelling:* The communication of definitive scripts is revolutionary in extending the centralised spreadsheet modelling to accommodate various participants with diverse styles and privileges of action. It extends the openness of the spreadsheet modelling activity at the network level and supports it with views that metaphorically represent the working space of each participant in this modelling activity. The interaction of participants is not limited to features provided by their views but can be endlessly extended (within the attributed privileges) through the communication of definitive scripts. Such communication does not need to follow any preconceived pattern or sequence. In the railway model, the interaction between participants (drivers, signalmen) can be established at any instant through the communication of definitive scripts that affects the state of shared observables in the distributed views of the model (e.g. the communication of definitive scripts concerning the observable **`clear position`** of the telegraph shared in the views of the two signalmen, such communication establishes the functionality of the telegraph device).

3) *supporting openness of interaction:* In DEM, the modelling activity can accommodate different scenarios and different construals for each scenario. The human role in leading the distributed modelling activity is central. Human participants collaboratively share insight and understanding to the extent that the computer-based interaction resembles the real world interaction with its openness and situatedness. In the distributed railway model, different scenarios can be played by human participants (e.g. the accident scenario, and the normal railway operation scenario). Also, a scenario might be construed and played by human participants in different ways (e.g. it might be the case that the inattention of the driver that was the main reason behind the accident, or the carelessness of one of the signalmen in operating the telegraph that lead to the catastrophe). Although the visual representation in the distributed EM railway model is unsophisticated in visual effect, the openness in interaction is different in character and far richer from what can be supplied by a VR scene of the accident where the number of scenarios and the construals of each scenario are preconceived.

The Virtual Electronic Laboratory
Illustrating the EM implications on Computer-based group learning

Motivation

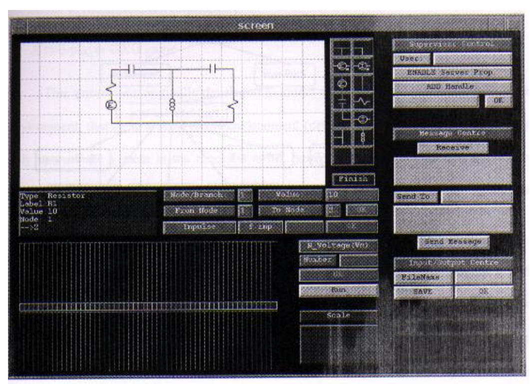


Figure 4.3 The Teacher's View in the VEL Model

Computer-based technology promises to play an important role in educating the future generation. However, no one can predict to what extent the computer will replace the teacher, the pencil and the book in modern schools and universities. Distance and group learning are emerging activities that tend to rely on computer-based technology and communication. These kinds of approaches to learning have a long history. There exist different learning theories that focus on observation and experimentation as an important aspect of learning. These include: constructivism, behaviourism, Piaget's Development Theory, Neuroscience, Brain-based learning, learning styles, multiple intelligences, right brain / left brain thinking, communities of practice, control theory, observational learning, Vygotsky and social cognition, problem-based learning, etc..

[17] The term 'provisional' indicates the possibility of change.

EM research [Roe99, COMICAL[18]] has directed special attention to providing a suitable computational framework that can support the implementation of different experience-based learning theories. Key EM concepts of observables, agency, dependency, the focus on state as experienced, the semantic relationship that is established between a computer-based model and its real world referent, and the use of definitive scripts in distributed modelling, give greater potential for EM to integrate the computer-based activity with the experiential group learning activity. EM computer-based artefacts reflect the modeller's mental model and his personal insight and understanding. This artefact is constructed based on past experience and can be amended in the light of newly gained experience and hence can support learning by experience.

About the model

The virtual electronic laboratory (VEL), developed by D'Ornellas [Dor98] and Sheth [She98], is a distributed model of an electrical laboratory used for educational purposes. The model offers similar features to those offered by other educational software for electronics (such as PSpice), and added value features in distributing the modelling activity and giving the teacher more control over the students' activity [19]. The participants in the model are the students and the teacher. The teacher acts as an external and internal agent who can create and shape the context of interaction between students and their views, and assign to students diverse agent roles and privileges to change the state of the model. The student's view is a restriction of the teacher's view. This enables the teacher to intervene as a super agent in the model. Interaction with the model is open ended. This is enabled through the visual interface associated with participants' views of the model (student or teacher) and / or the tkeden input window. Communication can be established both between students & teachers, and between students & students. Given the distributed client server configuration, student-student communication at the clients should pass via the server. Agency is assigned by the teacher to the student, to

[18] A project that aimed at introducing the concepts and practical use of EM tools to teachers in UK and to gain feedback on the potential use of Empirical Modelling models in schools in UK.

[19] Sheth (1998) identified four scenarios in a real life electronic laboratory: (1) the teacher and the lab assistant initially set up the equipment and the circuit on the circuit board containing all the required components. The students crowd around the table where the teacher is in control of the experiment. The teacher describes the circuit and its components to students and then asks students to change values of components for learning purposes; (2) each of the students either in-group of one or more create their own circuit on their own circuit board and monitor the outcome of their experiment. The teacher may assist students in building their own circuit. In this case the teacher does not have control over all the students who are working and can supervise only one group at a time; (3) the teacher may carry out the experiment on the computer and the students observe and ask questions. The teacher will demonstrate to students the effect of changing components values and the students learn from this interaction. This method is not based on a distributed architecture; (4) The teacher uses a software tool such as PSpice in aid of the experiment. Once this is done, the student will interact with the tool PSpice and make

observe (oracle) or change (handle) the values and parameters of the electronic circuit components and its design, using the LSD[20] notation for distributed communication [Sun99].

Conclusions drawn from the model

The model highlights added value that can be gained from using EM as a framework for computer-based learning. Specific features in the VEL model to which EM makes an essential contribution are:

- The extended teacher's computer -based control over the student activity in the distributed modelling environment. In various existing electronic lab scenarios the teacher has either full computer-based control over the modelling activity with little computer-based control and intervention from the student, or she/ he can have minor computer-based control over the student activity.
- The openness of the interaction with the distributed model. This is facilitated through the interface for communication, and the open-ended communication of definitive scripts that is radically different from traditional means of distributed modelling.
- The situatedness of the interaction with the distributed model. Interpersonal communication that can take place between the teacher and students is not preconceived. It reflects modeller's insights and experiential learning activity.

In general, EM technology can support diverse modes of interaction in computer-based interpersonal communication. Some built-in interaction modes are provided by the tools (normal, private, broadcast, and the interference modes). However, different configurations for communications and modes of interaction can be established, through using the LSD notation in dtkeden, to suit various needs of computer-based interpersonal interaction. In a group learning context, EM technology can potentially deliver computer-based support that *integrates* with, rather than *substitutes* for, the human role in the learning activity. This promises a greater role for the computer as a resource used by the teacher and the student in modern computer-based learning.

The Timetable
Illustrating the EM implications on strategic decision support

Motivation

Providing computer-based support for the human decision making activity is a challenging task. Various tools[21] and technologies[22] have been explored for this application. Despite the

---

observations and learn. The teacher moves round the classroom giving attention to each student for a short time. The teacher has no control over what the students are doing directly.

[20] This notation is similar in syntax to the one introduced in chapter 3, but can be used only in distributed Empirical Modelling to set the context of interaction for different communicating agents.

enormous advances in storage, speed, functionality, and interface design, the effectiveness of the computer-based decision support is relatively modest. This is mainly attributed to the challenges posed by the coherent integration of the human and automated activity (semi-automation) and supplying the computer-based support for the human cognitive activity.

EM technology seeks a computer-based framework that supports the continuous human engagement in the automated decision making activity, and the exploration of a software development paradigm that ensures a distinctive quality of interaction between the human and automated activity.

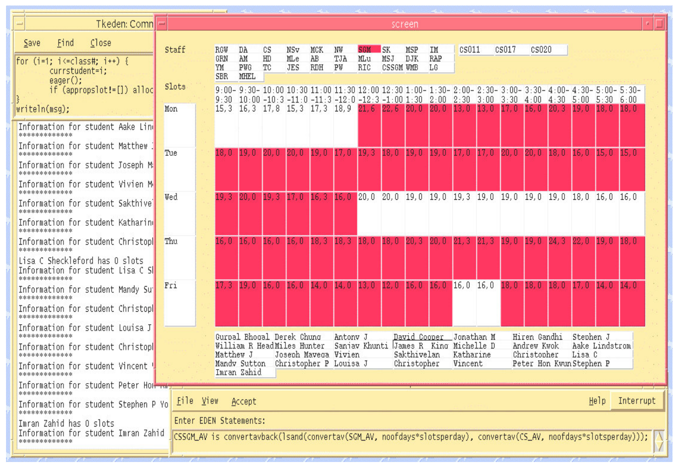


Figure 4.4 A snapshot of the Temposcope

About the model

A real timetabling application, developed by M. Beynon, S. Yung, S. Maad, A. Wong, and A. Ward is considered in [BRR00-1, BRR00-2, BWMWRR00] as a case study in computer-based decision support, the integration of human and automated activities, and the development of business solutions. It involves timetabling a week of oral presentations for final-year project students at Warwick. Each presentation requires a timeslot of 30 minutes duration between 9 am and 5 pm from Monday to Friday. The presentation is attended by the project supervisor, the second assessor, and a moderator. Staff are assigned to be moderators or assessors according to their suitability, availability, and workload criteria. The EM model developed is intended to provide semi-automated decision support for the human timetabler in recording and assessing the observables that are of crucial interest in developing the timetable. The model is similar to an instrument[23] in its use and development. The point of similarity to an instrument in its use stems from the scope for access and manipulation of *observables* of interest. The phases of the development of the timetabling instrument are: (1) an assembly phase where the component parts individually tested on artificial data sets are combined to create the first prototype; (2) a testing phase on real data generated from an old

---

[21] Spreadsheets, databases, expert systems, neural networks, intelligent agents

[22] e.g. artificial intelligence, hypermedia, networks, visual programming

[23] The term Temposcope (**T**imetabling **EM** **P**rojects **O**rals Instrument) is intended to suggest the use of the computer as an instrument.

timetable; and (3) a usage phase where the instrument is used in its intended mode (semi-automated or fully automated).

The semi-automated timetabling mode involves a high degree of interpretation and intervention from the human timetabler. Dependencies established between observables enable the semi-automation and the gradual construction of the timetable. A semantic relationship is established between the state of the model and its real counterpart (a working timetable satisfying all needs and constraints and accommodating unforeseen changes).

Supporting the decision making of the timetabler is much broader then deciding on the allocation of resources subject to well formulated constraints. Making human judgement, considering qualitative constraints, resolving singular situations are examples of timetabling activities that are hard to address with fully automated decision support.

Conclusions drawn from the model

Providing computer-based decision support relies on a high quality of integration of human judgement and automated processing in the decision making activity and a greater emphasis on the experiential aspect of a computer state. The open-ended engagement in the modelling activity and the semantic relationship between the state of the model and its real counterpart are distinctive qualities of model building in EM that better serve the decision support objective. The EM approach allows the entire timetabling activity to develop in a less constrained and less process-driven manner. It makes use of representations of state and agency that reflect the designer's construal and exploits these in both manual and automated aspects. Normally, the automation of the timetabling process exploits optimized representations of state that are difficult for the human interpreter to understand.

The Attribute Explorer
Illustrating the EM implications on visual exploration

Motivation

Real data with qualitative and imprecise characteristics are commonplace in many domains. Providing computer-based support for qualitatively richer exploration of data motivates a paradigm shift from conventional computer-based support for data analysis. This paradigm shift promotes:

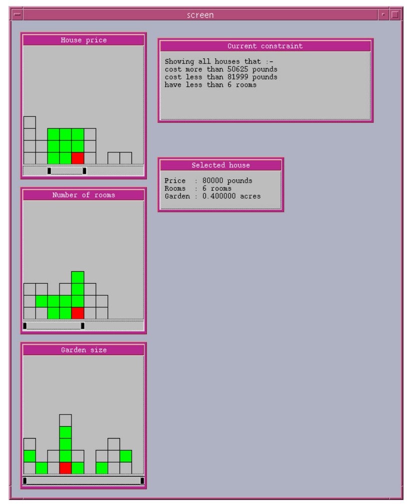
- visualisation in support of cognition (e.g. to complement abstract data processing);
- interactive exploration (e.g. to provide an alternative to preconceived search strategies subject to preconceived constraints);
- the discovery of hidden complex relationships between data (e.g. to complement computationally intensive analysis of tabulated data);

- the representation of qualitative relationships between data.

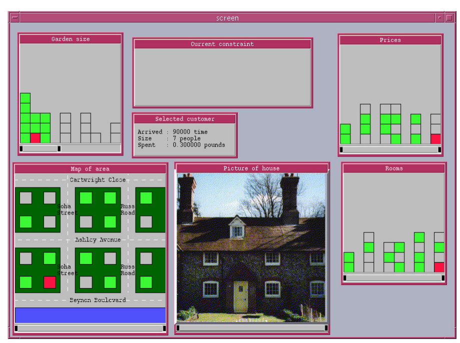
About the model

The EM Attribute Explorer, a data analysis model for definition driven interactive exploration of data is developed by C. Roe, firstly as an EM definitive script, then an appropriate visualisation is coupled to this script [Roe00]. The model is inspired from the Attribute Explorer (originally developed by Bob Spence, Imperial College London, and currently being further developed by him in collaboration with IBM Ease of Use Group). The model aims at offering an interface between the human and a data set. This interface acts as a lens to the data that can be looked at in different ways to cultivate our subjective understanding of the meaning of the data viewed. The definitive model can be customized to suit many different types of data (such as restaurant data, houses data, students marks, and weather records data). The first snapshot shows the visualisation of house data satisfying various constraints expressed in definitive scripts. The second snapshot shows the addition of non-numerical types (text fields, picture fields, map fields, etc.) to the visualisation of data about houses. This gives the modeller more immediate insight into the significance of the data in the real world.

Figure 4.5 The EM Attribute Explorer



Visualisation of houses data satisfying various constraints



Non-numerical types added to the interactive visual exploration

Conclusions drawn from the model

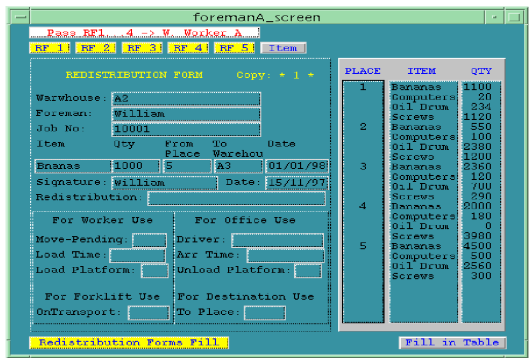
Conventional approaches to data analysis face many limitations. This motivates a paradigm shift in approaches for data analysis. Interactive visual exploration is a potential viable approach to data analysis that takes into account quantitative and qualitative features of data. EM principles, based on the concepts of observable dependency and agency for modelling personal subjective and provisional view of a domain, can potentially provide computer-based support to interactive visual exploration. The original Attribute Explorer is not a product of the EM group, but it clearly exploits dependency in a powerful manner. This has been demonstrated by C. Roe's re-implementation of the Attribute Explorer using a simple script. The Attribute Explorer epitomises one way in which EM concepts can potentially be exploited in visualization (other EM research along these lines is found in R. Cartwright's JAM and JAM2[24] tools and the Empirical World). There is a potential power in future use of the EM Attribute Explorer model. This stems from scope to link such exploration of data to pre-existing scripts.

The Warehouse
Illustrating the EM implications on participative BPM /BPR

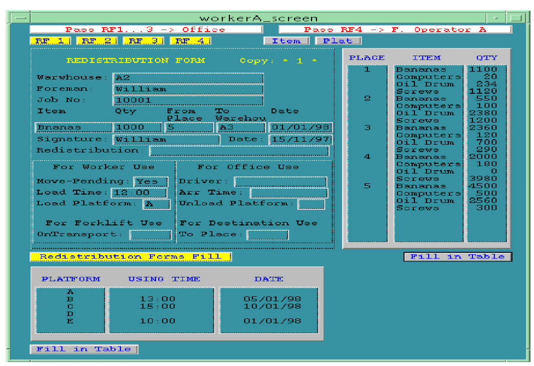
Motivation

Applying a computational framework to business process modelling is a challenging task. This is widely acknowledged in software system development and software engineering practices and is mainly due to the complexity of business processes and the difficulty of understanding these processes and expressing this understanding in a model [CRB00]. Despite the aspiration of business management to establish environments and routines that operate according to conventions and protocols as reliable as scientific laws, adopting abstract methodological approaches to BPM leads to major problems [BRR00-2]. These include:

Figure 4.6 Views of different Human Participants in the Warehouse ISM



Foreman Artefact

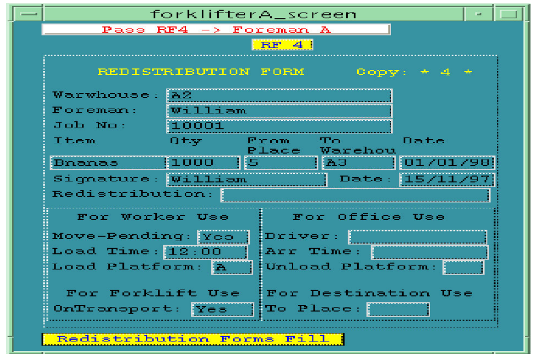


Warehouse Worker Artefact

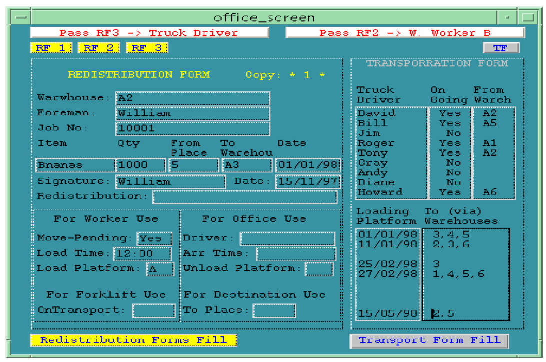
- enhancing the model incrementally in-line with further experiential knowledge of the business domain
- adding contextual information appropriate to different situations.
- supporting end-user participation in the modelling activity.

[24] /dcs/acad/wmb/public/projects/notations/JaM2API/DOCS/JaM2

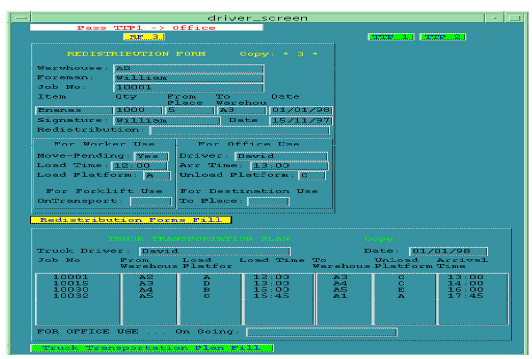
The limitations of abstract methodological approaches to modelling business processes motivate new modelling approaches that take into consideration the larger context of business processes (including the objectives of an organization, the viewpoints of people concerned in any particular process, the motives, the knowledge and expectations of users of systems being considered).



Forklift Operator Artefact



Office Artefact



Truck Driver Artefact

About the model

The warehouse[25] model [CRB00], developed by Y. C. Ch'en, is taken as a case study to illustrate the potential application of SPORE (Situated Process of Requirement Engineering) framework[26] in BPR. The case study was adopted by Jacobson et al (1992). The business process engineering involves introducing computer systems into the warehouse to offer automatic support to the storage and redistribution services. People in the warehouse include: the foreman responsible for the warehouse, the warehouse worker responsible for loading and unloading, the forklift operator who drives the forklift in the warehouse, the truck driver who drives a truck between different warehouses, the office

[25] The main function of a warehouse is to provide its customers with warehouse space. The operations of the warehouse include storing different kinds of items and using trucks to redistribute items.

[26] a problem oriented framework (based onn EM principles) in which requirements – viewed as solutions to the problems in the application domain - are developed in an open-ended situated manner [CRB00].

personnel who receive orders and requests from customers. An interactive situation model ISM is developed to analyse the business processes that operate in the warehouse application in order to re-engineer them. A key issue is to establish and maintain a semantic relationship between the ISM and its external referent. This involves explaining activities in terms of actual interaction amongst agents, then seeking to organize these into patterns that are reproducible and reliable. The modelling activity is based around patterns of information exchange (related to receiving, storing, and retrieving items from the warehouse) between warehouse personnel. Distributed modelling in developing the ISM makes it possible to separate the viewpoints of the agents in the model and to complement these with an external observer's interpretation. The above snapshots show the views of different human agents involved in the warehouse system.

Conclusions drawn from the model

Considering the wider context of business process modelling is a challenging task for conventional method-based modelling approaches. Empirical Modelling can potentially provide computer-based support for business process modelling in its wider context by adopting principles that aim at establishing a semantic relationship between the real world system and the corresponding computer-based artefact. EM technology emphasizes the human role in computer-based distributed business process modelling. Each participant in the business process modelling activity (human or human playing the role of devices) interacts with the distributed business process model in a situated manner through their own views and privileges for state changing actions. In the modelling activity, this reduces the burden of automating various roles of participants in a preconceived way, gives more room for personal insight, and can potentially support the detection of singular situations that might arise in the business activity when adopting a particular business process model.

4.3 Conclusion and Future Prospects of EM technology in Finance

This chapter has argued that EM technology can address the technical demands for the wider agenda for computing in finance by introducing a paradigm shift in the software system development activity. The EM approach to SSD takes into account the experiential, the situated, and the human centred aspect of this activity. In addressing the technical demands for the wider agenda for computing in finance, EM technology seeks a suitable framework for deploying current technologies to support the needed paradigm shift at the computational level. EM technology proposes new foundations for artificial intelligence, the support of the application of virtual reality technology in different domains (graphics, and social domains), and proposes a new view for database technology.

This chapter has overviewed different case studies that show the added value from using EM technology in addressing the wider agenda for computing. It asserts the claim of the potential prospect of EM technology in supporting the following shifts in perspective:

In software system development practices

- from programming to modelling [Yun92]
- from a methodological approach to an amethodological approach to software system development [Sun99]
- from a closed to an open development model for software systems [Sun99]
- from a closed-boundary system development to dynamic-boundary system development [RCR00]
- from independent designer, developer, user activity to collaborative designer, developer, user activity

In the modelling activity

- from formal specification of behaviour to experiential representation of state.
- from separation between the experience of the real world and of the model to establishing a semantic relationship between the computer-based model and its real world referent [Sun99, BRR00-1, BRR00-2]
- from modelling preconceived activities to modelling situated activities [Sun99]
- from knowledge representation to knowledge construction
- from individual modelling to distributed modelling [Sun99]

In computer uses

- from modelling products in isolation to modelling products in a situated context [FB01, FB01-2]
- from using the computer as a tool to using the computer as an instrument

In computer-based support to various activities in the real world

- from business processes specified by an abstract pattern to business processes developed from a situated activity [BRR00-2]
- from a “method-tool-user” culture to a “human computing” culture where human and computer agencies are more closely integrated [BRR00-2].
- from a methodological approach to decision support (based on problem identification, development of alternative solutions, selection of a solution), to an experience based approach to decision support (based on situated problem solving) [CRB00], [BRR00-3].

EM seeks to provide an approach that enables a human agent to engage in situated activity and aspires at using the computer as an instrument rather than a tool. A situation is construed in terms of the concepts of: observables, dependency, agency, and agent.

EM tools are still basically research tools, they illustrate the principles of the technology but do not have the robustness and consistency required for a commercial product [RCR00]. However, this doesn't undermine the potential of the EM tools to test the foundations and practical applications in various domains.

In the author's opinion, Empirical Modelling technology will be seen in the future more focused towards domains where:

- Group social activity is important. Examples include collaboration in engineering design; user/designer/developer collaboration in the software system development process in all its stages; b2b (business to business) collaboration; collaborative product design in manufacturing; group decision support; participative business process modelling and re-engineering; and collaborative financial market modelling.
- The human role is central. Empirical Modelling technology will find greater support to applications in the area of social science studies where the human role is central in guiding the analysis exercising control over the system state.
- Knowledge is weakly structured: this is typical in the early stages of exploring a new field of study or initiating a system development process.
- The machine cannot replace human thinking and human machine integration is more appropriate.
- Novel uses of the computer are demanded.

The paradigm shift supported by EM and the distinctive qualities of model building in EM motivates the exploration of the application of EM technology in the finance domain at the institutional, market, and investment levels. This is considered in subsequent chapters with reference to selected case studies.