Empirical Modelling for Educational Technology

W. M. Beynon
Department of Computer Science
University of Warwick
Coventry CV4 7AL, UK

Abstract

The potential merits of Empirical Modelling (EM) as a new approach to generating software for educational use are examined. Connections between EM and the learning process are discussed. EM is proposed as a method of dealing with knowledge that is gained through experience of interaction with artefacts. The philosophical implications of this thesis are considered, with particular reference to the work of William James (1842-1910). The relationship between EM and current research on multi-user spreadsheets and agent-oriented modelling is also described. A practical agenda for future applications of EM to education is outlined.

1 Introduction

This paper outlines the characteristic principles and techniques of a new approach to computer-based modelling known as Empirical Modelling (EM), and discusses its potential impact upon the technological infrastructure for education. EM addresses computer use in a wide variety of applications, and is relevant to education in many aspects (e.g. the construction of computer programs for educational use [16]; the adaptation of educational programs for different hardware and interface requirements [16, 20]; the simulation of classroom interaction), at many stages (e.g. in primary, secondary and higher education) and in many different contexts (e.g. for special needs [12], for craft-based disciplines, and potentially as a new foundation for Computer Science [18]). The paper will focus on putting EM in context with reference to what has been learnt from an extensive programme of wide-ranging practical case-studies. For more technical information, and for access to many additional references and screen shots of demonstration models, the reader may consult the EM website at http://www.dcs.warwick.ac.uk/pub/research/modelling.

This paper has 4 main sections. Section 2 identifies some of the key issues for educational software development; Section 3 introduces the general concept of EM, and outlines the approach with reference to case-studies linked to educational concerns; Section 4 examines the relationship between EM and contemporary research on multi-user spreadsheets and on agent-oriented modelling; Section 5 explores the philosophical background to EM with particular reference to the 'philosophic attitude' of Radical Empiricism [32] expounded by the American philosopher William James.

2 Context

2.1 Issues for Technology in Education

Current approaches to providing a technological infrastructure for education are beset by a number of problems. These can be viewed from several different perspectives:

- **the IT Management perspective** Most educational establishments have to employ some staff with a specialist knowledge of computer-based technologies. Outside higher education, the function of IT management is often supplied by one of the teaching staff who combines a hobbyist interest in computing with basic programming and operating system skills. Ongoing problems arise from the way in which commercial software packages evolve to demand new resources, and the speed with which hardware platforms reach obsolescence. In practice, it is hard to adapt existing software to run on new platforms, so that there is a high degree of dependence upon software and hardware suppliers. Consistent use of IT components in the teaching process is then expensive to resource in human and technological terms. Where such resources are unavailable, existing IT provision is compromised.

- **the Teacher perspective** Good teachers tend to customise resources such as textbooks, visual aids and IT support to suit their personal style and the demands of the curriculum. Where appropriate, they also tailor the use of these resources to suit the specific needs of pupils. Where IT is concerned, teachers wish to be able to carry out such customisation without specialist knowledge of computer software and systems, but this is generally impossible in practice. Even the minor revision of a software package frequently exposes a need for technical knowledge of programs and computer systems, often amounting to a degree of detailed insight that only the original developer could supply. Given the subject-oriented demands upon a teacher's time, this problem can only be resolved if it is possible to gain access to an intelligible incremental development process.

- **the Pupil perspective** To the imaginative pupil, IT products for educational use often appear too limited in scope. They operate with closed-world models and stereotyped modes of interaction that don't fully engage the imagination, and inhibit creativity and discovery through lack of openness and flexibility.

The limitations of current technology identified in these views are symptomatic of fundamental problems concerning computer-based system development in a wide range of applications. They have particular relevance in an educational context, since exploration and experiment have an essential role in the learning process in motivating pupils and engaging their imagination. EM has been concerned with several challenging issues bearing on the solution of these problems:

- developing system models in ways that allow flexible adaptation, extension and re-use even by users who aren't computer specialists, where models are evolved step-by-step (cf. F P Brooks' injunction that software should be grown [24]);

- developing techniques that allow machine-independent specification of software, and make it possible to generate implementations for particular architectures automatically or at least semi-automatically;

- using computers in ways that address the needs of the artist / craftsman, liberating creativity by putting the emphasis upon the creation of artefacts that exploit computer-based technology, rather than on animating the products of analysis and reflection in stereotyped ways;

- supporting an open-development rather than a closed-world engineering culture [23]: devising ways of interacting with computer-based technology that afford learning experiences that cannot be entirely preconceived (cf. the closely circumscribed knowledge about the solar system that can be learnt from an orrery with what can be learnt about flying from a high-quality flight simulator).

In this context, there is an apparent conflation of concerns: addressing fundamental issues of IT system development; and finding the most effective ways to exploit IT in education. The paradigm shift for computer-based modelling proposed in this paper addresses both issues, as it is itself intimately connected with the learning process.

2.2 Artefacts and the Learning Process

There is a tendency in educational philosophy to identify learning closely with the development of communication, language and background cultural experiences. This is countered by a recognition of the diverse types of skill and knowledge that contribute to intelligence [29], and of the role that interaction with artefacts and non-verbal interaction can play in learning and psychotherapy. In [4], for instance, Axline — a psychotherapist celebrated for her use of play therapy with children — observes that conventional theories of learning fail to account for the level of sophistication in reading, writing, spelling and drawing that can be acquired by a child who has withdrawn from communication. A related scepticism is expressed by Gooding, who identifies the need to redress the theoretical, linguistic emphasis in philosophy of science [31]. The proposals in this paper reflect this need.

The following brief synopsis gives the perspective on learning that informs this paper (for later ease of reference, this is summarised in Box 1). Learning begins with private experience, and with interactions that disclose persistent features, contexts and objects. It involves the correlation of experiences of different artefacts, and the acquisition of skills in their manipulation. Interaction reveals the extent to which change depends upon our actions, so disclosing the scope of personal agency, and the presence of independent agencies. Certain associations of features become identified with particular kinds of agency, and with generic patterns of interaction and stimulus-response. Communication with other agents evolves from pre-articulate interaction in a common environment, and from phenomenological uses of language, where utterances directly express aspects of the perceived current state. Such interaction provides the empirical basis for common experience, and the concept of objective knowledge. These in turn inform symbolic representations, public conventions for interpretation and the use of formal languages.

private experience / empirical / experiential

interaction with artefacts: identification of persistent features and contexts
practical knowledge: correlations between artefacts, acquisition of skills
identification of dependencies and postulation of independent agency
identification of generic patterns of interaction and stimulus-response mechanisms
non-verbal communication through interaction in a common environment
phenomenological uses of language
identification of common experience and objective knowledge
symbolic representations and formal languages: public conventions for interpretation

public knowledge / theoretical / formal

Box 1: An Empiricist Perspective on Learning

Box 1 is to be interpreted as delineating the empirical elements that inform personal learning. These range from private and primitive agency to sophisticated interactions with other individuals and artefacts in a common environment. The content of Box 1 has been informed by an approach to computer-based modelling to be described below. It serves as a frame within which to elaborate on a pragmatic and empiricist thesis such as is adopted by James ([32] p202): "the 'truth' of our mental operations must always be an intra-experiential affair".

Box 1 gives significant emphasis to private experiences that involve interaction with physical artefacts. Viewed as a progression of learning activities, it is to be distinguished from a typical instance of learning as encountered in education, where language plays a major role. The account of learning explored in this paper acknowledges the significance of language in communicating and representing knowledge, but aims to put this in its proper perspective.

The idea that *all* learning and knowledge is fundamentally mediated by language is so well-established that the concept of pre-articulate learning is often viewed with suspicion. The cue for such a philosophical position is to be found in Wittgenstein's words [45]: "a nothing would do as well as something about which nothing can be said". The contention of this paper is that appropriate use of computer-based technology liberates the design and construction of artefacts. In particular, suitably constructed artefacts can be used effectively in the systematic investigation of pre-articulate experience.

It cannot be denied that the predominant focus of education has been on communicating knowledge at the public, theoretical and formal end of the learning spectrum, where language is most effectively used. The main impact of the computer on education hitherto has been to reinforce the emphasis upon knowledge as centred upon language. The theory of computation legitimises the idea of the computer as executing sequences of operations that follow a recipe that can be formally specified. The role of human perception and interpretation in accessing the meaning of the executing program is not addressed in the classical perspective on computation. In conventional approaches to computer-based modelling, there is a sharp contrast between the formal treatment of computation and the informality surrounding cognitive issues concerning presentation and visualisation of results.

However informal the status of pre-articulate interaction and subjective knowledge in the learning process, their significance is evident in many areas. Sketches, scale models and physical prototypes have a vital role in practical engineering, especially in connection with the representation and communication of as yet uncertain and provisional knowledge. Gooding [31] highlights the use of similar representation techniques in Faraday's experimental work. Informal models of this type inform what Feynman calls the essential non-mathematical complement to a theory-based perspective on physics: "A physical understanding is a completely unmathematical, imprecise, and inexact thing, but absolutely necessary for a physicist." ([27] §2-1).

Other relevant illustrations include: the importance in historical accounts of presenting multiple viewpoints that subvert an objective reality; the usefulness of being able to assess the correctness of a calculation from many viewpoints; the experiential aspect of proof presentation and narrative; the fundamental role of non-verbal interaction in teaching musical instruments, crafts or sports; the importance of exploiting personal characteristics and potential when motivating pupils and students.

In so far as such components of our learning spectrum defy linguistic representation, and admit no formal specification, it could be argued that they are 'as nothing'. An alternative stance is that only now is the technology available to demonstrate the true significance and potential for representation and communication via constructed models. What is required to complement this technology is a philosophical account that can counter the reductionist view of the computer as merely a language processor. The topicality of this concept is evident from many recent publications: West's account of hermeneutic computing [44]; Naur's critique of logic and rules for knowledge representation [38]; R Brooks' demonstrations of the power of computer-based technology that does not rely upon symbolic representation and automated reasoning [25, 26]; B C Smith's assaults on the logicist foundations of AI [41]. EM is intended as a foundation for the wider view of computing commended by these authors.

3 Empirical Modelling

The above discussion motivates the development of an approach to computer-based modelling that is based upon a thesis about learning and cognition broad enough to embrace the role of the computer as an artefact and an instrument. Such an approach:

- is potentially good for applications in education because the principles of model construction are bound up with the learning process.

- can be the basis of a successful approach to computer-based model construction precisely because it establishes a direct link between the conception of the model and its construction;

Making allowance for the limitations of our current state of understanding and tool development [3, 30], EM is a promising candidate for a suitable approach. The evidence for this lies in the practical modelling tools that have been successfully developed (notably the tkeden interpreter), and the large body of associated practical case-studies that range over many application areas. To redeem the present limitations of these tools so that they can meet the system development challenges identified in this paper fully remains a major task, and is the subject of several ongoing research projects [2, 3]. For clarity, this paper adopts an idealised view of EM.

3.1 Principles of Empirical Modelling

EM is predicated on the thesis that cognition and learning are fundamentally concerned with a process of construing phenomena in terms of agency and dependency. The basis for this process of construal is empirical and pragmatic. It can lead to explanation of phenomena that presume such conviction about the reliability of stimulus-response patterns that they can be expressed using formal mathematical models. More typically, an explanation is provisional and tentative, and there can be such scepticism about its validity that the only possible representation is via a physical artefact with which the modeller interacts in an open exploratory fashion. The perspective on learning associated with EM is as described in Box 1, and the activities associated with learning are embodied in the process of model construction. EM is conceived as a collection of fundamental principles and techniques to trace and record comprehension through stages similar to those outlined in Box 1, as set out in [14].

EM aims to lay a principled foundation for non-linguistic, pre-articulate representations beyond the scope of symbolic AI. This involves the use of the computer to create physical artefacts that explicitly imitate phenomena in a manner that is similar to that involved in the development of a scientific instrument or an engineering prototype [39]. The semantics of such a physical artefact is defined by interaction, rather than circumscription, and so lends itself to open-ended use and interpretation that is not necessarily preconceived prior to its construction.

The central concept of EM is to elaborate a model of a phenomenon with reference to a projected causal account. Conceptually, this elaboration relies upon carrying out experiments in parallel upon the phenomenon and upon the model, and correlating the results. The stages in the elaboration are: the identification of observables, and of patterns of correlated change to observables ('indivisible relationships'); the identification of agents as the instigators of state-change; the classification of observables with respect to each agent, associated with identification of those observables presumed to account for stimulus-response patterns in their interaction; the construction of behavioural models for systems, whose animation is in general partially under the explicit control of the modeller.

The quality of the model developed in this manner is determined by the degree of insight into a phenomenon reflected in the modeller's explanatory framework, and by the pragmatic value of this explanatory framework in the context of the particular intended application of the model. The modelling activity proceeds in such a way that the explanatory framework and possibly the intended application evolve with the model. Empirical considerations govern this process. In every respect, the change of perspective associated with the 'experimental paradox' accounts for changes in the status, interpretation and classification of observables. That is to say, the very same activities that at an early stage in the modelling process are aimed at identifying what dependencies and agencies govern observables may subsequently be regarded as confirming that such relationships are indeed valid.

A common reaction to the concept of 'empirical modelling' is: *How does it differ from common-or-garden modelling? Isn't all modelling empirical by definition?* There are three different aspects that help to distinguish EM from conventional modelling: respectively philosophical, conceptual and technical in nature:

- EM addresses the concept of the computer as physical artefact and instrument;
- it involves recording the experimental contexts that inform the model;
- it exploits novel computational abstractions.

The idea of programming the computer as an artefact or instrument is itself controversial. The experiences that a user gains in interacting with the computer are not explicitly specified by formal programs, and topics such as human-computer interaction, interactive graphics and multi-media have an obscure status in the science of computer use. One application for computers is to provide visualisation and animation to illustrate a formally specified algorithmic process, or behaviour. In this context, the correspondence between what the user sees and what the computer calculates is precisely circumscribed, in that its interpretation is preconceived. The idea that all meaning must be mediated through preconception in this way is deeply embedded in philosophical prejudices concerning the primacy of language. Even Naur, despite his reservations about the limitations of logic for knowledge representation, seems to adopt this perspective on constructed models (see [38] §4.4: The Metaphysics of Constructed Models).

EM demands the rejection of this reductionist view of the potential of the computer as an instrument. This counterview is sustained by the evidence of experience. It is self-evident that the design sketch communicates information in an open-ended non-verbal manner; that Faraday developed his understanding of physical principles with reference to artefacts he constructed and explored; that a musical instrument is more than a piece of wood in the hands of a virtuoso. To specify the construction of a musical instrument sufficiently precisely to enable it to be manufactured as a physical artefact that permits certain basic modes of interaction is not to specify it as an instrument. Variations between instruments defy formal specification. How the perception and imagination of the musician interacts with the instrument can't be specified. The content of an artefact is not captured by a protocol for interaction with a formally specified mechanism. The interaction is open-ended, and unconstrained; it admits imaginative interpretation, influenced by the development of new skills and insights. The physical object is open to new modes of observation and investigation, and transcends any particular abstract repertoire of preconceived transformations and interpretations.

It is clear that only a valid claim to access to fundamental cognitive principles can legitimise an approach to modelling that purports to assist the construction of artefacts. Some justification is needed for supposing that Empirical Modelling is not merely empirical in the informal conventional sense that model development and refinement can be carried out in an iterative, or even trial-and-error, manner. The distinction between one view of empiricism and another is most starkly illustrated by considering how we should view an engineer who set out to build a bridge by constructing it and seeing whether or not it fell down. The engineering knowledge that informs bridge building is of course empirical in essence, but it is implicitly organised in a very highly structured manner into theories and particular facts derived from experience and precedents. All sorts of learning about bridge building is also relevant, representative of the entire spectrum of activities in Box 1.

The nature and structure of the models generated by EM is crucially significant here. In EM, the most appropriate ways of representing and organising models are not prescribed. A model comprises observables, dependencies and agents that are loosely assembled in such a way that they can be associated in several different ways to reflect different viewpoints and components. Effective documentation of such a model entails giving access to several different associations of model fragments. Which associations are most appropriate in the presentation of the model is a matter of judgement, but they will include direct representation of significant experimental contexts encountered in the evolution of the model, as experienced from several different viewpoints. In any case, each person who wishes to interpret the model is freely able to select and study particular associations. In contrast to interaction within a closed-world, such as a conventional computer model supplies, EM offers situated modelling with a potential for open interaction, that invokes creative observation and interpretation of the situation to which the model refers.

EM exploits two principal technical innovations in representing models as they evolve. Definitive (definition-based) representation of state is used to record dependencies between observables [16]. The results of observation-oriented and agent-oriented analysis are recorded in the special-purpose notation LSD. The models that are constructed are qualitatively different from conventional computer programs in something like the same way that an interconnected network of multi-user spreadsheets differs from an interactive calculator. In interacting with such models, there are no fixed entry points at which redefinitions can be made. New definitions can be introduced both to reflect new external observations and insights, and to refine the modelling instrument. Through such interaction, the provisional extensible informal LSD descriptions for agents can also be refined. It is in this fashion that EM respects the open-ended character of the artefact, so that the shaping of the computer model, together with its internal and external interactions, accompanies the analysis of the phenomena to which it refers.

The appropriate computational framework for EM is that provided by the Abstract Definitive Machine (ADM) [19]. Observables, dependencies and agents can be represented in the ADM in such a way that the modeller can interact with the model in the role of a superagent. Where the observed behaviour of a system is sufficiently well understood in stimulus-response terms, it is possible to incorporate particular behavioural patterns into the ADM model, but the applications of primary interest for EM are those that focus upon the early stages of the learning spectrum in Box 1. For such applications, the behaviour of the model is predominantly latent, and has to be defined and discovered through exploratory interaction.

The models that can be elaborated in this way are exceedingly diverse, according to the extent to which dependency and agency are emphasised. For instance, in an exercise in mathematical visualisation known as the lines demo, the entire model consists solely of a script comprising several hundred dependency relations [21], whilst, in an exercise in simulating classroom interaction, the main emphasis is upon constructing observation-oriented models of teacher and pupil agents.

3.2 The Empirical Modelling Process

Approaches to computer-based modelling are often described as methods. This terminology suggests a systematic procedure that is guaranteed to achieve a particular goal. It is questionable whether learning admits such a characterisation. There are elements of serendipity and discovery in creative learning, and no absolute assurance that preconceived patterns of action will achieve specified goals. The status of EM as a method may be likened to that of the so-called 'Scientific Method'. Though the pursuit of scientific knowledge follows certain general principles and empirical techniques, novel science is in no sense a routine exercise. It is an activity that takes place in an arena in which neither human imagination nor the recalcitrance of physical phenomena can be ignored.

Many aspects of the learning process (as set out in Box 1) are represented in the diverse applications of EM that have been developed in numerous case-studies. To some extent, the shifting focus of the Empirical Modelling Project, as it has been developed over more than a decade, has traced the stages of the learning process from the realm of private experience to that of public knowledge. The case-studies that have been developed over this period can be conveniently classified in this way.

3.2.1 EM for 1-agent Systems

The first focus of interest in EM is in what can be regarded as '1-agent systems'. To some extent, using a spreadsheet illustrates agency in a 1-agent system. The semantically interesting state — corresponding to the external state of the world of which the spreadsheet is a model — resides in the state of the spreadsheet. The only changes to the state are via actions on the part of the user. The nature of these actions is conditioned by the external semantics, and in general cannot be circumscribed as are the actions upon variables within a conventional computer program. In as much as neither their actions nor their interpretations can be anticipated, the user of the spreadsheet is a free agent.

Interaction with a spreadsheet is not in itself an archetypal example of a 1-agent system in EM terms. To conform to the semantic framework associated with EM, the emphasis in interpreting the spreadsheet has to be on what is experienced by the user, not on the abstract information encoded in the cells. This emphasis can be imputed to ordinary spreadsheet use, to the extent that a user may be able to directly apprehend the meaning of numerical information presented in tabular form, but the essential spirit of EM is better represented where there is an explicit experiential aspect to the model. This function can be served, for instance, by visualisation of the spreadsheet data. In an EM context, the variables that appear in definitive scripts typically have such an experiential significance — they may refer directly to entities visible to the computer user, such as points, lines, geometric attributes or windows on the screen for instance.

The experiential character of 1-agent models in EM has crucial significance in respect of the relationship between the computer model and its referent. In conventional symbolic representation, there is typically no sense in which the formal symbol resembles its referent. That is to say, a formal symbol offers no experience to the interpreter that can be recognised as associated with experience of the independent entity that it references. In contrast, the focus on observables and indivisible dependencies between observables in EM, when combined with mechanisms that make these directly perceptible in interaction, is the means to ensuring that what is observed of the artefact (albeit as in caricature rather than as in realism) conforms to what is observed of its referent. The fact that 'observation' can be interpreted literally in this context is fundamental when accounting for the perceived relationship between a feature of a phenomenon and its counterpart in the computer model (cf. the most primitive aspects of the learning process in Box 1). In elaborating the principles of EM, the generalisation to which the term 'observation' is subject is limited solely by what can be deemed to be directly apprehended by some observing agent (cf. §4.2 below).

The Empirical Modelling Project originated with the study of definitive notations for interactive graphics that apply the spreadsheet principle to the manipulation of visible states [5, 6]. Precedents for this kind of use are to be found in the research of the brothers B and G Wyvill [47]. 1-agent interaction of this nature has clear affinities with the kind of basic experimental interaction that a scientist or an engineer favours when trying to understand a complex phenomena. It also has some of the subjective exploratory character of the researches in which Faraday seeks to find means for the metaphorical representation of observables (such as electric currents and magnetic fields), and the indivisible relationships that link them together (cf. the relationship between the polarity of a magnetic field and the direction of a current).

Even in the context of interactive graphics, a range of interpretations for the EM process is evident. A relationship between observables may reflect a perceived connection between the state of two external features, such as the position of the lock of a door in relation to the hinge; it may serve to assist the designer to redesign an artefact in a conceptually convenient way (cf. parametric design); it may express a personal penchant on the part of the designer for a particular style of layout [10]. This is a simple illustration of the way in which the exploratory interactions of EM are associated with analysing external phenomena, refining the instrument by means of which such phenomena are to be represented, and identifying the personal protocols that are characteristic of the designer agent.

In 1-agent systems, it is possible to address private rather than objective experience. There need be no independent external observer, and the designer's model can be a product of fantasy. A discrepancy between the state of the world as is and as imagined is commonplace in 'what if?' use of a spreadsheet. The LSD description for the designer incorporates indivisible relationships between observables as they are conceived to be, capturing the idea of artefact as defined in its relation to interaction.

3.2.2 EM for Multi-agent Systems

More general application of EM principles involves postulating additional agency. The need for such agency manifests itself through an inability to attribute all state change involved in interaction with an environment to the intervention of the modeller. Since the effect of an action on the part of the modeller is mediated through dependencies between observables, agency can be viewed as complementary to dependency. Because of the nature of agency as represented in 1-agent systems, the most appropriate way to conceive other agents is as having the same general characteristics as the modeller. This is evidently quite appropriate when accounting for the actions of other human agents, discounting the potentially subjective nature of interactions. For inanimate agents, the stimuli and responses typically involve observables that cannot be directly sensed and manipulated by a human agent. Knowledge about the protocols for interaction of such agents has then to be represented in ways that are intelligible to a human agent. In EM, this is done by constructing artefacts that purport to imitate protocols in terms that are sensible to human perception and intelligence.

In this discussion, there is some prestidigitation over the status of observations. In a 1-agent system, the human agent typically acts in two roles: as the sole significant agent, and as the external observer. Where there are several agents, these can be configured in many different ways with respect to an external observer, or even to a choice of different observers. It is in connection with systems with more than one agent or observer that LSD descriptions become significant. An LSD description is a classification of observables from the perspective of an observer, detailing the observables whose values can act as stimuli for an agent (its *oracles*), which can be redefined by the agent in its responses (its *handles*), those observables whose existence is intrinsically associated with the agent (its *states*) and those indivisible relationships between observables that are characteristic of the interface between the agent and its environment (its *derivates*). The repertoire of possible state-changing actions of agents is also recorded (its *protocol*).

There are essentially two different ways in which EM can be applied to modelling concurrent systems:

Scenario 1 the modelling activity is centred around an external observer who can examine the system behaviour, but has to identify the component agents and infer or construct profiles for their interaction;

Scenario 2 the system can be observed from the perspectives of its component agents, but an objective viewpoint or mode of observation to account for the corporate effect of their interaction is to be identified.

Scenario 1 is the perspective appropriate for the experimental scientist, who has to speculate about the kind of agents and interactions that account for an observed phenomenon. It is also the appropriate framework within which an engineer works when trying to devise components and protocols to fulfil a known function.

Scenario 2 is the setting for the analysis of a complex system, where the behaviour of components can be observed, but the emergent behaviour is yet to be inferred. It is also the appropriate setting for co-operative working and concurrent engineering, where the viewpoints of the agents can be specified in isolation, but their corporate interaction is potentially subject to lead to conflict and incoherence.

In many applications, it is appropriate to consider both scenarios concurrently, with a view to reconciling global and local perspectives on the behaviour of a system. For instance, it may be that the profiles of the components of a reactive system and the intended system behaviour can be specified independently, but these two perspectives have yet to be reconciled.

Related research in the Empirical Modelling Project has involved the practical development of the ADM and the tkeden interpreter, and their application to issues such as programming-as-modelling [16], concurrent systems simulation [15], concurrent engineering [8, 1] and explanatory modelling [7]. The characteristic themes of this research are *circumscribing behaviour* and *establishing an objective viewpoint*; these are associated with the middle region of the spectrum of learning activities in Box 1. The overall progression associated with EM in such applications is: acquisition of personal conviction, recognition of generalisation, identification of objective knowledge. The potential for tracing this development process to its logical conclusion is illustrated in research into machine architectures for EM [3, 19], the translation of EM models into conventional procedural programs [3], and the construction of comprehensive knowledge representation models for specific applications.

3.3 EM in Practice

Almost all the models that have been developed using EM principles have been represented using the tkeden interpreter. This applies even to those that are constructed using the ADM [19], since this is at present implemented via a translator that acts as a front-end to tkeden. As outlined in the account of EM given above, the development of the tkeden model as a computer-based artefact is accompanied by the identification and analysis of agency as it is manifest in many diverse aspects. For instance, the modeller may address their own role as an agent, or be concerned with how the constituent agents are construed to interact, both in relation to the artefact itself and to the phenomenon to which it refers. This role cannot in general be formally specified in a conventional sense, but it may be explicitly described in LSD. Such a description is framed in terms of observables, not abstract mathematical variables, and evolves as the artefact is constructed. In practice, most model development using tkeden proceeds in such a way that the roles of agents are implicit in potential interaction with the model, in much the same way that user interaction with a spreadsheet is implicitly constrained.

A typical tkeden file comprises three kinds of construct: definitions, functions and actions. Definitions are formulated in terms of variables that represent scalar quantities, text strings, and recursive non-homogeneous lists, as well as visually significant elements such as points, lines, and shapes in the form of planar line drawings, and windows in the screen layout. Functions serve as user-defined operators on the RHS of definitions; these supplement standard built-in operators that are used to define scalar, structural and geometric relations. In the present interpreter, functions are specified procedurally, but there is a good rationale for specifying them by auxiliary definitive scripts. This is necessary when specifying operators that return a complex structure, for instance, where reference may need to be made to the components of that structure in definitions [6]. Actions are specified as procedures that are triggered by changes to the values of particular variables. Ideally, actions should take the form of redefinitions of variables, but in practice it is essential to permit actions that act as conventional procedures with side-effects, as this is the only way to express state-changes in a conventional architecture.

In the interpretation of a tkeden file, the definitive script represents the state that is interpreted by the modeller. In practice, there are many ways to conceive and interpret the script. For instance, a script can be organised into subscripts that represent stages in the development of the whole model, or into objects within the domain being modelled, or into viewpoints for the agents within that domain. In combination with sets of actions, scripts can also represent particular agents whose mode of observation and interaction may or may not be explicitly described in LSD. The user interface to tkeden includes features for script storage, retrieval and inspection to assist script organisation and development. The use of the ADM serves a similar function [15], as do higher-order definitions [30].

Whereas the focus of conventional computer programming methods is upon the representation of behaviours, EM is essentially concerned with state and patterns of observation of state. A typical strategy in model development is for the modeller to first simulate the role of the agents within the application, then to introduce actions to automate their interaction as and when appropriate. This process of automation is conceptually quite different from the circumscription of a behaviour, since it stems from discretion on the part of the modeller. Even when a model includes objects whose motion is synchronised with the observation of a clock using Newtonian mechanics, the sequence of states that is generated is to be interpreted as *observed* rather than *constrained* to conform to a pattern. This distinction is significant only because of the exceptional agency that the modeller can invoke — potentially intervening as a super-agent at any point to change the current state in ways that are quite unconstrained.

Model development in EM has an open flexible character. This reflects the freedom with which families of definitions, functions and actions can be selected and combined in meaningful ways. Since the interpretation of such a family is independent of order, the sequence in which its components are introduced is ultimately insignificant. The construction process builds upon incomplete models in an incremental fashion. At any stage, particular subfamilies can serve as experimental environments in which to test the validity of hypotheses about the structure and properties of the model.

The derivation of one model from another can follow many different patterns. It is common for a model to be extended through the introduction of extra definitions that reflect a new mode of observation of the same referent (e.g. the introduction of the connectivity graph to the railway track model in [8]). Wherever it is possible to conceive two models as simultaneously observed, they can readily be combined, possibly subject to introducing additional dependencies between the two sets of observations (see e.g. the adjunction of a chess clock to the model of a statechart and digital watch in [11]). Specialisation of a model through adaptation of particular definitions is also possible. These techniques lead to a type of re-use that is more radical than that associated with object-oriented modelling. In effect, the open-ended character of observation in EM means that an object can be invested with extra significance. Abstraction of this kind is more faithful to life, where objects are not stamped with a specific functionality, but can be organised and adapted in ingenious ways [42].

EM does not dictate a particular development strategy. The non-prescriptive exploratory character of the modelling process lends itself to adaptation in a way that can lead to unexpected results. For instance, in a recent student project, a graphical tool that had been developed with the generation and animation of statecharts in mind could at once be used in conjunction with existing models. A student who embarked on modelling an adventure game incidentally constructed a two-dimensional shape designer within one and the same environment. It is also possible to work from a preconceived design for a model, and to introduce the components of the model in whatever order seems most appropriate. For instance, a preliminary model can be developed from an LSD description that is animated via the ADM [15], and a user-interface can either be constructed first (cf. a Visual Basic approach), or developed on top of an pre-existing system model (cf. [9]). Co-operative development has been illustrated in a number of projects. The most ambitious experiment of this nature was the construction of an animation of the game of cricket by a team of ten graduate students. Such experiments have demonstrated the potential of EM for rapid prototyping.

The distinctive open-ended character of the products of EM is most clearly illustrated in research into their translation to conventional programs [3]. The advantage of making such translations is that it leads to portability and much more efficient model execution. The only and essential limitation is that the behaviour of the model must first be constrained by restricting the protocols of all its agents. In effect, circumscription is a pre-requisite for formal representation. This suggests that an appropriate application for EM is as a knowledge base from which to generate conventional programs to meet evolving requirements.

3.4 EM and the Learning Process

The connection between EM and the learning process is highlighted by considering two aspects of the knowledge representation task, as it arises in modelling or programming the state-changing agents within a complex system:

- in the early phases of analysis of such a system, knowledge is most effectively represented by artefacts that imitate the experience of interaction with components of the system. At this stage, the behaviour of agents is understood in reactive terms, with reference to particular patterns of stimulus-response, and as communicated to the environment through the indivisible propagation of change. Such artefacts serve to represent the empirical evidence that informs our understanding of what the relevant agents are, and how they appear to interact, as this develops;
- if and when the behaviour of a system is comprehensively understood, knowledge can be represented by means of formal abstract specifications of the protocols that the agents follow. The use of such a representation is predicated upon the idealised presumption that we have sufficient reliable insight into how the agents interact.

Much education emphasises knowledge representation of the latter type, making use of highly abstract models of behaviour in which all intuitions about agency are hidden (cf. [40] Chap. XIII). Formal software construction is conceived in a similar spirit, relying on formulaic transformation motivated by paradigms for sequential programming and by the mathematical theory of computation. To be most effective, the teaching process has to expose and explore the experiential roots of knowledge that can inform theories. For system development to become more intelligible and flexible, computer-based modelling must invoke the essential interaction with real-world experiment and analysis of the practising engineer. It is this common concern for relating two kinds of knowledge that links the two apparently distinct objectives identified in §2.1.

From an educational perspective, the most significant aspect of agent-oriented knowledge representation lies in its explanatory power. The computational and procedural recipes that are associated with classical algorithms are the counterparts of rote learning in educational terms. They are optimised to perform efficiently on machines that are presumed absolutely reliable, but they embody no understanding of the context in which they operate, and are quite oblivious to significant changes in that environment. To the extent that this is their sole function, optimisation is desirable, but in practice there is a penumbra around the desired behaviour of a system associated with faulty operations, inappropriate use and unforeseen contextual changes. It is important to be able to take account of this not only because such exceptional conditions may arise, but because the extensions and refinements involved in the development process can otherwise entail major re-engineering.

Good understanding is associated with developing models and recipes within an explanatory framework. The learning activities in Box 1 are concerned not merely with *what happens* but with *why it happens*. An expert car mechanic is familiar with the penumbra of undesirable engine behaviours, and knows how to find plausible explanations. Such knowledge is essentially empirical in nature, drawing upon previous experience of 'what if?' scenarios and the results of previously unrehearsed experiments.

A simple illustration of the way in which such considerations arise in EM is to be found in the models of noughts-and-crosses (OXO-models) we have constructed. In one such model, the criterion by which the computer agent determines when to make a move is based on the number of Os and Xs, and which player took the first turn. If the modeller (acting in the role of a superagent as EM allows in any context) simulates a human player who cheats by taking two successive turns, the computer itself responds in a similar fashion. A different behaviour would be exhibited if the criterion were to be "did my opponent make the last move?".

In contemplating the use of EM in an educational setting, it is possible to identify many potential applications, most of which are yet to be explored. In future implementations of EM, the functions of the underlying algebra over which dependencies are formulated should themselves be represented by definitive scripts and agents [6]. This concept will expose the two contexts in the mind of the human interpreter of a definition of the form x = f(a,b,c, ...): one in which the explicit evaluation of the function f is being considered, and another in which f represents an implicit atomic operation. Such a representation can be applied in at least two ways to express learning activities:

- what is first learned laboriously as a procedure (cf. the explicit evaluation of f) can be subsequently directly apprehended (cf. f as an implicit atomic operation);

- a sequence of actions to achieve a particular objective (cf. the definition that involves f) can be conceived as composed of subsequences (cf. the explicit evaluations of f) to achieve subgoals.

There is also scope for sophisticated forms of dependency that can be applied in a CAL setting. By way of illustration, in performing mathematical calculations, it is important to know what procedures to apply when checking the correctness of a result. For example, when multiplying two polynomials, we should check that the result is a polynomial, that terms of the same degree have been collated, that the degree of the result is correct, that the constant term is correct etc. In an EM framework, multiple agencies of this kind can be set up in a simple manner so that they operate with a degree of eagerness over which the pupil or teacher has discretion. As an alternative application of a similar mechanism, agents can be set up to monitor the learning process, and alert the pupil to sequences of actions that can be applied to achieve certain goals. The degree of assistance that such a monitoring system provides to the learner can range from reminder, to hint, to demonstration. Most importantly, such monitors can be readily customised to the needs of particular pupils, and placed under the control of teacher or pupil.

One consequence of the prevailing computer science culture is that it is hard to develop CAL systems that expose the learner to the creative and exploratory processes by which proofs are discovered, or literature is written. In principle, EM is very well-suited to the role of presenting states of mind that relate to the early stages of development of a product, as cited in Box 1. EM techniques can be applied to the presentation of proofs (cf. [17]), to constructing environments to represent the state of the reader's knowledge at some point in a novel, or to simulating the patterns of movement and interaction conceived by a director of a play. The special benefit that EM can bring to these situations is in empowering the pupil to explore the consequences of actions other than those represented in the final product in an open-ended fashion.

4 Empirical Modelling in Context

EM uses two principal techniques for analysis and representation: definitive representation of state, and observation-oriented analysis of agents. This section relates these two techniques to other independent research with which it has both significant affinity and crucial points of difference. It also includes a brief discussion of philosophical outlooks with which EM has an affinity.

4.1 Multi-user Spreadsheets

What we have termed 'definitive principles for state representation' are becoming widespread in many modelling applications. For the most part, this stems from developments and extensions of the spreadsheet concept. Conventional spreadsheets and their extensions provide ever more powerful functionality for business applications, but they have also been proposed as a framework for intelligent business software [28], adapted for image processing [34] and geometric modelling, and adopted as a basis for web interaction. EM draws together many of these themes, putting particular emphasis on broader interpretations of observable than can typically be recorded in spreadsheet cells (inc. entities apprehended through cultural convention, or sophisticated thought-processes) and on perceptualisation (graphs, line drawings, textual annotations, windows).

It is also becoming clear that definitive principles can influence the way in which developers interact. Nardi [37] is a particularly interesting study of the impact of spreadsheet use on the software development culture. The themes emerging from this study are consistent with our experience and aspirations for EM. A spreadsheet can serve as the focus for many different types of interaction, and is amenable to re-use and extension in novel ways. For instance, new spreadsheets can be developed by customisation of existing ones, or through the integration of spreadsheets. Unlike a program with closed preconceived functionality, a spreadsheet can evolve as an organic entity through the corporate efforts of several types of user and developer. In this development process, many levels of expertise are represented, ranging from the sophisticated skills required of application programmers, to macro-level programming, to simple cell redefinition, and the choice of parameters.

The most significant aspect of spreadsheet use is that it promotes a change in management culture. Communication via documents favours one-way batch interaction modes, where the design process is compartmentalised and the transfer of information between different participants involves making firm or even binding recommendations. In contrast, communication of spreadsheets entails sharing and exchange of experimental contexts. This potentially permits much more flexible interaction and greater scope for creativity. On the other hand, it also demands another kind of discipline, posing challenges for conflict resolution, and for maintaining consistency. Determining the explicit status of parameters with respect to the different participants is essential here, so that it is understood who has the authority to make or declare the limits on choices of values.

A similar pattern of use has been observed within EM. Re-use is common, and has led to the construction of quite sophisticated products over a very short period of time. It has proved easy to transfer scripts from one context to another, and to extend existing models by adding new viewpoints and dependency links. Some consideration has been given to the extension of these principles to co-operative design [1].

The perspective that EM affords upon definitive representation of state indicates that combining spreadsheet principles with conventional programming technologies is a problematic issue. When definitive relationships are used in conjunction with traditional programming paradigms, this entails marrying together two different kinds of knowledge, the one associated with recipes for action in an established closed-world context, the other with the direct representation and open exploration of an external state.

Supplying a richer control environment to spreadsheets can be viewed as a particular case of the more general problem of composing agency and dependency without compromising their integrity. EM tackles this problem by using definitive representations to specify the environments for agent action, and explicitly relating all state changes in the system to agent actions. It is much harder to keep conceptual control of how transitions interact with dependencies in traditional programming paradigms. Evidence for this can be seen by examining the ways in which constraint-based, rule-based and event-driven technologies for rapid prototyping deal with agency and dependency. Constraint-based approaches attempt to combine declarative assertions about relationships between observables derived from a circumscribed view of system behaviour with procedural mechanisms for constraint satisfaction. In rule-based approaches, it is hard to distinguish between the processes that update dependencies and the significant actions of agents. Event-driven paradigms attach too little weight to maintaining the integrity of the contexts in which state changes are triggered.

Interfacing spreadsheet-like principles and rapid prototyping paradigms of this type directly is a conceptually unsatisfactory way of trying to take a short-cut towards traditional programs. EM favours a different approach to generating conventional programs, based on the translation of models to efficient optimised programs after restriction of their functionality. The possibility of partial translation is also admitted, representing commitment to certain features of the interaction within the model about which conviction has been acquired.

4.2 Agent-based Modelling

The analysis of agency is at the heart of observation-oriented modelling, but the particular framework for state representation adopted in EM distinguishes it fundamentally from other approaches to agent-based systems. The adoption of a non-logicist viewpoint is central, and there is some affinity between our stance towards knowledge and intelligence and that of R A Brooks [25, 26].

Current work in agent-based systems can be classified into two schools [46]: *cognitive* and *reactive* approaches. These two approaches are distinguished by the nature of the observables that are deemed to effect communication between agents, and the nature of the processing activity that is presumed to govern their actions. In a cognitive approach, agent interaction is typically described with reference to a belief, desire and intention framework for agent-based reasoning. In a reactive approach, agent interaction is primarily expressed in terms of more primitive stimulus-response patterns.

EM is in spirit closest to a reactive viewpoint, but the observables to which an agent can be responsive are unconventionally subtle and complex. Observables in EM encompass any identifiable feature of a situation to which a value can be ascribed, such that this value can be construed as directly perceived by some agent. This concept embraces what is perceptible in any way by a human agent (e.g. through touch and hearing); what is invisible to a human agent, but is construed to be directly sensed by a device (e.g. an electric current); what can be directly apprehended as a result of acquired skill or familiarisation (e.g. the cues through which a computer user apprehends that it is time to enter input, or a musician recognises a chord, or a lawyer appreciates that a legal procedure is completed). In this way, EM blurs the distinction between the cognitive and the reactive aspects of agent-oriented modelling. The rationale for adopting such a broad — and in some respects, mysterious — concept of observable, is that, when accounting for agent interaction in a concurrent system, the immediacy and timeliness of knowledge matters more than its nature.

EM bears on two fundamental questions raised by current research into the foundations for multi-agent systems:

- can the concept of agent be formally specified? (cf. [35])
- is there a development methodology for reactive systems? (cf. [46])

EM gives insight into these questions, and favours negative answers to both, at least as far as the aspirations of the logicist are concerned.

As applied across the spectrum of learning represented in Box 1, EM accommodates three concepts of an agent within a unifying framework for model construction:

View 1 an entity comprising a group of observables with unexplored potential to affect system behaviour;

View 2 a View 1 agent that is capable of particular patterns of stimulus-response within the system;

View 3 a View 2 agent whose pattern of stimulus-response interaction can be entirely circumscribed and predicted.

An example of a View 1 agent might be a stone, as represented by the open-ended group of observables (space occupancy, volume, weight, colour, smoothness, coefficient of restitution etc.) that are intrinsically concerned with it as an entity. For many practical purposes, it is inappropriate to regard a stone as an agent, but in some contexts it is viewed as one ("the window was smashed by the stone"). As for unexplored potential, who is to say whether, on rubbing the stone whilst incanting some magic words, a geni might appear?

A table lamp might be an example of a View 2 agent. Switching the lamp on and off is an example of a reliable stimulus-response pattern. In an intermediate state of knowledge about table lamps, I may or may not be prepared for the bulb to explode unexpectedly, and it may be that such behaviour is outside the scope of my concern.

For many practical purposes, an alarm clock is an example of a View 3 agent. The functions it performs are precisely known, and entirely predictable. In object-oriented development terms, an alarm clock is a good example of an object-like abstraction.

The classification of agency according to these views is not a formal matter. Agency is being invoked as a conceptual device fundamentally associated with how phenomena are construed to occur. This construal is a pragmatic issue, informed by the scope of intended application of the model, and by what assumptions are being made about the context for observation and interaction. It is clear that a developing child will appreciate an alarm clock as first View 1, then View 2 and then View 3 agent, and that from the perspective of the through-and-through empiricist there is no absolute justification for any of these views.

McCarthy has observed that the concept of agency is most useful when applied to a phenomenon that is incompletely understood [36]. EM promotes the related but more radical view that agency is only meaningful in relation to the development of understanding, as it is informally described in Box 1. All three views of agency are necessary to this development, and there is no absolute sense in which any one particular view must prevail in a given context. In any case, the concept of a View 1 agent is so broad as to be almost vacuous, and the agency of a View 3 agent is so circumscribed as to be almost redundant. It is in the attempt to make a conceptual transition from View 1 to View 3 that agency is a forceful concept.

Where the development of agent-oriented models is concerned, the classification of experience upon which EM puts emphasis is between what has been and is being observed and what is believed on the basis of empirical evidence and past observation to be reliably verifiable. Definitive representation of state plays a fundamental role in shaping the development process, since one and the same action on the part of an agent can have radically different effects upon the system state according to context. In view of this, it is possible to develop models incrementally, without necessarily having to revise the protocols for agent action in order to reflect new insights into system behaviour.

By means of EM, it is possible to construct computer models of n-agent systems (n>0) in which the roles of agents with a degree of autonomy can be played by the modeller in the guise of superagent. Phenomena that are comprehensively understood can be represented either by models that incorporate View 2 agents with a preconceived and limited degree of autonomy, or by mechanistic (0-agent) models. This latter category of models comprises those that are formal in the conventional sense. The relationship between 0-agent models — as realised by formal (typically computer-based) models with an unambiguous operational interpretation, and n-agent models — as realised by physical artefacts (possibly but not necessarily computer-based), resembles the relationship between a scientific theory and the physical apparatus and experimental protocols used to establish and corroborate the theory. The open-ended and situated character of the development process is consonant with Wavish and Graham's observation [43] that, in a reactive architectural paradigm, each system must be individually hand-crafted through a potentially lengthy period of experimentation. On this basis, it is not appropriate to speak of a development methodology.

5 Philosophical Foundations

The philosophical stance in EM is not logicist. In the spirit of B C Smith [41], it aspires to principles based on foundations more general than conventional computational theory admits, in which physical realisibility has an essential role. It is commonly supposed that logic represents the most primitive foundation for description of phenomena, and cannot be underwritten by yet more fundamental principles. The primary characteristics of logic as a medium for public communication, concerned with generalisations and the formalisation of absolute knowledge, are in complete contrast to our primary experience of agency, which originates in personal and private perceptions of specific situations, and has an experimental quality. An archetypal example of fundamental activity that is more primitive than logic in its essence, is the process by which an optician diagnoses the characteristics of a patient's sight through the systematic use of artefacts. An adequate account of EM has to accommodate both the empirical and rational elements in human understanding, and explicate the relationship between the two (cf. [14]).

EM poses special problems of communication and evaluation. In the light of its grand claims to be addressing fundamental problems of system development, it is natural for its critics to look sceptically at what appear to be modest practical products. To some extent, this has to do with inappropriate expectations: a failure to recognise that the novelty of EM lies in the development process and the nature of the artefact it creates. The primary interest in EM is not in highly optimised programs to solve particular specific goals. Whilst it would be of great interest to combine EM principles with the exceptionally high degree of realism that commercial computer graphics systems offer (cf. [2]), the underlying mechanisms that can generate convincing virtual realities necessarily require deep empirical foundations. For EM, the primary focus of interest in this connection is on investigating the explanatory frameworks behind complex behavioural models, not on solving the problems of generating realistic real-time animation.

Another major concern of critics is whether the use of linguistic constructs such as definitive scripts and LSD descriptions can possibly be anything other than a disguised use of formal language. The appropriate response to this is that the variables in these scripts and descriptions stand for external observables, and purport to represent them directly in a phenomenological sense. That is to say, they are observables that have not been subjected to the complete process of empirical refinement that allows us to represent them by logical variables. They are instead, like the variables in spreadsheet cells, representative of quantities whose values and relationships are determined with reference to experience of their external referents.

The issue of how meaning and knowledge is shaped by experience is central to the work of the American philosopher William James. In his 'philosophic attitude' of Radical Empiricism [32], James argues that the basic content of experience is much more than discrete sensory particulars — it embraces complex 'conjunctive relations'. James identifies "the relation experienced between terms that form states of mind" as the most intimate conjunctive relation, and criticises traditional empiricists for deploying highly abstract theoretical concepts to refer to the most basic particular elements of our experience: "Conception disintegrates experience utterly" ([32] p70), "[it] performs on conjunctive relations the usual rationalistic act of substitution — [taking] them not as they are given in their first intention, as parts constitutive of experience's living flow, but only as they appear in retrospect, each fixed as a determinate object of conception, static, therefore, and contained within itself." ([32] p236).

James' attitude to experience is consistent with what informs the use of spreadsheets, and the more general related abstractions of EM. Each cell of a spreadsheet refers to a particular identity and all values ascribed to this cell are interpreted as referring to this same identity. Redefining the value of a cell is not specifying a new spreadsheet — the different states of a particular spreadsheet are all as terms in one state of mind. The evolution of the spreadsheet reflects its referent as it is experienced, and its openness to interaction eludes any attempt to fix it as "a determinate object of conception". Such examples of conjunctive relations illustrate how, in EM, more than discrete sensory particulars are presumed to be empirically accessible, and directly apprehended. No matter that such features of our experience are unexpectedly sophisticated; as James indicates, in giving an account of 'pure experience' it is not necessary to explain "how experiences ever get themselves made, or why their characters and relations are just such as appear" ([32] p132-3).

James' writings endorse the outlook commended by EM in many other respects. The principles that motivate agent-oriented analysis in EM are consistent with those advocated by James: "the healthy thing for philosophy is ... to try to solve the concrete questions of where effectuation in this world is located, of which things are the true causal agents there, and of what the more remote effects consist" ([32] p185-6). As the discussion of agency in the previous section illustrates, it would be absurd to sustain such an investigation in a conventional formal framework, and a pragmatic stance must determine what we deem to carry conviction. For instance, an absolute test of object identity is in general hard to formulate and apply, but — for many practical purposes — potentially fallible criteria suffice.

In the context of educational concerns, one of James' most interesting ideas is that primary knowledge is defined by relationships between experiences [37, 32], and that knowledge representation can be achieved without conventional formal symbolism in this way. In the EM process, the correlation of computer model with the phenomena to which it refers is established and mediated by experience in just such a fashion. Only by reference to such a philosophic attitude as James adopts does it seem possible to address the issue of how it is that a particular EM model evolves indefinitely towards a more comprehensive representation of a particular referent without conceptually reaching a point of termination. For instance, the product of a modelling exercise associated with what at first appears to be such a circumscribed task as playing noughts-and-crosses resembles a laboratory for generating OXO-like games rather than a specific game-playing program (cf. [13]).

James' reference to the 'speechlessness' of the sensations that define 'pure experience' is consonant with the use of artefacts for representation and communication in EM. The philosophical status of extra-linguistic abstractions of this nature is controversial (cf. [22]). There is a strong prejudice towards seeing reality as mediated via language. For instance, all the themes cited by Kvale in his essay on Themes of Postmodernity, are familiar themes in EM but for one: "A focus on the way societies use languages to construct their own realities." [33]. For the future, it will be helpful to explore the potential for EM to fulfill the aspirations towards speechless communication implicit in James' account.

6 Conclusions

This paper has argued that there is an intimate relationship between EM and the learning process. Work at Warwick lends some support to this thesis. By its nature, EM invites justification through practice and demonstration, and — contrary to any impression of 'argument by rhetoric' that might be given by this paper — the ideas developed here have been shaped and motivated by practical experience. The large number of imaginative student projects that have been developed in connection with the Empirical Modelling Project over the last few years indicates the versatility and potential of EM. Experience with undergraduate students who have been demotivated by orthodox approaches to programming shows that — when used in conjunction with appropriate documentation and personal direction — EM can empower the learner to experiment and explore.

As EM tools and methods mature, a broader agenda for future work to explore the potential benefits of EM as the basis of an educational technology will become viable. One practical programme currently under consideration involves the distribution to schools of 'seed' models that can potentially become the focus for interaction, extension and refinement through the exchange of model fragments and agents. It is to be hoped that investment in these models over a period of time will demonstrate that they can evolve in an intelligible manner in response to new insights, modified patterns of use and developments in technology, through contributions from many participants in the education process. There are also reasons to suppose that EM has good potential for use in a CAL context.

Acknowledgments

I am indebted to all the members of the Empirical Modelling Group, and especially to Paul Ness and Steve Russ, for invaluable discussion on the themes of this paper, and for drawing my attention to so many significant references.

References

[1] V. D. Adzhiev, W. M. Beynon, A. J. Cartwright, and Y. P. Yung. A computational model for multi-agent interaction in concurrent engineering. In *Proc. CEEDA'94*, pages 227–232. Bournemouth University, 1994.

[2] V. D. Adzhiev, A. A. Pasko, and A. V. Sarkisov. Hyperjazz project: development of geometric modelling systems with inherent symbolic interactivity. In *Proceedings CSG 1996*. Information Geometers, April 1996.

[3] J. A. Allderidge, W. M. Beynon, R. I. Cartwright, and Y. P. Yung. Enabling technologies for empirical modelling in graphics. University of Warwick, May 1997. submitted to PPG 1997.

[4] V. M. Axline. *Dibs: In Search of Self — Personality Development in Play Therapy*. Penguin, 1990.

[5] W. M. Beynon. Definitive principles for interactive graphics. In *NATO ASI*, volume 40 of *F*, pages 1083–1097. Springer-Verlag, 1988. also University of Warwick Computer Science Research Report 93.

[6] W. M. Beynon. Evaluating definitive principles for interactive graphics. In *Proc. Computer Graphics International 88*, pages 291–303. Springer-Verlag, 1989.

[7] W. M. Beynon. Agent-oriented modelling and the explanation of behaviour. In *Proc. International Workshop Shape Modelling Parallelism, Interactivity and Applications*. University of Aizu, Japan, September 1994.

[8] W. M. Beynon, V. D. Adzhiev, A. J. Cartwright, and Y. P. Yung. An agent-oriented framework for concurrent engineering. Technical report, oct 1994.

[9] W. M. Beynon, I. Bridge, and Y. P. Yung. Agent-oriented modelling for a vehicle cruise control system. In *Proc. ESDA92*, 1992.

[10] W. M. Beynon and A. J. Cartwright. A definitive approach to the implementation of CAD software. In *Intelligent CAD Systems II: Implementation Issues*, pages 125–145. Springer-Verlag, 1989.

[11] W. M. Beynon and R. I. Cartwright. Empirical modelling principles for cognitive artefacts. In *Proc. IEE Colloquium: Design Systems with Users in Mind: The Role of Cognitive Artefacts*, December 1995.

[12] W. M. Beynon and R. I. Cartwright. Empirical modelling principles in application development for the disabled. In *Proc. IEE Colloquium Computers in the Service of Mankind: Helping the Disabled*, March 1997.

[13] W. M. Beynon and M. S. Joy. Computer programming for noughts and crosses: New frontiers. In *Proc. PPIG94*, pages 27–37. Open University, January 1994.

[14] W. M. Beynon, P. E. Ness, and S. Russ. Worlds before and beyond words. Research Report 331, Department of Computer Science, University of Warwick, 1995.

[15] W. M. Beynon, M. T. Norris, R. A. Orr, and M. D. Slade. Definitive specification of concurrent systems. In *Proc. UKIT 1990, IEE Conference Publications 316*, pages 52–57, 1990.

[16] W. M. Beynon, M.T. Norris, S.B. Russ, M.D. Slade, Y. P. Yung, and Y.W. Yung. Software construction using definitions: An illustrative example. Research Report 147, Department of Computer Science, University of Warwick, September 1989.

[17] W. M. Beynon and S. B. Russ. Variables in mathematics and computer science. Research Report 141, Department of Computer Science, University of Warwick, 1989.

[18] W. M. Beynon and S. B. Russ. The interpretation of states: a new foundation for computation? Technical report, University of Warwick, February 1992.

[19] W. M. Beynon, M. D. Slade, and Y. W. Yung. Parallel computation in definitive models. In *Proc. CONPAR88*, pages 359–367, June 1988.

[20] W. M. Beynon and Y. P. Yung. Definitive interfaces as a visualization mechanism. In *Proc. GI90*, pages 285–292, 1990.

[21] W. M. Beynon, Y. P. Yung, A. J. Cartwright, and P. J. Horgan. Scientific visualization: Experiments and observations. In *Proc. Eurographics Workshop Visualization in Scientific Computing*, pages 157–173, 1992.

[22] G. Bird. *William James.* Routledge and Kegan Paul, 1986.

[23] P. Brödner. The two cultures in engineering. In *Skill, Technology and Enlightenment*, pages 249–260. Springer-Verlag, 1995.

[24] F. P. Brooks. No silver bullet: Essence and accidents of software engineering. *IEEE Computer*, April 1987.

[25] R. A. Brooks. Intelligence without reason. In *Proc. IJCAI-91*, pages 569–595, 1991.

[26] R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[27] R. Feynman, R. Leighton, and M. Sands. *The Feynman Lectures on Physics: Volume II.* Addison Wesley, Reading, Massachusetts, 1964.

[28] R. S. Freedman, R. P. Frail, F. T. Schneider, and B. Schnitta. Expert systems in spreadsheets: Modeling the wall street user domain. In *Proc. 1st International Conference on AI Applications on Wall Street*, pages 296–301, 1991.

[29] H. Gardner. *Frames of Mind: The Theory of Multiple Intelligences.* Basic Books, 1983.

[30] D. K. Gehring, Y. P. Yung, R. I. Cartwright, W. M. Beynon, and A. J. Cartwright. Higher-order constructs for interactive graphics. In *Proc. Eurographics UK Chapter, 14th Annual Conference*, pages 179–192, 1996.

[31] D. Gooding. *Experiment and the Making of Meaning.* Kluwer, 1990.

[32] W. James. *Essays in Radical Empiricism.* Bison Books, 1996.

[33] S. Kvale. Themes of postmodernity. In *The Fontana Postmodernism Reader*, pages 18–25. Fontana, 1996.

[34] M. Levy. Spreadsheets for images. Technical Report CSL-TR-94-607, Stanford University, 1994.

[35] M. Luck and M. d'Inverno. Agency and autonomy: a formal framework. Research Report 276, Department of Computer Science, University of Warwick, December 1994.

[36] J. McCarthy. Ascribing mental qualities to machines. Technical report, Stanford University, 1978.

[37] B. Nardi. *A Small Matter of Programming: Perspectives on End User Computing.* MIT Press, 1993.

[38] P. Naur. *Knowing and the Mystique of Logic and Rules.* Kluwer Academic Publishers, 1995.

[39] S. B. Russ. Empirical modelling: the computer as a modelling medium. *BCS Computer Bulletin*, pages 296–301, April 1997.

[40] B. Russell. *The ABC of Relativity.* George Allen and Unwin, 1969.

[41] B.C. Smith. Two lessons in logic. *Computer Intell. Vol. 3*, pages 214–218, 1987.

[42] H. Ossher W. Harrison. Subject-oriented programming (a critique of pure objects). In *OOPSLA '93*, pages 411–428, 1993.

[43] P. Wavish and M. Graham. Roles, skills and behaviour: a situated action approach to organising systems of agents. In *Intelligent Agents: Theory, Architecture and Languages*, volume 890 of *LNAI*, pages 371–385. Springer-Verlag, 1995.

[44] D. West. Hermeneutic computer science. *Communications of the ACM*, 40(4):115–116, April 1996.

[45] L. Wittgenstein. *Philosophical Investigations.* Oxford, 1954.

[46] M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1995.

[47] B. Wyvill. *An Interactive Graphics Language.* PhD thesis, Bradford University, 1975.