Analysing the *lack* of Demand Organisation

Philip Boxer, Boxer Research Ltd
and
Professor Bernard Cohen, Interoperable Systems Research Centre, City University

Abstract

We seek to develop means of intervention in Enterprises that will enable them to react in an effective, sustainable and timely fashion to changes in the ways that markets and demand are organized; that is, to act *strategically*.

We take an enterprise to be some entity that seeks to provide its clients with services that they value while maintaining its ability to do so in the face of changes in the demands of its clients and in the resources at its disposal. The services that clients value form around what the organization of their demands *lack*. The concept of *strategy* therefore rests on critically evaluating the ontology and semantics of the Enterprise in relation to these *holes* in demand organization.

We access ontology and semantics by constructing and manipulating hypothetical, first-order, mathematical models of the Enterprise's services and of its value-adding processes. Because an enterprise is an *anticipatory* system, its semantic domain must include representations of the enterprise's model of itself and of the market and demand organizations within which it competes. First-order (set) theory provides adequate expressive power here, but alternative, higher order, mathematical frameworks, such as Dubois' *hyperincursion*, provide inadequate power, particularly in relation to the analysis of the properties of *emergence*.

Knowing exactly why and where this mathematical *lack* manifests in the analysis process enables effective collaboration between *systems analysts* and *psychoanalysts*, and suggest directions for mathematical research.

Keywords: Anticipatory, Enterprise Modelling, Hole, Ontology, Stratification.

Introduction

An enterprise may be modeled as an interrelated collection of services that would, in concert, enable the enterprise to satisfy its needs. Since these services might, in general, be supplied by alternative providers, their descriptions, (or specifications) should be independent of their implementation by any provider, and each provider should be able to demonstrate that the service complies with its specification.

This paper is about an approach to modeling that can support a dialogue concerning the design of an enterprise's value-adding processes. In practice, the specifications of 'services' in an enterprise are rarely explicit. Making them explicit involves making a distinction between value chains and value ladders (this distinction between value ladders and value chains is elaborated in "Performative Organization", Boxer and Wensley 1997), but to compete on value ladders means making design specification explicit (an approach to this is outlined in Boxer, 1993). When this approach is applied to the industry contexts within which enterprises compete, it results in a distinction between Market and Demand Organization.

Our concern is to develop means of intervention in Enterprises that will enable them to react in an effective, sustainable and timely fashion to changes in the ways markets and demand are organized, that is, to act strategically. We take an Enterprise to be some entity that seeks:

- to provide its clients with services that they value and
- to maintain its ability to do so in the face of changes
  - in the demands of its clients and
  - in the resources at its disposal.

These services which clients value themselves form around *holes* in demand organization – around what demand organization *lacks*. The conception of "strategy" ultimately rests therefore on critically evaluating the ontology and semantics of the Enterprise in relation to these holes in demand organization (Boxer and Wensley, 1997).

The main mathematical problem facing the modeler in the business world arises from the 'anticipatory' nature of systems in that world. The semantic domain must include a representation of the business's model of itself (warts and all), so that the formal model may include among its consequences the kinds of decision that the business might make based on its model of itself. But it also has to include a model of the organization of the market organization and demand organization within which it is competing. Hence the need for something like Dubois' "incursive" (formal) systems (Dubois, 1996), and to differentiate between the notions of 'time' inside in the business (the 'simulation') and outside in the competitive environment (the 'world').

Approaching the client's frame of reference

We want to distinguish between the description of a service and its implementation both for technical and for business development reasons, especially for those services that are to be available from different (competing) suppliers.

- Technically, it has to do with quality: every supplier must 'guarantee' that the service provided satisfies some description that tells every client what properties of the service can be 'relied' on, whoever supplies it. This notion of 'rely/guarantee' is central to the formal specification and verification of programs, etc. (Cohen, Harwood and Jackson, 1984).

- In terms of business development, the client should be able to determine from a service description what its value might be to (i.e. to include it in a design of) the enterprise before investigating the cost of purchasing it from different suppliers; each supplier, on the other hand, should be able to evaluate alternative implementations (i.e. designs), all of which (demonstrably) satisfy the same service description, with respect to that supplier's culture/knowledge/value system.

There are obvious interpretations of 'added value' and 'competitive edge' in terms of the client's own culture/knowledge/value system - the client's frame of reference, but this does not satisfactorily account for 'stratification' (Boxer, 1996).

- So the third reason for distinguishing between service description and implementation is so that the descriptions can both provide and exploit 'languaging tools for articulating different forms of stratification'.

Each service implementer (or agent) may itself be modeled as an enterprise in a similar manner, its purpose being to supply services to its clients while in turn invoking services provided elsewhere. Some agents may be characterized as platforms: integrated collections of 'generic' services that support the implementation of many different service specifications (Cohen, 1996a). Similar contextual issues may therefore arise at many levels of service provision.

Thus, the interrelationships among services that comprise the context of a given enterprise may both add constraint to individual service specifications, as well as endowing their composition with emergent properties that no isolated service exhibits. It is therefore possible that an implementation that complies with an isolated service specification may fail to satisfy that specification in some context (Cohen, 1996a). This means that the choice of what is modeled and what is ignored becomes of particular importance.

Anticipatory Systems and Stratification

Another way of approaching this problematic of emergence is in terms of the effects of stratification - the way services are stratified will affect what emergent/higher order properties will be observed. Thus, not only does the client's frame of reference affect the ways in which 'added value' and 'competitive edge' are interpreted. It will also affect the way services are stratified.

An Enterprise may be construed as a complex anticipatory (in the sense of Rosen, 1987) system, as may be every agent serving within an Enterprise. In other words, one can attribute to each of an Enterprise's services an 'agent', 'implementing' the service in its own way and subject only to the rules governing its interoperation with other services. Such an 'agent' can be understood as having the same characteristics as an 'Enterprise'. As a result, this distinction of being "within", which can be made between Enterprise and the larger context in which clients value the services provided by the Enterprise and its competitors, can be applied in the same way between agent and Enterprise itself.

The way this "within" distinction is made between a particular relation of inside to outside is dependent on the ontology of the modeller. When applied recursively by a modeller, it results in a *stratification* of embedded 'insides' and 'outsides' (Van der

Vijver, 1996a; Boxer, 1996). So what happens when we seek to model an enterprise conceived of as being stratified in this way?

We come to this problem from different points in the 'analytic' universe: Philip Boxer from the psychoanalytic approach to the Enterprise, and Bernie Cohen from the systems analytic approach. The paper aims to consider the implications of bringing these two approaches together in the modelling of the Enterprise. We will argue that the modelling developed by Boxer in relation to clients' expectations of the services provided by an Enterprise, together with the 'knowledge' implicit in the stratification of the modelling of the Enterprise itself, provide an ontological setting within which the formal 'systems' models of Cohen can be elaborated and composed. As a result, the latter offers the possibility of illuminating the topological structures and logical consequences of the former. We make a start on this by articulating a semantic domain for stratification analysis.

Holes in Space

Formulating the *lack*

Rosen's 'Anticipatory Systems' (1987) provides an exceptionally clear exposition of the issues surrounding the relationship between ('natural') systems and models of them. In understanding more of the difficulties surrounding this relationship, this question of the *lack* can be approached.

Rosen treats 'abstraction' as the necessary selection of a set of 'observables' from a system's state space in terms of which a formal model can be expressed. Every formal model can therefore capture only a 'subsystem' of the system observed, and that subsystem is considered in the model to be 'closed', i.e. 'isolated' from the influence of any other part of the system's state space. An 'open' system is one in which this assumption may be violated.

To model an open system therefore requires either an extension of the closed system model, in which the additional influences from the environment are incorporated into state components, or the composition of a closed system model with another that captures some other natural system with which the former is deemed to interact. This definition is recursive, since the formal model + model of system observed can become a composite model which can itself be taken as a formal model in relation to a system observed. If we therefore take some composite model with some number of stratified levels of emergence as being the formal system, then 'error' can be treated as a 'bifurcation' that arises as the behaviour predicted by this formal model diverges from the behaviour exhibited by the system observed. What forms of 'error' can arise, then?

Types of Error

From the point of view of the formal system, two kinds of error can arise in claiming any kind of ability to state the 'truth' about the system observed, and a third kind of error can arise in the composition of the formal system and the model of the system observed. These can be summarised in terms of three characteristics of the model:

- the *operation* of the model in replicating the behaviour of the system observed;
- the internal composition of the formal *model* itself (whatever its levels of stratification, and however it can be formalised in a state space); and
- the *composition* of the organisation of the data, in terms of which the system observed is being described, in relation to the formal model.

With a 'Type I' error, model and composition are taken as being 'true' but the operation of the model in replicating the data produces wrong results. With a 'Type II' error, the composition is 'true' and the model's operation is 'correct' but the formal model itself is wrong. And with a 'Type III' error, the model is 'true' and is applied 'correctly', but the composition as a whole of model in relation to system observed is wrong.

In considering the relationship between observed systems and formal models of them, which class of error we think we are dealing with raises important questions therefore:

- with a Type I error, we can say that there is a *correspondence* error between the model and the system observed; and
- with a Type II error, we can say there is a *coherence* error in the particular way in which the model itself is constituted.

In the first of these, there has to be a presumption of an independently existing (‘natural’) reality in order to be able to claim a correspondence error between (‘natural’) system and model - a question of reference, while for the second, it has to be possible to be able to assert that a non-sense has arisen - a coherence error. For this second type of error, therefore, there is a need for some form of (normative) state space in which it is possible to assert that a coherence error has arisen.

So what is happening with a Type III error? In a Type III error, what is called into question is the point of view from which both the formal model and the model of the system observed have been constructed.

- With a Type III error, we can say that an *undecidability* error has arisen in the way the composition of the model in relation to the system observed has been framed.

If this undecidability error is to be distinct from errors of correspondence or coherence, a Type III error has to be an error in which it is not possible to determine whether or not a Type I or a Type II error has arisen. The very constitution of the model itself in relation to the system observed is unable to render ‘tractable’ the truth or falsity of the relation between the (‘natural’) system and the model.

So how are we to distinguish Type III errors without recourse to a ‘higher’ authority either of correspondence-to-reality or to coherence-in-state-space? (Maturana (1988), in seeking to argue against Aristotle’s ‘transcendental’ ontology, was arguing against the first of these in order to be able to assert the supremacy of the second ‘constitutive’ ontology).

Undecidability Error as ‘hyperincursion'

'Bifurcation' between the observed behaviour of a system and the predictions of a formal model of it is an empirical matter that can be decided only at the 'time' (in the natural system) when divergence occurs. On the other hand, some of the 'errors' manifested by the 'emergent properties' of a composite model are formal, and occur at the 'time' (in the modelling world) when the composite model's implications are derived (e.g. inconsistencies). Of course, some are also empirical in that, although the composite model is consistent, its predictions when imputed on the natural system ('now', rather than in the 'future') diverge from observation, i.e. are 'invalid'. So,

- an invalid model would exhibit a Type I (correspondence) error;
- an inconsistent model would exhibit a Type II (coherence) error.

Anticipative Systems Theory formulates undecidability as a hyperincursion. We could say therefore that, instead of there being a (first order) bifurcation, because the natural system had not yet reached its bifurcation with the model, there was instead a second order bifurcation –the hyperincursion becomes a bifurcation in anticipation itself: how we are to decide how we are to anticipate. Thus

- a hyperincursive model would exhibit a Type III (undecidability) error.

We might get a glimpse of potential Type III errors by monitoring the changing ontology and topology of the formal model as it is changed to accommodate the needs of the 'natural' system observed. Sudden and dramatic alterations to those structures might suggest the onset of 'undecidabilities', or at least raise red flags that signal serious conceptual alterations and motivate a closer look at strategies and plans.

When Maturana (1988) asserts 'the supremacy of the second 'constitutive' ontology', he seems to be demanding that model consistency take precedence over model validity. There is indeed no purpose to be served by even attempting to validate an inconsistent model, but this is a chronological precedence for the modeler. To take it as an epistemological precedence is to commit solipsism. But with Type III errors we are calling into question the ontology of the modeler - the modeller as frame.

Bifurcating Models

We now have a rather complicated situation in which we are talking about a number of different contexts in which different forms of error may arise and in relation to which it is not always clear what the notion of an 'error' is referring to. The operation of an observing model on some observed model results in the anticipation of a certitude (Lacan, 1988) - a prediction taken to be true. When we are talking about a Type I 'error' we are talking about the prediction having been proved to be wrong, as a result of which we are seeking to formulate an explanation. In this sense, therefore, an error always ends up being a 'bifurcation' between observing and observed model.

For an error to arise in this Type I sense, we need a relationship between the observed behaviour of one system, and the predictions about the behaviour of that system by another (formal) model. But in our case we have two types of model that we can speak about:

- a formal model of a natural system - such as might arise in seeking to 'explain' the behaviour of systems which themselves remain wholly 'Other' to us, even though we may approximate to the nature of their behaviour through describing them in terms of composite models.
- a formal model of a composite system - the composite system being some system of systems which, in being 'artificial' in nature, can themselves be described as formal systems, even though we may only be able to do so one-by-one. The formal modelling of this composite system has to be able to account for 'emergent' behaviours arising through the interactions between its constituent systems. The composite system may itself include a composite model of (a) natural system.

In the first case, we are talking about a *1st order* modelling process in that we have a model and a thing modelled, even though the thing modelled may itself be stratified. In the second case we are modelling the modelling of a natural system and we may say, therefore, that we are dealing with a *2nd order* modelling process.

In each of these cases, an error can only be said to arise as an *après coup* - after the event of the disconfirmation of the anticipation of the certitude, we might say. In speaking of what *type* of error we have made, we are therefore seeking to form some kind of explanation of how the bifurcation might have arisen. What kind of modeling situation we assume ourselves to be in will affect the form our explanations take. We can therefore further elaborate the versions of error arising as a result of a bifurcation between the anticipated behaviour resulting from the modeling process and the behaviour of the system (whether composite or not) modeled:

- an invalid model - a correspondence (Type I) error.
- an inconsistent model - a coherence (Type II) error.
- a bifurcating model - an undecidability (Type III) error.

Put in these terms, it is not quite right to say that a Type III error is therefore just a Type I error that has not yet happened. A Type II error can always be said to be a series of Type I errors (this being the normal defense of a current scientific paradigm), just as a Type III error can be said to be a series of Type II errors (this being the normal defense of the practices of science itself) …. At this point we are questioning the nature of the truth claims which support the practices of science itself (Lacan1966). But what kind of explanation is assumed for the bifurcating error leads to two forms of Type III undecidability:

- a ‘correspondence’ undecidability, if the assumption is that the error is a failure in the scope of the model (i.e. that ‘hidden variables’ seem to be involved); and
- a ‘coherence’ undecidability, if the assumption is that the error is a failure in the very constitution of the model itself.

Finally, with reference to the demand to make model consistency take precedence over model validity, this is only possible to argue in principle if we wish to argue that the ‘natural’ is *maya* - a construction, as indeed the radical constructivists seek to do (Boxer and Kenny, 1992). These are exactly the conditions which can be said to exist for the modeler of composite (artificial) systems, however, insofar as they are able to postpone indefinitely an encounter with the Real!

Holes in (state) space

'Anticipatory' systems can be considered as that class of systems which contain formal models of their own (sub-)systems which they use to predict the future behaviour of their own (sub-)systems in relation to the system(s) observed; and through which they are enabled to select their present actions. It becomes more useful therefore to speak of ‘bifurcations’ rather than errors arising from the divergence between the system's internal models of the behaviour of its (sub-)systems and those (sub-)systems’ ‘actual’ behaviour. We have now re-located the boundary from being between the ‘natural’ system and the model of it, to one between the model of the model-in-relation-to-the-’natural’-system and the model-in-relation-to-the-’natural’-system itself. But if we abandon the notion of an independently-existing reality, a ‘natural’ system can never be more that another open system.

For the anticipative model, ‘bifurcations’ will now arise from erroneous decision making (whether Type I or Type II) in relation to both explicitly modeled and emergent behaviours of the (sub-)systems. Reconstruction of the internal model may 'compensate' for such bifurcations, and may have to involve radical 'redesign'. We now have a way of speaking about the issues raised by 'service interaction' (Cameron and Cohen, 1996). It does not, of course, go into the manifestations of these issues in information system implementation (where the models of system and platform interact), nor into the 'legacy problem' (where old and new ontologies have to be reconciled). But it does provide a unifying framework in which all of these issues may be considered to be related, and to have related formal and methodological requirements.

Finally, both for these anticipative models and for the model-as-a-whole constituted in relation to the other ('natural') system(s), we can consider a 'stratified analysis', where bifurcations can arise at every level. If we formulate an undecidability as arising in relation to a *hole*, we must recognise two kinds of hole: those that can be filled and those that cannot, within the current stratification. A 'third cybernetics' might be formulated as one that seeks to address the problematics of such holes. A *third-order* cybernetics would then be one where the 'model' becomes what arises in relation to 'unfillable holes', experienced by the observer as the problematics of desire. In other words, an account of the dynamic properties of stratification would require a theory of the observer, in which correspondence and coherence undecidabilities, at *every* layer of *any* conceived stratification, could be analyzed in terms of the composition of models and the discovery of their emergent properties.

The modeling approach

The object of interest

The IS culture has grown up with a particular kind of analytical approach, mainly because it has taken its metaphors from first-order cybernetics and computing - hence the 'state machine', 'dataflow', relational', 'object-oriented', etc., approaches to conceptual modeling' and 'business process re-engineering'. However, this background itself constrains the choice of modeling language and therefore limits the diversity of articulative structure, and the strength of analytical power, that can be deployed.

Set against this, the apparent success of IS modeling techniques in certain business applications has encouraged the belief that they can be used effectively for purposes far beyond the limits of their applicability - a belief encouraged by those with interests vested in the modeling tools, as vendors, teachers, or practitioners. There is therefore a key problem to be considered: what is the relationship between the IS frame of reference and the client's frame of reference?

We could choose to work within the IS frame of reference. It would undoubtedly be an easier 'cut' to work with, given the natural leaning of IS towards formalization. But would it enable us to consider the ways in which the IS frame of reference did or did not restrict the strategic options open to the enterprise which it was supporting?

Our interest is in considering the ways in which to look at the enterprise as a whole (a relatively 'simple' one, whatever that means, at first). Our aim would be

- to work within the client's frame of reference in order to articulate (service) models and their stratification.
- To work in relation to the client's development objective for the enterprise in order to identify 'lacks'.

By considering the implementation of stratified service models on IS platforms, we may find that it is precisely in the way these models are, or have been, implemented on IS platforms that endows these holes with varying degrees of resistance-to-change.

Problems recently observed in enterprises as disparate as Telecommunications (‘feature interaction’, Cameron and Velthuijsen, 1993) and Healthcare (‘electronic patient records’, Cohen 1996b, 1997) show us that they concern the way the technology may not only not be unable to support certain forms of languaging processes between people, but that these processes have a logic of their own. It is insofar as a hole acquires resistance-to-change that they become objects. These objects are therefore our object of interest. Formulated as such, we would want to suggest that these objects *qua* holes are now ubiquitous and should be taken seriously by corporate strategists, standards writers and systems designers alike.

Developing a rich descriptive methodology

In these terms, the modeling techniques and notations currently favored by IS disciplines do not provide the requisite analytical power, nor do those of the computing platform communities. Richer and mathematically more sophisticated modeling frameworks are urgently needed and there are very few candidates.

Decon™ (an expert system shell, developed by BRL) is proposed as a rich descriptive methodology which can be used not only to ‘extract’ the axiomatics of the discursive practices in the client system, but also to generate stratifications which are relevant to the strategic options which the enterprise wishes to consider. An analysis of holes at all levels of strata make it possible to determine where the forms taken by particular services need to be migrated to a form of greater articulation. The extracted axiomatics means that this can be done within a formal framework.

Thus Decon™ becomes a means of identifying ('extracting') viable ontologies and structures, in terms of which we might construct formal models, that articulate a theory (or 'axiomatics'), whose closure might account for 'the discursive practices in the client system'. The basic thesis is that, if we define the aims of a business development process in terms of the changing problem domains in which the business is competing, then the <u>reason</u> that legacy IT exerts a disabling effect on business change is because of holes both in the information environment and in the network organization of services... and so on down the strata.

So what is being modelled?

The class of anticipatory systems is characterised by 'non-locality'. This means that an anticipatory system's global behaviour in relation to its context (whether successful or not) is emergent from the composition of its local parts and cannot be attributed to any one part in isolation. In particular, one cannot localise the model used by the system to anticipate the potential consequences of its actions.

The activation of this model, and the delivery of its predictive consequences, can be construed as one of the services invoked by the Enterprise, to be composed with other services so invoked in order to enable the Enterprise to act strategically. This modelling strategy escapes the criticism of reductionism because the whole is decomposed into a collection of inter-operating services, each described independently of its 'implementation’, of which there might be several possible, or even available.

The analytical problems raised by such a modelling strategy are legion, not least of which is the definition of what is ‘within’ the Enterprise. (This question relates to the questions of identification and embodiment raised in Van de Vijver’s (1996b) and in

Boxer, 1995). But they at least direct attention to the nature of the mathematical framework in which to construct service descriptions, so that their composition and the inference of their consequences can be performed.

Cohen, on the one hand, starts with a formal representation (in set theory), which is an abstraction and generalisation of the ontology and intended semantics of the Enterprise as an integrated collection of 'services'. Each service is modelled as a state space (an indexed collection of set-valued state components) and the events in which the service participates (each defined by precondition and postcondition predicates on the state space, Schuman, Pitt and Byers 1990). In general, the services so modelled are not isolated. They comprise each other's context and so 'act' on each other (if this were not so, then the enterprise would not be a 'system' but a mere aggregation of services). As a consequence, he has to address the following particular forms of relationship:

- The logical verification of the models' internal consistency.
- The logical derivation of the models' behavioural properties (closures), so that they may be validated by attempted refutation of their predicted behaviour in relation to and satisfaction of client expectations.
- The utility of the generalisation used, assessed by restricting some parameters of the generalised models to values that had been used in the past and showing that the resulting, more specific models indeed account for the behaviour of the enterprise’s 'legacy' services.

Boxer, on the other hand, starts from defining the relationship of the Enterprise as a whole to what is valued by the client, defined in terms of the Enterprise’s ‘knowledge’ of itself and its competitive environment. He starts by seeking to describe the activity logics and capabilities that comprise the underlying task structures in relation to which the Enterprise’s workgroup and information processes are constituted. This provides an entry-point for a more abstract form of description, which describes the organisation of the market, and demand contexts within which clients’ expectations are formed.

Boxer’s model of the Enterprise takes the form of a User-Defined Knowledge Base (UKB) which includes modelling the effects of stratification within the Enterprise itself. Through the instantiation of the objects and relations in the UKB, the formulations of clients’ expectations can also be tested for consistency (in the familiar, not the logical, sense) in relation to the data generated by legacy systems.

But in order to be able to articulate this model of the Enterprise’s ‘knowledge’, a formal syntax has to be developed - the Expert-Defined Knowledge Base (EKB). The description is therefore done in terms of distinct types of relation, and the Decon™ system is a software tool that supports the construction of such a relational model of the enterprise in the EKB. The particular relationships resulting from these types of relations are formulated in the UKB by inviting the enterprise’s agents to nominate those services, processes, activities, capabilities, etc. in terms of which they currently perceive the enterprise, and the organization of the market and of demand.

The resulting matrices can then be analyzed using the PAN™ system (developed by BRL). PAN™ is an analytical tool which uses the language of Ron Atkins’ Q-analysis

(Casti, 1994), and which provides empirical evidence for both the structural differentiation of an enterprise's services, but also the identification of 'holes' in both market and demand organization. Feeding these results back to the agents enables (and encourages) them to reconsider their own theories, to reformulate them and to perform this kind of analysis for themselves.

Anticipative systems

In Dubois' conclusions to "A Semantic Logic for CAST ...", he says: "Any systems theory deals with a semantic logic, corresponding to a meaning in the mind of the CAST engineer, which can be represented by mathematical equations." This sentiment differs from the approach in Decon™. What is this difference?

In order to formulate a 'User-Defined Knowledge Base', Decon™ needs a syntax. The concrete syntax provided in the 'Boxer' approach to micro-organizational design contains symbols that act as tokens for certain classes of objects and relations in terms of which the client's statements about his enterprise are articulated. The terms in these statements, whether expressed lexically using words or tokens, need to carry two aspects of 'meaning': their 'sense' and their 'reference'. Their reference is to things in the client's world -- his frame of reference -- and are therefore ontological in nature. Their sense is 'semantic', in Dubois' usage, where both the objects and the relations are mapped into a mathematical domain that provides a formal system with the analytical power necessary to determine the consistency of any model constructed in it and to enable the derivation of its logical consequences. As Rosen says,

> "What we require is a new universe in which we can create systems. But in this universe, the systems we create will have only the properties we endow them with; they will comprise only those observables [read 'objects'], and satisfy only those linkages [read 'relations'], which we assign to them. But the crucial ingredient of this universe must be a mechanism for making inferences, or predictions".

That is what Cohen understands Dubois to mean by "a meaning in the mind of the CAST engineer", what Bunge (1974-80) means by the 'fictional' nature of formal models, and what Rosen means by 'abstraction', which is the deliberate focus of the modeller on a (fictional) 'subsystem', and by 'imputation', which is the modeller's act of relating the model, and its logical consequences, back onto the observed world, with a view to the refutation of the model, or to an improved understanding of the world.

The language(s) of Decon™, then, need semantics for their syntax, and those semantics need to be expressed as models in a mathematical framework with suitable expressive and analytical power. For example, in the 'Glass Industry' knowledge base, there is a relation called 'supplies' which is defined to take named objects of certain classes in its domain and range, specifically product-process, or process-product. At this level, Decon™ allows the user to check formulations in the User-defined knowledge-base (UKB) for syntactic validity. The syntactical constraints on the UKB are not however making it possible to assert semantic constraints on its usage, such as whether every instance of it should exclude the identity (as the PAN™ analysis would suggest), that is that "x supplies x" is not acceptable even when "x" appears both a product and a process. Such constraints appear as 'invariants' in state space models. These ‘invariants’ are indeed not fully expressible in relational languages, even in those in use by the OO community. They also serve the much more important function of restricting the state space of composite system models to a subset of the product of the state spaces of their component 'subsystems', in order formally to capture the additional topology that arises from such composition and that, when imputed to the world of observation, accounts for both 'emergent' and 'pathological' behaviour (e.g. 'holes' or 'feature interactions'). So what are the characteristics of this ‘state space’. It is the expert-defined grammar (EDG) in Decon™ which begins to formulate these semantic constraints. The challenge of a formal approach to modelling is to formulate these constraints within a state space.

What we have done is to explore the overlapping areas of functionality and the ways in which our different approaches to the overall problem have created complementary methods in which the Decon™ toolset supports the formal modelling process between client and facilitator/modeller. As a result, the client is enabled to articulate an ontological structure with its associated ‘holes’ and conceptual flaws within the client’s world view.

In the sections below, we establish the equivalencies between the formal model (UKB) provided by this toolset, in relation to which the axiomatics (EKB) of the enterprise can be formulated, the logical consequences of which might reveal flaws of a different kind within the framework of a formal abstraction.

Analysing the lack

The toolset used by Boxer supports a modelling process between Enterprise and facilitator/modeller of the Enterprise’s relations to Client Expectations. As a result, the Enterprise is enabled to articulate not only an ontological structure which is 'lacking' in relation to the client's demand organisation, but an organisation of demand which may itself be lacking. Thus the EKB and UKB provide starting points for the construction of formal systems models, the logical consequences of which can reveal holes.

However, the relational structures produced in Decon™, and analyzed in PAN™, lack the topological depth and formal foundations that would be required if the enterprise model were to admit the analytical inference of *semantic* ‘holes’ – that is, errors in the agents’ models that logically preclude the satisfaction of the enterprise’s needs, or that do not adequately account for the enterprise’s interactions with its world.

We seek to overcome this deficiency by constructing a formal semantics, in set theory, of Decon’s relational structure. We also expect to find that set theory will ultimately prove to be inadequate for this purpose because the enterprise is an inherently *anticipatory* system (Rosen 1987), whose model must therefore be *hyperincursive* (Dubois 1996), and therefore beyond the essentially *first-order* descriptive and analytical power of set theory. However, knowing exactly where, and why, a first-order system becomes inadequate when modeling an enterprise would itself be instructive to the analyst and illuminating to the agent because it represents the onset of that knowledge diffusion concerning the nature of the PAN/Decon™

methodology itself. It would also provide the theoretician with a clearer image of the ‘holes’ in mathematics itself.

Nevertheless, our theoretical foundations allow us to infer errors (i.e. inconsistencies) in such models, but not, in general, to derive corrections to such errors. Model alteration must therefore be considered as a fallible task to be continually subjected to refutation, just like scientific theory construction. The internal coherence of established models and systems builds up resistance to such changes in much the same way as it does in science. That resistance, which manifests itself as reluctance to consider alternatives, augments and may be indistinguishable from the social and political resistances that are usually associated with Enterprises. In that case, we would really be following the precepts of Freud and Lacan in considering the effects of the 'Symbolic' in relation to the 'Real' and the 'Imagined'.

A Semantic Domain for Stratification Analysis

Introduction

Our objective, then, is to provide a formal foundation (in set theory) for relational models of enterprises, which are constructed when the agents of an enterprise participate in its analysis. For the purposes of this study, we shall use the following working definitions:

- An **agent** is an entity that exists to satisfy its needs in a world populated by other agents. Every agent has its own theory of its needs which is a composition of its theories of itself and of other agents' needs, the latter including those other agents' theories of it. An agent is a formulation within second order cybernetics, in that it is a system that has a distinguishable identity, and an organization which manifests itself as revealed preferences in its behaviour (anticipative behaviour). An agent within third order cybernetics becomes a second order agent whose anticipative behaviour is organized in relation to a lack.... a point of instability/hyperincursion. (Boxer 1996.)
- An **enterprise** is an agent which offers services that it expects (according to its theory) to be valued by other agents in its world.
- An agent perceives a service to be of **value** to it if the service, as described, fulfils a demand that derives from the agent's theory of its needs; that is, that the provision of a service would satisfy a demand.

When an enterprise is approached in these terms, it becomes necessary to describe the internal organization of an enterprise, the ways in which this is related to the external organization of the market in which it competes, and the ways in which this external market organization is itself related to the organization of demand to which it is a response.

Supply and Demand Organization

We assume that:

- Supply organisation identifies *services*
  - One engages in supply when one deploys *processes*.
  - A process *provides* a service
  - By *organising* a set of services.

	Name	Type	Interpretation
1	SR	Set	Every service that might be provided to or organised by a process
2	PR	Set	Every process that might be deployed
3	prv	PR→SR	The (unique) service that each process *provides*
4	org	PR↔SR	The services that each process *organises*

- Demand Organisation identifies *specifications.* We may need to introduce an ordering relation, *weaker than*, on specifications, because any service that satisfies a specification should also satisfy all 'weaker' specifications. For completeness, the formal definition of this relation, and some of its properties, is given in Appendix 1.

- One expresses demand when one articulates *design specifications*
- A design specification *decomposes* a specification
- into *demands* for a set of specifications.

5	SP	Set	Every specification that might be demanded or decomposed by a design
6	DE	Set	Every design that might be articulated
7	dec	DE→SP	The (unique) specification that each design *decomposes*
8	dem	DE↔SP	The specifications that each design *demands*

- Every service *satisfies* some specification

9	sat	SR→SP	The (strongest) specification that each service *satisfies*

- A Process *realises* a design if, and only if:

10	rea	PR→DE	The (unique) design that each process *realises.*

- And imposes the following constraints on it:

11	(dec⊞rea)⊆(sat⊞prv)	The service provided by the process satisfies the specification decomposed by the design
12	(dem⊞rea)⊆(sat⊞org)	Each of the specifications in the design's decomposition is satisfied by one of the services organised by the process

Demand Organization

We can now consider 'demand organisation' to be the successive decompositions of designs that leads a potential services *user* to express a specification, thereby *situating* that specification in a user's 'value ladder'.

- Demand is expressed by *users*.

13	US	set	All potential *users* (in a problem domain)

- Users articulate designs. We do not formalise the relation between the refinements and demands of designs, i.e. the *validation* that a design is fit for its purpose. This would require the establishment of a sound decidable calculus for the composition of specifications, and there isn't one. The onus of design validation therefore lies with the customer in relation to whose demands designs are situated.

14	art	US↔DE	The designs *articulated* by each user.

- To specify a need as a demand without decomposing it is to define a *problem* if there is no service which satisfies that problem directly. This problem arises in a *demand situation.*

– Those specifications that a user's designs decompose but do not demand constitute the user's *demand situation* in terms of 'smaller' problems.

15	dst	US↔SP	The specifications that comprise each user's *demand situations.*
16			$dst = (dec \circ art) \backslash (dem \circ art)$

– Those specifications that a user's designs demand but do not decompose constitute the user's (c-type) customer situations. 'Value ladders' are defined here as directed, but not necessarily acyclic, graphs – of designs, from refinement through demand to further refinement, and so on. This ladder of designs for a service specification might include that same specification among its demands. For example, a customer whose purpose is to provide air transport services might refine that purpose into a design that demands air transport services, e.g. for the transportation of components to assembly plants.

17	cst	US↔SP	The specifications that comprise each user's *customer situations*
18			$cst = (dem \circ art) \backslash (dec \circ art)$

– Those users whose designs construct customer situations are *customers*.

19		$CU \subseteq US$	Those users who are customers
20			$CU = dom(cst)$

– The choice about whether or not to demand in this sense is equivalent to the decision to 'outsource'.

– If we think in terms of classes of demand as being some measure of 'distance' from the user's need, then it is possible for a supplier to service a higher-level specification while the user is still servicing a lower level one. (e.g. when the enterprise uses a design consultant to design a product that it will make itself.)

Supply Structure

Symmetrically, we consider 'supply logic' to be the chain of organisations of processes that enable a *position* to provide a service, thereby positioning that service in a 'value chain'.

- Supply is expressed by *positions*.

21	PO	set	All positions in an industry sector.

- Positions' capabilities are determined by the processes that they deploy. We do not formalise the relation between provision and organization of processes, i.e. the *verification* that a process achieves its intent. This is the traditional task of engineering, the onus of verification lying with the engineering staff of the competitor whose positions frame processes.

22	cap	$PO \leftrightarrow PR$	The processes that comprise the position's *capabilities*

– Those services that a position's processes provide but do not organise constitute the position's *value profiles*.

23	vpr	$PO \leftrightarrow SR$	The services that are 'framed' by each position, which constitute the position's *value profiles*.
24			$vpr = (prv \circ cap) \backslash (org \circ cap)$

– Those services that a position's processes organise but do not deliver constitute the position's *infrastructure*. Infrastructure presents constraints on the positions that a supplier might take. Symmetrically, we might identify *superstructure* with constraints that limit users' potential for articulation. This concept should have something to do with the substitutivity of specifications, and the services that satisfy them, in the context of particular demand situations (the property known as 'reuse' in the IS world). Formally, we would expect each design (or decomposition of a demand situation) to determine an equivalence relation on specifications.

25	inf	$PO \leftrightarrow PR$	The processes that constitute each position's *infrastructure*.
26			$inf = (org \circ cap) \backslash (prv \circ cap)$

Market Organization

In a market, competitors seek to supply their services to customers whose customer situations include specifications that they can satisfy. Symmetrically, customers seek to acquire services from competitors whose positions include services that satisfy their specifications. In general, customers deal with competitors' organisations who *order* services from positions that they control.

27	CO	set	Competitor organisations.
28	ctl	$PO \rightarrow CO$	The (unique) competitor who *controls* each position.
29	ord	$CO \rightarrow (PO \leftrightarrow SR)$	The positions that each competitor orders its infrastructure to take.
30		$(dom \circ ord) \subseteq ctl^{-1}$	Competitors can issue orders only to positions that they control
31		$ran(ord) = inf$	All, and only, the positions taken by a competitor's infrastructure (i.e. all the services in its value profiles) are ordered by the competitor.
32	sup	$(CoxSR) \leftrightarrow (UsxSP)$	The services *supplied* by competitors to customer situations.

33	dom(sup) $\subseteq$ ran°ord	Competitors can supply only those services that occur in their controlled positions.
34	ran(sup) $\subseteq$ cst	To customers whose situations comprise specifications
35	sup $\subseteq$ (PaxUS)*sat	that are satisfied by the services with which they are supplied.

Discussion for Future Development

Propositions and composition/decomposition within value ladders

Given our earlier formulations of agent, enterprise and value, services might be known to the agent to be available (at an acceptable price etc.), either from among its own services or from other agents. To be in a position to estimate the value of a service, an agent must have articulated a model of its needs, which thereby take the form of demands. These demands can themselves be a composition of service descriptions (but not necessarily if they are at the 'top' of the compositions for this agent, whose emergent properties, themselves composed with the agent's model of its world, predict a satisfying outcome). Such demands give rise to the simplest form of proposition – the c-type proposition:

- The **c-type** service offering, which supplies ‘off the shelf’ services. The supplier has no involvement in the decomposition of the client’s needs. The supplier’s concern is focused on its own operational effectiveness with respect to those services for which it has identified a ‘market’. What is sold is the ability to supply a service.

Those remaining demands, for which there is a demand but no supply, constitute a 'lack' in market organization, which is relative to the agent's current formulation of its needs, resources and market knowledge. Even for demands for which there are currently services, there is no unique decomposition of any purpose. One decomposition may reveal a lack in market organization which cannot apparently be satisfied, while others might reveal completely different lacks, or no lack at all. Equally, decomposition might be erroneous in that composition of the service descriptions in the model because:

(a) it does not induce the emergent properties required by the demand ('invalidity'), or

(b) induces mutual constraints on the (isolated) services which they cannot jointly satisfy ('inconsistency').

Or decomposition might be ineffective in that:

(c) some of the service descriptions in the model cannot be effectively supplied ('infeasibility'), or

(d) the composite model does not admit inference of its internal consistency or of its emergent properties ('intractability'), or

(e) the service descriptions in the model admit individual satisfaction by services that, when themselves composed, fail to meet the agent's purposes ('destructive interaction').

In all these cases, however, what is being offered as a service is a capability for composition:

- The **K-type** service offering, which is configured to deliver services in a form that is appropriate to the demand situation in which it believes its customer to be. The client defines the need in the form of a demand, but the supplier decomposes that demand – the problem - into a set of (better) satisfiable service specifications. What is 'sold' is the ability to decompose the problem into a form which renders it (more) tractable.

There may however be difficulties in the formulation of need itself. The enterprise may have needs which are not even formulated as demands, and which therefore constitute a 'lack' in demand organization.:

(i) The agent's theory of its needs may itself be erroneous so that the composition of a collection of services that satisfied a demonstrably valid and feasible model would, when deployed, fail to satisfy the agent's needs ('bifurcation')
(ii) Decomposition might be conceptually impossible in that the agent's needs are not themselves sufficiently well-formulated to admit decomposition into services (this comes closest to the notion of a lack in Demand organisation), or
(iii) the agent's conceptual framework is not rich enough to admit any effective decomposition (lack of elaboration of signifying networks).

Services provided to the agent which aim to enable the customer to formulate demands which are satisfiable and/or decomposable are P-type:

- The **P-type** service offering, which collaborates with a potential service customer in the specification of the latter's need as a demand, before (jointly) decomposing it into the form of services which itself or others can supply. What is 'sold' is the ability to articulate need in the form a demand for services which are believed to be satisfiable, in other words to define a problem.

The 'lack' in mathematics itself

Services provided to an agent which overcome compositional difficulties are K-type, and the providers of K-type services must be able to identify and classify these difficulties by reasoning over the composition of service specifications. However, service composition is not a formally definable operator; that is, it occurs in no logical calculus.

A related compositional operator can be defined for axiomatic systems provided that all such compositions are conservative extensions of their parts. However, the composition of service specifications is not, in general, conservative (i.e. it is not definable as an operator) unless those specifications are orthogonal. The result of an orthogonal composition is neither stronger or weaker than its components; that is, all services that satisfy each of the specifications s and t separately (i.e. in isolation) also satisfy the composition of s and t. However, the composition of service specifications may be either:

(a) restrictive, where a composite specification is stronger than at least one of its components, that is, although services p and q satisfied specifications s and t, respectively, in isolation, together they fail to satisfy the composition of s and t. Composition may be so restrictive as to result in the inconsistent specification, which is satisfiable by no service.

(b) emergent, where the composition of two specifications s and t possesses properties exhibited by neither in isolation.

Emergent composition is a major factor in the development of 'value ladders'. If a service specification ,s, to be provided by an enterprise, e, can be successfully decomposed into two orthogonal service specifications, p and q, then its clients may as well acquire services that satisfy p and q separately, rather than acquiring s from e. If, on the other hand, s is an emergent composition of p and q, then e 'adds value' to services that satisfy p and q by composing them.

However, an enterprise that conceives a design for some service specification, s, as the emergent composition of service specifications p and q may encounter difficulties if:

(a) services that satisfy p and q are not available, or

(b) such services are available but they do not, together, satisfy s.

The enterprise may respond to these situations in a number of ways. It might undertake (or commission) the development of either a service that satisfies s directly (a 'custom' system), or of services x and y that separately satisfy p and q, respectively, and together satisfy s (although this latter may not be apparent to the suppliers of x and y, nor to the customer for s). On the other hand, it might abandon that design of s and seek a different composition that is more effectively satisfiable.

Knowledge Diffusion

The know-how of an agent is a function of its ability to articulate satisfiable demand situations in relation to any given problem. For any given set of problems defined by a value ladder, knowledge diffusion is defined by the rate at which the population of agents with that know-how increases over time. Knowledge diffusion affects value through the way in which it affects the balance between supply and demand in any given problem domain. The way in which this balance changes over time is further affected by the architecture of the supplying enterprises. Thus, the articulation and promulgation of service specifications, and the demonstration of their emergent properties, constitutes 'knowledge diffusion', as these descriptions and demonstrations become part of the enterprise's model of itself.

Once the stratified nature of a value ladder is understood in terms of a layering of restrictive and emergent compositions, it therefore becomes necessary to distinguish the different kinds of supplier-customer relationship constituting the form of the ladder itself. Within this context, the successful discovery of an effective emergent composition is a developmental step (a 'radical break'?) for the enterprise because it makes possible the creation of new forms of value.

In conclusion

What has been formulated here is a **primary stratification** in the relations between:

[[[Infrastructure :: Enterprise]::Market Organisation]::Demand Organisation]::Needs

Each of these 'layers' have holes in them in relation to the layer 'above'. Thus needs give rise to holes in demand organisation, just as demands give rise to holes in market organisation, etcetera. Neither is it clear where the 'top' is, because not only is this primary stratification itself an effect of the observer, but agents have needs, and infrastructure is made up of agents. Thus the whole stratification has properties like those of the Klein bottle in the sense that inside becomes outside and vice-versa. Within this primary stratification, there is then a **secondary stratification.** This is seen in the layering of the value ladder and in the organisation of supply relationships as much as it is present in the relations between agents within infrastructure.

All 'anticipatory systems' are emergent composites, no behavioural property of the whole being attributable to any of its parts in isolation. 'Agents' as space-time bounded (imaginary) entities are formulated in relation to a semantic (symbolic) domain which is itself lacking. Thus, even though a compositional operator might describe the presence of a having-been-composed, composition itself becomes something of the Real, a tuché marking the presence of a lack in the semantic domain. An interesting, and open, question concerning the nature of 'enterprise' itself might therefore be to understand the sense in which all emergent composites might be anticipatory.

Appendix 1: The ordering relation on specifications

Specifications are ordered under the relation *weaker than:*

A1	wkr: SP↔SP	The specifications that each specification is *weaker than*
A2	wkr°wkr = wkr	This relationship is transitively closed and
A3	$wkr \cap wkr^{-1}$ = ide(SP)	Reflexive and antisymmetric, i.e. it is a *complete partial order.*
A4	vac:SP	The weakest (i.e. *vacuous*) specification, satisfied by all the services
A5	inc:SP	The strongest (i.e. *inconsistent*) specification, satisfied by no service, where
A6	wkr^{-1}[vac] = wkr[inc] = SP	Every specification is stronger than **vac** and weaker than **inc**

References

Boxer, P.J. (1993) *The Role of IT in Supporting Business Development Processes.* in Proceeedings of The Hewlett-Packard Computer Users' European Conference, Birmingham. **Error! Bookmark not defined.**

Boxer, P.J. (1996) *The stratification of cause: when does the desire of the leader become the leadership of desire?* in Lasker, G.E., Dubois, D., & Teiling, B. (Eds.) “Advances in Modeling of Anticipative Systems” Institute for Advanced Studies in Systems Research and Cybernetics, Baden-Baden. Pp106-110. **Error! Bookmark not defined.**

Boxer, P.J. (1995) “Freud’s Project and topologizing organization” GOWG working paper **Error! Bookmark not defined.**

Boxer, P.J. and Kenny, V. (1992) “Lacan and Maturana: Constructivist Origins for a 3^0 Cybernetics”, in *Communication and Cognition* Vol 25 No 1 pp73-100.

Boxer, P.J. and Wensley, J.R.C. (1997) *Performative Organisation: learning to design or designing to learn? (in press)* **Error! Bookmark not defined.**

Bunge, M. (1974-80) *Treatise on Basic Philosophy* (8 vols). D. Reidel.

Casti, J.L. (1994) *Complexification* Abacus

Cameron, J. and Cohen, B. (1996) *Formal Approaches to Feature Interactions*. Tutorial Notes, FORTE/PSTV’96, Kaiserslautern, 1996.

Cameron, J. and Velthuijsen, H. (1993) *Feature Interactions in Telecommunications Systems*. IEEE Communications Magazine, August 1993.

Cohen, B., Harwood, W., and Jackson, M.I. (1984) *The Specification of Complex Systems*. Addison Wesley.

Cohen, B. (1996a) *The Description and Analysis of Services as Required and Provided by their Agents.* **Error! Bookmark not defined.**

Cohen, B. (1996b) *Models and Modelling Frameworks in Health Care Informatics.* Proc ‘Towards an Electronic Patient Record’, San Diego and London.

Cohen, B. (1997) *Formal Modelling Frameworks of the Principles Governing the Confidentiality of the Patient Record*, Proc ‘Towards an Electronic Patient Record’, Nashville.

Dubois, D. (1996) *Concept and Method of Incursion and Hyperincursion.* Proc. CAST ’96.

Lacan, J. (1966) “La Science et la Vérité”. *Écrits*. Éditions du Seuil, Paris.

Lacan, J. (1988) “Logical Time and the Assertion of Anticipated Certainty” Translated by Bruce Fink and March Silver. Newsletter of the Freudian Field 2 (1988): 4-22. Originally written in March 1945, this was originally published in Écrits pp197-213. (1966).

Maturana, H.R. (1988) “Reality: The Search for Objectivity or the Quest for a Compelling Argument”. *Irish Journal of Psychology*, 1988, 9,1, p33

Rosen, R. (1987) *Anticipatory Systems* Pergamon

Schuman, S.A., Pitt, D.M. and Byers, P.J. (1990) *Object Oriented Process Specification*, in “Specification and Verification of Concurrent Systems” (ed Rattray), Springer-Verlag.

Van de Vijver, G. (1996a) “Internalism versus Externalism: a matter of choice”, *Japanese Journal of Contemporary Philosophy*. Pp93-101.

Van de Vijver, G. (1996b) “On the Origin of Psychic Structure: a case-study revisited on the basis of Freud’s *Project*” University of Ghent