Towards a Structured Approach to Activity Theory Analysis: Capturing and Representing Parties' Understandings in Business Process Modelling

G. Michael McGrath

College of Business, Victoria University, Melbourne, Australia

ABSTRACT

Sachs (among others) has convincingly argued that many process change failures have resulted from too narrow a focus and that a more holistic view of work is required. Thus, we believe it essential that effective process modelling approaches and tools provide adequate support for the so-called "softer" aspects. In this context, activity theory would appear to have much to offer and a process modelling approach, centred on this theory, is the major focus of this paper. Activity theory incorporates notions of intentionality, history, mediation, motivation, understanding, culture and community and, in particular, it provides a framework in which the critical issue of context can be taken into account. Unfortunately, however, the activity theory literature reveals very little in the way of prescriptive guidelines or tools that might be employed by practitioners. As a first step towards addressing this limitation, we employ activity theory, as a framework for the development of a process modelling methodology (plus an associated automated repository and analysis tools). A prototype repository has now been implemented and we report on its evaluation as part of a collaborative business process modelling exercise we are undertaking with a large Australian organisation. We focus particularly on the representation and analysis of "understandings" - a term we use to encompass motives, beliefs, norms, values and the like.

Keywords: Activity Theory, Process Modelling, People Aspects

INTRODUCTION

Effective business strategies and plans might be developed using intuition, sheer luck, prayer, astrology or, even, by reading tea leaves or the entrails of a chicken! Today, more than ever before, traditional strategic and operational planning processes and approaches are under siege. Plans are increasingly developed in highly uncertain and ambiguous environments, and the process itself is buffeted by numerous forces, including the new economics of information, forcing organisations to rethink the very strategic bases of their businesses. Despite these difficulties, however, it seems that to increase the likelihood of better results, and to partly fulfil the relentless need for control and predictability, an informed approach is probably advisable. Business process modelling can be invaluable in this respect. A sound process modelling tool can be used: to derive a model of the current business; to help pinpoint weaknesses in current processes and policies; to develop and test alternative models; and to derive the systems required to support new processes.

Many process modelling tools and techniques have been proposed and used in business planning projects. Most of these have been adapted from techniques employed in information systems analysis and design activities. Examples include flowcharts, data flow diagrams, state transition diagrams and functional decomposition diagrams. Newer and more powerful tools have been derived from recent research into systems dynamics, workflow analysis, computer-supported collaborative work and group decision support systems. In our view, all current tools and techniques have strengths and weaknesses but very few tools provide any useful support for the critical "people-related", organisational behaviour issues so critical to the successful implementation of any new business process or system. In this context, Sachs (1995), over 20 years ago, put forward convincing arguments that many business process reengineering failures have resulted from too narrow a focus and that a more holistic view is required. In particular, she argued that "business process reengineering --- fails to recognise how work is carried out and the part that human ingenuity plays in it" and that "(1) if only the organizational (explicit structures) of work are considered in designing work and (2) the importance of learning is left out, there will be negative consequences in the conception and implementation of [process] design". More recent studies would appear to indicate that things have not improved much in recent years, with up to 70% of projects failing (see e.g. Self and Schraeder, 2009; Weiner, 2009; Burnes and Jackson, 2011).

Existing business process modelling tools have, in broad terms, answered the who, whom, what, where and when of process modelling, but not the how or the why. Specifically, they have enabled the recording of the individuals and teams enacting a process, the artefacts produced, and the sequencing of the functional tasks, but not the norms, beliefs and motives of the staff carrying out the process: ie hard factors but not soft factors. The research discussed in this paper provides a promising means of readily allowing the capture of the critical softer[1] process data.

In this context, activity theory would appear to have much to offer. Activity theory has its origins in Russian psychology research (Vygotsky, 1978). Activity theory incorporates notions of intentionality, history, mediation, motivation, understanding, culture and community and it is these aspects that have proved attractive to information systems (IS) researchers - in particular, it provides a framework in which the critical issue of context can be taken into account. Unfortunately, however, a perusal of the activity theory literature reveals little in the way of precise, prescriptive guidelines or tools that might be employed by practitioners: in the sense, for example, that one might follow well-established rules in designing a dataflow diagram (Bokij et al., 2007) or undertaking a structured analysis and design exercise (Hoffer, George, and Valacich, 2013). As a step towards addressing this limitation, we have used activity theory, as the basis for the development of a process modelling methodology, plus an associated automated repository and analysis tools, called SATBPA (System for Activity Theory Business Process Analysis). A prototype repository has now been implemented and we are currently evaluating its effectiveness as part of a collaborative study with an industry partner. In our work to date, we have focused particularly on the representation and analysis of understandings - a term we use to encompass motives, beliefs, norms, values and the like.

Our paper is organised as follows: in the following section, we provide an overview of our research design. We then briefly introduce details of the study we are undertaking, in order to provide context for examples presented later in the paper. Following that, we introduce some key elements of activity theory and our own interpretation and automated implementation of the theory (SATBPA). In Section 5, we introduce Speech Act Theory and, in Section 6, we then demonstrate how the essence of this theory may be employed to facilitate the representation of the softer elements of business processes - the focus of our interest here. Concluding remarks are presented in Section 7.The paper is organized as follows: background to the project is provided in the following section and this is followed by a summary of a preliminary requirements elicitation workshop (held in late-2014), aimed at gaining a much deeper understanding of client help-seeking behaviour. The DSS design and specification is then overviewed and discussed and this is followed by concluding remarks.

RESEARCH APPROACH

A number of authors have addressed the issue of employing IS design and development as an authentic research method. Hasan (2003: 4) claims that IS development, in many cases, should be considered a valid research activity (and method) because, not only is knowledge created about the development process itself, but also because "a deeper understanding emerges about the organizational problem that the system is designed to solve". Markus et al. (2002) put forward a similar case in arguing that IS development is a particular instance of an emergent knowledge process (EKP) and that this constitutes original research where requirements elicitation, design and implementation are original and generate new knowledge on how to proactively manage data and information in complex situations. Hasan (2003: 6) further contends that this often involves a staged approach, where "systems evolve through a series of prototypes" with results of each stage informing requirements for the next and subsequent iterations.The principal output of this specific project is a DNA modelling and simulation architecture and associated DSS that extends traditional Social Network Analysis (SNA) approaches. Specifically, it allows: i) multiple modelling methods to be employed – particularly agent-based modelling (Borschev and Filippov, 2004), system dynamics modelling and simulation (Maani and Cavana, 2000) and traditional SNA matrix manipulation (Hanneman and Riddle, 2007); ii) through simulation, studies of how dynamic networks evolve and behave over time; iii) models specified using different approaches to be integrated and to share data; and iv) 'what if' type analysis, thus permitting the possible impacts of new policies and intervention strategies to be simulated and evaluated.

Nunamaker et al. (1991) draw on an alternative research tradition in case studies and, in particular, action research. Again, using 'replication' strategies, each new instance (case or action research activity) builds upon and refines knowledge gleaned from previous studies (Yin, 1994). Nunamaker et al. (op cit.), go on to note that IS development is enabled by the uniqueness of the technology employed (which can, as a tool, mediate knowledge generation and the communication of same). This feature has been studied extensively by scholars in 'activity theory'. Notably, activity theorists emphasize the holistic nature of the IS development process and, in particular, the critical nature of the cultural and social context within which systems are developed (Nardi, 1996). The socio-technical view of IS, where hardware, software, people and processes are integrated into a complex, purposeful whole, is one of the key features that make information and communication technologies "like no other in the history of mankind" (Hasan, 2003: 4).

With respect to DSS design and development, Pornphol and McGrath (2011) argue that this is a legitimate research activity in its own right, which draws on the more established, traditional research approaches of the design sciences and especially case study/action research. Each new application of a DSS produces a new prototype version and extends knowledge of the research domain. This is akin to employing a multi-case (study) research strategy - with each new case refining and extending results of previous iterations - and finally, many research findings and outputs are actually inherent in the various conceptual models (and implementations of these) that constitute the DSS.

As a (partial) consequence of - and in parallel with - these studies, the last two decades have seen the rise of the 'Design Science' (DS) research approach; sometimes described as the 'Design Science Research Methodology' (DSRM) (Peffers et al., 2007-08). As far back as the 1960s, Simon (1981: 55) argued "Whereas natural sciences and social sciences try to understand reality, design science attempts to create things that serve human purposes". Hevner et al. (2004) proposed seven guidelines for conducting DS research. These are: i) that the research must produce an "artefact created to address a problem"; ii) the artefact should be relevant to the solution of a "heretofore unsolved and important business problem"; iii) its "utility, quality, and efficacy" must be rigorously evaluated; iv) the research should represent a verifiable contribution; v) rigor must be applied in both the development of the artefact and its evaluation; vi) the development of the artefact should be a search process that draws from existing theories and knowledge to come up with a solution to a defined problem; and, finally, vii) the research must be effectively communicated to appropriate audiences.

Peffers et al. (2007-08) reviewed much of the DS research and specified the process model presented in Figure 1 as a consensus of the better-known DSRMs proposed to that point in time. This represents the research design of our study, further detail on which is presented in the remainder of this section.

The problem identification and motivation stage involves defining the research problem and justifying the value of a solution (Peffers et al., 2007-08). In this instance, the problem and its importance (c.f. motivation) were detailed in the previous section, where it was noted that: i) the change process modelling and implementation successrate is poor; ii) improvements are (at least partially) dependent on addressing softer factors adequately; iii) activity theory can help here but a more prescriptive approach is required; and iv) SATBPA is a DSS designed for precisely this purpose. Thus, in terms of the DS research guidelines detailed by Hevner et al. (2004), the project is important, it addresses a previously-unsolved problem and it is original: particularly with regard to formal and rigorous modelling of key AT concepts and the DSS implementation of same. In addition it is certainly non-trivial (see the following sections).

As noted by Peffers et al. (2007-08), DS researchers often combine the first two stages (problem identification and objectives specification) and this is the approach taken here. The third stage is system design and development an during this stage the aretefat is created. Again, according to Peffers et al. (2007-08: 53):

"Such artifacts are potentially constructs, models, methods, or instantiations (each defined broadly) ----. Conceptually, a design research artifact can be any designed object in which a research contribution is embedded in the design. This activity includes determining the artifact's desired functionality and its architecture and then creating the actual artifact."

We maintain that a major research contribution of this project is the scheme we have developed for capturing, classifyingand analysing the results of AT analyses exercises and, in particular, understandings. The SATBPA database schema and

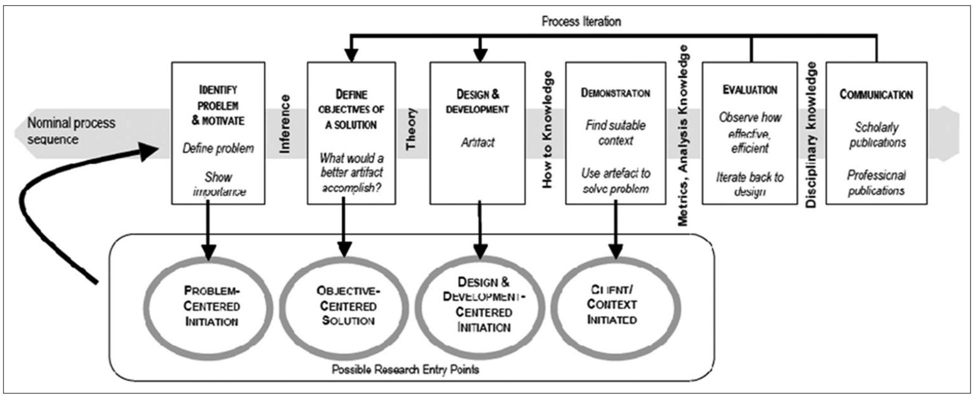

Figure 1: DSRM process model (reproduced from Hevner et al., 2004: 52)

accompanying very high-level *Prolog* code may be viewed as an 'embedded' DSS specification of this contribution and these are detailed in Sections 4-6 of this paper.

Stages four and five are demonstration and evaluation respectively and these are accomplished through use of the SATBPA DSS to support the AT analysis undertaken during the case study detailed in the following section. The final stage is communication where, according to Peffers et al. (2007-08: 54), artefact designers and developers must:

"Communicate the problem and its importance, the artifact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences, such as practicing professionals, when appropriate."

This, of course, is the primary objective of this paper.

A BRIEF CASE STUDY

Gigante is a large Australian company involved in, what broadly may be described as, the information services provision (ISP) business. It is suffering loss of revenue due to mistakes being made by its Customer Service Officers (CSOs) in taking and processing customer orders. The CSO workload is undoubtedly very high and this is a major contributor to their problems, which are exacerbated by a lack of training, new products being introduced at a rapid rate, poor process management and documentation procedures and inadequate performance measures[2]. The organisation's information systems are poorly integrated and this is, certainly, another significant reason cause of the high error rate. Our focus in undertaking this study, however, was on 'softer', people-related problems.

In undertaking this field study, initial modelling was conducted using the 'system dynamics' approach and its tools (Forrester, 1961; Vennix, 1996). Customers place an order with a *Customer Service Centre (CSC),* using an information system or via a telephone call, and request one or more information services. One CSO deals with a customer's request, ascertains the customer's needs and translates these needs into an order, comprised of one or more products (identified by product codes). The CSO then completes initial processing of the order within a provisioning system (separate information systems handle the provisioning and billing functions). Because there are several hundred products (and even more product codes), many errors occur at this point - particularly for newer products, the more complex product offerings and for products ordered less frequently. Product codes entered into the provisioning system are translated into: 1) service codes and sent to the billing system; and 2) network reconfiguration data, which is then forwarded to the engineers responsible for maintaining Gigante's own internal network. In many cases where an incorrect product code is entered, the network will, nevertheless, be configured correctly. From this time, the customer has access to the requested services. Normally, each time an information service is utilised, a call charge record (CCR) is debited against the customer's account. However, where there is an incorrect service code in the billing system, CCRs will be rejected and a bill for the revenue involved cannot be issued until the original provisioning error is corrected (and rejected CCRs re-entered).

Gigante operates in a highly-competitive market, regulatory and market conditions have virtually forced it to adopt a product differentiation business strategy and, consequently, it releases new products on a regular basis. To cope with new products CSOs must be trained. If, however, *actual training* (both formal courses and on-the-job) is inadequate, the knowledge gap increases (the *knowledge gap* is the difference between the knowledge *required* for CSOs to do their jobs effectively and efficiently and their *actual* knowledge). As the knowledge gap grows, the number of *rejects* (specifically, CCR rejects) increases. This has several impacts: 1) CSOs have to work harder which, ultimately, may lead to even more rejects; 2) this *workload* increase may result in greater staff *turnover,* replacement staff will (initially) not be as effective as their colleagues and, hence, further rejects are again the result; 3) the workload increase also means that staff have less time available for training, thus exacerbating the rejects problem even more; and 4) as the rejects level rises, *customer satisfaction* decreases. This leads to more *customer contact* (enquiries and complaints) which, in turn, further adds to the workload and, again, leads to higher staff attrition, less time available for training etc. Note also that too many unhappy customers results in an increase in the level of *defections* to Gigante's competitors, thus causing a significant *revenue loss.*

The costs of all this to Gigante are substantial. At times, the situation has become so bad (as evidenced by the size of the rejects database) that exceptional, "one-off" corrective action has had to be undertaken. Essentially, this involves clearing out the rejects database of certain CCR categories (e.g. CCRs of specific types older than 9 months) and the consequent loss of all revenue associated with the deleted CCRs. The organisation's policy is not to bill a customer for any debit older than 12 months and this accounts for a little less than 1% of all CCRs. Finally, on average, it takes 4 months to clear an error and this delay costs the organisation further - in terms of lost interest earnings opportunities[3].

Using the simulation capabilities of the system dynamics software package *ithink™,* the project team was able to demonstate the negative impacts of factors such as poor training, excessive management pressure and high levels of staff turnover. However, we did experience a number of difficulties.

Firstly, *ithink™* system dynamics models do not scale up all that well (and the same appears to be the case with other system dynamics tools). It is true that system dynamics is of most use at the strategic level and, as such, lower-level detail is rarely included in models. Even so, the scope of our study is such that our model has grown to an unwieldy size and the *ithink™* restriction, that sub-models may only be taken down one level, has caused us some difficulty (the same seems to apply to all other system dynamics modelling tools we have investigated).

Secondly, like most process modelling approaches, system dynamics is very coarse-grained. As noted by Dellen, Kohler and Maurer (1996), fine-grained process modelling is sometimes necessary in order to capture details of specific parties, resources, events and dependencies between them. This occurred on a number of occasions during our modelling sessions at Gigante. For example, on one site visit it was revealed that local work structure arrangements (even down to the level of which employees work well together) can have a major impact on error rates. The modelling team considered this information to be absolutely vital but it is difficult to see how it might be captured (at least, at a fine-grained level) using a system dynamics approach. The same applies to much other material brought up, relating to individual parties, systems, and their roles and dependencies.

As noted above, a great deal of the relevant domain information that remains to be captured requires a much more fine-grained modelling approach than we have employed to date. Furthermore, whatever approach is employed must be capable, both, of representing the softer aspects of the processes being studied and of coping with the (considerable)

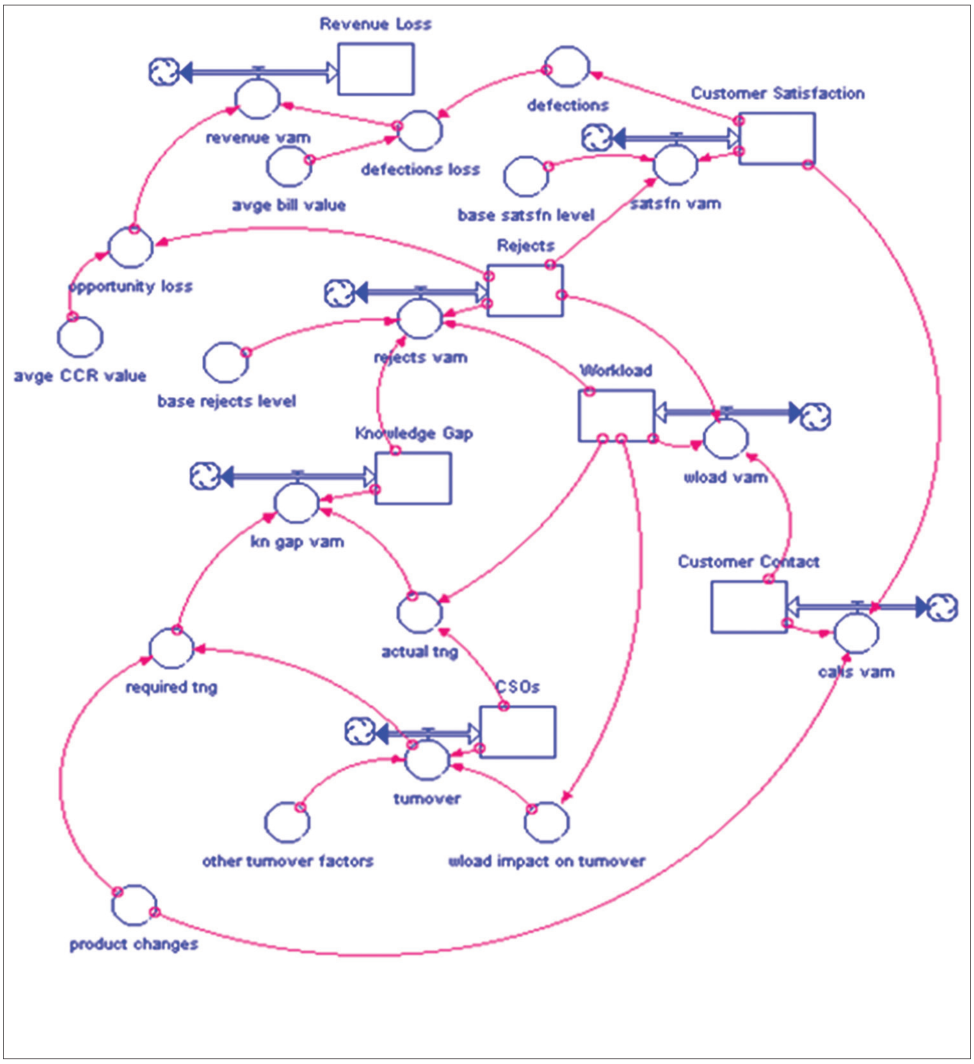

Figure 2: A system dynamics view of Gigante's ISP process

scope of our problem domain. In this context, activity theory would appear to have much to offer and, in the remainder of the paper, we turn our attention to our efforts to utilise this theory as the basis for a fine-grained business process modelling approach.

ACTIVITY THEORY AND SATBPA

In this section, we briefly introduce activity theory and our own interpretation and automated implementation of some key elements of the theory. For a detailed introduction, the reader is referred to Kuutti (1996). In classical activity theory (Vygotsky, 1978), an *activity* involves a *subject* working on an *object,* using a *tool,* to produce an *outcome.* A simple example, taken from our case study, is a customer purchasing an information service, via an information system, with the aim (desired outcome) of being able to utilise a working product. Looked at through the lens of the REA (resources, events and agents) abstracted modelling approach of McCarthy (1982), this may be viewed as: i) a *party-resource involvement (pri)*, where the party is the customer and the resource the information service; ii) a *resource-party-resource involvement (rpri),* where the additional resource in this compound structure is the information system employed as the activity tool; and finally iii) an *event-resource-party-resource involvement (erpri),* where the event is the activity outcome (a delivered, working information product).

The order in which these compound structures are built varies with the activity itself and will be impacted by analyst subjectivity. Abstracting our general activity model further though, produces the template presented in Figure 3. At the core of the template is the simple, 1st-level involvement, $i_1(E_{11}, E_{12})$, where E_1 and E_2 are entities, each of which must be one of a limited number of types *(resource, party, event and activity),* and i_1 is an involvement role, ideally expressed as a 1st-order logic predicate, with the declarative interpretation, $E_{11}\ i_1\ E_{12}$ (Kowalski, 1979)[4] . Examples are:

customers place information-service-orders;

CSOs take information-service-orders; and

information-service-orders are-placed-via (an) information-system.

At the next level, we have the term, $i_2(E_{21}, i_1(E_{11}, E_{12}))$. Kuutti (1996) suggests that activities may be broken down into actions and, then, into operations. Within SATBPA, we do not restrict ourselves to three levels and, instead, allow n-level hierarchies, where higher-level structures resursively operate upon lower-level structures, all of which we refer to as activities. Following from this, lower-lever activities may be ‘rolled up’, labelled and then used at higher levels. Thus, we might label the first of the above examples as a_{11} and used in the following 2nd-level activity:

CSOs take-orders-in a_{11}.

Similarly, labelling the above as activity a_{21}, we could employ this in a 3rd-level activity, with *information-system* as the subject and *processes-orders-in* as the involvement role. More generally, activities at level *n* have the form: $i_n(E_{n1}, i_{n-1}(E_{n-11}, \text{----} i_1(E_{11}, E_{12})\text{----})$.

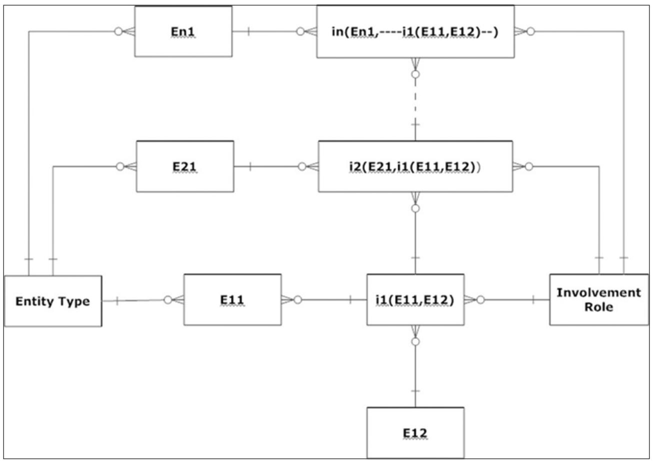

Figure 3: Activity modelling – entity-relationship (Chen, 1976) form

The final elements that must be taken into account in activity theory modelling are the division of labour and the norms, beliefs, values etc. (rules) that guide the community. We refer to these as *understandings.*

SATBPA is an automated tool that provides an interface to a database designed to record the detail of business processes in a form consistent with activity theory. SATBPA is intended to be a first step towards the development of a methodology for the structured application of activity theory in business process modelling. This addresses a major problem currently confronting potential activity theory practitioners — namely a distinct lack of any clear, prescriptive guidelines. A SATBPA user interface screen is presented in Figure 4. The screen in question is desiged to assist with the capture and analysis of understandings. We discuss this further in the following two sections.

SPEECH-ACT THEORY

Our scheme for representing understandings is based largely on the *Formal Language for Business Communication (FLBC),* detailed by Kimbrough and Moore (1997). Their language has its origins in *Speech Act Theory (SAT)* (Austin, 1962) and, in our view, they present convincing arguments (at a pragmatic level and within the restricted systems domain under consideration) for the superiority of FLBC over competing approaches (notably EDI languages and standards, tagged messages and natural language understanding).

At the core of FLBC is the *F(P)* framework. *P* is a proposition and *F* is described as an *illocutionary force.* Essentially, illocutionary forces provide context for propositions and typical examples quoted by Kimbrough and Moore (1997) include beliefs, desires and intentions. Their list of illocutionary points consists of *assertive* (statements of fact), *commissive* (promises), *directive* (commands), *declarative* (performatives) and *expressive* (attitudes). For example, the activity specification, *CSOs are responsible for service order entry*, represents the FLBC proposition, *P.* Stated blandly, the proposition may simply be an assertion of an organisational truth. If, however, the speaker is the CSC manager and the statement is made to the CSO team, heavily emphasised and with voice raised, then the propositional content may be interpreted by all CSOs present as a directive (i.e. to do better with their service order entry work) and this is the illocutionary force.

Constructs within the *F(P)* framework may be nested to any level and combined using a limited number of connectives. This is the source of the expressive power of the FLBC. In the following section, we illustrate that these facilities are precisely what are needed to enable the representation of the understandings that are central to our SATBPA interpretation and implementation of activity theory.

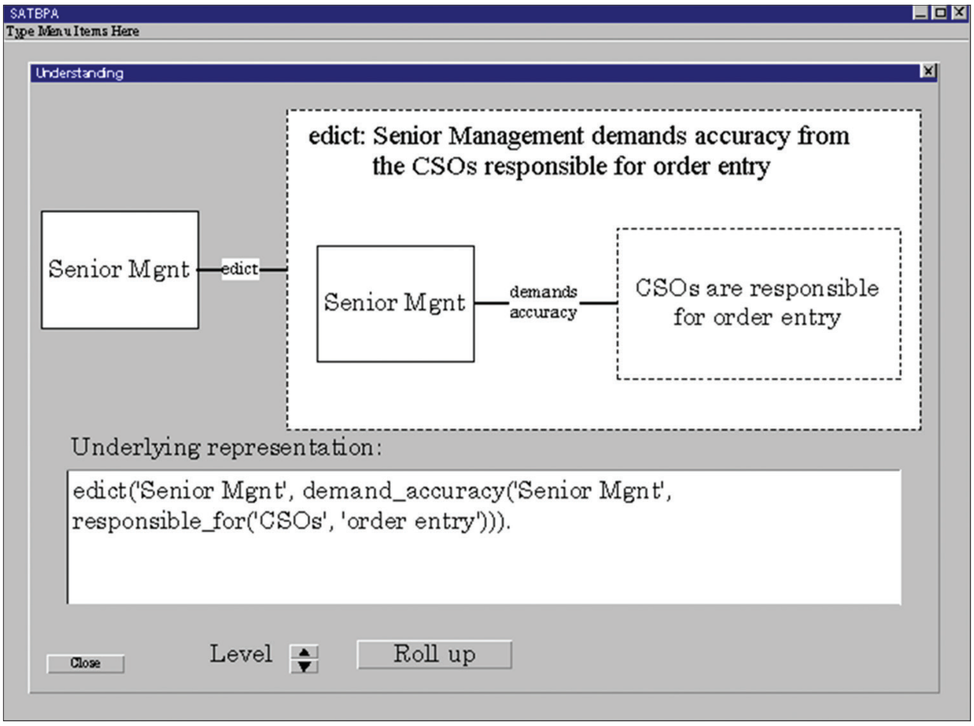

Figure 4: SATBPA screen example – representation of an understanding

Note, however, that we do not contend that SATBPA is a realisation of the (theoretical) universality claim[5] of SAT. On the contrary, our objective is to strike a reasonable balance between expressive power and structure. That is, to capture the critical contextual information, central to activity theory, a representation scheme considerably more powerful than that found in most information systems is required. On the other hand, a scheme with too few representational constraints may greatly limit the amount of convenient, useful analysis that can be performed on captured repository data. The *F(P)* framework does seem to provide a basis for this balance. When expressing *P,* the analyst may choose from an infinite number of predicates, but is restricted to a limited range of illocutionary forces (in our case: *norm, belief, value* (policy) *edict, motive, goal* and few others). Furthermore, within SATBPA, the user is further constrained by having to employ our data model, abstracted, E-R types (*party, resource, event* and *activity*). Whether we have, indeed, struck the right balance or not can only be confirmed by further applied field research. Early results from the Gigante study are encouraging.

REPRESENTATION OF UNDERSTANDINGS

Understandings are specified using precisely the same modelling template employed for base activities (see Figure 3): all that is required is to allow the use of illocutionary forces as involvement roles. In this way, we can gradually develop the nesting of propositions and illocutionary forces referred to in the previous section. The process can be extended indefinitely. Consider the example briefly discussed earlier, where the Gigante senior management demand that orders be processed accurately was hardly consistent with the CSO performance measure based on minimising customer wait-time. CSOs are responsible for order processing. This is an instance of a (relatively) low-level *pai (party-activity involvement)* activity relationship and, as such, may be expressed as the proposition:

responsible-for(CSOs, order-entry).

At the next level, senior management's demand for accuracy can be represented as the higher-level *ppai (part-party-activity involvement* relationship:

demand-accuracy(Senior-Mgnt, responsible-for(CSOs, order-entry)).

The illocutionary force at work here is an edict, which we may, in turn, denote as:

edict(Senior-Mgnt, demand-accuracy(Senior-Mgnt, responsible-for(CSOs, order-entry))).

Thus, we have established a nested SAT structure of the form, *F(P(P)).* Coincidentally, the minimum-wait-time performance measure (effectively, a senior management edict as well) may also be represented, using the same SAT structure, as:

edict(Senior-Mgnt, demand-min-wait-time(Senior-Mgnt, responsible-for(CSOs, order-entry))).

A major disadvantage of this representation scheme is that many users do not feel comfortable dealing directly with formal logic. We have, however, designed a user interface that permits the business process analyst to manipulate SATBPA database entities (plus other entities within the activity theory universe) by employing the usual GUI-style operations (see Figure 4). Note that the analyst may choose to manage complexity within SAT structures by a *Roll Up* of any substructure at any point in a modelling session and, thereafter, by simply referencing the identity of the rolled up substructure. Alternatively, the analyst may choose to work directly with the underlying logic and relational database.

The analyst may also *Roll Down* a substructure, in order to display its detail in visual form. This *Roll Up/Down* capability is facilitated to some extent by the binary trees which underpin our SAT structures. For example, referring again to the first edict above, its binary tree representation is illustrated in Figure 5. Note that here we have superimposed a new node on top of the edict itself, which means that the entire edict has been rolled up. The superimposed node enables labelling and identification of the rolled-up structure, with *C_id* standing for *condition/conclusion identifier* and the right-hand leaf, *C,* representing the actual value of the structure identifier. The *condition/conclusion* label was selected because, as we shall now see, any of our SAT structures may be employed as both conditions and conclusions in understandings.

Understandings may now be specified using the four binary relationships:

isa(U_id, understanding)

subject(U_id, S)

conclns(U_id, C)

condns(U_id, C')

where *U_id* is the understanding identifier, *S* is the subject (or "understander") and *C* and *C'* represent lists of conclusions and conditions respectively. Both these list have the form:

$c_0(C_0, c_1(C_1, --------, c_{n-1}(C_{n-1}, C_n))----)$

where c_0,----,c_{n-1} are logical connectors and each member of the set $\{C_0,-----,C_n\}$ is either a SAT structure or the identifier of a rolled-up version of the same. The conclusions list must contain at least one entry, while the conditions list may be empty. Again, both lists may be represented as binary trees. Thus understandings are effectively binary tree structures, with connectors at each major node and a SAT structure at each leaf or branch. A major benefit gained from this representation scheme is that trees (along with lists) are the data structures natural to *Prolog,* the language we employ to manipulate and analyse understandings. Note also that, as a simplification, we employ the standard *Prolog* list structure, $[C_0,----,C_n]$, to represent a sequence of conclusions or conditions connected by *"and"*.

Example

Gigante has gradually been centralising its order entry operations - an initiative prompted by senior management's belief that this will result in improved efficiency. Using the methods detailed above, we may specify the following condition/conclusions:

is(C1, centralisation(Senior-Mgnt, responsible-for(CSOs, order-entry)))

and

is(C2, belief(Senior-Mgnt, more-efficient-at(CSOs, order-entry))).

C1 simply asserts that senior management is centralising the order entry function (and, consequently, the CSOs responsible for that activity), while *C2* is the representation of senior managements' belief that the efficiency of the CSOs involved in the order entry function can be improved. CSOs, however, have expressed a great deal of skepticism concerning the centralisation initiative; specifically, it is their view that senior management's primary focus is on staff reductions. If we now add the additional rolled-up conclusion:

is(C3, belief(CSOs, will-be-fired(CSOs, order-entry)))

which is the representation of the CSOs' belief that their jobs are not all that secure, then, we may now specify two understandings:

isa(U1, understanding) *isa(U2, understanding)*

subject(U1, Senior-Mgnt) *subject(U2, CSOs)*

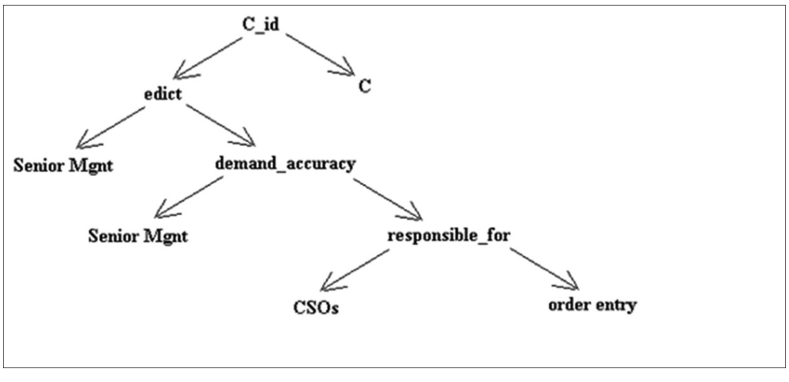

Figure 5: Binary tree representation of SAT structures

conclns(U1, [C2]) *conclns(U2, [C3])*

condns(U1, [C1]) *condns(U2, [C1])*

Thus, on the surface, senior management and CSOs have different understandings of the impact of the order entry centralisation programme. This, however, does not imply a contradiction. In fact, since it was quite evident from interviews we conducted with Gigante's senior management that they believed the desired efficiency improvements would come about as a result of staff reductions, a more complete version of senior management's understandings might now be represented as:

isa(U3, understanding) *isa(U4, understanding)*

subject(U3, Senior-Mgnt) *subject(U4, Senior-Mgnt)*

conclns(U3, [C3]) *conclns(U4, [C2])*

condns(U3, [C1]) *condns(U4, [C3])*

In effect, the CSOs understanding of the centralisation initiative was that it would result in job losses $(C1 \rightarrow C3)$. Senior management's understanding simply extended the consequential links to $C1 \rightarrow C3 \rightarrow C2$ (centralisation leads to job losses which, in turn, leads to efficiency improvements). Senior management's reluctance to acknowledge the middle link in the chain is perfectly understandable and is, in fact, an excellent example of the use of language as an organisational power/political tool (Pfeffer, 1992).

SUMMARY

We have presented details of a company that is losing a great deal of revenue due, in no small part, to poor coordination and collaboration between the many parties involved in its customer orders and billing functions. To gain a deeper understanding of the problem, we have developed a model of the relevant processes using system dynamics. This model has, indeed, improved our level of understanding and even allowed us to experiment with different workflow arrangements by utilising the built-in simulation capabilities provided with the automated modelling software we have employed. Also, with the system dynamics version, we have been able to capture and analyse some of the critical softer factors that must be addressed if the company is to satisfactorily resolve its problems. Finally, we have identified the need for both a much more fine-grained process modelling approach and for facilities that allow a more comprehensive range of softer factors to be captured, manipulated and analysed. We have proposed activity theory as the basis for one possible way forward here and detailed a specific approach (SATBPA). Our broad aim is to empower business analysts, change managers and others involved in process planning, redesign and implementation to better deal with the critical human aspects inherent in all non-trivial change initiatives.

Our major research output is the SATBPA tool: a DSS designed to capture the results of business process modelling exercises conducted using activity theory. A key feature of this system is that it is ale to represent parties' understandings: a term used to encompass norms, beliefs, values etc. Consistent with the design science research approach, much of the knowledge gained in this project is contained in the DSS artefact itself: particularly the data structures (including the logic-based rules) and the system databases. Moreover, the artefact will undergo refinement and continue to evolve as the DSS is employed in support of further AT-based process modelling exercises in the future.

END NOTES

1. By "softer" we broadly mean "behavioural", ranging from the individual level (encompassing attitudes, opinions, beliefs, knowledge, skills etc.) to the organisation-wide level (where we include aspects such as power and politics, organisation culture and learning).
2. At the commencement of our study, CSOs were measured mainly on customer wait-times - a performance measure hardly consistent with the need for accuracy (discussed further in Section 6).
3. We are unable to cite our sources without divulging Gigante's actual identity. However, we can reveal that there is substantial public pressure on Australian information service providers in general, not to bill customers for any debits greater than 3 months old.
4. Among other advantages (including the provision of a reasonably-clear declarative semantics), this allows convenient

manipulation in the artificial intelligence programming language, *Prolog* (Bratko, 1986).

5. The universality claim is that all that can ever be said (ever, by anyone and with whatever intent) can be captured by some (proposed or actual) operationalisation of speech-act theory.

REFERENCES

Austin, J.L. (1962): *How to Do Things with Words*, Oxford, UK: Clarendon Press.

Bokij, P., Chaffey, D., Greasley, A. and Hickie, S. (2006). *Business Information Systems (3rd ed.),* Edinburgh, UK: Pearson Education.

Bratko, I. (1986). *Prolog Programming for Artificial Intelligence,* Wokingham, UK: Addison-Wesley.

Burnes, B. and Jackson, P. (2011). "Success and failure in organizational change: An exploration of the role of values", *Journal of Change Management,* 11, 133-162.

Chen, P.P.S. (1976). "The Entity-Relationship Model: Towards a Unified View of Data", *ACM Transactions on Database Systems,* Vol.1, No.1, 9-36.

Dellen, B., Kohler, K. and Maurer, F. (1996). "Integrating software process models and design rationales", in *Proceedings of KBSE'96* (IEEE Computer Society, U. S.), 84- 93.

Forrester, J.W. (1961). *Industrial Dynamics,* Cambridge, MA: MIT Press.

Hasan, H. (2003). "Information Systems Development as a Research Method", *Australasian Journal of Information Systems,* Vol.11, No.1, 4-13.

Hevner, A.R., March, S.T. and Park, J. (2004). "Design Research in Information Systems Research", *MIS Quarterly,* 28(1), 75-105.

Hoffer, J.A., George, J. and Valacich, J. (2013). *Modern Systems Analysis and Design,* Pearson Education, Edinburgh, UK.

Kimbrough, S.O. and Moore, S.A. (1997). "On automated message processing in electronic commerce and work support systems: Speech act theory and expressive felicity", *ACM Transactions on Information Systems,* Vol.15, No.4, 321-367.

Kowalski, R.A. (1979). *Logic for Problem Solving,* North Holland, New York.

Kuutti, K. (1996). Activity theory as a potential framework for Human-Computer Interaction research. in Nardi, Bonnie A. (ed.), *Context and Consciousness: Activity Theory and Human-Computer Interaction*. MIT Press, Cambridge, MA, 17-44.

McCarthy, W.E. (1982). "The REA Model: "A Generalized Framework for Accounting Systems in a Shared Data Environment", *The Accounting Review,* Vol.57, No.3, 554-578.

Markus, L., Majchrzak and Gasser, L. (2002). "A Design Theory for Systems that Support Emergent Knowledge Processes", *MIS Quarterly,* Vol.26, No.3, 179-212.

Nunamaker, J.F., Chen, M. and Purdin, T. (1991). "Systems Development in Information Systems Research", *Journal of Management Information Systems,* Vol.7, No.3, 89-106.

Peffers, K., Tuunanen, T., Rothenberger, M.A. and Chatterjee, S. (2007-08). "A Design Research Methodology for Information Systems", *Journal of Management Information Systems,* Vol.24, No.3, 2007-08, 45-77.

Pfeffer, J. *Managing with Power: Politics and Influence in Organizations*, Harvard Business School Press, Boston, MA, 1992.

Pornphol, P. and McGrath, G.M. (2011). "Information Systems Development as an Emergent Knowledge Research Process", *International Journal of Interdisciplinary Social Sciences,* Vol.6, No.1, 301-317.

Sachs, P. (1995). "Transforming work: Collaboration, learning and design", *Communications of the ACM* 38, Vol.9, 36-44.

Self, D.R. and Schraeder, M. (2009). "Enhancing the success of organizational change: Matching readiness strategies with sources of resistance", *Leadership and Organization Development Journal,* Vol.30, 167-182.

Simon, H. (1969). *The Sciences of the Artificial,* MIT Press: Cambridge, MA.

Vennix, J.A.M. (1996). *Group Model Building: Facilitating Team Learning Using System Dynamics,* Wiley, Chichester, UK.

Vygotsky, L. S. (1978). *Mind and Society*. Harvard University Press, Cambridge MA.

Weiner, B.J. (2009). "A theory of organizational readiness for change", *Implementation Science,* Vol.4, No.67, 1-9.