

Association for Information Systems AIS Electronic Library (AISeL)

UK Academy for Information Systems Conference
Proceedings 2009

UK Academy for Information Systems

3-31-2009

Process Architectures In Higher Education

Ian Beeson

The University of the West of England, Bristol, UK., ian.beeson@uwe.ac.uk

Stewart Green

The University of the West of England, Bristol, UK., stewart.green@uwe.ac.uk

Richard Kamm

The University of Bath, Bath, UK., r.m.kamm@bath.ac.uk

Follow this and additional works at: <http://aisel.aisnet.org/ukais2009>

Recommended Citation

Beeson, Ian; Green, Stewart; and Kamm, Richard, "Process Architectures In Higher Education" (2009). *UK Academy for Information Systems Conference Proceedings 2009*. 11.
<http://aisel.aisnet.org/ukais2009/11>

This material is brought to you by the UK Academy for Information Systems at AIS Electronic Library (AISeL). It has been accepted for inclusion in UK Academy for Information Systems Conference Proceedings 2009 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

PROCESS ARCHITECTURES IN HIGHER EDUCATION

Ian Beeson

The University of the West of England, Bristol, UK.

Email: ian.beeson@uwe.ac.uk

Stewart Green

The University of the West of England, Bristol, UK.

Email: stewart.green@uwe.ac.uk

Richard Kamm

The University of Bath, Bath, UK.

Email: r.m.kamm@bath.ac.uk

Abstract

We have used the business process modelling method 'Riva' to model processes of programme management in two UK universities. The method depends on the identification of 'essential business entities' as the basis for defining a process architecture. The author of the method claims that organisations in the same business will have the same process architecture. In two attempts to produce process architectures for our case organisations, we could not produce any convergence in the outcomes. The exercise was however useful, as is the method. We make some suggestions regarding a core architecture for the area of activity under study, and make some observations on the method and the concepts used in it.

Keywords: process architecture, essential business entity, Riva.

Topic: IS methods and modelling.

1 Introduction

Organisations may try to improve their efficiency or effectiveness by identifying and modelling their most important business processes, and then use the models as a basis for the redevelopment of these processes. Piecemeal approaches that model one process at a time, as needs or opportunities arise, are unlikely to produce a coherent set of streamlined processes such as might be necessary for an organisation to meet its strategic objectives. Harmon (2003) recommends a coordinated approach in which a *process architecture* is built before individual processes are selected, modelled and supported by IT. The architecture could then be used to select which processes to develop, in which order, and to guide their development so as to ensure appropriate interrelation.

A number of process architecture methods have been proposed to date. For example, Kavakli and Loucopoulos have proposed a method based upon organisational goals (Kavakli and Loucopoulos,1997), Snowdon and Kawalek have proposed a method based upon Stafford Beer's Viable System Model (Snowdon and Kawalek, 2003), and Lunn has described a method involving the development of process maps (Lunn et al., 2003). Another method, *Riva* (Ould, 2005), identifies an organisation's process architecture according to the entities that comprise the organisation's principal units of work. For example, for Higher Education, entities might include programme, modules and, students, and a corresponding *Riva* process architecture would be derived from these. Ould makes the strong assertion that "a *Riva* process architecture is an invariant for an organisation that stays in the same business" (2005, p.171)): in other words, two organisations in the same business will have the same process architecture.

The work discussed in this paper describes an application of *Riva* to the delivery and development of undergraduate and postgraduate courses in two neighbouring UK universities. Two particular questions are addressed:

- What issues arise in using *Riva* to develop process architectures for this area of work in the two organisations?
- Are the resultant architectures substantially the same?

2 The *Riva* Approach to Process Architecture

The fundamental concept of the *Riva* approach is that an organisation's process architecture can be built up from the essential entities that are its subject matter, i.e. the entities that characterise the business. If an insurance company – to take one example of a type of business organisation - is characterised by having customers, policies, and claims, among other things, then the process architecture for this organisation should be based directly on these essential business entities. It is further assumed in *Riva* that organisations in the same business will have the same essential business entities (EBEs) and hence the same essential process architecture. Because they do things in different ways, organisations in the same business may differ in the

secondary entities they work with, and therefore in the detail of their fully elaborated architectures.

Riva is an attractive approach for a number of reasons:

- It provides a clear and practical method for developing a process architecture from a set of business entities.
- As the processes in the architecture are identified, the internal structure of each can be modelled using the established method of *role-activity diagramming* (RAD). (Ould's version of RAD is incorporated within *Riva*.)
- The process architecture and associated set of RADs can serve as a blueprint for the implementation of processes, which might be partly or fully automated.
- It includes a bold hypothesis of architectural invariance among businesses of the same type, which it ought to be possible to test.
- If the hypothesis is found to be true, a process architecture developed for one business could be transferred to, or reused in, another business of the same type.

The key starting concepts in *Riva* are *essential business entity* (EBE) and *unit of work* (UOW). EBE is somewhat vaguely defined by Ould (2005, Ch. 6), partly to ensure flexibility in the method. An EBE is something that characterises the business that an organisation is in, something that an organisation has to handle and 'cannot get away from', by virtue of being in that business. It can be concrete, like a car on a production line, or more abstract, like a production run. Ould contrasts essential with *designed* business entities (DBEs); a DBE is an entity that is not an essential part of the business, but rather a way of getting something done that could be done another way. So an invoice is a way of requesting payment, but not the only way. Ould's advice when modelling is to replace DBEs – where possible – by the EBEs they stand for or implement (Ould, 2005, p. 173). Ould defines *units of work* as 'those entities that have lifetimes during which we must look after them' (p. 176). These are the ones that are to be included in the architecture. Entities – even EBEs – that do not have to be looked after by the organisation, even if they are essential to it, are not units of work. A quality standard or our Chief Executive might be EBEs for us (we cannot get away from them), but they are not units or work for us (it is not our responsibility to look after them). UOWs can be essential or designed.

Different units of work may be dynamically related to one another. This happens when one type of UOW arises because another one calls for it or needs it. We may have ‘customer’ as a UOW, and then, because customers produce orders – which we also need to look after – we find we have ‘order’ as another UOW. Ould uses ‘generate’ as a catch-all term to cover all dynamic relationships between UOWs. In a UOW diagram, the fact that UOW A generates UOW B is shown by drawing an arrow from A to B and labelling it with an appropriate verb. The cardinality of the relationship (eg 1:1 or 1:m) can also be shown. The UOWs themselves are shown in hexagonal boxes (there will be examples later).

The dynamic (‘generate’) relationship between two UOWs can be realised in two different ways, described by Ould (2005, pp. 154-159) as *service* and *task force*. The former occurs where the generated UOW is associated with a “permanent service offered by someone else”. When on the other hand the part of the organisation handling the generating UOW sets up the means to handle the generated UOW itself, this is a task force arrangement. For example, customer orders might be handled by a separate order processing service (in another part of, or outside, the organisation), or might be handled as they arise as an integrated part of dealing with the customer.

The UOW diagram is further elaborated to become a *process architecture diagram* (PAD). The PAD is also the point of connection to a set of role activity diagrams (RADs) that show the internal structure of the processes and associations between them. (We do not go to the RAD level in this paper.) A process architecture diagram in *Riva* is made up from two main kinds of process, the *case process* (CP), and the *case management process* (CMP). (A third kind, the *case strategy process* or CSP is not yet fully defined in the method.)

A case process models the standard way that each instance of a particular UOW is handled within the organisation. There will be a distinct CP associated with each distinct UOW. A case management process models the flow of instances of the associated case process. A CMP schedules, activates, monitors and resources a succession of CPs. For instance, in an insurance company’s handling of claims, a CP (‘handle a claim’) might be instantiated each time a claim is processed, while the

associated CMP ('manage the flow of claims') organises and oversees the successive instantiations of the CP.

Relationships between processes in a process architecture diagram fall into three main types – *activate*, *compose(or interact)*, and *encapsulate* - abbreviated on the diagrams to A, C (or I) and E. These relationships are more precise than 'generates' at the UOW level, and reflect the particular ways in which processes can be interrelated. If one process at some point starts another one off, it activates it; if one process interacts at one or more points with another, it composes with it; and if one process includes another inside it, it encapsulates it.

Figures 1 and 2 show, in a simplified form, how service and task force arrangements between CPs and CMPs involve activate, compose and encapsulate relationships, in a process architecture diagram for the customer/order example. When the customer orders are serviced out, a negotiation takes place between the customer and customer order CMPs. When the customer orders are dealt with as part of customer handling, the customer order CMP is encapsulated in the customer CP.

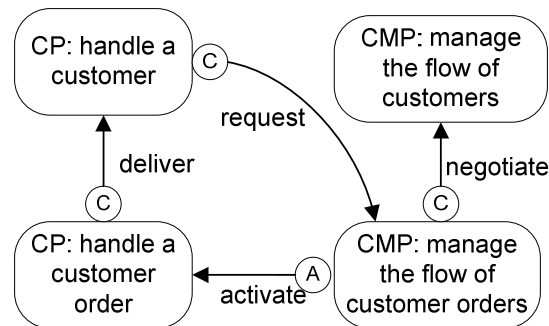


Figure 1: Process architecture showing a service arrangement

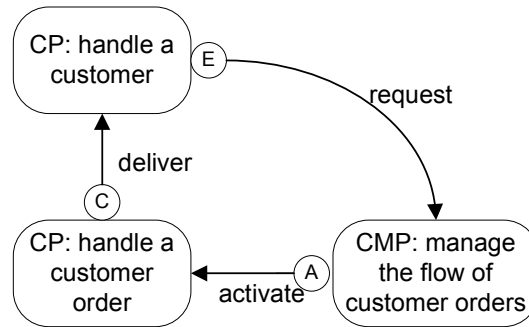


Figure 2: Process architecture showing a task force arrangement

Broadly, the *Riva* method comprises the following sequence of activities:

1. Determine the boundary of the organisation under consideration. (This is a crucial step because the placement of the boundary constrains the set of EBEs and thence UOWs for an organisation.)
2. Brainstorm the subject matter of the organisation to identify its essential business entities (EBEs).
3. Identify those EBEs that have a lifetime which the organisation must handle – the units of work UOWs.
4. Create a UOW diagram that shows any dynamic relationships between UOWs that pertain when one UOW generates (or calls for or demands or activates or requires) another.
5. From each UOW, derive a case process to deal with each single instance of the UOW, and a case management process to deal with the flow of cases.
6. Transform the UOW diagram into an initial ('first-cut') process architecture by connecting the case processes and case management processes together according to their relationships.

7. Apply heuristics identified by Ould (2005) and summarised below to the first-cut process architecture in order to produce a reduced ('second-cut') process architecture.

8. At this point the process architecture based upon EBEs has been created. It is the most fundamental architecture within *Riva* because it is based upon an organisation's EBEs. However, steps two to seven may now be repeated, this time considering Designed Business Entities (DBEs) as well as EBEs. The resulting architecture will be less fundamental, but closer to the actual organisation under consideration.

Ould notes that for an existing organisation, the first-cut process architecture often shows more than actually exists. Consequently, he has identified heuristics intended to reduce this process architecture to a more compact second-cut process architecture. He lists five heuristics:

1. For task-force arrangements, fold the case management process into the case process that requests it.
2. When one UOW generates precisely one other UOW, it may be possible to remove the associated case management process.
3. If UOWs are linked in a chain (A generates B, B generates C, etc), it may be possible to short circuit the chain (so that C delivers directly to A, for instance).
4. Where one UOW stands for a collection of UOWs that the first one generates, the CMP for the second UOW may be folded back into the CP for the first one.
5. If a CMP for a particular UOW has nothing corresponding to it in reality, it may be removed from the process architecture.

3. Background to the case study institutions

The two institutions whose process architectures we compare are UK universities in the same region. One (University A) is a post-1992 and the other (B) is a pre-1992 university. Our investigation centres on a School of Technology in University A and a School of Management in University B.

For the area of activity examined here, the management of taught programmes, the two universities are in what Ould would describe as the same business: producing degree-level education and awarding the qualifications based on the achievements of their students. They also operate within a similar regulatory context and on a similar scale. However, as is explained below, the way in which significant operational decisions are made differs and this may be reflected in the relevant process architectures.

The two institutions examined here are active in teaching undergraduate and postgraduate degree programmes and in research. There is a difference in the proportion of each School's income that comes from teaching and research, but not in the nature of the activities. The number of course offerings and the amount of teaching work undertaken are also similar. Both organisations undertake a range of teaching from undergraduate, through professional development to advanced Masters and research degree level. Both teach students registered within the School and also from other departments in their respective institutions. The two Schools under consideration each have about one hundred academic staff.

A significant difference lies in the principles around which degree programmes are designed. While both institutions nominally divide their programmes up into modules, each with a credit value, the important academic decisions, which affect student progress and the awarding of credit, are made differently.

At University B, credit is only awarded at the end of an academic year by the department which hosts the degree programme on which the student is registered: it is impossible to be awarded credit for individual modules under the normal regulations. Any student not meeting the criteria for passing a whole year is required to retake that

year, including those modules which were passed the first time around. At University A, credit is awarded by the department that owns the module, so credit-awarding decisions for the same student can be taken in different departments. Students can, under this system, accumulate credits at a variable pace and, once a module is passed, it will never need to be retaken.

With other decisions affecting student progress there is a similar distinction. At B, decisions about whether or not a student is guilty of plagiarism, has a justified case for mitigating circumstances or for late submission of coursework, etc, are all taken by the department that runs the student's programme, even where these issues concern modules owned by another department. At A, all of these decisions are made by the department that owns the relevant module.

The difference in processes arises from the continued existence, at University B, of the linear degree programme: there is a strong attachment to the idea of a student having to pass all elements of a traditional full-time year of degree study in order to move on to the next. While there are modular features there, notably the sharing of units by several degree programmes, the model of academic progression at B is one which University A has moved away from since 1994. At A, there is no distinction in the regulations between full and part time students, with no concept of them having to pass a whole year at a time. Degree programmes at A are important because they identify students' areas of study, and this is recognised in the management of Faculties. However, the award of credit is on a module-by-module basis: credit can be accumulated by students at different rates.

These differences are reflected in the contents of the respective quality documents. University A's Modular Assessment Regulations include a series of job descriptions to demonstrate the responsibility of particular office-holders in a faculty, Dean, Award Leader, Field Leader, Module Leader etc, for different stages of each procedure. This definition of responsibilities is less evident at University B. Here, a Code of Practice guides academic departments in their organisation of programmes, but it is less prescriptive about the responsibilities of individuals: the only job titles to be mentioned are Director of Studies (the equivalent of Award Leader at A) and Head

of Department. Some areas of the management of programme quality, notably curriculum and module development are not allocated to particular managers.

University A has a strongly modular programme. Each Faculty is required to maintain a curriculum document, to which the fields, in the Faculty i.e. subject groups, contribute modules. The construction of new degree programmes then becomes principally a matter of fitting together curriculum elements, often already in existence, from one or more fields. Like many of the pre-1992 universities, University B's teaching programmes are based largely on distinct, linear degree programmes. Although units are defined as modules with set amounts of credit, degree programmes are designed and implemented as single entities.

At University A, modules are essentially "owned" by the Fields. There is a clear line of responsibility for managing issues that arise in their staffing. The presumption is that, once a module has been offered, then it will run if enough students are taking it: staff illness means that another member of staff has to step in. The removal of a module is a collective decision, the implications of which are discussed by the Field Committee, representing all staff members in the subject area.

At University B, by contrast, most module development is left to the decisions of individual staff members. Where units are optional on the a degree programme they are often developed from the research of individual staff members, and so are not made available if that member of staff is unavailable for any reason, or leaves the institution. There is a strong sense of personal ownership of modules by individual members of academic staff rather than by the department or institution.

The description of the two institutions in terms of their structures and procedures sets the terms of the comparison of their process architectures, when defined more formally through the use of the *Riva* methodology. There are clearly activities in common, but also differences in the way in which important operational decisions are made: on what to teach, on whether to accept appeals or extenuating circumstances claims and on the award to students of credit and degree-level qualifications. The modelling exercise that follows investigates the extent to which these differences are

reflected in process architecture. This, in turn, provides an indication of whether processes might be interchangeable or reusable between institutions.

4. First Attempt at Process Architectures

We proceeded to apply the *Riva* method to produce process architectures for the area of programme management at the two Schools in Universities A and B. The two of us located at University A carried out the modelling for A, while the third author, who had worked previously at A but now works at B, did the modelling for University B. Though we had worked together to frame the problem area, the two modelling exercises were carried out separately. The idea was to develop the two architectures in isolation and then see how similar they were, or whether we could merge them.

4.1 School of Management at University B

The following diagrams illustrate the formal side of programme management in the University B School of Management. In the UOW diagram, note that the cloud symbol indicates that some external activity is generating one of the UOWs of interest to the organisation.

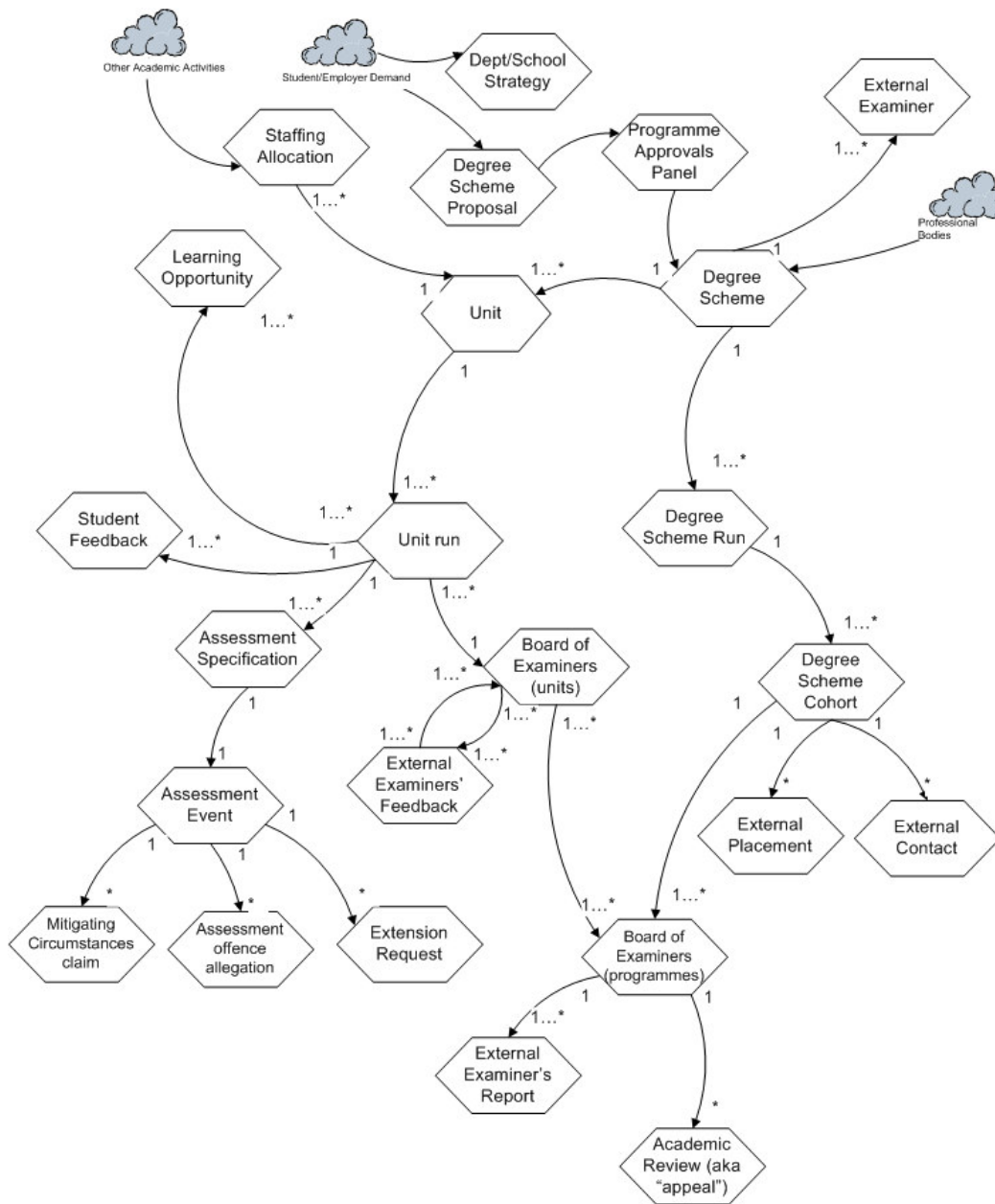


Figure 3: UOW diagram (University B)

The important points to highlight, linked to the previous comparison of the two academic institutions, are:

- Units (or “modules”) generate Unit runs which generate assessment events and Boards of Examiners for Units. The Board of Examiners for Units is there to ratify the marks that students are awarded, but not to award credit.

- Degree Schemes (or “programmes”) generate Degree Scheme Runs which generate Boards of Examiners for Programmes. The Board of Examiners for Programmes cannot change the marks that students have been awarded, but does award credit and handles all of the mitigating (extenuating) circumstances claims. Students may not appeal against their mark, but can appeal against a decision about progression or the award of qualifications.
- External examiners are in an ambiguous position, in that they are asked to provide feedback on the decisions of Boards of Examiners for Units, but are required to attend the Boards of Examiners for Programmes. Their reports are written after attendance at the latter and they therefore have no direct influence on the marks awarded.
- Units themselves can be generated by Degree Schemes but can also be generated by Staff Allocation. A new member of staff arrives, with particular research interests. They must be allocated teaching duties. This allocation will often involve the creation of a new unit around their research interests.

4.2 School of Technology at University A

The corresponding UOW diagram for University A’s School of Technology is shown at Figure 4.

Note here the division of the diagram into two parts, an operational part centred on Module Run and a developmental part centred on Programme Specification. Programme Specifications are seen as generated from work in curriculum design and faculty planning, and themselves generate more detailed specification work as well as two kinds of approval event. Module Run, itself generated by Programme Run, generates teaching and learning activity, assessment specifications and events, and the end of session examination board (for a disciplinary Field), as well as two kinds of feedback reporting. Programmes are seen as part of Schemes, which operate according to a schedule. The Scheme here is a whole set of programmes in a School or Faculty running under the modular assessment regulations. Programmes are reported on in an analogous way to Modules, and the culmination end of session examination board (Award Board), which makes progression decisions for students, is conducted at Scheme level.



Figure 4: UOW diagram (University A)

4.3 Comparing the UOW diagrams

We carried on from the UOW diagrams to develop first- and second-cut process architecture diagrams, but will not show those here; the PADs are essentially elaborations of the UOW diagrams, so similarities or differences can be adequately gauged at the UOW level

Comparing the two UOW diagrams, we had to conclude that, even though we felt that there was a broad similarity in the processes at the two sites, and even though we had shared and overlapping experience of the management and development of taught programmes, which might have been expected to produce similar analyses, the diagrams that we produced were quite dissimilar – in the names and numbers of UOWs, and in the relationships between them.

Some of the differences are terminological, so could be fixed by agreeing a shared vocabulary. Sometimes there were different decisions about whether something was important enough to include (eg, ‘mitigating circumstances’). More generally though, the diagram for A is a little more complex than that for B and shows a clearer separation between operational and development activity, as well as a little more emphasis on planning and reporting activity. The B diagram is more integrated round central concepts of degree scheme and cohort while the A diagram represents a more elaborated modular scheme. The situation at A looks more bureaucratic than at B, where a greater level of autonomy appears to be enjoyed at departmental level. Development processes at A are more formalised than those at B.

Did this mean that higher education, at least as a set of processes, was not the same in the A and B Schools? Does this in turn imply that organisations can be in the same business but still have different process architectures?

Because we felt that the situations at our two institutions were not that different, and we could imagine process evolution at either place that would take it in the direction of the other, our first thought was that we had mistakes in the modelling process.

We might have stayed too close to operational reality, partly because of our familiarity with and involvement in it, and that has led us to produce descriptive, low-

level architectures, so obscuring the truly essential business of higher education by focusing on contingent features. Perhaps we should have spent more time deciding what our EBEs were, and in particular what was an EBE and what a DBE. It could be that two people working on a model will tend to produce a more complex result than one person. Noting Ould's warning (2005, p. 181) not to be too liberal in adding 'generates' relationships to a UOW diagram, the thought also occurs that perhaps we had over-connected our diagrams, including too many relationships that were only 'interesting' and not really dynamic, possibly under the influence of using other modelling methods.

So we decided to go back to basics and redevelop the process architectures by isolating a small set of what we took to be the really fundamental elements of programme development and delivery in higher education (the core EBEs, as it were). To do this necessitated our taking a step back from operational realities, to consider what, so far as we knew from first principles, the inescapable components had to be.

5. Second Attempt at Process Architectures

5.1 Establishing a core architecture

For our second attempt at developing a process architecture for programme management in UK Higher Education, based on the nature of the business at our two institutions, we carried out a joint brainstorming exercise to find a set of core EBEs. The intention here was to establish a solid foundation from which to go on and develop our process architectures. We thought that if we had a common foundation, the two eventual architectures would be easier to compare. Differences between them then might then be understood in terms of DBEs: as we elaborated the common architecture to accommodate business processes at institutions A and B, we expected that they would diverge, but that the additions to the core diagram would be DBEs, reflecting different ways of doing things, but not a difference in the essential business, at the two places. We were effectively redrawing the boundary between EBEs and DBEs, and expecting organisational reality to be shaped more by the latter than the former. Still, if this worked, we would have established a core architecture, which might then be transferrable to other institutions.

We came up with the following five core EBEs in our brainstorming exercise:

- Student
- Staff member
- Curriculum element
- Teaching and learning activity
- Assessment event

These were the EBEs that we felt a University in the UK *really* could not get away from. All university departments will have a curriculum (otherwise there would be no way of giving it any kind of disciplinary identity), teaching and learning activities (whether formal or informal, real or virtual), assessment events (because without this no accreditation of a student's achievement is possible), staff, and students.

Our list is open to the objection that it is quite obvious, and could be worked out by anyone with a passing knowledge of higher education. This core would also work quite well for other levels of education, in the UK and elsewhere, so hardly captures the essential or distinctive quality of the area of activity in which we are interested. Still, what we were trying to do was not find a set of EBEs that distinguished higher education *from* other areas of activity, but find a set that provided a common core *within* HE institutions, as a starting point for development of full process architectures for different institutions in the sector.

Since we are interested in the lifetime of each of these five entities, each one of them becomes a UOW. The UOW diagram, with appropriate clouds and relationships added, is presented as Figure 5.

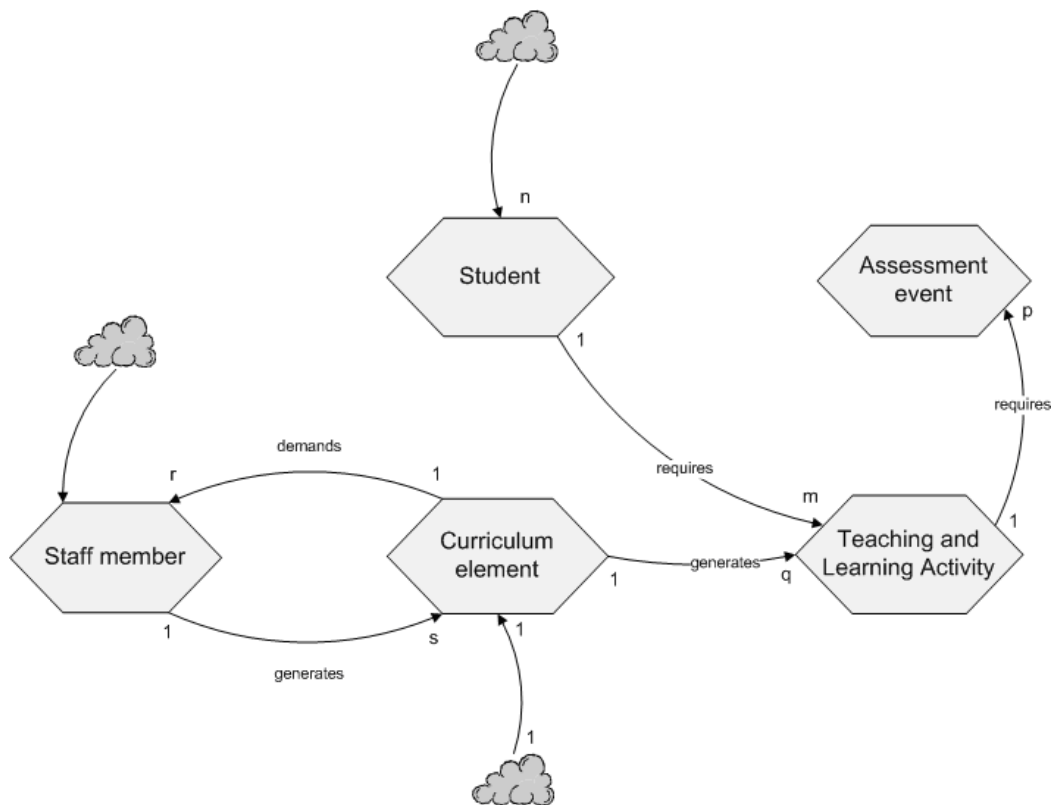


Figure 5: Core UOW diagram

Staff members generate curriculum elements in the sense of proposing ideas for what should be taught. Conversely, the content of teaching (derived from the curriculum elements) demands the existence of staff to cover them. These curriculum elements also generate teaching and learning activities: teaching activities for the staff and learning activities for the students. To know how effective learning activities have been requires assessment events, which are shown in the figure as being required by teaching and learning activity.

We went on to derive first- and second-cut process architecture diagrams from this core UOW diagram, and these are shown as Figures 6 and 7. For the first-cut architecture, the first step is to produce a CP and a CMP for each of the UOWs – except for Student, which does not get a CMP because Student is not generated by any other unit of work (only by a cloud). Then we consider whether the relationships between CPs (via their respective CMPs) are organised through services (with an independent CMP) or through task forces (with CMP encapsulated). In this case, all

the relationships seemed best understood as task forces, and so were modelled accordingly.

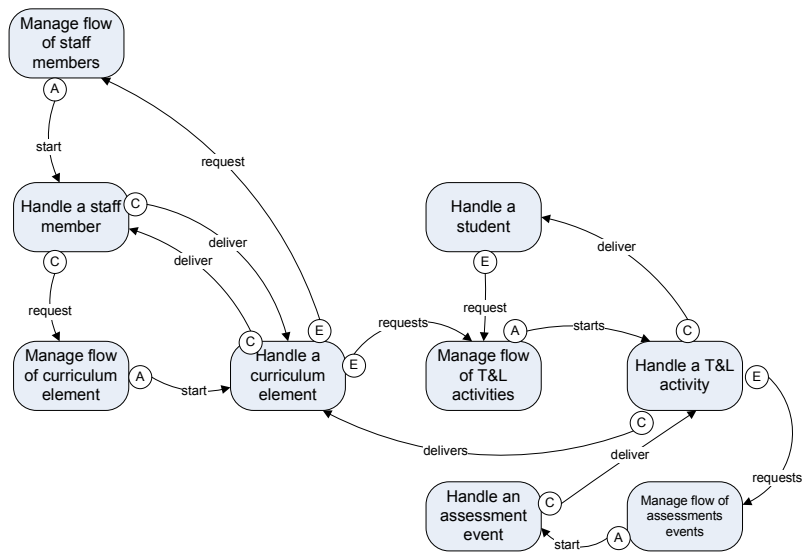


Figure 6: First-cut core process architecture

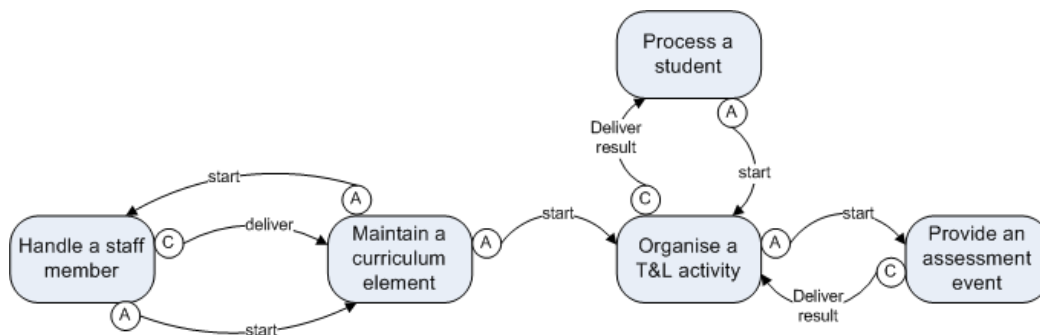


Figure 7: Second-cut core process architecture

Some of Ould’s heuristics were used to reduce the first-cut architecture to a second-cut equivalent. First, all the encapsulated CMPs were folded into the requesting CPs. Second, some of the delivery chains were rationalised. The chain of results delivery from “Handle an assessment” to “Handle a student” was retained because the

intermediary CP “Handle a Teaching and Learning activity” does actually make use of the result. But the deliver relationship between “Handle a teaching and learning activity” and “Handle a curriculum” was deleted because in practice nothing is actually delivered. Similarly, the deliver relationship between “Handle a curriculum” and “Handle a staff member” was deleted. As a final step, the names of some processes were changed into to more appropriate ones.

5.2 Adding Designed UOWs

To approach more closely the business processes at our two institutions (ie, the programme management activity in the two Schools), it is necessary to move beyond the core architecture and include additional business entities and units of work. Since we are now investigating the two sites separately, we are allowing the possibility that the additional entities/units of work will be different in the two places, and therefore not ‘essential’ to the business area in general. We will therefore treat any additional entities or units or work, at least provisionally, as *designed*.

In the particular context of UK Higher Education, because universities operate within a nationwide set of arrangements for the maintenance of quality and standards, these additional entities and units of work, even if they are not identical between institutions, are likely to reflect a fairly constrained area of choice between different ways of operating. This opens the possibility that between essential business entities common to all organisations in the same business and designed business entities specific to individual organisations there may lie an intermediate territory occupied by business entities selected from a restricted range.

As with the original modelling exercise, for this expansion from the core architecture we worked on the two cases separately, and compared outcomes subsequently.

5.2.1 Re-work for the School of Management at University B

The re-worked UOW diagram for University B is shown as Figure 8. The key relationships which needed to be illustrated when adding University B's DBEs to the UOW Diagram were those between units, assessments and marks on the one hand, and between Degree Programmes, cohorts and progression decisions on the other.

“Mark approval” is a unit of work that covers the receipt of a piece of assessed work, through marking, moderation and the formal acceptance of the mark. Assessment offence allegations (e.g. suspicions of plagiarism) can be generated within marking, hence by the “Mark approval” unit of work.

“Student progression decision” covers all the exam board activity concerning discussion of whether or not students have met the requirements of the particular stage of their programmes. It includes, for undergraduate programmes, degree classification decisions. Marks are not changed in any way within this process. But appeals can be generated. Students are allowed to appeal against decisions on progression to the next year and on the award of qualifications, but not against individual marks.

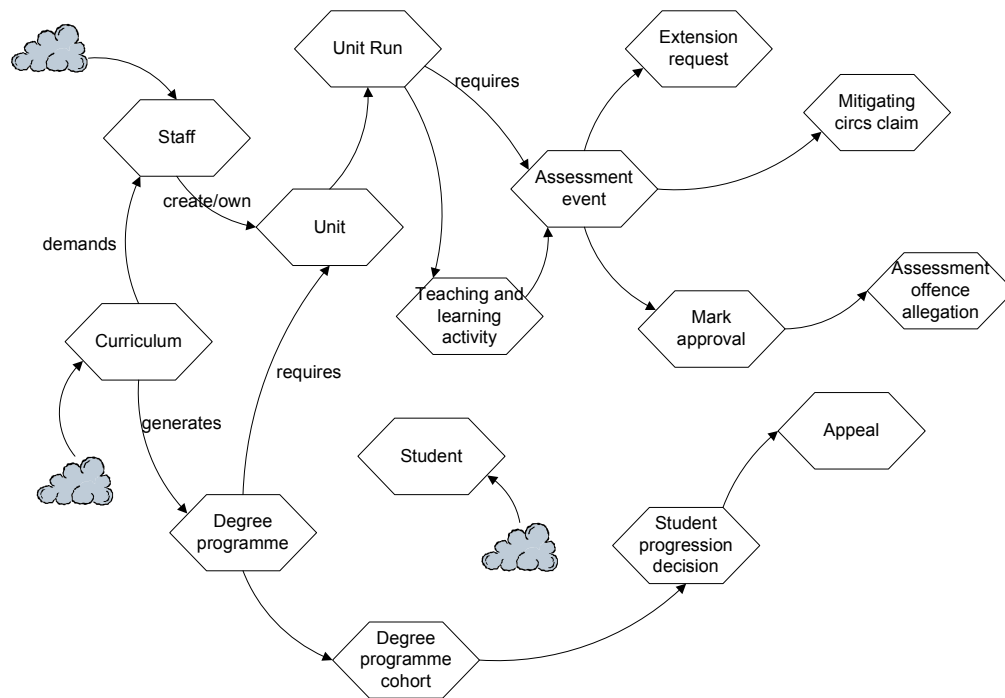


Figure 8: Redrawn UOW diagram (University B)

5.2.2 Re-work for the School of Technology at University A

The re-worked UOW diagram for University A is shown as Figure 9. Redeveloping from the new core has made a noticeable difference, as can be seen by comparing Figures 4 and 9. ‘Scheme’ is no longer visible, and ‘Programme Run’/ ‘Module Run’ have transmuted into ‘Programme Delivery’/ ‘Module Delivery’. The split between programme operation and programme development is less marked than previously, and the specification UOWs occupy the central position formerly held by the run UOWs. Starting from the core entities seems to have produced a reorientation away from an organisational view of programme management towards a more pedagogic perspective, though the view is still somewhat formal, with procedures, documents and delivery and approval considerations very much in the foreground.

The redevelopment has altered the UOW diagram for university A, but not radically enough to put the core entities at the centre. ‘Student’ looks marginal and ‘Staff Member’ appears subordinated to the specification UOWs. Although ‘Assessment Event’ has grown in prominence in the re-work, ‘Curriculum Element’ is squeezed between an idea and a proposal, and ‘Teaching & Learning Activity’ – the central element when you consider the business *educationally* – has ended up hidden behind ‘Class’.

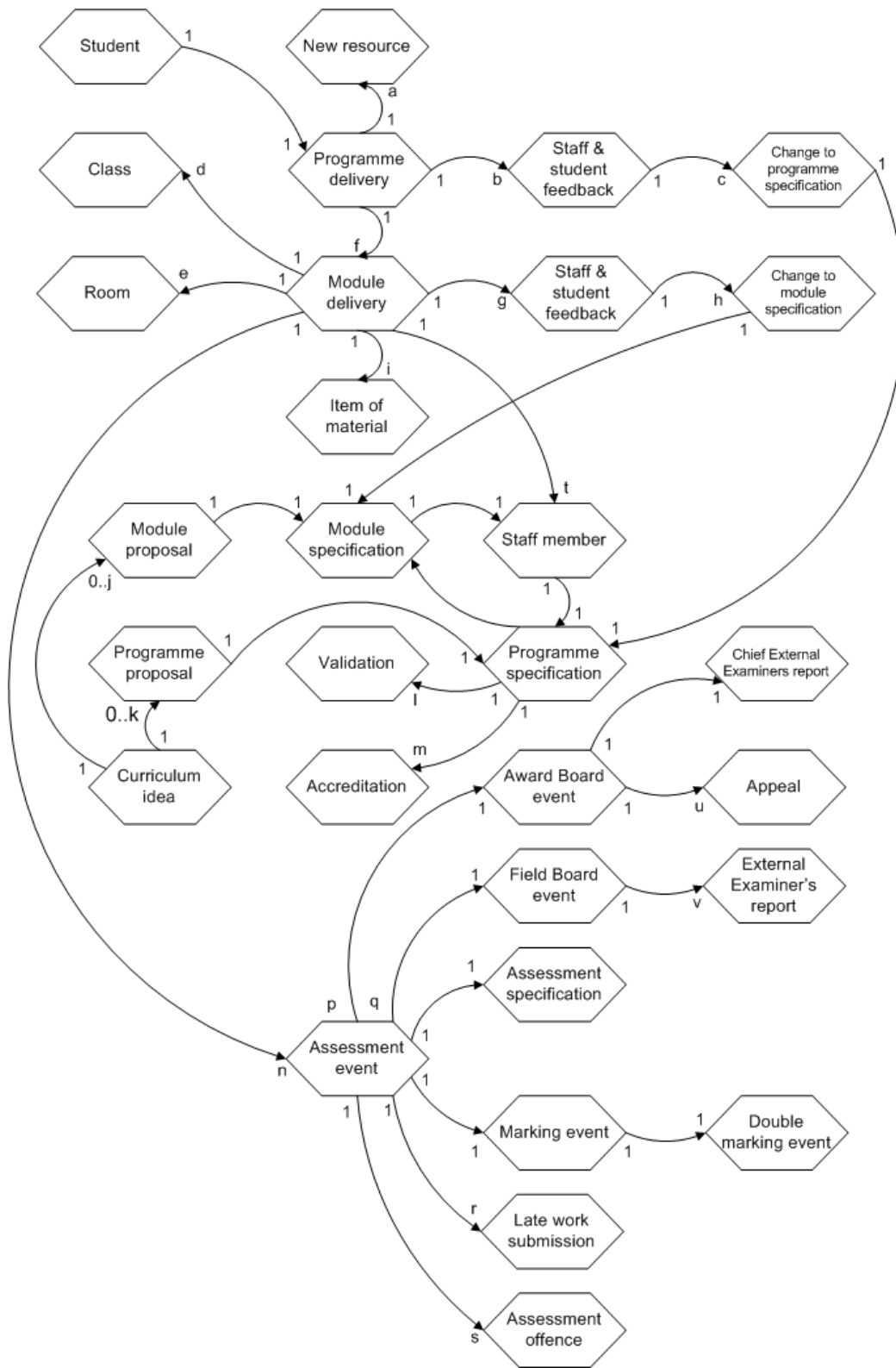


Figure 9: Redrawn UOW diagram (University A)

5.2.3 Comparing the A and B re-works

We have gone on the develop process architecture diagrams from these two new UOW diagrams, but there is not space to present them here. In any case, comparing the new UOW diagrams for A and B tells us enough about whether the developments have been convergent or not.

Clearly they have not. Figures 8 and 9 are if anything more dissimilar from one another than Figures 3 and 4, not least because the discrepancy in size between the A and B models has become more marked.

It was not the primary purpose of re-starting the architecture development from a set of core entities to produce convergent models, but rather to make a clearer separation between essential and designed business entities and units of work. However, it was at least an imaginable outcome that establishing a common core for the models would encourage a disciplined approach to modelling that involved working away from a central core and adding designed UOWs at the edges of the model. Agreeing a core evidently did not produce this effect. Something stronger must be at work which is pushing the modellers in different directions. In the B model, the degree cohort remains a strong presence, while in the A model all the core elements except for assessment event have been marginalised to a greater or lesser extent.

The likely explanation for these effects is the difference in organisational context remarked on earlier, namely that departmental autonomy is relatively well preserved in University B, while at A programme management is conducted within a university wide modular framework. The model for B can be smaller, because the key activities are under local control; while the model for A must pay attention to a more distributed control framework and a more formally articulated set of procedures.

6. Discussion

We set out in this paper to apply the *Riva* method in an exercise in modelling the process of programme management in two comparable UK universities. We were also interested in testing of Ould's proposition that, for any area of business, a fundamental process architecture can be identified which will not vary between organisations in that business area.

Our first attempts to model the processes of the two Schools that were our case studies produced such disparate results that it led us to doubt, not so much Ould's proposition, but the correctness of our own modelling. We therefore made another attempt, this time proceeding in two steps: first, we identified a small set of core EBEs that we thought would be common to all UK HE institutions; and then we tried to add the DBEs that we thought would capture local practical, structural and cultural differences between Universities A and B.

The core model, consisting solely of EBEs, we think could be applied not only to the institutions examined here, but also to other types of taught course in higher education. For example, the EBEs have been created at a sufficient level of abstraction to cover work-based or distance learning, since there is no mention of classroom activity. They could cover programmes where the assessment is based entirely on projects or on examinations, because there is no specification of the form of assessment which is to take place. And the EBEs will cover programmes where certification or the award of credit takes place at different points in the academic cycle, whether after each module is taken or only when a student has completed the entire programme.

The EBEs in the core model are however quite abstract. *Curriculum* in particular is sometimes a very formal entity, but not in every institution. In our Universities A and B, while A has a set process for handling new modules and programmes, as well as changes to these, "Curriculum" has no formal life history of its own at University B, but simply emerges from the processes of developing and managing the degree programmes.

When we compared the extended UOW diagrams for A and B in the second development attempt, having added different DBEs to the common core, we found no less divergence between the outcomes than in the first attempt. Our conclusion from this is that we not yet have the right distinction between essential and designed entities or units of work. It appears that there are real differences between the organisations which cannot simply be assigned to ‘designed’ elements. These real differences (the strength of ‘degree cohort’ at B, and the prominence of specifications and delivery at A) seem at least as important as the core ‘essential’ entities.

We do on the other hand see some similarities in the designed business entities emerging in the extended modelling, particular in relation to detailed procedural work such as assessment offences, appeals, and mitigating circumstances.

The position we have arrived at can be summarised as follows. Nothing that we have done lends support to Ould’s contention that organisations in the same business will have the same process architecture. We are reasonably sure that our A and B are in the same business, but we are unable to produce similar process architectures for them. This may of course be due to poor modelling on our part. In our second modelling attempt, we produced a core set of entities as a first step, and believe this approach may have some merit (but need to do further work on it). On the other hand, we have found that some entities we thought of as designed seem to be at least as important (in a particular organisation) as others we had deemed essential. We also found, that in some of the designed areas, there was some convergence between the two sites. Our analysis, in particular in relation to the essential/designed distinction, needs to be re-examined and deepened.

With regard to the use of *Riva* as a modelling method, we have found it very useful, for several reasons:-

- It puts a primary emphasis on an organisation’s “subject matter” – the key entities with which it has to deal – rather than on existing patterns of activity.
- It provides a systematic way of proceeding from the identification of business entities to the development of a process architecture and then to the detailed design of roles, activities and interactions within each of the processes in the architecture.
- It is sufficiently flexible to be used informally as well as formally – the diagrams do not have to be complete or validated before they are useful.

However, we also find some shortcomings or areas that might be improved in the method:-

- The claim that organisations in the same business will have the same process architecture is too strong and cannot be proved or disproved until it is made more precise what it means to be in the same business or have the same process architecture.
- The method does not define clearly enough any of the terms ‘essential business entity’, ‘designed business entity’, ‘unit of work’, or ‘generates’. As a consequence of this, it is difficult to determine whether a unit of work diagram (and hence a process architecture diagram) is accurately drawn, and the importance of designed business entities might be underestimated.
- The derivation of the first-cut architecture from the UOW diagram is claimed by Ould to be ‘mechanical’ (2005, p. 83), but in fact requires consideration of service and task force arrangements and the difficult transition from ‘generates’ in the UOW diagram to ‘activates/composes/encapsulates’ in the process architecture diagrams.

Some progress needs to be made in these areas if *Riva* is to become a general purpose method for establishing definitive process architectures for different business areas. It may be too ambitious to aim for complete architectures, but transferrable partial architectures would also be useful. From our experience, the greatest need is to achieve greater clarity in the identification of essential and designed business entities. As suggested above, it may be that the distinction between these two is not absolute, and that intermediate entities that are neither universal for the business area nor specific to the organisation need to be recognised. If the production of the UOW diagram can be made a surer exercise, the transition to the process architecture should become simpler, and might be achieved through the application of a single set of heuristics instead of the present two ‘cuts’.

References

- Harmon, P., 2003. *Business Process Change*. San Francisco: Morgan Kaufmann.
- Kavakli, V. and Loucopoulos, P., 1997. 'Goal-driven Business Process Analysis Application in Electricity Deregulation', *Information Systems* 4(3), pp. 187–207.
- Lunn, K., Sixsmith, A., Lindsay, A., and Vaarama, M., 2003. 'Traceability in requirements through process modelling, applied to social care applications', *Information & Software Technology*, 45(15) pp. 1045-1052.
- Ould, M.A., 2005. *Business process management: a rigorous approach*. Swindon: BCS.
- Snowdon R.A. and Kawalek, P., 2003, 'Active Meta-Process Models: a Conceptual Exposition', *Information and Software Technology*, 45 (15), pp. 1021-1029.