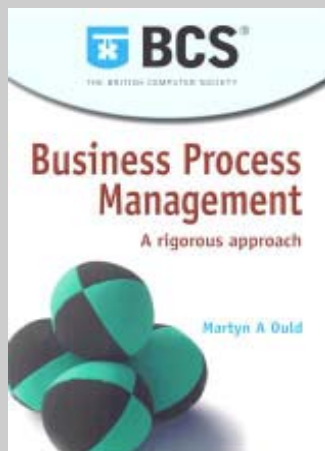


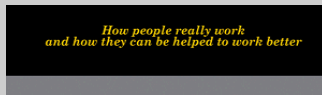
October 2005



Business Process Management: A Rigorous Approach

Martyn Ould

Meghan-Kiffer Press 2005
\$29.67, 340 pages



Keith Harrison-Broninski

Human Interactions: The Heart and Soul of Business Process Management

Keith Harrison-Broninski

Meghan-Kiffer Press 2005
\$26.37, 304 pages

Is There a Method to the BPM Madness? A Review of Two Books Describing a New Foundation for Business Process Management

Sue Bushell

Although the past five years have witnessed great progress in the adoption of business process management, deployments have been mostly tactical and limited in scope, applying to improvements in specific business functions or departments. In this context, not much attention has been paid to process-oriented modeling techniques and methods. But that is about to change, and for good reasons.

Most conceptualization and development methods are ad hoc, proprietary, or based on information systems (IT) methods – It's been a kind of BPM methods madness. But if we look out on the horizon of strategic and complex, "enterprise business process management," the need for industrial-strength process-oriented methods can no longer be swept under the rug.

Clearly, process conception, analysis, and development represent a different set of artifacts and constructs than those found in traditional software development, which is aimed at *information management*, not *process management*. These revelations, and prescriptions for appropriate methods that can be deployed by reality-based CIOs with a job to do, can be found in two recently released books on process-oriented methods:

- *Business Process Management: A Rigorous Approach*, Martyn Ould, 2005, Meghan-Kiffer Press. (www.mkpress.com)
- *Human Interactions: The Heart and Soul of Business Process Management*, Keith Harrison-Broninski, 2005, Meghan-Kiffer Press. (www.mkpress.com)

Sure, as a CIO with real-time pressures from the CEO and COO, you see that the real need is to create a simple framework of techniques that regular working folks can use to design and implement a new process quickly. Learning will occur more promptly for the employees doing the new process, and they can continue to evolve the process as necessary. You may also be thinking that this process stuff is just the same as building information systems from "objects." It's not. Object orientation is about building and maintaining IT systems; BPM is about building and maintaining business processes, and putting those capabilities in the hands of ordinary business people who get work done through their work processes – not their "objects." The typical IT practitioner may be thinking, "Who cares about 'processes versus objects' if I have a deadline I need to meet by Monday morning!" Fair enough! But if we kept our heads down to Monday-morning deadlines – without lifting our eyes to the horizon now and then – we would still be programming in machine language. It's now time to lift our eyes and gaze on the possibilities of actually meeting those Monday-morning deadlines with time to spare by moving on from IT development methods to process-oriented development methods. It's business wise to shift the way of



thinking, the mental models – and the way of doing – from the IT to the business-process level of abstraction. And it's not just because we can; it's because the business demands it.

In his eye-opening book, *Business Process Management: A Rigorous Approach*, Martyn Ould explains, "The key thing about a paradigm shift is that old ways of thinking just won't work in the new world. If your structural engineers have only had mud to work with in the past, their ways of specifying and designing buildings will be fine for mud buildings, but they won't make a lot of sense when the steel girder appear on the scene. Today's dedicated follower of fashion in information system development is likely to be speaking UML and using one of the various development approaches based on the UML. [But] we cannot view business process management systems as just another sort of information system." Information-oriented languages and information-oriented methods have evolved around a world of storing, retrieving, and updating information, not the dynamic world of process.

Ould elaborates, "Simply extending our information-based methods simply won't work – they don't have the necessary concepts at their heart. We're moving from the Information Age to the Process Age. We need purpose-built methods for working with processes to replace our methods for working with information." The process-managed enterprise demands that the center of automation be shifted from *information processing* to *process processing*. Ould continues, "The traditional paradigm views organizations as things that work with information, so our systems have been about looking after information. In the past we have specified our information systems in terms of what data they will store and how we can access it and change it and move it around. And we have designed our information systems in terms of data representations and operations on data." Processes need an architecture and conceptual framework of their own, not a data-oriented information systems paradigm. While new "service-oriented" architectures and techniques for software development offer flexible, loosely-coupled approaches to computing, these advances, though good, are aimed at technical people who build and manage information systems, e.g., programmers, not business analysts who want to build and manage business processes. To this end, a modeling framework based on techniques compatible with process-oriented architecture is needed to supersede information-oriented approaches to process modeling and analysis. This shift from the Information Age to the Process Age in terms of architecture and methods will serve as the cornerstone of BPM, now that the tinkering, tactical deployment phase is moving on to strategic enterprise BPM.

In effect, BPM assimilation in the enterprise has reached a watershed:

It's almost universally agreed that the BPM market opportunity is very high, that its compound annual growth rate (unrestrained by a bad economy) is among the highest for any software category, and that the potential ROI and rate of return has few peers. In 2000, most estimates of the BPM market were in the tens or perhaps low hundreds of millions in U.S. dollars. By some analyst forecasts, the BPM market will be between \$4 and \$6 billion U.S. dollars in 2005. ¹



Yet despite the best efforts of standards bodies, IT vendors are responding with a myriad of diverse offerings – typically either enhancements of legacy workflow/EAI software or concoctions based on an assortment of acquisitions. And these products, with their associated proprietary development methods, do not conform to any generally accepted set of principles for the construction of process-based systems:

The realization of those sketchy flowcharts drawn by business analysts on whiteboards requires an architecture built on the best of BPM's many standards: BPEL, BPMN, and WS-CDL. Alas, no actual vendor implementation of this architecture exists today.²

The market is awash with complexity, and may well be heading for meltdown, as the maintenance costs of unstructured, proprietary process systems start to outweigh their benefits. And while vendors and consultants are quite as aware of this state of affairs as analysts and users, so far they have been helpless to improve the situation. We have any number of process languages and standards bodies. But we do not have a universal set of *principles* on which all process systems can be based – a *complete, consistent, and fully business-oriented methodology for the conceptualization and construction of process-based systems*, fulfilling the vision of BPM's pioneers that a business process should directly implement

“the complete and dynamically coordinated set of collaborative and transactional activities that deliver value to customers.”³

Neither Ould nor Harrison-Broninski prescribes any specific process language. Neither of their works is the product of a committee, and neither mandates the use of any specific software. Each man is an alumnus of the legendary IPSE2.5 (Integrated Project Support Environment) project in the U.K.,⁴ an early attempt to build a process platform that is now being repurposed for the business world. Their works embody the principles of *Role Activity Theory* – an approach to process description based on Roles and their Interactions. Role Activity Theory has acquired a core of adherents over the 20 years since its invention, but never quite made it into the IT mainstream, which over the years has concentrated on building information-based, not process-based, systems.

Now, just when the business world needs a process-oriented method to solve real business problems, Ould and Harrison-Broninski have each taken the principles of Role Activity Theory and used them to show a way forward.

Ould's book describes a method, Riva, that can be used both to determine the “process architecture” of an organization – the “chunking” of its activities into separate processes – and to model the processes themselves. The Riva approach is to isolate *Essential Business Entities* (EBEs) – the high-level information objects with which the organization is concerned – and to identify certain of these with high-level *Units Of Work* (UOWs), termed *essential UOWs*. The EBEs and UOWs are the starting point for a more detailed analysis of constituent processes.



For instance, in a pharmaceutical organization, a Clinical Trial would be an EBE, with a corresponding essential UOW to run the trial. From processes that are created to carry out the essential UOWs, the organization then invents *designed* entities (for example, an Invoice) that arise from decisions about how to do business around the EBEs. For purposes of process design and implementation of computer support, both types of UOW (essential and designed) are treated as types of data object, and a generic *process type* is constructed around each one. When it comes to process enactment, each object of a particular type is identified with an actual process of the corresponding type. More precisely, Riva identifies an enacted process with the “lead Role” of the process. It is not so much processes that are created, but lead Roles that then carry the responsibility for a particular stream of activity associated with the corresponding data objects. Riva also conceptualizes three levels of process associated with each essential and designed business entity:

- A *case process*, which handles the work associated with a specific entity – the operation of a particular Clinical Trial, or settlement of a specific Invoice within such a Clinical Trial.
- A *case management process*, which deals with the concurrent operation of case processes – in particular, allocating access to shared resources. Riva associates with case management process actions to do with planning, reporting, monitoring, scheduling, resourcing, prioritizing, negotiating, reconciling, and so on. There is assumed to be a case management process for all Clinical Trials, another within a particular Clinical Trial for all Invoices, and so on.
- A *case strategy process*, which takes a longer-term view of management and deals with overall improvement to both forms of process, taking into account business and industry trends as well as operational issues. The case strategy process for Clinical Trials might look at how they can be re-engineered to cut cycle time dramatically, or how quality measures can be applied to improve efficiency.

Riva is attractive in its formality and simplicity: It sets out specifically to give process modeling both a rigorous underpinning and a methodology that can be applied to any form of organization. The modeling techniques of Riva are attuned well to routine, semi- or fully-automated activities (system-to-system, or human-to-system interactions), but human involvement is limited to key points. In other words, Riva brings S2S and H2S automation and business activities to the process table, where they can be managed as business processes, not IT systems; but still more is needed to take on the human-to-human interactions.

Taking the next step, Keith Harrison-Broninski introduces *Human Interaction Management (HIM)* and its supporting software, a *Human Interaction Management System (HIMS)*. The more flexible *human-driven* processes based on innovation and interaction – in which the process itself is typically shaped by participants as they go along – are the chief concerns of HIM. Common business practices, such as provisional decision-making, or negotiations of indefinite length and nature, are very hard to capture using Riva alone. And the clean separation of



case work and case management work is not sufficient for human-driven processes in which management must be integrally tied to the unfolding of the work itself.

Important to note, of both works, software support is not necessary to derive great benefit from the core principles. HIM, like Riva, is a *method* rather than a notational technique, programming language, or software protocol. HIM unifies various existing disciplines (Role Activity Theory, Cognitive Theory, Social Systems Theory, Learning Theory, multiple process-based Computer Science disciplines, and more) into a *complete theory of human, collaborative work*, and shows how this theory can be used not only to model, but also to manage, any human-driven business processes. HIM also offers a new approach to *quality*, showing how metrics based on human resource utilization (rather than traditional manufacturing metrics such as cost and cycle time) can be used to achieve new levels of operational efficiency and effectiveness. This approach doesn't negate traditional measures such as ROI, profit growth, profit per employee, and revenue and revenue growth. Rather, smart executives see these measures more as symptoms, not as causes. Just as "time" (cycle time, response time, inventory turn time, and so on) can be managed as a business variable – in addition to traditional systematic measures such as cost – human utilization metrics can be measured, not as symptoms, but as *causes* of productivity and performance. You cannot manage symptoms, but you can manage causes of those symptoms.

The claim of HIM is that human-driven processes are generally the part of the process "iceberg" that is hidden under the water – the almost invisible work that nevertheless underpins any organization:

Current approaches to analysis of business processes deal very well with the part above the water, and hardly at all with the part below. Human activity, where it is covered at all, is treated as if it was mechanistic – which is probably worse than leaving it alone. Hence, businesses are competing on only a small part of the activity they carry out: the routinized, regulated, mechanistic part. How much advantage could be gained by competing on the human activity as well?⁵

Harrison-Broninski identifies five main features of human working activity, and explains not only how they can be *modeled*, but also how they can be *managed*:

1. *Connection visibility*. In automated processes, the ultimate aim is to rationalize the distribution of data, logic, and control, regardless of "who" and "what" these resources are, or what they represent, or with what they naturally interact. In human collaborative situations, by contrast, quite the opposite is true. A human process creates *meaningful* connections between participants whose skills, responsibilities, authorities, and resources are quite distinct, and probably very different. To work with people, you need to know who they are and what they can do. Therefore, collaborative technology must provide a strong representation of *process participants*, the *roles* they play, and the *private information resources* that belong to each of them. HIM does this using Role Activity Diagrams (RADs) – a simple graphical approach to



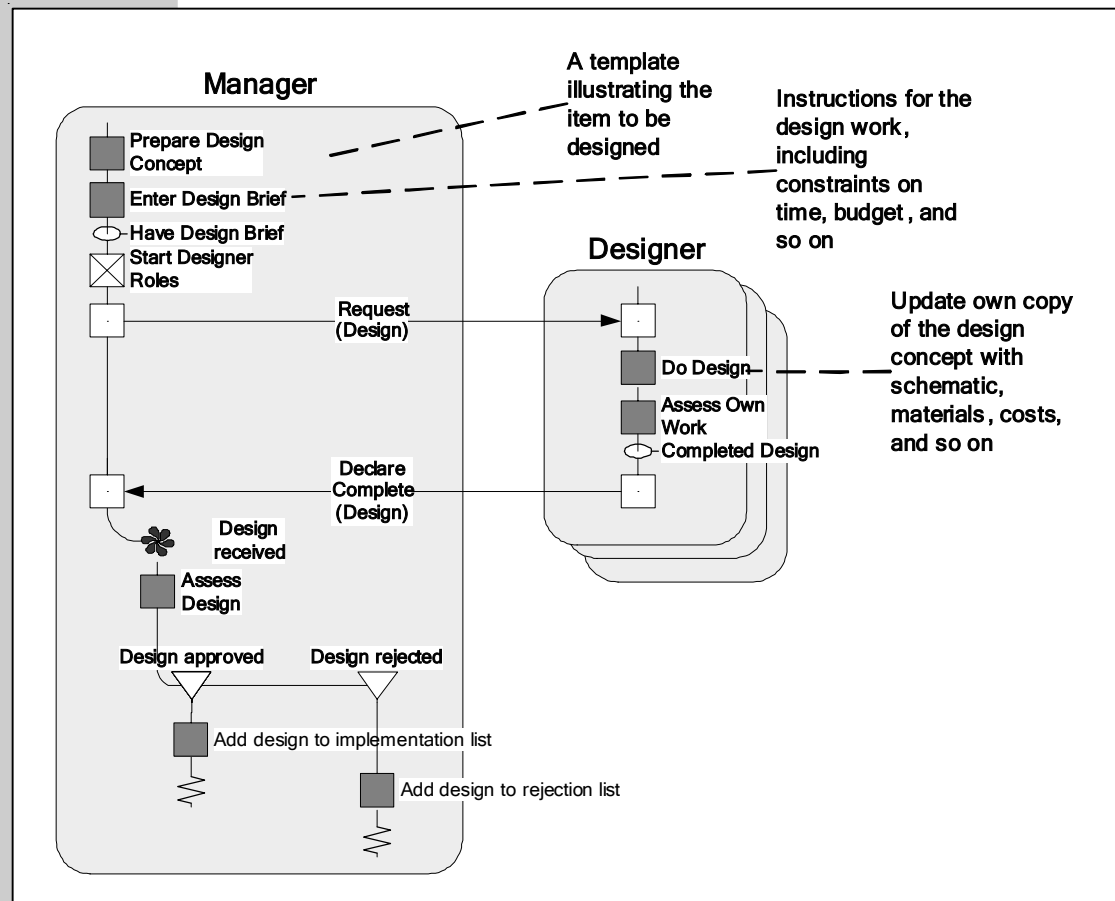


Figure 1. A Role Activity Diagram for an engineering design process.

depicting processes that anyone can understand in moments, with no need for training.

2. *Structured messaging.* Messaging is an enabling force for human-computer interaction, yet typically results in efficiency losses as well as gains. For example, the volume of email received by organizational workers is an increasing problem – sorting it by relevance and priority alone can consume much of a working day. If we are to manage our interactions with others better, they must be structured for us, under process control. For Harrison-Broninski, Role Activity Diagrams are again a way forward, because they allow us to show not only the communications between process participants, but also to impose structure on them – by showing their *intention*, describing the *content* of the message, and showing the interaction *in a process context* so that its dependencies and impacts can be understood.
3. *Support for mental work.* A large part of what humans do has little concrete output – at least not the kind of concrete output that is easily measured by existing management techniques or computer systems. Yet the time and mental effort invested in researching, comparing, considering, deciding, and generally *responding to information* – turning it into knowledge and ideas – is a critical part of the job of an interaction worker. Mental effort along with all



other human activity is described in HIM via a simple and generic pattern, Research-Evaluate-Analyze-Constrain-Task, or REACT (where Research has a sub-pattern, Access-Identify-Memorize, or AIM). REACT and AIM repeated, interleaved, nested, and split across process participants can be used to create process descriptions that reflect the reality of human activity. Harrison-Broninski also identifies a pattern for *collaborative transactions* that allows a complex problem to be managed as a group of simpler ones, with the necessary dependencies automatically catered for. All of these patterns are depicted via Role Activity Diagrams. More importantly, they can be modified “on the fly” in response to changing dynamics of the business.

4. *Supportive rather than prescriptive activity management.* Humans do not sequence their activities in the manner of a procedural computer program – “After doing x, I either do y or z, only depending on the outcome of x.” A person that worked like this would *be* a machine. On the contrary, people take action in different ways on different days, in response to their dealings with others, to changes in the state of resources to which they have access, and – if we’re realistic – to their mood at the time. HIM supports this in a simple way by changing the way we read a classical Role Activity Diagram. A particular activity or interaction has a *precondition* (a state of affairs in which it becomes available) and a *postcondition* (a state of affairs that is guaranteed to be the case on completion). HIM allows people to carry out any activity for which the precondition is true, at any time – no matter what activities they just did previously – and insists that any activity whose results violate the postcondition must be completely undone, in accordance with the business rules set by business managers, so as to prevent it derailing the process.
5. *Processes change processes.* Human activities are concerned often with solving problems, or making something happen. Such activities routinely start in the same fashion – by establishing a way of proceeding. Before you can design your new widget, or develop your marketing plan, you need to work out *how* you are going to do so – which methodology to use, which tools are required, which people should be consulted, and so on. In other words, process definition is an intrinsic part of the process itself. Further, this is not a one-time thing: It happens continually throughout the life of the process. Hence, actions and interactions in human-driven processes cause *continual change to the process itself*. Harrison-Broninski deals with this by developing a new approach to process management. The principle of separation of control allows him to distinguish management control (day-to-day facilitation of human activity, carried out as part of the process itself – ongoing resourcing, monitoring, and process redesign) from executive control (exercise of authority over the process via determination of its primary Roles, interactions, and deliverables). Process evolution can then be implemented under management control via the establishment of consensus among certain participants, based on *how they would proceed from now on* – Harrison-Broninski terms such a consensus an agreement. Each successive agreement can be documented and shared via an updated Role Activity Diagram.

With this last aspect of HIM, the principle of “separation of control,” Harrison-Broninski strikes new ground for modeling the *management* of human-driven processes. In general, though, HIM offers a variety of important new approaches. For example, process diagramming in HIM is very powerful – it uses a much smaller set of symbols than traditional Role Activity Diagram notation, while providing stronger semantics to permit the capture of the flexibility and dynamism inherent in human-driven processes.

Taken together, Ould and Harrison-Broninski provide the frameworks that the BPM world needs to untangle the complex web of processes in the organization, and, generally, put the BPM house in order. Both Ould and Harrison-Broninski are quite amenable to languages such as BPEL being used to implement “programmable” sequences of activity in a process. However, the fundamental achievement of these authors is to recognize that such IT-based notational techniques are not enough to deal with the world of process. Each current standards proposal addresses a single piece of the picture: BPEL for internal organization processing, CDL for organizational interaction, the UML (and other associated metamodels) for data design, and so on. But what is really needed is a step-by-step guide that brings together high- and low-level perspectives, interactions of all kinds (system-system, system-human, human-human), IT systems and management needs, and inter- and intra-organizational viewpoints.

For BPM to mature and become a stable business practice, its technologies must be applied using a complete and consistent process-based method. Taken together, Riva (for process architecture and process modeling) and Human Interaction Management (for human-driven process modeling and management) provide the overarching framework for 21st century business technology. Each is the outcome of academic research and industry practice going back over 20 years, gaining critical mass with the landmark work on Integrated Project Support Systems (IPSEs) carried out by Praxis in the late 1980s and early 1990s – so they have the weight of experience behind them.

Neither book is easy to come to grips with, but, then, neither is enterprise BPM. The complexity of enterprise process modeling and management is all the more reason to manage them properly. And that means approaching them in the right way – a consistent, complete way founded on extensive research, long experience, and careful thought. This is what Ould and Harrison-Broninski provide. The BPM practitioner needs to read both books carefully. They provide a new, long-awaited and much-needed foundation for enterprise business process management.

Sue Bushell is the Contributing Editor of *CIO Magazine*, Australia, and an independent writer assisting high-tech corporations in communicating with their customers through white papers, articles, and books. In addition to her acclaimed business technology and BPM articles in *CIO Magazine*, she is a major contributor to the landmark book, *Extreme Competition: Innovation and the Great 21st Century Business Reformation*, and can be reached at sbushell@bigpond.net.au

Notes

¹ "Taking The Pulse of BPM," March 22, 2005, David McGoveran takes the pulse of the current BPM market by talking to five key industry analysts. <http://www.bpm.com/FeaturePrint.asp?FeatureId=155>

² "What Is Business Process Modeling," July 20, 2005, Mike Havey (author of *Essential Business Process Modeling*, 2005, O'Reilly), <http://www.onjava.com/pub/a/onjava/2005/07/20/businessprocessmodeling.html>

³ Smith, H., Fingar, P., *Business Process Management: The Third Wave*, Meghan-Kiffer Press, 2003.

⁴ Snowdon R.A. , 1988, "A Brief Overview of the IPSE 2.5 Project," *Ada User*, Volume 9, No. 4.

⁵ Harrison-Broninski, Keith, *Human Interactions: the Heart and Soul of Business Process Management*, Meghan-Kiffer Press, 2005.

