

Going To Sea In A Sieve: A Response to the BPTrends Email Advisor of November 15, 2005

Keith Harrison-Broninski

In the November 15, 2005 BPTrends email Advisor, Paul Harmon writes:

BPMN is currently the best available notation for business managers and we ought to rally round it and put aside our favored idiosyncratic notations, for the common good. If our goal is to get business people to focus on processes and to learn to express ideas about processes in graphic form, we need a common process language. It will make everything easier. I suggest we all use BPMN.

Paul describes the enormous amount of work that has gone into achieving consensus on BPMN, and questions why any other notation should be necessary. As he rightly says, standardization is for the common good.

But standardization on what, in this case? BPMN is certainly the best available notation *for the type of business processes for which it was designed* – the processes that act like what computer scientists call “automata”. But there are other kinds of business process. Bear with me here, reader, since we need to understand what an automaton is before going further.

In plain language, an **automaton** is a machine that has different “states” at different times – a state can be thought of as a set of values for the machine’s properties. Further, an automaton is limited in its behavior, in that it cannot freely move from any state to any other state – from each particular state, there are only certain other states that it can go to next.

This notion of automaton is the basis of much of computer science. In particular, it underpins both the “process formalisms” known as Petri nets and pi-calculus – and these are in turn the basis of BPMN. So BPMN has (as Paul says) good claim to be the best graphical tool available for depicting business processes that act like automata.

But do all business processes act like automata?

I have been arguing for some time now, both online and in print, that *human collaborative activity* is not like this at all. There really are two kinds of business process: “mechanistic” and “human-driven”.

Mechanistic business processes are largely implemented by machines, with human involvement limited to key decision and data entry points – they are not unlike computer programs in their essential nature. Most processes involved in order-to-cash and supply chain are like this, and even some service processes (such as insurance claim handling) can be treated in this way.

Human-driven processes, on the other hand, are quite different. They are fundamentally collaborative, dynamic and innovative. Consider such everyday business activities as product design, IT outsourcing, complex sales, marketing, company growth/merger/divestment, health care, human resources and so on – such processes do not act like automata, since they are full of sudden jumps from one place to another, for two reasons:

1. People take action based on a variety of uniquely human responses to situations – if you do not recognize this you are burying your head in the sand. People change their mind, make provisional decisions, are affected by impulse and emotion, and allow one thing they are doing to affect other things they are doing. It’s just the way we are, all of us.
2. Human-driven processes are subject to continual change. This change is not a superficial aspect of such a process, but embedded right into it. Much of the work humans do is “deciding what to do next” – we change our processes as we go along, in

other words. The ability to do this well is often what makes a person successful. It may represent a large part of what they are employed to do.

Hence, trying to model human-driven processes as automata is like going to sea in a sieve. You spend your whole time baling out, since the equipment you are using is profoundly unsuited to the task at hand.

Actually, to deal properly with human-driven processes you need a lot more than an appropriate modeling notation – you need patterns for using that notation, and techniques for process *management* that support the continual process change endemic to human collaborative activity. For more on this, see www.human-interaction-management.info. However, let's stay with notation for now.

The simplest way I have found to represent collaborative human activity is a version of the notation known as Role Activity Diagrams (RADs) - stripped down to remove unnecessary complexity, and reinterpreted to match how people really behave. So I use such RADs to depict human-driven business processes, build software based on these diagrams, and have always found them to have the following advantages:

- Business people can understand the notation immediately without training
- You can generally reduce a complex set of documents to a single 1-page RAD.

References to articles and books about this use of RADs can be found at the link above. For the purpose of this article, the point is that RADs, reinvented in this way, provide a means of showing human activities that allows the people who will carry them out to understand what is going on, and that is amenable in principle to software support. To achieve this, the notation must *correspond to the processes in question*. And since humans are not automata, BPMN doesn't fit this criterion.

So what about standardization? The issue is easier to resolve than it may seem. What we need is not a single notation for both types of process – it is the ability to use the same software tools to design (and possibly support) both types of process. We must standardize on *process exchange* - and fortunately, we already have the ability to do this. For some years now the OMG has been working on a common means of representing any objects, both types and instances, whatever their provenance (known as **MOF**). There is even an XML dialect available for importing and exporting such objects between software tools (known as **XMI**).

So any process that can be saved in XMI format can be opened in any MOF-compliant tool – of which there are many, Rational Rose being only the most well-known. Further, since such tools typically integrate *process design* with lower-level *program design*, this is a very productive and practical route for process support via enterprise IT. The fact that there exist 2 diagrammatic representations of the process objects, according to whether they belong to a mechanistic or a human-driven process, then doesn't matter. What matters is the process content.

To finish, it is worth noting that in principle, BPMN could be bent and twisted to represent human-driven processes as well as mechanistic ones. Just as any "Turing-complete" programming language can in principle express any conceivable program, any process modeling notation containing a minimum set of constructs can express any possible process. However, this is not to say that one programming language - and one process modeling notation - is all that we need.

With respect to programming, it is taken as read by all system developers that different languages are appropriate for different circumstances – some are higher level but less efficient, some are customized with special constructs for mathematical calculation, some offer the means to build dynamic Web sites with minimum effort, and so on. IT is an engineering discipline, in which the professional chooses their tools to match the needs of the situation, rather than applying the one-size-fits-all approach of a hobbyist.

Similarly, with process modeling, BPMN may well be a sensible choice for any process that genuinely deals with automata: i.e., computing systems with human involvement at key points

only. However, applying BPMN to collaborative human activities is uncomfortable at best. Humans are not automata, as shown above, so modeling their collaborative activities in BPMN is tortuous and long-winded for any real-world process, especially given the continual change that is inevitable once the work starts.

It is easy to overlook the fact that simple “textbook” process examples do not in any way illustrate the true nature of human collaborative activity. As discussed in Appendix B of the Process Modeling Group Workshop Proceedings from June 2005 (<http://tinyurl.com/84bsj>), a description of a human-driven process may appear to be realistic while actually being far too simplistic. Once you start including the true complexity of the activities that are involved in even quite everyday processes (the example in the document is a Travel Agency), the modeling effort can quickly spiral out of control unless you adopt an appropriate approach.

In real-world practice, as opposed to textbook examples, modeling human-driven processes with BPMN is so cumbersome as to be effectively unusable. One team that attempted to model a Systems Engineering process in such a way persisted for 6 months then simply gave up in frustration. A person responsible for re-engineering IT support in a global manufacturing company spent several months drawing up process diagrams which no-one used, or could even understand, before turning to RADs and finding that everything they had written so far, and more, could be expressed in a single 1-page diagram – further, a diagram that elucidated the processes concerned so clearly that all involved were able to agree on a re-engineered version within 2 meetings. The diagram in question is available online as a Case Study at <http://www.human-interaction-management.info>.

In fact, using an inappropriate notation for human-driven processes is more than difficult – it is dangerous, since such an approach if implemented in process support software can only lead to systems that are unnecessarily restrictive. The people charged with doing the work will then be forced either to bypass the system or to do their work badly. Through no fault of their own, people’s work will be compromised by the method used to describe it – as happens all the time with existing workflow systems, when it comes to processes requiring genuine innovation and collaboration.

I listed above some examples of the many fundamental business processes that are human-driven rather than mechanistic. And there exist many more such processes that are not specifically commercial but nevertheless lie at the very root of our society: political/social negotiation, natural disaster prevention/management, crime solving, epidemic control, government policy implementation, running an election campaign, military action, and so on.

We got to grips with the management of mechanistic business processes in the 20th century – and now it’s the turn of human-driven processes. Perhaps if we were to deal with such processes more efficiently with the aid of IT, the world would be a better place to live in. We can make a start by settling on the right tools for the job.

Keith Harrison-Broninski (keith.harrison-broninski.info) is a consultant, writer, researcher and software developer working at the forefront of the IT and business worlds. He is:

- Author of the landmark book “Human Interactions: The Heart And Soul Of Business Process Management”, described by reviewers as:
[a key component of] the overarching framework for 21st century business technology a must read for Process Professionals and Systems Analysts alike
www.human-interaction-management.info
- Contributing “thought leader” to the BPMG book “In Search Of BPM Excellence” (BPMG, 2005, www.bpmg.org)

- Author of numerous articles on the topic of Human Interaction Management (HIM) and the Human Interaction Management System (HIMS)

There are a number of collaborative activities that go beyond workflow and knowledge management, which I call human interaction management, and that is going to be the next envelope pushed in the whole BPM space.

Peter Fingar, interview for CIO Magazine Australia, September 2005

- A founder member of the Process Modelling Group (www.process-modelling-group.org).

Along with his research and consulting work, Keith is now the CTO of Role Modellers Ltd. (www.rolemodellers.com). The company mission is to develop understanding and support of human-driven processes across industry, a field that Keith has pioneered. Their new product, humanedj (www.humanedj.com), will be released in 2006.