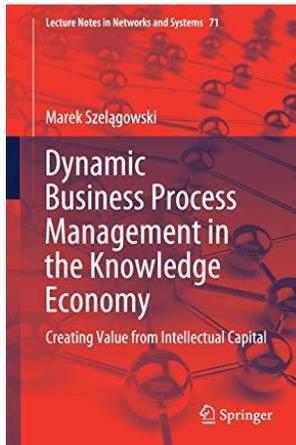


## **Dynamic Business Process Management in the Knowledge Economy: Creating Value from Intellectual Capital.** **By Marek Szelaowski.** **Springer, 2019. \$119.99 pp213**

**Reviewed by Paul Harmon**



Starting around 2000 the OMG and other groups began to work on a modern process design notation. What evolved and has become the widely accepted international standard was BPMN (Business Process Modeling Notation). This notation system supported about a dozen different specific notations and one could use whichever one worked best with the specific process problem that one faced. In spite of this seemingly comprehensive approach, and the flexibility built into the notation that allowed new more specialized notations to be defined and added, by 2010 there was a growing feeling that a new notation was needed. This feeling was driven by the fact that BPMN was designed to generate code. In fact, early implementations didn't completely satisfy this goal, but as time passed and new editions were issued, BPMN became increasingly rigorous and the very rigor that allowed the graphic diagrams to generate computer software code made software people feel that BPMN described processes in too static a manner.

In passing, the concerns about BPMN being too static never bothered business analysts much because they were primarily interested in describing business processes and not in generating software code, and so they were already used to the idea that the code provide a picture of the business situation rather than thinking of it as a rigorous algorithm.

The first break with BPMN came when process theorists at OMG began to think of defining a Decision Modeling Notation (DMN) that, in effect, described how business rules could be associated with processes to capture the logic involved in complex decision-making.

Later, this idea was extended and process companies began to think about how inferencing and rules (Artificial Intelligence techniques from the 80"s) could be incorporated into process diagrams to determine when certain processes would be used. In effect, *triggers* were added to process flows that specified that IF certain preconditions applied in a specific case, THEN a given set of processes would be used.

Capturing triggers and business rules in a process notation make it possible to think about much more complex business processes than those typically considered in the Nineties. Recall that the first large and complex business process diagrams were created to model the assembly of automobiles. These flow diagrams specified what hundreds of different individuals did, each modifying an ongoing car assembly in one or a few specific ways. The diagrams might have been large and looked complex, but, in fact, at any given station along the production line, a specific individual followed a very precise set of steps. This kind of application is where the original use of the terms procedure or workflow diagram came from.

Now consider a very different process: a potential patient comes into an emergency room at a major hospital complaining of various pains and symptoms. The physician who meets with the patient is faced with a huge number of possibilities, from fatigue or a heart attack to a knife wound or Ebola. To make matters more complicated, each major category of possible problems leads to an additional branching tree with anywhere from dozens to hundreds of additional possibilities. Add to that the fact that certain possibilities require immediate action. In some cases the receiving physician ends up beginning treatment for several different possibilities, simultaneously, just to assure that one of those possibilities doesn't prove fatal before it is even clearly diagnosed. Add to everything else the fact that each previous patient is rather different. Some are old and male, others young and female. Some are allergic to certain drugs or are already on drugs that limit what other drugs can be prescribed, etc. etc. Some have previously had one lung removed or are diabetic.

The hospital admission process isn't anything like the auto assembly process. We can define a very high level process: admit, diagnose, run tests, evaluate results... but beyond the high level sequence of very generic processes, each patient requires a slightly different set of specific actions. Each case is different. Indeed the most common way of talking about dynamic business processes, like hospital admission, is to speak of case management. We create a new file (digital or otherwise) when the patient arrives and we begin a new case. As we gather information we begin to define the specific case and to consider possibilities.

An early expert system, Mycin, was used to diagnose one rather specialized set of problems – meningitis infections. Mycin had over 10,000 rules. Imagine each of those rules as a little bit of an algorithm: An If...then branch with say three branches. Then imagine 10,000 such rules linked together to form a huge branching process diagram. This would be much more complex than anything developed to model auto assembly, and it would only be a model of the decision process that goes into the analysis of one specific set of diseases.

The analysis and design techniques that worked when we analyzed the precise steps taken to assemble a new car, don't work as well when we begin to analyze the complex, dynamic cases involved in hospital admissions. Of course there is

considerable overlap and many techniques are similar, but they are arranged in different ways and new techniques must be added and integrated.

OMG's theorists, as they moved beyond rule-based ways of representing decision making processes, ultimately arrived at what they termed Case Management Modeling Notation (CMMN). The basic idea here was to model a process as if it were a case. At a high level, each case would consider certain basic topics. Depending on circumstances, triggers would fire, adding new sets of processes or activities to the things the users would want to consider or do. Working one's way through such a CMMN diagram was very like working through an expert system, triggering only those sets of rules one needed, depending on the specific context, to solve the particular case.

Most companies are not using CMMN. Most companies have enough to do to simply model their major processes using conventional BPMN techniques. But most companies aren't going to be around long. A quick glance at the business market suggests that hundreds of publishers and booksellers are being replaced by Amazon. The same company is well on the way to replacing a huge number of retail merchants selling everything from soap to furniture. Our economies are increasingly being dominated by giant companies that rely on the Internet and AI techniques to serve customers via cell phones and computers, and there is no reason to think this trend will let up soon. A few highly digitalized high tech companies are where the money is being made today, and they will increasingly hire the best talent to create highly dynamic business processes that will cement their domination.

Dynamic Business Process Management is the wave of the near future. Any process analyst who wants to survive the next couple of decades will want to master dynamic business processes and develop a keen sense of where and how the various dynamic techniques can be used to model and integrate the Internet, social media and AI apps of all kinds. Think of business processes that interface with facial recognition and voice recognition that routinely recall buyer patterns from hundreds of early purchases and thus guide the customer toward a tailored product best suited for his or her specific needs.

*Dynamic Business Process Management in the Knowledge Economy: Creating Value from Intellectual Capital* by Marek Szlagowski is the most interesting book I have seen that focused on dynamic BPM. Mr. Szlagowski has been a process practitioner, and IT manager and a CIO. He is currently a researcher at the Systems Research Institute at the Polish Academy of Sciences. He writes well and provides clear explanations. His book is divided into four sections, one on traditional BPM, one on dynamic BPM, one on the problems of acquiring and using knowledge, and one on implementing dynamic BPM applications.

Szlagowski hits most of the right themes and provides lots of examples. I was hoping for one, in-depth, sustained case study that would provide a step-by-step picture of how one developed a CMMN application, and I didn't find it. Nor, for that matter, did I find a well-defined step-by-step methodology describing how to approach and accomplish a CMMN application. Perhaps it is too early for that. Neither DMN or CMMN have been in final release that long and few companies have used either to build significant applications. What Szlagowski does provide is lots of specific examples of problems one will face, intellectually or concretely, as one tries to build more dynamic applications and provides examples of approaches that get results.

As in other areas of new technology, a company is probably advised to choose a tool vendor who is actively working in dynamic business processes, has experience helping other lead-edge companies, and rely on the vendor for practical advice – keeping in mind that new technology evolves very fast and that other practices will probably work better in 3-4 years.

Case management, intelligent business process management or dynamic BPM, whatever you call it, is becoming more important. Process practitioners will want to learn about it. It's probably too early to find a book that can hold your hand and walk you through a dynamic development effort, but its not too early to find a book like Dynamic Business Process Management in the Knowledge Economy that can start you thinking about what's involved, tell you of some early successes, and give you some practical advice about the kinds of issues you need to learn more about.

## Author

### Author



Executive Editor and Founder, Business Process Trends In addition to his role as Executive Editor and Founder of Business Process Trends, Paul Harmon is Chief Consultant and Founder of BPTrends Associates, a professional services company providing educational and consulting services to managers interested in understanding and implementing business process change. Paul is a noted consultant, author and analyst concerned with applying new technologies to real-world business problems. He is the author of Business Process Change: A Manager's Guide to Improving, Redesigning, and Automating Processes (2003). He has previously co-authored Developing E-business Systems and Architectures(2001), Understanding UML (1998), and Intelligent Software Systems Development (1993). Mr. Harmon has served as a senior consultant and head of Cutter Consortium's Distributed Architecture practice. Between 1985 and 2000 Mr. Harmon wrote Cutter newsletters, including Expert Systems Strategies, CASE Strategies, and Component Development Strategies. Paul has worked on major process redesign projects with Bank of America, Wells Fargo, Security Pacific, Prudential, and Citibank, among others. He is a member of ISPI and a Certified Performance Technologist. Paul is a widely respected keynote speaker and has developed and delivered workshops and seminars on a wide variety of topics to conferences and major corporations through out the world. Paul lives in Las Vegas. Paul can be reached at [pharmon@bptrends.com](mailto:pharmon@bptrends.com)

[Facebook](#)[Twitter](#)[google](#) [plus](#) [Share / Save](#)