

March Sponsor



CONSULTING & EDUCATION

Learn how our comprehensive, integrated BPM methodology can work for your organization.



BPTrends Associates

EXPAND YOUR KNOWLEDGE

LEARN MORE ▶

BPM COURSES & CERTIFICATE PROGRAMS

- At one of our locations, or
- Onsite at your location – with your team



Corporate Education Group

+1.978.649.8200

How People Make Decisions Within Processes

Imagine production processes that are managed by supervisors who must make monthly unit production run decisions. Imagine that one supervisor routinely orders the right number of units, and each of several other supervisors routinely orders too few or too many units. Without going into the details of the cost of maintaining inventory, or restarting the production line to make a few more units when the supply falls short, we can't get precise about the costs, but suffice to say that good decisions save the organization money while bad decisions cost it money.

Imagine that it costs an oil company \$5 million dollars to drill a test hole for oil. On average, only 1 in 20 holes actually results in oil that can be exploited. An expert, who is right 2 in 20 times, saves the organization \$5 million in test drilling costs and generates additional revenue by enabling the organization to bring more oil to market, sooner. Every decision doesn't need to yield dramatic results. Often, just yielding better than average results is very valuable.

A decade ago, process analysts were much more likely to focus on the sequential flow of activities and on whether or not activities produced value. Today, many organizations have eliminated most non-value adding processes and their process analysts are focused on improving how decisions are made within their processes. As our examples suggest, a slight improvement in the decisions that are made in a process can lead to a major improvement in the organization's performance.

The Logic of Decisions

Let's begin by considering the logic involved in decision making. Decision theorists talk about a search space and starting and ending or goal states. In essence, one applies some type of decision rule over and over, generating all possible paths from the starting state to the goal state. An example is provided by the decision map shown in Figure 1 [1]. In this case, we are looking at a game that is played on a "board" with five squares in a row. One begins with two nickel coins on the left and two dime coins on the right. The starting and ending states are provided in red at the top. The rules and legal moves are provided in red at the bottom left. The decision space search is shown. The conclusion, in this case, is that there are two paths one can follow to reach the goal state.

Obviously we could have chosen a simpler problem where every decision resulted in a binary choice between true and false or 0 and 1. We can classify decisions by asking how many branches could be taken at any given point, and how many branching points there are.

Some problems, like the nickel and dime problem described in Figure 1, are easy to search. It has 27 states to consider. Others are harder. We could describe a search space where there were 10 branches at each point and where we kept branching through 20 points as a search space of 10^{20} states. Searching a space with 10^{20} states would be beyond the capacity of most of today's computers. Searching the space for chess, with its 64 squares and 16 pieces, each with different possible moves, results in a search space of 10^{120} events. This is such a large search space that it will never be searched using the systematic approach we used in Figure 1.

BPM

Business Process Management Conference Europe 2012

18-20 June 2012 London

- 2 co-located conferences
- 5 conference tracks
- 13 pre-conference seminars



NEW
Business Process Modeling 2011

A BPTRENDS SURVEY REPORT

BPTrends Associates
BPM TRAINING
 delivered in **AUSTRALIA and NEW ZEALAND**
 by
leonardo consulting

Torque management
 BP Trends Business Process Education
Now Available in EUROPE!
 Gain professional certification with our BPM courses
 need more details? [Click here](#)
BPTrends
 BUSINESS PROCESS TRENDS

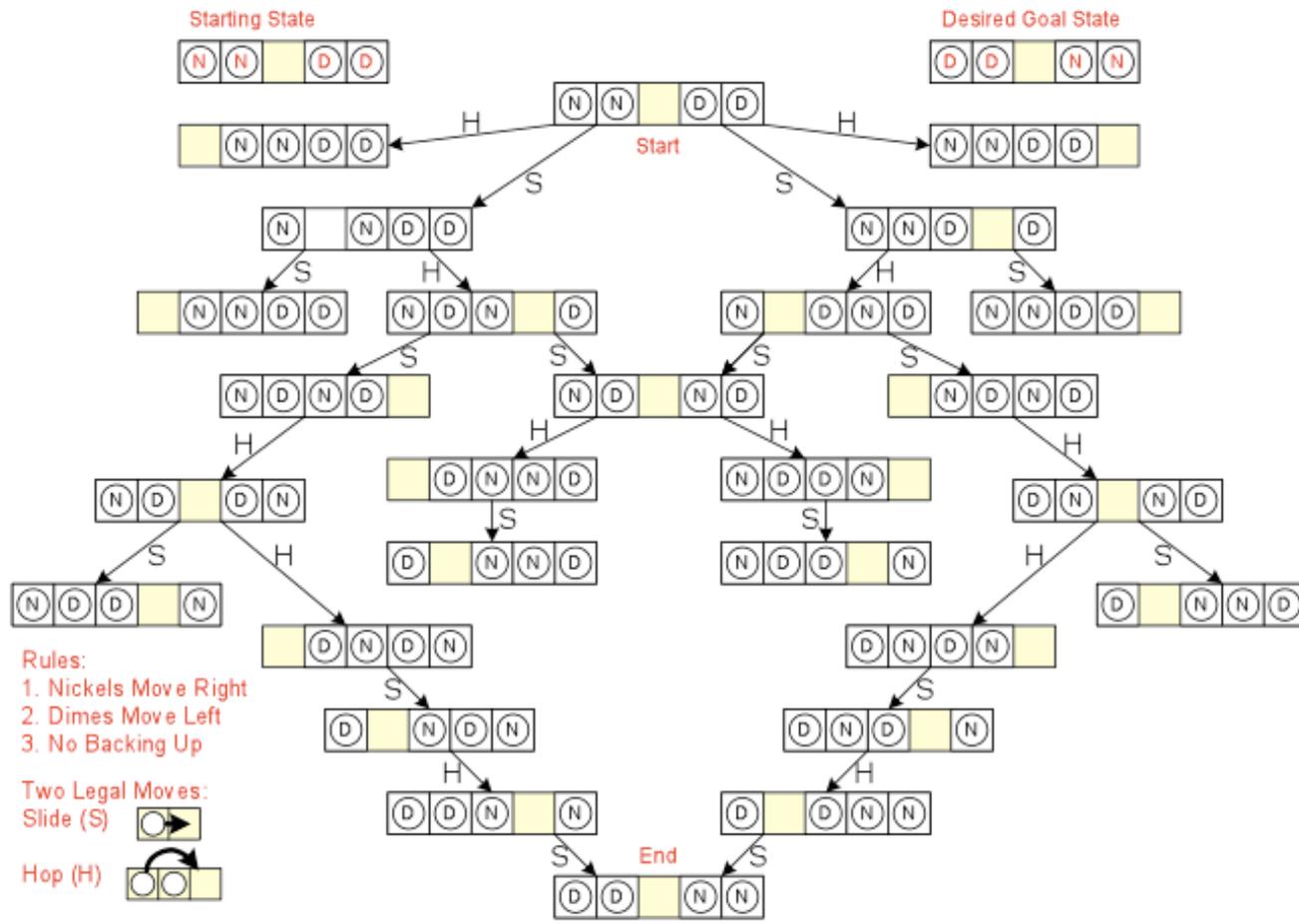


Figure 1. A search space for the nickel and dime problem.

To search very large search spaces, we shift from trying to identify all of the possible paths through a logical search space to relying on heuristic or experience-based rules. These rules assert that some things must be true and that other things can't be true. In some cases we associate probabilities with the outcomes. Using large numbers of rules, we can generate consistent decisions that are right in most cases.

Good expert systems were created in the Eighties that could do as well as the best oil experts in choosing places to drill test holes, or that could do as well as physicians in diagnosing complex illnesses. These systems were developed as a result of Artificial Intelligence (AI) developers working with human experts to formalize human knowledge in rules. In essence, one went over cases, asking the human expert: "Why did you make that decision? How did you know that you should do x rather than y?" It took hundreds of hours to capture all the rules that a human expert might use to solve a complex problem. The resulting systems weren't perfect any more than human experts are always correct, but they were as good as the best human experts and, in many cases, that is as much as we can expect.

Expert systems that could match real human experts in the diagnosis of diseases or the analysis of geological sites usually required 8,000 to 12,000 rules. By the end of the 1980s, two things happened. First, it was determined that it was too costly to maintain large expert systems. Human experts were constantly learning new things at a very rapid rate and the task of constantly updating a 10,000 rule expert system knowledge

global solutions



REGISTER TODAY!
[CLICK HERE](#)

bbc BUILDING BUSINESS CAPABILITY
 OCT 28 - NOV 2, 2012
 FORT LAUDERDALE, FL

Business Process Change
 A Guide for Business Managers and BPM and Six Sigma Professionals
 Second Edition
 Paul Harmon
 President, The Strategist
 MCF

NEW
 BPTrends Discussion Group
[Linked in](#)

base wasn't cost effective. At the same time, AI experts began to explore different types of decision systems that more closely mimicked the massive parallel neural networks found in the human brain. Without describing how Neural Nets work, they can learn in real time and deliver better results than the large rule based systems developed in the 1980s. Neural Nets and related technologies are at the root of newer data mining software and analytic tools.

Policy-Driven versus Experience-Driven Rules

Expert systems were created by working with human experts and formalizing their experiential knowledge as rules.

At the same time that AI practitioners were working on expert systems, relational database theorists began to work on storing procedures in databases. The idea was to make it possible to not only capture data, but also to manipulate the data in the same database. They quickly arrived at the idea of storing rules in relational databases to capture procedural information. Unlike the expert systems that AI developers were working with, most of the rules that database analysts wanted to include in their systems specified actions or constraints.

The work by the IBM Users Group and others resulted in the Business Rules movement, initially focused on deriving rules, top-down from policy statements. If the company had a policy of not granting credit, then there were rules that applied to various possible activities that forbade the granting of credit under various specific situations. From an expert systems perspective, this is, logically, pretty trivial, but Business Rules Systems had the immediate advantage of being relatively easy to maintain. Unlike Experience-Derived Rules, that required a human expert to re-conceptualize how to solve a problem in the light of new empirical results, if an organization changed its policy and decided to grant credit under certain circumstances, Business Rules Systems made it possible to quickly find the rules that needed to be modified.

Notice that I am not considering rules that tell a computer what to do. I am only concerned with rules that a human might use to make a business decision. We may choose to capture those decision rules in a software system, but the ultimate justification for the use of the rules is that they are used by humans to solve a business problem.

An interesting thing happened in the mid-Nineties. A number of software vendors ended up with expert system-building software tools that were no longer needed by those who had given up on building large expert systems. After some thrashing around, many of the vendors decided to reposition their tools as Business Rules tools. This, in turn, introduced the Business Rules folks to the idea of declarative rules that could be manipulated by Business Rules engines (algorithms) – systems that could begin with a goal and search individual rules, one at a time, to see if any possible solution was possible. In essence, expert systems tools became Business Rules tools – a role they could easily fulfill, since the Business Rules practitioners were making many fewer demands on the tools than the expert systems developers had made.

Processes Flow Models and Decisions

Ultimately, we want to end up considering how to actually improve decisions in business processes. Before we go there, however, there is another thing to consider. How do we represent decisions in process flow diagrams?

There are two schools of thought about showing the results of decisions on process flow diagrams. One school would use activity boxes and arrows and avoid showing decision points on a process flow model. This works fine on very high level models where branches defined by specific decisions aren't shown. As you drill down, and look at more detailed subprocesses, however, it is often the case that one moves from a specific activity, A, to either activity B or C based on a decision taken within activity A. Thus, for example, activity A might

describe a process for determining whether or not a bank is going to give a loan to a given business applicant. If the loan is to be granted, one moves to activity B, which begins with informing the applicant and inviting him or her to sign a loan agreement. On the other hand, if the loan is to be denied, one moves to activity C which informs the applicant that the bank can not grant the loan, and which may or may not involve suggesting alternative services. (See Figure 2.)

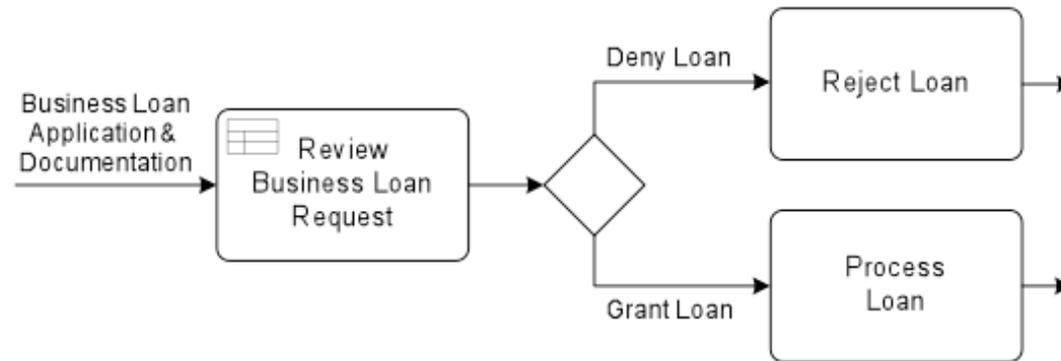


Figure 2. Process flow for business loan processing

The thing that bothers some purists in this model is that the decision to grant or deny the loan takes place **within** the Review Business Loan Request activity box, but the ultimate result is shown outside the box. Technically, the boxes on a flow diagram are where activity occurs and the arrows simply show which activity occurs next in the sequence and, placing a decision diamond outside the Review Business Loan Request activity box seems to confuse the two basic flow concepts. I have always found this to be a quibble. I try not to use too many diamonds, but at the appropriate level of decomposition, showing that a branch occurs and that some activities follow one outcome while another set of activities follows another seems a very practical thing to picture. I have never had problems explaining this to business users and they have never suggested they were concerned about the syntactical issues.

My concern here, however, is quite different. The diamond and the branches only show what results from the decision taken inside the Review Business Loan Request activity box. Some people refer to the diamond and its branches as a business rule. It is, but it's a very trivial use of the term. (E.g. If you don't grant the loan, do activity B next, else do activity C next.)

The real decision about granting or not granting the loan occurs within the activity box. (In the latest version of BPMN, there are symbols one can use to designate where rules are used in decisions that occur within processes. We have placed such a symbol on the Review Business Loan Request to indicate that business rules are involved in the activity.) If we think of the decision in terms of business rules, it probably involves anywhere from 100 to 250 rules. The loan team will consider the assets and liabilities of the business, the purpose of the loan, the ability of any item acquired to serve as collateral, and the degree of trust the bank has in the managers of the business. They may also consider outstanding balances they have with similar companies in the same industry (exposure to an industry downturn) and their projections of economic growth over the life of the loan, etc. If we were creating a set of business rules to formalize this decision process we would need multiple rules to cover each of these considerations. Or, to return to our discussion of problem solving, we would need to define the decision space for business loans, and define all of the rules we would use to determine how to get from the application submitted to a decision to either grant the loan, or deny it.

The point I would emphasize is not to worry about the diamonds and the branches. If you are serious about improving how decisions get made in business processes, then focus on activities that require decisions and look inside the activity boxes to determine how the decision is made. A simple activity box can easily use

hundreds or thousands of rules to make decisions. How you document the rules that are used within a given activity box is not something I want to explore here. Suffice to say that it wouldn't be efficient to try to subdivide the Review Loan activity into sub-activities or to somehow incorporate the actual rules in the flow diagram. If you are using the right kind of software modeling tool, you might be able to double click on the Review Loan box and open up a spreadsheet on which you could enter the rules, or, the software tool might simply move you to a separate business rules environment in which you could capture or review the rules to be used in making the decision.

If your analysis is limited to policy-driven rules, you would probably not have all the rules required to make the decision, but would, instead, have policy constraints that the employees would use as they make the decision. If you are using experience-driven rules, which have been generated from studying loan experts at work, there might be hundreds of rules and they might be used to completely automate the decision process. This level of detail is not our concern at this point.

Capturing Rules for Decision Making

Today, many BPM people are focused on trying to formalize decision making activities. Simple decisions can often be captured in decision tables or checklists. [2]. The more complex decisions rely on business rules, usually managed by software that serves, in effect, as a decision support system.

If one isn't careful, it's easy to get carried away trying to capture any and all decisions. This will lead to a level of complexity that isn't productive. Consider what I have already said about expert systems. We are still not ready to capture complex decisions that involve rules derived from human experience. We are, on the other hand, able to capture modest decisions that depend on human experience, and rather complex decisions based on policy-derived rules. Consider Figure 3. I've created a matrix that considers two things. First is the level of abstraction you are working at and second is the complexity of the decision.

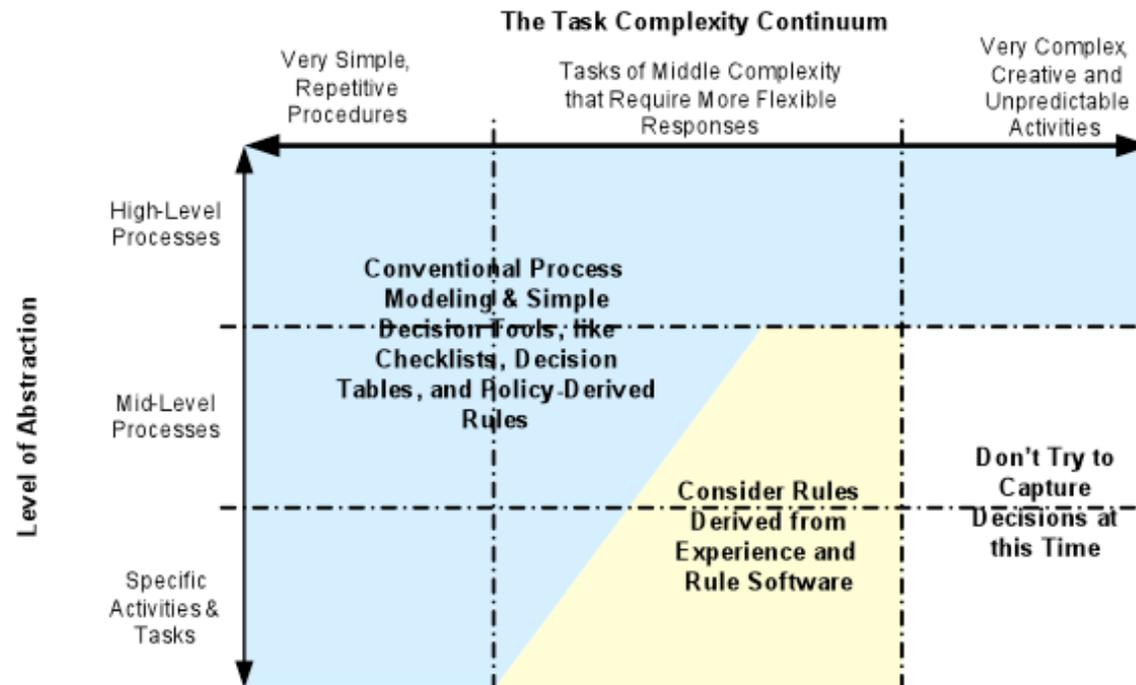


Figure 3. Levels of Abstraction and Task Complexity

At a high-level of abstraction, we can model any process as a sequence, and we don't need to focus on decisions in any detail. Consider the development of a new product (an auto design, a new drug, a patient diagnosis). At a high level, we can show the steps that the developer goes through. What we do not want to attempt is drilling down into the details. Figure 4, for example, provides one possible description of a high-level medical diagnosis process. At this level of abstraction we can define the steps the physician should go through.

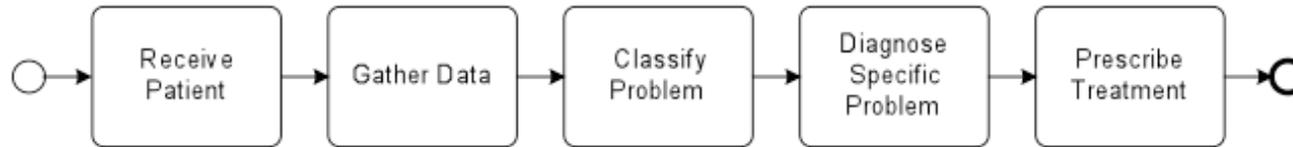


Figure 4. High-Level View of Medical Problem Diagnosis Process

We could easily decompose some of the processes described in Figure 4. We could, for example, spell out some of the steps a physician might go through to gather data or to classify a problem. But we will probably not want to drill down into the details of diagnosis. It's simply too complex and patient specific.

The problem in drilling down into a diagnosis is not that there are alternative possible paths. If it were simply a matter of choosing between alternative paths, it would, in my opinion, be a problem of middle complexity. This is the kind of thing we ask medical technicians to do, and we can define this in a step-by-step manner, with a checklist or decision table.

Looking at Figure 3, you can see that we've divided it into three areas. Some processes include activities that involve relatively few decisions, or at least decisions that can be easily formalized. Using checklists, decision tables or perhaps a few dozen business rules, we can tell employees, or a computer, exactly how to make the decision. Any process group ought to try to formalize those processes immediately.

A second area is where the interesting work is being done. These are more complex processes that require more complex decisions. In terms of rules, we are perhaps talking about hundreds of rules with a mix of Policy-derived and Experience-Derived rules. This can be done today, and maintained for a modest cost. Process practitioners ought to be exploring these possibilities and formalizing them where they can.

On the other hand, there is still an area to the right of Figure 3 that should be avoided. This area describes complex processes that depend on the extensive use of experience. You might be able to describe these decisions with rules if you put in the years of effort required to create an expert system like Mycin (which diagnosed meningitis infections) but you'd never be able to afford to maintain it. In the same vein, don't think about trying to analyze the CEO's job, or the decision process he or she will go through to identify a successor. You can do it at the top level, of course, as we did in Figure 4, but don't try to drill down. Similarly, don't try to analyze how a software architecture should be structured, or how your best sales people decide to pursue some prospects and not others, or how your best designers come up with creative marketing campaigns. In all these cases, hire the most experienced human talent you can find.

Capturing the decisions involved in processes is a great place for process people to focus. Formalizing the right decisions can improve your organization's efficiency and its profits. But over-reaching and trying to do too much is a recipe for failure.

Till next time,

Paul Harmon

[1] This example is described in more detail in *Expert Systems: Artificial Intelligence in Business*, which I coauthored with David King. (Wiley, 1983. p.27)

[2] For a powerful introduction to the use of checklists to support both procedures and decision making, see Atul Gawande, *The Checklist Manifesto*. (Henry Holt & Company, LLC, 2009)

BPTrends LinkedIn Discussion Group

We created a BPTrends Discussion Group on LinkedIn to allow our members, readers and friends to freely exchange ideas on a wide variety of BPM related topics. We encourage you to initiate a new discussion on this publication, or on other BPM related topics of interest to you, or to contribute to existing discussions. Go to LinkedIn and join the [BPTrends Discussion Group](#).

:: [email us](#)

:: [Visit BPTrends](#)
