



"The iThink software provides a framework in which I can work with both 'hard' data, like financial and engineering quantities, and softer relationship-type information. Building the diagram helps create a shared systems level understanding of a complex organization."

*--Organizational Development Executive,
Motorola*

Case Study

The Need for Visual Thinking

Organizational Development (OD) work is among the most challenging in business. What makes it so challenging is that an organization's structure and culture are suffused throughout all that it does. Thus, unlike people in the functional areas (e.g., manufacturing, sales, engineering, finance), OD practitioners can not focus on accomplishing a particular functional task. Instead, they have to work the issue of how the functional pieces play together.

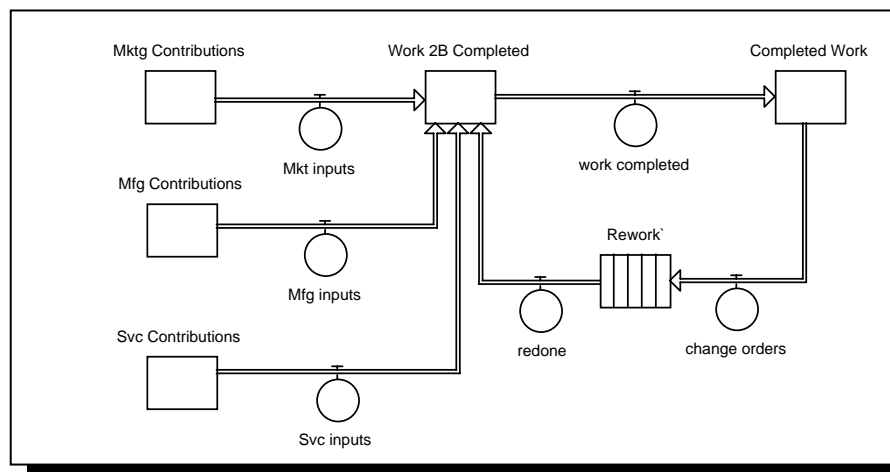
Traditionally, OD practitioners have relied primarily on qualitative tools and a high level of interpersonal skill to get their jobs done. Increasingly, however, companies are looking for a more rigorous approach – one that can relate OD contributions directly to the bottom line. Furthermore, as organizational structures become more complex, and the interdependencies between functions become more critical, strictly qualitative analysis simply isn't good enough any more.

The *iThink* software can help OD practitioners to meet the need for more rigor, without doing violence to the often inherently qualitative nature of OD subject matter, or requiring that the OD practitioner be a "quant jock." The *iThink* software is easy to use, and offers a generic, operational language for depicting organizational relationships which cross functional boundaries.

The Setting: A major computer company
The Topic: How to organize to reduce cycle time.
The Challenge: Building effective cross-functional teams

Background: One of the key questions in almost any high-tech company is how to reduce cycle time – the time from conception to introduction of a new product. One of the major reasons why cycle times are higher than they need to be is a lack of coordination across functions. To deliver a good product, all functions must contribute. But each function typically has its own goal-set, and reports to a functional boss who rewards for achievements relative to this goal-set. The question is how to make the pieces play together.

Step 1: Map the Current Process. To be effective, cross-functional teams must have a clear, shared understanding of how each member's contributions add value to the product's bottom line. This often is difficult to achieve because, in many ways, manufacturing operates in a very different world than engineering, marketing or service. An excellent basis for shared understanding, in this case, began with creation of an *iThink* map of the product development process (a simplified version appears below). The *iThink* software's generic building blocks offer an operational language that transcends function-specific dialects. The *iThink* map of the new product development process served as a focal point for numerous fruitful discussions.



Step 2: Model. After the basic map had been laid out, the next task was to incorporate the team's assumptions about how the map worked. It was this web of interdependencies that the team had to learn to manage – as a team! The *iThink* software makes it easy to include even very sophisticated behavioral assumptions. The graphical function is one way to do this. In this case, the change over time in engineering's "willingness to listen" to inputs from team members from other functions was one of the critical relationships captured this way.

Step 3: Simulate. The *iThink* model of the product development process provided an overarching context for cross-functional team discussions. Now, when, say, manufacturing or service said they needed something included in the design of the product, a simulation would show the whole team the bottom-line implications of both including, and not including it. As such, the simulation output served as a positive, operational context for resolving cross-functional issues.

Step 4: Celebrate! The *iThink*-based product development model became part of a new process for managing cross-functional teams. Team members met on a regular basis to discuss progress on the product and associated issues. Whenever there was a question relating to whether a particular feature should become part of the product (or should be eliminated), each team member provided their best guess as to the resulting impact on a set of function-specific variables represented in the model. Following the collection and discussion of inputs, the model was simulated, and the bottom-line implications – over the product's life cycle – were laid out for discussion. As a consequence of this process, cycle-times were reduced, and overall product profitability enhanced. Teams really began to operate as teams.

