Designing Quality In

A growing interest in *integrated product development* (IPD) as an approach to reducing time to market for new products has brought research and development functions into intimate contact with quality management practices. Engineers and designers, once considered independent contributors, have become members of cross-functional product development teams (PDTs), working to produce better products faster and with lower life cycle costs.

The cultural shock waves of this transformation are hilariously documented in the panels of the *Dilbert* comic strips, but the Council for Continuous Improvement (CCI) provides serious guidance for organizations trying to surf those waves. Through General Sessions (and the *Proceedings* that document them), CCI members can hear the testimony of experts (like MIT Professor Don Clausing) as well as share the experiences of their peers.

IPD (also known as *concurrent engineering*) is an interdisciplinary effort which requires team skills that can only be acquired through training and practice. The benefits, however, make the effort well worthwhile. “I have seen concurrent engineering reduce the number of change orders by 75%, compared to the standard design and manufacturing process,” **Mark Knodle**, a consultant with extensive direct experience in manufacturing, reported at a recent General Session. Based on eight years of working with the concurrent engineering process on a daily basis at a major manufacturer of heavy equipment, Mr. Knodle’s presentation focused on many of the details involved in implementing such an approach.

The integrated approach helps developers understand the significance of early design decisions. Decisions made in the concept phase typically drive 80% of the life cycle costs of an entire program, Mr. Knodle asserted. “Curiously, many companies try to shave money in development, but the more successful organizations put additional people up front to help them make the right decisions.” The IPD process helps bring the voice of the customer directly into the design process, avoiding costly change orders later in the product life cycle.

Designing for Manufacturability

Producibility is another important issue that IPD raises for product developers. Not only must a product meet continually evolving standards of customer satisfaction: it must do so in ways that are continually easier to achieve. “We are trying to simplify, standardize, and make the product more manufacturable,” Mr. Knodle said.

At another General Session, **William Lareau** of the **American Samurai Institute** addressed the principles of design for manufacturability (DFM) in greater detail. He presented a checklist that included sensible practices like *easy operator access and visibility*, *reduced part count*, *fewer processes*, *eliminate opportunities for incorrect assembly*, and *minimum transport*. Part count, Dr. Lareau noted, is a primary driver of cost, and even greater savings can be achieved by *eliminating* unnecessary processes rather than merely improving them.

Some of the DFM principles were less obvious, but equally important, such as *eliminate fasteners* and use the *lowest acceptable technology*. Dr. Lareau described fasteners as "deadly melanomas growing on the skin of your profitability." He strongly recommended commonizing fasteners to reduce the number of parts, lower material costs, and reduce time and effort devoted to purchasing, stocking, handling, assembly, and servicing. Regarding technology, Dr. Lareau observed that customers frequently request the latest technology, whether it is required or not. He urged CCI members to work towards shrinking the distance between customer *wants* and *needs*.

"If you look at the problems your products have," Dr. Lareau suggested, "you'll find that they involve issues addressed by these manufacturability principles. But if the information isn't there at the design stage, you can't expect your engineers to be aware of these concerns."

Don Clausing, Professor of Engineering Innovation & Practice at the **Massachusetts Institute of Technology**, echoed those concerns in another CCI presentation. "As engineers, we are very good at doing tasks right," Professor Clausing said, "but we have not been very good at picking the right tasks. We learn a partial design that emphasizes creativity and feasibility. But it takes more than creativity and feasibility to make a successful product."

Making Teams Work

Mr. Knodle, Dr. Lareau, and Prof. Clausing all pointed to teamwork as an essential element in the success of IPD projects. Prof. Clausing cited an MIT study which found "failures of cooperation" to be a significant factor in the weakness of American product development in the 1980s and early '90s. He observed that the traditional serial process ("throw it over the wall") inhibits teamwork.

Mr. Knodle emphasized the importance of having all involved functions properly represented on the product development team. This is particularly critical for the engineering and design component of the team, as it enables them to accurately incorporate customer and manufacturing requirements into their designs from the beginning. However, Mr. Knodle noted several challenges for team builders: Having everyone on the team from the very beginning is probably not cost effective, and managing participation on very large teams is also difficult, so downstream functions usually play a support role in the early stages. On the other hand, consistent membership on the team is also important, since accountability is threatened if people are constantly rotated on and off a team. "Most successful teams I've seen have two layers of membership: a core team and auxiliary members who support the team on a part-time basis. Typically, auxiliary members will support more than one program."

Dr. Lareau observed that a focus on engineering tools often ignores the important dimension of human interaction. "Throw away the old engineering management approach," he advised. " You can't standardize this process with policies and procedures. Once you write it down, you crush the intuition and creativity you were trying to nurture." He recommended co-locating the team — putting them in the same room for contiguous blocks of time to develop team cohesion — but he also warned against *groupthink*, which can suppress critical thought within teams, leading to irrational decisions. To combat this tendency, he suggested assigning each team member the role of critical evaluator, encouraging feedback from outside experts, and revisiting decisions even after consensus is reached ("second thoughts meetings").

Software Solutions

The principles of concurrent engineering are not limited to manufacturing applications. In another recent presentation at a CCI General Session, **Craig Kaplan** of the **I.Q. Company** discussed his experiences at IBM's Santa Teresa software laboratory, where they developed these concepts even further in an effort to reduce the cycle time of their software design process.

The Santa Teresa laboratory formerly used a sequential "waterfall process" for software design. "First you do design," Dr. Kaplan explained. "Then the output of the design goes to the people doing the coding; then the output of the coding goes to testing, and finally out into the field. It took us two years to get a product out the door because there were too many serial steps in this process." They realized that they could reduce the cycle time if they made the process more parallel. For instance, they could involve the test department in the beginning stages instead of waiting until development was done, so that testing could work in parallel with the people who are developing the code.

"That was our first step," Dr. Kaplan said. "But we can do even better than that using a *train process*, with teams running on a schedule like a train station." Each software release is like a train scheduled to depart from the station on a particular date. Product requirements for that release are the individual cars that compose the train: each car consists of different line items that are to be included new release, such as new functions that customers want. A team — including a tester, a developer, and an information developer — is associated with each "car." Each team is responsible for getting their car ready to go on the train; the train will pull out, whether their car is on it or not.

"These guys would work together like crazy," Dr. Kaplan reported, "sharing information from the beginning, trying to get their car onto the train. If they missed the train, then they were really under pressure to make the next train that pulled out, or there was hell to pay for missing the deadline. It really reduced our cycle time."

Pointing out that many people are accustomed to habitual ways of doing things, Dr. Kaplan warned, "You will encounter resistance. Some of the programmers who had been there for 20 years did not want to hear about this new thing. We had to make sure that someone communicated the need for change: the way we're doing things right now *isn't working*, and we need need new solutions."

Dr. Kaplan offered other suggestions for implementing such changes. "People who are traditionally from different areas have to work together on this new team — they must be willing to engage in cross-functional teamwork. Find customers who really want to try this new approach. That helps encourage management buy-in." Management must commit resources: they must staff and support the team, while communicating the new idea throughout the organization, monitoring the results, and feeding it back to the team.

Inspiring Change

The cultural and organizational obstacles to change are particularly evident in engineering and design disciplines, where traditions of creative independence have been institutionalized. However, the necessity for change in these disciplines — embracing the principles of teamwork and the methodologies of quality management — is equally evident. The networking opportunities offered by organizations like CCI represent powerful tools for effecting change. By communicating both theoretical information and practical experiences, CCI members focus on solutions, avoiding costly trial and error experimentation. Furthermore, directly sharing experiences among peers can help motivate and inspire reluctant individual contributors who remain suspicious of teams and TQM tools.