Four Reasons to Take a Rapid Prototyping Approach to Your PLM Implementation

by Richard Mizuno and Daniela Ilieva

Many companies view product lifecycle management (PLM) implementations the same old way they've viewed enterprise system implementations (like ERP) for years. As a result, companies tend to follow the same waterfall implementation approach. The waterfall approach, which originates from very structured project methodologies in the manufacturing and construction industries, is a linear and sequential approach to software implementations. Waterfall approaches rely heavily on written specs, and leave testing and user feedback until late in the development cycle. A waterfall approach works well when there is a desire to consciously limit user options and force the adoption of out-of the-box functionality, or when trying to impose de facto requirements on the user community in an effort to drive compliance (which is really effective in financial or reporting systems).

However, in today's fast-paced, highly visual and interactive world, it is difficult and unrealistic to drive enthusiasm and deliver strong results from PLM with an implementation approach that is documentation-driven, slow-moving and highly bureaucratic. Rapid prototyping or agile development, which is an iterative and collaborative approach characterized by short "requirement --> build --> test" cycles to design a product, has proven to be quite an effective technique for product development. A rapid prototyping approach lends itself well to changing requirements and facilitates learning from short trial-and-error prototyping cycles, which helps to accelerate delivery while ultimately maximizing the value of the system.

Maintaining agility during a PLM implementation, which typically involves a high degree of process redesign and evolving business requirements, and adopting a strategy that embraces a more iterative and collaborative solution design approach, has significant benefits and can lead to more optimal PLM solutions.

Here are four reasons why a rapid prototyping approach can lead to a more robust and adaptive PLM solution:

1. **Allows adaptation to changing or evolving requirements**. Most organizations set about PLM implementations with an eye toward streamlining and harmonizing the underlying product development process, with business and functional requirements likely evolving and becoming clearer over time. It's impractical to expect the user community to be able to define and agree on the complete set of requirements in advance. Users don't always know what they want until they see it.
2. **Demonstrates requirements with functioning prototypes instead of paper specifications**. With the paper designs of a waterfall approach, users need immense powers of deduction and imagination to extrapolate what their solution is going to look like. The rapid prototyping approach relies on working prototypes that allow users to see, touch and use what is essentially a beta version of their solutions. The quality and completeness of user feedback is more robust when users can engage with working models.
3. **Iteratively matures the solution over time while delivering incremental value with each release**. PLM applications lend themselves well to deployment strategies that call for the staged release of incremental functionality over time. For example, start first with CAD data management, then add document management, later add engineering change management, and so on until the total solution has been deployed. An iterative design approach helps capture incremental value as the solution is deployed and matured with each subsequent release.
4. **Consistently engages key stakeholders throughout the project**. The rapid prototyping process also provides an effective means of engaging users and stakeholders throughout the project, by seeking their feedback and gaining alignment as the solution is being built instead of waiting until end-user training just before "go-live."

What's Your View?

Have you used a rapid prototyping approach to implementing PLM with good success? Did you struggle with a waterfall approach?

Originally published on March 25th, 2013

What's your view? Add your question or comment

About the Authors

Richard Mizuno

richard.mizuno@kalypso.com

Richard likes to avoid housework by coaching, refereeing or watching (but certainly not playing) a game of lacrosse whenever he can.

Daniela Ilieva

daniela.ilieva@kalypso.com

Daniela loves outdoor sports, skiing and traveling.