Beyond PLM: Eight Tips for Bridging Product Data between PLM and Other Systems

by Prashant Jagarlapudi

Product Lifecycle Management (PLM) systems have been front and center for product development and engineering organizations for many years. An integral part of any PLM system is the product data record. It defines the data structures for raw materials and finished goods, and houses quality specifications, supplier relationships and any relevant trade information for the end product. As part of implementing and evolving a PLM solution, organizations gather product data from various sources across the company and create a product data record that is tracked through the innovation lifecycle. This means portions of the product data record often originate or terminate in systems beyond PLM.

As organizations mature in their PLM implementations, they need better ways to integrate data between PLM and these other systems. Integrations eliminate the need for duplicate data entry, reduce errors in data reconciliation, and enable faster data propagation between systems. These benefits help organizations make product decisions faster, using the latest valid data in their designs.

Bridging the gap with effective integration is a struggle for many companies in all industries. PLM implementations in different industries have slightly different integration needs. For example, common integrations in the food and beverage industry include:

- Raw material coding systems to PLM
- Output materials and formula BOM to manufacturing and ERP
- BOM integration to procurement systems
- Nutritional information to/from Laboratory Information Management (LIMS) Systems
- Trade information to data pools (e.g., GS1)

Common Integration Considerations

While it may be tempting to treat PLM integration projects as IT initiatives, this can cause companies to fall short of their integration goals. In many cases, the business users of these organizations’ systems use tools and processes that are not integrated cross-functionally. Introducing integration between systems and functions that have previously operated autonomously will inevitably create significant change to the way the organization works. Established ways of operating will have to be reconsidered and users will need to begin considering their jobs in the context of how they affect other functions.

Whether you are implementing a new PLM system, or maturing your system by adding new integrations, these eight tips will help ensure the success of the integration initiative, realize the business goals, and deliver an error free solution.

1. **Drive system integrations via business use cases**

   Business use cases provide the rationale for pursuing an integration project. Business use cases often identify requirements that are not immediately apparent when first considering the integration. A common challenge when integrating PLM with an execution system like ERP, for example, is how to handle the effectivity of a material or packaging specification change. If a change from PLM is simply propagated to downstream ERP and supply chain systems, item records in those systems might be changed before inventory or production plans can be adjusted. Integration logic must accommodate the business needs of both sets of users. Clear business use cases for engineering change management can clarify the technical requirements for this interface.

   Integration projects cannot be viewed only as technology initiatives. Identify the business value of integrated data, then work with business users to define owners and processes to maintain the data. After that, a joint business and technical team can work together to identify the requirements for the integration.

2. **Invest time in data mapping**

   A data mapping exercise is required for each system touch point in the integration. While a similar activity might have been performed during the PLM implementation, it was probably limited to the scope of PLM. As part of the data mapping activity for an integration, organizations need to consider the data mapping rules of both PLM and the other systems. There are often discrepancies in how data is represented. For example, PLM might consider a recipe to be a master blueprint, while ERP might require it to be facility specific.

   Functional areas that have previously operated in non-integrated silos often define a given attribute differently. While they may have worked around these differences in the past, an integrated future will require them to align their definitions. Invest time in data mapping between...
systems to ensure a smooth translation of data with no need for workarounds.

3. Build a cross functional system integration team

To complete data mapping and understand the direction and frequency of updates of data across various modules, successful implementation teams require representation from both systems. The integration team should consist of technical architects and functional subject matter experts for both systems. A joint integration team should be put in place for every integration effort, with resources allocated to various roles and functions. PLM integration work needs to be part of a broader project plan.

4. Pay attention to error handling

Integration project teams are often tempted to focus solely on the business requirements as defined by the user. However, problems like system outages, user errors, and data corruption can occur, and a successful integration design must account for this. The project team must address how the system will respond when such failures occur. Clear error messages, graceful failures, alerts, and caches are just a few of the techniques that should be considered.

5. Provide sufficient time and resources for testing

Just as with a PLM implementation project, system integration projects should allow for sufficient testing. Set time aside for unit testing of the interfaces exposed on both sides of the integration, and conduct a system integration test based on the business use cases identified at the start of the project. Be prepared for integration testing to be complex. By definition, it will involve multiple technical and functional teams, and resolving issues can be complicated. Testing should be planned conservatively with ample time budgeted for remediation.

6. Invest in training and support

Integration components require the same level of training and transition between project teams and the support organizations. In some cases, certain power users of the PLM system should be assigned to help train and troubleshoot issues that the business users may encounter as they utilize functionality covered by the integration. This will help ensure wider adoption and support by the business team.

7. Build room for ambiguity

System integration is never a smooth process. Despite careful planning, there may be bumps in the road and unknowns to overcome. The quality and completeness of data across multiple systems can vary. The use of systems across business units might depend on localized regional business rules. Project managers should plan for system and business ambiguity during the integration process, and business sponsors should be informed of these issues through regular program management meetings.

8. Place system integrations on the PLM roadmap

To be adequately funded and supported by the PLM organization, the integration effort should be prioritized on the PLM roadmap as a business-driven initiative. This provides visibility and support from the various stakeholders that are required to make the integration effort successful.

As organizations mature in their PLM implementation, they can identify and reap benefits from integrating their PLM platform with adjacent systems. The improved operational efficiency and quicker visibility of data can be a competitive advantage. However, if this effort is taken on as a technology-focused implementation, it can experience the same pitfalls as that of a poorly run PLM implementation.

Originally published on June 3rd, 2014

What's your view? Add your question or comment
About the Author

Prashant Jagarlapudi
prashant.jagarlapudi@kalypso.com
Prashant is a senior manager specializing in product lifecycle management (PLM) and innovation analytics.