Competitive Edge through Product Customization

How a Modular Product Architecture Increases Product Options, Reduces Costs, and Locks In Satisfied Customers

Abstract

Industrial manufacturers are facing unprecedented demand to deliver multiple product variations–which is driving up both costs and complexity in product development. The fact is, in every industry today, customers want more products that are specific to their unique requirements—and they want them quickly at an attractive price.

To win, industrial manufacturers must adapt to this mandate by quickly creating customized products in a profitable, high-quality way. For many forward-thinking manufacturers, finding a 'customization solution' requires looking at traditional manufacturing processes in a completely different light, and ultimately transforming their product architecture to support 'modular' design.

This white paper explains how a Modular Product Architecture (MPA) applies to industrial manufacturing, and how an MPA can increase the number of product options that you offer to your customers while decreasing your costs and time-to-market. But moving to an MPA isn't simple, and many executives and managers wonder:

- Do I need a Modular Product Architecture?
- How does an MPA help me create customized products more easily?
- What are the benefits compared to a traditional product architecture?
- What are the challenges of defining an MPA, and then implementing it?

This paper can serve as an important primer for senior executives and all managers in product development who are facing the growing demand for more customized products at lower costs.

Implementing Modular Design for the Industrial Equipment Market

Customers–both consumer and industrial–are driving a global trend of demanding more customized products. In late 2007, *Forbes* magazine named 'personalization' as one of the "10 Industrial Design Trends You Can't Ignore". At the same time, a *ThomasNet Industrial Market Trends* article identified "Personal" as one of the Top 7 Design Trends heading into 2008.

As buyers continue to shun the "take it or leave it" mentality of mass production in favor of customized product development, top manufacturers are responding by offering more and more options and product variants. And while offering more product variations does bring a competitive edge and improves customer satisfaction, managers must find a way to deliver these customized solutions while keeping costs low and meeting ever-tighter schedules.

In the following pages, we'll explore how a Modular Product Architecture enables industrial manufacturers to support a full portfolio of product variants, and we'll discuss the processes you need to follow in order to successfully implement modular design for your products.

Why do you need modular product design?

To meet the demands of customers, industrial manufacturers must deliver a wide range of product variants that are specific to customer, market segment, or regional needs. Consider the following scenario typical of an industrial equipment manufacturer struggling to compete in a saturated market. To differentiate itself from the competition and ensure consistent wins, a manufacturer offers custom products that are tailored to the client's needs and requirements. Not surprisingly, more customers sign on. This manufacturer's sales team, in order to make a sale, readily agrees to customer demands without a realistic understanding of the time and effort needed to implement the requested design changes. These promises create a tremendous challenge in managing customer expectations based on what the customer now expects for their price, and what's reasonable to deliver.

In this all-too-common scenario, each design must be built to meet the unique needs of each particular customer, which stretches engineering bandwidth thin. Because there is little design reuse, newly created parts must go through an extensive QA and testing process, which jeopardizing multiple schedules. As timelines shift, it becomes harder for project managers to estimate resource needs, making scheduling more difficult. And, because the scope of the project is open to interpretation by the customer and the sales manager, late-stage changes are inevitable, which negatively impacts development timelines and costs. Ultimately, the need for more resources and longer timelines drives product development costs up, and lowers profits.

Figure1 –Typical Industrial Pain Points Driving Modular Product Implementation

Current Pain Point	Eventual Result
Profitability	• Shrinking margins–even when revenues are increasing
Market complexity	• Struggling to respond to custom requests • Difficulty addressing the needs of specialized market sub-segments
Product complexity	• Need to increase part reuse • Testing times delaying time-to-market
Development complexity	• Need to outsource portions of the product • Need for distributed engineering
Service complexity	• Product is too expensive or too complex to service

The challenges mentioned above can best be described as those of a traditional–or 'integral'–product architecture, which consists of components and sub-assemblies tightly integrated and highly dependent on each other. Although it offers cost and performance advantages for one-off products, an integral architecture can make product development unwieldy as the volume of demand for new variants increases. An integral architecture is also very costly when designing multiple variants, and requires considerable testing and performance-tuning because customization increases product complexity.

An alternative is a Modular Product Architecture (MPA), where product options and variations are predefined to support easy customization. An MPA separates the components of a product design into reusable design elements, or "modules", like product building blocks. An MPA enables industrial manufacturers to offer customized solutions without performing extensive design rework, because products are assembled to meet the customer's specific needs, using specific combinations of these design building blocks. Since these building blocks are fairly independent of each other and have well-defined and stable interfaces, product design can be easily customized for each client by adding or omitting modules, depending on customer requirements. If a change is necessary, its overall effect can be limited to that particular module, instead of forcing a designer to completely rework an entire product design.

Modular Product Architecture also reduces the complexity of customization, making it applicable to a wide range of industrial products. For complex industrial products produced in low volume–such as printing presses, packaging machines, and power plants, managing product variations isn't a major challenge. Indeed, Modular Product Architecture is quite beneficial because it helps reduce overall product complexity–for example, the number of parts and part interactions, and electro-mechanical development complexity.

For industrial companies producing complex products in medium volume–such as agricultural equipment, earth-moving equipment, industrial machinery, turbines, industrial engines, and mining equipment, a Modular Product Architecture not only helps manage complexity, but also allows manufacturers to offer specialized product variants at lower costs.

For less-complex products produced in high volume–such as small appliances, power tools, and white goods, a Modular Product Architecture helps manufacturers quickly and cost-effectively generate market-specific product variants from a single product platform.

By properly implementing a modular design and ensuring that the new product architecture is fully aligned with the business strategy, industrial manufacturers can achieve impressive results. Specifically, manufacturers can achieve five key benefits—discussed in greater detail below–that are crucial to future success:

1. Balance product variety requirements with development costs
2. Control product complexity, increase standardization, and improve design reuse
3. Rationalize procurement operations and leverage outsourcing opportunities
4. Optimize manufacturing processes and assembly sequence, and minimize inventory costs
5. Minimize maintenance and servicing costs to better enable ongoing revenue streams

1. Balance product variety

A Modular Product Architecture helps manufacturers provide customers with more options, by allowing modules to be "mixed and matched" to create many flavors of the same product–yet limits how many versions of the product are in the marketplace by eliminating most one-off customizations. In order to support a product platform strategy and to cost-effectively provide product variations, an optimized modular architecture must contain the lowest number of modules possible to meet customer needs. Only by clearly understanding market requirements for the product variations up front, by agreeing on a platform strategy, and by explicitly defining the modules for the product family, can you then establish the optimal module design.

Modules are often referred to as one of the following: "common"–that is, found in all versions of a product within a product family; "variant"–meaning that alternate modules can be substituted; or "optional" - meaning that a completed product may contain this module, but it is not required. These 'common, variant and optional' modules are then combined to create each required product variation. By sensibly choosing an optimum modular architecture, you not only can increase the proportion of product variants to common modules (resulting in larger volumes and economies of scale), but also better manage the number of optional and variant modules.

For example, a leading Japanese producer of printing machinery recently needed to reach different segments of the market, each having its own specific product needs and required features. With an integral product design process, it would have been very costly to develop new products for each of these markets; moreover, getting the products to market on time would have been nearly impossible. Through a Modular Product Architecture, this Japanese manufacturer was able to serve these varied market segments at a lower cost by offering segment-specific modules that "plug in" to a common product platform. Because the majority of the product–the set of "common" modules–was identical between versions, only these small, segment-specific modules needed to be designed to tap into the new markets. As an additional realized benefit, their Modular Product Architecture shortened both RFQ response times and product testing times.

A well-defined modular architecture, combined with a manageable amount of variant and optional modules, also helps organizations establish consensus between cross-functional departments on what's "included" in a product. As a result, the ongoing proposal and scoping of new features and functions is more feasible. When a customer makes a request to your sales team for a feature that is outside normal product capabilities, the sales team is better able to recognize this as a change request. In this way, fewer engineering resources are required to scope the change request, since the impact is limited both to the functionality itself and to a limited number of integration points. Each change can then be evaluated on an individual basis, for instance:

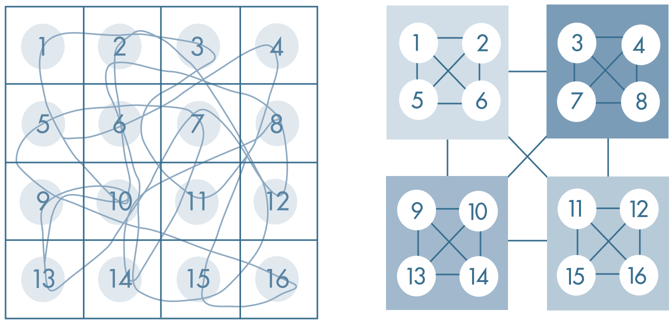
- Is it reasonable to customize this module for this client?
- Is this a new module that needs to be introduced to the product platform as a whole?
- Is this change unfeasible, given the constraints in time and resources?

Moreover, by developing a carry-over strategy—or a plan to reuse significant elements of earlier product models in new releases–manufacturers can ensure design reuse across product generations at the module level, as opposed to the broader component level.

2. Control product complexity, increase standardization, and improve design reuse

A Modular Product Architecture also helps manufacturers reduce and manage product complexity by initially segmenting a complex system design into chunks (modules), and then explicitly managing the complexity by defining and managing the critical interfaces between modules. In this way, the design can be evaluated, communicated, tested, and shared in smaller, more manageable pieces. Modules are then assembled to comprise a larger system. Changes to the design become more manageable because they are isolated to the affected module and its set interactions with the larger system (see Figure 2).

Integrated Design Modular Design



I=m-(m-11)2

Upper bound of 120 interactions per product.

Still an upper bound of 120 interactions, but with a modular architecture, 6 interactions are confined to each module, and the system has 6 interactions between the modules.

Figure 2: An integrated architecture versus a modular architecture.

A Modular Product Architecture also increases design reuse, and in turn, standardization, by limiting the overall number of design elements with which engineers are allowed to work. With a traditional, 'integral' product architecture, each product variation represents a unique design; component parts and electrical and mechanical interfaces throughout the product are tailored to that specific instance of the product.

For example, two product variations of a loader may each have unique lift arms despite similar bucket designs. Each lift arm includes washers, bolts, plates, and other components specific to that particular design. Despite the similar buckets, the bucket from one variation wouldn't fit the second variation because the mechanical and electrical interfaces don't match up. Because of the unique aspects, engineers working on future product variations of the loader have an infinite number of historical design versions to reference. Even if current customer requirements have been addressed in past designs–for example, a lift arm that can support a certain weight and a bucket that has a certain capacity–these designs must still be altered so that the interfaces are consistent, and later tested as a complete system.

With a Modular Product Architecture, on the other hand, designers work with predefined product modules with set interfaces. These predefined design 'chunks', like the lift arm and the bucket, are associated with specific sets of customer requirements. In the previous example, the two engineers, now working in a Modular

Product Architecture, have a choice between five possible lift arms and three different buckets. Even if each engineer chooses a different lift arm based on their customer's requirements, they can still use the exact same bucket because the lift arm is not custom to any specific product instance. In terms of part and component standardization, this now means that both product variations use the same set of washers, plates, and bolts–along with other components–for both buckets. When a new lift arm needs to be designed to meet a previously unmet customer requirement, the engineer has five lift arm designs to reference, as opposed to the potentially infinite number in the integrated architecture scenario.

Because of this manageable number of designs–each associated with a set of requirements–the likelihood of elements from the existing bucket versions being carried over to the new design is much greater. This reuse of designs also makes the design process much more efficient, and speeds time-to-market.

3. **Rationalizing procurement and outsourcing**

Once the design is segmented into modules, it becomes easier for organizations to leverage third-party resources to deliver aspects of product development. Modules can be separated from the larger design and developed concurrently, either by internal teams or third parties. Because modules have predefined parameters, the process of evaluating the ones that need to be outsourced, versus those that can be developed in house, can be made earlier in the product development process. Third-party vendors can be evaluated and selected before a customer order has even been placed, removing this step from the development timeframe. When a module is outsourced, its limited number of interfaces makes it is easier to integrate back into the larger product. This also helps protect your intellectual property, such as CAD models and design specifications, because only information pertaining to that module and its interfaces is shared with the third-party organization.

An example is a major distributor of global mail processing systems, whose products include equipment that manages inserting, sorting and weighing mail, and affixing postage. The equipment design architecture was separated into three modules: the user interface (UIC), the input module, and the finishing module. Because of its Modular Product Architecture, the company was able to outsource the design and manufacture of all three modules. In addition, because the interfaces between the three modules were standardized, different suppliers could work on each of the modules, designing and manufacturing them concurrently. Not only was the company able to significantly lower product development costs through outsourcing, but they were also able to maintain high quality by matching supplier capabilities to the requirements of the module.

The component standardization described above also makes procurement and sourcing easier, because fewer parts (in larger quantities) are being used across all manufactured products. Procuring fewer parts means that fewer suppliers need to be identified, and purchasing larger quantities tends to result in more advantageous pricing. With standardized components, you also can more accurately estimate future needs for parts based on projected orders, as your designs are now standard, and the cost benefits of creating parts in-house versus outsourcing can be more effectively analyzed.

4. **Optimize assembly, shorten test times, and minimize inventory parts**

The benefits of a Modular Product Architecture extend beyond system and detailed design into Manufacturing Process Planning, where a modular product design can improve parallel manufacturing operations. Here, new designs and engineering changes can be limited in scope to just the affected modules and their integration points; modules that aren't impacted by a design change can be passed to manufacturing while the new design is still taking place. Through component and part standardization and reuse, manufacturers can better plan for needed parts, and better control inventory levels.

Modular designs also improve testing and quality assurance. Modularity allows QA engineers to test products at the module level, as well as at the component and product levels. Testing teams can verify modules within a product or product family ahead of time–thus reducing test and acceptance time by cutting the overall number of verification tests needed for each configuration. As well, common modules don't have to be retested at the component level for each product variation.

Additionally, because modules have well-defined interfaces, there is only a limited number of integration points between modules within the broader product. This makes it is easier to anticipate the impact that changes to one module will have on the overall product, and allows verification tests to focus on potential impacts of late-stage design changes. Lastly, when product modules are carried over from one generation of a product family to the next, testing teams can reuse previously developed verification plans and limit the testing of previously tested modules–making it more efficient to verify new versions of a product.

5. **Minimize maintenance and servicing costs in support of ongoing revenue**
   A Modular Product Architecture also facilitates more effective long-term support of complex systems, thus improving customer satisfaction over the lifetime of a product. For instance, product modules can be tied to specific product information that's used to create product manuals and service documentation that precisely matches the specific configuration of the product. By working with more specific manuals, users and field service personnel alike can diagnose and resolve maintenance issues faster and easier, resulting in higher customer satisfaction. Having a set number of modules also means that training for repair technicians is easier, which not only improves the quality of their service, but also cuts down on the number of interactions required with the manufacturer.

Challenges of a Modular Product Architecture (MPA) Implementation

While many industrial equipment manufacturers are now realizing the value of a Modular Product Architecture, they are also seeing that implementing an MPA can be a rather daunting task. The challenges are two-fold: manufacturers must first redefine their existing product architectures from an 'integral' to a modular perspective; and then they must change their organization's processes and operations to support this new approach. Often, it is the latter–i.e., introducing new processes–that proves the most challenging.

Through more than 20 years' experience helping industrial manufacturers improve their product development processes, PTC has been able to identify five main considerations, detailed briefly below, that have helped organizations successfully make the transition from an integral product architecture to modular product design.

1. **Know Your Strategy**
   Often, the new Modular Product Architecture is defined by a well-meaning engineering department, whose goals are limited to improving 'ease-of-design' alone, and not taking into consideration the company's future product vision. This limited view can result in modular products that are not optimized either for your customer needs, your company's long-term vision, or future market trends.

   A Modular Product Architecture must be aligned with–and must enable–your corporate objectives, not just your engineering requirements. This alignment requires an analysis of your business strategy, so that you can prioritize the value drivers of your corporate stakeholders, and understand their importance relative to your architecture strategy. Because the stakeholders involved in the project have varied roles within the organization, these individuals also tend to have different perspectives, priorities, and considerations. While a VP of Engineering may see the main driver of a Modular Product Architecture as the need to increase innovation, a CFO may see reducing costs as more important. Once all of the drivers are understood, the entire group needs to reach consensus on the relative priorities, so that a definition of project success can be established, and expectations can be set.

   PTC uses a tool called the PTC Value Roadmap to help manufacturers prioritize business value drivers related to product development processes, which helps firms align strategy with product architecture. This unique tool is based on PTC's experience in the product development business, as well as current research from product development thought-leaders both in academia (including Harvard and MIT), interviews with leaders from over 900 manufacturing companies, reviews of published media articles, and other manufacturing resources. Using this tool, companies are able to drive discussion at any organizational level, while considering specific company issues.

   Understanding strategy not only means exploring "if" a company should modularize its product lines, but also "why". The PTC Value Roadmap makes it easy to answer the critical 'why' question by linking business strategy to product development improvement initiatives to required solution capabilities.

   For example, using the PTC Value Roadmap, the executive stakeholders of a power systems manufacturer collaborated to identify and prioritize value drivers for implementing an MPA practice. The combined PTC/customer team identified and ranked the company's highest priority value opportunities for MPA as being:

   - Improving Ability to Fulfill Demand
   - Designing for Ongoing Revenue Streams
   - Growing Market Share with Customer-focused Products
   - Lowering Product Costs
   - Improving Asset Utilization

PTC then worked with the customer to map these value drivers to their product architecture requirements–including "Increased Part Use" and "Increased Concurrent Subsystem Design"–and ranked the correlation.

For example, "Increased Part Use" was identified as having a strong correlation with "Improving Ability to Fulfill Demand", but only a weak correlation to "Growing Market Share with Customer-focused Products". Analyzing these correlations, and then establishing the relative priority of each value driver, allowed the customer to create a ranking of product architecture capabilities as goals for the MPA practice. Using this ranking, the customer was then able to put together a plan for implementation.

By aligning priorities, setting shared goals with associated product architecture requirements, and establishing a long-term plan, stakeholders can agree both on a course of action and measurable metrics of product success.

2. Prepare Your People

The new Modular Product Architecture must be defined, realized and deployed by various stakeholder groups across many internal departments. Coordinating these cross-functional resources can be a challenge. When implementing a Modular Product Architecture, organizations will run into process issues if there's unclear ownership of deliverables, or there's a lack of strategic responsibility for who will create product modules and module interfaces. As well, if there's a lack of early involvement by all stakeholders (up-stream and down-stream), this omission will lead to mismanaged expectations later in the adoption process.

Clear organizational sponsorship is necessary to ensure that the right resources from each group are identified and committed to supporting the process, especially for distributed enterprises. While the tendency is to focus on just those teams involved in module definition activities, it is just as important to identify and appoint the teams who will manage ongoing changes to the modules, as well as the engineers who will build and test the modules, and the leaders who will be responsible for managing adoption throughout the enterprise.

In terms of identifying specific roles and responsibilities, the organization must demarcate between module ownership and interface ownership. For the implementation to be most effective, it is critical to appoint a formal architecture manager who has strategic responsibility for all interfaces.

3. Understand Your Processes

Moving from an integrated product architecture to a Modular Product Architecture requires a great deal of commitment and involvement from Engineering. First, your Modular Product Architecture must be clearly defined by Engineering management. Second, the module specifications, interface specifications, and the change management process for controlling these specifications must be documented. And thirdly, the modules must be built and tested. PTC refers to these three phases as Definition, Realization, and Deployment, respectively. And though these phases require significant engineering dedication, it is not sustainable for an engineering organization to realize and deploy a Modular Product Architecture in isolation.

The fact is, you not only must have a full understanding of your entire organization's cross-departmental product development processes, but you must be aware of how modularization will impact those processes. For example, you need to understand your company's proposal response process in order to understand how the new modular architecture will affect that process in the future: How does the sales and marketing team prepare their proposals? What information do they need? How are special customer requests that are outside the current capability's scope addressed? What training will these teams need when the new architecture is rolled out? Where can users go for the information they need in their day-to-day activities? By fully understanding how the product architecture touches enterprise-wide product development processes, such as proposal response, you will be better equipped to ensure full and fast adoption.

4. Evaluate Today's PLM (Product Lifecycle Management) Technology

Another important consideration is whether your Information Technology (IT) system is capable of supporting a Modular Product Architecture. Your Product Lifecycle Management (PLM) system must be able to achieve the following:

- Create – The system must be able to create explicit product module interfaces directly in the module. It must enable the capture and development of ideas and intellectual capital into high-fidelity, structured product representations that provide realistic, interactive, and intuitive definitions of a product's look and feel, behavior, and means of production.
- Collaborate – The system must facilitate collaboration around module interfaces by coordinating module development teams across the extended enterprise. All stakeholders who participate in the processes of planning, developing, sourcing, manufacturing, documenting, and servicing the product must be able to communicate effectively.

- **Control** – The system must control the development of module interfaces as a separate function from product components, due to the strategic importance of the module interfaces. The system must: capture all product content in a single, trusted repository; automate and monitor key product development processes; and facilitate tight alignment of all stakeholders throughout a product's lifecycle.
- **Configure** – The system should drive configuration rules from the module interfaces. As well, the system must enable users to combine content components into simple or elaborate structures–thus producing higher-level deliverables such as manufactured products, services, and publications–and manage those evolving structures over time.
- **Communicate** – The system must communicate interface spec-ifications throughout the product lifecycle to enable concurrent engineering. It should drive timely and effective decision-making among internal and external parties, and ultimately deliver dynamic product content to the right audiences, in the right format, on demand.

5. Don't Underestimate Adoption

Lastly, to be successful and to deliver the expected ROI, the new MPA practice must be adopted and embedded into existing product development processes. In evaluating your adoption strategy, you must consider your organization's structure, culture and unique challenges. Lack of multi-department management buy-in to support the required process improvement (e.g., from Marketing, Engineering, Production, After-sales) can doom a Modular Product Architecture initiative right from the start.

Understanding and communicating your organization's strategy and value message, understanding the people most affected, and anticipating which cross-departmental product development processes will be impacted, is the foundation for successful adoption. With an Information Technology system that fully supports your new architecture, and a carefully thought out adoption plan, you are much better equipped to handle these challenges. PTC advocates a gradual (stepped) process improvement approach, ensuring that MPA definition, realization, and deployment practices are piloted, adapted, and fully adopted into daily development activities and development projects.

Defining, Realizing, and Deploying the Modular Product Architecture

Through our experience helping industrial and non-industrial customers implement a Modular Product Architecture, PTC has defined best practices for the three deliverable-based phases mentioned previously; Definition, Realization, and Deployment. These three phases ensure that implementation of the MPA practice is controlled and manageable, and are described in greater detail on the following pages. Figure 3 shows how these three phases fit within a generic stage-gate development process. In describing these stages, we make the following assumptions:

- Cross-functional involvement occurs throughout the three phases
- There is an increased upfront focus on system design activities in the product development process
- A commitment is made to assigning ownership and control at the architecture/system level

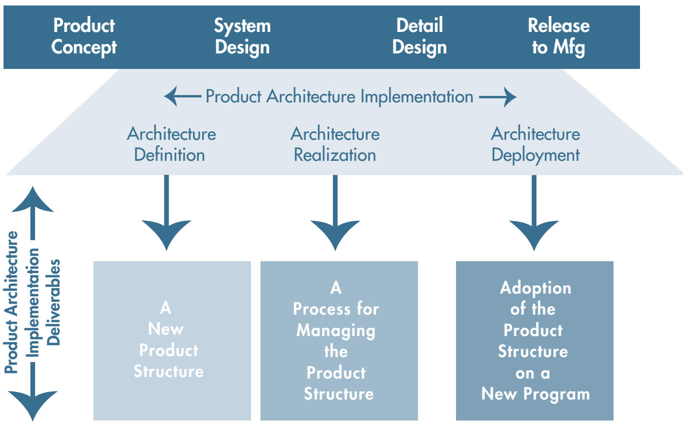


Figure 3. Three phases of an Improved Product Architecture Process, as they fall within a stage-gate product development process, and the expected outcome of each phase.

Product Architecture Definition

The Definition phase of MPA implementation is executed during system design, and is the most cross-functionally demanding phase of modular design. Here, it is critical to gain input from key functions, such as Marketing, Product Management, Manufacturing, Purchasing, Suppliers, Engineering, Finance, and Quality, to ensure that the architecture is defined, which ultimately enables the product to contribute to corporate success. Four typical system design process steps associated with a modular design approach are described below. They are sequential and based on specific deliverables, beginning with architecture requirements. The eventual result of the Definition phase is the explicit definition of a Modular Product Architecture (modules and interfaces), which forms your new product structure. Steps in this phase include:

Architecture Requirements:

- Establish business and product strategy
- Define and prioritize product architecture requirements
- Develop product variation plan

Technical Solutions:

- Define product functions and technical solutions to “sub-module” level
- Define product variant requirements
- Identify product interactions

Module Clustering:

- Develop module clustering alternatives
- Evaluate and select optimal module clusters based on product architecture requirements
- Develop commonality plan

Interface Identification:

- Identify and classify required control interfaces between defined modules
- Identify and classify inter-module interactions for risk mitigation
- Define a control interface master document

Product Architecture Realization

The Realization phase of MPA implementation focuses on documentation and control of both modules and module interfaces. The Realization steps include interaction and cooperation with the systems and organizations required to support the deployment phase. These steps are not necessarily sequential; however, the final result is an architectural plan that will be the foundation for the detail design of the product. Steps in this phase include:

Interface Management:

- Develop control interface specification documents
- Develop supporting skeleton and interface components in CAD

Module Management:

- Develop module specification documents
- Develop module space-claim components in CAD

System Management:

- Implement product data management infrastructure
- Implement change management process for product architecture (modules and interfaces)
- Implement configuration management process

Architecture Plan:

- Define detail design activities

Product Architecture Deployment

The steps below in the Deployment phase ensure that the architecture requirements are maintained throughout the remainder of the product development processes, primarily Detail Design. The primary focus is on module development and architecture management, including ownership, change management, configuration management, and control. Steps in this phase include:

Module Development:

- Module teams design/build/test against module specifications

Change Management:

- Module changes impacting one module governed by module change process. Module owners own/control module content
- Module changes impacting control interfaces governed by an architecture change control process. Architecture managers own/control interfaces

Architecture Management:

- System Engineering team maintains overall product structure and performance specifications of the system

Configuration Management:

- Product structure configurations captured and base-lined prior to release to manufacturing

Conclusion

No doubt, having the ability to quickly and cost-effectively deliver customer-specific and market-specific products is becoming increasingly crucial as a competitive differentiator in the industrial equipment market. Just the same, implementing new processes that can deliver multiple product variations can be challenging and costly. By adopting a Modular Product Architecture that makes variant design and generation and system design processes more efficient, companies can deliver improved customer service, greater product choice, and higher-quality products.

Moreover, companies can lower associated product costs, since the indirect expense (Engineering, Development, Manufacturing, Operations) of a single underlying product platform is spread across many product variants. In addition to many short-term benefits, a Modular Product Architecture also supports lower lifetime maintenance and servicing costs, and optimizes design, such that modules and components can be reused across product generations.

Despite its potential benefits, a Modular Product Architecture implementation can be challenging and time-consuming. Project leads must fully understand, communicate, and align their company's strategy and drivers for implementing a Modular Product Architecture among project stakeholders. Project leads must also fully understand the product development processes that will be affected, and clearly communicate the roles and task ownership of all members of the extended project team. On the IT side, the right technology infrastructure must be in place to support the project, and a phased process improvement plan should be put in place to ensure adoption throughout the organization, and to ensure realized sustainable benefits.

The three phases of implementation–Definition, Realization and Deployment–can only begin after careful planning is complete. Throughout the deployment of these phases, best-in-class organizations will involve not only Engineering, but also other stakeholders from all other departments, because all these parties will invariably be affected. Collaboration across all these enterprise-wide departments is crucial, and is an important factor in Modular Product Architecture adoption. Truly understanding the product development process landscape, addressing potential implementation challenges, and giving the proper weight to the "Definition" and "Realization" phases of implementation are the keys to MPA implementation success.

3307-IM-WP-0308