

Why RFQ Systems Struggle to Support Complex Manufacturing



Product launches can be stressful. They require close coordination across internal departments as well as suppliers who may be dispersed across the globe. Most organizations struggle to overcome miscommunication caused by siloed information, which results in teams working from disconnected data that cannot be easily consolidated and it is often out of date. This in turn leads to a lack of actionable data needed to make timely, critical decisions. These are threats that can impact start of production, costs, and performance quality.

Purchasing plays a critical role during a new product or program launch, sourcing BOMs created with hundreds of engineered parts. However, product BOMs can undergo numerous engineering changes as a result of feedback from the customer, Program Management, Finance, Quality, and suppliers, prior to start-of-production. It's very difficult to track the economic impact of these changes because the details are stored in disconnected RFQs.

By implementing a system that can track sourcing by the item, organizations can streamline their processes, drive transparency internally and with suppliers, and dramatically improve their product launch.

It's Buried In A File... Somewhere

A certain amount of chaos is generally accepted as typical product launch energy. The product launch process is highly complex: there can be hundreds of changes associated with a product BOM prior to start of production.

While engineering changes are managed through the PLM, most sourcing systems are disconnected from the PLM and lack the ability to properly manage an RFQ down to the part level. As a result, purchasing departments often resort to storing part-level information in external documents such as spreadsheets, which are attached to RFQs and saved on shared drives.

Additionally, Program Management sends Purchasing program enrichment information such as production volume scenarios, and other program related attributes, such as make/buy or prototype build dates, which Purchasing needs to take into account when sourcing.

Given all of this, it is difficult to track variances created by Engineering and Program Management that result in changes to items, and their impact to costs. It can be particularly difficult to determine which key data points reflect minor versus major revisions that affect pricing. For example, a minor change can be a small informative detail, whereas a major change could require a supplier quote to be retracted and requoted. To make things even more complicated, a new RFQ must be created for every change. This is challenging because a buyer needs to link these disconnected RFQs to track price evolution.

Further, spreadsheets and emails offer a snapshot of the moment they're created, not real-time information. Countless hours are spent tracking down current versions and re-confirming that everyone is working from the most accurate data — and racing to get back on track when it's discovered that they aren't. This becomes a challenge even during an “uneventful” program launch.




A certain amount of chaos is generally accepted as typical product launch energy. The product launch process is highly complex: there can be hundreds of changes associated with a product BOM prior to start of production. ◀

It's safe to assume that, at some point during every program launch, most stakeholders are working from confusing, conflicting, inaccurate - and out-of-date - information. It cannot be actionable. It derails quality. And it puts teams into firefighting mode to meet the program launch date.

Locking Down the Master Data

An efficient program launch is dependent on an integrated process, where everyone is working from the same standardized master data. This can be accomplished, but it's an entirely different approach from the current model on which most systems operate.

In a typical sourcing system, the RFQ is the highest level of consolidation. There can be hundreds of RFQs associated with a new product launch, organized by commodity, regions, and other non-item attributes. This data is in different languages and formats, not standardized: a unique item might have different metadata and identifiers in each of these systems. It is difficult to share between departments, particularly when each department is using different sets of data pertinent to their specific needs.



It is difficult to track variances created by Engineering and Program Management that result in changes to items, and their impact to costs.

For example, Engineering is disconnected from the purchasing system and typically works from engineering drawing and item numbers; Purchasing works from RFQ numbers; Quality works from PPAP numbers. Program Management creates production volume scenarios and phases. Supplier details are peppered across databases and personal contact lists.

The key is finding the underlying information that spans all processes: the master data. With product launches, that master data is derived in the BOM — the items listed in the BOM — coming from the Engineering PLM.

An Item-Centric Approach Calms the Chaos

There is another approach: an organization can consolidate by making program information the overarching top level of data of the product launch, with the item as the most specific level of information. This level of granularity enables all departments to communicate seamlessly, regardless of the number of engineering changes. The item-centric system supports an RFQ system while allowing data to be consolidated and reported at the program as well as the item level. When an item-centric approach leverages a portal to standardize and aggregate common components, each department contributes and maintains information to the item-centric master data in support of their specific requirements.

Complete and accurate data is then available transparently — and immediately— to all stakeholders involved in any program that requires the item. Engineering can focus on engineering; Program Management can focus on program management. And Purchasing is delivered all critical information needed from both Engineering and Program Management to properly source the launch.

This centralized, standardized, item-centric approach tracks each item associated with a drawing, including multiple revisions that may impact the various departments involved in launch. It monitors changes to production scenarios such as prototype build dates or capacity planning volumes. It tracks items that require different levels of quality certification, as identified in the APQP/PPAP process. It notes changes that may impact tooling and require recertification or new quotes by the supplier. The portal collects detailed and unique cost breakdowns, tied to item revision history.

For example, the system will map engineering item numbers to manufacturing item numbers, which may be unique to each plant and can include variable attributes such as color and finish. When these item attributes are linked to a master item number, it's possible to conduct robust analysis on a program, pricing, suppliers, quality, and so forth. This aids in calculating cost creep, as well.

An item-centric approach offers granular data within a few clicks. This is especially critical for highly engineered, complex components, where the risk of missing a change could lead to costly delays or errors.


An Item-Centric, Single Source of the Truth

This is where LiveSource comes in. With its Product Launch Portal, LiveSource connects every stakeholder, centralizing, streamlining, and managing the continuous change during the launch process. LiveSource offers an item-centric approach that delivers an integrated, single source of truth about an item throughout every stage of the product launch process. It solves the endemic problems of miscommunication, siloed information, and a lack of actionable data.

For many manufacturers, the PLM does not match the manufacturing item number or production configuration: one item number in the PLM may have multiple production part numbers, such as for each color option. There are several situations where this capability is important. For example, an ability to map color-specific part numbers allows purchasing to track price variances for different production part numbers. It is also critical that the sourcing system be able to identify each specific manufacturing part number and the plant responsible for it, to ensure the purchasing is complete and accurate.

LiveSource maps the engineering item number to the corresponding manufacturing part numbers in the ERPs and updates the system each time an engineering change is processed. The PLM pushes BOM and engineering changes into the LiveSource item library; data is synced with new or updated RFQs, PPAPs, and APQPs, as well.

LiveSource also streamlines the RFQ process. Buyers nominate a supplier for an award, initiating an automated approval workflow. Once approved, LiveSource forwards the RFQ information via an API with the customer's ERP, and a blanket purchase order or supplier schedule agreement is created.



An item-centric approach offers granular data within a few clicks. This is especially critical for highly engineered, complex components, where the risk of missing a change could lead to costly delays or errors.

With LiveSource, it's easy to pull history, associations, cost, and other data for each item. As suppliers provide information through the LiveSource portal, quality engineers and program managers are updated on each task's on-time status related to start of production. This creates efficiencies and increases accuracy, by reducing the need to look up or update items in multiple locations.

Managing Manufacturing Chaos

LiveSource manages the manufacturing chaos by managing changes on an item level. Working from the standardized, up-to-date master data set – a single source of truth – all departments are empowered with actionable data for decision making, communication is improved, and errors are minimized. With LiveSource, organizations are better equipped to deliver product launches on time with less chaos.