

Why CPQ IS Not Enough

08-Sep-2020



Picture this scenario. Your company's sales team is in the field. They are visiting a potential customer. All the important figures are there, sitting around the conference room table. Your sales lead is on their laptop, sharing the screen to the main projector. Running is the lightweight CAD viewer and composer software installed by your company's IT department. The sales lead never got any training on it, but the software is easy to use thanks to the building blocks created by Engineering. The sales team is listening to the customer, dragging and dropping bits and bobbles, panning and zooming around the screen, and fully impressing the customer with how your company can create a customized solution to fit their exact needs, all in real time. The customers are all sitting on the edges of their seats, captivated by how smoothly this experience has been. The conversation ends, and on the screen is a fully rendered solution. Everyone shakes hands and the sales lead presses send. The custom configuration is uploaded to the corporate servers. Now the real fun begins.

In a best-case scenario, the Engineering Manager reviews the work done by the sales team. In a worst-case scenario, that configuration is automatically entered straight into ERP and work orders

are being expedited to meet this unrealistic deadline set by the sales team. Either way, expletives are flying around the office.

“Who approved this configuration?”

“We can’t build this.”

“This will never work!”

“They promised it by when?”

History of CPQ

CPQ – Configure, Price, Quote – software started at about the same time as the personal computer, in the 1980s. As technology improved, so too did the software. The above scenario is exactly why CPQ software was created – to allow the sales teams to quickly put together a quote for the customer that is accurate and consistent among the growing worldwide sales teams. And, within that context, CPQ works great, so long as the rules that were defined in the CPQ system were setup correctly. But as technology grew, datasets moved from 2D to 3D to virtual; and as mass customization [1] becomes the new normal, the ability to properly configure a product requires more and more rules to ensure compatibility of design element options. And that is where CPQ hits its limits.

The Fallacy of CPQ

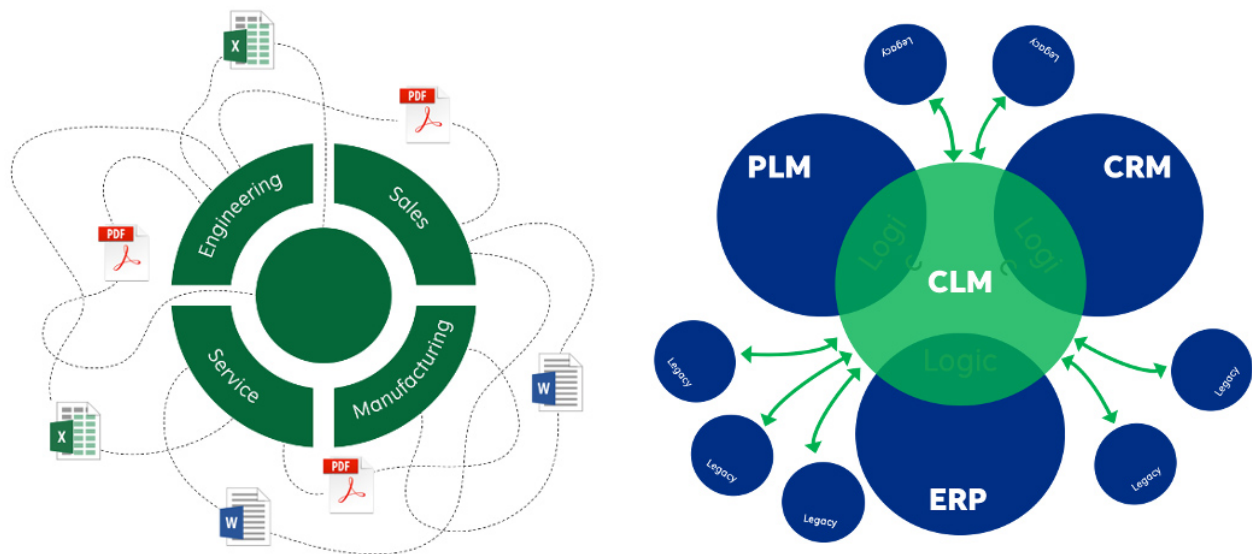
While it may be named a “configurator,” CPQ is not enough, both in terms of making the sale or in terms of driving revenue for the company. Why? Because as the example scenario shows, the ability of the sales team to create a viable configuration on the spot is highly dependent upon the building blocks that have been designed into the system, adequate training on using the tool, product knowledge, and current production capacity or schedule. In other words, simply dragging and dropping some pictures together, while it may look good, does not guarantee success. That is why many early CPQ workflows mandated review by Engineering before the order was processed.

In today’s world, that workflow is too slow. Turnaround times are constantly challenged and any way to trim a few days or weeks off the process means faster time to market and more revenue for the bottom line. Bypassing an Engineering review and moving straight into production does just that. That is, until the configuration that the sales team developed cannot be built or does not function. Then, instead of fast and efficient engineer-to-order workflow, the company is stuck in a loop of corrective action and production delays. Not exactly the promise that CPQ offers, nor the promise that the sales team provided the customer. And this does not include consideration for scheduling conflicts with other committed delivery dates.

Similarly, populating legacy configurators with increasingly complex sets of rules takes a long time of up-front processing and labor. To handle this part of the process, configurator engines have matured, moving from simple rules-based selections to constraint satisfaction engines to compile-based configurators [2]. Regardless of the method, human intervention is required to define what options and variants are compatible with each other. However, it gets increasingly difficult for a human to accurately define every possible viable combination as complexity increases.

In the meantime, sales automation has grown as well. The forward-facing tools have improved. CRM is now the forefront in managing client interactions and CPQ remains, for the most part, considered a back-office tool. Yet, as generations continue to expect more from the online buying process, the ability to customize their own product is demanded by consumers. But this trend is not limited to B2C shoppers, B2B customers expect the ease and convenience of online buying, too. These trends extend across industries, from computer buying to car buying to machine component buying. Out of consumer demand, CPQ has become part of the forward-facing buying experience and creating a lifecycle of its own.

Configuration Lifecycle Management



A Configuration Lifecycle Management system replaces the error prone manual exchange of information between functional groups in an organization with an automated integration of rules & dependencies with logging and traceability changes

CPQ on its own is no longer sufficient to meet the evolving customer requirements. The disparate systems in use by CRM, CPQ, PLM, and ERP each has its merits but does not provide a single source of truth. The sales team is using the latest data on their CRM system, but it may not have been updated to the latest engineering data resident within PLM. And CPQ often bypasses PLM and works with the sales tools that input directly into order processing, typically the domain of ERP. What is needed is a Configuration Lifecycle Management (CLM) system that integrates each of these systems and brings with it a single source of truth.

A properly implemented CLM system defines the rules from sales & marketing requirements matching (Market Intent) what is being offered to the customer, these rules are linked to Engineering, who own the data and understand the functional interactions of options and variants. (i.e. bill of material). The CLM system takes those rules and allows the sales teams to create custom configurations for the users that they know have been approved and are manufacturable. New configurations can be defined on the fly and be fully structured in PLM. The link from CLM to ERP

provides the sales team with a realistic expectation of when the system can be built, even if it has never been built before.

Changes made to one system are synchronized among all the business enterprise tools. As Engineering develops new products or new options, those rules propagate into CLM. A robust change and configuration management process ensures change impacts are defined, thereby identifying all data that needs to be updated including work orders and build sheets. As the sales team creates these new options, they can be confident that a robust configuration management process enabled by a tool like CLM guarantees a flawless sales order and a satisfied customer.

The New Story



Let us look at our story again, but this time with CLM.

The customer begins by visiting the supplier's website. The website received all valid combinations of the Products from the CLM configurator. From the website, the customer uses the web configurator to select the product and options they desire. This could involve "Guided Selling" leading the customer to special packages or campaigns. After submittal of the order, the integrated enterprise tools are validating the requested configuration with configurations that have already been defined. Upon validation the order is expedited to production, the ERP system is automatically updated with the order, and the ERP system provides a delivery date to the customer based on real time production schedules. On time delivery is assured. Customer satisfaction guaranteed.

If the CLM configurator allows for customization of the products, the website configurator will allow the customer to enter the data. This could be size of the product – for example, the length of a crane or size of a tent. After submittal of the order, the sales team or application engineers get notified of the order and follow up with the customer, clarifying that the selected options fit the customer’s intended application and use case. In the meantime, the integrated enterprise tools are comparing the requested configuration with configurations that have already been defined. If one exists, the order is expedited to production and the ERP provides a delivery date to the customer based on real time production schedules.

If a configuration has not been built before, Engineering is notified to verify and release the new product structure, creating any new datasets, as needed. The ERP system is automatically updated with the order and schedules the order for production providing feedback to the customer. On time delivery is assured. Customer satisfaction guaranteed.

While this utopian future seems very similar to the original story, it is only feasible because of robust processes and enabling tools. Without good change management and Configuration Lifecycle Management, none of this would be possible. The reality is, employees spend countless hours verifying and validating spreadsheets and manually updating data, often creating duplicate information because it is easier than searching for an existing configuration, only to find themselves suffering from cost overruns while they apologize to the customer for being late on delivery...again.

If you find yourself in that last situation, maybe it is time to see if Configuration Lifecycle Management is right for you.

For more information about CLM visit <https://configit.com/solutions/clm/>

References

- [1] M. E. Dollarihide, "Mass Customization," Investopedia, 04 SEP 2020. [Online]. Available: <https://www.investopedia.com/terms/m/masscustomization.asp>. [Accessed 07 SEP 2020].
- [2] Wikipedia, "Configure, Price and Quote," 01 APR 2020. [Online]. Available: https://en.wikipedia.org/wiki/Configure,_price_and_quote. [Accessed 04 SEP 2020].