Configuration Lifecycle Management

The development and use of product configurators has been highlighted as one of the more important IT enabled technologies to facilitate the realisation of mass customisation. However, the integration of these technologies into day to day business operations is complicated by the proliferation of configuration capabilities across a number of types of IT applications. This paper seeks to address the challenges of product configuration in an assemble-to-order environment via an automotive case study.

The purpose of the paper is to highlight a number of configuration gaps and propose an approach to enable an integrated approach for the consolidation of configuration across a number of business domains in the context of an assembly-to-order environment. We call this new area of enterprise software Configuration Lifecycle Management (CLM).
## Contents

INTRODUCTION ................................................................................................................................. 2  
CHALLENGES OF MASS CUSTOMIZATION .................................................................................. 2  
THE CLM VISION ............................................................................................................................. 5  
A CLM CASE STORY: JAGUAR LAND ROVER ............................................................................. 7  
CONCLUSIONS ................................................................................................................................ 11
INTRODUCTION

There is well established literature around mass customization and the application of configuration technology as a means of connecting customers to personalized products and services. Mass customization tries to achieve, at the same time, the cost efficiency of mass production from a manufacturing point of view and the desire to meet customer demands with flexible and customizable product offerings. For many goods, in particular high-valued products such as automobiles, the trend over the last three decades has been to offer ever more choice in the offerings to customers. Evidence for the trend to move towards mass customization can be seen from the fact that the Configure, Price, Quote (CPQ) segment is growing more rapidly than the underlying related areas of PLM, ERP, CRM (CPQ annually 25% vs. 8-10%)

A manufacturing company can be classified along two dimensions: the complexity of their products measured as the number of different variants and the volume of products, illustrated in the diagram below:

![Diagram showing complexity and volume classification]

Some examples of companies with low volume production and little variation (the lower left corner) are Fritz Hansen (manufacturer of exclusive handmade designer furniture) and Royal Copenhagen (manufacturer of hand painted porcelain). Other companies make highly complex products that are engineered specifically to customer requirements (lower rightmost corner). Examples of such companies are SMS Siemag who makes steel and aluminum process plants and Boeing who makes commercial airplanes.

In the top leftmost corner are those companies that produce standard products in high volumes. There are many examples of such mass produced products, e.g., pencils and iPhones, since traditionally manufacturing companies have focused on reducing cost by going from manual low volume production towards batch- and flow-based mass production and by doing so, increasing product volumes and reducing costs. This effort moves the companies from the bottom towards the top in the above diagram.

In the top rightmost corner are those companies that mass produce highly customized products, referred to as mass customization. At its core, mass customization is a significant increase in variety and customization without a corresponding increase in costs. Examples of companies that have embraced mass customization are Grundfos who produce highly customized pumps and most car manufacturers such as Audi and Land Rover.

When moving towards mass customization, almost every business process from development, manufacturing through to the delivery of such configurable products needs to be adapted to ensure both accuracy and efficiency. However, current enterprise software such as PLM, ERP, and CRM systems are developed for standard products with configurable products added as an afterthought and in a siloed and disconnected fashion. To achieve the cost efficiency of mass production, it is essential that the master system of record for product definition logic and configuration is maintained in a single enterprise system, called a Configuration Lifecycle Management (CLM) system. This paper defines the challenges of moving towards mass customization and presents the key properties of a CLM system.

CHALLENGES OF MASS CUSTOMIZATION
Increasingly, as products become more complex a key enabler for achieving the tailoring of products to customer needs is the application of computer aided configuration technologies. Such technologies enable the efficient and accurate definition of the allowable configurations thereby supporting the demand for customization and personalization.

Configurators can be segmented into three broad categories:

- **Sales Configurators** – used to define compatibility rules to enable pricing and the creation of quotes and orders.
- **Product Configurators** – used to define compatibility rules and configure the product (typically a bill of material, CAD model, and drawings).
- **Manufacturing Configurators** – used to define compatibility rules to drive manufacturing processes, scheduling, and resource allocation.

Two key IT technologies, Enterprise Resource Planning (ERP) and Product Lifecycle Management (PLM), have emerged to drive business efficiencies within product development and manufacturing over the last 20 years. Each technology brings unique value to the enterprise, and when combined, ERP and PLM provide the ability to allow direct sharing of engineering and manufacturing data through automated processes. Understanding the core functionality underpinning both ERP and PLM systems is increasingly important as the boundaries between these applications begin to blur as the vendors of these systems merge functionality at key touch points. For example, all major ERP applications such as SAP, Baan and J.D. Edwards have a configuration capability in their standard applications; similarly both Dassault’s Enovia PLM application and Siemens PLM solution include configuration components. But even so, product configuration applications continue to be implemented via a myriad of technical solutions via commercial sales configurators or as bespoke company specific solutions. The proliferation of configuration capabilities means that configuration data is increasingly likely to be distributed across multiple applications throughout the enterprise. The figure below illustrates how configuration data is maintained in multiple enterprise systems (A, B, C …) and communicated in various formats such as Excel and Word when products move through their lifecycle from being developed, over to priced and sold, to manufactured and eventually maintained.

Whilst technologies to address the problem of configuration continue to evolve, the integration of these technologies into day to day business operations is complicated by the fact that product configuration is not a process in itself. Product configuration is a step within a number of other business processes such as the product design process, the sales process and the order fulfillment process. As a consequence of this configuration capabilities can be realized via a range of specialist configuration applications, often focused upon a particular phase of the configuration lifecycle, or configuration capabilities can be embedded into enterprise focused applications.

The challenge facing companies would appear to be the navigation of the myriad of applications and the avoidance of unnecessary duplication of configuration activities. The diagram below illustrates how PLM defines engineering rules for variants and options, CRM defines sales rules for variants and options and ERP defines manufacturing rules for variants and options. Thus, there is no single coherent representation of the valid variants.
The reality for many companies is that the integration of these different types of configurators is problematic since integration across different configuration domains is difficult due to the differences in the configuration paradigms employed by the different configuration technologies to solve particular types of configuration problem. Some typical barriers to the deployment of configuration technologies are:

- **A lack of technology**: the difficulty of finding a solution that meets the needs of a company.

- **A lack of resources**: The time and maintenance associated with the creation of product models and their rules across the engineering and marketing domains.

- **The cost and complexity of integration**: The complexity associated with managing and synchronizing configuration master data across consuming applications such as CRM, ERP, CAD and PLM.

The integration landscape is further complicated where companies have developed their own in-house configuration technologies to meet their specialist configuration requirements. A study conducted by AMR noted that "*in-house developed applications are still used by 33% of companies for some or all of the configuration process.*" To illustrate this point, in 2011 Siemens conducted an internal review of the use of configurators across their business units. In total 96 different configurators were identified, 67 of which were developed in-house by different business units of Siemens. Of the remaining applications bought off of the shelf, 27 different vendors were identified as supplying configuration technologies to Siemens.

The siloed nature of the enterprise systems also means that propagation of changes is difficult and error-prone. Any change in the configuration rules will have to be replicated to all systems. Any change to the engineering rules must potentially be reflected in the sales configurator. For example, there may be a new legal rule that states that all vehicles in the US must have a reverse sensor on a particular date. Such a rule is a typical example of an engineering rule which may affect the behavior of the sales configurator. But deciding whether it influences the sellable combinations and if so, in what way, is a very difficult problem when the two configurators do not share a common set of rules.

Determining the correct price of a configurable products is another significant challenge. The price calculation of a configurable product is very complex due to the many ways a product can be configured. The price logic must be coded with a number of rules to support, for example, that certain options can be selected as stand-alone options at one price but may also be included in packs of options, in which case the price is included in the price of the pack. Such pricing rules are often coded up in the ERP system (since it is responsible for invoicing), but must be replicated to the sales configurator to provide accurate pricing when a customer places an order.

These issues are fundamentally caused by having different parts of the configuration logic in different enterprise systems. A CLM system supports the enterprise over the complete product lifecycle by providing a single version of configuration truth which includes all aspects of the product, including engineering, sales, manufacturing and maintenance of the product.
THE CLM VISION

A Configuration Lifecycle Management (CLM) system is the master system of record for configuration logic and the individual configurations. The figure below illustrates how a CLM system is integrating traditional enterprise systems such as ERP, CRM and PLM.

All configuration-related business processes access CLM for configuration definitions and queries. At the simplest level, product configuration is an interactive process in the selling of a product whereby a customer chooses a value or an option, which is then validated via a configuration engine before allowing the customer to make the next choice about the remaining variables to be configured. This process of choice followed by validation continues until all choices have been made and a complete and valid configuration has been found. The apparent simplicity of this process hides a significant amount of complexity to construct the rules used to drive the configuration process and it does not reveal all the processes involved in running a business with customizable products.

Product configuration is the process through which product features (which represent both technical aspects and customer choices) are modeled in a configuration application. A product model is developed which consists of a set of parameters called feature families, a set of possible values for the parameters called features, and rules describing dependencies among the features.

A product model consists of rules to control the technical parameters of the product as well as rules that control the commercial presentation of the product. A product model is then used by a configuration engine to assist in generating a configured product. Product configuration management is, therefore, the process through which a full or partial product configuration is solved via a configuration engine to generate an analytical, a digital or a physical representation of a product.

What variants of the products that is valid for a specific customer depends upon a range of parameters including the customer’s market and the local regulations and conformance conditions in that market. The exact time for the delivery of the product will also influence what options are available, as will the overall business decisions made from a profit perspective on what will be offered to what customers on a market by market basis.

The key idea in a CLM solution is that there is only a single version of configuration truth, i.e., the CLM system is the master of record. Thus, the product model contains rules to support all phases of configuration, from design, to ordering and manufacturing, to maintenance. We briefly describe how product configuration is an essential part of the seven key phases:

- **Develop.** During the development of new products, decisions have to be made on how to modularize the product so that it will be amenable to customization. The allowable variants and options must be defined so that the necessary dependent engineering, sourcing processes and other business processes can be initiated. In this phase, the allowable combinations forming the configuration space are being defined. This involves defining legal and technical constraints representing, for instance, as a concrete example the permitted radio frequencies for wireless door locking in specific markets. It is important to understand the consequences for
the downstream processes and profitability of the decisions taken on the offered options. This involves analyzing the space of allowed configurations.

- **Price.** Pricing is a key activity for any business. It involves analyzing the cost prices and the market acceptable sales prices and from these deducing the expected profit margins. With customizable products this is grossly complicated because the calculations have to be made over the full space of possible configurations and not just on a list of predetermined products.

- **Market.** When marketing a customizable product on a global scale there will be differences in the offering across the geographical markets. The differences arise as a consequence of local regulations and varying customer preferences. It is necessary to ensure that the offered configurations match the needs. Sales and Marketing departments often express the mapping of offerings to markets using large availability matrices which, in essence, are configuration rules.

- **Sell.** The sales process involves the key operation of deciding on a specific configuration meeting the customer requirements. A customer will be faced with a lot of choices and it is important to ensure accurate guidance when making the selections so that the end result will be a satisfactory product at an acceptable price. The result of the sales process is a valid order to be fed into the downstream systems.

- **Source.** Manufacturing of complex products involves using parts and assemblies from sub-suppliers. When the overall product is configurable the parts and assemblies vary according to the individual configurations. It is, therefore, important to ensure that the sourcing of parts is such that correctly configured sub-assemblies are delivered in the right sequence for the production line. Configurations must, therefore, be communicated correctly to the suppliers. Other constraints to be considered are the suppliers’ yearly or monthly capacity that must be taken into account in the production planning.

- **Build.** The building of the product is defined by the configuration. Each product will have its own defining configuration that is used for selecting the correct parts to be mounted. An error in a configuration is very costly if it makes its way down to the factory floor. The result can be that the end product will be useless or that the production line must stop for a period of time while the problem is overcome. Other constraints to be considered are build constraints on the production line reflecting known line labour balancing issues and lineside feeding constraints.

- **Maintain.** After a product is delivered it will typically go through service and maintenance where these activities are also dependent upon the configuration of the product. The configuration of the product can also be changed over time by upgrading sub-systems.

The engineering intent of a product sets the parameters of the features available to marketing and sales by which to configure a marketed product. All legal, quasi-legal, technical, engineering and licensing constraints are encoded in the engineering intent so that Marketing and Sales cannot overwrite these product rules and Manufacturing cannot receive orders for infeasible or unbuildable feature combinations. Examples of engineering rules include all features subject to third party licensing arrangements, e.g. Bluetooth and legally required features.

Marketing intent sets the commercial parameters which relate to the way in which the product is to be marketed, taking into account factors such as geographical markets, trim levels etc. Marketed intent is a subset of the engineering intent and working within the defined engineering capability of a product, Marketing can then configure and group features in the most appropriate manner for the market, region, trim level etc.

A further complication arises from the need to have control over the *effectivity* of key elements of the product definition. For example, in the automotive industry, product offerings are always under change: major changes are grouped into model years of a car line, which typically involves changes to all major sub-systems of the vehicles whereas minor changes can occur at any time and can be due to upgrading of some of the sub-systems for various commercial reasons. All changes need to be reflected in all the key processes involved in the business. In order to get alignment between all these processes, the exact timing of the effect of a change is of key importance. Any system that is to support the business must, therefore, have full control over the effectivity points and their interactions.
The following figure illustrates how the CLM system interfaces to all phases in the product lifecycle. It also shows how the configuration rules and resulting solution space, i.e., the set of all valid configurations, is central in the early phases of the life-cycle (“Develop”, “Price”, “Market” and “Sell”) whereas the actual configuration, i.e., the specific features specifying each ordered product, is the central object in the following phases.

A CLM CASE STORY: JAGUAR LAND ROVER

Jaguar Land Rover (JLR), owned by Tata Motors since June 2008, is a premium vehicle manufacturer selling approximately 300,000 vehicles per year; in 150 plus markets around the world. Jaguar produces high performance sports cars and saloons (XK, XF, XJ and F-type), whilst Land Rover produces class leading 4-wheel drive vehicles (Defender, Discovery 4, Freelander 2, Range Rover Evoque, Range Rover Sport and Range Rover). Research and design activities are undertaken in the UK at two development sites whilst manufacturing is undertaken at 3 UK based plants with overseas assembly occurring in India and a planned joint venture in China is underway. Whilst the impact of the recession, which started in 2008 was severe, JLR’s recovery, like the rest of the premium automotive sector, has been driven by strong demand in the developing BRIC markets.

JLR has concurrently embarked upon the introduction of both a new PLM application and a new ERP application. The concurrent introduction of such enterprise IT applications is highly unusual, however, this has provided a unique backdrop within which to examine the opportunities for the rationalization and optimization of product configuration capabilities to support the mass customization of products throughout the product lifecycle. Specifically this case study identifies those factors which enable the adoption of a CLM approach to support the enterprise over the complete product lifecycle using a single version of configuration truth.

Jaguar Land Rover’s varied ownership heritage has resulted in a highly challenging IT landscape. Following ownership by a number of companies including BMW and then Ford, JLR has inherited a large portfolio of applications as a consequence of various IT integration activities. From a product definition and configuration perspective information is authored and duplicated across many systems leading to wasted effort and the increased risk of inaccurate information being published to drive configuration activities.

JLR has a 20 year history in the use and development of configuration management capabilities. Both of JLR’s existing product definition authoring tools originated in the early to mid-1990s and are built upon an object oriented approach. Both use Boolean constructs meaning that they can handle "If-Then" rules and they both have a constraints capability in that they can handle tabular or multidimensional relationships which either must hold or are forbidden. One system was inherited from Ford and is a commercial application that was popular in the late 1990s and early 2000s and this application is used to author the engineered intent.

The second application was developed in the early 1990s by Land Rover (then a part of Rover Group). Three IT applications were concurrently developed in-house: a product definition application, a bill of material system and an order scheduling system. Each system was designed to operate with one another in support of a drive to improve efficiencies to enable the company to operate as a configure-to-order manufacturer. Whilst the bill of material system was replaced several years ago under the ownership of Ford, the product definition application used to define the marketed intent of JLR’s vehicles and the order scheduling system remains in the landscape today.
Having an in-depth knowledge of configuration technology and the experience of using this capability in an integrated IT environment caused JLR to ask the question “could it be possible to introduce a single rule authoring and configuration management application to serve the needs of the enterprise?” The ongoing evaluation of PLM applications in 2009, combined with the evaluation of a multi-phased deployment of an ERP application presented an opportunity to undertake a significant technical evaluation of the market for configuration applications. This technical evaluation finally resulted in the selection of Configit as the configuration development partner and Configit ACE as the CLM system. Presented and discussed here are some of the key conclusions of the technical evaluation of the market in a 20 month period between September 2009 and May 2011 comprising three distinct phases.

1. The desk top market evaluation exercise conducted from September 2009 to January 2010.
2. The Request for Information phase conducted from February 2010 to July 2010.

The initial desk top market evaluation exercise identified 35 specialist configuration applications, or best of breed companies, available on the market which satisfied a high level set of technical criteria. The 35 applications were identified via an extensive internet based search and review of published reviews and technical white papers on configuration applications/technologies by the likes of Gartner, Aberdeen Group and AMR. The websites of the vendors were used to identify the nature and scope of the functionality of the applications and reference was made to any application specifications as well as other documentation relating to the configuration technology underpinning the applications. Excluded from this list of specialist configuration applications were PLM and ERP applications which contained in-built configuration capabilities.

From this list of 35 companies, 12 were judged to cover the range of technical capabilities which JLR wanted to review in detail and these companies were invited to undertake a Request for Information (RFI) exercise. In addition to these specialist configuration companies, two market leading enterprise application companies, one providing a PLM solution and one an ERP solution, were also invited to participate. In total 9 companies, including the PLM and ERP companies, chose to participate in the RFI review. A detailed set of technical criteria was developed and used to evaluate the configuration capabilities of the companies participating in the study. In addition to this a detailed practical use case was developed and supplied to each company to model in their applications.

The use case demonstrated a range of configuration concepts to be modeled which are representative of the complexity of JLR’s vehicles. The table below summarizes the high level configuration concepts tested for. Upon completing the RFI phase three companies were invited to participate in the RFQ phase. During this next phase each company participated in a second series of in-depth technical evaluations and the use case was refined to demonstrate a range of more sophisticated modeling concepts, building upon the existing data.

<table>
<thead>
<tr>
<th>Configuration Concepts Tested</th>
<th>Configuration Principles</th>
</tr>
</thead>
</table>
| 1. Support for effectivity     | • Support for effectivity based configuration throughout the product lifecycle.  
                                  • Effectivity is a time point or set of points which define the availability period of an object e.g. a feature, a feature rule, a feature specification etc. in a product. |
| 2. Support for concurrent build periods | • Differentiation between production builds and prototype builds using effectivity.  
                                           • Releases managed around timing points and change points, with multiple programs of work being managed independently. |
| 3. Support of an integrated modeling environment | • Support for both perpetual and model year based releasing in the product modeling environment. |
| 4. Support for rule inheritance | • Creation of marketed intent will be defined within the bounds defined by the engineered intent. |
| 5. Support for the authoring of cross-carline rules | • Enable modular product modeling, for example, homologation rules and certain architectural technical constraints apply across multiple carlines. |
In their own right the PLM and the ERP applications contained a degree of inbuilt product modeling and configuration capability; however, two conclusions were drawn. Firstly, it was found that the breadth of the product modeling capabilities of these types of applications were restricted and less capable than the existing technologies employed within JLR and the specialist applications. The rule authoring capabilities of both applications were only at the level of truth-maintenance systems, meaning that complex modeling constructs would be difficult to support. It was also found that both applications adopted a strict hierarchical tree structure for the development of product models on a vehicle line by vehicle line basis with no ability to concurrently maintain multiple model years. Product model maintenance in both approaches would, therefore, be highly complex and restricted to a model year based approach to product model maintenance contrary to a configuration approach based upon the use of effectivity.

Secondly, the scope of the configuration capabilities of both applications was found to be restricted. The configuration functionality embedded within the ERP system is designed to support the front-end sales and order processes but is not suited to the back office requirements of Engineering. Equally, the configuration capabilities of the PLM application, due to its PDM origins, are focused towards CAD, prototyping, visualization, etc. as opposed to supporting the demands of front-end sales and order management. The configuration capabilities of these applications are unsuited to supporting the lifecycle configuration needs of complex products from concept to run-out via a single version of configuration truth.

Due to the complexity of JLR’s premium products, perpetual product models are used for modeling purposes and effectivity is used to create “virtual” model years for the purpose of releasing marketing intent to the sales systems. Information such as product and feature descriptions, price structure and feature visibility are associated at the model year variant level. A purely model year approach to product modeling is unsuited to a vehicle manufacturer with the complexity of products sold in the number of markets such as those of JLR’s. At any one time a vehicle line may have up to three model years to be managed concurrently; one model year in production and the subsequent models year for the next two model years in development. It should be noted that Ford North America does model on a model year basis, whilst Ford of Europe shares the same modeling philosophy as JLR due to its model and market complexity.

The concept of effective points is used to manage up-dates and control when changes are released into the production environment. The use of a perpetual view of a product in the modeling environment enables feature rules to flow through the model years and be inherited where required, therefore, meaning less maintenance or errors due to re-authoring. Whilst features are affected in and out as required over the life of a vehicle many features, mainly technical constraints, will remain unchanged and will cover multiple marketing model years. The best of breed configuration applications studied do not currently have a sophisticated approach towards effectivity, instead they rely upon release management processes as opposed to using a fully defined approach toward effectivity via the use of effective points associated to features, rules and build phases.

Of the 7 best of breed configuration applications reviewed, 6 used a constraints-based approach to rule definition whilst only Configit used a compilation-based approach. These applications were better able to express some of the more complex feature relationships demonstrated in the use case, and two applications were able to tackle the problem of writing cross-car line rules to be inherited across multiple vehicle lines. Due to the increased complexity of vehicles, the use of modular designs and the need to be able to define common technical constraints across vehicles/platforms means that there is a requirement to author common feature rules independent of the product ranges and then incorporate these common rules into specific product models. Such an approach we have referred to as being “modular” product modeling.

Through the adoption of modular modeling principles it is proposed that certain types of feature rules are authored as independent files which incorporate effectivity and are change managed in their own right. These independent modular files are then incorporated into the product line files that need to observe these rules. Such an approach, it is proposed, lessen the negative impact of product modeling practices that exhibit the characteristics of a tightly coupled design where feature inter-dependency is complex due to the fact that a complete vehicle line is modeled in its entirety and rules are inherited via the explicit use of structure. The concept of modular modeling appeared to present a challenge for those rules authoring applications that operated via explicit inheritance through the use of product structure, as opposed to those applications whose modeling approach is constructed around implicit inheritance. Product models, which are tightly coupled, tend to exhibit the following developmental characteristics, which are seen as disadvantages:
• A change in the rules applied to one feature/feature family can force a ripple effect of changes into other features/families and their corresponding rules.

• The compilation of a valid product model for a complete vehicle line will require more effort and/or time due to the increased inter-dependencies between features and their rules.

• A particular set of common feature rules cannot be reused because they are embedded into the complete product model.

A conclusion of the study is that no existing PLM or ERP system can supply the needed functionality for a lifecycle approach to product configuration. This can be explained from the cross cutting nature of configuration management across very many diverse business processes whereas PLM and ERP systems are focused only on a particular subset of these processes. PLM systems have the purpose of supporting product knowledge management in the development and maintenance of products whereas the main purpose of ERP systems is to support operational business requirements in areas such as finance, manufacturing, inventory and human resources. From a product configuration management perspective they differ in a range of aspects, which is summarized in Table 2.

Common for these is that not one of the systems supports these aspects well enough from a configuration management perspective leaving a gap to be filled by a configuration-centered system for configuration lifecycle management (CLM). When working in PLM systems the basic approach is project-based centered on a particular product to be developed, whereas in ERP systems the basic approach is transactional dealing with representing the business operations taking place in the extended enterprise. Similarly, PLM systems seek to support time to market for new products whereas the time cycle supported by ERP systems is directed towards supporting the time to customer from the receipt of an order to delivery of the product.

When it comes to parts management in bills-of-material (BOM) the focus of the systems differs: PLM is supporting the development and evaluation of the Engineering BOM whereas ERP is concerned with executing the manufacturing BOM. In the middle is the need of linking commercial and technical features in fully solved feature strings to be used in driving the planning and manufacturing. Performance metrics also differ a lot. In PLM the BOM explosion is typically triggered manually as part of an engineer’s work whereas in ERP the BOM explosion workload can easily go to 1000s per hour during planning and entering of orders. On top of these are the need to make configuration solves for guiding customer and resellers using interactive ordering tools, which can easily go to the need of 10,000s per hour. However, the most limiting of all aspects when it comes to managing configuration is the lack of a total view of the full configuration space as seen by PLM or ERP. Only partial knowledge about the technical or commercial rules is accessible in any one system but thorough analyses of the effects of the rules require a complete view of all rules and their interactions. The following table lists main aspects of PLM, ERP and CLM with Respect to Configuration Management.

<table>
<thead>
<tr>
<th>Purpose</th>
<th>PLM</th>
<th>CLM</th>
<th>ERP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approach</td>
<td>Project-based</td>
<td>Supports both project-based and transactional view with one set of configuration master data.</td>
<td>Transaction-based</td>
</tr>
<tr>
<td>Time cycles</td>
<td>Supports Time to Market</td>
<td>Supports both Time to Market and Time to Customer</td>
<td>Supports Time to Customer</td>
</tr>
<tr>
<td>BOM focus</td>
<td>Definition of an Engineering BOM.</td>
<td>Generation of solved feature strings.</td>
<td>Execution of a Manufacturing BOM.</td>
</tr>
<tr>
<td>Solve performance</td>
<td>BOM-explosion workload: 10’s per hour</td>
<td>Performs many types of solves also to support, among other things, BOM explosion at a rate of 10000s per hour</td>
<td>BOM-explosion workload: 1000s per hour</td>
</tr>
<tr>
<td>Types of constraints</td>
<td>Technical constraints</td>
<td>Technical and commercial</td>
<td>Commercial constraints</td>
</tr>
<tr>
<td>Change of product configuration offerings</td>
<td>By model year</td>
<td>Linking model years and running changes</td>
<td>Running</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>--------------</td>
<td>----------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Configuration space</td>
<td>Partially defined configuration space biased towards technical features</td>
<td>Fully defined configuration space linking technical and commercial features</td>
<td>Partially defined configuration space biased towards commercial features</td>
</tr>
</tbody>
</table>

**CONCLUSIONS**

Product configuration applications continue to be implemented via a myriad of technical solutions via commercial sales configurators, as parts of ERP systems and more recently PLM systems or as bespoke company specific solutions. The proliferation of configuration capabilities means that configuration data is increasingly likely to be distributed across multiple applications throughout the enterprise. In the context of assemble-to-order companies we conclude that consideration needs to be given to the notion of a master system of record for product definition logic and configuration. As a consequence we have argued for the need for a configuration-centered system for Configuration Lifecycle Management (CLM).

Configuration knowledge is consumed by many systems and users in the extended enterprise. A complication is that all systems and users do not have the same needs for their view on a configuration. For instance, for billing it is important to know the choice of features influencing the price whereas technical features driving parts for manufacturing are irrelevant. For a customer going to a web-site to explore the options a textual description of the features in understandable text in the local language is important whereas an engineer seeking knowledge about a product will most likely prefer a concise, easy to recognize description. Therefore, a successful CLM system must take the channel consuming the knowledge into account.

A CLM system adds business value by providing a centralized source of product definition with such features as:

- The ability to associate predicted market take rates to features and calculate part volumes compounded from those take rates.
- The ability to compare feature commonality or differences between products and product types.
- The support for architectural analysis and cross carline analysis to optimize commonality and complexity targets.
- The ability to analyses parts usage in BOMs based on what configurations are valid over time.