Enterprise information systems as a service: enterprise software from the cloud follows patterns from industry

J.C. ("Hans") Wortmann
Groningen University, The Netherlands
Hans Don
Vanenburg Software, The Netherlands
Jan Hasselman
Vanenburg Software, The Netherlands
Alex Wilbrink
Vanenburg Software, The Netherlands

Abstract

This paper draws an analogy between developments in enterprise software and in capital goods manufacturing industry. Many branches of manufacturing industry, especially automotive industry, have grown in maturity by moving from craftsmanship to mass production. These industries subsequently move from mass production towards mass customization, introducing lean practices. Finally, full maturity is reached by increased servitization.
This paper analyzes the developments of enterprise information systems in the same terms. The paper shows that the enterprise software follows a similar pattern as capital goods manufacturing industries, with a few interesting differences.
Enterprise software "from the cloud" is comparable to the move towards product-service systems (PSS) in other branches of industry. However, lean delivery of enterprise applications is still at the threshold of being practised. Lean delivery requires single versions of applications and delivery in multi-tenant mode. Combining lean delivery with large variety requires automated configuration of application systems components.

Keywords: Enterprise information systems, SaaS, ERP, Cloud computing, servitization

1. Introduction

Innovation in manufacturing companies has always gone hand-in-hand with investments in plant equipment and capital goods. However, over the last decades a substantial part of the investments has been spent in enterprise information systems (EIS), such ERP, CRM and SCM. This paper studies the question, if production and distribution EIS is comparable to the production and distribution of capital goods and other physical products.

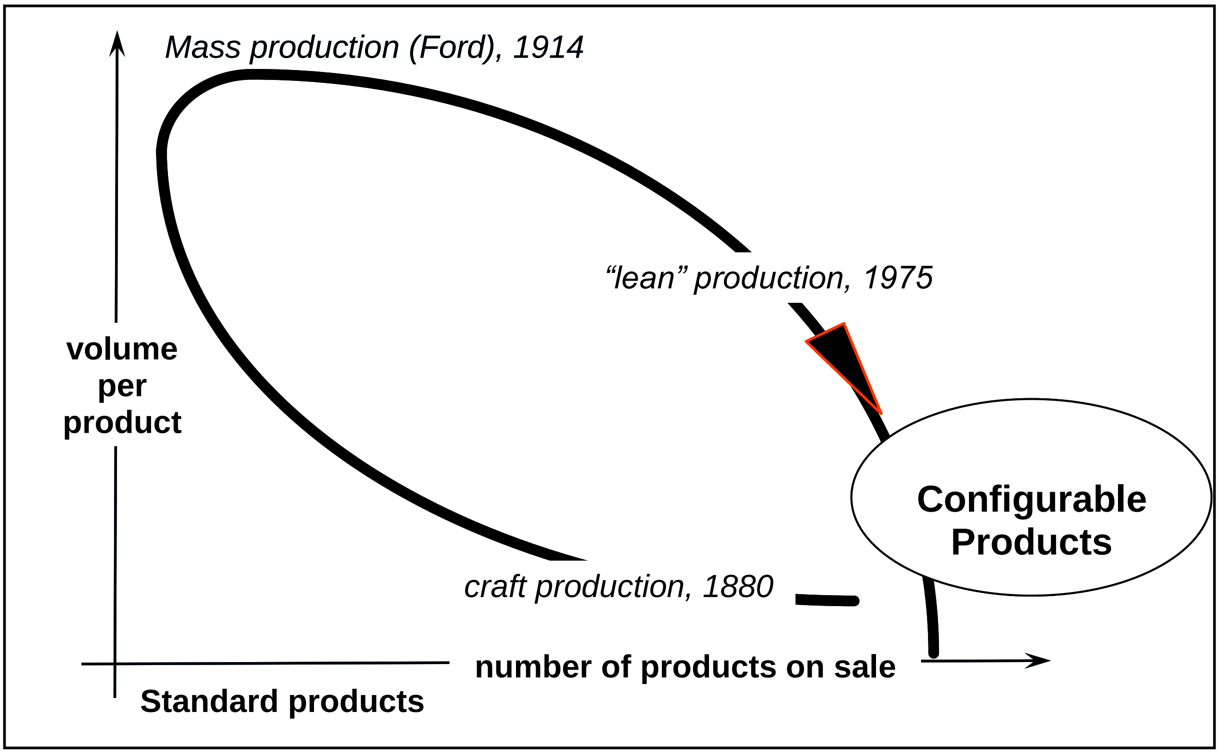
There is a striking analogy between the developments in many physical supply chains and the developments in the supply of enterprise software. In particular, the industry trends towards mass production, increasing variety, multi tier component supply, lean production and servitization can all be recognized in enterprise software supply networks. Cloud computing allows lean supply to be combined with servitization. However, the combination of large variety, lean supply, and servitization is not yet attained in enterprise software. Such a combination will require configurators of standard software components.
This paper provides a short overview of major trends in supply of physical goods in section 2. The analogy with enterprise software is discussed in section 3. Section 4 concludes the paper.

2. Manufacturing Industry Developments

2.1. From craft production to mass production

From the beginning of the industrial revolution till the start of the 20th century most production was “craft production” (or one-of-a-kind production). Craft production executes the process of manufacturing by hand with or with the aid of general tools, but not dedicated tools or machinery. A side effect of the craft manufacturing process is that the final product is unique. The product volume for each model was low, while the quality of the product could vary from low to extremely high quality.
Automotive industry provides an excellent example. The advent of mass production and the standardization of replacement parts guaranteed a parts' compatibility with a variety of vehicle models. Mass production has many drawbacks to craft production including that production quality can be lower than a craft produced item. However, the volume per product was high and the cost of the product was substantial lower and the buying community a lot larger.(See figure 1, derived from Womack et al. [1991])

Figure 1. Development of automotive industry

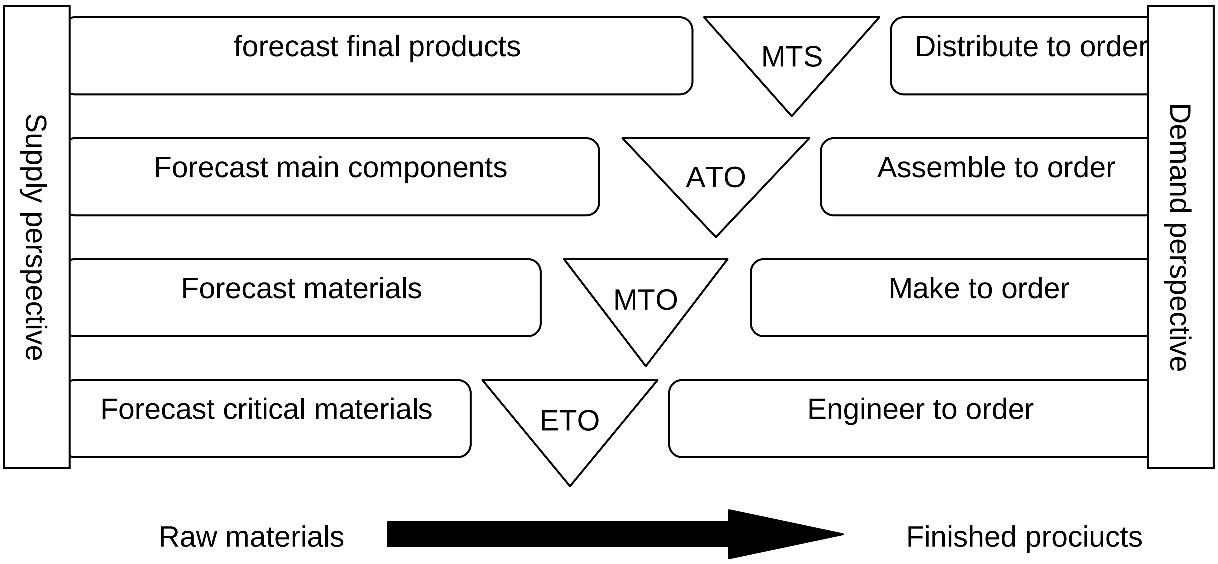


2.2. From mass production to mass customization (lean)

Lean manufacturing aims to bring back or exceed the quality of craft production and remedy the inefficiency of mass production through the elimination of waste. In order to align the variation of demand towards the supply ‘mass customization’ principles were applied.

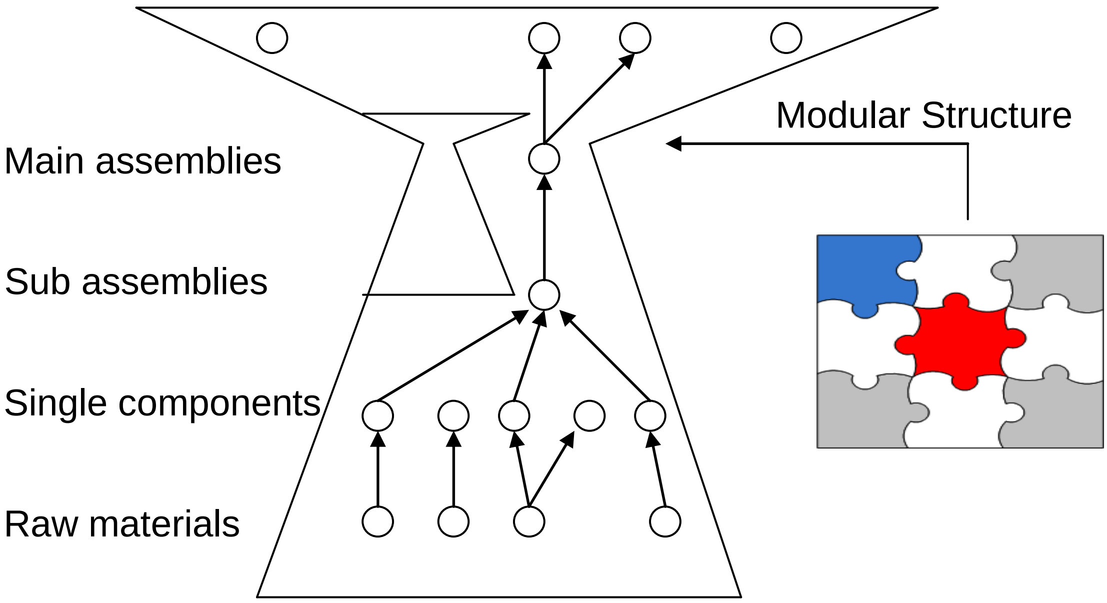
Most manufacturers realized though that craft production practices, as well as mass production and lean techniques were still needed to fulfil customer demand. Each of these practices had their own characteristics and logistical principles. Key distinction was the decoupling between demand and supply: the so called ‘customer order decoupling point - CODP’ (see figure 2, derived from Hoekstra and Romme [1992], and e.g. Ollhager [2010]).

Figure 2. Customer order decoupling point in classical manufacturing



In order to manage the variability of products, a new product modelling approach was introduced, allowing to represent product platforms or product families (see e.g. Forze and Salvador, 2002, and Wortmann and Alblas [2009]). Generic Bills of Material, generic process sheets (routings), generic pricelists were containing of all optional engineered products. Based on customer preferences the final product variants were *configured:* created, assembled using existing subassemblies or new produced components (see Figure 3).
As illustrated in the left part of figure 3, the Bill-of-Material structure for a platform takes the form of a diabolo (see Erens [1996]). The corresponding *generic* bill-of-material structure is more complex than the classical manufacturing Bill of Material, because it should allow representation of issues like:

Figure 3. Product families and platforms



- the parameters and constraints applicable to the platform,
- the various domains in which bills-of-materials are used,
- the versioning of the platform
- the modularity of the main components.

This last issue is illustrated by the right part of figure 3, showing standardization of interfaces. Finally, is should be noticed that the main components of a product family may again be a product family. For example, an engine or gearbox in a vehicle may be a generic product, just like a vehicle. This is illustrated in figure 3 (left) by the small diabolo inside the large diabolo. See Hegge [1992] and Alblas et al.[2010] for elaboration of the generic Bill-of-Material.

In should be noted that there are many configuration solutions which have more domain-specific knowledge than generic bills of material. For example in construction industry, a CAD system for piping solutions can be seen as a configurator, because it allows to create a huge variety of solutions from a limited number of standardized elements (such a pipes with varying lengths and diameters

2.3. From mass customization to multi-tier components

In order to fulfil the increasing market demand, manufacturers moved to pre-engineered products and services. Subassemblies and modules were standardized. This standardization made the production repeatable at lower costs, due to economies of scale. Tailoring was originally limited to final assembly processes and flexibility was limited to the options provided by the engineers of manufacturers. This results in short lead times: lead times were reduced since subassemblies were made on stock. Interfaces between the standard components were predefined and also standardized.

However, for many sub-assemblies the same principles apply, viz. that their markets enforce the CODP upstream. Accordingly, a world market appears for many products, with an increasing number of tiers and highly specialised manufacturers of materials, components, and subassemblies. The more mature the branch, the more the CODP moves upward. Accordingly, more sophisticated supply chain co-ordination is needed.
Moreover, the number of tiers in the supply network tends to increase, while the number of different suppliers per product tends to decrease. In other words, the manufacturing Bill of Material gets more levels and less components per level. This is also reflected in the Generic Bill of Material.

2.4 From Manufacturing towards Outsourcing and Servitization

In order to survive for manufacturing companies in developed economies, OEM manufacturing firms cannot comprise all value-adding manufacturing activities and at the same time they cannot restrict their activities to manufacturing.
On the one hand, the supply of components had to be outsourced to specialized firms who could apply economies of scale. Outsourcing of non-core activities up or down the supply chain created dynamic business networks to provide value in the most efficient way. Many of the components can be made at lower cost at outside the OEM.

On the other hand, OEM firms have to move beyond manufacturing and offer services and solutions, delivered through their products. Recent technological developments – especially in data capture and information processing – are enabling manufacturing firms to develop new business models, exploiting the potential of their products over the life cycles.

Servitization of capital goods was introduced a few decades ago. An example is the photo copying industry, who started to lease (rather than sell) their machines. In automotive industry, the fleet owners pushed servitization of passenger cars by requesting lease contracts with full service. Life cycle costing became popular in many markets where customers invest in capital goods, and vendors accordingly started to offer service contracts.
In academic literature, the trend towards servitization of physical products is also reflected. Of ten the term *product service system (PSS)* is used (see e.g. Mont [2002], Tukker and Tischner [2006], Aurich et al. [2009], and Shimomura and Hara [2010]). Key elements are:

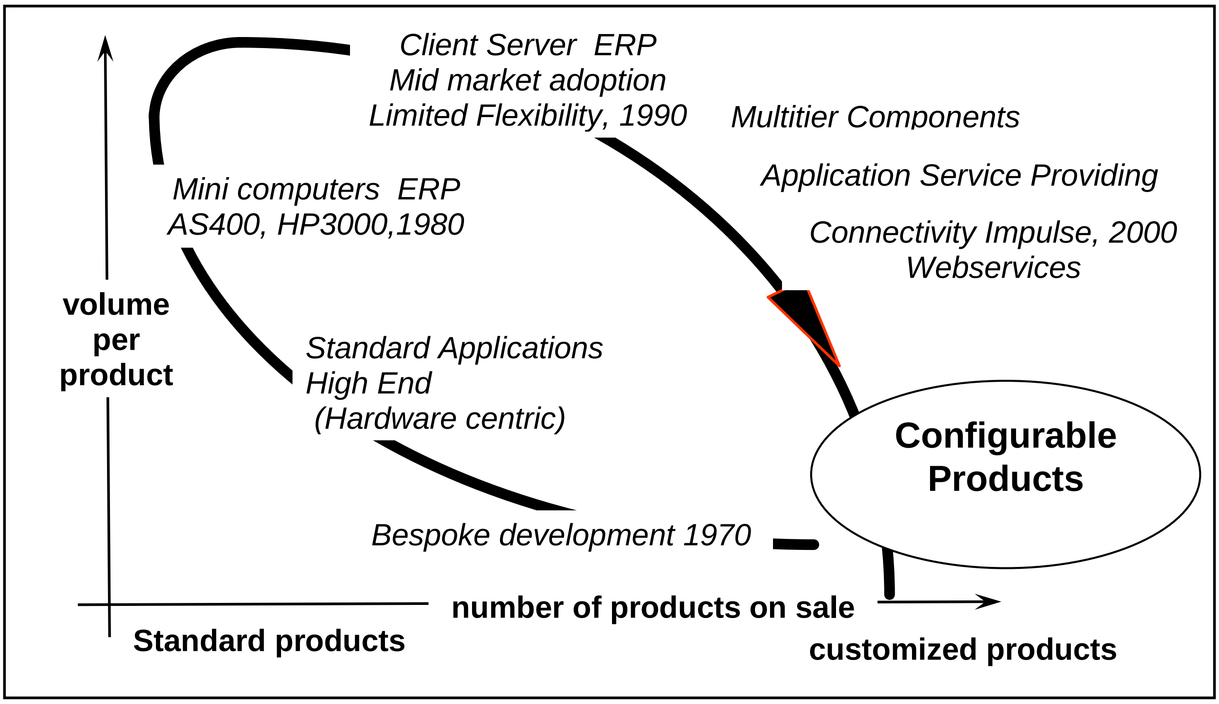
- a life cycle costing orientation with suppliers and customers
- focus on service level agreements
- ownership of the physical assets stays with the supplier (often an OEM)
- a network of service suppliers has to be organized for delivery.

3. Software Industry Developments

3.1 From bespoke software engineering to standard software

Several decades ago, in the 1970s and 1980s, application software was largely bespoke, and bespoke software engineering can be characterized as *craftsmanship*.
Over the years, *complete standard enterprise solutions* (Enterprise Resource Planning) were built, implemented and deployed. The shift from bespoke enterprise information systems to standard software which took place in the 1990-ties is completely in line with the move from craftsmanship to mass production in other branches of industry (see Figure 4).
The client/server based ERP solution was also adopted by the midmarket. The implementation costs were lower and competitive pressure –and therefore the need for automation- also increased for midmarket companies. This is again completely in line with the developments in other markets where the introduction of standard products leads to lower prices and therefore wide-spread adoption in lower market segments.

Figure 4. Development of enterprise applications market



3.2. From standard packages to multi-tier software components

Vendors of enterprise applications have always adopted mature standard software components, e.g. the database management system. However, other components, such as the license manager, the UI handler, the development toolkits (based on fourth generation languages), and the runtime environment were all developed and maintained by the ERP vendor.

The need for lower prices and more flexibility makes 'bespoke vendor specific development' replaced by solutions assembled from standard components. Large standard software component suppliers such as MicroSoft, Oracle and Google enter the market with software components and tools which are used by vendors of enterprise applications. This development is completely mirroring the development in physical goods manufacturing.

3.3. Application service providing (ASP) by hosting: a first step to servitization

The idea of hosting is not new. *Application Service Providing* (ASP) for long represented the idea of servitization of standard software products. It consists of the following elements:

- The ASP offers the hardware/communication infrastructure to the customer as a service. Therefore, customers do not own this infrastructure and have no operational responsibility. Later this has become known as *infrastructure-as-a-service (IaaS)*
- The ASP offers the systems software, server management, firewalls, database management system and other support software to the customer as a service. Again, the customer does not "own" this software: *platform-as-a-service (PaaS)*
- The ASP offers the application suite to the customer as a service. Therefore, the customer does not "own" the application but relies on the application service provider to make the application available: this offering is called *software-as-a-service (SaaS).*
- Payment by the customer to the ASP occurs on a periodic basis, based on actual use by the customer of the services provided by the ASP.

It is generally acknowledged that IaaS and PaaS may lead to substantial savings, due to economies of scale, due to virtualization and due to the law of Moore (continued increase of price-performance ration in ICT infrastructural assets). Therefore, pure ASP offerings have been in competition with more general outsourcing value propositions form ICT service providers to customers. Accordingly, the ASP market has grown over the last decade, but Application Service Providing per sé it is not a disruptive technology.

3.4. From standard software to limited configured packaged applications

Standard ERP provides limited flexibility. The flexibility of these applications is determined by parameters. However, parameters increase complexity and costs. The same holds for customization tools. Moroever, customization tools shift the decoupling point back from MTS to ETO and returns to craftsmanship. Altogether, the dominant delivery model of ERP is MTS. Moreover, the delivery model of standard software packages as ERP is not *lean.* The effort to implement these packages is dramatically high as compared to the mere software costs. After initial implementation, the dynamics of the customer cannot be easily captured to keep the system synchronized with the company progress. Last but not least, the dynamics of the vendor (new upgrades) cannot be properly managed.

These upgrading problems have motivated vendors of monolithic standard enterprise applications to strive for *componentization* of these monolithic applications. Many vendors announced that they split their ERP application to provide more flexibility for upgrading. Accordingly, vendors move towards an *assemble-to-order* mode of delivering configured enterprise applications.

There are three other forces which drive vendors towards this componentization and the accompanying standardizations of interfaces between their components. These are new technologies, integration issues, and business process management.

3.5 Servitization using single-code based software products

Maintaining multiple different versions of a software component implies multiple investments in knowledge, multiple integration problems, multiple functionality issues and technology issues to be solved. This continues to hamper efficiency of service delivery. Multiple versions can never become *lean* delivery. To solve these issues, a variety of customers should be served from a single code base: a service provider has only one version of running code of an application suite, from which all customers are served *software as-a-service* (SaaS).
Together, IaaS, PaaS and SaaS constitute *cloud computing.* The advantage for the service supplier is, that the single code base dramatically reduces the effort in knowledge management and integration. Companies such as Force (see www.Force.com) claim to offer such a solution. However, this form of software delivery is not easy. For example, the data of all customers have to be kept separated, which is called *multi-tennant* database management.

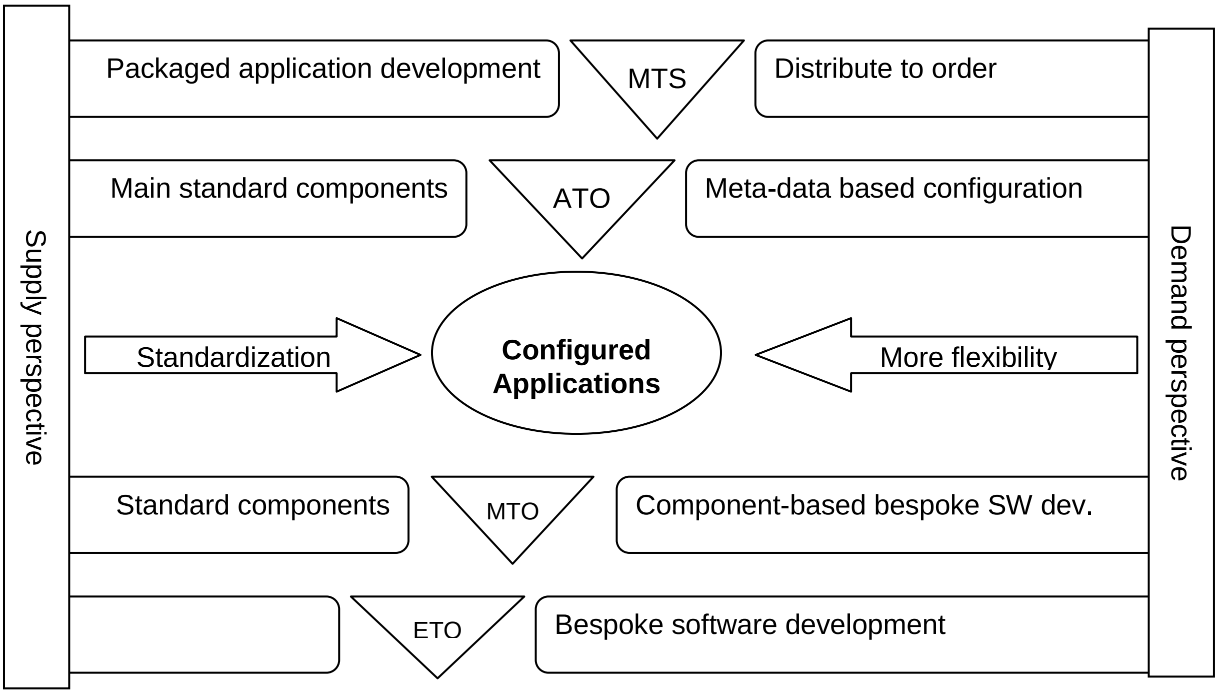
3.6 Servitization using single-code based *configurable* software products

The final step to a SaaS offering consists of *configurable* single-code based software components. In addition, a vast variety of vendors can offer other components for which there is a market, much like the *apps stores* in smart phones. Unlike these consumer apps, EIS components have considerable complexity due to at least the following three features:

- Components of EIS manage (structured) data
- Components of EIS collaborate via data integration or business process integration
- They use many devices, requiring adaptable user interfaces and interaction models.

An important question related to *what* is being configured. When thinking in analogies, then classical product configurators use *parameters* for configuration (see Section 2.2.).

Figure 5. Customer order decoupling point in enterprise information systems



If a configurator of enterprise information systems can work with a single code base for every component, it may lead to a very efficient delivery *(lean)* delivery model, which may cover a huge variety of requirements regarding functionality, deployment an ease of use. Such a development would complete the analogy of trends in physical goods and in enterprise information systems towards servitization, agility and variety.

4. Conclusion

In this paper, we explored the analogy between major trends in delivery of physical products as compared with the delivery of enterprise information systems. The similarities are striking.
Both in capital goods manufacturing and in enterprise information systems the earliest stage of maturity is characterized by craftsmanship. In capital goods, this is encountered as *engineer-to-order* manufacturing and delivery, whereas craftsmanship in enterprise information systems takes the form of *bespoke software development.*
The second stage of development is characterized by standard products. In enterprise information systems, it takes the form of *standard packages of application software*, such as ERP and CRM. Despite parameterization, these products remain standard products. Alternatively, when customization of these products is practised, the route back towards craftsmanship is paved.
Accordingly this paper conjectures that the next step in enterprise information systems is going to be an assemble-to-order delivery model based on configuration of standard main components. It will be a cloud service which has three distinct characteristics that differentiate it from traditional hosting. It is sold on demand, it is elastic -- a user can have as much or as little of a service as they want at any given time; and the service is fully managed by the provider (the consumer needs nothing but a personal computer and Internet access).

References

[1] Alblas, A.A., Zhang, L. and Wortmann, J.C. (2011) "Representing Function-Technology Platform based on Unified Modelling Language", *International Journal of Production Research,* accepted for publication.
[2] Aurich, J C, Wolf, N, Siener, M, Schweitzer, E (2009) "Configuration of product-service systems", *Journal of Manufacturing Technology Management*, **20** (5), pp. 591-605.
[3] Erens, F.J. (1996) "The synthesis of variety- developing product families". PhD dissertation, Eindhoven University of Technology.
[4] Forza, C. and Salvador, F. (2002) "Managing for variety in the order acquisition and fulfilment process: the contribution of product configuration systems". *International Journal of Production Economics*, 76(1), pp. 87-98.
[5] Hegge, H.M.H. (1992) "Intelligent product family descriptions for business applications". PhD dissertation, Eindhoven University of Technology.
[6] Hoekstra, Sj. and Romme, J.A.C. (1992) *Integral logistic structures,* McGraw-Hill, ISBN 0 7707552 8.
[7] Meijler, T.-D., Pettersen Nytun, J., Prinz A., and. Wortmann, J.C. (2010) "Supporting Fine-Grained Generative Model-Driven Evolution". *Journal of Software and Systems Modelling (SOSYM),* **9 (**3), pp. 403-424.
[8] Mont, O. (2002) "Clarifying the concept of product-service system", *Journal of Cleaner Production* **10** pp. 237-245.
[9] Olhager, J. (2010) "The role of the customer order decoupling point in production and supply chain management:. *Computers in Industry* **61**(9): pp. 863-868
[10] Shimomura, Y and Hara T (2010) "Method for supporting conflict resolution for efficient PSS Development", *CIRP Annals – Manufacturing Technology,* **59 (**1), pp 191-194
[11] Tukker, A. and Tischner, U. (2006) "Product-Services as a Research Field: Past, Present and Future. Reflections from a Decade of Research. *J. of Cleaner Prod.* **14** pp. 1552-1556.
[12] Womack, J.P., Jones, D.T., Roos, D. and Sammons Carpenter, D. (1991) *The machine that changed the world: the story of lean production*, HarperPerennial, ISBN 0 0609417 9.
[13] Wortmann, J.C. and Alblas, A.A. (2009) "Product platform life cycles: a multiple case study ", *International Journal of Technology Management*, **48** (2): pp. 188-201.