

System Software Product Line

Avionic Design Service GmbH
www.ads-avionics.com



1 Introduction

The concept of Software Product Lines has been developed for more than a decade. Being initially an academic topic, product lines are more and more incorporated into industrial processes nowadays.

This White Paper presents an integrated System and Software Product Line approach that covers the complete product development process. Instead of providing a green field solution, typical industrial organizational structures and decision processes are taken into account.

2 Product Line Approaches

Classic Software Product Lines require fundamental changes in engineering and management. They call for a complete transition from product orientation to feature based activities.

According to our experience, the engineering level is usually open-minded about product line approaches. As multi-variant issues are part of the daily business, the idea to focus on features and reuse rather than on complete products can easily be placed.

Management has typically a different view. By definition, project management is basically interested in a single project. Funding is usually done according to product variants. If a new customer orders an individual product variant, the respective budget may only be used for directly related development activities. Trying to reallocate those budgets on a cross product level is one of the most prominent pitfalls in product line application. In fact, a much more promising way is to accept product related budgeting as a boundary condition.

Another problem for product line concepts affects the transition process: Many concepts mainly describe how product lines work and which processes are applicable in which areas. However, in reality, product lines are rarely green-field developments. Development has usually started long time ago and a number of product variants have already been sold. Companies think about a transition to a product line when being faced by the needs of a growing market. Knowing how ideal product lines work is fine – it is however much more important to know how existing processes and already developed items can be transferred efficiently.

The product line concept described in this paper takes into account the needs of both project management and engineering. It empowers engineers to think in variations and commonalities and to develop product features rather than following deprecated clone approaches. At the same point of time it brings project management back into the driver's seat by providing a powerful control mechanism for the input to engineering as well as for the output, i. e. the product variants contracted by customers.

Providing a combined product and feature oriented approach

closes the gap between classic product lines and industrial business structures. This integrated approach enables to achieve a controlled development process, to offer a wide variety of product variants and to realize maximum economies of scale from internal commonalities.

3 Triggers for Product Lines

Product development usually starts with a single variant. This is especially true for complex systems where engineering activities are mainly driven by the needs of the first customer. In those early development phases, general technical questions and decisions demand the attention of developers and managers.

Although the possibility to sell several product variants to a number of possible customers might already be seen, there are in most cases no special activities to simplify future development of product variants. This is mainly due to risk avoidance for existing projects and due to the unclear business case.

The arguments remain the same for subsequent variants. Each one is developed in a separate project. This purely project driven development approach leads to duplicated efforts because similar tasks need to be done for each product. As items have been developed to fit to a specific product, there is a relatively high adaptation effort for other products. It is usually not simply possible to remove items or to add further ones. Each modification leads to further integration activities in order to achieve a consistent product.

Focusing on single product development might lead to repeated cost overruns and missed deadlines. There is probably a disappointing cost-benefit ratio. Quality issues, inconsistencies and incomplete traceability of changes possibly limit the ability to control the development process.

This is usually the trigger to put the development strategy into question. An obvious goal is to avoid redundant activities and to profit from product commonalities. The idea to overcome the described situation is to perform a variant independent investment, accepting an additional one-time effort, and to profit from reduced efforts for each subsequent product variant.

4 Product Line Engineering

Product line engineering is the sum of activities that are necessary to set up and operate a product line. It is key to separate product and domain engineering.

Product Engineering is basically driven by customer requirements. An important goal is to manage those requirements in an efficient way across all variants. The set of requirements that specifies a certain product variant is the input to the engineering process. It is also used to validate the final products.

Domain Engineering focuses on specification and implementation activities that satisfy the customer requirements. Specification encompasses the system architecture, system and software requirements and related tests. Implementation means software architecture, design, coding and test.

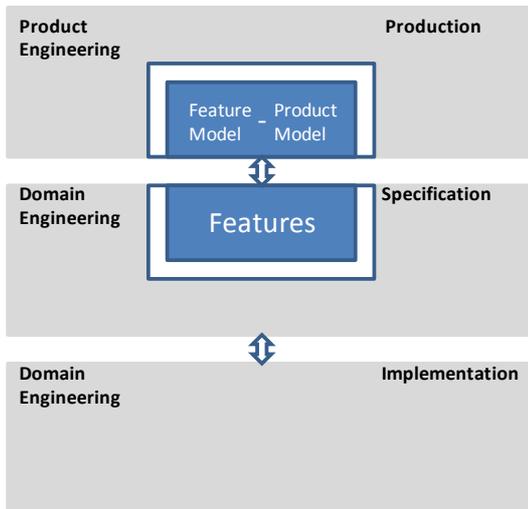


Figure 1: Separation of Product and Domain Engineering

4.1 Product Engineering

Efficient product engineering is a comprehensive development approach that covers all product variants at once instead of repeatedly focusing on single products. Product commonalities and variations are analyzed. The smallest building blocks for products are features. All features are defined and products are built by integrating those predefined features.

Product engineering starts with a definition of existing and planned features. In the next step a Feature Model is defined and products are defined by included features in a Product Model. All items like source code or documentation are organized according to features in a so-called Asset Base. Finally there is a production plan for product assembly from existing features and items respectively.

The main driver is effort reduction: Product commonalities are reflected in reusable components which are developed and maintained only once. Standardized interfaces allow easy integration or removal of items.

Furthermore, product development becomes more predictable. It is immediately transparent, which features can be taken over from the Asset Base and which ones still need to be developed.

The dynamic SSPL model in Figure 2 shows the development and integration process within a product line.

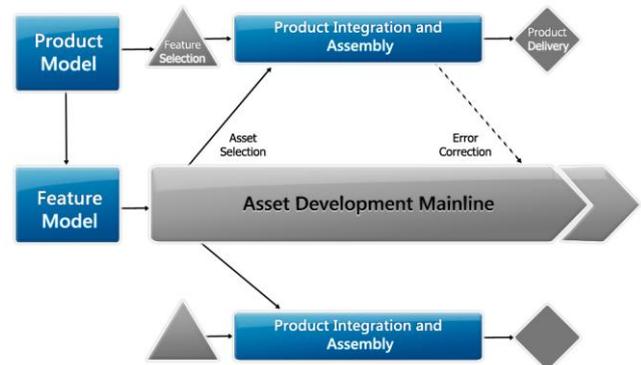


Figure 2: SSPL Dynamic Model

4.2 Domain Engineering

One of the goals of domain engineering is to develop items in such a way that they can be used by any product without specific adaptations. This requires that items are designed to be reusable. Product line oriented domain engineering thus leads to strategic reuse.

The Asset Base contains all items in all revision numbers. Each item may have several development lines. The most important one is the mainline that contains all approved states of a certain item.

Product integration is based on defined revision numbers for each required item. During the integration process items may be modified if required. This is done in an integration line. Product integration ends with product delivery. Item modifications, e. g. because of error corrections or continuous development, are merged back to the corresponding mainline.

Items that have been integrated to a certain product become part of a release baseline. Items that together implement a certain feature in a certain development state represent a feature baseline.

4.3 SSPL Process Model

Product and domain engineering are closely related. The process model in figure 3 shows the individual activities and their relations to each other.

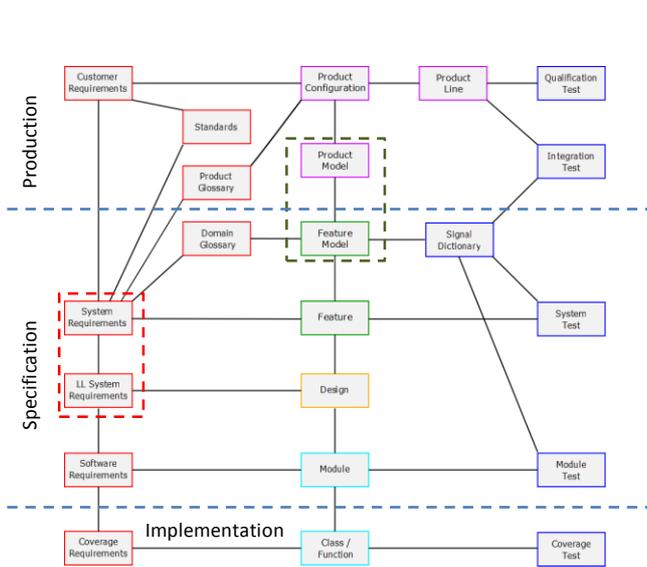


Figure 3: SSPL Data Model

This process model provides a sufficient level of abstraction to be applicable to a wide range of projects. Existing tools and process definitions can usually be mapped to activities of the data model. By customization and tailoring, the SSPL process model gradually transforms into a SSPL project model for a specific project.

5 The System and Software Product Line

A System and Software Product Line (SSPL) focuses on strategic reuse. Instead of taking over common items and then adapting them to fit the needs of a specific product variant, all kinds of variations are already implemented in the features.

The SSPL covers the complete system life-cycle from initial conception to maintenance and support. The SSPL is driven by Project Management. In order to be successful, a close cooperation between Systems Engineering, Software Engineering and Test Engineering is necessary.

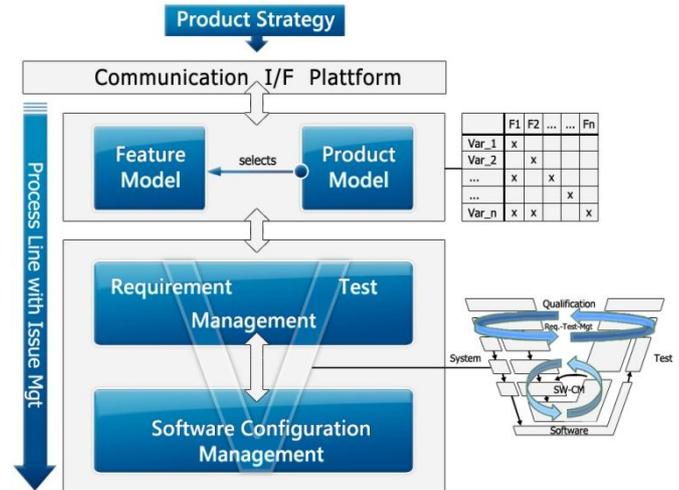


Figure 4: SSPL Static Model

The static SSPL model in Figure 4 shows the overall SSPL concept. The product strategy is defined in a dedicated communication platform that can be realized within a business intelligence system. This is the driver’s seat for project or program management to control the complete process. Product strategy decisions are taken on this level. The output of those decisions is the Feature Model on one side and the Product Model on the other side.

The Feature Model is the input for development activities that formally follow a V-Model: From system requirements to software requirements and finally to software engineering. There is no product dependency within the engineering activities: all items are related to features.

The Product Model is applied when products shall be integrated. This follows a dedicated production plan that describes how to assemble items for all features that have been defined for the product. Qualification tests are done again on this product level and the final product becomes visible on the communication platform level.

The V-Model within the domain engineering outlines the strict traceability of different levels of requirements and features. Of course, any kind of development process such as rapid prototyping can be applied.

Feature Model and Product Model can be regarded as a link between the management level that is driven by product variants and the engineering level that rather concentrates on product functionalities.

6 Summary

This White Paper presents an integrated System and Software Product Line –SSPL– approach. It takes into account typical industrial organizational structures and covers both engineering and management. The main focus is on companies that have been successful in developing a number of product variants and that are now facing the needs of a growing market.

The feature oriented approach in the engineering area allows a transition to strategic reuse. It helps to realize maximum economies of scale from product commonalities. Engineers are empowered to think in product commonalities and variations rather than following deprecated clone approaches.

At the same point of time it brings project management back into the driver's seat by providing a powerful control mechanism for the input to engineering as well as for the output, i. e. the product variants contracted by customers.

The SSPL is a combined product and feature oriented approach that closes the gap between classic product lines and business structures.