



# Podcast summary: Software as a product

## Best Practice Insight: Software Delivery InFocus Series

**Bola Rotibi**

July 2009

This is an accompanying document to the MWD Software Delivery InFocus podcast on “Feature-based product line delivery: software as a product”. It summarises the main guidelines and key messages highlighted in the discussion. To access the podcast audio please visit <http://mwd.libsyn.com/>

**MWD Advisors** is a specialist IT advisory firm which focuses exclusively on issues concerning **IT-business alignment**. We use our significant industry experience, acknowledged expertise, and a flexible approach to advise businesses on IT architecture, integration, management, organisation and culture.

[www.mwdadvisors.com](http://www.mwdadvisors.com)



Licensed under a Creative Commons License: [www.mwdadvisors.com/about/open.php](http://www.mwdadvisors.com/about/open.php)

SOME RIGHTS RESERVED

## Overview

With the increasing use of, and the important reliance on software components within manufactured products, and the sizeable market for commercial software applications, the concerns of the software delivery and management lifecycle are converging with the concerns of the delivery and management lifecycle of software-based products. The challenges facing organisations charged with delivering variations of software applications and software-based products are ones of consistency, repeatability and confident product re-use.

Software product lines, or SPL, is a mechanism and a methodology approach for managing the feature variation and evolution of software products and applications. It is seen by many of those responsible for delivering software-based applications and products as a contender for the most appropriate framework for bringing together and managing the convergence between the software delivery and product lifecycle processes, successfully and repeatedly.

In this podcast we discuss the challenges in developing software product lines. The discussion focuses on four key points:

1. How software product line development methodologies are helping to bring about successful re-use
2. Actions taken to align the software delivery process with product delivery processes
3. The importance of tools to implementing SPL methodology and processes
4. How to ensure that the business invests in SPL automation and processes for managing software variations that are aligned to product delivery processes

The discussion was led by Bola Rotibi, Principal Analyst at MWD.

The two guest speakers joining the podcast discussion were:

- James Cezo, Principal Engineer at Lockheed Martin Maritime Systems and Sensors, also known as MS2, a provider of surface, air, and undersea applications for U.S. and allied forces, including systems integration of network-centric naval combat systems.
- Dale Churchett, Senior Software Architect at HomeAway, a global online vacation rentals marketplace.

We thank both guest speakers for their valuable insights and contributions to the discussion.

## Key conclusions and recommendations

We begin with a summary of key conclusions and recommendations from the discussion.

### The do's and don'ts of SPL investment and implementation

In order to drive initial investment and on-going optimised adoption, the participants offered a number of recommendations based on experiences within their own organisations.

- **Develop the 'big picture' concept of why you want to implement a SPL approach** – It is important to get buy-in from all the major stakeholders. Being able to articulate the potential benefits to the development process and the product delivery process and how this will ultimately benefit the business will help sell the concept to upper management.
- **Think big but start small** – Balance big picture objectives with a plan to simultaneously start small so that the benefits can be proven quickly. Form a small team that will focus on the concepts, and another to carry out a pilot project to validate the technology approach, making sure that this fits with your existing tool sets. From here, once you have experienced some successes and have proof that it works, you can gain momentum and subsequently expand the team and enlarge the scope of the project.
- **Make variation management a core competency across the business, not just in the development team** – It is not enough to embed variation management approaches only within the development team. Do not forget to propagate this understanding up to the business owners, to marketing, into sales, and even up to the CEO. Permeating this approach through the business could offer you a major source of competitive advantage, so everyone needs to understand it and start thinking in terms of products and features.
- **Adopt a reactive model and marry it with an agile development process** – It is difficult to predict changing requirements a long way into the future, so react to requirements only as they come in and accommodate change as it is needed. This approach fits the agile development process nicely as short, sprint iterations make it possible to adapt to change quickly – you don't have to wait three months before you finally implement something.
- **As SPL is still a relatively new concept, seek education and support from specialist sources** – With old habits dying hard, SPL has yet to enjoy widespread adoption. However, the reported benefits should not be ignored. We encourage readers to explore the finer details of the actual technology and approach in order to broaden your understanding and be able to take a well-informed business case for change to your organisation.

## Podcast Summary

In this section we drill down into some of the detail under each of the four discussion points.

### 1. How software product line development methodologies are helping to bring about successful re-use

The discussion identified a number of ways in which software product line development methodologies offered a different and ultimately more productive approach to software development.

- **Intentional vs. opportunistic re-use** – Using software product line methodologies sets up effective software re-use from the beginning, with the intention that many products will be built from a set of common components. Using a software product line approach HomeAway has been able to achieve up to 95% of shared code between products. This contrasts to more traditional approaches where code is written and put ‘on the shelf’ to be used opportunistically by those willing to go and find it. Although there may be some initial benefits from ‘off the shelf’ software re-use, over time re-use can become more complex and diluted as new projects add new features or capabilities, without those new features and benefits being added back into the original software components. The result can be a multiplicity of similar yet variable software components for re-use. As time progresses developers have an increasingly complicated choice to make between the various available components.
- **Focus on variations between similar products** – Instead of taking a snapshot ‘clone and own’ approach to leveraging a new product from an existing product, the software product line approach focuses on the variations that are unique to the new project. In this way you engineer them into the same set of software artefacts that the original programme was with and continues to stay with, leading to multiple projects living out of the same set of software artefacts. By only needing to engineer the specific deltas, there is no need for a library approach: the deltas are inserted into the main, common software artefacts and then through the product line activation mechanism, you are able to instantiate a product specifically with key features enabled and key features selected on a product-by-product basis.
- **Identify early benefits but recognise that there are bigger long-term benefits to be had** – The strength of the SPL approach lies in the successful management and re-use of these variations. Trying to tease out what these variations are may take a bit of time, and may require you to have a number of iterations before you start to get good at it and become expert at recognising them. Once mastered, software product line methodologies offer long-term benefits.

### 2. Actions taken to align the software delivery process with product delivery processes

Both participants indicated that there was a distinct pattern in how the software delivery process aligned with the product delivery process. In other words, it is about aligning IT practices and philosophies with those of the business.

#### **The benefits of software product line approaches is felt first in development but this filters through the business over time once the results become apparent**

In both cases, the paradigm shift driven by a software product line approach and the resulting benefits were initially recognised with the development organisation. This is because it requires a significant mind shift in approach to delivering systems. Over time, the impact of the new approach seeps through to the business owners as they begin to see the effects on their own delivery processes.

## Best Practice Insight: Software Delivery InFocus Series

For HomeAway this has meant that they can ship a new international version of their website in approximately 30 hours of work (compared to the 2000 hours of work it took to build the original site). This now puts IT well ahead of the business who haven't been used to developing new requirements so quickly. Now the business has begun to recognise the speed at which IT can work and is starting to realise the benefits. This is filtering through the organisation up to the sales and marketing department who are more efficient in providing feedback on products and features and requesting new requirements.

**Acknowledging resistance to change and taking the appropriate steps is critical**

Both Lockheed Martin and HomeAway encountered resistance to change. Interestingly however, resistance came from varying quarters. Taking a software product line approach can represent a very different approach to building a product. For development teams software product line can present a significant challenge from a testing, quality assurance and configuration management perspective. For business managers, a proposed change in approach can provoke feelings of 'we've always done things this way and it works for us – we don't need to change'.

At Lockheed Martin, although their software developers and system engineers readily took to the software product line approach as a way of accelerating the development, it was more of a challenge to convince the programme managers who felt that their historical track record was evidence that things were already being done 'the right way'. In order to address this the team conducted a series of incremental prototype projects that demonstrated the validity of the software product line approach and showed how the particular methodologies enabled the pro-active growth of integrated re-use strategies.

At HomeAway, c-level executives were quick to appreciate the need to treat software development as a product line. The developers, however, were fairly entrenched in the legacy of the development system that was already running, despite identified issues with quality, speed and an excessive bug count. To shift the system over to a software product line approach-based tool would have been counter-productive. Instead, the first software product line features were introduced slowly, one step at a time, making it so easy that 'the developers didn't even notice it was there'. This provided a solid foundation for gradual change that was more easily adopted by the development team.

### 3. The importance of tools to implementing SPL methodology and processes

Both participants agreed that tools play an important part in the process – that's how the job gets done. However, an accompanying change in mindset is also critical.

- **New tools can drive fundamental changes in mindset** – Thinking about variations and different ways of managing applications can change traditional processes and can evolve the way developers and engineers think about the applications they are building. Tools can be critical to this as they challenge the way that developers are traditionally taught, i.e. to enable re-use through inheritance, through interfaces or through runtime configuration. Without the right tools, developers may go off and create their own way to implement variations. As an example of this, Churchett points to a system that had 30 different mechanisms; all were ingenious and clever, but nonetheless this meant 30 different ways of managing. With the right tool this was reduced to one or two ways.
- **Focus teams on multiple programmes/projects** – Instead of organising teams around just one particular product, expose teams to multiple programme requirements so they get used to working with common code. In this way, instead of driving divergence and multiple copies of components, engineers and developers are more able to readily adopt the mind shift needed to embrace building in specific feature variation within the software or within the requirements domain.

## Best Practice Insight: Software Delivery InFocus Series

Ultimately, whatever tools you use, this is irrelevant to the business. The business does not really care how you deliver the products, but that the product is delivered against their business objectives, on time, within budget and that it is of the highest quality possible. The choice of tool you adopt is felt more keenly by the development team, which feels the most benefit of day-to-day automation. This then translates into the streamlining benefits that are felt by the business.

#### 4. How to ensure that the business invests in SPL automation and processes for managing software variations that are aligned to product delivery processes

**Once SPL tools and processes are in place, the benefits become apparent relatively quickly, but getting initial buy-in isn't always easy**

Getting the business to invest in the software delivery team obtaining the benefit from the day to day experience of implementing SPL, requires you to demonstrate on a step by step basis the speed by which one can deliver an application that addresses a specific business outcome or goal. By doing this the business is able to witness the evidence themselves, making it hard for them to counter the value proposition of investing in an SPL strategy and tooling framework.

It's still early days for SPL so do not assume that there is a widespread understanding of what it is and what it can do for you. Although presenting technical arguments to all the stakeholders is a necessary first step, unless there is a good understanding of alternative or more traditional approaches to software re-use, it may be tough to get initial buy-in. It may be the case that it is possible to demonstrate evidence of the benefits to the business only once the software product line tools are already installed and are actually being used. This is clearly a frustrating 'chicken and egg' situation. In the case of HomeAway, the decision to invest relied heavily on Churchett's experience with SPL in previous roles.

However, once in place, it is possible to quite quickly show some immediate tangible benefits. One significant benefit experienced by HomeAway was the immediate improvement in quality issues and a cessation of the common practice of cross-team code breaking; there was no longer a need to 'roll back and constantly patch everyone's software'.