Production Planning for a Software Product Line

Dr. Gary J. Chastek and Linda M. Northrop
Software Engineering Institute

Dr. John D. McGregor
Clemson University

The goal of using a software product line (SPL) approach is to predictably develop multiple software-intensive systems (products) in an efficient, timely, and cost-effective manner that takes economic advantage of the features common to the products. Achieving this goal requires more than reusable (core) assets. It requires production planning that formulates a production strategy, devises a production method, and composes a production plan that is followed for each product. We present a three-step approach to production planning that guides an organization from the goals for the SPL to a comprehensive production plan.

The benefits of an effective production system are well understood. For example, in manufacturing, the Toyota Production System (TPS) produces high-quality, low-cost automobiles and is widely credited with providing Toyota with a significant competitive advantage. The TPS is based on a set of development rules [1], two of which are of particular relevance to SPLs:

- **Rule 1 – How People Work.** Tasks are rigorously specified yet the overall processes are highly flexible.
- **Rule 2 – How People Connect.** Customers and suppliers are connected directly and unambiguously, with defects and schedules handled rapidly.

Just as producing automobiles is Toyota's focus, producing software-intensive products is the main business of an SPL organization. An SPL is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way [2]. This definition is consistent with the one traditionally given for any product line. But it adds more, putting constraints on the way in which the systems in an SPL are developed. Why? Because substantial production economies can be achieved when the systems in an SPL are developed from a common set of assets in a prescribed way, in contrast to being developed separately, from scratch, or in an arbitrary fashion. It is exactly these production economies that make the SPL approach succeed in ways that earlier forms of software reuse failed.

Building a new product (system) becomes more a matter of assembly or generation than one of creation; the predominant activity is integration rather than programming. For each SPL, there is a production plan that specifies the exact product-building approach. And software components are not the only thing being reused across products. Other core assets include requirements, architecture, designs, documentation, budgets and schedules, tools, process definitions, performance models, testing artifacts, and much more.

The organizations that we have studied[1] have achieved remarkable benefits that are aligned with commonly held business goals, including:

- Large-scale productivity gains.
- Decreased time-to-market.
- Increased product quality.
- Decreased product risk.
- Increased market agility.
- Increased customer satisfaction.
- More efficient use of human resources.
- The ability to effect mass customization.
- The ability to maintain market presence.
- The ability to sustain unprecedented growth.

These benefits give organizations a competitive advantage and are derived from the reuse of the core assets in a strategic and prescribed way.

Implied by the definition of an SPL are two coordinated roles in product production. The core asset developer creates reusable artifacts that will be used to build multiple products. The product developer assembles products from these core assets—not arbitrarily, but in a *prescribed way*. In [3], the authors surveyed a number of organizations and found that these two roles are often embodied in separate teams that are not co-located. The artifacts produced during production planning are intended to coordinate the interactions of these two roles.

The core assets are designed to be reused, and reused in particular ways that accommodate the variation that is inherent in separate products. Although the product developers construct the products, they are constrained in what they can do because of how the core assets are designed. Production planning provides the opportunity for technical planning on how the core assets should be designed to facilitate efficient product variation and integration [4]. The production plan ultimately describes how to use the core assets to build products.

These described TPS rules illustrate some of the implications of the SPL definition for product production. In a product line production system, the prescribed way is intended to rigorously specify the roles in the organization. This specification is actually broken into pieces that correspond to the individual core assets used to build the products. Each asset is accompanied by a description of how to use the asset in product building. This attached process is used to populate portions of the production process. As different products are defined and different assets are selected for inclusion in the products, different attached processes are included in the production plan and *how people work* is changed.

In an SPL organization, the core asset and product developers are connected by a delivery process in which core assets are made available to the product developers. This is developed from a common set of core assets from the SPL definition. An explicit feedback mechanism allows the product developers to inform the core asset developers about any defects or to request desired enhancements. This is a realization of Rule 2 of the TPS.

In the following sections, we provide an overview of our approach to production planning, list several documented production planning experiences in both industry and the DoD, and describe a production planning workshop we have used with industrial and DoD customers. Due to the size of the artifacts, readers should examine the production planning artifacts of the Software Engineering Institute's (SEI's) Pedagogical Product Line [5]. This example SPL provides an extensive set of product line artifacts in addition to the production plan.

Approach

Our production planning technique involves three activities: context, prerequisites for planning, and planning activities [6]. Each activity produces an artifact that plays a specific role in the production system. We will describe the context in which the activities are conducted, the prerequisite actions that must occur prior to production planning, and then detail each of the three planning activities.

Context

Production planning occurs within the context of the SPL, the product line organization, and the narrower context defined by the product and production constraints. These constraints are identified during early product line analysis activities.

Activities such as scoping and market analysis identify production constraints. The structure of the organization can also impose constraints. The relationships among the customers, program offices, and contractors constrain production. These constraints include required divisions in process responsibilities brought on by the geographic distribution of personnel and legal divisions between the program office and the contractor.

Required properties of the products impose constraints. For example, the need for DO-178B certification imposes the requirement that the tools and processes used to produce the core assets and products be approved for this level of quality.

Prerequisites for Planning

Production planning begins early in the planning for the product line but it can only be completed when certain prerequisites have been met. The requirements for the production system are captured in a set of production scenarios. A production scenario takes the form shown in Table 1. Sufficient production scenarios are created to cover the various ways in which products will be produced.

Source of Stimulus	Who or what is initiating product production.
Stimulus	The event or action that initiates product development.
Environment	The state of the production environment of the product line at the time of this scenario (e.g., all core assets are completed and available for use).
Artifact	The production system artifact can be a product or a core asset.
Response	How the production system responds to the request to produce a specific product. For example, how long will it take to produce this product?
Response Measure	The measure may be calendar days from purchase contract to deployment, cost in dollars or days of effort, etc.

Table 1: *Production Scenario*

Planning Activities

The planning activities form a logical sequence that move planners from determining the goals of the production system to identifying the specifics of the production schedule for each product.

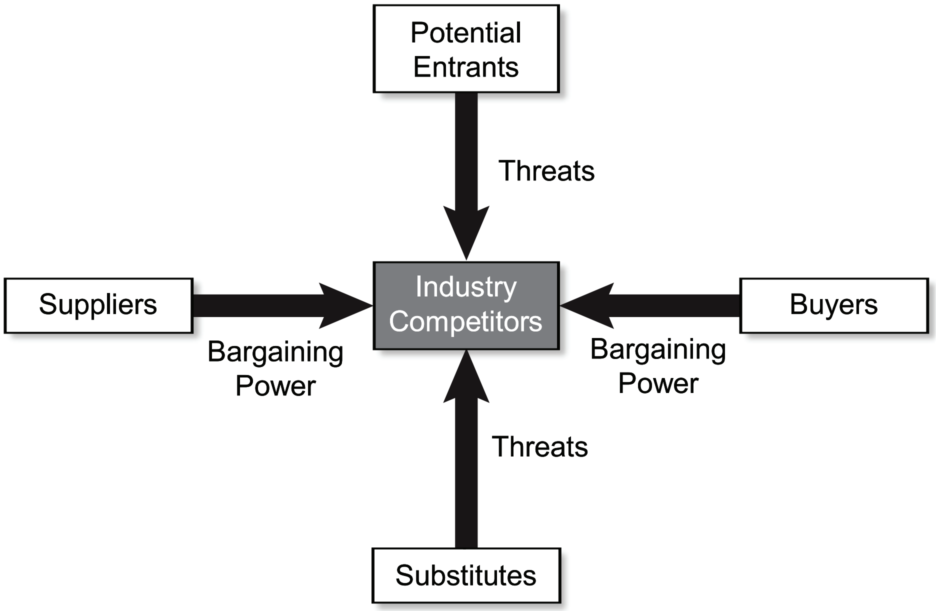
Defining the Production Strategy

The production strategy is a high-level statement of how the organization expects to achieve the goals of the product line. The breadth and longevity of a product line requires a goal-driven approach to keep the organization focused. The technique for defining the production strategy begins with the business goals of the product line organization.

The SEI *What to Build* pattern [2] focuses on defining which products are part of the product line, its scope, and developing a business case that justifies the investment in the product line. The business case is predicated on the organization's business goals and identifies the factors that are critical to the success of the product line.

We use Porter's Five Forces Model [7], illustrated in Figure 1, and the critical factors output from application of the *What to Build* pattern, to identify the strategic actions that will be the basis of the production strategy.

Figure 1: *Porter's Five Forces Model*



The boxes represent the market forces and the arrows represent threats and power. For example, potential entrants into a market represent a threat while buyers use their bargaining power to obtain discounts or improved products. Here we provide a brief description of each force:

- **Potential Entrants.** How can we raise the cost to others of entering the market by the means we use to produce the products? In an SPL context, this usually results in a strategic action to automate as much as possible, amortizing automation costs over the set of products. This in turn increases the cost of entry for potential competitors. A flexible production method can also respond to the needs of customers faster than product producers who have not yet entered the market.
- **Substitutes.** How can we differentiate

our product from the substitutes through the means of production? The economies of scale of the product line support a strategic action to lower prices while the economies of scope support an increase in features to resolve the threat of substitutes.

- **Buyers.** How can we better respond to buyers' requests through attributes of the production process? One strategy action is to adopt short iterations that provide enhanced functionality quickly. Another is to provide buyers with access to the status of defect fixes so they can track those that are important to them.
- **Suppliers.** How can we lower the prices we pay suppliers by the production techniques we use? The use of open source software is one strategic action. Another is to take advantage of the economies of scale of the product line to negotiate lower license fees for specific components.
- **Industry Competitors.** How can we gain advantage over the competition by different choices of production techniques? One strategic action is building multiple versions of products simultaneously. By establishing collaborative production arrangements with suppliers where we obtain early copies of future versions of their components, our products can be released much sooner after the release of the new components.

The strategy that results from this activity links the business goals of the product line to a first, high-level statement about how products will be produced. The strategy provides an essential input into the development of the production method. A detailed description of production strategy development can be found in [8].

The DoD context adds a layer to the usual customer/supplier relationship. The customer and program office each have goals for the product line and some of these goals can be achieved through the appropriate production techniques. The production strategy is ultimately the responsibility of the contractor who will build the product line, but it must encompass the goals of the customer and program office.

Engineering the Production Method

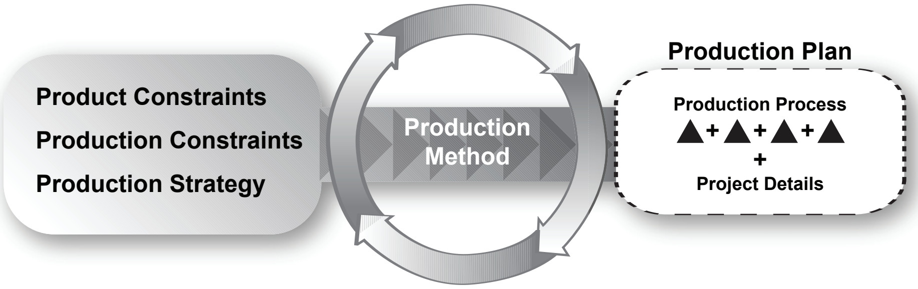
The production method [9] bridges the gap between the production strategy and production plan to provide a comprehensive view of the entire SPL development. While the production method is intended to describe how to produce a product, it also defines constraints on how the core asset developers can design their assets. The method becomes the vehicle for coordination between the core asset and product development teams.

> *"The production method bridges the gap between the production strategy and production plan to provide a comprehensive plan of the entire SPL development."*

The method encompasses the processes, tools, and models needed to completely describe a development effort. For example, the production method, in response to the production strategy, might adopt an Agile process model for productbuilding teams. The method would describe the roles and tasks for customers or customer surrogates and development team members. The method would define work products such as user stories, unit tests, and a software architecture, and assigns responsibility for their creation to specific roles.

The scope of the method varies from one organization to another. The method incorporates the processes and associated tools and models for building products, but the method may be expanded to include processes for specifying products and for product deployment. In some product line organizations, the production method may also include management activities related to estimating and scheduling production.

Figure 2: *Production Plan*



Development of the production method usually begins either with the single system development method that is in use as the product line organization is formed or with a standard software development method. Method engineering techniques [10] are used to elaborate that method into the full production method that addresses the scale and variability of the product line.

Developing the Production Plan

The production plan (shown in Figure 2) is the product builder's guide. It prescribes how products are produced from the core assets. It includes a process to be used for building products (the production process) and lays out the project details to enable execution and management of the process. The production plan is structured around the product building process defined in the production method. Just like a product specification, the production plan includes variation points; these include the variation points in the product specification as well as points related to the potential variations in the production system.

The production plan gives a step-by-step description of how to build a product. The initial description is high-level, but it is particularized as choices are made at variation points and core assets are selected to implement those choices. Each core asset is accompanied by an *attached process*. The attached process for an asset is the user's guide for that asset. The attached process solves the common software reuse problem of the learning curve for a new asset. The attached process for a core asset gives a step-by-step process for using the asset and should provide sufficiently detailed information to combat any difficulties in understanding how the asset works. The attached process for a selected asset is added into the product's production process when that asset is selected to be used to build the product. By making core asset selections in the prescribed order and adding the attached process for each asset as it is selected, a fully instantiated plan is developed just in time to guide the product developer [11, 12].

Production Planning Experiences

Our experience has shown the value of production planning:

- A survey was conducted of SPL Hall of Fame members to capture their experiences with production planning.

The full results of that survey are available in [3]. The most important result (for the purposes of this article) is that production planning was found to have a positive effect on the success of the product line. Organizations that did not sufficiently focus on production planning wished they had. Cummins, Inc. included more robust architecture and production planning practices in its second generation product line [13].

- Paul Jensen describes Overwatch's experience with production planning in a DoD context [14]. This experience illustrated the need to plan for production as early as possible. An attempt to change tools in the midst of core asset development was difficult and resulted in inadequate tool support.
- In [15], the authors describe a *user guide* that is essentially a production plan at Rolls Royce. The user's guide, for the core asset base, describes how to use the core assets to build products. It was built as a core asset and delivered with the core asset base.
- In [16], the authors provide a production planning technique that relies on the feature model. Production planning was decomposed into planning about how to include each feature in a product. By maintaining this traceability, the product line production plan is more easily transformed into the product-specific production plan. As features are selected during product definition, the production plan is composed.

The SEI developed a Production Planning Workshop [17] in response to requests for assistance on initiating a production planning capability. The workshop is an intensive planning session that is intended to expedite an organization's planning, providing a two-day introduction to the production planning process. Our experience with the introductory workshop has shown us that these two days are very useful in accelerating the production planning exercise.

Summary

A robust production planning technique—such as the one we have described—produces a production plan that is actionable and evolvable. The organization extracts production goals from the overall product line goals, creates a production strategy that resolves the competitive forces through the means of production, elaborates the strategy into a set of mutually consistent processes, tools, and models, and finally operationalizes the method in a detailed production plan. This sequence of successive refinements ensures that the production plan contains sufficient detail for core asset and product builders to accomplish their tasks predictably and efficiently. The traceability provided by this approach ensures that changes to product line goals or the discovery of additional production constraints can be propagated through the artifacts promptly.

A production system does not come free with reusable components, services, or even a product line architecture. A production plan is necessary. Developing and maintaining an effective and efficient production system is critical to the success of a product line.◆

References

1. TPS. "Toyota Production System Terms." Toyota Motor Manufacturing Kentucky, Inc. 2008 <www.toyotageorgetown.com/terms.asp>.
2. Clements, Paul, and Linda Northrop. Software Product Lines: Practices and Patterns. Boston: Addison-Wesley Publishers, 2002.
3. Chastek, Gary, Patrick Donohoe, and John D. McGregor. "A Study of Product Production in Software Product Lines." SEI, Carnegie Mellon University. Technical Note CMU/SEI-2004-TN-012. Mar. 2004 <www.sei.cmu.edu/pub/documents/04.reports/pdf/04tn012.pdf>.
4. Northrop, Linda, and Paul Clements. "A Framework for Software Product Line Practice, Ver. 5.0." SEI, Carnegie Mellon University. 2007 <www.sei.cmu.edu/productlines/framework.html>.
5. SEI. "Arcade Game Maker – Pedagogical Product Line." SEI, Carnegie Mellon University. 2008 <www.sei.cmu.edu/productlines/ppl/>.
6. Chastek, Gary, Patrick Donohoe, and John D. McGregor. A Production System for Software Product Lines. Proc. of the 11th Annual Software Product Line Conference. Kyoto, Japan, 2007: 117-128.
7. Porter, Michael E. "Competitive Advantage: Creating and Sustaining Superior Performance." Free Press, 1998.
8. Chastek, Gary, and John D. McGregor. "Formulation of a Production Strategy for a Software Product Line." SEI, Carnegie Mellon University. Technical Report CMU/SEI-2008-TN-023.
9. Chastek, Gary, Patrick Donohoe, and John D. McGregor. "Applying Goal-Driven Method Engineering to Product Production in a Software Product Line." SEI, Carnegie Mellon University. Technical Report to appear 2009.
10. Henderson-Sellers, Brian. "Method Engineering for OO Systems Development." Communications of the ACM 46 (Oct. 2003): 73-78. <http://portal.acm.org/citation.cfm?coll=GUIDE&dl=GUIDE&id=944242>.
11. Chastek, Gary, and John D. McGregor. "Guidelines for Developing a Product Line Production Plan." SEI, Carnegie Mellon University. Technical Report CMU/SEI-2002-TR-006. June 2002 <www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr006.pdf>.
12. Chastek, Gary, Paul Donohoe, and John D. McGregor. "Product Line Production Planning for the Home Integration System Example." SEI, Carnegie Mellon University. Technical Note CMU/ SEI-2002-TN-029. Sept. 2002 <www.sei.cmu.edu/pub/documents/02.reports/pdf/02tn029.pdf>.
13. Dager, Jim. Nomination presentation at Product Line Hall of Fame Session. 12th Annual Software Product Line Conference. Limerick, Ireland, Sept. 2008.
14. Jensen, Paul. Experiences With Product Line Development of Multi-Discipline Analysis Software at Overwatch Textron Systems. Proc. of the 11th Annual Software Product Line Conference. Kyoto, Japan, 2007.
15. Habli, Ibrahim, Tim Kelly, and Ian Hopkins. Challenges of Establishing a Software Product Line for an Aerospace Engine Monitoring System. Proc. of the 11th Annual Software Product Line Conference. Kyoto, Japan, 2007.
16. Lee, Jaejoon, K.C. Kang, and Sajoong Kim. A Feature-Based Approach to Product Line Production Planning. Proc. of the Third Software Product Line Conference. Boston, 2004: 183-196.
17. SEI. Software Product Lines. "Production Plans for Software Production Lines." 2008 <www.sei.cmu.edu/productlines/production_plan.html>.

Note

1. The SEI has published several detailed case studies of successful product line organizations and the benefits they have enjoyed. You can find these case studies in [2] and on the Web at <www.sei.cmu.edu/productlines/spl_case_studies.html>. You can also find references to product line efforts at <www.sei.cmu.edu/productlines/plp_hof.html> and <www.sei.cmu.edu/productlines/plp_catalog.html>.

About the Authors

Gary J. Chastek, Ph.D., is a senior member of the technical staff at the SEI in the SPL Initiative. Before joining the SEI, Chastek designed and implemented Ada compilers. Chastek's current research interests include production planning, variability management, and the use of open source development techniques in SPLs. He received his doctorate in computer science from the University of Pittsburgh in 1983.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213-2612
Phone: (412) 268-2826
E-mail: gjc@sei.cmu.edu

Linda M. Northrop is director of the Research, Technology, and System Solutions Program at the SEI, where she leads work in software architecture, SPLs, model-based engineering, integration of software-intensive systems, predictable software construction, and ultra-large-scale systems. She is coauthor of "Software Product Lines: Practices and Patterns" and "Ultra-Large-Scale Systems: The Software Challenge of the Future."

SEI
4500 Fifth AVE
Pittsburgh, PA 15213-2612
Phone: (412) 268-7638
E-mail: lmn@sei.cmu.edu

John D. McGregor, Ph.D., is an associate professor of computer science at Clemson University, a visiting scientist at the SEI, and a partner in Luminary Software, a software engineering consulting firm. His research interests include SPLs, model-driven development, and component-based software engineering. McGregor's latest book is "A Practical Guide to Testing Object-Oriented Software."

School of Computing
Clemson University
Clemson, SC 29634
Phone: (864) 656-5859
E-mail: johnmc@lumsoft.com

Web Sites

Software and Systems Process Improvement Network (SPIN)

www.sei.cmu.edu/collaborating/spins

A SPIN is an organization of professionals in a given geographical area who are interested in software and systems process improvement, and this Web site is the place to sign up. Joining a SPIN recognizes a commitment and loyalty to improving the state of software and systems engineering, as well as placing one in contact with a network of experts within their community. It is a practical forum for the interchange of ideas, information, and mutual support. Each regional SPIN is slightly different, based on the vision of the founders and the needs of the community. SPINs are made up of professionals from all sectors—industry, government, and academia (including students)—and include defense contractors, professional organizations, and independent consultants.

Software Product Lines (SPLs)

www.softwareproductlines.com

Devoted to the community of software engineers and managers interested in using SPL approaches to develop their software, the goal of this site is to provide software developers, product managers, and development managers with practical information on SPL issues, ranging from introductory concepts to advanced techniques. Learn about SPL concepts, the first steps for using SPL approaches, the benefits that may help convince bosses to use SPL, success stories, expert perspectives, and resources to use for learning more about SPL.

The Goldratt Institute

www.goldratt.com

What should an organization do when confronted with low overall performance results, difficulties securing or maintaining a strategic advantage in the marketplace, financial hardships, seemingly constant firefighting, poor customer service, and chronic conflicts between people? Utilizing the Theory of Constraints (TOC) may help. The Goldratt Institute, birthplace of the TOC, is a leading provider of TOC expertise, development, implementation, and education. Their approach begins with the development of corporate strategy and fans out to all operational aspects of a given organization, tightly integrating the strengths of Lean Six Sigma into an overarching TOC-based solution. Once the barriers that block parts of an organization from working together as an integrated system are removed, the result is significant and sustainable improvement in each problem area. This Web site provides robust, customizable processes, expert-level training and certification, technical support, mentorship, training materials, and planning in support of all TOC practices.

Validating Java for Safety-Critical Applications

http://javolution.org/doc/Man33955.pdf

With the real-time extensions, Java can now be used for safety-critical systems. It is therefore imperative to ensure that virtual machine implementations not only conform to the Real-Time Specification for Java (RTSJ) but also that efficiency and predictability are up to a certain standard. In particular, if the overhead incurred by RTSJ implementations are beyond a certain threshold, they may not suitable for safety-critical systems. With this in mind, the Web site outlines the development and maintenance of a test suite that addresses conformance as well as performance, and proved to be extremely useful in the critical process of selecting the Java Virtual Machine Platform.