Introduction to Software Product Lines

Patrick Donohoe
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
**Report Documentation Page**

<table>
<thead>
<tr>
<th>1. REPORT DATE</th>
<th>2. REPORT TYPE</th>
<th>3. DATES COVERED</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 OCT 2014</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. TITLE AND SUBTITLE</th>
<th>5a. CONTRACT NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Software Product Lines (Slides)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. AUTHOR(S)</th>
<th>5b. GRANT NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Donohoe /Patrick</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</th>
<th>5c. PROGRAM ELEMENT NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. PERFORMING ORGANIZATION REPORT NUMBER</th>
<th>10. SPONSOR/MONITOR’S ACRONYM(S)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

12. DISTRIBUTION/AVAILABILITY STATEMENT  
Approved for public release, distribution unlimited.

13. SUPPLEMENTARY NOTES  
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:  
   a. REPORT  
      unclassified  
   b. ABSTRACT  
      unclassified  
   c. THIS PAGE  
      unclassified

17. LIMITATION OF ABSTRACT  
   SAR

18. NUMBER OF PAGES  
   24

19a. NAME OF RESPONSIBLE PERSON

---

Standard Form 298 (Rev. 8-98)  
Prescribed by ANSI Std Z39-18
Business Success Requires Software Prowess

Software pervades every sector. Software has become the bottom line for many organizations, even those who never envisioned themselves in the software business.
Few Systems Are Unique

Most organizations produce families of similar systems, differentiated by features.

A reuse strategy makes sense.
What Is a Software Product Line?

A *software product line* 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.

- a new application of a proven concept
- an innovative, growing concept in software engineering
Software Product Lines

Product lines
- take economic advantage of commonality
- bound variation
How Do Product Lines Help?

Product lines amortize the investment in these and other *core assets*:

- requirements and requirements analysis
- domain model
- software architecture and design
- performance engineering
- documentation
- test plans, test cases, and test data
- people: their knowledge and skills
- processes, methods, and tools
- defect elimination
- budgets, schedules, and work plans
- components and services

**PRODUCT LINES = STRATEGIC REUSE**
Organizational Benefits

Organizations use product line practices to

• achieve large-scale productivity gains
• improve time to market
• maintain market presence
• sustain unprecedented growth
• achieve greater market agility
• compensate for an inability to hire
• enable mass customization
• get control of diverse product configurations
• improve product quality
• increase customer satisfaction
• increase predictability of cost, schedule, and quality
# Costs of a Software Product Line

<table>
<thead>
<tr>
<th>Core Assets</th>
<th>Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>Must support variation inherent in the product line</td>
</tr>
<tr>
<td>Software Components</td>
<td>Must be designed to be general without a loss of performance; must build in support for variation points</td>
</tr>
<tr>
<td>Test Plans, Test Cases, Test Data</td>
<td>Must consider variation points and multiple instances of the product line</td>
</tr>
<tr>
<td>Business Case and Market Analysis</td>
<td>Must address a family of software products, not just one product</td>
</tr>
<tr>
<td>Project Plans</td>
<td>Must be generic or be made extensible to accommodate product variations</td>
</tr>
<tr>
<td>Tools and Processes</td>
<td>Must be more robust</td>
</tr>
<tr>
<td>People, Skills, Training</td>
<td>Must involve training and expertise centered around the assets and procedures associated with the product line</td>
</tr>
</tbody>
</table>
Economics of Product Lines

Economics of Product Lines

Product Line Practice

Contexts for product lines vary widely, based on

- nature of products
- nature of market or mission
- business goals
- organizational infrastructure
- workforce distribution
- process discipline
- artifact maturity

But there are universal essential activities and practices.
The Three Essential Activities
Different Approaches - 1

**Proactive:** Develop the core assets first.
- Develop the scope first and use it as a “mission” statement.
- Products come to market quickly with minimum code writing.
- Requires up-front investment and predictive knowledge

**Reactive:** Start with one or more products.
- From them, generate the product line core assets and then future products; the scope evolves more dramatically.
- Much lower cost of entry
- The architecture and other core assets must be robust, extensible, and appropriate to future product line needs.
Different Approaches - 2

**Incremental:** In either a reactive or proactive approach, it is possible to develop the core asset base in stages, while planning from the beginning to develop a product line.

- Develop part of the core asset base, including the architecture and some of the components.
- Develop one or more products.
- Develop part of the rest of the core asset base.
- Develop more products.
- Evolve more of the core asset base.
- …
The SEI Framework for Software Product Line Practice℠

The SEI Framework for Software Product Line Practice is a conceptual framework that describes the essential activities and twenty-nine practice areas necessary for successful software product lines.

The Framework, originally conceived in 1998, is evolving based on the experience and information provided by the community.

Version 4.0 –
in Software Product Lines: Practices and Patterns

Version 5.0 –
http://www.sei.cmu.edu/productlines/tools/framework/index.cfm

℠ Framework for Software Product Line Practice is a service mark of Carnegie Mellon University.
## Framework Version 5.0

<table>
<thead>
<tr>
<th>PRACTICE AREAS</th>
<th>ESSENTIAL ACTIVITIES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software Engineering</strong></td>
<td><strong>Technical Management</strong></td>
</tr>
<tr>
<td>Architecture Definition</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>Architecture Evaluation</td>
<td>Make/Buy/Mine/Commission Analysis</td>
</tr>
<tr>
<td>Component Development</td>
<td>Measurement and Tracking</td>
</tr>
<tr>
<td>Mining Existing Assets</td>
<td>Process Discipline</td>
</tr>
<tr>
<td>Requirements Engineering</td>
<td>Scoping</td>
</tr>
<tr>
<td>Software System Integration</td>
<td>Technical Planning</td>
</tr>
<tr>
<td>Testing</td>
<td>Technical Risk Management</td>
</tr>
<tr>
<td>Understanding Relevant Domains</td>
<td>Tool Support</td>
</tr>
<tr>
<td>Using Externally Available Software</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The product line architecture is central to success.
The **right architecture** paves the way for system **success**. The **wrong architecture** usually spells some form of **disaster**.
At the Heart of Successful Product Lines

A pressing need that addresses the heart of the business

Long and deep domain experience

A legacy base from which to build

Architectural excellence

Process discipline

Management commitment

Loyalty to the product line as a single entity
The Product Line Adoption Endgame

To have an **operational software product line**.

To do that, an organization must

- have
  - a core asset base
  - supportive processes and organizational structures
- develop products from that asset base in a way that achieves business goals
- prepare itself to institutionalize product line practices
Widespread Use of Software Product Lines

Successful software product lines have been built for families of among other things

- mobile phones
- shipboard command and control systems
- satellite ground-station systems
- avionics systems
- command and control/situational awareness systems
- pagers
- engine control systems
- mass storage devices
- billing systems
- Web-based retail systems
- printers
- consumer electronic products
- acquisition management enterprise systems
- financial and tax systems
- medical devices
- fish farm management software
In a Nutshell

Software product lines epitomize the concept of strategic, planned reuse.

The product line concept is about more than a new technology. It is a new way of doing one’s software business.

There are essential product line activities and practices areas as well as product line patterns to make the move to product lines more manageable.
Contact Information

Patrick Donohoe
Software Engineering and Acquisition Practices Directorate
Telephone: +1 412-268-7616
Email: pd@sei.cmu.edu

World Wide Web:
www.sei.cmu.edu/productlines

U.S. Mail:
Software Engineering Institute Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA
Telephone: +1 412-268-7616
Email: pd@sei.cmu.edu

Customer Relations
Email: customer-relations@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257