15 Jahre Software-Produktlinien: Einführung und aktueller Stand

Mini-Tutorial

Dr. Andreas Birk (Software.Process.Management), Prof. Dr. Klaus Schmid (Universität Hildesheim)
PIK2009, Berlin
25. Mai 2009

Agenda

What is a Software Product Line (SPL)?

History of the SPL discipline

Concepts & approaches to SPL development

SPL case study

Key questions of SPL research & practice
Terminology: Product Family

We consider a set of programs to constitute a **family**, whenever it is worthwhile to study programs from the set by …

**first** studying the common properties of the set and

**then** determining the special properties of the individual family members.

*(David L. Parnas, 1976)*

---

Terminology: Product Line

A **software product line (SPL)** is …
a set of software-intensive systems
that share a common, managed set of features
satisfying 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.

*(Clements and Northrop, SEI, 2002)*
Terminology: Product Line

**Marketed (Software) Product Line:**
A set of products that are marketed together as sharing a common set of concepts or features.

**Engineered (Software) Product Line:**
A set of products that are engineered together so as to share major parts of their implementation.

Reuse

Build something
from existing or pre-produced items
Advantages of Reuse

Quality

Productivity

Time

Reuse Items

Test Cases

Estimates

Architecture

Models

Requirements

Code

Subsystems

Rules

Documentation
But…

...most traditional software reuse approaches actually failed?

So what is new?

opportunistic
vs.
systematic

explicit differentiation in:
• development for reuse
• development with reuse

Agenda

What is a Software Product Line (SPL)?

History of the SPL discipline

Concepts & approaches to SPL development

SPL case study

Key questions of SPL research & practice
History of the SPL discipline

- Parnas: Product Families (1976)
- European Research Programs (late 1990ies)
- First Conference of International SPLC/PFE Series (1996)
- Reuse Libraries
- Framework-Based Reuse
- SEI Book (2002)
- Model-Based and Generative Approaches
- Increasing acceptance of SPL approaches in industry

Agenda

What is a Software Product Line (SPL)?

History of the SPL discipline

Concepts & approaches to SPL development

SPL case study

Key questions of SPL research & practice
Very often, composition and customization require extensive modification of "reused" items. This reduces or even outweighs the expected reuse benefits significantly.

Reuse, the Old Way: Shortcomings

Library did not contain needed items

Selection tedious

Mostly adaptation is needed

Too little composition

Too much customization

People tend to build from scratch
Reuse: Product Lines

Platform / Core Assets

Select & Compose

Customize

Product

Product Lines: Technical Challenges

PLE Temple

2-4 times lower cost
Higher quality

Product Line Engineering

Good Software Engineering Practices
SPL Type: Business Rationale

**Product Suites**
- Existing applications are integrated over time
- Customer uses more than one product from SPL
- SPL motivation: Reduced cost of ownership
- Example: Microsoft Office

**Customized Products**
- New applications are instantiated from the framework
- Customer uses only one product from SPL
- SPL motivation: Delivery efficiency
- Examples: Cummins, Bosch, Danfoss

---

**Setting up a Product Line**

Anticipate

Future Products
Scope: *Product Family to be developed as Product Line*

Define Requirements & Domain Architecture

Develop Platform / Core Assets

Establish SPL Processes & Practices
Operating a Product Line

Legend: PL = Product Line; PF = Platform; Prod = Product; Cust = Custom; Req = Requirement; Impl = Implementation

Challenges of SPL: Requirements

Anticipate Requirements

Describe Commonality & Variability

Maintain large requirements base over very long lifetime

Split requirements into platform and variant

Derive variant- & version-specific requirements views
Variability Management

Core Idea of Product Line Engineering

- Product 1
- Product 2
- Product 3

Core idea: Similarity of Products = Commonality + (reg.) Variability + Product-specific parts

Develop once  →  Make selectable  →  Single development

Variability Management

Product-oriented variation

<table>
<thead>
<tr>
<th>Generic parts</th>
<th>Variation product 1</th>
<th>Variation product 2</th>
<th>...</th>
<th>Variation product n</th>
</tr>
</thead>
</table>

Concept-based variation

<table>
<thead>
<tr>
<th>Generic parts</th>
<th>Concept 1</th>
<th>Concept 2</th>
<th>...</th>
<th>Concept n</th>
</tr>
</thead>
</table>

Product Definition

Product 1: concept 1, 2, 4, 5, ...
Product 2: concept 1, 3, 4, 6, ...
Definitions:

- **Product-specific:**
  an artifact (or characteristic) that is specific to a product

- **Variability:**
  an artifact (or characteristic) that exists in some, but not all, products of the product line

- **Commonality:**
  an artifact (or characteristic) that exists in all products of the product line

**SPL Types: Product Instantiation**

- **Constructive**
  • Customization based on framework, library, or shared components

- **Compositional**
  • Fixed building blocks with parametrization
  • Parametrization during build process or during installation; never exposed to end user

- **Generative**
  • Model-driven development with artefact generation

- **Combination**
  • For instance, generation of shared components with subsequent customization
Mapping Organization and Architecture

Separate platform team

Platform under control of product engineering projects

Mixed

Phased combination

Product Lines: Technical Challenges

Anticipate scope & requirements

Define & maintain domain architecture

Develop & maintain platform / core assets

Establish mature software engineering practices
Challenges of SPL: Organization

Very many stakeholders

Diverging interests

Changes of requirements & products

Product Line Framework

**Process: Overall SPL Lifecycle**

- **Initiation and Preparation**
  - Initiation, evaluation, and decision
  - Planning and preparation, establishment of infrastructure

- **Set-up and Operation**
  - Development of core assets
  - Development of product instances
  - Evolution of core assets
  - Evolution of SPL

- **Closure**
  - Discontinuation of core assets
  - Discontinuation of SPL

Different life cycle phases require different organizational structures and processes.

---

**Agenda**

What is a Software Product Line (SPL)?

History of the SPL discipline

Concepts & approaches to SPL development

**SPL case study**

Key questions of SPL research & practice
Case Study: Small

Market Maker Software AG
Small organization (< 25 developers total; ~50 people total)
Company started in 1984 with a focus on developing stock market software
Since then development of various products (product line)
  • Desktop / PC-based systems
  • Delphi-/C++-based development
Large range of customers (from consumer-market to large banks)

New product line (i* product line) started in 1999
  • Web-based products
  • Java-development
  • Development started from scratch

Case Study: Small

Market Maker: Product Line Engineering Results
New product line highly successful
  – Running for more ~ten years now
  – Large number of products

  + Reduction of Time-to-Market: 2-4
  + Break-Even: after about five products
  + Reduction of maintenance cost: ~60%
  + Reduced cost of quality (reliability in the field)
  – Increase of issue resolution time
Case Study: Large

Philips Medical Systems
• Large organization (> 1000 developers)
• Focus on medical imaging systems, e.g.,
  – Acquisition: X-Ray, Magnetic Resonance Imaging (MRI)
  …
  – Image processing, viewing & storing
• World-wide development
• Development structured around product groups
  – cross product group reuse
  – core functionality: imaging products (storing, retrieving, processing)

Case Study: Large

Product Line Engineering Results
• Successful transition
• Integration of additional companies
  + Reduction of Effort: 2-4
  + Time-to-Market reduction: ~50%
  + Product defect density: <50%
  + Reduction of maintenance cost: ~60%
  + Common look and Feel
  + Better product planning and use of roadmaps
Some Literature

Further Material
- Product line engineering from a practitioner perspective
- Families Evaluation Framework
- Many industrial case studies!

http://www.spl-book.net

Gleichheit in Vielfalt –
Produktlinien die Zukunft der industriellen Softwareentwicklung

iX-Archiv, 5/2008, Seite 110

Linden, Schmid, Rommes
Product Lines in Action
Springer, 2007

Agenda

What is a Software Product Line (SPL)?

History of the SPL discipline

Concepts & approaches to SPL development

SPL case study

Key questions of SPL research & practice
Key Questions of SPL Research & Practice

How to model complex real-world variability?

How to integrate SPL with other methodologies?

How to create awareness of SPL demands?

How to switch over from non-SPL development to SPL development?

Kontakt

Dr. Andreas Birk
Software.Process.Management
info@swpm.de
http://www.swpm.de

Prof. Dr. Klaus Schmid
Universität Hildesheim
schmid@sse.uni-hildesheim.de
http://www.sse.uni-hildesheim.de