More on Software Architecture and Product Lines

SEI Update

GSAW 2000

Linda M. Northrop
Director, Product Line Systems Program

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This work is sponsored by the U.S. Department of Defense.
Today’s Talk

Context (again!)

Software Architecture
- ABD
- ATAM<sup>SM</sup>
- ABAS

Product Lines
- Supporting Trends
- Benefits
- Pervasiveness
- DoD Response
- Impact
- SEI Product Line Practice Framework

Conclusion
What Is a Product Line?

A product line is a group of products sharing a common, managed set of features that satisfy specific needs of a selected market.
Product Line
Cottage Industry
Assembly Production
Software Product Lines

- Market strategy/Application domain
- Architecture
- Components

Product lines
- take economic advantage of commonality
- bound variability
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 data
- people: their knowledge and skills
- processes, methods, and tools
- budgets, schedules, and work plans
- components

product lines = strategic reuse
The Key Concepts

Use of a common asset base in production of a related set of products
The Key Concepts

Use of a common asset base in production

of a related set of products

Architecture

Production Plan

Scope Definition Business Case
Customers/Collaborators

Caterpillar
Robert Bosch Co.
Hewlett Packard
LLNL
EPA
FAA
USCG
NRO/CCT
JNTF
DMSO
US Army SOA: TAPO
US Army CECOM
US Navy TENA
US Airforce: F-22
NASA

Lucent
AT&T
Thomson-CSF
Ericsson
Raytheon
Siemens
Schlumberger
Cummins Engine Co.
Nokia
Telesoft S.p.A
Boeing
CelsiusTech
Buzzeo
ALLTEL
Motorola
General Motors
Today’s Talk

Context (again!)

Software Architecture

- ABD
- ATAM\textsuperscript{SM}
- ABAS

Product Lines

- Supporting Trends
- Benefits
- Pervasiveness
- DoD Response
- Impact
- SEI Product Line Practice Framework

Conclusion
Necessary Changes

The architecture is the foundation of everything.
Risks and Mitigation Strategies

Poor quality architecture is among the top ten risks associated with software product lines.

To mitigate the risk, care must be taken during architecture definition and evaluation.

Our architecture work has therefore focused on:
- Architecture-Based Design
- Architecture Analysis: Architecture Tradeoff Analysis Method$^{SM}$ (ATAM)
- Analyzable Designs: Attribute-based Architectural Styles (ABAS)
Architecture-Based Design

A refinement method designed to organize the earliest design decisions
Architecture Design Considerations

Variabilities and Commonalities
• use cases with variation points and quality scenarios

Software Templates
• categorize design elements into types

Architectural Drivers
• combination of functional and quality requirements that drive the design
ABD Method Within the Life Cycle

- **Requirements Analysis**
  - Functional Requirements
    - abstract
    - use cases
  - Quality Requirements
    - abstract
    - quality scenarios
    - architecture options
  - Constraints

- **ABD Method**
  - Abstract Components
  - Software Templates
  - Constraints
  - Requirements

- **Concrete Component Design**

- Business Case
- Architect's Experience
- Legacy Systems
Engineering Quality Attributes

We need to identify and analyze risks at the stage of architecture design.

To do this we need suitable architecture analysis techniques.

And we need analyzable designs.

How do we get these?
ATAM is an architecture evaluation method that

- focuses on multiple quality attributes

- illuminates points in the architecture where quality attribute tradeoffs occur

- generates a context for ongoing quantitative analysis

- utilizes an architecture’s vested stakeholders as authorities on the quality attribute goals
The ATAM

We have been developing the Architecture Tradeoff Analysis Method (ATAM) for over two years.

The purpose of ATAM is: to assess the consequences of architectural decision alternatives in light of quality attribute requirements.
ATAM Steps

1. Present ATAM
2. Present business drivers
3. Present architecture
4. Identify architectural styles
5. Generate quality attribute utility tree
6. Elicit and analyze architectural styles
7. Generate seed scenarios
8. Brainstorm and prioritize scenarios
9. Map scenarios onto styles
10. Present out-brief and/or write report
ATAM Techniques

- Utility Tree Generation
- Style-Based Elicitation/Analysis
- Scenario Brainstorming/Mapping
Building Upon Styles and Design Patterns: ABASs

Architectural styles and design patterns are a wonderful (and necessary) idea.

They describe the essence of a recurring design problem and its solution.

Attribute-based architectural styles (ABASs) add explicit quality attribute analysis models to reason about the costs/benefits of a pattern or style.
ABAS Motivation

Why add a quality attribute-based modeling framework to an architectural style?

• to make architectural design more routine and predictable
• to have a standard set of important attribute-based analysis questions associated with the style
• to tighten the link between design and analysis
Analysis Models

To aid in structuring an ABAS and in understanding each attribute, we are using attribute characterizations.

For each attribute, the characterization describes:
- the stimuli of interest
- the architectural style (and its parameters)
- the responses
1. **Problem Description and Criteria:** characteristics of the problem solved.

2. **Stimuli/Responses:** the ABAS’s quality attribute specific stimuli and the measures of the responses.

3. **Architectural Style:** components, connectors, parameters, topology, constraints.

4. **Analysis:** formally relating quality attribute models to the style; heuristics for reasoning about the style.
ABAS Support

We are building a handbook with a collection of ABASs

Analysis  Design

© 2000 by Carnegie Mellon University 27
Example Handbook Contents

**Performance:**
- Concurrent Pipelines
- Multiple Messages
- Synchronization
- Cache
- Client/Server

**Modifiability:**
- Abstract Data Repository
- Layers
- Publish/Subscribe
- Data Indirection

**Availability:**
- Analytic Redundancy
- Simplex
- Trimodular Redundancy

**Security:**
- Firewall
- Virtual Private Network
- Encryption/Decryption

**Usability:**
- Undo
- Cancel
- Visualization
Connections

ABAS

ATAM
ABD

Quality Attribute Workshop
Architecture

Component quality (CBE)

Component Assembly Plan (CBE)
production plan

Software Product Lines

© 2000 by Carnegie Mellon University
Today’s Talk

Context (again!)

Software Architecture
- ABD
- ATAM\textsuperscript{SM}
- ABAS

Product Lines
- Supporting Trends
- Benefits
- Pervasiveness
- DoD Response
- Impact
- SEI Product Line Practice Framework

Conclusion
Trends Supporting Product Lines

Proliferation within major organizations of self-sustaining architecture centers.

Growing acceptance of the importance of architecture.

Standardization of commercial middleware.

Growing popularity of the notion of "rapid development."

Community acceptance of well-defined processes for software development.

Growing acceptance in the software engineering community of the importance of product line practices and the rising recognition of the amazing cost/performance savings that are possible.
Benefits

Improved productivity
  by as much as 10x

Decreased time to market (to field, to launch...)
  by as much as an order of magnitude

Decreased cost
  by as much as 60%

Increased quality
  as measured by customer satisfaction
# Pervasiveness

**First Software Product Line Conference**  
**Paper Submissions**

<table>
<thead>
<tr>
<th>Region</th>
<th>Submissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>North America</td>
<td>25</td>
</tr>
<tr>
<td>Europe</td>
<td>25</td>
</tr>
<tr>
<td>Asia</td>
<td>7</td>
</tr>
<tr>
<td>So America/So Africa</td>
<td>2</td>
</tr>
<tr>
<td>Academic</td>
<td>27</td>
</tr>
<tr>
<td>Industry</td>
<td>32</td>
</tr>
<tr>
<td>* of these, 10 from DoD community*</td>
<td></td>
</tr>
<tr>
<td>Research</td>
<td>19</td>
</tr>
<tr>
<td>Experience</td>
<td>18</td>
</tr>
<tr>
<td>Either/both</td>
<td>12</td>
</tr>
</tbody>
</table>
Some DoD Options

Scope the product line and develop the architecture
Acquire a product line architecture
Acquire the core asset base
Acquire a product built using product line technology
Acquire a product and some set of reusable assets
Acquire products built from a government asset base
Acquire an entire product line
Acquire products built from a non-government asset base
DoD Strategy

Domain Understanding •••••• Scope Definition

DoD Options •••••• Business Case

Product Line Goals ••••••
DoD Response

30 from the DoD community participated in SEI Second DoD Product Line Practice Workshop (March 1999).

• They talked about how they were doing or going to do product lines, not how it would be impossible in the DoD.

The current Defense Science Board Task Force is recommending product lines.
Bosch is a major participant in ESAPS

New Bosch Research Center in Pittsburgh

Two dashboard prototypes

New dashboard business unit

Glass Dashboard R&D Effort

Corporate Domain Engineering Team

Product Lines are one of three goals of new Bosch Initiative in Software Systems (BISS)
Web-based, evolving document

Describes product line essential activities

Describes essential and proven product line practices in the areas of
  • software engineering
  • technical management
  • organizational management

Addresses development and acquisition contexts
Current Status of Framework

Version 2.0 is now on our Web Site

http://www.sei.cmu.edu/plp/framework.html

Version 2.0 differs from Version 1.0

✔ modified list of practice areas
✔ added nine additional practice area descriptions
✔ improved acquisition context coverage
✔ improved the six practice area descriptions in V1.0
✔ included an FAQ section

Currently known to be used by 20 organizations in their product line efforts
Product Line Essential Activities

Product Line Development / Acquisition Process

Core Asset Development / Acquisition

Management

Domain Engineering

Application Engineering
Practice Area Categories

SOFTWARE ENGINEERING

TECHNICAL MANAGEMENT

ORGANIZATIONAL MANAGEMENT
Software Engineering Practice Areas

- Understanding Relevant Domains
- Mining Existing Assets
- Architecture Definition
- Architecture Evaluation
- Component Development
- Testing
- Requirements Engineering
- COTS Utilization
- Software System Integration
Technical Management Practice Areas

- Data Collection, Metrics and Tracking
- Product Line Scoping
- Configuration Management
- Process Modeling
- Planning
- Make/Buy/Mine/Outsource Analysis
- Technical Risk Management
- Tool Support

* in Version 1.0
☆ in Version 2.0
Organizational Management Practice Areas

- Achieving the Right Organizational Structure
- Building and Communicating a Business Case
- Funding
- Market Analysis
- Developing and Implementing an Acquisition Strategy
- Operations
- Training
- Customer Interface Management
- Technology Forecasting
- Launching and Institutionalizing a Product Line
- Organizational Risk Management

© 2000 by Carnegie Mellon University
Key Themes Among Successful Product Lines

Long and deep domain experience

A legacy base from which to build

Architectural excellence

Management commitment
Remarks

The SEI framework for software product line practice is intended to be a living document.

Version 2.0 is the second iteration.

Future versions will incorporate
• Additional practice area descriptions
• Usage scenarios
• Practice area dependency descriptions

The SEI conducts *product line technical probes* based upon the framework to examine whether an organization is “fit for product lines.”
Today’s Talk

Context (again!)

Software Architecture
- ABD
- ATAM\(^{SM}\)
- ABAS

Product Lines
- Supporting Trends
- Benefits
- Pervasiveness
- DoD Response
- Impact
- SEI Product Line Practice Framework

Conclusion
Conclusion

There has been much technological and experiential progress in the last year both in software architecture and software product lines.

The time is right to make software product lines a DoD reality.

Join us at:
- Third DoD Product Line Practice Workshop (March 13-14, 2000)
- SPLC1 (August 28-31, 2000)

...to push the frontier further.
<table>
<thead>
<tr>
<th>For Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Telephone</strong></td>
</tr>
<tr>
<td><strong>FAX</strong></td>
</tr>
<tr>
<td><strong>Email</strong></td>
</tr>
<tr>
<td><strong>World Wide Web</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>U.S. mail</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>