

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Home

Keynotes
Program
Tutorials
Workshops
Panels
Registration
Local Information

CFP
Submissions
Demonstration
Doctoral symposium
Product Line Hall of
Fame

Organization
Program Committee
Sponsor

call-for-participation (pdf)

call-for-participation (txt)

The 11th International Conference of Software Product Line Conference draw to a successful close. We have 220+ participants from 20+ countries. We would like to thank all those who contributed to the success of this conference. The next [SPLC 2008](#) conference will be held in Limerick, Ireland, 8-12 September 2008. We look forward to seeing you in Limerick.

Software Product Lines - Power for Competitive Advantage

The current trend of globalization is pressuring industries to explore ways to meet diverse needs of the global market most effectively and efficiently. Over the last decade or so, software product line has emerged as one of the most promising software development paradigms in drastically increasing the productivity of IT-related industries, and the product line community has grown and is still growing rapidly. At this very important juncture, the Software Engineering Center (SEC) of Information-Technology Promotion Agency (IPA), Japan is proud to sponsor the Eleventh International Software Product Line Conference (SPLC 2007) in Kyoto, Japan, the first ever software product line conference held in Asia.

SPLC is the most prestigious and leading forum for researchers, practitioners, and educators in the field. SPLC 2007 will provide a venue for exchanging, sharing, and learning technologies and industrial experiences to the community. The conference will feature research and experience papers, tutorials, workshops, panels, and demonstrations. Especially, we will organize special workshops to address the needs of specific industrial segments, including automobile, mobile communications, embedded controllers, and enterprise software.

We encourage you to submit technical papers, and proposals for panels, tutorials, workshops, and demonstrations. We look forward to interacting with you at SPLC 2007.

Topics of interest of SPLC 2007 include, but are definitely not restricted to:

- Industrial experiences in product line engineering
- Techniques and tools for product line engineering
- Evolution of product line assets

New!

October. 22, 2007

[Keynote slides are uploaded.](#)

June. 15, 2007

[Registration site open!](#)

May. 31, 2007

[Doctoral Symposium](#) deadline extended to June 10.

日本語による情報
Information in Japanese

- Business issues for product lines
- Organizational and process issues for product lines
- Product line life-cycle issues

Special features of SPLC 2007 are:

- Special interest group workshops will be organized
 - To address the needs of specific industries including automobile, mobile communication, embedded controller, and enterprise SW
 - To encourage participation from industry
- Proceedings will be published in two volumes to encourage participation:
 - Main volume: technical papers
 - Addendum: poster papers, workshop position papers, panel position papers, demo summaries, summary of Hall of Fame inductees - this volume will have a separate ISBN

Contact Information:

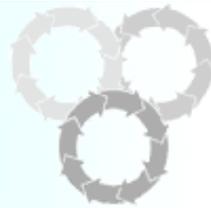
For general information, see organization page.

For web site information, contact Hayato Kanai

[h-kanai at jaist.ac.jp](mailto:h-kanai@jaist.ac.jp)

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Keynotes

Wednesday, 12 September Dr. Yoshihiro Matsumoto

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)



ASTEM Research Institute of Kyoto

Yoshihiro Matsumoto is Adviser of ASTEM Research Institute of Kyoto. He started his career in Toshiba Corporation in 1954, where he took initiative in the applications of software to real time control systems and in building Toshiba Software Factory for those domains. After he spent 35 years in Toshiba, he switched to academic field and served a professor at Kyoto University, Osaka Institute of Technology, and Musashi Institute of Technology. He received Dr. Eng. degree from the University of Tokyo, and Fellow in 1982/Life Fellow in 2004 both from the IEEE.

Experience of a Software Factory from Domain Preparation to Product Line Adoption,

日本語による情報
Information in Japanese

A software factory project for the software application to electric power generation (EPG) in Toshiba Software Factory started its domain preparation in the beginning of 1960, developed the first product line assets from 1971 to 1976, and began to adopt the product line assets to real projects in 1976. Afterwards, the EPG product line has overcome three big generation changes, and more than 150 large EPG systems have been produced under the product line adoption. In this talk, experience especially of domain engineering and asset development will be touched.

Yoshihiro Matsumoto: A Guide for Management and Financial Controls of product Lines, Proceedings SPLC2007, Kyoto, September 10-14 (2007)

Yoshihiro Matsumoto: Essence of Toshiba Software Factory,

Thursday, 13 September Dr.-Ing. Martin Verlage



Vice Executive Director vwd group Technology
vwd Vereinigte Wirtschaftsdienste GmbH
Frankfurt am Main, Germany

Dr.-Ing. Martin Verlage heads the software development and technical units of vwd group, a leading service provider for financial market data in Europe that focuses on solutions for retail banking, private banking and wealth management. Martin studied Computer Science at the Technical University of Kaiserslautern, Germany, and received a Diploma as well as a Ph.D. in Computer Science (Doktor der Ingenieurwissenschaften). For three years he worked as team leader of the Software Process Modelling Group of the Fraunhofer Institute of Experimental Software Engineering (IESE). In 1999 he joined market maker Software AG, a Kaiserslautern based software and service company, where he built up a new product line for intranet und internet applications. He was also responsible for research projects in software product lines. After six years he joined vwd Vereinigte Wirtschaftsdienste in Frankfurt, Germany, the major shareholder of market maker as Director Delivery Systems and Quality Management. In 2006 he became Vice Executive Director vwd group Technology, now being responsible for integrating vwd group's software development units and product platforms. Additionally, he acts as Director Technology of vwd's subsidiary FIDES Information Services, Zurich, Switzerland. Martin worked in international projects studying software product lines (e.g., ESAPS, CAFÉ, Families), published scientific conference and journal papers, continues to be member of program committees, and volunteers as reviewer for international journals.

**The Invisible Man-Month
or
What is the real value of a core asset?**

[keynote slides](#)

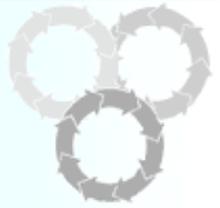
When motivating product line engineering, the usual suspects are lower cost, faster time-to-market, and better quality. Well-known to software engineers are the latter two, whereas development cost usually is a subject of discussion between project leaders, head of software development and upper management. And, of course, cost has to be less than the - expected - value of the software developed in a project.

This keynote pinpoints to economical phenomena when building up a product line, especially when existing software is to be replaced by flexible, configurable, and modular components. Software engineers should know at least a small portion of how software assets are represented in their company's balance sheet, because this may hinder or boost launching a new product line. Components represent value and hence building one single new component which replaces several existing ones destroys and creates value at the same time. But this is only the case, if the old

components were properly accounted. Is the new component 3rd party software or is it build by the company's software development unit? Was development effort recorded and accounted? What might look like prestidigitation is in fact a serious issue when management commitment is required to build new product lines. Building up or destroying values might influence the decision when a top-notch core asset is to be created.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Program

Home
Keynotes
Program
Tutorials
Workshops
Panels
Registration
Local Information

CFP
Submissions
Demonstration
Doctoral symposium
Product Line Hall of Fame

Organization
Program Committee
Sponsor

call-for-participation (pdf)
call-for-participation (txt)

日本語による情報
Information in Japanese

Monday, 10 September 2007			
09:00-12:30	WORKSHOPS W1: Open Source Software and Product Lines (OSSPL07 Asia) W2: Service-Oriented Architectures and Product Lines (SOAPL - 07) W3: (Canceled) W4: Dynamic Software Product Line (DSPL 07)	TUTORIALS T1. Introduction to Software Product Lines <i>Linda Northrop</i> T2. Domain-Specific Modeling and code generation for Product Lines <i>Juha-Pekka Tolvanen</i> T3. Using Feature Models for Product Derivation <i>Olaf Spinczyk, Holger Papajewski</i> T4. Building Reusable Testing Assets for a Software Product Line <i>John D. McGregor</i>	
13:30-17:00		TUTORIALS T5. Introduction to Software Product Adoption <i>Linda Northrop, Larry Jones</i> T6. Generative Software Development <i>Krzysztof Czarnecki</i> T7. The Scoping Game (Canceled) <i>Mark Dalgarno</i> T8. Predicting Product Line Payoff with SIMPLE <i>Paul Clements, John McGregor, Dirk Muthig</i>	
Tuesday, 11 September 2007			

09:00-12:30	WORKSHOPS W5: Managing Variability for Software Product Lines W6: 4th Software Product Line Testing Workshop (SPLIT2007) W7: Visualization in Software Product Line Engineering (ViSPLE 2007) W8: Product Lines in Practice - Taking Stock (PLIP07)	TUTORIALS T9. New Methods Behind the New Generation of Software Product Line Success Stories <i>Charles W. Krueger</i> T10. Variability Management for Product Lines with a Generative Technique <i>Stan Jarzabek</i> T11. Improving Product Line Development with the Families Evaluation Framework (FEF) <i>Klaus Schmid, Frank van der Linden</i>	
13:30-17:00		TUTORIALS T12. Transforming Legacy Systems into Software Product Lines <i>Danilo Beuche</i> T13. Production Planning in a Software Product Line Organization <i>Gary J. Chastek, John D. McGregor</i> T14. Leveraging Model Driven Engineering in Software Product Line Architectures <i>Bruce Trask, Angel Roman</i>	DOCTORAL SYMPOSIUM

Wednesday, 12 September 2007

	Conference Registration
08:30-09:00	OPENING REMARKS
09:00-10:15	KEYNOTE ADDRESS 1 Dr. Yoshihiro Matsumoto
10:15-10:40	Break

10:40-12:10	<p>T11A FEATURE MODELING moderator: <i>Jean-Marc Jezequel</i> Automating Mappings between Use Case Diagrams and Feature Models for Software Product Lines <i>Alexandre Braganca, Ricardo J. Machado</i> Reasoning about Feature Models in Higher-Order Logic <i>Mikolas Janota, Joseph Kiniry</i> Feature Diagrams and Logics: There and Back Again <i>Krzysztof Czarnecki, Andrzej Wasowski</i></p>	<p>T11B EXPERIENCE 1 moderator: <i>Kolb, Ronny</i> Experiences with Product Line Development of Multi-Discipline Analysis Software at Overwatch Textron Systems <i>Paul Jensen</i> Software Product Lines Approach in Enterprise System Development <i>Yuzo Ishida</i> Introducing Software Product Line Engineering for Metal Processing Lines in a Small to Medium Enterprise <i>David Sellier, Mike Mannion, Gorka Benguria, Gorka Urchegui</i></p>
12:10-13:30	Lunch	
13:30-15:00	<p>T12A COST/REQUIREMENT moderator: <i>Klaus Schmid</i> Impacts of Architecture and Quality Investment in Software Product Line Development <i>Makoto Nonaka, Liming Zhu, Muhammad Ali Babar, Mark Staples</i> Comparing Costs and Benefits of Different Test Strategies for a Software Product Line: A Study from Testo AG <i>Dharmalingam Ganesan, Jens Knodel, Ronny Kolb, Uwe Haury, Gerald Meier</i> Using Requirements Management Tools in Software Product Line Engineering: The State of the Practice <i>Danilo Beuche, Andreas Birk, Heinrich Dreier, Andreas Fleischmann, Isabel John, Ramin Tavakoli Kolagari, Heidi Galle, Gerald Gerald Heller, Dirk Janzen, Thomas von der Massen</i></p>	<p>T12B DEMO Kumbang Tools <i>Varvana Myllarniemi, Mikko Raatikainen, Tomi Mannisto</i> BigLever Software Gears and the 3-Tiered SPL Methodology <i>Charles W. Krueger</i> The Rhapsody/Gears Bridge – SPL for MDD <i>Martin Bakal, Charles W. Krueger</i></p>
15:00-15:30	Break	
15:30-17:00	<p>T13A PRODUCTION moderator: <i>Dirk Muthig</i> The 3-Tiered Methodology: Pragmatic Insights from New Generation Software Product Lines <i>Charles W Krueger</i> Development and Configuration of Service-based Product Lines <i>Alexander Gruler, Alexander Harhurin, Judith Hartmann</i> A Production System for Software</p>	<p>T13B DEMO nTools <i>Kazuyuki Natsume</i> Modeling and Building Software Product Lines with pure::variants and Eclipse <i>Danilo Beuche</i> MetaEdit+: Domain-Specific Modeling and Product Generation Environment <i>Juha-Pekka Tolvanen</i></p>

	Product Lines <i>Gary J. Chastek, Patrick Donohoe, John D. McGregor</i>	
	Reception	
Thursday, 13 September 2007		
	Conference Registration	
09:00-10:15	KEYNOTE ADDRESS 2 Dr. Martin Verlage	
10:15-10:40	Break	
10:40-12:10	T21A VARIABILITY MANAGEMENT moderator: <i>Klaus Schmid</i> Automating Product-Line Variant Selection for Mobile Devices <i>Jules White, Douglas C. Schmidt, Egon Wuchner, Andrey Nechypurenko</i> Supporting Product Derivation by Adapting and Augmenting Variability Models <i>Rick Rabiser, Paul Gruenbacher, Deepak Dhungana</i> Optimization of Variability in Software Product Lines <i>Felix Loesch</i>	T21B EXPERIENCE 2 moderator: <i>David Weiss</i> A Guide for Management and Financial Controls of Product Lines <i>Yoshihiro Matsumoto</i> Building Software Product Line from the Legacy Systems <i>Kangtae Kim, Hyungrok Kim, Woomok Kim</i> A Component Model supporting Decomposition and Composition of Consumer Electronics Software Product Lines <i>Chong-Mok Park, Seokjin Hong, Kyoung-Ho Son, Jagun Kwon</i>
12:10-13:30	Lunch	
13:30-15:00	T22A EXPERIENCE 3 moderator: <i>Kentaro Yoshimura</i> Challenges of Establishing a Software Product Line for an Aerospace Engine Monitoring System <i>Ibrahim M Habli, Tim P Kelly, Ian Hopkins</i> Minimally Invasive Migration to Software Product Lines <i>Hans Peter Jepsen, Jan Gaardsted, Danilo Beuche</i> Dynamic Complexity and the Owen Firmware Product Line Program <i>Holt Mebane, Joni T. Ohta</i>	T22B DEMO Design Verification Tool for Product Line Development <i>Natsuko Noda</i> CIDE: Decomposing Legacy Applications into Features <i>Christian Kastner</i> DOPLER: An Adaptable Tool Suite for Product Line Engineering <i>D. Dhungana R. Rabiser P. Grünbacher K. Lehner C. Federspiel</i>
15:00-15:30	Break	

15:30-17:00	<p>T23A ASPECT-ORIENTATION/MDA moderator: <i>Frank van der Linden</i> A Case Study Implementing Features using AspectJ <i>Christian Kraestner, Sven Apel, Don Batory</i> Product Line Implementation using Aspect-Oriented and Model-Driven Software Development <i>Markus Voelter, Iris Groher</i> Higher-Order Transformations for Product Lines <i>Jon Oldevik, Oystein Haugen</i></p>	<p>T23B PANEL SW Product Line Evolution and Life Cycle</p>
	Banquet	
Friday, 14 September 2007		
	Conference Registration	
09:00-10:30	<p>PANEL Considerations of Long Term Product Support on Software Product Lines</p>	
10:30-10:50		
10:50-11:35	<p>T31A SHORTEPAPER 1 moderator: <i>Natsuko Noda</i> Tailoring Infrastructure Software Product Lines by Static Application Analysis <i>Horst Schirmeier, Olaf Spinczyk</i> A Variability Modeling Method for Adaptable Services in Service-Oriented Computing <i>Soo Ho Chang, Soo Dong Kim</i></p>	<p>T31B SHORTEPAPER 2 moderator: <i>Jaejoon Lee</i> Why does it take that long? Establishing Product Lines in the Automotive Domain <i>Christian Tischer, Andreas Mueller, Markus Ketterer, Lars Geyer</i> Software Product Line Impacts Beyond Software Development <i>Scott G Decker, James C Dager</i></p>
11:35-11:45	Break	
11:45-13:00	HALL OF FAME CLOSING	
13:00-14:00	Lunch	

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Tutorials

The following tutorials will be held in conjunction with SPLC 2007:

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報

Information in Japanese

Monday, September 10th, Morning

T1. [Introduction to Software Product Lines](#)

Linda Northrop

T2. [Domain-Specific Modeling and code generation for Product Lines](#)

Juha-Pekka Tolvanen

T3. [Using Feature Models for Product Derivation](#)

Olaf Spinczyk, Holger Papajewski

T4. [Building Reusable Testing Assets for a Software Product Line](#)

John D. McGregor

Monday, September 10th, Afternoon

T5. [Introduction to Software Product Adoption](#)

Linda Northrop, Larry Jones

T6. [Generative Software Development](#)

Krzysztof Czarnecki

T7. The Scoping Game (Canceled)

Mark Dalgarno

T8. [Predicting Product Line Payoff with SIMPLE](#)

Paul Clements , John McGregor , Dirk Muthig

Tuesday, September 11th, Morning

T9. [New Methods Behind the New Generation of Software Product Line Success Stories](#)

Charles W. Krueger

T10. [Variability Management for Product Lines with a Generative Technique](#)

Stan Jarzabek

T11. [Improving Product Line Development with the Families Evaluation Framework \(FEF\)](#)

Klaus Schmid, Frank van der Linden

Tuesday, September 11th, Afternoon

T12. [Transforming Legacy Systems into Software Product Lines](#)

Danilo Beuche

T13. [Production Planning in a Software Product Line Organization](#)

Gary J. Chastek , John D. McGregor

T14. [Leveraging Model Driven Engineering in Software Product Line Architectures](#)

[With specific examples using the Eclipse Frameworks EMF, GEF, & GMF](#)

Bruce Trask, Angel Roman

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Workshops

The purpose of the workshop program is to provide a platform for bringing together people from industry, academia, and research institutions to present and discuss recent proposals and practices around software product line development.

The following workshops will be held in conjunction with SPLC 2007:

- [International Workshop on Dynamic Software Product Line \(DSPL 07\)](#)
- [Managing Variability for Software Product Lines](#)
- [Open Source and Product Lines 2007, Asia OSSPL07 Asia](#)
- [Service Oriented Architectures and Product Lines](#)
- [What is the Connection? \(SOAPL - 07\)](#)
- [The 4th Software Product Lines Testing \(SPLiT2007\) Workshop](#)
- [Value-Based Product Line Engineering \(VBPL'07\) \(Canceled\)](#)
- [Workshop on Visualisation in Software Product Line Engineering \(ViSPLE 2007\)](#)
- [Product Lines in Practice - Taking Stock\(PLiP07\) \(Canceled\)](#)

日本語による情報
Information in Japanese



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

Panels

Panel 1: SW Product Line Evolution and Life Cycle (Thursday 15:30-17:00)

Abstract: There are many forces which can impact a software product line. These forces can cause the product line to evolve the existing assets, force a new product line, cause a split in the product line, or a merger of two product lines. Determining what direction to take with the product line life cycle has costs and benefits associated with it. Panelist are asked to draw a picture of the product line life cycle showing the various decision points expressed in the questions and give the criteria used for the various choices.

Overview: This panel addresses questions regarding software product line evolution and life cycle. The panel is to discuss and address some of the following issues and questions:

- At what point do you consider adopting a new product line?
- Are paradigms such as Aspect Oriented Programming or Model-Driven Software Development catalysts for a new product line?
- Should the adoption of a new software development paradigm cause a product line change?
- What criteria are used for allowing localized rearchitecting of the product line?
- At what point does enough localized rearchitecting cause it to be considered a new product line?
- When should you adapt a product line versus starting new?
- When should a product line be split into two?
- How do we merge multiple product lines?
- Can product line degradation during support of sold products be avoided or is it just part of the product line life cycle?
- What happens if a company merger causes two similar product lines to exist within the new company?
- What are the financial impacts of these various decisions?

日本語による情報
Information in Japanese

Panel 2: Considerations of Long Term Product Support on Software Product Lines (Friday 9:00-10:30)

Abstract: This panel looks at the impacts that product lines have on the support of products once they are in production. Depending on the industry, products in production may need to be enhanced and bugs fixed for 5, 10, even 30 years after first being released.

Overview: This panel is to address some of the following issues and questions:

- How does organizational structure impact the long term support of the product line? What organizational structures have worked well and which ones have not?
- After a product is launched into production, how does a company ensure that the original product line is not degrading by making fixes or enhancements? Why do we care if it does?
- What product maintenance processes have been useful in preventing product line degradation while supporting sold products?
- How do future and current product organizations work together in a product line environment?
- Can product line degradation during support of sold products be avoided or is it just part of the product line life cycle?
- How does the life expectancy of a product line affect your product support approach?

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Registration

The followings are an overview of registration fee and accommodations.

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

IMPORTANT:

Deadline for Early Registration: August 24th, 2007, 23:59 JST (GMT+9)

Deadline for Registration: September 5th, 2007, 23:59 JST (GMT+9)

REGISTRATION FEE:

MAIN CONFERENCE

	EARLY REGISTRATION	LATE REGISTRATION
Standard	JPY 52,000	JPY 60,000
Student	JPY 32,000	JPY 40,000

Fees per person include for the whole conference: lunches, coffee breaks, reception, banquet and proceedings

TUTORIALS

	EARLY REGISTRATION	LATE REGISTRATION
1 Tutorial	JPY 11,000	JPY 13,000

Fees per person for one tutorial include: coffee break and tutorial documents (the lunch is NOT included)

WORKSHOPS

	EARLY REGISTRATION	LATE REGISTRATION
1 Workshop	JPY 10,000	JPY 13,000

Fees per person include for the whole day: lunch and coffee breaks

BANQUET

Extra Ticket
JPY 12,000

ACCOMMODATION:

A block of rooms in the following hotels has been reserved for this conference;

HOTEL GRANVIA KYOTO

NEW MIYAKO HOTEL

RIHGA ROYAL HOTEL KYOTO

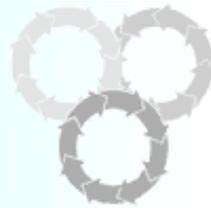
KYOTO DAI-NI TOWER HOTEL

HOTEL HOKKE CLUB KYOTO

日本語による情報
Information in Japanese

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Local Information

Conference venue

Kyoto Research Park (KRP), Kyoto, Japan

for more information on KRP, please visit <http://www.krp.co.jp/english/index.html>

Access: <http://www.krp.co.jp/english/access/index.html>

[HYPERDIA-timetable](#)

Please specify "Kansai Airport" "Kyoto" "Tambaguchi" for Start Point or Destination.

[Google Maps for SPLC 2007](#)

[Public transport in Kyoto](#)

Accommodation hints

- You can reserve the following hotels at conference rates via the registration site

[HOTEL GRANVIA KYOTO](#)

[NEW MIYAKO HOTEL](#)

[RIHGA ROYAL HOTEL KYOTO](#)

[KYOTO DAI-NI TOWER HOTEL](#)

[HOTEL HOKKE CLUB KYOTO](#)

- If you want to stay in a Japanese-style hotel ("Ryokan"),
you can find information [here](#).

Please note that each Ryokan only has a small number of rooms.

We recommend you make an arrangement as early as possible.

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報

Information in Japanese

Welcome to Kyoto

Kyoto was Japan's capital and the emperor's residence from 794 until 1868. It is now the country's seventh largest city with a population of 1.4 million people and a modern face.

Over the centuries, Kyoto was destroyed by many wars and fires, but due to its historic value, the city was not chosen as a target of air raids during World War II. Countless temples, shrines and other historically priceless structures survive in the city today.

Tourist Attractions

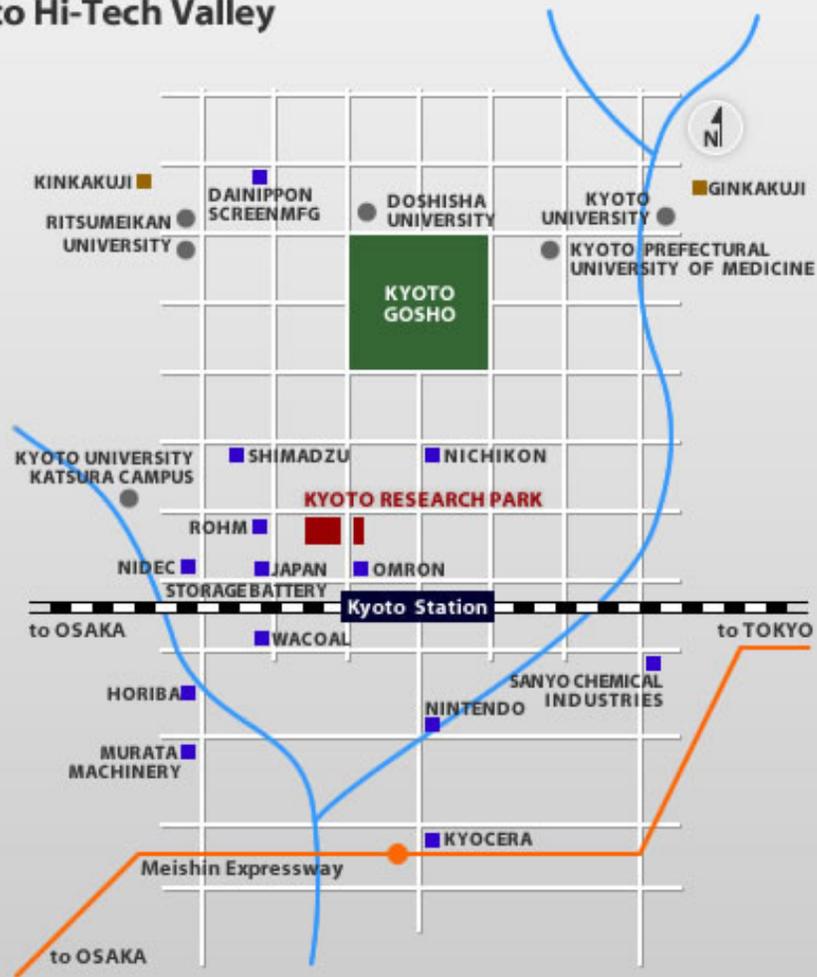
- Kinkakuji (Golden Pavilion) and Ginkakuji (Silver Pavilion)
- Toei Uzumasa Eigamura (aka Toei Movie Land)
- Kyoto Gosho (Kyoto Imperial Palace)
- Kiyomizudera (Pure Water Temple)
- Nishi Honganji
- Gojū-no-tō (Toji Temple)
- Sanjusangendo
- Heian Shrine
- Ninomaru

[Kyoto City Tourism and Culture Information System](#)



Map of Kyoto

Kyoto Hi-Tech Valley



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



CFP

Submission Deadlines

Paper submission (firm!) Feb 28, 2007

Author notification ~~Apr 19, 2007~~

Camera ready paper deadline (firm!) ~~Apr 30, 2007~~

Author notification on May 5th

Camera ready deadline is scheduled on **middle of June**.

(exact date will be notified from publisher to authors)

With SPLC 2007, *the* premier forum for practitioners, researchers and educators to present and discuss the most recent ideas, innovations, trends, experiences, and concerns in the area software product lines and software product family engineering comes to Asia for the first time .

The objective is to continue the dialogue between software product line practitioners and researchers on the benefits, obstacles, and weaknesses of applying software product line principles, techniques, methods, processes, and tools in an industrial or organizational setting.

Scope

Topics of interest of SPLC 2007 include, but are definitely not restricted to:

- Industrial experiences in product line engineering
- Techniques and tools for product line engineering
- Evolution of product line assets
- Business issues for product lines
- Organizational and process issues for product lines
- Product line life-cycle issues

We invite three classes of original, unpublished submissions:

- RESEARCH PAPERS describe original research contribution (theoretical, conceptual) to the field of software product line engineering. A research paper should clearly describe the problem that has been tackled, the state of the art with respect to the problem, the solution that is suggested and the potential -- or even better the evaluated -- benefits of the contribution.

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報
Information in Japanese

We also call for short research papers, which are intended to report ideas in their early stages, and are not required to show complete/evaluated result.

- EXPERIENCE PAPERS describe the history of a software product line accompanied by a critical review of experience within one or more development phases within family and/or application engineering. An experience report describes lessons learned in domains of specific industries like automobile, mobile communication, embedded controller, or enterprise SW.

We also call for short experience papers: Short experience papers are intended to report their useful and informative experiences briefly, and highlight their main interesting points.

- POSTERS describes research result or on-going research projects. They are displayed in a dedicated poster area at the conference, and presented in poster sessions.
Poster papers will appear in a separate proceedings.

Review Process

At least three members of the SPLC 2007 program committee will review each submission. The reviews will be the basis for making final decisions about which submissions to accept for presentation at the conference.

Accepted Submissions

Each accepted paper submission will be allotted a maximum of ten pages (full papers) or six pages (short papers) in the conference proceedings. The final version of accepted papers must conform to the proceedings publication format. Authors are required to present their work in a technical session.

Each accepted poster submission will allotted two pages in the separate proceedings. Poster format will be announced to accepted authors. Authors are required to present their work in poster sessions.

PC members will vote for the best paper from full papers, but no paper may be selected if there is no agreement on the best paper.

Submission Guidelines

Papers should be submitted in PDF format. The results described must be unpublished and should not be under review elsewhere.

Submitted papers must conform to the IEEE proceeding 8.5x11-inche, Two-Column Format (*), and should not exceed ten pages (including all text, figures, references and appendices).

(*) <http://www.computer.org/portal/site/cscps/index.jsp> INFORMATION FOR AUTHORS -> Formatting

Short research/experience papers are also conform the same format, and should not exceed six pages.

Poster papers are also conform the same format, and should not exceed two pages.

The submission deadline is firm! If you are unable to submit a paper electronically, please contact one of the two chairs at least two weeks prior to the submission deadline.

Contact

For more information about submissions, feel free to contact Tomoji Kishi (tkishi at jaist.ac.jp) and Dirk Muthig (dirk.muthig at iese.fraunhofer.de).

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Demonstration

Call for demos:

We seek proposals for demonstration of tools related to the many aspects of the Product Line approach such as analysis (modeling), design of Product line architectures, derivation of Product Line members, handling variability, testing, documentation, etc. Authors of accepted demos will have 2-page papers published in conference proceedings.

Please e-mail your proposal in free format, maximum 2 pages to:

Stan Jarzabek ([stan at comp.nus.edu.sg](mailto:stan@comp.nus.edu.sg)) and Charles Krueger

([ckrueger at biglever.com](mailto:ckrueger@biglever.com))

Important Dates:

Submit your proposal by May 1, 2007

Notification by June 1, 2007

Send 2 page camera-ready paper describing your demo for publication in proceedings: June 15, 2007.

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Fame

Organization

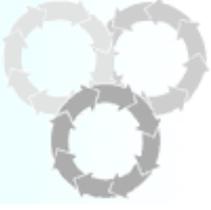
Program Committee

Sponsor

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報
Information in Japanese



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

PRODUCT LINE HALL OF FAME

A hall of fame serves as a way to recognize distinguished members of a community in a field of endeavor. Those elected to membership in a hall of fame represent the highest achievement in their field, serving as models of what can be achieved and how. Each Software Product Line Conference culminates with a session in which members of the audience nominate systems for induction into the Software Product Line Hall of Fame. These nominations feed discussions about what constitutes excellence and success in product lines. The goal is to improve software product line practice by identifying the best examples in the field. Nominations are acted on by a panel of expert judges, who decide which nominees will be inducted into the Hall of Fame.

You can read about the current members of the Software Product Line Hall of Fame at http://www.sei.cmu.edu/productlines/plp_hof.html. Inductees from 2006 will be announced at the SPLC 2007 Hall of Fame session.

Criteria for Election to the Software Product Line Hall of Fame

Members of the software product line hall of fame should serve as models of what a software product line should be, exhibiting most or all of the following characteristics:

The family that constitutes the product line is clearly identified, i.e., there is a way to tell whether or not a software system is a member of the product line, either by applying a known rule or a known enumeration.

The family that constitutes the product line is explicitly defined and designed as a product line, i.e., the commonalities and variabilities that characterize the members of the product line are known and there is an underlying design for the product line that takes advantage of them.

The product line has had a strong influence on others who desire to build and evolve product lines, and has gained recognition as a model of what a product line should be and how it should be built. Others have borrowed, copied, and stolen from it in creating their product lines or in expounding ideas and practices for creating product lines.

The product line has been commercially successful.

There is sufficient documentation about the product line that one can understand its definition, design, and implementation without resorting solely to hearsay.

Contact:

David Weiss ([weiss at avaya.com](mailto:weiss@avaya.com)), Avaya Labs Research

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報
Information in Japanese

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Organization

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報

Information in Japanese

Organizers:

- **General Chair**
Kyo Kang, POSTECH, Korea
[kck at postech.ac.kr](mailto:kck@postech.ac.kr)
- **Program Co-chairs**
Tomoji Kishi, JAIST, Japan
[tkishi at jaist.ac.jp](mailto:tkishi@jaist.ac.jp)
Dirk Muthig, IESE, Fraunhofer, Germany
[muthig at iese.fraunhofer.de](mailto:muthig@iese.fraunhofer.de)
- **Workshop Chair**
Peter Knauber, Mannheim University of Applied Science, Germany
[p.knauber at hs-mannheim.de](mailto:p.knauber@hs-mannheim.de)
- **Tutorial Chair**
Charles Krueger, BigLever Software Inc., USA
[ckrueger at biglever.com](mailto:ckrueger@biglever.com)
- **Panel Chair**
Jim Dager, Cummins Inc., USA
[jim.c.dager at cummins.com](mailto:jim.c.dager@cummins.com)
- **Demonstration Co-Chairs**
Stan Jarzabek, National University of Singapore, Singapore
[stan at comp.nus.edu.sg](mailto:stan@comp.nus.edu.sg)
Charles Krueger, BigLever Software Inc., USA
[ckrueger at biglever.com](mailto:ckrueger@biglever.com)
- **Doctoral symposium Chair**
Jaejoon Lee, IESE, Fraunhofer, Germany
[Jaejoon.Lee at iese.fraunhofer.de](mailto:Jaejoon.Lee@iese.fraunhofer.de)
- **Product Line Hall of Fame**
David Weiss, Avaya Labs, USA
[weiss at avaya.com](mailto:weiss@avaya.com)
- **Local Arrangement Chair**
Atsushi Sawada, Kyoto University, Japan
[sawada at media.kyoto-u.ac.jp](mailto:sawada@media.kyoto-u.ac.jp)
- **Financial Chair**
Natsuko Noda, NEC/JAIST, Japan
[nnoda at jaist.ac.jp](mailto:nnoda@jaist.ac.jp)
- **Industrial Relation Chair**
Kazuo Iwano, IBM Japan, Japan
[iwanoalt at jp.ibm.com](mailto:iwanoalt@jp.ibm.com)
- **Publicity**
USA
John McGregor, Clemson University, USA
[johnmc at cs.clemson.edu](mailto:johnmc@cs.clemson.edu)
Asia
Yoshikazu Hayashi, SRA-KTL, Inc., Japan
[yosikazu at sra.co.jp](mailto:yosikazu@sra.co.jp)
Europe
Klaus Pohl, Lero, University of Limerick, Ireland and
SSE, University of Duisburg-Essen, Germany
[pohl at sse.uni.due.de](mailto:pohl@sse.uni.due.de)

11th International

Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Program Committee

Home	Gary Chastek, SEI, USA Paul Clements, SEI, USA
Keynotes	Sholom Cohen, SEI, USA
Program	Krzysztof Czarnecki, University of Waterloo, Canada
Tutorials	Stefania Gnesi, ISTI-CNR, Italy
Workshops	Hassan Gomaa, George Mason University, USA
Panels	Masayuki Hirayama, IPA/SEC, Japan
Registration	Stan Jarzabek, National University of Singapore, Singapore
Local Information	Jean-Marc Jezequel, IRSIA, France
CFP	Isabel John, Fraunhofer IESE, Germany
Submissions	Michael Kircher, Siemens, Germany
Demonstration	Charles W. Krueger, Biglever, USA
Doctoral symposium	Kwanwoo Lee, Hansung University, South Korea
Product Line Hall of Fame	John D. McGregor, Clemson University, USA
Organization	Natsuko, Noda, NEC Corp. / JAIST, Japan.
Program Committee	Sooyong Park, SOGANG University, South Korea
Sponsor	Klaus Pohl, Lero, University of Limerick, Ireland and SSE, University of Duisburg-Essen, Germany
call-for-participation (pdf)	Klaus Schmid, University of Hildesheim, Germany
call-for-participation (txt)	Andr van der Hoek, University of California, USA Frank van der Linden, Philips, The Netherlands Rob van Ommering, Philips, The Netherlands Karl Reed, La Trobe University, Australia Martin Verlage, Vereinigte Wirtschaftsdienste, Germany David M. Weiss, Avaya, USA Kentaro Yoshimura, Hitachi, Japan

日本語による情報

Information in Japanese

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Sponsor

Sponsors



Carnegie Mellon
Software Engineering Institute



Information-technology Promotion Agency, Japan

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Corporate Sponsors

Sponsors

[日本語による情報](#)

[Information in Japanese](#)

HITACHI

[Hitachi, Ltd.](#)



[MITSUBISHI ELECTRIC CORPORATION](#)

NEC

NEC Corporation

NRI Dream up the future.
**Nomura
Research Institute**

Nomura Research Institute, Ltd.

 **pure-systems**

pure-systems

RICOH

RICOH COMPANY, LTD

TOSHIBA

Leading Innovation >>>

TOSHIBA CORPORATION

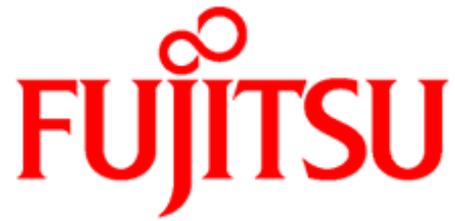
Supporters



BigLever Software



CATS CO.,LTD.



FUJITSU LIMITED



IBM



OMRON Corporation



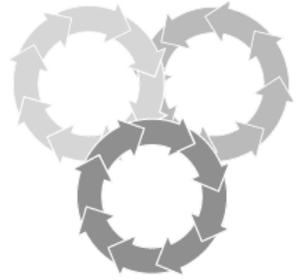
TOSHIBA SOLUTIONS CORPORATION

TOSHIBA SOLUTIONS CORPORATION

11th International
Software Product Line Conference
(SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

<http://sec.ipa.go.jp/SPLC2007/>



*"Software Product Lines –
Power for Competitive Advantage"*



The current trend of globalization is pressuring industries to explore ways to meet diverse needs of the global market most effectively and efficiently. Over the last decade or so, software product line has emerged as one of the most promising software development paradigms in drastically increasing the productivity of IT-related industries, and the product line community has grown and is still growing rapidly. At this very important juncture, the Software Engineering Center (SEC) of Information-Technology Promotion Agency (IPA), Japan is proud to sponsor the Eleventh International Software Product Line Conference (SPLC 2007) in Kyoto, Japan, the first ever software product line conference held in Asia.

SPLC is the most prestigious and leading forum for researchers, practitioners, and educators in the field. SPLC 2007 will provide a venue for exchanging, sharing, and learning technologies and industrial experiences to the community. The conference will feature research and experience papers, tutorials, workshops, panels, demonstrations, doctoral symposium, and Product Line Hall of Fame.

We look forward to interacting with you at SPLC 2007!

• Important dates:

- Jun 15, 2007 Early registration starts.
- Aug 17, 2007 Late registration starts.

For detailed information on programs and fees, please visit our conference home page.

• Conference venue:

Kyoto Research Park (KRP), Kyoto, Japan

For more information on KRP, please visit <http://www.krp.co.jp/english/index.html>

Sponsors



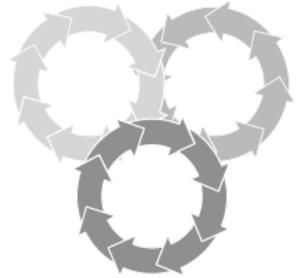
Information-technology Promotion Agency, Japan





11th International
Software Product Line Conference
(SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan
<http://sec.ipa.go.jp/SPLC2007/>



Organization

General Chair

Kyo C. Kang (kck@postech.ac.kr)
POSTECH, South Korea

Program Co-chairs

Tomoji Kishi (tkishi@jaist.ac.jp)
JAIST, Japan

Dirk Muthig (dirk.muthig@iese.fraunhofer.de)
IESE, Fraunhofer, Germany

Workshop Chair

Peter Knauber (p.knauber@hs-mannheim.de)
Mannheim University of Applied Science, Germany

Tutorial Chair

Charles Krueger (ckrueger@biglever.com)
BigLever Software Inc., USA

Panel Chair

Jim Dager (jim.c.dager@cummins.com)
Cummins Inc., USA

Demonstration Chair

Stan Jarzabek (stan@comp.nus.edu.sg)
National University of Singapore, Singapore

Doctorial Symposium Chair

Jaejoon Lee (Jaejoon.Lee@iese.fraunhofer.de)
IESE, Fraunhofer, Germany

Product Line Hall of Fame

David Weiss (weiss@avaya.com)
Avaya Labs, USA

Local Arrangement Chair

Atsushi Sawada (sawada@media.kyoto-u.ac.jp)
Kyoto University, Japan

Financial Chair

Natsuko Noda (n-noda@cw.jp.nec.com)
NEC/JAIST, Japan

Publicity Chairs

USA: John McGregor (johnmc@cs.clemson.edu)
Clemson University, USA

Asia: Yoshikazu Hayashi (yosikazu@sra.co.jp)
SRA-KTL, Inc., Japan

Europe: Klaus Pohl (pohl@lero.ie)

*Lero, University of Limerick, Ireland
and SSE, University of Duisburg-Essen, Germany*

Program Committee

Gary Chastek (gjc@sei.cmu.edu)
SEI, USA

Paul Clements (clements@sei.cmu.edu)
SEI, USA

Sholom Cohen (sgc@sei.cmu.edu)
SEI, USA

Krzysztof Czarnecki (czarnecki@acm.org)
University of Waterloo, Canada

Stefania Gnesi (stefania.gnesi@isti.cnr.it)
ISTI-CNR, Italy

Hassan Gomaa (hgomaa@gmu.edu)
George Mason University, USA

Masayuki Hirayama (m-hiraya@ipa.go.jp)
IPA/SEC, Japan

Stan Jarzabek (stan@comp.nus.edu.sg)
National University of Singapore, Singapore

Jean-Marc Jezequel (Jean-Marc.Jezequel@irisa.fr)
IRISA, France

Isabel John (john@iese.fhg.de)
IESE, Fraunhofer, Germany

Michael Kircher (Michael.Kircher@siemens.com)
Siemens, Germany

Charles W. Krueger (ckrueger@biglever.com)
BigLever, USA

Kwanwoo Lee (kwlee@hansung.ac.kr)
Hansung University, South Korea

John D. McGregor (johnmc@cs.clemson.edu)
Clemson University, USA

Natsuko Noda (n-noda@cw.jp.nec.com)
NEC Corp. / JAIST, Japan

Sooyong Park (syPark@sogang.ac.kr)
SOGANG University, South Korea

Klaus Pohl (pohl@lero.ie)
*Lero, University of Limerick, Ireland
and SSE, University of Duisburg-Essen, Germany*

Klaus Schmid (schmid@sse.uni-hildesheim.de)
University of Hildesheim, Germany

Andre van der Hoek (andre@ics.uci.edu)
University of California, USA

Frank van der Linden (Frank.van.der.Linden@philips.com)
Philips, The Netherlands

Rob van Ommering (Rob.van.Ommering@philips.com)
Philips, The Netherlands

Karl Reed (k.reed@latrobe.edu.au)
La Trobe University, Australia

Martin Verlage (mverlage@vwd.de)
Vereinigte Wirtschaftsdienste, Germany

David M. Weiss (weiss@avaya.com)
Avaya, USA

Kentaro Yoshimura (kentaro.yoshimura@hitachi-eu.com)
Hitachi Europe, France

CALL FOR PARTICIPATION

11th International
Software Product Line Conference

S P L C 2 0 0 7

Kyoto, Japan 10 - 14 September 2007

<http://sec.ipa.go.jp/SPLC2007/>

<http://www.splc.net/>

•O Software Product Lines ? Power for Competitive Advantage •P

The current trend of globalization is pressuring industries to explore ways to meet diverse needs of the global market most effectively and efficiently. Over the last decade or so, software product line has emerged as one of the most promising software development paradigms in drastically increasing the productivity of IT-related industries, and the product line community has grown and is still growing rapidly. At this very important juncture, the Software Engineering Center (SEC) of Information-Technology Promotion Agency (IPA), Japan is proud to sponsor the Eleventh International Software Product Line Conference (SPLC 2007) in Kyoto, Japan, the first ever software product line conference held in Asia.

SPLC is the most prestigious and leading forum for researchers, practitioners, and educators in the field. SPLC 2007 will provide a venue for exchanging, sharing, and learning technologies and industrial experiences to the community. The conference will feature research and experience papers, tutorials, workshops, panels, demonstrations, doctoral symposium, and Product Line Hall of Fame.

We look forward to interacting with you at SPLC 2007!

- Important dates:

- Jun 15, 2007 Early registration started.
- Aug 24, 2007 Late registration starts.

For detailed information on programs and fees, please visit our
conference home page.

- Conference venue:

Kyoto Research Park (KRP), Kyoto, Japan

For more information on KRP,
please visit <http://www.krp.co.jp/english/index.html>



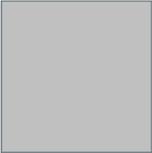
The Invisible Man-Month or What is the Real Value of a Core Asset?

Keynote
11th Software Product Line Conference
SPLC 2007
Kyoto, Japan

Martin Verlage
Vice Executive Director vwd group Technology

September 13, 2007

All Rights Reserved © Martin Verlage, 2007



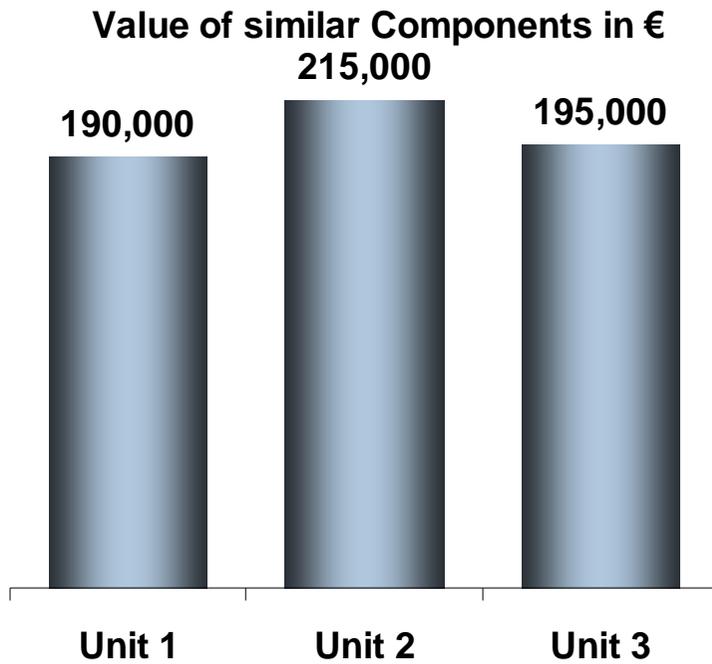
Contents

- **A Small Product Line Engineering Assignment**
- **Background**
- **Motivation**
- **Part I – The Value of Software**
- **Part II – The Invisible Man Month**
- **Concluding Remarks**

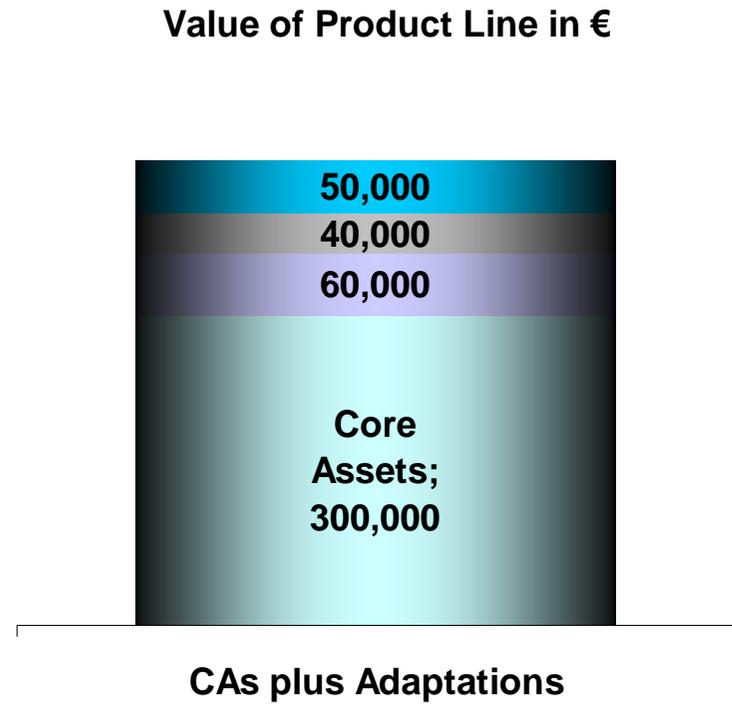
Acknowledgement

I would like to thank Thomas Kiesgen and Karsten Schudt for inspiring discussions and their valuable comments on earlier versions of these slides.

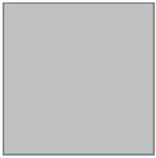
Question



Sum: 600,000



Sum: 450,000

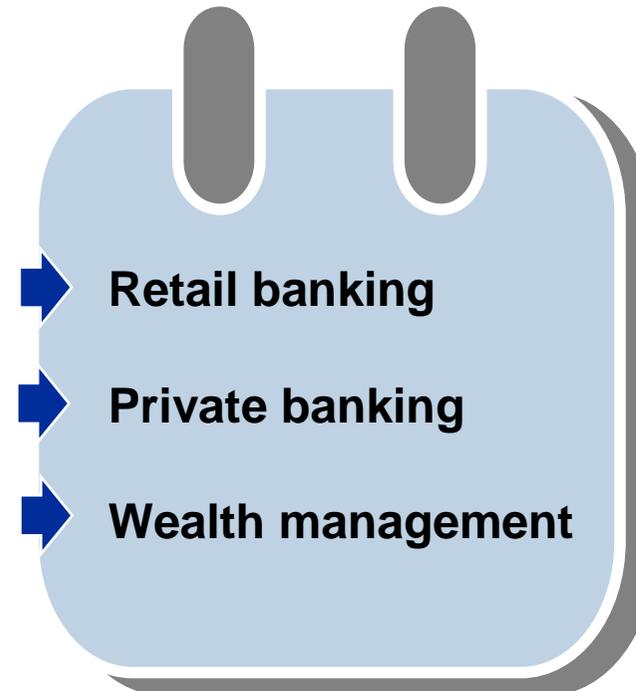


Background

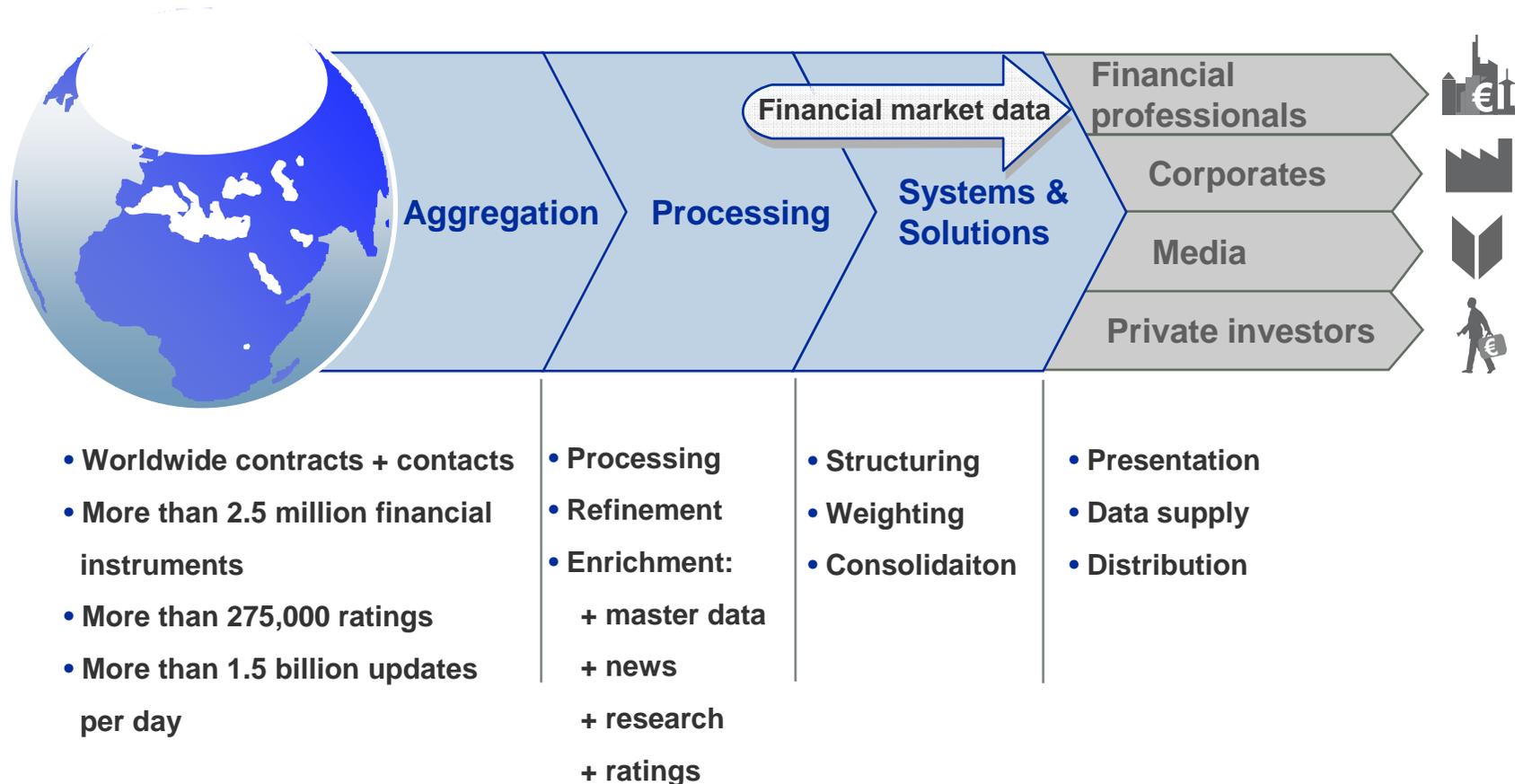
vwd group - Mission

The vwd group is a leading service provider for financial information in Europe.

The company aggregates, processes and disseminates global financial market data for retail banking, private banking and wealth management.



vwd group - Company Objective



vwd group - Facts and Figures

Foundation of the
parent company:

1949

Total turnover 2006:

EUR 54.9 million

Employees 2007:

330

Company Structure:

vwd Vereinigte Wirtschaftsdienste AG

ISIN DE 000 520470 5

Subsidiaries

Gatrixx NetSolutions AG (100%)

GeVaSyS GmbH (51%)

vwd TransactionSolutions AG (60%)

market maker Software AG (57.33%)

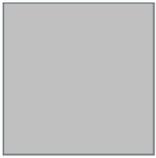
incl. Lenz + Partner AG (51.29%)

FIDES Information Services AG (100%)



Corporate Actions:

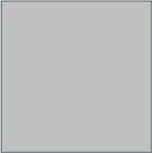
Downstream merger with b.i.s. AG in summer 2007



Motivation

Why PLEs Should Care About Accounting

- **Software Engineers focus on technical issues (big surprise ☺)**
- **Software Engineering often is about “big things”**
 - Changing the way software is developed
 - Investments for training, tools, and processes
- **Product Line Engineering (PLE) is even about “bigger than big things”**
 - Not a single process (e.g., testing), but significant parts of the organization are transformed (e.g., separation into application engineering and domain engineering)
 - Drastically change the way how an organization develops software
 - “Guerilla tactics” isn’t applicable, often the term “big bang” is used
- **Product Line Engineering is considered to affect the wealth state of a software developing company; PLE is motivated by economic effects**
- **The wealth of a company is not under control by software engineers**
 - Management Commitment
 - Budget (money and people)
 - Priority (time and people)
 - Understand the decision making processes and the factors that are relevant to decisions about Product Line Engineering



Expectations

- **This is not a crash course in accounting**
- **This is not a lecture about standards and standard setting bodies**

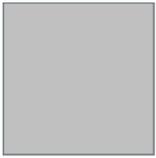
- **But**
 - Some insight into how accounting represents the value of a piece of software.
 - What are the effects to the balance sheet if product lines are introduced?
 - Why is it so difficult to get a clear answer?

- **Goal of this talk**
 - Help to establish understanding about some non-technical issues in introducing a product line.
 - Point to selected economic pitfalls when being concerned with introduction of a product line.

The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**



The Value of Software

What is the Value of a Core Asset?

Alternatives of defining the term “value”

- Value = discounted accumulated revenues [cash view]
- Value = discounted accumulated (future) earnings [external economic view]
- Value = ability to save/reduce cost [internal economic / controlling view]
- Value = accumulated cost [budget view]
- Value = fitness to solve problems [customer’s view]
- Value = readiness for future business [strategic view]
- Value = pride [emotional view]
- Value = market capitalization (includes expectations) [shareholder’s view]

Product Line evangelists usually stress a mixture of the above definitions

- “faster time-to-market”
 - Bigger market share and higher prices result in higher revenues
 - Competitive advantage by reacting faster to changing markets
- “the break even is with instance #4, from there on, development of a new instance is cheaper than with traditional development”

How to Measure Value?

It looks like “price” and “market value” aren’t good ideas for defining the value of a product line!

- **Wikipedia (Sep 8, 2007):**
Value (economics) -- In general, the economic value of something is how much a product or service is worth to someone relative to other things (often measured in money)
...
It can be either an assessment of what it could or should be the price (valuation), or an explanation of its actual market value (price).
- **Is there a market for product line software?**
 - PL seldomly aren’t sold “as is”
 - A PL is intentionally not a product, but part of multiple products
 - In larger organisations there could be an internal market for using PLs
 - Often the decision is “make or buy” instead of choosing between alternative PLs
- **How are the revenues scattered across the PL features?**
 - What is the relative value of product line adoptions (i.e., which percentage of the revenues really is going to PL)?
 - What percentage of a generic feature is used?
 - In niche markets or when testing new markets the entry just can be done by the presence of a generic PL which lowers risk and investment

Cost of Software

Software Development Phase	Stage
Problem Definition	Preliminary Project Stage
Requirements Analysis	
Specification	
Design	Application Development Stage
Coding	
Test	
Training	Post Implementation Stage
Maintenance	

- **Sample list of costs**

- Salaries
- Communication
- (Avoidance of) Risk
- Travel Expenses
- Project Marketing
- Licenses
- Floor space
- Overhead
- ...

- **Alignment to phases**

- **Hypothesis**

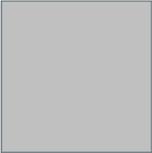
$$\text{value of product line} = \text{sum of all costs}$$

- **Representation in the corporate balance sheet**
- **Value is not for eternity**

The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**



Terms (1)

- **Amortization** refers to expensing the acquisition cost less the residual value of intangible assets in a systematic manner over their estimated useful economic lives so as to reflect their consumption, expiration, obsolescence or other decline in value as a result of use or the passage of time.
- A corresponding concept for tangible assets is **depreciation**.
- Amortization is recorded in the financial statements of an entity as a reduction in the **carrying value** of the intangible asset in the **balance sheet** and as an **expense** in the **income statement**.
- Including assets in the balance sheet is named **recognition**. The result of recognition is **capitalization**.
- If not recognized, all expenses are recorded in the Loss & Profit account.

Example – Depreciation of a Server

- Purchase date: Oct 10, 2007
- Purchase Cost: € 6,000
- Depreciation method: Straight-line depreciation
- Salvage value / Residual value: € 0
- Server depreciates over 5 years
- Expense Schedule

Year	2007	2008	2009	2010	2011	2012	
Balance sheet value at beginning of the year [€]	6,000	5,600	4,400	3,200	2,000	800	
Expense [€]	(400)	(1,200)	(1,200)	(1,200)	(1,200)	(800)	Σ 6,000
Carrying Value by End of Year [€]	5,600	4,400	3,200	2,000	800	0	

In accounting, numbers enclosed in brackets indicate negative values.

The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**

Terms (2)

- **Intangible assets** are defined as those non-monetary assets that cannot be seen, touched or physically measured and which are created through time and/or effort. [...]
- The Uniform Commercial Code (Section 9-102(a)(42)) defines “general intangibles” as
“any personal property...other than accounts, chattel paper, commercial tort claims, deposit accounts, documents, goods, instruments, investment property, letter of credit rights, letters of credit, money, and oil, gas, or other minerals before extraction. The term includes payment intangibles and software.”

[Wikipedia.org, Intangible_asset]

An Interpretation of Intangible Assets

COMPETITIVE INTANGIBLES

Market Value of Equity Significantly Higher than Book Value - The assets that truly underpin our success today and in the future do not appear on the balance sheet. This can be seen by comparing SAP AG's market capitalization, which was €51.0 billion at the end of the year (2005: €48.5 billion), with shareholders' equity, which was €6.1 billion (2005: €5.8 billion). The difference is chiefly due to certain intangible assets that the applicable accounting standards do not allow to be recorded (at all or at fair value) on the balance sheet. They include **customer capital** (our customer base and customer relations), **the employees and their knowledge and skills**, **our ecosystem of partners**, **software we developed ourselves**, our **ability to innovate**, the **brands** we have built up – in particular the SAP brand itself – and our **organization**.

[SAP AG, Annual Report Fiscal Year 2006]

Oracle – Statement about Hyperion’s Intangible Assets

In performing our preliminary purchase price allocation, we considered, among other factors, our intention for future use of acquired assets, analyses of historical financial performance and estimates of future performance of Hyperion’s products. **The fair value of intangible assets was based, in part, on a valuation completed by a third party valuation firm using an income approach and estimates and assumptions provided by management.** The rates utilized to discount net cash flows to their present values were based on our weighted average cost of capital and ranged from 10% to 22%. These discount rates were determined after consideration of our rate of return on debt capital and equity and the weighted average return on invested capital. The following table sets forth the components of intangible assets associated with the Hyperion acquisition:

(Dollars in millions)

	Fair Value	Useful Life
Software support agreements and related relationships...	\$ 500	9 years
Developed technology	\$ 521	5 years
Core technology	\$ 211	7 years
Customer relationships	\$ 182	9 years
Trademarks and other	\$ 46	10 years
Total intangible assets	\$ 1,460	

Customer relationships and software support agreements and related relationships represent the underlying relationships and agreements with Hyperion’s installed customer base. **Developed technology, which comprises products that have reached technological feasibility, includes products in most of Hyperion’s product lines.** Core technology represents a combination of Hyperion processes, patents and trade secrets related to the design and development of its software products. This proprietary know-how can be leveraged to develop new technology and improve Oracle’s software products.

[Oracle Annual Report Fiscal Year 2007]

Accounting Practices of Selected German Companies

Company	Market Cap in Mio €	Applied Std	Capitalization of SW developed internally	
				Reason, if not
SAP AG	49,117.6	US-GAAP	no	intangible assets that the applicable accounting standards do not allow to be recorded
Software AG	2,005.0	IFRS	no	Allocation of expenses for research and development not possible
Nemetscheck AG	241.6	IFRS	yes	
Utimaco Software AG	139.3	IFRS	yes	
P&I Personal und Informatik AG	165.9	IFRS	yes	
Isra Vision Systems AG	82.9	IFRS	yes	
Fabasoft AG	37.8	IFRS	no	No feasibility
Beta Systems Software AG	41.6	IFRS	yes	
Valor Computerized Systems	83.1	IFRS	no	No feasibility
PSI AG	63.8	IFRS	Yes	
Atoss Software AG	34.3	IFRS	no	--- no further explanation ---
Parsytec AG	34.5	IFRS	no	--- no further explanation ---

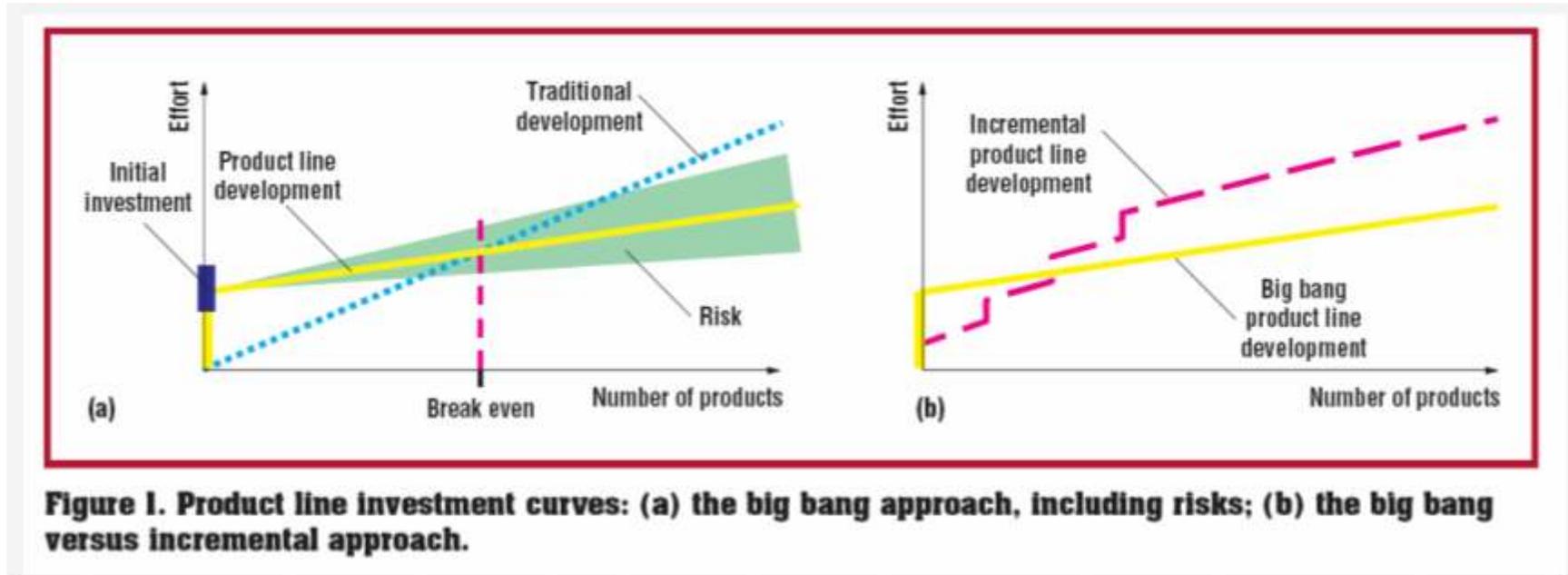
List of companies randomly selected from the Prime Software Performance Index (© Deutsche Börse AG), 67 companies total

The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**
- **Software is an “intangible asset”. Companies may decide to a certain extent, whether software is capitalized or not.**

Product Line Investments



Klaus Schmid, Martin Verlage: The Economic Impact of Product Line Adoption and Evolution. IEEE Software 19(4): 50-57 (2002)

Product Line Investments and Amortization

- **Scenario / Assumptions**

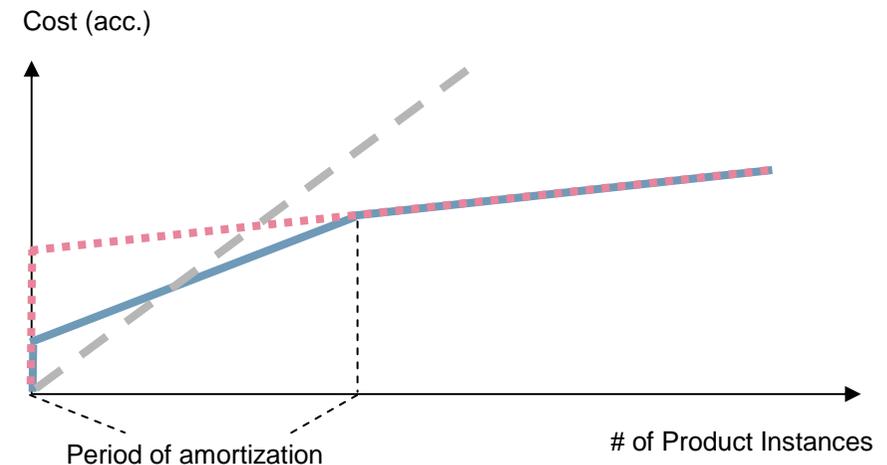
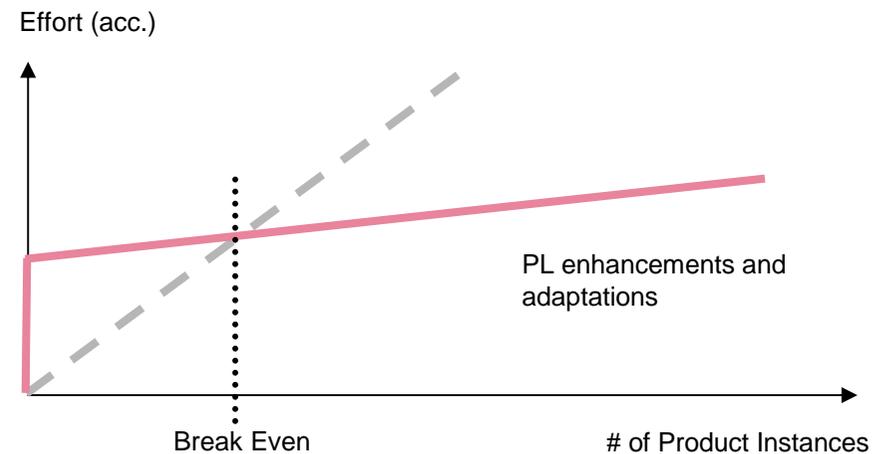
- 100 man months of project effort
- 1 man month costs 10,000 €

- **Without capitalization**

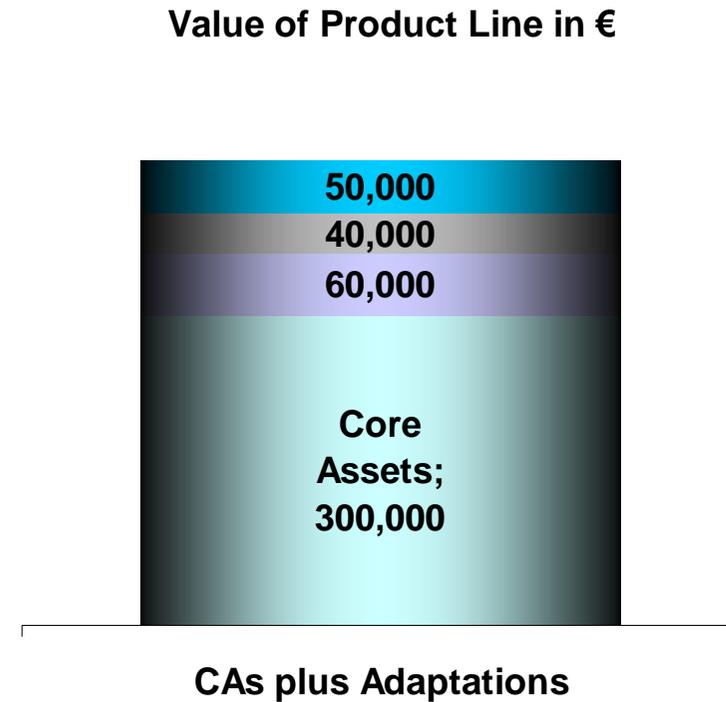
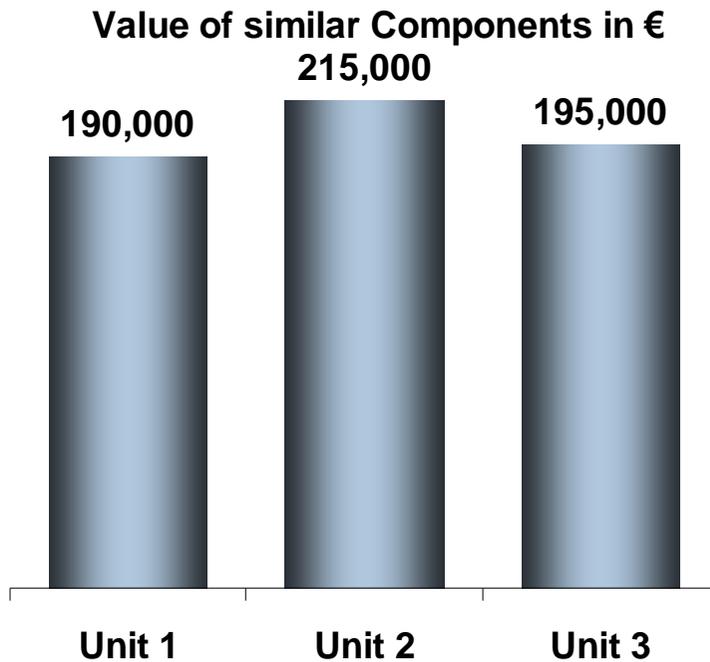
- Effort = Cost
- 1,000,000 € of expenses in the first year

- **With capitalization**

- Effort for preliminary project stage, post implementation stage and non-technical effort during development is 400,000 €. This amount is not to be capitalized.
- Effort of development stage is 600,000 €.
- Expenses are 120,000 € p.a. over 5 years.
- Effect on profit compared to “without capitalization”
 - + 480,000 € in Year 1
 - - 120,000 € in Year 2 – Year 5



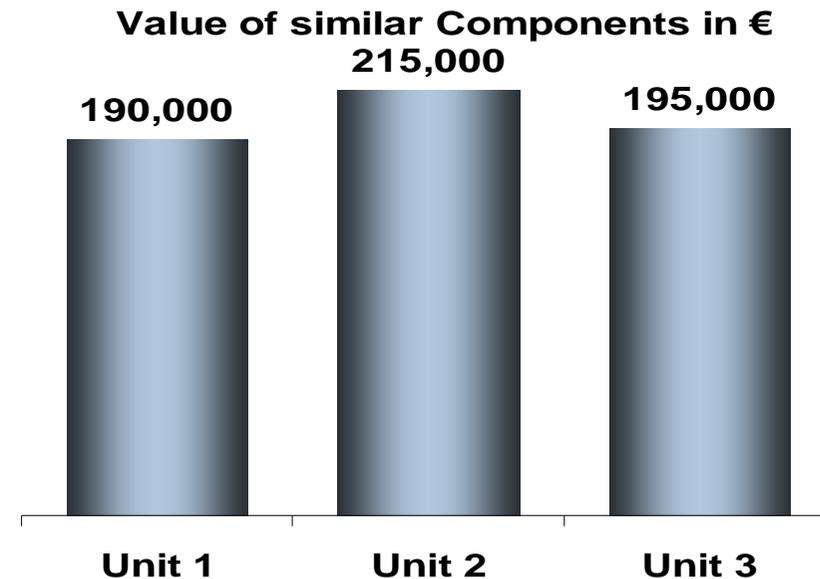
Recalling the Example



The Effect of Building a Product Line to Replace Systems

- **Example**

- Three development units have developed software which should be replaced by a software product line
- The units do not belong to the same company or group
- Software is recognized
- There is no budget assigned to maintenance, the effort is paid out of a general budget of the respective unit
- Unit 2 builds the new product line in one year on basis of their existing software and does maintenance



Book Value after 1 year

- Software of Unit 1: 152,000 €
- Software of Unit 2: 172,000 €
- Software of Unit 3: 156,000 €

Product Line Licenses

(1/3 of Core Assets plus Adoptions)

- Unit 1 purchases for 140,000 €
- Unit 3 purchases for 160,000 €

Effects on Book Value and Liquidity – without Capitalization

Transaction		Unit 1	Unit 2	Unit 3
1	Book Value at t	0	0	0
2	Book Value at t + 1y (Amortization 20%)	0	0	0
3	Stop using SW for U1 and U3 Expenses for Special Amortization	0	0	0
4	Building PL on existing SW Effect on Liquidity	0	(235,000)	0
5	Purchase License from U2	(140,000)	300,000	(160,000)
6	Capitalization of PL at t + 1y	140,000	0	160,000
7	Overall Effect on Liquidity (Rows 4 and 5)	(140,000)	65,000	(160,000)
8	Expenses for Amortization at t + 2y (20% of Row 6)	(28,000)	0	(32,000)
9	Overall Effect for Profit & Loss at t + 2y (Rows 3, 5 and 8)	(28,000)	65,000	(32,000)

Effects on Book Value and Liquidity – with Capitalization

Transaction		Unit 1	Unit 2	Unit 3
1	Book Value at t	190,000	215,000	195,000
2	Book Value at t + 1y (Amortization 20%)	152,000	172,000	156,000
3	Stop using SW for U1 and U3 Expenses for Special Amortization	(152,000)	0	(156,000)
4	Building PL on existing SW Effect on Liquidity	0	(235,000)	0
5	Purchase License from U2	(140,000)	300,000	(160,000)
6	Capitalization of PL at t + 1y	140,000	407,000	160,000
7	Overall Effect on Liquidity (Rows 4 and 5)	(140,000)	65,000	(160,000)
8	Expenses for Amortization at t + 2y (20% of Row 6)	(28,000)	(81,400)	(32,000)
9	Overall Effect for Profit & Loss at t + 2y (Rows 3, 5 and 8)	(180,000)	218,600	(188,000)

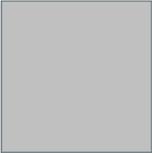
The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**
- **Software is an “intangible asset”. Companies may decide to a certain extent, whether software is capitalized or not. The difference between both ways is often significant.**

Remarks & Questions

- **General rule of thumb**
 - Capitalization is good for start-up companies
 - Reduction of expenses, increases earnings
 - Values of software parts are represented in the balance sheet
 - Capitalization for mature software producers requires careful consideration
 - Recognized software increases equity capital, hence Return on Equity Capital (RoE) is going down
 - Amortization of recognized software and capitalization of software developed newly is roughly the same
 - A way to document value of a company
- **When software can be capitalized or not, ...**
 - Who decides? – Members of the Board (CEO, CFO)
 - Is it allowed to change policy from time to time? - No
- **Freedom of choice is not an ideal**
 - Interpretation of economic figures of a company needs remarks and additional statements
 - Comparison between companies isn't easy
- **Standards, Regulations, and Laws**



Summary of Part I

In accounting software is a special case

- **Intangible asset**
- **Understanding changed (or still changes?) over time**

- **Regulations are modified and extended**
- **Appraisal of new regulations**
- **Interpretations of regulations (Recommendations) are modified from time to time**
- **„Best practices" change**

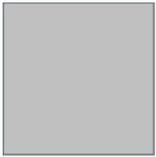
- **Companies determine their own course**

- **Concluding Remarks**
 - There is no single rule to be applied
 - You have to check how software developed internally is accounted

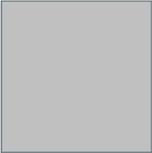
The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**
- **Software is an “intangible asset”. Companies may decide to a certain extent, whether software is capitalized or not. The difference between both ways is often significant.**

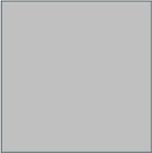


The Invisible Man-Month



Domain Engineering Budgets

- **General question: How visible is maintenance effort in a product line organization?**
- **Documentation of effort (cost) is practice in organizations that**
 - Perform contract work
 - Have a strong project management framework
- **But often organizations have budgets assigned to sub-units**
- **Domain Engineering is likely to receive general budgets**
 - How to control, what the money is used for?
 - What are adequate budgets?
 - Is domain engineering cheap or expensive?



The Kingdom of Eh-da

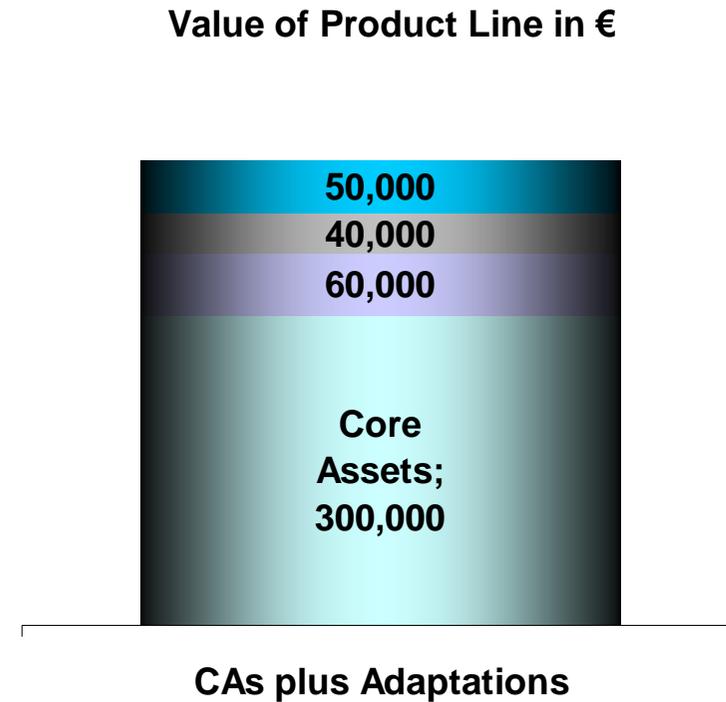
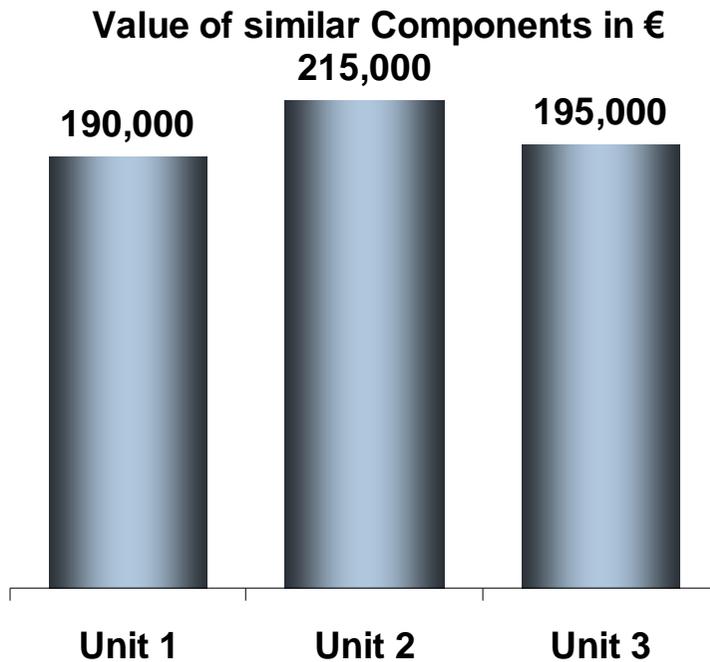
A king ruled a kingdom where 12 earls had to protect crown, land, and people. The tribes of the north stretched their greedy hands out to the kingdom. A tax was raised to finance an army. Each earl had to draft and to provide 1,000 soldiers and for that he got tax money from the king. The soldiers were based in each shire in order to flexibly build the army when the enemy would attack. To be prepared for fighting against a strong enemy the soldiers were expected to exercise every day.

The earl of the most southern shire was a clever guy. The risk to get attacked was the lowest for him among all other shires. So the earl of the most southern shire assigned 500 soldiers to handcraft tools which were sold across the whole kingdom.

One day the king visited the southern shire and he wanted to see the 1,000 soldiers. The earl took him to the balcony of the castle and because the place in front of the castle was too small for all of the soldiers they formed up in groups by a hundred. As soon as the first group left the place, the soldiers of that group took different uniforms to be presented a second time. The king was happy and continued the army's on-site inspection at the other shires.

Since the enemy knew about the kingdom's army he never attacked, but presented his soldiers from time to time at the border. The balance of power worked. Year by year the earl of the south became more rich and powerful, and he lived happily ever after.

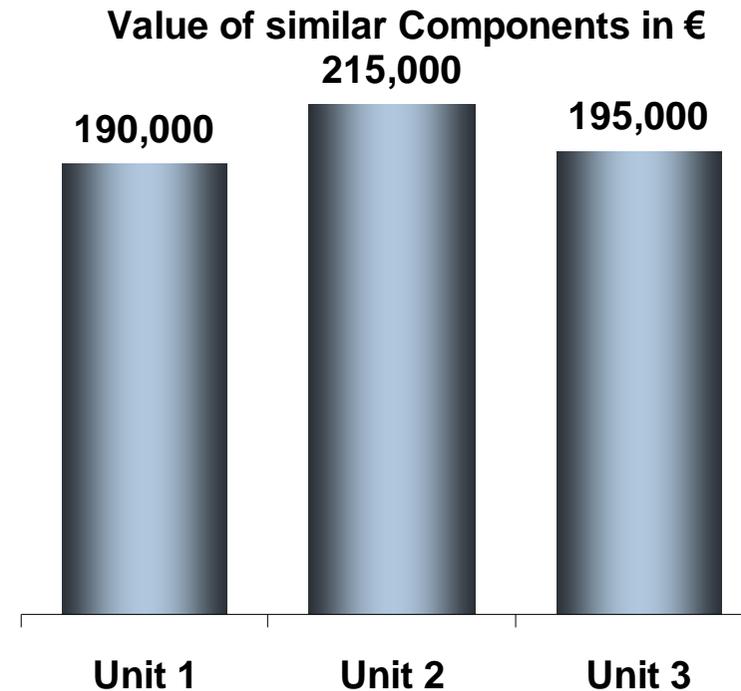
Recalling the Example



The Effect of Building a Product Line to Replace Systems

- **Example**

- Three development units have developed software which should be replaced by a software product line
- The units do not belong to the same company or group
- There is no budget assigned to maintenance, the effort is paid out of a general budget of the unit
- Unit 2 builds the new product line and does maintenance



Maintenance Cost = 15% of Development Cost

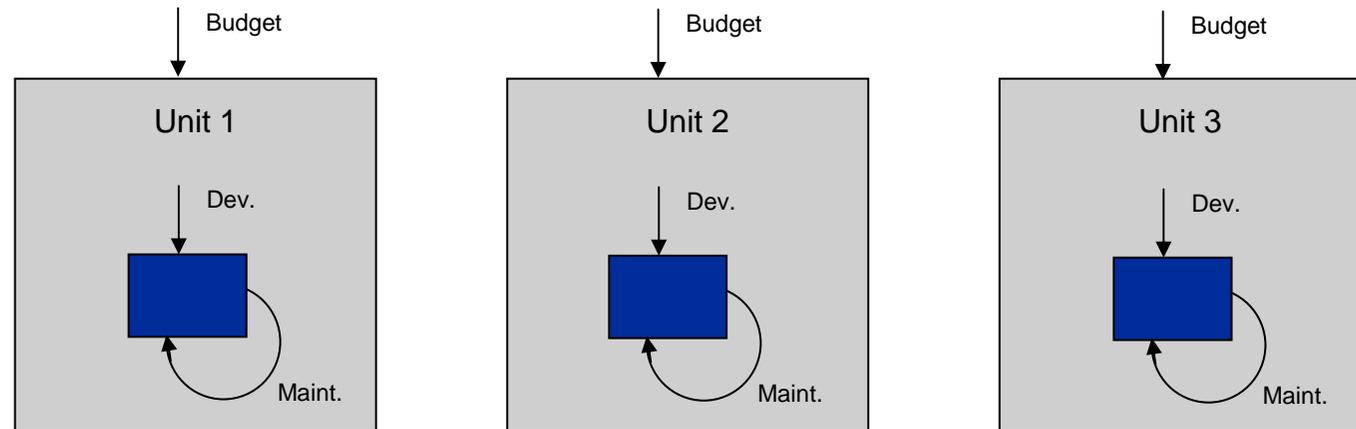
- Unit 1 spends effort equal to 28,500 € per annum
- Unit 2 spends effort equal to 32,250 € per annum
- Unit 3 spends effort equal to 29,250 € per annum

Maintenance for the Product Line

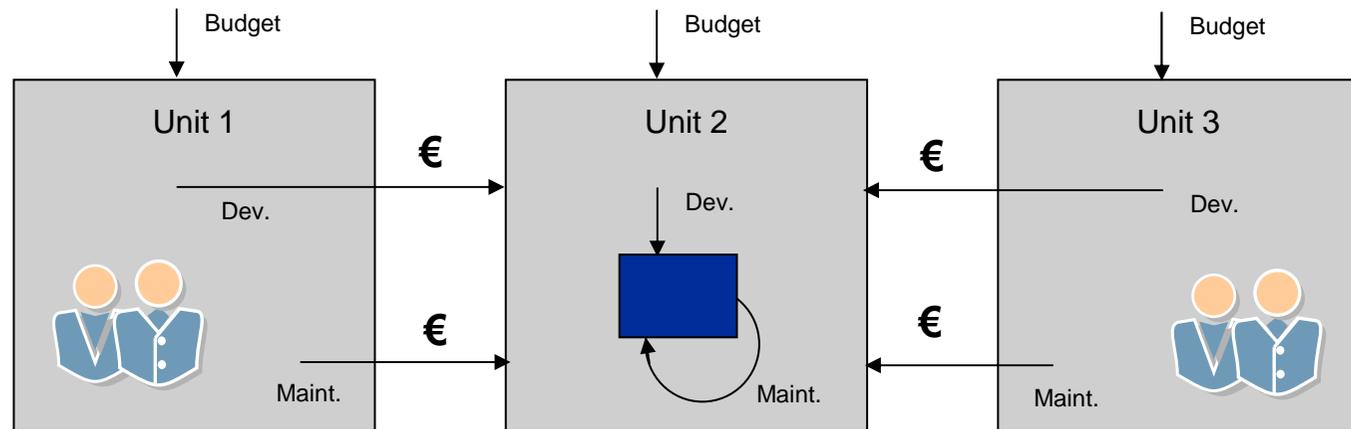
- 67,500 € per annum (15% of 450,000 €)
- is 22,500 € for each unit

Visualization of Effects

Before



After



The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**
- **Software is an “intangible asset”. Companies may decide to a certain extent, whether software is capitalized or not. The difference between both ways is often significant.**
- **Not only product line investments, but also maintenance costs are relevant.**

Effects of Product Line Maintenance

Transaction		Unit 1	Unit 2	Unit 3
1	Maintenance at t (black-box costs)	28,500.00	32,250.00	29,250.00
2	Maintenance at t + 1	0	67,500.00	0
3	Cash-flow	(22,500.00)	45,000	(22,500.00)
4	Available Resources [person months] *	2.85	-3.525	2.95
5	Revenues ** (11,550 multiplied with Row 4)	32,917.50	(40,713.75)	34,072.50
6	Benefit (p.a.) Row 5 + Row 3	10,417.50	4,286.25	11,572.50

* Assumption was 10,000 € internal cost per person month (see Slide 26)

** Assumption is: 900 € per day, 70% can be sold which equals to 11,550 € per person month at 220 workdays p.a.

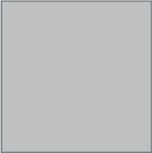
Working Models of Collaboration

- **Build product line from scratch**
 - Nothing has to be replaced
 - Major decision: funding (cash) or collaboration (developers)
- **Outsourcing**
 - Product line is developed by a joint venture
- **One partner is first and opens product for all other partners, which becomes a product line**
 - EPOC by Psion -> Symbian
 - Founders of Symbian Nokia, Ericsson, Motorola, and Psion
 - Many others followed
 - If successful: license fees replace funding
- **Product line adopted by partner who receives immediate benefits**
 - Replacement of expensive license
- **Give it away as Open Source?**

The Thread



- **Product Line Engineering is not purely technical, economic and organizational issues are also relevant.**
- **Among other definitions the equation “value = cost“ is important.**
- **In accounting, “depreciation” and “amortization” is a usual way to distribute the effect of cost to the company value across multiple years.**
- **Software is an “intangible asset”. Companies may decide to a certain extent, whether software is capitalized or not. The difference between both ways is often significant.**
- **Not only product line investments, but also maintenance costs are relevant.**
- **A working model for collaboration in setting up and maintaining a product line across companies or units requires a clear cost model and analysis of cash flows and effects to the balance sheet.**



Summary of Part II

- **Black-box costs**

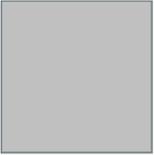
- No explicit relationship between cost and output
- Common practice
 - Budgets are assigned to organizational units
 - No distinction between different types of tasks
- Common problem in software development units with multiple products and responsibilities
 - Seldomly clear project definition
 - No tracking of “time and material” wrt. tasks and projects

- **Maintenance Costs**

- Keep an eye on cash flow, saving internal costs may cause cash flow
- Include the effects of saved maintenance effort to each partner

- **Issues for Product Line Engineering**

- Black-box should turn into white-box, when organizational unit boundaries are crossed



Summary

- **The purpose of this talk was to help to establish understanding about some non-technical issues**
- **and to point to selected economic phenomenons when being concerned with introduction of a product line.**

- **Software is an intangible asset**
- **Software in accounting is a complex matter**
- **There is room for interpretation, actual practices in a company may vary**

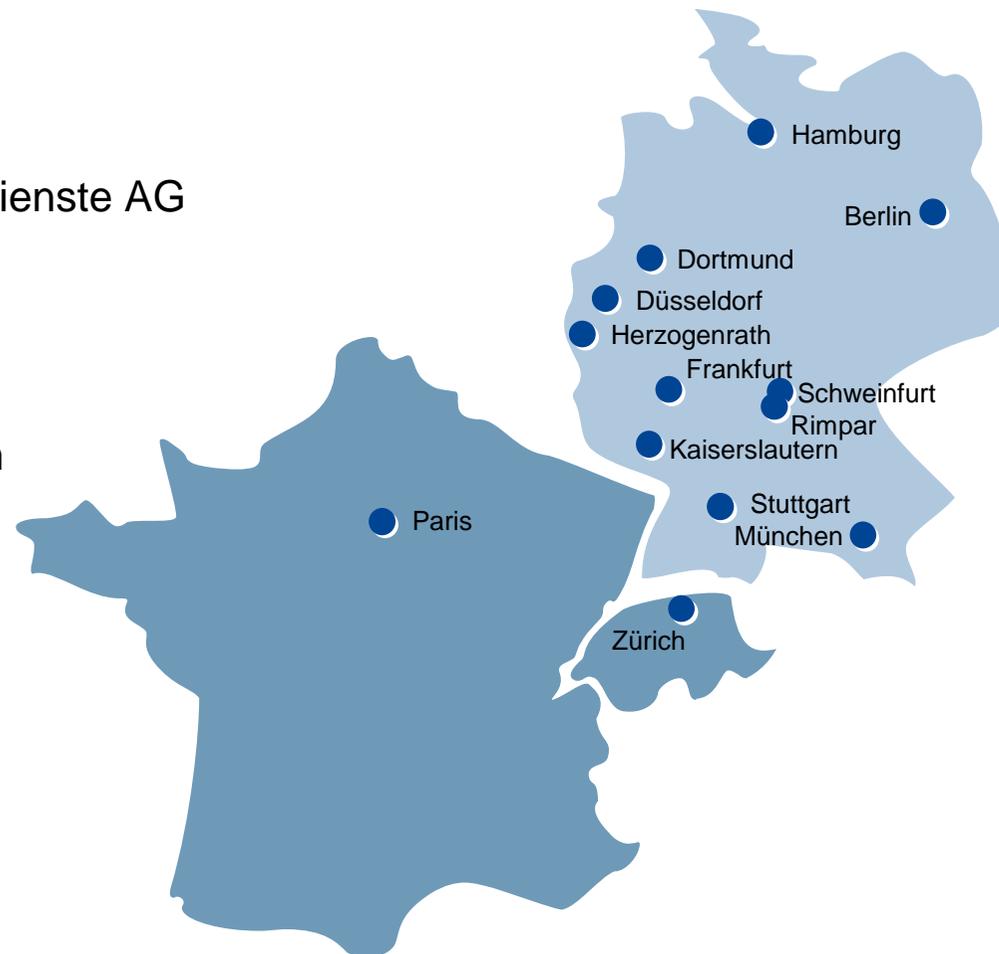
- **PLE is likely to result in large investments.**
- **Complex structures and economic effects might influence performance of PL projects.**

Contact Details

vwd group Technology

Martin Verlage
Vice Executive Director

vwd Vereinigte Wirtschaftsdienste AG
Tilsiter Strasse 1
60487 Frankfurt am Main
Germany
Phone: +49 69 50701-225
E-Mail: mverlage@vwd.com
www.vwd.com



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T1. Introduction to Software Product Lines

Linda Northrop, Software Engineering Institute, USA

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報
Information in Japanese

Abstract

Software product lines have emerged as a new software development paradigm of great importance. A software product line is a set of software intensive systems sharing a common, managed set of features, and that are developed in a disciplined fashion using a common set of core assets. Organizations developing a portfolio of products as a software product line are experiencing order-of-magnitude improvements in cost, time to market, staff productivity, and quality of the deployed products.

This tutorial will introduce the essential activities and underlying practice areas of software product line development. It will review the basic concepts of software product lines, discuss the costs and benefits of product line adoption, introduce the SEI's Framework for Software Product Line Practice, and describe approaches to applying the practices of the framework and includes material on the following topics

- Software product line concepts: What software product lines are and aren't; basic terminology; strategic reuse; success stories
- Costs and benefits: economics of software product lines; individual and organizational benefits
- The three essential activities: Core asset development; product development; organizational and technical management
- Product line practice areas: Practice area categories; the SEI's Framework for Software Product Line Practice
- Making it happen: Applying the practice areas; gauging an organization's ability to succeed with software product lines
- Wrap-Up: Current research and remaining challenges

Bio

Linda Northrop is Director of the Product Line Systems Program at the Software Engineering Institute (SEI), a position she has held since 1995. There she leads the work in software architecture, software product lines, predictable component engineering, and ultra-large-scale systems. Under her leadership, the SEI has developed software architecture and product line methods that are used worldwide, a series of five highly acclaimed books, and software architecture and software product line curricula. She is coauthor of *Software Product Lines: Practices and Patterns* and more than 50 other publications, has worked directly on successful software product lines, chaired both the first and second international Software Product Line Conference (SPLC1 and SPLC2), and is currently SPLC Steering Committee Chair. Prior to her career at the SEI, Linda was professor of computer science at the State University of New York (SUNY) College at Brockport and a Distinguished Visiting Professor at the United States Air Force Academy. She was also formerly associated with both Eastman Kodak and IBM as a software

engineer. As a private consultant, she also worked for an assortment of companies covering a wide range of software systems.

Linda has given product line tutorials at SPLC 2000, 2004, 2005, 2006; at OOPLSA, and AOSD and has developed and taught product lines courses as part of the SEI Software Product Line Curriculum.



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

T2. Domain-Specific Modeling and Code Generation for Product Lines

Juha-Pekka Tolvanen, MetaCase, Finland

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報

Information in Japanese

Abstract

Current modeling languages provide surprisingly little, if any, support for product line development. They are either based on the code world using the semantically well-defined concepts of programming languages (e.g. UML, SDL) or based on an architectural view using a simple component-connector concept. In both cases, the languages themselves say nothing about a product family or its variants. This situation could be compared to that of a programmer being asked to write object-oriented programs where the language does not support any object-oriented concepts.

However, recent advances in metamodeling and related technology (e.g. metamodeling tools, Software Factory concept (Greenfield et al. 2004) as well as tool frameworks provide better support for language creation. In particular, metamodel-driven tools can markedly reduce the time and effort required to develop a product family-specific method, along with its tool support and code generation. Industrial experiences of this approach show remarkable improvements in productivity (5-10 times faster variant creation) as well as capability to handle complex and large product lines (more than 100 product variants).

The main part of the tutorial addresses the guidelines for implementing methods for product family development: how to identify the necessary modeling language constructs, how to relate existing assets to a product family-specific method, and how to organize models describing family components and models specifying variants. The tutorial includes exercises for participants to create a domain-specific language for an imaginary product line. Although a special attention will be paid to the language creation principles, it will also be demonstrated that variation knowledge can be put into the generators or the platform for which variant information is an input. Finally, alternative tool support for domain engineering and application engineering is discussed.

Bio

Juha-Pekka Tolvanen is the CEO of MetaCase and co-founder of DSM Forum. He has been involved in model-driven development and tools, notably method engineering and metamodeling since 1991. He has acted as a consultant world-wide for modeling language development and has written over 50 articles for software development magazines and conferences, such as Embedded Systems, JavaSpektrum, DevX, Journal of AIS, MoDELS and Software Product Line Conference (www.cs.jyu.fi/~jpt). Juha-Pekka holds a Ph.D. in computer science in 1998 and he is also an adjunct professor (docent on software development methods) at the University of Jyväskylä, Finland.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T3. Using Feature Models for Product Derivation

Olaf Spinczyk, University of Erlangen-Nuremberg, Germany

Holger Papajewski, pure-systems, Germany

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Abstract

In general the implementation of a software product line leads to a high degree of variability within the software architecture. For an effective development and deployment it is necessary to resolve variation points within the architecture and source code automatically during product/variant derivation. Given the complexity of most software systems tool support is necessary for these tasks.

This tutorial shows how feature models combined with appropriate tools can provide this support. At first the importance of the separation of problem space modelling and solution space modelling is discussed. Concepts how to connect both spaces using constraints and/or generative approaches are shown. Furthermore some typical patterns of variability in the solution space are shown and their automatic resolution in common languages like C/C++ and Java is demonstrated. Integration of code generators, aspect-oriented programming and software configuration management systems into the derivation process is also discussed.

The tutorial is accompanied by short demonstrations of the presented concepts with freely available tools (fmp, XVLC, pure::variants, OAW) and thereby gives attendees an idea where to look when concepts are put into practice.

Bio

Olaf Spinczyk is an Assistant Professor of Computer Science at the Friedrich-Alexander University gen-Nuremberg, Germany. He received a Ph.D. from the University of Magdeburg, Germany, in 2002 for his research on the combination of aspect-orientation and product-line concepts for the construction of embedded operating system families. An important part of this work was the design of the AspectC++ language, which he started in 2001. AspectC++ is an aspect-oriented language extension for C++ in the style of AspectJ. Today he is the main designer and developer of the AspectC++ weaver, which he presented at various AOSD and OOPSLA demonstrations and tutorials. His current research is focused on the combination of generic and generative programming with AOP in AspectC++, and on applying these techniques in the implementation of the CiAO operating system family.

Holger Papajewski is CTO of the pure-systems GmbH. pure-systems is a software company specialized in services and tool development for the application of product line technologies in embedded software systems. Before he joined pure-systems he worked as researcher at the University of Magdeburg and was involved in network software and real-time operating system development. Among other things he was responsible for the development of OSEK-compatible family members of an object-oriented operating systems family.

At pure-systems he is responsible for the management of a software product line development with

日本語による情報
Information in Japanese

several hundred thousands lines of code, written in C++ and Java. Holger has (co-) authored several scientific articles during his academic career and also taught students in the field of operating system development.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T4. Building Reusable Testing Assets for a Software Product Line

John D. McGregor, Clemson University, Luminary Software, Software Engineering Institute, USA

Abstract

This tutorial focuses on the test assets and test processes created by a software product line organization. The tutorial will allow participants to consider how to modify existing testing practices to take advantage of strategic reuse. The software product line approach blends organizational management, technical management and software engineering principles to efficiently and effectively produce a set of related products. The major test assets: test plans, test cases, test data, and test reports are created at multiple levels of abstraction to facilitate their reuse. A product line organization also defines a test process that differs from the test process in a traditional development organization. This tutorial will allow participants to consider how to modify existing testing practices to take advantage of strategic reuse. At the end of this tutorial you will be able to:

- Understand the basic concepts of testing in software product line organizations.
- Understand the benefits, costs and risks of creating reusable test assets.
- Define a test process for your product line organization.
- Identify the steps necessary to initiate these activities for your organization.

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of Fame

Organization

Program Committee

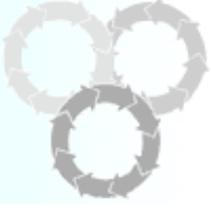
Sponsor

call-for-participation (pdf)

call-for-participation (txt) **Bio**

日本語による情報
Information in Japanese

Dr. John D. McGregor is an associate professor of computer science at Clemson University, a founding partner of Luminary Software, and a Visiting Scientist at the Software Engineering Institute. He is co-author of two books on software engineering, including "A Practical Guide to Testing Object-Oriented Software Engineering." Dr. McGregor teaches graduate software engineering courses, courses in the SEI's software product line curriculum, and has presented numerous tutorials at a variety of conferences. He consults with numerous software development organizations.



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

T5. Introduction to Software Product Adoption

Linda Northrop, Software Engineering Institute, USA

Larry Jones, Software Engineering Institute, USA

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

Abstract

The tremendous benefits of taking a software product line approach are well documented. Organizations have achieved significant reductions in cost and time to market and, at the same time, increased the quality of families of their software systems. However, to date, there are considerable barriers to organizational adoption of product line practices. If an organization is sold on the concept, how is it to proceed? Phased adoption is attractive as a risk reduction and fiscally viable proposition. This tutorial describes a phased, pattern-based approach to software product line adoption. A phased adoption strategy is attractive as a risk reduction and fiscally viable proposition. The tutorial begins with a discussion of software product line adoption issues and then presents the Adoption Factory pattern. The Adoption Factory pattern provides a roadmap for phased, product line adoption. The tutorial covers the Adoption Factory in detail, including focus areas, phases, subpatterns, related practice areas, outputs, and roles. Examples of product line adoption plans following the pattern are used to illustrate its utility. The tutorial also describes strategies for creating synergy within an organization between product line adoption and ongoing CMMI or other improvement initiatives.

Bio

Linda Northrop is Director of the Product Line Systems Program at the Software Engineering Institute (SEI), a position she has held since 1995. There she leads the work in software architecture, software product lines, predictable component engineering, and ultra-large-scale systems. Under her leadership, the SEI has developed software architecture and product line methods that are used worldwide, a series of five highly acclaimed books, and software architecture and software product line curricula. She is coauthor of *Software Product Lines: Practices and Patterns* and more than 50 other publications, has worked directly on successful software product lines, chaired both the first and second international Software Product Line Conference (SPLC1 and SPLC2), and is currently SPLC Steering Committee Chair. Prior to her career at the SEI, Linda was professor of computer science at the State University of New York (SUNY) College at Brockport and a Distinguished Visiting Professor at the United States Air Force Academy. She was also formerly associated with both Eastman Kodak and IBM as a software engineer. As a private consultant, she also worked for an assortment of companies covering a wide range of software systems.

Linda has given product line tutorials at SPLC 2000, 2004, 2005, 2006; at OOPLSA, and AOSD and has developed and taught product lines courses as part of the SEI Software Product Line Curriculum.

Dr. Lawrence G. Jones is a Senior Member of the Technical Staff in the Product Line Systems Program at the Software Engineering Institute (SEI) of Carnegie Mellon University with over 36

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報
Information in Japanese

years experience in software development, management and education. Before joining the SEI, Larry served a career in the US Air Force and is the former Chair of the Computer Science Department at the Air Force Academy. Larry is a Senior Member of the IEEE, is the current Chair of the ABET Computing Sciences Accreditation Commission, and is also an ACM Director on the Computing Sciences Accreditation Board.

Larry is a frequent contributor to the software literature and a speaker at software conferences. Recent product line tutorial presentations include: SPLC in 2004, 2005, 2006; SEPG conferences in 2005; European SEPG conferences in 2005. Additionally he teaches product line courses as part of the SEI Software Product Line Curriculum.



T6. Generative Software Development

Krzysztof Czarnecki, University of Waterloo, Canada

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Abstract

Product-line engineering seeks to exploit the commonalities among systems from a given problem domain while managing the variabilities among them in a systematic way. In product-line engineering, new system variants can be rapidly created based on a set of reusable assets (such as a common architecture, components, models, etc.). Generative software development aims at modeling and implementing product lines in such a way that a given system can be automatically generated from a specification written in one or more textual or graphical domain-specific languages (DSLs).

In this tutorial, participants will learn how to perform domain analysis (i.e., capturing the commonalities and variabilities within a system family in a software schema using feature modeling), domain design (i.e., developing a common architecture for a system family), and implementing software generators using multiple technologies, such as template-based code generation and model transformations. Available tools for feature modeling and implementing DSLs as well as related approaches such as Software Factories and Model-Driven Architecture will be surveyed and compared.

The presented concepts and methods will be demonstrated using a sample case study of an e-commerce platform.

Bio

Krzysztof Czarnecki is an Assistant Professor at the University of Waterloo, Canada. Before coming to Waterloo, he spent eight years at DaimlerChrysler Research working on the practical applications of generative programming. He is co-author of the book "Generative Programming" (Addison-Wesley, 2000), which is regarded as founding work of the area and is used as a graduate text at universities around the world. He was a keynote speaker the 2006 International Conference on Generative Programming and Component Engineering (GPCE) and will be the program chair for MoDELS 2008. His current work focuses on realizing the synergies between generative and model-driven software development.

日本語による情報

Information in Japanese

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T8. Predicting Product Line Payoff with SIMPLE

Paul Clements, Software Engineering Institute, USA

John McGregor, Clemson University, Luminary Software, Software Engineering Institute, USA

Dirk Muthig, Fraunhofer Institute for Experimental Software Engineering, Germany

Abstract

There are many decision points in software product line engineering that require a sound economic basis for resolving. The Structured Intuitive Model for Product Line Engineering (SIMPLE) was designed to help software product line decision-makers weigh the economic costs and benefits of decision alternatives, such as

- Should we convert our existing products to a software product line or keep them as separate projects?
- Should we add a new product to the software product line?
- Should we structure our family of products as one software product line or several?
- How much will it cost to make a change to all of the members of our product line?
- What will be the return on investment for converting a product portfolio to a software product line?

SIMPLE is based on a small number of cost functions, which model different dimensions the cost of taking a decision alternative, as well as a set of benefit functions, which model the corresponding benefits of that alternative. The cost functions include the cost of setting up the core asset base, the cost of using that core asset base to build a product, the cost of building the part of a product that does not come from the core asset base, and the cost of making organizational changes to launch and institutionalize product line practices.

SIMPLE has been featured in IEEE Software, in a 2004 SPLC panel, in the Software Engineering Institute's Framework for Software Product Line Practice, a recent book about product line engineering, and in papers that recount its use in industry.

The primary aim of the tutorial is to help a decision-maker understand the basics of software product line strategies, the advantages of the approach, and what will be involved in adopting a product line approach. Additionally, the aim is to help them assess where their organization stands in relation to the capabilities needed to launch a product line and to give them the necessary tools to tailor adoption and execution strategies to their organization.

Bio

Home
Keynotes
Program
Tutorials
Workshops
Panels
Registration
Local Information

CFP
Submissions
Demonstration
Doctoral symposium
Product Line Hall of Fame
Organization
Program Committee
Sponsor

call-for-participation (pdf)
call-for-participation (txt)

日本語による情報
Information in Japanese

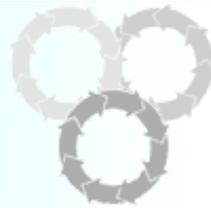
Dr. Paul C. Clements is a Senior Member of the Technical Staff at Carnegie Mellon University's Software Engineering Institute. He is co-author of five books in software engineering and software architecture, including "Software Product Lines: Practices and Patterns" and "Software Architecture in Practice." He regularly teaches three of the two-day courses in the SEI's software architecture curriculum, and has presented numerous half-day and full-day tutorials at major conferences.

Dr. John D. McGregor is an associate professor of computer science at Clemson University, a founding partner of Luminary Software, and a Visiting Scientist at the Software Engineering Institute. He is co-author of two books on software engineering, including "A Practical Guide to Testing Object-Oriented Software Engineering." Dr. McGregor teaches graduate software engineering courses, courses in the SEI's software product line curriculum, and has presented numerous tutorials at a variety of conferences. He consults with numerous software development organizations.

Dr. Dirk Muthig heads the "Product Line Architectures" department at the Fraunhofer Institute for Experimental Software Engineering (IESE). He has been involved in the definition and development of Fraunhofer's PuLSE™ (Product Line Software Engineering) methodology since 1997, as well as in the transfer of product line and architecture technology into diverse industrial organizations. Dirk Muthig has organized workshops at SPLC and ICSR in 2004 and 2006, as well as teaches product line engineering at the Technical University of Kaiserslautern for several years. He received a diploma in computer science, as well as a Ph. D., from the Technical University of Kaiserslautern.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T9. New Methods Behind the New Generation of Software Product Line Success Stories

Charles W. Krueger, BigLever Software, USA

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Abstract

A new generation of software product line (SPL) success stories is being driven by a new generation of methods, tools and techniques. While early SPL case studies at the genesis of the field revealed some of the best software engineering improvement metrics in decades, the early generation SPL methodologies tended to be large, complex, and offer an overabundance of options and choices, making adoption in practice extremely difficult to comprehend, justify and achieve. In this tutorial, we explore a simpler new generation SPL methodology, referred to as the *3-Tiered SPL Methodology*. It is based on observations and firsthand experiences during deployments and operation of the latest generation of successful commercial SPL practices, including highly acclaimed Software Product Line Hall of Fame inductees Salion and LSI Logic/Engenio. The 3-Tiered SPL Methodology has supported full scale operational transitions to software product line practice, involving hundreds of software engineers, millions of lines of source code, and billions of dollars in commercial product lines.

Attendees will learn very advanced and yet very practical concepts in the 3-Tiered SPL Methodology that are yielding order-of-magnitude improvements in time-to-market, engineering cost, product quality, and portfolio scalability. It features best practices including minimally-invasive and agile strategies, reactive product and core asset scoping, software mass customization sans application engineering, bounded product line combinatorics, model-driven and aspect-oriented integrations with SPL, and end-to-end software product portfolio line lifecycle management.

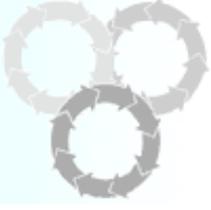
The case studies and demonstrations featured in the tutorial will provide pragmatic insights into how the new generation methods, tools and techniques are enabling companies of all types and sizes to realize a new level of benefits, in terms of both technical and business impact. While these new success stories exhibit the 10x engineering improvements we have come to expect from the earlier generation, what is most unexpected is that the new 3-Tiered SPL Methodology requires up to 100x less time, cost and effort to make the transition to product line practice and to achieve return on investment.

Bio

日本語による情報
Information in Japanese

Charles Krueger, PhD, is the founder and CEO of BigLever Software, a leading provider of software product line development tools and services. He is a thought leader in the software product line development field, with 20 years of experience in software development practice. He has proven expertise in leading commercial software product line development teams, and helping companies establish some of the industry's most highly acclaimed software product line practices, including Salion, 2004 Software Product Line Hall of Fame inductee, and LSI Logic, 2006 Software Product Line Hall of Fame inductee.

Dr. Krueger is an author and speaker for over 30 articles, columns, book chapters, and conference sessions. He has co-chaired the International Conference on Software Reuse, is a frequent organizer and speaker for the International Software Product Line Conferences, and moderates the SoftwareProductLines.com practitioner community website. Most recently, Dr. Krueger presented at SD Best Practices and Architecture & Design World, was an invited author for the Communications of the ACM special issue on Software Product Lines, and was featured in a Dr. Dobbs podcast spotlighting Software Product Lines. He received his PhD in computer science from Carnegie Mellon University.



11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan

T10. Variability Management for Product Lines with a Generative Technique

Stan Jarzabek, National University of Singapore, Singapore

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報

Information in Japanese

Abstract

The presented approach helps handle inter-dependent variant features with system-wide (delocalized) impact on Product Line (PL) assets. Such variability requires a mechanism to manage one-to-many, complex mappings between variant features and affected components, as well as the impact of feature inter-dependencies on components. Conventional architecture-centric and component-based approaches do not have explicit mechanisms to manage that. The symptoms of the problem include explosion of similar component versions and difficulties to find, customize and integrate component configurations during derivation PL members. Both problems complicate a PL Architecture and hinder reuse productivity.

In the tutorial, we show how these problems can be alleviated by complementing architecture/component approaches with a generative technique of XVCL (<http://xvcl.comp.nus.edu.sg>). The idea of the approach is to build generic, adaptable software structures, with explicit record of customizations required to accommodate variant features in their legal combinations into PL members. XVCL helps us to build such generic structures, as a PL Architecture from which to derive custom PL members. Differences among PL members caused by variant features are specified as deltas from these generic structures and automatically propagated to PL members. XVCL mechanisms are simple, but they effectively address a number of problems that are difficult to solve in the PL practice such as avoiding explosion of similar components, component selection/customization during product derivation, PL Architecture evolution, or selective injection of variant features (any changes, in fact) into PL members that need them, without affecting other PL members that do not need them. We can evolve PL members in independent directions, without ever disconnecting them from the PL Architecture. We base the tutorial on industrial projects and on studies that explain in detail technical underpinnings of the solutions.

Bio

Stan Jarzabek is an Associate Professor at the Department of Computer Science, School of Computing, National University of Singapore (NUS). He spent 12 years of his professional career in industry and 20 years in academia. Stan is interested in all aspects of software design, in particular techniques for design of adaptable, easy to change (high-variability) software. He has published over 90 papers in international journals and conference proceedings (his recent paper received the ACM Distinguished Paper Award and two other papers were selected as "best papers" at leading conferences). Stan was a Principal Investigator in a multi-national collaborative project involving universities (NUS and the University of Waterloo) and companies in Singapore and Toronto.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T11. Improving Product Line Development with the Families Evaluation Framework (FEF)

Klaus Schmid, University of Hildesheim, Germany

Frank van der Linden, Philips Medical Systems, The Netherlands

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報

Information in Japanese

Abstract

This tutorial introduces the Families Evaluation Framework (FEF), a systematic approach for evaluating and improving the product line development practices of an organization. Using the concepts of the FEF, practitioners are capable of analyzing their current development practices, determine an appropriate target situation and identify a possible migration path.

The approach aims both to support the transition to a product line approach as well as the ongoing improvement. As more and more organizations are interested in the transition to a product line engineering approach or even improving an existing approach, such assessment and improvement approaches are of prime importance for industrial practice.

The FEF is comparable to the Product Line Practice Framework / PLTP of the SEI, however, it has been built upon the experience of a large number of European companies in the Families project. Further, the FEF is well aligned with the CMM-I.

The FEF is an exhaustive framework for describing product line level competence in an organization, which was initially developed as part of the Families project. It relies on four dimensions: the business, architecture, process, and organizational dimensions. These four dimensions are mostly independent and are thus independently assessed. The approach relies on the assumption that there is no single product line profile that fits each organization. As a consequence, product line assessment starts with the determination of a target profile. Then an analysis of the current status is performed, leading to action steps, describing how to get from "here" to "there".

In this tutorial, we will provide the practitioner with an overview of the approach and the core concepts of how to apply this approach. It is not the aim of such a limited tutorial to replace assessment training.

Bio

Prof. Dr. Klaus Schmid leads the software systems engineering group at the University of Hildesheim. He worked in various product line projects; both in research projects like ESAPS, Café, and Families and in industrial projects. He has authored numerous papers, in particular, in the area of product line scoping and economics and also gave various tutorials on product line scoping and modelling related topics.

Dr. Frank van der Linden works at Philips Medical Systems CTO Office. He received his Ph.D. in pure Mathematics (algebraic number theory) in 1984 at the University of Amsterdam. Since then he was employed by Philips, first at Philips Research, and since 1999 by Philips Medical Systems. In this time he did a lot of work in the area of software engineering especially considering software architecture and software development processes, with an emphasis on software product line engineering. His present job is to initiate and execute cooperation projects in software engineering improvement with external, mainly European, parties. As part of this job he was the project leader of the three successive ITEA projects on product line engineering: ESAPS, CAFÉ and FAMILIES. During these projects a series of SWAPF & PFE workshops were organized, all with Frank van der Linden as program chair, and editor of the proceedings. As a follow up he was general chair of the SPLC 2005 conference, and program co-chair of the SPLC 2006 conference.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T12. Transforming Legacy Systems into Software Product Lines

Danilo Beuche, pure-systems, Germany

Home
Keynotes
Program
Tutorials
Workshops
Panels
Registration
Local Information

Abstract

Not every software product lines starts from the scratch, often organizations face the problem that after a while their software system is deployed in several variants and the need arises to migrate to systematic variability and variant management using a software product line approach.

The tutorial will discuss issues coming up during this migration process mainly on the technical level, leaving out most of the organisational questions. The goal of the tutorial is to give attendees an initial idea how a transition into a software product line development process could be done with respect to the technical transition.

The tutorial starts with a brief introduction into software product line concepts, discussing terms such as problem and solution space, feature models, versions vs. variants.

Tutorial topics are how to choose adequate problem space modelling, the mining of problem space variability from existing artefacts such as requirements documents and software architecture. Also part of the discussion will be the need for separation of problem space from solution space and ways to realize it. A substantial part will be dedicated to variability detection and refactoring in the solution space of legacy systems.

CFP
Submissions
Demonstration
Doctoral symposium
Product Line Hall of Fame
Organization
Program Committee
Sponsor

call-for-participation (pdf)
call-for-participation (txt)

Bio

Danilo is managing director of the pure-systems GmbH. pure-systems is a software company specialized in services and tool development for the application of product line technologies in embedded software systems. When he joined the GMD First (now Fraunhofer FIRST) in 1995, he started to work in the field of embedded operating systems and software families and continued at the University Magdeburg, where he also received his PhD in this area. His work on tool support for feature based software development finally lead to the founding of pure-systems in 2001. At pure-systems he works also as consultant in the area of product line development mainly for clients from the automotive industry.

He has been tutorial presenter, speaker, workshop organizer and panelist at conferences such as AOSD, ISORC, SPLC and OOPSLA. He is also author of articles in scientific journals and software developer magazines. During his university career and also to a limited degree later on he has been teaching students as tutor, teaching assistant and lecturer in the areas of operating system development and software engineering.

日本語による情報
Information in Japanese

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T13. Production Planning in a Software Product Line Organization

Gary J. Chastek, Software Engineering Institute, USA

John D. McGregor, Clemson University, Luminary Software, Software Engineering Institute, USA

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

Abstract

Most software product line organizations recognize the need for two roles: core asset developers and product builders. These roles may both be assumed by an individual or each may be assumed by persons who are in different administrative units, in different geographic locations, or of vastly different skill levels. For example, a corporation may have one lab assigned to produce core assets and other labs around the world to use those assets to produce products. The greater the separation among these people the greater the need for communication and coordination regarding product production.

Production planning is used in many industries to coordinate the efforts of external suppliers who supply parts and to structure the assembly line where products are produced. The need for coordination in a software product line organization is even greater than in hard goods manufacturing because product production is less constrained by physical properties or industrial standards. Our research has shown that organizations that fail to plan production are more likely to fail than those that do plan. The goal of this tutorial is to provide participants with techniques for conducting production planning.

We will cover the complete product line life cycle from adoption until a first generation of products is developed. We use a business strategy development tool, Porter's Five Forces model, to guide strategy development. We will use the Software Process Engineering Meta-model and an instantiation of it, the Eclipse Process Framework for method development and documentation. For the production plan we will use a document template that has been used with numerous clients.

日本語による情報
Information in Japanese

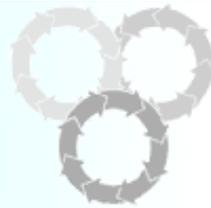
Bio

Dr. Gary J. Chastek is a senior member of the technical staff at the Software Engineering Institute in the Software Product Line Initiative. He has presented tutorials and lead workshops at SPLC and OOPSLA. Dr. Chastek's current research interests include production planning, variability management, and the use of aspect-oriented development in a software product line.

Dr. John D. McGregor is an associate professor of computer science at Clemson University, a founding partner of Luminary Software, and a Visiting Scientist at the Software Engineering Institute. He is co-author of two books on software engineering, including "A Practical Guide to Testing Object-Oriented Software Engineering." Dr. McGregor teaches graduate software engineering courses, courses in the SEI's software product line curriculum, and has presented numerous tutorials at a variety of conferences. He consults with numerous software development organizations.

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



T14. Leveraging Model Driven Engineering in Software Product Line Architectures

Bruce Trask, MDE Systems, USA

Angel Roman, MDE Systems, USA

[Home](#)

[Keynotes](#)

[Program](#)

[Tutorials](#)

[Workshops](#)

[Panels](#)

[Registration](#)

[Local Information](#)

[CFP](#)

[Submissions](#)

[Demonstration](#)

[Doctoral symposium](#)

[Product Line Hall of Fame](#)

[Fame](#)

[Organization](#)

[Program Committee](#)

[Sponsor](#)

[call-for-participation \(pdf\)](#)

[call-for-participation \(txt\)](#)

日本語による情報

Information in Japanese

Abstract

Model Driven Engineering (MDE) is a new innovation in the software industry that has proven to work synergistically with Software Product Line Architectures. It can provide the tools necessary to fully harness the power of Software Product Lines. The major players in the software industry including commercial companies such as IBM, Microsoft, standards bodies including the Object Management Group and leading Universities such as the ISIS group at Vanderbilt University are embracing this MDE/PLA combination fully. IBM is spearheading the Eclipse Foundation including its MDE tools like EMF, GEF and GMF. Microsoft has launched there Software Factories foray into the MDE space.

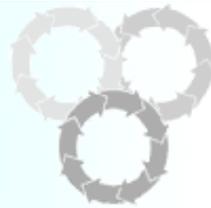
The goal of this tutorial is to educate attendees on what MDE technologies are, how exactly they relate synergistically to Product Line Architectures, and how to actually apply them using an existing Eclipse implementation.

The process of Developing Software Product Line Architectures can be a complex task. However, the use of Model Driven Engineering (MDE) techniques can facilitate the development of SPLAs by introducing Domain Specific Languages, Graphical Editors, and Generators. Together these are considered the sacred triad of MDE. Key to understanding MDE and how it fits into SPLAs is to know exactly what each part of the trinity means, how it relates to the other parts, and what the various implementations are for each. This tutorial will demonstrate the use of the Eclipse Modeling Framework (EMF) and Eclipse's Graphical Editor Framework (GEF) to create an actual MDE solution as applied to a sample SPLA. When building Graphical Modeling Languages using GEF and EMF one may find themselves worrying about specific implementation details related to EMF or GEF. To address this issue, the Eclipse community has created a generative bridge between EMF and GEF called The Graphical Modeling Framework (GMF). During this tutorial we will also illustrate how to model the visual artifacts of our Domain Model and generate a Domain Specific Graphical Editor using GMF.

Bio

11th International Software Product Line Conference (SPLC 2007)

Sept. 10-14, 2007, Kyoto, Japan



Managing Variability for Software Product Lines

Organizer:

Paul Clements (clements at sei.cmu.edu)

Dirk Muthig

Description:

Managing variability is the essence of software product line practice. Variability enters the product line picture through the need for different features, deployment on different platforms, the desire for different quality attributes, and the accommodation of different deployment scenarios. Eventually, every need for variability manifests itself in one way or another in the actual artifacts that populate a product line's core asset base, whether those artifacts are software, tools, documents, or other kinds of assets.

"Variation mechanisms" are the name we give to the constructs that achieve variation at the artifact level. Catalogs of these mechanisms have been published, and they come in a bewildering variety. Selecting the correct variation mechanism(s) can have a dramatic effect on the cost to deploy new products, react to evolutionary pressures, and in general maintain and grow the product line. But selection remains an ad hoc process in nearly all product line organizations.

This workshop series is intended to fill the void between variability requirements visible to those who deal with features and other product-level concerns, and the variation mechanisms visible to creators and consumers of a product line's core assets.

Topics of interest for the workshop include, but are not limited to:

- Reasoning frameworks for variability selection
- Factors that affect the selection of variability mechanisms
- Cost models to enable reasoned selection of variability mechanisms
- Variability mechanisms especially suited for non-software artifacts
- Binding time issues from an strategic or economic viewpoint

The goal of the workshop series is to codify a body of knowledge for the informed and purposeful selection of variation mechanisms to use in a software product line's core assets.

Submission / Important Dates:

Download [PDF](#) file.

Deadline for paper submission: August 1, 2007

Deadline for author notification: August 15, 2007

Home

Keynotes

Program

Tutorials

Workshops

Panels

Registration

Local Information

CFP

Submissions

Demonstration

Doctoral symposium

Product Line Hall of

Fame

Organization

Program Committee

Sponsor

call-for-participation (pdf)

call-for-participation (txt)

日本語による情報
Information in Japanese

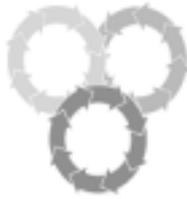
Participants in the SOAPL Workshop will include product line practitioners who have experience in using service-oriented architectures, or who are interested in migrating their software product lines to SOA in the future. These include practitioners in product line engineering as well as product line management roles. We also welcome participation from architects/developers of SOA-based systems who are interested in applying product line practices in development of their systems.

Submission:

Please visit above URL.

Important Dates:

Submit a 3-6 page position paper or experience report by July 1st with notifications of paper or experience report acceptance by July 15, 2007.



11th Software Product Line Conference (SPLC 2007)
"Software Product Lines –
Power for Competitive Advantage"
September 10-14, 2007 Kyoto, Japan

第 11 回ソフトウェアプロダクトライン国際会議 (SPLC 2007) 参加募集

<http://sec.ipa.go.jp/SPLC2007/>

グローバル化の進展の中、企業は世界市場の多様な要求に効果的かつ効率的に対応することが求められています。ソフトウェアプロダクトラインは大幅に生産性を改善する開発パラダイムとして注目され、その適用やコミュニティも拡大してきています。ソフトウェアプロダクトライン国際会議(SPLC)は実務者から研究者まで幅広い背景の参加者が集う、この分野で最も権威のある会議です。今回京都で開催される第 11 回ソフトウェアプロダクトライン国際会議(SPLC 2007)は、アジア地区で初めての開催となります。是非ともご参加ください。

● 会議の概要

日程：2007 年 9 月 10 日(月) ～ 14 日(金)

会場：京都リサーチパーク(KRP) 京都市下京区

協賛：社団法人 情報処理学会

参加申込：上記 SPLC2007 の WEB ページで詳細確認の上、お申込みください。

早期割引締切：8 月 24 日、登録締切：9 月 5 日

参加費： 本会議： 52,000 円 (早期割引) 60,000 円 (通常)
1 チュートリアル： 11,000 円 (早期割引) 13,000 円 (通常)
1 ワークショップ： 10,000 円 (早期割引) 13,000 円 (通常)

● プログラム

○9 月 10 日, 11 日：ワークショップ・チュートリアル・博士シンポジウム

9月10日(月)	
9:00-17:00 ワークショップ	9:00-12:30 チュートリアル
W1: Open Source Software and Product Lines (OSSPL07 Asia)	T1. Introduction to Software Product Lines <i>Linda Northrop</i> T2 Domain-Specific Modeling and code generation for Product Lines <i>Juha-Pekka Tolvanen</i>
W2: Service-Oriented Architectures and Product Lines (SOAPL - 07)	T3 Using Feature Models for Product Derivation <i>Olaf Spinczyk, Holger Papajewski</i> T4 Building Reusable Testing Assets for a Software Product Line <i>John D. McGregor</i>
W4: Dynamic Software Product Line (DSPL 07)	13:30-17:00 チュートリアル T5. Introduction to Software Product Adoption <i>Linda Northrop, Larry Jones</i> T6. Generative Software Development <i>Krzysztof Czarniecki</i> T8. Predicting Product Line Payoff with SIMPLE <i>Paul Clements, John McGregor, Dirk Muthig</i>

9月11日(火)	
9:00-17:00 ワークショップ	9:00-12:30 チュートリアル
W5: Managing Variability for Software Product Lines	T9 New Methods Behind the New Generation of Software Product Line Success Stories <i>Charles W. Krueger</i>
W6: 4th Software Product Line Testing Workshop (SPLiT2007)	T10. Variability Management for Product Lines with a Generative Technique <i>Stan Jarzabek</i>
W7: Visualization in Software Product Line Engineering (ViSPLE 2007)	T11. Improving Product Line Development with the Families Evaluation Framework (FEF) <i>Klaus Schmid, Frank van der Linden</i>
W8: Product Lines in Practice – Taking Stock (PLiP07)	13:30-17:00 チュートリアル
	T12. Transforming Legacy Systems into Software Product Lines <i>Danilo Beuche</i>
	T13. Production Planning in a Software Product Line Organization <i>Gary J. Chastek, John D. McGregor</i>
	T14. Leveraging Model Driven Engineering in Software Product Line Architectures <i>Bruce Trask, Angel Roman</i>

※13:30-17:00 博士シンポジウム

○9月12日～14日：本会議

- ・ 基調講演 (1) : 松本吉弘 (京都高度技術研究所)
- ・ 基調講演 (2) : Martin Verlage (vwd Vereinigte Wirtschaftsdienste GmbH)

9月12日(水)		9月13日(木)		9月14日(金)	
オープニング					
基調講演(1)		基調講演(2)		パネル	
論文発表 フィーチャモデリング	論文発表 経験論文(1)	論文発表 変動性管理	論文発表 経験論文(2)	論文発表 ショートペーパー(1)	論文発表 ショートペーパー(2)
論文発表 コスト・見積り	デモ発表	論文発表 経験論文(3)	デモ発表	プロダクトラインの殿堂 クロージング	
論文発表 製造	デモ発表	論文発表 アスペクト /MDA	パネル		
レセプション		バンケット			

第 11 回ソフトウェアプロダクトライン国際会議 (SPLC 2007)

プログラム

○ 9月10日, 11日 : ワークショップ・チュートリアル・博士シンポジウム

9月10日(月)	
<p>9:00-17:00 ワークショップ</p> <p>W1: Open Source Software and Product Lines (OSSPL07 Asia)</p> <p>W2: Service-Oriented Architectures and Product Lines (SOAPL - 07)</p> <p>W4: Dynamic Software Product Line (DSPL 07)</p>	<p>9:00-12:30 チュートリアル</p> <p>T1: Introduction to Software Product Lines <i>Linda Northrop</i></p> <p>T2: Domain-Specific Modeling and code generation for Product Lines <i>Juha-Pekka Tolvanen</i></p> <p>T3: Using Feature Models for Product Derivation <i>Olaf Spinczyk, Holger Papajewski</i></p> <p>T4: Building Reusable Testing Assets for a Software Product Line <i>John D. McGregor</i></p> <hr/> <p>13:30-17:00 チュートリアル</p> <p>T5: Introduction to Software Product Adoption <i>Linda Northrop, Larry Jones</i></p> <p>T6: Generative Software Development <i>Krzysztof Czarnecki</i></p> <p>T8: Predicting Product Line Payoff with SIMPLE <i>Paul Clements, John McGregor, Dirk Muthig</i></p>

9月11日(火)	
<p>9:00-17:00 ワークショップ</p> <p>W5: Managing Variability for Software Product Lines</p> <p>W6: 4th Software Product Line Testing Workshop (SPLiT2007)</p> <p>W7: Visualization in Software Product Line Engineering (ViSPLE 2007)</p> <p>W8: Product Lines in Practice – Taking Stock (PLiP07)</p>	<p>9:00-12:30 チュートリアル</p> <p>T9: New Methods Behind the New Generation of Software Product Line Success Stories <i>Charles W. Krueger</i></p> <p>T10: Variability Management for Product Lines with a Generative Technique <i>Stan Jarzabek</i></p> <p>T11: Improving Product Line Development with the Families Evaluation Framework (FEF) <i>Klaus Schmid, Frank van der Linden</i></p> <hr/> <p>13:30-17:00 チュートリアル</p> <p>T12: Transforming Legacy Systems into Software Product Lines <i>Danilo Beuche</i></p> <p>T13: Production Planning in a Software Product Line Organization <i>Gary J. Chastek, John D. McGregor</i></p> <p>T14: Leveraging Model Driven Engineering in Software Product Line Architectures <i>Bruce Trask, Angel Roman</i></p>

※13:30-17:00 博士シンポジウム:

○9月12日～14日：本会議

- ・ 基調講演 (1) : 松本吉弘 (京都高度技術研究所)
- ・ 基調講演 (2) : Martin Verlage (vwd Vereinigte Wirtschaftsdienste GmbH)

9月12日(水)	
会議登録	
08:30-09:00 開会	
09:00-10:15 基調講演(1) Dr. Yoshihiro Matsumoto Experience of a Software Factory from Domain Preparation to Product Line Adoption	
10:40-12:10 T11A: 論文発表 フィーチャモデリング Automating Mappings between Use Case Diagrams and Feature Models for Software Product Lines <i>Alexandre Braganca, Ricardo J. Machado</i> Reasoning about Feature Models in Higher-Order Logic <i>Mikolas Janota, Joseph Kiniry</i> Feature Diagrams and Logics: There and Back Again <i>Krzysztof Czarnecki, Andrzej Wasowski</i>	10:40-12:10 T11B: 論文発表 経験論文(1) Experiences with Product Line Development of Multi-Discipline Analysis Software at Overwatch Textron Systems <i>Paul Jensen</i> Software Product Lines Approach in Enterprise System Development <i>Yuzo Ishida</i> Introducing Software Product Line Engineering for Metal Processing Lines in a Small to Medium Enterprise <i>David Sellier, Mike Mannion, Gorka Benguria, Gorka Urchegui</i>
13:30-15:00 T12A: 論文発表 コスト・見積もり Impacts of Architecture and Quality Investment in Software Product Line Development <i>Makoto Nonaka, Liming Zhu, Muhammad Ali Babar, Mark Staples</i> Comparing Costs and Benefits of Different Test Strategies for a Software Product Line: A Study from Testo AG <i>Dharmalingam Ganesan, Jens Knodel, Ronny Kolb, Uwe Haury, Gerald Meier</i> Using Requirements Management Tools in Software Product Line Engineering: The State of the Practice <i>Danilo Beuche, Andreas Birk, Heinrich Dreier, Andreas Fleischmann, Isabel John, Ramin Tavakoli Kolagari, Heidi Galle, Gerald Heller, Dirk Janzen, Thomas von der Massen</i>	13:30-15:00 T12B: デモ
15:30-17:00 T13A: 論文発表 製造 The 3-Tiered Methodology: Pragmatic Insights from New Generation Software Product Lines <i>Charles W Krueger</i> Development and Configuration of Service-based Product Lines <i>Alexander Gruler, Alexander Harhurin, Judith Hartmann</i> A Production System for Software Product Lines <i>Gary J. Chastek, Patrick Donohoe, John D. McGregor</i>	15:30-17:00 T13B: デモ
レセプション	

9月13日(木)	
会議登録	
09:00-10:15 基調講演(2) Dr. Martin Verlage The Invisible Man-Month or What is the real value of a core asset?	
10:40-12:10 T21: 論文発表 変動性管理 Automating Product-Line Variant Selection for Mobile Devices Jules White, Douglas C. Schmidt, Egon Wuchner, Andrey Nechypurenko Supporting Product Derivation by Adapting and Augmenting Variability Models Rick Rabiser, Paul Gruenbacher, Deepak Dhungana Optimization of Variability in Software Product Lines Felix Loesch	10:40-12:10 T21B: 論文発表 経験論文(2) A Guide for Management and Financial Controls of Product Lines Yoshihiro Matsumoto Building Software Product Line from the Legacy Systems Kangtae Kim, Hyungrok Kim, Woomok Kim A Component Model supporting Decomposition and Composition of Consumer Electronics Software Product Lines Chong-Mok Park, Seokjin Hong, Kyoung-Ho Son, Jagun Kwon
13:30-15:00 T22A: 論文発表 経験論文(3) Challenges of Establishing a Software Product Line for an Aerospace Engine Monitoring System Ibrahim M Habli, Tim P Kelly, Ian Hopkins Minimally Invasive Migration to Software Product Lines Hans Peter Jepsen, Jan Gaardsted, Danilo Beuche Dynamic Complexity and the Owen Firmware Product Line Program Holt Mebane, Joni T. Ohta	13:30-15:00 T22B: デモ
15:30-17:00 T23A: 論文発表 アスペクト/MDA A Case Study Implementing Features using AspectJ Christian Kraestner, Sven Apel, Don Batory Product Line Implementation using Aspect-Oriented and Model-Driven Software Development Markus Voelter, Iris Groher Higher-Order Transformations for Product Lines Jon Oldevik, Oystein Haugen	15:30-17:00 T23B: PANEL SW Product Line Evolution and Life Cycle
バンケット	

9月14日(金)	
会議登録	
09:00-10:30 パネル	
10:50-11:35 T31A: 論文発表 ショートペーパー(1) Tailoring Infrastructure Software Product Lines by Static Application Analysis <i>Horst Schirmeier, Olaf Spinczyk</i> A Variability Modeling Method for Adaptable Services in Service-Oriented Computing <i>Soo Ho Chang, Soo Dong Kim</i>	10:50-11:35 T31B: 論文発表 ショートペーパー(2) Why does it take that long? Establishing Product Lines in the Automotive Domain <i>Christian Tischer, Andreas Mueller, Markus Ketterer, Lars Geyer</i> Software Product Line Impacts Beyond Software Development <i>Scott G Decker, James C Dager</i>
11:45-13:00 プロダクトラインの殿堂 閉会	

※プロダクトラインの殿堂：プロダクトラインの成功事例について殿堂入りの議論をする催し。

http://www.sei.cmu.edu/productlines/plp_hof.html