



Successful Product Line Development with Strong Requirements Engineering and Management

Andreas Birk and Gerald Heller

Tutorial at RE 2006, 14th IEEE International Requirements Engineering Conference
September 11, 2006, Minneapolis/St. Paul, MN, USA

- Handout Slides -

andreas.birk@sdm.de, gerald.heller@hp.com



Agenda

- Introduction
- Fundamentals of SPL and REM
- Strong REM practices for SPL development
- Growing your REM practices
- New promises: Agile techniques
- Summary



Agenda

- Introduction
- **Fundamentals of SPL and REM**
 - What is SPL engineering?
 - Challenges of REM in SPL
 - Importance of REM in SPL
- Strong REM practices for SPL development
- Growing your REM practices
- New promises: Agile techniques
- Summary

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

3



What is a Software Product Line?

- Software Engineering Institute (SEI):

A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

4

Carnegie Mellon: SEI SPL Practice

Address: <http://www.sei.cmu.edu/productlines/>

Carnegie Mellon Software Engineering Institute

Home Search Contact Us Site Map What's New

About the SEI Management Engineering Acquisition Work with Us Products and Services Publications

ENGINEERING

Software Product Lines

What is a Software Product Line?

A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

Why are Software Product Lines Important?

Software product lines are rapidly emerging as a viable and important software development paradigm allowing companies to realize order-of-magnitude improvements in time to market, cost, productivity, quality, and other business drivers. Software product line engineering can also enable rapid market entry and flexible response, and provide a capability for mass customization.

Join the Software Product Line Community

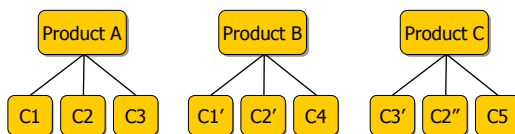
The 10th International Software Product Line Conference (SPLC 2006), is scheduled for August 21-25, 2006, Baltimore, Maryland, USA. SPLC is the leading venue for practitioners, researchers, and educators to reflect on the software product line achievements made during the past decade, assess the current state of the field, and identify key challenges still facing researchers and practitioners. SPLC 2006 is the tenth official gathering of the software product line community. SPLC 2006 will feature research and experience papers, topical panels, tutorials, workshops, demonstrations, birds-of-a-feather discussions, and other opportunities for members of the product line community to interact.

September 11, 2006 RE 2006, Minneapolis/St. Paul, MN, USA

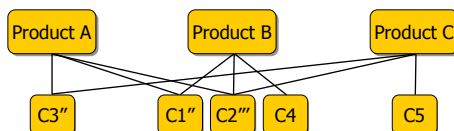
5

Advantages of product line architectures

Product component architecture without product line



Product component architecture with product line



Advantages of product line architectures

- Ability to manage large product portfolios
- Improved development productivity
- Enhanced product quality
- Shorter release cycles
- Rapid responses to customer and market demands
- Faster qualification of new development team members

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

6



SPL-driving scenarios

- Suites
 - Existing applications are integrated over time
 - Customer uses more than one product from SPL
 - SPL motivation: Reduced cost of ownership
 - E.g.: Microsoft Office
- Specialized products
 - New applications are instantiated from the framework
 - Customer uses only one product from SPL
 - SPL motivation: Delivery efficiency
 - E.g.: Cummings, Bosch

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

7



REM for Suites

- Incremental domain analysis after the fact
 - Identify existing shared or similar requirements
- Stepwise approach with long-term strategy
- Re-architect and re-build existing applications with improved functionality
- Customer usage scenarios are key enablers for SPL integration strategy
- Customer requirements drive product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

8

REM for specialized products

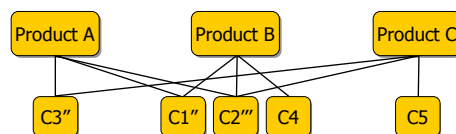
- Upfront domain analysis
- Requirements are typically not shared across customers
- Each customer needs to have the full requirements perspective
 - Flexible requirements reporting is mandatory
- Re-use is sensitive as the customer pays for functionality which was already developed
- Customer may not be interested in product line

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

9

SPL architecture and organizational structure



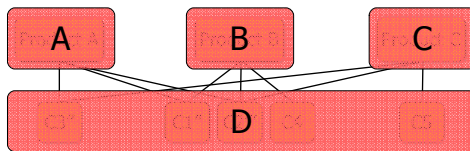
- Product line architecture: A set of core assets (C1-C5) shared by several products (A, B, C)

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

10

SPL architecture and organizational structure



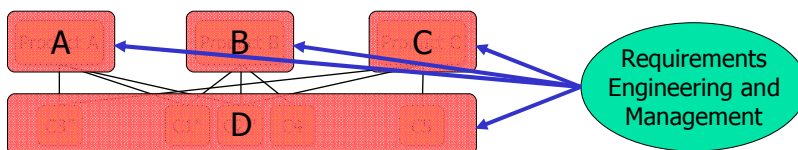
- Product line architecture: A set of core assets (C1-C5) shared by several products (A, B, C)
- Organizational structure: Projects (A-D) perform the development of core assets and products

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

11

SPL architecture and organizational structure



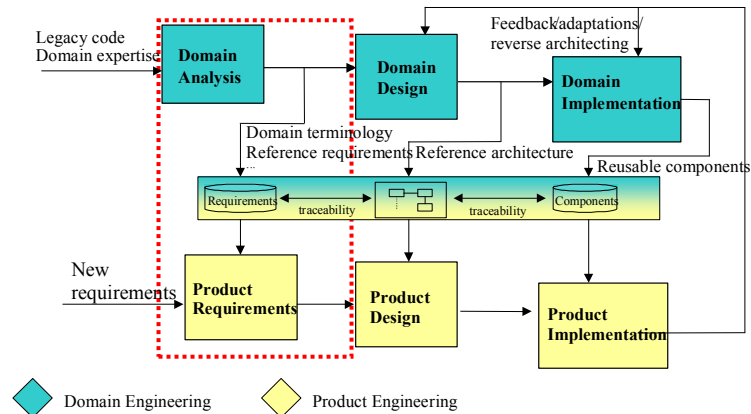
- Product line architecture: A set of core assets (C1-C5) shared by several products (A, B, C)
- Organizational structure: Projects (A-D) perform the development of core assets and products
- REM: Processes and a set of tasks that coordinate requirements within and for the projects

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

12

Two instances of REM process in SPL

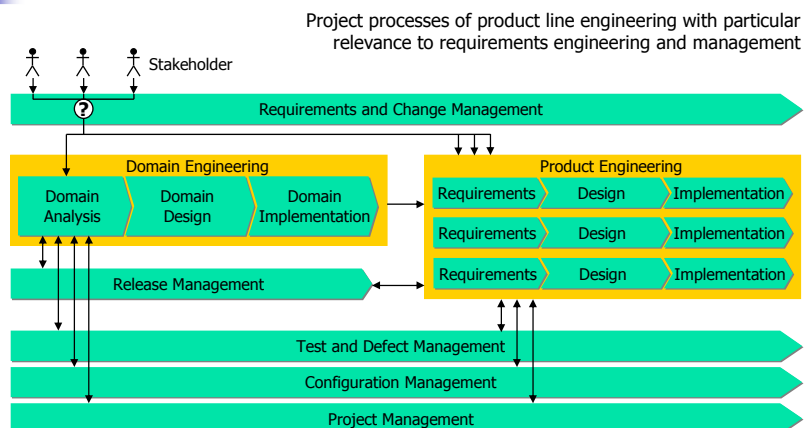


September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

13

REM interacts with many project processes



September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

14

Key challenges of REM for SPL

Category	Challenges
Organization and Management	Justification of the platform approach as a process model by a cost / benefit-analysis
	Independent platform team
	Difficult cooperation between platform and product development teams
	Proof of justification of the platform team
	High communication overhead
	Poor configuration management
Requirements engineering	Influence of the architecture on requirements negotiation is not taken into account
	No description of variability for domain analysis
	Missing domain analysis and domain description
	Discussions on design and not on requirements level
	No explicit requirements process
	Missing tool support
Product- vs. platform-specific	Sequence of integrating requirements into the platform
	No explicit prioritization of requirements
	Realization of platform requirements in products
	Strong influence of the pilot client
Architecture	No use of the architectural advantages
	Poor description of the generic architecture

Source: A. Berk, G. Heller, I. John, T. von der Maßen, K. Müller, K. Schmidt:
Product Line Engineering: The State of the Practice. IEEE Software, 20(6), 2003.

15

SPL requirements are harder ...

... because of the inherent complexity of SPL

- Two-stage RE phase: Product and platform
- Many product variants
- Long-living requirements
- Many different stakeholders
- Change management of requirements
- Versions and variants of requirements
- Tool support difficult to establish

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

16



Agenda

- Introduction
- Fundamentals of SPL and REM
- **Strong REM practices for SPL development**
- Growing your REM practices
- New promises: Agile techniques
- Summary

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

17



Agenda: Strong REM practices for SPL development

- **REM in the realm of project and program management**
- REM roles and organization for product lines
- Portfolio planning and requirements prioritization
- Product line architecture
- Documenting product line requirements
- Establishing the right communication structures
- Tool support for REM in product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

18



Project(s) versus SPL program

- The structure of a SPL program is determined by three dimensions
 - Platform/product dimension: Platform, N products
 - Time/release dimension I: N releases at a time
 - Time/release dimension II: Maintain past releases (of platform AND applications)
- Program involves meta-project management: Coordination across projects

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

19



Project and program management interact with REM

- Keep strong requirements focus during project and program management
 - Requirements impact on plan
 - Planning and re-planning impact on requirements
- Maintain common objectives across the teams
 - Acknowledge contributions to common objectives
 - Continuously “pray” the importance of SPL

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

20



REM practices that support program management

- Requirements change management
- Requirements estimation
- Management of dependencies between requirements and schedule

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

21



Toolbox: Project and program management

- Visual project / program landscape
- Uniform SPL-wide project management practices
- Continuous requirements management
- Requirements change management
- Continuously improved prediction capabilities ("agile planning")
- Continuously maintained and disseminated estimation base

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

22

Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- **REM roles and organization for product lines**
- Portfolio planning and requirements prioritization
- Product line architecture
- Documenting product line requirements
- Establishing the right communication structures
- Tool support for REM in product line development

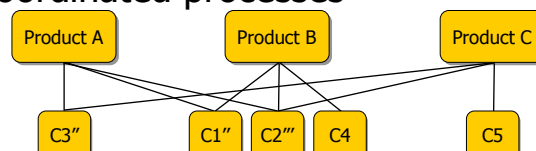
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

23

Roles and organization

- REM roles and processes need to be adapted to support the various organizational SPL set-ups
- Program management establishes coordinated processes



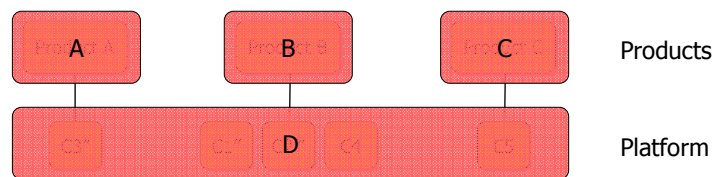
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

24

Roles and organization

- This is the *classical* platform approach
- 4 development departments
 - 3 for products, 1 for platform
- How to organize REM processes?



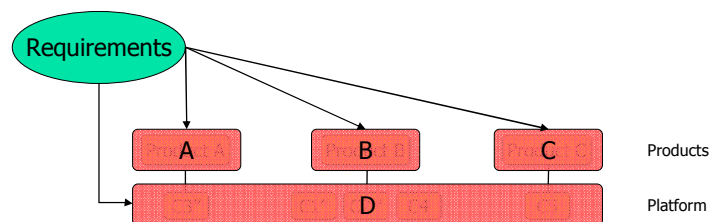
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

25

Roles and organization: Central REM – Domain analysts

- **Domain analysts** from marketing and R&D
 - Collect requirements for all products centrally
 - Analyze requirements for their commonality potential
 - Dispatch requirements to products and/or platform
 - Establish relationships between product and platform requirements
 - Meet regularly to synchronize, reduce backlog, monitor progress and manage escalations



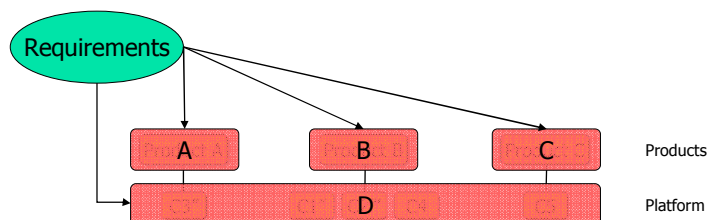
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

26

Roles and organization: Central REM - Advantages

- Provides portfolio-wide oversight
- Eases balancing activities
- Ensures liveliness of platform
- Fosters visibility between products and framework
- Reduces stress about late surprises



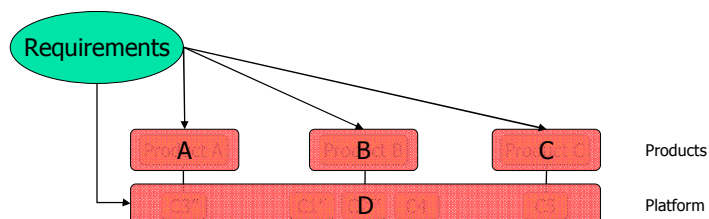
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

27

Roles and organization: Central REM - Challenges

- Too much time to get requirements for products
- Continuous availability of team members
- Tracing and monitoring of decisions



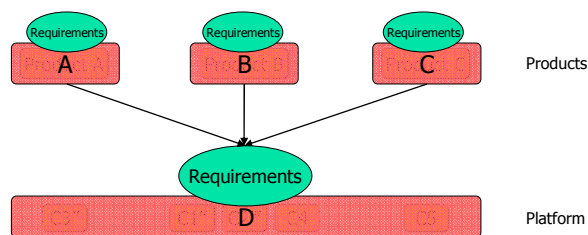
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

28

Roles and organization: Decentralized REM

- Requirements are 'dumped' from product teams
- Domain analysis for requirements takes place in platform team alone



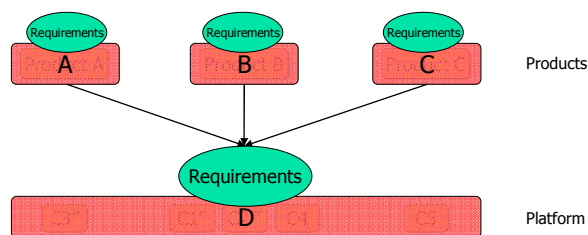
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

29

Roles and organization: Decentralized REM - Advantages

- Fast action from product perspective
- Simple organizational structure with clear separation of concerns



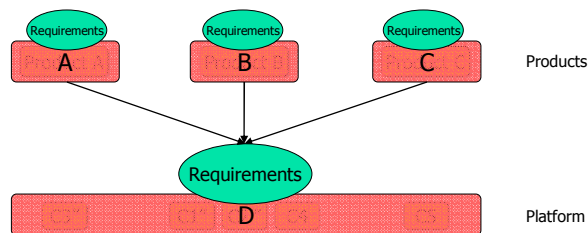
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

30

Roles and organization: Decentralized REM – Challenges

- Shared requirements may not be recognized
- Platform team needs competent domain experts
- Platform may get weaker over time



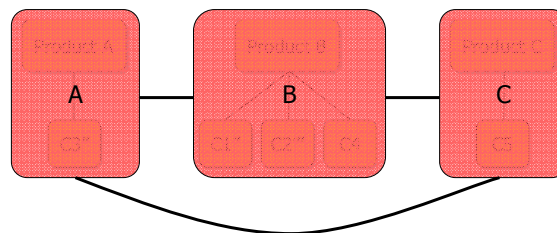
September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

31

Roles and organization: No platform team

- Shared ownership over components
- Requires an extremely good collaboration
- Works well in initial phases of the product line for suites



September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

32



Toolbox: Roles and organization

- Establish an organization which ensures good collaboration in your company's context
- Depending on the maturity of the SPL different organizations might be chosen
 - E.g.: Once the SPL is more mature, a central RM team is recommended
- Establish a ***domain analyst*** role and explicitly assign someone to it

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

33



Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- REM roles and organization for product lines
- **Portfolio planning and requirements prioritization**
- Product line architecture
- Documenting product line requirements
- Establishing the right communication structures
- Tool support for REM in product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

34



Portfolio planning

- The portfolio planning process provides guidance for efficient SPL requirements management
- Deliverables of this process are
 - Investment guidance between products
 - Relative priority of products
 - Release time-window expectations
 - Portfolio-wide requirements

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

35



Prioritization

- Starting point: Best practices for single product development
 - E.g.: Spreadsheet from Karl Wiegers web-site

Feature	Stakeholder 1		Stakeholder 2		Stakeholder 3		Total		Total		Total		Relative		Relative	
	Benefit	Penalty	Benefit	Penalty	Benefit	Penalty	Benefit	Penalty	Value	Value %	Cost	Cost %	Risk	Risk %	Priority	
	Relative	Relative	Relative	Relative	Relative	Relative	Relative	Relative	Value	Value %	Cost	Cost %	Risk	Risk %	Priority	
F1	7	3	5	3	5	3	6.2	3.0	9.2	10.6	2	6.1	1	3.8	1.07	
F2	3	7	9	7	9	7	5.4	7.0	12.4	14.3	5	15.2	3	11.5	0.54	
F3	5	5	5	5	5	5	5.0	5.0	10.0	11.5	3	9.1	2	7.7	0.69	
F4	2	1	2	1	2	1	2.0	1.0	3.0	3.5	1	3.0	1	3.8	0.50	
F5	4	9	4	9	4	9	4.0	9.0	13.0	15.0	4	12.1	4	15.4	0.55	
F6	4	3	4	3	4	3	4.0	3.0	7.0	8.1	3	9.1	2	7.7	0.48	
F7	6	2	6	2	6	2	6.0	2.0	8.0	9.2	4	12.1	3	11.5	0.39	
F8	9	8	9	8	9	8	9.0	8.0	17.0	19.6	7	21.2	8	30.8	0.38	
F9	3	4	3	4	3	4	3.0	4.0	7.0	8.1	4	12.1	2	7.7	0.41	
Totals	43	42	47	42	47	42	44.6	42.0	86.6	100.0	33	100.0	26	100.0		

- Looks comprehensive, what else?

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

36



Prioritization for SPL

- Additional elements for SPL platform requirements prioritization
 - Relative importance of products
 - Release timeline for product variants
 - Priority is a function of time!
 - Frequency of a specific requirement across variants
 - Alignment of requirements with strategic goals
 - as defined in the SPL architecture
- The maturity of the SPL determines the approach
 - Initial: Select one product and drive it to success
 - Evolution: Maximize ROI

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

37



Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- REM roles and organization for product lines
- Portfolio planning and requirements prioritization
- **Product line architecture**
- Documenting product line requirements
- Establishing the right communication structures
- Tool support for REM in product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

38



Architectural criteria facilitate REM decisions

- What parts of the system does a requirement affect?
- Does a requirement belong to platform or product?

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

39



Document architecture to enable effective REM

- Document domain, platform and product architecture explicitly
 - SPL objectives
 - Architecture models
 - Architectural scenarios
- Generate architecture documentation from development artefacts
- Ease access to information about architectural dependencies and constraints

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

40



Enable effective interaction between REM and architecture

- Task and role definition
 - Strong role of the domain analyst
- Architecture review board and change control board
- Negotiation processes
- Decision policies
- Persistent documentation of decisions
- Easy access to documentation

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

41



Toolbox: Product line architecture

- Architecture models and scenarios
- Efficient generation of architecture documentation from implementation artifacts
- Strong role of the domain analyst
- Architecture review board and change control board with clearly defined tasks and interactions

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

42



Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- REM roles and organization for product lines
- Portfolio planning and requirements prioritization
- Product line architecture
- **Documenting product line requirements**
- Establishing the right communication structures
- Tool support for REM in product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

43



Characteristics of SPL requirements

- Large collection of long-living requirements
- Platform vs. application requirements
- Several different types of requirements attributes and interrelations (e.g., status)
- Feature associations and interactions
- Various different uses, users, views, and operations on requirements

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

44



Two levels of requirements documentation

- Macro level:
Elements and structure of requirements documentation
- Micro level:
Documentation of each individual requirement

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

45



Macro level: Central requirements base plus supplementary documents

- Central requirements base
 - Usually using REM tool
- Supplementary documentation for information types not supported by REM tool
 - Graphics, formal models, document structures etc.
 - Complex, SPL-specific requirements notations
 - Informative documents for specific target groups (should be generated from requirements base)
- Conceptual domain model
 - Definition of domain and platform architecture on requirements level

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

46



Lessons learned for requirements documentation

- Define meta model for requirements documentation
- Define stakeholder-specific views on requirements base
- Generate view-specific documents from the requirements base
- Ensure documentation and document quality

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

47



Techniques for documenting SPL requirements

- Requirements base
 - Usually part of specialized REM tool
 - Note: Little SPL support available
- Documents
 - Text documents
 - Spreadsheets
- Models
 - UML models
 - Feature and variability models
 - Domain model

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

48



Micro level: Defining individual SPL requirements

- Each requirement as an identifiable entity (i.e., data record)
- Requirements status
- Temporal relations
- Variants of requirements
- Versions of requirements

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

49



Different kinds of requirements status

- Content completion status
 - Outline, draft, complete
- Decision making status
 - Work in progress, ready for review, approved, ...
 - Individual (product) requirement versus mapping to SPL architecture and release schedule

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

50



Ensure documentation and document quality

- Mandatory templates for requirements base and requirements documents
- Usage guidelines and checklists for requirements base and documents
- Quality gates for systematic requirements definition and document completion
- Clearly defined central responsibilities for quality assurance and approval of requirements definitions

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

51



Toolbox: Requirements documentation

- Central requirements base
- Stakeholder-specific requirements views
- Conceptual domain model
- Strong and customized tool support
 - Typed links and attributes, transformation, reporting
- Version management on requirements level
- Requirements status tracking
- Change management for requirements
- Explicit document management
- Ensure documentation and document quality

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

52

Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- REM roles and organization for product lines
- Portfolio planning and requirements prioritization
- Product line architecture
- Documenting product line requirements
- **Establishing the right communication structures**
- Tool support for REM in product line development

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

53

REM challenges often related to communication

Category	Challenges
Organization and Management	Justification of the platform approach as a process model by a cost / benefit-analysis
	Independent platform team
	Difficult cooperation between platform and product development teams
	Proof of justification of the platform team
	High communication overhead
	Poor configuration management
Requirements engineering	Influence of the architecture on requirements negotiation is not taken into account
	No description of variability for domain analysis
	Missing domain analysis and domain description
	Discussions on design and not on requirements level
	No explicit requirements process
	Missing tool support
Product- vs. platform-specific	Sequence of integrating requirements into the platform
	No explicit prioritization of requirements
	Realization of platform requirements in products
	Strong influence of the pilot client
Architecture	No use of the architectural advantages
	Poor description of the generic architecture

Source: A. Brä, G. Heller, I. John, T. von der Maßen, K. Müller, K. Schmidt:
Product Line Engineering: The State of the Practice. IEEE Software, 20(6), 2003.

54



REM communication needs

- Elicit and negotiate requirements from users and product sponsors
- Negotiate requirements between products and platform
- Bring products and platform development in touch with user viewpoints
- Conduct decision processes
- Blend knowledge and experience from different sources
- Pertain and use knowledge and decisions

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

55



Elements of communication structures

- Participants
- Information flow
- Organization
- Process
- Tools

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

56



Integrated REM decision and negotiation process

- Involve all relevant roles
 - Product sponsors and users
 - Program and project managers
 - Requirements engineers
 - System architects
- Consider product/platform mapping of requirements early
- Enforce decision making and negotiation on requirements level (instead of design level)

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

57



Communication infrastructure

- Define policies for decision making, information flows, and documentation
 - Clear role definition
 - Decision boards and meeting structures
 - Distribution lists
 - Notification mechanisms
 - Archiving
- Support persistent documentation
 - Discussion groups or Wikis
 - Best integrated with central requirements base (REM tool)

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

58



Environment for effective communication

- Make product sponsors aware of SPL concepts and decisions
- Foster an open communication culture
 - Workshops and meetings
 - Bridge organizational boundaries
- Review communication infrastructure regularly
- Review knowledge management and qualification measures regularly

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

59



Toolbox: Communication structure

- Comprehensive role involvement
 - E.g., integrate architects into customer negotiation
- Architecture review board
- Change control board
- Active and frequent communication
- Job rotation
- SW tools that support transparent group communication and group decision making
- Architecture models and scenarios

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

60



Agenda: Strong REM practices for SPL development

- REM in the realm of project and program management
- REM roles and organization for product lines
- Portfolio planning and requirements prioritization
- Product line architecture
- Documenting product line requirements
- Establishing the right communication structures
- **Tool support for REM in product line development**

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

61



Tools for REM in SPLs (1/2)

- Poor tool support known for quite some time
 - e.g.: Raised on SPLC 2002
- Major tool pain points are
 - Limited scalability functionality
 - Explicit variability support
 - Role-based working environments (views)
 - Configuration and change management
- Most companies implement their own solutions

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

62

Tools for REM in SPLs (2/2)

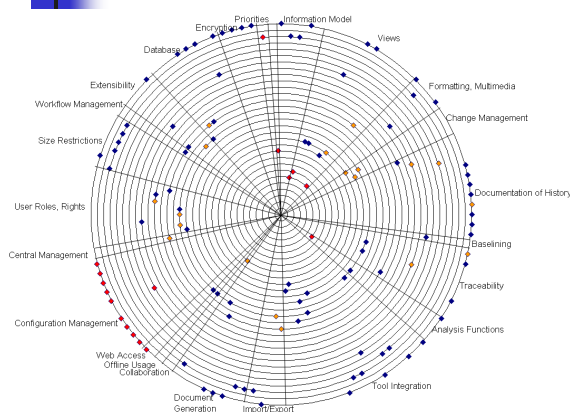
- Only few information about the needs is available
- A recent study details requirements along with their importance for SPLs
- Five REM tools were evaluated using those SPL requirements
- Conclusion:
 - Current commercial tools do not sufficiently support REM for SPLs

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

63

Applying REM tools in product line development



Properties of RM tools particularly important in product line contexts

- Explicit modeling concepts for requirements representation
- Changeability and adaptability of requirements representation and functionality
- Graphical presentation and visualization of requirements and their interrelations
- Collaborative work during requirements management
- Management of multiplicity in requirements definition (multiple products and projects, as well as interrelations between them)

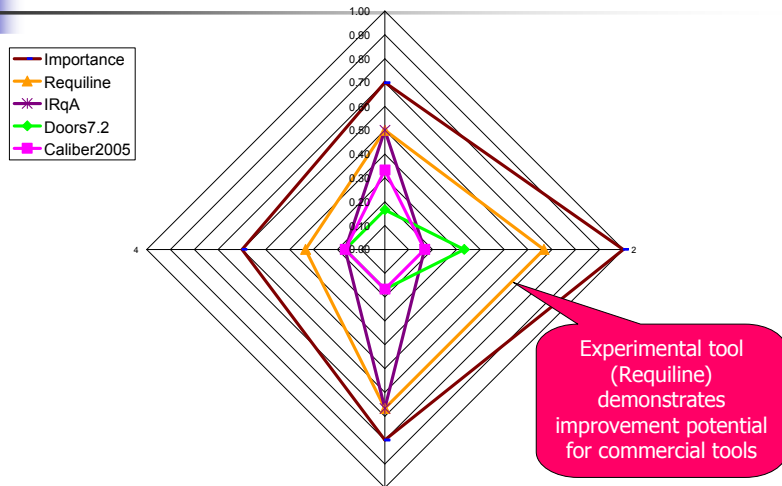
Source: Report of the GI Working Group on Requirements Management Tools for Software Product Lines, RWTH Aachen, 2006 (to appear).

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

64

REM tool sample comparison



September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

65

Toolbox: REM tools for SPL

- Deploy REM tools for efficiency purpose
- Establish own weighted criteria list and select tool based on own criteria

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

66



Agenda

- Introduction
- Fundamentals of SPL and REM
- Strong REM practices for SPL development
- **Growing your REM practices**
 - Establish and improve REM
 - Be aware of the power game
- New promises: Agile techniques
- Summary

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

67



Establishing and improving REM in a SPL context is particularly challenging

- Often a serious change to the organization
- May interfere significantly with existing power balance within the organization
- Goes far beyond R&D
- Impacts processes beyond REM
- Changes organization, process, and tooling simultaneously

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

68



Systematic change management highly recommended

- For establishing and improving REM in a SPL context, systematic change management is often much more important than in other process change efforts
- Plan and prepare change systematically
 - See: Rising, Mans: Fearless Change
- Based on "best practice"
 - See: Clements, Northrop: Software Product Lines
- Involve stakeholders early
- Start as small as possible (and as big as necessary ...)
- Continuously extend and improve

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

69



Be aware of the power game

- REM and SPL changes may interfere significantly with the existing power balance and cause strong opposition
- Be aware of power situation and consider it throughout all change activities
- Make explicit and visualize power system
 - Existing strategic goals
 - Sociographs
 - Objectives maps

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

70



Agenda

- Introduction
- Fundamentals of SPL and REM
- Strong REM practices for SPL development
- Growing your REM practices
- **New promises: Agile techniques**
- Summary

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

71



Agile development

- Promises of agile development
 - Deliver value more frequently
 - Deliver value which is needed by customer
- Techniques
 - Iterative development
 - Customer is part of the team
 - Embrace change
 - ...
- Principle: Fast feedback

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

72



Agile SPL

- SPL problem statement
 - Platform moves slowly – long release cycles
 - Disconnect between application and platform
- Establish processes in a way that they are able to incorporate unplanned feedback
 - Many traditional processes ask for feedback, but aren't really able to reflect it in a timely manner
- Agile techniques which will help SPL development
 - Develop the platform evolutionary in iterations
 - Involve application teams
 - Validate partial results with application teams

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

73



Agile SPL REM storyboard (1/2)

- Establish goals for a specific release
- Create development plan with fixed iterations
- Develop **themes** or work packages for iterations
 - No detailed requirements!

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

74



Agile SPL REM storyboard (2/2)

- Run the planning for the first iteration
 - Get together with consumers of this iteration
 - Ask what they like to consume by the end of the first iteration
- Run the planning for iteration n+1
 - Run retrospective of iteration n
 - Decide on process improvement
 - Develop and select next set of requirements with consumers of iteration n+1

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

75



Agile cautions

- The discipline is still pretty new, consider step-wise improvement
- Agile platform development process must fit into the overall development process
- SPL documentation requirements may not be compatible with agile approaches

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

76



Agenda

- Introduction
- Fundamentals of SPL and REM
- Strong REM practices for SPL development
- Growing your REM practices
- New promises: Agile techniques
- **Summary**

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

77



Conclusions

- REM in SPL is REM²
- Practices for mastering REM in SPL exist and need always be customized
- Design REM infrastructure before you start with SPL
- Continuously learn and improve to achieve higher levels of REM practices

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

78



Our Top Five REM Practices

- Strong Domain Analyst role
- Regular communication between platform and product teams
- Requirements base using REM tool
- Portfolio planning & architecture models to facilitate requirements decisions
- Requirements change management

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

79



Agenda

- Introduction
- Fundamentals of SPL and REM
- Strong REM practices for SPL development
- Growing your REM practices
- New promises: Agile techniques
- Summary
- **Appendix: Recommended information sources**

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

80



References – Books and papers on SPL and architecture

- P. Clements and L. Northop: Software Product Lines: Practices and Patterns. Addison-Wesley, 2002.
- K. Pohl et al.: Software Product Line Engineering: Foundations, Principles, and Techniques. Springer, 2005.
- Böckle et al.: Software Produktlinien: Methoden, Einführung und Praxis. dpunkt.verlag, 2004. (in German)
- D. Dikel et al.: Software Architecture: Organizational Principles and Patterns. Prentice Hall PTR, 2001.
- A. Birk et al.: Product Line Engineering: The State of the Practice. IEEE Software, 20(6), 2003.
- P. Clements et al.: Documenting Software Architectures: Views and Beyond. Addison-Wesley, 2003.

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

81



References – Books and papers on REM and Agility

- S. Robertson and J. Robertson: Mastering the requirements process (2nd Ed.), Addison-Wesley, 2006.
- K. Wiegers: Software Requirements (2nd Ed.), Microsoft Press, 2003.
- K. Wiegers: More about Software Requirements, Microsoft Press, 2006.
- S. Bayer: Practicing agile requirements, Cutter Consortium, 2002.
- C. Rupp: Post agilum, Java magazin, 2005. (in German)

September 11, 2006

RE 2006, Minneapolis/St. Paul, MN, USA

82



References – Websites, conferences

- SEI: Software product lines
 - <http://www.sei.cmu.edu/productlines>
- SPLC conferences
 - <http://www.splc.net/>
- Community page from Charles Krueger
 - <http://www.softwareproductlines.com>
- CAFE (EU-Project)
 - <http://www.esi.es/Cafe>
- ESAPS (EU-Project)
 - <http://www.esi.es/esaps>
- VRAPS (book website; templates)
 - <http://www.vraps.com>