Model-based Engineering in Medical Device Development – Issues & Solutions

INCOSE International Workshop Phoenix, AZ, USA January 29, 2011

Copyright © Siemens AG 2011. All rights reserved.
Contents

- Goals
- Brief look on Siemens Healthcare
- SYNGO products
- Business challenges
- Model-based Engineering: Issues & Solutions
- Recommendations
- Further Information

Copyright © Siemens AG 2011. All rights reserved.
Contents

- Goals
- Brief look on Siemens Healthcare
- SYNGO products
- Business challenges
- Model-based Engineering: Issues & Solutions
- Results and Summary
- Further Information
Goals of this Talk

- Discuss the experiences using engineering models (in different phases of product development of a next generation imaging platform)

- Identify and share needs and requirements with colleagues from diverse industries to advance product development tooling
in-vivo diagnostics (imaging)

X-Ray
Computed Tomography
Magnetic Resonance
Molecular Imaging
Ultrasound
Oncology

in-vitro diagnostics (laboratory systems)

Immunodiagnostics
Nucleid Acid Testing
Clinical Chemistry
Hematology
Urin Analysis
Lab Automation
Near Patient Testing
Siemens Healthcare
Development of Sales and Employee Numbers

Sales according to region¹)

- Germany: 9%
- Asia & Australia: 17%
- Europe (without Germany): 31%
- Americas: 43%

Employees according to region²)

- Germany: 23%
- Asia & Australia: 17%
- Europe (without Germany): 19%
- Americas: 41%

¹) Basis: FY 2009 acc. To customer locations. ²) Figures worldwide as of Sept. 30, 2009

Copyright © Siemens AG 2011. All rights reserved.
Contents

- Goals
- Brief look on Siemens Healthcare
- **SYNGO products**
- Business challenges
- Model-based Engineering: Issues & Solutions
- Results and Summary
- Further Information
SYNGO Products

- Clinical Imaging Applications
  - syngo.via

- Radiology Image Management (PACS)
  - syngo.plaza

- Radiology Information Systems (RIS)

- Cardiology IT Systems

- Enterprise Image and Information Access

- Multi-Site and Regional Solutions

- Services

Copyright © Siemens AG 2011. All rights reserved.
synigo.via Product

Project Summary

**synigo.via**: Next generation imaging software covering the entire reading process

**Project data:**
- > 5,000 single product requirements
- 7+ million lines of code C++/C#
- Several hundred developers in 5 locations
- Clinical applications for Radiology, PACS, X-Ray, CT, MI, Oncology, Particle Therapy and MR.
Contents

- Goals
- Brief look on Siemens and Vector
- SYNGO products
- Business challenges
- Model Management: Issues & Solutions
- Results and Summary
- Further Information
Business Challenges in Medical Device Industry

- Industry regulations
- Cycle time down
- Costs down
- Complexity growth

Need to industrialize product development

Focus of this talk

Application Life-cycle Management Tools

Platforms

Globally Integrated Systems

Model-based Engineering

Copyright © Siemens AG 2011. All rights reserved.
Disclaimer:
The content discussed in this presentation needs to be considered as work in progress.
## Industry Issues and Pain Points

<table>
<thead>
<tr>
<th>Pain Points</th>
<th>Business Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>① Product structure intransparent, domain model partially incomplete</td>
<td>Technology-driven product platform, no link to business drivers</td>
</tr>
<tr>
<td></td>
<td>- Opaque relationship between problem &amp; solution space</td>
</tr>
<tr>
<td></td>
<td>- Re-scoping sessions w/ customers on basis of 50+ specs.</td>
</tr>
<tr>
<td>② Ambiguity and lack of accuracy of specifications</td>
<td>Textual-based specifications are subject to interpretation</td>
</tr>
<tr>
<td></td>
<td>- Textual use case descriptions work only for smaller projects</td>
</tr>
<tr>
<td></td>
<td>- Natural language subject to interpretation, inconsistent, incomplete</td>
</tr>
<tr>
<td>③ Controlling architectural complexity</td>
<td>Redundancy of architecture components due to lack of understanding of problem- and solution space</td>
</tr>
<tr>
<td></td>
<td>- Business needs not consistently linked to features</td>
</tr>
<tr>
<td></td>
<td>- Too much variability in software architecture</td>
</tr>
<tr>
<td>④ Lack of V&amp;V efficiency</td>
<td>Test specifications in natural language are mostly executed in a manual way only</td>
</tr>
<tr>
<td></td>
<td>- Test cases manually created</td>
</tr>
<tr>
<td></td>
<td>- 5000+ pages of requirements</td>
</tr>
</tbody>
</table>
Pain point 1: Product Structure Intransparent, Domain Model Partially Incomplete

Selected issues:
- Opaque relations between problem- & solution space
- Re-scoping sessions w/ customer on basis of 50+ engineering specs.

Solutions:
A. Requirements Engineering Meta-model
B. Feature Model
Solution A: Requirements Engineering Meta Model – Benefits & Tool Requirements (1)

**Problem Space**

- **Business Plan**
  - Market Analysis, SW Platform Roadmap, Business Case, etc.
- **Charm DB**
  - Stakeholder Request
- **MRS**
  - SW Feature
- **SW Feature**
  - Market Requirement
- **SWRS**
  - SW Req. (Problem)
  - SW Req. (Solution)
- **Analysis**
  - Use Case Specifications, Load Profiles, Concept Papers

**Characteristics:**
- Provides needed artifacts, their attributes and relationships to each other.
- Using meta-model, it is also possible to prescribe the way how / which data are captured.

**Benefits:**
- Lean artefact infrastructure, no redundancy
- Guidance for engineering tasks with structured input
- Established link between business drivers, requirements, design and test
A Feature represents a characteristic of a product which provides a business value and supports purchase decisions [...]. A Feature structure requirements in a meaningful way and is no specification in itself (only "container").

A Feature Model is a hierarchical "tree" to describe the structure, dependencies, commonalities and variabilities of Features within a product or product line (e.g. SW Platform, Finished Medical Device).

Requirements specify the Feature. They represent the functionality already implemented within the product and functionality that is planned for future versions. Depending on the project phase requirements are less or more detailed (e.g. Market Req., SW Req.).

Stakeholder Requests are wishes towards an existing or future product. Stakeholder Requests (SR) are gathered from various sources (e.g. end customer, business units) and may have an impact on different levels and phases of the product lifecycle.
Solution A: Requirements Engineering Meta Model – Concepts (3)

Problem Space

- Business Plan
  - Market Analysis, Roadmap, Business Case, Version Goal

- SW Feature
  - Market Requirement

Solution Space

- DB
  - Stakeholder Request

- Analysis
  - Use Case Specifications, Load Profiles, Concept Papers, etc.

- Architecture
  - SW Req. (Problem)
  - SW Req. (Solution)

- SWRS
  - Architectural Build. Block
  - Design Description

Copyright © Siemens AG 2011. All rights reserved.
Solution A: Requirements Engineering Meta Model – Concepts (4)

**Problem Space**

- **Business Plan**
  - Market Analysis, Roadmap, Business Case, Version Goal

- **MRS**
  - Market Requirement
  - SW Feature

- **SWRS**
  - SW Req. (Problem)
  - SW Req. (Solution)

**Solution Space**

- **DB**
  - Stakeholder Request

- **Analysis**
  - Use Case Specifications, Load Profiles, Concept Papers, etc.

- **Architecture**
  - CDS
    - Architectural Build. Block
    - Design Description

**Example: Feature - „Demo of Images“**

**Market Requirement:** As a Clinician, I want to mark images to support presentations for a medical conference.
Solution A: Requirements Engineering Meta Model – Concepts (5)

Problem Space

Business Plan
- Market Analysis, Roadmap, Business Case, Version Goal

MRS
- SW Feature
- Market Requirement

SWRS
- SW Req. (Problem)
- SW Req. (Solution)

Solution Space

DB
- Stakeholder Request

Analysis
- Use Case Specifications, Load Profiles, Concept Papers, etc.

Architecture
- Architectural Build. Block
- Design Description

Use Case: „Demo of Images“ with activity diagram to derive software requirements.
Solution B: Feature Model

Characteristics:
- Hierarchical tree to describe the structure, dependencies and commonalities
- Lays out the basics for variant management and impact analysis

Benefits:
- Higher level abstraction of grouping of requirements into sellable units: From 5,000 product requirements to 800+ features (factor ~ 6)
- Visual domain model for healthcare workflows (tree & graphical)
- Reduction in time to understand aspects of the system
Solution B: Feature Model & Its Relations (1)

How can I manage huge amounts of Requirements?

How do I scope a product version efficiently?
Solution B: Feature Model & Its Relations (2)

Stakeholder Requests

Application Use Cases

Architecture Model

How do I manage stakeholder requests from different businesses most effectively?
Pain point 2: Ambiguity and Lack of Accuracy of Specifications

Issues:
- Textual use case descriptions work only for smaller projects < ~ 100 requirements
- Natural language subject to interpretation, usually inconsistent, incomplete with incorrect version (and conflicting)

Root causes:
- Textual requirements engineering do not scale for platform projects
- Missing versioning
- No direct access to single requirements
- Lack of product structure
- Inconsistently executed change management process

Solutions:
C. Application Use Cases
Solution C: Graphical Modeling of Clinical Workflows (1)

**Characteristics:**
- Used to describe clinical workflows that consist of a collection of steps in a defined sequence together with accompanying specification of pre-/post-conditions, business rules, performance aspects, etc.

**Benefits:**
- Increase expressiveness of clinical workflows to describe dynamic behaviors of clinical workflows
- Early analysis of stakeholder requests from customers
- Improved impact analysis of change requests
- Joint modeling sessions to describe the needs from the customer's point of view
Solution C: Graphical Modeling of Clinical Workflows (2)

Stakeholder Requests
- Feature 1
- Feature 2
- Feature 3
- Feature 4
- Feature 5
- Feature 6
- Feature 7
- Feature 8
- CT-Obility
- IGR... Export of tabular results
- IGR... Progress visualization and information
- IGR... Scorable tabular results

Application Use Cases
- How can I analyze a feature or a stakeholder request?
Pain point 3: Controlling Architectural Complexity

Selected issues:

- Business needs not consistently linked to features/requirements; dependencies between features not easily visible
- Too much variability in software architecture

Solutions:

D. Architecture Model Mapping
Solution D: Architecture Model Mapping

Characteristics:
- Identifies links between features and their implementation
- Explicit modeling of variability in the architecture

Benefits:
- Architectural decisions motivated by features and product-line variability
- Enabling reduction of architectural complexity
- Support impact analysis for (de-) scoping sessions
- Early identification of architectural risks
- Improved accuracy of early effort estimates
- Reduction of number of scoping sessions
Solution D: Architecture Model Mapping

Which component implements which feature?
Pain point 4: Need to Increase V&V Efficiency (*)

Issues:
- Test cases partially manually generated from textual use cases
- Cycle times for system test too long
- Compliance requirements QSR 21 CFR §820.30, ISO 13485, EU MDD 93/42

Solutions:
- E. Model-based System Testing

(*) Not yet piloted
Solution E: Model-based System Test (2)

Benefits:
- Early identification of requirements defects through validation by testers
- Effort reduction for test (cycle time, cost ~ -30%) and increase of test coverage
- Decrease number of defects
- Model-based testing is highly structured, reproducible and efficient
- Increase reuse of development artifacts
- Quicker impact analysis by parsing model for late requirements changes
Solution E: Model-based System Test

Stakeholder Requests

Architecture Model

Application Use Cases

Which application use case is covered by which test case?
Benefits from Model-based Engineering

- Precise calculation of % requirements complete
- Assurance for complete traceability
- Processes are followed through workflow guidance
- Understand which feature relates to which component
- More precise estimates through understanding of feature complexity
- Ensure testability of requirements
- Generate test cases from requirements models
- Automatically identify test cases which failed
- Reduction in engineering costs
- Increased compliance with FDA
- Higher engineering productivity through integrated engineering workflows
- Know which features need to be elicited
- Quick understanding of requirements through navigability in models
- Reduction of review times / efforts with customer
- Understand impact of requirements / change requests
- Improved requirements quality
- Semi-automated traceability
- Automated model checking of requirements
Major Changes to Development Approach
Model-based Engineering

- Requirements Engineering Meta-Model
- Feature Model
- Graphical Modeling of Clinical Workflows
- Architecture Model Mapping
- (Model-based System Test)
Contents

- Goals
- Brief look on Siemens and Vector
- SYNGO products
- Business challenges
- Model-based Engineering: Issues & Solutions
- Results and Summary
- Further Information
Potential Business Impact using Model-based Engineering

- Model-based engineering approach can cut development cycle-time by ~20% (*)

- Reduction of review effort by ~30% due to feature reviews (*)

- Model-based testing can cut effort by ~30% while increasing test coverage (*)

(*) Data from large-scale industry projects
Key Take-Aways for Model-based Engineering

- Seamless model-driven engineering is only partially tool-supported; the biggest gap remains in requirements engineering.
- Tool vendors need to stronger leverage the experience of leading development organizations and uptake it into technology roadmaps.
- Acceptance of model-based engineering is a huge organizational change management endeavor, only 10% of organizations have already gained practical experience.
- Continuous assessment and verification of business benefits for model-based engineering is a must to maintain sponsorship from management.
Thank you for your attention!
Contents

- Goals
- Brief look on Siemens and Vector
- SYNGO products
- Business challenges
- Model-based Engineering: Issues & Solutions
- Results and Summary
- Further Information
References


- Siemens Healthcare Imaging & Therapy SYNGO: Optimized Requirements Engineering Approach, Siemens Healthcare, 2010

- Siemens Healthcare Imaging & Therapy SYNGO: Microsoft Team Foundation Server Business Case, May 2010


- Dehla Sokenou, Erwin Tratar: Modell-basierte Software Entwicklung – Vom Requirements Engineering bis zum Testen


- US Food & Drug Administration, Design Control Guidance for Medical Device Manufacturers; March 11, 1997

Documented Experiences and Best Practices from Various Industry Projects

Software & Systems Requirements Engineering: In Practice
2009
McGrawHill

Link to web site
McGrawHill
Arnold Rudorfer
Director Software Initiative and Process Improvement
Siemens Healthcare AG

Hartmannstrasse 16
D-91052 Erlangen

Phone: +49 9131 – 82 2299
Fax: +49 9131 – 84 8691
Mobile: +49 174 1537825

Email: arnold.rudorfer@siemens.com