How Reference Architectures support the evolution of Product Families; the Darwin research project

by Gerrit Muller  Embedded Systems Institute
e-mail: gaudisite@gmail.com
www.gaudisite.nl

Abstract

TBD

Distribution

This article or presentation is written as part of the Gaudi project. The Gaudi project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

This work has been carried out as part of the Darwin project under the responsibility of the Embedded Systems Institute. This project is partially supported by the Netherlands Ministry of Economic Affairs under the BSIK program.
High Level Problem Statement

Installed Base Business
- costly
- high effort
- diversity and # of configurations

Life Cycle Management
- costly
- high effort
- too late

Development efficiency
- costly
- high effort
- too late

Innovation rate
- too low
- too late

see next slides
The Innovation Challenge

Challenge: how to apply change locally for exploration of potential value and feasibility?

Postulate 1: for effective exploration the following properties must be maintained
- patient throughput
- system responsiveness
- image quality
- safety
- reliability

Postulate 2: a system architecture that supports this level of exploration also supports the next phases of innovation: scaling-up and engineering

Postulate 3: a system architecture that supports this level of exploration also supports life cycle business over many generations
evolvability problem statement

exploration is difficult

reliable realization is difficult

engineering is difficult

too much
time, effort, cost
from idea to tryout
too much
and unpredictable
development
time, effort, cost
from tryout to realization
some new features
late relative to competition
too much
material and labor cost

innovation life cycle

1
10
100

tryout
exploration of innovative features
scale up
for clinical use
scale up
for volume sales

volume
time

How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
DERAproblemStatement

Gerrit Muller
### Evolvability Problem Analysis

<table>
<thead>
<tr>
<th>Problems</th>
<th>Observed Causes</th>
<th>Suspected More Specific Root Causes</th>
</tr>
</thead>
</table>
| exploration is difficult                                                | observed causes
| too much time, effort, cost from idea to tryout                        | 25 years of historical growth                                                  | coupling (dependencies)
| reliable realization is difficult                                       | lack of overview large amount of detailed documentation                        | higher than needed
| too much and unpredictable development time, effort, cost              | size and complexity of realization                                             | ineffective structure (decomposition, interfaces)
| from tryout to realization                                              | size and complexity of organization                                            | insufficient underpinning of decisions by value and cost
| engineering is difficult                                                | inherent complexity of system and context                                      | unbalance in core/key/base
| some new features late relative to competition too much material and    | human and cultural factors high level of expertise conservatism                | diversity of configurations
| labor cost                                                               |                                                                                |                                                                                |

How Reference Architectures support the evolution of Product Families

Gerrit Muller

version: 0.4
June 23, 2016
DERAproblemAnalysis
Darwin Project Goal

- scientifically sound
- suitable for PhD
- specific methods, techniques and patterns
- based on modeling and Reference Architectures
- faster to market
- less effort
- more predictable
- to improve the evolvability
- of product families
- within industrial constraints
- and while maintaining other qualities
- market response to anticipated and unexpected changes
- very relevant for MR
- also relevant for others
- (partially) validated
- diverse products
- installed base diversity
- patient throughput
- system responsiveness
- image quality
- safety
- reliability
- people, process, project duration, and cost
- scientifically sound
- suitable for PhD
- specific methods, techniques and patterns
- based on modeling and Reference Architectures
- faster to market
- less effort
- more predictable
- to improve the evolvability
- of product families
- within industrial constraints
- and while maintaining other qualities
- market response to anticipated and unexpected changes
- very relevant for MR
- also relevant for others
- (partially) validated
- diverse products
- installed base diversity
- patient throughput
- system responsiveness
- image quality
- safety
- reliability
- people, process, project duration, and cost

How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
DERAprojectGoal

6     Gerrit Muller
Darwin Research Model: Industry as Laboratory

- **source of inspiration**
  - Philips MR scanner

- **application playground**
  - industry

- **challenge problems**
  - apply new engineering methods

- **research**
  - evaluate results
  - improve

- **hypothesis**

How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
Sources of Change

customer context
- humans
- other systems
- legislation
- reimbursement

technical architecture
- clinical applications
- workflow applications
- domain specific technology
- generic technology

business architecture
- competition
- organization
- business model
Sources of Change

customer context
- humans
- other systems
- legislation
- reimbursement
- USA

business architecture
- clinical applications
- workflow applications
- competition
- organization
- business model
- PMW
- PII

technical architecture
- RF coils
- gradient amplifier
- domain specific technology
- generic technology
- Windows Vista
- PCI-X
- database

other systems
- PACS
- RIS

Reimbursement
- legislative

Other examples
- USA
- business model
## Darwin Research Questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Related Research Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>How to transform into an evolvable product family architecture?</strong></td>
<td>value analysis, e.g. real option roadmapping</td>
</tr>
<tr>
<td><strong>How to support decision making?</strong></td>
<td>reference architecture</td>
</tr>
<tr>
<td><strong>business wise</strong></td>
<td>physical models, functional models, budgeting, figures-of-merit, state-diagrams, time-lines</td>
</tr>
<tr>
<td><strong>technological</strong></td>
<td>repository meta-data analysis</td>
</tr>
<tr>
<td><strong>by visualization</strong></td>
<td>dynamic dependency analysis</td>
</tr>
<tr>
<td><strong>by high-level modeling</strong></td>
<td>semantic analysis</td>
</tr>
<tr>
<td><strong>How to create overview?</strong></td>
<td>reference architecture</td>
</tr>
<tr>
<td><strong>by visualization</strong></td>
<td>physical models, functional models, qualities, behavior models</td>
</tr>
<tr>
<td><strong>by high-level modeling</strong></td>
<td>clustering, structure, set-based design</td>
</tr>
<tr>
<td><strong>How to mine the realization for implicit know how?</strong></td>
<td></td>
</tr>
<tr>
<td><strong>What are practical guidelines?</strong></td>
<td></td>
</tr>
<tr>
<td><strong>for decomposition</strong></td>
<td></td>
</tr>
<tr>
<td><strong>for interface definition</strong></td>
<td></td>
</tr>
<tr>
<td><strong>What are patterns that support evolvability?</strong></td>
<td></td>
</tr>
</tbody>
</table>
RA = Business Arch. + Technical Arch. + Customer Context

How Reference Architectures support the evolution of Product Families

Gerrit Muller
1. Functional Decomposition

2. Construction Decomposition

3. Allocation

4. Infrastructure

5. Choice of integrating concepts
Decomposition and Interfaces

How Reference Architectures support the evolution of Product Families

Gerrit Muller
Interface much more than functions + parameters

black box (interface) level:
- protocols
- functions
- parameters
- formats

standard

behavior
characteristics

white box (implementation) level:
- protocols
- functions
- parameters
- formats

complying implementations

realizations
limitations
constraints
opportunities

behavior
characteristics

How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
ECMAblackWhite

Gerrit Muller
How Reference Architectures support the evolution of Product Families

ARMRintegrationAndDiversity
Distribution Scenario's

A: "Thin Servers"
- Clients
- Network
- Thin Servers

B: "All-in-one" Combi's
- All-in-one Combi's
- Network

C: "All-in-one" server
- Thin Clients
- Network
- All-in-one Server (PACS or HIS)

D: "Modular"
- Client
- Client
- Server
- Server

Legend:
- acquisition
- image handling
- information handling
- generic technology
Simplistic Architecture

How Reference Architectures support the evolution of Product Families

17 Gerrit Muller

version: 0.4
June 23, 2016
ARMRsimplisticArchitecture
Future Simplistic Architecture

How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
Gerrit Muller
How Reference Architectures support the evolution of Product Families

version: 0.4
June 23, 2016
ARMRcodeAssets

Gerrit Muller
Long Term Vision:
Reference Architecture +
Sample implementation
of Framework and
Components

Applications

Services

Computing Infrastructure

Domain Infrastructure

Framework

Reference Architecture

i.e. Internationalization

i.e. tunes, themes
Conclusion: Refactoring the Architecture is a must