Product Families and Generic Aspects

by Gerrit Muller    Buskerud University College

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

Most products fit in a larger family of products. The members of such a product family share a lot of functionality and features. It is attractive to share implementations, designs et cetera between those members to increase the efficiency of the entire company.

In practice many difficulties pop up when product developments become coupled, due to the partial developments which are shared. This article discusses the advantages and disadvantages of a family approach based on shared developments and provides some methods to increase the chance on success.
## Typical Examples of Generic Developments

<table>
<thead>
<tr>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common components</td>
</tr>
<tr>
<td>Standard design</td>
</tr>
<tr>
<td>Framework</td>
</tr>
<tr>
<td>Family architecture</td>
</tr>
<tr>
<td>Generic aspects, functions, or features</td>
</tr>
<tr>
<td>Reuse</td>
</tr>
<tr>
<td>Products (in project environment)</td>
</tr>
</tbody>
</table>
Claimed Advantages of Generic Developments

- Reduced time to market: building on shared components
- Reduced cost per function: build every function only once
- Improved quality
- Improved reliability: maturing realization
- Improved predictability
- Easier diversity management: modularity
- Increases uniformity
- Employees only have to understand one base system: less learning
- Larger purchasing power: economy of scale
- Means to consolidate knowledge
- Increase added value: not reinventing existing functionality
- Enables parallel developments of multiple products
- “Free” feature propagation: product-to-product or project-to-project
Experiences with reuse, from counterproductive to effective

**bad**
- longer time to market
- high investments
- lots of maintenance
- poor quality
- poor reliability
- diversity is opposed
- lot of know how required
- predictable too late
- dependability
- knowledge dilution
- lack of market focus
- interference
- but integration required

**good**
- reduced time to market
- reduced investment
- reduced (shared) maintenance cost
- improved quality
- improved reliability
- easier diversity management
- understanding of one base system
- improved predictability
- larger purchasing power
- means to consolidate knowledge
- increase added value
- enables parallel developments
- free feature propagation
### Successful examples of reuse

<table>
<thead>
<tr>
<th>Homogeneous Domain</th>
<th>Cath lab</th>
<th>MRI</th>
<th>Television</th>
<th>Waferstepper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Dominated</td>
<td>Car</td>
<td>Airplane</td>
<td>Shaver</td>
<td>Television</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Limited Scope</th>
<th>Audio Codec</th>
<th>Compression Library</th>
<th>Streaming Library</th>
</tr>
</thead>
</table>
Limits of successful reuse

struggle with integration/convergence with other domains

TV: digital networks and media
cath lab: US imaging, MRI

poor/slow response on paradigm shifts

TV: LCD screens
cath lab: image based acquisition control

software maintenance, configurations, integration, release

MRI: integration and test
wafersteppers: number of configurations

how to innovate?
Drivers for Generic Developments

Customer value
- application adaptability
- availability variations
- new features originating from different products
- timely availability
- reliability

Internal benefits
- asset creation
- increase economy of scale

Extrovert driver

Introvert driver

- reliability
- quality increase
- predictability
- availability integrated base product
- maturity

Design for configurability
Availability variations
Availability of accumulated feature set
Shared architectural framework
Maturity
Predictability
Availability integrated base product
Increase economy of scale
Asset creation
Internal benefits
Customer value
Extrovert driver
Introvert driver

Product Families and Generic Aspects
version: 2.3
March 6, 2013
GDdrivers
Granularity of generic developments shown in 2 dimensions
Modified Process Decomposition

- Customer-Oriented Process
- Policy and Planning Process
- Product Creation Process
- Shared Assets Creation Process
- People, Process, and Technology Management Process

Customer Roadmap
- Budget, plan
- Technology, Process, and People roadmaps
- Needs & Feedback
- Technical Product Doc.
- Product-related processes
- People Technology Process
- Support
- Product needs and feedback

Business Drivers
- Product roadmap, budget, plan
- Generic assets roadmap
- Budget, plan
- Needs & Feedback
- Generic assets
- People Technology Process
- People Technology Process
- People Technology Process
- Needs & Feedback

Product needs and feedback
- Info
- Order
- Product
- $\$

Support
- Needs & Feedback
- Technical Product Doc.
- Product-related processes
- People Technology Process
- People Technology Process
- People Technology Process
- Needs & Feedback

People, Process, and Technology Management Process
- People Technology Process
- People Technology Process
- People Technology Process
- People Technology Process
- People Technology Process
- Needs & Feedback
- Needs & Feedback
- Needs & Feedback

Product Families and Generic Aspects

version: 2.3
March 6, 2013
GDprocessDecompositionFamily

Gerrit Muller
Financial Viewpoint on Process Decomposition

- **Management**
- **Cashflow generation**
- **Tomorrow's Cashflow**
- **Strategic Asset Generation**
- **Assets**

Customer

Needs and Feedback

Product Needs and feedback

Material

Order

Information

Presales

Sales

Logistics

Production

Service

Support

Budget, plan

Product roadmap

ProductRelated processes

Product related processes

Product related documentation

Technical documentation

Feedback

Business drivers

Customer roadmap

Product roadmap

Technology, process and people roadmaps

Budgets

Product generics roadmap

Management

Information

Support

Tomorrow's Cashflow

Strategic Asset Generation

Assets

Product Families and Generic Aspects

version: 2.3

March 6, 2013

Gerrit Muller
Value and Feedback Flow

Customer-Oriented Process

Product Creation Process

Shared Assets Creation Process

People, Process, and Technology Management Process

Policy and Planning Process
Propagati
on Delay Plat
form Feature to Market

feature 1
feature 2
Platform integration
test
Release

Product integration
product feature 1
product feature 2
test
Release

Product feature 1
Product feature 2

feature 1
feature 2

Release

Product Families and Generic Aspects
version: 2.3
March 6, 2013
GDpropagationDelay
Sources of Failure in Generic Developments

Technical

• Too generic
• Innovation stops (stable interfaces)
• Vulnerability

Process/People/Organization

• Forced cooperation
• Time platform feature to market
• Unrealistic expectations
• Distance platform developer to customer
• No marketing ownership
• Bureaucratic process (no flexibility)
• New employees, knowledge dilution
• Underestimation of platform support
• Overstretching of product scope
• Nonmanagement, organizational scope increase
• Underestimation of integration
• Component/platform determines business policy
• Subcritical investment
Models for Generic Development

- **Policy and planning**
- **Customer oriented process** (sales, service, production)
- **Product Creation Process**
- **Create generic components**
- **People and technology management process**

**Customer**

- **Lead customer**
  - Direct feedback too specific?
- **Carrier product**
  - Product feedback product specific?
- **Platform**
  - Feedback problem too generic
- **Technology push**
  - No feedback