Six Sigma Revisited

EuroSPI² 2006 – 13th October 2006
Dr. Thomas Fehlmann
Euro Project Office AG
www.e-p-o.com
Agenda

- What is Six Sigma? How to apply it for Software?
- Defect Elimination with Six Sigma for Software
- Knowledge Management for Software Development
- The Convergence Factor
- Use of the Convergence Factor for Defects Elimination
Agenda

- What is Six Sigma? How to apply it for Software?
- Defect Elimination with Six Sigma for Software
- Knowledge Management for Software Development
- The Convergence Factor
- Use of the Convergence Factor for Defects Elimination
Statistical Process Control

Input

Process

Specifications
Checklists
Templates

Resources

Methods
Tools
ICT-Support

Output

\[ Cp = \frac{USL - LSL}{6\sigma} = \frac{1}{6} \]

Outside tolerance!
Statistical Process Control

Specifications
Checklists
Templates

Resources

Methods
Tools
ICT–Support

Input

Process

Output

\[ Cp = \frac{USL - LSL}{6\sigma} > 1 \]
1. Measure process capability $C_p$ per development phase
   - Select the critical parameters using Quality Function Deployment (QFD)
   - Analyze the transfer function between phases

2. Aggregate process capability to predict development success
   - Process response per phase
Six Sigma Value Chain with CPM
“Critical Functional Response”

System Requirements

Subsystem Requirements

Subassembly Requirements

Component Requirements

Manufacturing Process

Cp₁ = ok!

Cp₂ = ok!

Cp₃ = ok!

Cp₄ = ok!

System Response

Subsystem Response

Subassembly Response

Component Response

Manufacturing Specification

Not ok!

Wuff!
Transfer Functions

- **$Y = f(x)$**: 
  - Response to Business’s Needs is a function of the Use Case Solution $BN = f(UC)$

- **Select critical parameters**
  - Representative profiles for business and technical requirements
  - Given technical requirements profile $x = <x_1, \ldots, x_n>$, response profile to business needs is $Y = f(x) = <\varphi_1(x), \ldots, \varphi_m(x)>$. 

Use Case Solution (UC) $\rightarrow$ Business Needs (BN)

$Y = f(x)$

$X$
What a Software Development Process must be able to do

- Develop with known Process Capability
  - $C_p$ – Metrics
    - $USL = \text{Upper Specification Limit}$
    - $LSL = \text{Lower Specification Limit}$
    - $\delta = \text{Process Capability}$
    - $\sigma = \text{Standard Deviation}$

- Keep requirements variation under control
SW Product Deployment

- Realization of customer requirements
  - Comprehensive Quality Function Deployment (QFD)
  - Process Capability
    - Statistical samples
    - \( Cp = \? \):
      - How to get \( Cp \)?
      - From data collected?!
Agenda

- What is SPC? How to apply it for Software?
- **Defect Elimination with Six Sigma for Software**
- Knowledge Management for Software Development
- The Convergence Factor
- Use of the Convergence Factor for Defects Elimination
What is Software?

- Software is a service offered over time and space
  - Ready-to-call Knowledge
  - Ready-to-use Know-how
- Knowledge Acquisition
A-Defects and B-Defects

- **Wrong or missing requirements**
- **Customer's or user's needs not detected**
- **Bad communication**

- **Missed deadlines**
- **Wrong estimates**
- **Wrongly implemented requirements**
- **Missed specifications**
- **Program malfunction**
Critical Parameter Management for SW Requirements

- For each topic level, select representative requirements sample
  - Must be New, Unique and Difficult (NUD)!
- Establish knowledge transfer functions
  - From solution concept to business
  - From engineering to solution concept
  - Include for process and quality requirements
- Establish statistical relevance of the sample
The Deming Value Chain for SW

- **Decision**
  - Business Needs (BN) → Voice of the Customer (VoC)
  - BN → Competitive Analysis (LT)

- **Realization**
  - Use Cases (UC) → BN
  - UC → Decision

- **Enablers**
  - Technology (Arch) → BN
  - Critical to Quality (CtQ) → BN
  - Capability Maturity (CMM) → CtQ
  - SW Components (SC) → UC
  - Unit Tests (UT) → SC

- Critical to Quality (CtQ) → BN

- Technology (Arch) → BN

- Acceptance Test (AcT) → BN

- Application Test (AT) → UC

- Unit Tests (UT) → SC

- SW Components (SC) → UC

- Use Cases (UC) → UC

- Enablers

- Voice of the Customer (VoC)
  - #Opinion polls

- Competitive Analysis (LT)
  - #Market Share

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)
  - #FP

- Unit Tests (UT)
  - #Bugs

- Acceptance Test (AcT)
  - #Bugs

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)

- Business Needs (BN)

- Use Cases (UC)

- SW Components (SC)

- Unit Tests (UT)

- Acceptance Test (AcT)
What is SPC? How to apply it for Software?
Defect Elimination with Six Sigma for Software
Knowledge Management for Software Development
The Convergence Factor
Use of the Convergence Factor for Defects Elimination
Knowledge Management

- How to describe the goals?
  - Select representative NUD goals $g_1, g_2, \ldots$
  - Profile them according value for the customer

- What is the best solution?
  - Use domain know-how and business knowledge
  - Select the best solution approach \{s_1, s_3, s_7\} for each of the representative NUD goals
  - Associate solutions to the goals they support \{s_1, s_3, s_7\} $\times g_1$

- Use Ishikava–diagrams (Fishbone) for association!
The Origin of Quality Function Deployment

Strong Dependency: 9
Medium Dependency: 3
Weak Dependency: 1

\{s_1, s_3, s_7\} \leftarrow g_1

s_1: Solution Element
s_3: Solution Element
s_7: Solution Element

g_1: Business Need
Knowledge Terms

- The QFD matrices (= Ishikava diagrams) are knowledge terms that describe the cause/effect relationships of the software system under development.

- NUD requirements are a statistical sample from the potentially infinite set of all knowledge terms about the system (“all” requirements).
  - Statistical analysis for SW requirements explains how representative the samples are.

- Well-known techniques exist for NUD requirements selection:
  - Voice of The Customer
  - Analytical Hierarchical Process (AHP)
  - Quality Function Deployment
Agenda

- What is SPC? How to apply it for Software?
- Defect Elimination with Six Sigma for Software
- Knowledge Management for Software Development
- The Convergence Factor
- Use of the Convergence Factor for Defects Elimination
Quality Function Deployment for Cause/Effect Analysis

- Goal Profile

<table>
<thead>
<tr>
<th>Goals (WHAT)</th>
<th>Importance of selected goals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Goal profile is known</td>
</tr>
</tbody>
</table>

Goal profile is known
# Quality Function Deployment – Analysis of Influence Factors

## Cause/Effect matrix

<table>
<thead>
<tr>
<th>Goals (WHAT)</th>
<th>Influence Factors &amp; Handles (HOW)</th>
<th>Importance of selected goals (WHAT)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Goal 1" /></td>
<td><img src="image2" alt="Factor 1" /> <img src="image3" alt="Factor 2" /> <img src="image4" alt="Factor 3" /></td>
<td><img src="image5" alt="Importance 1" /></td>
</tr>
<tr>
<td><img src="image6" alt="Goal 2" /></td>
<td><img src="image2" alt="Factor 1" /> <img src="image3" alt="Factor 2" /> <img src="image4" alt="Factor 3" /></td>
<td><img src="image5" alt="Importance 2" /></td>
</tr>
<tr>
<td><img src="image7" alt="Goal 3" /></td>
<td><img src="image2" alt="Factor 1" /> <img src="image3" alt="Factor 2" /> <img src="image4" alt="Factor 3" /></td>
<td><img src="image5" alt="Importance 3" /></td>
</tr>
<tr>
<td><img src="image8" alt="Goal 4" /></td>
<td><img src="image2" alt="Factor 1" /> <img src="image3" alt="Factor 2" /> <img src="image4" alt="Factor 3" /></td>
<td><img src="image5" alt="Importance 4" /></td>
</tr>
</tbody>
</table>

- Red = strong relation
- Blue = medium relation
- Green = weak relation

**HOW MUCH:** Find profile for Influence Factors!

# Quality Function Deployment – Analysis of Influence Factors

## Cause/Effect matrix

<table>
<thead>
<tr>
<th>Goals (WHAT)</th>
<th>Influence Factors &amp; Handles (HOW)</th>
<th>Importance of selected goals (WHAT)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="9" alt="9" /> <img src="3" alt="3" /> <img src="9" alt="9" /></td>
<td><img src="9" alt="strong relation" /></td>
</tr>
<tr>
<td></td>
<td><img src="9" alt="9" /> <img src="9" alt="9" /> <img src="9" alt="9" /></td>
<td><img src="9" alt="medium relation" /></td>
</tr>
<tr>
<td></td>
<td><img src="9" alt="9" /> <img src="3" alt="3" /> <img src="9" alt="9" /></td>
<td><img src="1" alt="weak relation" /></td>
</tr>
<tr>
<td></td>
<td><img src="1" alt="1" /> <img src="1" alt="1" /> <img src="3" alt="3" /></td>
<td><img src="yellow" alt="Effects of Influence Factors" /></td>
</tr>
</tbody>
</table>

9 = strong relation
3 = medium relation
1 = weak relation
The Convergence Factor

\[ \kappa = \frac{|z - y|}{\sqrt{m}} = \sqrt{\frac{\sum_{j=1}^{m} (\zeta_j - \psi_j)^2}{m}} \]

\[ \kappa = \text{Minimum!} \]
The Convergence Factor

- The Convergence Factor measures the gap between
  - the importance of solution requirements, and
  - the response they effectuate on the goal topic

- Small gap means *robust* selection of solution components
  - The transfer function yields repeatable results!
  - Even if we don’t know much more than the three level cause/effect relationships
What is SPC? How to apply it for Software?
Defect Elimination with Six Sigma for Software
Knowledge Management for Software Development
The Convergence Factor
Use of the Convergence Factor for Defects Elimination
- Check requirements on all levels with QFD
- Check whether implemented profiles match planned profiles
  - For SW Components
  - For Use Cases Design
  - For Business Needs
Result is a product that meets business needs
  - E.g. gain market acceptance according to Lanchester Theory
- Then measure it using Functional Sizing #FP
Conclusion

- QFD provides Critical Parameter Management for software requirements
- Convergence Factor $K$ is quality metric for cause/effect analysis
  - Detects A-defects: Missing requirements!
- Works both for process and product quality
  - Deming Value Chain for Software