Dependable Systems

System Dependability Evaluation

Dr. Peter Tröger

Alison Cartlidge et al.. An Introductory Overview of ITIL® V3. ISBN 0-9551245-8-1
Dr. Ralf Kneuer, Capability Maturity Model Integration (CMMI).
Qualitative Dependability Investigation

• Different approaches that focus on structural (qualitative) system evaluation
  • Root cause analysis
    • Broad research / industrial topic about error diagnosis approaches
    • Specialized topic in quality methodologies
  • Development process investigation
    • Procedures for ensuring industry quality in production
    • Software development process
  • Organizational investigation
    • Non-technical influence factors on system reliability
Root Cause Analysis

• What caused the fault? - Starting point of dependability chain
  • Peeling back the layers
  • Must be performed systematically as an investigation
  • Establish sequence of events / timeline
Root Cause Analysis

- Class of approaches and algorithms for identifying the root cause of a problem
  - Iterative process of continuous improvement
  - Can be performed in reactive or pro-active fashion
- Applied in different fields, competing definitions and understandings
  - Accident analysis in safety-critical systems, e.g. aviation industry
  - Quality control in industrial manufacturing
  - Investigation of formally described business processes
  - Hardware failure analysis
- Risk management for huge hardware / software projects
- Assumes broken process or alterable cause (e.g. no physical faults)
RCA: 5 Whys

• Originally developed by Sakichi Toyoda for car manufacturing process
  • Limit number of dive-in‘s to avoid tracing the chain of causality

• Example: The Washington Monument was disintegrating.
  • Why? Use of harsh chemicals.
  • Why? To clean pigeon poop.
  • Why so many pigeons? They eat spiders and there are a lot of spiders at monument.
  • Why so many spiders? They eat gnats and lots of gnats at monument.
  • Why so many gnats? They are attracted to the light at dusk.
  • Solution: Turn on the lights at a later time.
5 Whys

- Limited steps ensure that investigator moves through layers
- Has danger of stopping too early at symptoms
- Results are not reproducible
- Investigators cannot consider reasons behind their own information - need teams
- Recommendation to use observation instead of deduction
  - Deduction does not allow proper ranking of answers
- Depends on completely honest answers and complete problem statement
- Only good for low risk problems
RCA: Why-Because Analysis

• Mainly intended for accident analysis (train, bus, airplane, security, industry)

• Central notion of a causal factor

• Output is directed acyclic Why-Because graph (WBG)
  • Showing causal connections between all events and states of behavior
  • Nodes express causal factors, arcs express cause-effect relationship
  • Different subgraphs can be tested, results are combinable

  • **Causal sufficiency test** -  
    *Will an effect always happen if all parent causes happen?*

  • **Counterfactual test** (Dawid Lewis 1975, philosophical logician) -  
    *If the cause would not have existed, could the effect still have happened?*

      • If „no“ for two effects, then the cause is a **necessary causal factor** (NCF)
Example [Peter B. Ladkin]

Interurban train collides with Indian Pacific at end of block

0. Indian Pacific delayed at end of block

1. Interurban train fails to halt upon driver sighting Indian Pacific

2. Interurban train driver proceeded at up to 50 kph, omitting to exercise extreme caution

3. Interurban train driver believed that block clear

4. Signaller issues permission to enter block

5. Rule 245 did not guide operations as intended

6. Signal failure

7. Indian Pacific does not use radio comms

8. Lack of train indicator board

9. Phone comms failure

10. Required communications procedures

11. Interurban train proceeds into block

12. Interurban train fails to halt upon driver sighting Indian Pacific

13. Signaller believes block clear

14. Signaller fails to determine location of previous train to have passed signal, the Indian Pacific

15. Phone technology unreliable

16. Phone comms time-consuming

17. Inappropriate phone technology
RCA: Ishikawa / Fishbone Diagram

• Invented by Japanese quality control statistician for failure prevention

• Identify (categorized) sources for variation

• Analysis tool for systematic cause-and-effect analysis

  • List problem to be studied in the ‘head of the fish’

  • Label main ‘fish bones’

    • 4 M‘s: Methods (process), machines (technology), materials (raw, consumables), manpower (physical and brain work)

    • 4 P‘s: Place, procedure, people, policies

    • 4 S‘s: Surroundings, suppliers, systems, skills

  • Identify factors per category that may affecting the problem / issue

  • Repeat with sub-factors
Fig: Application of Fish-bone diagram to craft based workshops in Indian design
FMEA

- **Failure Mode and Effects Analysis**

- Engineering quality method for early concept phase - identify and tackle weak points

- Introduced in the late 1940s for military usage (MIL-P-1629)
  
  - Later also used for aerospace program, automotive industry, semiconductor processing, software development, healthcare, ...

- Main goal is to identify and prevent critical failures

- Performed by cross-functional team of **subject matter experts**

- Most important task in many reliability programs
  
  - Six Sigma certification, reliability-centered maintenance (RCM) approach
  
  - Automotive industry (ISO16949, SAE J1739)
  
  - Medical devices
FMEA Goals

• Examples:
  • Identify and prevent safety hazards
  • Minimize loss of product performance
  • Improve test and verification plans
  • Improve process control plans
  • Consider changes to product design or manufacturing process
  • Identify significant product or process characteristics
  • Develop preventive maintenance plans for machinery and equipment
  • Develop online diagnosis techniques
FMEA Types

- **SFMEA (S: System)**
  - Improving the overall system design for avoiding complete failures
  - Analysis on highest possible level of abstraction
  - Targeting system-related deficiencies, such as safety or subsystem integration
  - Focus on functions and relationships that are unique to the system as a whole
  - Consider human interactions and services provided

- **DFMEA (D: Design)**
  - Focus on failure modes reasoned by design deficiencies in subsystems
  - Focus on parts that can be prototyped before high volume production
FMEA Types

• **PFMEA (P: Process)**
  
  • Analyze manufacturing and assembling processes
  
  • Influence design of machinery, selection of tooling and component parts
  
  • Built to design safely, with minimal downtime, scrap, or rework
  
  • Includes shipping, incoming parts, transportation of material, storage, conveyors, tool maintenance and labeling
  
  • Typically assumes that the product design is finished

• Do not mix up design failures and causes („incorrect material specified“) with process failures and causes („incorrect material used“)
FMEA Types

• Other, less popular variants

  • **Concept FMEA**: Short version of FMEA to select from system design alternatives, results in prioritized list of concerns

  • **Reliability-Centered Maintenance (RCM)**: Analytical process to identify preventive maintenance requirements

    • FMEA on manufacturing or operational equipment as central part

  • **Software FMEA**: Determine if software is fault-tolerant with respect to hardware failures, identify missing requirements

  • **Hazard Analysis**: Identify safety-related risks in system lifetime

  • **Human Factors FMEA**: Type of system FMEA to pays attention to the interaction between users (humans) and equipment

  • **Service FMEA**: Focus on installation or service of equipment during operation
FMEA Types

• Other, less popular variants
  
  • **Business Process FMEA**: Minimize inefficiencies by improving workflows, organizational management, and decision making

    • Similar to PFMEA, but business process steps replace manufacturing steps

  • **Failure Mode, Mechanisms, and Effects Analysis (FMMEA)**: Extends FMEA by identifying high priority failure mechanisms

    • Determine operational stresses and environmental parameters

  • **Failure Mode, Effects, and Diagnosis Analysis (FMEDA)**: FMEA extension to derive more details about the effects of failure modes

    • Generates failure rates for safety-related effect categories

    • Used to develop online diagnosis techniques, needed for IEC61509 compliance
FMEA

- Main assumption: System is vulnerable to certain failure modes
  - Examples: Electrical short-circuiting, corrosion, deformation
- Identify relevant failure mode candidates based on past experience
- Effect analysis for (specific) failure modes - what happens to the functionality visible to the end user?
  - Examples: Degraded performance, noise, injury
- Top-Down FMEA: Start with one function failure and find according failure modes
  - Typical for certification procedure, were the undesired functional problems are specified by the requirement documents
- Bottom-Up FMEA: Find all the effects caused by all failure modes
  - Failures are prioritized according to their consequences, how frequently they occur, and how easily they can be detected
# FMEA Worksheet

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Potential Failure Mode</th>
<th>Potential Effect(s) of Failure</th>
<th>Severity (S)</th>
<th>Potential Cause of Failure</th>
<th>Current Prevention Controls</th>
<th>Probability of Occurrence (O)</th>
<th>Current Detection Controls</th>
<th>Probability of Detection (D)</th>
<th>RPN = S<em>O</em>D</th>
<th>Criticality = S*O</th>
<th>Recommended Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FMEA Worksheet

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Potential Failure Mode</th>
<th>Potential Effect(s) of Failure</th>
<th>Severity (S)</th>
<th>Potential Cause of Failure</th>
<th>Current Prevention Controls</th>
<th>Probability of Occurrence (O)</th>
<th>Current Detection Controls</th>
<th>Probability of Detection (D)</th>
<th>RPN = S<em>O</em>D</th>
<th>Criticality = S*O</th>
<th>Recommended Action</th>
</tr>
</thead>
</table>

- Focus only on main concerns in the expert team

- Starting point is typically some system hierarchy of components / process steps

- **Functions** need to have a **given standard of performance or requirement**, so that failure modes can be clearly identified

- Analyze severe failure modes by other techniques (e.g. FTA)

- Focus only the most serious **effect** (system/end user consequence) of a failure

- **Severity** indicator ranks the effect, typically based on company standards
FMEA Worksheet

- **Cause** is the specific reason for the failure („why?“, „due to“)
  - If the cause occurs, the corresponding failure mode occurs
  - In maintenance analysis, cause is a 'maintenance-actionable' item
  - Failure mechanism != cause

- **Occurrence** expresses the likelihood of cause + failure mode
  - Relative meaning rather than absolute value, must be ok to say „never happens“

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Potential Failure Mode</th>
<th>Potential Effect(s) of Failure</th>
<th>Severity (S)</th>
<th>Potential Cause of Failure</th>
<th>Current Prevention Controls</th>
<th>Probability of Occurrence (O)</th>
<th>Current Detection Controls</th>
<th>Probability of Detection (D)</th>
<th>RPN = S<em>O</em>D</th>
<th>Criticality = S*O</th>
<th>Recommended Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FMEA Worksheet

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Potential Failure Mode</th>
<th>Potential Effect(s) of Failure</th>
<th>Severity (S)</th>
<th>Potential Cause of Failure</th>
<th>Current Prevention Controls</th>
<th>Probability of Occurrence (O)</th>
<th>Current Detection Controls</th>
<th>Probability of Detection (D)</th>
<th>RPN = S<em>O</em>D</th>
<th>Criticality = S*O</th>
<th>Recommended Action</th>
</tr>
</thead>
</table>

- **High severity** must always be considered, regardless of occurrence ranking.

- **Prevention controls** are intended (!) to reduce the likelihood of occurrence.
  - Should only consider things that are already planned or in place - do not put 'want-to' items or functions here.

- **Detection controls** describe how the failure mode or cause is detected.
  - Example: In DFMEA, testing is a detection control.

- Controls do not influence the severity of a failure mode.
Starter Questions [Carlson]

- For **functions** in a system / design FMEA:
  - „What are the primary purposes of this item?“
  - „What is the item supposed to do? What must the item not do?“
  - „What is the standard of performance?“
  - „What functions occur at the interfaces?“
  - „What safety-related functions are important for this item?“

- For **functions** in a process FMEA:
  - „What is the operation? What is the part or assembly? What is the tool?“
  - „What is the primary purpose of the operation?“
  - „What is the standard of performance?“
Starter Questions [Carlson]

• For **failure modes** in a system / design FMEA:
  • „How can the item fail to perform the intended function?“
  • „How could the item perform an unintended function?“
  • „What could go wrong at the interfaces?“
  • „What has gone wrong in the past?“
  • „How could the item be abused?“
  • „What are the team member concerns with this item?“

• For **failure modes** in a process FMEA:
  • „Why would a part be rejected at this operation?“
  • „What could go wrong with this operation?“
Starter Questions [Carlson]

• For effects in a system / design FMEA:
  • „What is the consequence of the failure?“
  • „What will the customer experience?“
  • „Will the failure cause potential harm to the end users?“
  • „Will the failure cause potential violation of regulations?“
  • „What would a failure mean to adjacent parts and subsystems?“

• For effects in a process FMEA:
  • „Will the failure cause potential harm to equipment or operators?“
  • „What will be the consequences on downstream processing or plant level?“
  • „Will the failure cause potential violation of regulations?“
Starter Questions [Carlson]

- For **causes** in a system / design FMEA:
  - „How can the failure occur? What is the mechanism of failure?“
  - „What circumstances could cause the item to not perform the intended function?“
  - „Are there possible system interactions, degradations, operating environments, customer usages, or assembly issues that could cause the failure?“

- For **causes** in a process FMEA:
  - „What could cause the operation to fail in this manner?“
Starter Questions [Carlson]

• For **recommended actions** in a system / design FMEA:
  • „What can be done to reduce severity by modifying the design?“
  • „If the product fails, how can the user be protected from potential harm or injury?“
  • „How can the current design be made more robust?“
  • „What tests or evaluation techniques needs to be added or modified to improve detection capability?“
  • „If the recommended actions are implemented, will that be sufficient to address all high severity and high RPN risk?“

• For **recommended actions** in a process FMEA:
  • Similarly for process / assembly / manufacturing
### Example: System FMEA of ATM [asq.org]

<table>
<thead>
<tr>
<th>Function</th>
<th>Potential Failure Mode</th>
<th>Potential Effect(s) of Failure</th>
<th>S</th>
<th>Potential Cause of Failure</th>
<th>D</th>
<th>RPN</th>
<th>CRIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispense amount of cash requested by customer</td>
<td>Does not dispense cash</td>
<td>• Customer angry &lt;br&gt; • Incorrect entry to demand deposit system &lt;br&gt; • Discrepancy in cash balancing</td>
<td>8</td>
<td>Out of cash</td>
<td>5</td>
<td>Internal low-cash alert</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Machine jams</td>
<td>3</td>
<td>Internal jam alert</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Power failure during transaction</td>
<td>2</td>
<td>None</td>
<td>10</td>
</tr>
<tr>
<td>Takes too long to dispense cash</td>
<td>Dispenses too much cash</td>
<td>• Bank loses money &lt;br&gt; • Incorrect entry to demand deposit system</td>
<td>6</td>
<td>Bills stuck together</td>
<td>2</td>
<td>Loading procedure</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Denomination s in wrong trays</td>
<td>3</td>
<td>Two-person visual verification</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Heavy network traffic</td>
<td>7</td>
<td>None</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Power failure during transaction</td>
<td>2</td>
<td>None</td>
<td>10</td>
</tr>
</tbody>
</table>

RPN prioritizes differently from criticality.
## Example: Design FMEA for disk brake [Carlson]

<table>
<thead>
<tr>
<th>Item</th>
<th>Potential Failure Mode</th>
<th>Potential Cause of Failure</th>
<th>Current Prevention Controls</th>
<th>Current Detection Controls</th>
<th>Recommended Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Brake System</td>
<td>Vehicle does not stop</td>
<td>Mechanical linkage break due to corrosion</td>
<td>Designed per material standard MS-845</td>
<td>Environmental stress test 03-9963</td>
<td>Change material to stainless steel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Master cylinder vacuum lock</td>
<td>Carry-over design with same duty cycle requirements</td>
<td>Pressure variability testing on system level</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Loss of hydraulic fluid due to back off of connector</td>
<td>Designed per torque requirements - 3993</td>
<td>Vibration step-stress test 18-1950</td>
<td>Modify connector from crimp style to quick connect.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Loss of hydraulic fluid due to hydraulic lines crimped or compressed</td>
<td>Designed per material standard MS-1178</td>
<td>DOE tube resiliency test</td>
<td>Modify design from MS-1178 to MS-2025 to increase strength.</td>
</tr>
</tbody>
</table>
Example: Severity Ranking in Automotive Industry

<table>
<thead>
<tr>
<th>Effect</th>
<th>Severity of effect on product</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>System fails to meet safety / regulatory requirements</td>
<td>Failure mode affects safe operation or rules, without warning</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Failure mode affects safe operation or rules, with warning</td>
<td>9</td>
</tr>
<tr>
<td>Loss or degradation of primary function</td>
<td>Loss of primary function</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Degradation of primary function</td>
<td>7</td>
</tr>
<tr>
<td>Loss or degradation of secondary function</td>
<td>Loss of secondary function</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Degradation of secondary function</td>
<td>5</td>
</tr>
<tr>
<td>Annoyance</td>
<td>Appearance or audible noise, noticed by &gt;75% of customers</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Appearance or audible noise, noticed by 50% of customers</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Appearance or audible noise, noticed by &lt;25% of customers</td>
<td>2</td>
</tr>
<tr>
<td>No effect</td>
<td>No discernible effect</td>
<td>1</td>
</tr>
</tbody>
</table>

• Example of Design FMEA severity scale for automotive industry [Carlson]
Example: NASA Spacecraft Severity Ranking

- Category 1, **Catastrophic** - Failure modes that could result in serious injury or loss of life, or damage to the launch vehicle.

- Category 1R, **Catastrophic** - Failure modes of identical or equivalent redundant hardware items that, if all failed, could result in Category 1 effects.

- Category 2, **Critical** - Failure modes that could result in loss of one or more mission objectives as defined by the GSFC project office.

- Category 2R, **Critical** - Failure modes of identical or equivalent redundant hardware items that could result in Category 2 effects if all failed.

- Category 3, **Significant** - Failure modes that could cause degradation to mission objectives.

- Category 4, **Minor** - Failure modes that could result in insignificant or no loss to mission objectives.
Example: Occurrence Ranking in Automotive

<table>
<thead>
<tr>
<th>Mode Likelihood</th>
<th>Associated Cause</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very high</td>
<td>New technology / design with no history</td>
<td>&gt;= 1 in 10</td>
</tr>
<tr>
<td></td>
<td>Failure is inevitable with new design</td>
<td>1 in 20</td>
</tr>
<tr>
<td></td>
<td>Failure is likely with new design</td>
<td>1 in 50</td>
</tr>
<tr>
<td></td>
<td>Failure is uncertain with new design</td>
<td>1 in 100</td>
</tr>
<tr>
<td>High</td>
<td>Frequent failures expected due to knowledge from similar designs / experiment</td>
<td>1 in 500</td>
</tr>
<tr>
<td></td>
<td>Occasional failures expected due to knowledge from similar designs / experiment</td>
<td>1 in 2000</td>
</tr>
<tr>
<td></td>
<td>Isolated failures expected due to knowledge from similar designs / experiment</td>
<td>1 in 10,000</td>
</tr>
<tr>
<td>Moderate</td>
<td>Only isolated failures expected due to knowledge from similar designs / experiment</td>
<td>1 in 100,000</td>
</tr>
<tr>
<td></td>
<td>No failures expected due to knowledge from similar designs / experiment</td>
<td>1 in 1,000,000</td>
</tr>
<tr>
<td>Low</td>
<td>Failure is eliminated through preventive control</td>
<td>Failure eliminated</td>
</tr>
</tbody>
</table>

- Example of Design FMEA occurrence scale for automotive industry [Carlson]
Example: Detection Ranking in Automotive

<table>
<thead>
<tr>
<th>Opportunity for Detection</th>
<th>Detection by Design Control</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>No current design control</td>
<td>10</td>
</tr>
<tr>
<td>Not likely to detect</td>
<td>Controls have weak capabilities</td>
<td>9</td>
</tr>
<tr>
<td>Prior to launch</td>
<td>Pass / fail testing after design freeze</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Test to failure testing after design freeze</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Degradation testing after design freeze</td>
<td>6</td>
</tr>
<tr>
<td>Prior to design freeze</td>
<td>Pass / fail testing before design freeze</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Test to failure testing before design freeze</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Degradation testing before design freeze</td>
<td>3</td>
</tr>
<tr>
<td>From design controls</td>
<td>Strong control capabilities</td>
<td>2</td>
</tr>
<tr>
<td>No applicable</td>
<td>Failure mode cannot occur since it is prevented</td>
<td>1</td>
</tr>
</tbody>
</table>

• Example of Design FMEA occurrence scale for automotive industry [Carlson]
FMEA Steps [asq.org]

1. Identify FMEA scope in cross-functional team (design, quality, testing, support, ...)

2. Identify functions in the investigation scope (verb + noun)

3. Per function, identify all ways a failure could happen - potential failure modes

4. Identify consequences / effects per failure mode - on the system itself, related systems, product, service, customers or regulations

5. Perform severity rating (S) per effect - From insignificant to catastrophic, add only highest ranked effects to the further analysis

6. Identify root causes per failure mode, rank by probability of occurrence (O)

7. List tests / procedures (process controls) that are in place to keep the causes away

8. Determine detection rating (D) per control - from certain detection to no detection, based on level of process control installed

9. Risk priority number (RPN) = S × O × D, Criticality (CRIT) = S × O
Risk Priority Number

- Numerical ranking of the risk of each potential failure mode
- Heated debate in the community about the proper utilization of this value
- Important: Always cover severity, regardless of RPN!
- Low RPN can be used to identify cost reduction candidates
- High risk factor should always lead to multiple **recommended actions**
  - Approval by management is a strategic advantage
  - If ‘review‘ is part of the actions, make sure that there is a follow-up on results
Risk Priority Number - Limitations

- **Subjectivity**: Helps in prioritizing recommended actions, no risk comparison across FMEAs useful

- **Lack of detection information**: Some companies only compute RPN=SxO

- **Holes in the scale**: Although RPN is an integer scale, it is not continuous or proportional

- **Duplicate RPN numbers**: Failures with widely differing severity can be evaluated as having the same importance

- **RPN Thresholds**: Management loves RPN thresholds, leads to ‘numbers game’
  - Teams lower numbers to avoid excessive consequences
  - Most important aspect for a successful FMEA is to have a honest team
Example: NASA Spacecraft FMEA

Ground Rules for Failure Mode Analysis

• Only one failure mode exists at a time.

• All inputs (including software commands) to the item being analyzed are present and at nominal values.

• All consumables are present in sufficient quantities.

• Nominal power is available.

• All mission phases are considered in the analysis; mission phases that prove inapplicable may be omitted.

• Connector failure modes are limited to connector disconnect.

• Special emphasis will be directed towards identification of single failures that could cause loss of two or more redundant paths.
Example: NASA Spacecraft FMEA
Flow Diagram of Overall System Analysis

- Define System and Requirements
- Define Hardware Subdivision Level for Analysis
- Identify Design Requirements to be Verified

- Complete Work Sheets
  - Failure Modes
  - Failure effects
  - Severity Classification
  - Corrective Measures

- Obtain Functional & Reliability Block Diagrams
- Define Ground Rules Assumptions & Mission Phases

- Identify Problem Areas
- Recommend Corrective Action
- Document the Analysis
Software FMEA

• FMEA on system functional level, detailed design (logic) level, or code level
  • Typically supported by data flow diagrams

• Typical objectives
  • Identify missing software requirements
  • Analyze output variables
  • Analyze system response in reaction to different environmental input
  • Analyze interfaces in addition to functions
  • Identify software response to hardware anomalies

• Since software is a logical construct, hazards must be identified first [Goddard]
Software FMEA - Failure Definition [Raheja]

• Failure to perform a function reliably
• Failure to perform a function safely
• Failure to perform a function when (not) needed
• Performing functions that are not in the specification
• Failure to stop a task at the right time
• Loss of input or output
• Corrupted performance by an operating environment
• Failure from an incorrect request
• Incomplete execution
• Inability to execute critical interruptions
Software FMEA - Cause Definition [Raheja]

- Useful to define **cause types**
  - Physical hardware effects
  - Coding or logic errors
  - Input / output error
  - Data handling
  - Definition of variables
  - Interface failures
  - Failed hardware / power outage / loose wires or cables
  - Communication failure

- Omissions in the specification
  - Insufficient or corrupted memory
  - Operational environment
  - Inaccurate input, such as from sensors
Prioritize FMEA Projects

- Amount of FMEAs typically limited by budget -> Preliminary risk assessment
- Rank component risks and multiply them
- Example for new bicycle product [Carlson]

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Risk from System FMEA</th>
<th>Safety Concerns</th>
<th>New Technology</th>
<th>New Applications</th>
<th>Field Problems</th>
<th>Regulatory Risks</th>
<th>Supplier Concerns</th>
<th>Other</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>36</td>
</tr>
<tr>
<td>Front Wheel</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Rear Wheel</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Sprocket</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Chain</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Seat</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Handlebar</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Hand brake</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>36</td>
</tr>
<tr>
<td>Suspension</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
</tr>
</tbody>
</table>
Most Common FMEA Mistakes [Carlson]

- Some FMEAs do not recommend any action at all
- Some FMEAs recommend mostly testing, other ineffective actions
- Some FMEAs fail to address all high-risk failure modes
- Some FMEA teams do not include representatives from test or analysis department
- Some companies focus on part or subsystem failures and miss the interfaces
- Failure to link between FMEA and field information
- „Missing the forest for the trees“ - Too much detail makes it difficult to identify areas of higher risk
- Late FMEAs are less effective
- Team meeting attendance and right choice of people is crucial
- FMEA in itself as procedure must be evaluated, many ways for doing it wrong
FMEA Success Factors [Carlson]

• Avoid 'lost in space' issues
  • Clear definition of the scope of the FMEA project
  • Get team consensus on boundaries
  • Gather relevant information and documentation before the meeting

• Collect subject matter experts, typically group of 4-8 people

• One participants must act as FMEA facilitator - brainstorming guidance, encouragement, discussion control, decision making, conflict and time management

• Past FMEA's and field data must be re-used to save time

• Goal is to improve the system design, not to understand it

• Take causes to the level of root cause (e.g. by FTA), not controls

• Avoid voting, prefer consensus
Failure Mode Effects and Criticality Analysis (FMECA)

• Similar to FMEA, but relies on a different set of scales, including **criticality**

• Standards: MIL-STD 1629A and SAE ARP5580, originally developed by NASA

• **Qualitative FMECA**
  
  • Rate the **severity** of potential effects (catastrophic, critical, marginal, minor)
  
  • Rate the likelihood of **occurrence** per mode (frequent, reasonably probable, occasional, remote, extremely unlikely)
  
  • Compare the failure modes using the **criticality matrix**

• **Quantitative FMECA**
  
  • Series of calculations to rank items and failure modes
### Qualitative FMECA Example [Carlson]

<table>
<thead>
<tr>
<th>Items</th>
<th>Functions</th>
<th>Failures and Causes</th>
<th>Local Failure Effects</th>
<th>Next Higher Level Effects</th>
<th>End Effects</th>
<th>Severity Class</th>
<th>Failure Detection Method</th>
<th>Compensation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicycle Brake Pad</td>
<td>Primary means of friction between the brake caliper against the wheel</td>
<td>Excessive wear of pad due to wrong material</td>
<td>No controlled friction to the wheel</td>
<td>Wheel does not slow down when break lever pulled</td>
<td>Bicycle does not stop, causing accident</td>
<td>Catastrophic</td>
<td>Brake testing procedure #1234</td>
<td>Select new material with better durability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Pad material cracks (cured too hot)</td>
<td>Erratic friction against wheel</td>
<td>Wheel motion chugs during maneuver</td>
<td>Bicycle operator dissatisfied</td>
<td>Marginal</td>
<td>No detection, only by owner</td>
<td>Revise curing procedure</td>
</tr>
</tbody>
</table>
Quantitative FMECA

- Calculate **expected failures** for each item (from reliability data)
- Identify the **mode ratio of unreliability** for each potential failure mode
  - Percentage of all item failures that will be due to the investigated failure mode
    - Can be derived from reliability prediction data (see last course unit)
- Rate the **probability of loss** resulting from each failure mode
  - Probability that the failure mode leads to a system failure
- **Mode criticality** [item, failure mode] = Expected failures x mode ratio x probability of loss
- **Item criticality** [item] = Sum(mode criticalities)
- Output: Critical item list, single failure points list, failure mode list
## Quantitative FMECA Example [Carlson]

<table>
<thead>
<tr>
<th>Items</th>
<th>Operating Time (h)</th>
<th>Expected Failures</th>
<th>Functions</th>
<th>Failures and Causes</th>
<th>Ratio of Unreliability</th>
<th>Probability of Loss</th>
<th>Mode Criticality</th>
<th>Item Criticality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicycle Brake Pad</td>
<td>5475</td>
<td>0,548</td>
<td>Primary means of friction between the brake caliper and the wheel</td>
<td>Excessive wear of pad due to wrong material</td>
<td>0,85</td>
<td>0,75</td>
<td>0,349</td>
<td>0,361</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Pad material cracks (cured too hot)</td>
<td>0,15</td>
<td>0,15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Hazard & Operability Studies (HAZOPS)

- Process for identification of potential hazard & operability problems, caused by **deviations from the design intend**
  - Difference between deviation (failure) and its cause (fault)
  - Conduct intended functionality in the safest and most effective manner
  - Initially developed to investigate chemical production processes, meanwhile for petroleum, food, and water industries
  - Extended for complex (software) systems

- Qualitative technique
  - Take full description of process and systematically question every part of it
  - Assess possible deviations and their consequences
  - Based on **guide-words** and multi-disciplinary meetings
HAZOPS

• First identify system entities and their attributes (e.g. state diagrams)

• Use of keywords to focus the attention

  • **Primary keywords** - Focus attention on a particular aspect of the design intent / process condition / investigated parameter
    
    • Examples for plants:
      flow, pressure, react, corrode, temperature, level, mix, absorb, erode, ... isolate, vent, inspect, start-up, shutdown, purge, maintain, ...
    
    • Examples for Java Language Definition analysis:
      Type default value, type value range, name scope, class modifier, class name, field modifier, field type, method modifier, method name, formal parameter, ...

  • **Secondary keywords** - When combined with a primary keyword, suggest possible deviations of the system from the design intent
    
    • Standardized list, not all combinations are appropriate
## Plant Example - Secondary Keywords

<table>
<thead>
<tr>
<th>Word</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>The design intent does not occur (e.g. Flow/No), or the operational aspect is not achievable (Isolate/No)</td>
</tr>
<tr>
<td>Less</td>
<td>A quantitative decrease in the design intent occurs (e.g. Pressure/Less)</td>
</tr>
<tr>
<td>More</td>
<td>A quantitative increase in the design intent occurs (e.g. Temperature/More)</td>
</tr>
<tr>
<td>Reverse</td>
<td>The opposite of the design intent occurs (e.g. Flow/Reverse)</td>
</tr>
<tr>
<td>Also</td>
<td>The design intent is completely fulfilled, but in addition some other related activity occurs (e.g. Flow/Also indicating contamination in a product stream)</td>
</tr>
<tr>
<td>Other</td>
<td>The activity occurs, but not in the way intended (e.g. Flow/Other could indicate a leak or product flowing where it should not)</td>
</tr>
<tr>
<td>Fluctuation</td>
<td>The design intention is achieved only part of the time (e.g. an air-lock in a pipeline might result in Flow/Fluctuation)</td>
</tr>
<tr>
<td>Early</td>
<td>Usually used when studying sequential operations, this would indicate that a step is started at the wrong time or done out of sequence</td>
</tr>
<tr>
<td>Late</td>
<td></td>
</tr>
</tbody>
</table>
### Plant Example - Combinations [Wikipedia]

<table>
<thead>
<tr>
<th>Parameter / Guide Word</th>
<th>More</th>
<th>Less</th>
<th>None</th>
<th>Reverse</th>
<th>As well as</th>
<th>Part of</th>
<th>Other than</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow</td>
<td>high flow</td>
<td>low flow</td>
<td>no flow</td>
<td>reverse flow</td>
<td>deviating concentration</td>
<td>contamination</td>
<td>deviating material</td>
</tr>
<tr>
<td>Pressure</td>
<td>high pressure</td>
<td>low pressure</td>
<td>vacuum</td>
<td>delta-p</td>
<td></td>
<td></td>
<td>explosion</td>
</tr>
<tr>
<td>Temperature</td>
<td>high temperature</td>
<td>low temperature</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>high level</td>
<td>low level</td>
<td>no level</td>
<td></td>
<td>different level</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>too long / too late</td>
<td>too short / too soon</td>
<td>sequence step skipped</td>
<td>backwards</td>
<td>missing actions</td>
<td>extra actions</td>
<td>wrong time</td>
</tr>
<tr>
<td>Agitation</td>
<td>fast mixing</td>
<td>slow mixing</td>
<td>no mixing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reaction</td>
<td>fast reaction / runaway</td>
<td>slow reaction</td>
<td>no reaction</td>
<td></td>
<td></td>
<td></td>
<td>unwanted reaction</td>
</tr>
<tr>
<td>Start-up / Shutdown</td>
<td>too fast</td>
<td>too slow</td>
<td></td>
<td>actions missed</td>
<td></td>
<td></td>
<td>wrong recipe</td>
</tr>
<tr>
<td>Draining / Venting</td>
<td>too long</td>
<td>too short</td>
<td>none</td>
<td>deviating pressure</td>
<td></td>
<td>wrong timing</td>
<td></td>
</tr>
<tr>
<td>Inertising</td>
<td>high pressure</td>
<td>low pressure</td>
<td>none</td>
<td></td>
<td></td>
<td>contamination</td>
<td>wrong material</td>
</tr>
<tr>
<td>Utility failure</td>
<td>high pressure</td>
<td>low pressure</td>
<td>failure</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Word</td>
<td>Meaning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>No part of the intention is achieved. No use of syntactic components</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>More</td>
<td>A quantitative increase, the data value is too high (within or out of bounds)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Less</td>
<td>A quantitative decrease, the data value is too low (within or out of bounds)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>As Well As</td>
<td>Specific design intent is achieved but with additional results</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Part Of</td>
<td>Only some of the intention is achieved, incomplete</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reverse</td>
<td>Reverse flow - flow of information in wrong direction, iteration count modified in wrong direction, logical negation of condition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other Than</td>
<td>A result other than the original intention is achieved, complete but incorrect</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Narrowing</td>
<td>Scope or accessibility is narrower than intended.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Widening</td>
<td>Scope or accessibility is enlarged</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Equivalent</td>
<td>The same design intent is achieved in a different way (without any side effect)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
HAZOPS Procedure

Select a section of the plant

Have all relevant Primary Keywords for this plant section been considered?

Yes

No

Select a Primary Keyword not previously considered (e.g. Pressure)

Have all relevant Secondary Keywords for this Primary Keyword been considered?

Yes

No

Select a relevant Secondary Keyword not previously considered (e.g. More)

Are there any Causes for this deviation not previously discussed and recorded?

No

Yes

Record the new Cause

Are associated consequences of any significance?

No

Yes

Record the Consequence/s

Record any Safeguards identified

Having regard to the Consequences and Safeguards, is an Action necessary?

No

Yes

Record the agreed Action
HAZOPS Documentation

• Apply in a systematic way all relevant keyword combinations to the design

• Per combination, record
  
  • Deviation (keyword combination)

  • Cause (potential causes for the deviation)

  • Consequence (from both the deviation and the cause)
    
    • Should not take credit for protective systems or instruments in the design, since not all operational conditions are clarified at this point

  • Safeguards (which prevents the cause or safeguards the system against it)
    
    • Can be hardware, software, or procedures

  • Actions (which either remove the cause or mitigate the consequences)
HAZOPS for Java Language Specification
[Kim, Clark, McDermid] - Deviants Example

<table>
<thead>
<tr>
<th>Attribute: Field modifiers</th>
<th>FieldModifiers&lt;sup&gt;opt&lt;/sup&gt; Type VariableDeclarators ;</th>
</tr>
</thead>
</table>
| Causes                     | - The modifier static is specified where the field should be an instance variable.  
                            | - The modifier static is not specified where the field should be a class (static) variable. |
| Consequences               | A field becomes a class variable instead of an instance variable (or vice versa), causing different results or compile errors. |
| Attribute: Field modifiers | Guideword: OTHER_THAN                                   |
| Causes                     | The number of the specified modifiers increases.       |
| Consequences               | The behaviour of a field is changed/restricted or compile-time error occurs. |
| Attribute: Field modifiers | Guideword: MORE                                        |
| Causes                     | The number of the specified modifiers decreases. (e.g. from 2 modifiers to 1 or no modifier) |
| Consequences               | The behaviour of a field is changed or compile-time error occurs. |
| Attribute: Accessibility   | Guideword: WIDENING                                    |
| Causes                     | An access modifier is changed from protected to public, or from private to public. |
| Consequences               | The fields that were not accessible become accessible. For example when a field is changed from private to default access, any entities within the same package can access the field either by SimpleName or QualifiedName. |
| Attribute: Type compatibility of a field type | Guideword: AS_WELL_AS |
| Causes                     | - Class type T is declared instead of class type S, provided that S is a subclass of T.  
                            | - Interface type K is used instead of interface type J, provided that J is a subinterface of K. |
| Consequences               | Widening reference conversion regards a reference as having some other type. |
| Attribute: Type compatibility of a field type | Guideword: PART_OF |
| Causes                     | Class type S instead of class type T is declared, provided that S is a subclass of T. |
| Consequences               | classCastException may arise if the actual reference value is not a legitimate value of the declared type at run time. |
Reliability Models for IT Infrastructures

- System reliability in a commercial environment is determined by many factors:
  - Software and hardware reliability
  - Training of maintenance personnel
  - „Business processes“ how maintenance is handled
  - The way the IT department is organized
  - ...
- Impact of management organization on reliability is an emerging research field
- Standards for IT organization, based on best practices
  - Describe which processes have to be established in an IT department
  - Provide reference models for organization of IT department
Standards

- ISO/IEC 20000: IT Service Management
- Capability Maturity Model Integration (CMMI®)
- Control Objectives for Information and related Technology (COBIT®)
- Projects in Controlled Environments (PRINCE2®)
- Project Management Body of Knowledge (PMBOK®)
- Management of Risk (M_o_R®)
- eSourcing Capability Model for Service Providers (eSCM-SPTM)
- Telecom Operations Map (eTOM®)
- Six Sigma™.
Software Process Evaluation and Improvement

• Software reliability can be derived from level of trust into the development process

• Relevant for software supplier evaluation in public bidding

• Most famous approach is the **Capability Maturity Model (CMM)** by CMU Software Engineering Institute (SEI), extended to **CMMI**
  
  • Customer wants: Completion in time, budget, and functionality with high quality
  
  • Management wants: High customer satisfaction and productivity, control

• Application areas for CMMI
  
  • System engineering
  
  • Software engineering
  
  • Integrated product and process development
  
  • Supplier sourcing
CMMI Maturity Levels

- **Level 1: Initial**
  - Processes unpredictable, poorly controlled and reactive

- **Level 2: Managed**
  - Processes characterized for projects and is often reactive.

- **Level 3: Defined**
  - Processes characterized for the organization and is proactive. 
    (Projects tailor their processes from organization's standards)

- **Level 4: Quantitatively Managed**
  - Processes measured and controlled

- **Level 5: Optimizing**
  - Focus on process improvement

(C) Wikipedia
Capability Maturity Model Integration - Structural Overview

Maturity Level N

- Process Area 1
  - Specific Goals
    - Specific Practices
  - Commitment to Perform
- Process Area 2
  - Generic Goals
    - Ability to Perform
    - Directing Implementation
    - Verifying Implementation
  - Commitment to Perform
- Process Area 3
  - Specific Practices

success in

CMMI for Acquisition
CMMI for Development
CMMI for Services

Generic Practices
### Capability Maturity Model Integration

- Maturity level consists of the results in multiple process areas
- Version 1.3 contains 22 process areas

#### Process Area

<table>
<thead>
<tr>
<th>Process Mgmt.</th>
<th>Project Management</th>
<th>Engineering</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Process Management:
  - Project Planning (PP)
  - Project Monitoring and Control (PMC)
  - Supplier Agreement Management (SAM)
  - Organizational Process Focus (OPF)
  - Organizational Process Definition (OPD)
  - Organizational Training (OT)
  - Organizational Process Performance (OPP)

- Engineering:
  - Integrated Project Management (IPM)
  - Risk Management (RSKM)
  - Organizational Innovation and Deployment (OID)

- Support:
  - Configuration Management (CM)
  - Process & Product Quality Assurance (PPQA)
  - Measurement and Analysis (MA)
  - Decision Analysis and Resolution (DAR)
  - Technical Solution (TS)
  - Product Integration (PI)
  - Verification (VER)
  - Validation (VAL)
  - Causal Analysis and Resolution (CAR)
Example: Specific Goals in Project Planning (Project Management Area, Level 2)
Example: Specific Goals in Requirements Management (Engineering Area, Level 2)

- **SG 1 Manage Requirements**
  - SP 1.1 Obtain an Understanding of Requirements
  - SP 1.2 Obtain Commitment to Requirements
  - SP 1.3 Manage Requirements Changes
  - SP 1.4 Maintain Bidirectional Traceability of Requirements
  - SP 1.5 Identify Inconsistencies between Project Work and Requirements

1 Typical tools:
- requirements list
- requirements database

---

(C) Dr. Ralf Kneuper
Example: Specific Goals in Measurement and Analysis (Support Area, Level 2)
Six Sigma

• Coined by Motorola engineer (1986), meanwhile Motorola trademark
  • Started as simple statistical technique to reduce defects in manufacturing
  • Intended to improve quality by improving manufacturing processes
• Meanwhile common quality improvement strategy (Sony, Honda, TI, Canon, ...)
• Measured process quality level and its standard deviation is set in relation to target value and its range of tolerance
  • Target value and mean should be close together
  • Range of tolerance shouldn’t be much smaller than the standard deviation
• Next tolerance threshold should be at least six standard deviations away from the mean, so that the specified limit is never reached
  • In practice, mean is typically shifted by 1.5 standard deviations from target value
  • Translates to 99.99966% defect-free produced units
Six Sigma

- Management framework for processes, techniques, and training
- Data-driven systematic evaluation approach
  - Asks tougher and tougher questions until quantifiable answers are received
  - Measure of process capability, related to defect rate and complexity of a product
  - Defect: Any process output that does not meet customer expectation
- Advantage: Well defined targets by specification of statistical metrics
  - Popular among manufacturing and service organizations
- Quality improvement does not automatically lead to financial benefit
  - Mathematical models for choosing an optimal improvement strategy
  - Might demand process tweaking or replacement
Six Sigma Metrics

- **Yield (Y):** Measure of process capability
  
  - Ratio of defect free units vs. units produced (‘opportunities’)

  \[
  Y = \frac{\text{Defect free units}}{\text{Number of opportunities}} \times 100\%
  \]

- **Defects per Million Opportunities (DPMO)**
  
  - Typically estimated from a sample of units - defects per unit (DPO)

  \[
  DPMO = DPU \times 10^6 = \frac{\text{Number of defects}}{\text{Number of opportunities}} \times 10^6
  \]

- **Sigma Quality Level:** Measure of quality for the output produced by an organization
  
  - Level directly related to DPMO metric
Six Sigma Levels

• Level 6 allows less than four (3.4) defects per million opportunities (DPMO)

• Practice: Mean expected to move up to 1.5 standard deviations over time

• Experience that processes get worse in the long term

• Assumption that initial 6-sigma process will degrade to 4.5-sigma

• Designed to prevent underestimation of defect levels

• Does not reflect product specifics (pace maker vs. mass mail advertiser)

<table>
<thead>
<tr>
<th>Level</th>
<th>DPMO</th>
<th>Defective</th>
<th>Yield</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>691462</td>
<td>69 %</td>
<td>31 %</td>
</tr>
<tr>
<td>2</td>
<td>308538</td>
<td>31 %</td>
<td>69 %</td>
</tr>
<tr>
<td>3</td>
<td>66807</td>
<td>6.7 %</td>
<td>93.3 %</td>
</tr>
<tr>
<td>4</td>
<td>6210</td>
<td>0.62 %</td>
<td>99.38 %</td>
</tr>
<tr>
<td>5</td>
<td>233</td>
<td>0.023 %</td>
<td>99.977 %</td>
</tr>
<tr>
<td>6</td>
<td>3.4</td>
<td>0.00034 %</td>
<td>99.99966 %</td>
</tr>
<tr>
<td>7</td>
<td>0.019</td>
<td>0.0000019 %</td>
<td>99.9999981 %</td>
</tr>
</tbody>
</table>
Six Sigma Project Selection

- Each process improvement opportunity is treated as *project*
- Criteria's for choice [Pande]
  - *Business benefits:* Impact on customers, business strategy, finances; urgency
  - *Feasibility:* Resources needed, expertise available, complexity, success probability
  - *Organization impact:* Learning and cross-functional benefits
  - Also: Top management commitment
- Project selection itself should follow a strategy
- Once project is chosen, walk through methodology for improvement
Six Sigma Project Selection

- Analytic hierarchy process (AHP) project selection strategy [Kumar]

- Assign weights to projects according to criterias:
  - Duration for completion
  - Costs of project
  - Probability of success
  - Strategic fit of the project
  - Increase in customer satisfaction
  - Increase in Six Sigma quality level
  - Reduction of cost of poor quality (CoPQ)
  - Manpower requirement (green belts and black belts)

AHP Explained [Wikipedia]
DMAIC Methodology

• Aimed at improving existing process by using Six Sigma (‘project‘)

• 5 stages in a cycle
  
  • **Define**: Identify the problem in terms of deficiencies in the *critical to quality (CTQ)* parameter
  
  • **Measure**: Define and use metrics to rank process capability, get gap to target
  
  • **Analyze**: Determine cause-effect relationship between process performance gaps (measured in CTQ) and process inputs
    
    • Examples: 5 Whys, data mining, experiment design, FMEA, ...
  
  • **Improve**: Implement solution for identified problem from ‘define‘ stage
  
  • **Control**: Sustain the improvement

• Strength in tool box due to mathematical foundation
**DEFINE**
- Why must this project be done **NOW**?
- What is the business case for the project?
- Who is the customer?
- What is the current state?
- What will be the future state?
- What is the scope of this project?
- What are the tangible deliverables?
- What is the due date?

**CONTROL**
- During the project, how will I control risk, quality, cost, schedule, scope, and changes to the plan?
- What types of progress reports should I send to sponsors?
- How will I assure that the business goals of the project were accomplished?
- How will I maintain the gains made?

**MEASURE**
- What are the key metrics for this business process?
- Are metrics valid and reliable?
- Do we have adequate data on this process?
- How will I measure progress?
- How will I measure ultimate success?

**IMPROVE**
- What is the work breakdown structure for this project?
- What specific activities are necessary to meet the project’s goals?
- How will I re-integrate the various subprojects?
- Do the changes produce the desired effects?
- Any unanticipated consequences?

**ANALYZE**
- Current state analysis
- Is the current state as good as the process can do?
- Who will help make the changes?
- What resources will we need?
- What could cause this change effort to fail?
- What major obstacles do I face in completing this project?
Six Sigma Roles

- **Executive leadership**: Empowers the other role holders with freedom and resources
  - Responsible to drive new ideas for improvement

- **Champions**: Drawn from upper management, responsible for Six Sigma deployment
  - Act as mentors for black belts

- **Master black belts**: Full-time in-house experts
  - Coach and trainer for the cross-department implementation of Six Sigma

- **Black belts**: Apply Six Sigma methodology full-time to specific projects
  - Rule of thumb: One black belt per 100 employees (1% rule)

- **Green belts**: Common employees helping the black belts along with their normal job

Rising number of participating employees is expected to increase profitability
Six Sigma Trends

• Design for Six Sigma (DFSS / DMADV) approach
  • Optimize both customer needs and organizational objectives in engineering work
  • Aims at early design phase, not at improving existing process

• 5 stages in a cycle
  • Define: Develop new product development (NPD) strategy
  • Measure: Understand customer requirements
  • Analyze: Develop conceptual design after analyzing design options
  • Design: Develop product or process design
  • Verify: Develop prototype, evaluate effectiveness of the design

• Design process or service „with the end in mind“
ITIL

- Information Technology Infrastructure Library (ITIL), latest version v3
  - Started as set of recommendations by the UK Government
  - Concepts and guides for **IT service management**
  - Supports to deliver business-oriented quality IT services
- Core publications: *Service Strategy, Service Design, Service Transition, Service Operation, Continual Service Improvement*
  - Broad tool support from vendors
  - High costs for certification and training
  - Methodology sometimes over-respected at the expense of pragmatism
ITIL

• Replaces technology-oriented view on IT service management with an „end-to-end“ approach

• Get rid of „technology silos“ and „islands of excellence“

• Management system, expected to be ...
  • ... more focused on business needs and related to business processes
  • ... less dependent on specific technology due to service thinking
  • ... more integrated with other management approaches and tools

• ITIL cycle is typically initiated by a change demand in business requirements
ITIL v3 - Service

• „A service is a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks.“

• Supports the customer / user in achieving his defined goals

• **Service value** for the customer = **utility** (what) and **warranty** (how)

• **Service level**, the service quantity and quality demands, should be measurable

• Current and future demands on service level are specified in a **service-level agreement**

• **Service assets** (resources, capabilities to control them) support **business assets**

• **Service providers:**
  
  • Type 1 (internally for one business unit)
  
  • Type 2 (for multiple business units in the same organization)
  
  • Type 3 (serving multiple external customers)
ITIL v3 - Service Lifecycle
ITIL v3 - Service Strategy

• Service strategy has to rely on acknowledgment that customer buys satisfaction of needs, not a physical product


• The four Ps of strategy

  • Perspective: Distinct vision and direction (e.g. what services to provide)
  
  • Position: The basis on which the provider will compete (e.g. market check)
  
  • Plan: How the provider will achieve the vision
  
  • Pattern: Distinct fundamental patterns in decisions and actions of time
ITIL v3 - Service Strategy

• Service Provisioning Models
  • Selected by customers, used by providers
  • Managed Service: Business unit requiring a service fully funds the service provision for itself
  • Shared Service: Provisioning of multiple services through shared resources and infrastructure, targeting one ore more business units
  • Utility: Provisioning based on customer requirements (how much, how often, what time)

• Organization Design and Development
  • Development stages of the organization itself, interfaces to services, service analytics, strategies for service sources (internal, shared, outsourcing, ...)
ITIL v3 - Service Strategy

• Service strategy process also contains side activities

  • **Financial management** for quantification of the money value of services - budgeting, accounting, charging

  • **Service portfolio management (SPM)** as continuous process for controlling investment across the service lifecycle
    
    • Planned services, live services, retired services

    • **Define, Analyze, Approve, Charter** the portfolio

• **Demand management**

  • Example: Excess capacity generates costs without generating value

  • Based on utility and warranty terms for service value
ITIL v3 - Service Strategy

• Roles and responsibilities
  
  • Business Relationship Manager (BRM) - Close contact to customer
  
  • Product Manager (PM) - Development and management of service lifecycle
  
  • Chief Sourcing Officer (CS) - Leading sourcing strategy development
ITIL v3 - Service Design

- Design IT services to meet current and future agreed business outcomes

- Service design has to consider ...
  - Agreed business outcomes
  - Support for service lifecycle, risk management, security, resiliency
  - Definition of measurement methods and metrics, skill development

- The four Ps of design
  - **People:** Peoples, skill and competencies involved in the provisioning
  - **Products:** Technology and management systems used in service delivery
  - **Processes:** Processes, roles and activities involved in service provisioning
  - **Partners:** Vendors, manufacturers, and suppliers used to assist the provisioning
ITIL v3 - Service Design

- **Service Design Package (SDP):** Defines all aspects and requirements of the service
  - Produced for each new IT service, major change, or IT service retirement

- **Service Catalogue Management (SCM):** Maintaining the central source of service information for business areas, describing available IT services

- **Service Level Management (SLM):** Negotiates, agrees and documents IT service targets with the business (e.g. SLA)

- **Capacity Management:** Management of capacity and performance-related issues for services and resources, to match IT to business demands
  - **Capacity Management Information System (CMIS)**

- **Availability Management:** Reactive activities (monitoring, measuring, analysis, event management, incident management) vs. proactive activities (planning, design, recommendations)
  - **Availability Management Information System (AMIS)**
ITIL v3 - Service Design

- **IT Service Continuity Management (ITSCM)**
  - Maintain appropriate risk reduction measures and recovery options

- **Information Security Management (ISM)**
  - Considered within the corporate governance framework
  - Align IT security with business security, ensure information security
    - *Information availability* - should be usable when required
    - *Information confidentiality* - observed or disclosed to only those who have a right to know
    - *Information integrity* - Completeness, accuracy, no unauthorized modification
    - *Information authenticity and non-repudiation* - Business transactions and information exchange can be trusted
ITIL v3 - Service Design

• **Supplier Management**
  - Obtain value for money from suppliers, while meeting the business targets

• **Supplier and Contract Database (SCD)**

• **Roles and responsibilities**

  • **Service Design Manager** - Coordination and deployment of quality solution designs for services and processes

  • **IT Designer / Architect** - Coordination and design of required technologies, architectures, strategies, designs, plans

  • **Service Catalogue Manager, Service Level Manager, Availability Manager, IT Service Continuity Manager, Capacity Manager, Security Manager, Supplier Manager ...**
ITIL v3 - Service Transition

• Deliver services into operational use
  • Receive SDP’s from design stage, deliver into operational stage every element required for ongoing operation
  • Needs to ensure that service can operate in foreseeable extreme or abnormal circumstances

• Change management
  • Can apply to different scopes of change and release management
  • Ensures that service changes are recorded, evaluated, authorized, prioritized, planned, tested, implemented, documented, reviewed
  • Optimize business risk of changes
ITIL v3 - Service Transition

• **Service Asset and Configuration Management (SACM)**
  - Identify, control and account for service assets and configuration items (CI)
  - **Support by Configuration Management System (CMS)**

• **Knowledge Management**
  - *Data-Information-Knowledge-Wisdom* structure
  - Make information available to the right person at the right point in time

• **Transition Planning and Support**
  - Identify, manage, and control risks of failure across transition activities
  - Improves ability to handle high volume of change and releases
ITIL v3 - Service Transition

• **Release and Deployment Management**
  
  • Covers whole assembly and implementation of new/changed services for operational use
  
  • From release planning to early life support
  
  • Deliver changes at optimized speed, risk and cost

• **Service Validation and Testing, Evaluation**
  
  • Provide objective evidence that the new / changes service supports the business requirements, including agreed SLAs

• Roles and responsibilities
  
  • Not anticipated as separate group of people, ‘flow of experience and skills‘
ITIL v3 - Service Operation

• Deliver agreed levels of service to customers, manage applications and infrastructure that support the service delivery

• Only in this stage, services actually deliver business value

• Balancing of conflicting goals
  • Internal IT view vs. external business view
  • Stability vs. responsiveness
  • Quality of service vs. cost of service
  • Reactive vs. proactive activities

• Operational health of services summarizes relevant indicators relevant for execution of vital business functions
ITIL v3 - Service Operation

- **Event Management Process**
  - Handle change of state that has significance for the management of a configuration item or IT service
  - May indicate correct or incorrect functioning (tape change vs. hardware outage incident)
  - Depends on monitoring, but always generates / consumes notifications
  - Response to an event can be automated or may require manual intervention

- **Incident Management Process**
  - Handle unplanned service interruption, quality reduction, or configuration item error state not impacting the service
  - Restore normal service as quickly as possible
  - Functional vs. hierarchical escalation
ITIL v3 - Service Operation

- **Request Fulfillment Process**
  - Source and deliver services
  - Provide information about services and procedures for obtaining them

- **Access Management Process**
  - Manage confidentiality, availability and integrity of data and intellectual property
  - Manage service access rights and user identity

- **Problem Management Process**
  - Problem is a cause of one or more incidents
  - Responsible for investigation of incidents for identification of the problem
  - Understand causes, document workarounds and request changes to permanently resolve the problems
ITIL v3 - Service Operation

- **Service Desk Function**
  - Single central point of contact for all users of the IT
  - Logs and manages all incidents, service requests and access requests
  - First-line investigation and diagnosis, keeping users informed
  - Different organization approaches:
    - Local service desk (close to the user)
    - Centralized service desk (higher volume with fewer staff)
    - Virtual service desk (staff in many locations, appear as one team)
    - Follow the sun (24h coverage, automated call passing)
ITIL v3 - Service Operation

• Technical Management Function
  • People providing technical expertise and management of IT infrastructure
  • Mainly assistance function

• Application Management Function
  • Focus on software rather than infrastructure
  • Applications treated as one component of service

• IT Operations Management Function
  • Operations control - routine tasks, centralized monitoring from cockpit
  • Facilities management - data centers, computer rooms, recovery sites
ITIL v3 - Continual Service Improvement

- Continual Service Improvement (CSI) maintains value for customers
  - Should be embedded in organization, instead of ad-hoc approaches after failure
- CSI model - Contrast current position with long-term goals and objectives
ITIL v3 - Continual Service Improvement

- 7-Step Improvement Process
ITIL v3 - Continual Service Improvement

- **Service Measurement**
  - Metric types:
    - **Technology metrics** - components, applications; performance, availability
    - **Process metrics** - critical success factors (CSFs), key performance indicators (KPIs)
    - **Service metrics** - results of service, computed from technology metrics
  - Validate earlier decisions, help in deciding for corrective actions

- **Service Reporting**
  - Summarize historical development of data collected
  - Dedicated **CSI manager role** recommended, all other activities as part of the other lifecycle stages
ITILv3 - Service Lifecycle
ITILv3 - Service Lifecycle

Continual Service Improvement (CSI)
- 7-Step Improvement Process
- Service Measurement
- Service Reporting

Service Strategy (SS)
- Strategy Generation
- Financial Management
- Service Portfolio Management
- Demand Management

Service Operation (SO)
- Event Management
- Incident Management
- Request Fulfilment
- Problem Management
- Access Management

Service Design (SD)
- Service Catalogue Management
- Service Level Management
- Capacity Management
- Availability Management
- IT Service Continuity Management
- Information Security Management
- Supplier Management

Service Transition (ST)
- Transition Planning and Support
- Change Management
- Service Asset & Configuration Mgmt
- Release and Deployment Mgmt
- Service Validation and Testing
- Evaluation
- Knowledge Management
CoBiT

- *Control Objectives for Information and related Technology (COBIT)*
- International model for defining control and audit goals for IT, started in 1996
  - Maintained by *Information Systems Audit and Control Association (ISACA)*
  - Association of auditors, reaction on business impact of IT
- Collection of best practices for auditing in IT, no standardization on its own
- Considers:
  - Technical standards (EDIFACT, ISO, ...)
  - Codes of conduct for business (EU, OECD, ...)
  - Qualification criteria for IT systems and processes (ISO 9000, common criteria, ...)
  - Professional standards from the domain
  - Documented industry practices
CoBiT

• Describes how to rate the maturity level of IT processes
  • Considers IT quality (efficiency, robustness), IT security (integrity, trustability), and IT fiduciary (compliance to financial and lawful rules)
• Processes are defined to rely on IT resources, structured into:
  • Data - structured and non-structures information elements
  • Application systems - collection of manual and programmed processes
  • Technologies - Hardware, operating systems, databases, networks, ...
  • Facilities - Resources for housing and operation of IT systems
  • People - Human part in IT system operation
• All these factors must be planned, developed, implemented, operated and monitored in a coordinated fashion - meta framework for compliance checks
CoBiT

• Metrics: Key Goal Indicators (KGI), Critical Success Factors (CSF), Key Performance Indicators (KPI)

• KGI ranks resulting output, KPI ranks performance of producing these outputs

• Example: Change Management Process
  • KGI‘s: Number of successful changes, reduction of service interruptions due to changes (goals of the process itself)
  • CSF‘s: Useful software and hardware inventory (rating of applied improvements)
  • KPI‘s: Number of changes leading to failures, number of delayed changes (rating of applied improvements)

• Using COBIT in an organization
  • Auditing of IT processes according to COBIT
  • Using COBIT as internal evaluation tool
CoBiT

• ITIL covers one part of COBIT scope (IT service management processes)

• Maturity of IT processes can be ranked with COBIT

  • Becomes relevant for BASEL II or Sarbanes-Oxley rating of organizations

• Top-down approach in CoBiT analysis

  • Derive IT architecture (resources) from IT goals (processes) from business goals (requirements)