MUAC SSAS Process
ES2 WS1-2013 Software Safety Assessment Workshop
7-8 May 2013

Marinella LEONE, ASD MUAC
Morten Trier HANSEN, ENG MUAC
Overview

• SSAS procedure in MUAC SMS:
  • Overview of SSAS central process and sub-processes

• Method for SW Assurance in projects/developments
  • Process and tools (AMC) adopted for projects/developments @MUAC and between MUAC and manufactures

• Method for SW Assurance in maintenance:
  • Maintenance process with SW assurance as an integrated set of activities

• Conclusions
SSAS procedure in MUAC SMS

1. **System or SW safety requirements** are derived → SW requirements are specified
1. System and SW safety requirements

- **Operational Risk Assessment** → Functional risk analysis and mitigation process to derive SWAL requirements as well as functional and availability requirements

- **Technical Risk Assessment** → drive requirements and keep trace of procedures to recover from failures of sub-systems
Operational Risk Analysis

- Operational Risk assessment is constructed on the basis of Functional failure analysis for operational service functions:
  - Allocation of SWAL to interfacing sub-systems according to severity and likelihood
  - Requirements are propagated to feeding sub-systems
Technical Risk Assessment (TRA)

FMEA: Failure mode effect analysis:
- Assess effects of all failures + define a corrective maintenance activity

MAEA: Maintenance activity effect analysis:
- Assess effects of all maintenance activities

---

**FMEA**

<table>
<thead>
<tr>
<th>Component</th>
<th>Failure mode</th>
<th>TE</th>
<th>OE</th>
<th>MA</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;HW&gt;</td>
<td>Failure</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;SW&gt;</td>
<td>Crash, loop</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;interfaces&gt;</td>
<td>Overload, corruption..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**MAEA**

<table>
<thead>
<tr>
<th>MA</th>
<th>Proc.</th>
<th>When</th>
<th>TE</th>
<th>OE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replace</td>
<td>MPR</td>
<td>Night</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- <corrective, e.g. replace HW> |
- <adaptive, e.g. install release> |
- <preventive, e.g. health check>
1. **System or SW safety requirements** are derived → SW requirements are specified

2. **Traceability** through the relevant system architectural levels down to the design and with verification records

3. **SW** satisfies requirement to **level of confidence** equivalent to criticality of software

4. **Configuration management** processes in place

5. **Relevant stakeholders** are involved
Overview

- SSAS procedure in MUAC SMS:
  - Overview of SSAS central process and sub-processes.

- Method for SW Assurance in projects/developments
  - Process and tools (AMC) adopted for projects/developments @MUAC and between MUAC and manufactures

- Method for SW Assurance in maintenance:
  - Maintenance process with SW assurance as an integrated set of activities

- Conclusions
Derivation of Tender Safety Requirements

- Coming from SIL (IEC61508) and FHA/FTA or RBD for apportionment of requirements

- Moving to SWAL from Functional failure analysis:
  - Allocation to interfacing system according to severity and likelihood
  - Requirement propagation to feeding sub-systems

- Easy in principle to change approach at the beginning of a new project. However initial effort to align expectations of stakeholders and some lessons learned are lost

IEC 61508
ED-109
ED-153
[SOW - 647] The Contractor shall:

- adopt the procedures, guidance and templates of the Customer's SMS for the following activities, as required:
  - Safety Management Plan (SMP).
  - Preliminary System Safety Assessment (PSSA).
  - System Safety Assessment (SSA).
  - System Safety Case (SSC).

- adopt the **EUROCAE Guidelines ED-153** for the development and/or selection of all software deliverables, or demonstrate that the Contractor's method of software development and/or selection is fully consistent with ED-153.

- adopt the international standard IEC61508 (Part 2; particularly Tables A.16 to A.18 and B.1 to B.5) for the development and/or selection of all hardware deliverables, or demonstrate that the Contractor's method of hardware development and/or selection is fully consistent with IEC61508. This is to ensure the hardware is consistent with the requirements for Mean Time Between Failures (MTBFs) and software integrity.


[SOW - 823] The Contractor shall comply with the **SWAL-3** as required by the compliance tables.
Compliance to the DSRs shall be demonstrated as follows:

- for Software DSRs: SWAL compliance via individual compliance statements with supporting evidence for each applicable Objective from the compliance tables in ED-153, as follows:
  - **Newly Developed Software (NDS).** The Contractor shall show compliance to SWAL 3 objectives in the tables from Sections 3 to 7.1 with the following structure:
    a) If all NDS are developed by the same supplier and the same process is applied, only one set of SWAL compliance tables (from Section 3 to Section 7.1) shall be provided by the Contractor.
    b) If any NDS is developed by a different supplier or according to a different process, a separate set of SWAL compliance tables (from Section 3 to Section 7.1) shall be provided by the Contractor for that software.

  - **Commercially available Off-The-Shelf (COTS).** The Contractor shall show compliance to SWAL objectives in the tables from Section 3 and Section 7.2 with the following structure:
    a) Separate sets of SWAL compliance tables (i.e. from Section 3 and Section 7.2) shall be provided for each COTS item.

    Note: Compliance in Section 3 can be demonstrated via reference to the NDS Section 3 tables, if the evidence provided in those tables have accounted for the COTS. (Section 3 provides objectives relating to overall project initiation, planning and safety, in which it would be valid to include COTS evidence).

  - **Non-COTS Reused Software (NCRS).** All the SWAL 3 objectives in the compliance tables from Sections 3 to 7.2 shall be used, thus including both development & COTS tables. This is because a properly substantiated combination of the NDS and COTS approaches is acceptable to the Customer when demonstrating SWAL 3 compliance; i.e. a lack of development evidence for the NCRS can be mitigated by COTS evidence and vice versa. The following structure shall be used:
    a) If evidence for NCRS is provided from the same supplier and follows the same process, only one set of compliance tables shall be provided by the Contractor.
    b) If any NCRS is developed by a different supplier or following a different process then separate SWAL compliance tables will be provided by the Contractor for that software.
Compliance Reports

- Ability to meet objectives?
  - CRs from start
- Discrepancy of processes?
  - Company processes
  - ED-153 expectation table
- Quality of documents?
  - Constructed assurance
- In-service history?
  - Monitoring method
- Unintended (unspecified/unused/unneeded) functionality?
  - Identify/assess
Component definition

- **NDS** – possible to design/define the correct decomposition; appropriate level of detail for SWAL analysis (SRS)?
- **NCRS** - Existing software might not be modular or might be decomposed in too low level components

- Creating logical CSCI level to abstract from detail?
- Artificial documentation structure not reflected by software packaging. Difficult/redundant to redefine interfaces at logical level that are covered by low level components (in specifications and tests)
SSS & SRS

- Low level SSS and poor SRS?
- Facing fear for “explosion” of SW requirements and tests
- Sometimes missing in the SRS (startup management, shutdown management, logic to transform inputs into outputs, mode of operations, error handling, boundary condition, etc…) directly incorporated in design documentation (e.g. algorithm)

- No need for SRSs? Important when different actors involved and criticality/complexity of the system
- Knowledge gap not reconciled between system and software engineers?
Approach for changes to legacy

- **NCRS** - No SRS exist for some legacy software (often just SSS, maybe SRS at lower level of component)

- SRS that introduce change are refined up to the level that the logic/algorithms supporting the corresponding functions can be tested

**Continuous Integration during SW development for NCRS**
Overview

• SSAS procedure in MUAC SMS:
  • Overview of SSAS central process and sub-processes.

• Method for SW Assurance in projects/developments
  • Process and tools (AMC) adopted for projects/developments @MUAC and between ANSP and manufactures

• Method for SW Assurance in maintenance:
  • Maintenance process with SW assurance as an integrated set of activities

• Conclusions
• Maintenance is easy:
  • No PMP, CMP, SDP, RMP, SVP, … - just:

  ![Diagram showing EQM and subsystems]

• But not so easy anyway:
  • Process changes have large consequences
  • One process needs to fit all
Safety in EQM

Objectives:
- Protect staff from complexity of safety standard

IEC 61508
ED-153

Development

EQM

MNP

Strategy
Organisation
Processes
A: CfM
B: Procedures
C: Documentation

Safety

Annex D

Development

Annex D

Development

MUAC - SSAS Procedure
**ED-153 example: Requirements**

<table>
<thead>
<tr>
<th>Object</th>
<th>Title</th>
<th>Topic</th>
<th>SWAL3</th>
<th>SWAL1</th>
<th>Clarification</th>
<th>Evidence</th>
<th>Process</th>
<th>Detail</th>
<th>Evidence</th>
<th>Maintenance</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.4</td>
<td>SWA</td>
<td>SWAL3</td>
<td>y</td>
<td>y</td>
<td>y</td>
<td>Documented software requirements.</td>
<td>EKATM, DEHY-10</td>
<td>Sura</td>
<td>Sura</td>
<td>TE2 &lt;sub&gt;sub&lt;/sub&gt; AND SBS</td>
<td>Software requirements may be documented as part of the subsystem requirements document - the requirement is an existence of the requirements, not on a specific document.</td>
</tr>
</tbody>
</table>

**Obj 4.3.4**

- The developer shall 4.3.4.1 establish and document software requirements, using software requirements standards rules as defined per Objectives 4.3.6 & 4.3.10.
- The Software requirements shall:
  - specify the accuracy, timing, performances, and safety / security of software, robustness to abnormal operating conditions, and maintainability;
  - be complete and correct;
  - comply with the System Requirements;
  - an identification of the configuration/initial data range.

**What does it mean?**

**EQM translation**
### EQM example: Requirements

**Subsystem Requirements Analysis (SuRA)**

<table>
<thead>
<tr>
<th>Id</th>
<th>Requirement</th>
<th>EOM reference</th>
<th>S&amp;M</th>
<th>SWAL 4</th>
<th>SWAL 3</th>
<th>ED-153 reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All subsystem requirements must be documented</td>
<td>TE2 SSS</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 4.3.12</td>
</tr>
<tr>
<td></td>
<td>Subsystem requirements must specify:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- functional behaviour</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- timing performances</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- software resource usage and margins (e.g. memory, CPU load, disk space, communication bandwidth, ...) on target hardware</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- adaptation/configuration data ranges and interface boundaries</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- robustness to abnormal operating conditions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- overload tolerance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subsystem requirements must specify hardware requirements, e.g. MTBF and MTTR, for maintainability.</td>
<td>TE2 SSS</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subsystem requirements review must verify that requirements are:</td>
<td>RID</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 5.4.3</td>
</tr>
<tr>
<td></td>
<td>- complete and complete</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- correct</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- feasible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- unambiguous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- traceable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subsystem requirements review must verify that requirements are:</td>
<td>RID</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 5.4.3</td>
</tr>
<tr>
<td></td>
<td>- complete and complete</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- correct</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- feasible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- unambiguous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- traceable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>All requirements at the highest level (e.g. WBS level 1) must be traceable</td>
<td>TE2 SSS (traces)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 4.3.15 a</td>
</tr>
<tr>
<td></td>
<td>Traceability must be traceable</td>
<td>RID</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 5.4.12</td>
</tr>
</tbody>
</table>

**Service & Maintainability**

- Obj 4.3.4
- Obj 4.3.4
- Obj 4.3.4
## ED-153 example: Failure analysis

<table>
<thead>
<tr>
<th>Obj No</th>
<th>Obj Title: Topic</th>
<th>Requirements</th>
<th>SWAL1</th>
<th>SWAL4</th>
<th>Clarification</th>
<th>Evidence</th>
<th>Process</th>
<th>Detail</th>
<th>Evidence</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| 3.3.2  | Failure Effects | Annex A Section A.2.3.3  
The effects of failure occurrence shall 3.3.2.1 be excluded. | N      | N      |              |          |         |        |          | Refer to 3.3.1 |

The hazards associated with software failure occurrences shall 3.3.2.2 be identified in order to further complete the list of hazards initiated during Risk Assessment and Mitigation process (e.g. FHA and further completed during FSSA).

The TRA assesses the effect on the functions of the equipment, i.e. the output it delivers (Operational Effect). The effect of certain provision can only be assessed by OPS and is subject of FHA.
## EQM example: Failure analysis

<table>
<thead>
<tr>
<th>Id</th>
<th>Requirement</th>
<th>EOM reference</th>
<th>S&amp;M</th>
<th>SWAL 4</th>
<th>SWAL 3</th>
<th>ED-153 reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sub-1-1</td>
<td>Subsystem design must be reviewed against architectural design constraints and design standards</td>
<td>RID</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 5.4.5 d</td>
</tr>
<tr>
<td></td>
<td>Algorithms must be described.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 4.3.4</td>
</tr>
<tr>
<td></td>
<td>Subsystem design must describe the use, version and configuration of COTS tools.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Obj 7.2.1, 7.2.4, 7.2.8, 7.2.10</td>
</tr>
<tr>
<td>Sub-2</td>
<td>Effect of failure of HW, SW and interfaces must be described.</td>
<td>TE2 TRA</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 3.1.5, 3.3.1, 3.3.2</td>
</tr>
<tr>
<td></td>
<td>Effect of failures of COTS tools must be described.</td>
<td>TE2 TRA, TE2 SSS</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 3.3.2</td>
</tr>
<tr>
<td></td>
<td>Effects of undesired COTS configurations or performance or stability issues.</td>
<td>TE2 SSS</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 7.2.6</td>
</tr>
<tr>
<td></td>
<td>COTS tools must not affect stability or performance of system.</td>
<td>TE2 SSS</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 7.2.6</td>
</tr>
<tr>
<td></td>
<td>Effect of release installation of software.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 4.5.4</td>
</tr>
<tr>
<td></td>
<td>Subsystem test description must be complete and accurate.</td>
<td>TE2 SSS</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 5.4.3 e</td>
</tr>
<tr>
<td>Sub-3-1</td>
<td>Software must be broken down into software items.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 4.3.5</td>
</tr>
<tr>
<td></td>
<td>Software requirements must be allocated to software items.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 4.3.15 b</td>
</tr>
<tr>
<td></td>
<td>Software item interfaces must be described.</td>
<td>TE2 SSDD</td>
<td>X</td>
<td></td>
<td></td>
<td>Obj 3.1.1</td>
</tr>
</tbody>
</table>

**Obj 3.3.2**

**Subsystem Design (SuD)**
Phase applicability depends on SWAL:

<table>
<thead>
<tr>
<th>Phase</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SU</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
<th>SWAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Requirements Analysis (SuRA)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Design</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subsystems Requirements Analysis (SuRA)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subsystem Design (SuD)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Design (SuD)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coding and Unit Testing (CUT)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Integration (Si)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subsystem Integration (SyI)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Integration (SI)</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td>SWAL</td>
<td></td>
<td>SWAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Subsystems Requirements Analysis (SuRA)**
- Specify system design, identify subsystem requirements.
- Design software architecture (HW, SW, COTS).
- Trace design to SW requirements.
- Decompose SW into software units.
- Software components integration.

**Subsystem Design (SuD)**
- Develop software, perform code review.
- Plan software verification, integrate software units.
- Execute software verification.
- Execute subsystem verification, verify SWL compliance.

**MNP scope**
- Specific test descriptions.
- Final system verification.
- Initial software releases in TES and test.
- Initial subsystem releases in ONL.
- Execute ONL verification.
- Execute subsystem releases.

---

**MUAC - SSAS Procedure**
MNP: EQM compliance matrix

- Completed as part of each MNP for each CI
- No “cryptic” requirements
- Fulfillment of each requirement to be justified with reference to evidence
Development process summary

SWAL 4:
- Subsystem requirements, review, traceability
- Subsystem design
- Subsystem tests, traceability
- Systematic failure mode analysis
- Subcontractor deliverable review/acceptance (SWAL)
- Development tools identification
- Tools and COTS assurance

SWAL 3 (in addition to SWAL 4):
- SW decomposition, SW item requirements, review, traceability
- SW item tests, traceability
- Design choices/rationales + standards
- Failure mode testing

Service & Maintainability:
- Coding standards
- COTS from reputable vendors
Reflection over EQM extension

Did it help?
## Ongoing: Security extensions

<table>
<thead>
<tr>
<th>Phase</th>
<th>System Requirements Analysis (SyGA)</th>
<th>System Design (SyD)</th>
<th>Subsystem Requirements Analyzing (SSA)</th>
<th>Subsystem Design (SSD)</th>
<th>Software Requirements Analysis (SSRA)</th>
<th>Software Design (SSD)</th>
<th>Coding and Unit Testing (CUT)</th>
<th>Software Verification (SV)</th>
<th>Subsystem Verification (Sv)</th>
<th>Subsystem Integration (SI)</th>
<th>System Verification (SV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase applicability</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
<td>SWAL 2</td>
<td>SWAL 4</td>
<td>SWAL 2</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
<td>SWAL 4</td>
</tr>
<tr>
<td>Inputs</td>
<td>MC(S)</td>
<td>TST SSS</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSS</td>
<td>TST SSS</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
</tr>
<tr>
<td>Outputs</td>
<td>Authorization/realization.</td>
<td>TST SSS</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
<td>TST SSD</td>
</tr>
</tbody>
</table>

**Security assessment**

**Secure coding**

**Virus scanning**

**Hardening test report**

**MNP scope**
Overview

• SSAS procedure in MUAC SMS:
  • Overview of SSAS central process and sub-processes.

• Method for SW Assurance in projects/developments
  • Process and tools (AMC) adopted for projects/developments @MUAC and between ANSP and manufactures

• Method for SW Assurance in maintenance:
  • Maintenance process with SW assurance as an integrated set of activities

• Conclusions
Conclusion 1

- Difference between MUAC and suppliers’ expectations for development processes and tools (AMC) application in projects
- Difference of development processes and compliance approaches for recent and old legacy systems during maintenance @MUAC
- Involvement/understanding of stakeholders about SSAS application
- Compliance for COTS and legacy – Problems of in-service experience monitoring in evolving configuration and used environment
- Unintended functions (potential customization and configurations for NCRS and COTS)
Conclusion 2

- Component definition (SWAL3)
  - System decomposition (where is what I need? What should I analyse? I should get there… long way and many requirements I need an iterative method to focus in the detail review)
- SRS completeness and correctness:
  - Design documents cover the behaviour and input/output… usefulness not seen
  - Gap between system engineer terminology and software engineer terminology
  - System-software engineer is the same = usefulness?
- Identification of pitfalls after selection of AMC (need guidance on available options and limitations)
  - E.g. Ed-153 – Depth of design transparency
Conclusion 3

- SW Safety Requirements? Where do they come from? Forgot? Are we focusing just on quality process?

- Invest on Software Safety Requirements derivation and implementation analysis:
  - Analysis of DSR from RA and mapping in SRS/design
  - SW Safety Assessment Techniques, e.g. SW FMEA and SHARD
  - HF-Safety Techniques, e.g. HMI design assessment
  - Safety impact assessment of unspecified functions (on the table and in validation)
  - Review of Risk Assessments and SRS from analysis of occurrences
SQS, SSQ, QSS?

• Management process evolution:
  • EC 1035/2011, sec 3.2: “Air navigation service providers may integrate safety, security and quality management systems into their management system” – that’s what we want.
  • Safety and Security is about management of specific risks
  • Quality defines and assures the overall process
Thank you for your attention

Questions?

marinella.leone@eurocontrol.int
morten-trier.hansen@eurocontrol.int