AMASS
Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems

AMASS Usage Scenario 4:
Safety and Security Co-Assessment

2nd EAB Workshop
Västerås, September 17, 2018

Thomas Gruber
WP4 Leader
Introduction

• Co-assessment is a central prerequisite for efficient assurance of safety and security (& other concerns):
  – Traditionally, co-engineering is supported by applying separate, safety specific and security specific tools.
  – For a few years, combined approaches have been a topic in research and are now producing first tools as results.

• Using separate tools has drawbacks:
  – results may (and mostly do) influence the assumptions for applying the other one.
  – An additional analysis of the mutual influences between the quality attributes (Supporting/Conflicting/Dependency Impact Relationship) and of the trade-offs between them is necessary.
  – At least one additional iteration of the (concern-specific, parallel) assurance steps is required to integrate the trade-off analysis results.
Dependency relationship.
• The claim A of one attribute depends on the fulfillment of claim B of another attribute.
• E.g. a fail-safe claim (safety) depends on safety system not tampered (security).

Conflicting relationship.
• The assurance measure of attribute A is in conflict with the assurance measure of attribute B.
• E.g. “strong password or blocking a terminal after several failed login attempts” (security) conflicts with “emergency shutdown” (safety).
• Resolution of such a conflict needs to be noted in the Assurance Case.

Supporting relationship.
• Assurance measure of attribute A is also applicable to assurance of attribute B => one assurance measure can be used to replace two separate ones.
• E.g., encryption can be used for both confidentiality (security) and to check data integrity instead of checksum (safety).
  => This means two goals can be addressed by one argumentation.
AMASS Reference Tool Architecture

**Architecture-Driven Assurance (STO1)**
- System Architecture Modeling for Assurance
- Assurance Patterns Library Management
- Additional Activities Supporting Assurance Case
- Contract-Based Assurance Composition

**Multi-Concern Assurance (STO2)**
- System Dependability Co-Analysis/Assessment
- Dependability Assurance
- Contract-Based Multi-concern Assurance

**AMASS Platform Basic Building Blocks**
- Access Manager
- Data Manager
- System Component Specification
- Assurance Case Specification
- Evidence Management
- Compliance Management
- Common Assurance & Certification Metamodel (CACM)

**Cross/Intra-Domain Reuse (STO4)**
- Automatic Generation of Process-Based Arguments
- Automatic Generation of Product-Based Arguments
- Reuse Assistant
- Process-Related Reuse via Management of Variability at Process Level
- Product-Related Reuse via Management of Variability at Product Level

**Seamless Interoperability (STO3)**
- Collaborative Work Management
- Tool Quality Assessment and Characterization

**AMASS**

First Guidelines edition delivered

Functionalities identified. Implementation provided and validated (core & P1 prototypes)

WP4

Independent Assessment
- Certification Liaison
- Safety/Security Assessment

Product Engineering
- Component Release
- Module Assurance Case Development

Cross/Intra-Domain Reuse (STO4)

2nd EAB Workshop, Västerås, Sept 17, 2018
How Standards deal with Co-Engineering

SAE-J3061 Cybersecurity Guidebook for Cyber-Physical Vehicle Systems

Guidebook and not a standard

Only available for a few months, then back to Work in Progress

Multiple methods proposed, but no consistent approach (e.g. risk rating differs on used method)

Process copied from ISO26262
Alignment is needed but cybersecurity needs to include later stages

„Potential Communication Path“
How Standards deal with Co-Engineering

ISO/SAE 21434 WD Road Vehicles - Cybersecurity Engineering

• Based on SAE-J3061 but much more detailed guidance
• Scope:
  – Requirements for cybersecurity risk management for road vehicles, their components and interfaces, throughout engineering (concept, design, development), production, operation, maintenance, and decommissioning.
  – A framework that includes requirements for cybersecurity processes and a common language for communicating and managing cybersecurity risk among stakeholders
  – applicable to road vehicles that include electrical and electronic (E/E) systems, their interfaces and their communications
  – Standard does not prescribe specific technology or solutions related to cybersecurity
  – Engineering rigor depends on CAL (Cybersecurity Assurance Level)
How Standards deal with Co-Engineering

IEC62443 Industrial communication networks – Network and system security

Security
Lifecycle
# How Standards deal with Co-Engineering

**IEC 62443 Industrial communication networks - Network and system security:**
Mapping between safety and security lifecycles

<table>
<thead>
<tr>
<th>Lifecycle Phase</th>
<th>Functional Safety</th>
<th>IACS Cybersecurity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target of Evaluation</td>
<td>• Equipment under control (EUC)</td>
<td>• Zones and Conduits based on logical grouping of assets</td>
</tr>
<tr>
<td>Failure Likelihood</td>
<td>• Random failures due to operational and environmental stresses</td>
<td>• Threats: Internal, external or combination</td>
</tr>
<tr>
<td></td>
<td>• Systematic failures due to errors during safety life cycle</td>
<td>• Vulnerabilities due to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o component or system design flaws</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o making non-validated changes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o not following security practices and procedures</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o Threats exploiting vulnerabilities Lead to failure</td>
</tr>
<tr>
<td>Risk Analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Consequence Severity</td>
<td>• Impact on environment, health and safety of personnel and the general public</td>
<td>• Loss of availability and/or data integrity has direct impact and loss of</td>
</tr>
<tr>
<td></td>
<td></td>
<td>confidentiality has indirect impact on functional safety</td>
</tr>
<tr>
<td>Risk Categorization</td>
<td>• Based on likelihood and severity; risk may be quantified</td>
<td>• Based on likelihood and severity; risk is currently qualitative</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Risk categorization for every security requirement;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>multi-dimensional problem</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Assigned to Zone with target SL for each zone/conduit</td>
</tr>
<tr>
<td>Risk Mitigation Measures</td>
<td>• Relies on independent protection layers concept</td>
<td>• Relies on security countermeasures within conduits connected to the Zone, and</td>
</tr>
<tr>
<td></td>
<td>• Safeguards reduce likelihood of consequence evaluated</td>
<td>defense in depth concept</td>
</tr>
<tr>
<td></td>
<td>• Identifies integrity requirements for safeguards; for SIF assigns target SIL</td>
<td>• Countermeasures reduce likelihood</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Identifies requirements for countermeasures to meet the Zone Target SL for each</td>
</tr>
<tr>
<td></td>
<td></td>
<td>threat vector</td>
</tr>
<tr>
<td>Implementation of Measures</td>
<td>• Safety manual for components</td>
<td>• Security manual for components</td>
</tr>
<tr>
<td></td>
<td>• Quantitative SIL verification for SIF</td>
<td>• Verification through different Levels of testing for target SL</td>
</tr>
<tr>
<td>Operation and</td>
<td>• Restrict access to IACS components to competent personnel with necessary access</td>
<td>• Restrict access to IACS components to competent personnel with necessary access</td>
</tr>
<tr>
<td>Maintenance</td>
<td>privileges</td>
<td>privileges</td>
</tr>
<tr>
<td></td>
<td>• Periodic testing of measures</td>
<td>• Periodic testing of measures</td>
</tr>
<tr>
<td></td>
<td>• Demand rate and component failures to be monitored</td>
<td>• Frequent reviews to identify new vulnerabilities and</td>
</tr>
<tr>
<td></td>
<td>• Awareness and training</td>
<td>take appropriate action, if necessary</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Awareness and training</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cyber risk reassessment after each software or hardware change</td>
</tr>
<tr>
<td>Management System</td>
<td>• Defines requirements for competency, training, verification, testing, audit,</td>
<td>Defines requirements for competency, training, verification, testing, audit, MOC,</td>
</tr>
<tr>
<td></td>
<td>MOC, and documentation</td>
<td>and documentation</td>
</tr>
</tbody>
</table>

---

2nd EAB Workshop, Västerås, Sept 17, 2018
Two Ways to Realize Co-Engineering

Using separate tools

- MORETO
- IEC 62443
- WEFACT
- Security Process

Start

IAEC 61508
EN 50126/8/9
EN/ISO 13849
IEC 62061
IEC 61511

Using one combined tool

- e.g. APIS FMEA
- FMVEA

Start

IAEC 61508
EN 50126/8/9
EN/ISO 13849
IEC 62061
IEC 61511

IAccess

The real challenge is the trade-off analysis

- AQUAS
- AMASS
Co-Engineering Processes

- 2 ways of realizing co-engineering
- AMASS prefers efficient combined tools
- Other projects rely on separated ones, e.g. AQUAS, whose interaction point approach is similar to "potential communication paths" in SAE-J3061 "Cybersecurity Guidebook for Cyber-Physical Vehicle Systems"
- Standardization for safety and security is still separate. In the case study we used:
  - For safety
    - IEC 61508 Functional Safety of Electrical / Electronic / Programmable Electronic safety-related systems
  - For security
    - IEEE 1686-2013 - IEEE Standard for Intelligent Electronic Devices Cyber Security Capabilities, and
    - IEC 62443 Industrial communication networks – Network and system security
Multiconcern Assurance Scenario Overview

• Developing an Industrial Automation domain CPS in Case Study 1: Industrial and Automation Control Systems (IACS)

• Different tools are used for system analysis and requirements generation (so far MORETO, in the 3rd project year FMVEA)
  – FMVEA is included in the recent delivery of the 3rd AMASS platform iteration P2 as an external tool.

• The AMASS Platform is used for assurance & certification-specific activities:
  – Security analysis and security requirements allocation in compliance with the requirements of IEEE 1686-2013 “IEEE Standard for Intelligent Electronic Devices Cyber Security Capabilities” and IEC 62443 “Industrial communication networks - Network and system security”
  – Combined safety and security analysis in compliance with IEC 61508 and IEC 62443 avoiding iterations due to conflicts detected in the trade-off analysis

• The company aims to be able manage safety and security analysis; risk assessment based on a common model in the AMASS Platform
Higher-level objectives & expected gains

• **O2:** define a *multi-concern assurance* approach to ensure not only safety and security, but also other dependability aspects such as availability, robustness and reliability.

• **Metrics**
  – Effort for assurance and certification
  – Effectiveness in failure/threat identification capabilities
  – Number of requirements fed back into the model
  – Time needed for separate safety and security engineering process and the co-engineering process
  – Architectural/design modifications saved by combined safety/security co-engineering

• **G1:** to demonstrate a potential gain for design efficiency of complex CPS by reducing their assurance and certification/qualification effort by 50% (STO1&2).
Intro to MORETO & WEFACT/FMVEA Scenarios

- Case Study 1: Industrial Automation domain: Industrial and Automation Control Systems (IACS)
- Usage Scenario 2: Perform safety and security co-assessment
- Timeline:

<table>
<thead>
<tr>
<th>Platform Iteration</th>
<th>1st Iteration - Pcore</th>
<th>2nd Iteration – P1</th>
<th>3rd Iteration – P2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool</td>
<td>MORETO (Eclipse)</td>
<td>MORETO (Enterprise Architect)</td>
<td>FMVEA (Browser) (+WEFACT (Eclipse))</td>
</tr>
<tr>
<td>Safety Standards</td>
<td>-</td>
<td>-</td>
<td>IEC 61508</td>
</tr>
<tr>
<td>Security Standards</td>
<td>IEEE 1686 IEC 62443¹</td>
<td>IEEE 1586 IEC 62443</td>
<td>IEC 62443</td>
</tr>
</tbody>
</table>

¹) Not yet for RTU
Scenario in 2nd Iteration P1

• Design System in MORETO model editor or import SysML model
• Add security relevant properties including already present security controls into model
• Start requirements generation
• Feed back corresponding security controls into the model (with IEC 62443: corresponding to SL-T (Target security level))
Scenario in the 2nd Iteration P1 with MORETO

• Workflow

- Import model via SysML
- OR
- Create model in MORETO
- Enhanced System model
- Feed back mitigation measures into model
- Safety & Security Requirements
- Internal DB

IEEE, EC, ISO, CENELEC

AMASS platform P1

AMASS platform P1
MORETO Workflow

Modeling

MORETO Diagrams

MORETO Modeling Process

Network Topology Diagram
Data Flow Diagram
Remote Terminal Unit
Internal Block Diagram

Security Analysis

Security Requirements

Reporting

Scripts
Patterns
Drag and Drop
CSV

Automatically
Manually
Self-requirements generation
Manual Requirements Generation

Security Requirements Generation

Full Documentation
4 different diagrams for the system modeling process:

- **Block Definition Diagram (BDD)** for network elements
- **Internal Block Diagram (IBD)** for detailed modeling
- **Dataflow Diagram (DFD)** for Threat Modeling
- **Requirement diagram for security requirements**
External / Intermediate / Internal Layer

The external layer = network architecture

The internal layer = further details about components

The intermediate layer = component details
Requirements Generation

- **Manual Mode:**
  - With Drag and Drop, or
  - Importing a CSV File

- **Automated mode:**
  - With patterns
  - With Scripts
WEFACT and FMVEA Presentation

- Workflow

[Diagram showing workflow with WEFACT and FMVEA integrated, mentioning tools like Papyrus, MORETO, CHESS, and models like SysML and ReqIF.]
Activities in Iteration 3

- The WEFACT workflow engine executes the tool FMVEA
- The user creates the system model with dependability-relevant properties in FMVEA or imports a SysML model and enhances it with the required properties.
- The user can add rules in FMVEA or re-uses a previously created threat and failure database with these rules. In the case study, the rules correspond to the requirements of the applied standard.
- The database is applied to the system model yielding respective safety and security requirements.
  - In the case study we focus on safety and security, but these rules are not restricted to these quality attributes. The user could as well include multiple concerns, e.g. add a performance requirement like a WCET or a maximum memory usage.
- Two outputs:
  - These security requirements are fed back into the model manually and/or automatically.
  - The requirements are imported in WEFACT to create the executable assurance processes which create the evidences
WEFACT Lifecycle Activity for HARA/TARA

- Eclipse RCP application WEFACT is started
- Its process model (edited in WEFACT or imported from EPF-C in UMA dialect format) has associated lifecycle activities.
- In the Lifecycle the HARA/TARA phase is reached by the workflow
- The process starts the FMVEA tool
FMVEA Starts with the Model Editor

- Model (with dependability relevant properties) can be edited in FMVEA tool or imported eg. from Papyrus/CHESS via SysML
Safety/Security Rules Are Defined or Re-Used from DB

- FMVEA allows do define rules

Rule Definition

- Or to use a previously created database, e.g. realizing the rules of a specific standard
Automated Rule-based Safety/Security Analysis

- FMVEA applies its rules on model elements and thereby performs the FMVEA safety and security analysis.

### Analysis Results

<table>
<thead>
<tr>
<th>#</th>
<th>Affected Element</th>
<th>Threat</th>
<th>Likelihood</th>
<th>Impact</th>
<th>Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NFC</td>
<td>Robot receives wrong information from NFC communication from machinery</td>
<td>2</td>
<td>F1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>WIFI</td>
<td>Eavesdropping of wifi Signal</td>
<td>2</td>
<td>C4</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>Robot</td>
<td>Behaviour is not under control</td>
<td>5</td>
<td>F3</td>
<td>5</td>
</tr>
</tbody>
</table>
- Requirements generated according to the rules

<table>
<thead>
<tr>
<th></th>
<th>Affected Element</th>
<th>Requirement</th>
<th>Description</th>
<th>SL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WIFI</td>
<td>The WIFI Communication shall be encrypted</td>
<td>Encrypt the WIFI with appropriate algorithm.</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Limit the range of the WIFI signal</td>
<td>Restrict the area where the WIFI signal is active.</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>NFC</td>
<td>Integrate a sensor for product validation</td>
<td>Install a Sensor on Robot to detect wrong information from machinery.</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The Robot shall alarm when it detects irregularities</td>
<td>After the robot received wrong information from the machinery he was able to detect the error and then reports the error.</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Robot</td>
<td>The Robot shall have emergency shutdown</td>
<td>Robot needs emergency stop which is always active. (Physical)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The Robot shall only operate in a restricted area</td>
<td>Restrict the movement of the robot with physical borders.</td>
<td>2</td>
</tr>
</tbody>
</table>

- Export function to open format ReqIF
## FMVEA in an Excel File

- This shows a sheet for a manual FMVEA analysis -> Effort intensive

<table>
<thead>
<tr>
<th>Element</th>
<th>Threat / Failure Mode</th>
<th>STRIDE(failure)</th>
<th>Direct Effect</th>
<th>System Effect</th>
<th>Impact</th>
<th>Cause</th>
<th>Likelihood</th>
<th>Risk</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFC Zone</td>
<td>Manipulated data</td>
<td>Tampering with data, Denial of Service</td>
<td>Machinery receives wrong information from NFC communication from robot</td>
<td>Machine stops (wrong data detected)</td>
<td>F2</td>
<td>Insider attack</td>
<td>2</td>
<td>4</td>
<td>No clarification of how the robot behaves, we decided that the robot should stop. But this may result in the worst financial impact.</td>
</tr>
<tr>
<td>NFC Zone</td>
<td>Manipulated data</td>
<td>Tampering with data, Denial of Service</td>
<td>Robot receives wrong information from NFC communication from machine</td>
<td>Robot stops and alarms (wrong data detected)</td>
<td>F1</td>
<td>Insider attack</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>NFC Zone</td>
<td>Manipulated data</td>
<td>Tampering with data, Denial of Service</td>
<td>Robot receives wrong information from NFC communication from product</td>
<td>Robot stops and alarms (wrong data detected)</td>
<td>F1</td>
<td>Insider attack</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Jamming</td>
<td>Denial of Service</td>
<td>Jamming of NFC Signal</td>
<td>Machinery and Robot stop because weak or anomalous signal is detected</td>
<td>F3</td>
<td>Jammer signal reaches factory</td>
<td>5</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eavesdropping</td>
<td>Information Disclosure</td>
<td>Eavesdropping of NFC Signal</td>
<td>Cannot detect eavesdropping.</td>
<td>C4</td>
<td>Signal reaches the receiver’s antenna</td>
<td>5</td>
<td>20</td>
<td>Countermatrix: Use authenticated encryption with key that was exchanged in a safe environment beforehand.</td>
<td></td>
</tr>
<tr>
<td>Wi-Fi Zone</td>
<td>Manipulated data</td>
<td>Tampering with data, Denial of Service</td>
<td>Robot receives wrong instructions via wifi.</td>
<td>Robot does not work properly.</td>
<td>O2</td>
<td>Attack</td>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Mobile Device</td>
<td>Wrong configuration is written on NFC tag by the mobile device</td>
<td>Tampering with data</td>
<td>We assume that the robot does not find the correct product because the product received a wrong NFC tag from the mobile device</td>
<td>The robot cannot execute its command as it cannot find the associated NFC tag.</td>
<td>F2</td>
<td>Insider Attack</td>
<td>3</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Robot</td>
<td>Manipulated configuration of the robot</td>
<td>Denial of Service</td>
<td>Behaviour is not under control.</td>
<td>Unexpected behaviour of the robot.</td>
<td>F3</td>
<td>Insider attack</td>
<td>5</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Entire System</td>
<td>High Attitude Electromagnetic Pulse (HREP)</td>
<td>Denial of Service</td>
<td>Electronic devices are damaged.</td>
<td>Production is stopped completely, multiple devices harmed, devices may work wrongly, unexpected incidents</td>
<td>4</td>
<td>Attack</td>
<td>1</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>
• Requirements lead to security controls whose presence/correctness must be assured.
• This leads back to WEFACT activities which implement the executable assurance processes for these new requirements
Scenario Outcome (expected in P2)

- Reduced effort for assurance and certification through automation and co-assessment
- Effectiveness in failure/threat identification capabilities by following standards
- Update of the model according to the requirements created
- Time needed for separate safety and security engineering process is significantly reduced when applying the combined co-engineering process
- Iterations for deriving requirements for architectural or design modifications reduced by combined safety/security co-engineering
Y2 Achievements: Delivery for P2 on August 31st, 2018

- System dependability co-analysis via ConcertoFLA
- Contract-based trade-off analysis in parameterized architectures
- Abstract functions in the contracts specification
- Contract-based trade-off analysis with Analytical Network Process
- Multiconcern contracts in assurance via argument-fragment generation
- General extensions to contract based multi-concern assurance
- Many functional extensions delivered in: D4.6 Prototype for multi-concern assurance (c)

FMVEA - Failure Modes, Vulnerabilities & Effects Analysis

External tool: Medini Analyzer

P1 ☑ openCert Assurance Case Editor

SiSoPLE for enabling process-related co-assessment
System Dependability Co-analysis via ConcertoFLA

- A compositional technique to qualitatively assess the dependability of component-based systems. Users can
- decorate CHESSML models with dependability-related information
- execute Failure Logic Analysis (FLA) techniques (based on Failure Propagation Transformation Calculus (FPTC) and using the FlaMM meta model), and
- calculate the failure behaviour of a component-based system at system-level, based on the specification of the failure behaviour of the individual components
- get results back-propagated onto the original model.
Medini Analyzer

medini analyze — a Model based and System oriented Solution

Safety Analyzer available for a long time
Features:
• Eclipse based
• PHA, HARA, FMEA, FMEDA, FTA, HAZOP, Safety requirements & plans

SESAMO: prototype for security analysis

Cybersecurity analysis now available as a mature commercial tool
Features:
• Attack trees, TARA, Security FMEA, Security requirements allocation, confirmation measures
• AMASS integration via
• SCADE Architect

Cybersecurity threat monitoring and analysis methods required to protect vehicles from attacks
openCert Assurance Case Editor

• Edit in graphical GSN syntax, internal data in SACM,
• Support multiconcern argumentation
• Argument patterns,
• Assurance case contracts
• Implements Basic Building block „Assurance case specification“
• Delivered in Pcore and P1
General Extensions to Contract-based Multi-concern Assurance

- Extensions in CHESS for contract-based trade-off analysis
- Model treats conflicts between dependability attribute-specific goals
Out of defined processes for generating evidences, corresponding arguments for the assurance case can be generated.

Feature strongly related to WP6 and therefore described there.
Extended Safety Architect

- All4Tec commercial tool
- Supports FMEA, FTA with automatic detection of the FE (feared events),
- extended during the MERgE ITEA and French Clarity Project to support Safety and Security Co-Analysis
- For describing failure and threat propagation, Safety Architect provides safety view, security view & merged view
- dysfunctional analysis techniques applied for automatic fault or attack tree generation
- interfaces with many system engineering tools, such as Capella, System Architect, Papyrus, and the AMASS platform
• Apart from Contract-Based Multiconcern Assurance (STO2), tool is also related to Architecture-driven Assurance (STO1)
• deploy methods for monitoring and diagnosing Cyber-Physical System (CPS) models in Simulink
• translating informal system specifications into formal specification expressed in the extended Signal Temporal Logic (STL)
• Tool integrates existing monitoring techniques at AIT to the Simulink environment (CS3 in P2)
• Novel methods developed for system diagnosis and error localization in the Simulink models upon the detection of the specification violations.
Conclusion

• Guidance from standards getting slowly improved
• Other projects take partly comparable approaches
• Progress based on CS1, US2 demonstrated
• Integrating heterogeneous external tools essential
• Coupling workflows of single-concern tools necessary
• Few combined (integrated) tools available