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

Technical Vision

Second EAB Workshop
Västerås, Sept 17, 2018

Barbara Gallina, Ph.D.
TM, WP6 Leader, T6.1-2 Leader
AMASS technical contribution

AMASS objectives partially achieved

Goal \{G\} partially achieved

Objective \{O\} partially achieved

Scientific Technical Objective \{STO\} partially contributes to Objective \{O\}

Context of goal \{G\}, objective \{O\}, scientific technical objective \{STO\}, system \{X\} and Standard(s) \{Y\}

Evaluation framework given in D1.3

#m max 6
#n max 4

G={G1, G2, G3, G4}
O={O1, O2, O3, O4, O5, O6}
STO={STO1, STO2, STO3,STO4}

D1.3 - Evaluation Framework and Quality Metrics
O1: define a holistic approach for **architecture-driven assurance** to leverage the reuse opportunities in assurance and certification by directly and explicitly addressing current technologies and HW/SW architectures needs.

**Architecture-driven assurance** contributes to AMASS objective O1

Context of system \{X\} and Standard \{Y\}

Leverages reuse opportunities are in place

Support for Design/Assurance patterns constitutes a reuse opportunity

Contract-based assurance decomposition constitutes a reuse opportunity

Claim, claim decomposition, evidence

**Architecture-Driven Assurance (STO1)**

- System Architecture Modeling for Assurance
- Assurance Patterns Library Management
- Contract-Based Assurance Composition
- Additional Activities Supporting Assurance Case

Leverages reuse opportunities are in place

 kiểm tra lại để chắc chắn rằng nội dung được hiển thị chính xác và dễ đọc.
Architecture-driven assurance

Support for Design/Assurance patterns constitutes a reuse opportunity

Context of system \{X\} and Standard \{Y\}

Design/Assurance patterns have been conceptually analysed/designed

Claim, claim decomposition, evidence

Design/Assurance patterns have been conceptually analysed/designed

Claim, claim decomposition, evidence

2nd EAB Workshop, Västerås, Sept 17, 2018
01: define a holistic approach for *architecture-driven assurance* to leverage the reuse opportunities in assurance and certification by directly and explicitly addressing current technologies and HW/SW architectures needs.

**Contract-based assurance decomposition**

- **Assertion modelling and patterns**
  - **Contract-decomposition has been applied**
  - **Claim, claim decomposition, evidence**

- **Contract-editing capabilities in enabled**
  - **Contract-decomposition is implemented**

---

2nd EAB Workshop, Västerås, Sept 17, 2018
Architecture-driven assurance – Evidence on ...

Contract-decomposition has been applied

Context of a space system and ECSS Standards

CS4
Architecture-driven assurance – Evidence on ...

Assertion modelling and patterns → Context of system \{X\} and Standard \{Y\} → Claim, claim decomposition, evidence

Create a new Assumption or Guarantee based on patterns
- Select whether you want to create an assumption or a guarantee and choose a general pattern type.
- Select which kind of Assertion you want to define:
  - Assumption
  - Guarantee
- Select which Pattern you want to use:
  - General Patterns Type
    - Global Invariant Patterns
      - For nearly all systems we want to define conditions, which shall always hold, regardless of the state the system is currently in. The Global Invariant Pattern allows the definition of those conditions, as they do not have a restricted scope but need to be fulfilled at all points in time. Example: The supply voltage is always in the range from 5V to 12V.
    - Simultaneity Patterns
      - These Patterns are used to specify the dependency system behavior, that happens simultaneously. They can express the dependency of one condition to another or can state that a specific event is only allowed to occur while a certain condition holds. Example: While ignition occurs, car.key.status is ‘INSERTED’.
    - Trigger Reaction Patterns
      - System behavior can also stand in some trigger reaction relation to each other. So does some event occurrence always need to trigger another event or result in the satisfaction of a specific condition. Example: Whenever crash.detected occurs then in response airbag.ignition occurs during within 50ms.
Multi-Concern Assurance

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.

Multi-Concern Assurance contributes to AMASS objective O2

Context of system \{X\} and Standard \{Y\}

Safety and security are ensured

Contract-based multi-concern assurance is supported

System dependability co-assessment is supported

Dependability assurance modelling is partly supported

Claim, claim decomposition, evidence

Multi-Concern Assurance (STO2)

System Dependability Co-Analysis/Assessment
Dependability Assurance Modelling
Contract-Based Multi-concern Assurance

- Process engineer(s) addressing the security & safety process
- Architect jointly interacting with safety and security managers

ARP4761 DO-326A

2nd EAB Workshop, Västerås, Sept 17, 2018
Multi-Concern Assurance – Evidence on ...

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

System dependability co-assessment is supported

Context of automotive system and Standard ISO 26262 + Handbook SAE J3061

Security-informed safety process can be configured

SoftwareUnit: DesignAndImplementation: BVRModel

Activities

- (+) UnitDesign
  - "Commonality Point"

- UnitDesignReview
  - "Commonality Point"

- (+) UnitImplementation
  - "Commonality Point"

- (+) UnitImplementationReview
  - "Commonality Point"

ConcernChoice

DesignReviewSafety
- "Variability Point - ISO 26262"

DesignReviewCybersecurity
- "Variability Point - SAE J3061"

Safety implies ((DesignReviewSafety and (not DesignReviewCybersecurity)) and (not DesignReviewMultiConcern))

Cybersecurity implies ((DesignReviewCybersecurity and (not DesignReviewSafety)) and (not DesignReviewMultiConcern))

MultiConcern implies ((DesignReviewMultiConcern and (not DesignReviewSafety)) and (not DesignReviewCybersecurity))

2nd EAB Workshop, Västerås, Sept 17, 2018
02: define a *multi-concern assurance* approach to ensure not only safety and security but also other dependability aspects such as availability, robustness and reliability.

Contract-based multi-concern assurance is supported

Context of system \{X\} and Standard \{Y\}

Claim, claim decomposition, evidence
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.

Dependability assurance modelling is partly supported via internal Open Source Solutions: extension of ConcertoFLA (Dependability modelling extension) and via external Solutions: MORETO plugin.

The extension has been designed.

Claim, claim decomposition, evidence
Seamless interoperability contributes to AMASS objective O4

Management of tool integration is supported

OSLC connector developed

Context of system \( \{X\} \) and Standard \( \{Y\} \)

Management of collaborative work is supported

Claim, claim decomposition, evidence

Seamless Interoperability (STO3)

2nd EAB Workshop, Västerås, Sept 17, 2018
Seamless interoperability contributes to AMASS objective O4

Context of system \( \{X\} \) and Standard \( \{Y\} \)

Management of collaborative work is supported

Management of tool integration is supported

Integrity connectors have been conceived

Seamless interoperability of the platform with other tools is used in the development of CPSs.

Tool Integration Management

Collaborative Work Management

Tool Quality Assessment and Characterization

Seamless Interoperability (STO3)

Claim, claim decomposition, evidence

2nd EAB Workshop, Västerås, Sept 17, 2018
Cross and Intra Domain Reuse contributes to AMASS objective \( \{O\} \)

Context of system \( \{X\} \) and Standard \( \{Y\} \)

Reuse assistant supports reuse

Variability management at Product/Process level supports reuse

Cross-domain and intra-domain assurance reuse approach to improve mutual recognition agreement of compliance approvals and to help assess the return of investment of reuse decisions.
O3: consolidate a cross-domain and intra-domain assurance reuse approach to improve mutual recognition agreement of compliance approvals and to help assess the return of investment of reuse decisions.

Variability management
Process level supports reuse

Context of system ACS and Standard ECSS-40

Claim, claim decomposition, evidence
Compliance management contributes to AMASS objective O3

Compliance management can be shown, argued about, proved

Context of and Standard ECSS-40

Process-based argument generation is supported

Compliance checking is partly supported

O3: consolidate a cross-domain and intra-domain assurance reuse approach to improve mutual recognition agreement of compliance approvals and to help assess the return of investment of reuse decisions.
Compliance Management – Evidence on ...

O3: consolidate a cross-domain and intra-domain assurance reuse approach to improve mutual recognition agreement of compliance approvals and to help assess the return of investment of reuse decisions.

Compliance checking is partly supported

Context of ISO 26262

Compliance checking has been partly implemented

Compliance checking has been designed

Claim, claim decomposition, evidence

Regorous

EPF Composer

Transformation Engines

Process Structure & Compliance Annotations

Rule Set

Process Description

Compliance Annotated Process

Compliance Rule Base

Models & Annotations

Interpret

Formalizes

Supports

Analyzes

Improves

Compliance Effects

Standard

Custom Categories

- standard_requirements_iso_26262_software_unit_design
  - r1.address_software_unit_design_process
    - r1.1_address_software_unit_design_process
      - addressswunitdesigntprocess
  - r2.specify_software_units
    - r2.1.specify_software_units
      - performprovidesswarchitecturaldesign
      - performprovidessafetyrequirements
      - performspecifysoftwareunit
  - r2.2.incomplete_requirements_for Specify_software_u
    - -performspecifysoftwareunit
  - r3.describe_software_unit_specification
    - r3.1.strict_software_unit_specification_description
      - selectmandatorynotationsforswdesign
    - r3.2.taylored_software_unit_specification_description
      - providerationalformselectmandatorynotationsf
      - -selectmandatorynotationsforswdesign
  - rule_set_r_iso_26262-software_unit_design
Thank you for your attention!

Any questions