Structured Assurance Cases

and C&A

First Presented at:

ICSM 2003 in Amsterdam
expanded for
ICSM 2005 in Budapest
Structured Assurance – What’s in it for C&A

What DAA/DER* gets out of Structured Assurance

- Identification of all the artifacts to support the submission
- ‘Vertical’ relationships among artifacts – what they support
- Demonstration that (at least in theory) the submitter has given thought to meeting assurance expectations

What’s Missing

- ‘Horizontal’ relationships among artifacts – dependencies
- Process(es) executed by project in generating the artifacts
- Adequacy of time allocated to the processes that produce the artifacts

* DAA/DER = Designated Approving Authority/Designated Engineering Representative
Missing – Adequacy of Processes & Schedule

What are the sequences and dependencies?

Are the activities that produce the product & evidence scheduled reasonably? (i.e., sequence, duration, budget)
What a DAA/DER Would Really Like to See

As a minimum – the “standard” material

- The assurance artifacts
  - Preferably presented in order of the assurance case
- Documentation of how they support assurance expectations
  - Preferably with a structure that matches that of the controlling documents

For higher levels of confidence

- Evidence of planning the development of evidence drivers
- Evidence of adequate schedule for developing these drivers
- Evidence of adequate scheduling of the activities that result in execution evidence
- Evidence of execution of that schedule
How to Meet DAA/DER High-Assurance Expectations

Demonstrate that

} not only do the artifacts conform, but also
} the process addresses the criteria fully, early & continuously
The V-Chart View Helps Develop the Plan

- Specifies both development and assurance process activities
- Identifies and sequences requisite evidence
- Positions evidence production to where it optimizes C&A success
Basis – Waterfall Process and V-Chart

Concept
Plan & Rqmts
Prod. Design
Detail Design
Code
Unit Test
Integ. Test
Syst. Test
Accept. Test

V-Chart View

Change the shape from waterfall to fall and rise

Traditional Waterfall

Accept. Test
System Test
Integration Test
Unit Test
Code
Detail Design
Product Design
Plans and Requirements

Concept
NOTE: In the context of this paper, “Acceptance Test” includes all customer acceptance tests, not only the supplier’s. This may include testing purpose-built software and/or pre-purchase evaluation of COTS software.
Development V-Chart
Typically, Evaluation Criteria Get Written on the Way Up

The “Just-in-Time” Syndrome
Development V-Chart
We *Should* Establish Evaluation Criteria on the Way Down

- Needs
  - Needs Stmt
  - Accept Criteria
  - Design Spec
  - Test Spec
  - Design Spec
  - Test Spec
  - Design Spec
  - Test Spec

- System Requirements
  - Test Spec
  - Design Spec
  - Test Spec
  - Design Spec
  - Test Spec

- Subsystem Requirements
  - Test Spec
  - Design Spec
  - Test Spec
  - Design Spec
  - Test Spec

- Element Requirements
  - Test Spec
  - Design Spec
  - Test Spec
  - Design Spec
  - Test Spec

- Accept. Test
  - Test Spec
  - Test Spec
  - Test Spec

- System Test
  - Test Report
  - Test Report
  - Test Report

- Subsystem Test
  - Test Report
  - Test Report
  - Test Report

- Element Test
  - Test Report
  - Test Report
  - Test Report

Earlier intermediate products on this side

Same end-products on this side
Be sure we’re asking for – and *testing* for – the right stuff before we work our way downward.
**Development V-Chart**

Plan Early for Verify & DFT/DFM

**Needs**
- Needs Stmt
- Plan V&V
- Accept Criteria
- Design Spec
- Test Spec

**System Requirements**
- Test &V
- Design Spec
- Plan V&V
- DFT/DFM Feedback

**Subsystem Requirements**
- Test &V
- Design Spec
- Plan V&V
- DFT/DFM Feedback

**Element Requirements**
- Test &V
- Design Spec
- Plan V&V
- DFT/DFM Feedback

**Elements**

DFT – Design for Test(ability)
DFM – Design for Maintainability

Testers provide early feedback – better still, they should participate in design process.
The Development V-Chart – helps provide process confidence

... but how does this relate to building the Structured Assurance Case?
Yes, for two reasons:

1. To provide evidence that the design was done right in the first place.
2. When the requirements change, one can retrace the design process and know where to change the design.

Recording the rationale for decomposition helps with both.
Customer Assurance V-Chart
How can the customer develop assurance requirements?

Add an earlier phase that defines assurance case expectations before a product is designed and produced support a claim and how evidence supports the subclaims.

Define the processes, methods, and tools for evaluating the claims.

- Expectations, standards, desiderata
- Are decomposed into
  - Sub-claims re Expectations
  - Sub-claims re Expectations
  - Sub-claims re Expectations
- Evidence-Item Classes
- Body of Evidence

Argument
Add a earlier phase that defines assurance case expectations before a product is designed and produced support a claim and how evidence supports the subclaims.

Claim evaluation
- Sub-claim evaluation
- Appraisal
- Sub-claim evaluation
- Appraisal
- Sub-claim evaluation
- Appraisal
- Acceptance

Appraisal
- Claim evaluation
- Appraisal
- Sub-claim evaluation
- Appraisal
- Sub-claim evaluation
- Appraisal
- Acceptance

Argument
Add a earlier phase that defines assurance case expectations before a product is designed and produced support a claim and how evidence supports the subclaims.

Argument
Add a earlier phase that defines assurance case expectations before a product is designed and produced support a claim and how evidence supports the subclaims.

Argument
Add a earlier phase that defines assurance case expectations before a product is designed and produced support a claim and how evidence supports the subclaims.
Three main options:
- Evidence classes
- Arguments
- Appraisal methods

Customer Assurance V-Chart
How much of this gets communicated to the supplier?
Customer Assurance V-Chart
How much of this gets communicated to the supplier?

Three main options:
- Evidence classes
- Arguments
- Appraisal methods

Evidence classes: Recommended: Tell supplier what kinds of evidence are expected (and maybe in what form to present it)

Customer Assurance V-Chart
Expectations, standards, desiderata
are decomposed into

Claims re Expectations
are decomposed into

Sub-claims re Expectations
are decomposed into

Sub-claims re Expectations
are decomposed into

Evidence-Item Classes
Body of Evidence
Expectations, standards, desiderata are decomposed into:
- Evidence classes
- Arguments
- Appraisal methods

Arguments:
Neutral: Tell supplier why that evidence is expected. May support development and certification, but can help to “game the system.”

Three main options:
- Evidence classes
- Arguments
- Appraisal methods

Customer Assurance V-Chart
How much of this gets communicated to the supplier?
Customer Assurance V-Chart
How much of this gets communicated to the supplier?

Three main options:
- Evidence classes
- Arguments
- Appraisal methods

Appraisal Methods:
Not recommended:
- Adds little to development and certification processes.
- Can easily be used to game the system.

Evidence-Item Classes
are decomposed into
Sub-claims re Expectations
are decomposed into
Sub-claims re Expectations
are decomposed into
Sub-claims re Expectations
are decomposed into
Claims re Expectations
are decomposed into
Expectations, standards, desiderata
are decomposed into

Body of Evidence

Appraisal

Claim evaluation

Sub-claim evaluation

Sub-claim evaluation

Sub-claim evaluation

Acceptance
Expectations, standards, desiderata

Claim: Expectations

Sub-claims re Expectations

Sub-claims re Expectations

Evidence-Item Classes

Body of Evidence

Acceptance

Claim: evaluation

Sub-claim: evaluation

Sub-claim: evaluation

Synthesis structure is a Claim: evaluation assurance case.

Decor is a Body of Evidence

A strong Assurance Case requires well-planned, well-gathered and well-organized evidence.

Remember the “rationale” for decomposition? This is where it comes in handy.

Evidence is gathered in accordance with the customer-identified evidence classes.
Arguing a Safety Case: Basic Supplier Approach

CLAIM
Architecture is safe

ARGUMENT
Partitioning is proper
EVIDENCE Partitioning Documentation

ARGUMENT
Multi-versioning Dissimilarity is properly provided
EVIDENCE Multi-versioning Documentation

ARGUMENT
Safety Monitoring is provided
EVIDENCE Safety-Monitoring Documentation

Note that there is no justification that the three given arguments support the claim.
CLAIM
2.3
Architecture is safe

ARGUMENT
2.3.1
Partitioning is proper
EVIDENCE
2.3.1
Partitioning Documentation

ARGUMENT
2.3.2
Multi-versioning Dissimilarity is properly provided
EVIDENCE
2.3.2
Multi-versioning Documentation

ARGUMENT
2.3.3
Safety Monitoring is provided
EVIDENCE
2.3.3
Safety-Monitoring Documentation

Still no justification, but references (in this case, based on DO-178B) are provided.
Arguing a Safety Case: Justified Supplier Approach

CLAIM 2.3
Architecture is safe

ARGUMENT
Partitioning, multi-versioning, and safety monitoring are necessary and sufficient for a safe architecture

CLAIM 2.3.1
Partitioning is proper

EVIDENCE 2.3.1
Partitioning Documentation

CLAIM 2.3.2
Multi-versioning Dissimilarity is properly provided

EVIDENCE 2.3.2
Multi-versioning Documentation

CLAIM 2.3.3
Safety Monitoring is provided

EVIDENCE 2.3.3
Safety-Monitoring Documentation

Justification can be provided by the supplier to support the claim ...
but there is no guarantee that this will satisfy the customer.
Arguing a Safety Case: Requirements-Based Approach

ARGUMENT 2.3
Partitioning, multi-versioning, and safety monitoring are necessary and sufficient for a safe architecture

REQUIREMENT 2.3
Architecture is safe

REQUIREMENT 2.3.1
Partitioning is proper

EVIDENCE 2.3.1
Partitioning Documentation

REQUIREMENT 2.3.2
Multi-versioning Dissimilarity is properly provided

EVIDENCE 2.3.2
Multi-versioning Documentation

REQUIREMENT 2.3.3
Safety Monitoring is provided

EVIDENCE 2.3.3
Safety-Monitoring Documentation

Customer defines that architecture must be safe

Customer specifies what “safe architecture” means

Customer identifies expectations regarding proof that architecture is safe
ARGUING 2.3
Partitioning, multi-versioning, and safety monitoring are necessary and sufficient for a safe architecture.

EVIDENCE 2.3.1
Partitioning Documentation

EVIDENCE 2.3.2
Multi-versioning Documentation

EVIDENCE 2.3.3
Safety-Monitoring Documentation

ARGUMENT 2.3
Partitioning, multi-versioning, and safety monitoring are necessary and sufficient for a safe architecture.

CLAIM 2.3
Architecture is safe

ARGUMENT 2.3
Partitioning, multi-versioning, and safety monitoring are necessary and sufficient for a safe architecture.

CLAIM 2.3
Architecture is safe

Works remarkably like a V-Chart, doesn’t it?
The Bottom Line –
C&A Process is Labor-Intensive and Expensive

- **Structured Assurance Cases at process start**
  - Set clear expectations
    - saves the government money
    - saves the supplier (and thus government) time and money
  - Easier for both to prepare and evaluate evidence
  - Provide reduction/avoidance of failure due to misunderstood expectations

- **Structured Assurance Cases during process repetition**
  - Easier to identify inadequacies
  - Traceability structure helps eliminate redundant evaluation
    - no need to re-check where no change in evidence has taken place