Modular Certification

Tim Kelly
E-mail: tim.kelly@cs.york.ac.uk

Acknowledgements to the Industrial Avionics Working Group (IAWG)

High Integrity Systems Engineering Group
Department of Computer Science
THE UNIVERSITY of York

Overview

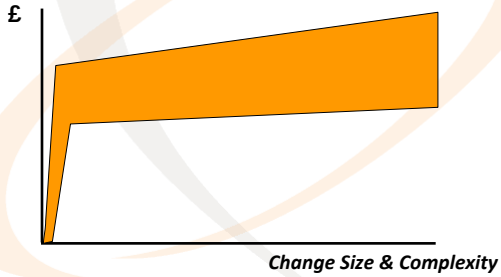
- Background to the IAWG study
- Motivation for Modular Certification
- Representing Modular Safety Cases
- IAWG Studies
  - Aims
  - Experience
- Safety Case Contracts
- Follow-on Work
- Applications of Modular GSN

History of IAWG

- Initially formed in 1979
- IAWG companies support a programme of joint activity
  - led, through EAP to Eurofighter Typhoon
- Since then the Companies have continued to work together
- Companies represented:
  - BAE SYSTEMS
  - Westland Helicopters
  - General Dynamics (United Kingdom) Limited
  - Smiths Aerospace
  - Selex S&AS
- One of current work areas is ***modular and incremental certification techniques for software***

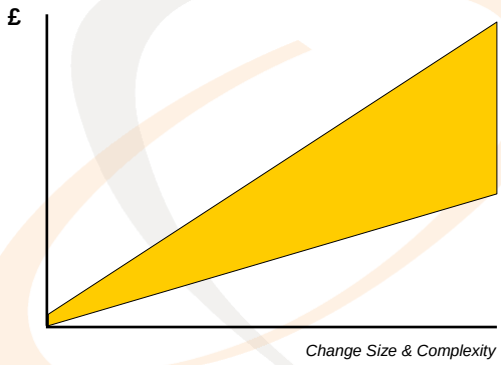
Modular Certification – Why?

- The costs of *change* have become a major part of the cost of ownership of a system
- Currently, the costs of re-certification of a system following any change account for the greater part of the cost of change



Cost of Re-Certification is Related to System Size and Complexity

Modular/Incremental Certification Ambition

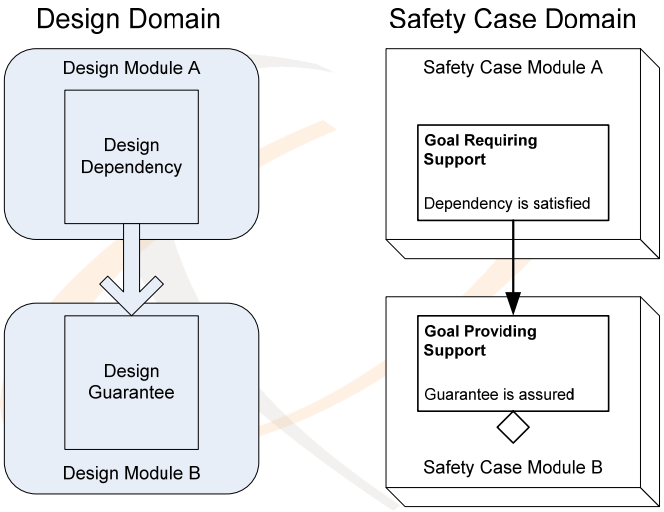


Cost of Re-Certification is Related to Change Size and Complexity

Modular Certification – Basic Principles

- Apply principles of software architecture and design to the **safety case** domain
  - **High Cohesion** – e.g safety concerns for a specific arch element
  - **Low Coupling** – Coupling unavoidable but minimise
  - **Information Hiding** – Key claims and properties visible
  - **Well-defined interfaces** – e.g. assumptions
- Align boundaries of safety case 'modules' with design boundaries to 'contain' change
  - **Aim:** A change to a design element should then only affect the corresponding safety case module, and not impact the entire safety argument

Modular Certification – Basic Principles



Representing Modular Safety Cases

- **Goal Structuring Notation (GSN)** - A structured method and notation for developing and presenting arguments
  - Developed by the University of York (15 years ago)
  - Usage to date GSN has largely been used for arguments that can be defined stand-alone and in toto
    - Some examples of GSN wallpaper exist
- To support modular safety case construction GSN was extended in 2001
  - QinetiQ funded work at York on Compositional SCs

Modular Safety Case 'Interfaces'

Externally visible properties:

1. Claims 'publicly' addressed by the module
2. Evidence presented within the module
3. Context (e.g. assumptions) defined within the module

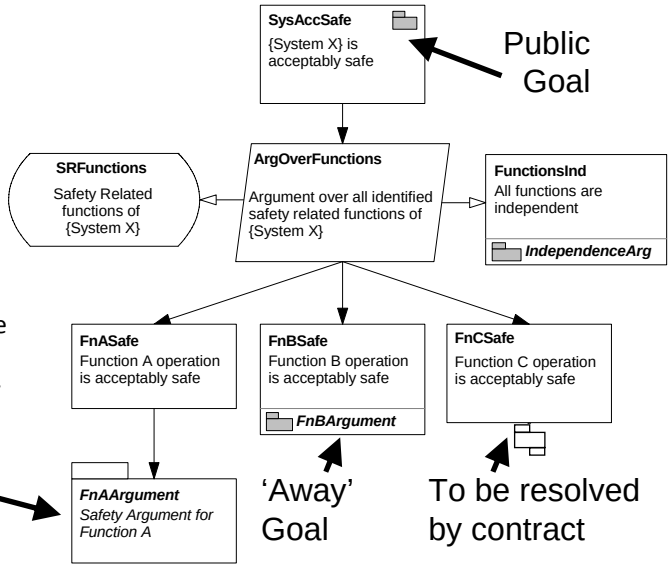
But also need to consider <u>interdependencies</u> …

4. Claims requiring support
5. Reliance on specific claims addressed elsewhere
6. Reliance on specific evidence presented elsewhere
7. Reliance on specific context defined elsewhere

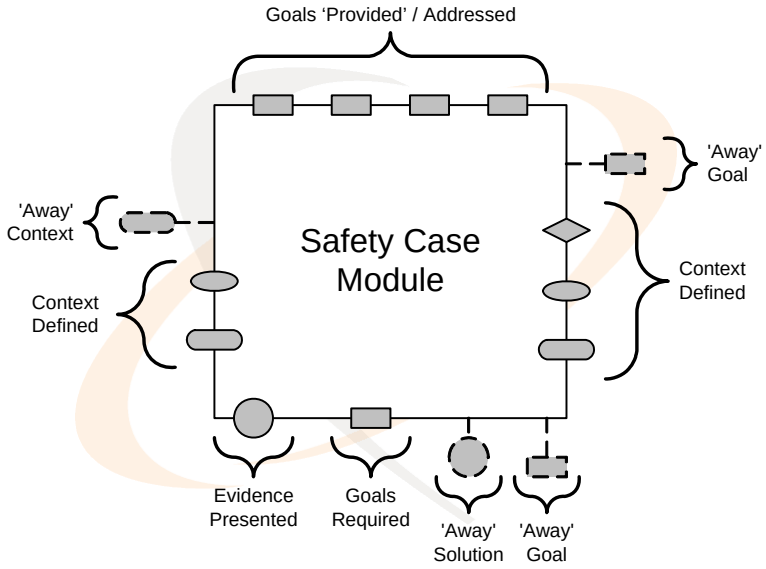
GSN Module Extensions

Extensions:

- Ability to mark a goal as *'public'*
- Ability to refer to goals defined in other modules
- Ability to refer to modules
- Ability to place one argument in the context of another

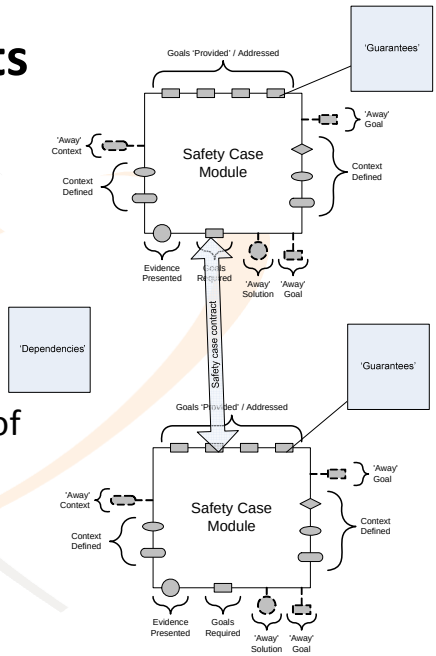


GSN Based Safety Case Interface



Safety Case Contracts

- Safety Case Modules can be composed if:
  - Goals Match (both ways)
  - **Context is compatible**
- Results can be recorded in a safety case contract
- Establishes a defined record of the inter safety case agreement
  - Supports management of change

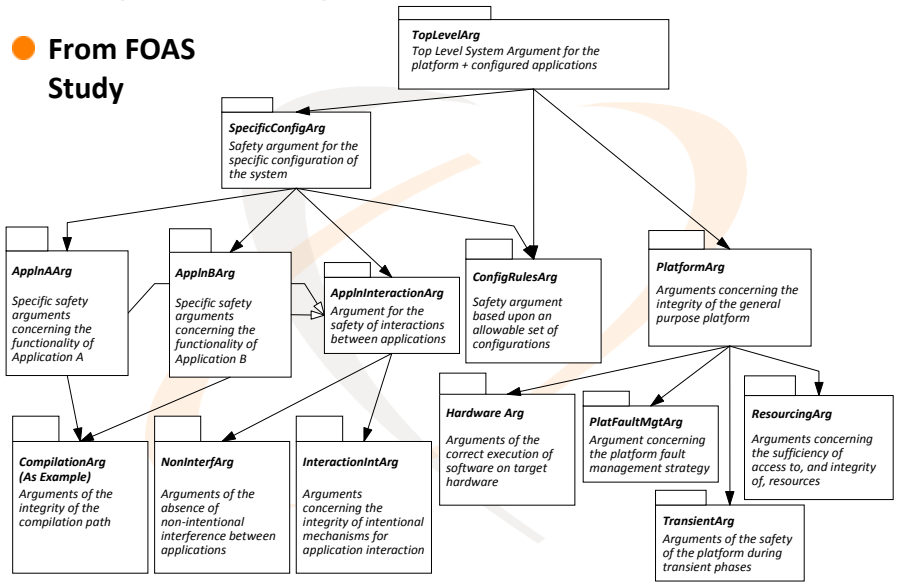


Initial Incremental Certification PV Study Methodology

- 3-Year Private Venture (PV) funded activity running from 2004 to 2006
- The **Agusta Westland Merlin Mk 1 helicopter** was used as the model for the study
- Modular Safety Case Structure developed
- A set of hypothetical but realistic change scenarios was devised and the effects of change studied
  - Analogy with S/W Architecture Scenario-based Assessment
- Software changes included flight safety related and non-flight safety related systems

Example: Safety Case Architecture for IMA

- **From FOAS Study**



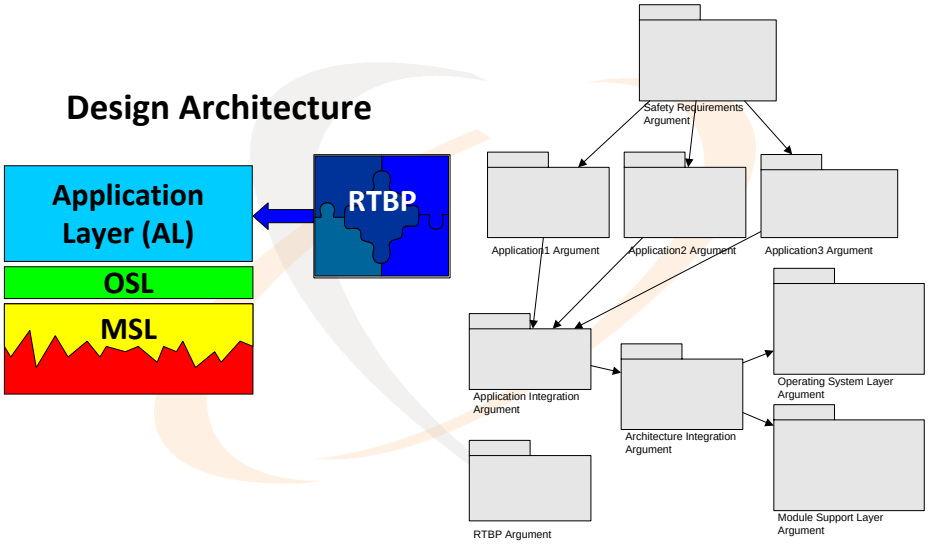
Change Scenarios

- Credible **change scenarios** include:
  - Hardware Vendor Change
  - Addition of a single application
  - Removal of a single application
  - Modification of existing application
  - Addition of extra processing nodes
  - Remove of processing nodes
  - Change of Databus
- Which safety case modules (arguments and evidence) would have to change in each case?
- Is the change local, non-local, architectural?

Hawk Parallel Study

- The results of the PV study convinced the MoD that additional work would be beneficial
- UK MoD funded a 'hot' research task
  - **Hawk AJT** aircraft chosen
  - Developing a modular safety case for a new system **in parallel** to monolithic project safey case
- New Mission Computer is IMA using an ASAAC-compliant three-layer stack
- Study aims:
  - Show that a modular safety case can be produced for a representatively sized project
  - Demonstrate that the proposed benefits can be achieved
  - **Multi-party** modular safety case development
  - Involve MOD appointed safety assessors (QinetiQ TES) to assess viability
- Hoped that Hawk project will transition to modular safety case

Safety Case 'Architecture'



Safety Case Architecture

- Software architecture defined in the following terms (Bass et al., 1998):

"The structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them"

- Safety case architecture can be defined in similar terms:

The high level organisation of the safety case into components of arguments and evidence, the externally visible properties of these components, and the interdependencies that exist between them

Safety Argument Overview

- Argument over software related elements within the System
  - (Functional) Blocks in the **Application** Layer
  - Modules in the **OSL (Operating System Layer)**
  - Modules in the **MSL (Module Support Layer)**
  - **Run-Time Blue Print**
    - The software modules that interpret the data are responsible for guaranteeing configuration of the system
- **Integration** Arguments regarding
  - **Architecture**
    - Integration of OSL and MSL
    - Provision and Performance of services
    - E.g. Scheduling arguments
  - **Application** Layer
    - Integration of the Software Applications
    - Integration of the Arguments for each Block
    - E.g. Absence of Unintentional Interactions
  - Overall **Integration**
    - Integration of the Applications with the Architecture

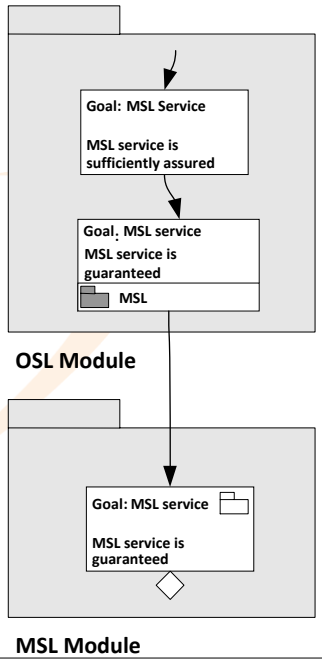
Dependencies / Guarantees & the Safety Argument 1

- Overall Argument Strategy
  - Top Level Claims that each **Safety Requirement** is adequately assured
  - Supported by claims that guaranteed behaviour is adequately assured
- Use of **Dependency Guarantee Relationships (DGRs)**
  - Identifies the guaranteed behaviour for each software related element
  - For each 'Guarantee' the related 'Dependencies' are identified
  - The related 'Dependencies' are the behaviour needed from other elements to meet the 'Guarantee'
- **Integration of arguments**
  - Integration of the arguments over the software related elements is achieved by linking the arguments over the 'Dependencies' and the 'Guarantees' between the elements
- **Mechanism for the argument**
  - Argument creates a 'Daisy Chain' that begins with a Safety Requirement in one element that is supported by a Guarantee in another element, whose associated Dependencies are supported in turn by Guarantees in another element …

Linking Safety Case Modules

- When developing the argument for a module, it may be necessary to make a claim to support the argument which is outside the scope of that module
  - e.g. The OSL argument may need to make a claim about the MSL behaviour to support it's safety argument
  - "I know I need the argument supporting the claim to be made, but I'm not going to make it here"
- Away goal and module references can be used to reference goals in other argument modules
  - But 'hard-wired' – i.e. Module A specifies up-front the argument in Module B which will be used to support the claim
- This means that if Module B changes, Module A will also need to change to point to a new goal in Module B

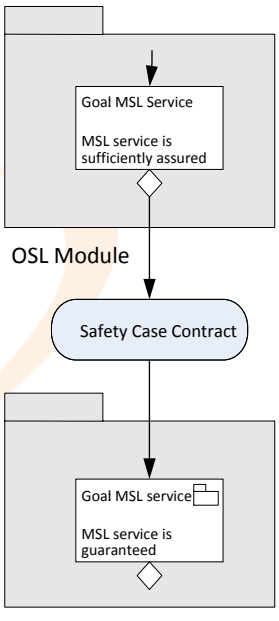
Linking Safety Case Modules



Linking Safety Case Modules with Contracts

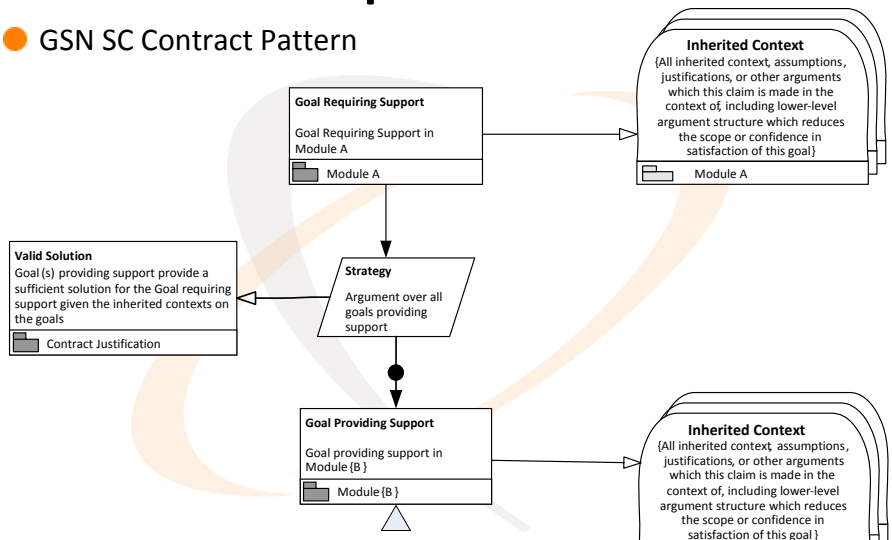
- An alternative approach to link safety case modules together is to form a **contract** between the two modules
  - Goal *to be supported* links to the contract, rather than directly to the supporting claim
  - Contract then identifies the goal *providing support* in the other module
- Provides a 'buffer' between the goals in the two modules
- If the supporting module changes, only the contract needs to be updated (to identify the new supporting argument) and not the module requiring support
  - To not 'open' the safety case at all is valuable
- In this way the module is 'isolated' from the changes in the supporting module

Linking Safety Case Modules with Contracts



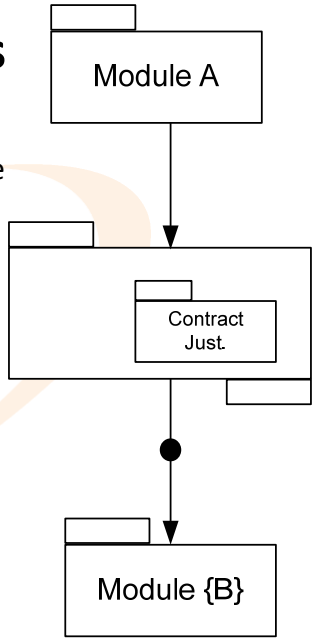
IAWG Proposed Solutions

- GSN SC Contract Pattern



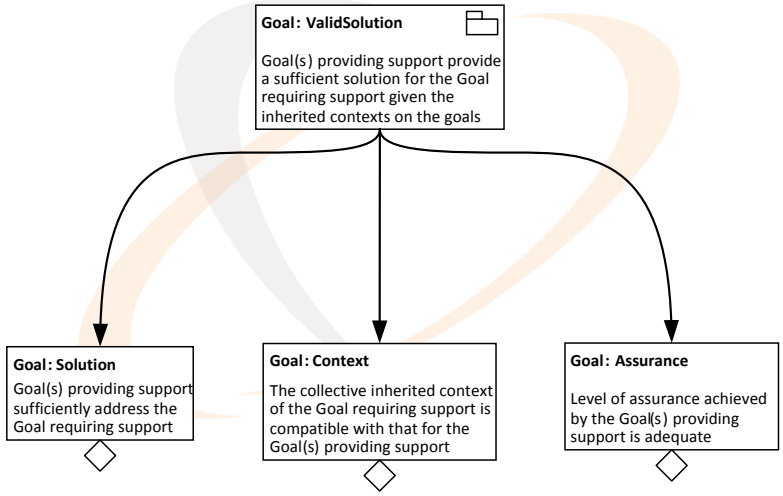
IAWG Proposed Solutions

- GSN Safety Case Contract Architecture



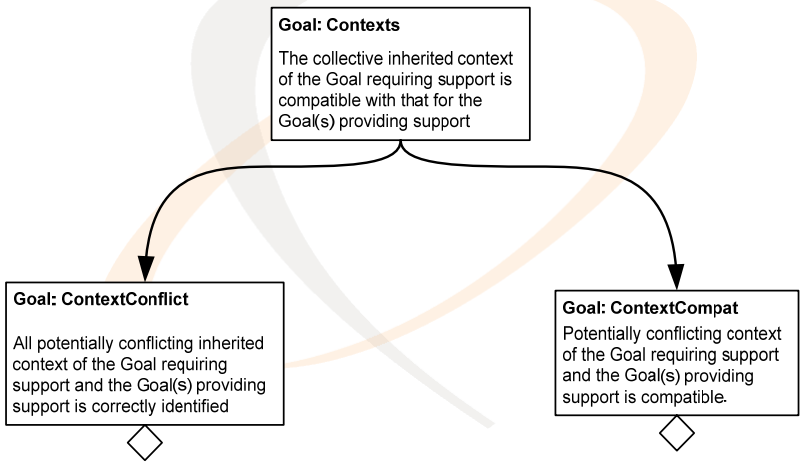
IAWG Proposed Solutions

- GSN Safety Case Contract Justification Argument Module



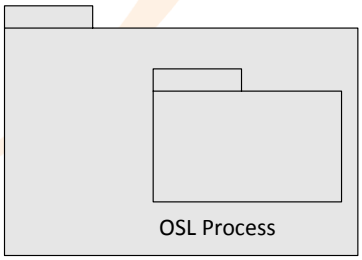
IAWG Proposed Solutions

- GSN SC Contract Context Compatibility



Containment

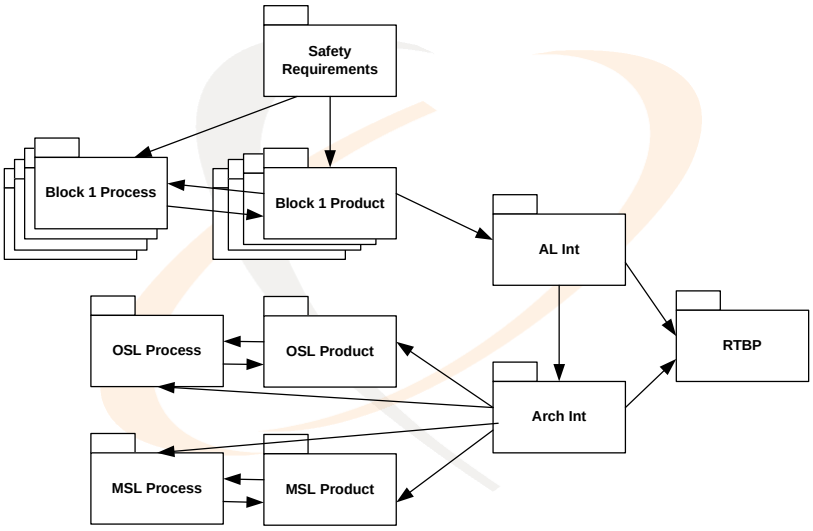
- It is not always necessary for each safety case module to have visibility of all other safety case modules
- Where elements of one safety case module are of limited applicability, their scope can be limited through containment
- Particularly useful for Process arguments which relate only to a single product argument module



OSL

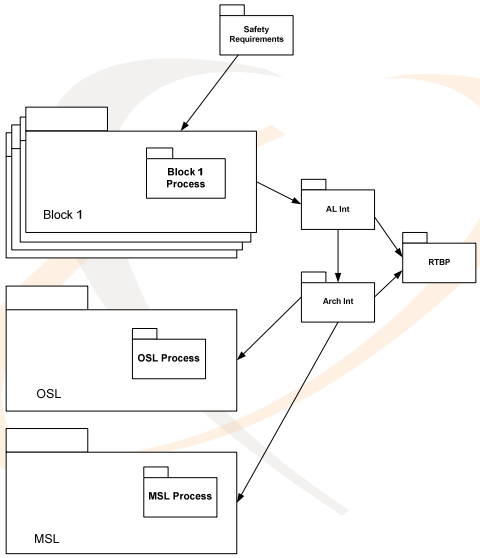
Containment Simplifying the SCA

Without Containment



Containment Simplifying the SCA

With Containment



Current IAWG Mod Cert Status

- 'Final' Safety Case Delivered
  - Customer-driven change to programme meant that system was not completely developed within research programme timescales, so:
    - Argument is thought to be complete for most modules, using template approach
    - Evidence is incomplete
- IAWG Modular Software Safety Case Process delivered to MoD
  - Part A – Process Definition
  - Part B – Guidance
  - Expected to be provided via the MoD Acquisition Management System website
- Next Study about to begin: Agusta Westland Future Lynx Avionics

Feedback 1

The IAWG's research into Modular Certification has a high profile within the MoD's research and equipment acquisition communities. The "hot research", parallel development approach with the Hawk AJT Mission Computer has enabled this research to have a very real exploitation focus. There are still some minor refinements required and areas left to be de-risked to achieve the greatest benefits and widest exploitation. The IAWG are keen to address these, following this, the MoD will seek to achieve much wider adoption of the Modular Certification approach.
Chris Nicholas, dstl, Research Customer Representative

Feedback 2

Modular Certification is a non-trivial task and IAWG have made great steps towards the realisation of this critical technique which will be an enabler for many future programmes!
Brian Jepson, BAE SYSTEMS, Internal ISA

[Assessment report] identifies key outstanding issues and concludes that if these issues are adequately addressed, the modular approach developed by the IAWG should be satisfactory for the certification of a modular software safety case and hence suitable for adoption on Hawk and other types.
Philip Vale, QinetiQ, External ISA

Further Work

- Dependency Guarantee Relationship (DGR) notation
- Auto-Generation/Validation of DGRs
- Assurance (Levels) in the modular safety case
- Modularity of Evidence
- Mutual optimisation between design architectures and safety case architectures
- Determining whether a system is receptive to modular certification
- Extending the modular approach to other dependability properties, e.g. security – SafSec Coherence Study
- Safety Considerations for RTBP

Applications of Modular GSN

- IAWG
- LM C130J Safety Case Books
- Smiths Industries Common Core System for Boeing 787
- MIL-STD 1760 Based Safety Cases – Jacobs Australia
- Process and Product Cases
- Multi-Attribute Dependability Cases
  - HIRTS DARP & QinetiQ VTID

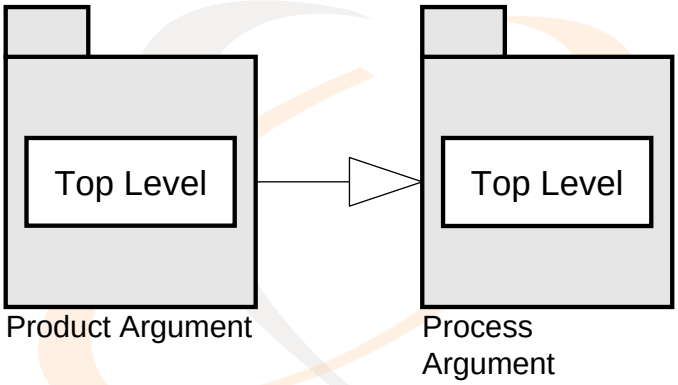
Tool Support for Modular GSN:

- Supported in University of York's freeware GSN addon for Microsoft Visio
- ERA Technology's GSN Casemaker
- High Integrity Solutions ISIS tool
- Support being added into Adelard's ASCE tool

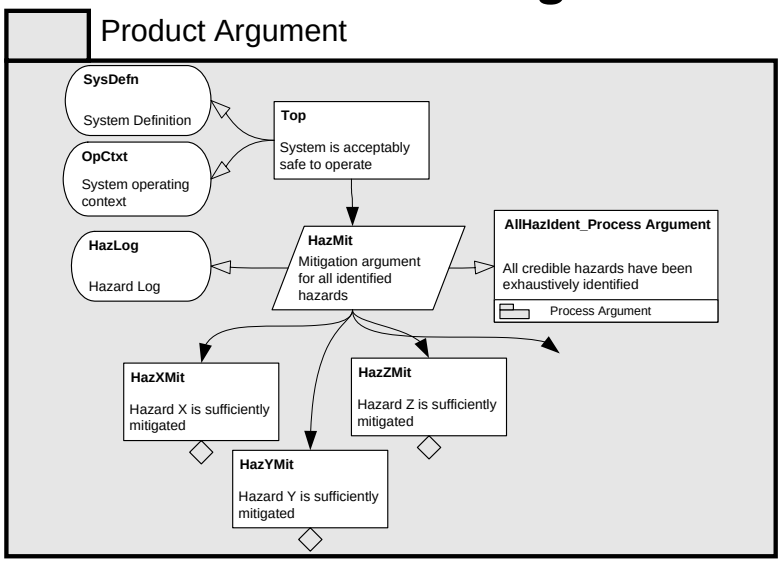
Summary

- Difficulties experience in managing the scale and complexity of current safety cases
- Many of the ideas from software architecture can be 'stolen'
- GSN Extended to support Modular Safety Case Construction in 2001
- IAWG has performed 2 studies to date, 1 more about to start
  - Positive feedback
- Refined & Extended Approach
  - IAWG Modular Safety Case Process
- Modular GSN useful in a wide variety of applications

Process and Product Argument 1



Process and Product Argument 2



Process and Product Argument 3