

---

# Assurance Networks for Complex Systems

Colin Brain  
SE Validation  
20 May 2009



# Agenda

---

- Introduction
- Motivations for system assurance
- Architectural views
  - Decision view
  - Goal view
  - Frame view
  - Evaluation view
- Integration
- Conclusions & Discussion



# Introduction

---

- Work in progress
- Evolution of work going back at least 10 years
- .. particularly in Anglo-Swedish joint research forum
- .. and European REVVA consortium
- Aspects routinely applied in SE Validation's safety, system and simulation validation business
- Other parts based on experience in INCOSE, including our workshops and tutorials
- .. but much here brought together for the first time

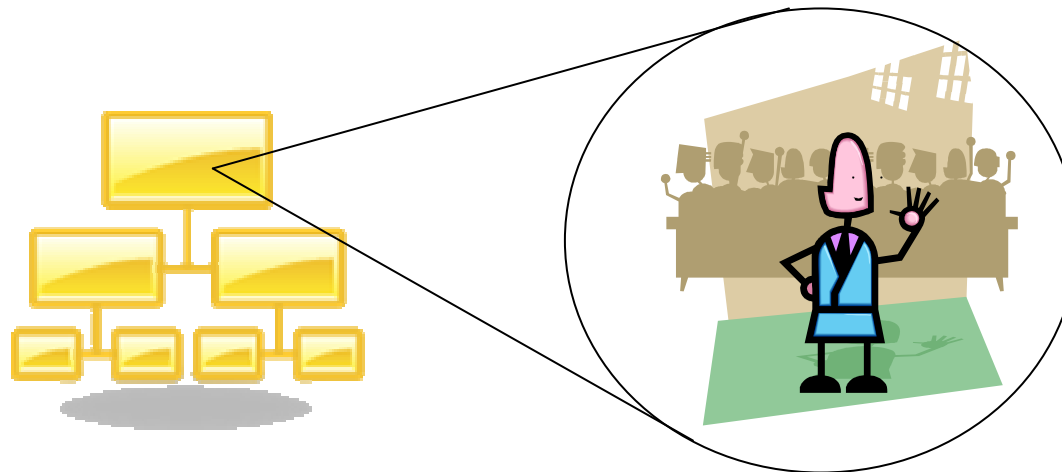


# Introduction: Complexity vs Complication

---

- Complicated system:
  - made up of a large number of interconnected components
  - created or operated under a single logical enterprise
  - components have the same or related life cycles

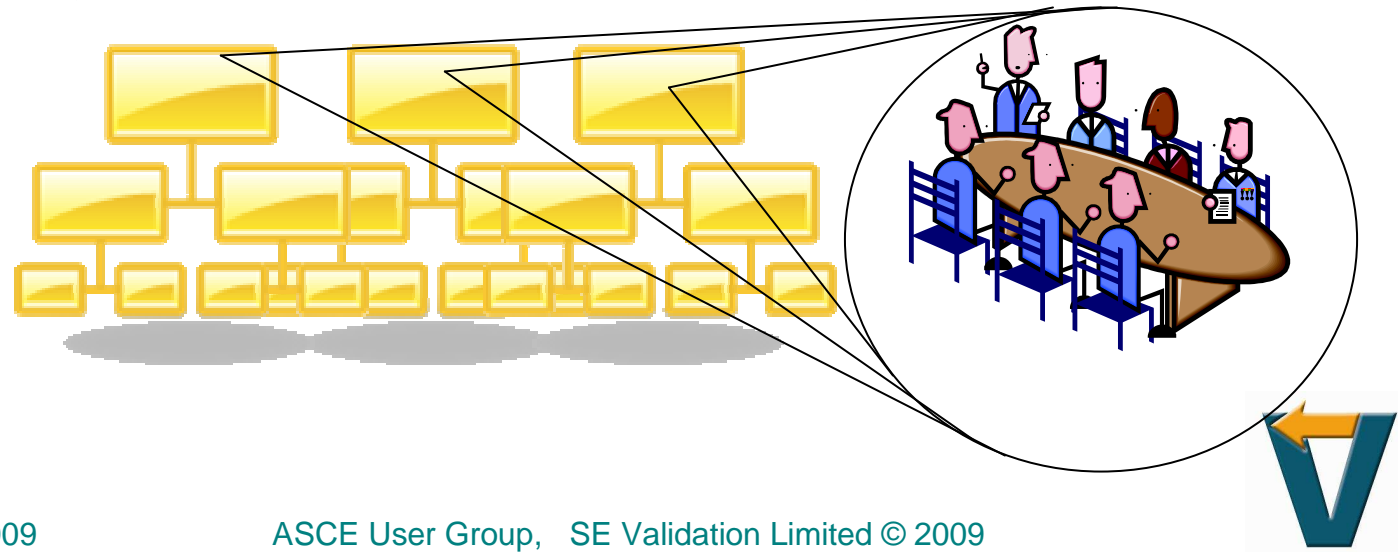
*The space shuttle is a complicated system*



# Introduction: Complexity

---

- Complex system (system-of-systems):
  - component systems have different, uncorrelated life cycles
  - development and / or operation takes place under different enterprises with different cultures
  - no single logical design authority or future vision
  - many interconnection interfaces never systematically designed and are not subject to any agreed standards



# Motivations for system assurance

---

- Complicated and complex systems – and related assurance products
- Risk aversion – in society, customers and suppliers
- Pace of technological change
- Complexity & distribution of supply chain
- Increasingly required by law, regulations, contracts and standards
- Emergence of goal based standards
  - encourage innovation but more difficult to produce
  - more difficult to comprehend and audit



# Motivation: Standards and Guidance

---

- Draft update to ISO 15026
- US NDIA handbook on System Assurance
- Simulation Interoperability Standards Organisation (SISO) Generic Methodology for Verification and Validation of Models and Simulations (GMVV)
- Object Management Group (OMG) standardisation efforts for evidence repositories and argumentation frameworks
- MODAF, etc
- **Customers!**



- ISO 15026 provides life cycle requirements for:
  - development, operation, maintenance, and disposal of systems and software products
  - that are critically required to exhibit and be shown to possess properties related to safety, security, dependability, or other characteristics
- Defines an assurance case as the central artifact for:
  - planning, monitoring, achieving and showing the achievement and sustainment of the properties
  - related support of other decision making



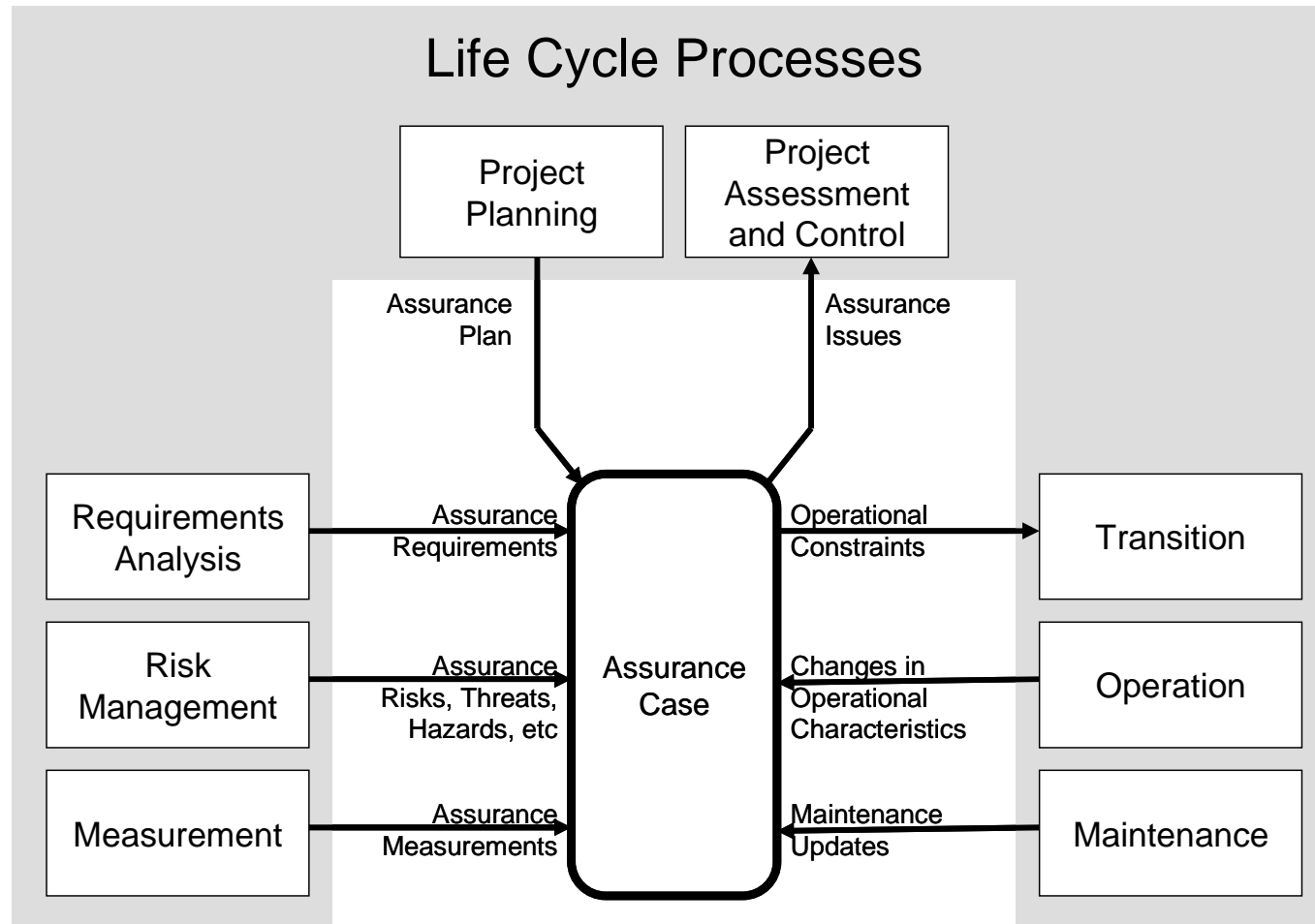
# ISO 15026 General Requirements

---

- **Project** shall establish & maintain an assurance case to ensure that:
  - Goals for designated critical properties are formulated
    - Product assurance objectives, &c. selected
    - Requirements for achievement of these defined.
      - Measures & satisfaction criteria selected & traced
  - Approaches planned, designed, implemented, demonstrated and documented
  - Extent of achievement continuously monitored, documented & communicated to stakeholders & managers
  - Assurance case developed & maintained **as element of system**
  - Requirements of the approval authority are satisfied



# ISO 15026 - Role of assurance cases



# System Assurance Networks

---

- Number of architectural frameworks available, incl:
  - MODAF / DODAF / NAF
  - Service oriented architectures
  - Business process modelling frameworks
- .. but for complexity we also need to support:
  - collaboration
  - organisational and system hierarchy
  - functional allocation (logical to physical)
  - experimental and conceptual-modelling frames to obtain behaviour data
  - asynchronous change
  - varying ontologies and taxonomies



# Architectural Views - Background

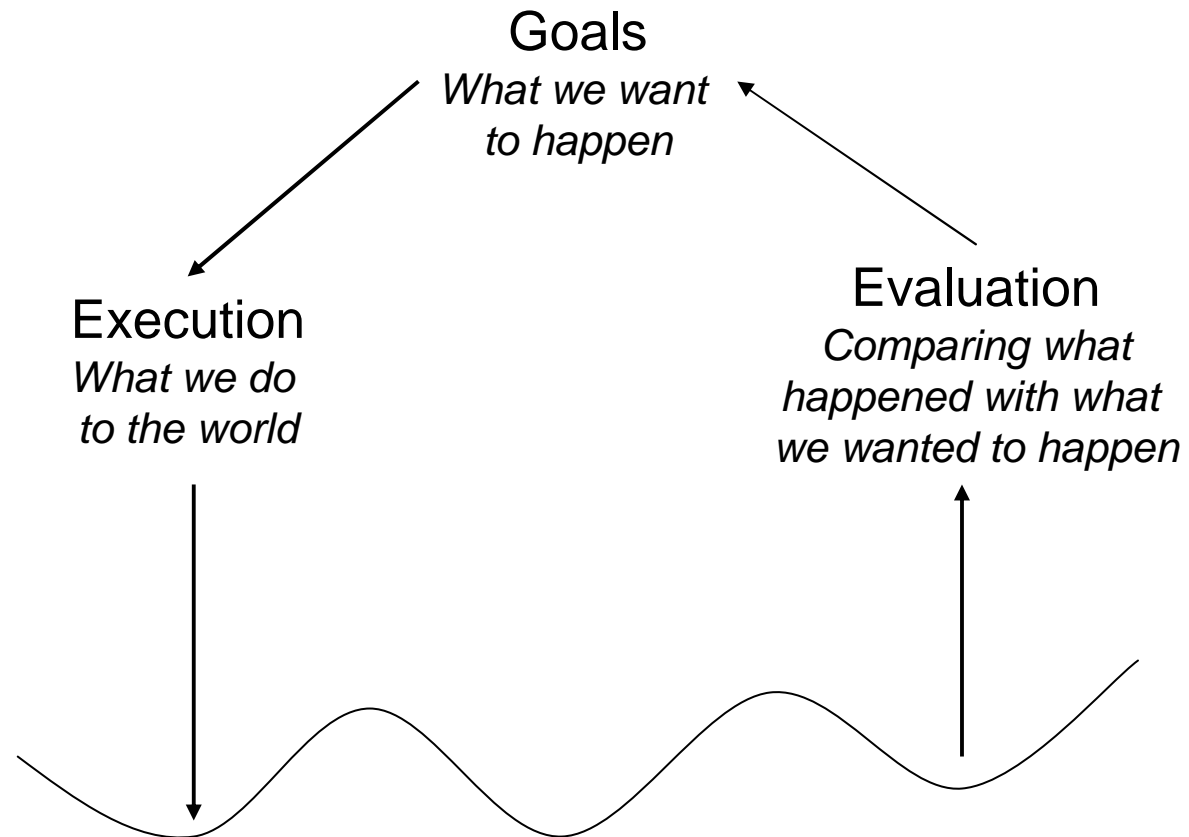
---

- Problem Solving – Norman model
- Need for explicit structural models
- Decisions
- Problem frames, verification & validation



# Norman's model

---



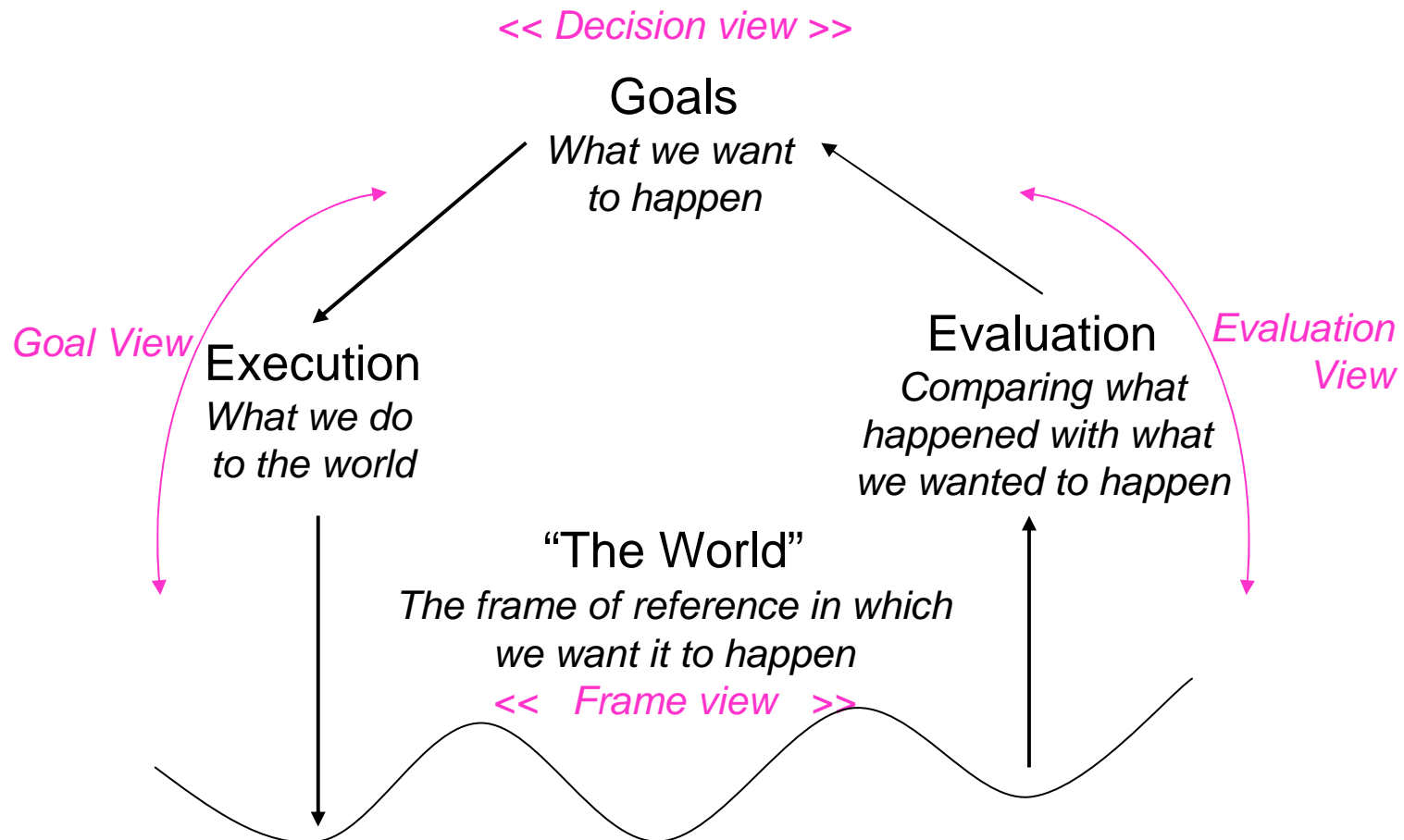
# Norman's model of problem solving

---

- Execution (Planning):
  - establishing the goal
  - forming the intention
  - specifying the action sequence
  - executing the action
- Evaluation:
  - perceiving the system state
  - interpreting the system state
  - evaluating the system state (with respect to the goals and intentions)



# Norman model - views



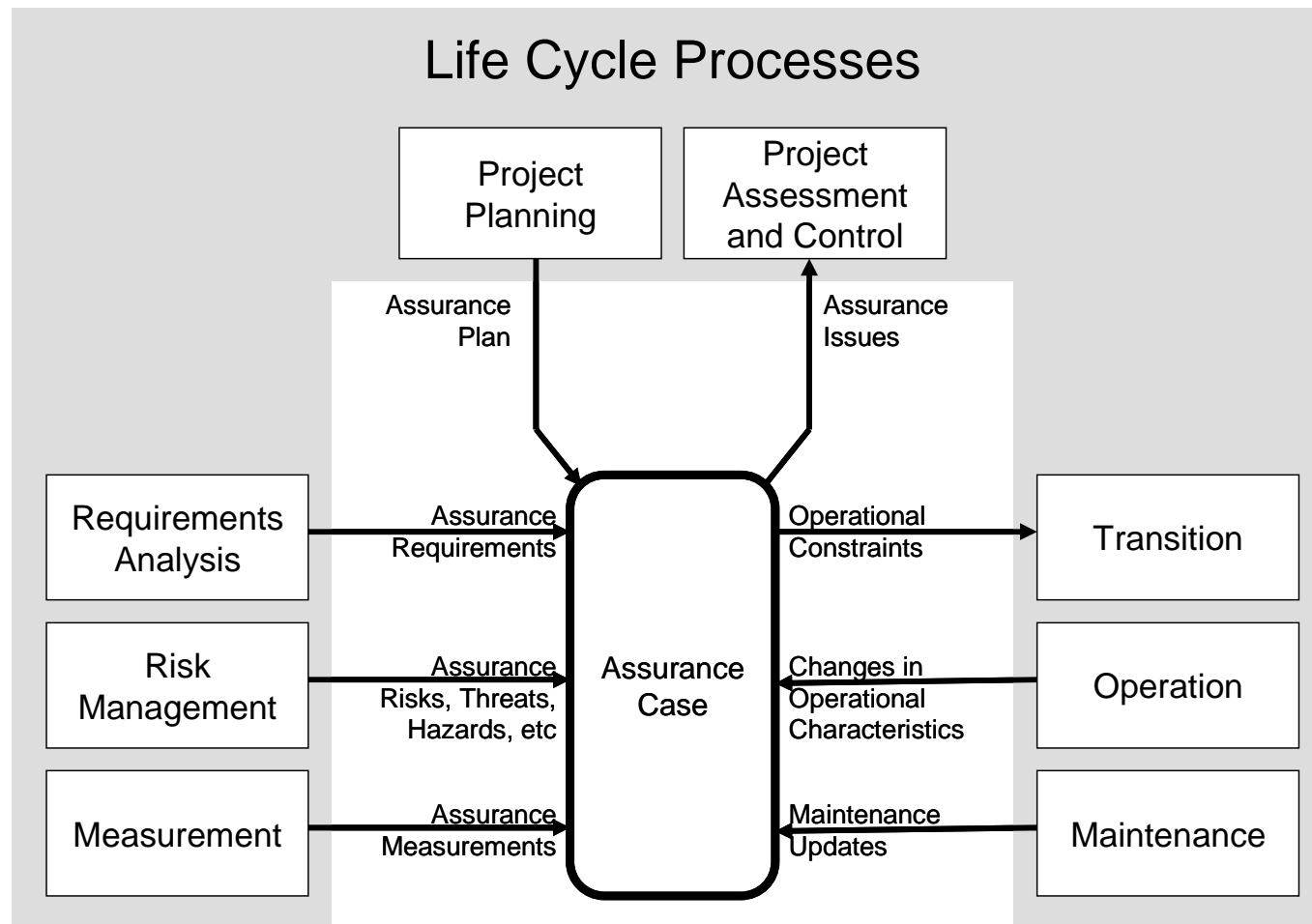
# Decisions



Source: Systems Process Inc



# ISO 15026 - Recap



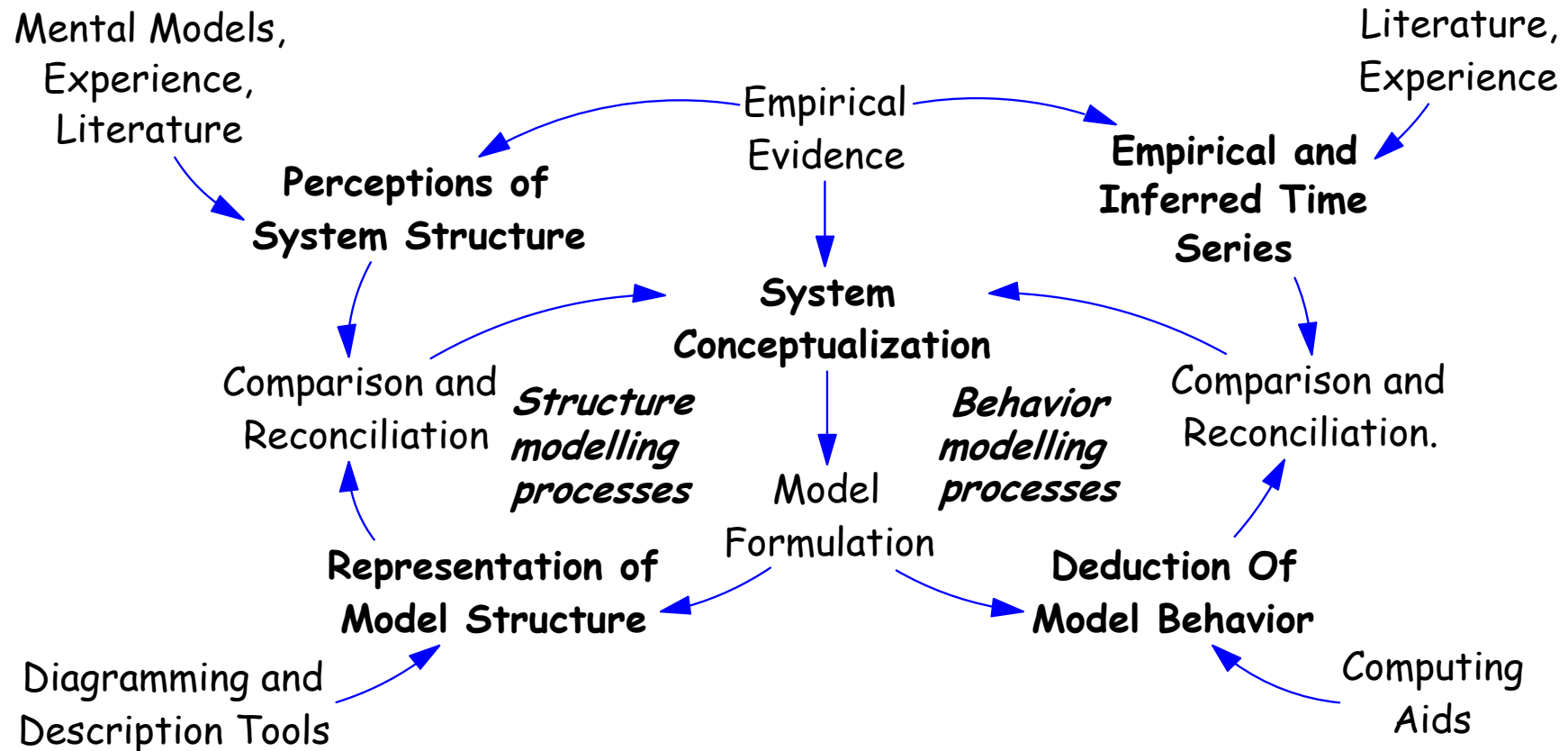
# Decisions and views



Source: Systems Process Inc



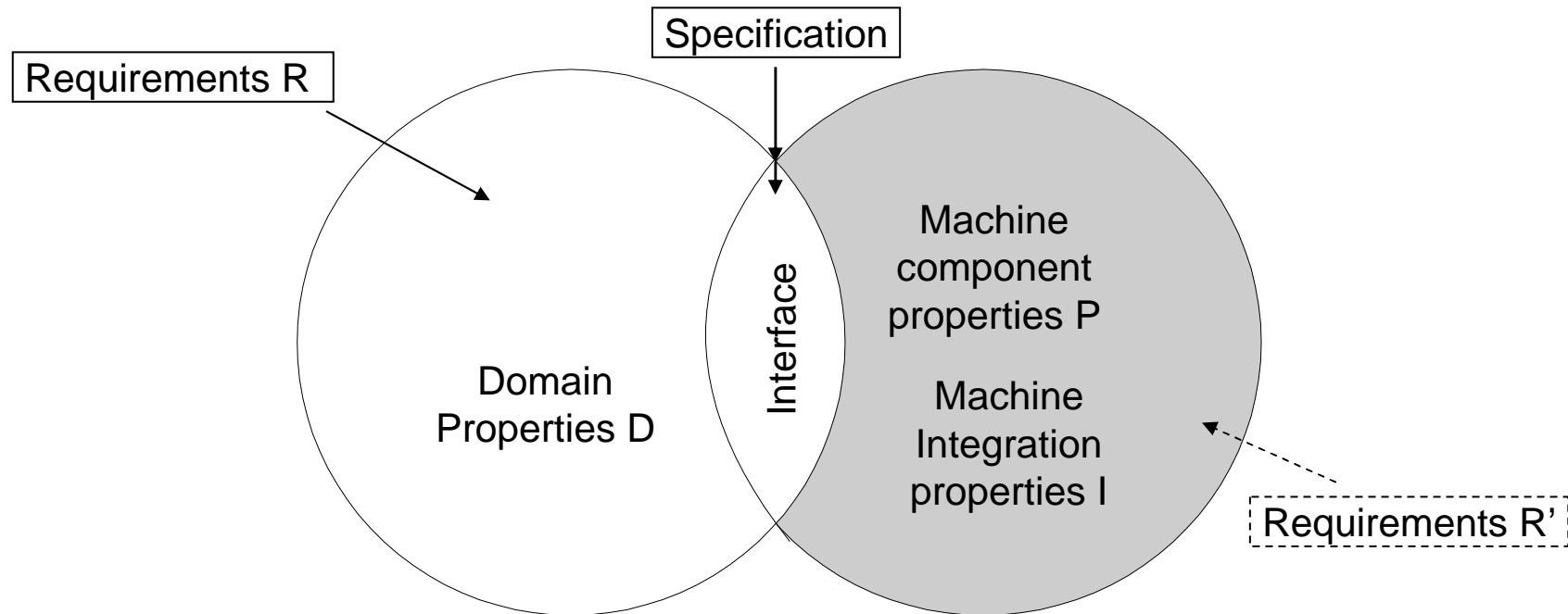
# Frame views - two kinds of modelling processes



Source: George P. Richardson, University of Albany



# The world and the machine



*Satisfaction Arguments:*

**WORLD**

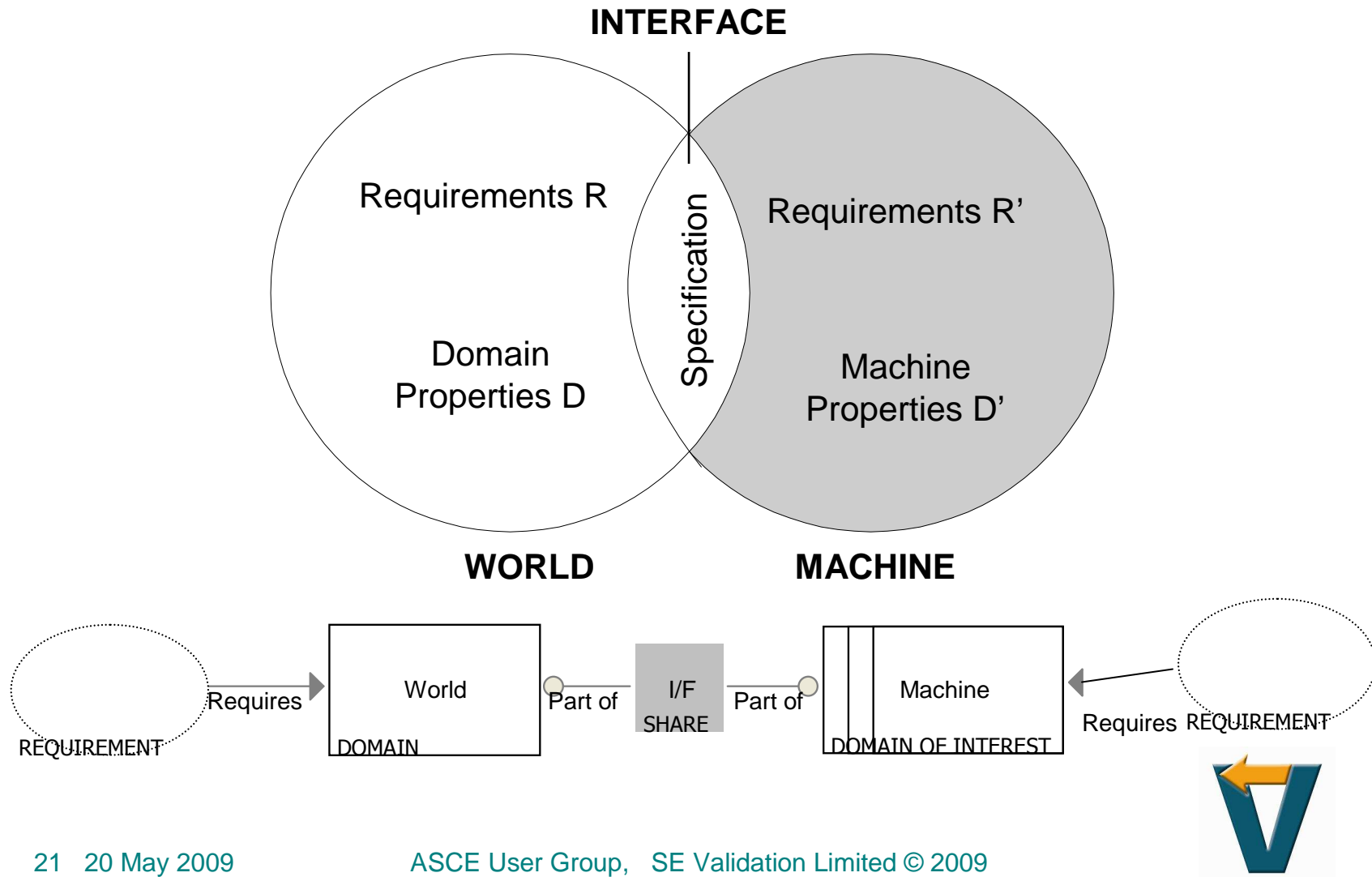
**MACHINE**

Validation:  $D, S \vDash R$  Specification AND Domain Properties IMPLY Requirements

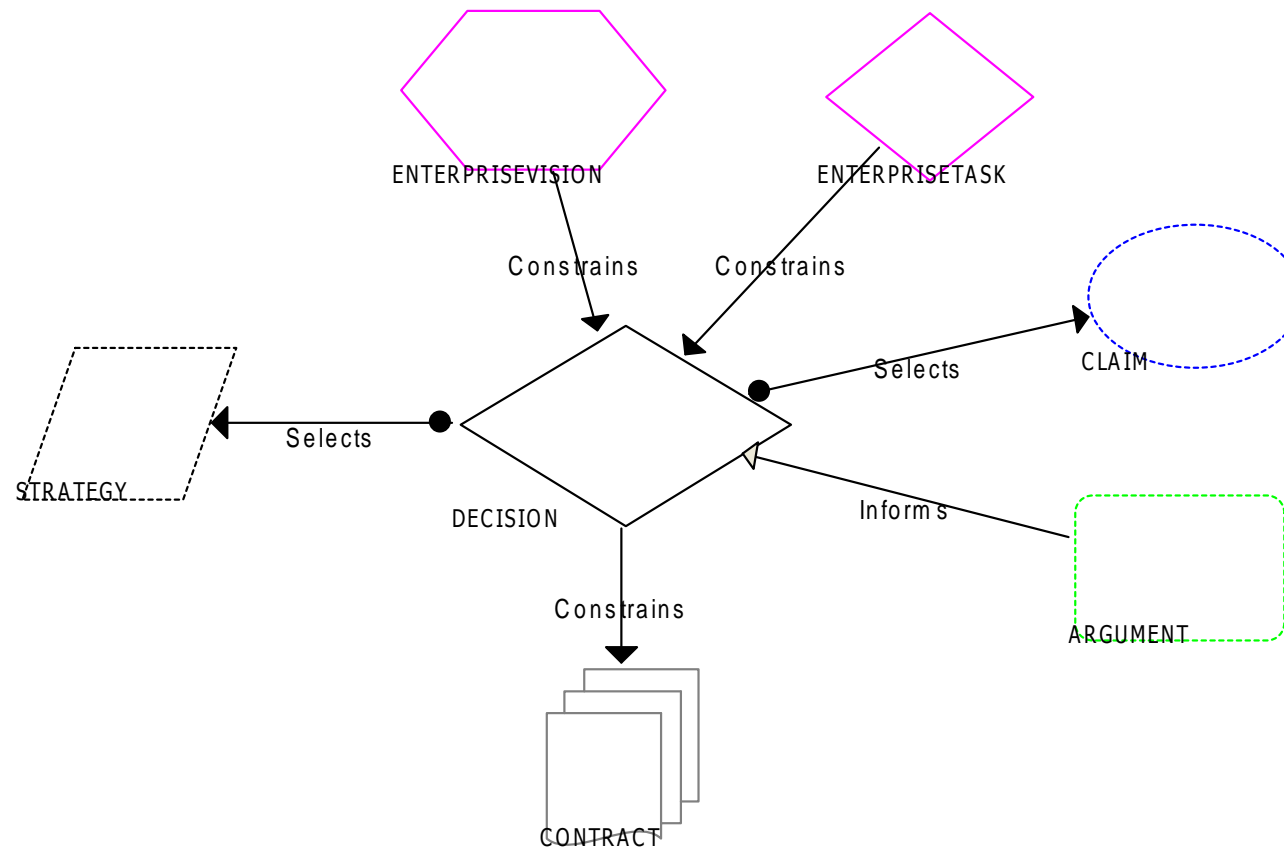
Verification:  $P, I \vDash S$  Component AND Integration Properties IMPLY Specification



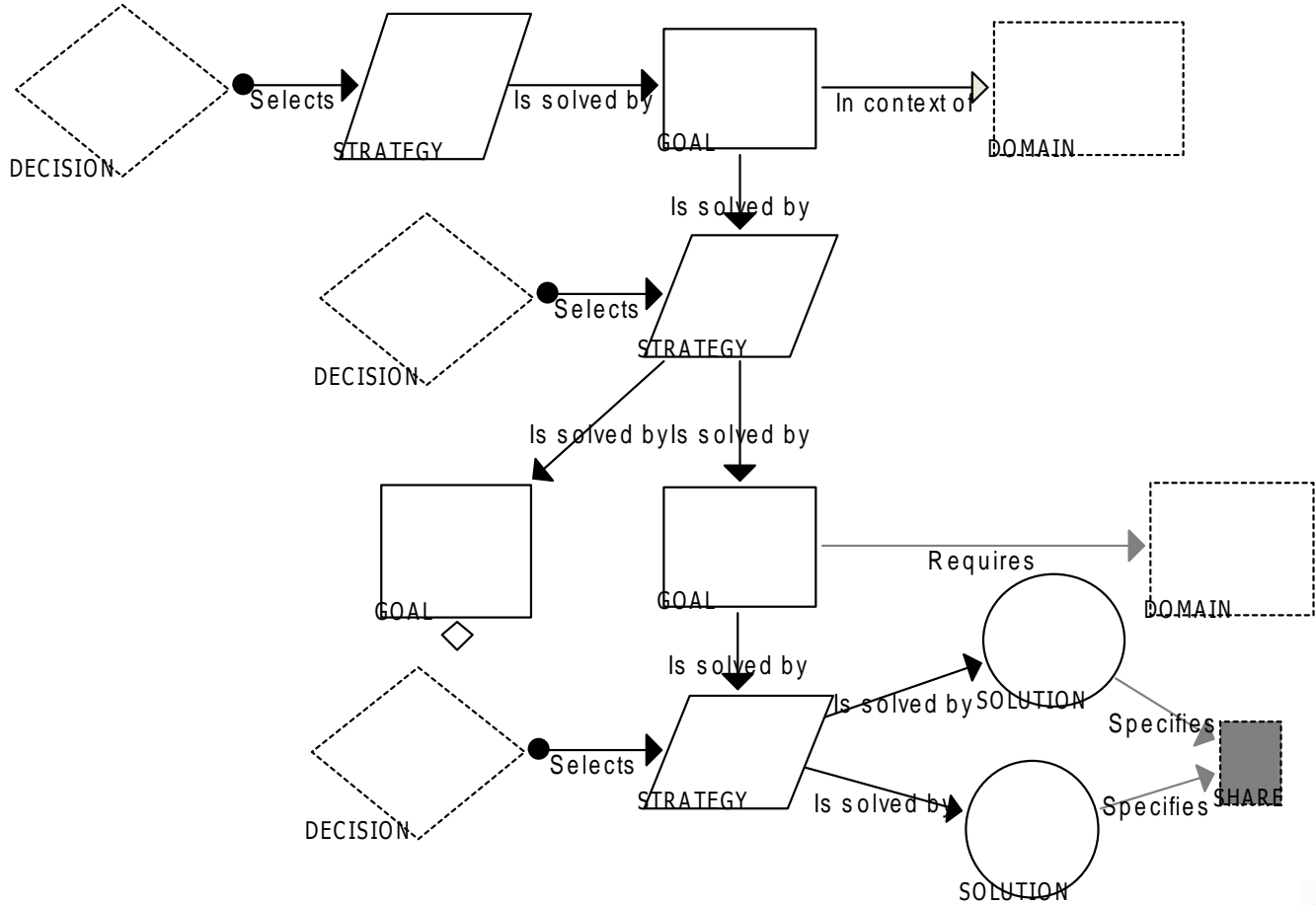
# ASCE Frame view schema



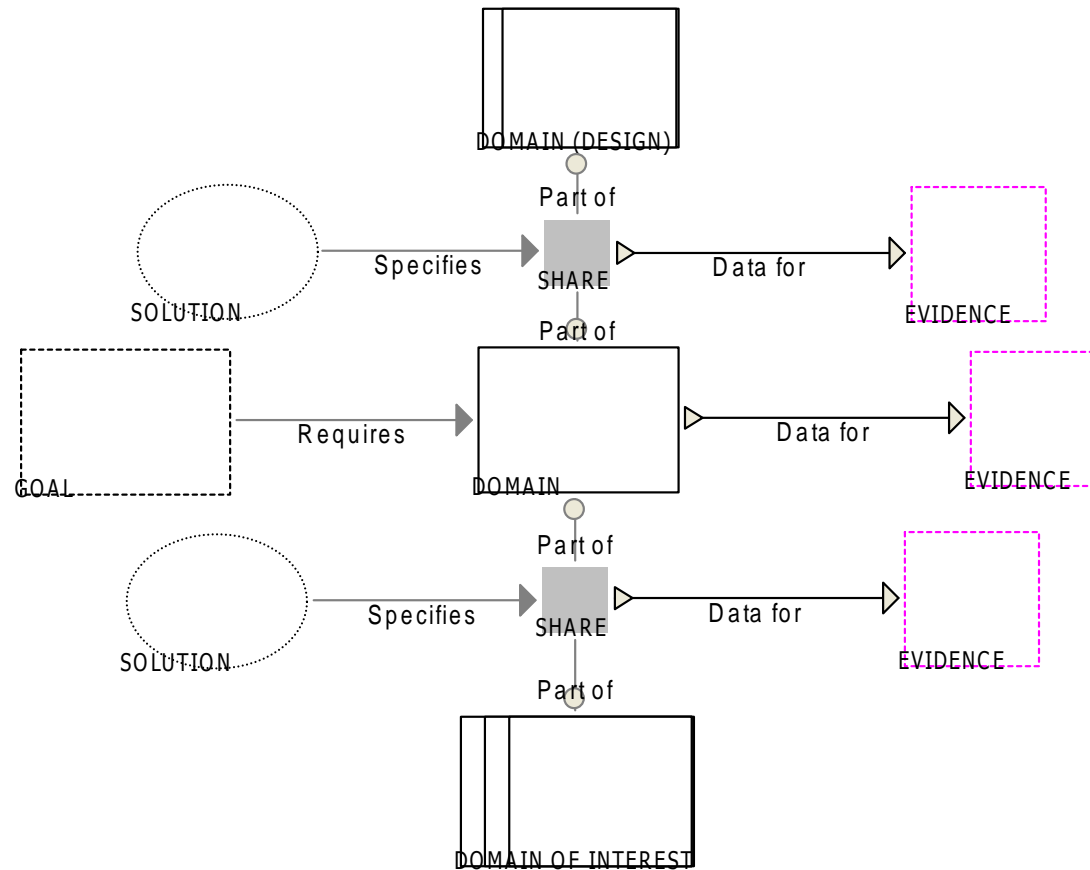
# Decision View



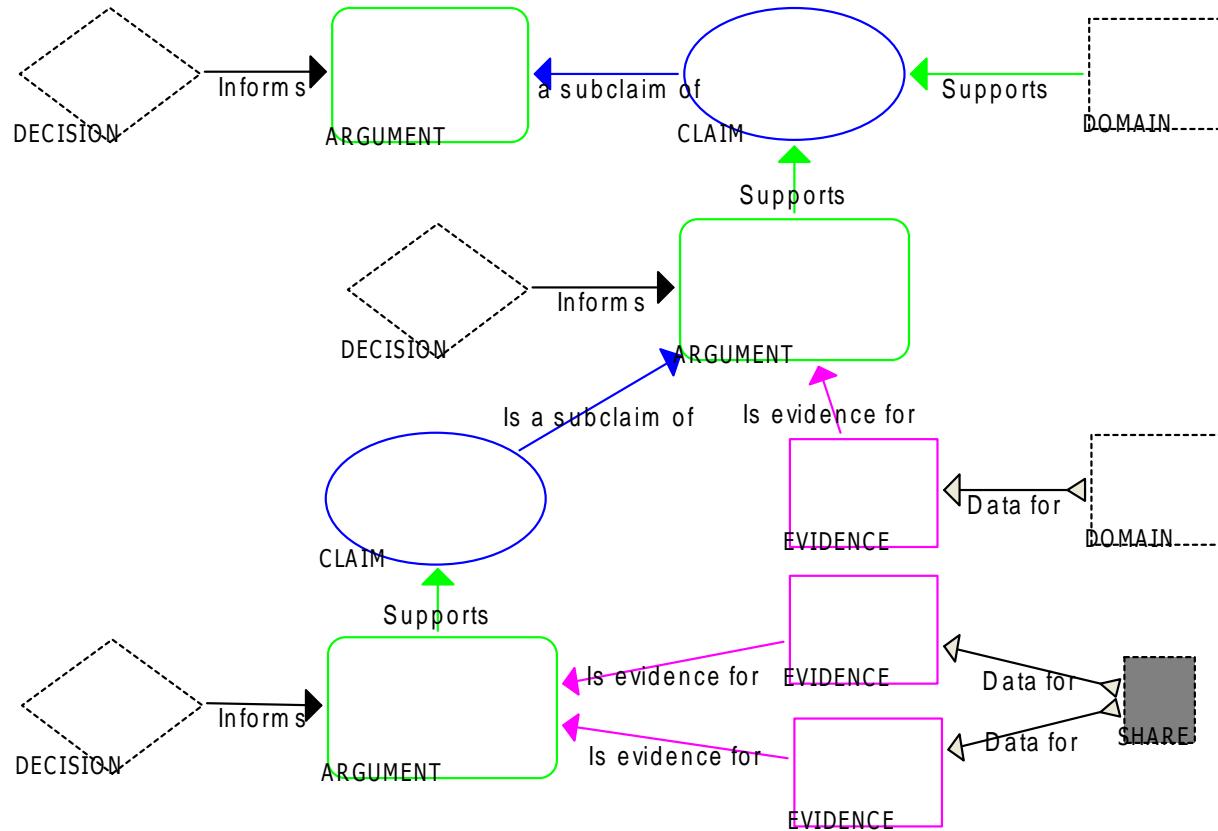
# Goal view



# Frame view



# Evaluation view



# Integration

---

- Very important for collaboration on complex systems
- Configuration & version control for asynchronous updates
- Need to interact with multiple decision cycles
- ASCE features:
  - Source verification
  - Maintenance of multiple views
  - Maintain the core electronic ‘case’ and the publication templates, not necessary each document
  - Simple open XML file structure, easy to configuration control



## Conclusion

---

- Four view framework appears capable of supporting full system assurance for complex systems
- Evolutionary, not revolutionary, approach
- Each view builds on proven approaches
- Appears to be capable of solving many of the system safety assurance issues that we have experienced
- Amenable to XML-based tool approaches using ASCE as one of the major tools
- Enhanced schema have been produced and prototyping started



## Further work

---

- Building prototype application
- View and node templates?
- Navigation between and within views
- Support for multi-view patterns and frames/contents
- XML import and export using style sheets driven from ontologies and taxonomies
- Computer supported querying, aggregation and verification across multiple views:
  - Answer-set programming?
  - Error / failure budgets and confidence limits
- Links to behavioural modelling and testing



---

Thanks for Listening

Colin Brain

SE Validation

[cjb@sevalidation.com](mailto:cjb@sevalidation.com)

[www.sevalidation.com](http://www.sevalidation.com)

01722 502674

