EXTENDING MBSE FOR DECISION PATTERNS AND TRACEABILITY

John Fitch, ESEP

Project Performance International

jfitch@ppi-int.com

Presented this topic at MBSE-CON-2024 Conference in early May

Requested to share with the INCOSE Orlando chapter

Added supplemental content from three prior presentations:

- Leveraging Decision Patterns tutorial at IS2023
- Case Study - Extending LML to Enable Decision Patterns and Traceability presentation at IS2023
- Leveraging Decision Patterns to Tame Complexity and Accelerate Solution Delivery – September 2022 INCOSE GfSE Webinar

In order to:

- Provide better background on the newer concepts
- Include a second case study example of extending a language (LML)

PPI SyEN monthly Newsjournal articles:

- Introduction to Decision Patterns: Edition #107 (December 2021)
- Decision Patterns – So What?: Edition #111 (April 2022)
- Reverse Engineering Stakeholder Decisions from Their Requirements: Edition #113 (June 2022)
- Extending the Lifecycle Modeling Language (LML) to Enable Decision Patterns and Traceability: Edition #125 (June 2023)
- Rethinking Requirements Derivation – Part 1: Edition #129 (October 2023)
- Rethinking Requirements Derivation – Part 2: Edition #130 (November 2023)

- Decisions are the human thinking process that transforms a problem definition (requirements/goals) into a solution description (design)
- But design decisions are poorly captured into today's system modeling languages and tools
- This failure has significant impact on the value delivered to stakeholders
- The fixes are fairly simple and well (but not widely) understood – a demonstration example exists that highlights language and tool gaps

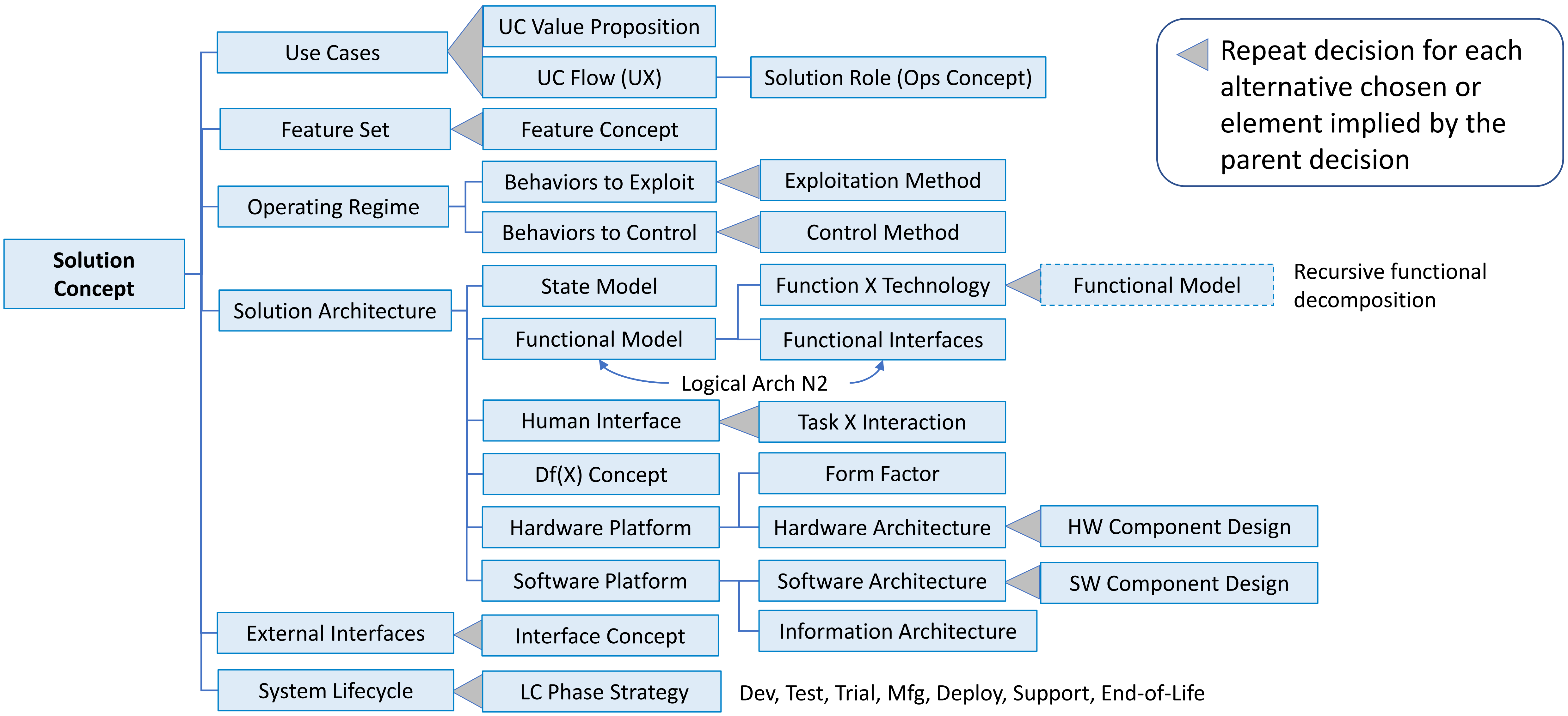
- LML and Innoslate show a lot of promise as a decision capture platform
- But there are many details to work out to optimize the results

- Lessons learned from LML can easily be extended to SysML 2.0 and its supporting tools or other MBSE platforms

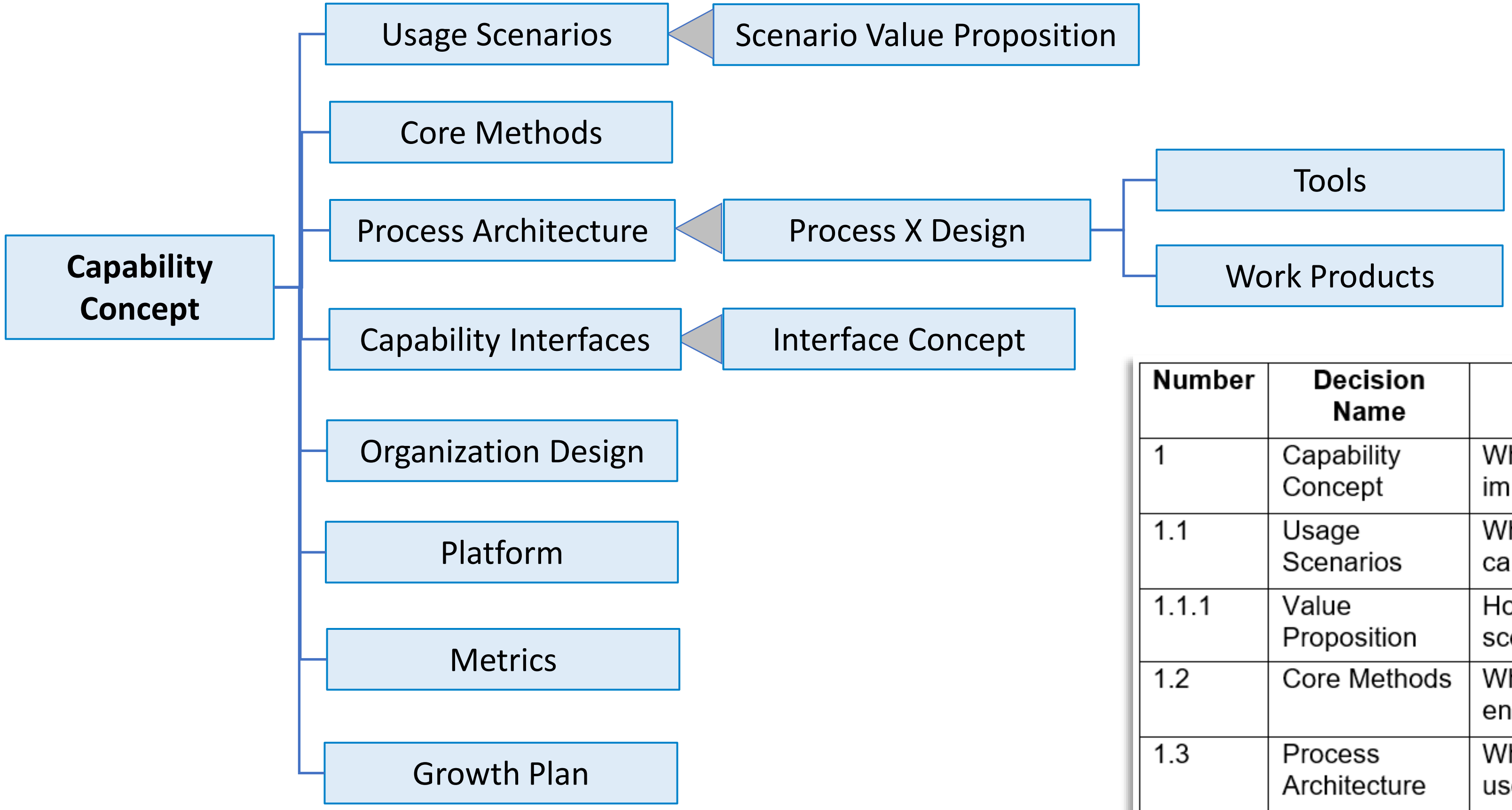
- Loss of the thinking that provides the rationale behind the design
- Failure to visualize, communicate & integrate the factors needed for high-confidence design decisions
- Inefficiencies in the face of change
- Inability to perform multi-decision optimization/tradeoffs
- Loss of the derivation traceability thread that is the source of all requirements
- Inability to leverage past decisions as patterns to accelerate/improve thinking

- Definition: A decision is a fundamental question or issue that demands an answer or solution - not the alternative chosen
- Design = decision making
- A system design is the result of numerous decisions (that must be consistent)
- These decisions follow patterns that can be used to jump-start any project
- An explicit decision model enables proactive, efficient & effective design; ad hoc decision-making just the opposite
- Decisions create requirements, i.e., all requirements are derived requirements
- Decision traceability demands capture of decision rationale and consequences (a rich data structure)

Product / System Design Decision Pattern



Process Capability Design Decision Pattern



Simplified pattern for business, management or technical processes, such as:

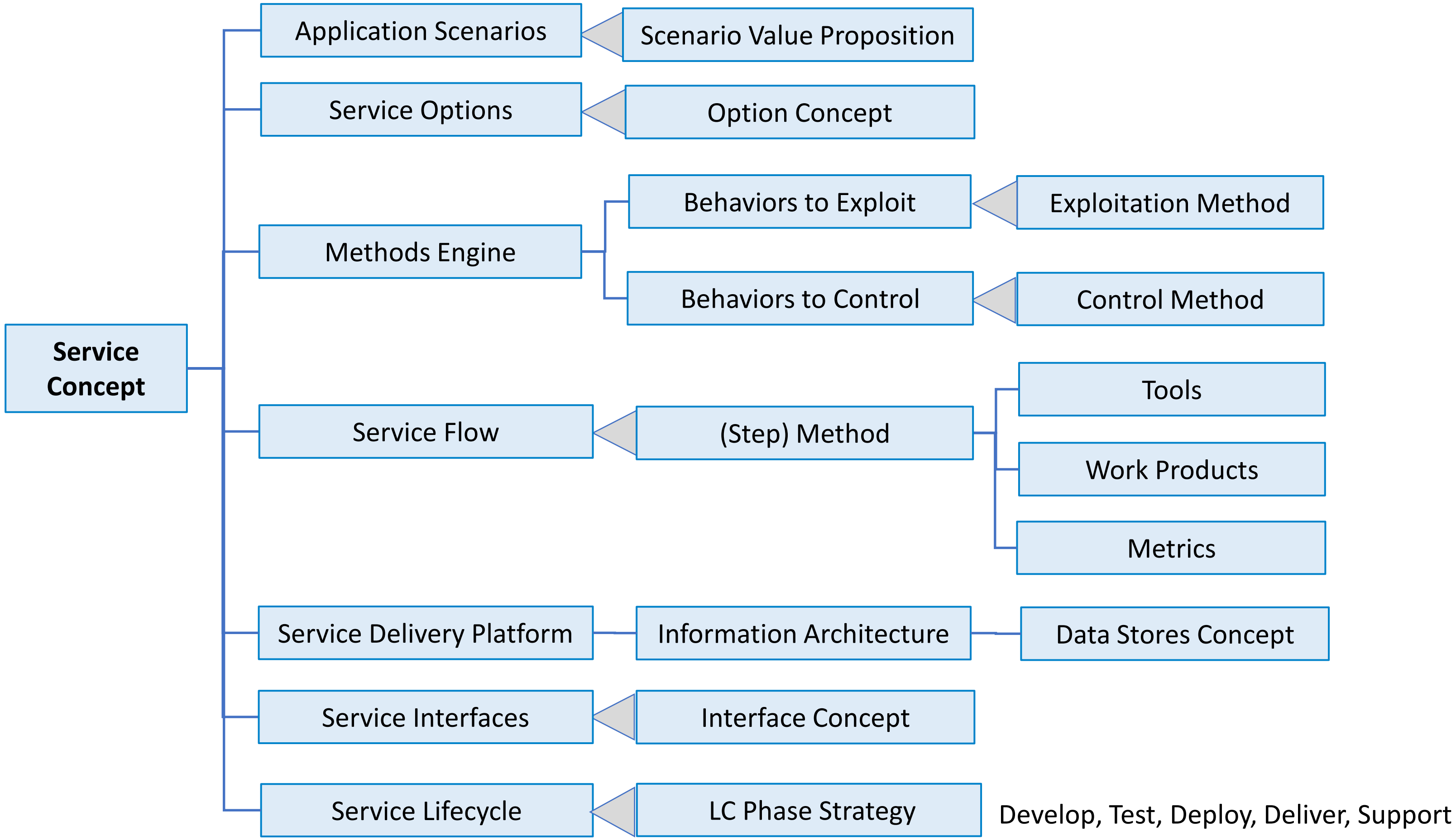
- Technology Roadmapping
- Requirements Management
- System Design
- Manufacturing Operations Management

Number	Decision Name	Decision Description	Decision Class
1	Capability Concept	What is the top-level architecture, design or implementation concept for this capability?	Single Answer
1.1	Usage Scenarios	Where (in which situation, scenarios) will we apply this capability?	Multiple Answer
1.1.1	Value Proposition	How will this capability offer unique value in this usage scenario?	Single Answer
1.2	Core Methods	What methods or combination of methods provide the engine for this capability?	Multiple Answer
1.3	Process Architecture	What process architecture, framework or flow will we use to deploy this capability?	Multi-part Answer
1.3.1	Process Design	How will this part of our process operate?	Single Answer

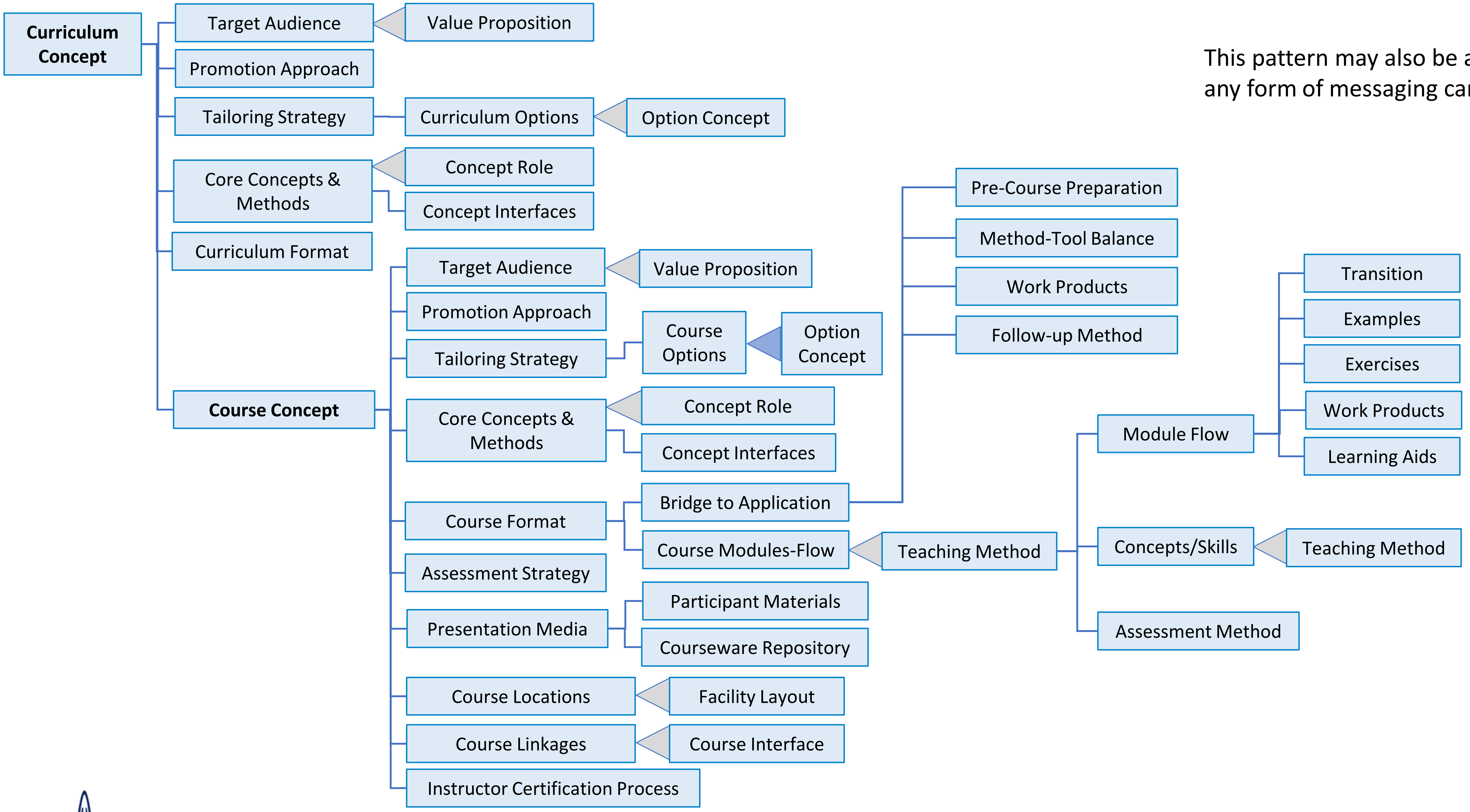
Decision Class governs the "fan-out" of the decision model:

- Single Answer (Technology)
- Multiple Answer (Portfolio)
- Multi-part Answer (Architecture)

Service Design Decision Pattern

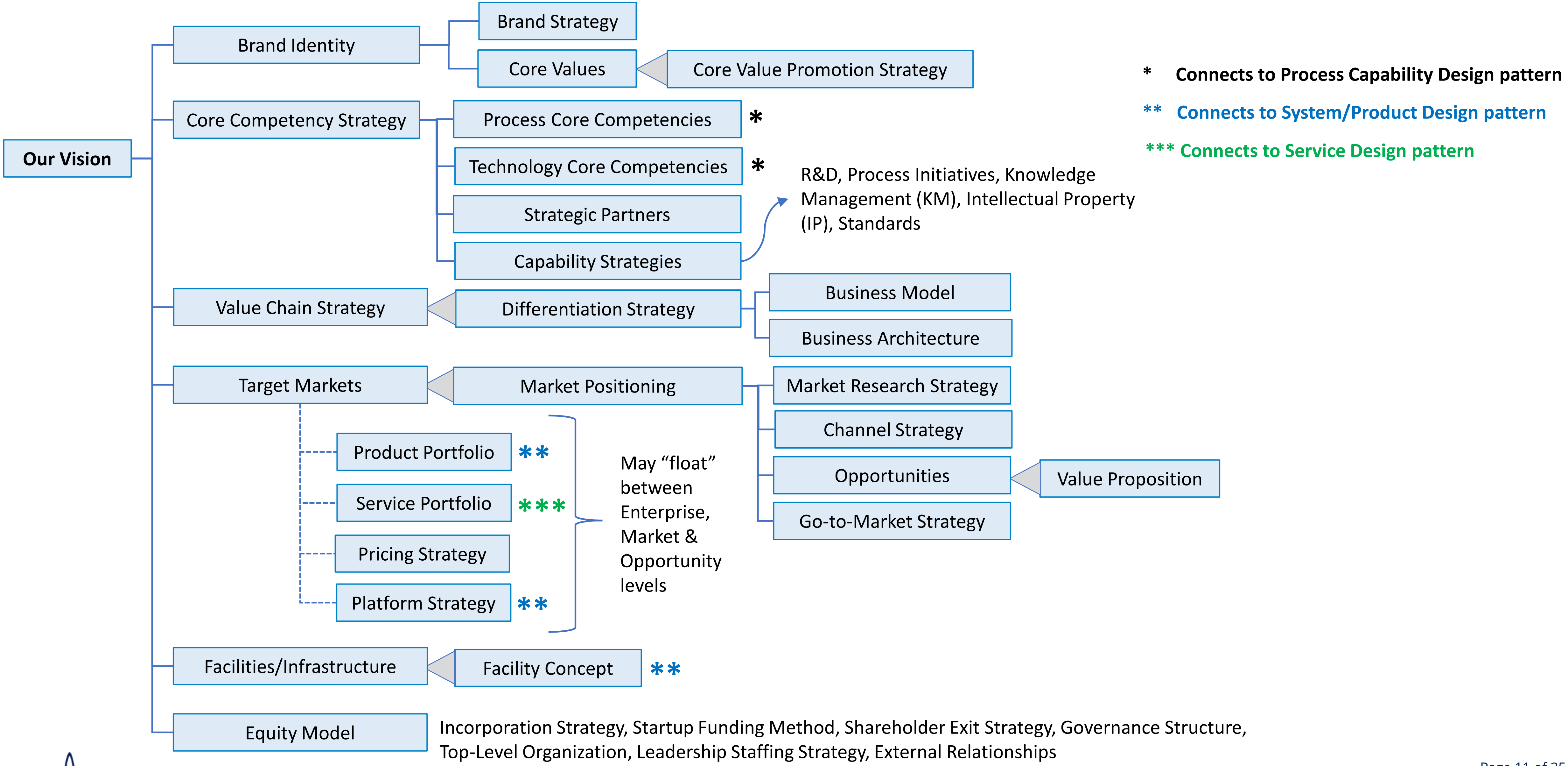


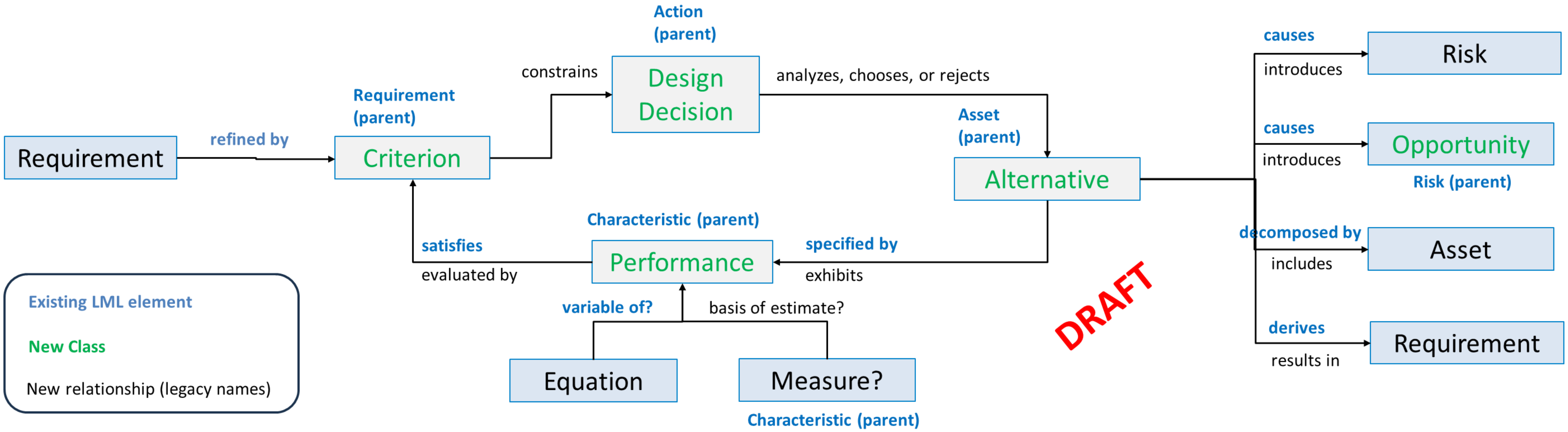
Curriculum/Courseware Design Decision Pattern



This pattern may also be applied to any form of messaging campaign

Enterprise Strategy Decision Pattern





My Nth rodeo in mapping decision data to a structured language and tool schema. Initial mapping shown above

PROCESS:

- Populated rich examples to highlight information gaps (entity classes, relationships, attributes)
- Visualized examples in Innoslate to uncover and highlight software capability gaps

NEXT: Engage LML community to work through information modeling tradeoffs -> elegance

Essential skill: Two-dimensional mapping process

Decision-Centric SE Information Metamodel

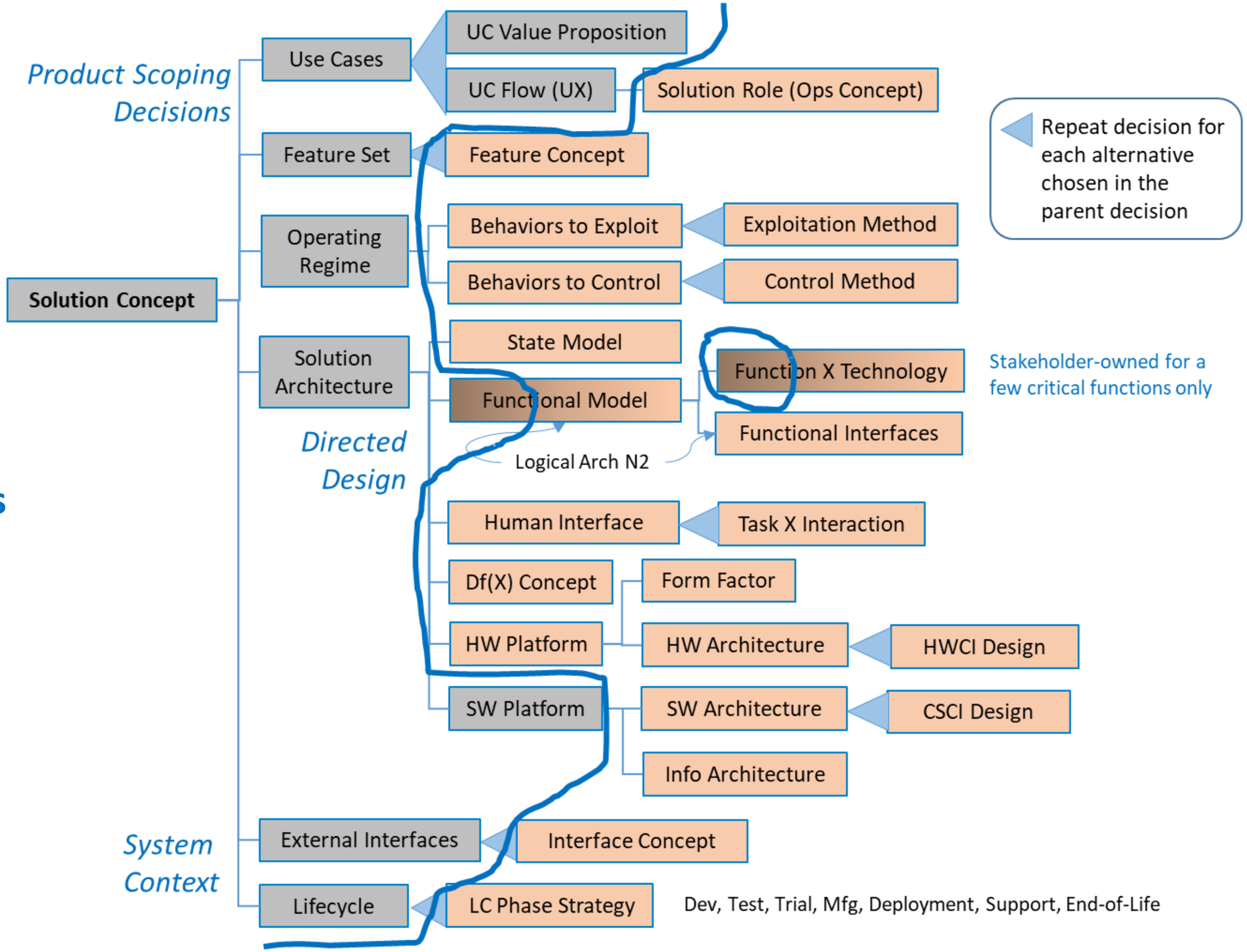
Decision Pattern – Problem Domain Decomposition

Decision Pattern	Requirement	Criterion	Decision	Alternative	Performance	Derived Requirement	Risk	Mitigation
1. Solution Concept								
1.1 Use Cases to Support								
1.1.1 Use Case Value Proposition								
1.1.2 Use Case Flow								
1.1.2.1 Subsystem Role (Ops Concept)								
1.2 Feature Set								
1.2.N Feature Concept								
1.3 Operating Regime(s)								
1.3.1 Research Strategy								
1.3.2 Behaviors to Exploit								
1.3.2.1 Exploitation Method								

Source document paragraph

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

One Model with Many Uses

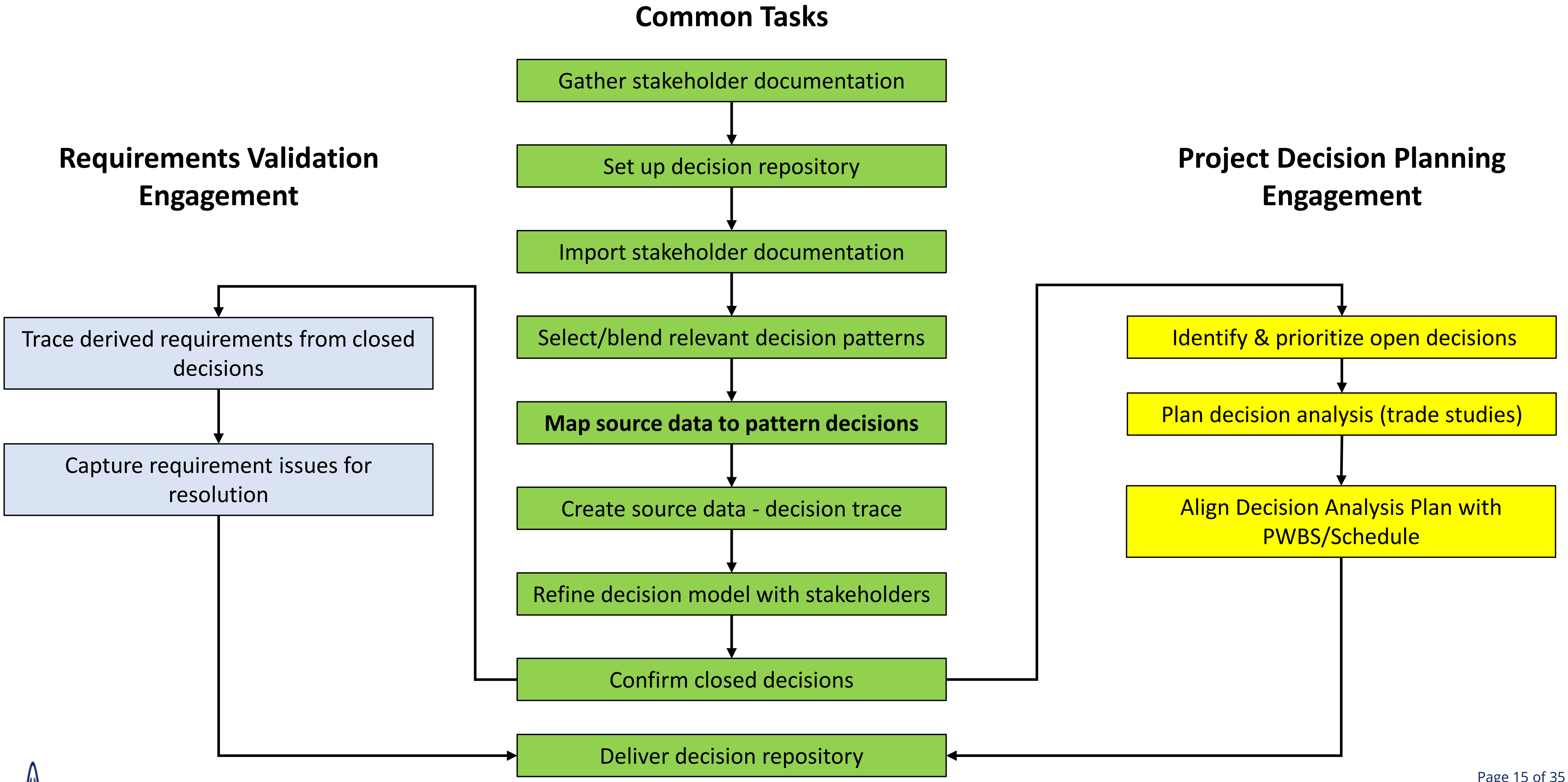


Validate Requirements from "Closed" Decisions

Plan Decision-Making for "Open" Decisions

Discover Decision "Frontier"

Decision Pattern Engagements



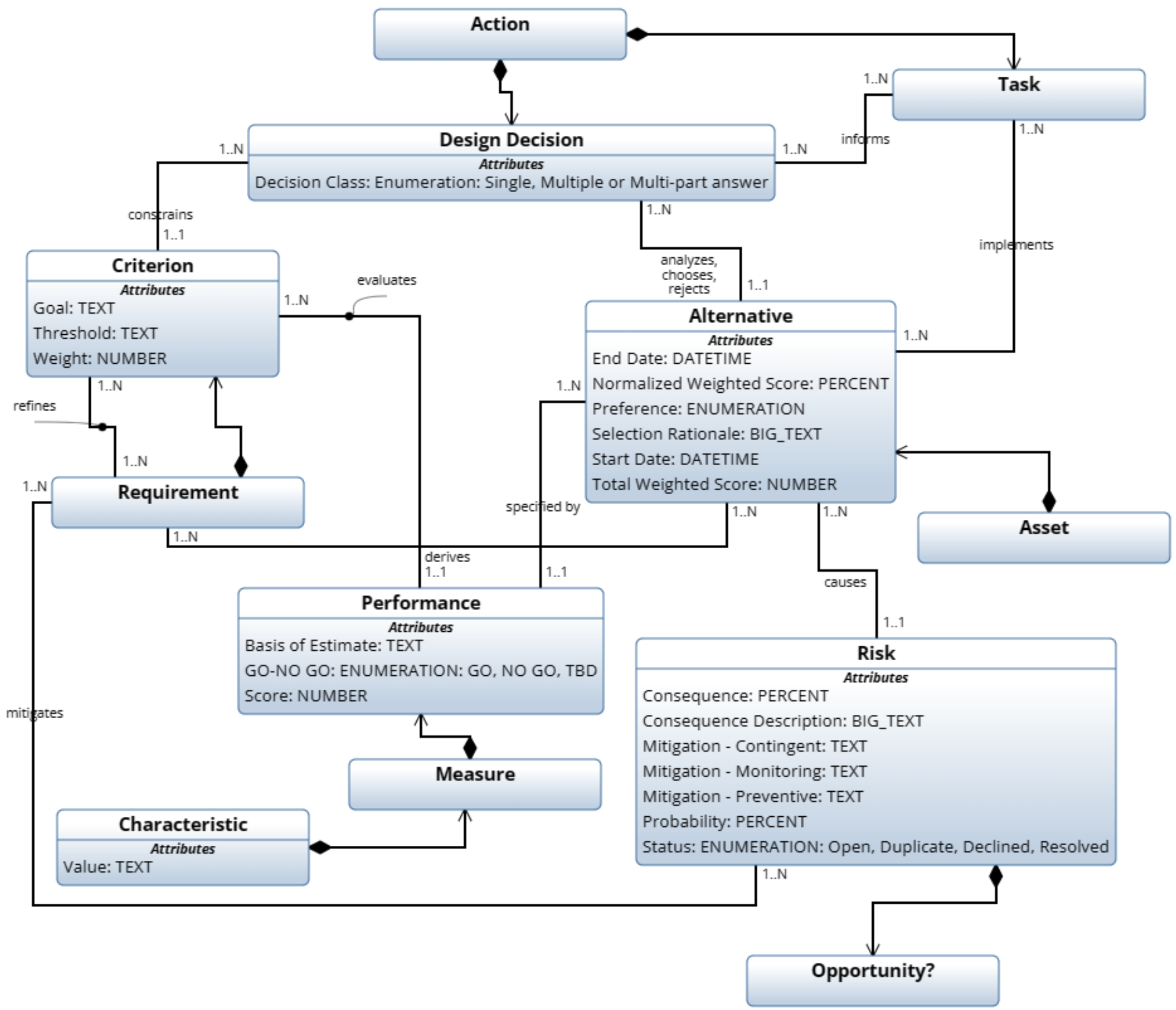
As implemented in Innoslate V4.9. All new classes have been implemented as subclasses

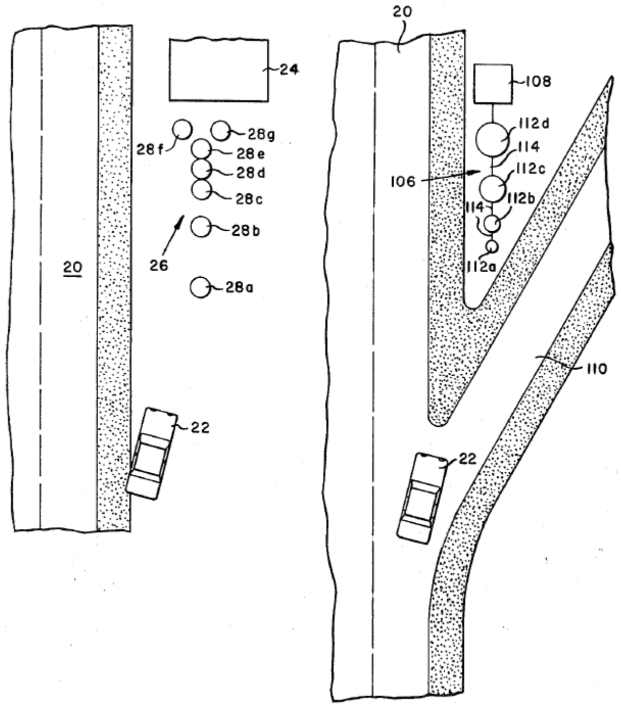
New subclasses

- Design Decision
- Criterion
- Alternative
- Performance
- Opportunity

Open issues

- Attributes for Performance to enable Weighted Score evaluation?
- Opportunity as subclass of Risk (or as further generalization of discrete uncertain events)?




20
24
28f
28g
28e
28d
28c
28b
28a
26
22
108
112d
114
112c
106
112b
112a
110

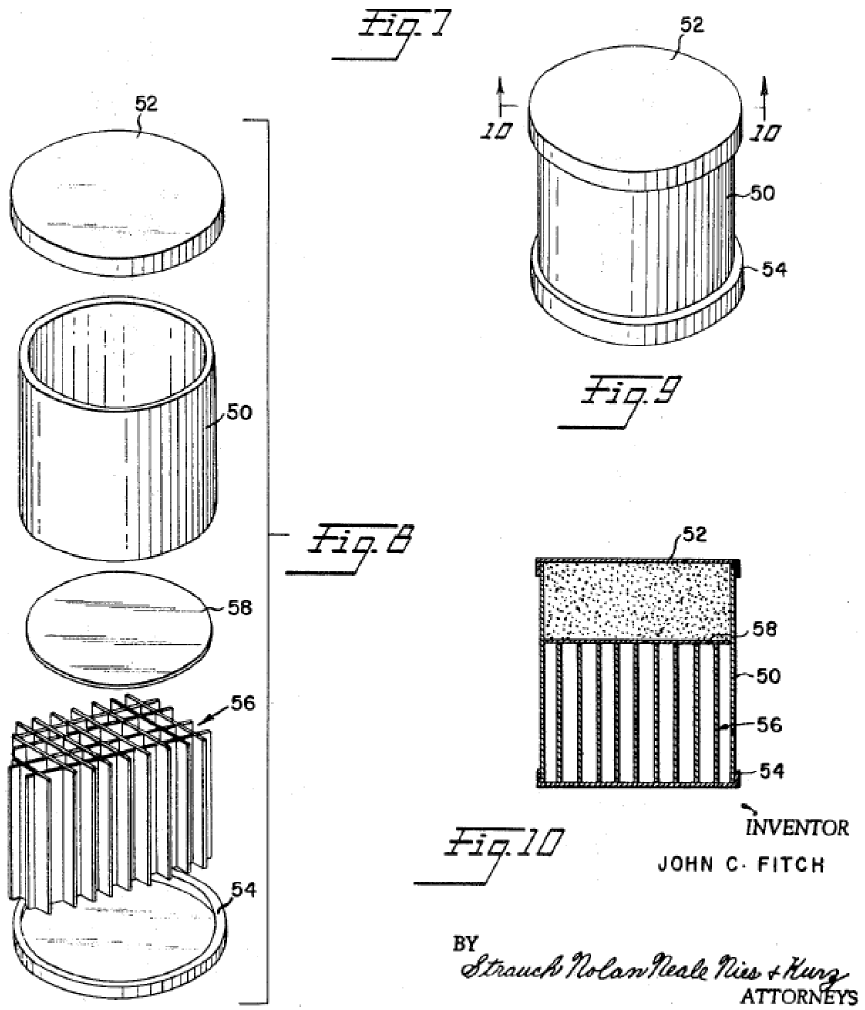


Fig. 7
52
10
10
50
54
Fig. 9
52
50
58
56
54
Fig. 8
52
58
50
56
54
Fig. 10
INVENTOR
JOHN C. FITCH
BY
Strauch Nolan Neale Nies & Kurz
ATTORNEYS

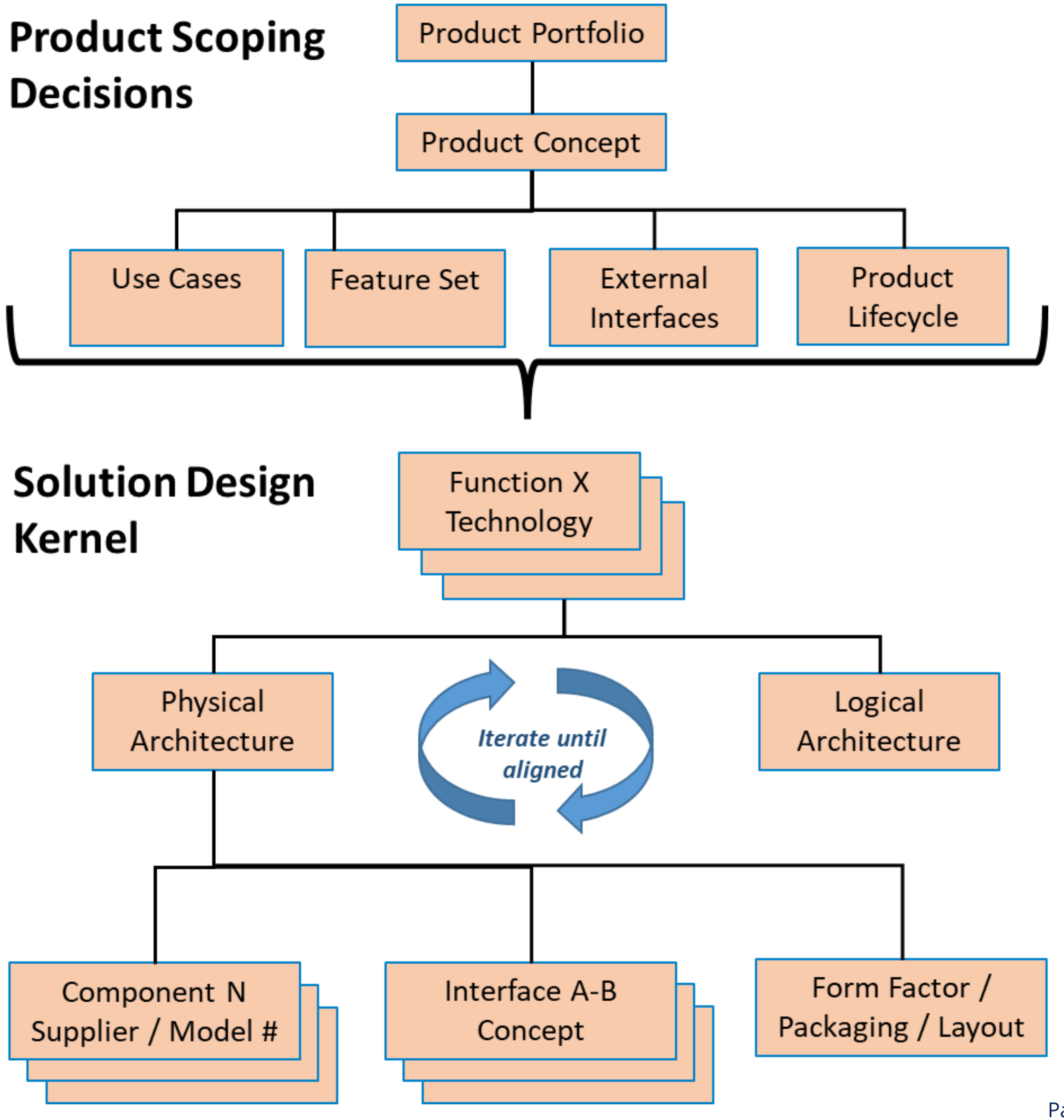
Use decision patterns to frame the problem, accelerate solution development and increase stakeholder value.

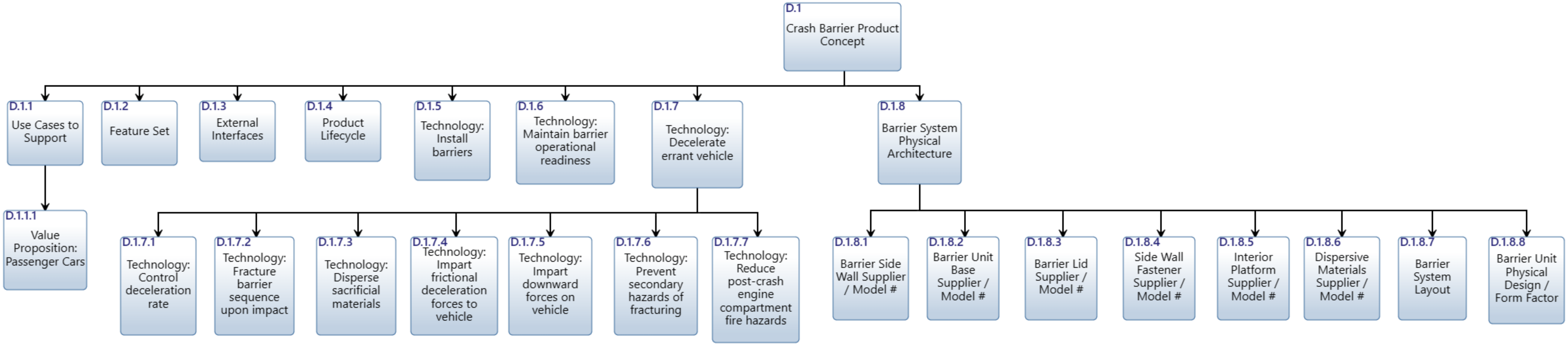
Requirements Validation

- Reverse engineer stakeholder decisions to validate requirements & bound project scope

Project Decision Planning -> Design

- Proactively identify & prioritize “open” decisions; plan analysis to inform them. Execute the design plan





Reverse engineered 25 decisions from the Crash Barrier patent using the Product Design decision pattern.

GAP: Inefficient process for instantiating cross-project decisions (seeding current project decisions & criteria from the pattern)

GAP: Visualizing pattern “where-used” traceability to support continuous pattern refinement.

Visualize design decisions and alternatives in a compact table form. Tabular equivalent to a multi-panel Decision Breakdown Structure (hierarchy)

Design Summary

- Conduct reverse engineering “Decision Blitz”
- Communicate decision priorities, status, analysis plans, or current design thinking

Multiple variants

- Brainstorm alternatives to evaluate
- Add Selection Rationale for alternatives

Decision Name	chooses Alternative
D.1 Crash Barrier Product Concept	A.1 Array of energy absorbing barrier units
D.1.1 Use Cases to Support	A.1.1.a Passenger cars
D.1.1.1 Value Proposition: Passenger Cars	A.1.1.1 Low cost barriers with high occupant protection performance via limited and "smooth"
D.1.2 Feature Set	A.1.2 Variable capacity solution using modular components
D.1.3 External Interfaces	A.1.3.c Barrier-Ambient Environment Interface
	A.1.3.b Barrier-Highway Infrastructure Interface
	A.1.3.a Automobile - Barrier Interface
D.1.4 Product Lifecycle	A.1.4 Set of modular components, assembled and configured in field. Near-zero maintenance.
D.1.5 Technology: Install barriers	A.1.5 Onsite assembly and configuration of barrier units
D.1.6 Technology: Maintain barrier operational readiness	A.1.6 Waterproof barrier units with tamper-resistant lids
D.1.7 Technology: Decelerate errant vehicle	A.1.7 Progressive fracturing of barrier units to transfer momentum and create friction
D.1.7.1 Technology: Control deceleration rate	A.1.7.1 Barrier units with differing masses spaced to "smooth" the deceleration forces
D.1.7.2 Technology: Fracture barrier sequence upon impact	A.1.7.2 Frangible cylindrical barrier units with break points
D.1.7.3 Technology: Disperse sacrificial materials	A.1.7.3 Dispersive material absorbs vehicle momentum
D.1.7.4 Technology: Impart frictional deceleration forces to vehicle	A.1.7.4 Build-up of dispersive material creates bulldozer effect
D.1.7.5 Technology: Impart downward forces on vehicle	A.1.7.5 Elevated dispersive materials above vehicle center of mass imparts downward force
D.1.7.6 Technology: Prevent secondary hazards of fracturing	A.1.7.6 Barrier units constructed to minimize size of broken "shards".
D.1.7.7 Technology: Reduce post-crash engine compartment fire hazards	A.1.7.7 Engine compartments fill with fire-retardant dispersive materials (sand)
D.1.8 Barrier System Physical Architecture	A.1.8 Configurable array; units of similar shape, varying in size and fill
D.1.8.1 Barrier Side Wall Supplier / Model #	A.1.8.1 TBD: Sheet of plastic with breaklines and rivet holes
D.1.8.2 Barrier Unit Base Supplier / Model #	A.1.8.2 TBD: Circular plastic base
D.1.8.3 Barrier Lid Supplier / Model #	A.1.8.3 TBD: Circular plastic lid with tamper-resistant closure
D.1.8.4 Side Wall Fastener Supplier / Model #	A.1.8.4 Standard rivets, size TBD
D.1.8.5 Interior Platform Supplier / Model #	A.1.8.5 TBD: Elevated variable-height platform - interior pedestals + circular divider
D.1.8.6 Dispersive Materials Supplier / Model #	A.1.8.6 Dry sand or equivalent
D.1.8.7 Barrier System Layout	A.1.8.7 Series of barrier units of increasing size/mass arranged linearly. See Figure N
D.1.8.8 Barrier Unit Physical Design / Form Factor	A.1.8.8 Cylindrical containers in a discrete range of sizes

Decision ID & Name	Decision Question	Alternatives Considered	
DD.1 - Services Portfolio	What set of services will we deliver to these customers?	Project Decision Jump-Start (PDJS)	ALT.1.a
DD.1.1 - Service Concept	What is the top-level concept for this service? What will be offered in what situations? What makes it unique?	Capture stakeholder decision context + project decision baseline	ALT.1.1.a
DD.1.1.1 - Application Scenarios	In what scenarios or situations will this service be delivered?	Requirements Validation (RV)	ALT.1.1.1.a
		Project Decision Planning (PDP)	ALT.1.1.1.b
DD.1.1.1.1 - Value Proposition: Requirements Validation	How will the service deliver value in the Requirements Validation scenario or situation?	Significant improvement to requirements quality + stakeholder concurrence	ALT.1.1.1.1
DD.1.1.1.2 - Value Proposition: Project Decision Planning	How will the service deliver value in the Project Decision Planning scenario or situation?	Aligned problem definition with project design scope	ALT.1.1.1.2
DD.1.1.2 - Service Options	What are the primary service options (bundles of work products) that will be offered?	Decision coaching - Decision-centric Digital Thread	ALT.1.1.2.f
		Requirements Validation (RV) standalone	ALT.1.1.2.a
		Project Decision Planning (PDP) standalone	ALT.1.1.2.b
		RV + PDP bundle	ALT.1.1.2.c
		RV + RQM bundle	ALT.1.1.2.d
		MBSE tool extension for decision management	ALT.1.1.2.e
DD.1.1.3 - Methods Engine	What methods or combination of methods provide the engine for this service?	Pattern-based decision reverse engineering	ALT.1.1.3.a
		Reqt - Decision - Reqt traceability	ALT.1.1.3.b
DD.1.1.3.1 - Behaviors to Exploit	What human behaviors or scientific principles will be exploited to create value within this service?	Pattern-driven continuous improvement	ALT.1.1.3.1.a
		Continuous derivation traceability	ALT.1.1.3.1.b
DD.1.1.3.1.1 - Exploitation Method: Pattern-driven continuous improvement	How will the service exploit this behavior/principle to deliver value?	Jumpstart creation of customer-owned knowledge assets	ALT.1.1.3.1.1.a
DD.1.1.3.1.2 - Exploitation Method: Continuous derivation traceability	How will the service exploit this behavior/principle to deliver value?	Continuous requirement, decision and plan alignment	ALT.1.1.3.1.2.a
DD.1.1.3.2 - Behaviors to Control	What human behaviors or scientific principles will be controlled (regulated, suppressed or avoided) to realize value?	Human doubts about patterns - the belief that every project is unique.	ALT.1.1.3.2.b
DD.1.1.3.2.1 - Control Method: Human doubts about patterns	How will the service control or suppress this unwanted behavior/principle?	Offline reverse engineering creates believable, traceable decision model	ALT.1.1.3.2.1.c
DD.1.1.4 - Service Flow	What series of steps will deliver this service? How will the engagement flow?	See Process N2:	ALT.1.1.4.a

Crash Barrier - Decision Breakdown Structure (DBS)

- **Crash Barrier Product Concept**
  Array of energy absorbing barrier units
  - **Use Cases to Support**
    Passenger cars
  - **Technology: Decelerate errant vehicle**
    Progressive fracturing of barrier units to transfer momentum and create friction
    - **Technology: Control deceleration rate**
      Barrier units with differing masses spaced to "smooth" the deceleration forces
    - **Technology: Impart downward forces on vehicle**
      Elevated dispersive materials above vehicle center of mass imparts downward force
  - **Barrier System Physical Architecture**
    Configurable array; units of similar shape, varying in size and fill

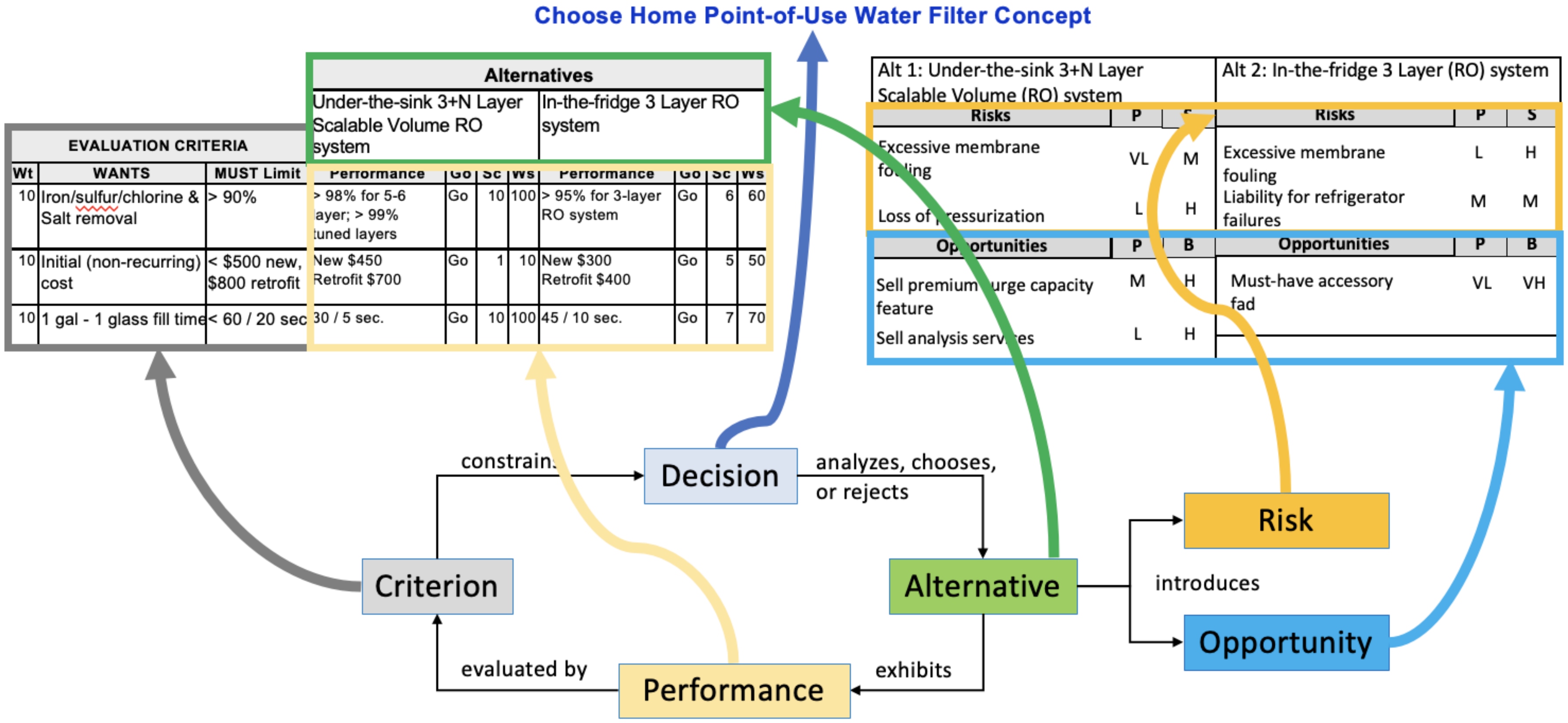
NEED: Support for general purpose multi-panel box defined by either relationships or attributes to enable compact user-configurable visualizations

Design Decision (Name)

-> *chooses* -> Alternative

Reverse engineered 25 decisions from the Crash Barrier patent using the Product Design decision pattern.

GAP: Lack "one-page" graphical design summary. No multi-panel decision "boxes" with Decision Name + Alternative(s) chosen and/or analyzed.



Capture the data required to fully inform the decision analysis process (Screening & scoring).

GAP: Inability to visualize evaluation matrix data in compact form

GAP: Inefficient data entry using standard Entity editors

GAP: No built-in weighted score or normalized weighted score calculations

Crash Barrier - Evaluation Matrix Data

Criteria	Crit Weight	Design Decision	chooses Alternative	exhibits Performance	Score	Weighted Sc...	evaluated by Criterion
CR.1.a Death/Injury reduction per crash	2	D.1 Crash Barrier Product Concept - Criteria-Performance Product Concept Eval Matrix data: Criteria and associated alternative performance	Array of energy absorbing barrier units	Pf.CR.1.i.Alt.1 $X in new highway equipment cost	6	12	CR.1.i Compatibility with existing highway maintenar
CR.1.b Range of vehicles (crash scenarios) mitigated	2			Pf.CR.1.h.Alt.1 Very limited collateral damage	9	18	CR.1.h Collateral damage to other vehicles, infrastruc
CR.1.c Lifecycle cost per installation	2			Pf.CR.1.g.Alt.1 $X K restoration cost	3	6	CR.1.g Barrier post-crash restoration cost
CR.1.d Barrier useful life	2			Pf.CR.1.f.Alt.1 X% loss of vehicle value	8	16	CR.1.f Damage to errant vehicle
CR.1.e Reconfigurability / reuse	2			Pf.CR.1.e.Alt.1 X% component reconfigurability/reuse	4	8	CR.1.e Reconfigurability / reuse
CR.1.f Damage to errant vehicle	3			Pf.CR.1.d.Alt.1 20-25 year life	6	18	CR.1.d Barrier useful life
CR.1.g Barrier post-crash restoration cost	4			Pf.CR.1.c.Alt.1 $X K LC cost	9	36	CR.1.c Lifecycle cost per installation
CR.1.h Collateral damage to other vehicles, infrastructure	5			Pf.CR.1.b.Alt.1 X% of vehicle crash scenarios	8	40	CR.1.b Range of vehicles (crash scenarios) mitigat
CR.1.i Compatibility with existing highway maintenance	5			Pf.CR.1.a.Alt.1 X% death/injury reduction	8	40	CR.1.a Death/Injury reduction per crash

Capture the data required to fully inform the decision analysis process (Screening & scoring).

GAP: Inability to visualize evaluation matrix data in compact form

GAP: Inefficient data entry using standard Entity editors

GAP: No built-in weighted score or normalized weighted score calculations

The Performance cells in a typical Evaluation Matrix are first-class entities with multiple attributes, not just relationships

Desire direct input to matrix

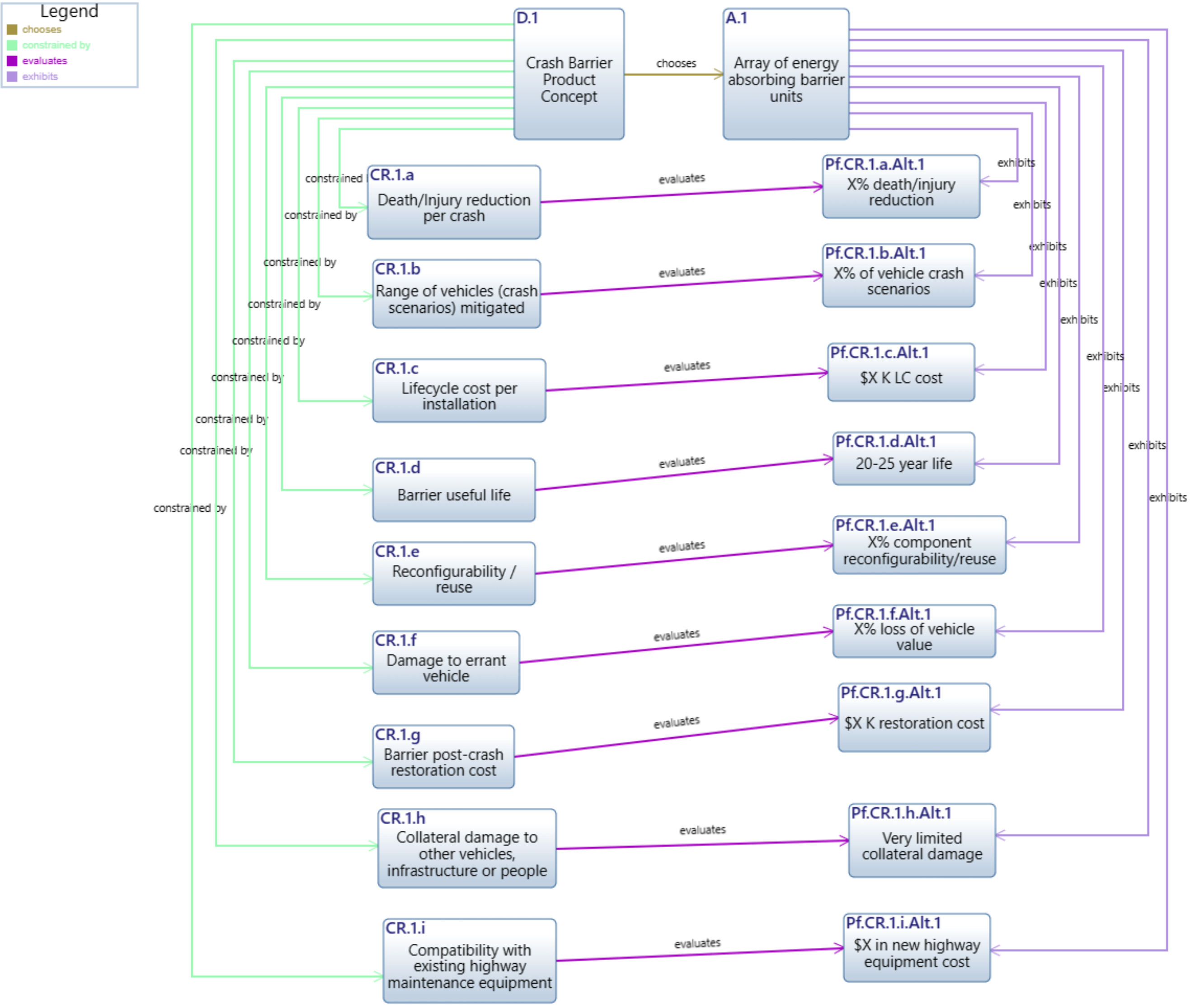
Current process:

- Create Performance entities
- Associate with Alternative
- Associate with Criterion
- Edit attributes

Visualize decision data

Move between equivalent views:

- Matrix
- Radar
- Tornado?



A bit of a maze

Radar Diagram can visualize weighted scoring judgments (performance against criteria) for a single alternative vs objective/goal value

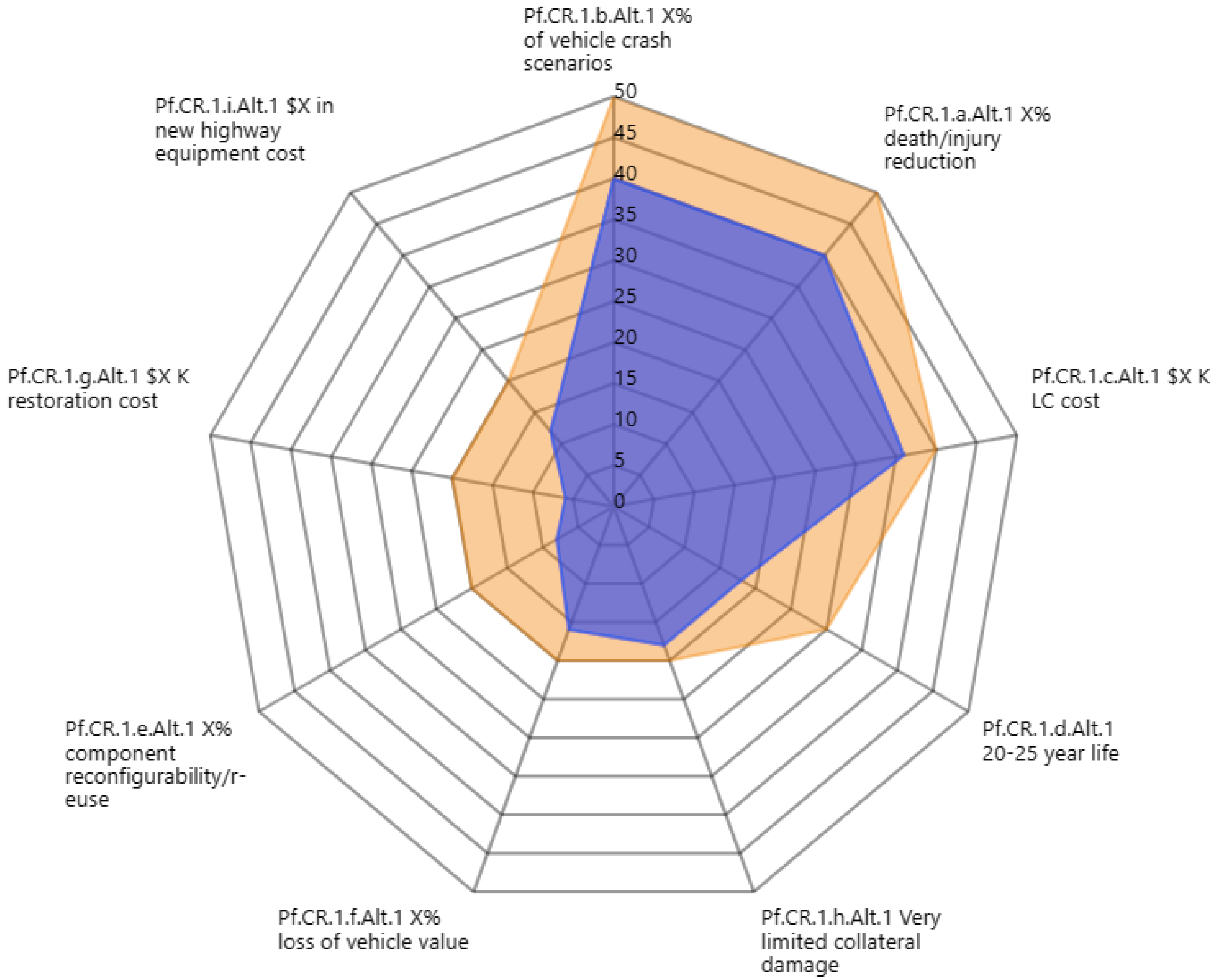
Usability GAPS:

- Inefficient entry of Performance data
- Manual diagram setup process; no defaults

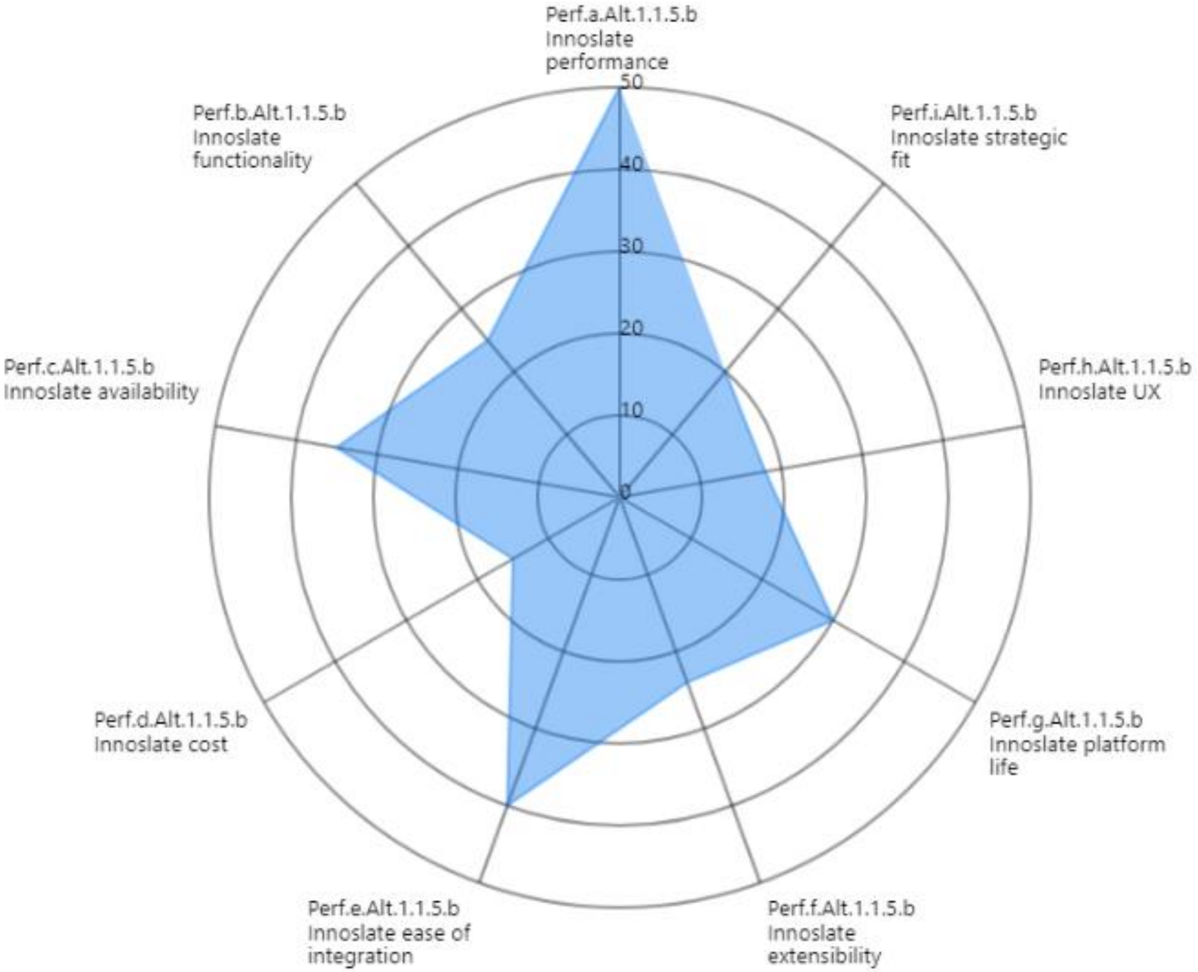
Capability GAPS:

- No multiple-alternative comparisons; multiple side-by-side charts hard to compare
- Can’t sort criteria by weight or weighted score attributes

Product Concept decision:
Fitch Inertial Barrier alternative



Radar Diagram



Service Delivery Platform decision: Innoslate alternative

Risk Matrix

	Negligible	Minor	Moderate	Serious	Critical
High					
Medium High					
Medium				• Rsk.3.ALT.1.1.5.b R-D-R traceability thread in Innoslate	• Rsk.1.ALT.1.1.5.b Limited Innoslate use on complex
Medium Low				• Rsk.2.ALT.1.1.5.b Innoslate decision-focused features	• Rsk.6.ALT.1.1.5.b Innoslate UX poor for new decision-focused
Low			• Rsk.5.ALT.1.1.5.b Innoslate market shared eroded by		• Rsk.4.ALT.1.1.5.b LML fails to adequately incorporate

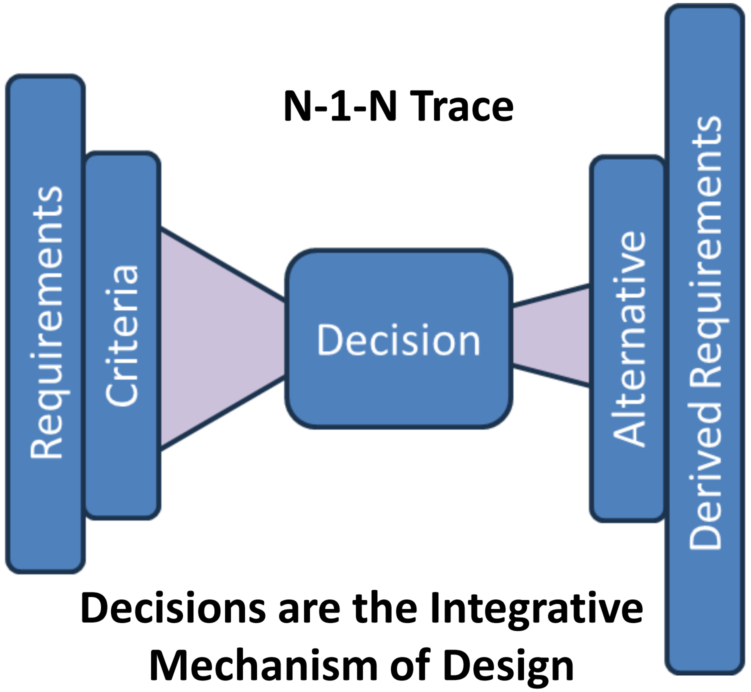
Crash Barrier - Decision-to-X Traceability

Requirement	Criteria	Design Decision	chooses Alternative	derives Requirement
	CR.1.i Compatibility with existing highway maintenance	D.1 Crash Barrier Product Concept Product Concept Eval Matrix data: Criteria and associated alternative performance	Array of energy absorbing barrier units	
	CR.1.h Collateral damage to other vehicles, infrastructure			
	CR.1.g Barrier post-crash restoration cost			
	CR.1.f Damage to errant vehicle			
	CR.1.e Reconfigurability / reuse			
	CR.1.d Barrier useful life			
	CR.1.c Lifecycle cost per installation			
	CR.1.b Range of vehicles (crash scenarios) mitigated			
	CR.1.a Death/Injury reduction per crash			
		D.1.1 Use Cases to Support	Passenger cars	R.62 Vehicle size ranges
				R.59 Vehicle mass ranges (2000-4500 lbs)
				R.58 Vehicle initial velocity ranges (up to 60 MPH)
		D.1.1.1 Value Proposition: Passenger	Low cost barriers with high occupant protection	R.53 Variable capacity solutions for different sites
		D.1.2 Feature Set	Variable capacity solution using modular components	R.56 Vary barrier unit number
				R.55 Vary barrier unit mass (fill)
				R.54 Vary barrier unit array layout
		D.1.3 External Interfaces	Barrier-Ambient Environment Interface	R.14 Durability across relevant environment(s)
			Barrier-Highway Infrastructure Interface	R.64 Site layout constraints
			Automobile - Barrier Interface	R.63 Vehicle-barrier impact profiles

MANY Requirements -> Criteria

One Decision

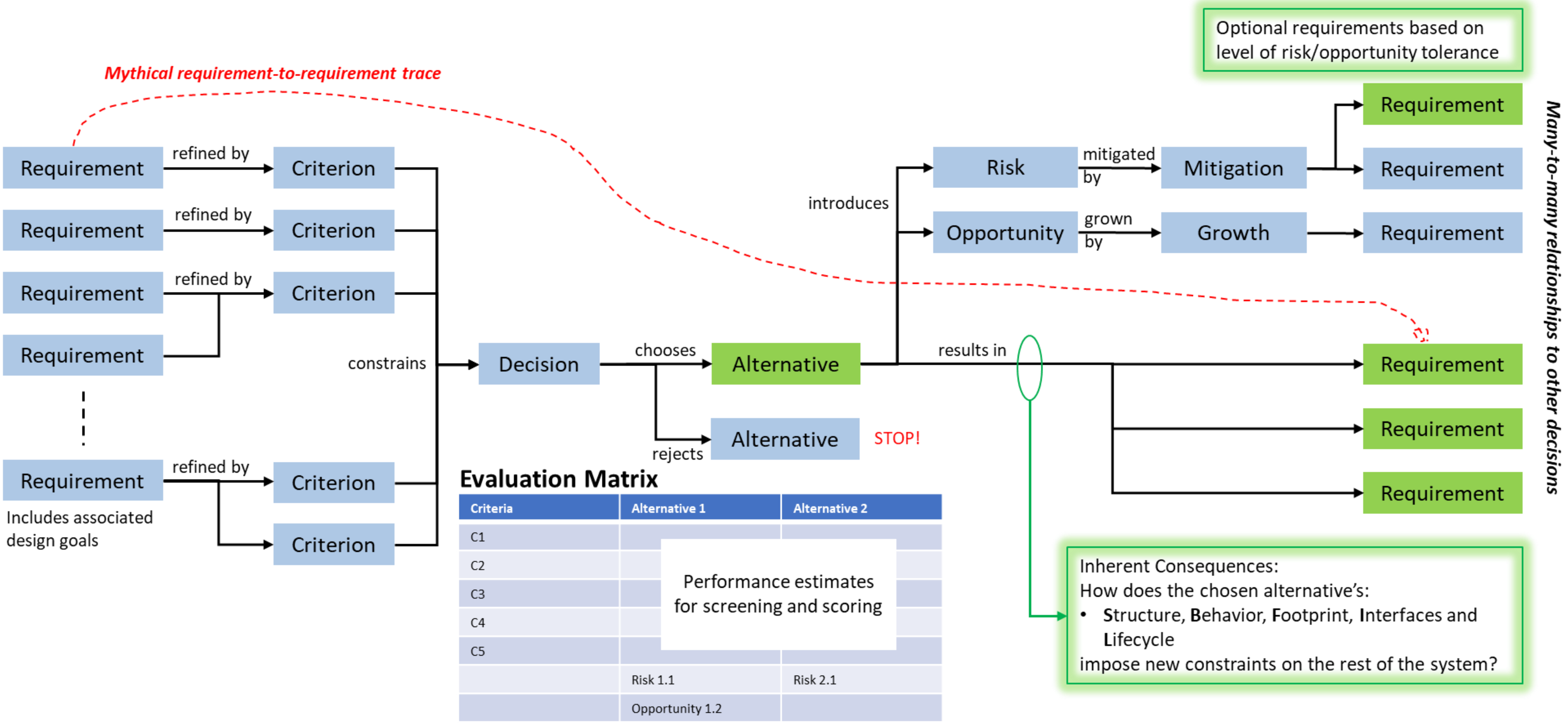
MANY Requirements



Need: "Decision-in-the-Middle" view to communicate how multiple requirements/goals drive a decision, which then creates multiple derived requirements based on the chosen alternative.

GAP: Display of N-1-N traceability topology is painful (Manual Spider Diagram setup)

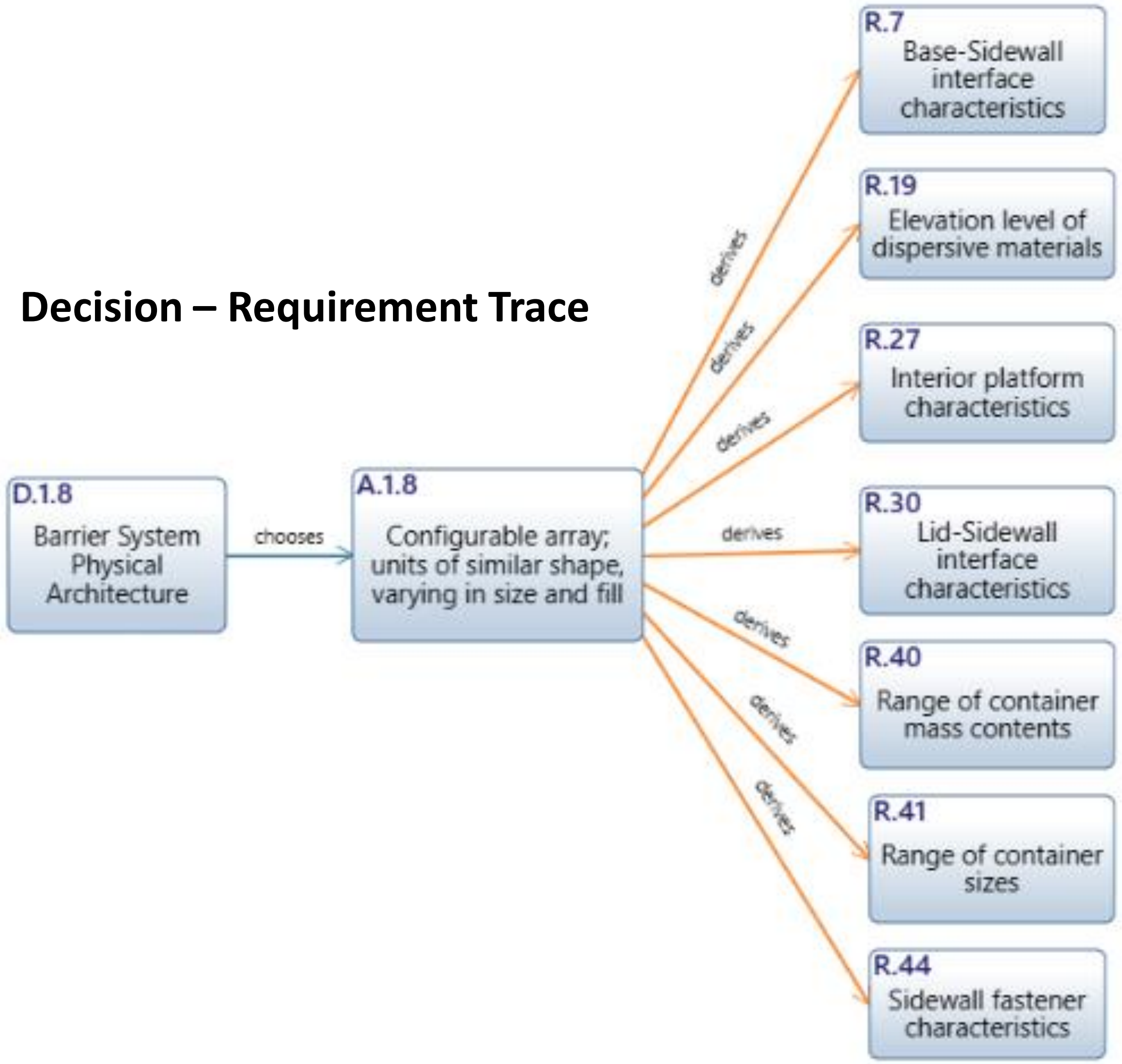
Requirement – Decision – Requirement Trace (N-1-N Trace)



Evaluation Matrix

Criteria	Alternative 1	Alternative 2
C1		
C2		
C3		
C4		
C5		
	Risk 1.1	Risk 2.1
	Opportunity 1.2	

Decision – Requirement Trace



Visualize how design decisions, through the alternatives chosen, create all "downstream" model entities

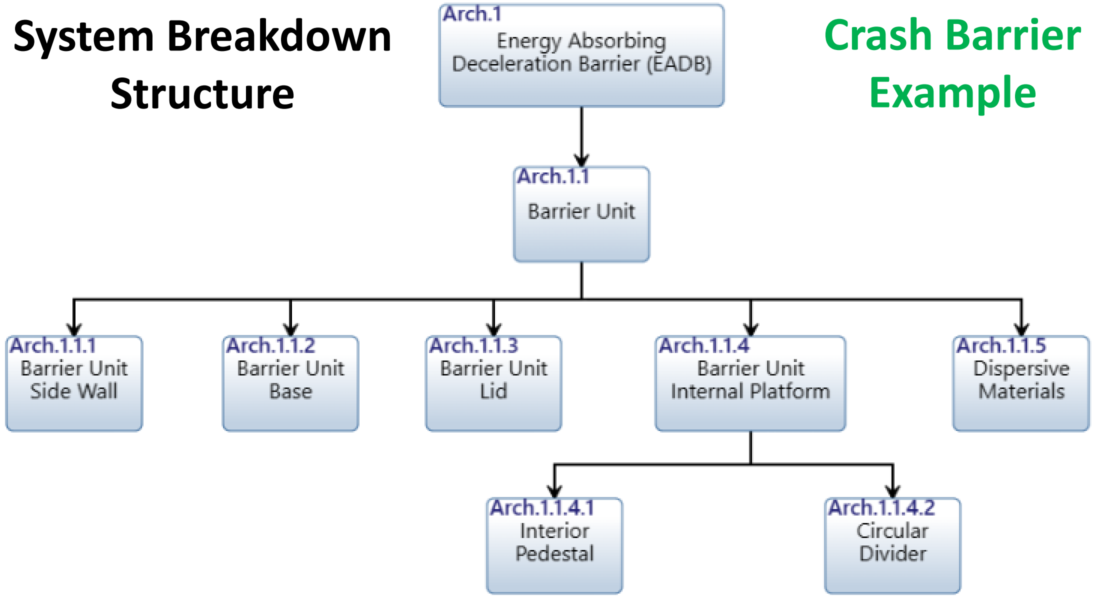
Decisions create Requirements

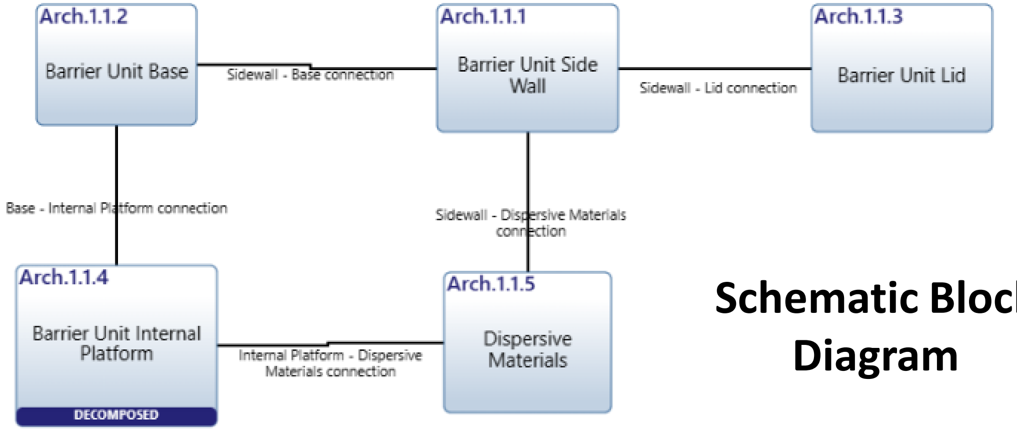
Inherent consequences of chosen alternative's:

- Structure
- Behavior
- Footprint
- Interfaces
- Lifecycle

System Breakdown Structure

Crash Barrier Example





Schematic Block Diagram

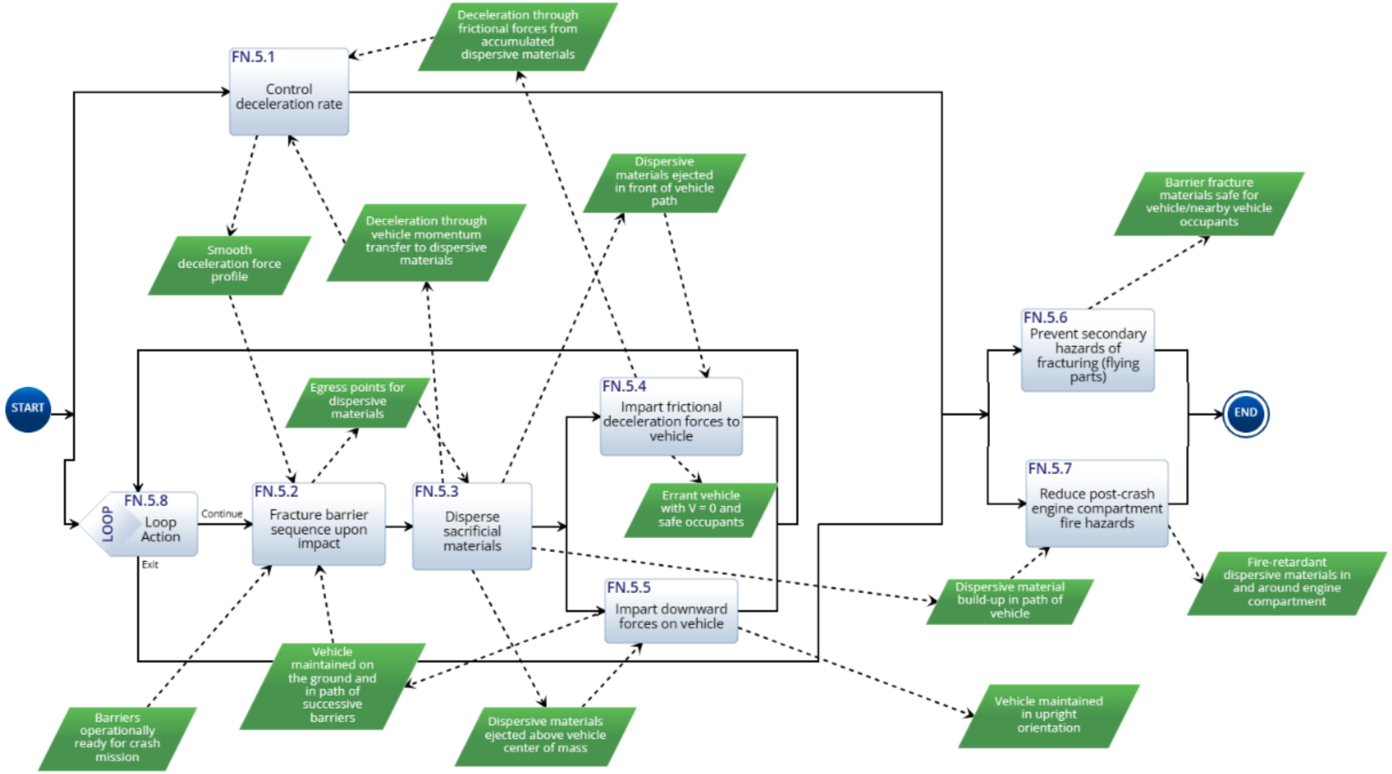
Visualize how design decisions, through the alternatives chosen, create all “downstream” model entities

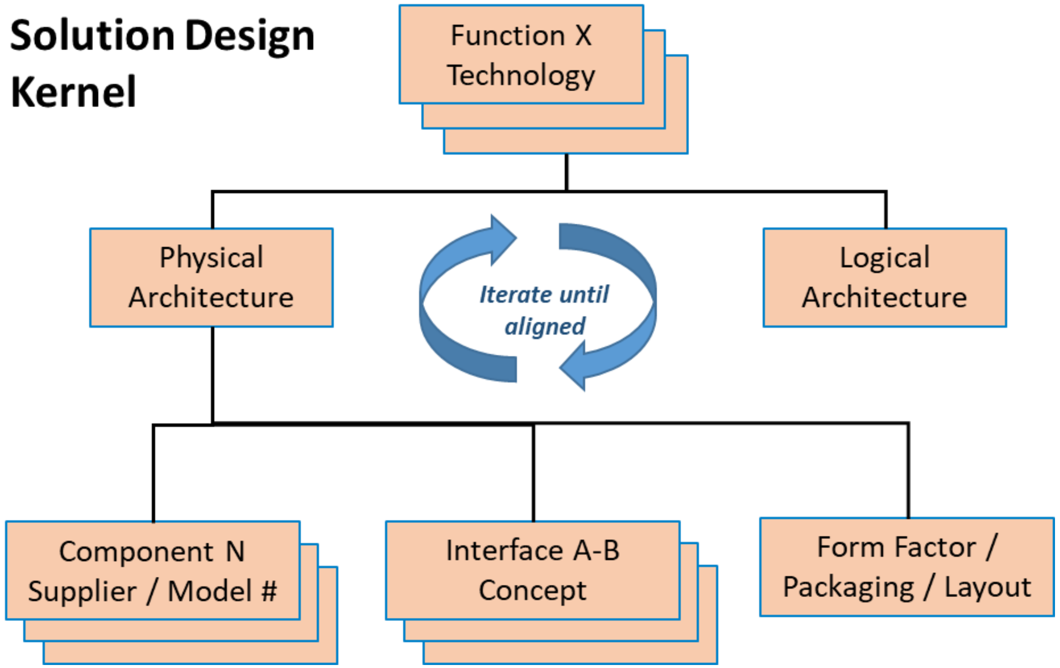
Decisions Create Architecture

Architecture decisions define system structure:

- Assets (system elements)
- Conduits (interfaces)

GAP: Alternatives from multiple decisions may ***shape*** each system element and interface. Difficult to quickly visualize these many-to-many relationships. Reuse N-1-N view?

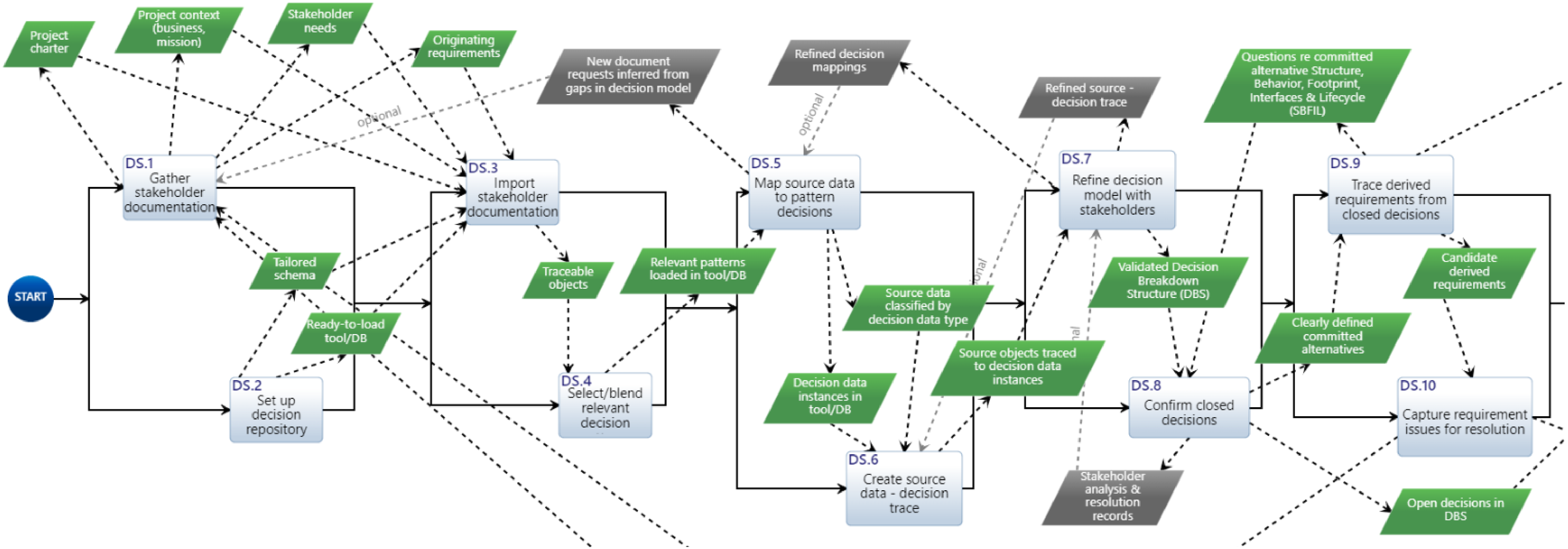




Model functional requirements to fully represent the as-designed behavior of the system, consistent with its physical architecture/design

GAP: Efficient methods to iterate and align physical and functional architectures, traced from design decision alternatives. (N-N-N relationships)

GAP: Maintenance of multiple overlapping designs during development



LML / Innoslate able to rigorously capture and visualize engagement flow (use case design)

Let's get started!

- Examples demonstrate that LML and Innoslate provide a great foundation for capturing design decisions and decision traceability
- But my prototypes are not likely the optimum extensions to LML **(or SysML 2.0)**
- Seeking your time to work through language tradeoffs & software features to support:
    - Rapid project decision framing through use of a decision pattern
    - Decision analysis capture and communication
    - Decision-to-everything traceability

- Who is available to dive in? How can we get this accomplished?

Decision patterns are proven and available

Most of your MBSE tools can be extended with a modest one-time effort while we wait for the standards and vendors to catch up

Project Decision Jump-start Services provide immediate payback

You can take ownership of a set of decision patterns that will:

- improve the value delivered to your stakeholders
- accelerate solutions into reality

PROJECT PERFORMANCE INTERNATIONAL

Thank you for attending this presentation!

Learn more about how to leverage decision patterns and traceability in your projects with Project Performance International (PPI) Project Decision Jump-Start Services.

Scan the QR code below or visit www.ppi-int.com/corporate-services/ppi-project-decision-jump-start-landing/ to discover how John Fitch can help you visualize stakeholders' decisions, validate project requirements, and plan for effective design decision-making.

Scan the QR code to Learn More