Design for Uncertainty

Dirk Pons

School of Engineering,
Christchurch Polytechnic Institute of Technology,
New Zealand

Overview

- Developed a descriptive meta-model for design.
- Focus here is on how the design objectives (needs) are determined, how early design concepts are formed, and how uncertainties are treated during design.
- Application is illustrated with data from a domestic dishwasher design.

1 Introduction

Development of engineering systems, e.g. plant or products, is a complex activity:

- Need to optimise multiple system attributes

  e.g. function, cost, resource usage, style

- Involves diverse activities

  e.g. design, risk assessment, decision making, construction, operation

- Complex failure modes exist.

  e.g. electronic functionality and active control systems

2a Problem definition

- Development of complex engineering systems is typically managed by decomposing the task into separate professional bodies of knowledge.
- These interface with each other in limited ways.
- Complexity arises from the incomplete nature of problem definition and the enormous number of design paths that may be taken.

2b Hypothesis

- That a descriptive meta-model could be developed to integrate the bodies of knowledge on design theory and uncertainty into a consistent epistemology.
- This is worth doing because it could increase understanding of the availability, strengths and weaknesses of the constituent methodologies, and identify issues for further study.

3 Other approaches

- Many design models exist

  e.g. Pahl & Beitz (1988), Hubka and Eder (1996), Finger & Dixon (1989), BS7000 (1989), Pugh (1991), Hales (1994), Crisp (1986), Candy et al (1996), Wallace (1987), Yassine et al (1999), Clarkson & Hamilton (2000)

- But do not specifically include uncertainty

  exceptions include Vose (1996), Wood & Antonsson (1989), Ullman & D'Ambrosio (1995), Ullman (2001), Clemen (1996), Ridgman (1996)

4 Method

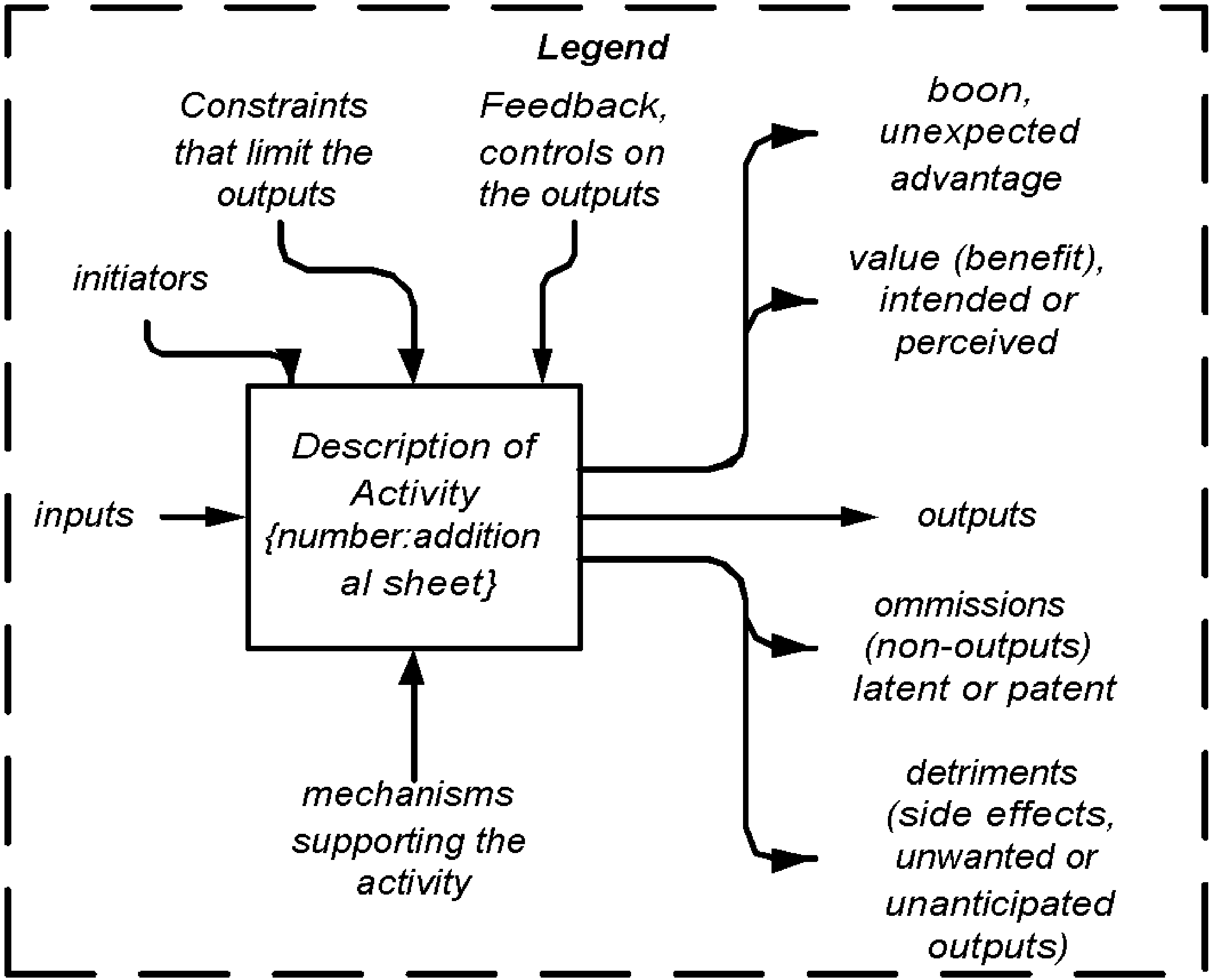
Applied dynamic process analysis (DPA):

- decomposes the process into sub actions,
- deduces initiating events, controls, inputs required, outputs, and mechanisms that are presumed to support the action,
- represented in IDEF0 notation.

4 IDEF0 Legend

- Inputs, Controls, Outputs, Mechanisms differentiated by placement relative to block…
- Emphasis is on activities rather than objects,

i.e. a process view rather than a state view


Legend
Constraints that limit the outputs
Feedback, controls on the outputs
initiators
Description of Activity {number:addition al sheet}
inputs
outputs
boon, unexpected advantage
value (benefit), intended or perceived
ommissions (non-outputs) latent or patent
detriments (side effects, unwanted or unanticipated outputs)
mechanisms supporting the activity

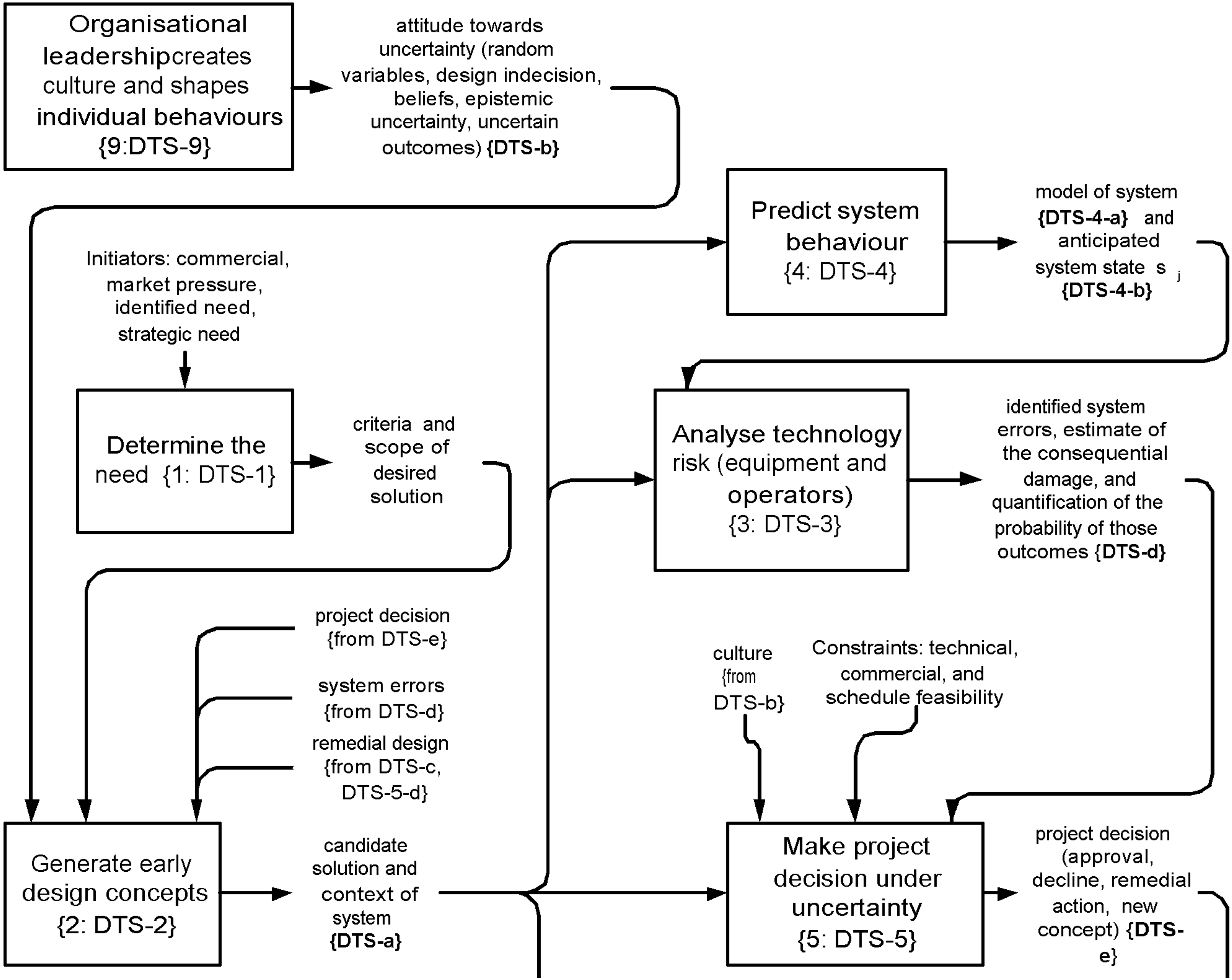
4 IDEF0 Legend

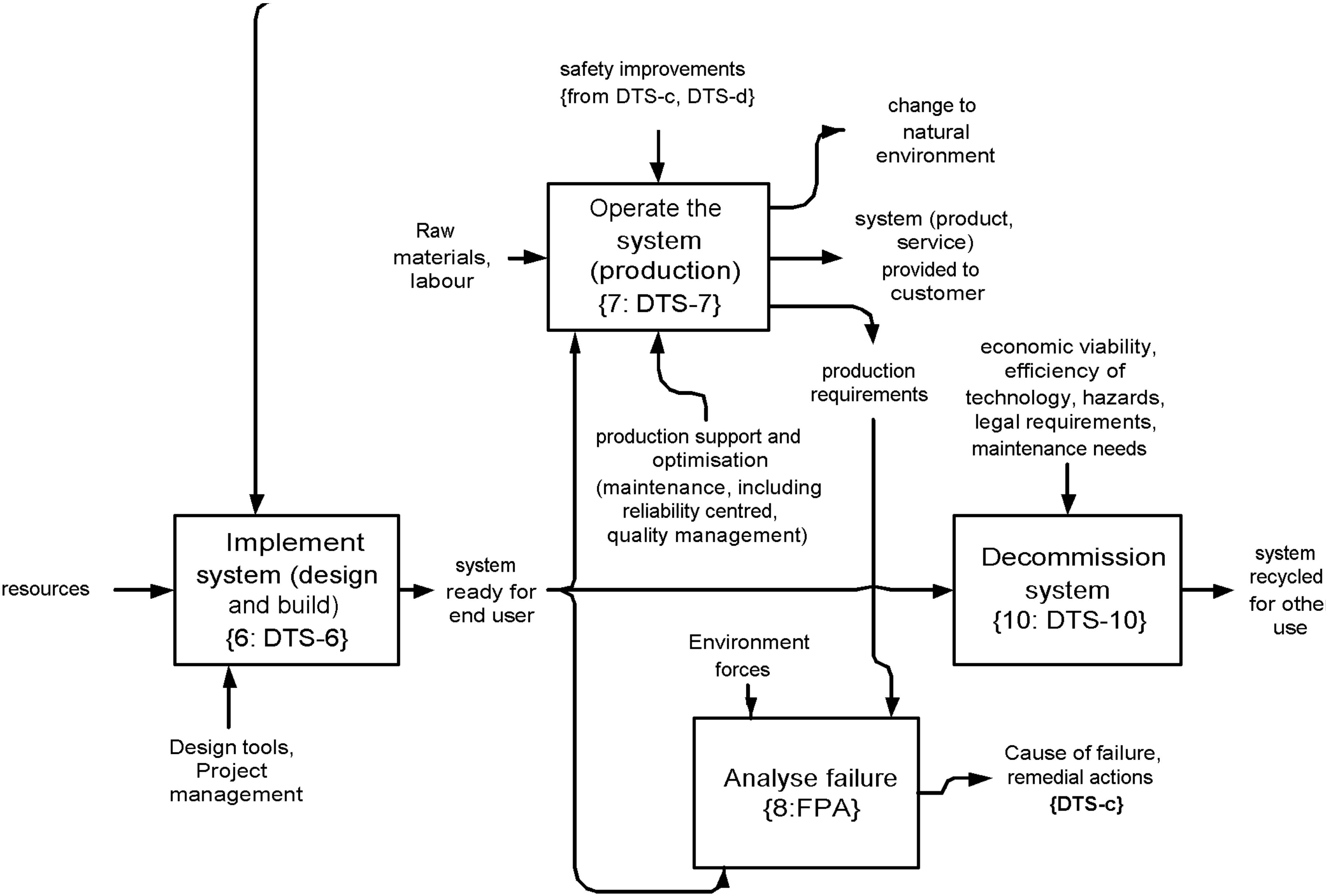
- Activities may be concurrent and at different stages of completeness.
- Can represent complex and uncertain relationships.
- Accommodates qualitative (textual) variables.

5 Results

Top view: Design, develop, and deploy technical system under uncertainty:

- determine the need {1}
- generation of candidate solutions {2}
- assessment of the system technology risk {3}
- simulation of system behaviour {4}
- decision is made on which concept to adopt {5}
- implementation occurs (design and build) {6}
- system operated {7}
- failures are analysed {8}
- individual behaviours {9}
- decommissioned {10}


Organisational leadershipcreates culture and shapes individual behaviours {9:DTS-9}
attitude towards uncertainty (random variables, design indecision, beliefs, epistemic uncertainty, uncertain outcomes) {DTS-b}
Predict system behaviour {4: DTS-4}
model of system {DTS-4-a} and anticipated system state s j {DTS-4-b}
Initiators: commercial, market pressure, identified need, strategic need
Determine the need {1: DTS-1}
criteria and scope of desired solution
Analyse technology risk (equipment and operators) {3: DTS-3}
identified system errors, estimate of the consequential damage, and quantification of the probability of those outcomes {DTS-d}
project decision {from DTS-e}
system errors {from DTS-d}
remedial design {from DTS-c, DTS-5-d}
culture {from DTS-b}
Constraints: technical, commercial, and schedule feasibility
Generate early design concepts {2: DTS-2}
candidate solution and context of system {DTS-a}
Make project decision under uncertainty {5: DTS-5}
project decision (approval, decline, remedial action, new concept) {DTS-e}



Design, develop, and deploy technical system under uncertainty {DTS}

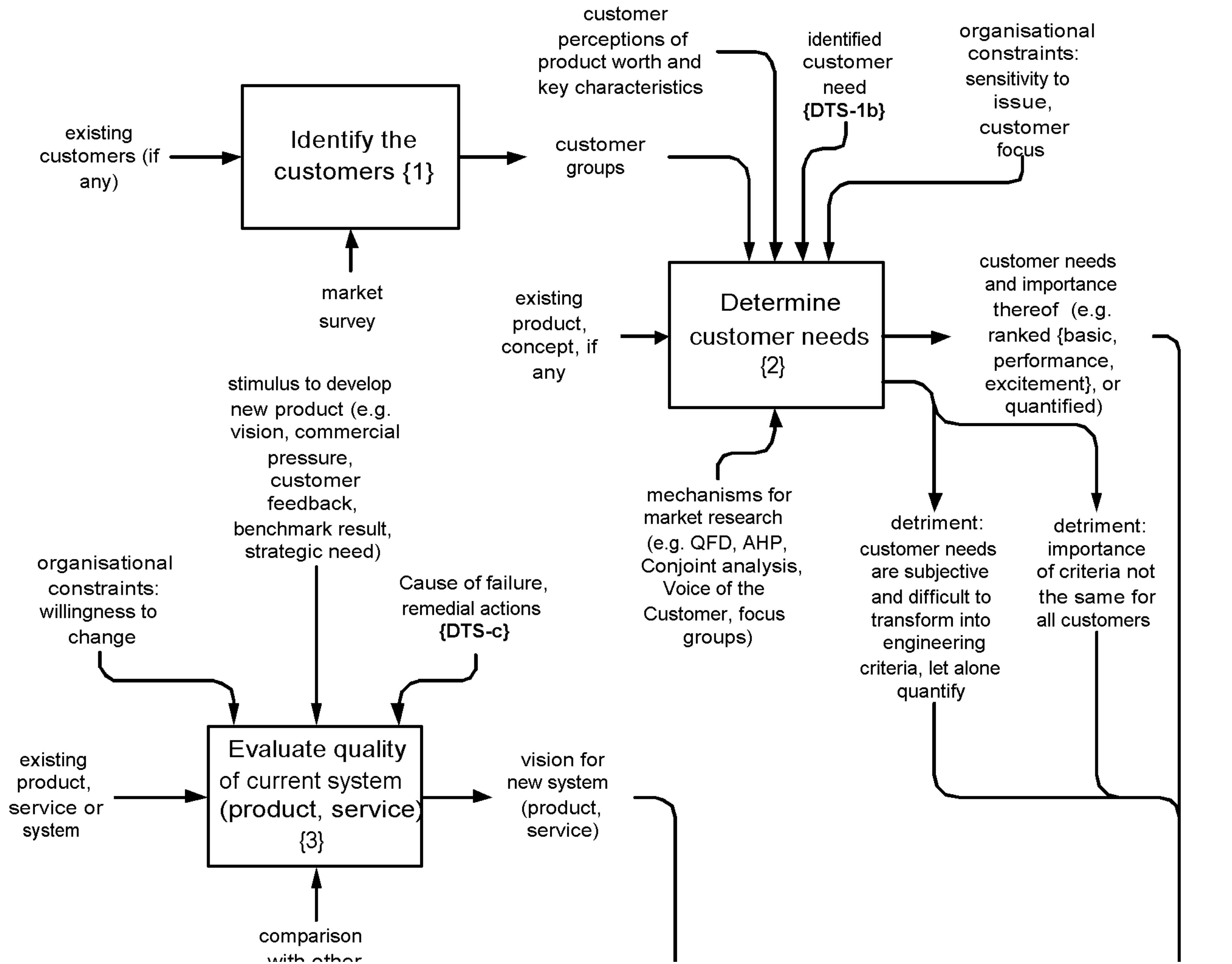
Determine the need {DTS-1}

- Determine customer needs {2}

  e.g. quality function deployment (QFD) (Bergman and Klefsjö, 1994; Martin et al, 1998), voice of the customer (Gustafsson, 1996), and analytical hierarchy process (AHP) (e.g. Perego and Rangone, 1996; Gustafsson, 1996), and focus groups

- Evaluate quality of current system {3}

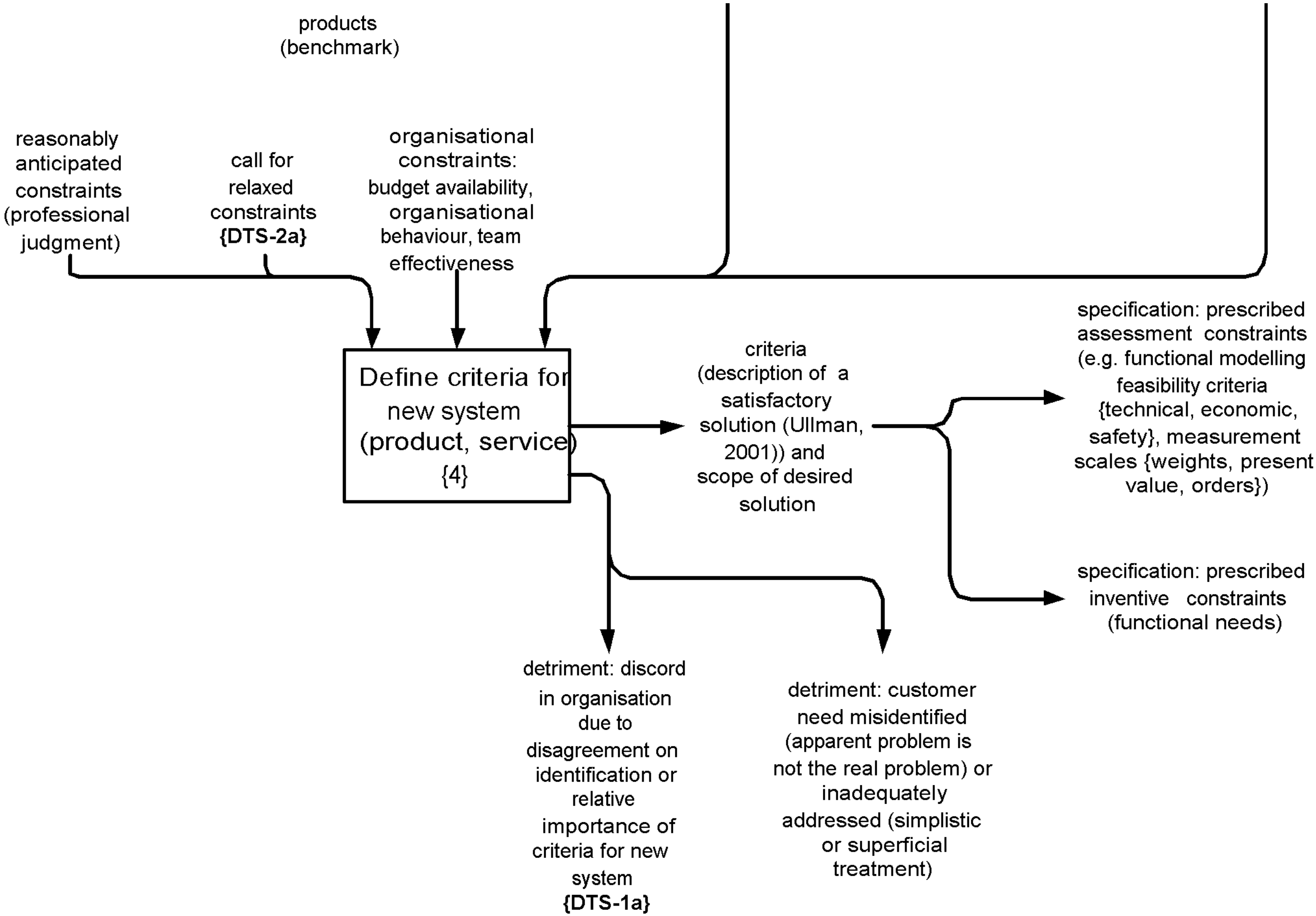
- Note: Detriments (failed processes) are readily represented…


existing customers (if any)
Identify the customers {1}
market survey
customer groups
customer perceptions of product worth and key characteristics
identified customer need {DTS-1b}
organisational constraints: sensitivity to issue, customer focus
existing product, concept, if any
Determine customer needs {2}
mechanisms for market research (e.g. QFD, AHP, Conjoint analysis, Voice of the Customer, focus groups)
customer needs and importance thereof (e.g. ranked {basic, performance, excitement}, or quantified)
detriment: customer needs are subjective and difficult to transform into engineering criteria, let alone quantify
detriment: importance of criteria not the same for all customers
stimulus to develop new product (e.g. vision, commercial pressure, customer feedback, benchmark result, strategic need)
organisational constraints: willingness to change
Cause of failure, remedial actions {DTS-c}
existing product, service or system
Evaluate quality of current system (product, service) {3}
vision for new system (product, service)
comparison

Determine the need {DTS-1}

Define criteria for new system {4}:

- Set criteria that describe a satisfactory solution. (Ullman, 2001)
- In reality, constraints may be partly qualitative and incomplete.
- They may also have different strengths or degrees of compulsion. (Owen, 1993)



Determine the need {DTS-1}

Generate early design concepts {DTS-2}

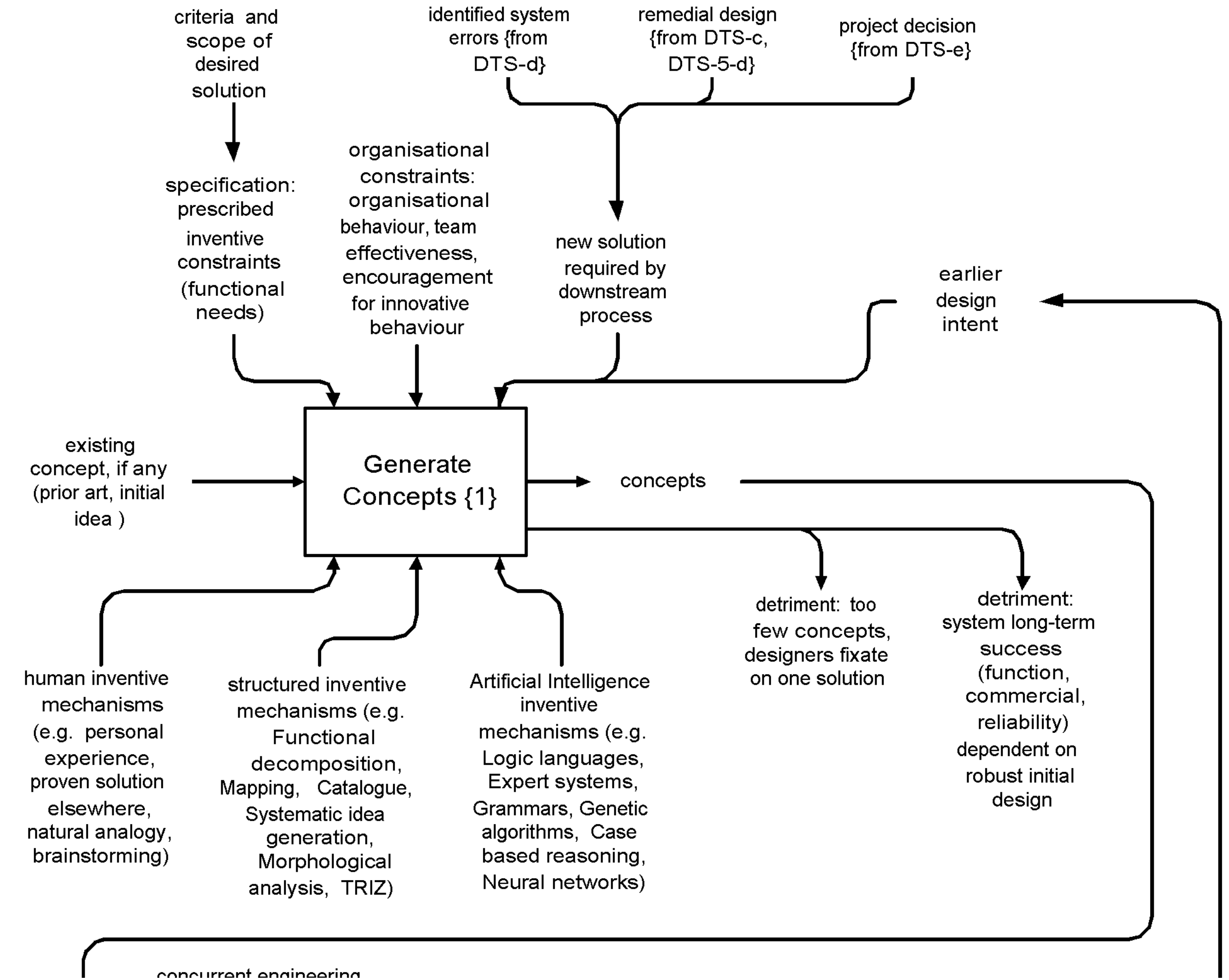
Generate concepts {1}:

- incremental improvement vs innovative
- variety of inventive mechanisms

e.g. personal experience, proven solution elsewhere, natural analogy, brainstorming, functional decomposition (Pahl and Beitz, 1988), mapping, catalogue (Kersten, 1996), systematic idea generation, morphological analysis (Hague et al, 1996), theory of inventive problem solving (TIPS/TRIZ) (Zlotin and Zusman, 1999), logic languages, expert systems (Cunningham and Smart, 1993), grammars (Andersson, 1994), genetic algorithms (Schmidt and Cagan, 1993), case based reasoning, and neural networks (Noguchi, 1998).

Generate early design concepts {DTS-2}

- Several concepts may be considered simultaneously and be in various positions within the design activity.
- Design is not set of discrete state transitions but as a system of multiple simultaneously active threads.


criteria and scope of desired solution
identified system errors {from DTS-d}
remedial design {from DTS-c, DTS-5-d}
project decision {from DTS-e}
specification: prescribed inventive constraints (functional needs)
organisational constraints: organisational behaviour, team effectiveness, encouragement for innovative behaviour
new solution required by downstream process
earlier design intent
existing concept, if any (prior art, initial idea)
Generate Concepts {1}
concepts
detriment: too few concepts, designers fixate on one solution
detriment: system long-term success (function, commercial, reliability) dependent on robust initial design
human inventive mechanisms (e.g. personal experience, proven solution elsewhere, natural analogy, brainstorming)
structured inventive mechanisms (e.g. Functional decomposition, Mapping, Catalogue, Systematic idea generation, Morphological analysis, TRIZ)
Artificial Intelligence inventive mechanisms (e.g. Logic languages, Expert systems, Grammars, Genetic algorithms, Case based reasoning, Neural networks)

Generate early design concepts {DTS-2}

Refine concepts {2}:

- Takes into account concurrent engineering constraints from elsewhere

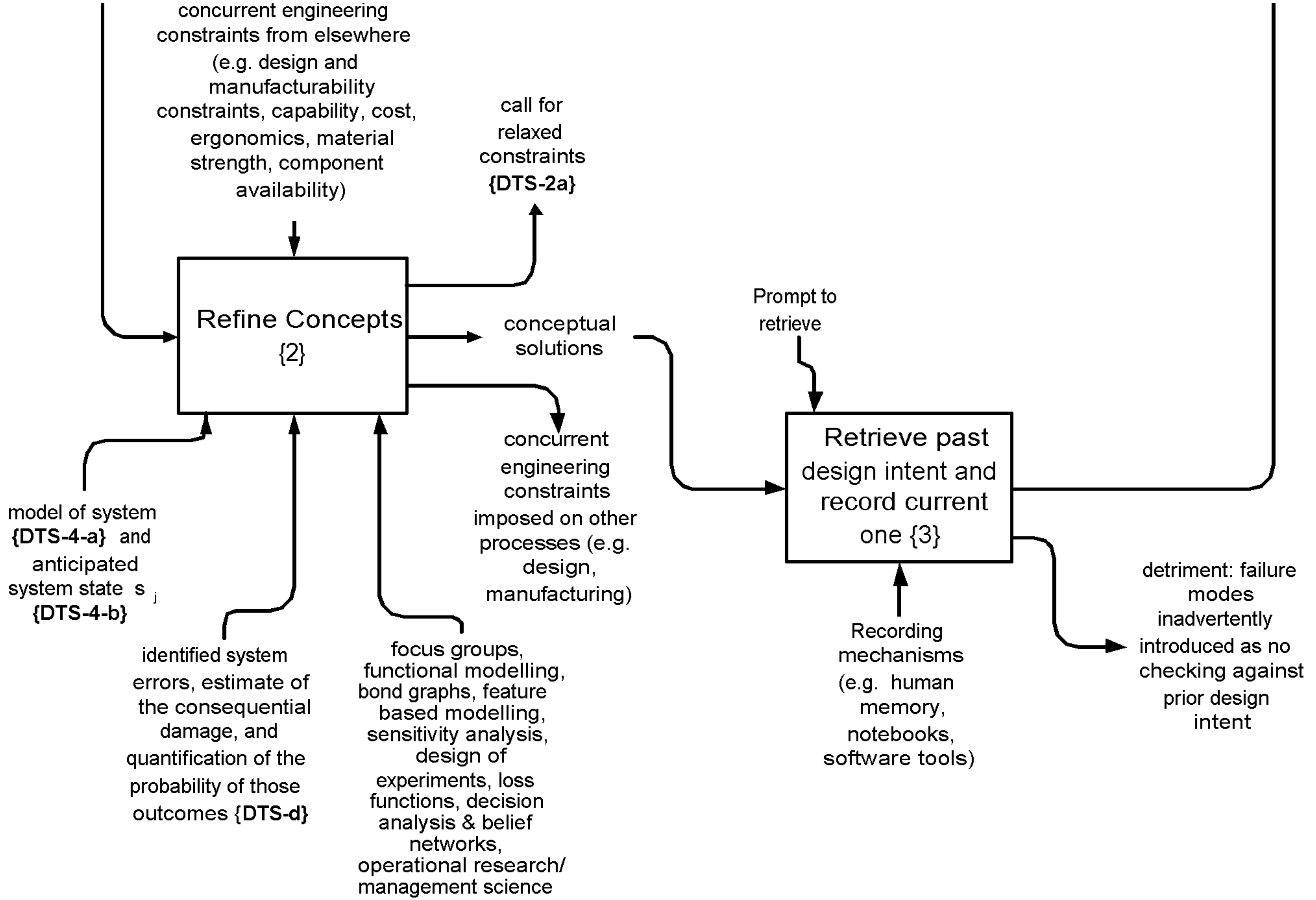
  e.g. design and manufacturability, production capability, cost, ergonomics, material strength, component availability

- Imposes concurrent engineering constraints on other processes

  overlapping concept in design structure matrix (Yassine and Falkenburg, 1999)

Retrieve past design intent and record current one {3}:

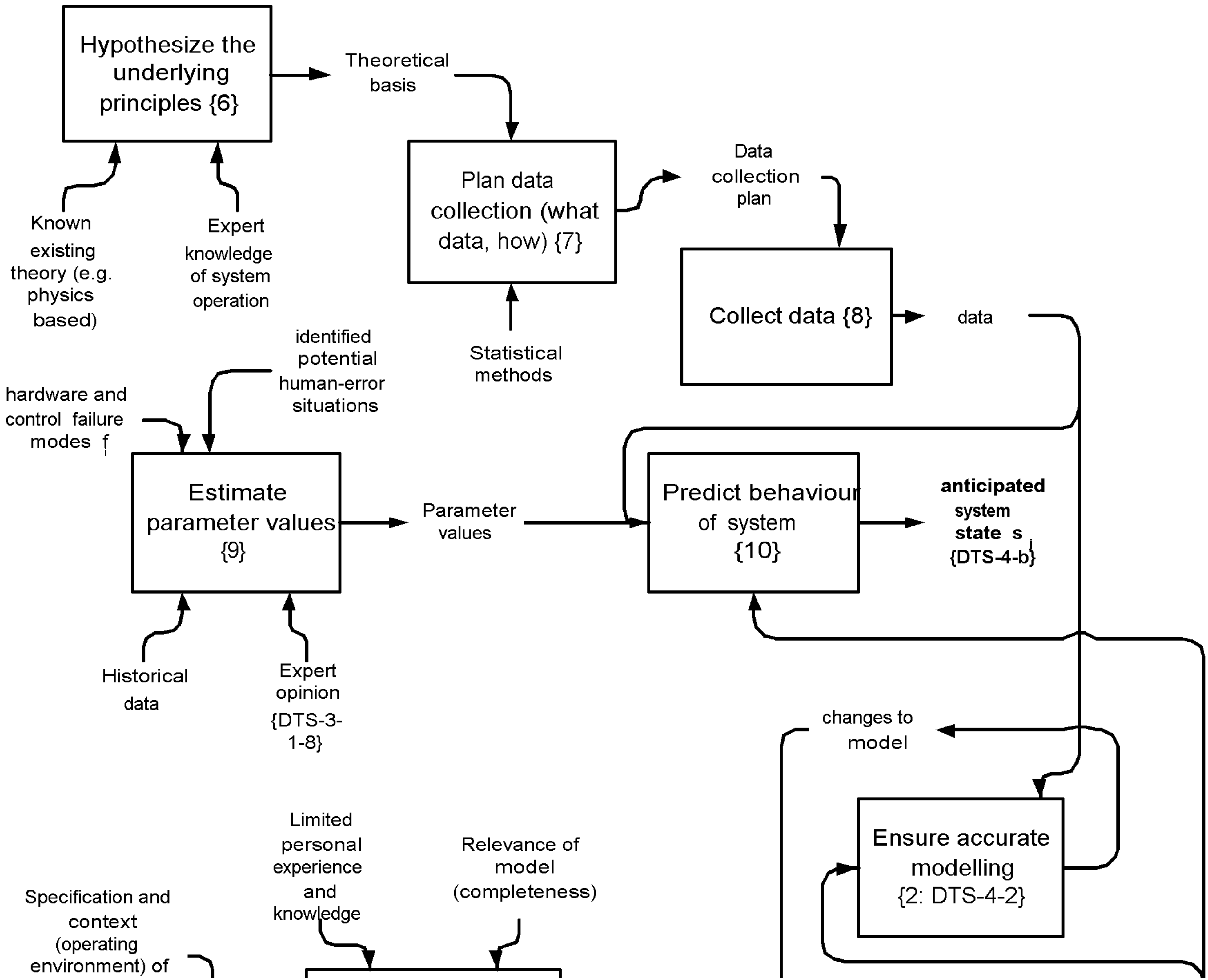
- Prevents the re-introduction of known failure modes.
- Few mechanisms – suggests an underdeveloped area of design.

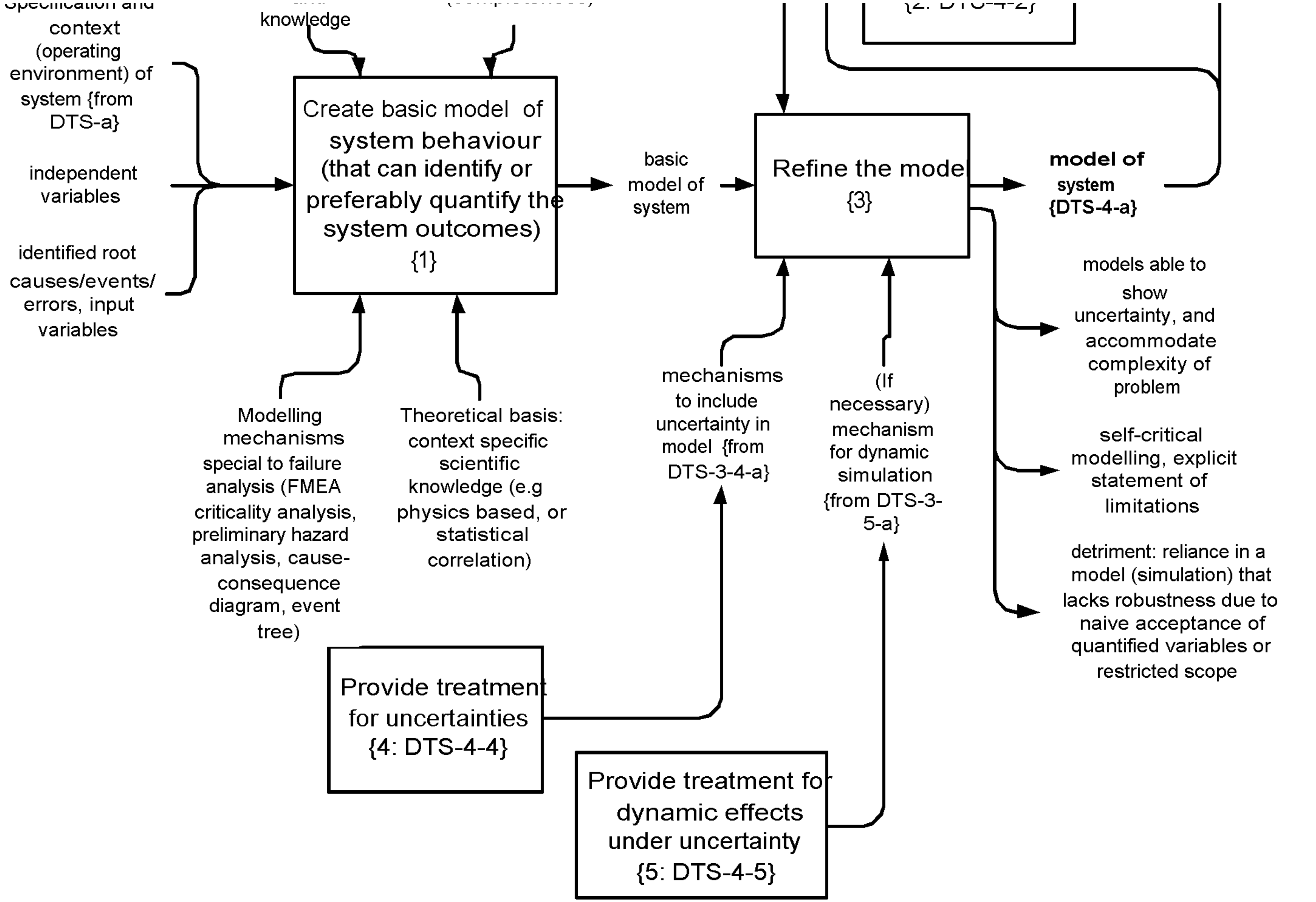


Generate early design concepts{DTS-2}

Predict system behaviour {DTS-4}

- Create a model of system behaviour {1}
- Refine it {3}
- Include treatment for uncertainties {4}
- Include dynamic effects {5}
- Ensure robust modelling {2}
- Collection of data {6, 7, 8, 9}
- Predict behaviour of the system {10}.


Hypothesize the underlying principles {6}
Theoretical basis
Known existing theory (e.g. physics based)
Expert knowledge of system operation
Plan data collection (what data, how) {7}
Data collection plan
Statistical methods
Collect data {8}
data
identified potential human-error situations
hardware and control failure modes f
Estimate parameter values {9}
Parameter values
Predict behaviour of system {10}
anticipated system state s
{DTS-4-b}
Historical data
Expert opinion {DTS-3-1-8}
changes to model
Ensure accurate modelling {2: DTS-4-2}
Limited personal experience and knowledge
Relevance of model (completeness)
Specification and context (operating environment) of

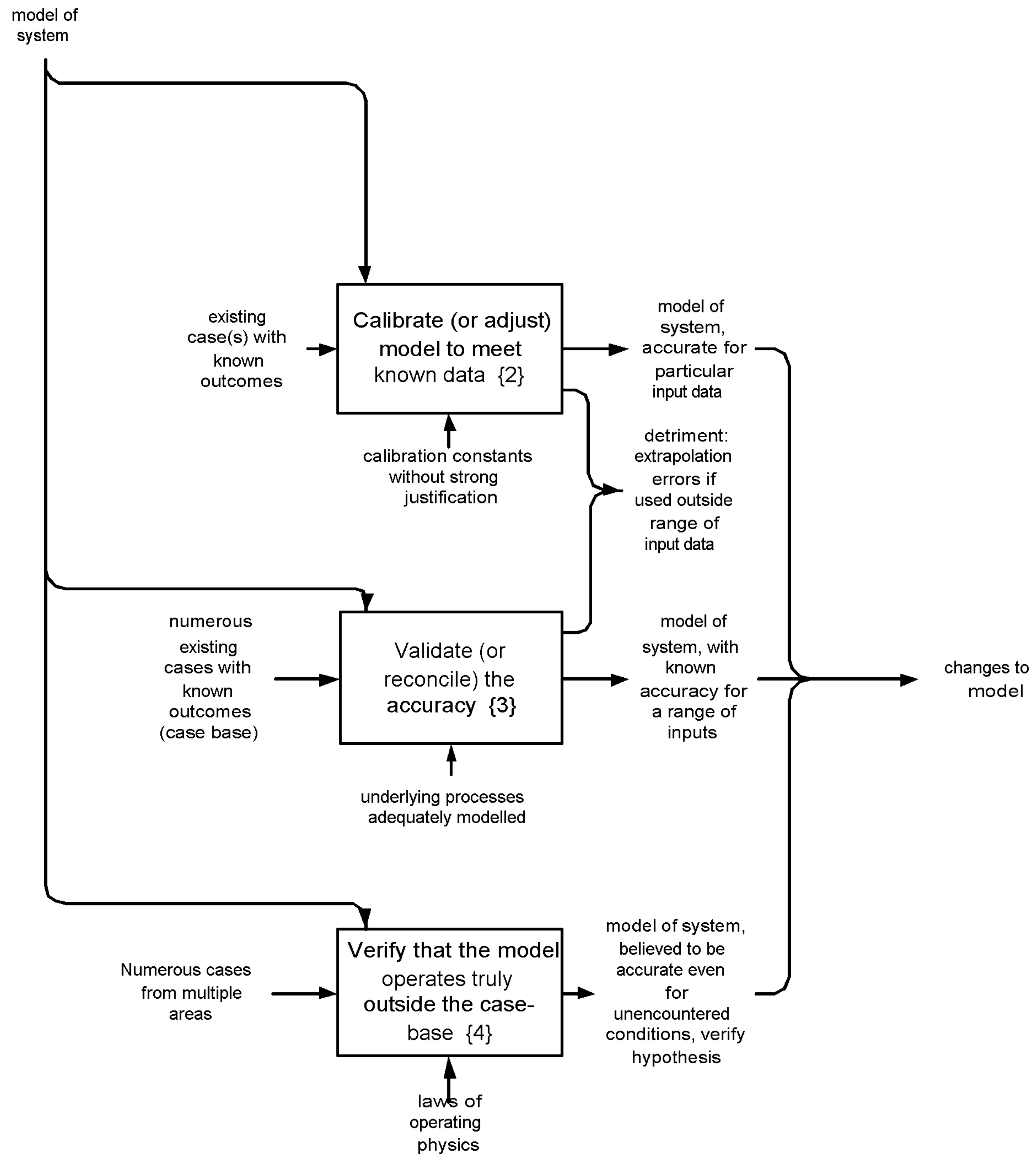


Predict system behaviour {DTS-4}

Ensuring accurate modelling {DTS-4-2}

Increasing levels of accuracy:

- Calibrate to known data {2}
- Validate for a range of inputs {3}
- Verify operation outside the case-base {4}.

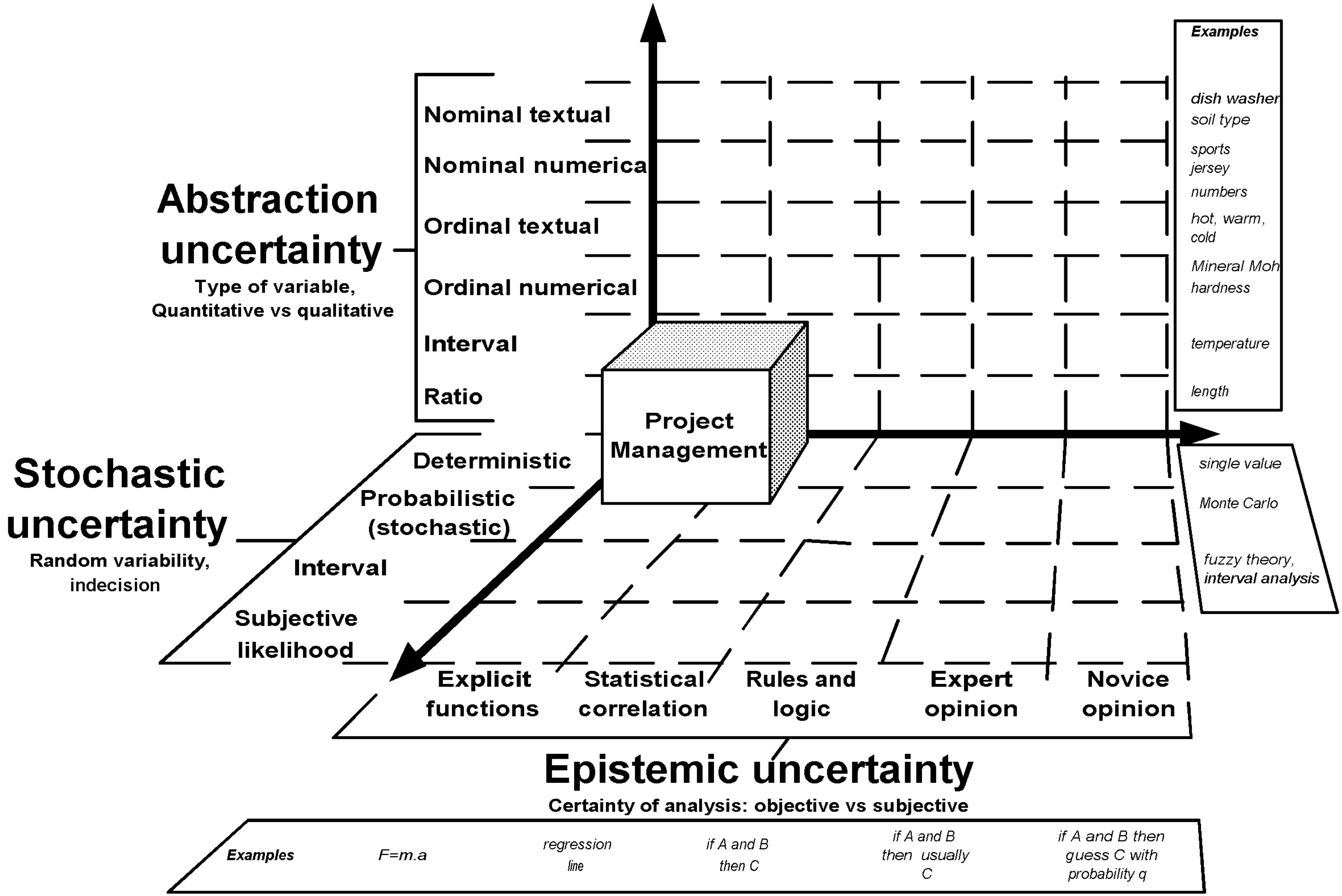


Ensure accurate modelling {DTS-4-2}

Provide treatment for uncertainties {DTS-4-4}

- Acknowledge the existence of uncertainty {1} – organisational and personality factors.

- Accommodate the three primary dimensions of uncertainty (stochastic, abstraction, and epistemic)…



Examples
dish washer
soil type
sports
jersey
numbers
hot, warm,
cold
Mineral Moh
hardness
temperature
length
Nominal textual
Nominal numerica
Ordinal textual
Ordinal numerical
Interval
Ratio
Abstraction
uncertainty
Type of variable,
Quantitative vs qualitative
Project
Management
Stochastic
uncertainty
Random variability,
indecision
Deterministic
Probabilistic
(stochastic)
Interval
Subjective
likelihood
single value
Monte Carlo
fuzzy theory,
interval analysis
Explicit
functions
Statistical
correlation
Rules and
logic
Expert
opinion
Novice
opinion
Epistemic uncertainty
Certainty of analysis: objective vs subjective
Examples
F=m.a
regression
line
if A and B
then C
if A and B
then usually
C
if A and B then
guess C with
probability q

- Accommodate stochastic uncertainty {2}

  e.g. single values (worst or plausible bounds, random what-if analysis, mean values), ranges (sensitivity analysis, moment methods, PERT, fuzzy theory), or full distributions (algebra of random variables, controlled interval method, Monte Carlo)

- Accommodate Abstraction {3}

  e.g. quantitative vs qualitative variables, assigning weights is problematic

- Accommodate epistemic uncertainty {4}

  e.g. explicit functions (axioms, mathematical equations), correlation (statistical regression), logic (boolean) & rules, expert opinion (conditional probability, decision theory, fuzzy theory), and novice opinion

- Modelling tools incapable of accommodating all three dimensions of uncertainty.
- Yet these uncertainties frequently arise in design studies.

Design for system integrity (DSI) methodology

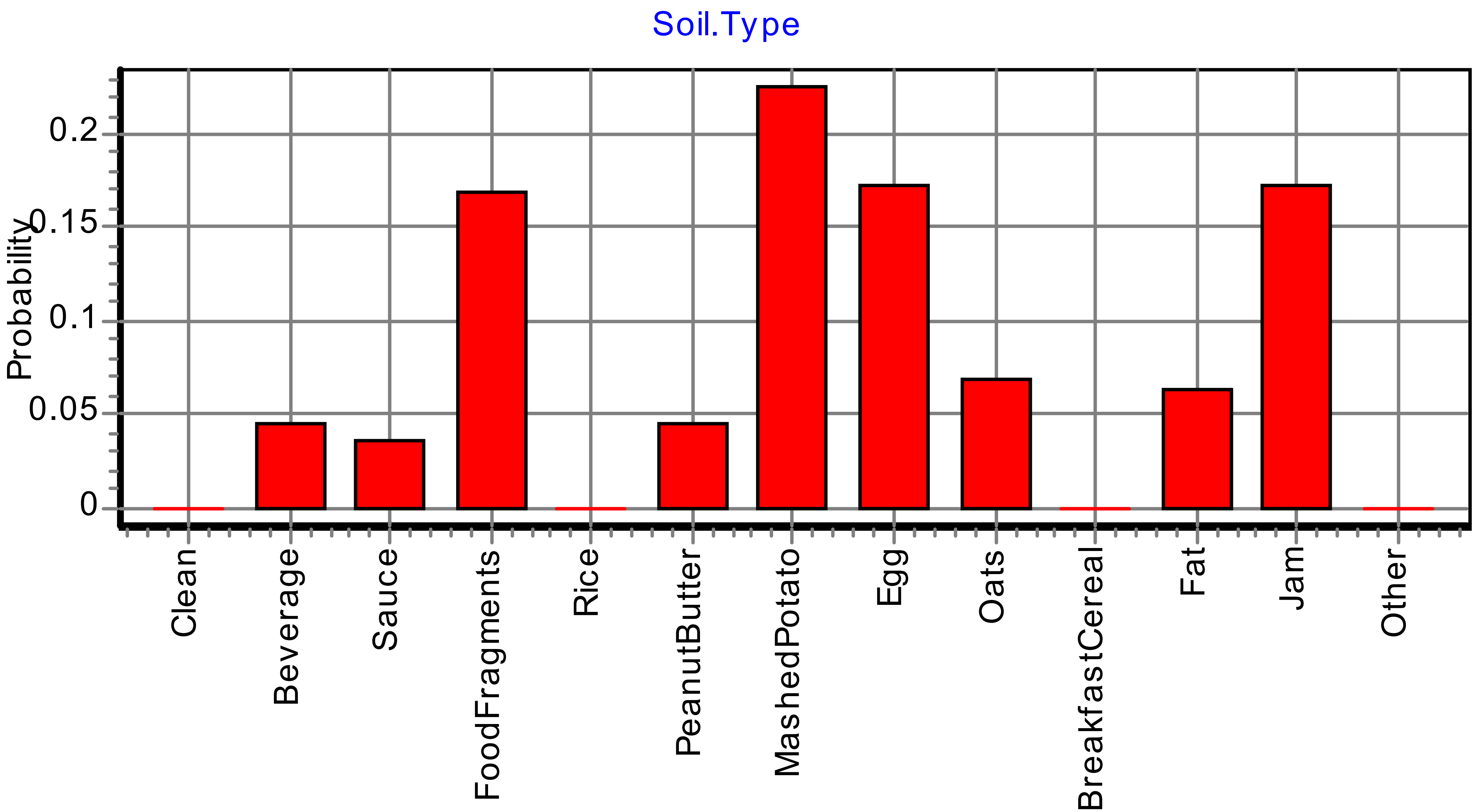
- DSI is a probability based approach.
- Prototype is implemented in software
  computationally demanding
- Accommodates all three forms of uncertainties in one model.
- Also includes catalogue features and multiple viewpoints

Support for the engineering design environment.

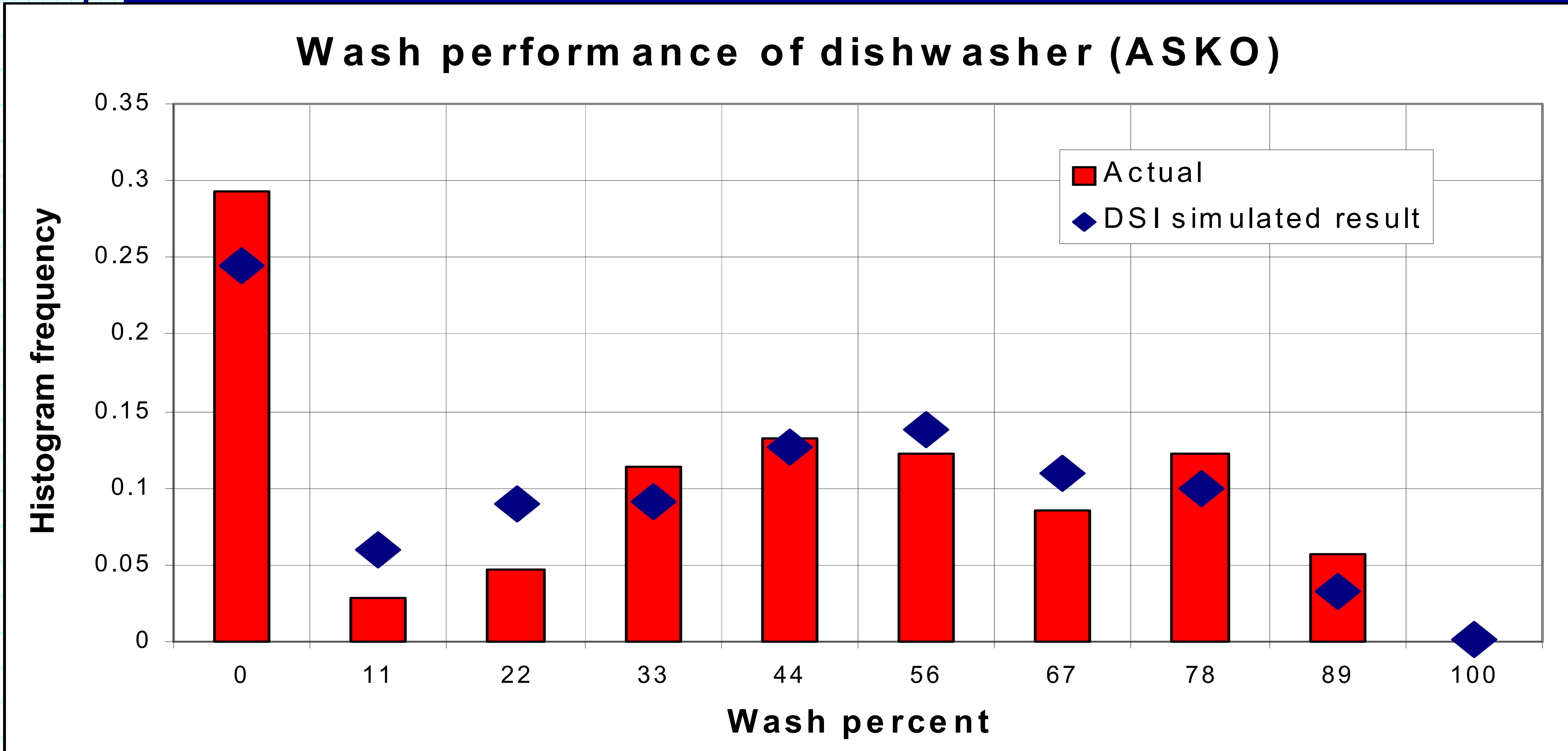
DSI example

Wash performance for domestic dishwashers:

- Given a description of the product configuration,
- And the soil load…


Soil.Type
Probability
0.2
0.15
0.1
0.05
0
Clean
Beverage
Sauce
FoodFragments
Rice
PeanutButter
MashedPotato
Egg
Oats
BreakfastCereal
Fat
Jam
Other

- Then predict the wash performance metric,
- And its variability…



Model of wash performance (points) compared to measured data (bars). Note similar spread, with some dishware washed well, others poorly. Consumer tests only report the average.

7 Conclusions

What has been achieved in the wash model?

- That qualitative variables carry important information for system modelling in design.
- That it is possible to include abstraction, stochastic and epistemic uncertainties in system models for design.

7 Conclusions

What has been achieved by the global model?

- An integrated design epistemology.

  Creation of a single conceptual framework that includes diverse bodies of design knowledge.

- Identification of process detriments.

  Provided as an outcome of systematic modelling approach.

- Inclusion of organisational and personality factors in design.

  Basic capability shown, with potential for extension.

End

Thank you for your interest.
Questions?

- Please see full paper for references and discussion on limitations of the model.
- Other areas of the larger model {DTS} are covered in other papers.