*CAS = Complex Adaptive Systems

Exploiting CAS* as a Force Multiplier - Its Application to Policy, Acquisition, Assessment and Operational Employment

A presentation to: Complex 07, Brisbane

Patrick Beautement
Systems Concepts and Assessments

03 July 2007

Agenda

- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

Background

- DRA / DERA:
  - Air / Land Agile C2 (Cellular automata / HiLoCa) with 32 AOG, Ramstein
  - 1998 - Study on exploiting emergence as force multiplier
  - 1999 - Uncertainty in command and control http://www.dodccrp.org/html4/events_past.html#1999
- With Institute of Human and Machine Cognition
  - 1999 - DARPA work on Coalition interoperability (CoAX)
    - autonomous agents / distributed systems in complex uncertain context http://www.aiai.ed.ac.uk/project/coax/index.html
- Santa-Fe Institute - ARCS (Adaptive and Resilient Computing Security):
  - 2002: http://discuss.santafe.edu/defense/agenda
  - 2003: http://discuss.santafe.edu/bnadaptive/
  - 2004: http://www.arcs-workshop.org/

Background

- QinetiQ:
  - Military workshops: Command Agility and Intelligence
  - 2004 - DoD Complex Systems / Agility in 'Edge Organisations'
  - 2006 - DSTO / DSTL collaboration on Complex System Engineering
- Some common themes / realities:
  - Humans are active problem-solvers, not dumb process-followers
  - Higher-level abstractions are used - hypotheses / 'abstract i2'
  - Boundaries (self / non-self) are blurred - interdependencies complex
  - Adversaries and the environment must not be 'simplified' away
  - 'Always-on' ***federations*** - can't be optimal, diversity is essential
  - Uncertainty, heterogeneity and the unexpected - are a certainty

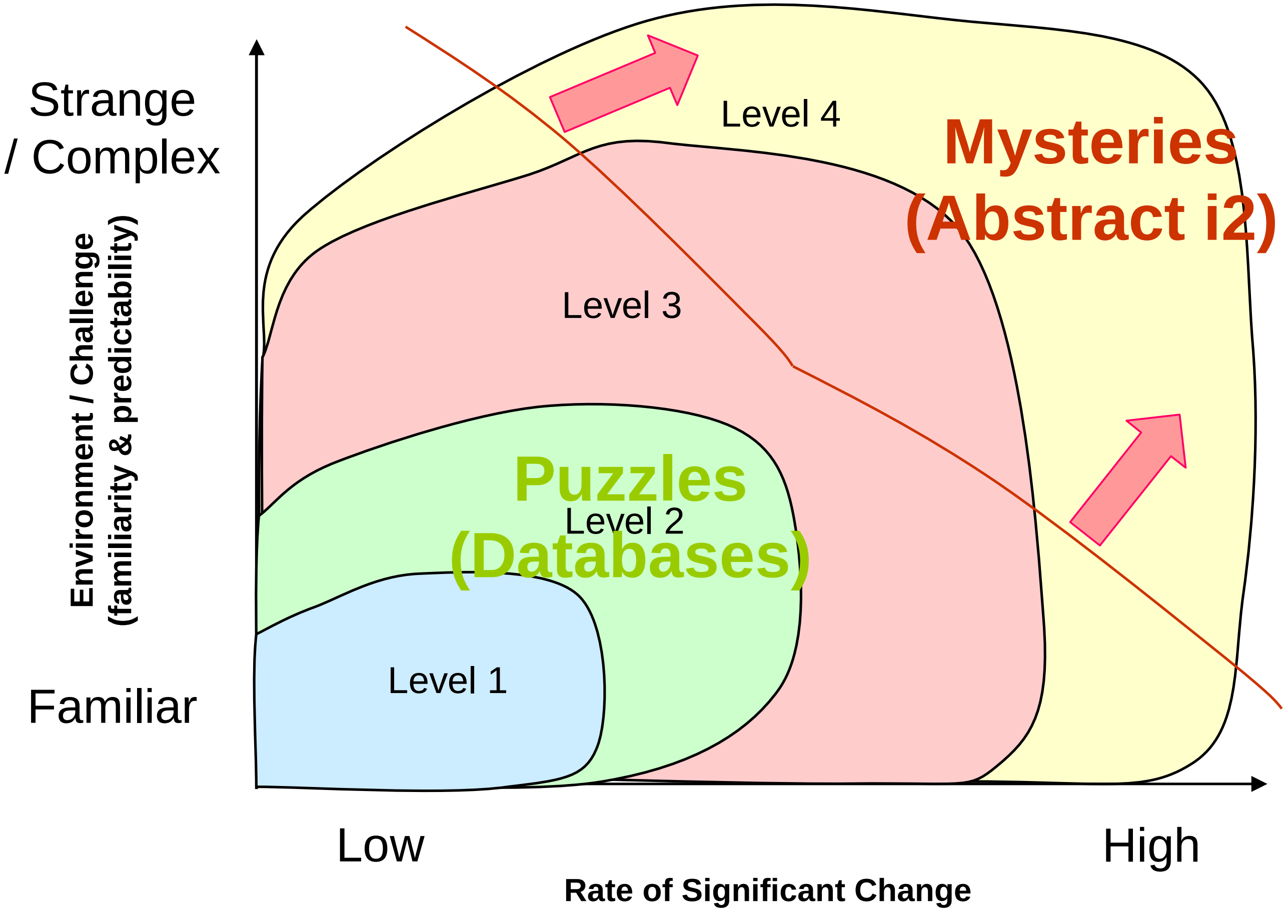
Agenda

- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

Operational context for CAS - Uncertainty

- We have trouble 'finding adversaries'- we look in our pictures and 'they' are not there. Why? Probably because they are:
  - Not on a defined battlefield - not where we expect them to be
  - Not constrained by notions of boundaries - they act wherever / whenever
  - Not part of a western-style 'fighting force' - asymmetric:
    - anything is a 'weapon' to be used to exert decisive influence
  - Not commanded 'from the centre' - instead, use resilient social networks
- From our point of view, they are probably:
  - Culturally 'strange' - different motivations, allegiances, values etc
  - Not necessarily part of 'them', 'out there' … they are "***Among the People***" ... and able to mimic apparently 'harmless' behaviours:
    - subvert our assets, mindsets, ways of working etc to their purpose

Level 3 / 4, abstract i2, is decisive in conflict and includes: hunches, intentions, hypotheses, weights of evidence, networks of relationships and other advanced abstractions that cannot be represented and manipulated as if they are facts



Any pre-defined 'sensor grid' is predicated on Level 1 / 2 and is useless against opponent who operates in the 'mysteries' region.

Level 4 - Bizarre - where ability to reason about non-things is vital. The data you don't have is significant etc. Hence is cognitive mind-game - you can't search databases for non-things!

Level 3 - eg Fallujah. Is as much about absence of signal as presence of signal.

Level 1 / 2 - Conventional Western warfighting. We can be structured - presumption is that things are easy to sense and identify, ie the 'object' has significance (not the background)

Starting Points - Agility and CAS

- To operate purposefully and effectively in a changing real world enterprises must be able to adapt, be agile:
  - Enterprises must be agile enough to generate novelty / deceive
  - Novelty cannot be defined a-priori, it must be generated at run-time
  - Enterprises use novelty to generate 'option spaces' / wiggle room
- Enterprises employ various aspects of CAS to:
  - Self-organise, regulate (autopoiesis), maintain / sustain (autonomic)
- What is self in CAS? Taking a perspective:
  - Human self: defined by cognitive, endocrine, immune system, possessions?
  - Systemic Self: System, System-of-systems, Federation
  - Acknowledging concurrent scales (time, extent of effect etc)

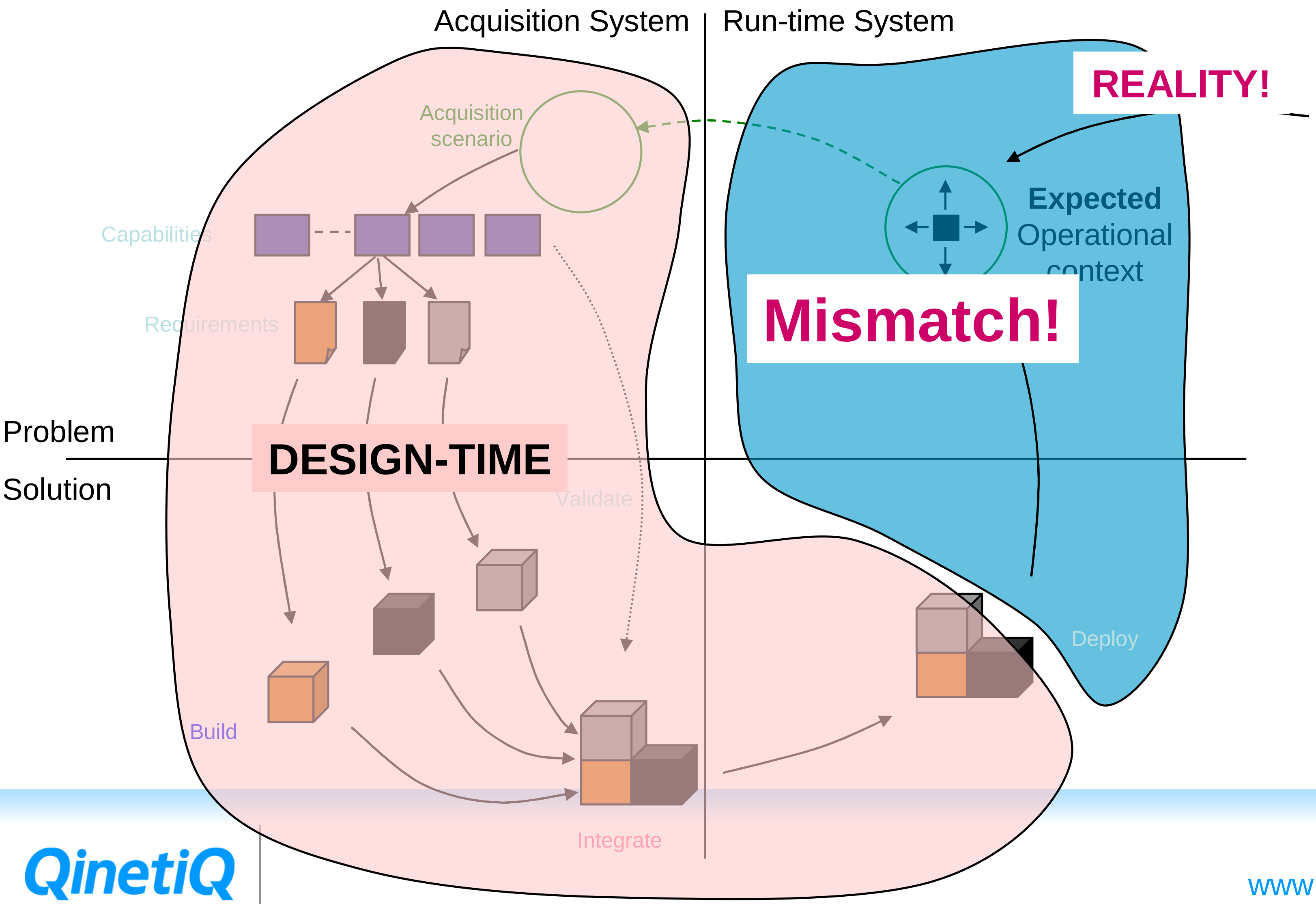
Exploiting CAS Purposefully

- Three aspects of influence:
  - Top-down - by the 'directing mind', shaping and tuning
  - Self-adjustment - through regulatory mechanisms
  - Bottom-up - emergence and spontaneous-organisation
- Factors to consider to enable 'influence-able enterprises':
  - 'Design-time' precursors that need to be in place
  - Assembling / 'growing' the necessary structures or letting them emerge
  - Opportunities and mechanisms available to be influenced
  - Run-time exploitation of the many levels at which interactions take place and the degrees of 'coupling' between these levels - federation dynamics
- How do these map to the wider environment?

Agenda

- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

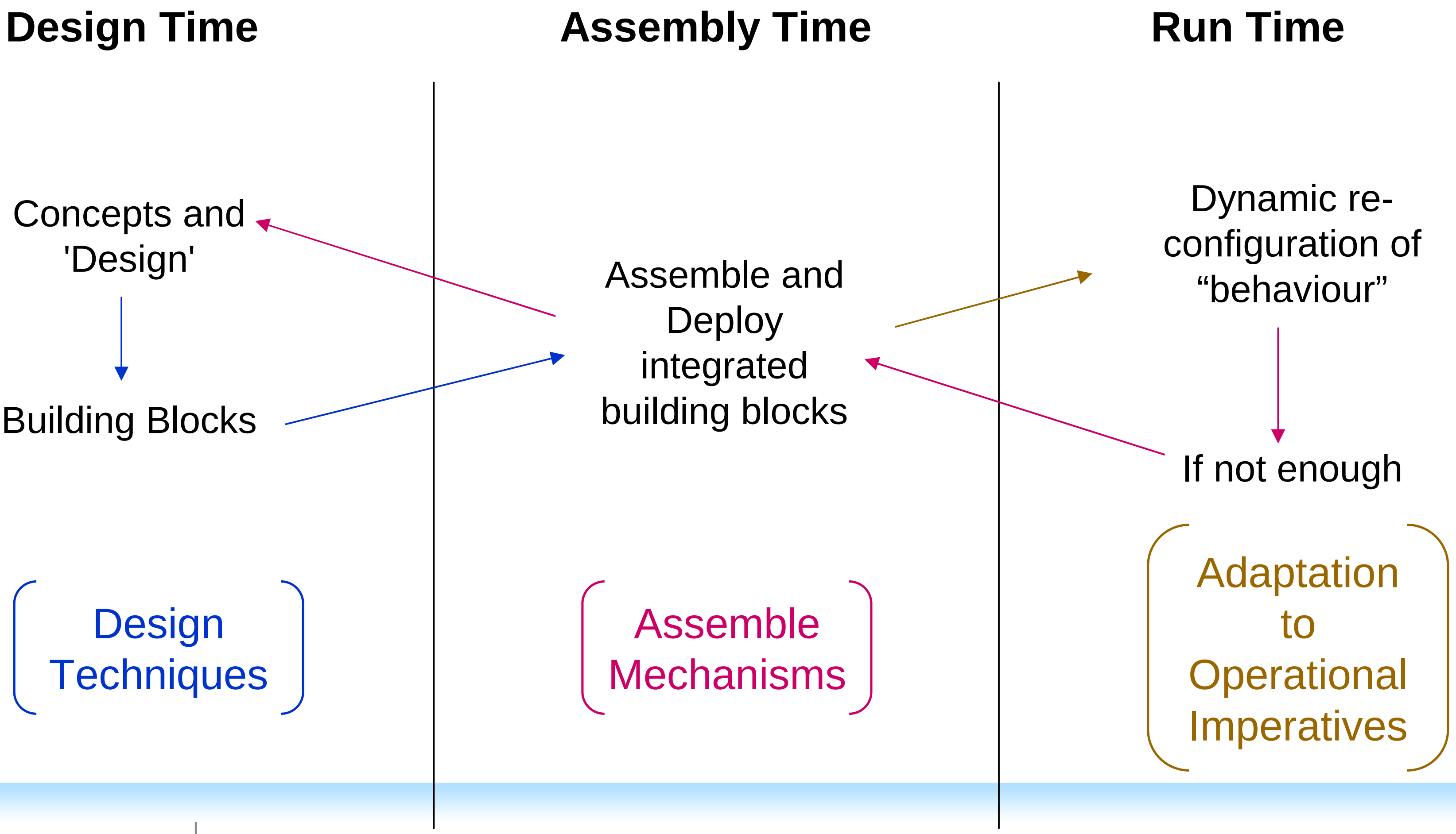
Industrial Age - How it Used to be (is!)



Agenda

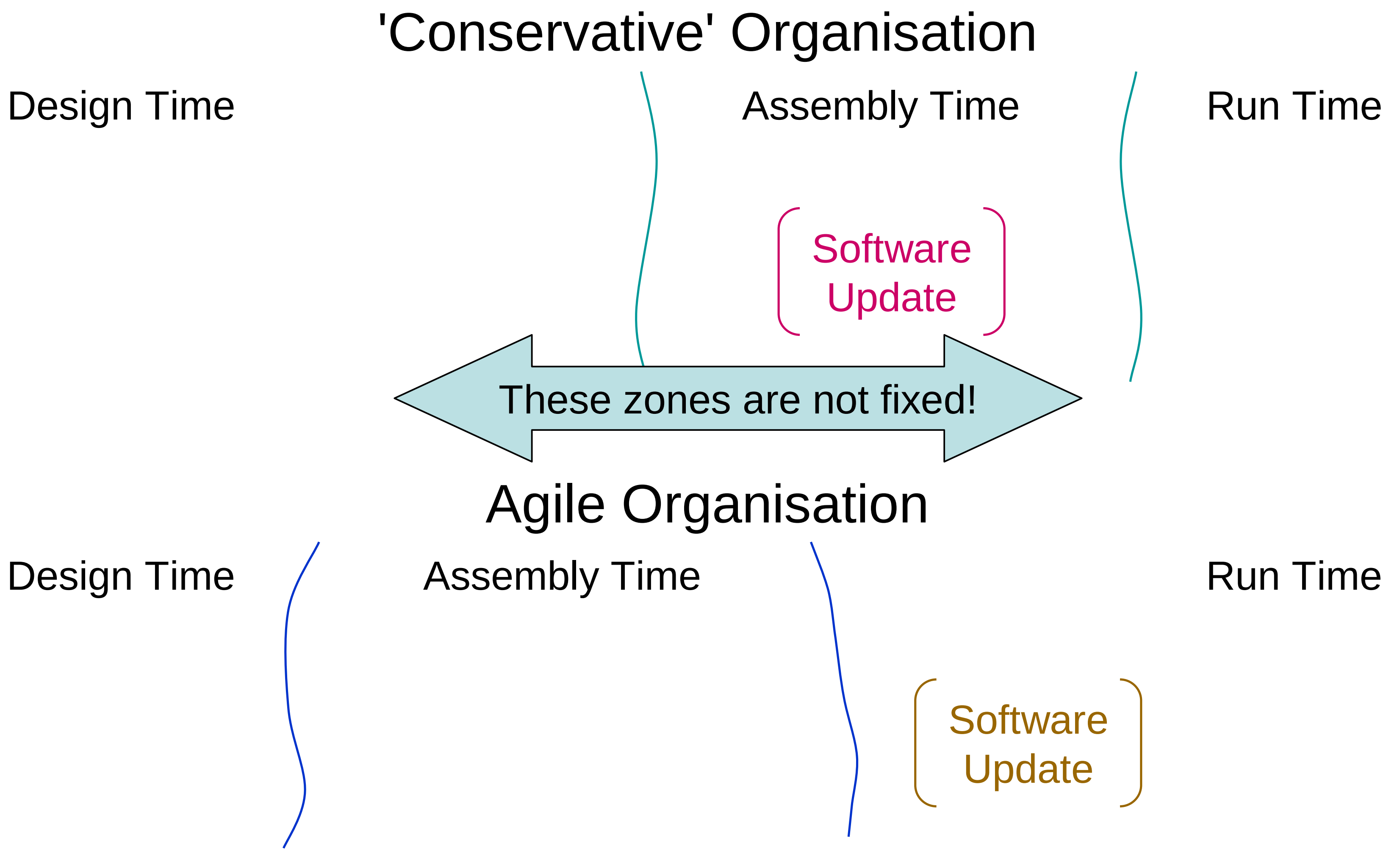
- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

Design, Assemble and Run-time (DART)

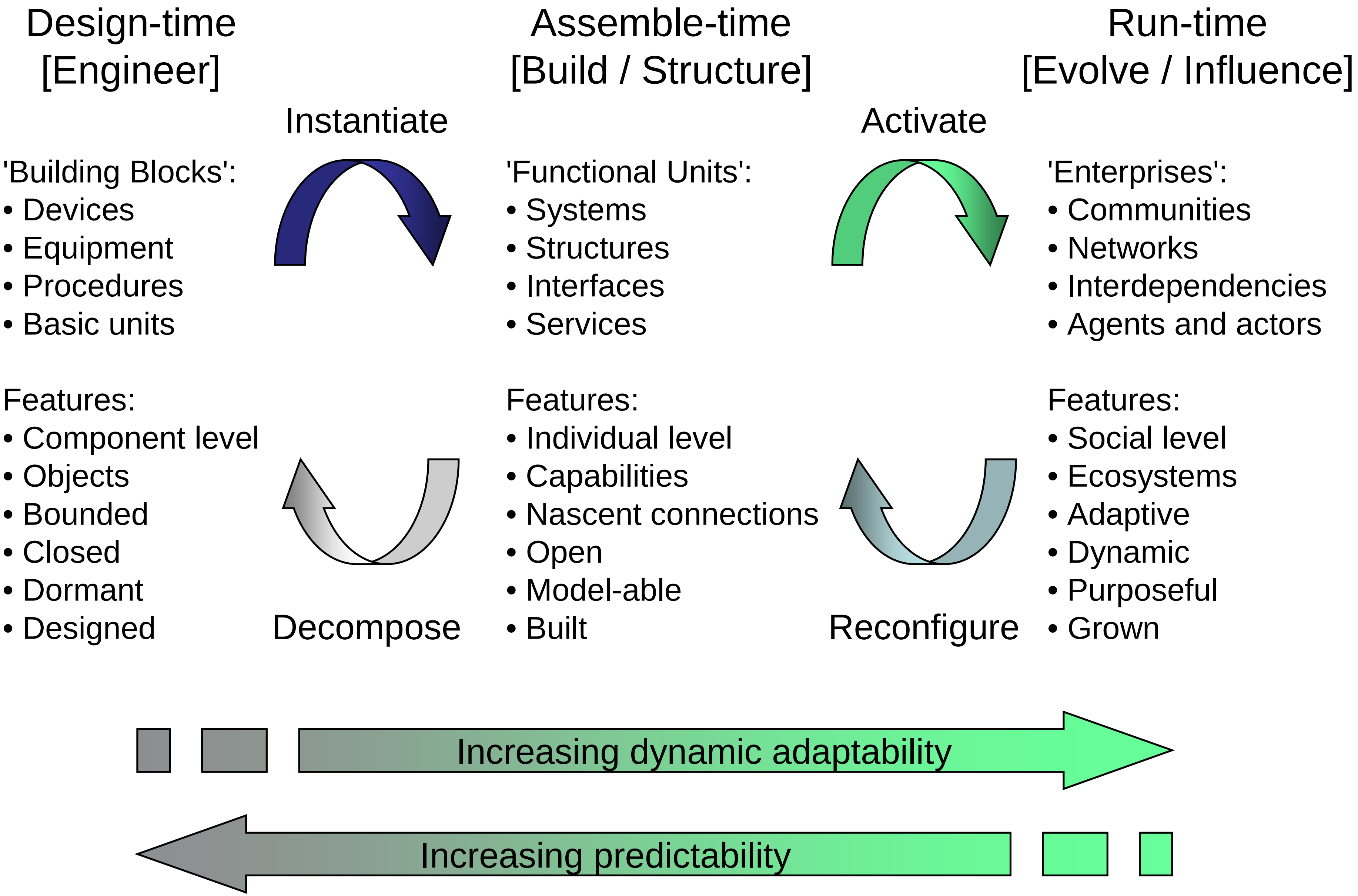


Design Time
Assembly Time
Run Time
Concepts and 'Design'
Building Blocks
Assemble and Deploy integrated building blocks
Dynamic re-configuration of “behaviour”
If not enough
Design Techniques
Assemble Mechanisms
Adaptation to Operational Imperatives

Design, Assemble and Run-time (DART)



DART Mechanisms



Design-time
[Engineer]

'Building Blocks':
- Devices
- Equipment
- Procedures
- Basic units

Features:
- Component level
- Objects
- Bounded
- Closed
- Dormant
- Designed

Instantiate

Decompose

Assemble-time
[Build / Structure]

'Functional Units':
- Systems
- Structures
- Interfaces
- Services

Features:
- Individual level
- Capabilities
- Nascent connections
- Open
- Model-able
- Built

Activate

Reconfigure

Run-time
[Evolve / Influence]

'Enterprises':
- Communities
- Networks
- Interdependencies
- Agents and actors

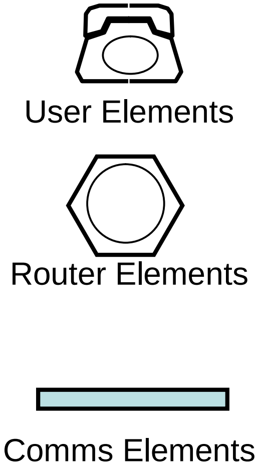
Features:
- Social level
- Ecosystems
- Adaptive
- Dynamic
- Purposeful
- Grown

Increasing dynamic adaptability

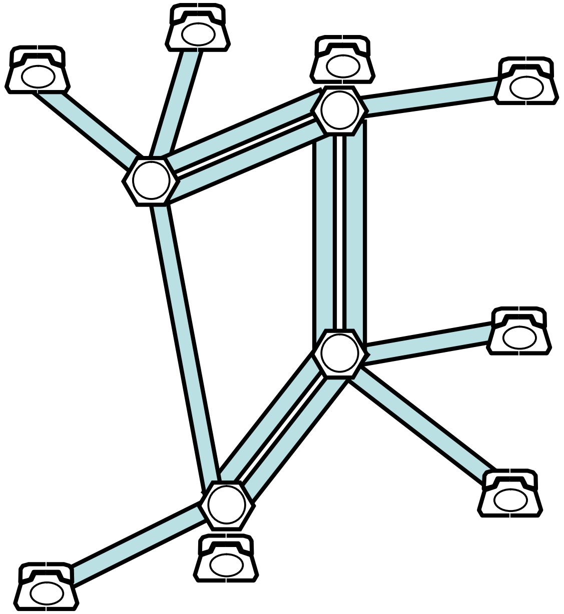
Increasing predictability

Comms Networks

Design Time



Assembly Time



Run Time

Primary Means

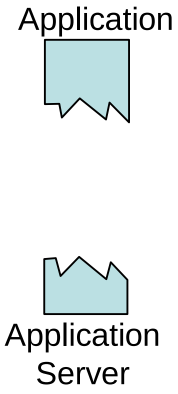
- Dynamic network management

Secondary means

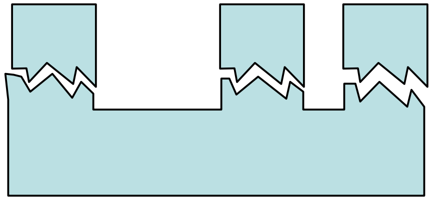
- Re-building (going back to assembly-time)

Applications and Services

Design Time



Assembly Time



Run Time

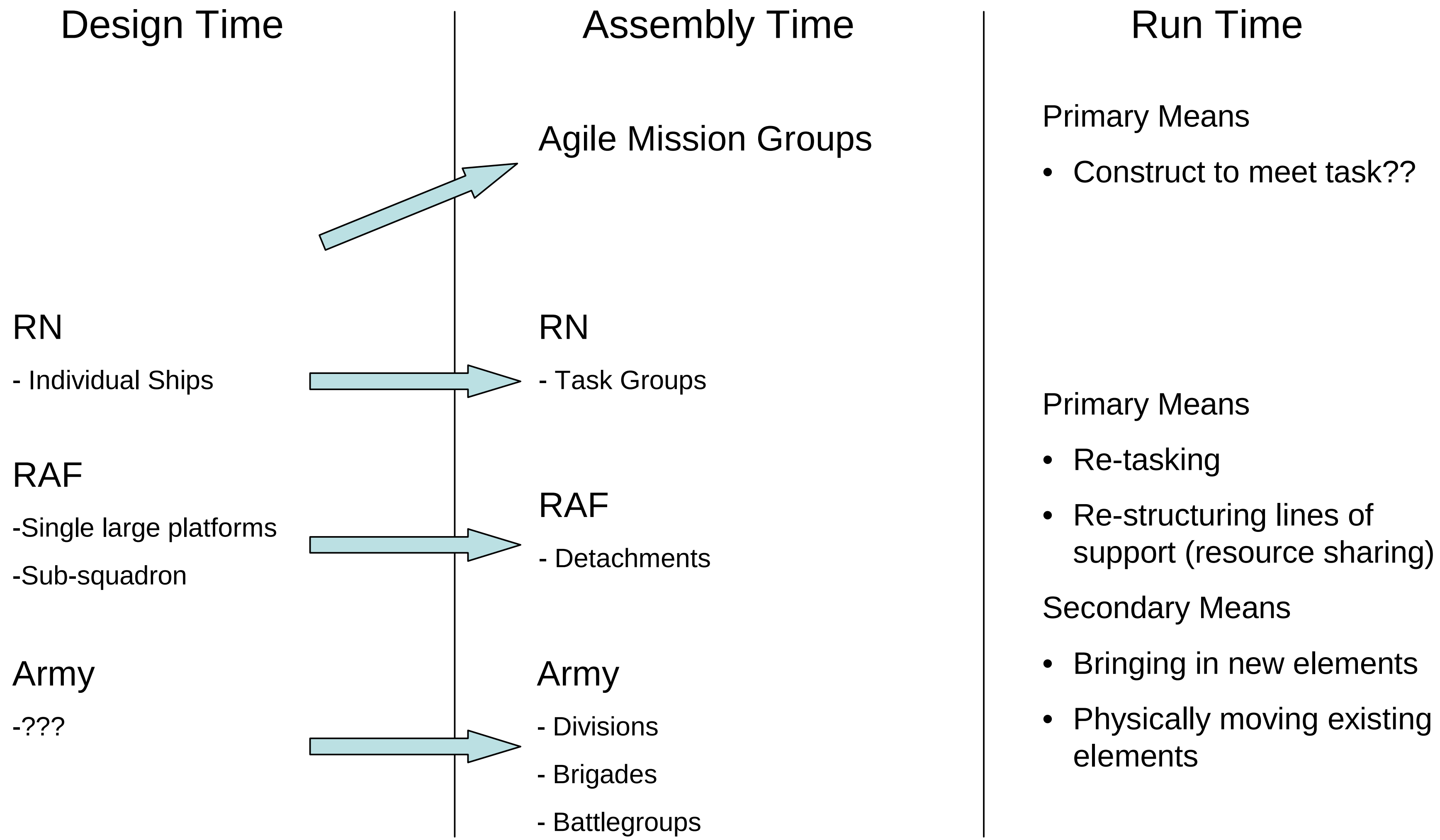
Primary Means

- Dynamic reconfiguration

Secondary means

- Plug and play (Plug and play needed for software upgrade more than agility)

Organisational Networks



Design Time

RN

- Individual Ships

RAF

-Single large platforms

-Sub-squadron

Army

-???

Assembly Time

Agile Mission Groups

RN

- Task Groups

RAF

- Detachments

Army

- Divisions

- Brigades

- Battlegroups

Run Time

Primary Means

- Construct to meet task??

Primary Means

- Re-tasking
- Re-structuring lines of support (resource sharing)

Secondary Means

- Bringing in new elements
- Physically moving existing elements

Social Networks

Design Time

- Trained commanders
- Liaison officers

Assembly Time

- Command networks/structure
- Communities of interest
- Informal Groups
- Lines of support

Run Time

Primary Means

- Dynamic COI’s
- Dynamic lines of support

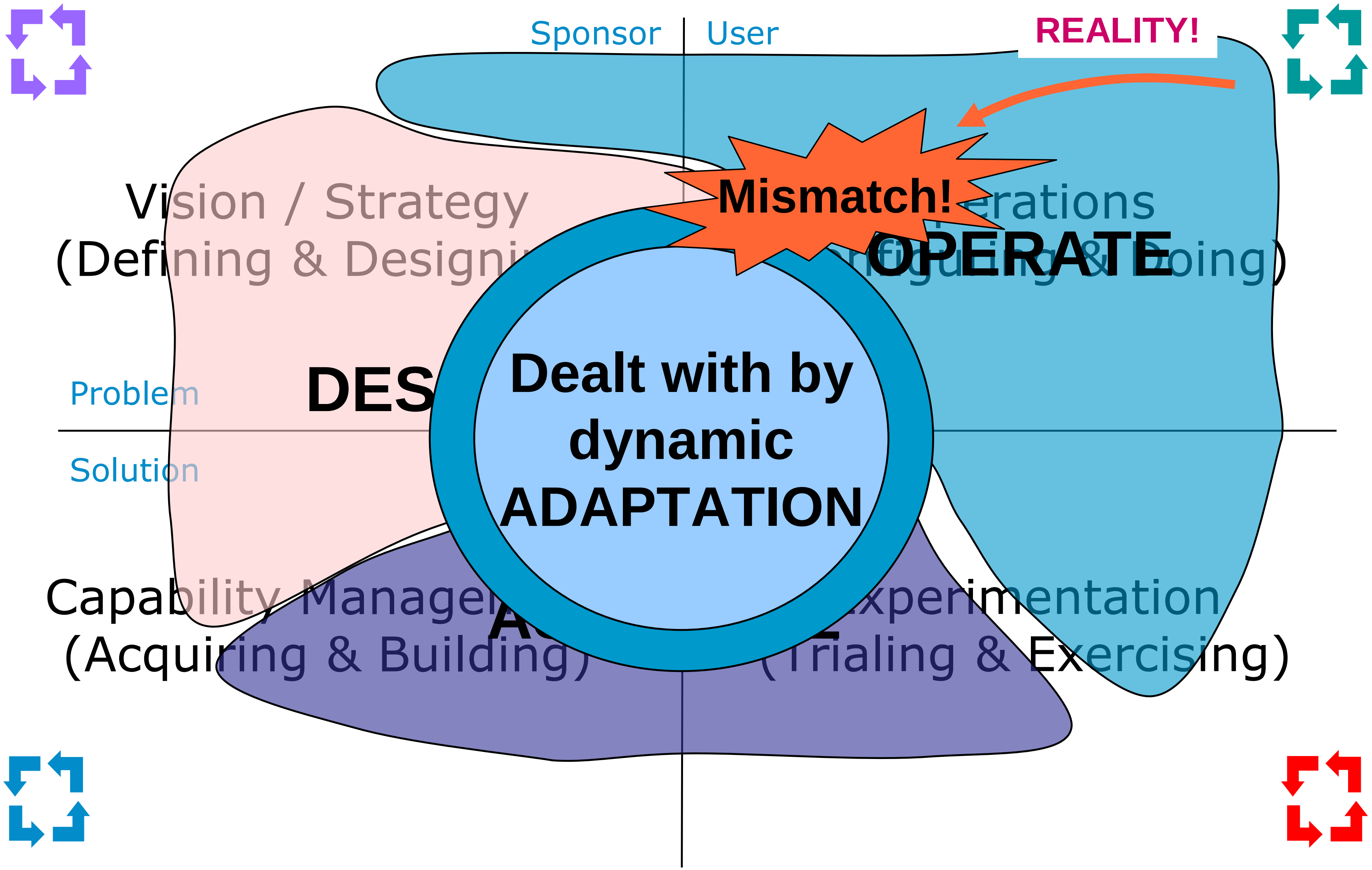
Secondary Means

- Change to command networks / structures

Agenda

- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

The DART Framework



Sponsor | User

REALITY!

Mismatch!

Vision / Strategy
(Defining & Designi

erations
nfiguring & Doing)

OPERATE

Problem

DES

Solution

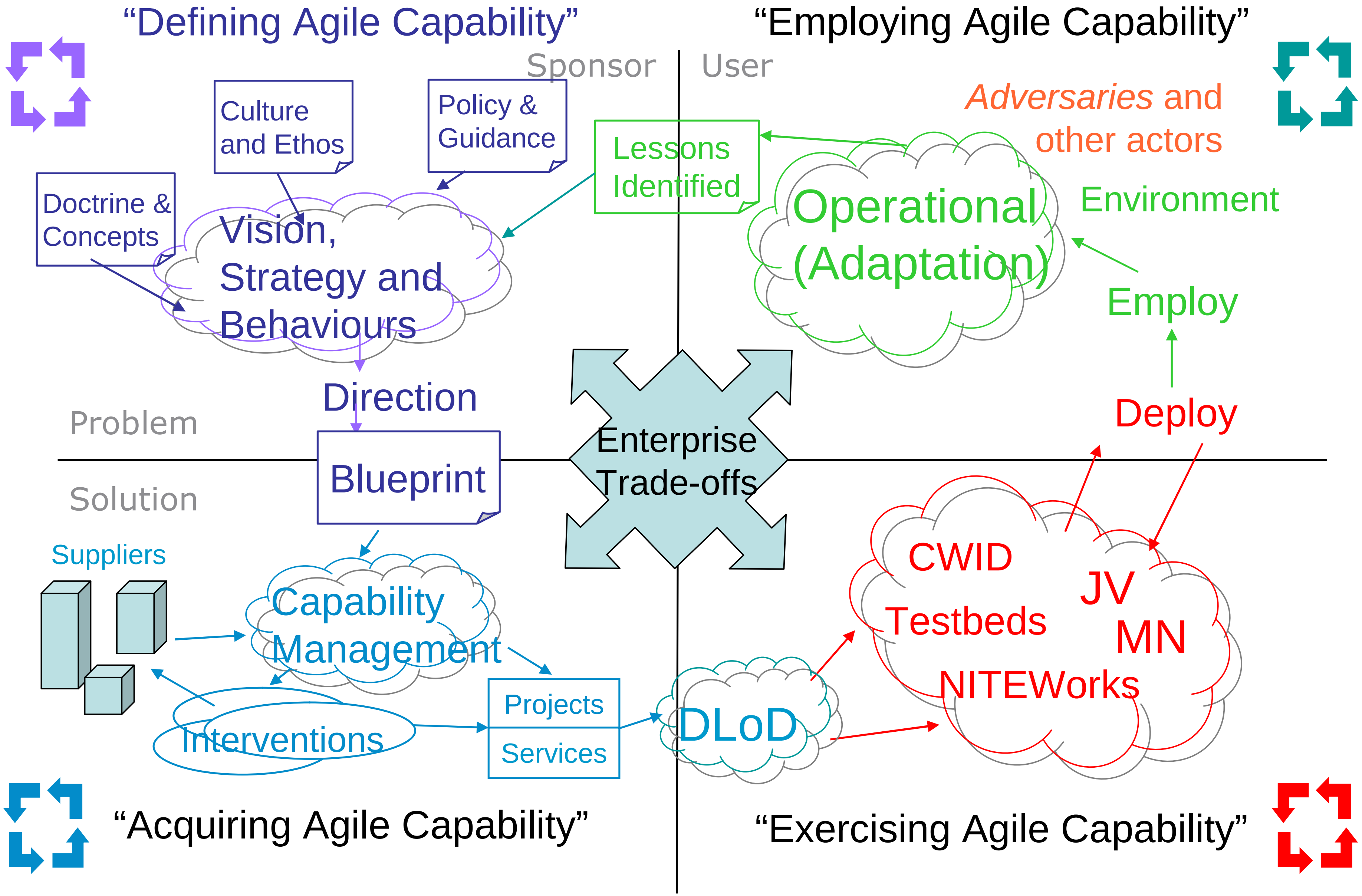
Dealt with by dynamic ADAPTATION

Capability Managem
(Acquiring & Building)

AC

xperimentation
(Trialing & Exercising)

The DART Framework



Exploiting DART - Vision / Strategy

- For ***Defining and Designing*** given the expected complexity:
  - Accommodating the range of possible future challenges
- Environment (Challenges, 'adversaries' / actors - what's success?):
  - Given competition among agencies - success is defending your power
- Appropriate approaches and mechanisms:
  - Design: From 'ethos', develop statements of policy and guidance
  - Assemble: Develop capability strategy and plans
  - Run-time: Use leadership and / or satisficing to hone policy / adjust plans
- Nature of outcomes:
  - Coherent and consistent guidance (blueprints) to acquisition
  - Provision of success values / metrics to experimentation
  - Provision of operational capabilities consistent with national aims

Exploiting DART - Capability Management

- For ***Acquiring and Building*** to meet the nature of complexity:
  - No single Project - managing diverse federations
- Environment (Challenges, 'adversaries' / actors - what's success?):
  - Given financial / risk stringency - success is delivery within time and cost
- Appropriate approaches and mechanisms:
  - Design: Identify federate characteristics across DLoDs*
  - Assemble: Build / acquire and configure capabilities in line with blueprint
  - Run-time: Through-life management support
- Nature of outcomes:
  - Provision of capability to support vision and strategy - national confidence
  - Interaction with experimentation - 'predator / prey' reduction of vulnerabilities
  - Trade-off project risks with operational agility benefits

* DLoDs = Defence Lines of Development

Exploiting DART - Experimentation

- For ***Trialling and Exercising*** and experiencing the complexity :
  - Complex, open-ended and unbounded experimental context
- Environment (Challenges, 'adversaries' / actors - what's success?):
  - Given need to be credible and useful - success is defended budgets
- Appropriate approaches and mechanisms:
  - Design: Define CAS-appropriate simulations / models experiments
  - Assemble: Configure federations of operational tools and simulations
  - Run-time: Experiments are plug and play in an 'always-on' environment
- Nature of outcomes:
  - Indicate policy constraints / inconsistencies, flawed assumptions, unreasonable expectations
  - Challenge and 'stress' acquisition / component providers / recruitment
  - Provision of adaptation-ready capabilities

Exploiting DART - 'Operations'

- For ***Configuring and Doing*** and adapting to the complexity:
  - Uncertainty, concurrency and wide scope of 'operational' environments
- Environment (Challenges, 'adversaries' / actors - what's success?):
  - Given intent and threat from adversaries - success is value-dependant
- Appropriate approaches and mechanisms:
  - Design: Conceive relationships, authorities, possibilities and constraints
  - Assemble: Form 'agile mission groupings' - be adaptation-ready
  - Run-time: Employ CAS interventions: eg, influence via tuning gradients, changing delegation of authority, force mix, environmental changes
- Nature of outcomes:
  - Robust operational outcomes in-line with national aspirations
  - Provision of credible insights to capability provision, doctrine etc
  - Hardened experimental context - the 'looser' learns the most

Agenda

- Background
- The Drivers - Uncertainty, Agility and CAS
- Current Approach to Provision of 'Capability'
- The Design, Assemble and Run-time (DART)
- Exploiting CAS within the DART Framework
- Summary / Way Ahead

Summary / Way Ahead

- Agility* is essential to enable operational adaptation:
  - The world is increasingly 'joined up' in federati[illegible]undaries between 'us' and 'them' are blurred - *we need optio[illegible]ptimisation*
  - 'Competition' is more diverse and d[illegible]d - the unexpected should be expected - *we need to actively [illegible] adaptive capabilities*
- We should increase our u[illegible]nding of the demands and benefits of agility and its systemi[illegible]isational aspects:
  - Characteris[illegible]n, Assemble, Run-time concepts and mechanisms
  - Ident[illegible]employ appropriate CAS toolsets / methods (mindset)
  - [illegible]e education in complex systems engineering
- [illegible]posefully employ DART to enable CAS to be exploited as a Force Multiplier (including deception, vulnerabilities and counter-CAS)

Complex system engineering enables federated 'adaptive enterprises'

* Agility includes: robustness, resilience, responsiveness, flexibility, innovation, adaptation.

Associates and Partners

- QinetiQ:
  - Anthony Alston, Patrick Beautement, Lorraine Dodd
  - Neil Briscombe / Mike Kirton - D3C (Dynamic, Dependable Distributed Computation)
  - David Allsopp - DIF DTC (Data and Information Fusion DTC)
- DoD: Use Strong Angel as a test case?
- DSTO: Anne-Marie Grisogono
- DSTL: Niki Jobson / Jim Moffat
- EU: Exystence Complex Systems network
- BT: Robert Ghanea-Hercock
- Institute of Human and Machine Cognition: Jeff Bradshaw
- Santa-Fe Institute: Business Network members
- Southampton University: Nick Jennings
- Warwick Business School: Yasmin Merali

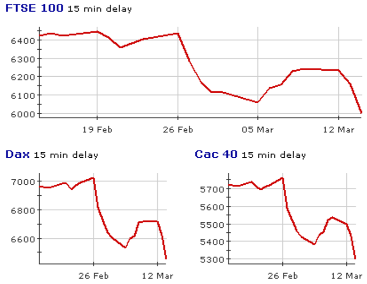
Questions?

Anthony Alston
Tel: +44 (0)1684 897416
Email: ajalston@qinetiq.com

Patrick Beautement
Tel: +44 (0)1684 896057
Email: pbeautement@qinetiq.com

Lorraine Dodd
Tel: +44 (0)1684 896135
Email: ldodd@qinetiq.com

AT119, QinetiQ Ltd,
Alan Turing Building,
Malvern Technology Park,
St Andrews Road,
Malvern, Worcs. WR14 3PS, UK.



QinetiQ

The Global Defence and Security Experts