Raytheon

*Customer Success Is Our Mission*

# Strategies for Competition beyond Open Architecture (OA): Acquisition at the edge of chaos

**NIRAJ SRIVASTAVA, PhD.**
**Sr. Manager**
**Advanced Concepts & Technology**
**Space and Airborne Systems**
**Raytheon Company**
**+1 202-904-2181 (office)**
**+1 310-469-8562 (cell)**
**Niraj.Srivastava@raytheon.com**

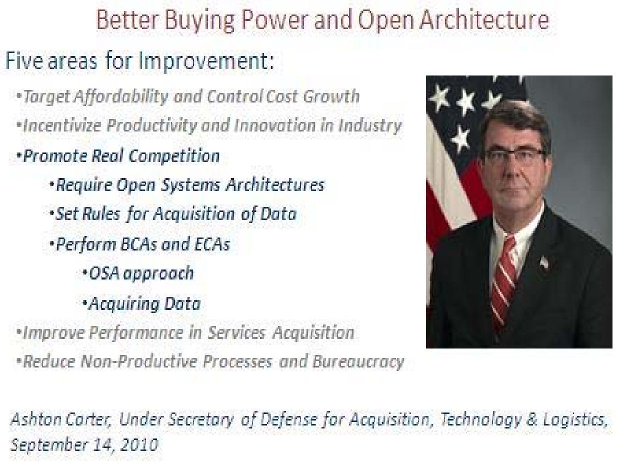
**MICHAEL L. RICE**
**President, R2E Inc.**
**2004 Victoria Drive, Stafford, VA 22554,**
**540-720-0454,**
**mrice.r2einc@verizon.net**

# NPS Study:
# "Acquisition at the edge of chaos"

Raytheon
Space and Airborne Systems

## Acquisition Issues

- **Threats: more complex and agile**
  - Shorter lifetime and increased uncertainty
  - Adversary evolution has accelerated
  - Sudden shifts in behavior arising from small changes
- **System:  cost and complexity growing**
  - Weapon Systems Acquisition Reform Act of 2009, dictates measures to ensure competition for better life cycle pricing.
  - Systems Engineering perspective
    - MOSA/OA principles have produced some good results – RDT&E,
    - Post deployment competition cost savings has not materialized.



Better Buying Power and Open Architecture

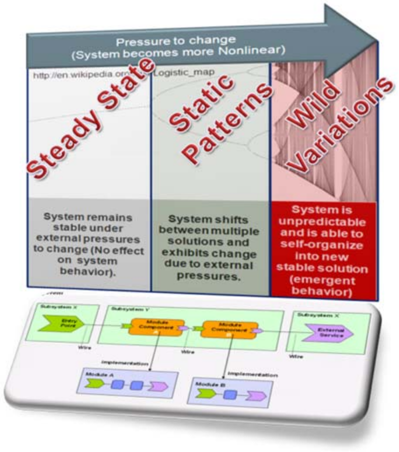
Five areas for Improvement:

- *Target Affordability and Control Cost Growth*
- *Incentivize Productivity and Innovation in Industry*
- *Promote Real Competition*
  - *Require Open Systems Architectures*
  - *Set Rules for Acquisition of Data*
  - *Perform BCAs and ECAs*
    - *OSA approach*
    - *Acquiring Data*
- *Improve Performance in Services Acquisition*
- *Reduce Non-Productive Processes and Bureaucracy*

*Ashton Carter, Under Secretary of Defense for Acquisition, Technology & Logistics, September 14, 2010*

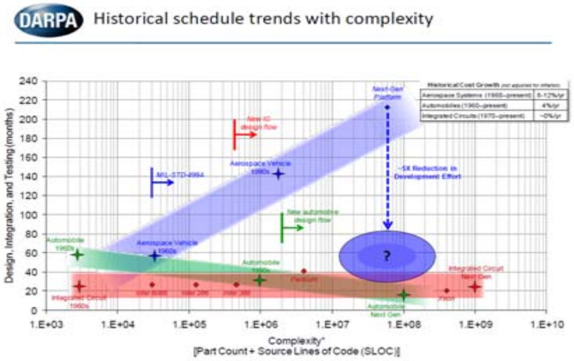
## CCRL Approach

- **System engineering practices have become outdated –**
  - Need to transition from steady state and static patterns to complex adaptive systems
- **Emergent behavior for the marketplace is innovation and agility**
- **Component Competition Readiness Level (CCRL) defines and measures competition readiness at the component-level throughout the lifecycle.**
  - Introduces agility into complex dynamic of the acquisition process
  - CCRL is a set of specific OSA related tasks
  - CCRL tasks are applied to the time-driven Acquisition maturity model(DoD5000).


Pressure to change
(System becomes more Nonlinear)
http://en.wikipedia.org/...Logistic_map
Steady State
Static Patterns
Wild Variations
System remains stable under external pressures to change (No effect on system behavior).
System shifts between multiple solutions and exhibits change due to external pressures.
System is unpredictable and is able to self-organize into new stable solution (emergent behavior)
Subsystem X
Subsystem Y
Subsystem X
Entry Point
Module Component A
Module Component B
External Service
Wire
Wire
Wire
Implementation
Implementation
Module A
Module B

## Value

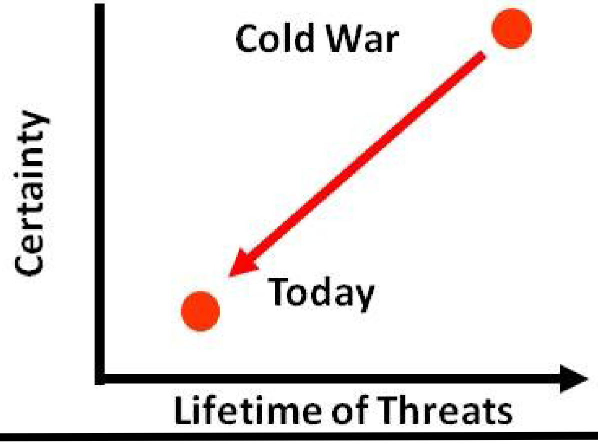
- **CCRL drives market development without prime lock-ins in post deployment context**
- **Current trends in OSA acquisition drive government/industry needs to align Data Rights Strategy (DRS)**
  - CCRL aligns DRS with OSA Strategy and Platform-driven architecture
  - Aligns DRS strategy within the Systems Engineering maturity model.
- **This effort positions CCRL as a measure for marketplace evolution for driving  agility and innovation with affordability in the Defense Acquisition.**


DARPA Historical schedule trends with complexity

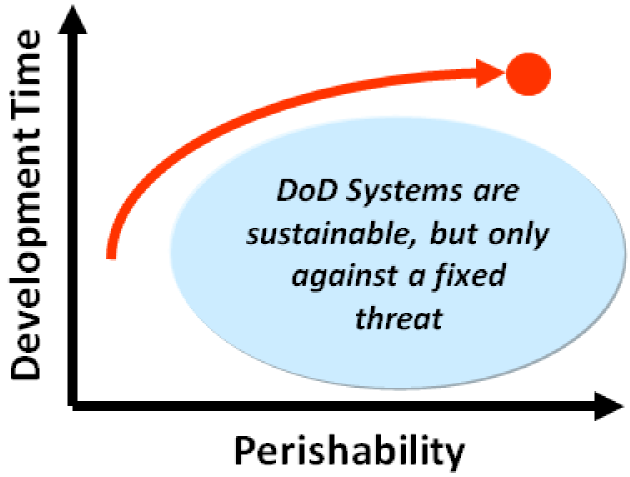
# Transformative Forces in DOD

*“Stress is the engine of innovation“*

## We are increasingly faced with threats that are surmountable, but are highly unpredictable

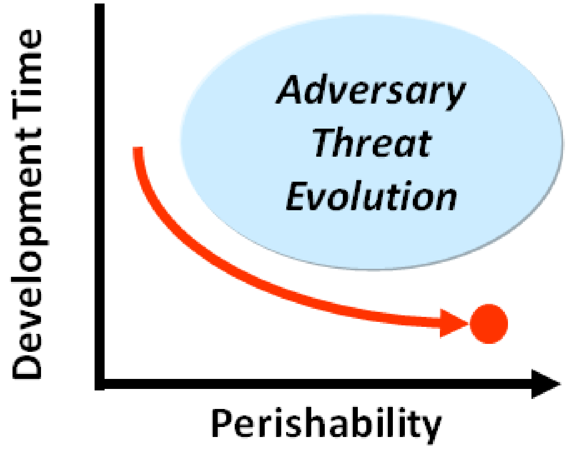


## Defense Systems Perishability

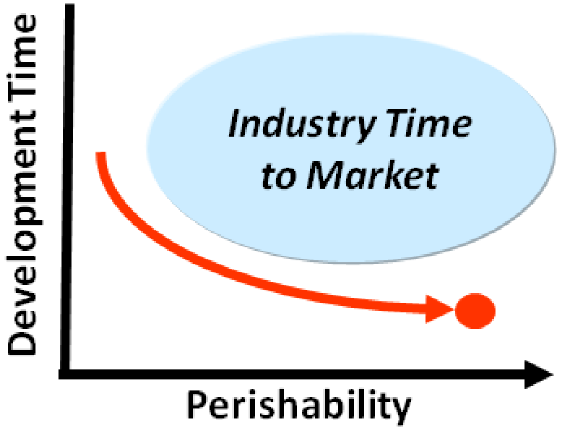


*DoD Systems are sustainable, but only against a fixed threat*

## Adversary Counter Measure Timelines



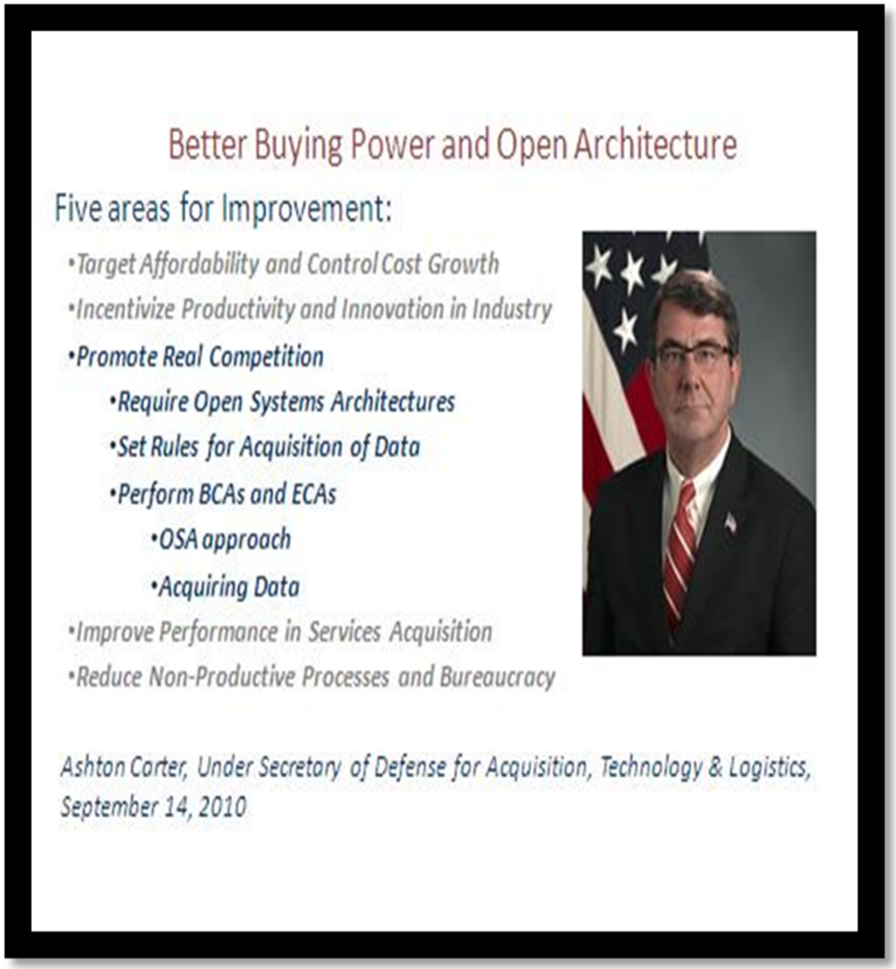
*Adversary Threat Evolution*



*Industry Time to Market*

# Better Buying Power

- Better Buying Power (BBP 1.0): Guidance for Obtaining Greater Efficiency and Productivity in Defense Spending (Dr. Ashton Carter, USD A,T&L)
- BBP 2.0 (DoD, 2013, Hon. Frank Kendall, USDAT&L): Seven initiatives to obtain greater efficiency and productivity in defense spending.
- Initiative No. 5 includes Open Systems Architecture -- “Promoting effective competition.”
- Competition: The single most powerful tool to the Department to drive productivity.



Better Buying Power and Open Architecture

Five areas for Improvement:

- *Target Affordability and Control Cost Growth*
- *Incentivize Productivity and Innovation in Industry*
- ***Promote Real Competition***
  - ***Require Open Systems Architectures***
  - ***Set Rules for Acquisition of Data***
  - ***Perform BCAs and ECAs***
    - ***OSA approach***
    - ***Acquiring Data***
- *Improve Performance in Services Acquisition*
- *Reduce Non-Productive Processes and Bureaucracy*

*Ashton Carter, Under Secretary of Defense for Acquisition, Technology & Logistics, September 14, 2010*

**Do more with less**

# Open Systems Architecture

- Merging technical architecture with an open business model

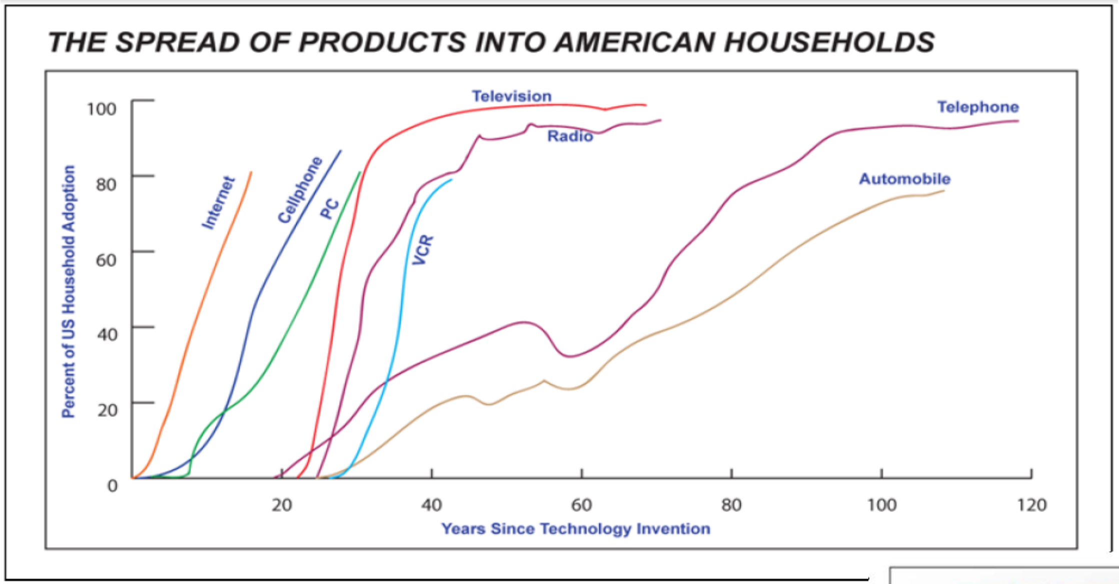
| Technical Architecture / Reference Framework | Open business model |
|---|---|
| • Defined and accepted open standards<br>• Published key interfaces<br>• Design disclosure that makes business sense<br>• Produce modular, loosely coupled, highly cohesive systems. (MOSA)<br>• Validate and certify conformance (PART / OAAT) | • Need to define appropriate metrics<br>• Must consider incentives and motivation<br>• Need to measure robustness of business community – competition<br>• Appropriate use of intellectual property and data rights – levels in layered architecture<br>• Appropriate lifecycle contractual context - FoS |

- Grading Open Systems Architecture in DoD

| In Acquisition Systems Engineering: | In post-deployment: |
|---|---|
| • An “A” in terms of driving down RDT&E / initial procurement cost | • “C-D” (Fair/Poor): Government has not fully realized fiscal relief due to lack of competition and recapture of investment |

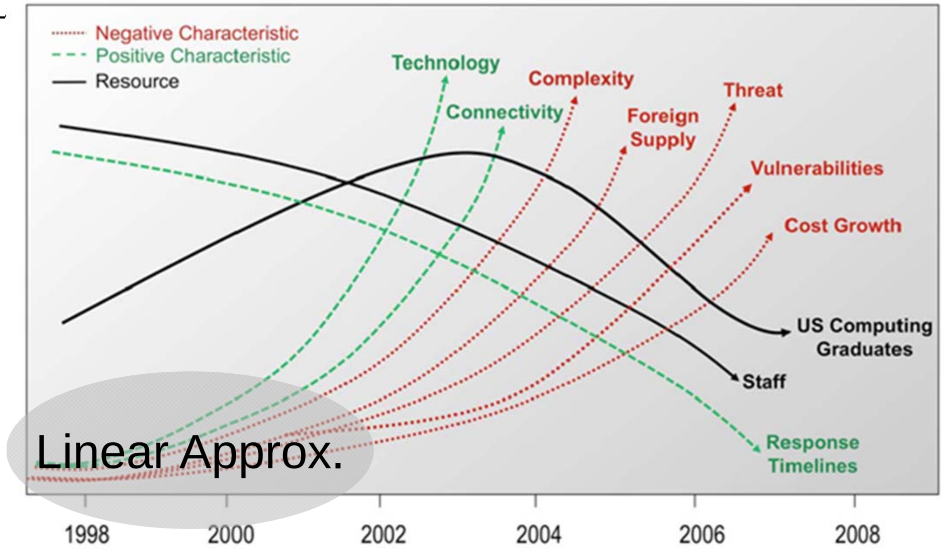
- Why this post-deployment shortcoming?
  - Cultural behavior (Government / Contractor)
  - Industry implementing from a Corporate Enterprise commonality perspective
  - Difficulty aligning Data Rights strategy with Systems Engineering maturity model and platform-driven architecture
  - Lack of governance/measures for consistent and repeatable outcomes

# Nonlinear growth and response



Commercial market has evolved to keep pace with technology adoption speed

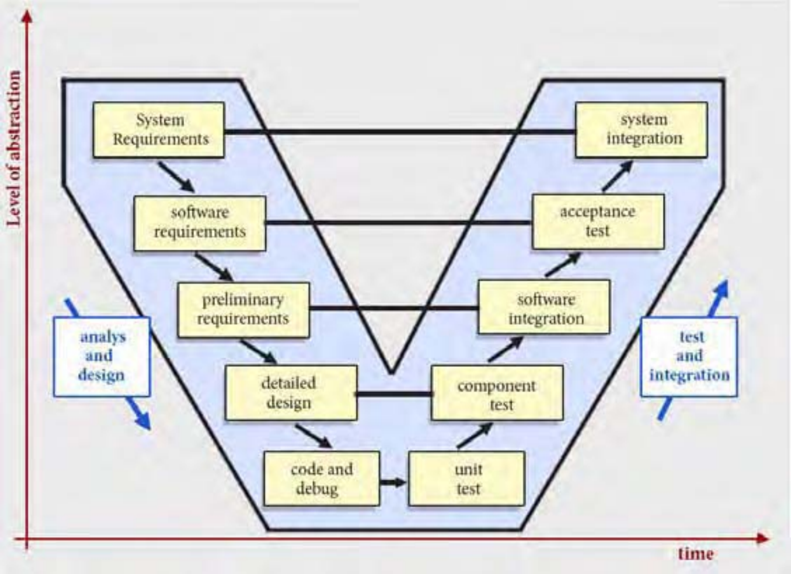
“It falls into our adversaries hands at a similarly accelerated Rate”

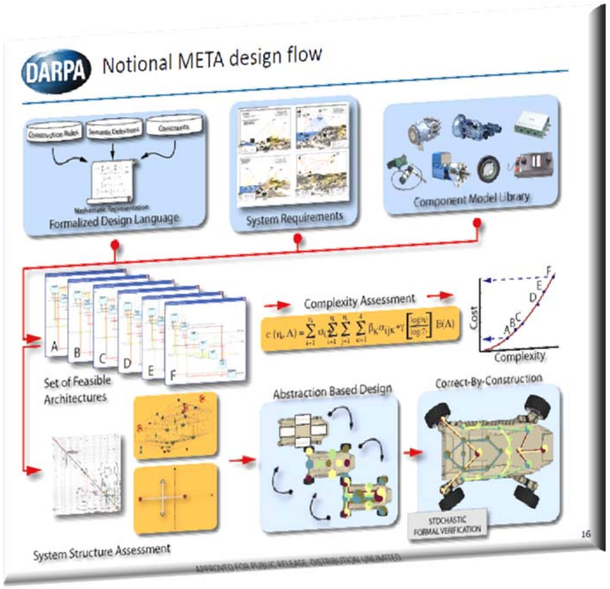


Linear Approx.

# Classical Systems Engineering is No Longer Sufficient for the Solution

- Linear Thinking
  - V-System and variants
    - CMM level 5 processes designed for structured software development tasks including requirements definition, architecture design, module development, and documentation production.
    - Difficult to be agile and adaptive
- Reductionism
  - Historically successful; search for the basic constituents
  - DARPA META program
    - Common feature of a complex system is its behavior- when parts change; the behavior of a system can sometimes be predicted—but often cannot (unintended consequences)
- Rapidly evolving, large-scale massively interconnected systems are not just scaled up versions of manufacturing project.

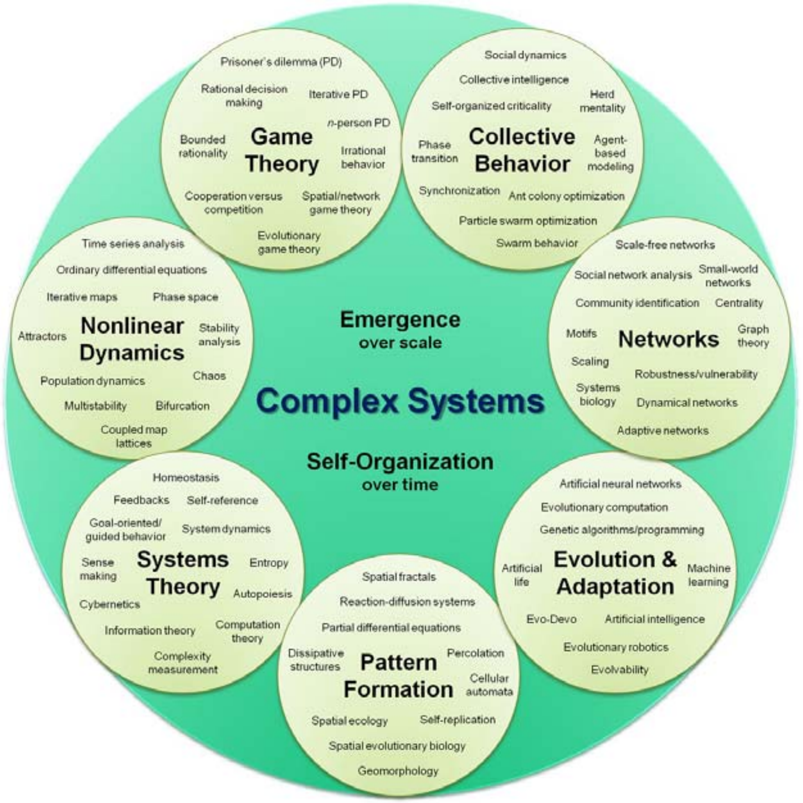



DARPA
Notional META design flow
Construction Rules
Semantic Definitions
Constraints
Mathematic Representation
Formalized Design Language
System Requirements
Component Model Library
A
B
C
D
E
F
Set of Feasible Architectures
Complexity Assessment
Cost
Complexity
Abstraction Based Design
Correct-By-Construction
STOCHASTIC FORMAL VERIFICATION
System Structure Assessment
16
APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED

# Language of Complexity

***Complex systems is a new approach to science that studies how relationships between parts give rise to the collective behaviors of a system and how the system interacts and forms relationships with its environment.*** ***(Wikipedia 2012)***

- ***Interconnectedness*** with the environment and itself
- ***Non-linearity*** of coupling and extreme sensitivity to initial conditions
- Applicability of the ***principle of superposition*** not valid
- ***Emergence*** of system properties and behaviors
- ***Self-organization***: A flock of birds organize themselves into the most appropriate formation,
- A ***power-law distribution*** of event sizes
- Many software development experts agree that a software development team is a ***complex adaptive system (CAS)***, because it is made up of multiple interacting parts within a boundary, with the capacity to change and learn from experience.



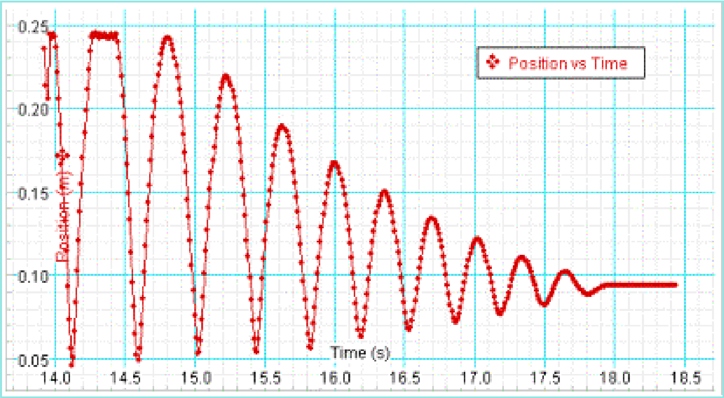
http://upload.wikimedia.org/wikipedia/commons/d/de/Complex_systems_organizational_map.jpg

# Edge of Chaos

Order        Complexity        Disorder

Stability        Oscillation        Chaos        Randomness

Short-Term Order
Long-Term Order

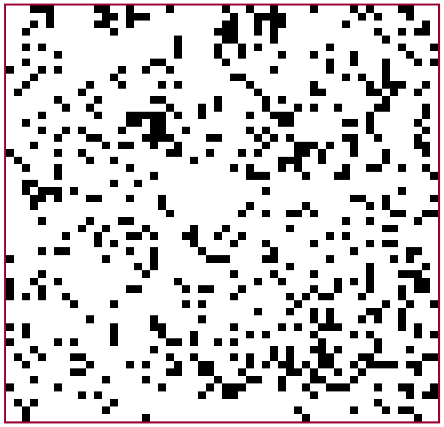


Source: http://www.personal.psu.edu/ref7/apparatus/2005%20competition/flores.htm

Short-Term Order
Long-Term Disorder

Source: http://background-wallpaper.110mb.com/background-wallpaper-fractals2.php

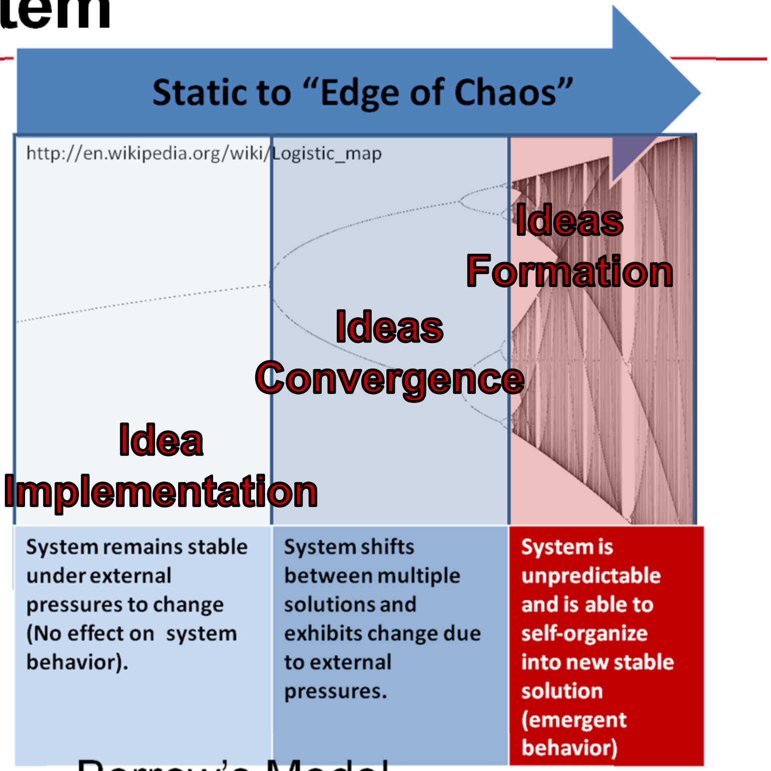
Short-Term Disorder
Long-Term Disorder



# SDLC as a Complex Adaptive System

| Complex Systems | Software |
|---|---|
| Large number of interacting elements (High dimensionality) | Large number of elements like subroutines, modules, packages, classes, functions, etc., |
| Interacting with the environment | Interacting with multiple entities with volatile requirements |
| Hierarchies consisting of local laws with global emergent properties | Stepwise refinement, top-down design, bottom up meaningful if different hierarchical levels of a have distinguishable characteristics |

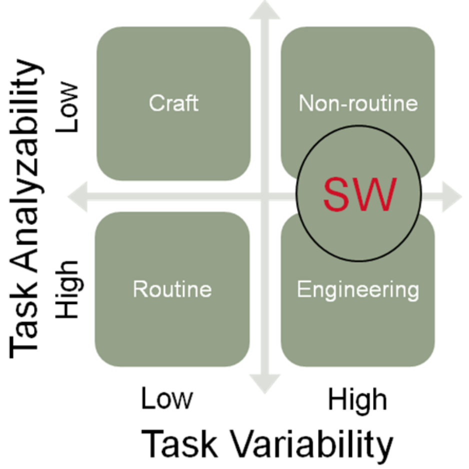
Software systems qualify as the class of complex adaptive systems that should be developed using a “complex development process”



http://en.wikipedia.org/wiki/Logistic_map

| | | |
|---|---|---|
| System remains stable under external pressures to change (No effect on system behavior). | System shifts between multiple solutions and exhibits change due to external pressures. | System is unpredictable and is able to self-organize into new stable solution (emergent behavior) |

Perrow’s Model



# Cont.

- Open Source Software also appears to follow self-organizing system
  - Self-organizing processes, when modeled as growing networks, display non-random attachment of nodes
  - Social networks, collaborative networks, and other self-organizing systems (e.g., the Internet, WWW pages, U.S. firm sizes, cities, economic systems, word usage in languages, ecosystems) often have another interesting property; they have highly skewed distributions, which under a log-log transformation results in a linear relationship. This is called a power-law relationship.
  - Study of projects at SourceForge show

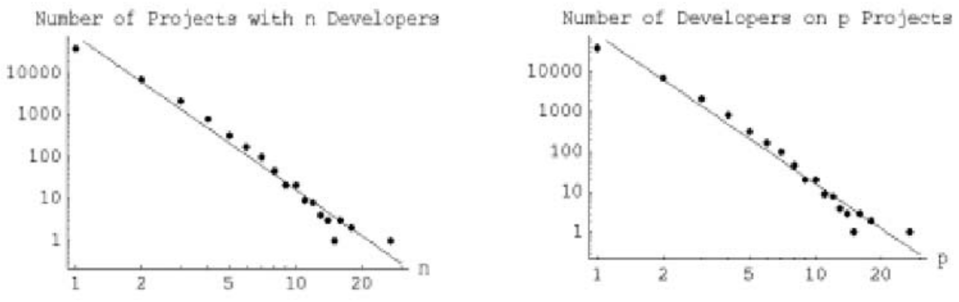


Figure 2. Power Law Relationships: OSS Project Size and Developer Project Membership

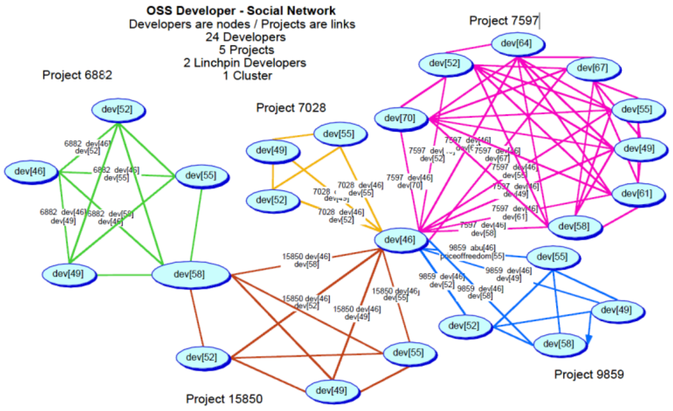
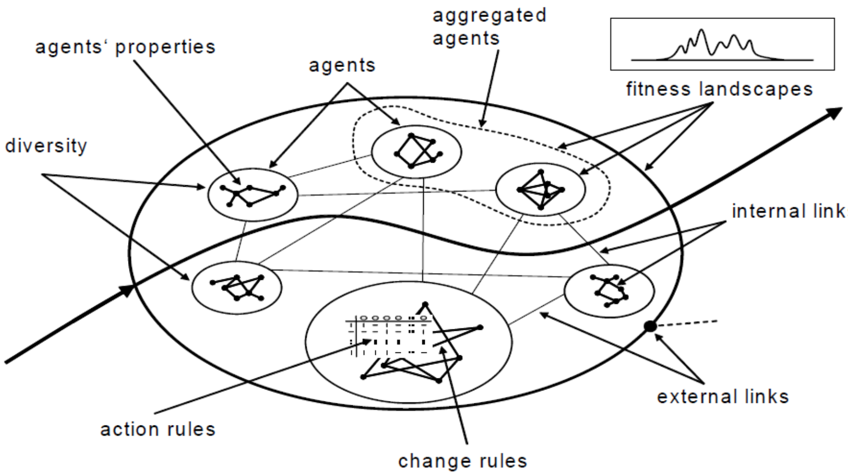


Figure 1. Developer Social-Network, Linked by Joint Project Membership, Cluster of Size 24

# Principles of designing a self-organizing enterprise

- Characterize entities, roles and interactions through models
- Develop Information Model and automate workflows
- Simplify processes; distribute and delegate decision authority to lowest levels possible
- Connect developers and consumers throughout the development process using the value system
- Free-scale networks: High cluster coefficient with a small diameter. Information easily and quickly diffuses through the network, even as nodes continuously join and leave the network.

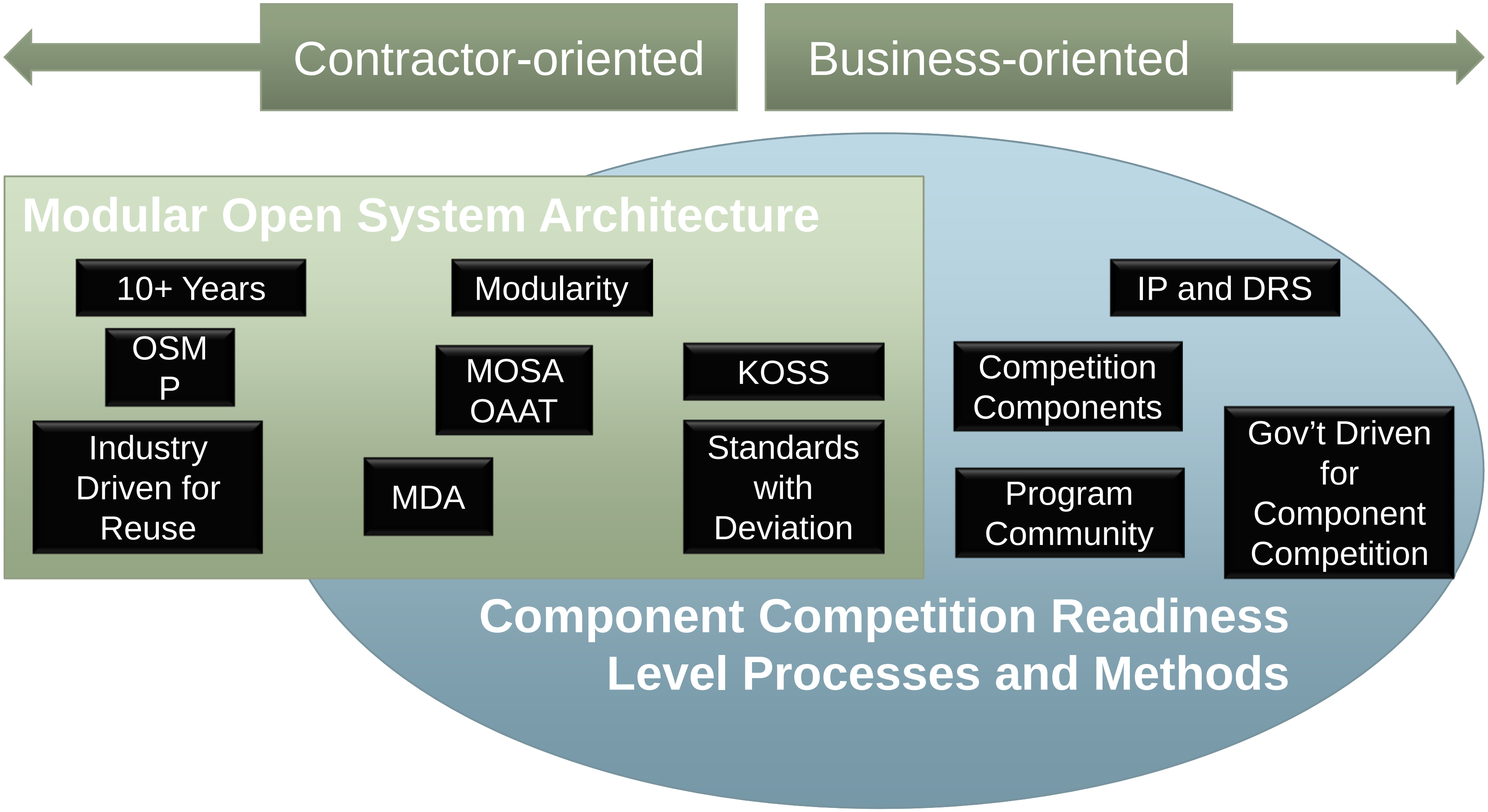


Meike Tilebein, “Principles of Emergence - A Generic Framework of Firms as Agent-Based Complex Adaptive Systems”

# Component Competition Readiness Level (CCRL)

- Measuring maturity levels of both the Open Business Model and the Technical architecture / Reference Framework.
- Establishing a business ecosystem to foster the proper dynamics between the business and technical framework.
- Complementing Technology Readiness Level (TRL) with component-level metrics relating to integration, interoperability and program readiness for Component Competition.

# MOSA and CCRL



Contractor-oriented

Business-oriented

Modular Open System Architecture

10+ Years

OSM P

Industry Driven for Reuse

Modularity

MOSA OAAT

MDA

KOSS

Standards with Deviation

IP and DRS

Competition Components

Program Community

Gov't Driven for Component Competition

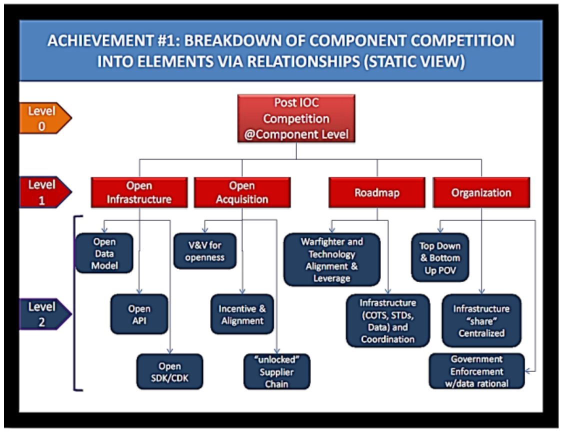
Component Competition Readiness Level Processes and Methods

Industry drove MOSA to win contracts by reusing and refactoring software and hardware modules.

Government program teams will need to drive CCRL-measured processes and activities to build the dynamic process and ecosystem of a program where competition is the emergent behavior of the system.

# CCRL: Top Three Levels

- Level 0: Goal
  - Reduce total ownership cost through agility and adaptability
- Level 1: Drivers
  - Technical drivers were addressed through Open Infrastructure and Roadmaps.
  - Business drivers were addressed through Open Acquisition and Organization.
- Level 2: Measurable Objectives
  - Inter-relationship of objectives that generate a complex dynamic behavior resulting in competition



ACHIEVEMENT #1: BREAKDOWN OF COMPONENT COMPETITION INTO ELEMENTS VIA RELATIONSHIPS (STATIC VIEW)

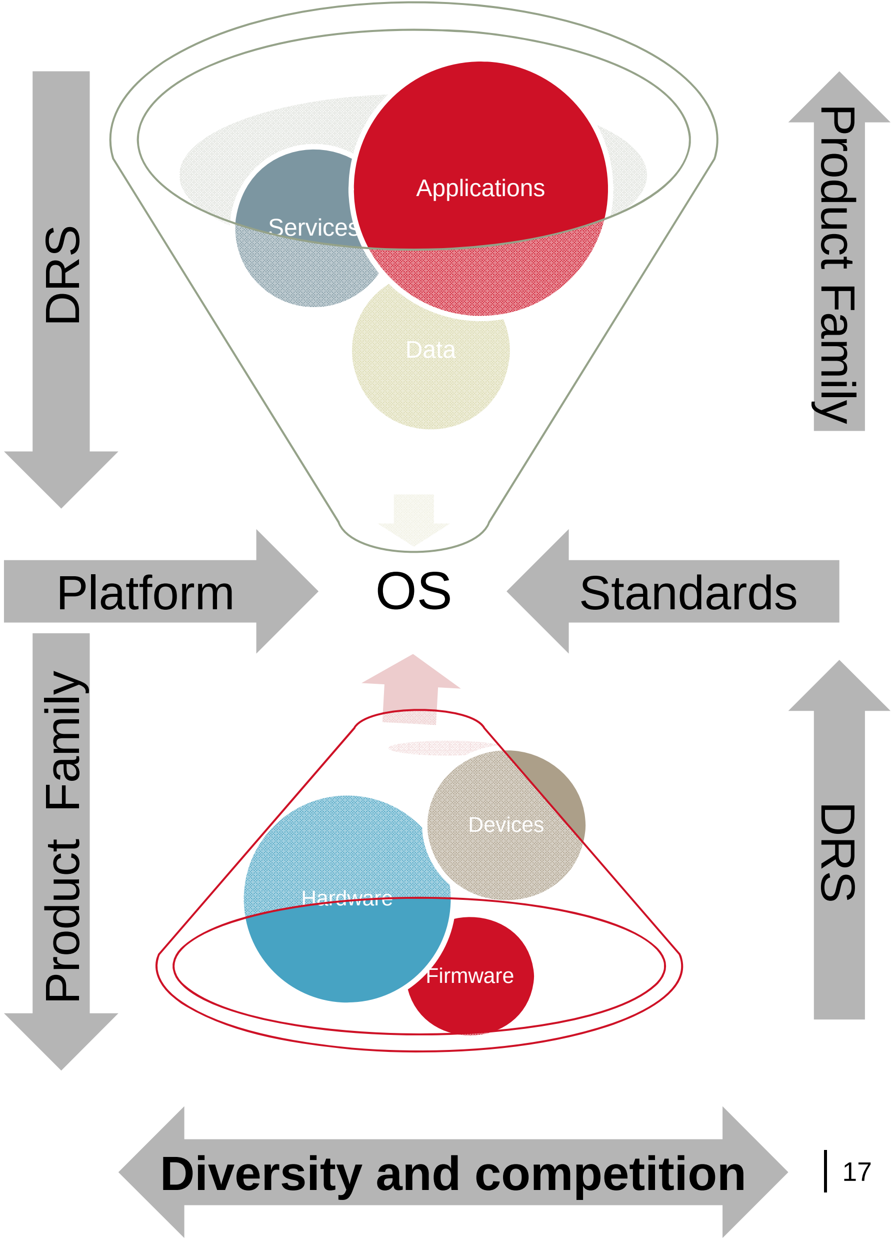
Level 0
Post IOC Competition @Component Level
Level 1
Open Infrastructure
Open Acquisition
Roadmap
Organization
Level 2
Open Data Model
V&V for openness
Warfighter and Technology Alignment & Leverage
Top Down & Bottom Up POV
Open API
Incentive & Alignment
Infrastructure (COTS, STDs, Data) and Coordination
Infrastructure "share" Centralized
Open SDK/CDK
"unlocked" Supplier Chain
Government Enforcement w/data rational

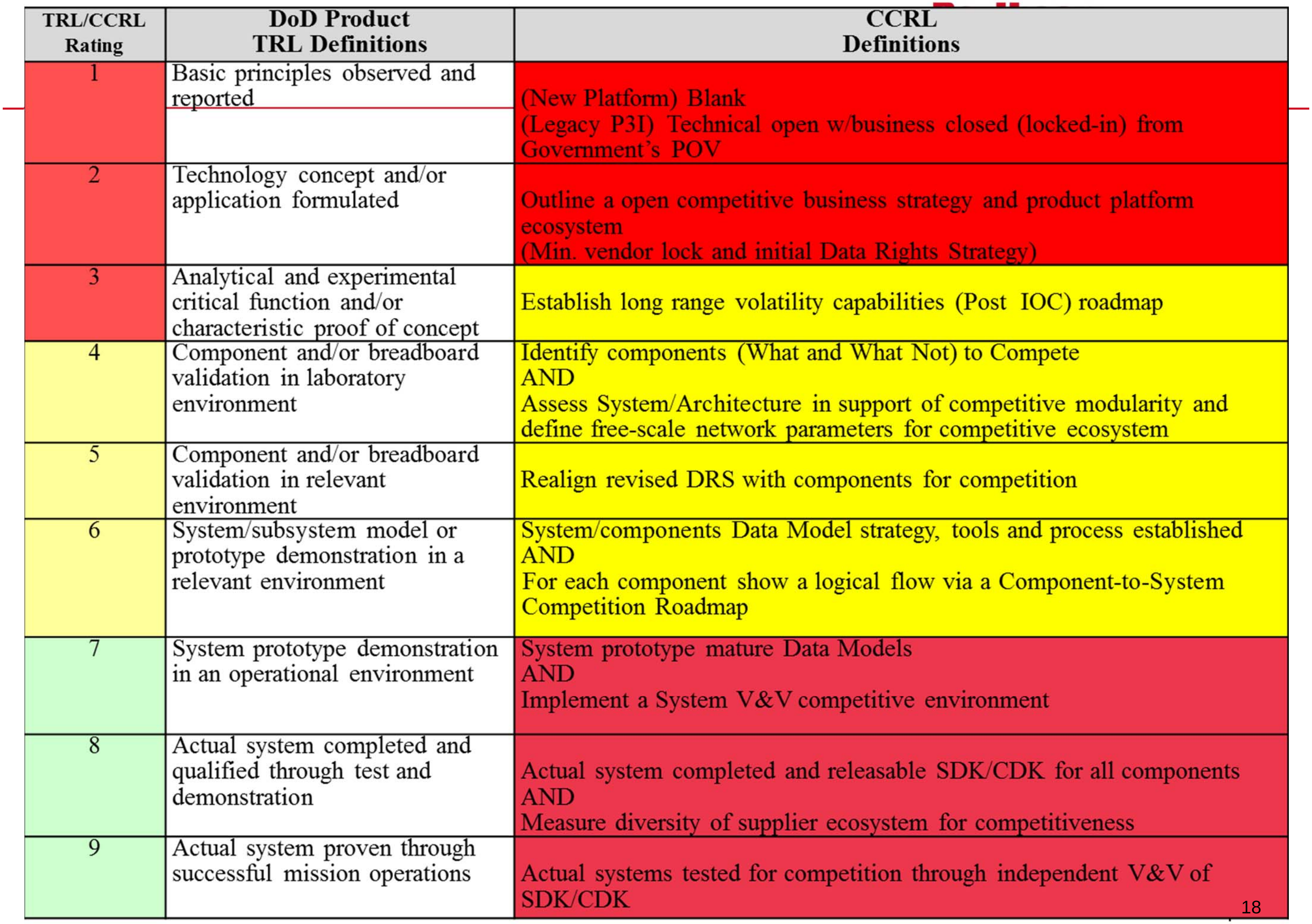
# Open Infrastructure Composition

- Open community, (then) Open Architecture
- Stakeholders drive the development Interface Technology Requirements via measurable objectives:
  - Common Data Models
  - Open Application Programming Interface
  - Open Software AND Component Development Kits (SDK/CDK)
- With third-party evaluations to judge the openness of the infrastructure.

# Open Acquisition Marketplace

- V&V for transparency
- Adequate incentives and alignment to promote good behavior
- Assurance suppliers are not locked-in
- Measurable (free-scale network ?) robust supplier network
- Aligned with platform-based product family development
  - Layered Bowtie/hourglass architecture
- Alignment of Data Right Strategy (DRS) with open architecture Component Competition strategy.





| TRL/CCRL Rating | DoD Product TRL Definitions | CCRL Definitions |
|---|---|---|
| 1 | Basic principles observed and reported | (New Platform) Blank<br>(Legacy P3I) Technical open w/business closed (locked-in) from Government's POV |
| 2 | Technology concept and/or application formulated | Outline a open competitive business strategy and product platform ecosystem<br>(Min. vendor lock and initial Data Rights Strategy) |
| 3 | Analytical and experimental critical function and/or characteristic proof of concept | Establish long range volatility capabilities (Post IOC) roadmap |
| 4 | Component and/or breadboard validation in laboratory environment | Identify components (What and What Not) to Compete<br>AND<br>Assess System/Architecture in support of competitive modularity and define free-scale network parameters for competitive ecosystem |
| 5 | Component and/or breadboard validation in relevant environment | Realign revised DRS with components for competition |
| 6 | System/subsystem model or prototype demonstration in a relevant environment | System/components Data Model strategy, tools and process established<br>AND<br>For each component show a logical flow via a Component-to-System Competition Roadmap |
| 7 | System prototype demonstration in an operational environment | System prototype mature Data Models<br>AND<br>Implement a System V&V competitive environment |
| 8 | Actual system completed and qualified through test and demonstration | Actual system completed and releasable SDK/CDK for all components<br>AND<br>Measure diversity of supplier ecosystem for competitiveness |
| 9 | Actual system proven through successful mission operations | Actual systems tested for competition through independent V&V of SDK/CDK |