2014-05-20

A decision framework for systems of systems based on operational effectiveness

Young, Bonnie

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/43420

NAVAL
POSTGRADUATE
SCHOOL

A Decision Framework for Systems of Systems Based on Operational Effectiveness

Bonnie Young
Naval Postgraduate School
Professor, Systems Engineering
bwyoung@nps.edu
703-407-4531

SoS Decision Framework

Research Objectives:

- ❖ To develop a framework that enables SoS “design” decisions that are based on operational effectiveness
- ❖ To achieve “purpose-driven” SoS’s

Definition of SoS

"An SoS is a set or arrangement of systems that result when independent and useful systems are integrated into a larger system that delivers unique capabilities...."

- (From "OSD SE Guide for SoS, 2008" (ODUSD(A&T)SSE))

Types of SoS

1. **Virtual** – Virtual SoS lack a central management authority and a centrally agreed upon purpose for the system-of-systems. Large-scale behavior emerges—and may be desirable—but this type of SoS must rely upon relatively invisible mechanisms to maintain it.
2. **Collaborative** – In collaborative SoS the component systems interact more or less voluntarily to fulfill agreed upon central purposes. The Internet is a collaborative system. The Internet Engineering Task Force works out standards but has no power to enforce them. The central players collectively decide how to provide or deny service, thereby providing some means of enforcing and maintaining standards.
3. **Acknowledged** - Acknowledged SoS have recognized objectives, a designated manager, and resources for the SoS; however, the constituent systems retain their independent ownership, objectives, funding, and development and sustainment approaches. Changes in the systems are based on collaboration between the SoS and the system.
4. **Directed** - Directed SoS are those in which the integrated SoS is built and managed to fulfill specific purposes. It is centrally managed during long-term operation to continue to fulfill those purposes as well as any new ones the system owners might wish to address. The component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the central managed purpose.

(From "OSD SE Guide for SoS, 2008" (ODUSD(A&T)SSE))

SoS Types
(Bonnie's Definition)

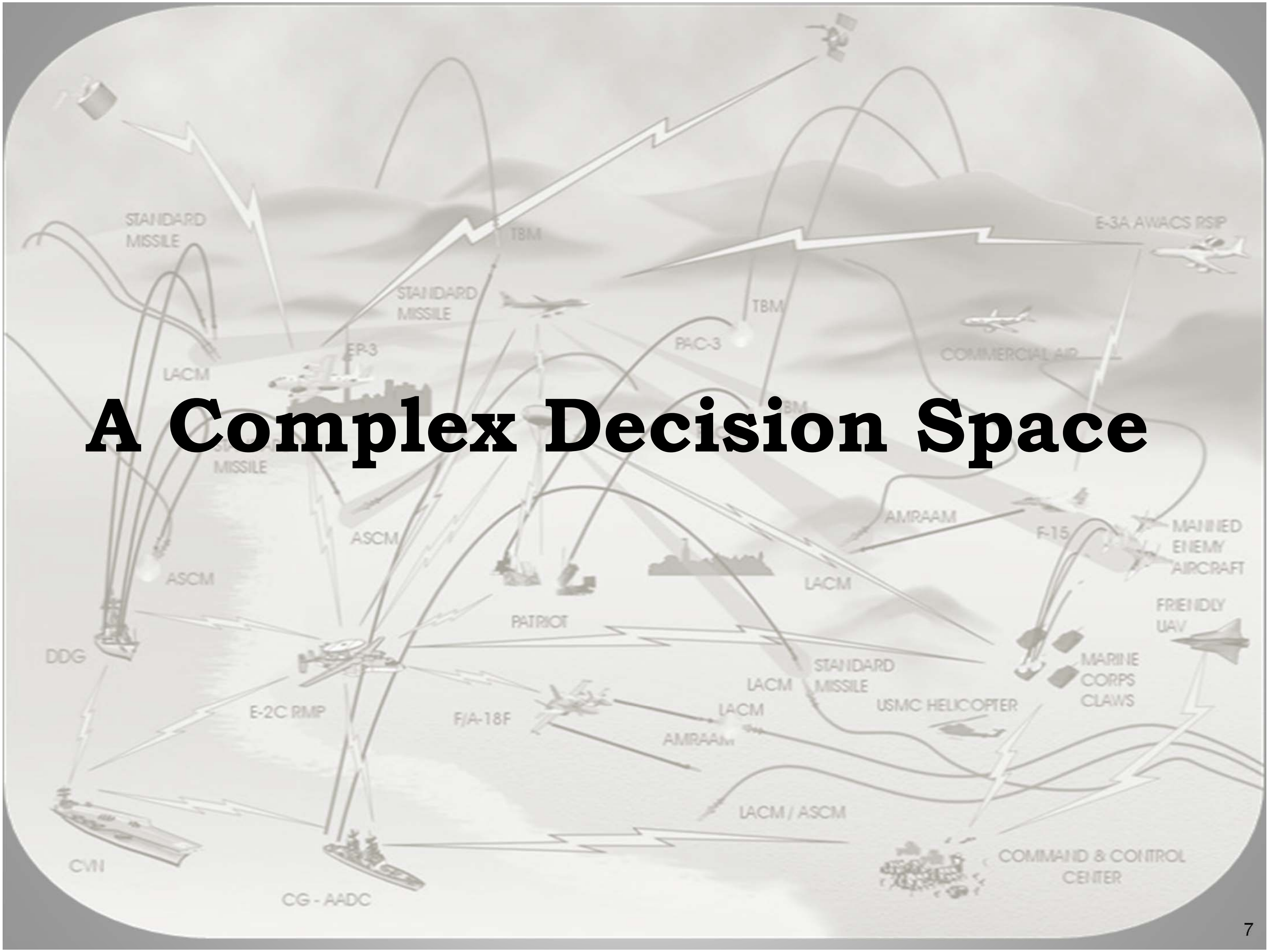
1. **Legacy System SoS** – an SoS made up of legacy systems. Design decisions are limited to the architecture and interfaces; bottoms-up design & development.
2. **Clean Slate SoS** – an SoS whose design originates from a "clean slate". The SoS is designed from scratch with little or no legacy system constraints. Design can be optimized based on the operational missions and objectives.
3. **Hybrid SoS** – an SoS comprised of a hybrid of new and legacy systems; and major upgrades to existing systems. Design decisions are concerned with the architecture, choice of participating systems, interfaces, and prioritization of upgrades to existing systems.
4. **Self-Organizing SoS** – an SoS whose constituent systems "self-organize" or collaborate in a changing manner as systems enter or exit the SoS and/or as emergent SoS behavior is needed to meet operational objectives. Self-organized SoS are formed by decisions made by the systems that decide to collaborate with one another.

Self-Organizing SoS

- Envisioned characteristics: **agile**, **adaptable**, **reactive**, **evolving**, **proactive**, and **harmonious** (Nichols & Dove, 2011)

- **Technical Requirements**:
  - Systems must be communicating with one another
  - Systems must have resident (embedded) capability to understand the operational mission needs
  - Systems must determine whether they can offer capability by joining the SoS (or forming one with other systems)

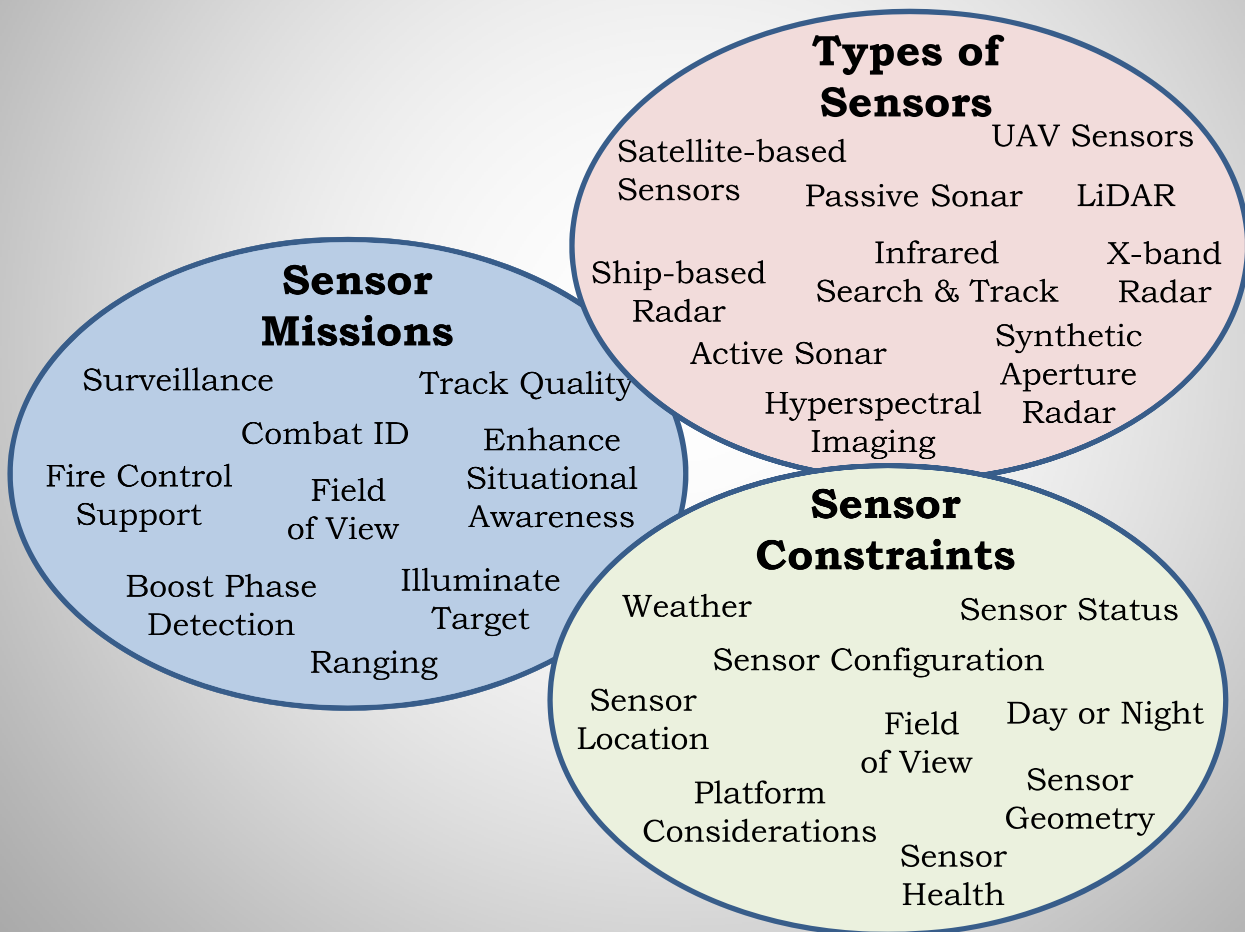
A Complex Decision Space



What makes the Decision Space Complex?

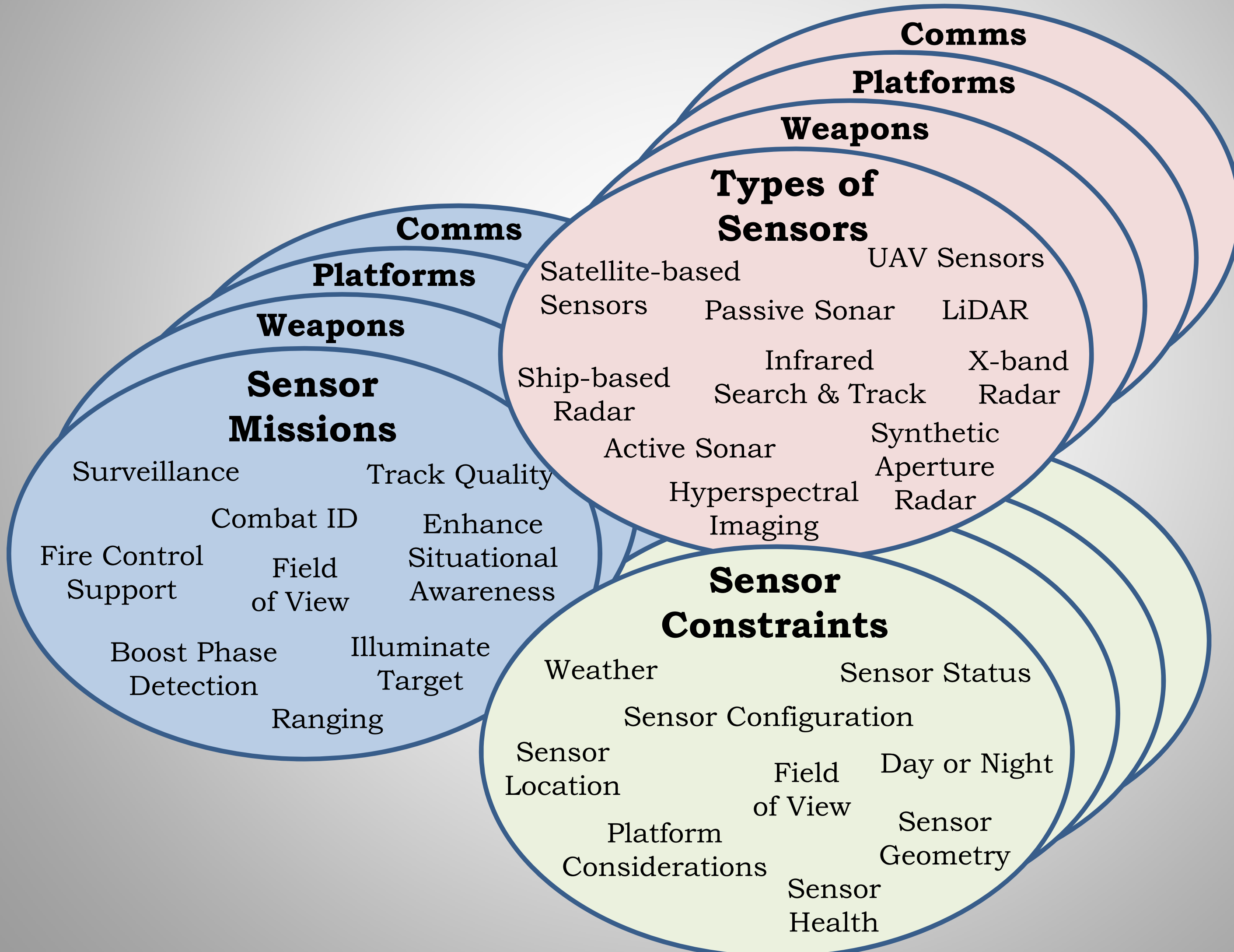
- Time-criticality
- Threat complexity
- Prioritization of operational objectives
- Limits to situational awareness
- Changing nature of operation
- Distribution and heterogeneity of warfare assets
- Command and control complexity

Sensor Resources
Leading to Decision Complexity



Warfare Resources

Leading to Decision Complexity

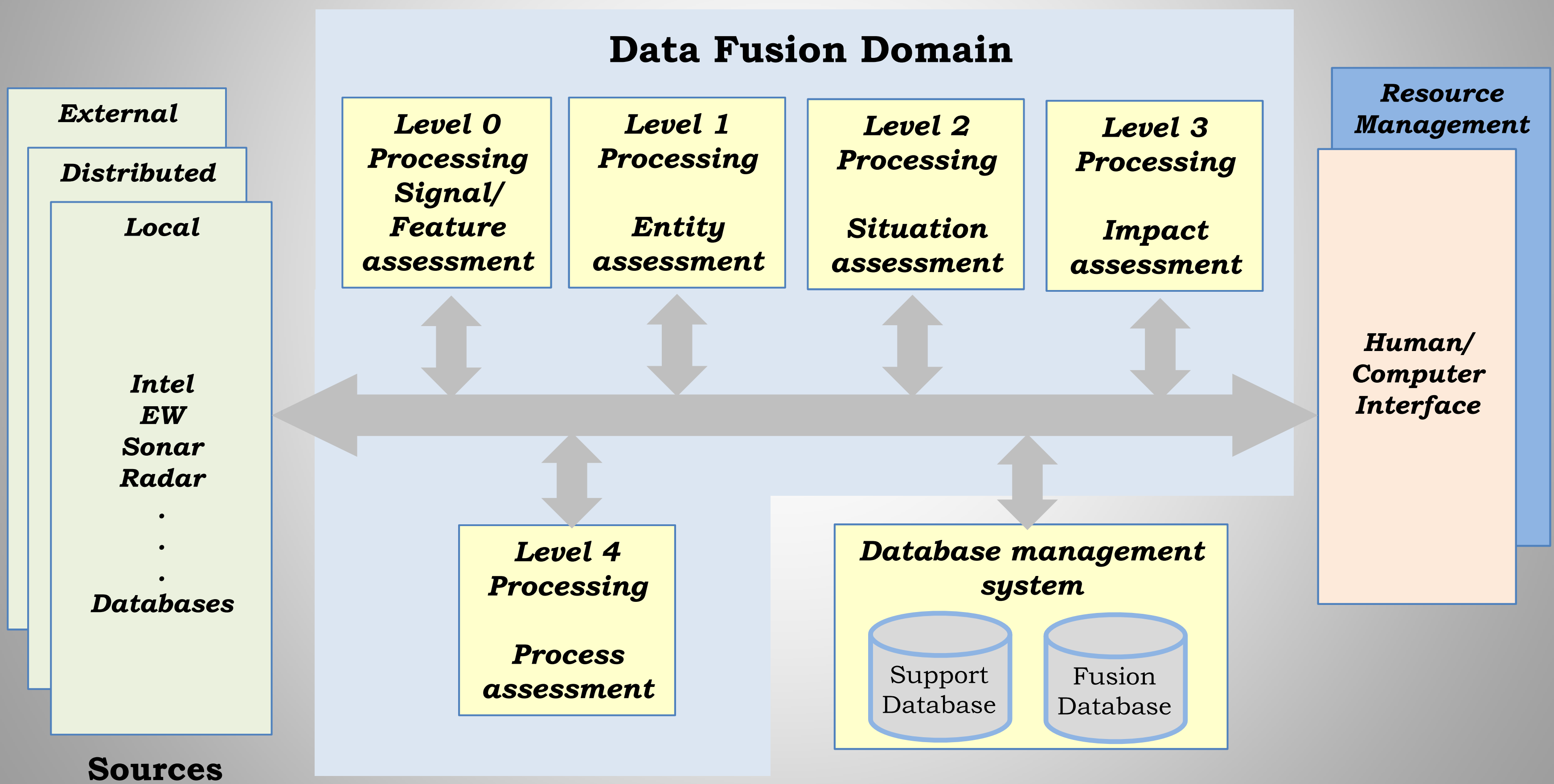


Strategies

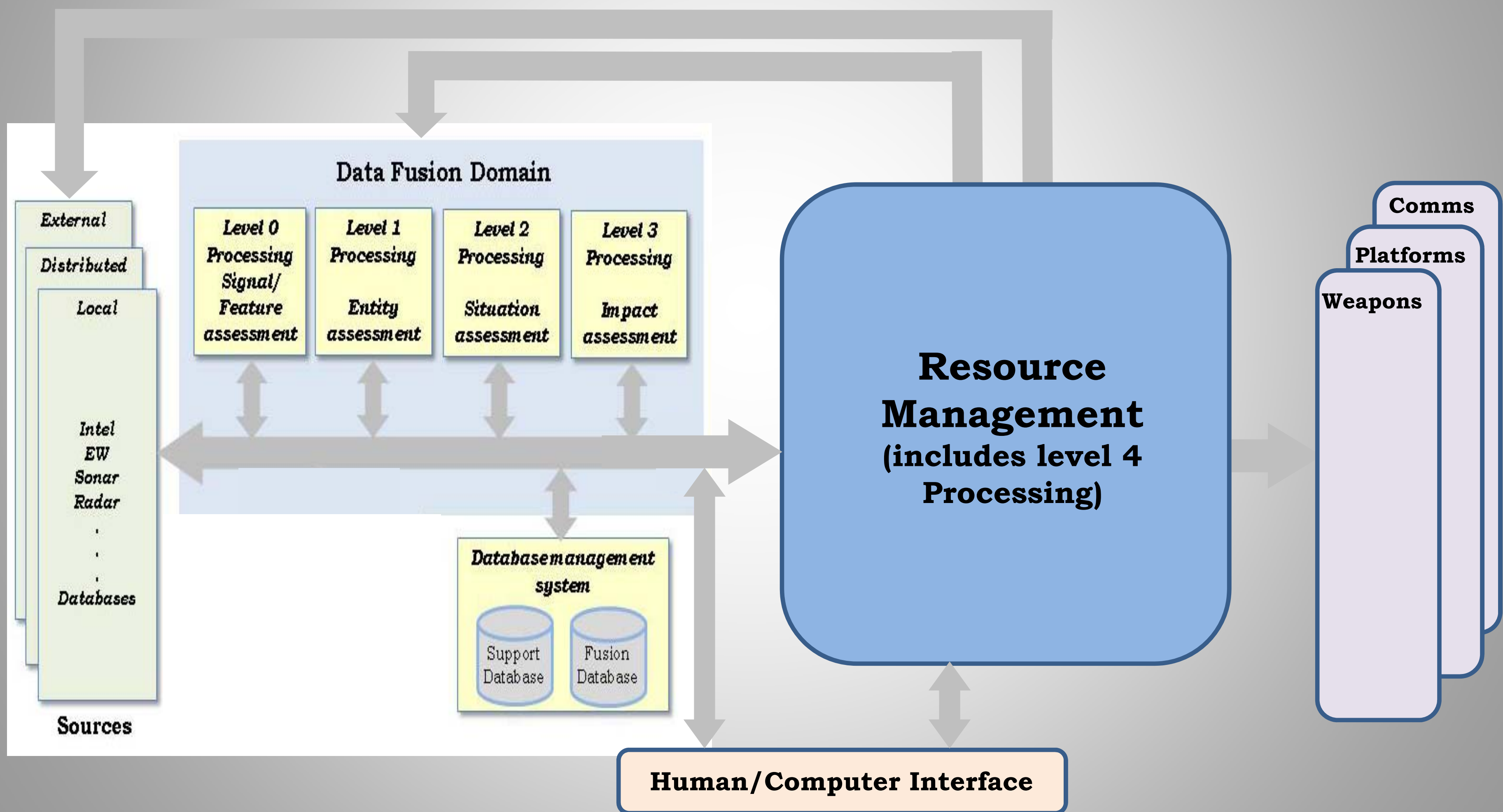
- Use warfare resources collaboratively as Systems of Systems (SoS)
- Use an NCW approach to network distributed assets
- Achieve situational awareness to support resource tasking/operations
- Fuse data from multiple sources
- Employ common processes across distributed warfare resources
- Use decision-aids to support C2

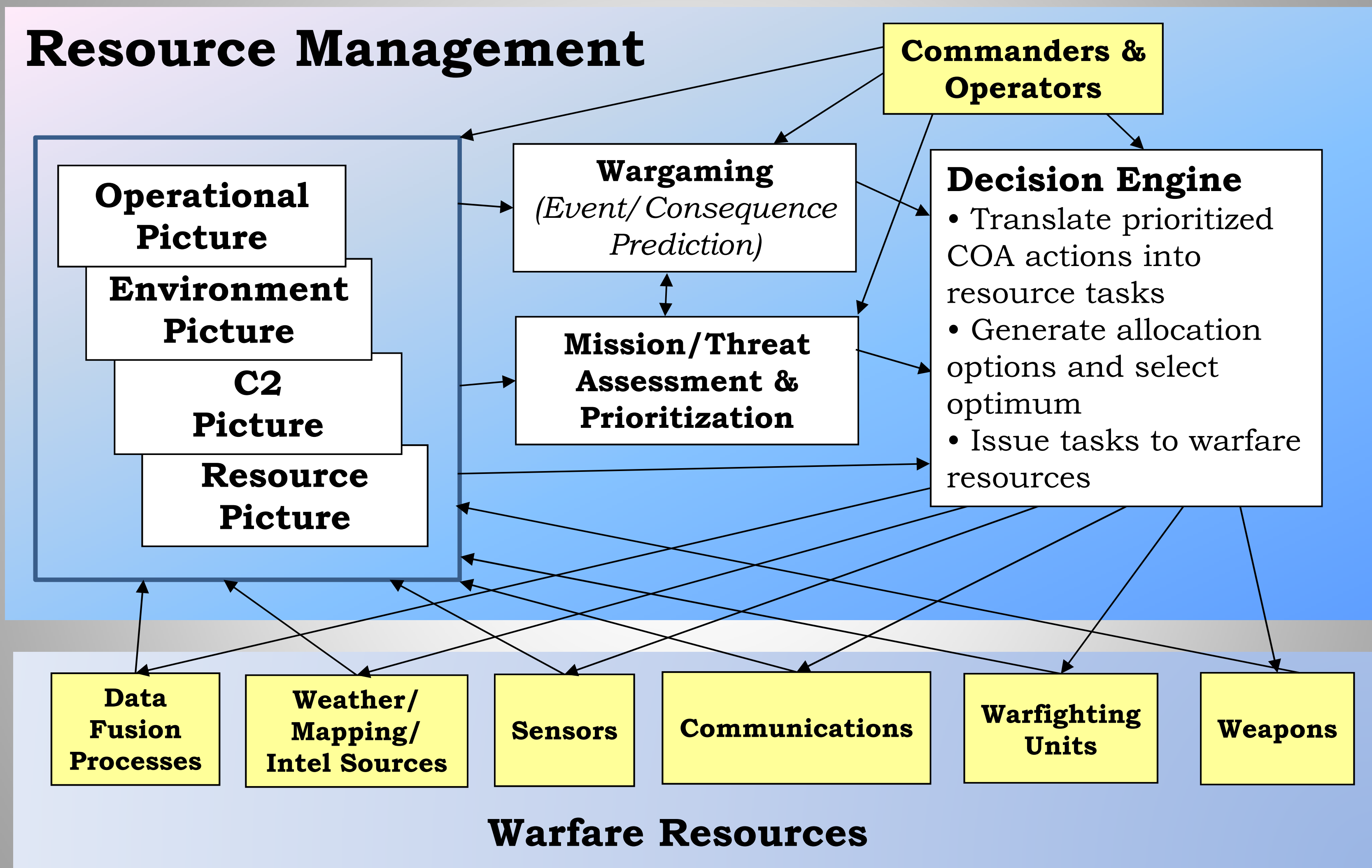
Over-arching Objective: To most effectively use warfare resources to meet tactical operational objectives

JDL Data Fusion Model: Data-Centric Framework



Shift to a Decision-Centric Framework





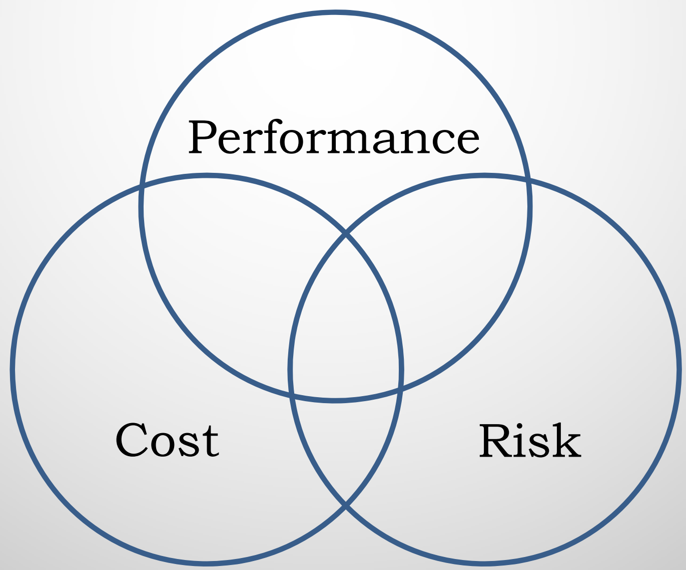
Resource Management
Commanders & Operators
Operational Picture
Environment Picture
C2 Picture
Resource Picture
Wargaming
(Event/Consequence Prediction)
Mission/Threat Assessment & Prioritization
Decision Engine
• Translate prioritized COA actions into resource tasks
• Generate allocation options and select optimum
• Issue tasks to warfare resources
Data Fusion Processes
Weather/ Mapping/ Intel Sources
Sensors
Communications
Warfighting Units
Weapons
Warfare Resources

Conceptual RM Capability

- **Architecture Considerations**
  - Distributed RM “instances”
  - Synchronization
  - Hybrid: dummy C2 nodes and RM C2 nodes
- **Continuous On-going RM Process**
  - Operational situation/missions are changing
  - Decision assessments must change in response—instead of a single assessment
- **Level of Automation**
  - How much of the RM concept is automated?
  - RM is a decision-aid
  - Human C2 decision-makers must be able to manipulate information, prioritizations, and taskings

Applying SE Design Methods to Distributed Resource Management

An analogy exists between the SE design process and operational C2 decisions for resource management

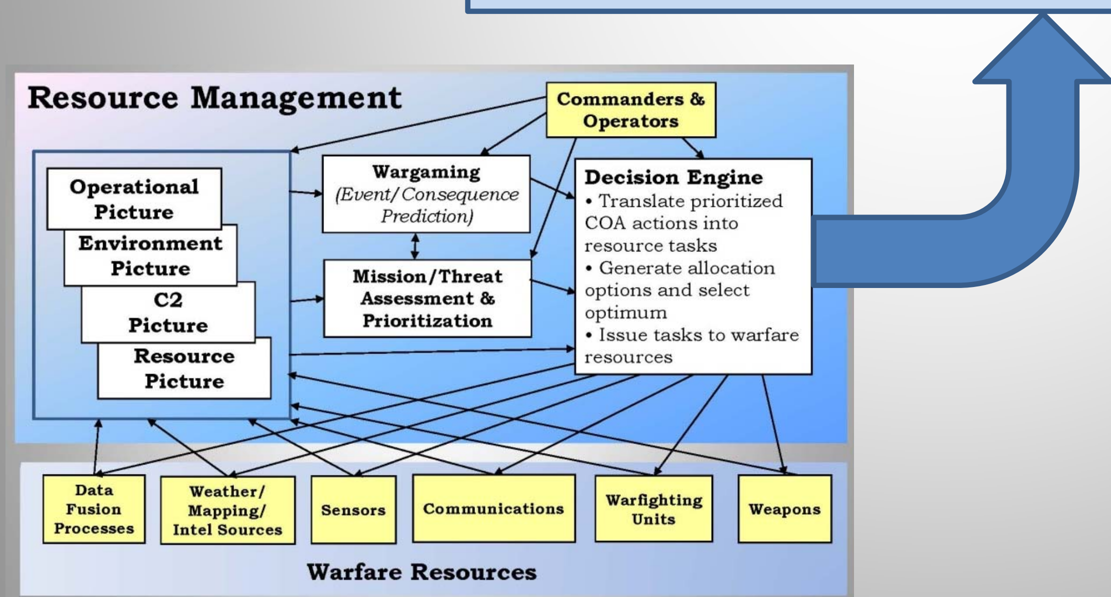


Resource Management Decision Assessments

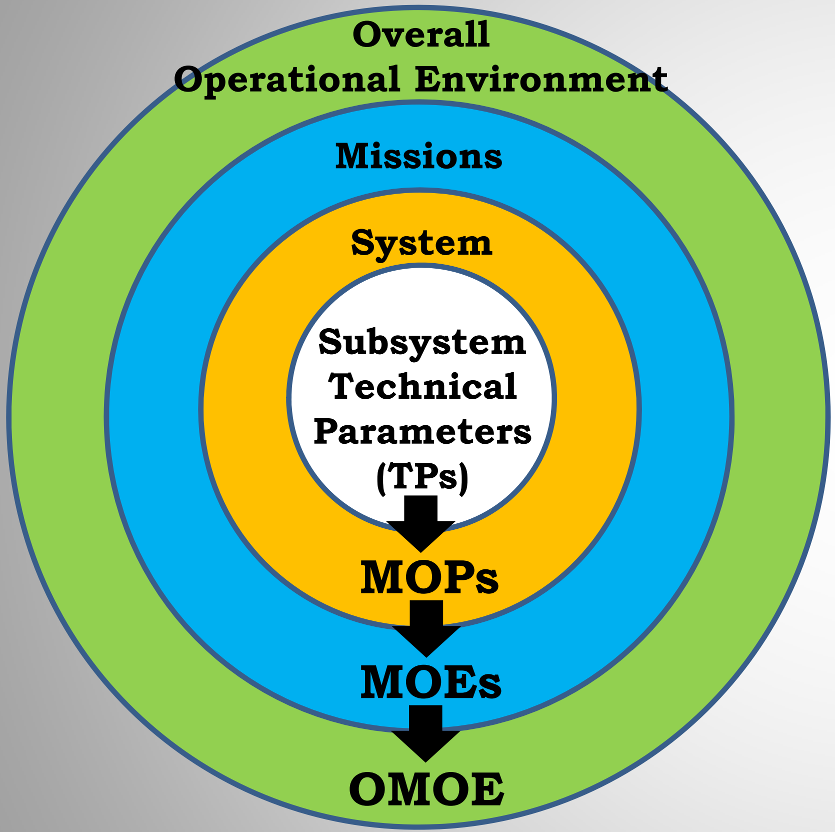
Performance
OMOE Decision Engine

"Cost"
Decision Cost Engine

"Risk"
Decision Confidence Engine



Measures of Merit for a System



$$MOE_j = \Sigma\ w_i\ MOP_i$$

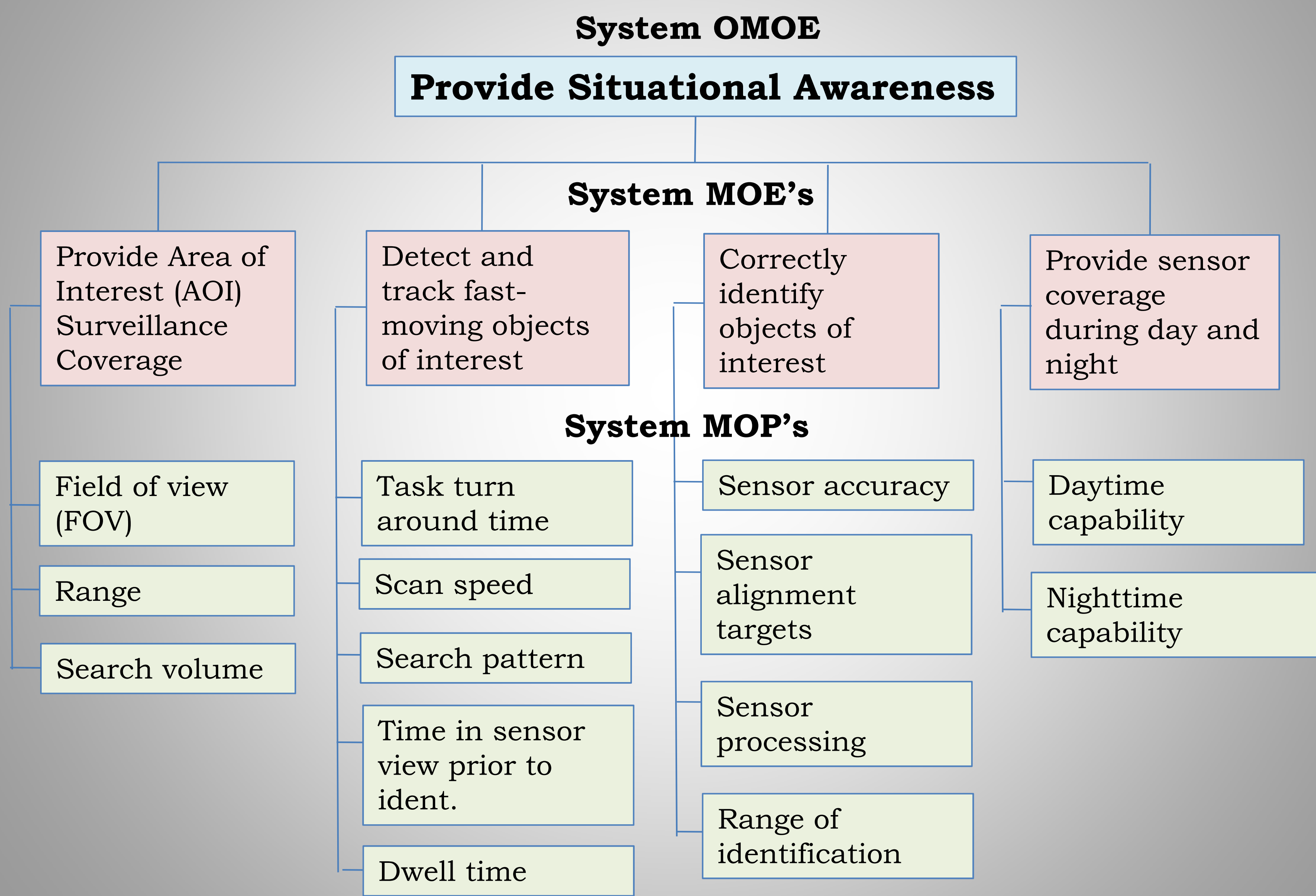
$$OMOE = \Sigma\ w_j\ MOE_j$$

MOPs – measure inherent attributes of system behavior

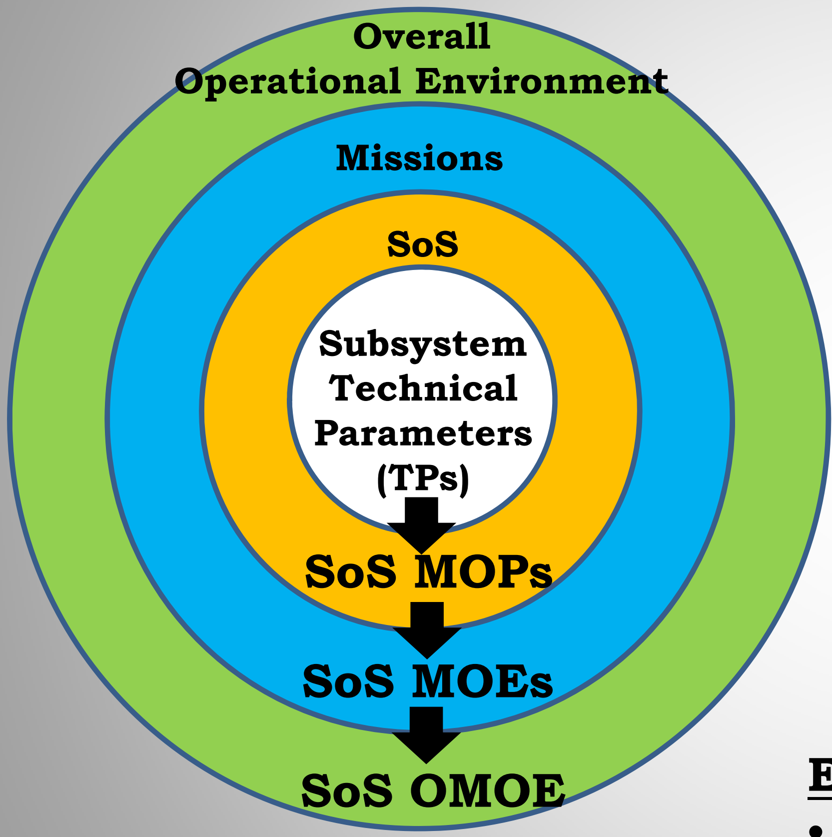
MOEs – measure how well a system performs against a single operational mission

OMOE – a single metric that measures how well a system performs across multiple operational missions

Examples of Performance Measures



Measures of Merit for a SoS



SoS MOPs – measure inherent attributes of SoS behavior

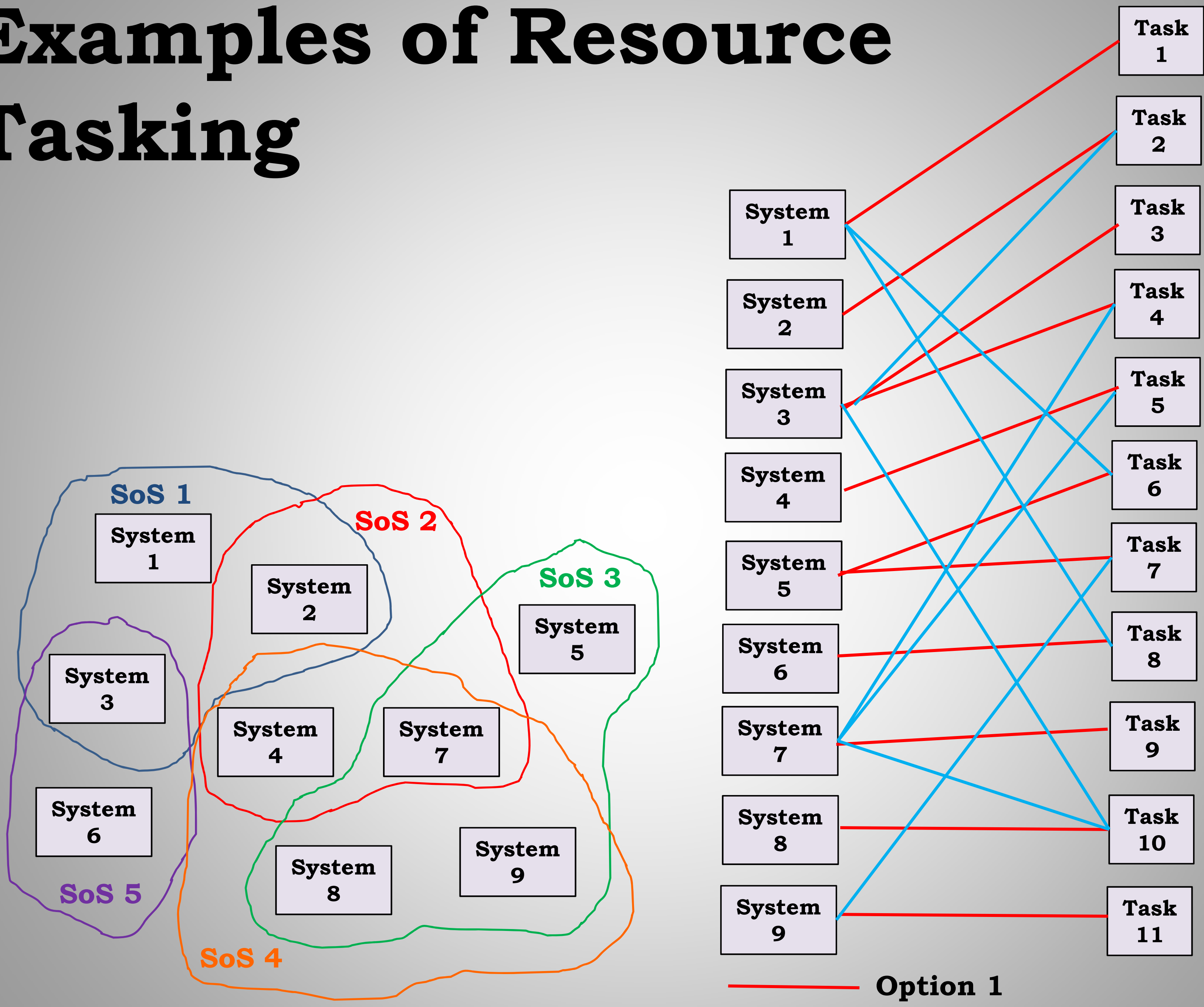
SoS MOEs – measure how well a SoS performs in an operational environment

SoS OMOE – a single metric that measures how well a SoS performs across multiple operational missions

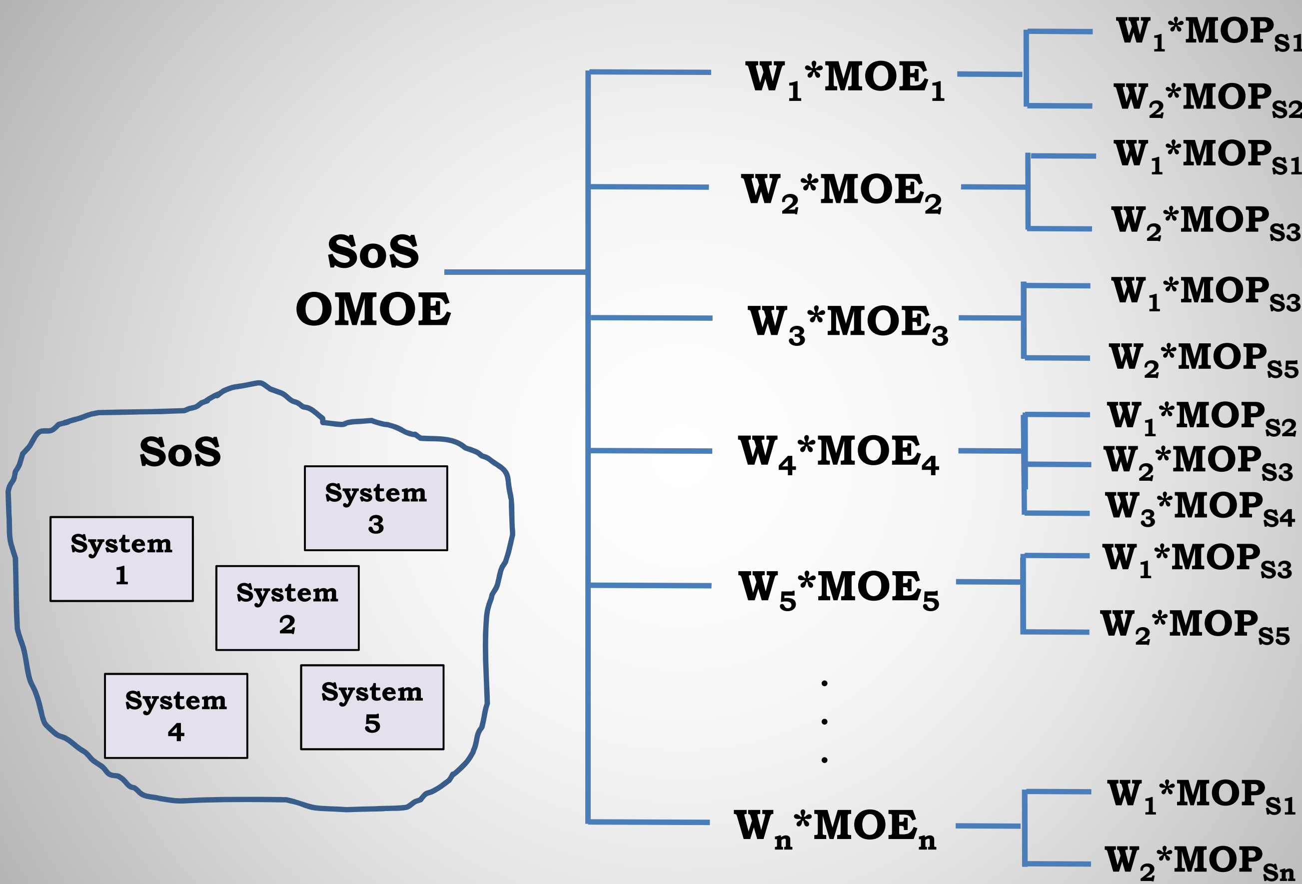
Example SoS MOPs:

- Level of interoperability achieved
- Overall Technical Readiness Level
- Accuracy of SoS Situational Awareness
- SoS Decision Response Time
- Synchronization of SoS Datasets

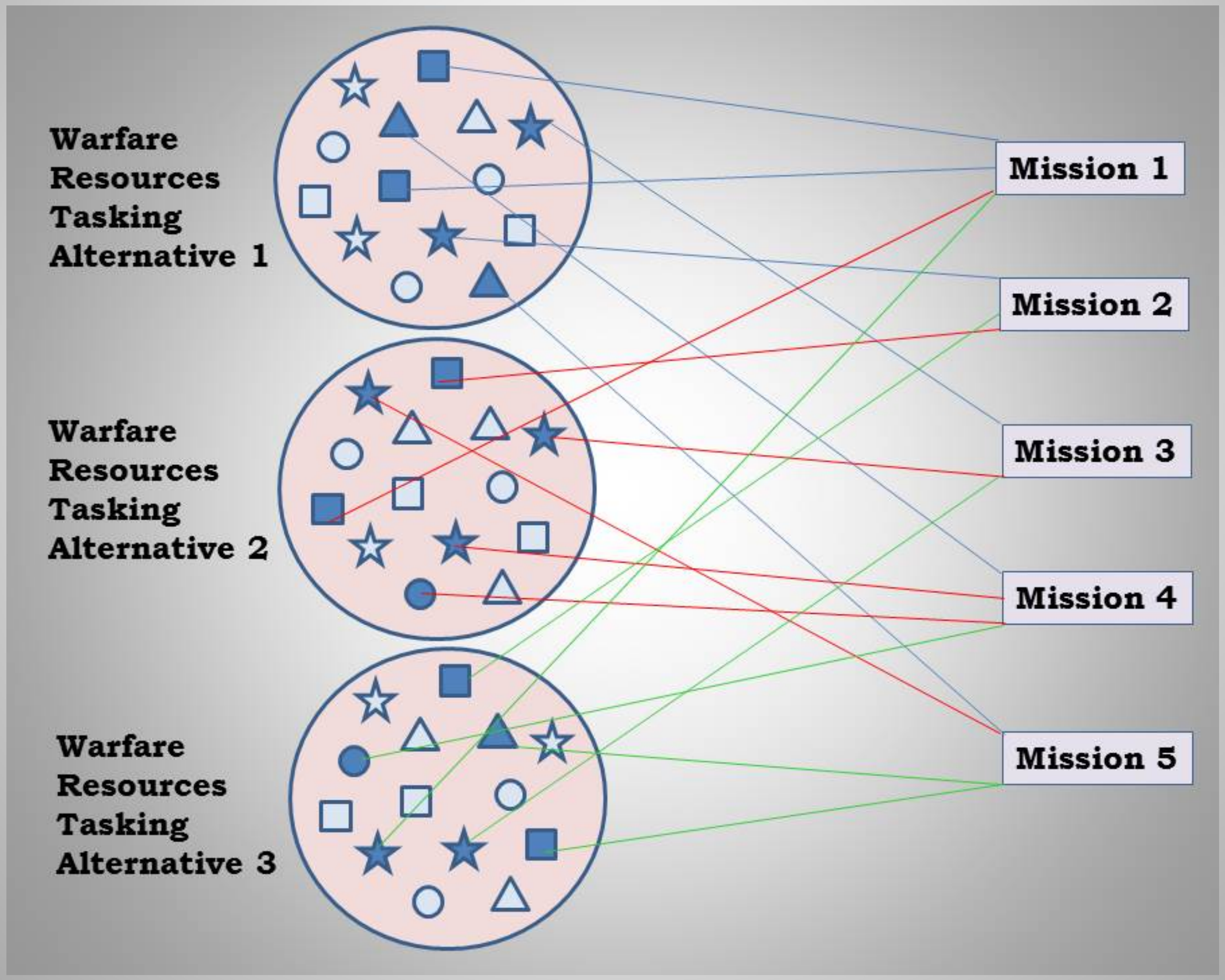
Examples of Resource Tasking



Hierarchy of Performance Effectiveness



SoS Tasking Alternatives for Multiple Missions



Performance/OMOE "Decision Engine"

- The idea is that given an understanding of the performance of each system, an automated "decision engine" could generate tasking alternatives (assigning systems to collaborative SoS's) and compute OMOE values for each SoS alternative to support optimized SoS "design" decisions.
- Self-organizing SoS: this could be taken one step further to enable each system to determine if it's participation in a SoS increases the SoS OMOE value. (If so, a decision to collaborate could be made.)

Resource Management Decision Assessments

Performance
OMOE Decision Engine

“Cost”
Decision Cost Engine

“Risk”
Decision Confidence Engine

Cost Considerations for Resource Management

- **Operational Costs** – defensive weapons, fuel, power
- **Maintenance Costs** (due to usage) – preventive maintenance, spares, repairs
- **Safety Costs** – manned vs. unmanned

Remember! For RM, the systems are already developed and paid for—so cost is treated differently

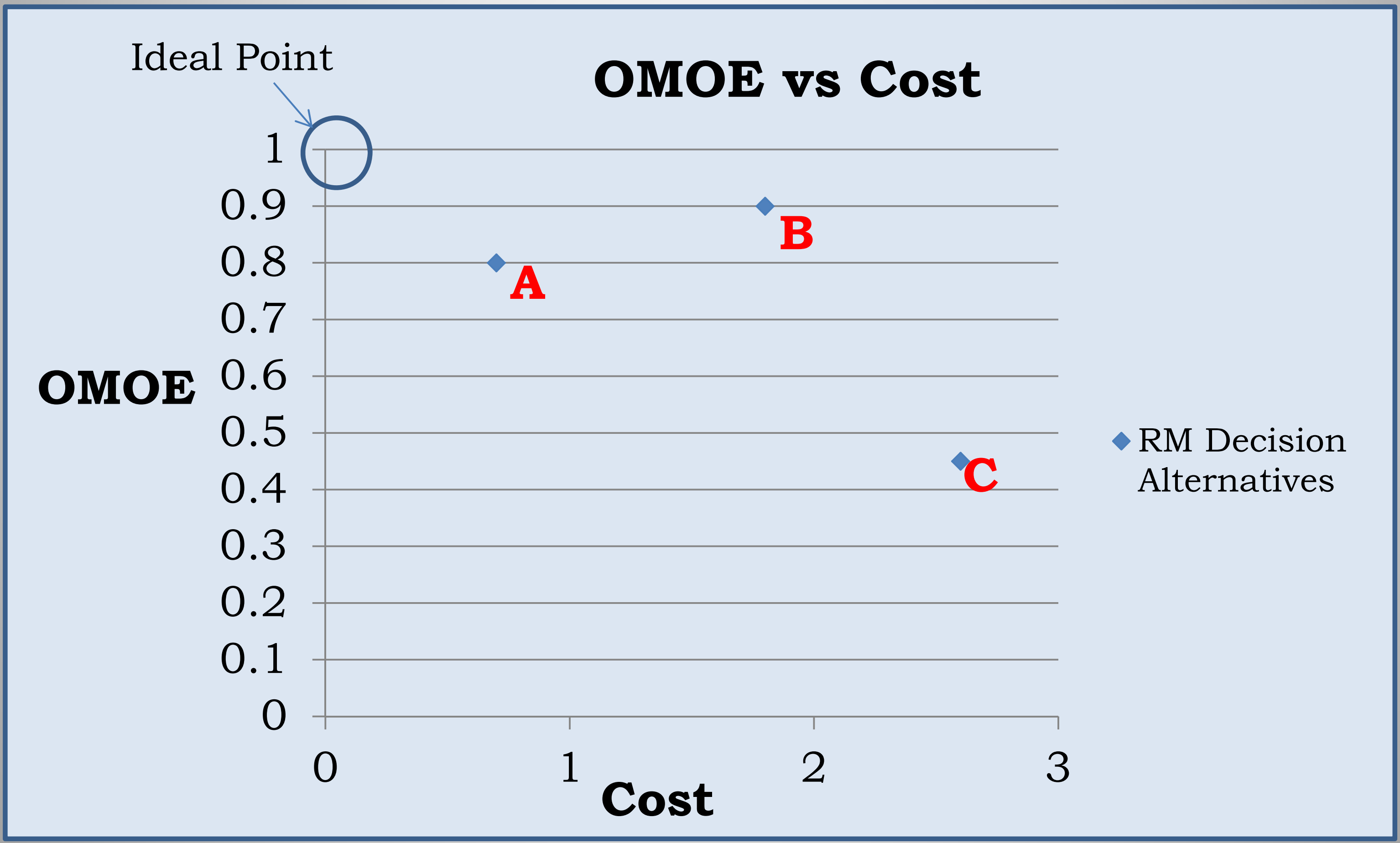
Decision Cost Engine Concept

- Provides methods to quantitatively represent the cost associated with the use of each warfare resource
- May provide relative cost levels or values
- Relative values are used to further refine the overall relative ranking of resource tasking decision alternatives

Decision Cost Engine: 3 Concepts

1. **"After the fact"** – shifting OMOE scores up or down based on relative cost levels
2. **"Red Flag"** – associating an "identifier" with very costly warfare resources to highlight decision alternatives that include their use
3. **"Hierarchical Weightings"** – the most comprehensive approach would assign cost ratings to all resources and weightings to compute an overall "cost" for each decision option

Combining Performance and Cost Assessments



Resource Management Decision Assessments

Performance
OMOE Decision Engine

“Cost”
Decision Cost Engine

“Risk”
Decision Confidence Engine

Decision Confidence Engine

- Determines a “level of confidence” associated with each resource tasking option
- Based on:
  - Information reliability (or “goodness”)
  - Data fusion performance
  - Sensor error
  - Communication error
  - Computational error
  - Mis-associations, incorrect identifications, dropped tracks, poor track quality, etc.

Sources of Decision Error

- Sensor Observations (SO)
- Communications (C)
- Data Fusion Processing (DFP)
- Association (A)
- Attribution (At)
- Identification (Id)
- Threat Prioritization (TP)
- Mission Identification/Prioritization (MP)
- Resource Information (Health, Status, Configuration, Location, etc.) (RI)

Notional Decision Confidence Level:

$$P_{\text{Decision Accuracy}} = P_{SO} * P_{C} * P_{DFP} * P_{A} * P_{At} * P_{Id} * P_{TP} * P_{MP} * P_{RI}$$

Decision Confidence Engine (continued)

- Hierarchical probability model – that includes all possible sources of error
- As the operational situation changes, model is updates with error estimates
- Errors are summed hierarchically to calculate an overall confidence level for each resource tasking option

Summary Comparison

Decision Assessment for System Design	Decision Assessment for RM Operations
System is in design phase	Systems are in operation
To select the most operationally effective design	To select the most operationally effective SoS/resource tasking
Single decision	Continuum of decisions
Projected performance against operational mission requirements	Projected performance against actual operational missions/threats
Cost in terms of estimated $ for acquisition and total lifecycle	Cost in terms of known cost to operate & maintain; safety
Risk in terms of ability to meet requirements	Risk in terms of decision uncertainty or level of confidence

Conclusions

- A decision framework providing decision assessment methodologies can address the complexity involved in effective resource management for tactical operations.

- Applications from Systems Engineering provide methods for operational performance, cost, and risk assessments of resource tasking alternatives.

- Future command and control stands to benefit from adopting a decision paradigm in addition to the traditional data-focused perspective.

Future Work

- Objective hierarchy modeling
- Techniques for generating resource tasking alternatives
- Continued development of the OMOE decision engine, cost decision engine, and decision confidence engine
- Designing warfare resources with an emphasis on being “taskable” and having “multiple uses”