Enterprise Architecture and COTS-Intensive System Acquisition Strategies

Diane Mularz, Jim Smith, Duane Hybertson

{mularz, dhyberts}@mitre.org, jds@sei.cmu.edu

January 2003

The opinions presented here are those of the authors, and do not represent the position of the MITRE Corporation, the Software Engineering Institute, or their sponsors.

Carnegie Mellon Software Engineering Institute

MITRE CI2

Topics

- The Challenge: Modern system acquisition forces and their implications
- An Approach: EPIC - A modern process for reconciling COTS product approaches with the architecture-based acquisition
- Strategies: Representative approaches and issues
- Summary

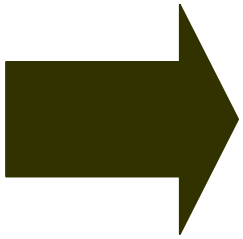
Carnegie Mellon Software Engineering Institute

MITRE CI2

Modern System Acquisition Forces and Their Implications

Forces

- **Keep pace with changing business demands**
  - Unpredictable threats, risks, economic conditions, rapid mission changes, changes in major players and organizations, multi-enterprise missions, business processes changing to accommodate new models of business,……
- **Keep pace with changing technologies and products**
  - Not just infrastructure anymore; broad application level products with applicability to government problem space
  - Ever-changing market options based on demands of users



Implications

- **Framework for technology and implementation decisions required:**
  - Enterprise architecture (EA)-based acquisition
  - Ensures technical solution aligns with changing business needs
- **Leverage commercial investments in products and technologies:**
  - COTS-based systems (CBS) solution space
  - Enables rapid alignment with market offerings

Carnegie Mellon Software Engineering Institute

MITRE CI2

Reconciling Divergent Pressures

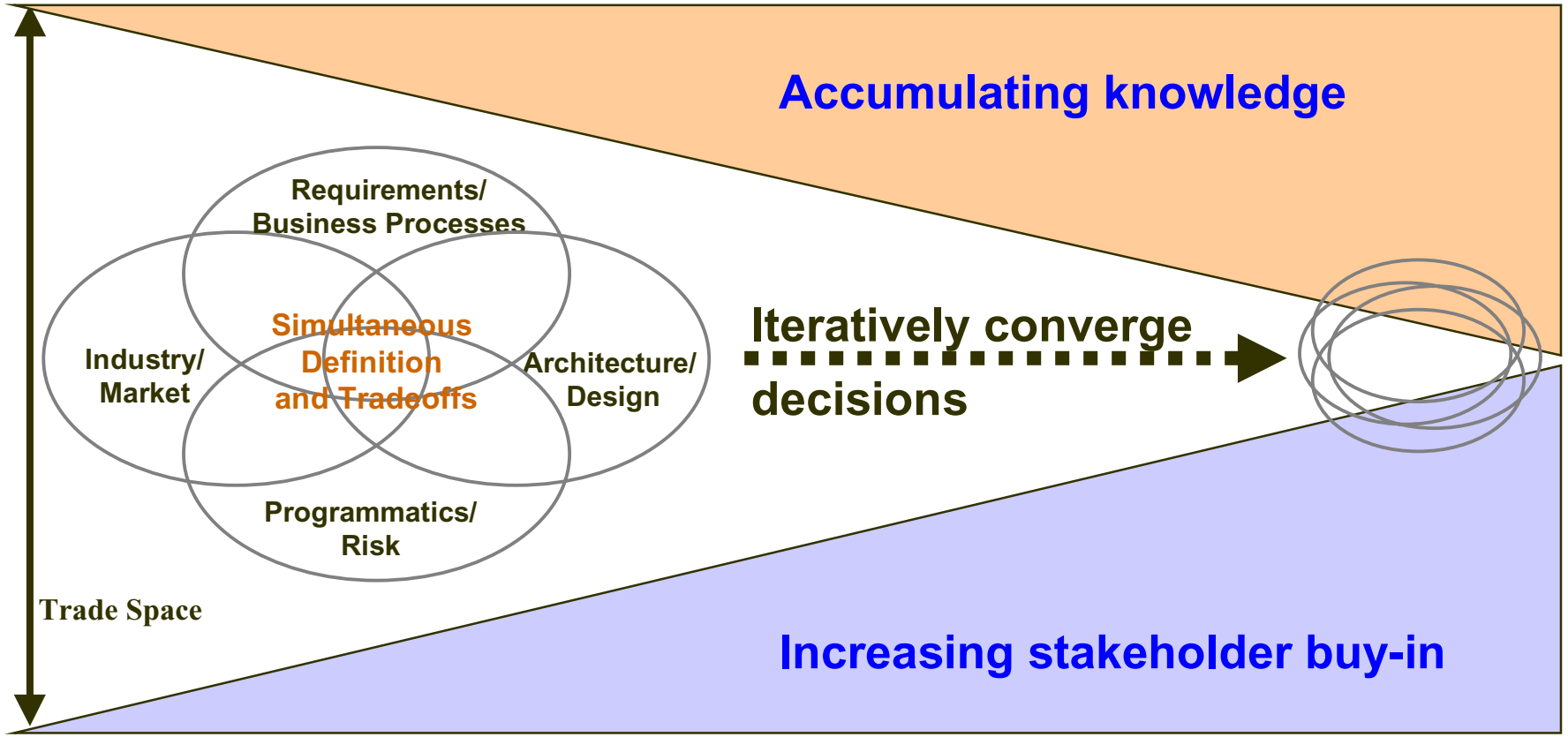
- **Enterprise architecture (EA) and COTS-based systems (CBS) tend to drive solutions along divergent paths:**
  - Enterprise Architecture-based acquisition
    - Must consider business needs and processes of the enterprise as drivers for technical solutions
    - Must stay aligned with changing requirements and business models
  - COTS-intensive solution space
    - Must maintain awareness of marketplace
    - Must define a flexible architecture that can exploit latest market offerings
    - Focus is on integration vs. development

Reconciling these divergent pressures requires an evolutionary process that supports simultaneous trades across business needs, market offerings, and architecture tempered by risks: EPIC

Carnegie Mellon Software Engineering Institute

MITRE CI2

EPIC: An Evolutionary Process for Integrating COTS-based Systems

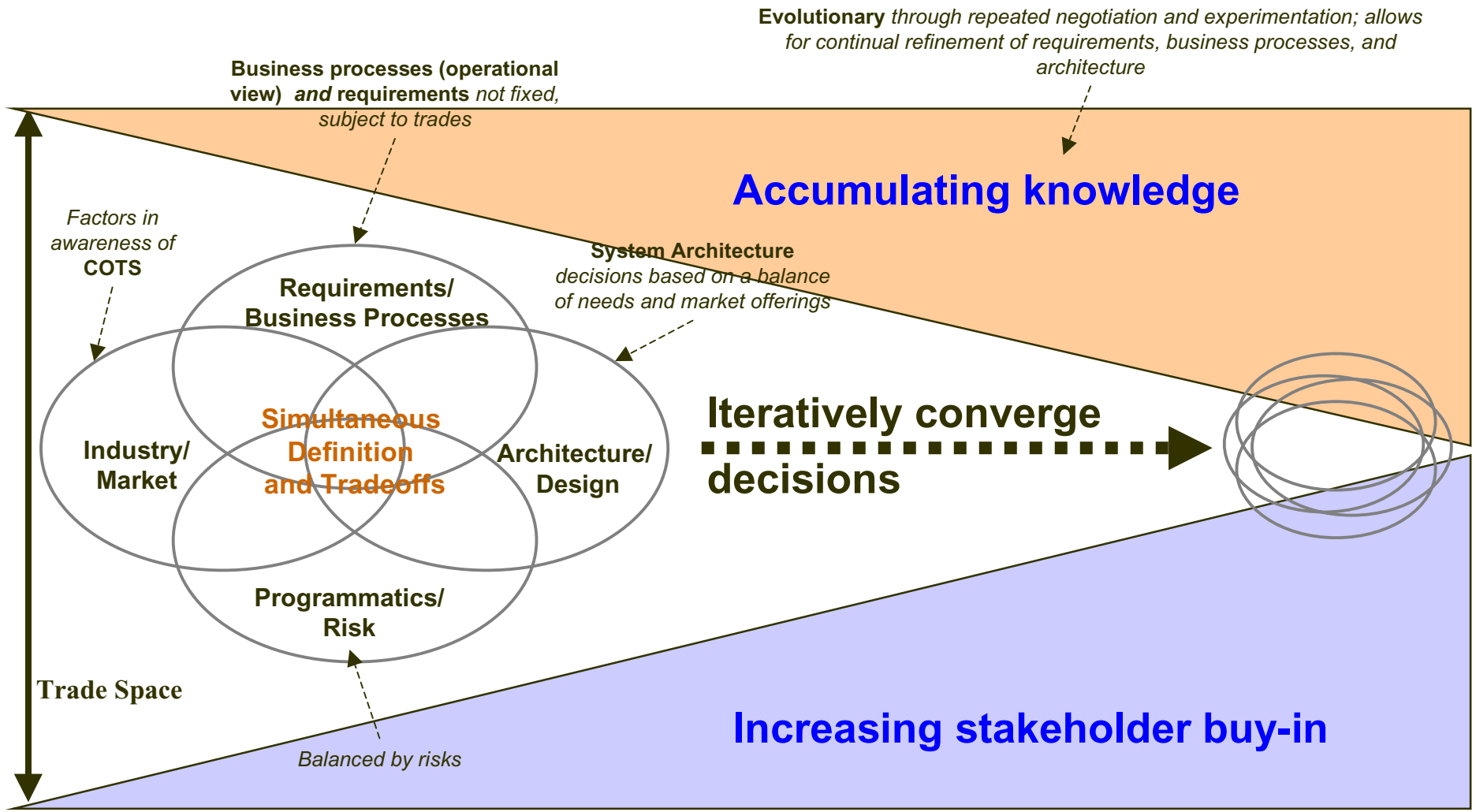


From 'Evolutionary Process for Integrating COTS-Based Systems (EPIC)' SEI, TR-2002-005, November 2002

Carnegie Mellon Software Engineering Institute

MITRE CI2

EPIC Aligns With Modern Business Realities



Carnegie Mellon Software Engineering Institute

MITRE CI2

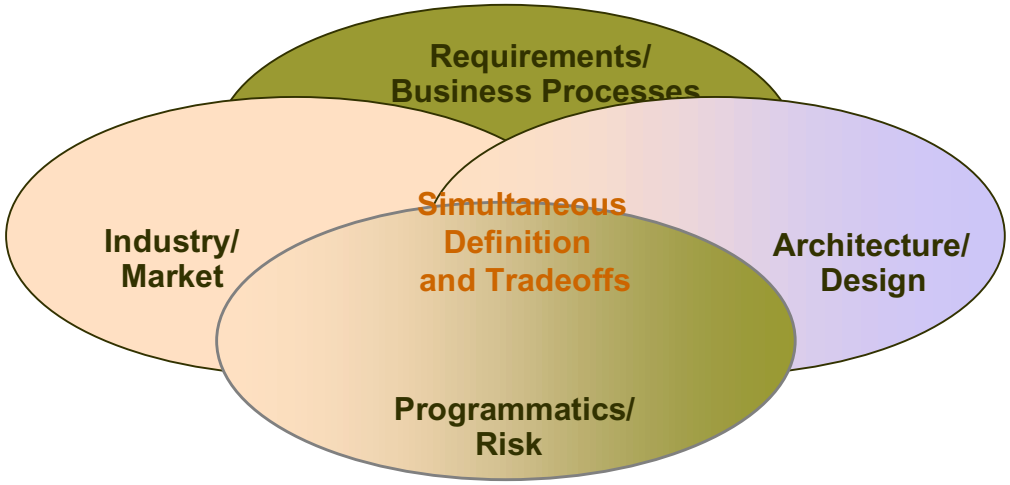
Possible Acquisition Strategies

- There are infinitely many possible programmatic, contractual, etc., strategies to accomplish this: *there is no one right approach*.
- No matter which strategy is employed, there are a number of decisions which much be addressed for a successful outcome.
- The following slides describe possible strategies based on the *allocation of execution responsibilities*, together with a brief discussion of some of the trade-offs which need to be considered in the context of any program.

Execution-based Allocation Strategies Explored

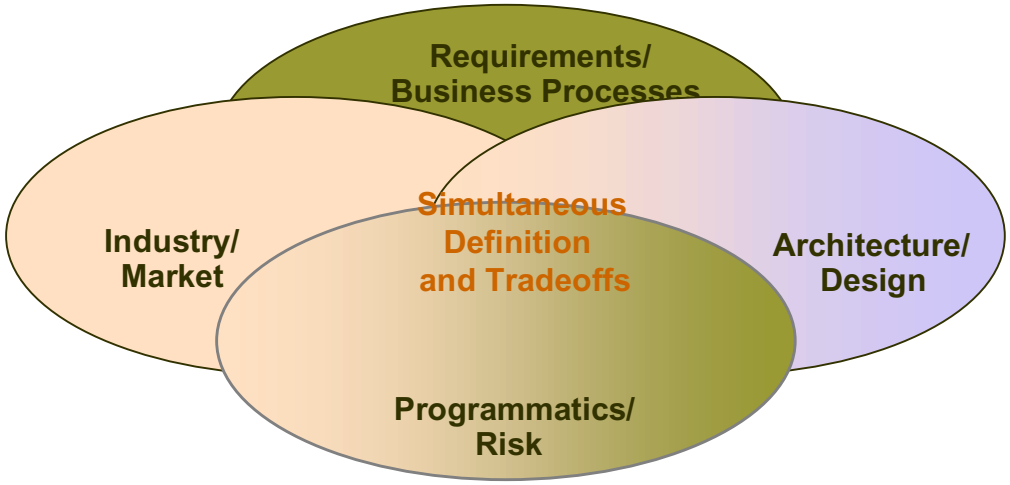
- **Three commonly-used strategies, based on different allocations of execution responsibility, are presented and discussed:**
  - **Strategy #1: "Functional" allocation**, with specific acquisition responsibilities assigned to discrete organizations (both Government and contractor)
  - **Strategy #2: "Project based" allocation**, where responsibilities are assigned according to the scope of the effort (e.g., enterprise, project "*x*," etc.)
  - **Strategy #3: "Site based" allocation**, where responsibilities are assigned on the basis of geographic "spheres of influence"

Strategy #1: Functional Allocation



- **Enterprise Architect**
  - Enterprise-level architectural/business process decisions (i.e., Scope and Enterprise levels of the Zachmann Framework, Levels I and II of the FEAF, or Operational Architecture views in the C4ISRAF)
- **System Developer**
  - System architecture (i.e., below the enterprise-level as defined above)
  - Market/technology forecasting
  - System implementation/spiral management/product selection/modernization decisions
- **Sustainment**
  - Maintenance of fielded systems

Strategy #1: Items for Consideration

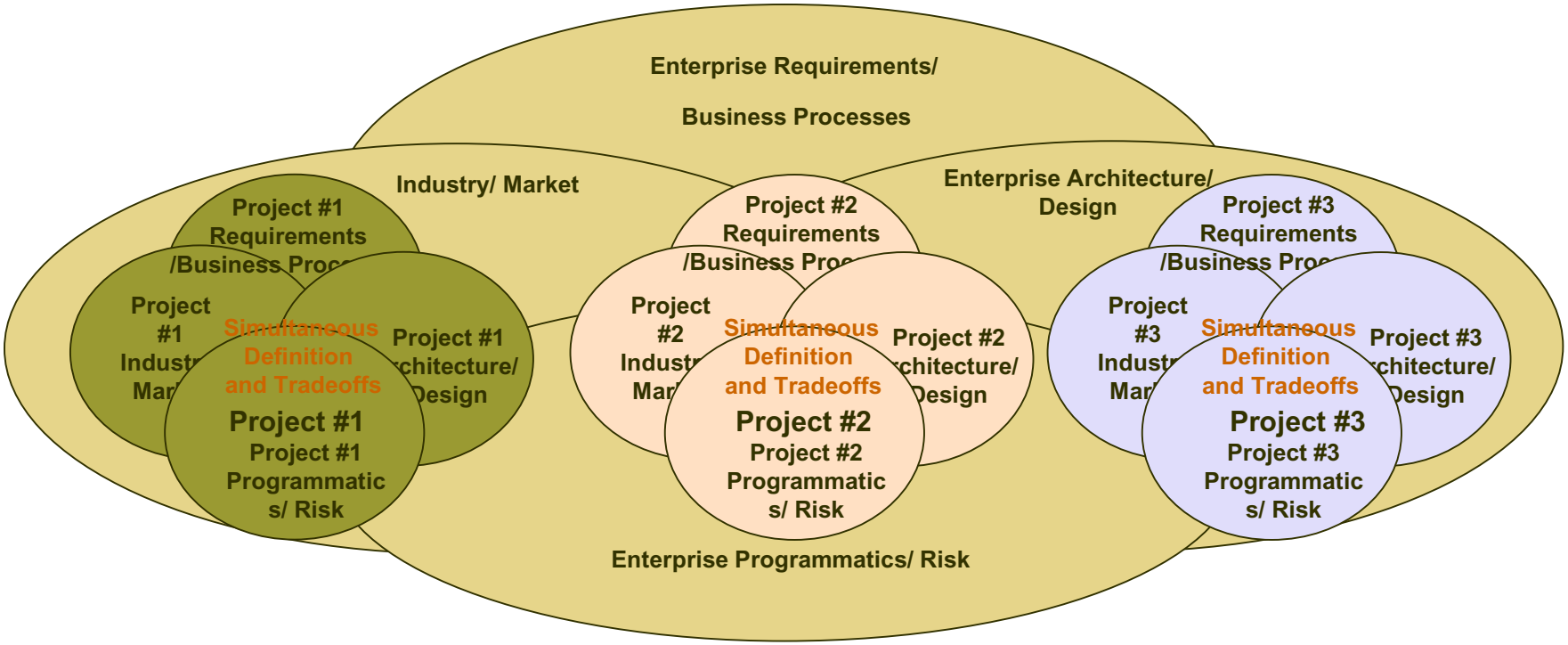


- Division of architectural responsibilities across organization/contract boundaries
- Reconciling evolving business processes across organization/contractual boundaries
- Integration/sustainment of continuously-evolving systems
- Incentives to "play nice"

Carnegie Mellon
Software Engineering Institute

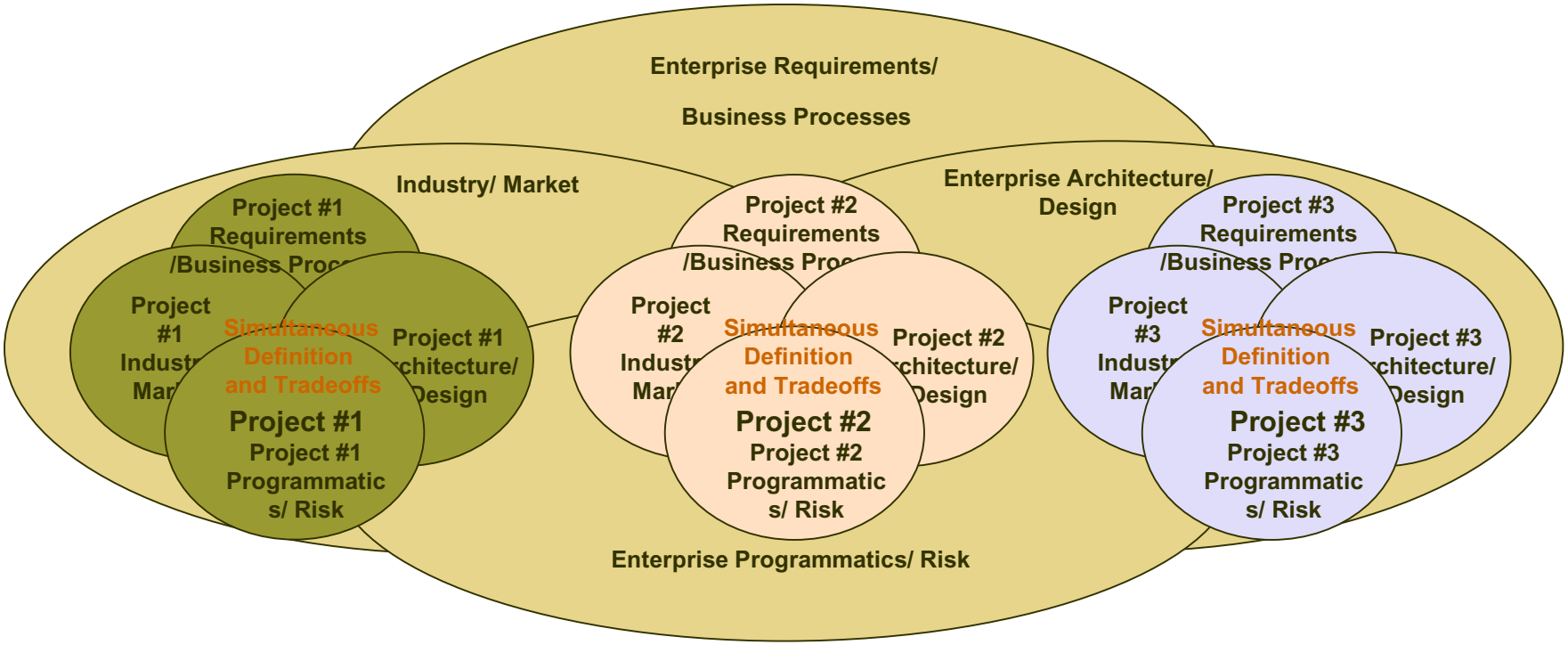
MITRE CI2

Strategy #2: Project-based Allocation



- **Enterprise architect**
  - Governs overall enterprise architecture and its realignment based on project demands/outcomes
  - Decides on projects to be developed, order of acquisition/development, and their degree of parallelism
- **Project Developers**
  - Each developer (Government entity, or contractor) is allocated requirements and business processes. Contractor has responsibility for project-specific requirements, business processes, architecture, market survey, standards, …with additional requirement to demonstrate that project is EA compliant

Strategy #2: Items for Consideration

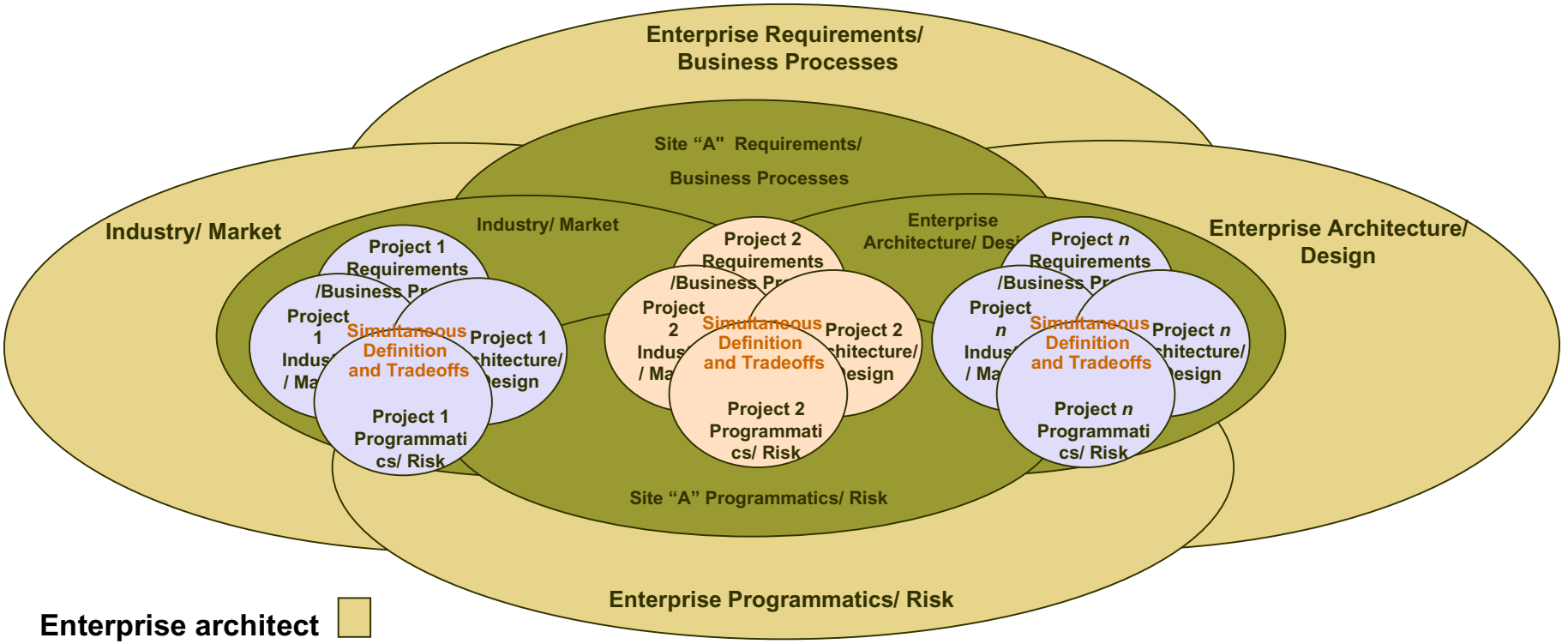


- Reconciling project "clashes" (e.g., business processes, architectural compliance, market selections, etc.)
- Maintaining EA compliance with continuously-evolving architecture, systems, requirements, etc.
- Clearly-defined roles and responsibilities
- Incentives to "play nice"

Carnegie Mellon
Software Engineering Institute

MITRE CI2

Strategy #3: Site-based Allocation

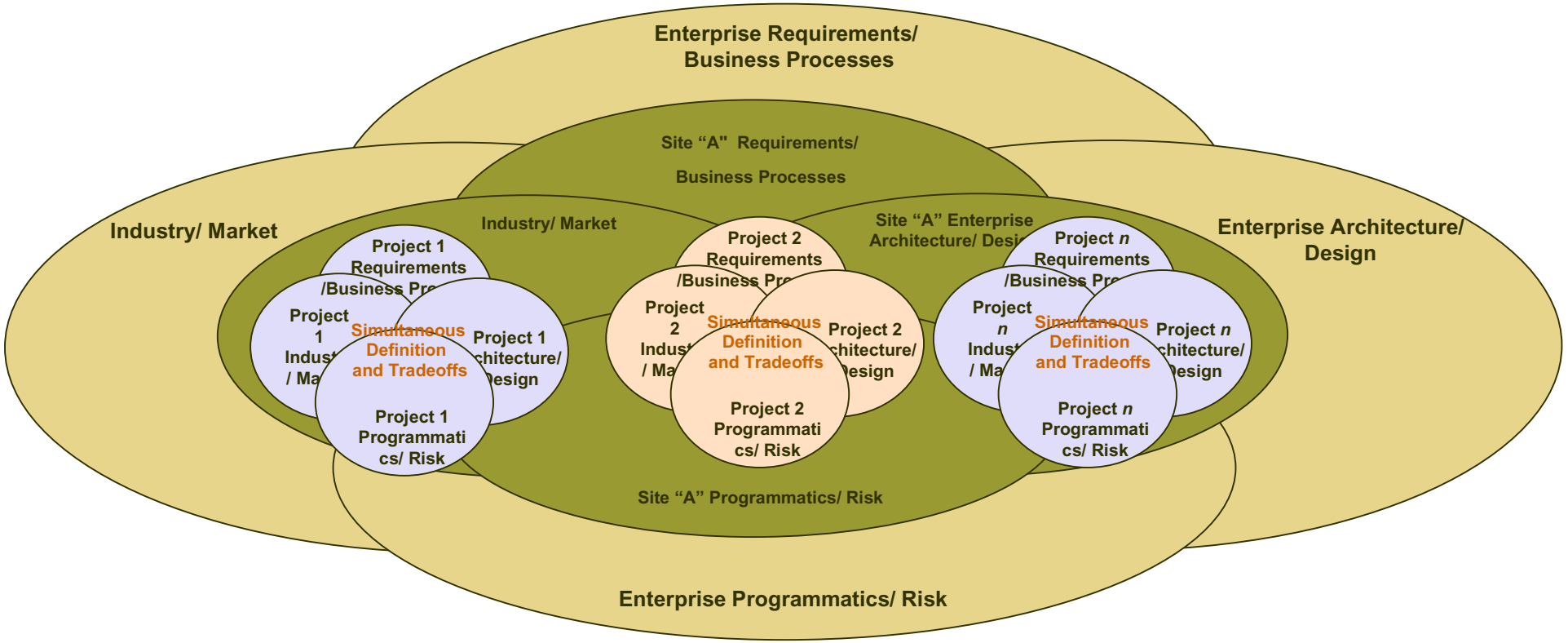


- **Enterprise architect**
  - Governs overall enterprise architecture and its realignment based on project demands/outcomes
  - Allocates site responsibilities to site integrators
- **Site integrator**
  - Each site integrator is allocated requirements and business processes. Responsibility for site-specific requirements, business processes, architecture, market survey, standards, etc., with oversight of site projects to ensure EA compliance
- **Project developers**
  - Responsible for development and sustainment of systems under site integrator direction

Carnegie Mellon Software Engineering Institute

MITRE CI2

Strategy #3: Items for Consideration



- Maintaining EA compliance across multiple sites
- Synchronizing architectural/business process/requirements changes across multiple sites
- Clearly-defined roles and responsibilities
- Incentives to "play nice"

Carnegie Mellon
Software Engineering Institute

MITRE CI2

Summary

- COTS and EA have the potential to ensure flexible architectures that can adapt to changing business needs and the marketplace, HOWEVER,
- COTS-based systems require iteration and negotiation across multiple spheres of influence, THEREFORE
- Allocation of responsibilities to each of those spheres can help or hinder the advantages of COTS and EA as acquisition strategies