An Introduction to Influence Maps: Foundations, Construction, and Use

Jim Smith
NDIA Systems Engineering Conference
October 29, 2009

Software Engineering Institute | **Carnegie Mellon**

Overview

This presentation will provide an overview of Influence Maps (IMs), a graph-based technique—central to Influence Mapping Analysis (IMA)—for understanding system-of-systems interoperability issues
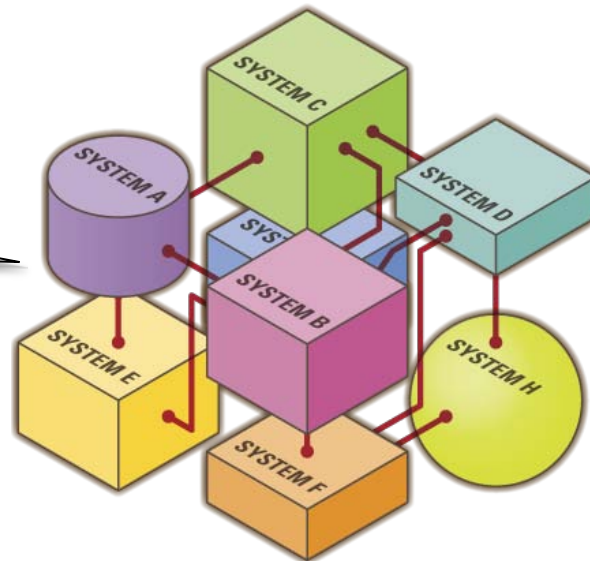
Topics include:

- Foundations for IMs
- Construction and use of IMs for governance- and acquisition-related interoperability risks

Multiple Perspectives on System of Systems -1

SoS are a collection of integrated and interoperable hardware and software entities providing capabilities that fulfill specific functional and operational needs
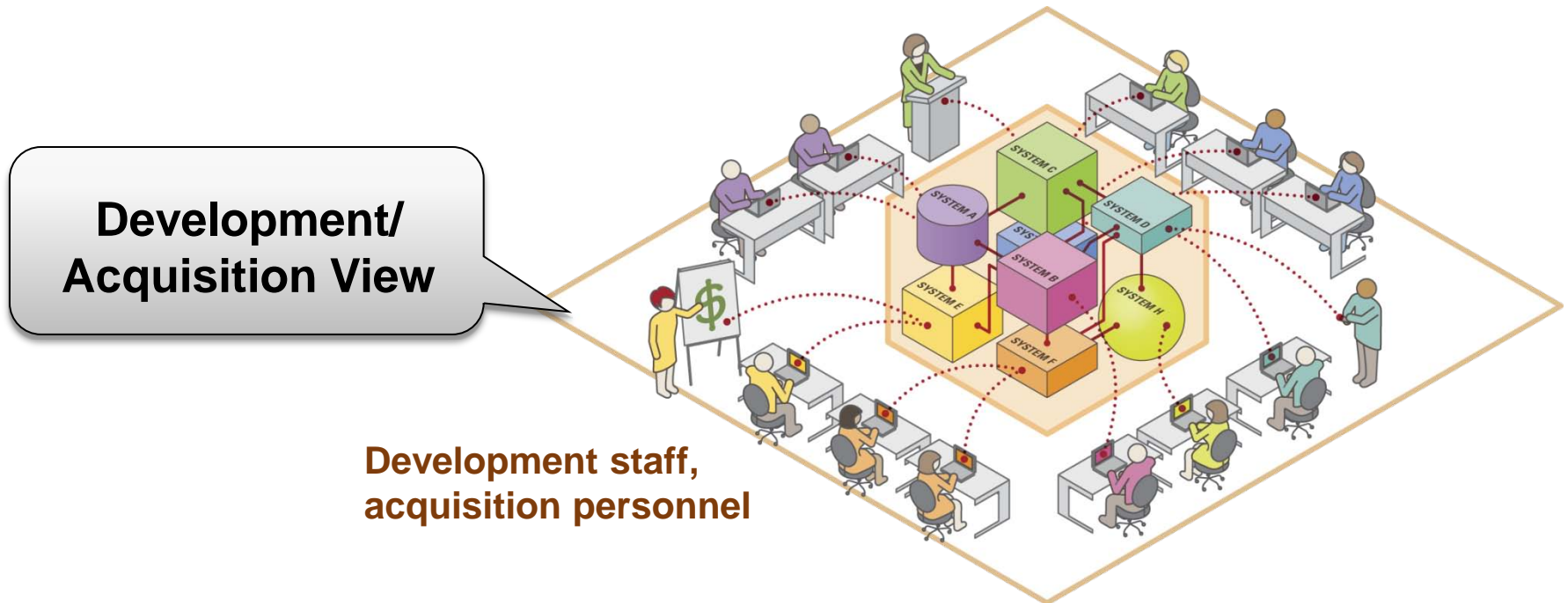
Technical View

But…systems of systems are more than interoperating hardware and software systems

3

Multiple Perspectives on System of Systems -2

An SoS is a collection of people and organizational entities involved in acquiring and composing "systems of systems" that provide capabilities to fulfill specified functional and operational needs

Development/ Acquisition View

Development staff, acquisition personnel

People systems are as important as technical systems

Software Engineering Institute | **Carnegie Mellon**

Multiple Perspectives on System of Systems -3

SoS provide capabilities that enable a collection of operational users to achieve the effects they need to meet their business/mission goals
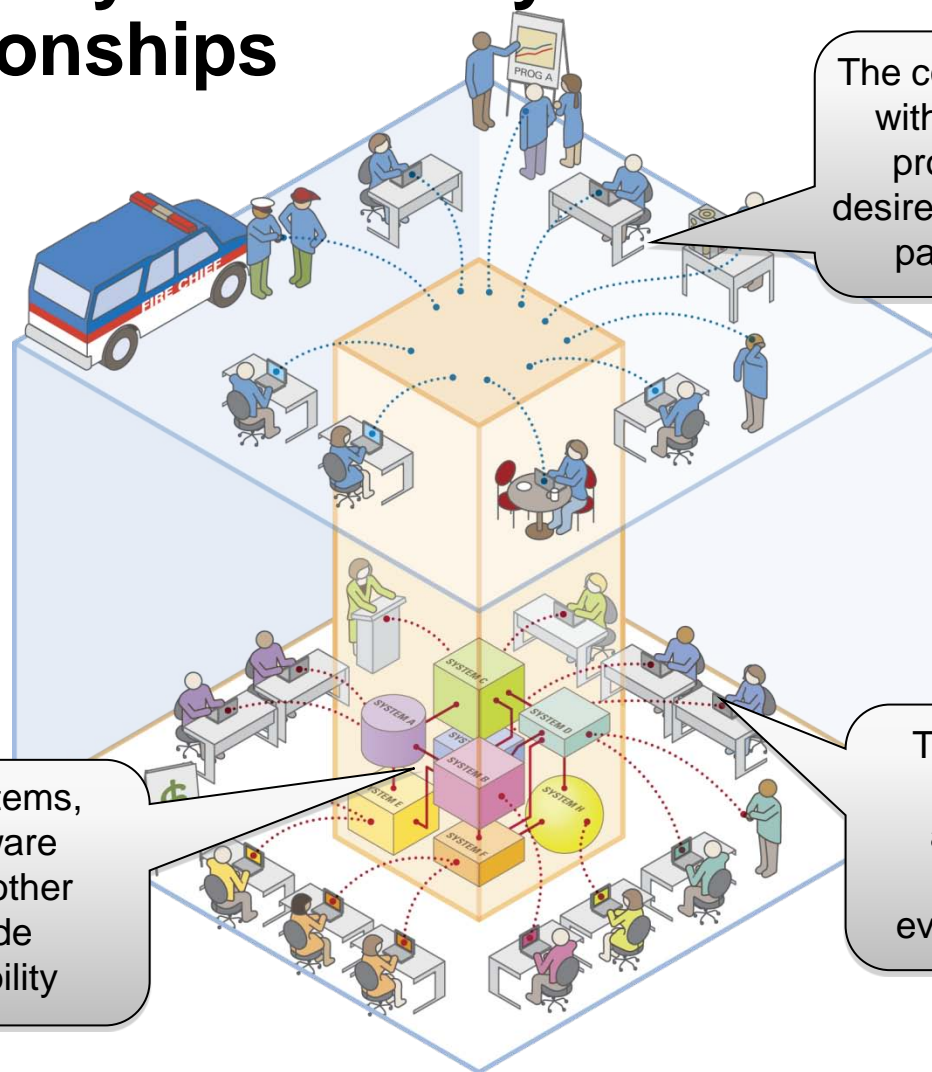
- Evolves to enable dynamically changing operational effects within the operational user's context of use

- Is likely to use technical and organizational assets outside of the original design context
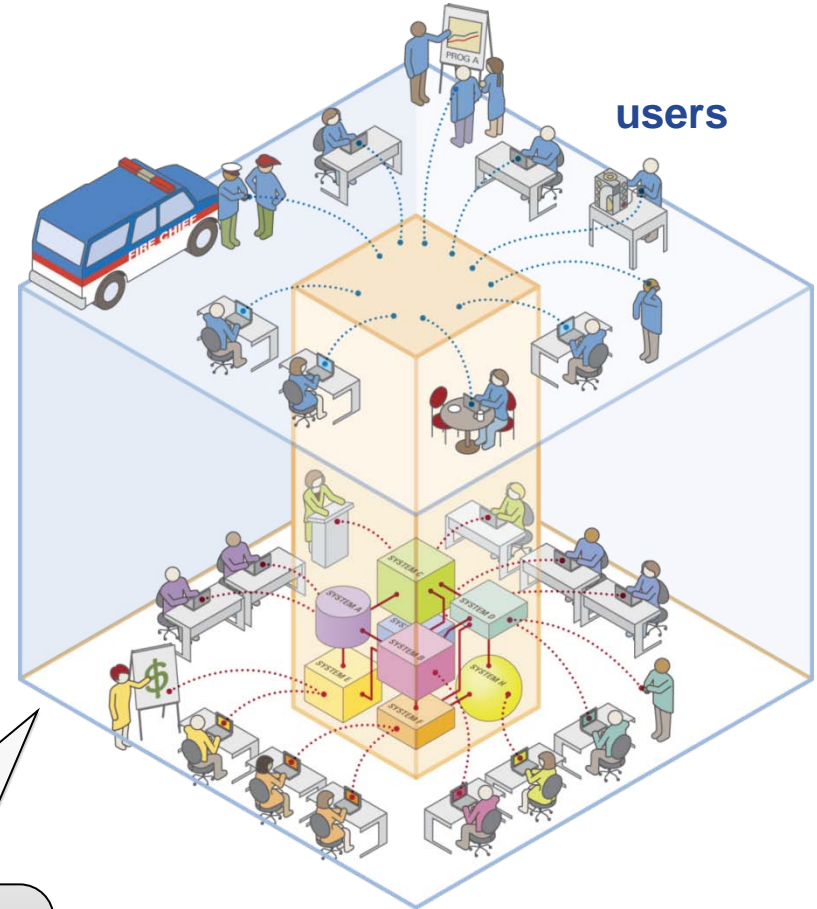
users

Operational Effects/ User View

Key Point: Systems of Systems Result from Interrelationships

The composition of capabilities with users and operational processes that achieves desired operational effects for a particular context of use

Aggregation of systems, hardware or software components, and other devices to provide operational capability

The people, organizations, and interrelationships associated with building, acquiring, fielding, and evolving systems of systems

Software Engineering Institute | Carnegie Mellon

Understanding Interrelationships and Influence

Various techniques exist to represent interrelationships in socio-technical systems, including:

- Network diagrams
- Functional Flow Block Diagrams
- Conceptual Graphs
- IDEF0/IDEF3
- PERT Charts

Challenges in applying to systems of systems

- Complexity of resulting representation
- Difficulties in representing/reasoning about "background" knowledge
- Problems in representing/reconciling conflicting/contradictory influences

Influence Maps (IMs)—and IM Analysis (IMA)—provides a simple way to identify, understand, and analyze influence interrelationships

- Permits the discovery of influences that impact governance, acquisition, and engineering for systems of systems
- Supports the identification, characterization, and reconciliation of ambiguous and contradictory influences
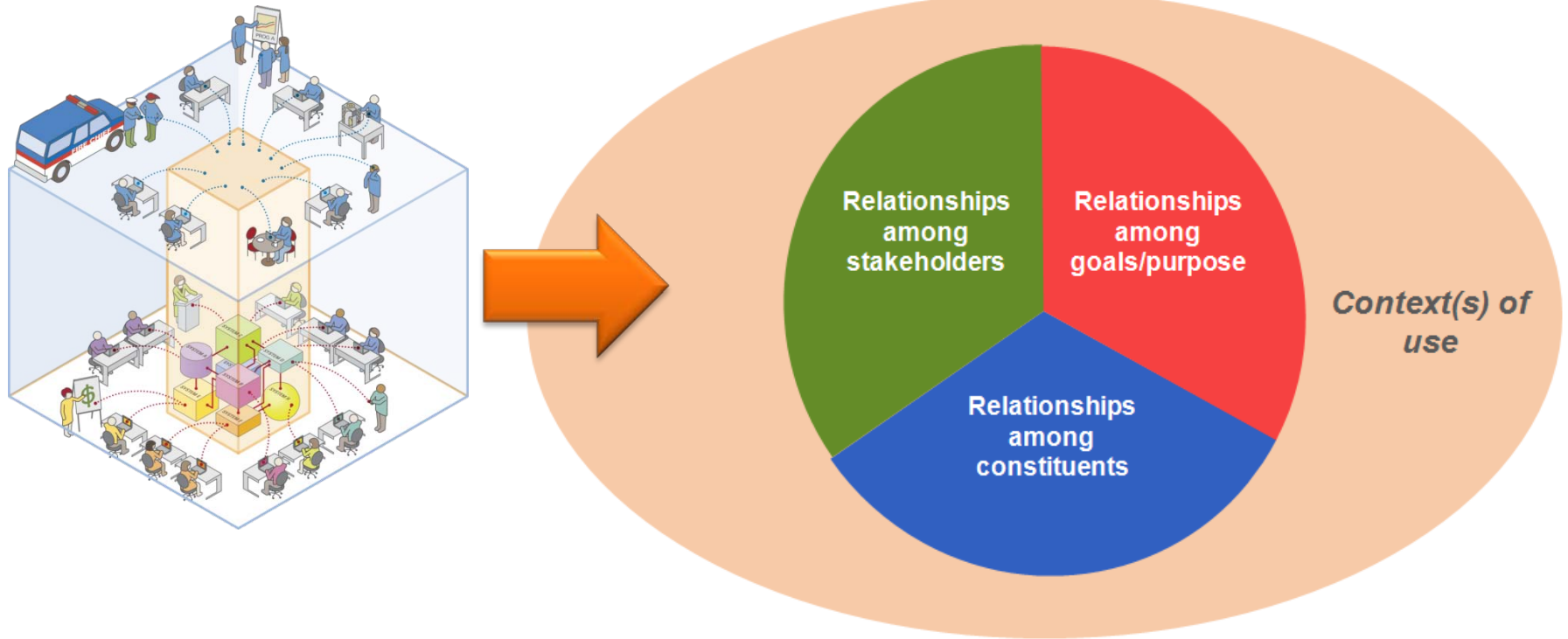- Input to formal analysis/reasoning framework and decision aids

Key Concepts of IMs and IMA

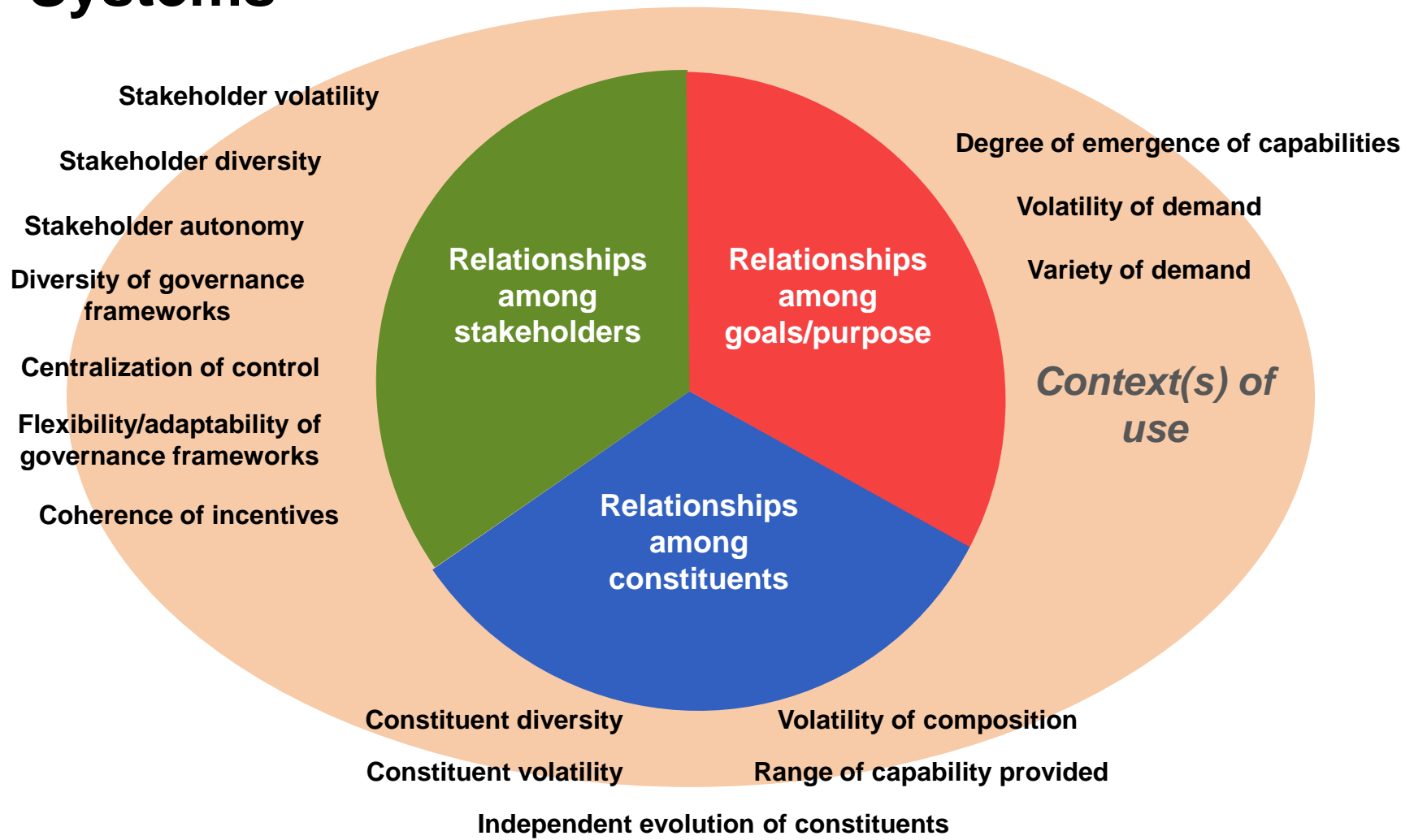Influence mapping analysis is built around several key concepts

- Use of IMs to identify, characterize, and understand influence relationships
- Resolving divergent perceptions of the actual conditions: the so-called "ground truth"
- Analyzing patterns of influence relationships for indicators of SoS risks
- Use of a contextually-driven discovery process

Understanding the Relationships Implied by the SoS Perspectives

Relationship Characteristics of Systems of Systems

Stakeholder volatility

Stakeholder diversity

Stakeholder autonomy

Diversity of governance frameworks

Centralization of control

Flexibility/adaptability of governance frameworks

Coherence of incentives

Relationships among stakeholders

Relationships among goals/purpose

Relationships among constituents

Degree of emergence of capabilities

Volatility of demand

Variety of demand

Context(s) of use

Constituent diversity

Constituent volatility

Volatility of composition

Range of capability provided

Independent evolution of constituents

Forms of Interrelationships

The interrelationships in socio-technical systems—and the workings of their influence—vary widely:

- Contract language
- Statutory/regulatory requirements
  - Defense Appropriation Act
  - HIPAA
  - Federal Acquisition Regulations
- Technical requirements
- Reporting requirements
- "Giver/receiver" relationships
- Funding
- Individual trust

Managing Divergent Perceptions -1

SoS participants can have a different understanding of relevant influence relationships

- Important influence relationships are often implicit, or only tacitly acknowledged

These inconsistencies can—and frequently do—lead to unfortunate technical and programmatic decisions that result in:

- Cost growth
- Schedule delays
- Performance shortfalls
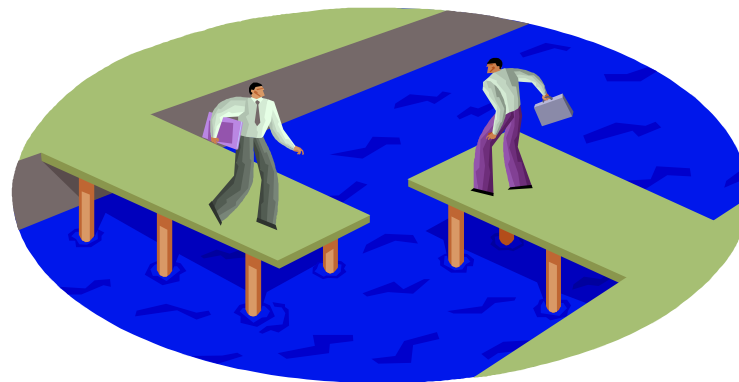
Managing Divergent Perceptions -2

Explicit versus tacit/implicit, "official truth" versus "ground truth"

- *Official truth* is reflected explicitly in various policy statements, organization charts, program plans, directives, memoranda, etc.
- Frequently at odds with real conditions (e.g., actual—versus ideal—programmatic relationships, "back channel" communications) that define the *ground truth*
- Much of this information exists as tacit or implicit knowledge

Before you can understand what is actually happening—and why—underlying assumptions and expectations must be made explicit

- What is the influence?
- Between what parties?
- For what purpose?

- How effected?
- With what assurance?
- As verified by?
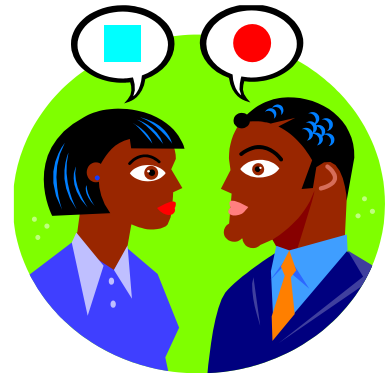
Patterns of Influence Relationships

Patterns of influence relationships can provide indications of potential risks

Examples

- Cycles, or loops
  (e.g., "A" has a schedule dependency on "B," which has a schedule dependency on "C," which has a schedule dependency on "A")

  – Can lead to programmatic "race conditions" because of the delay between the time that an event occurs (e.g., delivery date delayed by rework to correct problems discovered during testing) and when it becomes known to other participants

- Hidden—or indirect—dependencies
  (e.g., "A" has a schedule dependency on "B," which has a backwards compatibility relationship with "C," resulting in "A" having an indirect dependency on "C")

  – Can result in major impacts from seemingly unrelated decisions

Construction and Use of IMs for IMA

IM Analysis (IMA) comprises four steps:

1. Construct "strawman" IMs

2. Refine/extend IMs during discovery process

   – Create multiple node- and agreement-centric IMs representing relevant stakeholders' perspectives

3. Prepare composite IMs, and analyze for inconsistencies, gaps, clashes, patterns

4. Develop risk mitigation strategies

Three types of IMs:

Context-centric: Provides a high-level overview of the entire system-of-systems context, including all relevant participants

Node-centric: Represents influence relationships, as seen from the perspective of a single participant

Agreement-centric: Provides detailed representation of an individual influence relationship, including semantics

Strawman IMs

Context-centric IM provides a high-level, overview of entire system-of-systems context

- Includes major participants and influence relationships that comprise the key technical and programmatic drivers

Strawman context-centric IM based on documentation provided by subject organization and IMA team expert judgment

- Serves as basis for discovery process

- Could use outputs from a Critical Context Analysis (CCA) as an input

16

Example: Strawman Context-Centric IM
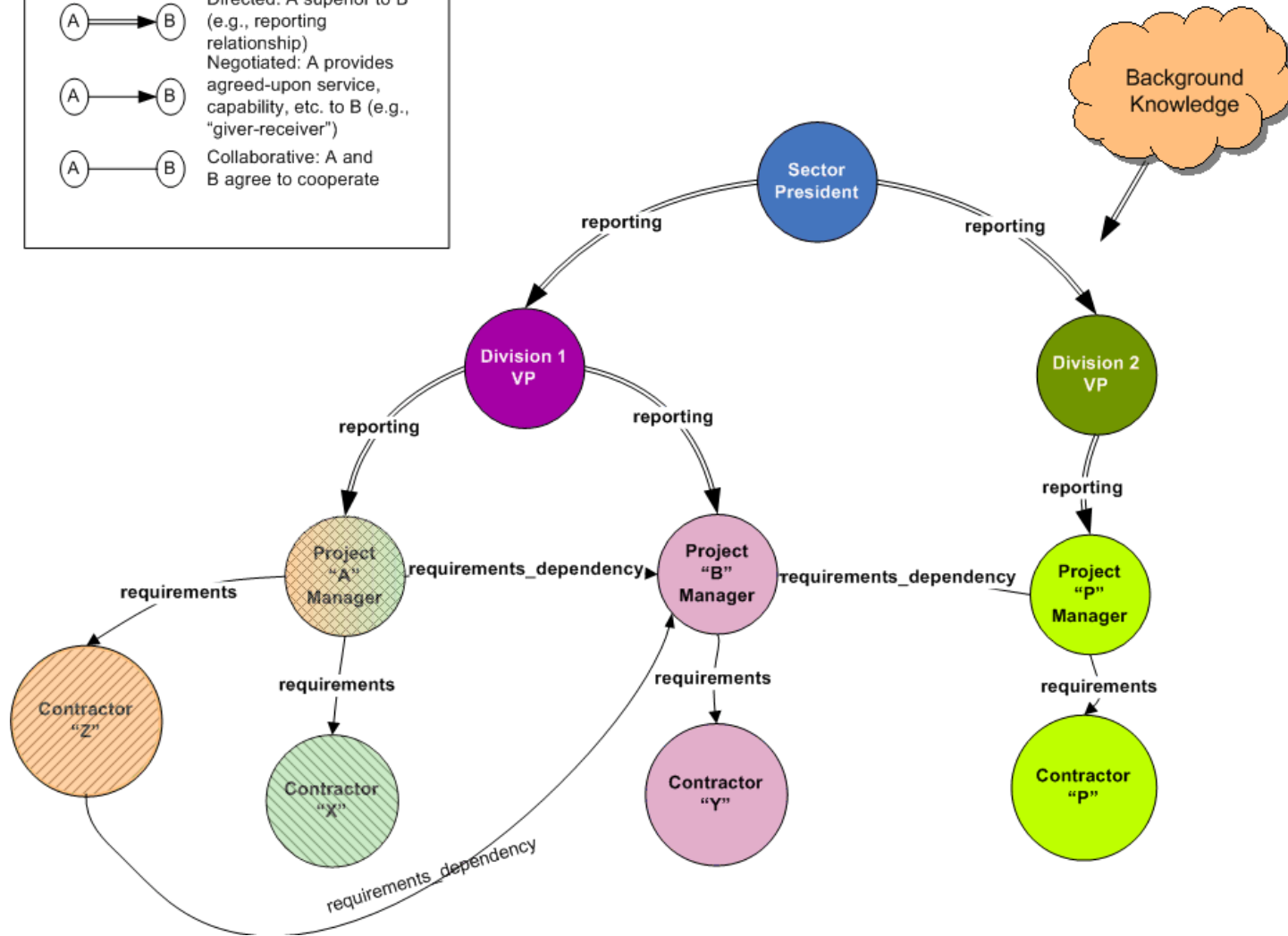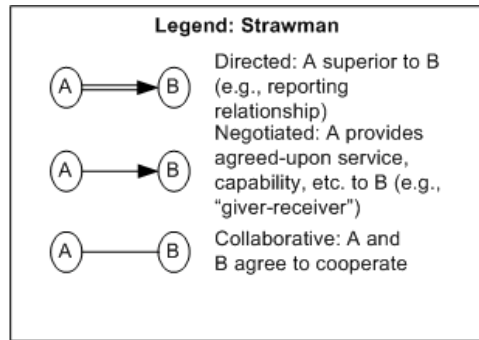
Discovery

Three goals of discovery process:

1. Identify the most critical, pacing requirements that drive the SoS context

2. To identify relevant internal and external stakeholders and characterize their key concerns, motivations, needs, etc.

3. To develop context-, node-, and agreement-centric IMs that reflect the "ground truth" for the relevant influence relationships

Contextually-Driven Discovery

Uses scenarios that relate to a participant's context within an SoS (e.g., acquisition program office, operational tester), augmented with influence relationship templates, to structure participant interviews

- Example: Your program's budget has been cut as a result of a Congressional Committee "mark." How do you evaluate the impact of this action on your ability to satisfy you program cost, schedule, and performance goals? Who do you interact with in making this determination? What information do you need to evaluate? etc.

Captures and characterizes assumptions about other stakeholder roles and responsibilities

Templates Support Contextually-Driven Discovery

Aids elicitation of influence relationships

- Different templates for different SoS perspectives (e.g., acquisition program office) and contexts (e.g., budget cut, schedule slip)
- Lists typical classes of nodes with which subject would reasonably be expected to have influence relationships (e.g., milestone decision authority, user community)
- Used to record types of relationships, how they are documented, their status, etc.

Serves as input to generation of "discovery" IMs

- Each row defines one or more influence relationships

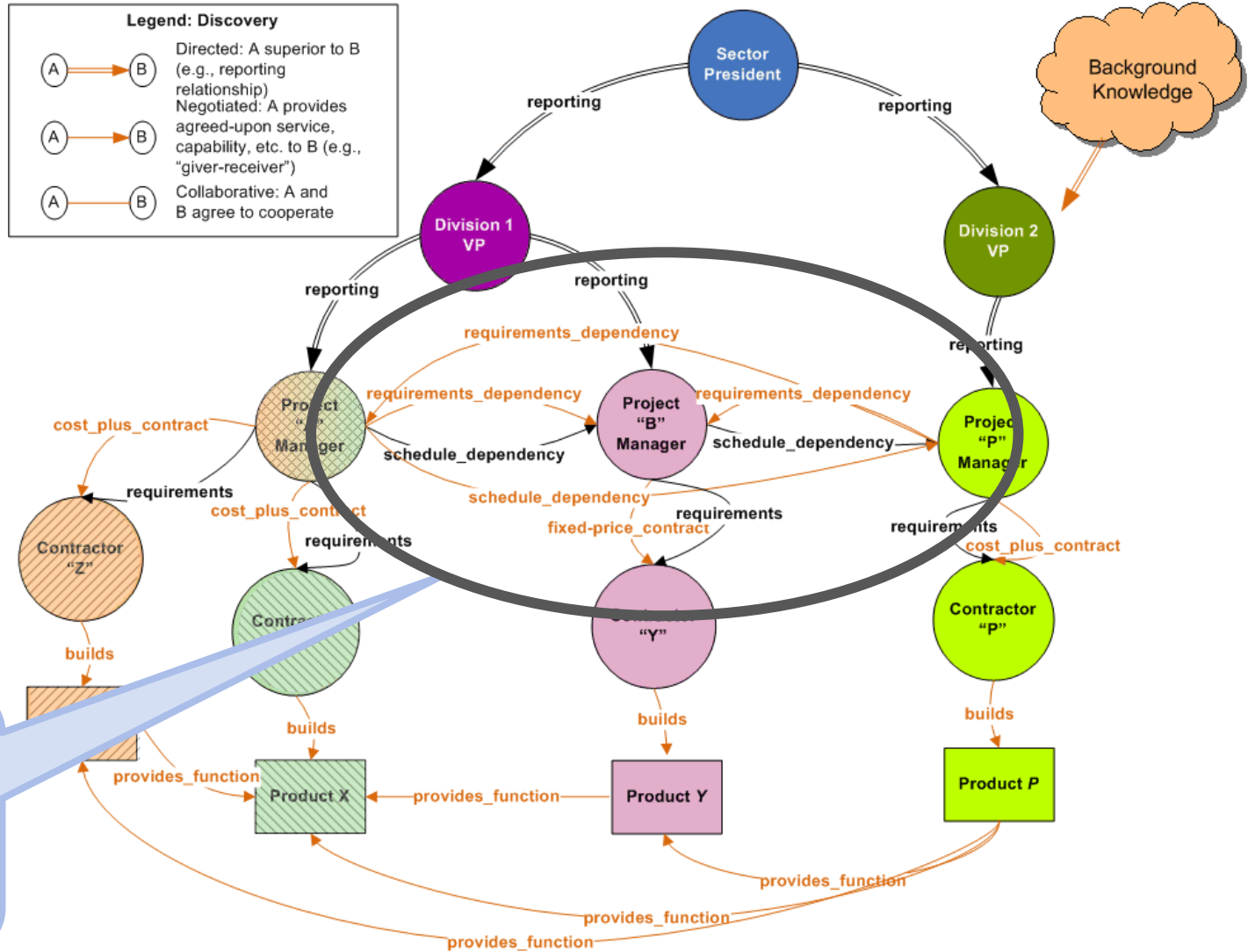# Context-Centric Influence Map

Provides a high-level, overview of entire system-of-systems context

- Includes major participants and influence relationships that comprise the key technical and programmatic drivers

Elaborates/updates the strawman context-centric IM based on information gained during interviews

Example: Context-Centric IM

Note greater detail and additional relationships when compared to strawman context-centric IM
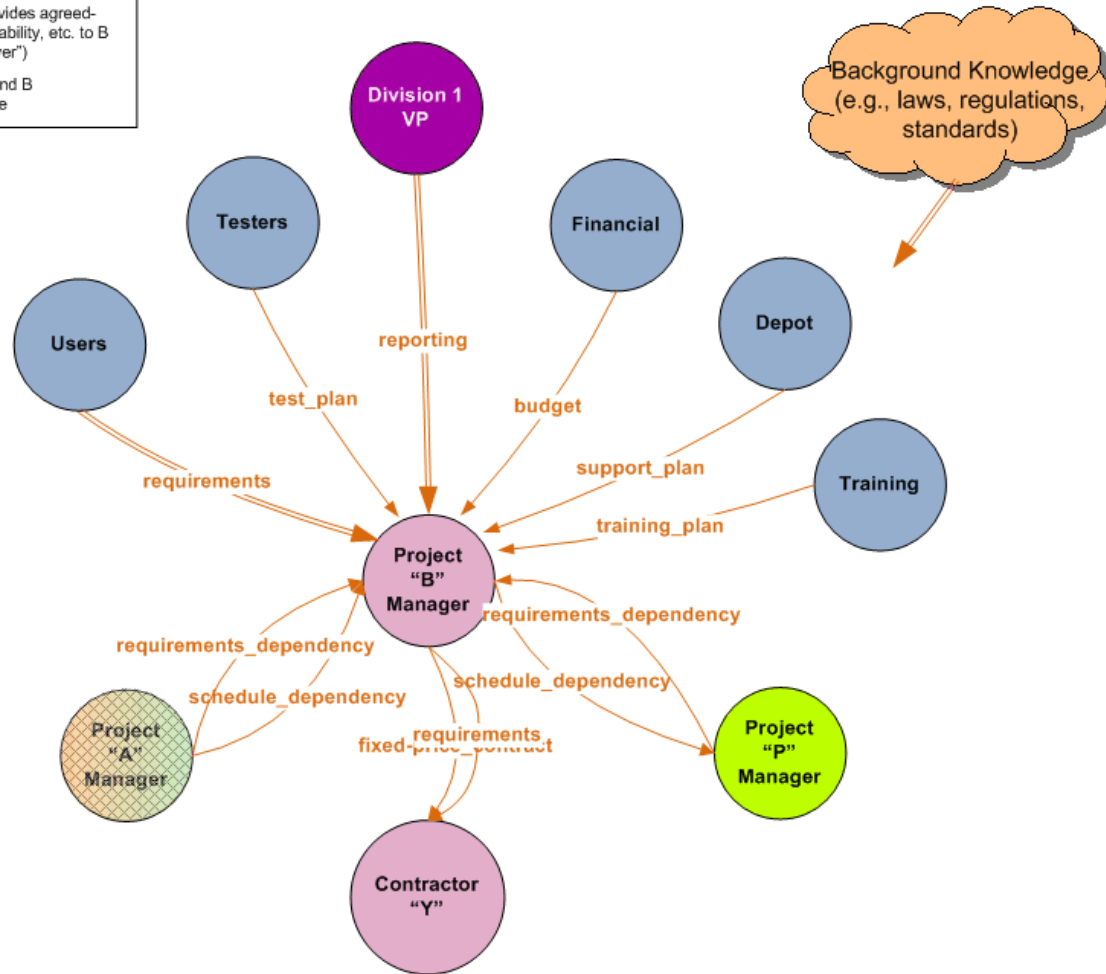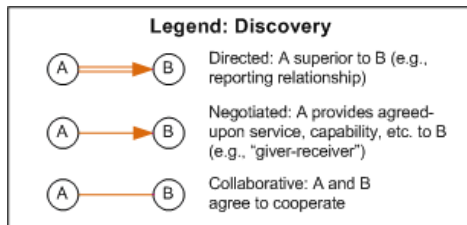
Node-Centric IM

Represents a view of immediate influence relationships (i.e., not via an intermediary agent/agency) from the perspective of a particular participant

Developed for each node (e.g., participant, organization) in a given SoS context

Example: Node-Centric IM
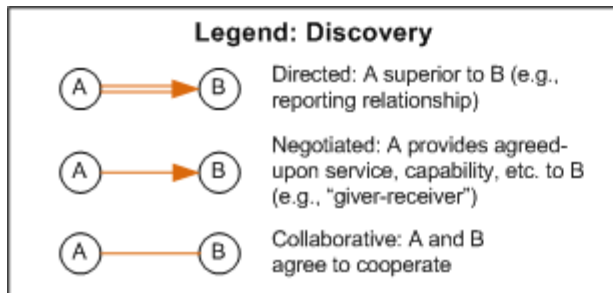
Agreement-Centric IM

Provides detailed representation of an individual influence relationship

- Includes semantics of the relationship
- Example: what—exactly—does "delivered" mean?
  - Installed and ready to turn on?
  - Or, shrink-wrapped, on a pallet, in some loading bay?

Developed for influence relationships identified as most critical to success in given SoS context

Example: Agreement-Centric IM

Legend: Discovery

A ⟶ B   Directed: A superior to B (e.g., reporting relationship)

A ⟶ B   Negotiated: A provides agreed-upon service, capability, etc. to B (e.g., "giver-receiver")

A — B   Collaborative: A and B agree to cooperate

Agreement:
- deliver_no_earlier_than
- deliver_no_later_than
- agreed_functionality
- agreed_quality_attributes
- agreed_assurance

Project "B" Manager ⟵ requirements_dependency — **Project "P" Manager**
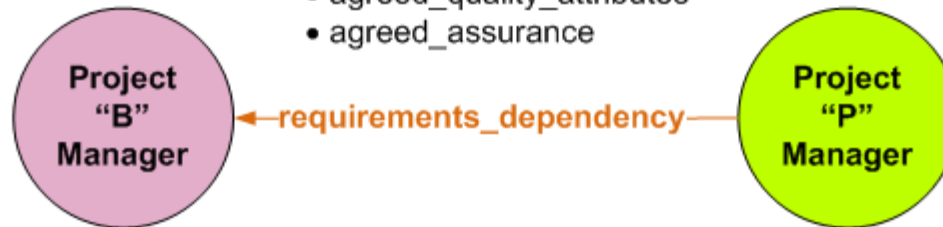
Receiver:
- earliest_need_date
- latest_need_date
- min_acceptable_functionality
- desired_functionality
- required_quality_attributes

Giver:
- offered_delivery_date
- offered_functionality
- offered_quality_attributes

Analysis

Identify potential influence interrelationship risks

Characteristics of analysis approach

- IMA team compares the IMs developed during the preparation and discovery steps to identify and characterize
    - Differences with respect to the strawman IMs
    - Differences between participants' perspectives of a given influence relationship
    - New, changed, deleted, or missing influence relationships
- Analysis is performed from 3 perspectives
    - Context-centric
    - Node-centric
    - Agreement-centric
- IMA team members use identified patterns of influence relationships

Software Engineering Institute | Carnegie Mellon

Context-Centric IM – Analysis -1

Provides a composite of top-down context-centric IM developed during interviews, overlaid with individual node-centric IMs

Highlights divergent perceptions of influence relationships obtained during interviews

- New, or "discovered" influence relationships (i.e., influence relations not apparent from context-centric, top-down perspective)
- Deleted influence relationships, that appear in a context-centric view, but not at the node-centric perspective
- Conflicted influence relationships, for which different participants have divergent interpretations

Provides input to node IM – analysis and risk mitigation planning

Context-Centric IM – Analysis -2

Three-step process for construction of "analysis" IMs:

1. "Normalize" the IMs developed during discovery phase
   - Common naming scheme (i.e., resolve synonyms/homonyms)
   - "Apples-to-apples" comparison of relationship characteristics (e.g., schedule-to-schedule, functionality desired-versus-promised)

2. Prepare a composite IM at the appropriate level (i.e., context, node, or agreement)
   - Overlay different stakeholders' views

How "A" sees the relationship

How "B" sees the relationship

Composite view of the relationship
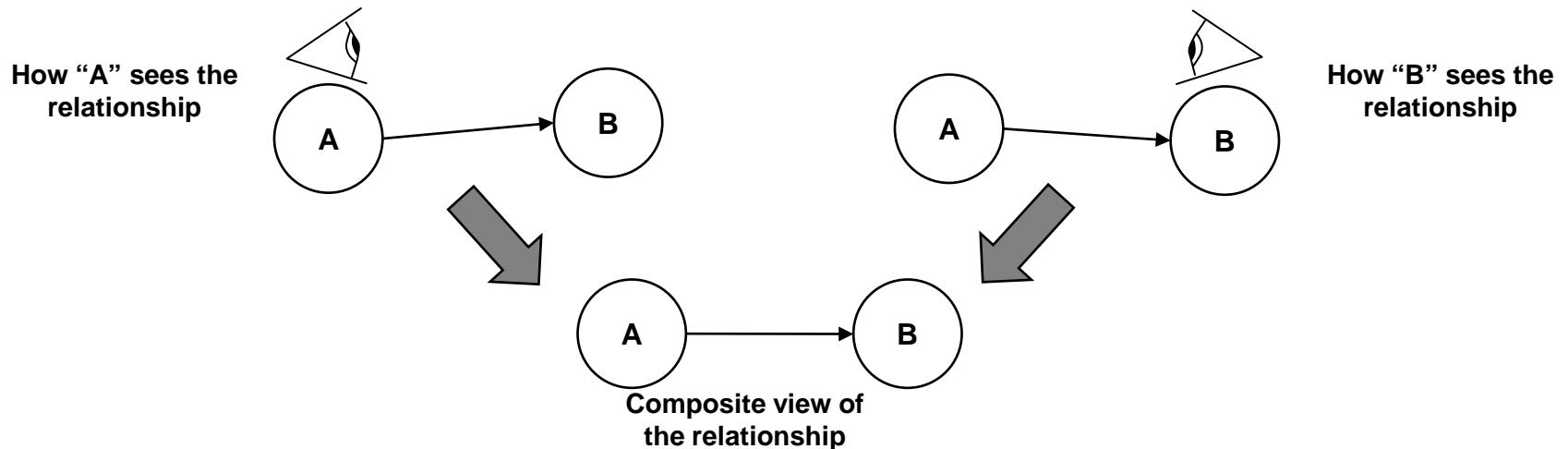
Context-Centric IM – Analysis -3

Three-step process for construction of "analysis" IMs (continued):

3. Examine the resulting composite IMs, highlighting any omissions, conflicts, or differences of interpretation on a relationship-by-relationship basis. For example:

    –   Do both stakeholders participating in a given relationship "see" the same thing? For example, do both parties in a "giver-receiver" relationship agree on the delivery date? What they even mean by "delivered"? The required functionality? How that functionality will be assured?

    –   Does only one stakeholder perceive the existence of a relationship? Are the stakeholders "talking past each other"?

        •   Does one stakeholder perceive the relationship as a schedule dependency, while the other one sees a backwards-compatibility relationship?

        •   They could both be referring to the same relationship, but their respective reference frames could prevent them from realizing that these relationships are—in fact—the same

Example: Context-Centric IM – Analysis
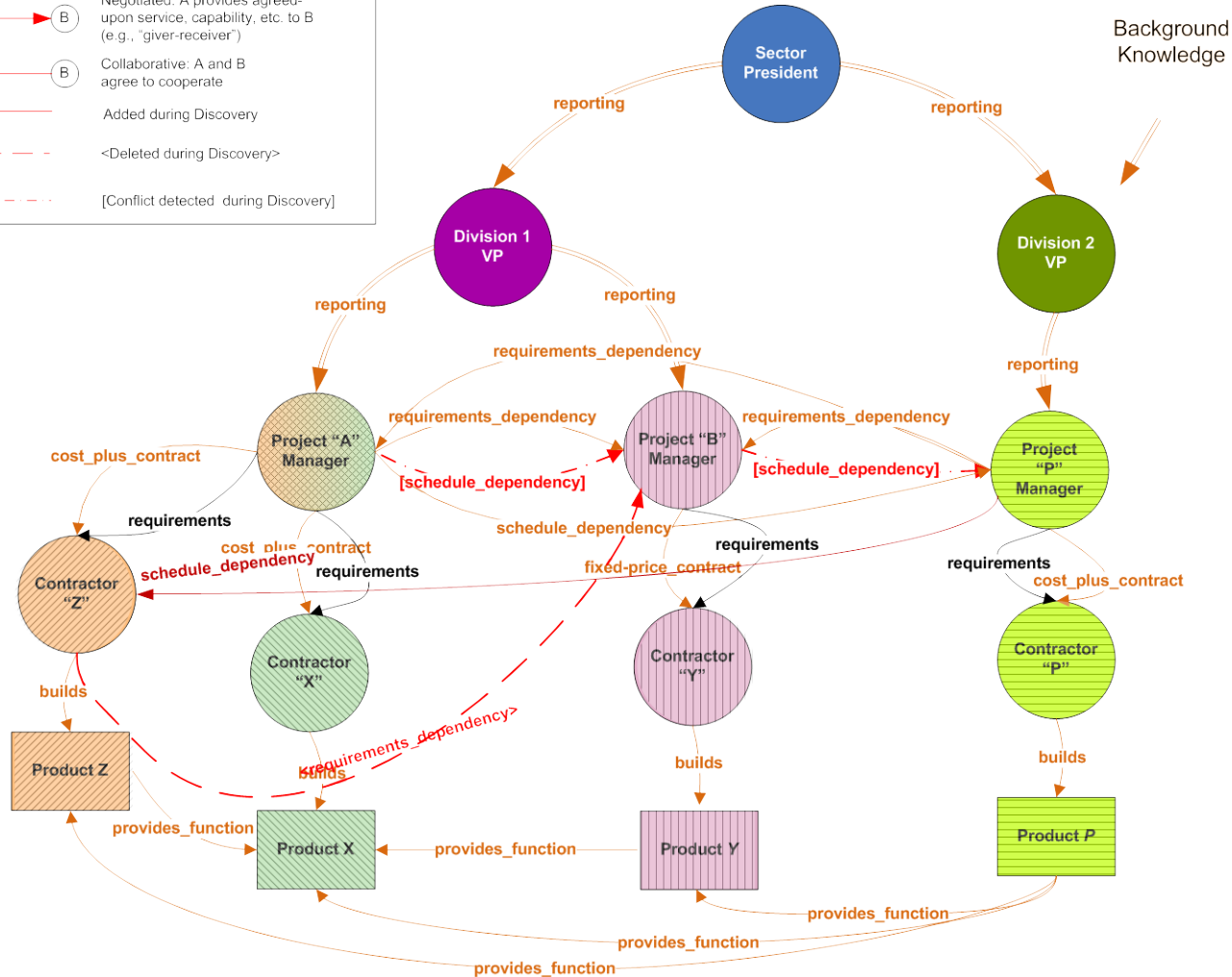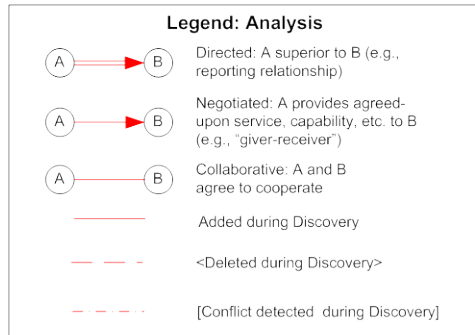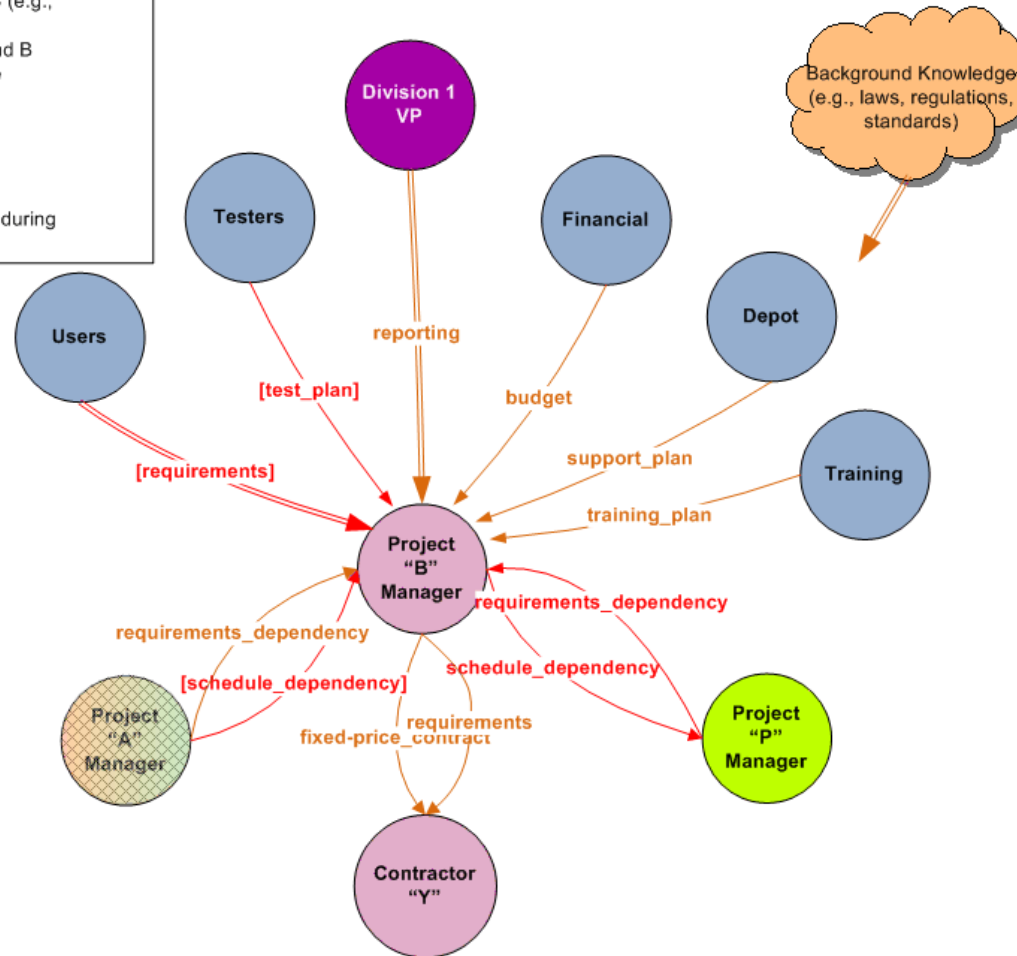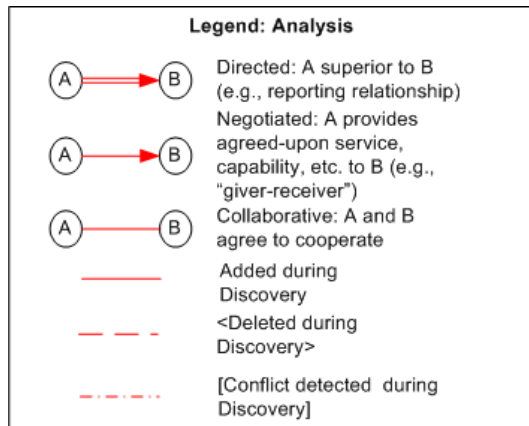
Node-Centric IM – Analysis

Characterize divergent perceptions of influence relationships

- Refinement of the context IM – analysis
- Captures/identifies top-level changes, additions, deletions, and conflicts for relevant influence relationships
- Supports prioritization of relationships requiring detailed analysis at agreement-centric level

Provides inputs to risk mitigation planning

Example: Node-Centric IM – Analysis
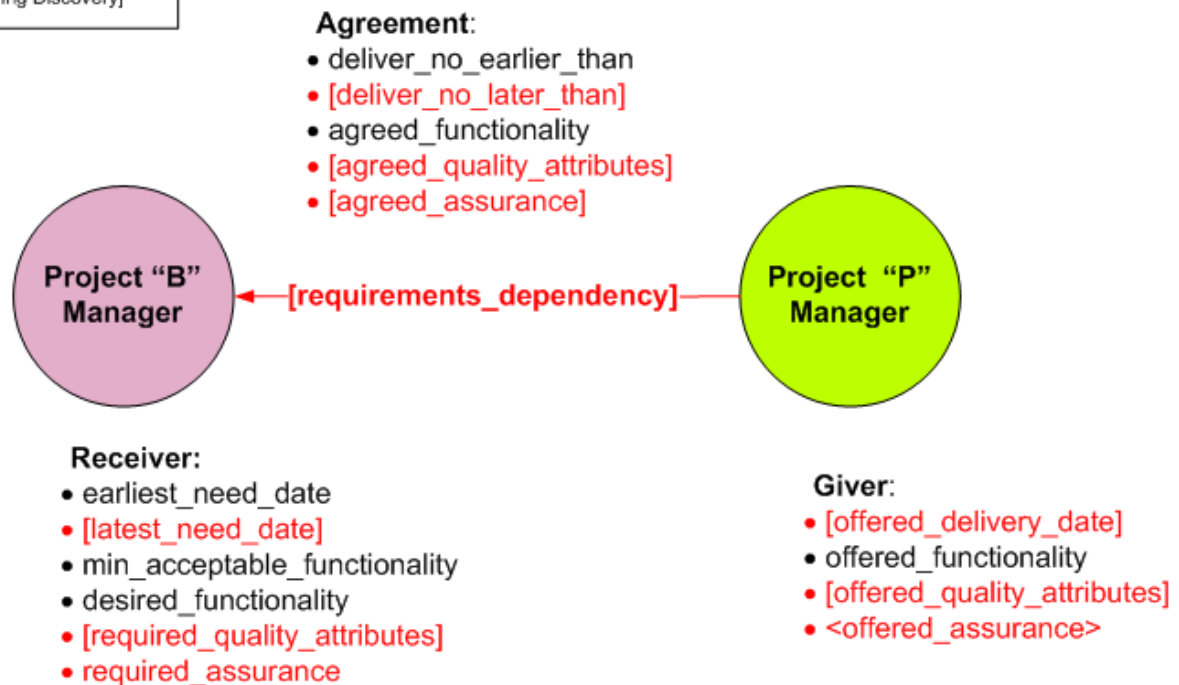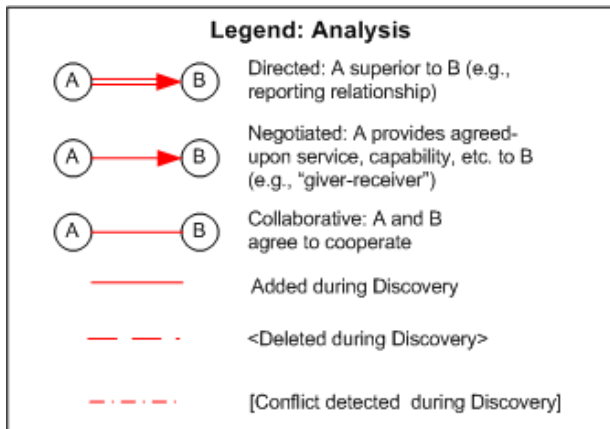
Agreement-Centric IM – Analysis

Captures detailed enumeration of changes, additions, deletions, and conflicts at the individual agreement level

- How has an agreement changed?
- What is the impact of that change?

Provides input to risk mitigation planning

Example: Agreement-Centric IM – Analysis

Legend: Analysis

A ⟹ B   Directed: A superior to B (e.g., reporting relationship)

A → B   Negotiated: A provides agreed-upon service, capability, etc. to B (e.g., "giver-receiver")

A — B   Collaborative: A and B agree to cooperate

———   Added during Discovery

— — — ·   <Deleted during Discovery>

— · — · — ·   [Conflict detected during Discovery]

Agreement:
- deliver_no_earlier_than
- [deliver_no_later_than]
- agreed_functionality
- [agreed_quality_attributes]
- [agreed_assurance]

Project "B" Manager ◄—[requirements_dependency]— **Project "P" Manager**

Receiver:
- earliest_need_date
- [latest_need_date]
- min_acceptable_functionality
- desired_functionality
- [required_quality_attributes]
- required_assurance

Giver:
- [offered_delivery_date]
- offered_functionality
- [offered_quality_attributes]
- <offered_assurance>

Mitigation Planning

Mitigation strategy and plans developed for prioritized risks identified during analysis phase

- Develop and implement mitigation strategy and plans in facilitated workshop
- Monitor for any changes
- Maintain and update IMs
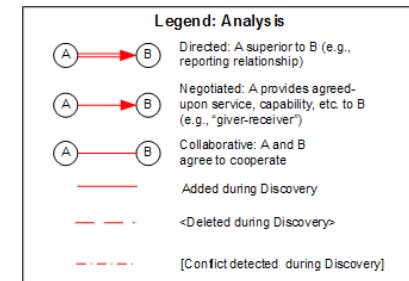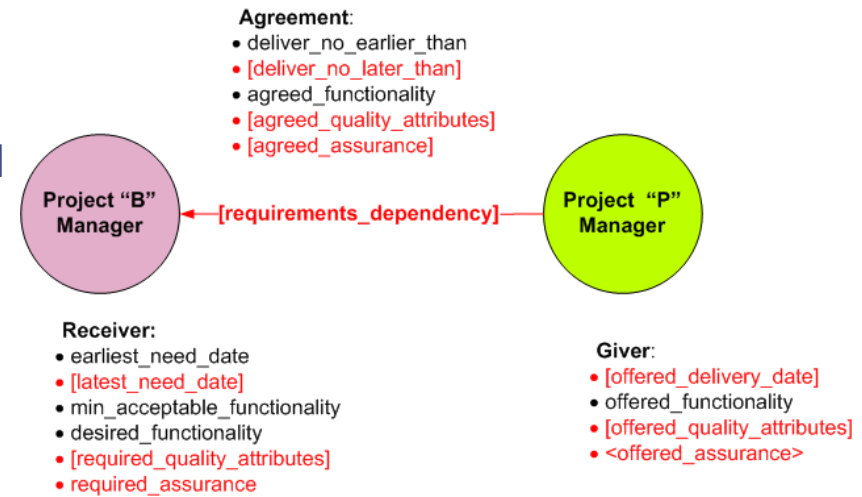
Software Engineering Institute | Carnegie Mellon

Example: Mitigation of Requirements Risk Identified in Analysis Agreement-Centric IM

Analysis identified changes in the requirements dependency between "P" and "B"

- Changes to "need" and "delivery" dates, and desired/offered quality attributes
- "B" has articulated new requirements for assurance, while "P" has dropped some previously-offered assurances

Analysis provides a basis for "B" and "P" to negotiate a new agreement—or identify that no agreement is possible

Identified aspects of the agreement— which may have been previously unstated— that need to be watched for future changes (e.g., quality attributes) based on their potential to affect cost, schedule, or performance

Agreement:
- deliver_no_earlier_than
- [deliver_no_later_than]
- agreed_functionality
- [agreed_quality_attributes]
- [agreed_assurance]

Project "B" Manager ←[requirements_dependency]— Project "P" Manager

Receiver:
- earliest_need_date
- [latest_need_date]
- min_acceptable_functionality
- desired_functionality
- [required_quality_attributes]
- required_assurance

Giver:
- [offered_delivery_date]
- offered_functionality
- [offered_quality_attributes]
- <offered_assurance>

Legend: Analysis

- A →→ B Directed: A superior to B (e.g., reporting relationship)
- A → B Negotiated: A provides agreed-upon service, capability, etc. to B (e.g., "giver-receiver")
- A ─ B Collaborative: A and B agree to cooperate
- ────── Added during Discovery
- ─ ─ ─ <Deleted during Discovery>
- ─·─·─ [Conflict detected during Discovery]

SUMMARY

Key Points

Conventional governance and acquisition techniques and processes provide an incomplete, and often incorrect, understanding of how the dynamics of systems of systems bear on the eventual success or failure of the enterprise

IMs—and associated IMA method:

- Permit identification of disconnects between stakeholder perspectives of influence interrelationships, and deviations from "official truth"
- Is useful for any organization involved in systems of systems with multiple stakeholders, and conflicting goals
  - Particularly relevant for program managers, senior executives, and policy makers
- Provides sufficient detail for focused mitigation actions

Contact Information

Jim Smith

Senior Member, Technical Staff

Research, Technology, and System Solutions (RTSS) Program

+1 703-908-8221

jds@sei.cmu.edu

Dennis Smith

Senior Member, Technical Staff

Research, Technology, and System Solutions (RTSS) Program

+1 412-268-6850

dbs@sei.cmu.edu