Tactics for Global Software Development: When to Do What?

James Herbsleb
Carnegie Mellon University
Multi-site Delay

Modification Request (MR) interval
Last Modification - First Modification
All changes over 2 year period

<table>
<thead>
<tr>
<th>Work Days</th>
<th>Network Element A</th>
<th>Network Element B</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>multi site 12.7</td>
<td>single site 4.9</td>
</tr>
<tr>
<td>20</td>
<td>multi site 18.1</td>
<td>single site 6.9</td>
</tr>
</tbody>
</table>

Carnegie Mellon School of Computer Science
Working Across Boundaries . . .

- Issue resolution paralysis
  - even small issues can take days
- Very difficult to stay “in the loop”
  - constantly surprised, “swimming upstream”
- Misalignment
  - undiscovered, conflicting assumptions
- Nonexistent or impaired social networks
  - loss of critical problem-solving mechanism
- Ineffective collaborative sessions
  - “What was decided?”
- Don’t know what you don’t know
Coordination is the Key

- Managing dependencies between tasks

Conway’s Law

▪ “Any organization that designs a system will inevitably produce a design whose structure is a copy of the organization's communication structure.”

▪ Implication: Modularity works as a coordination strategy

▪ Problem: Modularity has major limitations
Conway’s Law

Components

Teams

Software

Organization

Isomorphism
Conway’s Law

Components

Teams

Software

Homomorphism

Organization
What about the Connectors?

Components

Teams

Software

Organization
Complexity and Uncertainty

What kind of coordination is required?

What kind of coordination is required?
Coordination Requirements: Complexity

- Examples
  - How “big” is an API?
  - How complicated are API usage policies?
  - Features with implementations spanning components
  - Challenging non-functional requirements
    - Performance
    - Security
    - Availability
    - Etc.
Coordination Requirements: Uncertainty

- Examples
  - Allocation of functionality to components
  - Modification and refinement of component interfaces
  - Volatile requirements
  - Dependencies on other systems that are changing
    - Hardware
    - Firmware
    - Middleware
    - Etc.
Congruence

Components

Software

Teams

Organization

Carnegie Mellon
School of Computer Science
What determines coordination capability?
Coordination Mechanisms

- Preparation, e.g.,
  - Plans
  - Specifications
  - Defined processes
- Shared representation, e.g.,
  - Metrics dashboard
  - Posting test results
  - “Living” documents
- Communication, e.g.,
  - Meetings
  - “Informal” communication
Distance Breaks Down Communication

Communication

Gap

Within site

Lack of unplanned contact
Knowing who to contact about what
Difficulty of initiating contact
Ability to communicate effectively
Lack of trust, or willingness to communicate openly

Across sites
Distance Breaks Down Preparation and Shared Representations

Meeting of Minds

Gap

Variation in practices
Variation in understanding
Interpretation depends on context
Lack of shared notations
Little ability to anticipate actions

Within site
Across sites

Variation in practices
Variation in understanding
Interpretation depends on context
Lack of shared notations
Little ability to anticipate actions

Meeting of Minds

Gap

Variation in practices
Variation in understanding
Interpretation depends on context
Lack of shared notations
Little ability to anticipate actions

Within site
Across sites
Many Factors Affect Coordination Capability

- Organizational factors, e.g.,
  - Geographic distribution
  - Divergent processes
  - Different management practices
  - Communication infrastructure

- People factors, e.g.,
  - Experience working together
  - Domain and technology expertise
  - Language skills
Achieving Congruence

- Matching coordination requirements and coordination capability
Thinking About Tactics

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>
Changing the Game

Complexity

Low

High

Uncertainty

High

Low

Reduce Complexity

Carnegie Mellon
School of Computer Science
Changing the Game

High

Low

High

Uncertainty

Low

Complexity

Reduce Uncertainty
Beware of Architectural Change

- Lessons from the history of photolithographic alignment equipment*

Future Work

- “Coordination view” of an architecture
- Measuring congruence of architecture and organization
- Architectural tactics for improving congruence
Conclusion

- Architectural decisions create the “coordination landscape”
- Architectural structure and organizational structure are strongly related
- Congruence is necessary for project success
- Complexity and uncertainty present different problems
- Need research on measuring congruence, devising tactics for improving it