SIEMENS

GSD Workshop

The Context

- Siemens is a large global electronics company
  - 190 countries
  - ~33K software engineers
- Most development efforts of any consequence are using multi-site teams
- Siemens as a whole has CMMI process maturity goals
- Each operating company has/is investing considerably in their organizational processes

Work to Date

- Conducted retroactive case studies from numerous Siemens projects*
- Conducted “best practice interviews” from numerous multi-site projects*
- Established and executed “experimental” global studio project using student teams from 5 countries
- Conducting data collection and analysis on the relationship between specific aspects of software architecture and collaboration

* Sponsored by the Siemens Software Initiative and conducted with Jim Herbsleb from CMU

The Problem

- Defining effective organizational processes for multi-site development is problematic
- Contexts differ drastically from one project to the next
- Organizational learning comes at great expense and is typically not applicable across projects
- Unclear what is the appropriate balance of overhead/risk for a given project
- Project management has few tools to monitor progress in project

Current Approach

- Typically current approaches are:
  - Organizationally defined milestones, templates, and processes
  - Agile processes
  - Ad-hoc high-overhead approaches (e.g. personnel exchange, extensive travel by key individuals, …)
- Without extensive travel or collocation of key personnel problems are discovered late and with great consequence
  - Current solution is to define finer grained tasks, however this is labor intensive and doesn't always give early warning of problems

The Idea

- Rather than focusing on completion of the task at hand, focus on establishment of a shared mental model across team
  - Goal of artifacts and processes is to establish a common understanding with respect to particular tasks
  - Would make sense to pay attention to the extent to which these mechanisms are successful in establishing a common understanding
  - Without common understanding it is possible to produce something (i.e. create the appearance of progress), but likely not as intended
- If a sufficient shared mental model doesn't exist it doesn't make sense to start execution of the task
  - This would give an indication of when it would be necessary to augment interactions
  - Seems like it would likely minimize the risks of producing inappropriate artifacts

How Might This Work?

- Several ways in which a manager might measure the development of such a shared mental model
  - Monitoring of the social networks that exist in the projects could highlight:
    - The extent to which parties that are expected to communicate are actually communicating
    - Patterns of communication that indicate an issue of some kind
  - Explicit “sync meetings”
    - This may provide an indication of the current level of shared understanding that exists
  - Explicit validation steps
    - This could help validate the understanding of particular items
  - Perhaps project infrastructure could also be instrumented to provide useful information (e.g. server logs, CM logs, …)

What Have We Done

- Collected data from GSP project
  - SNA data
  - Collaboration data from:
    - Wiki
    - Logs
    - Meeting minutes
    - Project plan
    - …

Next Steps???

- Analyze data
  - Attempt to categorize interactions using “protocol analysis”
  - Correlate “synchronization interactions” with task outcome
- Thoughts?????