

# Orchestration of Global Software Engineering Projects

Position Paper

C. Bartelt, M. Broy, C. Herrmann, E. Knaus, M. Kuhrmann, A. Rausch,  
B. Rumpe, K. Schneider

Technische Universität Clausthal, Leibniz Universität Hannover  
Technische Universität München  
RWTH Aachen

# Agenda

- Promises of Global Software Engineering
  - Results in practice
  - What is Global Software Engineering all about?
  - Scope
  
- A general approach
  - Example
  - Views
  
- Conclusion

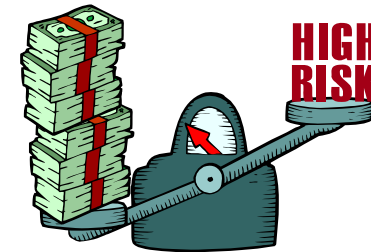
## Promises

- Main significant forces for global development (Software Engineering):
  - Economically, e.g.
    - Personnel costs
    - Networks of companies and suppliers
  - Organizational
    - Global organized company → global projects
    - Global personnel pool
  - Strategically
    - Close to market (big players like Microsoft, SAP etc.)
    - Political aspects
  
- Forces and (real) settings are mirrored by several „distribution styles“
  - Buyer/Supplier (e.g. simple development tasks)
  - Collaborative work in distributed teams



## Promises – Results in Practice

- Global SE has additional risks
  - (risks of classical SE + )
  - Variability of a distributed project and its settings
  - Distances, including
  - Professional and social issues
  
- Fact: Global projects show the same lacks as co-located projects
  - Each communication lack → additional effort → costs
  - Each underspecified requirement → additional effort → costs
  - Task often take much longer → costs



*Originally Global SE should reduce costs of software projects...*

## Promises – What is Global SE all about?

- Key question: *What is the difference between classical and global SE?*
  
- 1<sup>st</sup> notion: *In global SE*
  - *work is allocated to people at distributed sites with*
  - *different SE cultures.*
  
- 2<sup>nd</sup> key challenge: *Establish appropriate communication and collaboration.*
  
- 3<sup>rd</sup> approach: *(re-)orchestrate the existing communication and collaboration cultures of all participating parties...*
  - Levels:
    - project set-up and management
    - processes and information flow
    - artifact and product models

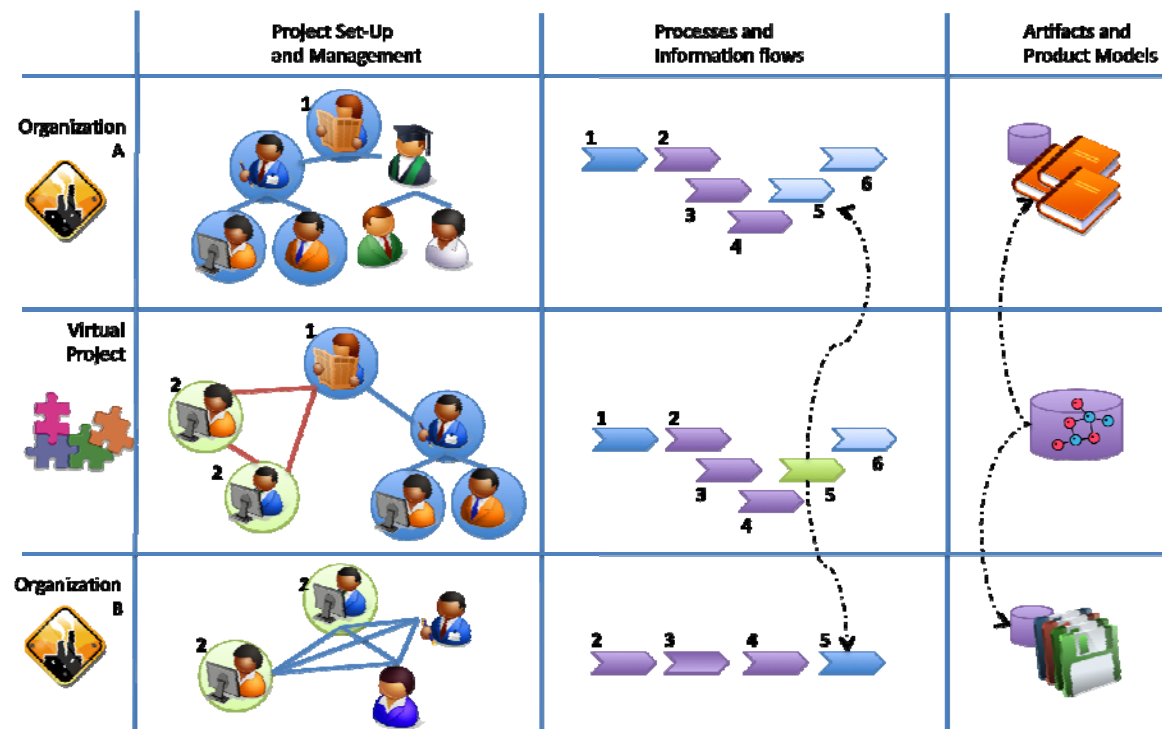
## A General Approach

- Considering the three levels, each one has to be mentioned from
  - Organizational and
  - Technical points of view
  
- Need for (re-)orchestration, coupling and integration
  - At all levels
  - Explicit interfaces are required
  
- Our approach covers: communication, processes and technology
  - Tracing and consistency of dependencies of data and information
  - Constitution of a Global SE, multi-site development process
  - Constitution of a Global SE project organization

## A General Approach - Example

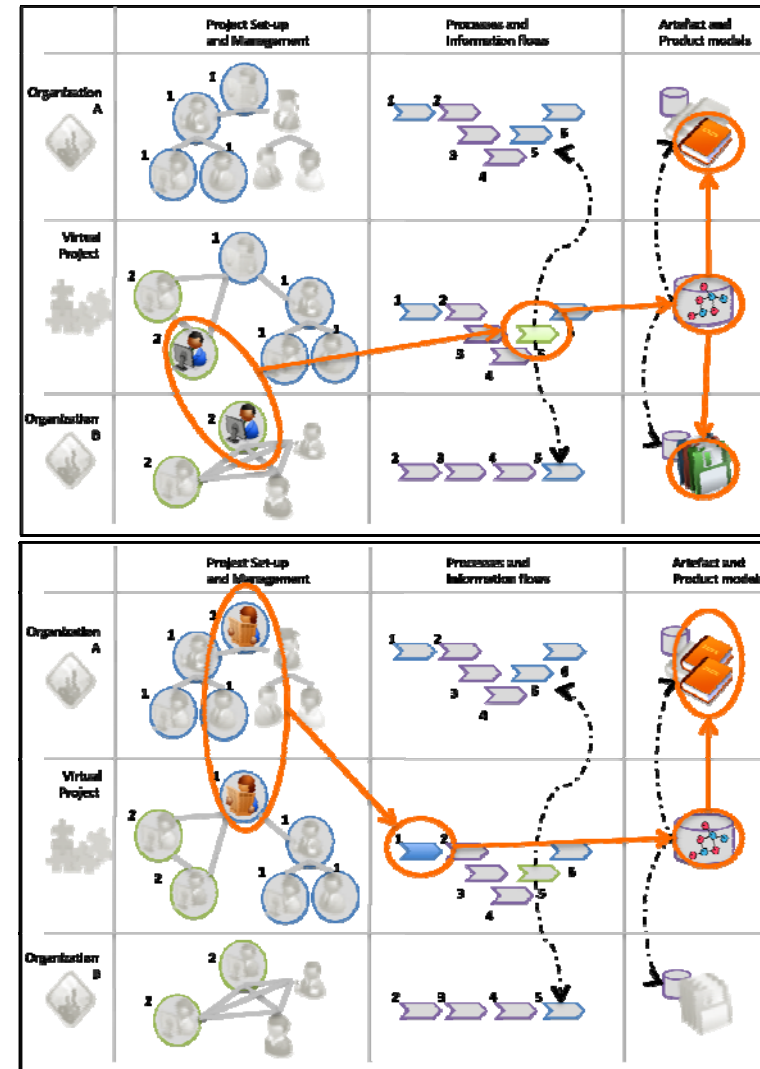
- Organizations *A* and *B* will cooperate in a distributed setting
  - An integrated project
  - Mixed teams
  - Common process
  - Distributed data storage

- Each organization still has its own ones...



## A General Approach - Example

- The approach:
  - Provide *task-specific* views  
(Sample 1: A developer has to perform some work...)
  - Provide *role-specific* views  
(Sample 2: A project manager has to determine some project state)
  - *There exist several views...*
- Views span all levels to be stated
  - project set-up has impacts on process
  - has impact on artifacts...



## Detailed discussion and working hypotheses

- Sample and setting show
  - High degrees of variability and uncertainty
  - Simple integration wouldn't work...
  - Simple provision of development artifacts using e.g. a shared folder is not suitable..
  - ...
  
- For each level → **key questions** has to be considered
  - To get an idea of what has to be done at a particular level
  - To collect best practices (if available) to combine them (bridging the levels)
  - To create new methods and techniques if required



## Detailed discussion and working hypotheses

- Level: Project Set-up
  - What are responsibilities?
  - What are communication paths? → Is there a correlation?
  - What means “project management”?
  - Is there a correlation between communication and responsibility?
  - Is there some kind of “virtual super project”?

## Detailed discussion and working hypotheses



- Level: Processes
  - What are appropriate process interfaces?
  - How can information flow be “guided”?
  - What strategy for integration or coupling is the most promising?
  - How can we identify the integration options and appropriate process-interfaces or define some kind of common development process?
  - Reason: The harmonization of processes is necessary to build a common understanding of the whole global project. Understanding means a common vocabulary (terminology, ontology), a common set of milestones, deliverables, common strategies for coordinating the distributed (sub-)projects and knowledge of the requirements related to process-relevant artifacts

## Detailed discussion and working hypotheses



- Level: Artifacts
  - Who owns an artifact?
  - Are artifacts consistent and free of redundancies?
  - Are there redundancies e.g. because the specification is mirrored at the developers' location? And if so, are both copies of the specification consistent?



## Detailed discussion and working hypotheses

- Key questions are just an outline
- Further questions relevant for Globale SE might be:
  - What are additional/other problems with Global SE?
  - Does the *view-based* approach cover all problems?
  - Do the intended levels cover all problems?
  - In how far, selective improvements are possible?
  - How to handle change and variability in a Global SE project?
- Hypotheses:

*Integration and coupling not only solves problems, but generates new...*

## Conclusion

- Global SE is reality but challenging
  - Level-building makes problem identification easier
  - View-building makes handling of complex structures easier
  
- Open questions
  - In sum: “What is the right amount of Global SE?”
  
- A weighted strategy might be advantageous, covering
  - People
  - Methods
  - Tools

