



Best practices for Configuration Management in Distributed Organizations

Lars Bendix
bendix@cs.lth.se

Department of Computer Science
Lund University
Sweden



Distribution

Distribution is good:

- it gives a larger pool of talents and specialists
- allows cooperation between departments/companies
- facilitates integration for mergers and acquisitions
- allows around the clock work
- gives more flexibility in scaling up and down projects

Distribution is bad:

- it is more complex to manage
- it creates silos between groups
- people don't understand and trust each other
- you loose control over remote teams/people
- ...



CM and Distribution

Configuration Management already handles "distribution":

- programmers are rarely co-located
- developers are often distributed (also in time)
- we handle development AND maintenance
- where is the Project Manager?



The SWAT team

Inner circle:

- Lars Bendix, CS@LTH and Sigrun, Sweden
- Marc Girod, Ericsson, Ireland
- Jan Magnusson, Sony Mobile Communications, Sweden
- Christian Pendleton, Softhouse Consulting, Sweden

Outer circle:

- Ulf Asklund, Sony Mobile Communications, Sweden
- Andreas Göransson, Hypergene, Sweden
- Fredrik Hugosson, Axis, Sweden
- Tero Kojo, FinPro, Finland
- Annabella Loconsole, Malmö University, Sweden
- Tatiana Shpichko, Nokia, Finland
- Ulf Steen, ABB, Sweden

Recruited from SNESCM:

<http://snescm.cs.lth.se/>



Our mission



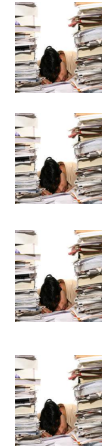
We were asking ourselves:

- is there something here we don't understand?
- is there something that others have overlooked?

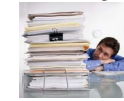
- what challenges exist?
- which ones involve CM?
- (which ones do not involve CM?)



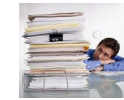
The method



Challenges



CM in DD



Cases



Analysis



Discussions



Challenges 1



Jiménez et al., 2009:

- communication
- group awareness
- software configuration management
- knowledge management
- coordination
- collaboration
- project and process management
- process support
- quality and measurement
- risk management



Challenges 1



Jiménez et al., 2009:

- **communication**
- *group awareness*
- **software configuration management**
- *knowledge management*
- **coordination**
- **collaboration**
- *project and process management*
- process support
- *quality and measurement*
- risk management



Challenges 2



da Silva et al., 2010:

- **effective communication**
- cultural differences
- **coordination**
- *time zone differences*
- *trust*
- asymmetry in processes, policies and standards
- *physical distance*
- IT infrastructure
- *different knowledge levels or knowledge transfer*
- **tracking and control**
- **cooperation**
- people management/conflict resolution
- language barriers
- *task allocation*
- identification of roles and responsibilities
- *knowledge management*
- **scope and change management**
- *overall visibility*
- *differences in technologies used*
- creating team spirit
- *project planning*
- quality
- *intellectual property issues/ confidentiality and privacy*
- different stakeholders
- *schedule management*
- **synchronizing work between distributed sites**



Challenges 2



da Silva et al., 2010:

- **effective communication**
- cultural differences
- **coordination**
- *time zone differences*
- *trust*
- asymmetry in processes, policies and standards
- *physical distance*
- IT infrastructure
- *different knowledge levels or knowledge transfer*
- **tracking and control**
- **cooperation**
- people management/conflict resolution
- language barriers
- *task allocation*
- identification of roles and responsibilities
- *knowledge management*
- **scope and change management**
- *overall visibility*
- *differences in technologies used*
- creating team spirit
- *project planning*
- quality
- *intellectual property issues/ confidentiality and privacy*
- different stakeholders
- *schedule management*
- **synchronizing work between distributed sites**



CM & DD 1



Fauzi et al., 2010:

- **dispersed software teams do not get information on what other teams are doing**
- **difficult to know the traceability of each module**
- *the definition of modifications or problems to be handled is unclear*
- *dependency*
- *delay and increased time required to complete change requests*
- **working in different SCM environments**
- *change requests are handled at various levels in the project*
- *lack of a planned baseline*
- *lack of coding standards*
- *code ownership*
- *unclear flow of development*
- *tool selection*
- **artefacts with different versions and content at each site**



CM & DD 2



Pilatti et al., 2006:

- *minimize dependencies between distributed teams*
- **work with one instance of SCM environment**
- **all CIs required for a build should be put under CM**
- **projects should define one build coordinator**
- **establish and clarify CM before starting project**
- **CM engagement in the beginning should be prioritized**
- **always plan and document baselines (in CM plan)**
- *re-plan activities due to scope floating across teams*



Results 1



Strongly related to Configuration Management:

- effective communication
- coordination
- tracking and control
- cooperation
- scope and change management
- synchronizing work between distributed sites
- communication
- software configuration management
- coordination
- collaboration
- work with one instance of SCM environment
- dispersed software teams do not get information on what other teams are doing
- difficult to know the traceability of each module
- working in different SCM environments
- artefacts with different versions and content at each site



Results 2



Weakly related to Configuration Management:

- time zone differences
- trust
- physical distance
- different knowledge levels or knowledge transfer
- task allocation
- knowledge management
- overall visibility
- differences in technologies used
- project planning
- intellectual property issues/confidentiality and privacy
- schedule management
- group awareness
- knowledge management
- project and process management
- quality and measurement
- minimize dependencies between distributed teams
- re-plan activities due to scope floating across teams
- the definition of modifications or problems to be handled is unclear
- dependency
- code ownership
- tool selection



Results 3



Not related to Configuration Management:

- cultural differences
- asymmetry in processes, policies and standards
- IT infrastructure
- people management/conflict resolution
- language barriers
- identification of roles and responsibilities
- creating team spirit
- quality
- different stakeholders
- process support
- risk management
- lack of coding standards
- unclear flow of development



The contributions



Dear PM – and CM,

Configuration Management can support DD too if:

- it remembers its “old virtues”
- it pays attention to the questions developers often ask
- it collects data that can answer those questions
- it helps transform that data into knowledge
- it makes the knowledge easily accessible

CM can:

- take responsibility for solving some of the challenges
- provide help for alleviating other challenges
- be too busy to bother with “exotic” challenges



Future work



Now – that was a lot of challenges. What about solutions?

Some strongly related challenges are invariant to distribution

Other strongly related challenges are variant to distribution

Most (all?) weakly related challenges will require “something extra”

We want to do a SCM4GSD 2.0 project:

- inner/outer circles
- resources and money
- cases and experiments
- 3 year duration, start early 2013

If interested, drop me an email...