Distributing a lean organization:
Maintaining communication while staying agile
(LESS 2010, Helsinki)

Sebastian Meyer, Eric Knauss, Kurt Schneider

{sebastian.meyer,eric.knauss,kurt.schneider}@inf.uni-hannover.de

Software Engineering Group

Leibniz Universität Hannover

Outline

- Motivation
- Project Setup
- Our GQM Approach
- Problems
- Conclusions

Motivation

- There are just little few developers who are skilled in every aspect/technology of software development.
  - It is a common practice to build teams consisting of specialists
  - Specialists needs to be integrated from outside or inside the company
- Not in all cases a co-located team can be formed
  - Integrating these teams can turn into an extreme overhead
- Communication is one of the major keystones in agile development
- We wanted to understand communication in a (distributed) agile project to understand which additions are absolutely necessary for distributing teams
  - And how they change the agile character

Project Setup (I)

- We used our XP-Lab as a base for the observations
  - Two 4h Spikes (Tuesdays)
  - One get-together to kickoff the project
  - Working one 40h week (Tuesday – Friday + Monday)

- For the distribution, we made use of the GloSE-Lab
  - A distributed lab between (in this case) two universities
  - Part of the Global Software Engineering project in Germany

- We built two teams:
  - A dispersed team with 4 students at each site
  - A co-located team with 7 students
  - All students where in the Master program

Project Setup (II)

- To connect to the remote site, we used:
  - Skype
  - A dedicated video conference system
  - A distributed MindMap
  - A desktop/document sharing tool

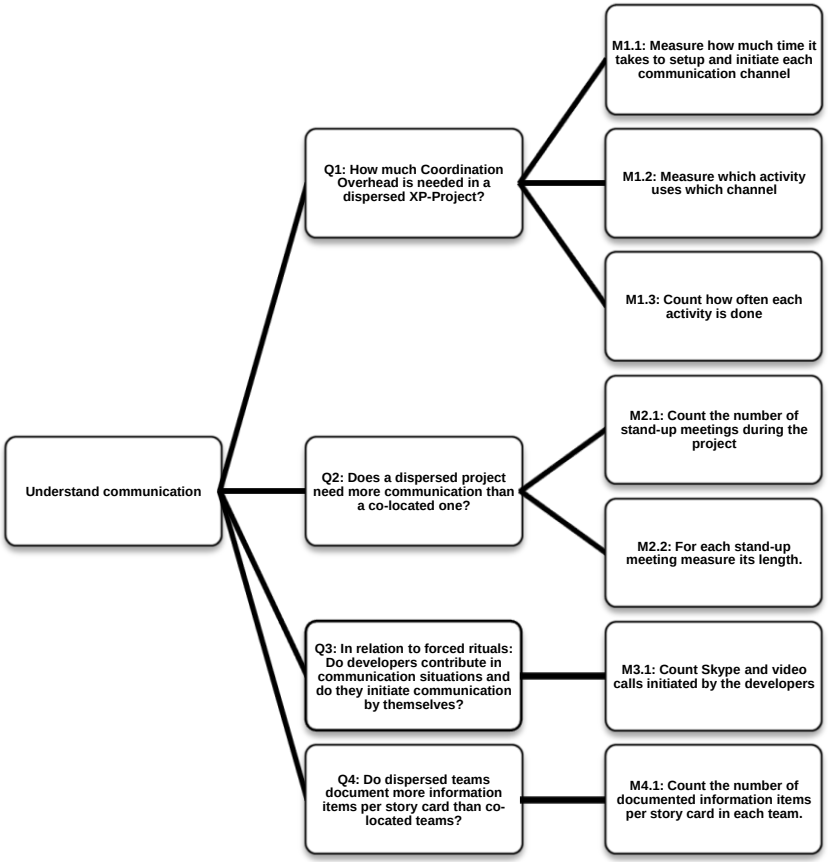
- Coaches and on-site customer where located in Hannover

- Both teams worked on Android Projects in Java
  - Using Eclipse with integrated Android tools
  - SVN for code sharing

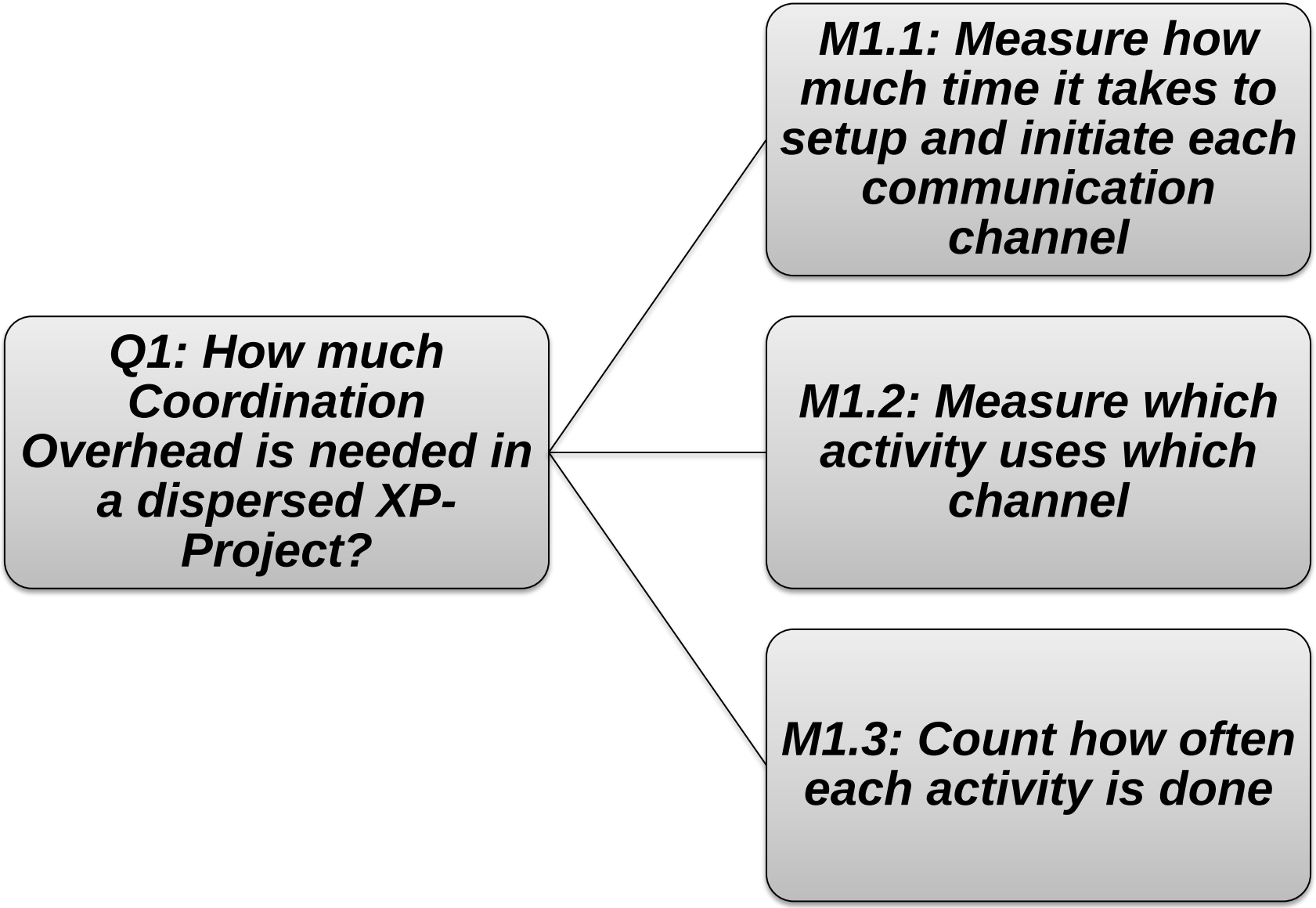
The GQM Tree

- We used the GQM method with the following goal:

  Understand the communication in distributed XP from the viewpoint of an XP coach and the developers



Question 1


Q1: How much Coordination Overhead is needed in a dispersed XP-Project?
M1.1: Measure how much time it takes to setup and initiate each communication channel
M1.2: Measure which activity uses which channel
M1.3: Count how often each activity is done

Hypothesis for Question 1

- **Hypothesis 1**: We assume that communication in a dispersed XP project needs considerably more technical and organizational effort than in the co-located case and has influence on the question, whether the project stays in time and budget.

Measurements for Question 1

Channel	Metric 1.1		Metric 1.2				
	Initialize	Setup Time	Planning Game	Standup-Meeting	Dev. - Customer	Dev. – Dev.	Coach – Dev.
Voice	**30min** (setup)	**1min** (start skype)			X	X	X
Video		**1h forerun** (get conference rooms) **+ 10 min** (init connection)	X	X			
Shared Story Cards	**1 h** (create accounts, dev. training)	**15 min** (setup notebook and beamer) **+ 4 min** (log-in all sites)	X	X		X	X
Desktop Sharing	**2 h** (create accounts, training of developers, init structure)	**2 min** (start and login) **+ 2 min** (login others, get control)		X	X	X	X
Shared Whiteboard		**3 min** (login all sites)	X	X	X		X
Shared Tracking Information		**3 min** (start and login all sites) **+ 1 min** (share file)	X	X			(X)
Σ setup time[3]	-	-	36 min work / 1 h time	40 min work / 1 h time	4 min	19 min	19 min
Occurrences	1	63	3	5	31/8[4]	16	const .

Question 2

Q2: Does a dispersed project need more communication than a co-located one?

M2.1: Count the number of stand-up meetings during the project

M2.2: For each stand-up meeting measure its length.

Measurements for Questions 2

- **Hypothesis 2:** We assume that one stand-up meeting per day is enough, as in the co-located case but that it will take more time (approx. 50 % more) compared to a co-located setting.

Date	25.06.10	26.06.10	27.06.10	28.06.10	31.06.10
Length	19:48 min	31:29 min	26:00 min	27:32 min	37:52 min

- Average length of **dispersed** meeting: 28:32 min
- Average length of **co-located** meeting: ~15 min

Question 3

Q3: In relation to forced rituals: Do developers contribute in communication situations and do they initiate communication by themselves?

M3.1: Count Skype and video calls initiated by the developers

Measurements for Question 3

- **Hypothesis 3:** We assume that the developers contribute less than in a co-located project. Furthermore, we expect very few self-initiated communication situations. Rituals like planning poker or 5-point-evaluation improve the situation

- Skype calls initiated by developers: 16
- Video conferences initiated by developers: 0

- Co-located team and parts of the distributed team that where on the same location communicated a lot
- Few communication between the distributed locations

Question 4

Q4: Do dispersed teams document more information items per story card than co-located teams?

M4.1: Count the number of documented information items per story card in each team.

Measurement for Question 4

- **Hypothesis 4:** We assume that they document about 50 % more than in a co-located setup. Video channels decrease amount of documented information items and allow to value interactions over documents.

	Co-Located Team	Dispersed Team
Requirements total	135	127
Documented Req.	54	28
Not documented Req.	81	99

Contradiction!

Problems

- We assume that the company wants to be lean
  - Needs a clear commitment
- A technical difficult communication channel that needs complex and expensive systems that are not available all the time requires much more organizational overhead
  - Example: Video conference room must be available when needed
  - This is not necessarily a problem of the used tools
- While our XP course is as realistic as possible, it‘s still a university course with very limited time
- Both sites were in the same time zone

Conclusions (I)

- Our goal was *to understand the communication in distributed XP from the viewpoint of an XP coach and the developers*
- The communication needs are not very different from the co-located case
- We could detect, that the co-located parts of the dispersed team communicated more
  - Inter-site communication *seems* to have a higher border
- The use of a dedicated video conference system was a great help
  - For group calls
  - For planning games and contact with the customer
  - We even had distributed lunch

Conclusions (II)

- To effectively distribute agile teams, we needed:
  - voice and video communication
  - tools for sharing desktops, story cards, whiteboards, and tracking information.

- Supposedly simple technologies like whiteboards or story cards are very difficult to substitute in a distributed settings.

- Setup time remains a major issue
  - The addition of many small time amounts to setup and keep the channels running adds to the coordination effort of managing the different skills in an agile team.

- Distributed communication feels much more intense