Teaching Agile Software Development at University Level: Values, Management, and Craftsmanship

Martin Kropp, FHNW
Andreas Meier, ZHAW

CSEE&T 2013, San Francisco, May 20, 2013

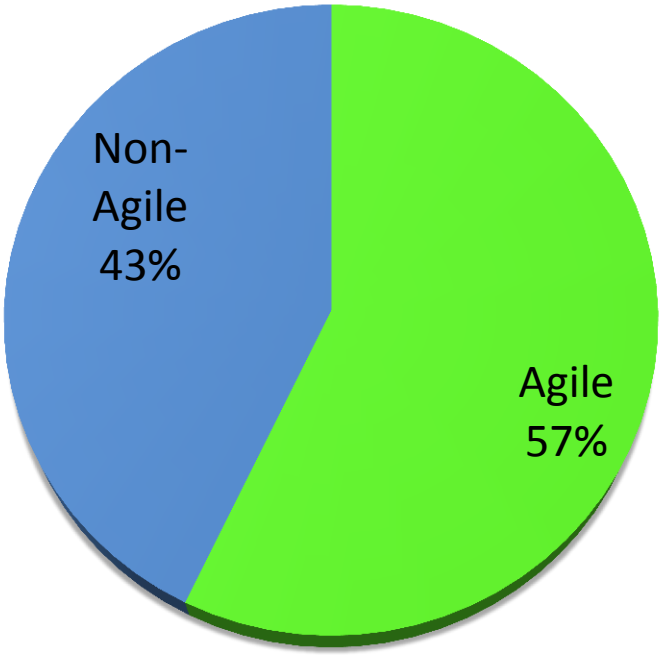
Agenda

- Motivation
- Pyramid of Agile Competences
- Agile Software Engineering Course
- Evaluation

Motivation

- Swiss Agile Study 2012 (SAS)
  - 140 IT companies
  - 194 IT Professionals
- Results?

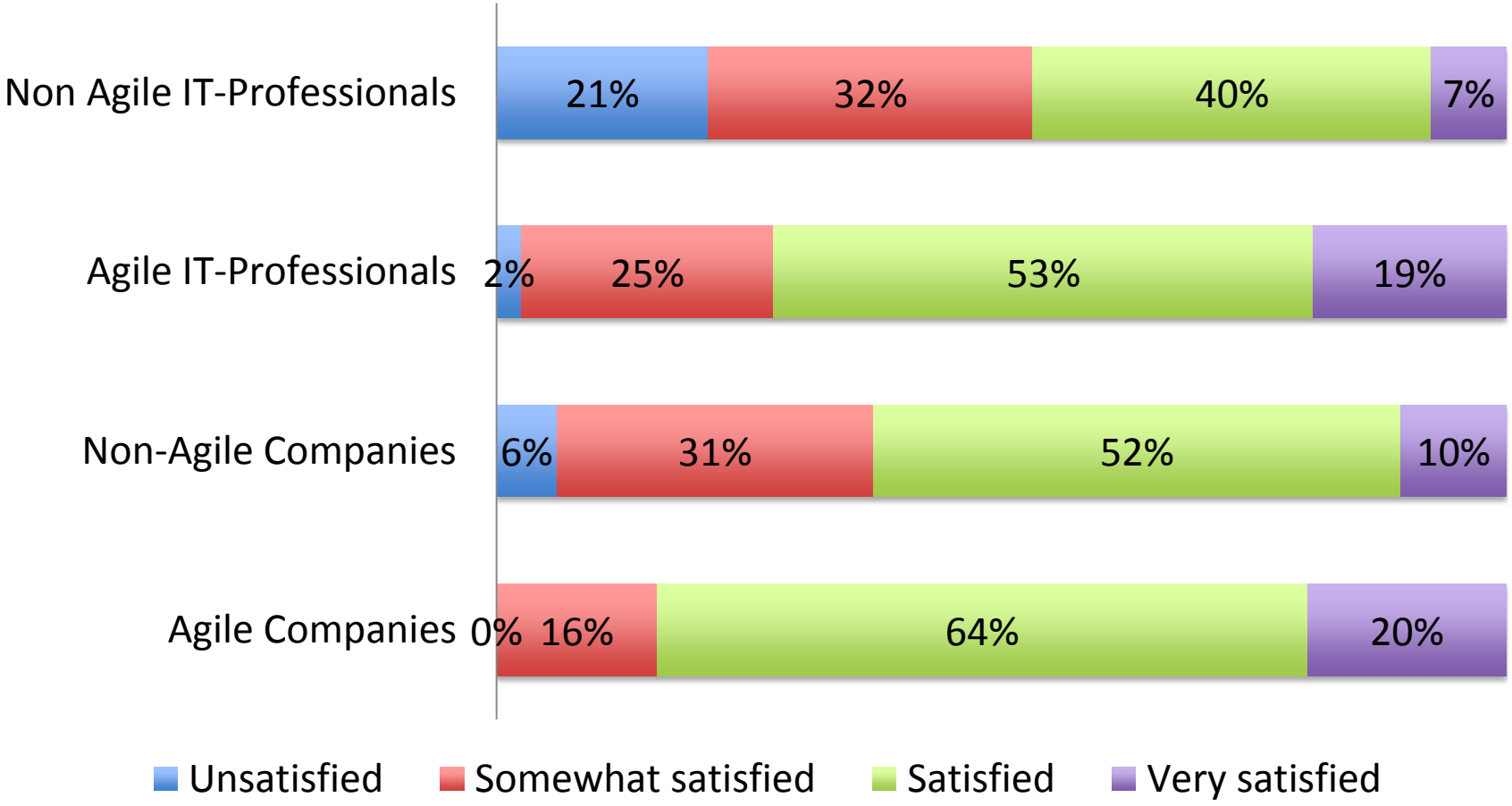
Company Survey



www.swissagilestudy.ch

SAS Results: Satisfaction

How satisfied are you with your current methodology?



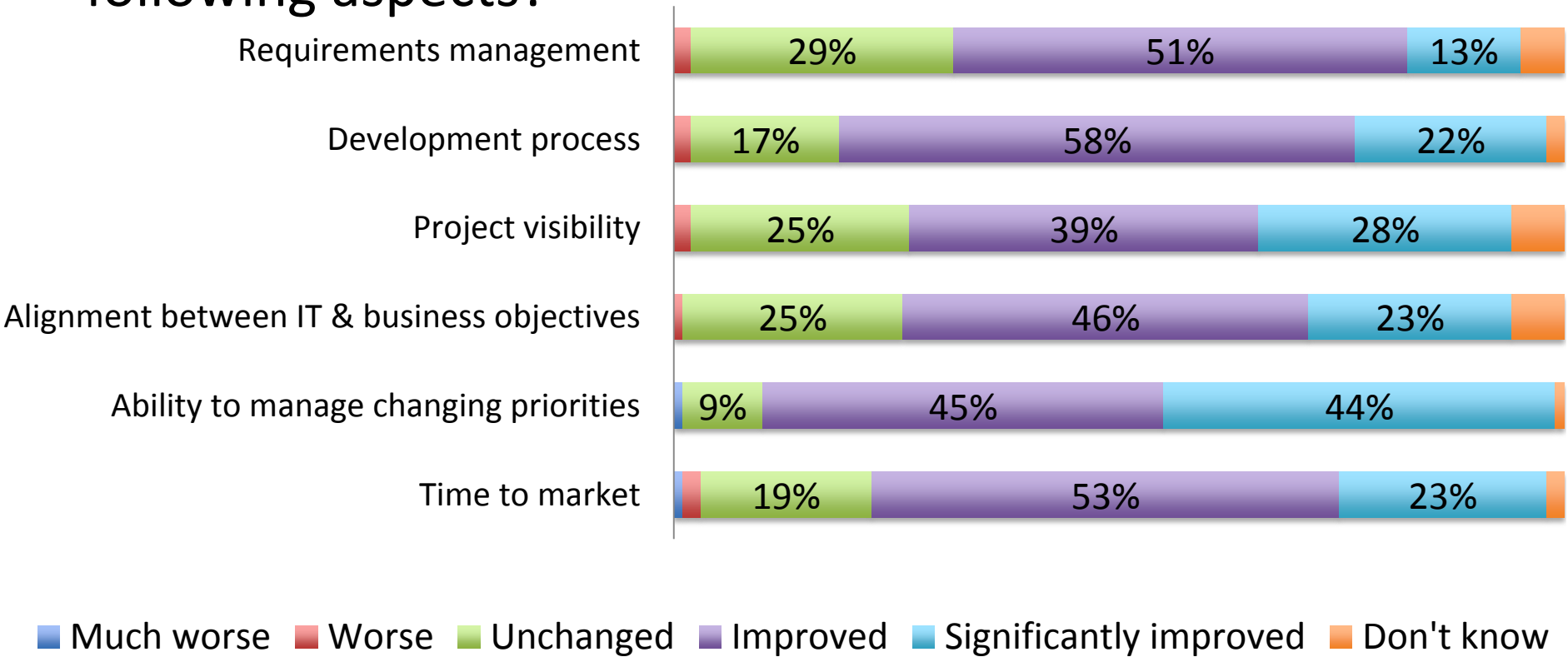
Motivation

SAS shows very promising results:

- much higher satisfaction with agile methodologies than with plan-driven ones

SAS Results: Agile Influence

How has agile software development influenced the following aspects?



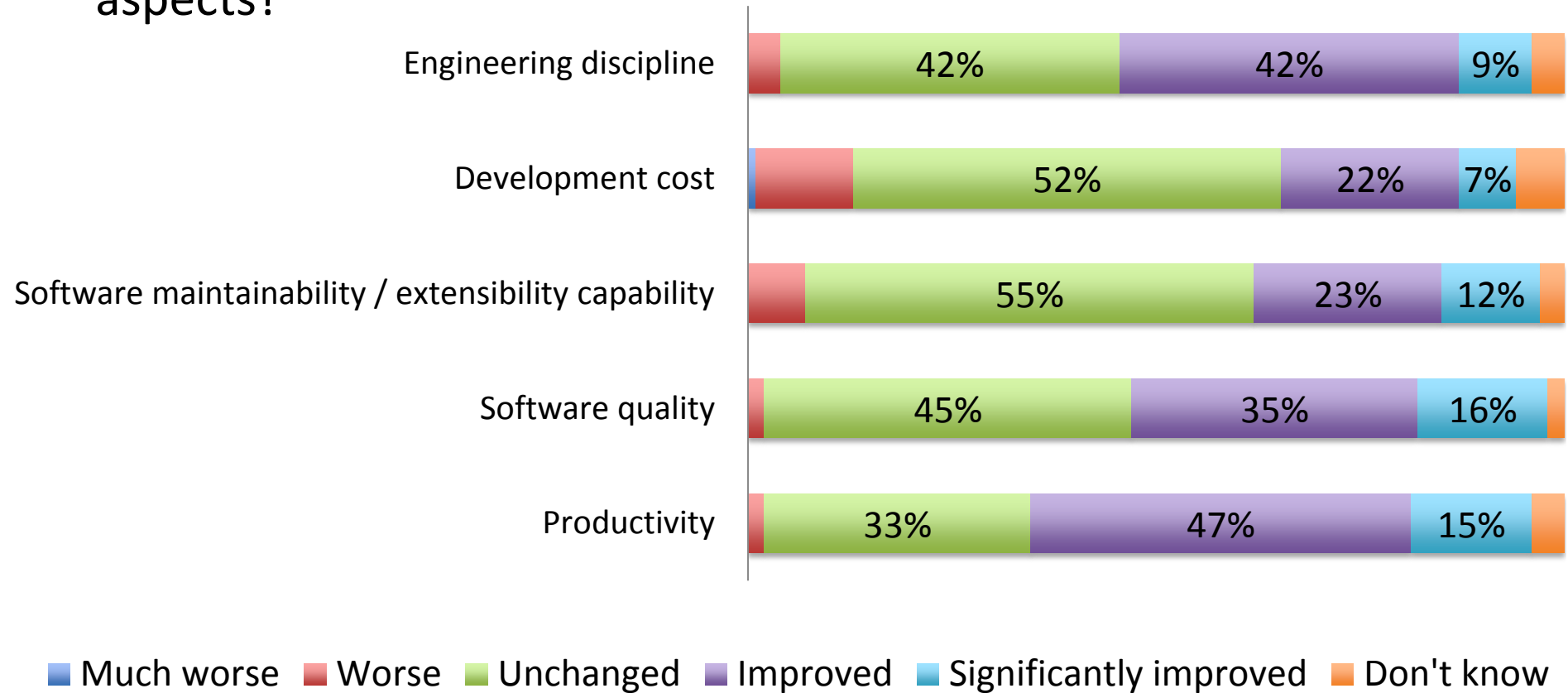
Motivation

SAS shows very promising results:

- significant improvement in the ability to manage changing priorities
- improvement of the development process in general
- much faster time-to-market

SAS Results: Agile Influence

How has agile software development influenced the following aspects?



Motivation

SAS shows very promising results at first view, there are also disappointing findings

- Development cost
- Software quality
- Software maintainability

have not really improved as much as expected

Motivation

Pros:

- Major improvements in project management aspects

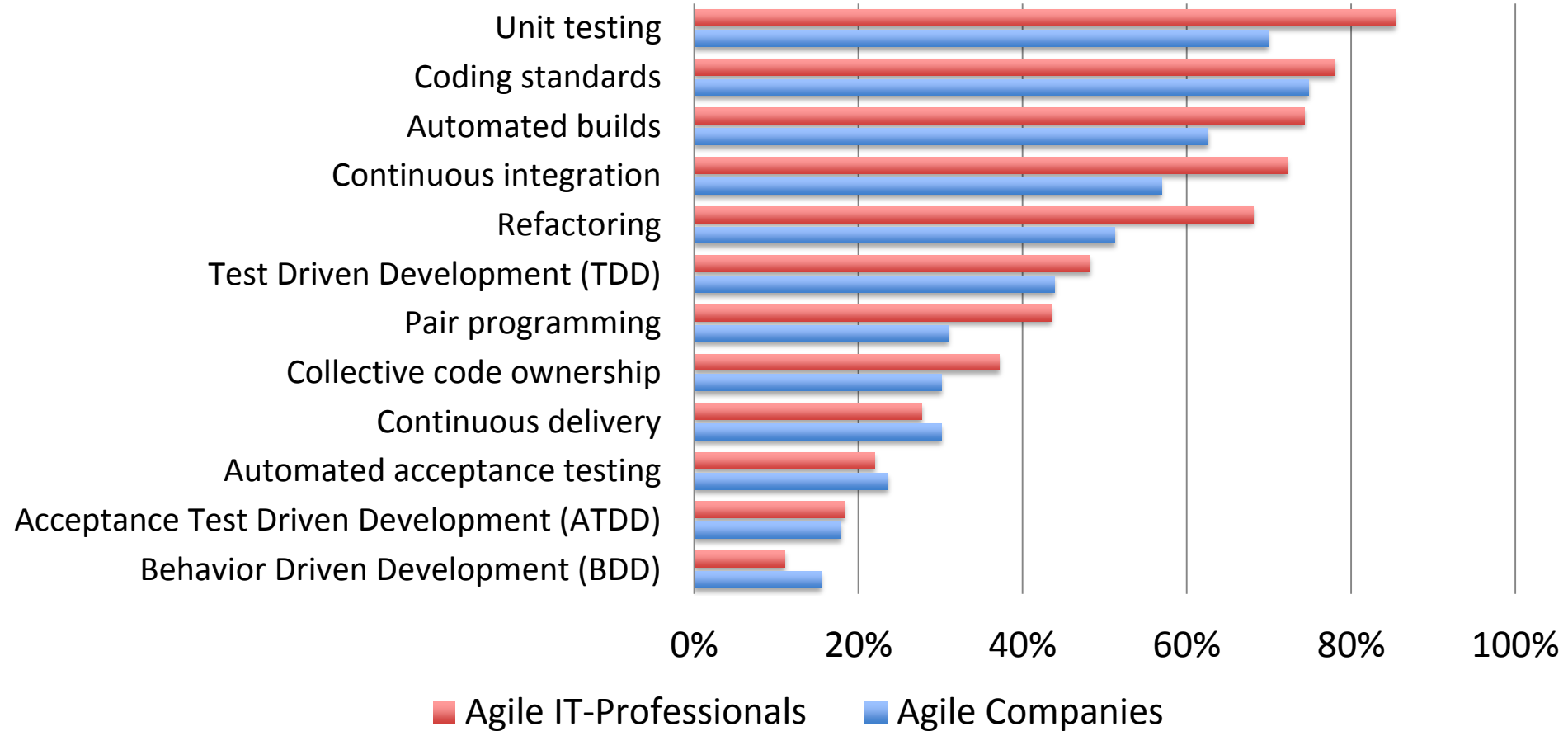
Cons:

- Minor or no improvements in financial, technical or quality aspects

Reasons?

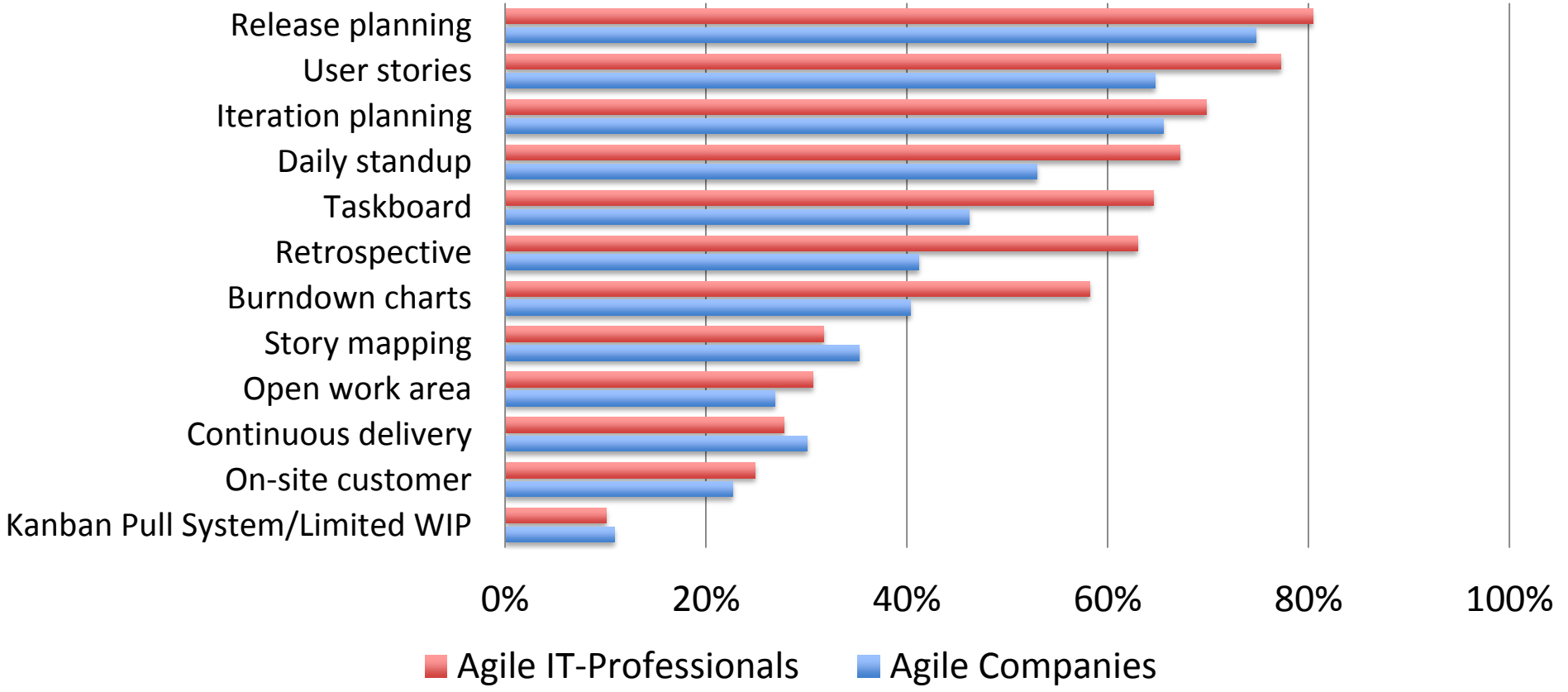
How Agile is Applied

- Engineering Practices



How Agile is Applied

- Managing Practices

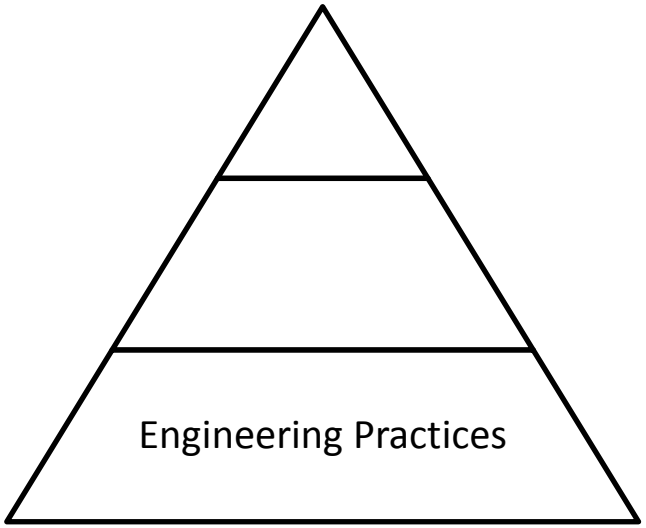


and Agile Education …

- What does this mean for teaching agile software development?
- Which skills and competences do the students need?

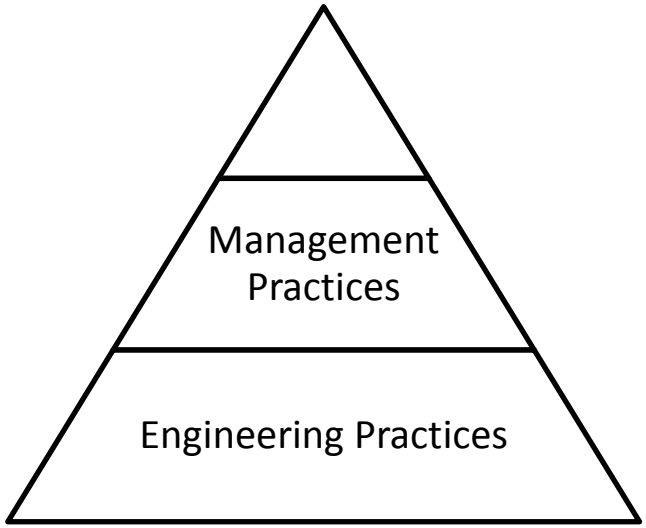
Pyramid of Agile Competences

- Technical skills or *engineering practices*,
  i.e. unit testing, clean code, test-driven development, collective code ownership etc.
- Engineering practices are mostly competences that refer to the single individual
- Software Craftsmanship
- builds the foundation of the pyramid



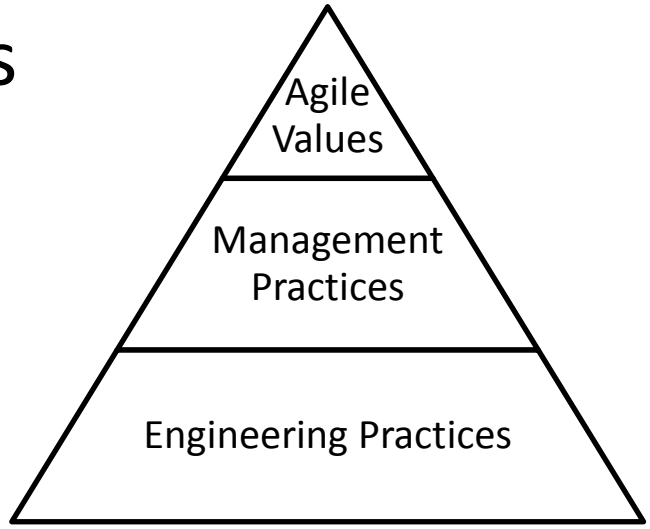
Pyramid of Agile Competences

- Agile *management practices* define how agile projects are organized and run
- I.e. iterative planning, short release cycles, small releases, strong customer involvement and highly interactive teams
- Management practices are typically team aspects, which require appropriate social competences



Pyramid of Agile Competences

- On top of these competences come the *agile values*, which are articulated in the *Agile Manifesto*
- i.e. are characteristics like mutual respect, openness, and courage
- The most difficult to teach



Agile Education Concept

- All three levels should be considered
- Bachelor and Master level
- Appropriate teaching methods should be applied

Agile Software Engineering Course

- 16-week semester class in the last year of the undergraduate level (B.Sc.)
- The students completed one Java programming project in an agile team of six to eight members during the course of the semester
- Per week there were a 2 hours lecture with the whole class and a 2 hours programming workshop with half the class
- 27 students were enrolled

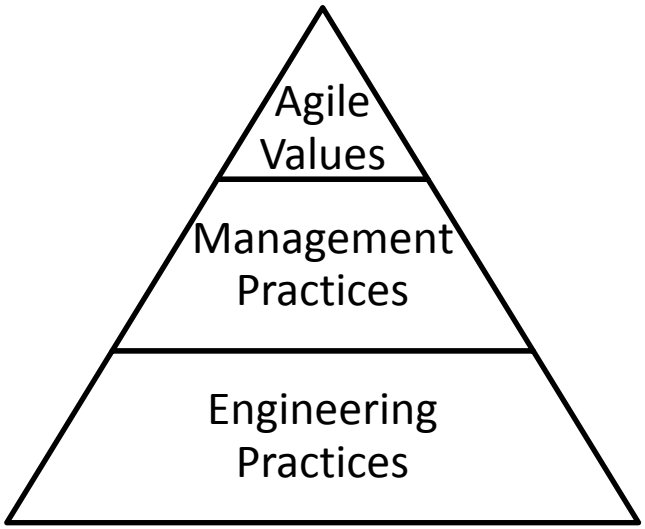
Agile Software Engineering Course

- Distribution of lectures, workshops and self-study:

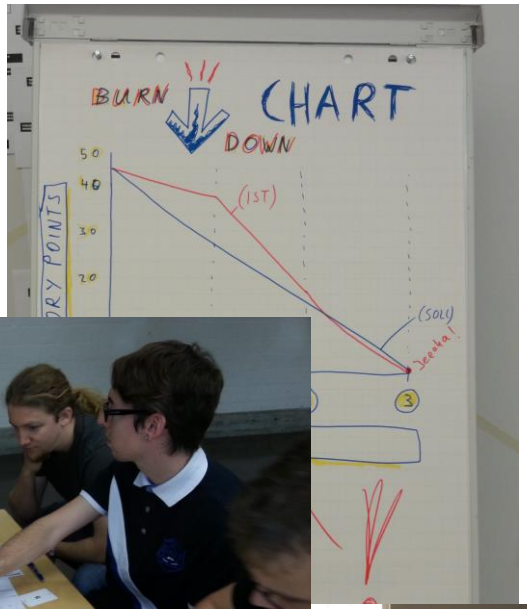
Lectures	32 h
Workshops	32 h
Self-study	56 h
Total	120 h

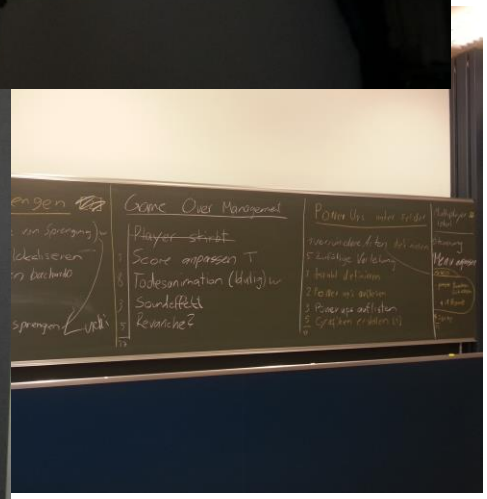
Agile Software Engineering Lecture

W	Lecture	Workshop
1	eXtreme Programming	Installation IDE and Plug-Ins Coding Assessment 1
2	eXtreme Programming Version Control	Coding Assessment 2 Version Control System (SVN)
3	eXtreme Programming Project Automation	Build Scripts (Ant)
4	Continuous Integration	CI (Jenkins Build Server)
5	Unit Testing	JUnit
6	Unit Testing / Mock Objects Clean Code / Code Smells	JUnit EasyMock
7	Refactoring	Refactoring
8	Introduction to Test-Driven Design / Scrum	TDD, The Craftsman articles
9	Scrum	Agile Game Development
10	Scrum	Agile Game Development
11	Agile Estimating and Planning	Agile Game Development Planning Poker
12	Metrics Agile Teams	Agile Game Development Metrics (EMMA)
13	User Stories Agile Principles	Agile Game Development
14	Demonstration of computer games	Agile Game Development



Agile Game Development





Course Evaluation

Items	*Excellent*	*Good*	*Bad*	*Very bad*
The content of this course is…	12	11	0	0
This course was divided into engineering- and management practices and agile values. How would you judge this concept?	12	11	0	0
How did the agile values come across in the lectures and workshops?	1	19	1	0
In the student project, you worked in a Scrum team of 6 to 8 fellow students. How would you judge this concept?	9	11	4	0
How would you judge the workshops in part one?	1	20	1	0
How would you judge the workshops in part two?	6	14	3	0

Course Evaluation

What did you like best about the course?

"… the development of the computer game in a Scrum team".
"… that the material in the course was not only covered theoretically but I also had the opportunity to apply and deepen it in the workshops".
"… the practical relevance".
"… that the topics covered were interesting and important. I had the opportunity to practice the newly learned in the student project. That was great!"

University of Applied Sciences Northwestern Switzerland

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

Kontakt

- Martin Kropp
  Institut für Mobile und Verteilte Systeme
  Fachhochschule Nordwestschweiz
  martin.kropp@fhnw.ch

- Andreas Meier
  Institut für angewandte Informationstechnologie
  Zürcher Hochschule für Angewandte Wissenschaften
  meea@zhaw.ch