

Lessons Learned in Agile Testing @ Siemens

TestNet 2012

LC Nieuwegein, The Netherlands

October 2, 2012

Peter Zimmerer

Principal Engineer

Siemens AG

Corporate Technology

81739 Munich, Germany

peter.zimmerer@siemens.com

<http://www.siemens.com/corporate-technology/>



Siemens Sectors and Divisions

Energy

Divisions

- Fossil Power Generation
- Wind Power
- Solar & Hydro
- Oil & Gas
- Energy Service
- Power Transmission



Healthcare

Divisions

- Imaging & Therapy Systems
- Clinical Products
- Diagnostics
- Customer Solutions



Industry

Divisions

- Industry Automation
- Drive Technologies
- Customer Services



Infrastructure & Cities

Divisions

- Rail Systems
- Mobility and Logistics
- Low and Medium Voltage
- Smart Grid
- Building Technologies
- OSRAM



Corporate functions

- Corp. Finance
- Corp. Technology
- Corp. Developm.
- ...

Corporate Technology

**Corporate Technology drives an
Integrated Technology Company**

Contents

Understand

- Misconceptions
- Challenges – Key questions for transition to agile testing

Practice

- Project examples on agile testing @ Siemens
- Siemens testing conference

Takeaways

In 2001 there was a memorable day in software development ...

Edsger W. Dijkstra, 1968:
Go-to statement
considered harmful

Fred Brooks, 1975:
Mythical Man Month –
*adding manpower to a
late software project
makes it later*

Philip B. Crosby, 1979:
Quality is Free

Fred Brooks, 1986:
No Silver Bullet –
*Essence and Accidents
of Software Engineering*

Manifesto for Agile Software Development

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

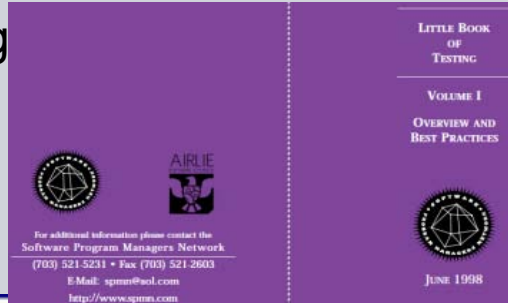
© 2001, the above authors
this declaration may be freely copied in any form,
but only in its entirety through this notice.

<http://agilemanifesto.org/>

Misconceptions (1)

Agile testing is new and completely different*

- Iterative & incremental development, risk-based, value-based, context
- Coding guidelines and standards
- Unit testing



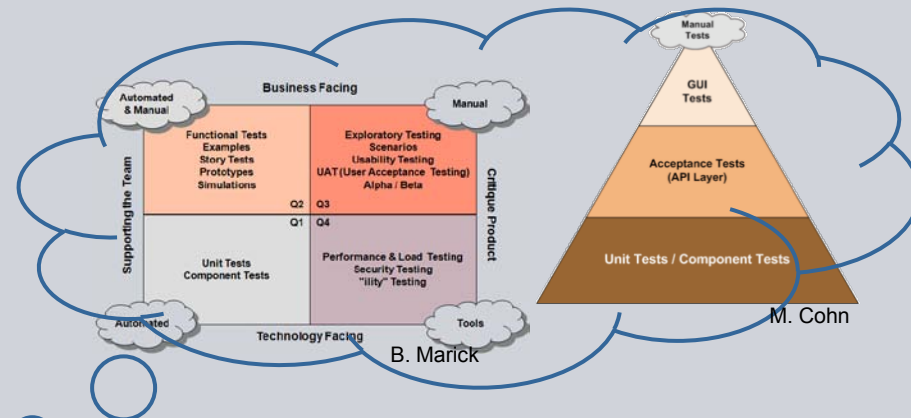
Best Testing Practices (Little Book of Testing, SPMN, 1998):

- Make evaluation everyone's job – a team responsibility
- Make preventive testing part of all specification work
- Use tests as progress and milestone gates
- Test early, test often
- Design and develop testware as deliverable components
- Provide integrated test tooling and infrastructure support

Ten Principles for Agile Testers

- Provide continuous feedback
- Deliver value to the customer
- Enable face-to-face communication
- Have courage
- Keep it simple
- Practice continuous improvement
- Respond to change
- Self-organize
- Focus on people

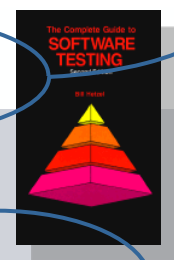
• Enjoy!



Misconceptions (2)

Preventive testing is built upon the observation that one of the most effective ways of specifying something is to describe (in detail) how you would accept (test) it if someone gave it to you.

David Gelperin, Bill Hetzel (<1990)



Agile testing is new and completely different*

- xTDD

The capability to capture test cases at the point of origin – the developer – could be very useful and may become essential. ... test cases should be written at the same time when the problem is specified. The point of origin is not only the developer, it could also be any person who is involved in the development process: ... the designer and even the customer who can demand which tests must be done...

Peter Zimmerer
Siemens Workshop Software Testing
Princeton NJ, USA, June 15, 1994

Everybody can do everything

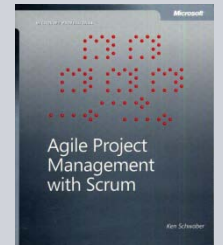
- Roles, responsibilities

Unit testing and acceptance testing – 100% automation trivialization

- Test levels, test environment, global development

Too much focus on process, dogmatism

- Scrum certification, Scrummer-fall, evangelists, self-marketing



Missing focus on engineering

- Good requirements, architecture, design, tests
- Clean (test) code



bbv Agile Poster
2011

***But it contains great ideas and enhancements ...**

Are you really agile ???

Question

Which of the following statements is from the agile manifesto?

- A) Individuals and interactions over contract negotiation**
- B) Individuals and interactions over comprehensive documentation**
- C) Individuals and interactions over processes and tools**
- D) Responding to change over contract negotiation**

Question

Which of the following statements is *not* from the agile manifesto?

- A) Working software over comprehensive documentation**
- B) Responding to change over processes and tools**
- C) Responding to change over following a plan**
- D) Customer collaboration over contract negotiation**

Question

According to the 9th Principle behind the Agile Manifesto:

“Continuous attention to

<what>

and

<what>

enhances agility”

???

Question

According to the 9th Principle behind the Agile Manifesto:

“Continuous attention to

<what>

and

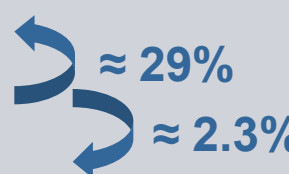
<what>

enhances agility”

- A) ... self-organizing teams ... face-to-face conversation ...**
- B) ... technical excellence ... good design ...**
- C) ... changing requirements ... delivering working software ...**
- D) ... Scrum Master certification ... facilitating retrospectives ...**

Question

Number of hits @  search on September 28, 2012 for

agile	68,600,000	
“agile testing“	551,000	
“agile training“	146,000	 <p>≈ 29%</p> <p>≈ 2.3%</p>
“agile testing training“	41,800	
“testing training“	1,780,000	
“agile improves quality“	???	
“testing improves quality“	???	
“agile testing improves quality“	???	

Challenges – Experiences

- **Preconditions of *agile* (i.e. *agile engineering practices*) not fulfilled**
 - No clean code
 - “Retrospective & passive” reviews vs. pair programming vs. xTDD
 - Inadequate unit testing (→ increased regression risk)
 - Low degree of test automation
 - Deficient sustainability of test automation infrastructure (technical debt)
- **Continuous pace, shortness of iterations, over-commitments**
- **More meetings due to less (or missing) documentation**
- **Volume and speed of change**
 - Drifting of the test basis
 - Missing, insufficient, changing test oracles
 - Regression testing – efforts for analysis of results, bug fixing

Challenges – Key questions for transition to agile testing

- How to **integrate** testers into the developers **team** and provide holistic, high quality results?
- Agile testing and **leadership**: What is the proper role of leadership and management within agile test teams?
- How to define and establish **roles** like test manager, test architect, and test (automation) engineer within an agile team?

- How to blend **traditional** testing approaches with their agile counterparts?
- How to attack a lack of **test automation** in larger-scale projects?
- How to change (reinvent?) traditional **planning** and **metrics** for more agile-centric approaches that engage stakeholders?

- How to handle **legacy** systems?
- How to realize agile in highly **regulated** environments?
- How to introduce and establish agile testing approaches in a **sustainable** way and how to show their **impact** and **value** for the business?

Project example 1 (1)

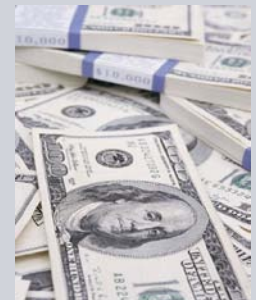
Software platform for unified communications solutions

- **Started in 2003**
- **Risk-driven approach, proactive risk management**
- **Prioritization of features**
- **Early customer feedback, short iterations**
- **Incremental development, early and frequent tests**

- **At the beginning small selected team (20 persons) located in “one room”**

- **Mixture of RUP, Scrum, and XP – UPXS**

- **3 Tester roles**
 - Test Manager
 - System Test Engineer
 - Unit Test Engineer



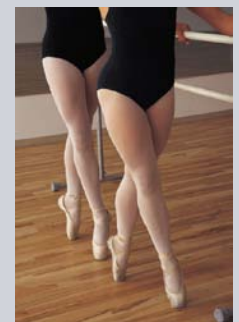
Value Delivery



Collaboration



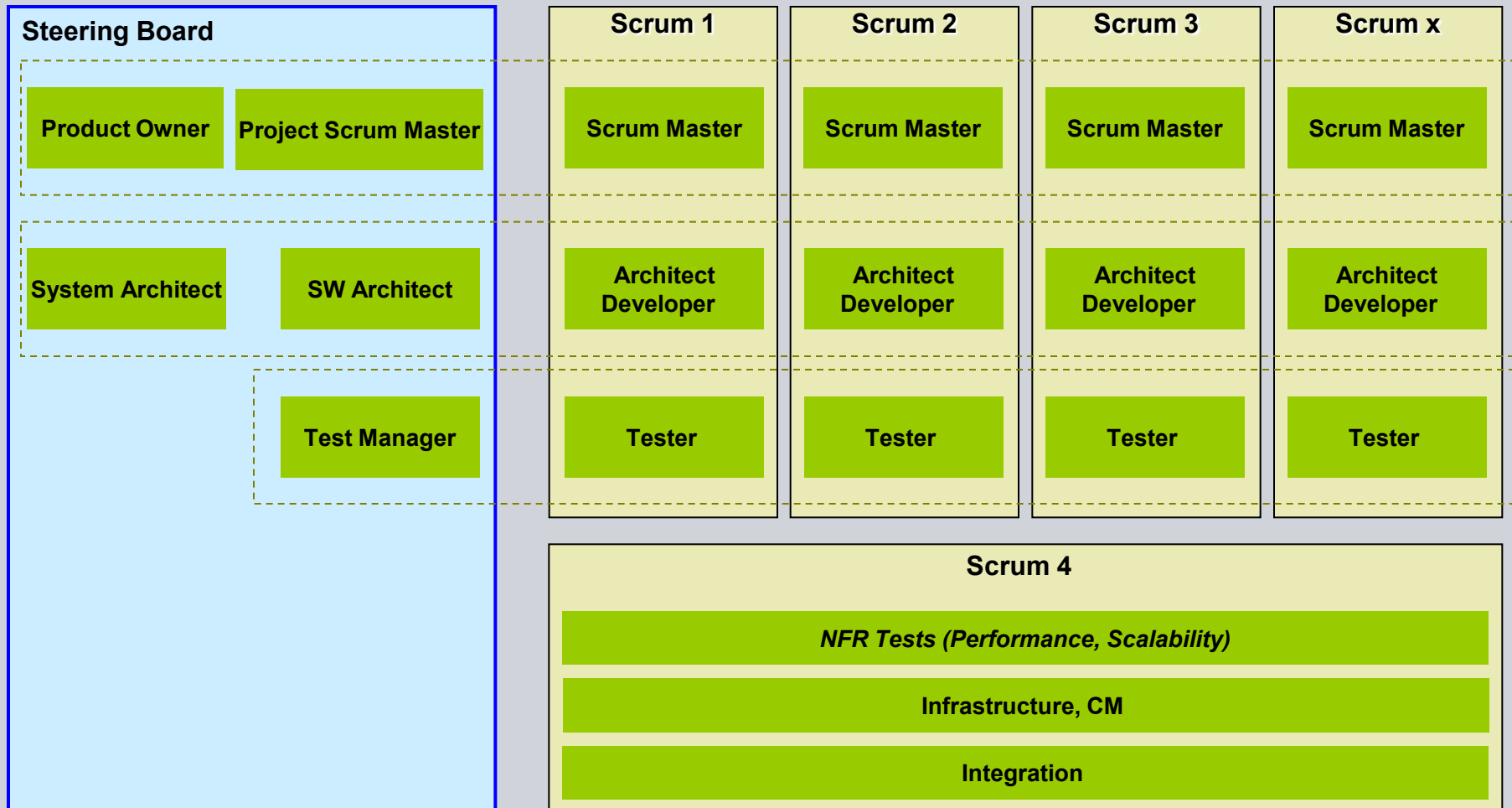
Courage



Discipline

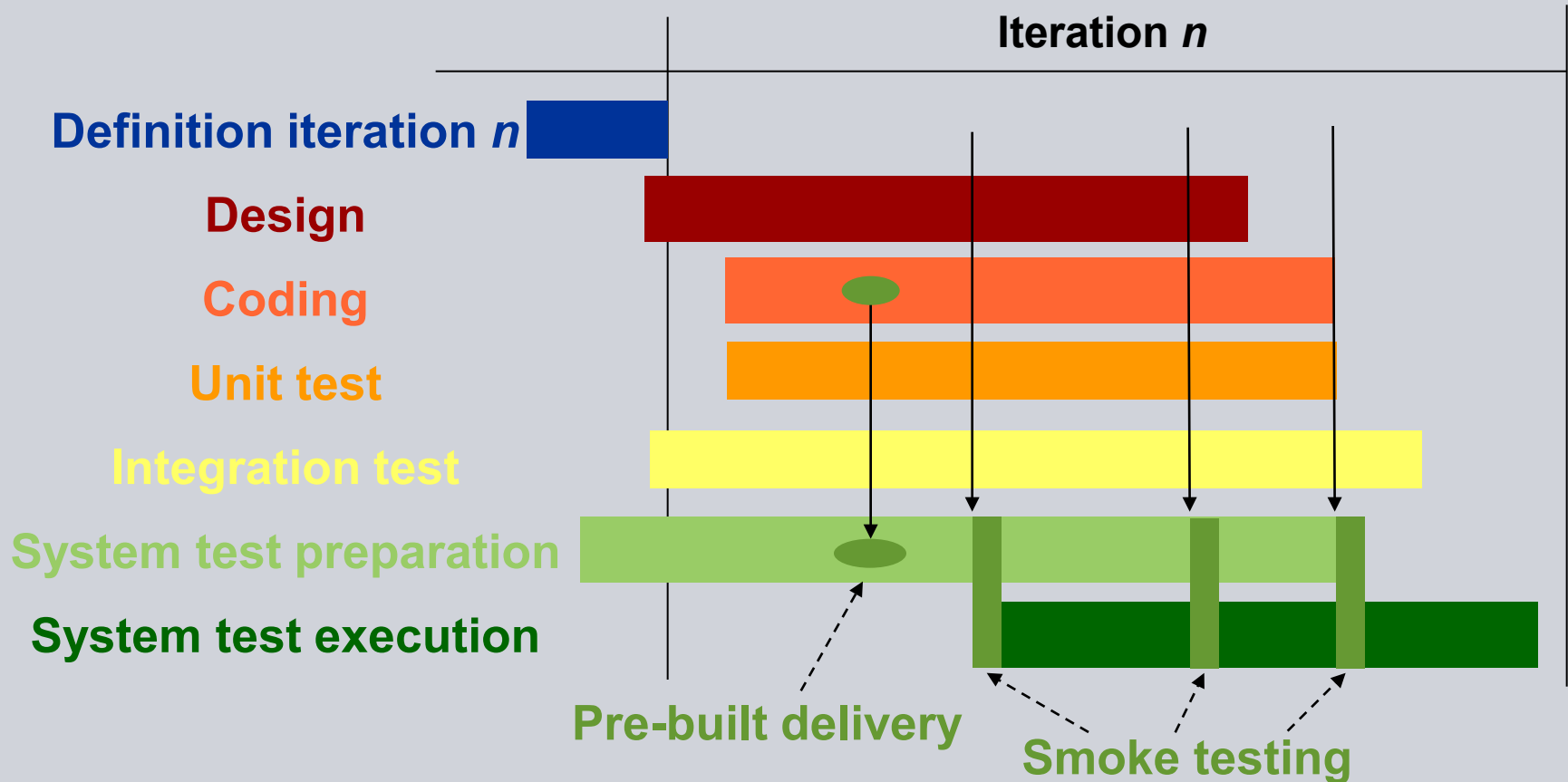
Project example 1 (2)

Project team structure



Project example 1 (3)

Example for a test workflow visualizing xTDD



Siemens testing conference in 2011:

SIEMENS

10 Years after the Agile Manifesto – How Did Testing Benefit From It?

Agile is mainstream ... there are good reasons to be agile today ...

- But there are misconceptions and misinterpretations ...

Quotes from signatories and practitioners about Agile software development 10 years later

- *I don't have a sound-bite answer for you on that.* Kent Beck
- *I'd say we transformed the industry.* Ward Cunningham
- *It's had a pretty significant effect on the industry.* Scott Ambler
- *You still have to do it [Agile] well.... You can do Agile poorly.* Ian McLeod
- *Sometimes, developers can call practices "Agile" when they are really not.* Damon Poole



<http://agilescout.com/10-years-of-agile-are-we-better-yet/>

**Which promises of agile development came true
in the area of testing**

and how far did we benefit from agile adoption

within testing in the past 10 years?

I thank agile for
... ??? ...

I blame agile for
... ??? ...

I thank agile for
... ??? ...

SIEMENS

Good Agile
whole-team view to
software testing

Developers and
Testers work
together

better
communication
between develop-
ers and
tester

Putting the
individual
first!

Team
Communication

- Past Feedback
- Flexibility
- good communication
between different
roles

I thank agile for
... ??? ...

SIEMENS

Early defect
detection.

For getting
testing involved
early.

transparent
& true
plan

I thank agile
for pointing out
that testing is an
important and integral
part of development

...breaking
up boundaries
& formalisms
of tradition.
Waterfall
model

make things
transparent

I blame agile for
... ??? ...

-Agile is used too often
as a Buzzword

... that
it might be used
as an excuse for
sloppiness.

For executives
who think
Agile is Free.

Being too
dogmatic.

Lack of
Documentation

...rising "agile"
expectation in
management
which still thinks
+acts "water-
fall"

I blame agile for
... ??? ...

teams are
working well, but
between teams
there is only few
communication

Bad Agile
Position that Agile
Testing doesn't adapt
to more difficult
contexts.

Working only
^{current} on user stories
limited look ahead
of upcoming user
stories/tasks

Communication
outside the
team/between
teams

only few
best practices
for large-scale
development

I blame agile
for not providing
enough strategy
concepts for trans-
forming "traditional"
to agile.

Takeaways

***Agile* did a great marketing job to recollect & enhance good practices**

***Agile Testing* is here to stay**

**Although some things are not really new
agile testing requires new skills and capabilities – *context matters!***

Upgrade your testing approach by agile (testing) practices *now*

Agile Testing provides new opportunities for testers – go for it!

Be honest to yourself and to your colleagues – are you really *agile*?

**“Some” *process* is needed
but good *engineering practices* are essential for sustaining success**