

# You Can Get There From Here

---



**Challenges, opportunities**

---

---

Slide 1  
5 June 2008

Keith Braithwaite  
© Zühlke 2008

---



## With Zühlke since 2006

- Business Unit Leader—Centre for Agile Practice
- Introduced Agile practices to
  - Wholesale finance
  - Retail finance
  - Mobile telecoms

## Previously:

- WDS Global
  - development manager in Singapore, 24x5 distributed Agile
- Senior Software Engineer at Penrillian
  - Anglo-indian company, embedded systems

# Agile

---



## Who is (doing) Agile?

# Agile

---



Who is (doing) Agile?

What does it mean to you?

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Value Programmer Productivity

---



**A necessary precondition for Agility**

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---



Huh?



## Thought experiment:

- Developer asks for a bigger monitor to improve productivity

# Value Programmer Productivity







## Thought experiment:

- Developer asks for a bigger monitor to improve productivity
- In your organization, what would happen?

# The Monitor Question—Manager's Responses

---



**Request dismissed out of hand**

# The Monitor Question—Manager's Responses

---



## Request dismissed out of hand

- Mental note: “not a team player”

# The Monitor Question—Manager's Responses

---



Referred out of existence

# The Monitor Question—Manager's Responses

---



## Referred out of existence

- Never-ending upwards chain seeking budget authority

# The Monitor Question—Manager's Responses

---



## Referred out of existence

- Never-ending upwards chain seeking budget authority, or
- Ditto seeking permission for a “non-standard” monitor

# The Monitor Question—Manager's Responses

---



## Lip-service analysis leading to refusal

# The Monitor Question—Manager's Responses

---



**Hard-core analysis**



# The Monitor Question—Manager's Responses

---



## Hard-core analysis

- Could go either way, but honest whatever the outcome

# The Monitor Question—Manager's Responses

---



**Lip-service analysis leading to authorization**

# The Monitor Question—Manager's Responses

---



**Authorised without question**

# What is a Monitor Worth?



## Developers aren't cheap

annual cost	days	cost/day
£60,000	200	£300

# What is a Monitor Worth?

---



**Monitors aren't expensive**

# What is a Monitor Worth?

---



**Monitors aren't expensive**

£300 gets you 24"

# What is a Monitor Worth?



---

How much more productive would they need to be?

actual work	201
paid-for work	200
<b>required increase</b>	<b>0.50%</b>

# What Could a New Monitor Achieve?

---



## Czerwinski's study at Microsoft

- between 10% and 44%

## Anderson, University of Utah

- 52% for 24" relative to 18"
- 44% for two 20" relative to 18"





# What Could a New Monitor Achieve?

---

**Call it 20%**

# What Could a New Monitor Achieve?



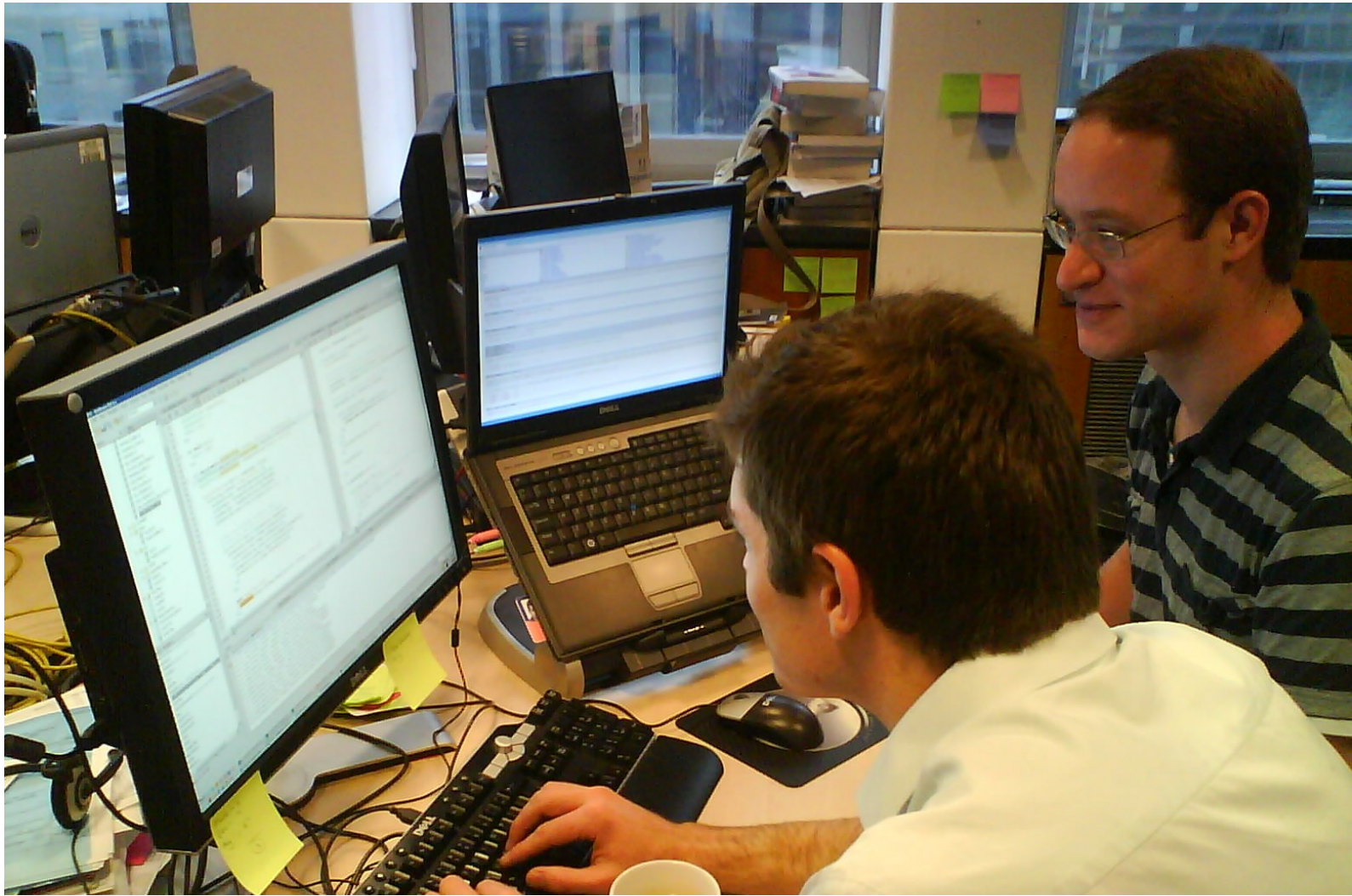
---

## Call it 20%

% increase	20%
paid-for work	200
actual work	240
would cost	£72,000
saving	£11,700

# No Brainer







## Who is doing pair programming?

# Well known, fixed requirements

---



## Introduction of Agile was resisted by one team

- They built emulators of mobile phone handsets

# Well known, fixed requirements

---



## Introduction of Agile was resisted by one team

- Requirement is absolutely fixed
  - Phone does not grow new menus or buttons

# Well known, fixed requirements

---



## Introduction of Agile was resisted by one team

- Estimation data was solid
  - They'd built dozens of these things, all the same way



# Well known, fixed requirements

---



## Introduction of Agile was resisted by one team

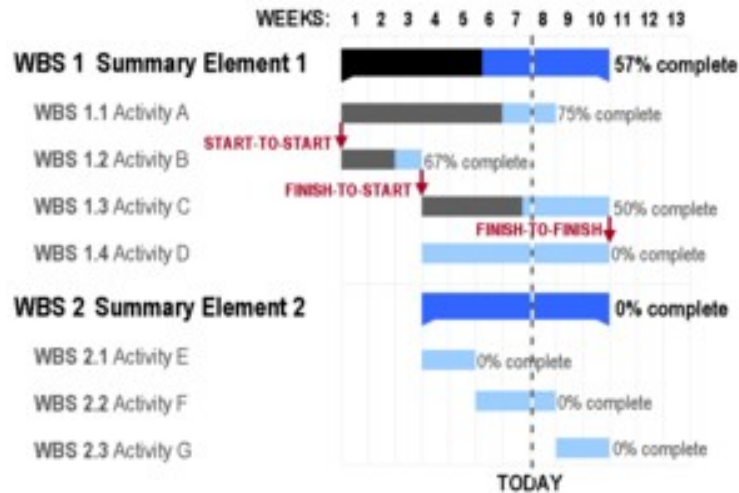
- Planning and tracking by Gantt chart was perfect
  - They were right to resist: all the pain of change for very little gain

# Well known, fixed requirements



## Introduction of Agile was resisted by one team

- Planning and tracking by Gantt chart was perfect
  - They were right to resist: all the pain of change for very little gain

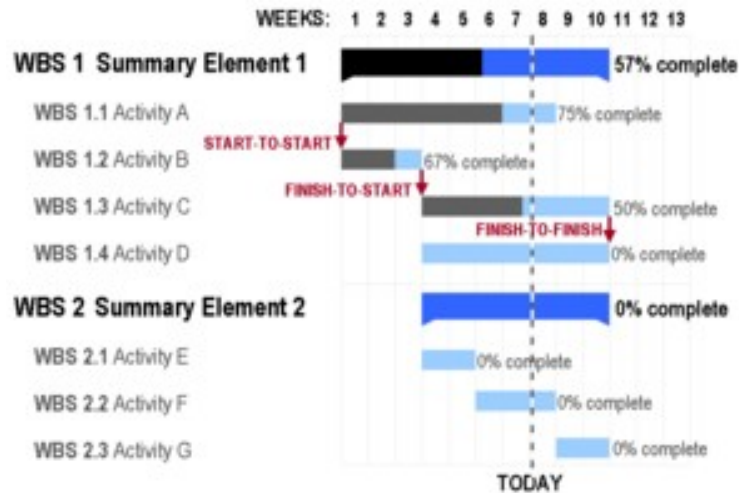


# Well known, fixed requirements



## Introduction of Agile was resisted by one team

- Planning and tracking by Gantt chart was perfect
  - They were right to resist: all the pain of change for very little gain



# Well known, fixed requirements

---



Introduction of Agile was resisted by one team

# and they were right!

# Who here has a project like that?



Who here has a project like that?

---

**zühlke**  
empowering ideas

**No, you don't**

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Test-first Process Improvement

---



**Know what you want to achieve**

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Test-first Process Improvement



Know what you want to achieve



**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008



# Test-first Process Improvement

---



## Know what you want to achieve

- “Be better” is not a goal
- Neither is “be Agile”

# Test-first Process Improvement

---



**Know what you want to achieve**

**It could be**

- Reduced time-to-value
- Reduced defect rates
- Improved customer/user satisfaction
- Improved accuracy of planning
- The list goes on...

# Test-first Process Improvement

---



**Know what you want to achieve**

**It could be**

- Reduced time-to-value
- Reduced defect rates
- Improved customer/user satisfaction
- Improved accuracy of planning
- The list goes on...

**Ask what these things are *worth***



## Who is using “checked examples”?

# Example: Time-to-Value

---



Once Fund Manager's internal IT department

# Example: Time-to-Value

---



Once Fund Manager's internal IT department

6-month release cycle

- *Best case*
- Manual testing
- Sometimes low quality

# Example: Time-to-Value

---



Once Fund Manager's internal IT department

Goal: release new features monthly

# Example: Time-to-Value

---



Once Fund Manager's internal IT department

Value: calculated through ability to bring new kinds of business to market earlier



# Example: Time-to-Value

---



Once Fund Manager's internal IT department

Tracking: Within a year releases could be made every 6 weeks, and falling

# Example: Time-to-Value

---



Within a year releases could be made every 6 weeks, and falling

This was not achieved by tricky scheduling

# Example: Time-to-Value

---



Within a year releases could be made every 6 weeks, and falling

This was not achieved by self-organising voodoo

# Example: Time-to-Value

---



Within a year releases could be made every 6 weeks, and falling

This was achieved by doing the damn thing right

# Example: Time-to-Value

---



Within a year releases could be made every 6 weeks, and falling

This was achieved by doing the damn thing right

hundreds of checked examples for each release

# Example: Time-to-Value

---



Within a year releases could be made every 6 weeks, and falling

This was achieved by doing the damn thing right

hundreds of checked examples for each release

those turned into automated acceptance tests

# Example: Time-to-Value

---



**Within a year releases could be made every 6 weeks, and falling**

**This was achieved by doing the damn thing right**

**hundreds of checked examples for each release**

**those turned into automated acceptance tests**

**2 developers, 1 tester and a user in a room, and get it done**



## Who is doing continuous integration?



# What Will Happen Along the Way?



# What Will Happen Along the Way?

---

**zühlke**  
empowering ideas



---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Some Developers Will Get Upset



# Some Developers Will Get Upset



# Some Developers Will Get Upset

---



## Collective Code Ownership

- Some developers have a huge amount of ego invested in “their” code

# Some Developers Will Get Upset

---



## No recognition for “Rock stars”

- Agile prefers the consistent team contributor over the temperamental genius

# Some Developers Will Get Upset

---



## Testing

- Some programmers consider themselves too clever to need to test

# Some Developers Will Get Upset

---



## Testing

- Some developers resent having to postpone “real work” to fix the build



# Intermezzo



- 
- who knows their build status?
  - who's build is always green?

# It Isn't Broken so they aren't Fixing it

---



**One Team adopted automated testing**

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# It Isn't Broken so they aren't Fixing it



## One Team adopted automated testing

Expected Non-hedge Orders						
Value Date	Primary CCY	Type	Against Ccy	Amount	FxType	FxRate
19-Mar-2008	CHF	FBY	USD	10000.00	Non Hedge	1.0000
19-Mar-2008	DKK	FBY	USD	10000.00	Non Hedge	1.0000
19-Mar-2008	GBP	FBY	USD	5000.00	Non Hedge	2.0000
19-Mar-2008	NOK	FBY	USD	10000.00	Non Hedge	1.0000
20-Mar-2008 <i>missing</i>	CHF	FBY	USD	10000.00	Non Hedge	1.0000
20-Mar-2008 <i>missing</i>	DKK	FBY	USD	3234.56	Non Hedge	1.0000
20-Mar-2008 <i>missing</i>	EUR	FBY	USD	785.71	Non Hedge	1.4000
20-Mar-2008 <i>missing</i>	GBP	FBY	USD	8889.76	Non Hedge	2.0000
20-Mar-2008	JPY	FSL	USD	4032768	Non Hedge	0.0010
20-Mar-2008 <i>missing</i>	NOK	FSL	USD	5555.12	Non Hedge	1.0000
20-Mar-2008 <i>missing</i>	SEK	FBY	USD	4016.23	Non Hedge	1.0000
25-Mar-2008 <i>missing</i>	JPY	FSL	USD	1000000	Non Hedge	0.0010
21-Mar-2008 <i>surplus</i>	CHF	FBY	USD	20000.0	Non Hedge	1.0
21-Mar-2008 <i>surplus</i>	DKK	FBY	USD	13234.56	Non Hedge	1.0
21-Mar-2008 <i>surplus</i>	EUR	FSL	USD	3658.73	Non Hedge	1.4
21-Mar-2008 <i>surplus</i>	GBP	FBY	USD	23334.64	Non Hedge	2.0
21-Mar-2008 <i>surplus</i>	JPY	FSL	USD	1.4032768E7	Non Hedge	0.0010
21-Mar-2008 <i>surplus</i>	NOK	FBY	USD	4444.88	Non Hedge	1.0
21-Mar-2008 <i>surplus</i>	SEK	FSL	USD	1967.54	Non Hedge	1.0

Agile Offshore

5 June 2008

Keith Braithwaite  
© Zühlke 2008

# It Isn't Broken so they aren't Fixing it

---



## One Team adopted automated testing

- Braced for the usual post-release nightmare

# It Isn't Broken so they aren't Fixing it

---



## One Team adopted automated testing

- Braced for the usual post-release nightmare
- But the phones did not ring

# It Isn't Broken so they aren't Fixing it

---



## One Team adopted automated testing

- A defect-free release
- The approach now mandatory for all new build



## Who is letting tests drive their development?

# Some Managers Will Get Upset

---



---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---



# Some Managers Will Get Upset

---



## No more Gantt charts

- No %complete figures to fudge
- No assignment of tasks
- No planning daily activities six months in advance

# Some Managers Will Get Upset

---



## No more Gantt charts

- No %complete figures to fudge
- No assignment of tasks
- No planning daily activities six months in advance

**Instead: real decisions, real responsibility, real management**

# The Truth Will Set You Free



# The Truth Will Set You Free

---



But you might not like that

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Transparency

---



---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# Transparency

---



## One project ditched the Gantt chart

---

**Agile Offshore**

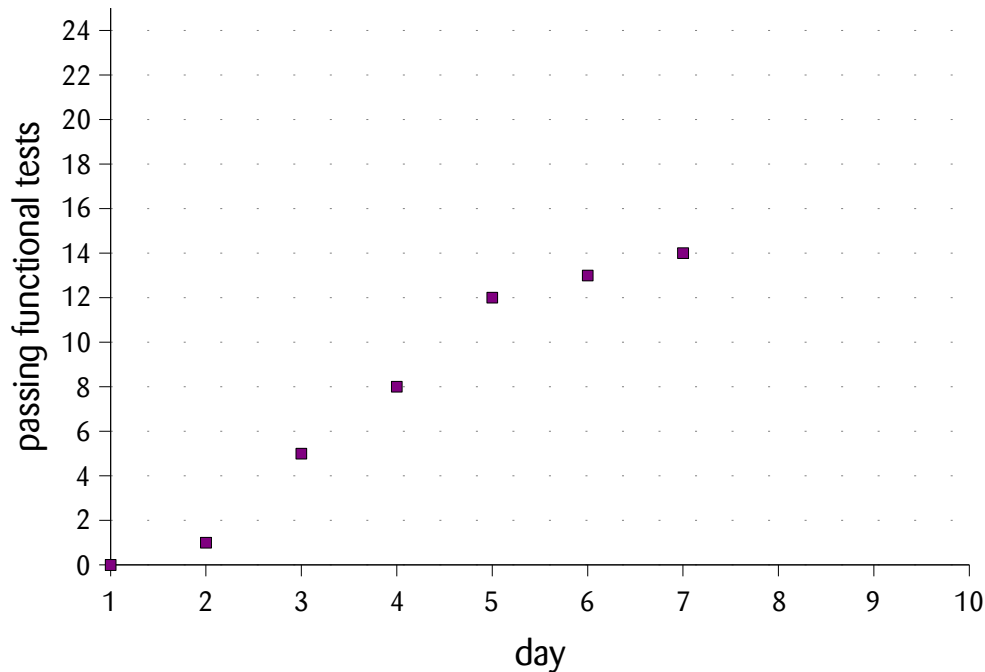
5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

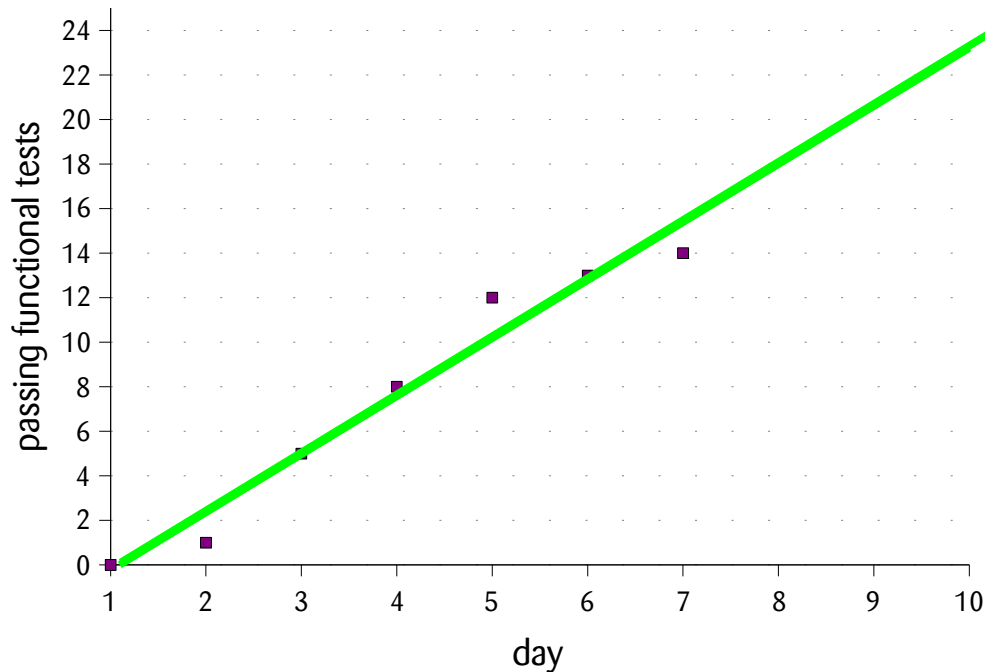
## One project ditched the Gantt chart

- Reported number of (functional) tests written, passing



## One project ditched the Gantt chart

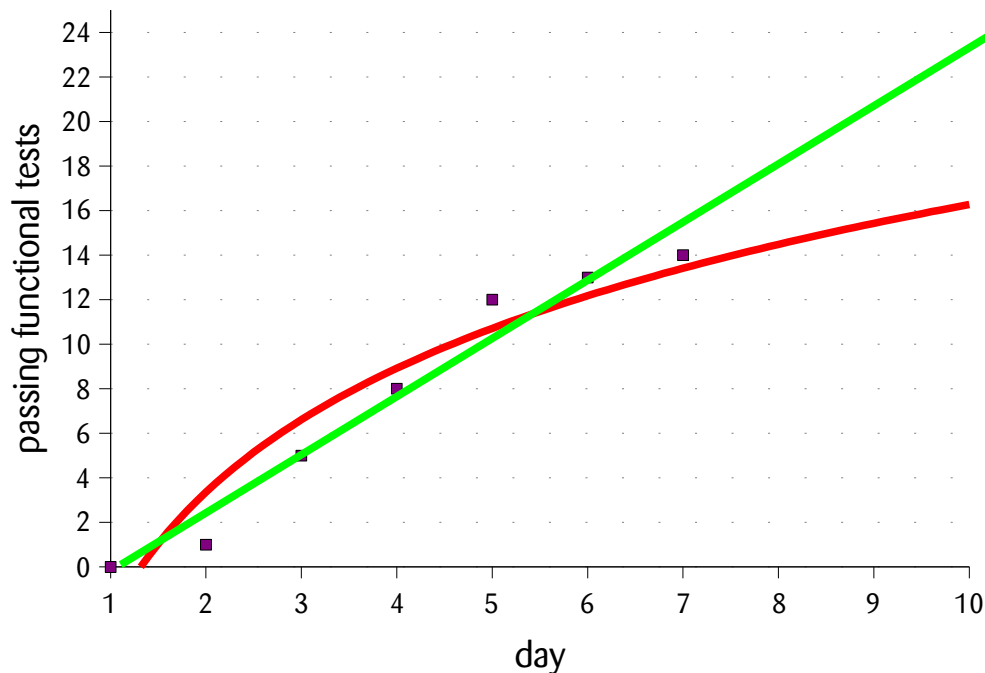
- Reported number of (functional) tests written, passing
  - Trended over time





## One project ditched the Gantt chart

- Reported number of (functional) tests written, passing
  - Trended over time
  - Projections based on a few likely scenarios



# Transparency

---



“The first time we’ve ever had a status report form a project that we believe”

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

**“The first time we’ve ever had a status report from a project that we believe”**

- Therefore, realistic intervention possible
- project did improve

# People About the Place will Get Upset

---



Being in close proximity to an Agile team can be a strain



# People About the Place will Get Upset

---



Being in close proximity to an Agile team can be a strain

- They're talking *all the time*

# People About the Place will Get Upset

---



## Being in close proximity to an Agile team can be a strain

- They're talking *all the time*
- They're never doing any *real work*
  - Translation: they appear to be enjoying themselves

# People About the Place will Get Upset

---



## Being in close proximity to an Agile team can be a strain

- They're talking *all the time*
- They're never doing any *real work*
  - Translation: they appear to be enjoying themselves
- They've got all this untidy stuff stuck on the walls

# People About the Place will Get Upset

---



## Being in close proximity to an Agile team can be a strain

- They're talking *all the time*
- They're never doing any *real work*
  - Translation: they appear to be enjoying themselves
- They've got all this untidy stuff stuck on the walls
- How come *they* get the big monitors?



# People About the Place will Get Upset

---



## Being in close proximity to an Agile team can be a strain

- They're talking *all the time*
- They're never doing any *real work*
  - Translation: they appear to be enjoying themselves
- They've got all this untidy stuff stuck on the walls
- How come *they* get the big monitors?
- What's so special about those \* ^ & # \$ \* § & 's?

# People About the Place will Get Upset

---



## Being in close proximity to an Agile team can be a strain

- They're talking *all the time*
- They're never doing any *real work*
  - Translation: they appear to be enjoying themselves
- They've got all this untidy stuff stuck on the walls
- How come *they* get the big monitors?
- What's so special about those \* ^ & # \$ \* § & 's?



# What *is* so Special About Them?



---

They deliver. Every time.

# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

Other groups can respond badly

# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

Other groups can respond badly

- Envy

# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

Other groups can respond badly

- Envy
- Jealousy

# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

Other groups can respond badly

- Envy
- Jealousy
- Not happy at being shown to be incompetent
- Old established ideas shown to be wrong
  - Oh, how much money have the wasted over the years...



# What *is* so Special About Them?

---



They deliver. Every time.

This can be *big* shock to the rest of the organization

Other groups can respond badly

- Envy
- Jealousy
- Not happy at being shown to be incompetent

# When Are You Done?



# When Are You Done?

---



When there are no more opportunities for improvement



---

**Agile Offshore**  
Slide 99  
5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# When Are You Done?

---



**Don't hold your breath!**



---

**Agile Offshore**  
Slide 100  
5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# When Are You Done?

---



Indicators that you're on the right track:



---

**Agile Offshore**  
Slide 101  
5 June 2008

Keith Braithwaite  
© Zühlke 2008

---

# When Are You Done?

---



## Indicators that you're on the right track:

- High customer/user satisfaction



---

**Agile Offshore**  
Slide 102  
5 June 2008

# When Are You Done?

---



## Indicators that you're on the right track:

- High developer satisfaction



---

**Agile Offshore**  
Slide 103  
5 June 2008

# When Are You Done?

---



## Indicators that you're on the right track:

- Complete visibility of project status



---

**Agile Offshore**  
Slide 104  
5 June 2008



# When Are You Done?

---



## Indicators that you're on the right track:

- Credible



---

**Agile Offshore**  
Slide 105  
5 June 2008

# When Are You Done?

---



## Indicators that you're on the right track:

- High customer/user satisfaction
- High developer satisfaction
- Complete visibility of project status
- Credible



---

**Agile Offshore**  
Slide 106  
5 June 2008



- 
- Who is doing retrospectives?

# When is it Working?

---



## Signs of Progress

# When is it Working?

---



## Signs of Progress

- Estimates Converge

# When is it Working?

---



## Signs of Progress

- Estimates Converge
- Quality remains high over long periods
  - Internal and external

# When is it Working?

---



## Signs of Progress

- Estimates Converge
- Quality remains high over long periods
  - Internal and external
- Teams find and action new approaches that work better

# When is it Working?

---



**If you “adopt Agile” and after a year are still doing it the same way, you’re doing it wrong**

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---



# Tell Me Your Stories

---



**email:** [kbr@zuhlke.com](mailto:kbr@zuhlke.com)

**blog:** <http://peripateticaxiom.blogspot.com>

**website:** <http://www.keithbraithwaite.demon.co.uk/professional/>

**twitter:** [http://twitter.com/keithb\\_b](http://twitter.com/keithb_b)

# Questions?

---

**zühlke**  
empowering ideas

---

**Agile Offshore**

5 June 2008

Keith Braithwaite  
© Zühlke 2008

---