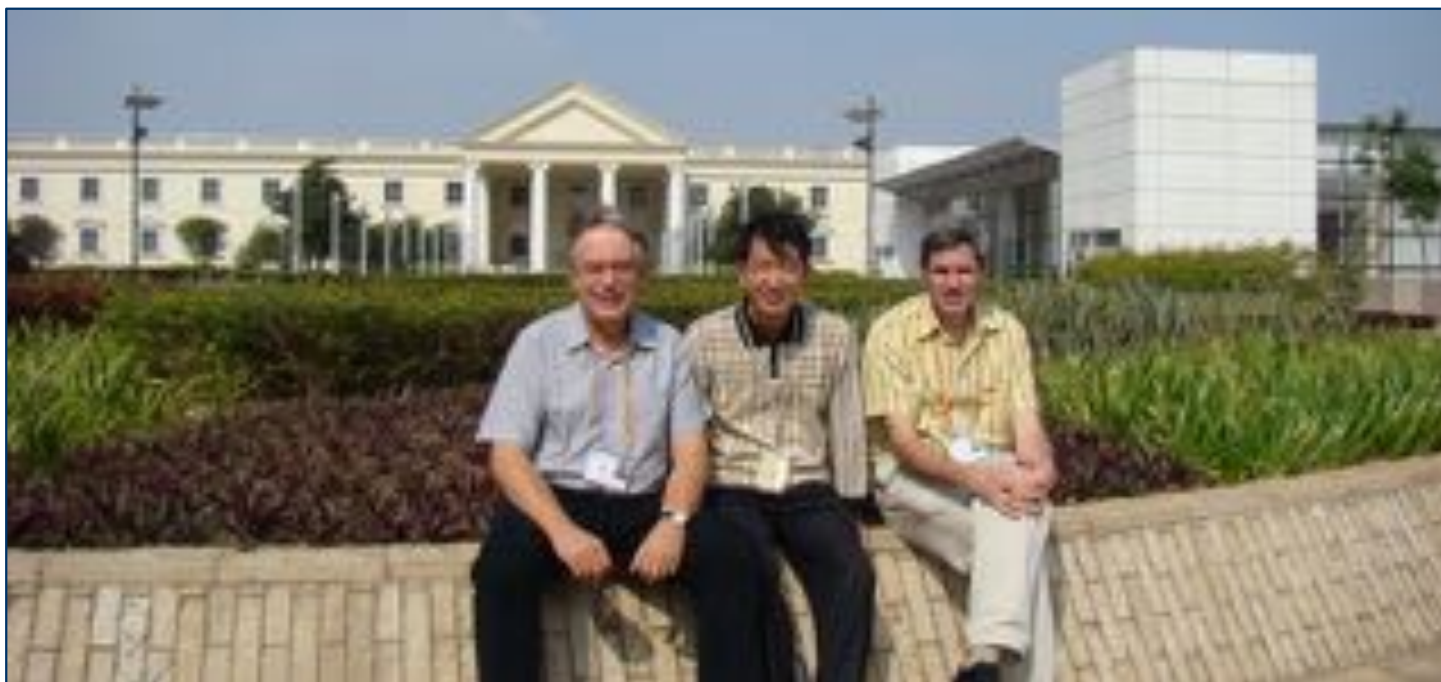


7000 testers, of 9000?

Laatste telling 26000...



Out-of-comfortzone TPI, een 10 jarig jubileum bij Huawei, China.

“

2007 Shenzhen

- TPI original (boek)



Test Maturity Matrix

Impact of dependencies

	Key Area / Scale	0	1	2	3	4	5	6	7	8	9	10	11	12	13
1	Test Strategy		A					B				C		D	
2	Life-cycle Model		A			B									
3	Moment of Involvement		A	A				B				C		D	
4	Estimating and Planning				A							B			
5	Test Specification Techniques		A		B										
6	Static Test Techniques					A		B							
7	Metrics						A			B			C		D
8	Test Automation					A			B			C			
9	Test Environment				A				B						C
10	Office Environment				A										
11	Commitment and Motivation		A				B						C		
12	Test functions and Training				A			B				C			
13	Scope of Methodology					A						B			C
14	Communication			A		B							C		
15	Reporting		A			B		C					D		
16	Defect Management		A				B		C						
17	Testware Management			A			B				C				D
20	Low-level Testing					A		B		C					

TPI original is best streng

2007 Shenzhen

- TPI original (boek)
- Huh *geen* boek?
- Eerst management
- Na een paar interviews wist hij het al...
- Proefballonnetjes
- Jij bent technisch Kees, doe jij dit interview?



Samenwerken met 'rolmodellen' erg leerzaam

2008/9 60 dagen China

- De poet werd verdeeld



Proposed RBT Process for Huawei
(Zhang Bo/Kees Blokland)

Security Level:

	A	B	C	D	E	F
	system part	quality characteristic	risk (what can go wrong and what is the impact)	probability	impact	risk class
1						
2	HA(Include DPS) (Base on DS)		从实现机制的角度评估结果和从特性耦合的角度评估的结果不同。			
3	特性耦合情况下的主备切换 (应该放到每个具体特性的FSA中考虑) switch will influence other function	Functionality	实现机制稳定,但是对具体特性的主备切换可能存在很多问题并且影响很大。 The mechanism of switch is good, but a lot of function will be test when switch	H	H	A
4	主备单板数据批量备份 master/slave board data batch backup	Functionality	备份不成功,导致配置数据不一致,切换后导致业务中断,静态业务不受影响,交互特性多。 Backup failure, cause the configuration data of master board is different from slave board, so cause the service disrupted when switch happened. a lot of function interact with this function point	L	H	B
5	主备单板数据实时备份 master/slave board data real time backup	Functionality	备份不成功,导致配置数据不一致,切换后导致业务中断,静态业务不受影响,但是后面可能在批量备份时发现数据不一致的情况,交互特性多。 Backup failure, cause the configuration data of master board is different from slave board, so cause the service disrupted when switch happened. a lot of function interact with this function point	H	H	B

Figure 7 Feature Level PRA example

Aardbeving Sichuan 2008

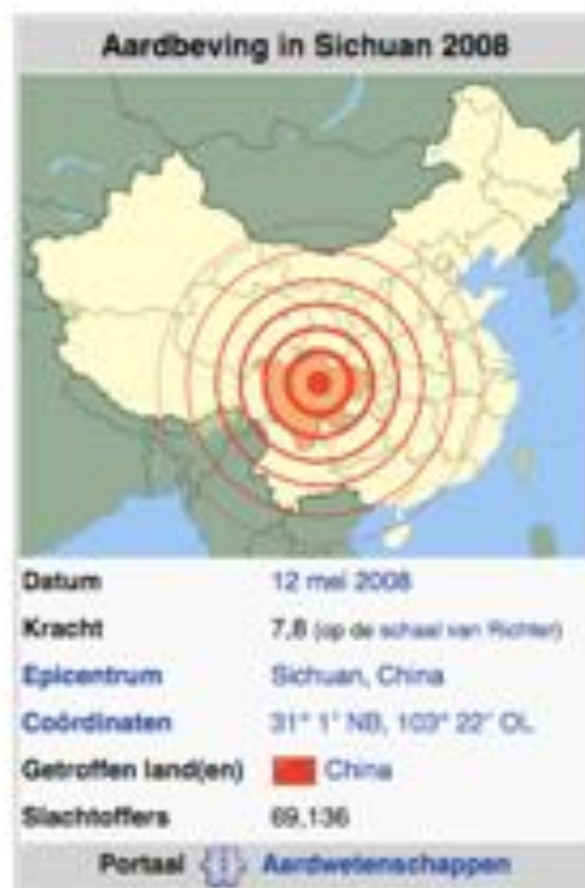
De zware **aardbeving in Sichuan** vond plaats om 14:28 uur (lokale tijd) op maandag 12 mei 2008 in de Chinese provincie Sichuan. Het epicentrum lag in het arrondissement **Wenchuan**. De aardbeving had een kracht van 7,8 op de **Schaal van Richter** en richtte zware schade aan. In sommige gebieden zijn hele dorpen en steden van de kaart geveegd. Er zijn volgens officiële cijfers 69.136 doden en 374.061 gewonden geteld.

In de miljoenenstad **Chengdu** (90 kilometer ten noordwesten van het epicentrum) werd de aardbeving ook hevig gevoeld en was er ook schade. Bovendien was de aardbeving in andere gebieden van **China** en in andere landen voelbaar. Volgens de eerste schatting was er voor de economie een schade van 8 miljard **euro**.

Tijdens de aardbeving vonden **aardverschuivingen** plaats, waardoor rivieren geblokkeerd werden en meren werden gevormd. Het **Tangjashanmeer** is daar een voorbeeld van. Meer dan honderduizend mensen in het rampgebied moesten daarom geëvacueerd worden.

Zie ook [bewerken]

- **Tangjashanmeer**



2008/9 60 dagen China

- De poet werd verdeeld
- Acceptatietest
- RBT
- Technieken
- Estimation
- Agile TPI
- Vriendengroep



Proposed RBT Process for Huawei
(Zhang Bo/Kees Blokland)

Security Level:

	A	B	C	D	E	F
	system part	quality characteristic	risk (what can go wrong and what is the impact)	probability	impact	risk class
1						
2	HA (Include DPS) (Base on DS)		从实现机制的角度评估结果和从特性耦合的角度评估的结果不同。			
3	特性耦合情况下的主备切换 (应该放到每个具体特性的PSA中考虑) switch will influence other function	Functionality	实现机制稳定, 但是对具体特性的主备切换可能存在很多问题并且影响很大。 The mechanism of switch is good, but a lot of function will be test when switch	H	H	A
4	主备单板数据批量备份 master/slave board data batch backup	Functionality	备份不成功, 导致配置数据不一致, 切换后导致业务中断, 静态业务不受影响, 交互特性多。 Backup failure, cause the configuration data of master board is different from slave board, so cause the service disrupted when switch happened, a lot of function interact with this function point	L	H	B
	主备单板数据实时备份 master/slave board data real time backup	Functionality	备份不成功, 导致配置数据不一致, 切换后导致业务中断, 静态业务不受影响, 但是后面可能在批量备份时发现数据不一致的情况, 交互特性多。 Backup failure, cause the configuration data of master board is different from slave board, so cause the service disrupted when switch happened, a lot of function interact	H	H	B

De context tekent zich af

Beijing 2011

- Upgrade naar TPI Next



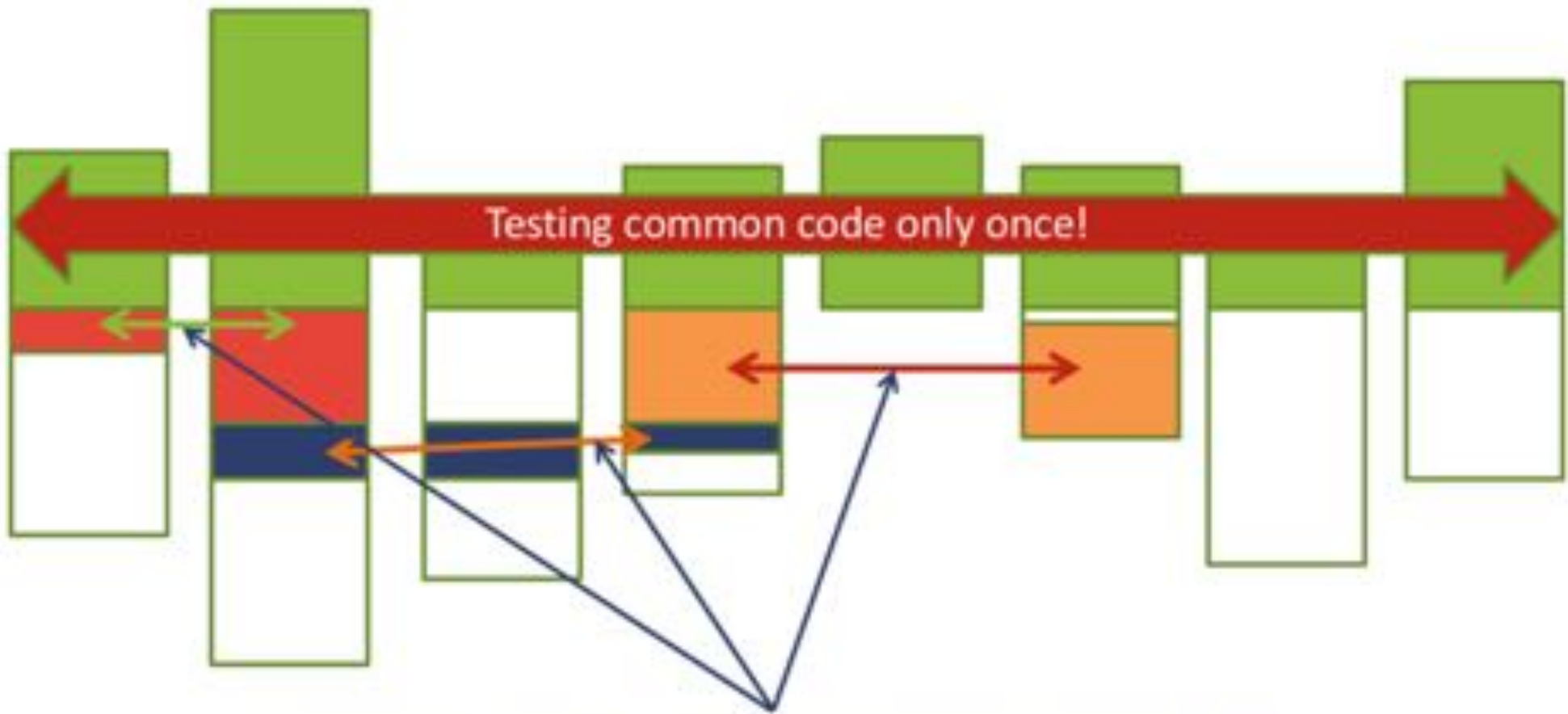
		Initial	Controlled			Efficient				Optimizing			
1	Stakeholder commitment		A	B	B	C	F	H	H	K	M	M	
2	Degree of involvement		A	B	C	E	H	H	J	L	L		
3	Test strategy		A	A	B	E	F	F	H	K	L		
4	Test organization		A	D	D	E	I	I	J	J	K	L	L
5	Communication		B	C	C	D	F	F	J	M	M		
6	Reporting		A		C	C	F	G	G	K	K		
7	Test process management		A	A	B	B	G	H	J	K	M		
8	Estimating and planning		B	B	C	C	G	H	I	I	K	L	L
9	Metrics		C		C	D	G	H	H	I	K	K	
10	Defect management		A	A	B	D	F	F	H	J	K	L	L
11	Testware Management		B	B	D	E	I	I	J	L	L	L	
12	Methodology practice		C		D	E	F	H	J	J	M	M	
13	Tester professionalism		D	D	E	E	G	G	I	I	K	K	M
16	Test environment		C	D	D	E	G	H	J	J	L	M	M

TPI Next transparanter, motiveert meer

Beijing 2011

- Upgrade naar TPI Next
- Complexiteit
- MBT
- Outsourcing
- Requirements
- Agile

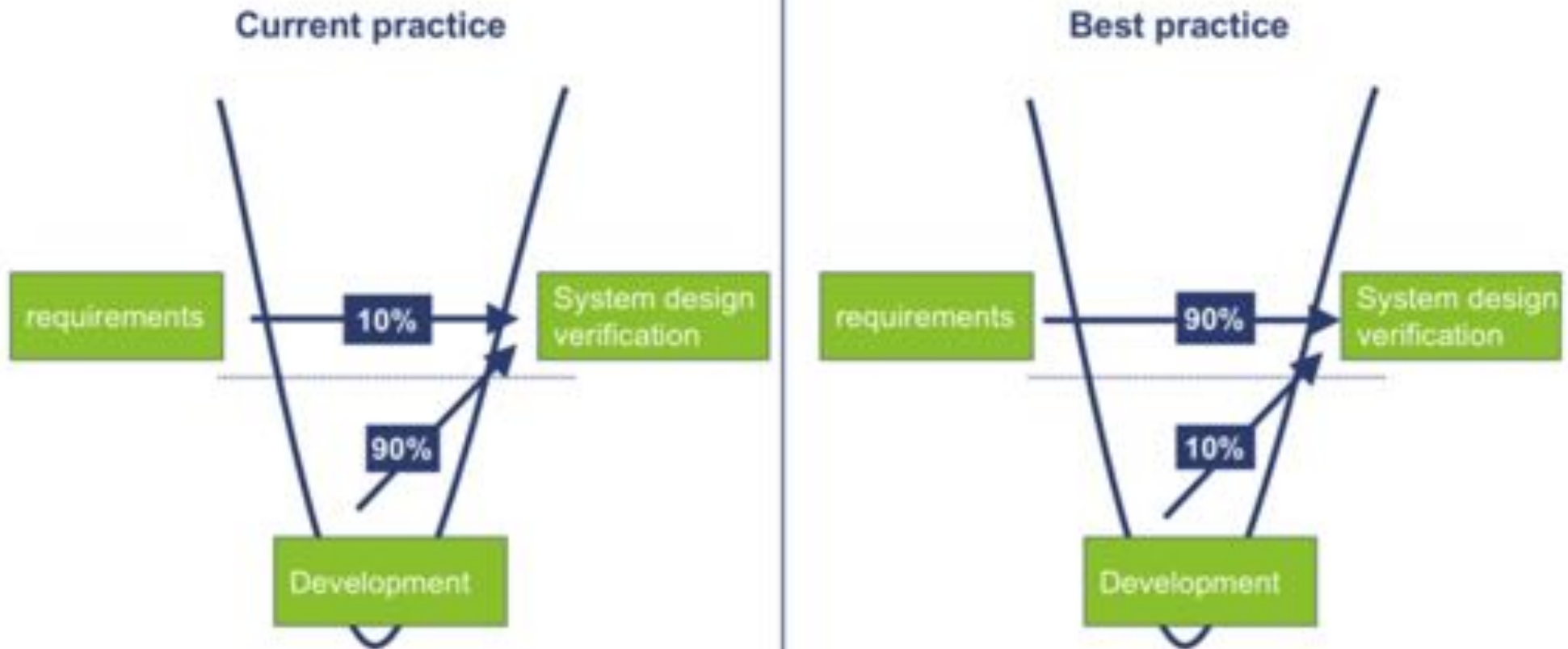




common code in two or more products

Effect van een snel groeiende organisatie

Testen wat gebouwd is of wat bedoeld is?



Dit zou ik nu anders brengen...

Beijing 2011

- Upgrade naar TPI Next
- Complexiteit
- MBT
- Outsourcing
- Requirements
- Agile



Terugkerende thema's

Shenzhen 2011

- Testmanagement training



Xi'an 2012

- Te goed!
- Kan het wat minder?



Key areas		Initial	Controlled				Efficient				Optimizing		
1	Stakeholder commitment		A	B	B	C	F	H	H	K	M	M	
2	Degree of involvement		A	B	C	E	H	H	J	L	L		
3	Test strategy		A	A	B	E	F	F	H	K	L		
4	Test organization		A	D	D	E	I	I	J	J	K	L	L
5	Communication		B	C	C	D	F	F	J	M	M		
6	Reporting		A	C	C		F	G	G	K	K		
7	Test process management		A	A	B	B	G	H	J	K	M		
8	Estimating and planning		B	B	C	C	G	H	I	I	K	L	L
9	Metrics		C	C	D		G	H	H	I	K	K	
10	Defect management		A	A	B	D	F	F	H	J	K	L	L
11	Testware management		B	B	D	E	I	I	J	L	L	L	
12	Methodology practice		C	D	E		F	H	J	J	M	M	
13	Tester professionalism		D	D	E	E	G	G	I	I	K	K	M
Context: testen was even 'klaar' met verbeteren													
15	Test tools		L	L	L		F	G	G	I	L	M	M
16	Test environment		C	D	D	E	G	H	J	J	L	M	M

Xi'an 2012

- Te goed!
- Kan het wat minder?
- Hardware teststrategie
- Verder veel 'gewone zaken'
- Testautomatisering
- Dekking unit tests
- Outsourcing → twee blokken verderop



Context: testen was even 'klaar' met verbeteren

Chengdu 2014, *out of comfortzone*

Eerste keer:

- TPI alleen
- Agile TPI (TI4Agile)



		Forming				Norming				Performing			
1	Stakeholder commitment	1	2	3	4	1	2	3	4	1	2	3	
2	Planning & Estimation	1	2	3	4	1	2	3	4	1	2	3	4
3	People	1	2	3	4	1	2	3	4	1	2	3	
4	Interaction	1	2	3	4	1	2	3	4	1	2	3	4
5	Teamwork	1	2	3		1	2	3	4	1	2	3	4
6	Test process	1	2	3		1	2	3	4	1	2	3	
7	Test management	1	2	3	4	1	2	3		1	2	3	
8	Test profession	1	2	3	4	1	2	3	4	1	2	3	
9	Test automation	1	2	3		1	2	3	4	1	2	3	4
10	Regression & E2E testing	1	2	3	4	1	2	3		1	2	3	4
11	Defect management	1	2	3		1	2	3	4	1	2	3	
12	Test environment	1	2	3		1	2	3		1	2	3	4

TI4Agile kijkt naar Testen en Agile (samen)werken

Chengdu 2014, *out of comfortzone*

Eerste keer:

- TPI alleen
- Agile TPI (TI4Agile)
- Michael Bolton



Chengdu 2014



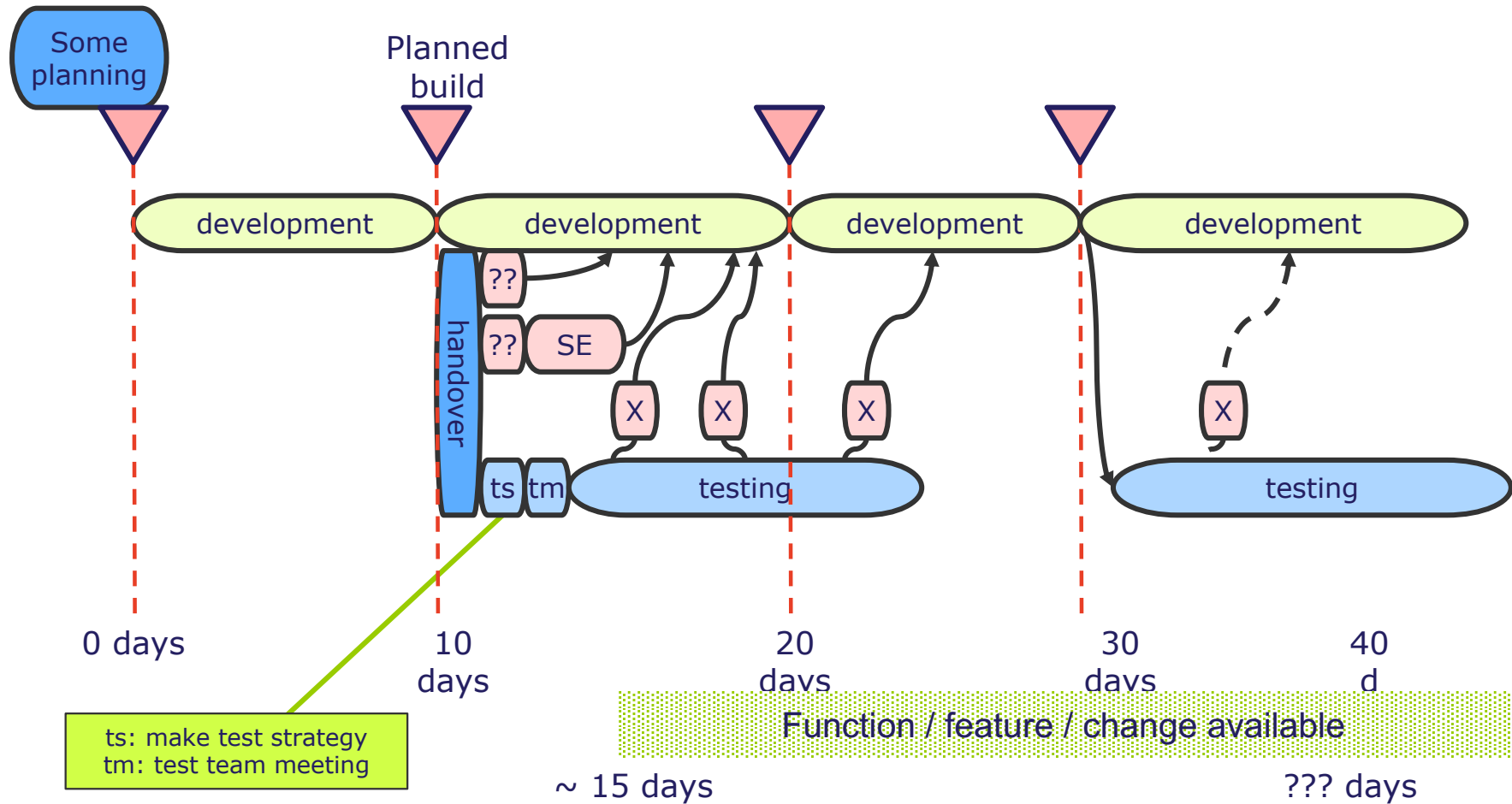
Stap uit je comfortzone en vertrouw op jezelf!

Wuhan 2015

- Lift
- Starbucks
- Mijlpalen en sprints
- Risicoanalyse
- Three part testing story



Current sequential work flow



Testing: Test's law on software failures

$$R = L \times I$$

Risk *Likelihood* *Impact*



NO RISK NO TEST



Tell a Three-Part Testing Story

A story about the status of the **PRODUCT**...

...about what it does, how it failed, and how it *might* fail...
...in ways that matter to your various clients.

A story about **HOW YOU TESTED** it...

...how you operated and observed it...
...how you recognized problems...
...what you have and *have not* tested yet...
...what you won't test *at all* (unless the client objects)...

A story about how **GOOD** that testing was...

...the risks and costs of testing or not testing...

Leren van mijn rolmodel Michael Bolton

...how testable (or not) the product is...

...what you need and what you recommend.



Vragensessies

CLASS Topic

- ✓ How to do your Tax Budget
- ✓ How to evaluate a tax process?
- ✓ How to evaluate market result?
• when we are not taking?
- ✓ How the other Company do tax?
- ✓ How to deal with liability law in the EU?
- ✓ How to deal with non-financial reporting?
- ✓ How to better work with digital tools?
- ✓ How to manage the digital tax world?
- ✓ How evaluate the best country?
- ✓ How to tell a better story?



Leren van de vragenstellers

Shenzhen 2016

This is the plan, make it happen!

This is the plan, make it happen!

This is the plan, make it happen!

This is the plan, make it happen!

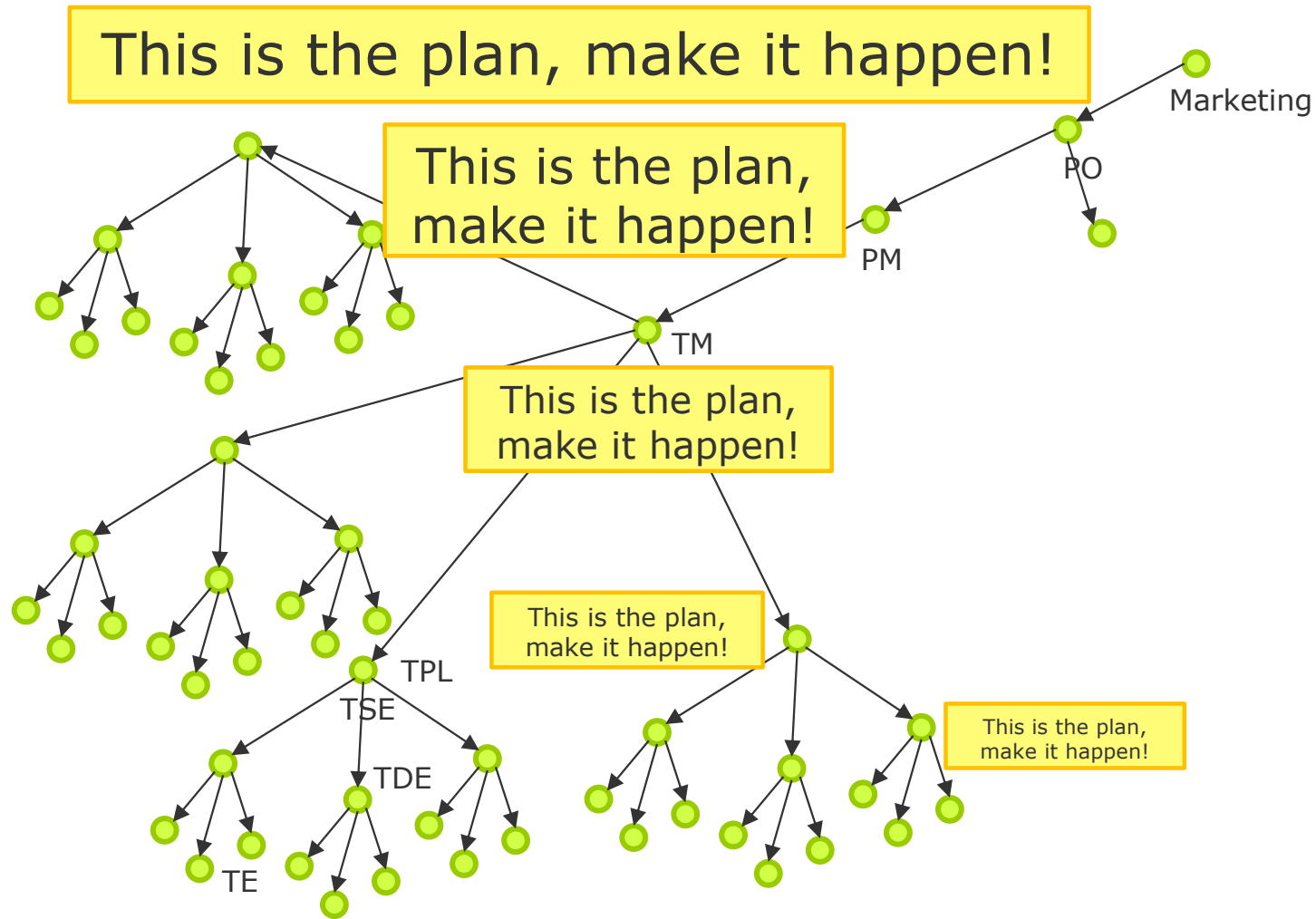
This is the plan, make it happen!

This is the plan, make it happen!

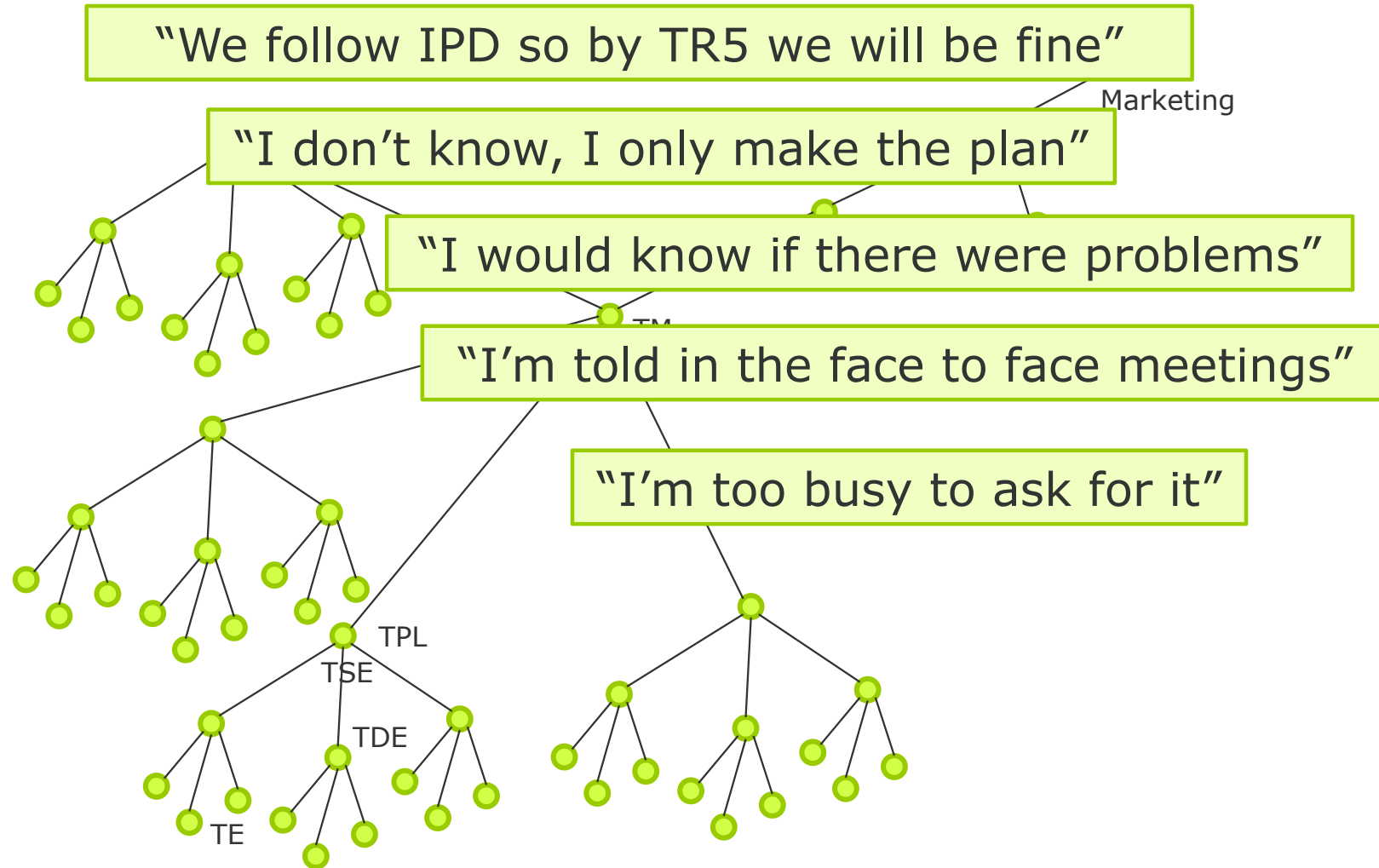


Leren van observeren

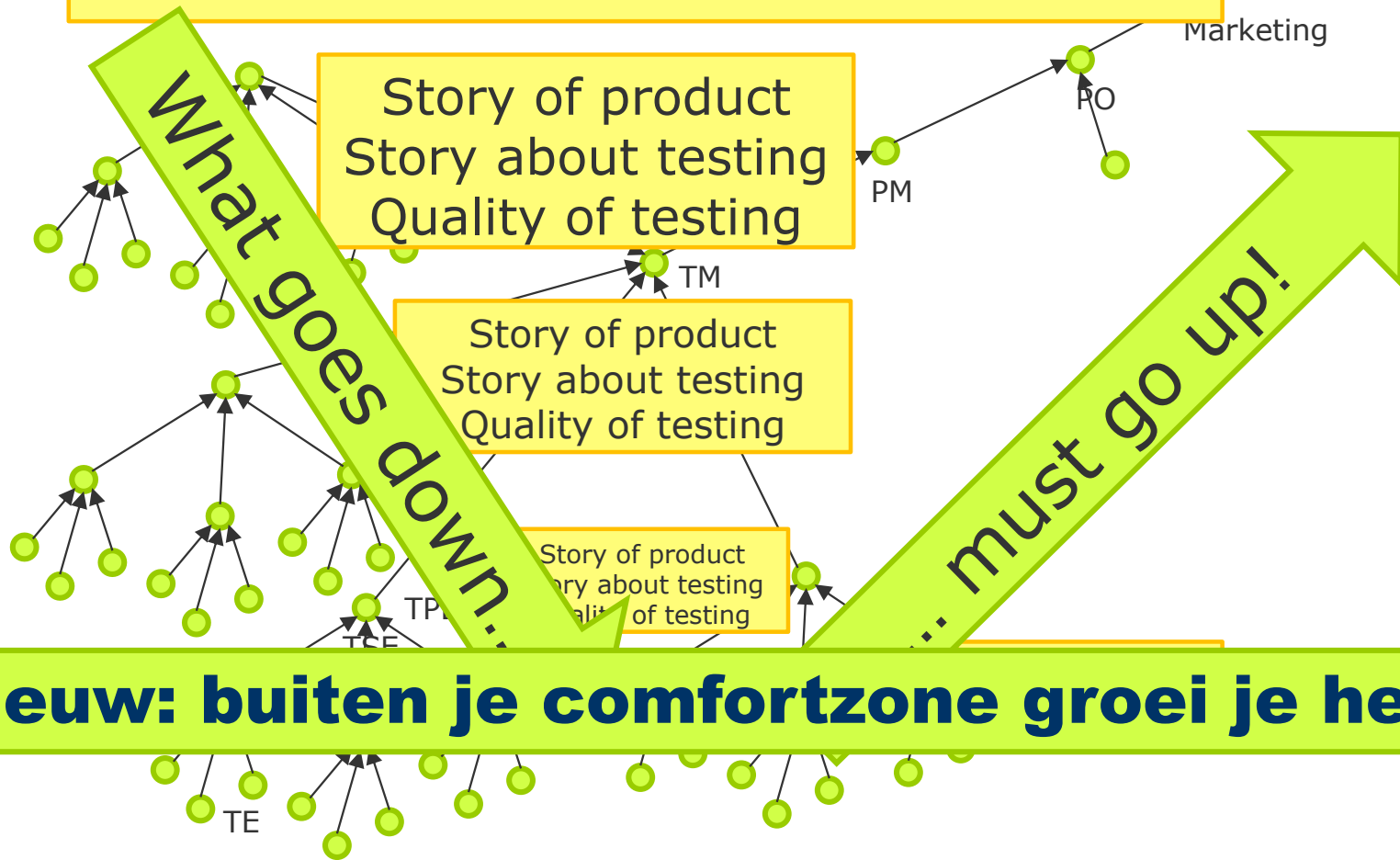
This is the plan, make it happen!



Status?



Story about the product
Story about the testing
Story about the quality of the testing



Hangzhou 2017

- Storyboard
- Terugkerend thema





Hangzhou 2017

- Storyboard
- Terugkerend thema
- Waar is TPI gebleven?



Key areas – development	Forming				Norming				Performing			
13. Code quality	1	2	3	4	1	2	3		1	2	3	4
14. Software architecture	1	2	3		1	2	3	4	1	2	3	
15. CI/CD	1	2	3	4	1	2	3	4	1	2	3	3
16. Development testing	1	2	3	4	1	2	3	4	1	2	3	4
17. Development profession	1	2	3	4	1	2	3	4	1	2	3	
18. Metrics	1	2	3	4	1	2	3		1	2	3	

100% statement coverage is not enough

```
unit testLeftPath (Input)
  assertion
```

```
unit testRightPath (Input)
  assertion
```

```
unit testElseOf2 (Input)
  assertion
```

```
unit testElseOf3 (Input)
  assertion
```

```
unit testLeftPathB (Input)
  assertion
```

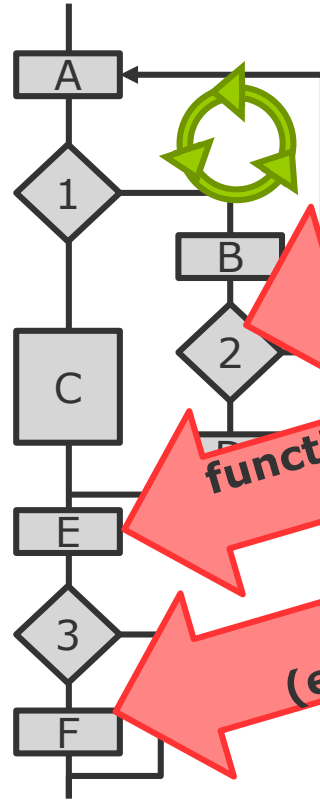
```
unit testLeftPathC (Input)
  assertion
```

```
unit testRightPathB (Input)
  assertion
```

```
unit testRightPathC (Input)
  assertion
```

- what can go wrong
- learn the code

(yet) unexpected input from 'user' of the function



testing boundaries
testing multiple conditions

function() with unexpected results

error handling
(e.g. divide by zero)

Hangzhou 2017

- Storyboard
 - Terugkerend thema
 - Waar is TPI gebleven?
-
- 4 juli zomerworkshop



Context Driven Test Improvement!