

Testfall statt Zufall

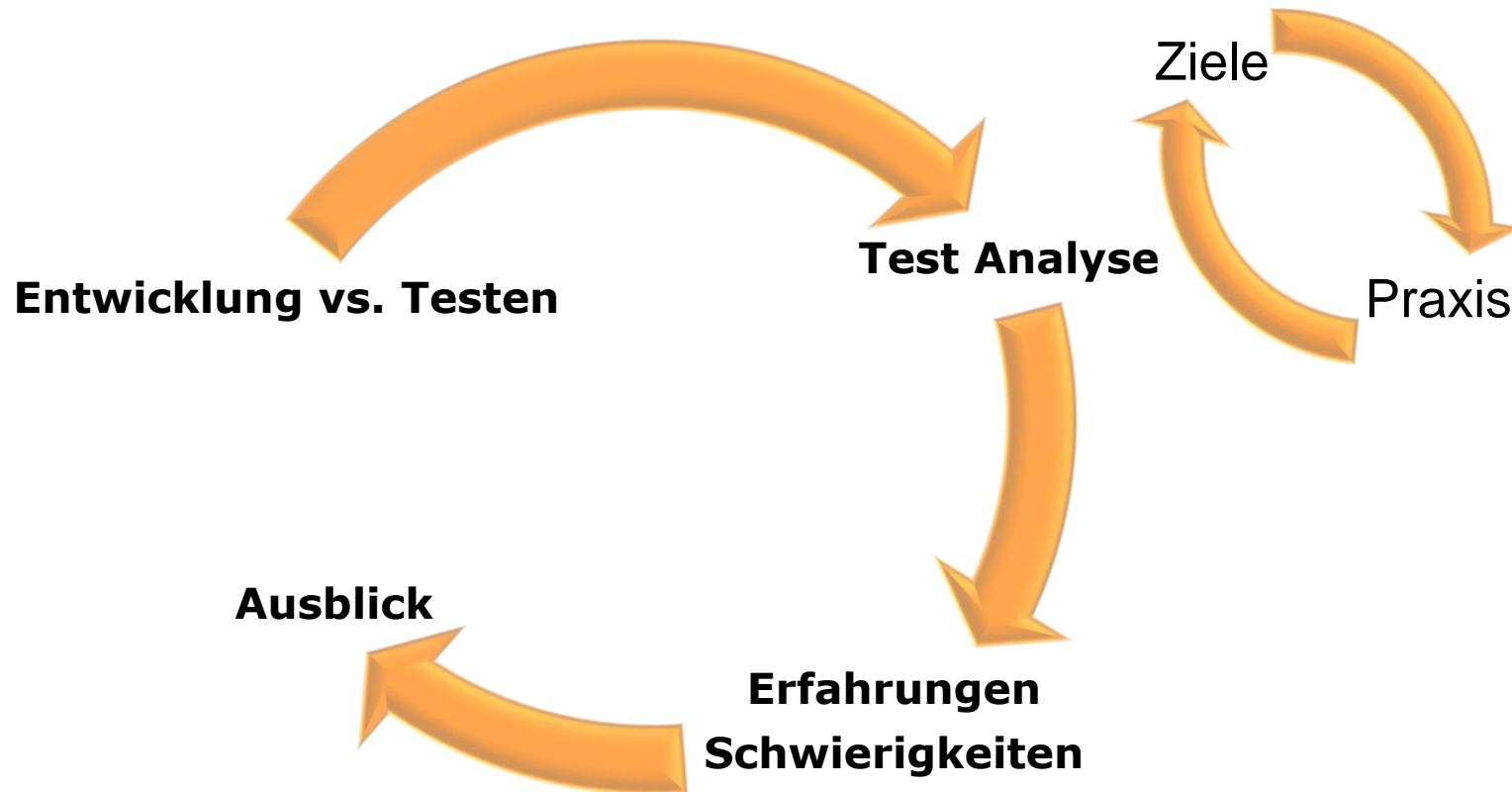
Technologie Forum Zug – 28. Oktober 2010

Fabian Wigger – Roche Diagnostics Ltd.



Testfall statt Zufall

Agenda



Roche Diagnostics Ltd

Testing in der Roche

- 1520 Mitarbeiter in Rotkreuz
 - Diagnose-Geräte und Workflowsysteme für Labor, Kliniken,...
 - Forschung, Entwicklung, Produktion, Support
- ~100 Mitarbeiter im Testing Bereich
 - Geräte Tests (HW & SW), Spital & Labor IT Lösungen

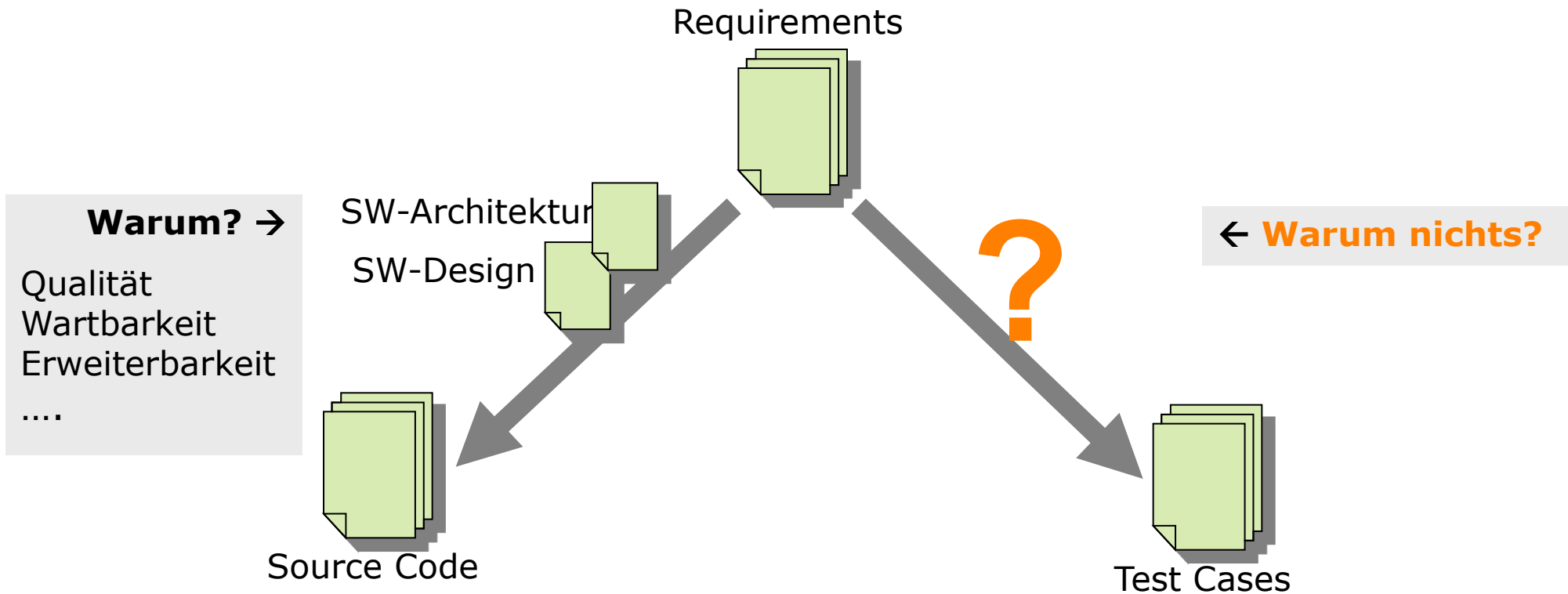
Meine Tätigkeiten:

- Test Analyst für verschiedene Produkte
- Testfall Methodik Arbeitsgruppe
- Test Tools
- Test Reporting



Software Testen vs. Software Entwicklung

Engineering Ansatz

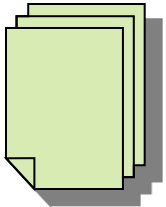


Testen ist eine Engineering Aufgabe!

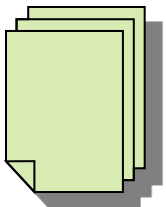
Software Testen – Der strukturierte Ansatz

Wie sieht unser Engineering Ansatz aus?

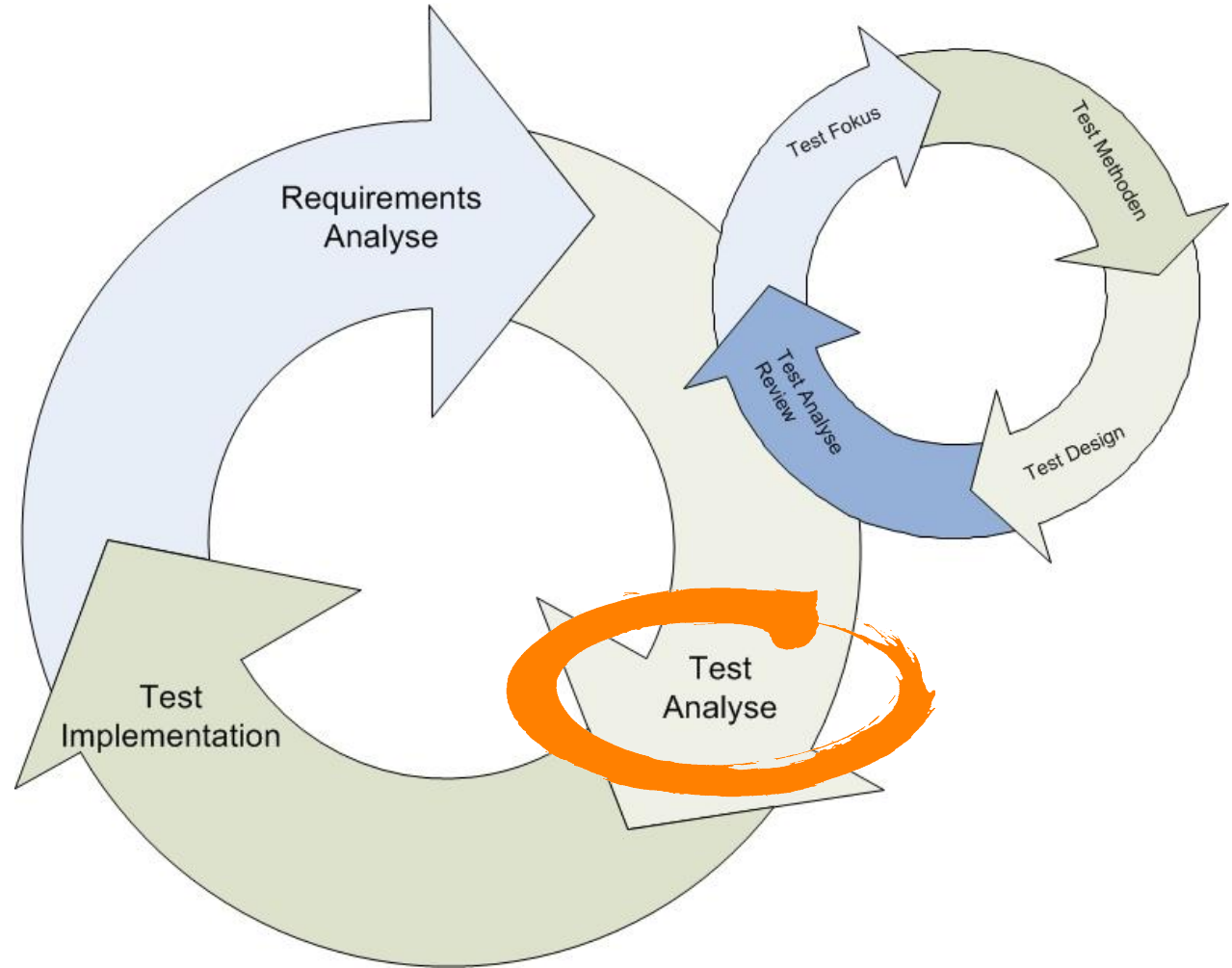
Requirements



Test Subject Data		Test Case Metadata	
Classification	Test Case	Tester	Test Case
Test Case ID	None	None	None
Author			
Version	1.0	Version	1.0
Test Object			
Object		Object	
Test Features			
Feature	Test Case	Feature	Test Case
Test Case		Test Case	
Test Case		Test Case	
Test Case		Test Case	
Test Case		Test Case	
Test Methods			
Method	Test Case	Method	Test Case
Test Case		Test Case	
Test Design			
Design	Test Case	Design	Test Case
Test Case		Test Case	
Review Comments			
Comment	Test Case	Comment	Test Case
Test Case		Test Case	

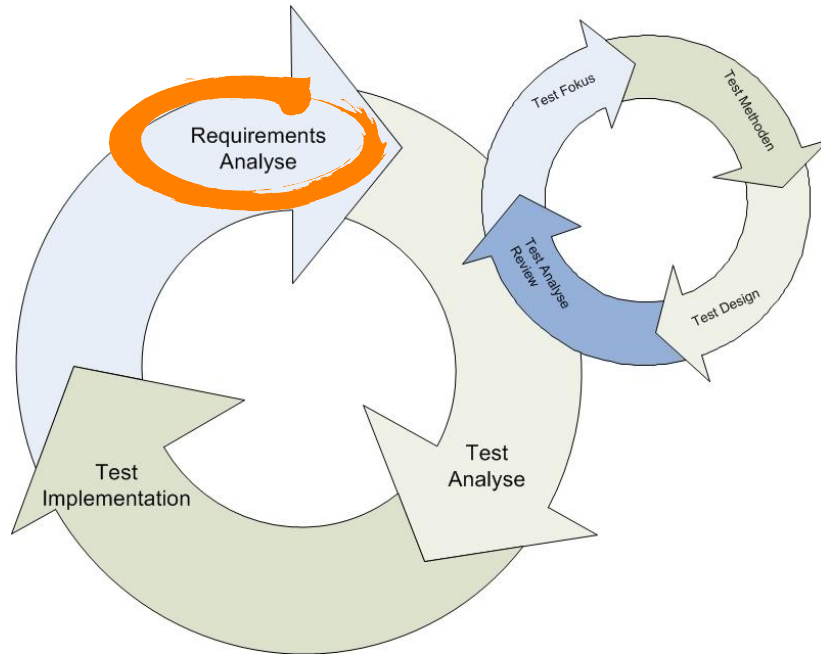


Test Cases



Software Testen – Der strukturierte Ansatz

Requirements Analyse – Ziele



→ *Habe ich die nötigen Informationen?*

Ziele:

- Testbarkeit überprüfen
- Vollständige Requirements
- Arbeitspakete identifizieren

Software-Entwicklung:

- *Sind die Requirements vollständig?*
- *Wer programmiert welche Teile?*

Software Testen – Der strukturierte Ansatz

Requirements Analyse - Praxis

Requirements Review Meetings

Review Prozess definieren

Checkliste für Tester

- Allgemein

- Workflow klar
- Requirements konsistent

- Testbarkeit

- Exakte Definitionen (kein „sollte“, „ungefähr“, ...)
- Daten Bereiche

- Nicht-Funktionale Aspekte

- Performance
- Sicherheit
- ...

Checklist Requirement Review		Test Case Methodology
Requirement Package Name	Template Checklist Requirement Review.dot	Version xx
Requirement Engineer (Creator)		
Test Analyst (Reviewer)		
Comments		

Checklist: Requirement Review

General

Review Criteria	Comment
<input checked="" type="checkbox"/> Are the Requirements written understandable? → Try to rephrase the statements in your own words Are the requirements complete? → Input / Output (valid ranges) → data character (mandatory, optional, read-only,...) → data handling (retrieve, add, modify, delete) → workflows (standard case, alternative cases) Are the Requirements consistent with each other?	
Are the priority and the risk of the requirement appropriate?	
Are the definition and terms clear?	

Testability

Review Criteria	Comment
<input checked="" type="checkbox"/> Are ambiguous terms used? → not "some", "reasonable", "higher", "faster", "easy" / ... Are the precision and range defined for numbers?	
Are inverse requirements explicitly documented? → Exception Handling, invalid ranges...	
Are the supported configurations specified?	
Are the compatibility requirements specified? → backward, database version, other installations	

Non-Functional

Review Criteria	Comment
<input checked="" type="checkbox"/> Performance: Are Requirements needed for Response Time, Throughput, Volume, Load security: Are Requirements needed for: → User rights and restrictions → data security (data export, interfaces,...) Usability: Are Requirements needed for Look and Feel? Language: Are Requirements needed for Internationalization/Localization	

Report Specific

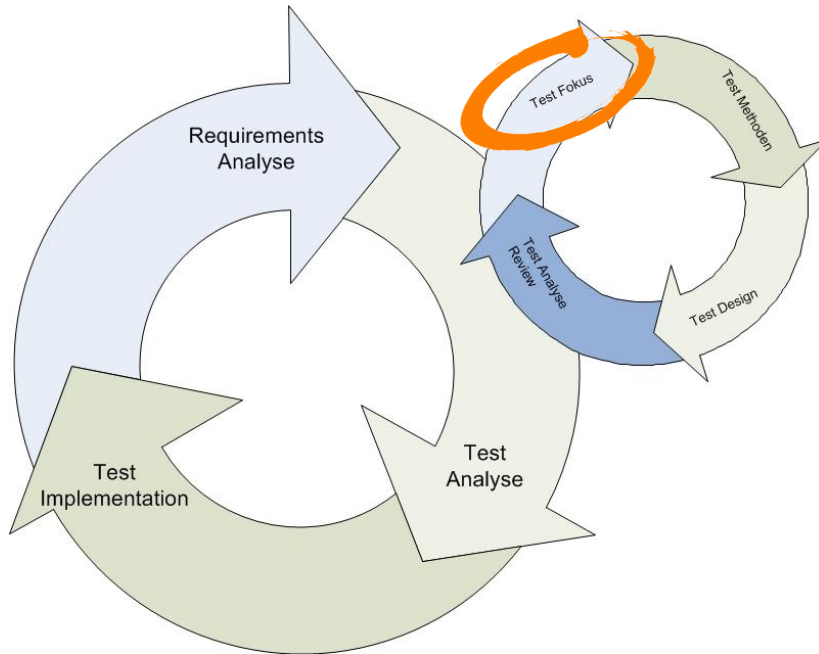
Review Criteria	Comment
<input checked="" type="checkbox"/> Is the general formatting defined? → margins, header/footer, format (landscape/portrait) Is a report example available? Is the data format defined? → set of data, format, page breaks, table headers,... Is the set of data's defined? → filter criteria	

\\pilot.roche.com\departments\DP\NVA\Shared Documents\VM Documents\Test Case Methodology\Requirement Based Testing\Template Checklist Requirement Review.doc

1 / 1

Software Testen – Der strukturierte Ansatz

Test Fokus - Ziele



→ *Was teste ich überhaupt? Was nicht?*

Ziele:

- Test Schwerpunkt klar aufzeigen
- Dokumentieren was nicht getestet wird
- Priorisierung

Software-Entwicklung:

- *Welche Funktionen sind in der Version 1 implementiert?*
- *Unterstützt wir Mehrsprachigkeit, Verschlüsselung,...?*

Software Testen – Der strukturierte Ansatz

Test Fokus - Praxis

Test Fokus Liste:

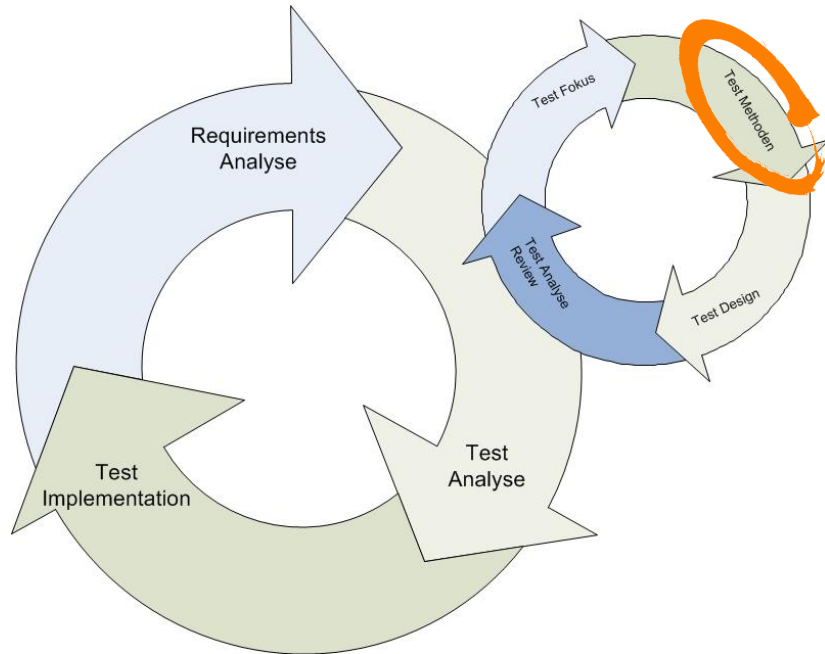
Covered (✓)	Test Focus	Comment / Rationale
-	Workflows	
-	Inputs	
✓	Outputs	- QC Ranges (min and max) send to instrument
-	Security	
-	Portability	
-	Internationalization	
-	Performance	
-	Usability	

Beispiel Portabilität:

- Funktionen müssen auf verschiedenen Browsern getestet
- Funktionalität "sollte" Browser unabhängig sein

Software Testen – Der strukturierte Ansatz

Test Methoden - Ziele



→ *Wie komme ich nun zu Testszenarien und Testdaten? Welche Methoden eignen sich dazu?*

Ziele:

- Effiziente und effektive Testfälle
- Optimale Testabdeckung

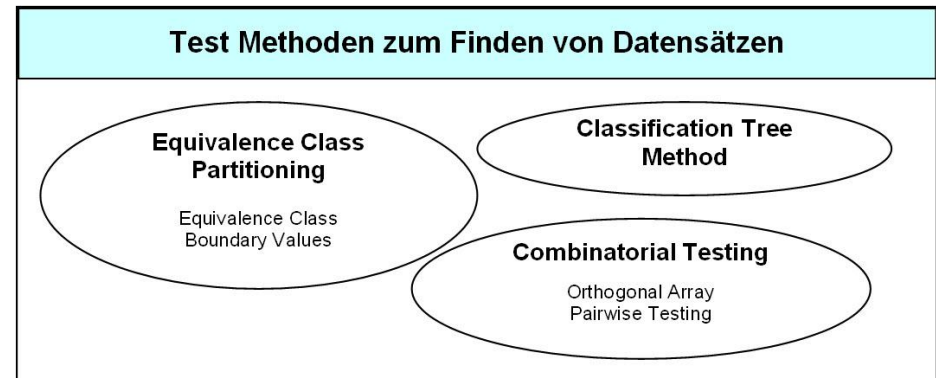
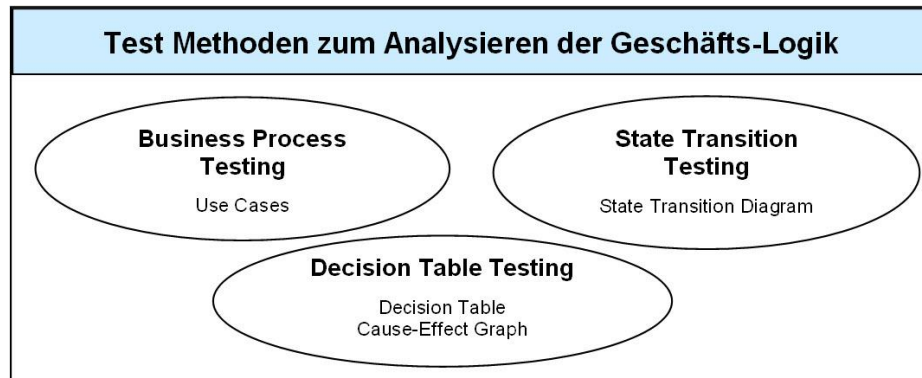
Software-Entwicklung:

- *Welche Libraries, Frameworks, API's gibt es bereits?*
- *Welche Plattformen soll ich verwenden?*

Software Testen – Der strukturierte Ansatz

Test Methoden - Praxis

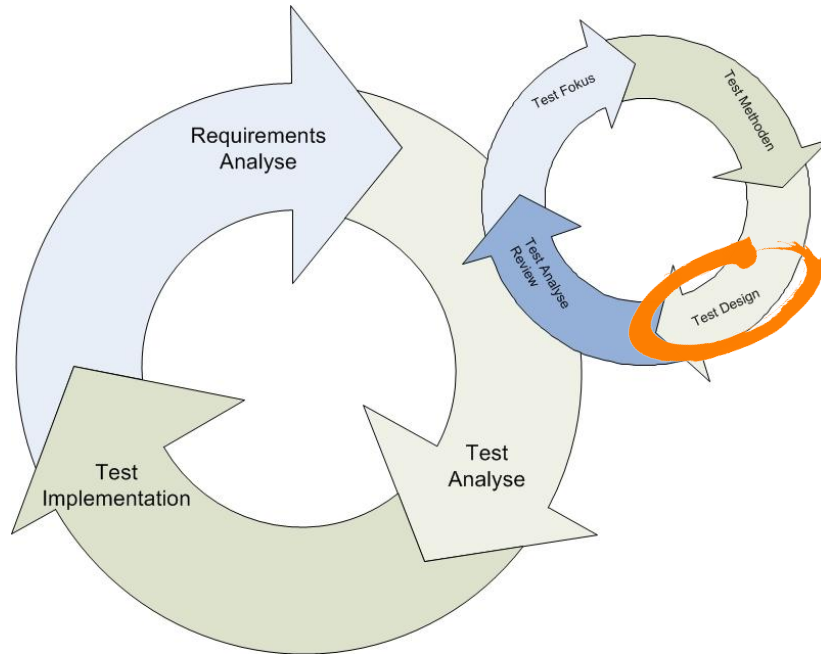
Vom Workflow zu den Test Daten



Anwendungsbereich - „Step by step“ Anleitung - Praxis-Beispiele

Software Testen – Der strukturierte Ansatz

Test Design - Ziele



→ *Wie viele Testfälle brauche ich nun?*

Ziele:

- Wiederverwendbarkeit
- Logisch strukturierte Testfälle
- Klares Testziel pro Testfall

Software-Entwicklung:

- *Wie viele SW-Klassen brauche ich nun?*
- *Wie gruppiere ich meine Klassen zu Modulen?*

Software Testen – Der strukturierte Ansatz

Test Design - Praxis

Auflistung der Testfälle

Test Case	Test Type	Comment / Rationale
Check QC Ranges Calculation and Rounding Rules	Systemtest	1 test case required with 3 different test data. Only when test pass for all 3 test data, the calculation is correct (otherwise it's difficult to say if rounding or "4999" rule is wrong)
LED with real instrument	Exploratory Test	GLU test with "Hemo-Q", because this instrument shows result with 1 decimal digit
QC Range Calculation	Regression Test	Update existing Regressiontest for QC range with LED

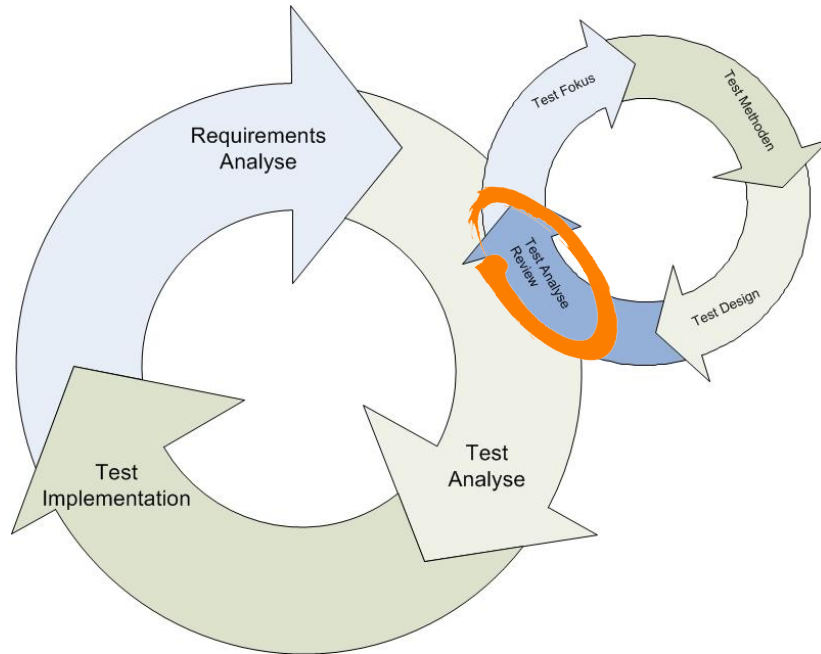
→ Testfall Namen und Testziel

→ Strukturiert oder explorativ getestet

→ Regressionstest

Software Testen – Der strukturierte Ansatz

Test Analyse Review - Ziele



→ *Habe ich an alles gedacht? Oder sogar an zu viel?*

Ziele:

- Ungenügende Testabdeckung aufdecken
- Reduzieren des Testaufwands

Software-Entwicklung:

- *Review des Software Designs*
- *Software Architektur laufend wird überprüft*

Software Testen – Der strukturierte Ansatz

Test Analyse Review - Praxis

Review Kommentare ins Word-Dokument

Test Case	Test Type	Comment / Rationale
Check QC Ranges Calculation and Rounding Rules	Systemtest	1 test case required with 3 different test data. Only when test pass for all 3 test data, the calculation is correct (otherwise it's difficult to say if rounding or "4999" rule is <u>wrong</u>)
LED with real instrument	Exploratory Test	GLU test with "Hemo-Q", because this instrument shows result with 1 decimal digit
QC Range Calculation	Regression Test	Update existing Regressiontest for QC range with LED

→ Review durch:

- Test Analyst
- Software Entwickler
- Requirements Engineer

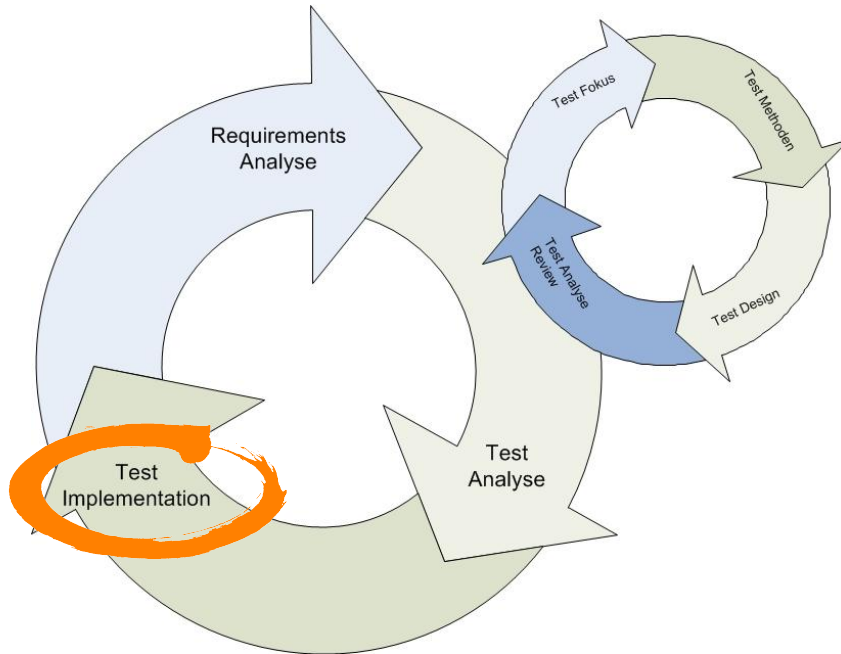
→ Review bevor Software vorhanden ist

Comment [c1]: TC for the calculation for absolute numbers as documented on the ICATT Description is missing? One Example for it = QC Test SO2 mapped on [Main] Test Mapping (RiliBAEK) as manufacture Range with configured Range type= ABS and added Ranges.

Comment [c2]: How you will test GLU Tests with Hemo Q-Instrument we don't have this instrument in our cit1k inventory? Maybe you mean Haemochron?

Software Testen – Der strukturierte Ansatz

Test Implementierung - Ziele



→ *Wie schreibe ich nun meine Testschritte?*

Ziele:

- Lesbarkeit und Wartbarkeit erleichtert
- Ähnlicher Detaillierungsgrad
- Jeder Test Analyst kann jeden Testfall ausführen

Software-Entwicklung:

- *Kodierrichtlinien*
- *Software Design Pattern*

Software Testen – Der strukturierte Ansatz

Test Implementierung - Praxis

Test Patterns (~20 Patterns Mit Antipattern)

Test Case Linking:		
How should...	Pattern	Anti-Pattern
... requirements and issues be linked to test cases?	Link only requirements and issues relevant to the test goal	Link requirements and issues touched by a test case, but not in focus of the test goal

Pattern zu Testfall, Test Schritte, Test Run

Testfall Nomenklatur (~16 Schlüsselwörter)

	Convention	Example
U . .	[MENU] aName > aEntry	[MENU] Workplaces > Sample Tracking
	[TOOLBAR] aIcon	[TOOLBAR] Save
	[TREE] aName > aEntry	[TREE] Administration > Clients > KIS > Default System
	[ALERT] aName	[ALERT] Quality Control
	[MAIN] aName	[MAIN] Order Input, XML-ID: 9988

GUI Objekte, Aktionen, Parameter

Software Testen – Der strukturierte Ansatz

Erfahrungen & Schwierigkeiten

Welche Erfahrungen haben wir gemacht?

- Transparenz geschaffen durch die Testanalyse
- Generelles Feedback: „Review freundliche Analyse“
- Frühere Erkennung von Problemen durch die Test Analyse
- Aufwendiges Umschreiben der Testfälle reduziert

Welche Schwierigkeiten ergaben sich?

- Publizieren der Informationen; Schulung der Tester
- Anwendung in der Praxis (Detaillierungsgrad)
- Verständigung → Verschiedene Testgruppen



Software Testen – Der strukturierte Ansatz

Ausblick

Was erhoffen wir uns noch?

- Aussagekräftige Testabdeckung
- Bessere Abschätzung des Testaufwands
- Analyse der Fehler nach der Auslieferung
→ Warum haben wir den Fehler nicht gefunden
- Akzeptanz der Test-Abteilung steigern

Die Testfall Methodik ist ein kontinuierlicher Verbesserungs-Prozess. Die Workshops werden weitergeführt und anhand der Rückmeldungen und Erfahrungen laufend erweitert und angepasst....

...weil Testen eine Engineering Aufgabe ist!





We Innovate Healthcare

Fabian Wigger - fabian.wigger@roche.com
- Xing: Fabian Wigger
- <http://www.testinghasnovalue.blogspot.com> (coming soon)