

2.– 5. September 2013  
in Nürnberg



# Herbstcampus

Wissenstransfer  
par excellence

## Akzeptanztests und Regression

Komplexe Schnittstellen und Prozesse mit FitNesse testen

Frank Ganske

[frank.ganske@mathema.de](mailto:frank.ganske@mathema.de)

# Agenda

---

- **Begriffe und fachlicher Hintergrund**
- Beispiele, Vorstellung des Werkzeugs
- Das Konzept
- Die Umsetzung des Konzepts
- Resümee

# Begriffe

---

- Entwicklungsabteilung testet automatisiert
  - Seilschaft statt Freiklettern, schnelle Rückmeldungen
  - Manuelle Tests investigativ und für neue Funktionen
- Akzeptanztests vs. Komponententests
  - Gesamtsystem zentral vs. Einzelverantwortung Entwickler
  - „Tiefer hängende Früchte“ für Regressionsprüfung
- Komplexität, welcher Art?
  - CERN: viele Daten, viele Rechenknoten → so eher nicht
  - Ebay: viele Browser, viele Betriebssysteme → so auch nicht

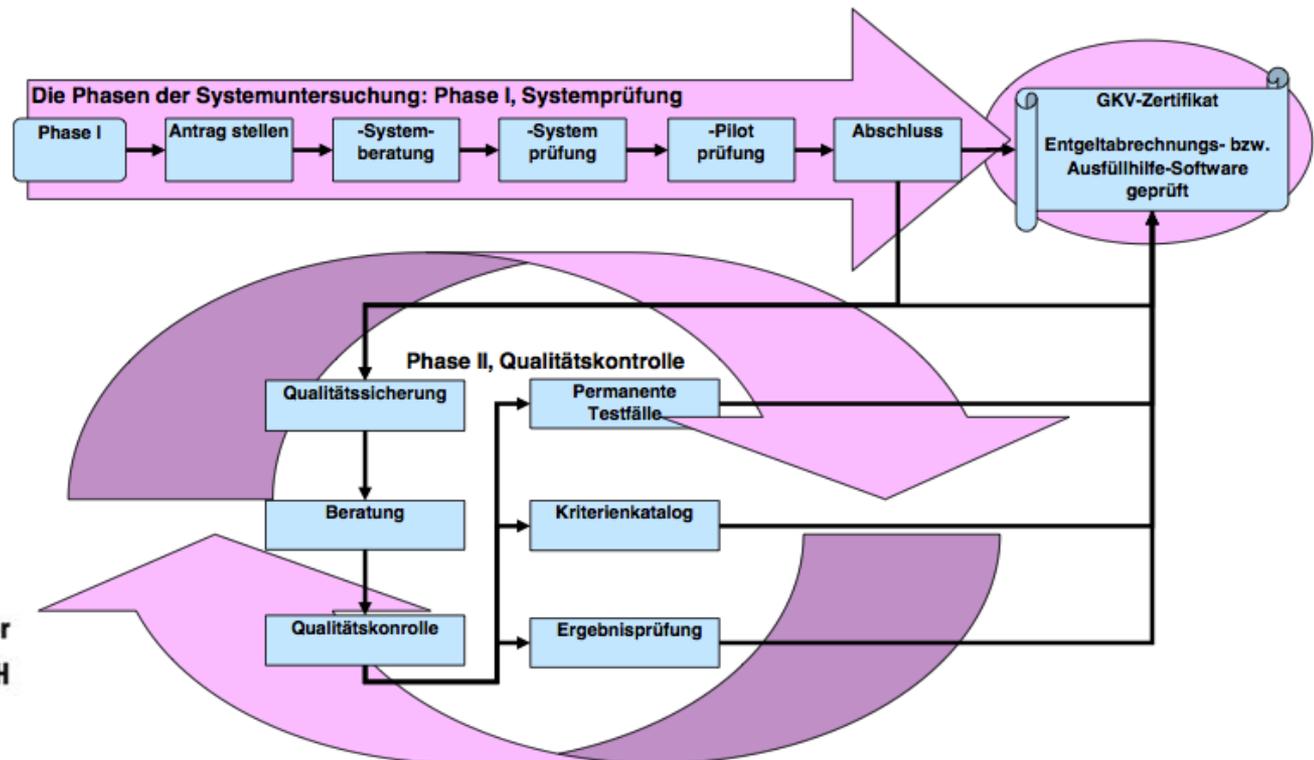
# Hintergrund - Standardsoftware

- Personalabrechnung in Deutschland, Sozialversicherung
- DEÜV ...



# ITSG

Informationstechnische Servicestelle der  
Gesetzlichen Krankenversicherung GmbH



[http://www.gkv-ag.de/upload/Das\\_Verfahren\\_Ueberblick\\_521.pdf](http://www.gkv-ag.de/upload/Das_Verfahren_Ueberblick_521.pdf)

# Agenda

---

- Begriffe und fachlicher Hintergrund
- **Beispiele, Vorstellung des Werkzeugs**
- Das Konzept
- Die Umsetzung des Konzepts
- Resümee



[WorkShop](#) > [TutorialSuite](#)

## Tutorial00Test

Test

Edit

Add

Tools

► [Included page: .WorkShop.SetUp \(edit\)](#)

[Expand All](#)

[Collapse All](#)

### Ein Fixture und ein Test in FitNesse - Hallo Welt!

Script-Tabellen verwenden die vorherige Fixture-Instanz bis eine neue gesetzt wird.

Setzen von **java.lang.Object** als Fixture-Klasse:

Script	java.lang.Object
\$id=	toString
show	get class

Prüfung, die neue Tabelle verwendet die selbe Instanz des Fixture:

Script		
check	toString	\$id

Prüfung, jetzt wird ein anderes Exemplar des Fixture angesprochen:

Script	java.lang.Object	
check not	toString	\$id

Wird das Fixture-Programm in [.WorkShop.SetUp](#) gesetzt, muss es in den Tests nicht mehr angegeben werden. -> [Tutorial01Test](#)

► [Fußnoten:](#)

[Expand All](#)

[Collapse All](#)

## FitNesse

- Ein Wiki für Tests und Test-Suiten
- Steuert beliebige Fixture-Programme
- Ausgereift und stabil
  - Source Code Management
  - Continuous Integration

## Real (gekürzt)

**VwePayroll. AkzeptanztestsKomplex**  
 TEST RESULTS [\[history\]](#) Tests Executed OK

**Assertions:** 18 right, 0 wrong, 0 ignored, 0 exceptions (32,467 seconds) [Expand All](#) / [Collapse All](#)

▸ *Precompiled Libraries* [Expand All](#) / [Collapse All](#)

▸ *Included page: .VwePayroll.SetUp (edit)* [Expand All](#) / [Collapse All](#)

Scenario	Firmen/Betriebsstätten - Firmen - Einstellungen, Eingaben		
field:	sofortmeldepflichtiger Betrieb	value:	false

Script		
Firma format	MDE000	▸ Scenario
Demo Firma kopieren	\$FIRMA->[MDE996]	▸ Scenario

Scenario	Mitarbeiter, Eingaben		
show	field:	Eintrittsdatum arbeitsrechtlich	value: 2012-01-02
show	field:	Eintrittsdatum SV-rechtlich	value: 2012-01-02

Script		
Demo Mitarbeiter abrechnen	1	▸ Scenario

Script		
Firmen/Betriebsstätten - Firmen - Grundlagen - Mitarbeiter-Prüflauf	\$FIRMA->[MDE996]	▸ Scenario
SV-Meldewesen - Aufbau DEÜV-Meldungen	\$FIRMA->[MDE996]	▸ Scenario

[Front Page](#) | [User Guide](#)  
[root](#) (for global !path's, etc.)

- Wirkliche Welt
- Test-Fixture:  
Neue Firma,  
Mitarbeiter,  
Monatsabschluss
- 3 Testeingaben
- 505 Eingaben in  
17 Dialogen

# Werkzeugwahl - Entscheidung

---

- FitNesse enthält **alternative** Möglichkeiten:
  - FIT von Ward Cunningham (<http://c2.com> ...)
  - FitLibrary von Rick Mugridge (... die Autoren des FIT-Buchs)
  - SLIM von Uncle Bob, Robert Cecil Martin (FitNesse, Clean Code)
- SLIM ist nativ und bietet das Scenario-/Library-Konzept
  - Makrosprache, keine Kontrollstrukturen sondern (Über-)Steuerung durch Ebenen: Library, Suite, Test, Tabelle
  - Auf- und Zuklappen der Komplexität
  - Natürlichsprachigkeit ist möglich

# Agenda

---

- Begriffe und fachlicher Hintergrund
- Beispiele, Vorstellung des Werkzeugs
- **Das Konzept**
- Die Umsetzung des Konzepts
- Resümee

## Die drei Iterationen des Konzepts

---

- Erster Ansatz: ein Fixture(-programm) je Serverfunktion
- Besser: Serialisierung der Kommunikation als XML
- Die Lösung: dynamische Daten und Strukturen

# Die drei Iterationen des Konzepts

---

- Erster Ansatz: ein Fixture(-programm) je Serverfunktion
  - Ergebnis: ein Kommunikationsframework
- Besser: Serialisierung der Kommunikation als XML
- Die Lösung: dynamische Daten und Strukturen

# Die drei Iterationen des Konzepts

---

- Erster Ansatz: ein Fixture(-programm) je Serverfunktion
  - Ergebnis: ein Kommunikationsframework
- Besser: Serialisierung der Kommunikation als XML
  - Ergebnisse: ein Programm für alle Fixtures, das Metamodell
- Die Lösung: dynamische Daten und Strukturen
  - Ausschließlich testspezifische Angaben im Test deklarieren
  - Separate Bereitstellung der allgemeinen Daten
  - Verwendung von Demo-Daten des Release-Standes
  - *Solche Tests liefen ohne Änderung über mehrere Jahre.*

# Die drei Iterationen des Konzepts

---

- Erster Ansatz: ein Fixture(-programm) je Serverfunktion
  - Ergebnis: ein Kommunikationsframework
- Besser: Serialisierung der Kommunikation als XML
  - Ergebnisse: ein Programm für alle Fixtures, das Metamodell
- Die Lösung: dynamische Daten und Strukturen
  - Ausschließlich **testspezifische** Angaben im Test deklarieren
  - **separate** Bereitstellung der **allgemeinen** Daten
  - Verwendung von Demo-Daten des Release-Standes
  - *Solche Tests liefen ohne Änderung über mehrere Jahre.*

# Agenda

---

- Begriffe und fachlicher Hintergrund
- Beispiele, Vorstellung des Werkzeugs
- Das Konzept
- **Die Umsetzung des Konzepts**
- Resümee

## Sprache - technisch

---

- Kommunikation zwischen den Systemen, die Schnittstelle
- Wo steckt die getestete Logik?
  - Client beteiligt, Autsch → Oberflächentests ...
- Konkret realisierte Schnittstellen: CORBA, EJB, ...
  - Remoteschnittstelle, serialisierbar, veröffentlicht und als Bibliotheken (maschinenlesbar) verfügbar
  - Kaum Veränderungen und wenn, dann Erweiterung

# Sprache - minimalistisch

---

- **Methoden des Fixture-Programms:**
  - Mit **lookup:method:** werden die erforderlichen Datenstrukturen für den Schnittstellenaufruf instanziiert
  - Mit **field:value:** werden Knoten des Modells über XPath-Expressions angesprochen und Werte gesetzt
  - Mit **call:** wird die Funktion im Server ausgeführt und das Ergebnis ins Modell übernommen
  - Mit **field:** werden die Werte des Ergebnis abgefragt
  - Mit **field:caption:** werden natürlichsprachige Bezeichner gesetzt
  - Mit **nodes:value:** kann man mehrere Felder oder keins (nur falls vorhanden) setzen
  - Mit **nodes:** werden mehrere Elemente abgefragt
  - Mit **function:** werden Ergebnisse von XPath-Expressions als Text ausgegeben, ohne Knoten zu ermitteln
  - Mit **method:** (ohne lookup: ) werden Daten und Bezeichner übernommen, wenn Parametertypen eindeutig sind
  - Mit **nodes:append:**, bzw. **field:append:** werden Modellelemente kopiert
  - Mit **ignore:** setzt man Fehlermeldungen, die beim nächsten call erwartet werden

# Sprache - minimalistisch

---

- Methoden des Fixture-Programms:

- Mit **lookup:method:** werden die erforderlichen Datenstrukturen für den Schnittstellenaufruf instanziiert
- Mit **field:value:** werden Knoten des Modells über XPath-Expressions angesprochen und Werte gesetzt
- Mit **call:** wird die Funktion im Server ausgeführt und das Ergebnis ins Modell übernommen
- Mit **field:** werden die Werte des Ergebnisses abgefragt
- Mit **field:only:** werden nur die angegebenen Bezeichner gesetzt
- Mit **nodes:value:** kann man mehrere Felder oder keins (nur falls vorhanden) setzen
- Mit **nodes:** werden mehrere Elemente abgefragt
- Mit **function:** werden Ergebnisse von XPath-Expressions abgefragt, ohne Knoten zu ermitteln
- Mit **method:** (ohne lookup: ) werden Daten und Bezeichner übernommen, wenn Parametertypen eindeutig sind
- Mit **nodes:append:**, bzw. **field:append:** werden Modellelemente kopiert
- Mit **ignore:** setzt man Fehlermeldungen, die beim nächsten call erwartet werden

# Sprache - fachlich

- Testaufgaben zur Systemuntersuchung:

	<b>Pflichtenheft zum DV-gestützten Entgeltabrechnungs- und Meldeverfahren für die Sozialversicherung</b>	<b>Testaufgaben 2012/2013 Basismodul Personal-/Testfall-Nr.: 150 Breit-Gossmann</b>

**Personalstammdaten:**

Familienname/ Geburtsname	Breit-Gossmann	PLZ, Wohnort Land	92549 Stadlern	Geb.-Datum Geb.-Ort	31.12.1947
Vorname	Heiner	Straße/ Hausnummer	MMM-Str. 1	Versicherungsnummer	19 631247 B 493
Vorsatzwort/ Geb.-Vorsatzwort		Anschriftenzusatz	III. Etage	Staatsangehörigkeit Elterneigenschaft	deutsch nein
Namenszusatz/Titel Geb.-Namenszusatz		Geburtsland EU- Staatsangehöriger		Rentenbezug	nein

Monat	laufendes Arbeitsentgelt	Einmalig gez. Arbeitsentgelt	KK / Einzugstelle	Beitrags- gruppe	Personen- gruppe	Tätigkeits- schlüssel	Besonderheiten/Anmerkungen
10	Eintritt: 01.10.2012 als Estrichleger						
10	1.900,00		AOK NW	1111	101	331122211	BBNR GT: 151 413 64 / Gefahrarif: 17
11	1.900,00		AOK NW	1111	101	331122211	Es liegt eine Aufforderung zur Abgabe einer gesonderten Meldung „Rente wegen Alters“ nach § 194 SGB VI (Rentenbeginn 01.02.2013) mit Grund der Abgabe 57 vor.
12	1.915,28		AOK NW	1111	101	331122211	incl. Nachzuschläge (Nacht 1) für 4 Std. Tarifzuschlag 35%
01	1.900,00		AOK NW	1111	101	331122211	
02	Austritt zum 28.02.2013						
02	1.900,00		AOK NW	3321	119	331122211	Regelaltersrente ab 01.02.2013

[http://www.gkv-ag.de/upload/TA\\_2012\\_2013\\_Stand\\_01.07.2012\\_oe\\_2162.pdf](http://www.gkv-ag.de/upload/TA_2012_2013_Stand_01.07.2012_oe_2162.pdf)

# Sprache - fachlich

- Testaufgaben zur Systemuntersuchung:

	<b>Pflichtenheft zum DV-gestützten Entgeltabrechnungs- und Meldeverfahren für die Sozialversicherung</b>	<b>Testaufgaben 2012/2013 Basismodul Personal-/Testfall-Nr.: 150 Breit-Gossmann</b>

**Personalstammdaten:**

Familienname/ Geburtsname	Breit-Gossmann	PLZ, Wohnort Land	92549 Stadlern	Geb.-Datum Geb.-Ort	31.12.1947
Vorname	Heiner	Straße/ Hausnummer	MMM-S	Versicherungsnummer	19 631247 B 493
Vorsatzwort/ Geb.-Vorsatzwort		Anschlussszusatz	III. E	Staatsangehörigkeit Elterneigenschaft	deutsch nein
Namenszusatz/Titel Geb.-Namenszusatz		Geburtsland EU- staatsangehöriger		Rentenbezug	nein

23

Monat	laufendes Arbeitsentgelt	Einmalig gez. Arbeitsentgelt	Einzugstelle	Beitrags- gruppe	Personen- gruppe	Tätigkeits- schlüssel	Besonderheiten/Anmerkungen
10	Eintritt: 01.10.2012 als Estricharbeiter						
10	1.900,00		AOK NW			331122211	51 413 64 / Fahrartarif: 17
11	1.900,00		AOK NW	1111	101	331122211	Es liegt eine Aufforderung zur Abgabe einer gesonderten Meldung „Rente wegen Alters“ nach § 194 SGB VI (Rentenbeginn 01.02.2013) mit Grund der Abgabe 57 vor.
12	1.915,28		AOK NW	1111	101	331122211	incl. Nachzuschläge (Nacht 1) für 4 Std. Tarifizuschlag 35%
01	1.900,00		AOK NW	1111	101	331122211	
02	Austritt zum 28.02.2013						
02	1.900,00		AOK NW	3321	119	331122211	Regelaltersrente ab 01.02.2013

[http://www.gkv-ag.de/upload/TA\\_2012\\_2013\\_Stand\\_01.07.2012\\_oe\\_2162.pdf](http://www.gkv-ag.de/upload/TA_2012_2013_Stand_01.07.2012_oe_2162.pdf)

## Sprache - syntaktisch

---

- Szenarios zitieren die Oberfläche oder Anforderung
  - „Firmen/Betriebsstätten - Firmen - Grundlagen“
  - „..., Neu“ oder „..., Suchen“ oder „..., Eingaben“ anhängen
  - „Erfassung für Testfall 150 10/2012“, „Abrechnung ...“, „Prüfung ...“
- Aufteilung in Erfassung / Verarbeitung / Prüfung
- Technische Details sind versteckt, aber aufklappbar
  - Keine Variablen oder Methoden auf oberster Ebene
  - Legende ist der Ist-Zustand, nichts Neues einführen

## Sprache - inhaltlich

- Komplexität schön versteckt, aber wie Daten eingeben?
  - Variablen ...
  - Eingabe-Szenarios:

Scenario	Firmen/Betriebsstätten - Firmen - Einstellungen, Eingaben		
field:	sofortmeldepflichtiger Betrieb	value:	false

Script	
Firma format	MDE000
Demo Firma kopieren	\$FIRMA



▼ Scenario			
Scenario	Firmen/Betriebsstätten - Firmen - Einstellungen, Eingaben		
field:	sofortmeldepflichtiger Betrieb	value:	false
505ms			

- Felder intern referenziert über XPath-Ausdrücke
  - Sonderfälle Mehrfachauswahl und Funktionsergebnis
- Im Test **immer** natürlichsprachige Bezeichner verwenden

# Agenda

---

- Begriffe und fachlicher Hintergrund
- Beispiele, Vorstellung des Werkzeugs
- Das Konzept
- Die Umsetzung des Konzepts
- **Resümee**

# Der Punkt. Was treibt mich an?

---

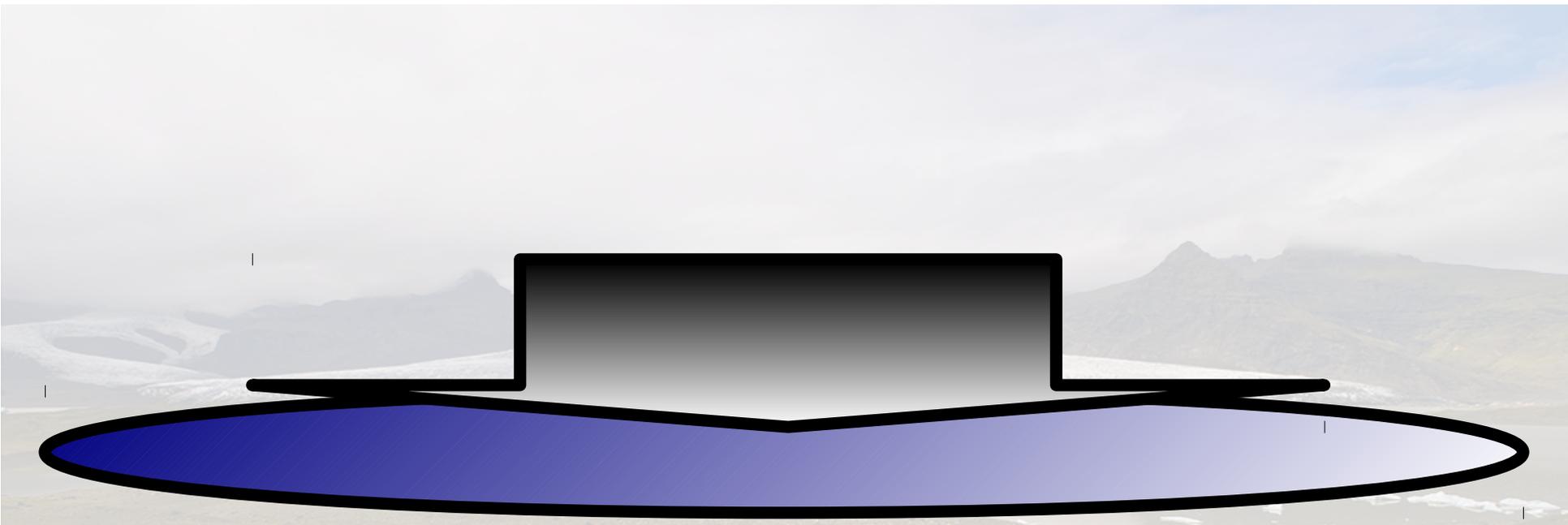
- Veränderung erwarten, das Veränderte benutzen!
  - Strukturen dynamisch ermitteln
  - Vorgabedaten verwenden
  - Abstraktion - Der Test ist der Test und nur der Test



## Der Punkt. Was treibt mich an?

---

- **Veränderung erwarten, das Veränderte benutzen!**
  - Strukturen dynamisch ermitteln
  - Vorgabedaten verwenden
  - **Abstraktion - Der Test ist der Test und nur der Test**



2.– 5. September 2013  
in Nürnberg



# Herbstcampus

Wissenstransfer  
par excellence

Vielen Dank!

Frank Ganske

[frank.ganske@mathema.de](mailto:frank.ganske@mathema.de)