

**Arbeitskreis Testmanagement**



# **Wege durch den Testmanagement-Dschungel**

TAV 30, München, 18.06.2010

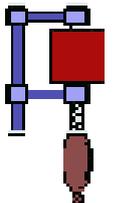
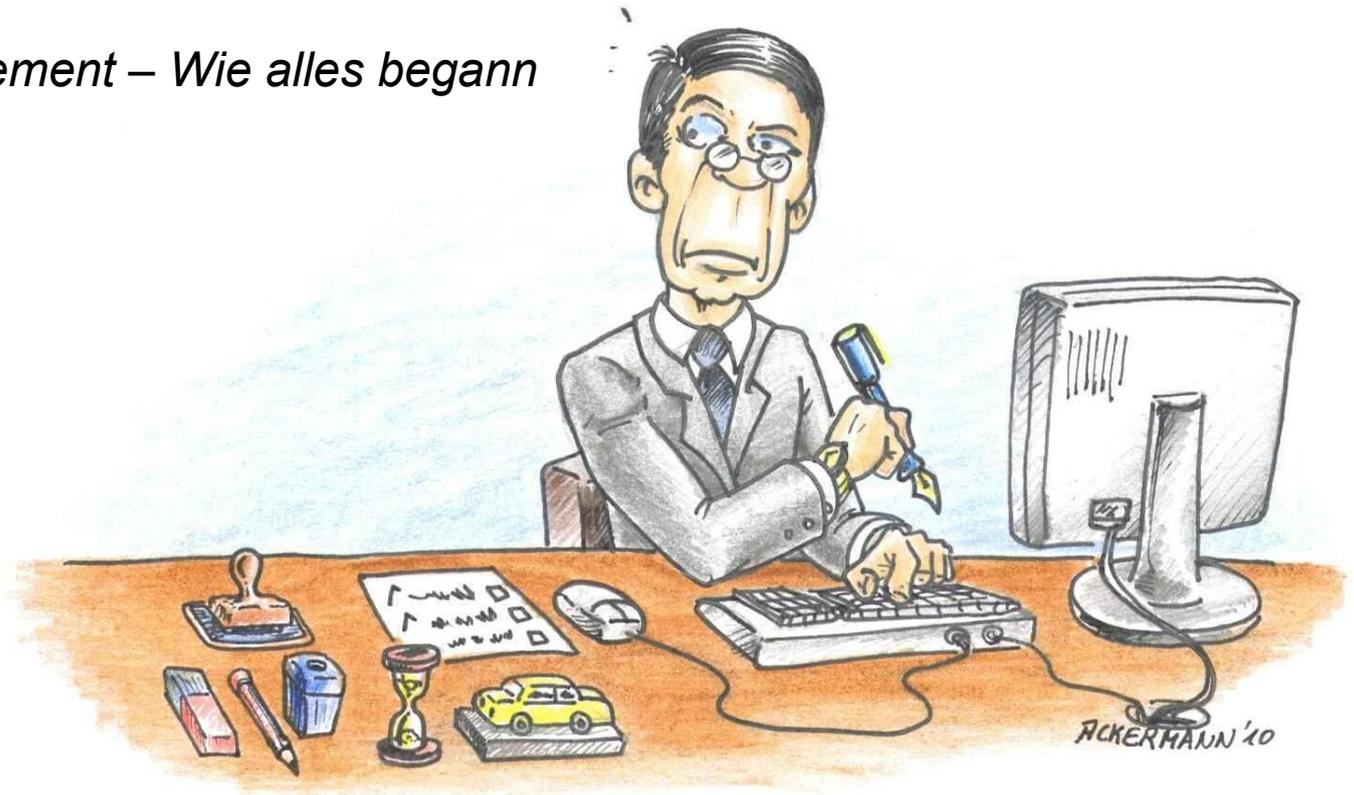
Autoren:

Dr. Andreas Birk  
Hans-Josef Eisenbach  
Holger Hanisch  
Bernhard Moritz  
Anton Schlatter  
Maud Schlich



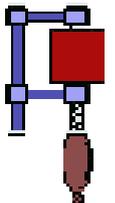
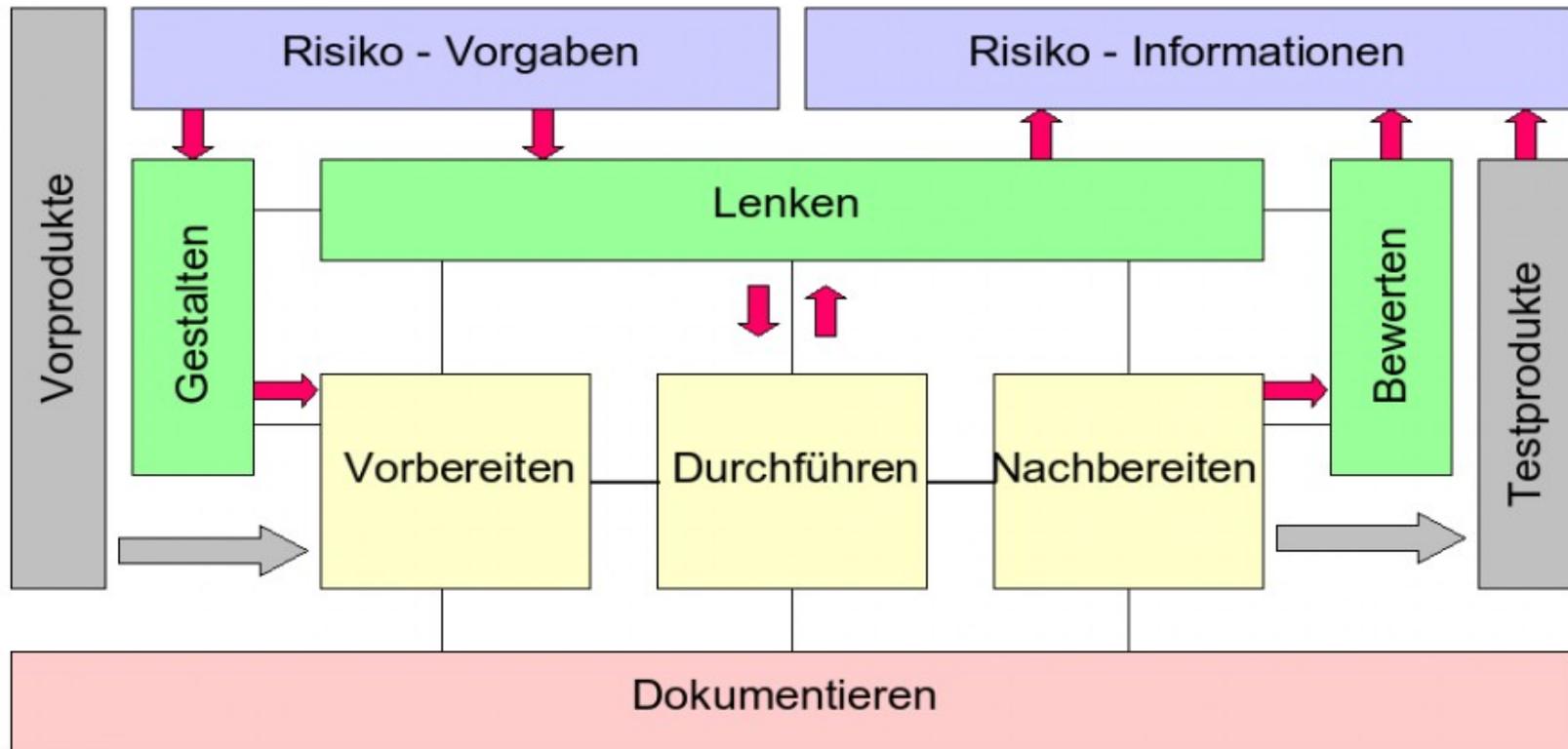
# Übersicht

- *15 Jahre AK Testmanagement – Wie alles begann*
- *Ärger im Paradies*
- *Testmanagement Praxis*
- *Standortbestimmung*
- *Ausblick*



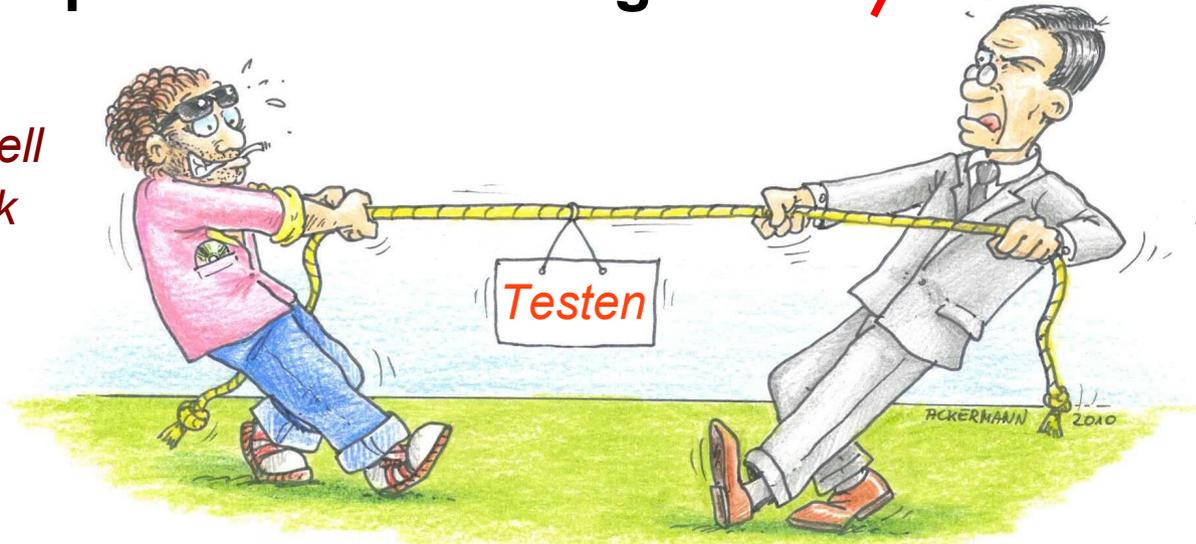
# Wege durch den Testmanagement-Dschungel

Unser Ausgangspunkt:



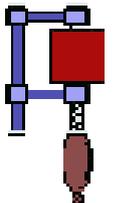
# Streitgespräch: Testmanagement $\neq$ Testmanagement ?

*Findus Schnell  
Internet-Bank*



*Hartmut Messbecher  
Klassische Filial-Bank*

- Methodische Vorgaben sperren Könner ein und führen zu suboptimalen Ergebnissen
- Produkt und Menschen sind der Mittelpunkt des Testens, nicht der Prozess
- Erst exploratives Testen bringt die echten Anforderungen ans Tageslicht
- Selbstorganisation im Team schafft motivierte Mitarbeiter
- Änderungen am Produkt aus Fehlerbehebung oder neuen Anforderungen werden sofort umgesetzt und getestet
- Der Test soll helfen, Anforderungen zu spezifizieren und zu verfeinern.
- Saubere Vorgaben strukturieren die Arbeit und liefern nachvollziehbare Ergebnisse
- Einheitliche Prozesse liefern messbare Qualität, das Produkt muss sich unterordnen
- Für exploratives Testen gibt es kein Endkriterium, es ist nicht planbar und kalkulierbar.
- Die Zuweisung von Teilaufgaben macht effektives Controlling erst möglich.
- Änderungen aus Fehlern oder neuen Anforderungen können nur zu bestimmten Zeitpunkten berücksichtigt werden.
- Quatsch, der Test dient ausschließlich dazu, Fehler aufzudecken!



# Schulen des Testens

Brett Pettichord stellte 2003 vier „Schulen des Testens“ vor:

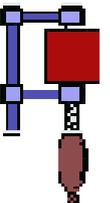
- *Analytical School*
- *Factory oder Standard School*
- *Quality School*
- *Context-Driven School*

2007 wurde eine fünfte Schule mitbetrachtet:

- *Agile School*

Vgl.: Brett Petticord: Schools of Software Testing. März 2007

[http://www.io.com/~wazmo/papers/four\\_schools.pdf](http://www.io.com/~wazmo/papers/four_schools.pdf)



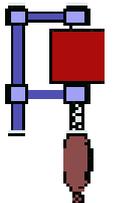
# Analytical School

*“The analytical school sees testing as rigorous and technical with many proponents in academia.”*

Fokus: Wissenschaftlich fundierte Techniken anwenden, um korrekte Funktion von Software nachweisen

Kernkonzepte:

- *Spezifikation der Software*
- *Struktur der Software*
- *Auswahl der passenden Testtechniken*
- *Berechnung von Code-Coverage.*



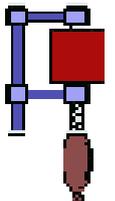
# Factory oder Standard School

*“The standard/factory school sees testing as a way to measure progress with emphasis on cost and repeatable standards.”*

Fokus: Bestimmung der Produktqualität, um Aussagen über den Projektfortschritt und das Erreichen der Projektziele zu machen

Kernkonzepte:

- *Fortschrittskontrolle des Entwicklungsprozesses*
- *Systematischen Durchführung des Testprozesses*
- *Klar vorgegebene Testabläufe*
- *Auch weniger qualifizierte Tester zu guten Testergebnissen befähigen*



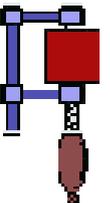
# Quality School

*“The quality school emphasizes process,  
policing developers and acting as the gatekeeper.”*

Fokus: Den Entwicklungsprozess steuern

Kernkonzepte:

- *Testergebnisse liefern Qualitätskriterien*
- *Quality Gates im Projektverlauf*
- *Testen als Instrument der Qualitätssicherung*



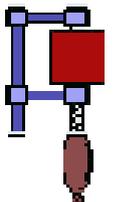
# Context-Driven School

*“The context-driven school emphasizes people, seeking bugs that stakeholders care about.”*

Fokus: Der Mensch bestimmt Schwerpunkt und Ablauf des Testens je nach den gegebenen Rahmenbedingungen

Kernkonzepte:

- *Menschen entscheiden, was wie und wann getestet wird*
- *Menschen befinden als Stakeholder über die Qualität der Software*
- *Berücksichtigung des spezifischen Kontexts der Software-Entwicklung*



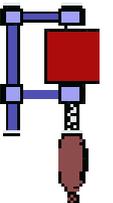
# Agile School

*“Software is an ongoing conversation.  
Testing tells us that a development story is complete.”*

Fokus: Werden die in der Story enthaltenen Anforderungen erfüllt?

Kernkonzepte:

- *Tests müssen automatisiert werden*
- *Test driven development*
- *Testen ist Aufgabe in der Kundenrolle*



# Schulen - Gibt es sie in der Praxis?

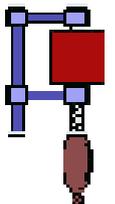


## Umfragen:

- TAV 26, 2007, 39 Teilnehmer („Psychotest“)
- ignite, 2010, 22 Teilnehmer (zus. Fragen zur Einstellung, zum Hintergrund)

Zur Verteilung der Schulen in der Praxis haben wir zwei Hypothesen:

- *Testmanager haben eine eindeutige Präferenz für eine Schule*
- *In Deutschland überwiegen die Factory- oder Quality-Testmanager*



# Der Fragebogen

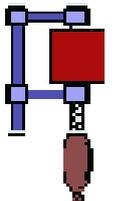
## Beispiel-Frage „Psychotest“ (TAV und iqnite):

*Was sind für Sie die wichtigsten Eigenschaften eines Testmanagers?*

- F - Projektplanungs- und -steuerungsfähigkeit
- Q - **Kenntnis und Einhaltung der definierten Testprozesse**
- A - **Technischer Sachverstand und Kenntnis von etablierten Testmethoden**
- C - **Teamintegration und Kommunikation**

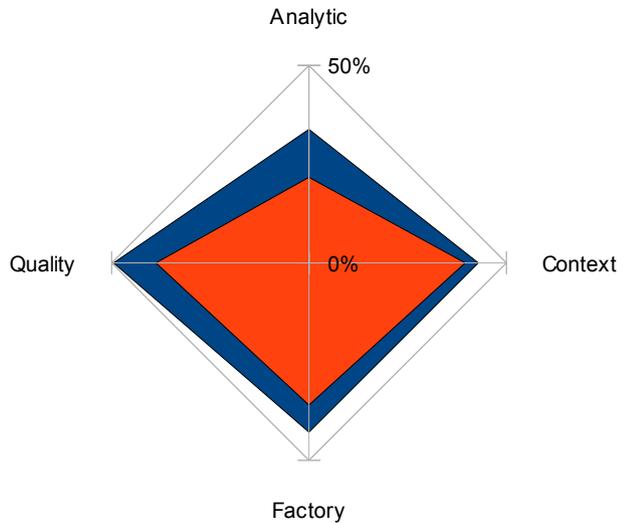
## Fragen zur „Einstellung“, Werte 1..10 (neu für iqnite)

- **Testen ist eine technische Aktivität, die formale und wissenschaftlich fundierte Prinzipien beachten muss (z.B. Testabdeckungsmaße).**
- **Testen ist ein Mittel, um Fortschritt bei der Software-Entwicklung festzustellen**
- **Testen sichert die Qualität im Projektablauf, indem es Vorgaben an die Entwicklung macht und deren Einhaltung überprüft**
- **Beim Testen stehen die Stakeholder-Interessen im Vordergrund: Es muss vor allem die Bugs finden, die aus Stakeholdersicht am problematischsten wären.**

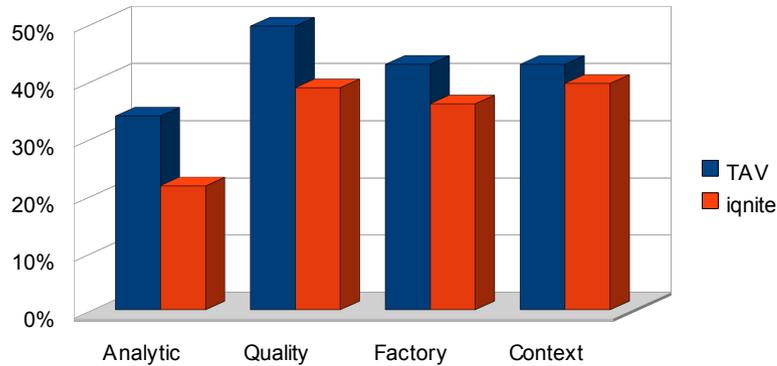
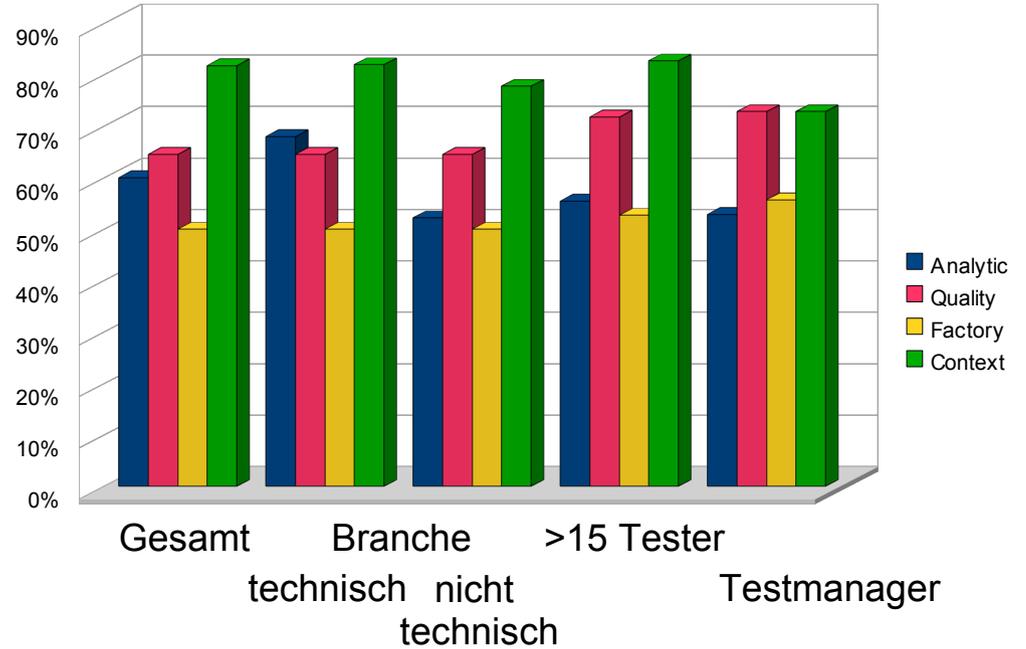


# Auswertung der Fragebögen

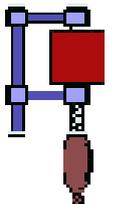
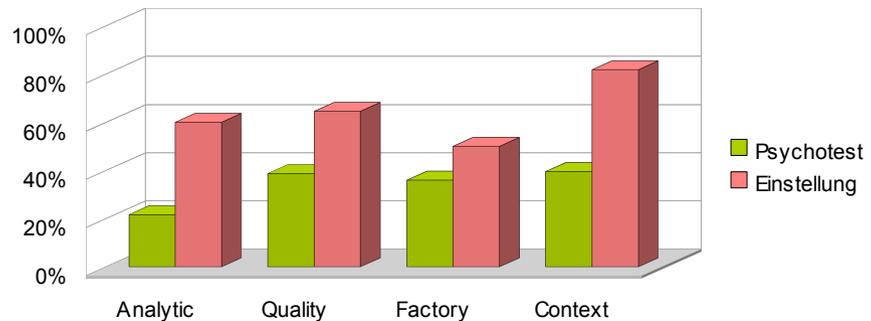
„Psychotest“ TAV vs. iqnite:



iqnite: „Einstellung“



iqnite: „Psychotest“ vs. „Einstellung“:



# Resümee aus der Befragung

Unsere Hypothesen konnten nicht bestätigt werden.

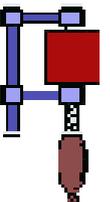
Für das Testmanagement können die 5 Schulen in zwei Gruppen zusammengefasst werden:

- *Analytische Schule entfällt (TM-fern- nur Basisprozess)*
- *Factory- und Quality-School sind aus TM-Sicht eng beeinander*
- *Context-driven und Agile-School liegen aus TM-Sicht ebenfalls eng beieinander*

Unsere Bezeichnungen der Gruppen:

- *Planungsbasiert: Prozesse, Berücksichtigung von Vorgaben, Kontrolle*
- *Erfahrungsbasiert: Personen, Reaktion auf Umfeldfordernisse*

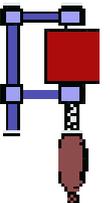
Möglicherweise hilft uns diese Unterscheidung allein noch nicht wirklich weiter.



# Kategorien zur Unterscheidung der Gruppen aus TM-Sicht

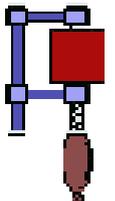
Im Positionspapier Testmanagement haben wir das Thema Testmanagement nach 7 Kategorien gegliedert und Fragen bzw. Aufgaben dazu definiert.

- *A - Inhaltlich/technische Vorgaben*
- *B - Organisation: Aufbau, Ablauf Aufbauorganisation*
- *C - Ergebnisse, Dokumente*
- *D - Ressourcen, Infrastruktur*
- *E - Mitarbeiter und Teams*
- *F - Konfigurationsmanagement*
- *G - Qualitätsmanagement*



# Zuordnung TM-Kriterien zu Gruppen

	Kriterien	Planungsbasiert	Erfahrungsbasiert
<b>A</b>	<b>Testzeitpunkt</b> <b>Anwendung von Testmethoden</b> <b>Standards, Normen</b> <b>Orientierung</b>	nach der Entwicklung Vorgabe durch TM Ja Prozess	während der Entwicklung Best Practices durch das Team Nein Produkt/Menschen
<b>B</b>	<b>Definition von Testfällen</b> <b>Software-Änderungen</b> <b>Testorganisation</b> <b>zu testende „Losgröße“</b> <b>Rolle des Testmanagers</b>	vollständig vor Testbeginn Sonderfall Abgrenzung Entwicklung vs. Testen gesamtes Produkt konkrete Arbeitszuteilung	gestaltbar bis Testausführung kurzfristige Änderungen Normalfall Entwicklung integriert Testen Inkremente bis zum Gesamtprodukt Management des selbstorganisierten Ablaufs
<b>C</b>	<b>Ziel</b>	Fehler finden	Anforderungen ermitteln
<b>E</b>	<b>Qualifikation der Mitarbeiter</b>	Angeleitete Tester	Selbständige Generalisten
<b>G</b>	<b>Stellenwert der Qualität</b>	Qualität wird erst im Test bewertet	Qualität ist Teil der SW-Entwicklung



# Bewertung der Sichten

## Planungsbasiert

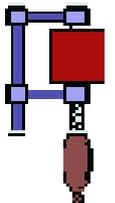
- *TAV-TM ist historisch der Factory/Quality School zugeordnet*
- *Termine und Kosten kontrollieren, Pläne einhalten*
- *Reproduzierbare Ergebnisse und Dokumentationen sind Pflicht*

## Erfahrungsbasiert

- *Standards und Normen sind Agilen Projekten fremd*
- *Selbstorganisation ersetzt Mitarbeiterführung*
- *Abgrenzung zwischen Entwicklern und Testern unmöglich*

## Gemeinsamkeiten

- *Fragen zur Automation bzw. Werkzeugeinsatz*
- *Ressourcen, Infrastruktur und Konfigurationsmanagement*

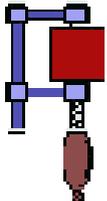
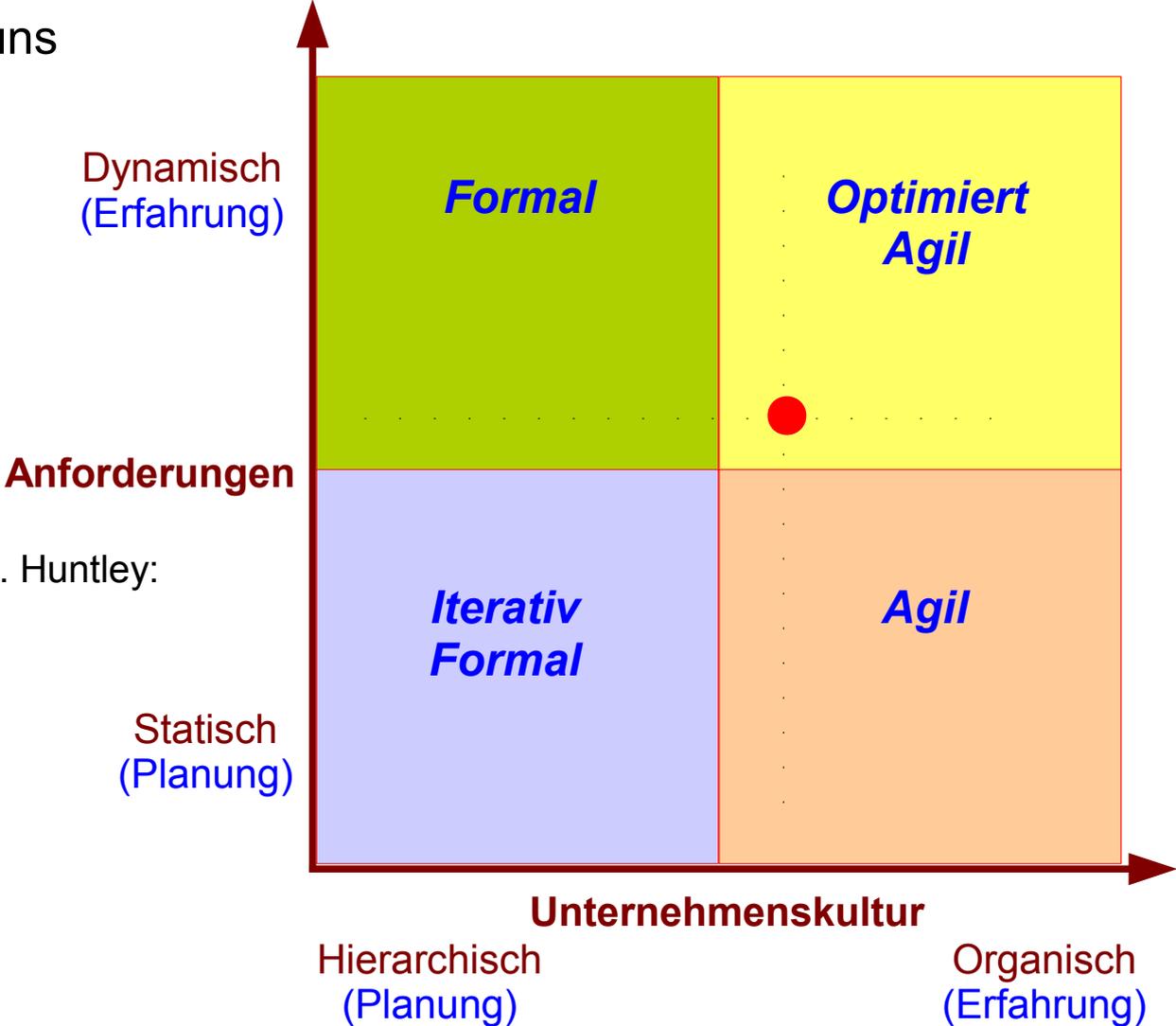


# Schema zur Standortbestimmung ?

Möglicherweise fehlen uns  
weitere Dimensionen

Nach:  
Vishnu Vinekar, Christopher L. Huntley:

*Agility versus Maturity:  
Is There Really a Trade-Off?*  
IEEE Computer, Mai 2010.



# Überlegungen zur Standortbestimmung

## Schritt 1: Positionsbestimmung

- *Katalog aus 'Kriterien zur Unterscheidung der Gruppen aus TM-Sicht' erstellen*
- *Zu jedem Eintrag wird die Frage nach der Unternehmenskultur bzw. zur Stabilität der Anforderungen gestellt.*

*Beispiel: Kriterium „Anwendung von Testmethoden“:*

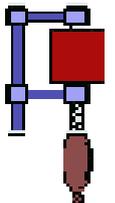
- ◆ *Regler hierarchisch-organisch: Testmethoden werden vorgegeben oder im Basisprozess ausgewählt*
- ◆ *Regler statisch-dynamisch: Vollständigkeit des Tests kann oder kann nicht gewährleistet werden*

## Schritt 2: Handlungsspielräume ermitteln

- *Auswerten des Kriterienkatalogs*
- *Testmanagementrolle ableiten*

## Schritt 3: Gestaltung des Testmanagements

- *Freiräume ausnutzen*
- *Grenzen erkennen*



# Wie geht es weiter?

Neuer Schwerpunkt: Testmanagement bei kontinuierlicher Weiterentwicklung

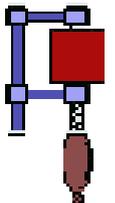
Unser Fokus bisher:

- *großer Lieferumfang*
- *Testen nach Fertigstellung des Gesamtprodukts*
- *Einmalprojekt*
- *Eigenständiges Testprojekt*

Fokus der kommenden Diskussion:

- *kleiner Lieferumfang*
- *Testen während der Entwicklung des Gesamtprodukts*
- *Kontinuierliche Weiterentwicklung*
- *Application Lifecycle Management unterstützt diese Integration*
- *Testen in Produktentwicklung integriert*

Fragestellung: Welche Auswirkungen treffen das Testmanagement?



# Referenzen

Brett Petticord: Schools of Software Testing. März 2007

[http://www.io.com/~wazmo/papers/four\\_schools.pdf](http://www.io.com/~wazmo/papers/four_schools.pdf)

TAV Arbeitskreis Testmanagement:

<http://www.caseconsult.com/tavtm>

Vishnu Vinekar, Christopher L. Huntley:

*Agility versus Maturity: Is There Really a Trade-Off?* IEEE Computer, Mai 2010.

Dank

Die Autoren bedanken sich bei Thomas Ackermann aus Aachen für die Karikaturen.

