

# IT-Projekt-Management

SoapUI (Pro)

Dr. The Anh Vuong

email: [vuongtheanh@netscape.net](mailto:vuongtheanh@netscape.net)

http: [www.dr-vuong.de](http://www.dr-vuong.de)

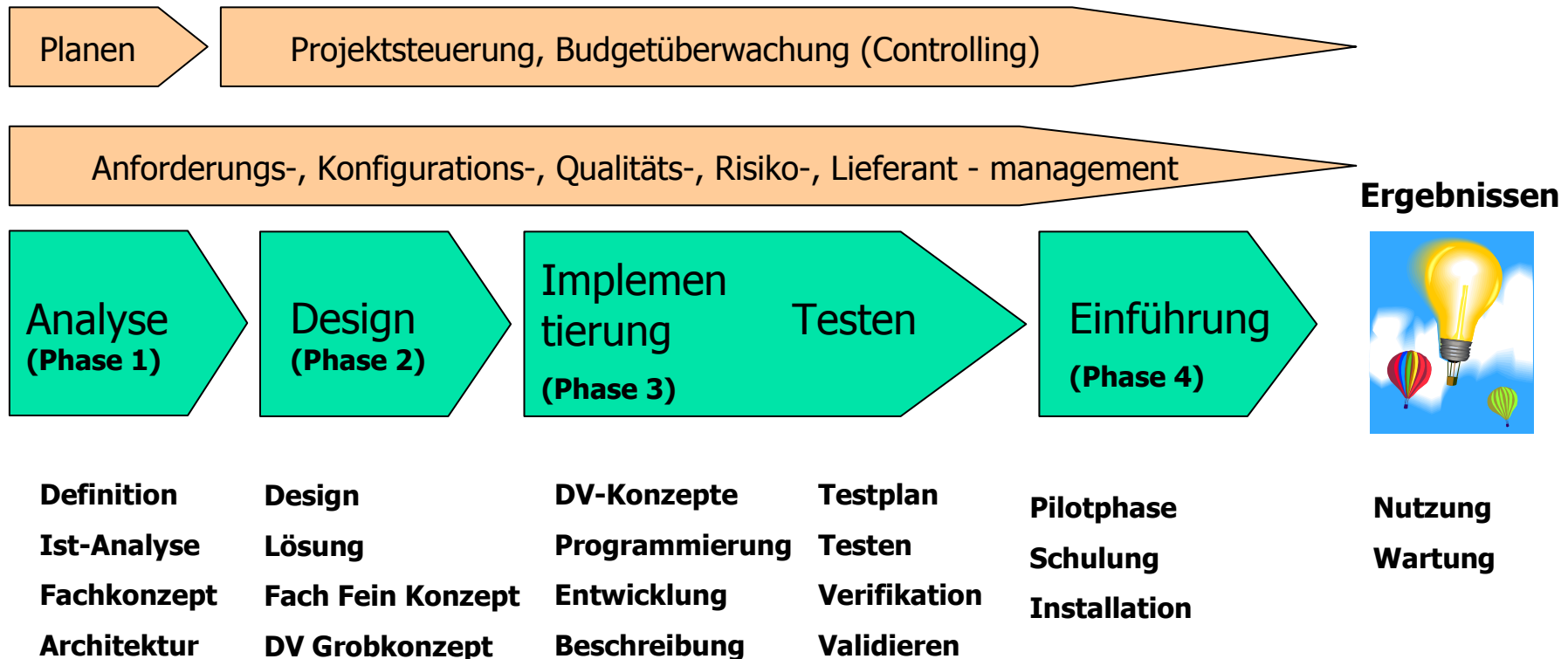




# Testen von IT System

---

# IT-Projekte: Entwicklungsprozesse -1 -



# Software Fehler

## → Stromausfall USA August 2003

Acht Staaten im Nordosten der USA und Teile Kanadas blieben für **fünf Tage ohne Strom**.

Insgesamt waren **50 Millionen Menschen betroffen**.

Ursache: **Softwarefehler des Managementsystems zur Überwachung und Steuerung von Stromnetzen** beim Erzeuger FirstEnergy.

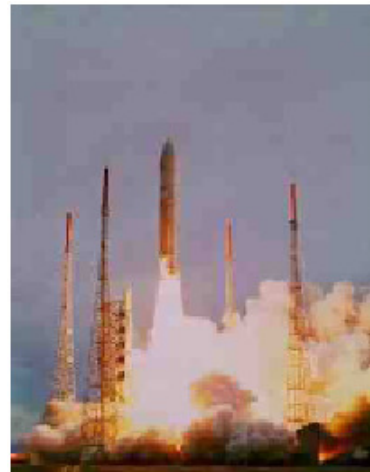
## → Jungfernflug Ariane 5 am 4. Juni 1996

Nach zunächst erfolgreichem Start veränderte die **Rakete** ihre Flugbahn und **explodierte**.

**Materialschaden** ca. **600 Millionen Euro**.

Prestigeverlust und die wirtschaftlichen Verluste der ESA waren beträchtlich.

Ursache: **Softwarefehler im Flugkontrollsystem** der Rakete.



Start der Ariane 5 ...



kurze Zeit später...



# SW Fehler

---

Die Ariane 501 Untersuchungskommission gab in ihrem Bericht<sup>1</sup> u.a. zu bedenken:

...,**dass es nicht ausreicht, einzelne Bauteile oder Schnittstellen zu testen.**

Eine **Überprüfung des gesamten Systems** wäre notwendig gewesen, auch wenn eine „hundertprozentige“ Genauigkeit nicht zu gewährleisten gewesen wäre.

Zudem wäre es **in der Verifikations-/Validationsphase** wichtig gewesen, dass die **Programmteile und die dazugehörige Dokumentation** strengen **Reviews unterzogen worden wäre**. An diesem Prozess hätte man **Projektpartner und zusätzlich externe Experten beteiligen** sollen.

**Source**

<http://ravel.esrin.esa.it/docs/esa-x-1819eng.pdf> („*Ariane 5 - Flight 501 Failure, Report by the Inquiry Board*“, Juli 1996)



# Verifikation

---

**Überprüfung, dass ausgewählte Arbeitsergebnisse ihre spezifizierten Anforderungen erfüllen.**

**„Haben wir das Produkt *richtig* gebaut?“**

**Grundlagen: Fachkonzept, DV Konzept**



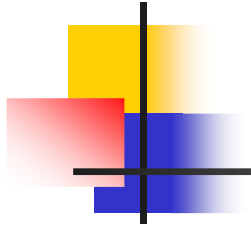
# Validation

---

**Testen, dass ein Produkt oder eine Produkt-Komponente seinen/ihren geplanten Zweck erfüllt, wenn es/sie in seine/ihre Zielumgebung gebracht wird.**

**„Haben wir das *richtige* Produkt gebaut?“**

**Grundlagen: Lastenheft**



## Verifikation

- Erfüllt die IT-Produkt die Anforderungen?
- Hat die Entwickler verstanden, wie es funktionieren soll?
- Wurden die Schnittstellen korrekt implementiert?
- Haben wir das Produkt *richtig* gebaut?
- Modul-, Integrations-Test

## Validation

- Erfüllt das entwickelte IT Produkt die Bedürfnisse und Erwartungen des Kunden?
- Haben wir entwickelt, was es bezwecken soll?
- Funktioniert das System im Gesamt- System?
- Haben wir das *richtige* Produkt gebaut?
- Abnahme-Test





# Test: Prozess

---

- Teststrategie festlegen
- Testkonzept erstellen
- Testfälle (Test Szenarien) entwickeln
- Test planen
- Test durchführen
- Testdaten erheben und analysieren
- Maßnahme festlegen
- Ergebnisse kommunizieren



# Test-Strategie

---

- Testphasen
- Testobjekt und Testumgebung
- Test-Schnittstelle
- Liste der verwendeten Testmethoden
- Test-Tools



# Test Verfahren in IT

---

- **Statik Test & Dynamik Test**
  - Statik: Walk throughs, Review
  - Test fälle, Test Szenarien
- **Box Test**
  - White-Box
  - Black-Box
  - Visual Test
  - Gray-Box



# Szenarien - Testfälle

---

- Grundlast
- Höchstlast
- Worst Case



# Test Typen in IT

---

- Funktionstest (gegen Anforderungen)
- Technischer Test
- Schnittstell-Test
- Integrationstest
- Abnahme Test
- Regressionstest
- Last-test / Performanz Test
- Boundary-Test und Stress-Test



# Last Test

---

- Ziele des Last-Tests sind die Prüfungen, wie das zu testende System bei der Belastung reagiert.
- Die Kennzahlen sind die angeforderte Mengegerüst (von den Kunden):
  - Anzahl der der Benutzer
  - Anzahl der Parallel Zugriff
  - Anzahl der zur Verfügung eingestellten Datenkanäle, Geräten
  - Bandbreite der für die Verbindungen.



# Boundary Test

---

- Ziele des Boundary-Tests sind die Prüfungen, wie das zu testende System bei seinen Grenzwerten reagiert.
- Die Grenzwerte des IT-Systems sind die maximale Anzahl der:
  - Datenbankzugriffe
  - eingeschalteten Datenkanäle
  - Verbindungen der IP-Netzgeräte
  - Begrenzung der Bandbreite der WAN-Verbindungen.



# Stress Test

---

- Ziele des Stress-Tests sind die Prüfungen, wie das zu testende System bei chaotischen Dateneingaben reagiert.
- Stress für das IT-System kann aus folgenden Gründen entstehen:
  - Die Anforderungsanzahl eingeschalteter Datenkanäle erhöht sich extrem.
  - Die Anzahl der Zugriffe erhöht sich spontan, z.B. bei Brandfall 100 Anrufe zum Hauptserver im Minutenbereich.
  - Benutzer geben bei Überlastung eine Folge von falschen Eingaben in das System ein.





# Test-Tools

---

- Open Source
  - JMETER : WebTest
  - Selenium
  - SoapUI (Pro)
- Professionell
  - HP Unified Functional Testing (UFT)
  - Micro Focus SilkTest
- Test Management
  - JIRA
  - TMT