# Herzlich willkommen
# DevDay 2016 – Software modellieren, aber richtig!

Arif Chughtai, Björn Reber

DIGICOMP

{ DEVDAY } by DIGICOMP

# Referenten

- Arif Chughtai
  - Berater und Trainer
  - Software-Engineering
  - Java-Technologie
- Björn Reber
  - Software-Entwickler und Trainer
  - Java-Technologie
  - OpenSource und SAP

# Inhalt

- Es geht nicht um…
- Ganz kurz nochmal…
- Modellieren im Software-Kontext
- Die Wunschvorstellung
- Die harte Realität
- Software modellieren, aber richtig!
- Beispiele mit EnterpriseArchitect
- Zusammenfassung

# Es geht nicht um…

# Es geht nicht um…

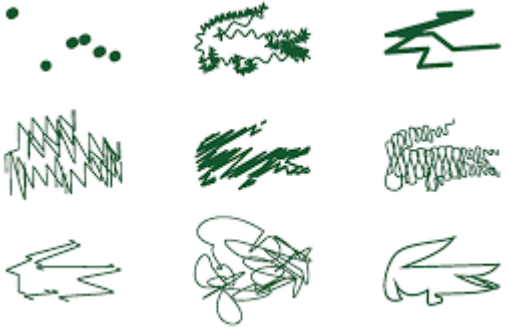
- **…Tools…**
  **…, die es von selber richtig machen ;-)**
  **(die gibt es nicht!)**
- A Fool with a Tool…
- A Fool without a Tool…

# Ganz kurz nochmal…

Was ist ein Modell?

**Original**

Abstraktion



**Modelle**

# Ganz kurz nochmal…

## Modelle nutzen

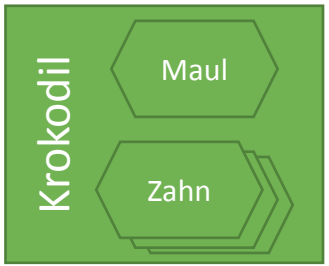
**Modell**

Verwendung

- Kommunikation
- Dokumentation
- Analyse
- Simulation
- Erstellung / Generierung
- …

# Ganz kurz nochmal…

## Modelle repräsentieren



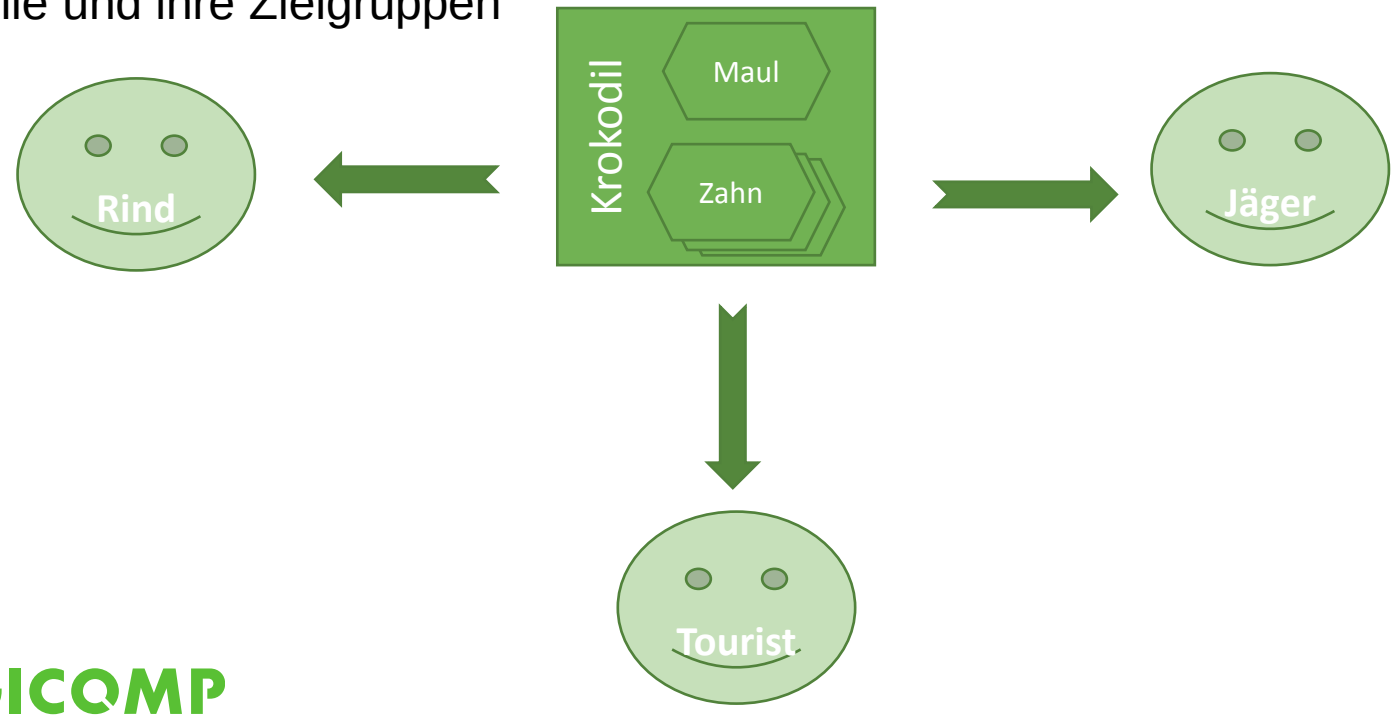

Krokodil:

- Maul
- Zahn (n)

Krokodil = Maul + ∑ Zahn

# Ganz kurz nochmal…

## Modelle und ihre Zielgruppen



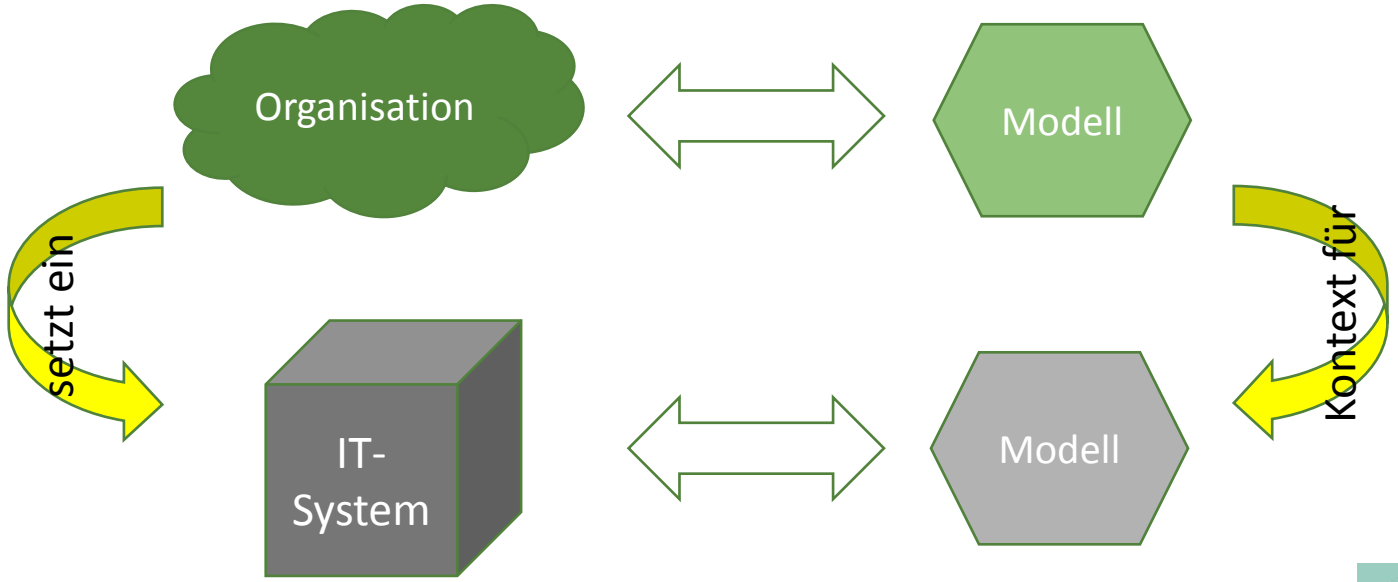
# Ganz kurz nochmal…

## Modelle erstellen

- Zahlreiche Methoden und Mittel
- Standards
- Werkzeuge

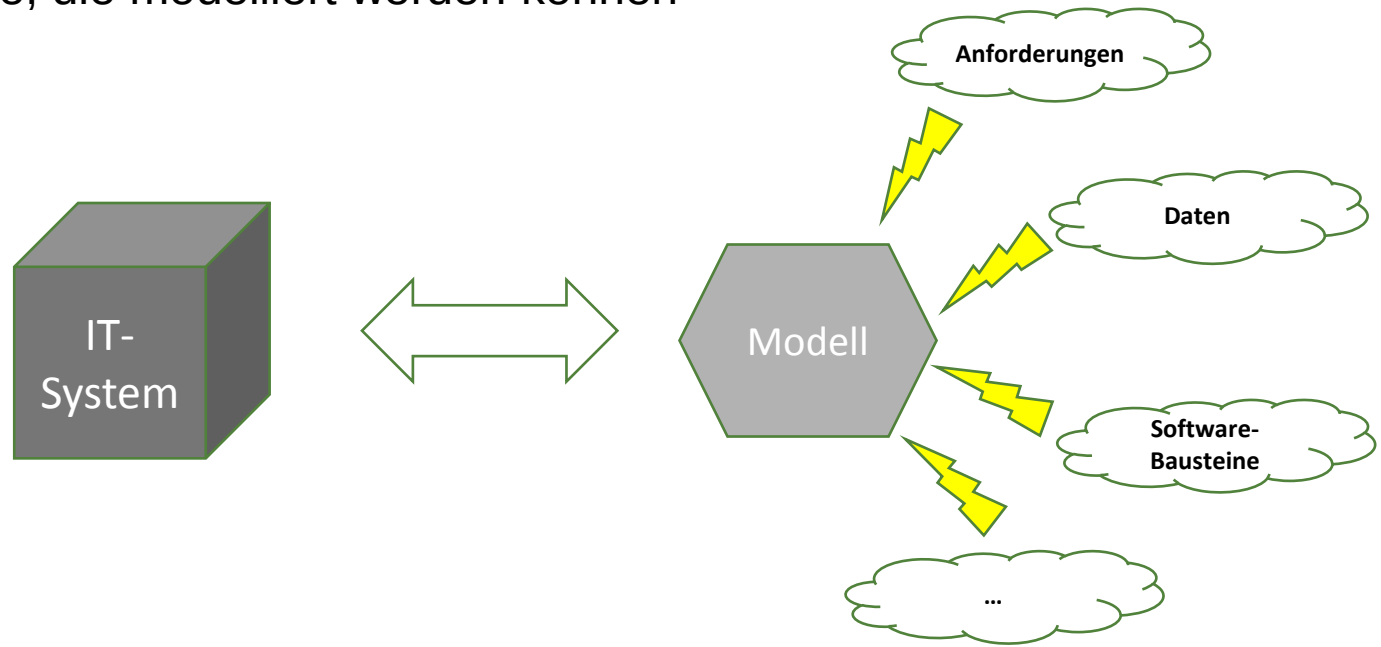
# Modellieren im Software-Kontext

## Abstraktionsebenen



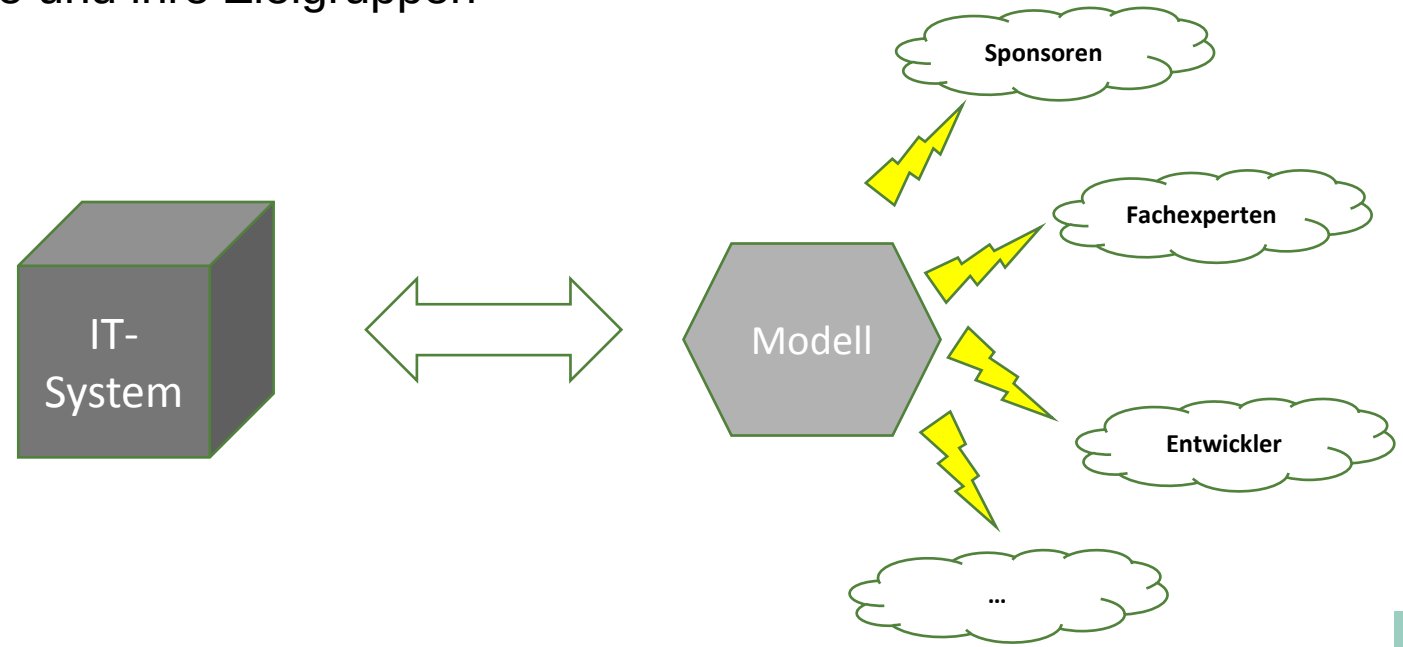
# Modellieren im Software-Kontext

## Aspekte, die modelliert werden können



# Modellieren im Software-Kontext

## Modelle und ihre Zielgruppen



# Die Wunschvorstellung

Modelle sollen…

- Kommunikation unterstützen
- Komplexität reduzieren
- Zusammenarbeit erleichtern
- Wissen festhalten und explizit machen
- Entscheidungsfindung unterstützen
- Übersicht schaffen
- Qualität verbessern
- Verständnis schaffen
- Prüfen und Testen ermöglichen
- Nachvollziehbarkeit ermöglichen
- …

# Die harte Realität

Modellierung…

- …wird nicht praktiziert
- …bringt keinen Nutzen
- …verursacht hohe Aufwände
- …wird abgelehnt
- …

# Die harte Realität

Modelle…

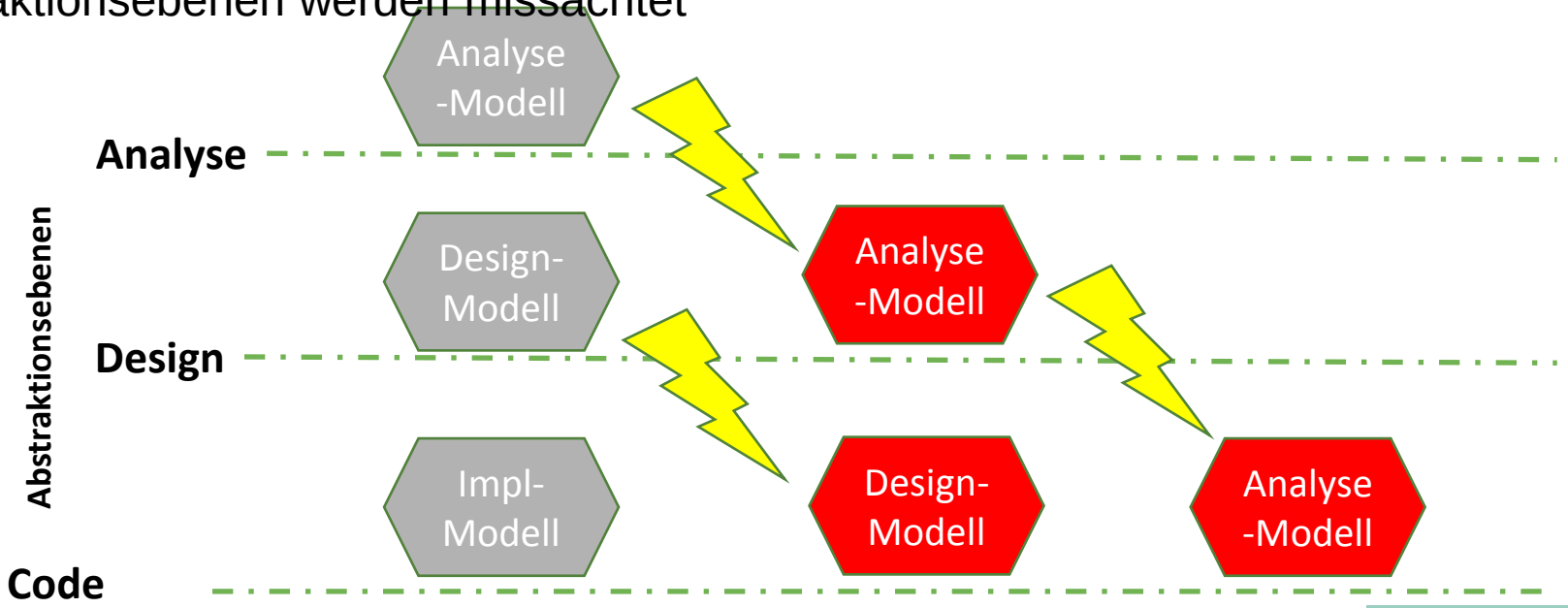
- …werden nicht genutzt
- …bringen keinen Nutzen
- …werden abgelehnt
- …

# Die harte Realität

- **Gründe???**

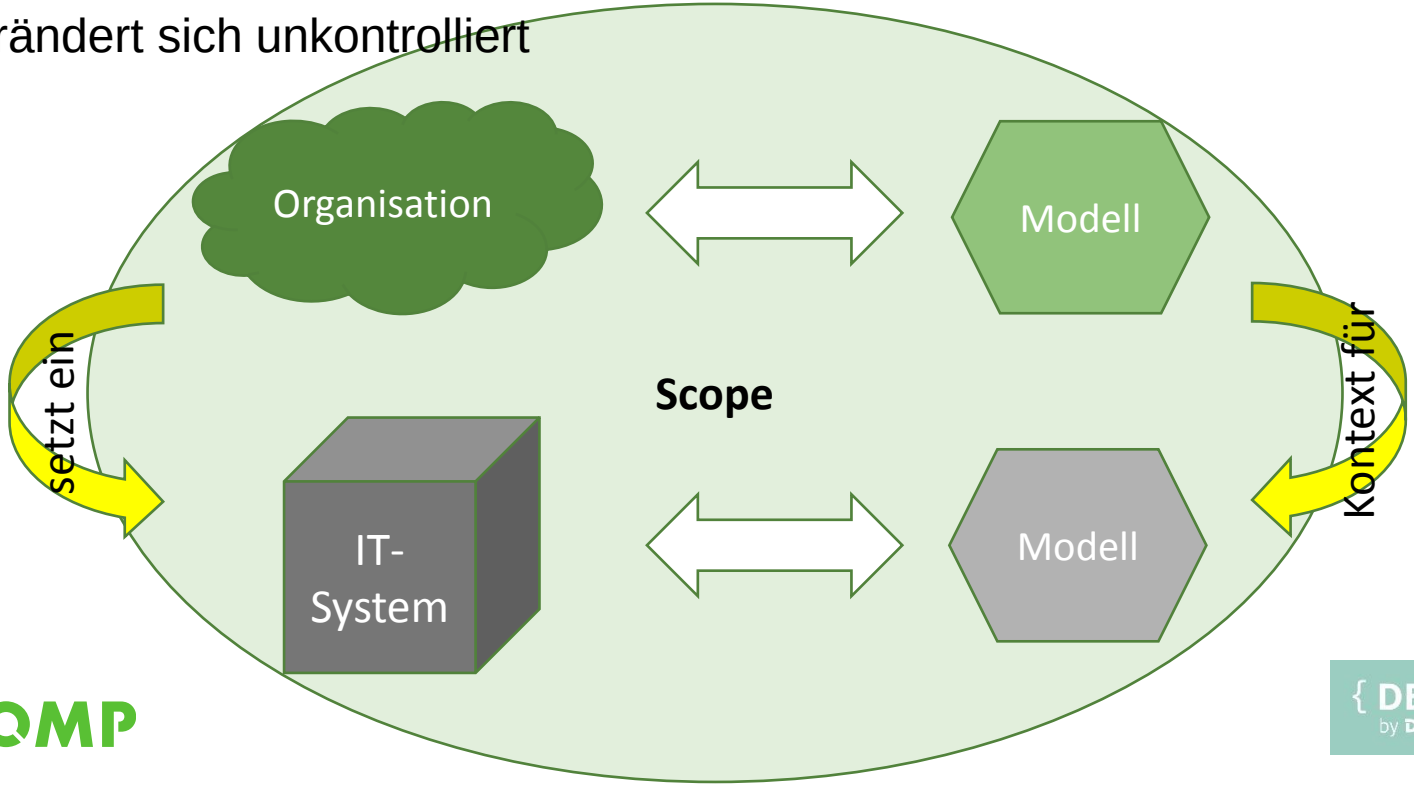
# Die harte Realität

Abstraktionsebenen werden missachtet



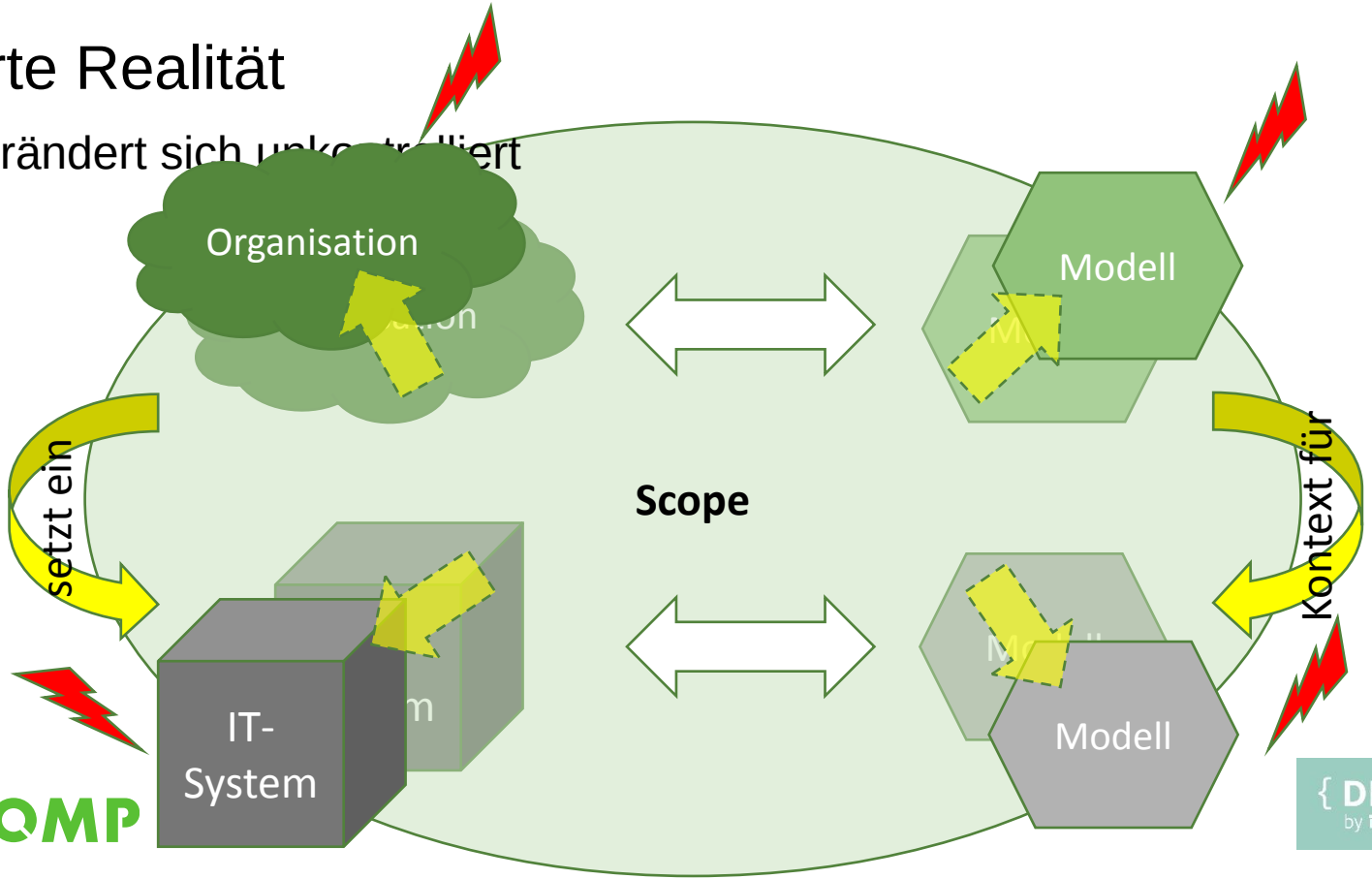
# Die harte Realität

Scope verändert sich unkontrolliert



# Die harte Realität

Scope verändert sich unkontrolliert



# Software modellieren, aber richtig!

## So geht das…

- Modelle müssen einen Nutzen bringen
- Modelle müssen zum Kontext passen
  - Abstraktionsgrad, Zielgruppen etc.
- Modelle müssen eingeführt werden
- Modelle müssen verfügbar sein
- Modelle müssen benutzerfreundlich repräsentiert werden

# Software modellieren, aber richtig!

## So geht das…

- Modelle müssen (fortlaufend) gepflegt werden
  - Wartbarkeit
- Mittel (Werkzeuge, Standards etc.) müssen zum Kontext passen
  - Die „richtigen“ Mittel auswählen
  - Mittel und Modell unterscheiden

# Software modellieren, aber richtig!

## So geht das…

- Mittel müssen beherrscht werden
  - Die Mittel "richtig" anwenden/benutzen
  - Richtlinien
  - Methodik
- Verständnis für Modellieren schaffen
  - Modellieren als anerkannte Aufgabe etablieren
  - Modelle als relevante Arbeitsergebnisse etablieren

# Software modellieren, aber richtig!

## So geht das…

- Nutzen aufzeigen
  - Quantifiziert
- Aufwand kennen und kommunizieren
  - Modellierung und Modelle ohne Aufwand gibt es nicht!
- Ablehnung pro aktiv begegnen
  - „Im Code liegt DIE Wahrheit!“, „Modelle sind Theorie!“ etc.

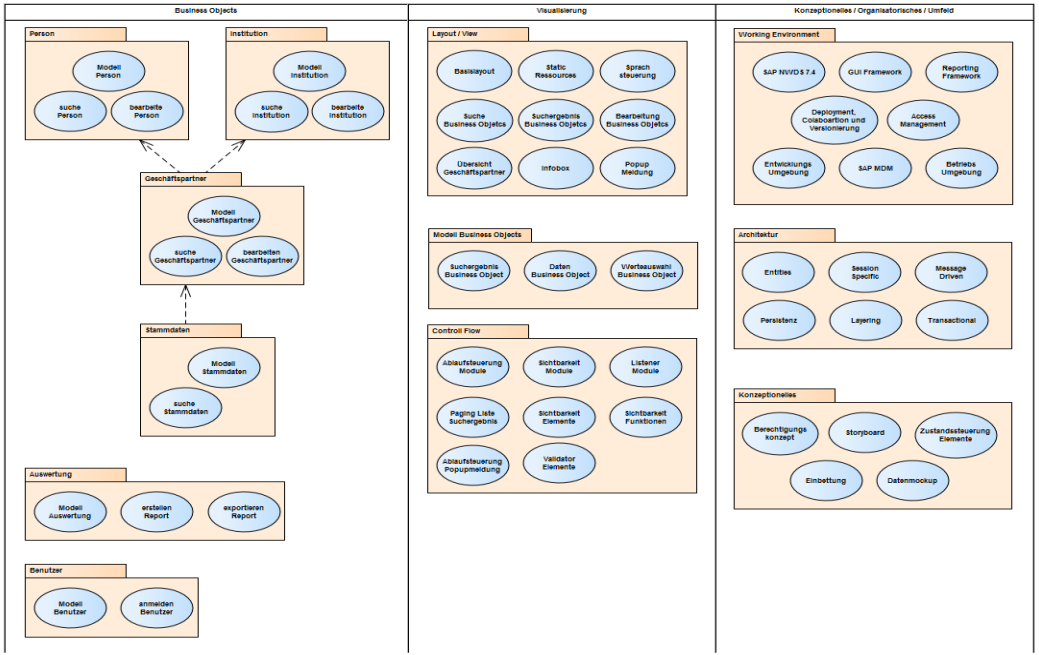
# Beispiele mit EnterpriseArchitect

Beispiele zeigen…

- Zum Kontext passende Modelle
- Wartbare Modelle
- Benutzerfreundliche Modelle (Einsatz von Farben etc.)
- Richtlinien und Methodik
- …

# Beispiele mit EnterpriseArchitect

## Use-Case-Modell: Modell für Aufwandsschätzung

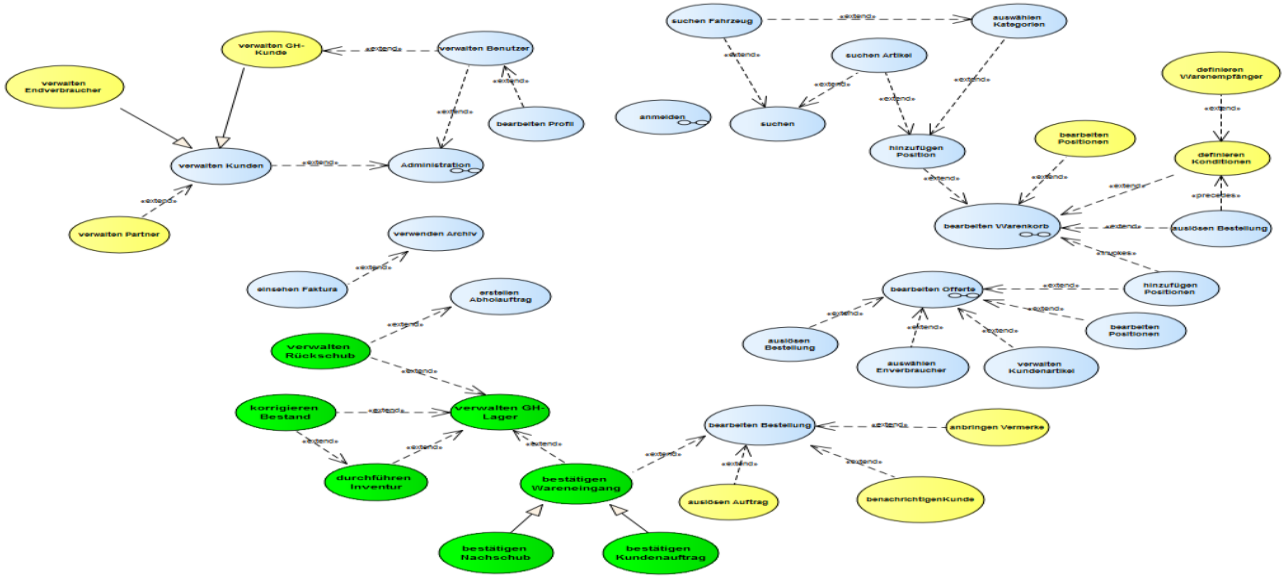



DIGICOMP

{ DEVDAY } by DIGICOMP

# Beispiele mit EnterpriseArchitect

## Use-Case-Modell: Use Cases eines IT-Systems

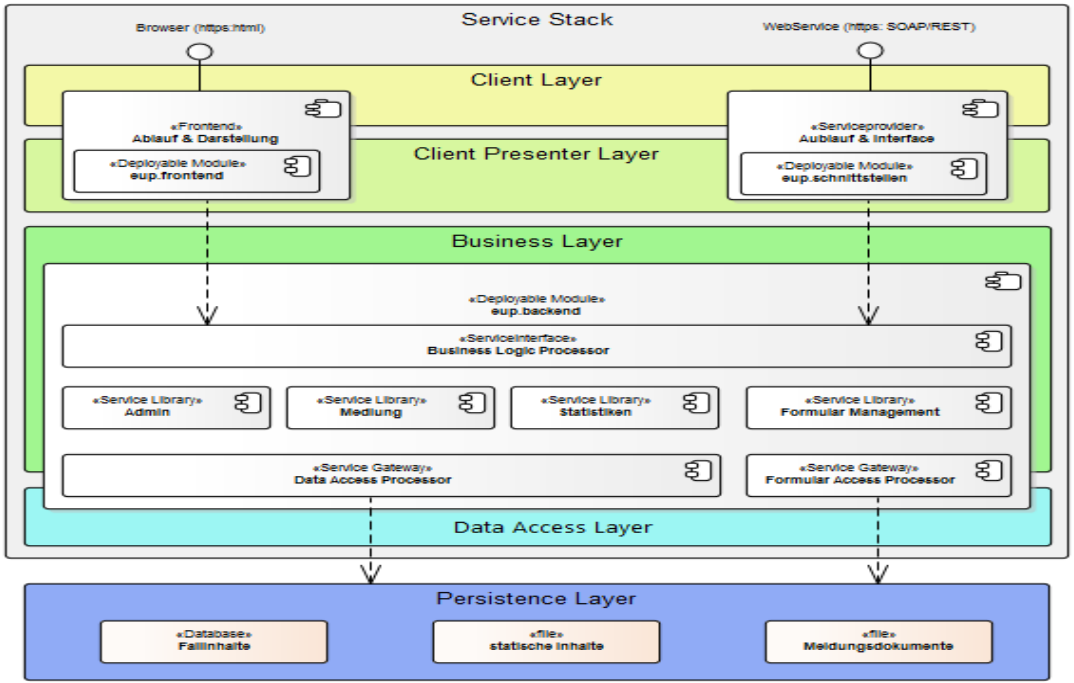



DIGICOMP

{ DEVDAY } by DIGICOMP

# Beispiele mit EnterpriseArchitect

## Architektur-Modell: Logische Sicht und Verteilungssicht




DIGICOMP

{ DEVDAY } by DIGICOMP

# Zusammenfassung

- Auf Nutzen achten
  - Kontext beachten!
- Nutzen erhalten
  - Weniger ist mehr!
- Modellierung und Tools beherrschen
- Richtlinien für Standards und Tools
- Nicht in Schönheit sterben ;-)

# Zusammenfassung

## Definition

***Abbildung***. *Ein Modell ist immer ein Abbild von etwas, eine Repräsentation natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.*
***Verkürzung***. *Ein Modell erfasst nicht alle Attribute des Originals, sondern nur diejenigen, die dem Modellschaffer bzw. Modellnutzer relevant erscheinen.*
***Pragmatismus***. *Pragmatismus bedeutet so viel wie Orientierung am Nützlichen. Ein Modell ist einem Original nicht von sich aus zugeordnet. Die Zuordnung wird durch die Fragen für wen, warum und wozu relativiert. Ein Modell wird vom Modellschaffer bzw. Modellnutzer innerhalb einer bestimmten Zeitspanne und zu einem bestimmten Zweck für ein Original eingesetzt. Das Modell wird somit interpretiert.*

Stachowiak, Herbert 1973

# Abschluss

- **Danke… :-)**
- **Fragen ???**