

# ABBILDUNG VON CONSTRAINTS FÜR BENUTZEROBERFLÄCHEN

HANS-GEORG SCHLADITZ

# AGENDA

- Problemstellung und Problem
- Grundlegende untersuchte Ansätze
  - Object Constraint Language (OCL)
  - UML-Profiles
  - Domain Specific Language (DSL)
- Konzept of Proof



# PROBLEMSTELLUNG UND PROBLEM

## **Zielstellung:**

- Ziel I. Finden von Möglichkeiten um Constraints möglichst einfach zu definieren
- Ziel II. Überblick über aktuelle Ansätze (Frameworks und Konzepte) zur Erstellung von Benutzeroberflächen aus Modellen mit Constraint-Definition geben

**Fokus:** UML-Klassendiagramm, Constraints vom Typ **Invariants**  
sowie **Initial & derived Values**

# OBJECT CONSTRAINT LANGUAGE - OCL

- OCL = Erweiterung von UML um die Möglichkeit zusätzliche Randbedingungen zu beschreiben

## Vorteile:

- alles abbildbar
- Widerspruchsfrei
- Hoher Generierungsgrad

## Nachteile:

- sehr komplex und detailliert
- schwer erlernbar
- schwer wiederzuverwenden (nicht modellübergreifend)
- schwache Toolunterstützung



# UML-PROFILES

- Profile erweitern UML!

## Zielstellung von UML-Profilen:

UML als Standard derart anzupassen, dass es auch für **neue Anwendungsgebiete** jenseits der Standard-Systemmodellierung passt

### Vorteile:

- bisher ungedeckte Anforderungen werden erfüllbar
- hinreichende Toolunterstützung

### Nachteile:

- erfordert tiefes Wissen über UML
- erfordert Wissen über die Domäne
- Mehraufwand durch Profilerstellung

# DOMAIN SPECIFIC LANGUAGE - DSL

- DSL = eine **Sprache** für eine **bestimmte Domäne**, für einen **bestimmten Personenkreis** (Domänenexperten) und für einen **bestimmten Zweck**

## Vorteile:

- Trennung von Aufgabenbereichen
- Trennung von Funktionalitäten
- Vorteile aus DSLs:
  - einfach erlernbar
  - Geringe Komplexität
  - Hohe Lesbarkeit

## Nachteile:

- Sprachdefinition nötig
- UML-Referenzierung nicht out-of-the-box



# CONCEPT OF PROOF – PROTOTYP

## Grundlage UML-Klassendiagramm

- Eclipse Papyrus = Framework zur Modellierung von UML-Modellen

### Vorteile:

- Open Source
- weite Verbreitung
- Import/Export von UML-konforme Dateiformaten möglich
- Zudem Ecore-konform



# CONCEPT OF PROOF – PROTOTYP

## Language Workbenches:

Meta Programming System ([MPS](#)) vs. [Xtext](#) von Eclipse

**Problem:** UML-Referenzierung nicht out-of-the-box

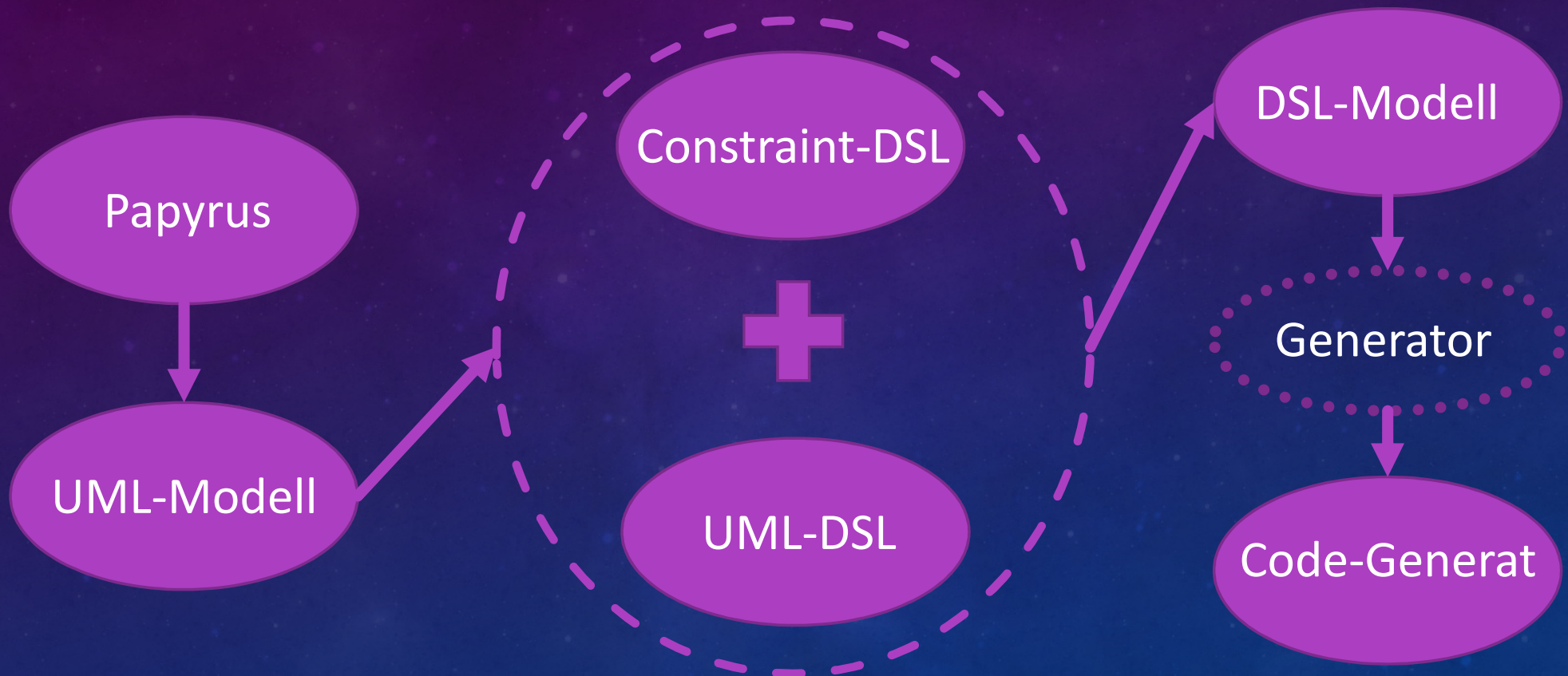
**Lösung:** Projekt von Christian Dietrich, Mitarbeiter von Itemes AG

Projekt bietet eine fertige [UML-DSL](#)

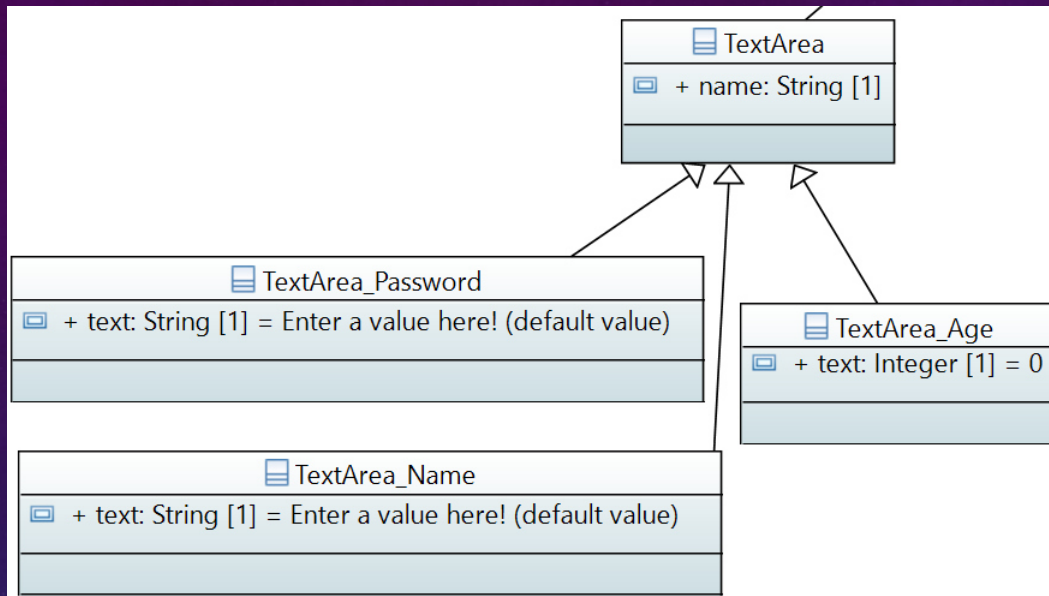
- Umgesetzt mit Xtext
- Ermöglicht Referenzierung auf UML-Modelle



# CONCEPT OF PROOF – PROTOTYP



# CONCEPT OF PROOF – PROTOTYP



```
property Name mapsTo Model.TextArea_Name.text
property Password mapsTo Model.TextArea_Password.text
property Age mapsTo Model.TextArea_Age.text
```

**PropertyEqualityConstraint:** Name "!=" Password

**NumberRestrictionConstraint:** Age ">" 18

A formular	
Name:	Enter a value here! (default value)
Password:	Enter a value here! (default value)
Age:	0
Error: Rule broken: Name != Password	Error: Rule broken: Age > 18



FRAGEN?    VIELEN DANK!

