

# Faktor-IPS

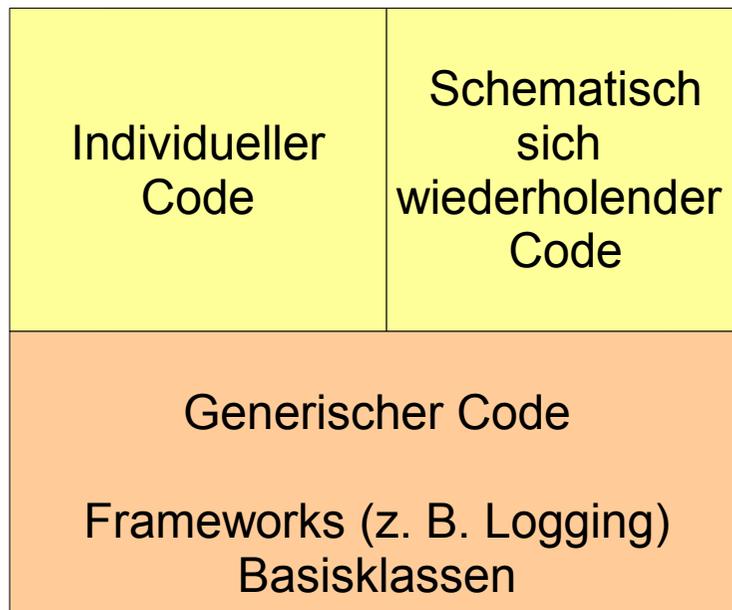
Modellgetriebene Softwareentwicklung mit  
Faktor-IPS

## Faktor-IPS

- Faktor-IPS ist ein Werkzeug zur modellgetriebenen Entwicklung versicherungsfachlicher Systeme
  - Bestandssysteme
  - Außendienstsysteme
  - Schadensysteme
  - Produktserver
- Besonderer Fokus liegt auf der einheitlichen Abbildung des Produktwissens.
- Mit Faktor-IPS werden die fachlichen Modelle und die Produktinformationen verwaltet .

# Modellgetriebene Softwareentwicklung: Motivation

## Referenzanwendung



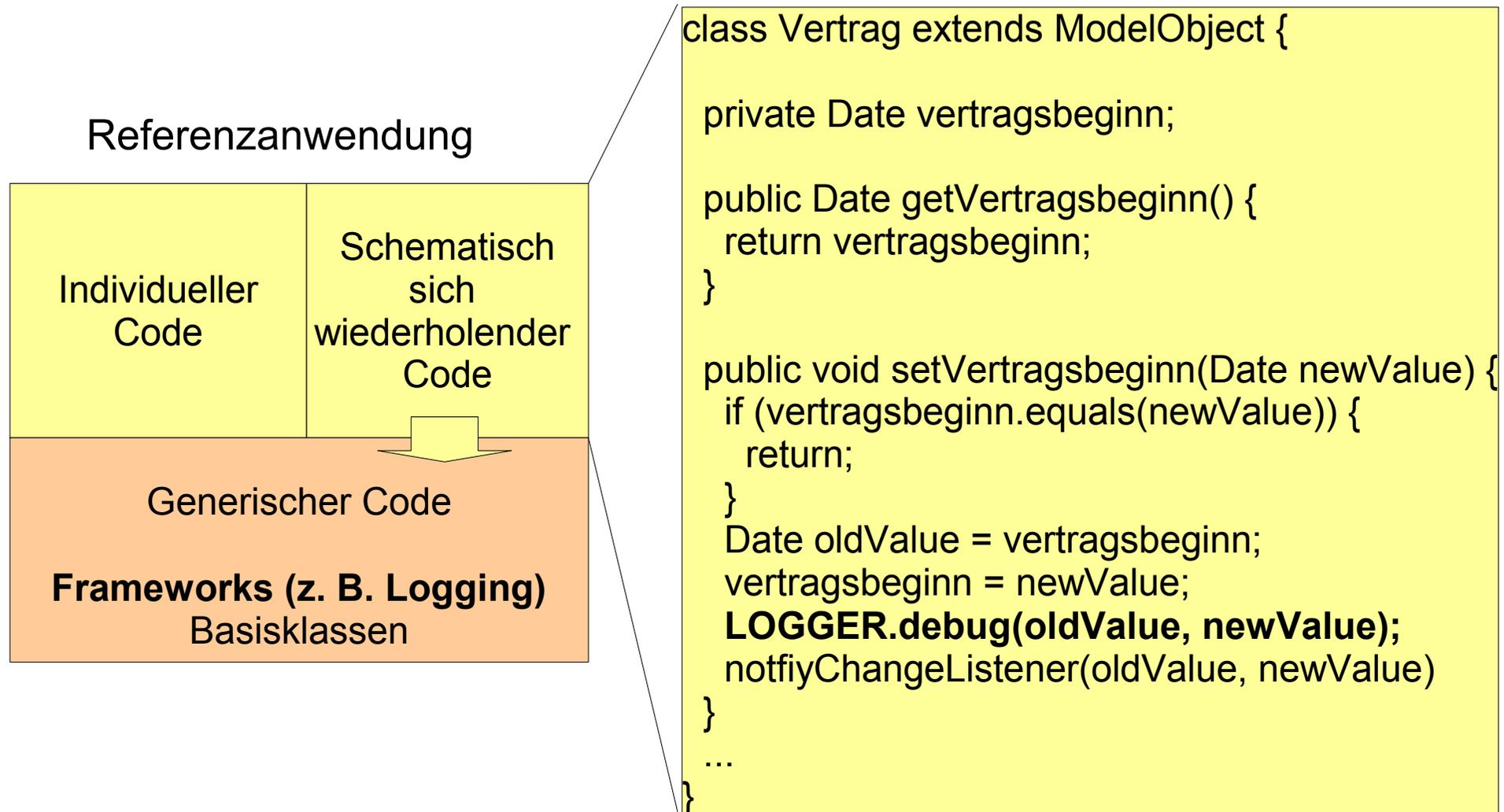
# Modellgetriebene Softwareentwicklung: Motivation

## Referenzanwendung

Individueller Code	Schematisch sich wiederholender Code
Generischer Code Frameworks (z. B. Logging) Basisklassen	

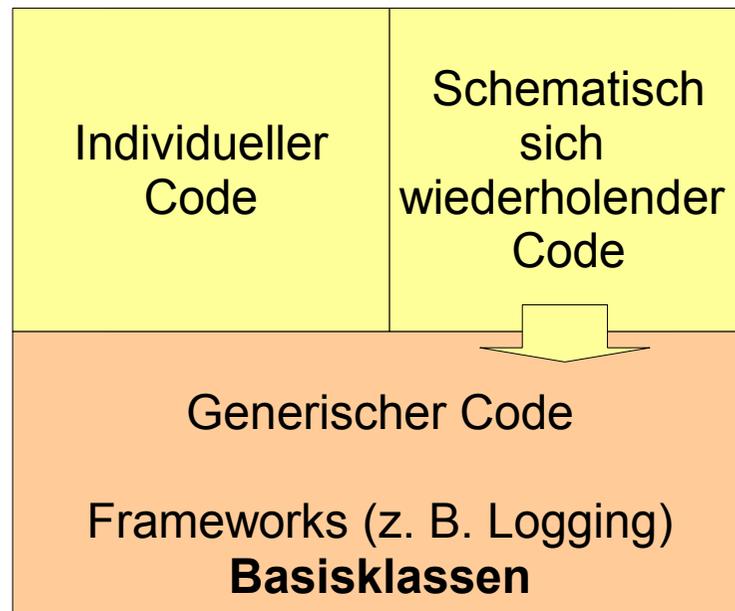
```
class Vertrag extends ModelObject {  
  
    private Date vertragsbeginn;  
  
    public Date getVertragsbeginn() {  
        return vertragsbeginn;  
    }  
  
    public void setVertragsbeginn(Date newValue) {  
        if (vertragsbeginn.equals(newValue)) {  
            return;  
        }  
        Date oldValue = vertragsbeginn;  
        vertragsbeginn = newValue;  
        LOGGER.debug(oldValue, newValue);  
        notifyChangeListener(oldValue, newValue)  
    }  
  
    ...  
}
```

# Modellgetriebene Softwareentwicklung: Motivation



## Modellgetriebene Softwareentwicklung: Motivation

### Referenzanwendung

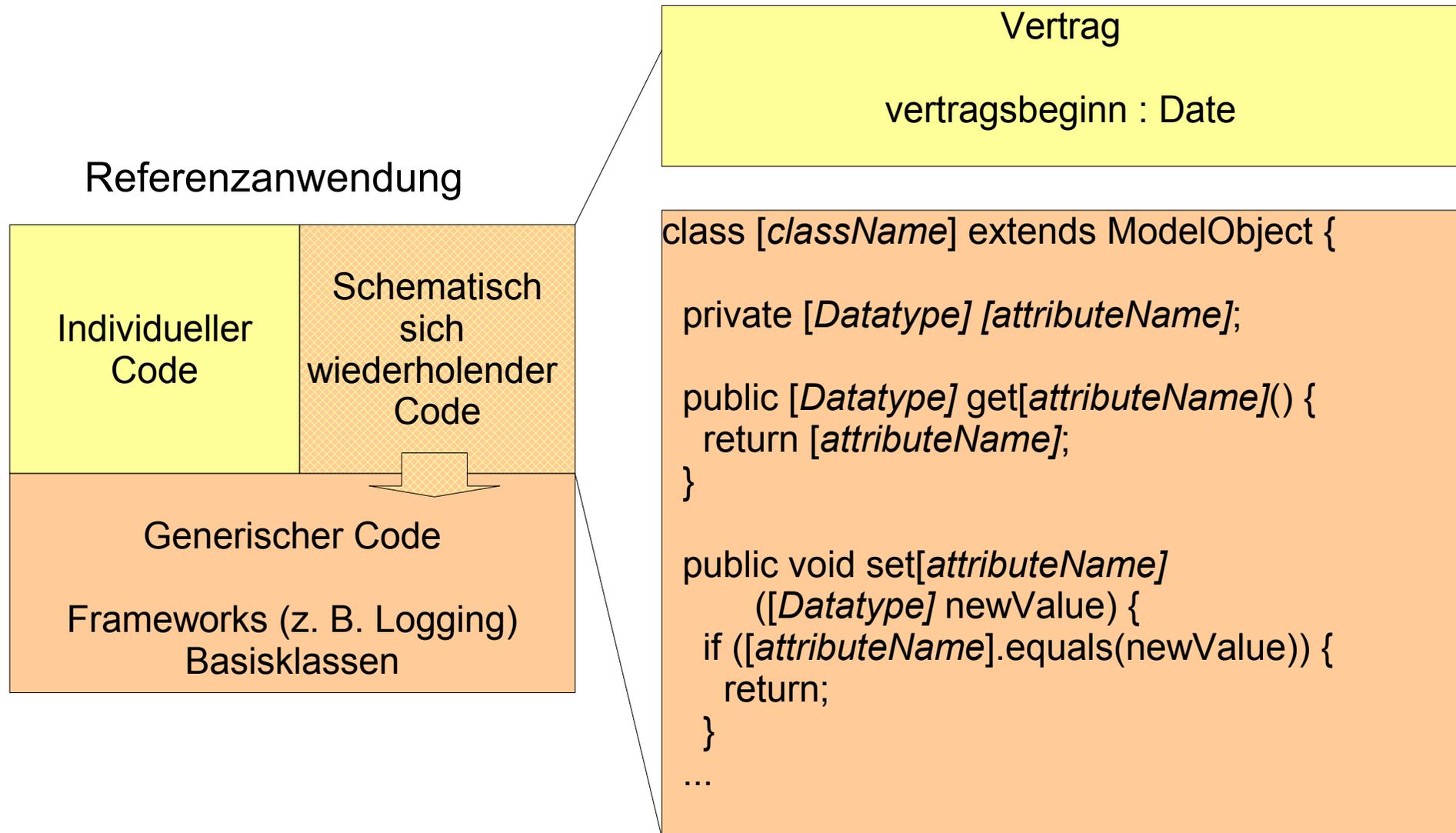


```
class Vertrag extends ModelObject {
    private Date vertragsbeginn;

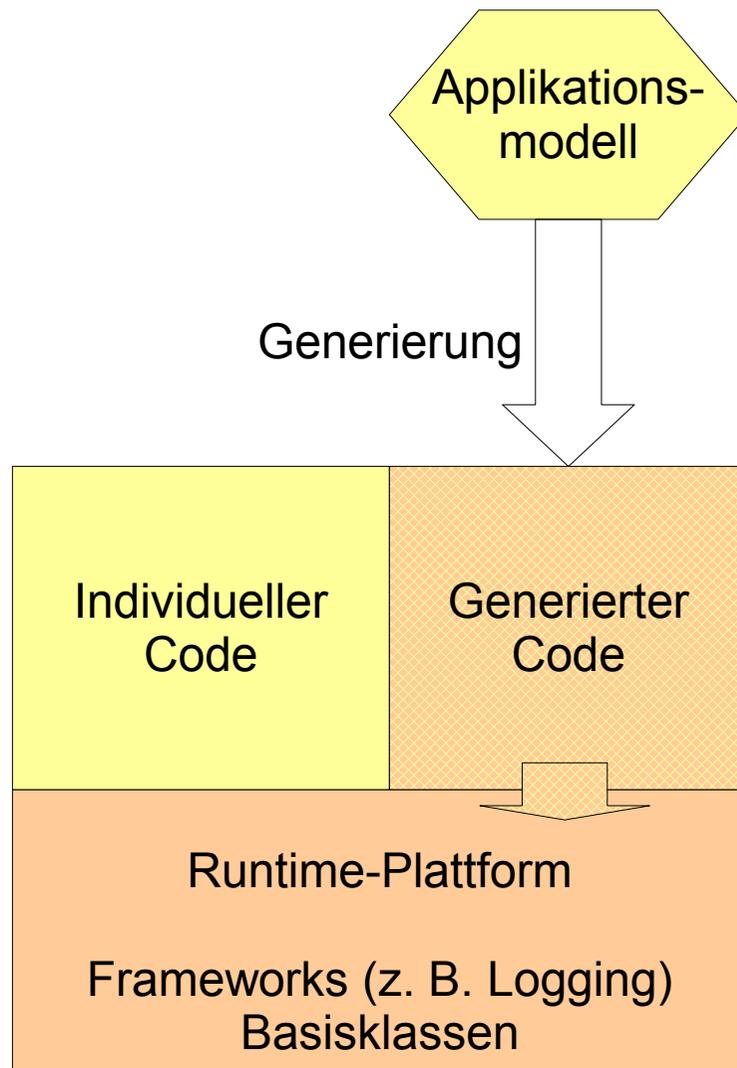
    public Date getVertragsbeginn() {
        return vertragsbeginn;
    }

    public void setVertragsbeginn(Date newValue) {
        if (vertragsbeginn.equals(newValue)) {
            return;
        }
        Date oldValue = vertragsbeginn;
        vertragsbeginn = newValue;
        LOGGER.debug(oldValue, newValue);
        notifyChangeListener(oldValue, newValue)
    }
    ...
}
```

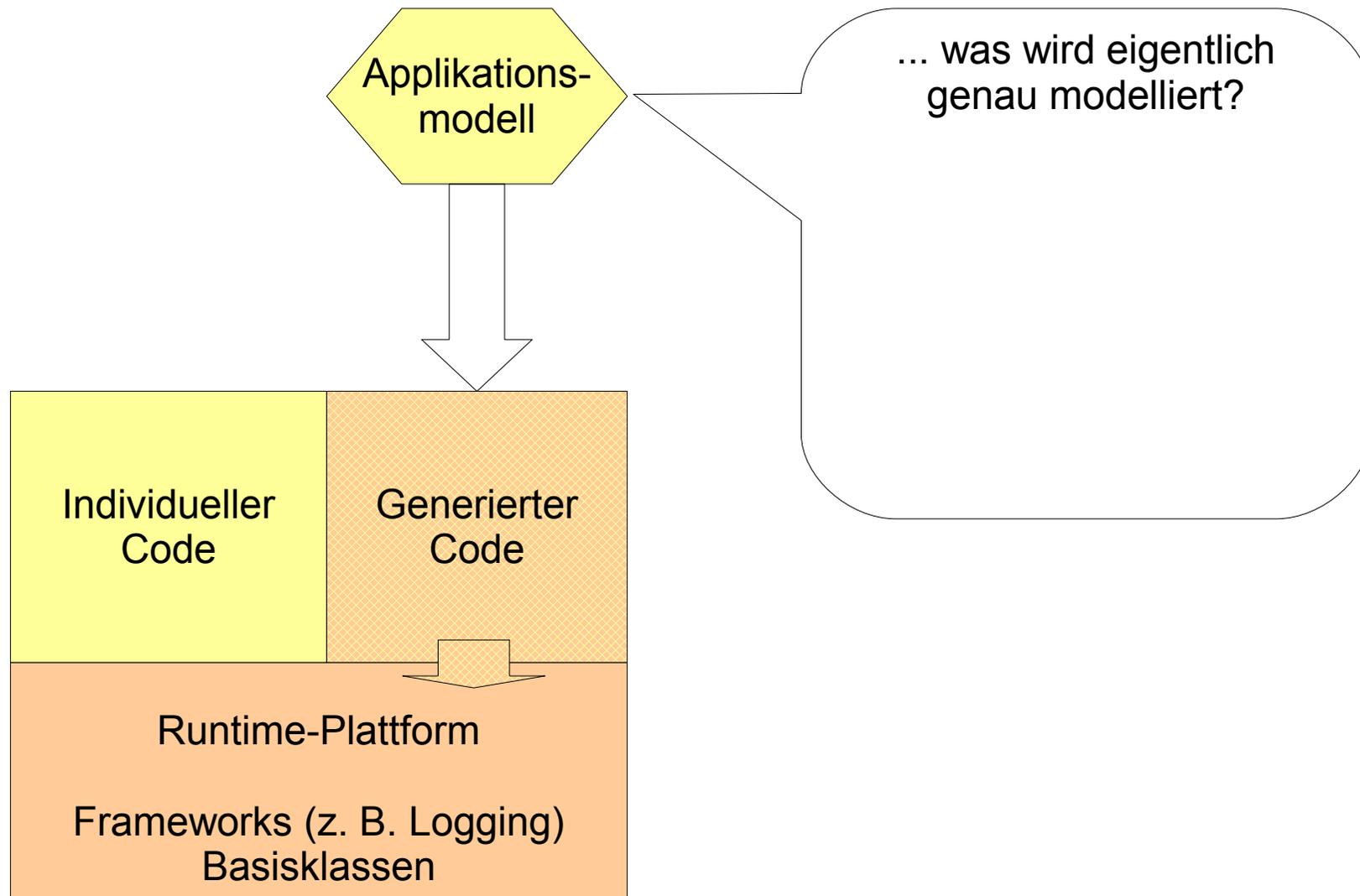
## Modellgetriebene Softwareentwicklung: Motivation



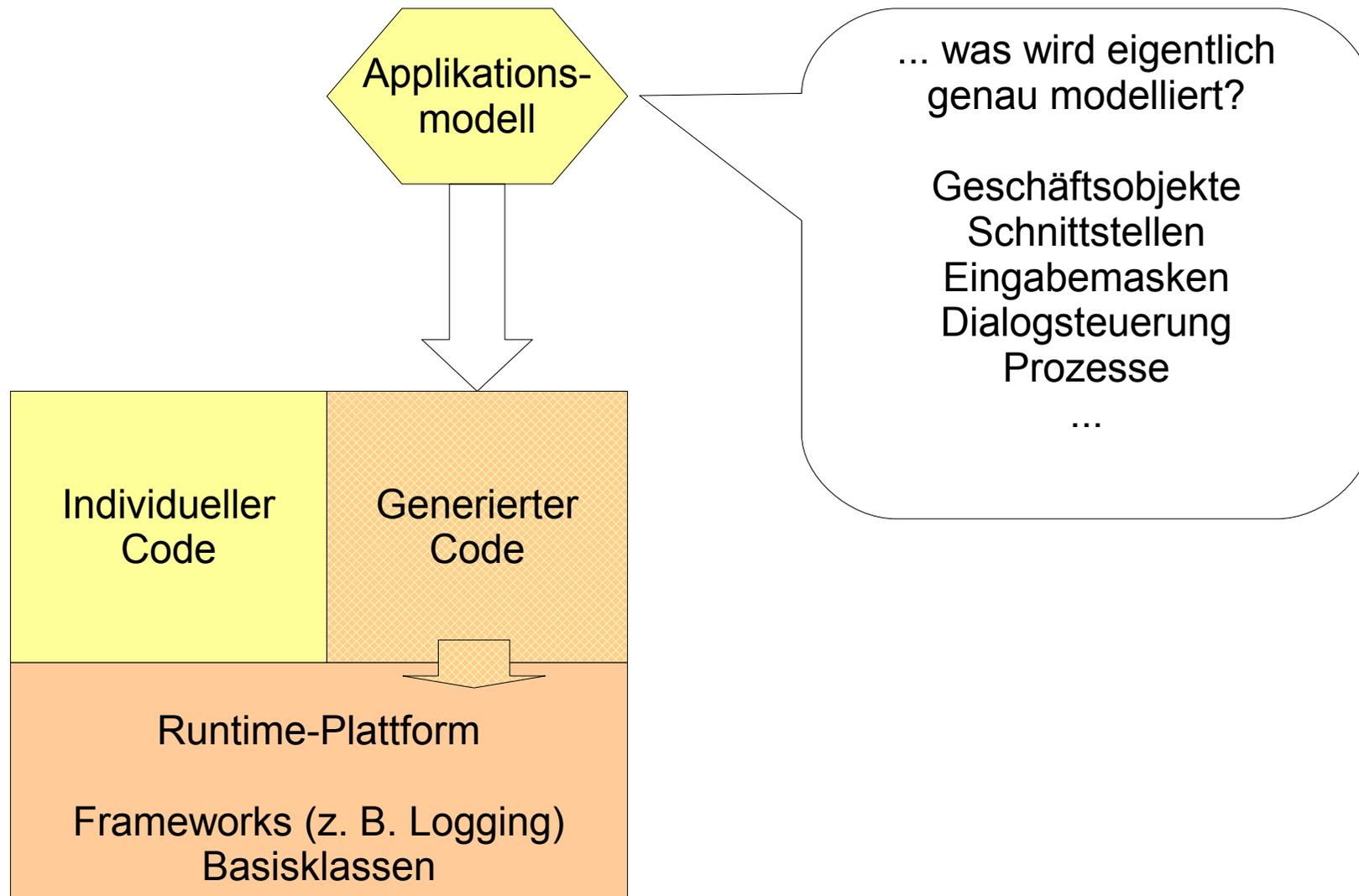
# Modellgetriebene Softwareentwicklung: Motivation



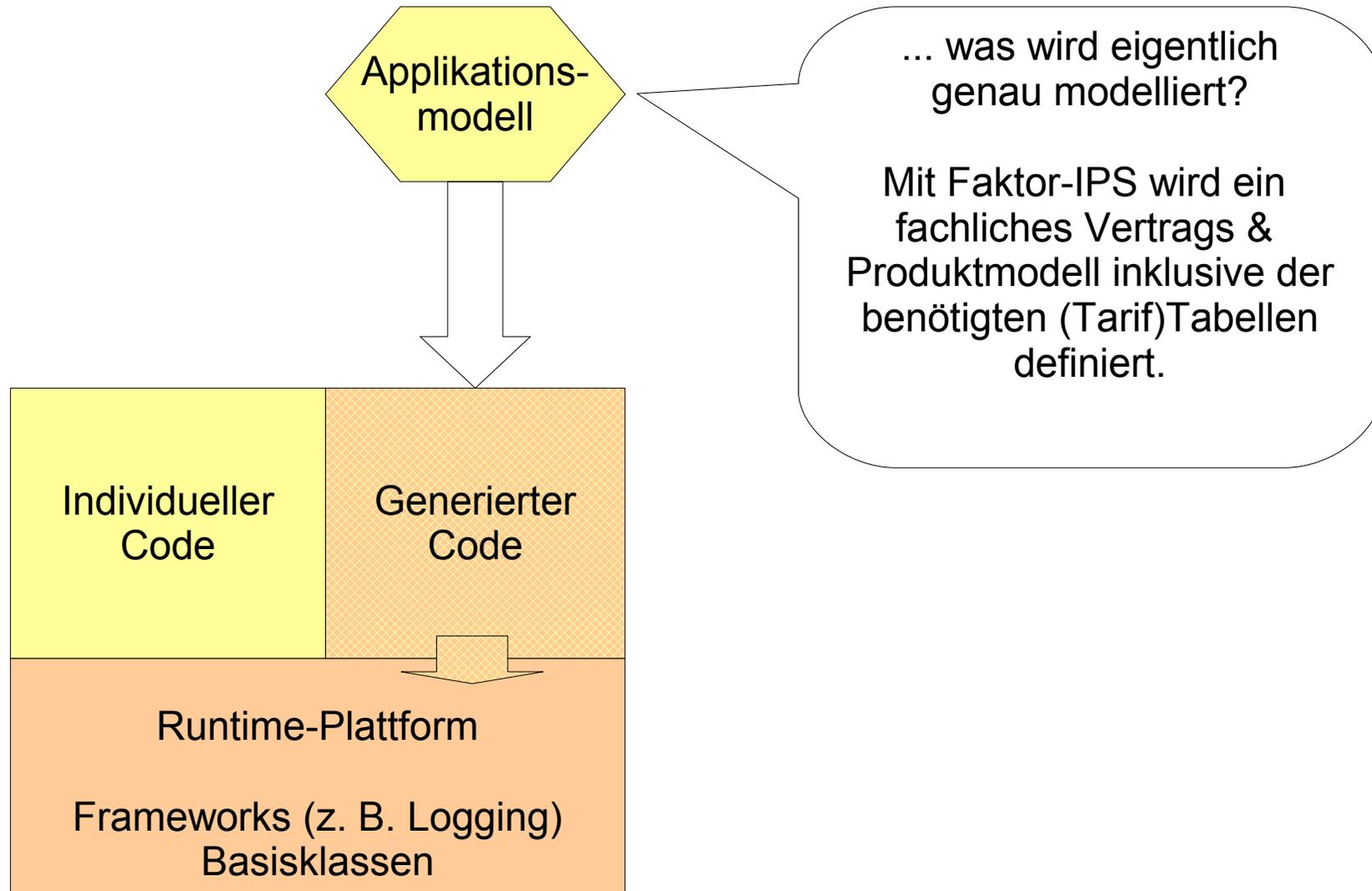
Aber ...



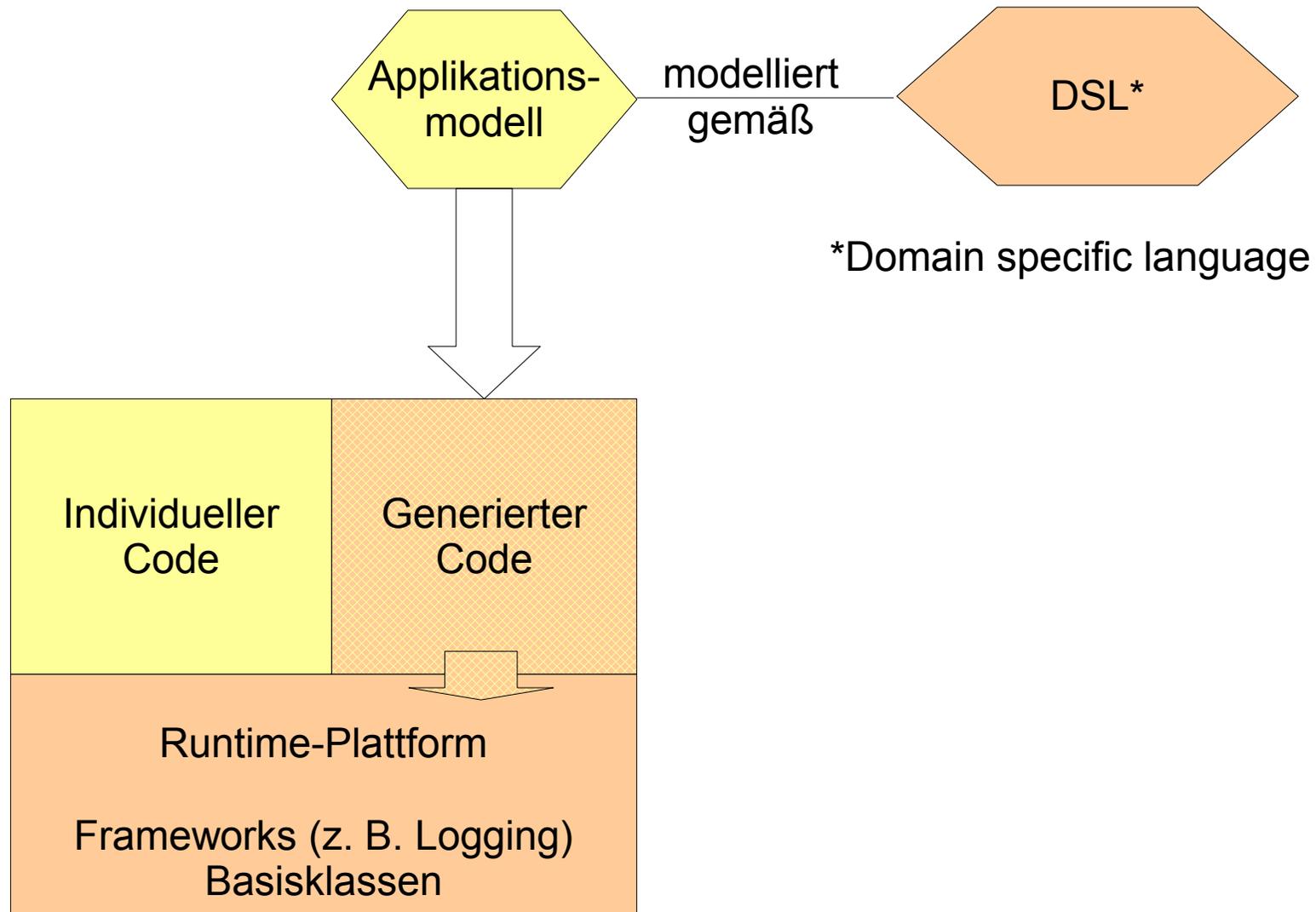
Aber ...



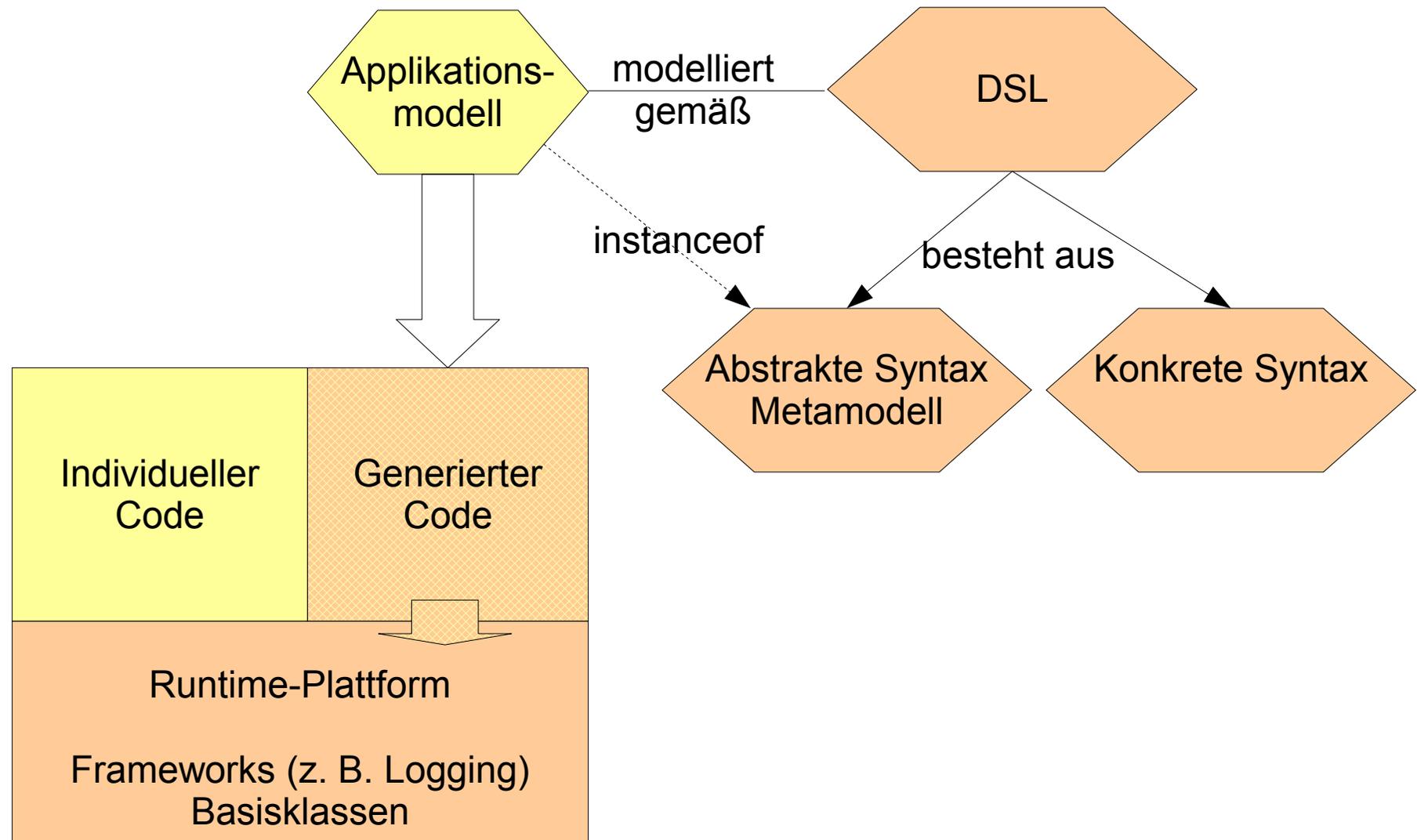
Aber ...



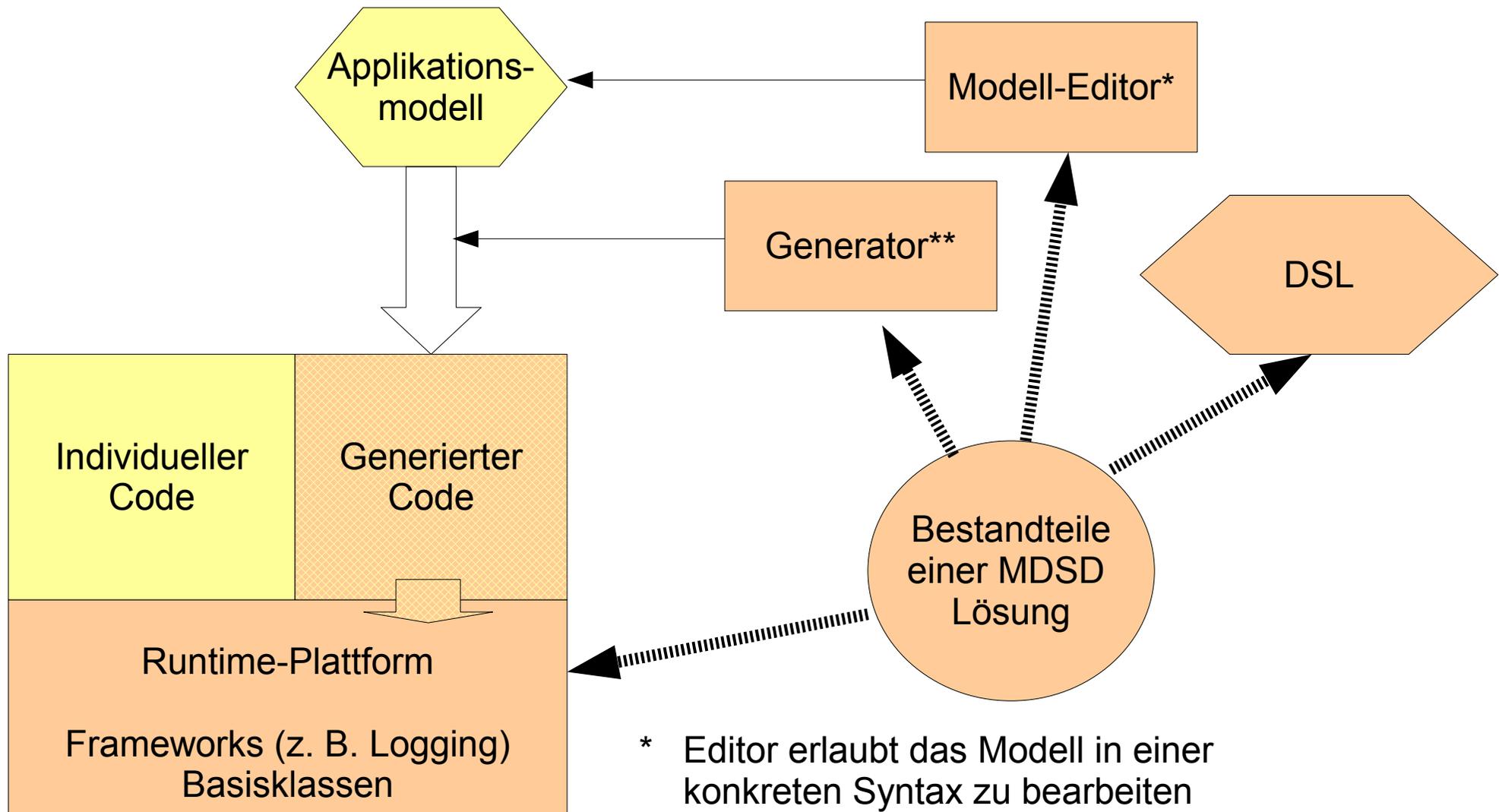
## Domain specific language



## Domain specific language



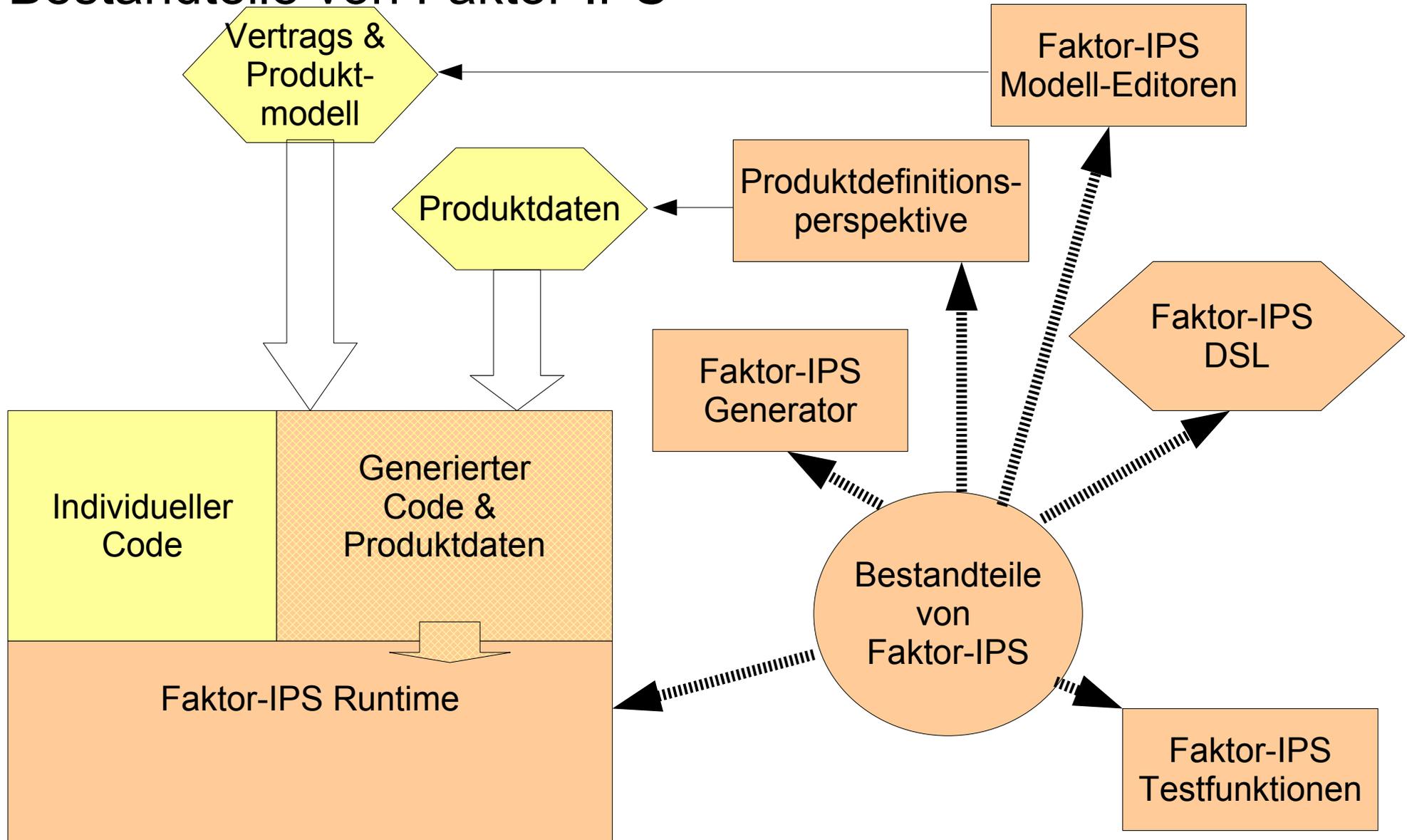
## Bestandteile einer MDSD Lösung



\* Editor erlaubt das Modell in einer konkreten Syntax zu bearbeiten

\*\* Generator arbeitet auf dem Metamodell

## Bestandteile von Faktor-IPS

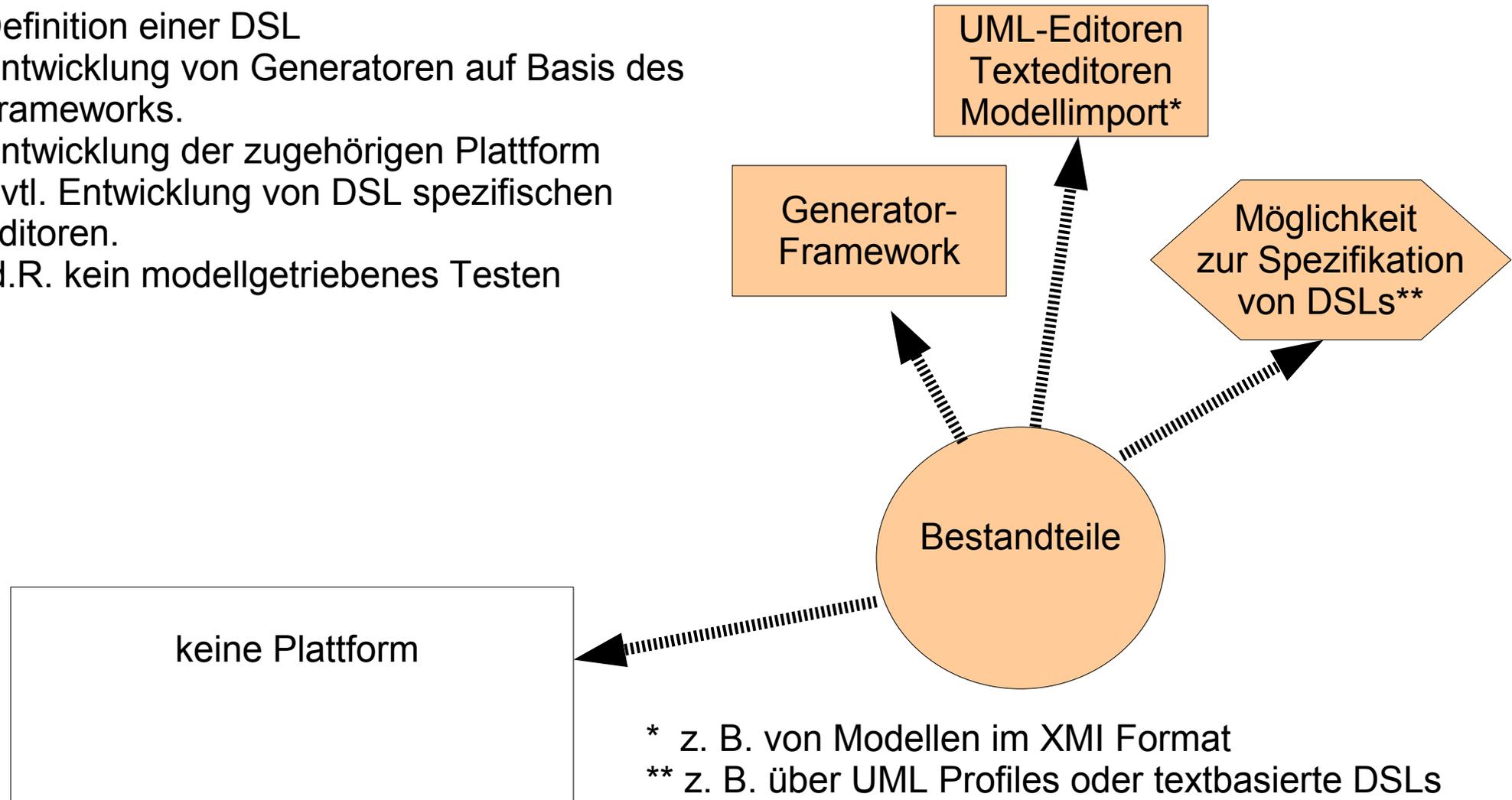


## Faktor-IPS: DSL

- Geschäftsobjekte Vertrag & Produkt
  - Teilmenge von MOF, UML für Klassen
  - Fachlicher Zusammenhang Vertrag / Produkt
- Produktbausteine
  - inkl. Abbildung Produktänderungen im Zeitablauf
- Formelsprache
- Tabellen
  - inkl. Zusammenhang zu Produktklassen / Bausteinen
- Aufzählungstypen
- Testfalltypen & Testfälle

## Entwicklung einer MDSD Lösung mit generellen MDSD Werkzeugen

- Definition einer DSL
- Entwicklung von Generatoren auf Basis des Frameworks.
- Entwicklung der zugehörigen Plattform
- Evtl. Entwicklung von DSL spezifischen Editoren.
- I.d.R. kein modellgetriebenes Testen



## Leistungsumfang Faktor-IPS Runtime

- Effizienter Zugriff auf Produktdaten über ein Repository
- Ablauffähig auf allen Plattformen auf denen eine JVM > 1.4 zur Verfügung steht.
  - Windows
  - Unix, Linux
  - OS / 390, Z/OS
- Leicht integrierbar
  - in JEE Architekturen (EJBs)
  - in JSE Architekturen
  - über Webservice
  - über JNI
- Keine Abhängigkeit zu andern Bibliotheken