

# HyperAdapt: Aspektorientierte Adaption für das Web

## Motivation

- Nutzer greifen von verschiedenen Orten mit unterschiedlichen Geräten und individuellen Zielen auf Webanwendungen zu
- Entwicklungsaufwand eines sich daran anpassenden (adaptiven) Systems steigt exponentiell mit Anzahl der Adaptionziele (z.B. Ortsanpassung, Übersetzung, etc.)

## Gewinnung neuer Erkenntnisse über Aspektinteraktionen

- Interaktionen zwischen Aspekten können ungewollt, aber ebenso Ergebnis einer bewussten Entscheidung des Autors sein
- Automatische Konfliktlösungsstrategien unter Nutzung der Semantik von Aspekten
- Semiautomatische Auflösung von Konflikten mit Einbeziehung des Autors

## Konzept

### Erweiterung invasiver Softwarekomposition (ISC) auf XML-Dokumente

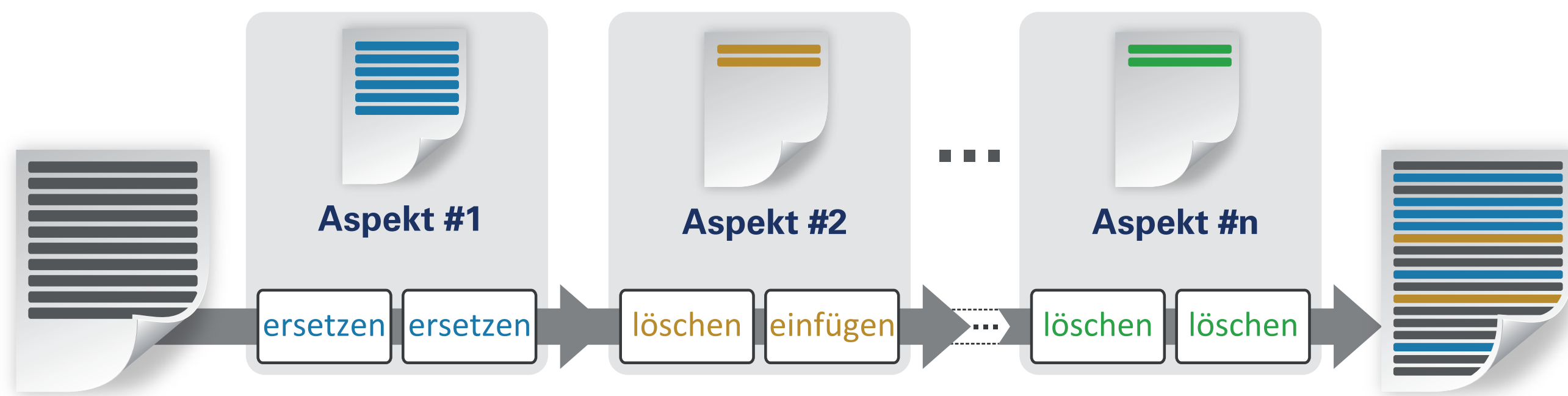
- ISC als Basis für typischeres Weben und Schnittstellendefinition zwischen XML-Komponenten
- Entwicklung eines Mechanismus zur Deklaration von Erweiterungs- und Variationspunkten in XML-Dokumenten
- Entwicklung einer Pipeline zum typischeren Verweben von XML-Fragmenten
- Aufgliederung von Aspekten in atomare Operationen

### Behandlung von Aspektinteraktionen

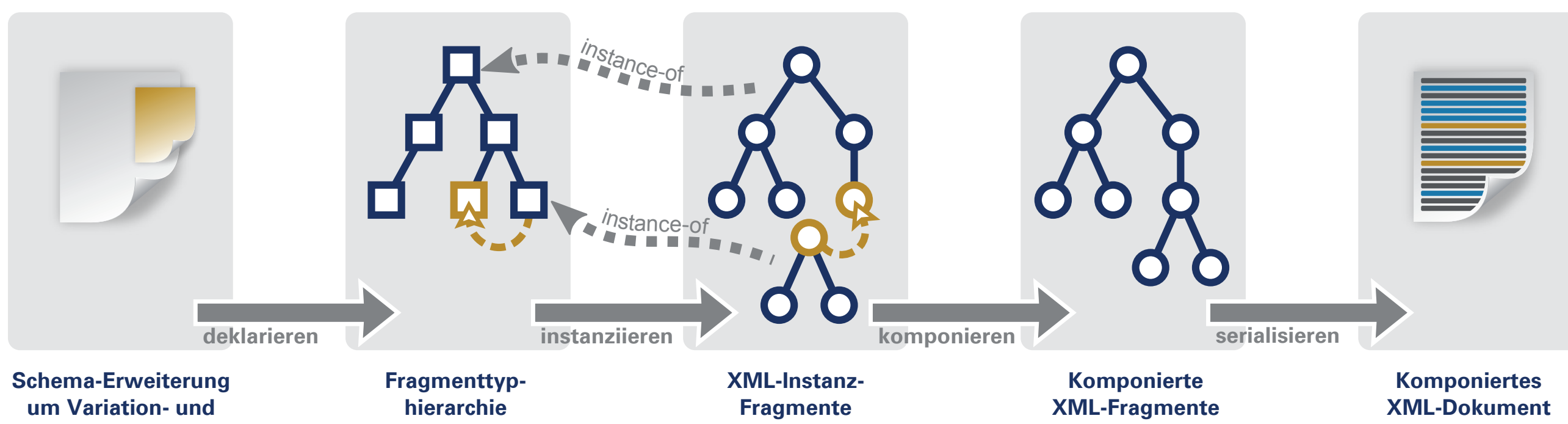
- Klassifizierung von Adaptionstechniken anhand ihres Konfliktpotentials
- Erweiterung dieser Adaptionstechniken um semantische Beschreibung zur vereinfachten Analyse auf Konflikte
- Auflösungsstrategien für Kombinationen bestimmter Adaptionsaspekte, anschließend Erweiterung auf generelle Aspekte

### Einführung von Beschränkungen auf Aspekte und Kernanwendungen

- Sperrung von Teilen einer Webanwendung gegen die Erweiterung durch Aspekte
- Knüpfung von Bedingungen an die Reihenfolge der Ausführung von Aspekten (z.B. Vor- und Nachbedingungen)



▲ Einzuwebende Aspekte werden auf atomare Operationen heruntergebrochen



▲ Durch Schema-Erweiterungen können Dokumente mit Variations-Elementen erweitert und komponiert werden

- Konflikte zwischen Adaptionenzielen können auftreten, deren Lösung bisher nur aufwändig und manuell durch den Autor möglich ist
- Lösungsidee: Aspektorientierte Programmierung (AOP) nutzt das Prinzip der Auftrennung eines Programms nach Belangen und bietet verschiedene, spezialisierte Sichten auf die Anwendung

## Entwicklung eines generischen Ansatzes für Aspekte in XML

- Adaption als „Add-on“ für vorhandene Webanwendungen oder andere Einsatzfelder von XML

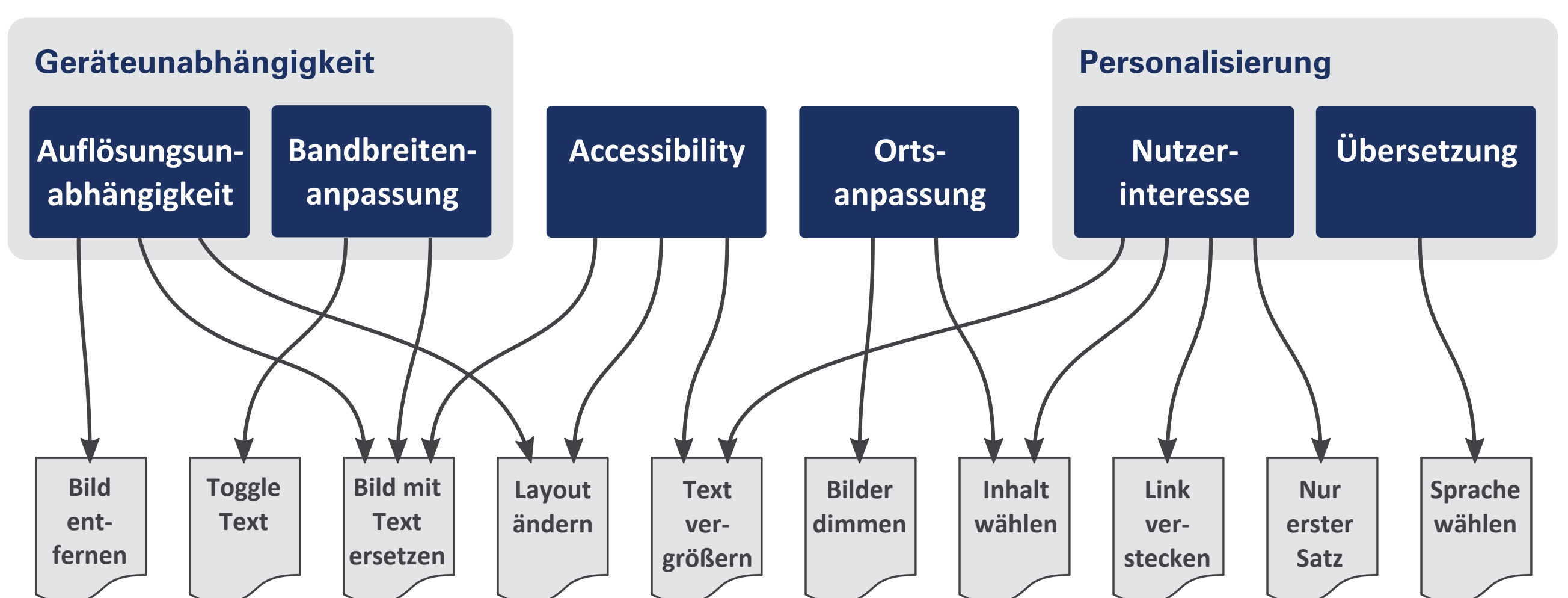
## Vision und Forschungsziele

### Kapselung von Adaptivität in Aspekten

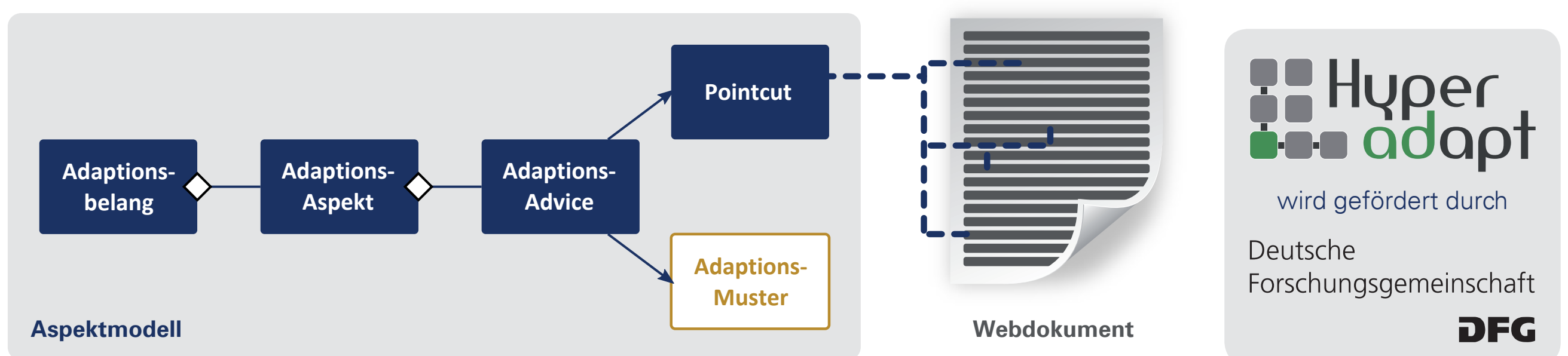
- Einsatz von AOP zur Modellierung von Adaption in Webanwendungen
- Organisation von Aspekten nach Zielen d. Adaption (Internationalisierung, Geräteunabhängigkeit, Standortanpassung, etc.)

### Sicheres Weben mehrerer Aspekte

- Gewährleistung der Typsicherheit („Passgenauigkeit“) von Aspekten
- Aufstellen von Verträgen zwischen Aspekten und der Anwendung
- Einsatz verschiedener Konfliktlösungsstrategien



▲ Adaptionsbelange werden mit Hilfe typischer, wiederkehrender Adaptionsmuster umgesetzt



▲ Aspekte können Adaptionsmuster nutzen und werden mittels Pointcuts auf ein XML-Dokument abgebildet

