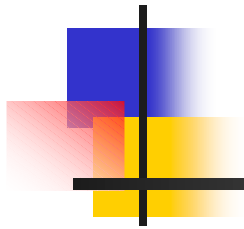


# Modeling Web Services Variability with Feature Diagrams



**Silva Robak**  
Uniwersytet Zielonogórski

Bogdan Franczyk  
Universität Leipzig

WS-RSD'02 Net.Object.Days Erfurt, October 2002



# Modeling Web Services Variability with Feature Diagrams

---





- ✍ Web Services (WS) vs. components
- ✍ Why feature diagrams for WS?
- ✍ Feature models:
  - ✍ Feature notion, feature diagram, used notation
- ✍ Examples:
  - ✍ WS communication technologies
  - ✍ WS models: interface, presentation, security
  - ✍ WS consumer: presentation model
- ✍ Conclusion and future work



# WS vs. Components

---

## Component






-  Provides functionality through well defined standards
-  Contains a full declaration of its static dependencies
-  Is equipped with explicit configuration mechanisms (required interfaces as opposed to the more traditional provided interfaces)
-  Is equipped with version identification



# WS vs. Components

---

## WS

-  Larger level of granularity
-  Existence of appropriate services allowing finding and discovering WSs
-  Service-oriented architecture containing 3 roles: producer, consumer, provider
-  Dynamic nature (parts evolve separately)
-  Run-time environment based on high performance application server or message broker



# Why Feature Diagrams for Web Services?

---





- Generic description of a range of diverse systems
  - Feature diagram and its instances
    - Commonality and variability of some WS aspects
  - Description transparency




# Feature Model

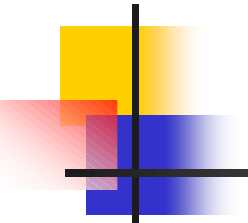
---

## FODA- feature model:

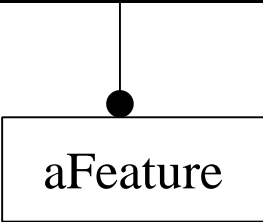
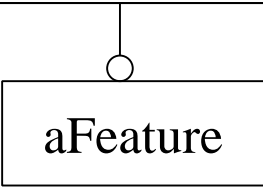
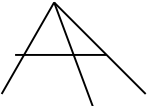

-  Feature diagram
-  Composition rules
-  Issues and decisions
-  System feature catalog.

 Feature notion: *A feature* is a visible characteristic of concept (e.g. system, component, etc.), which is used to describe and distinguish different instances of the concept

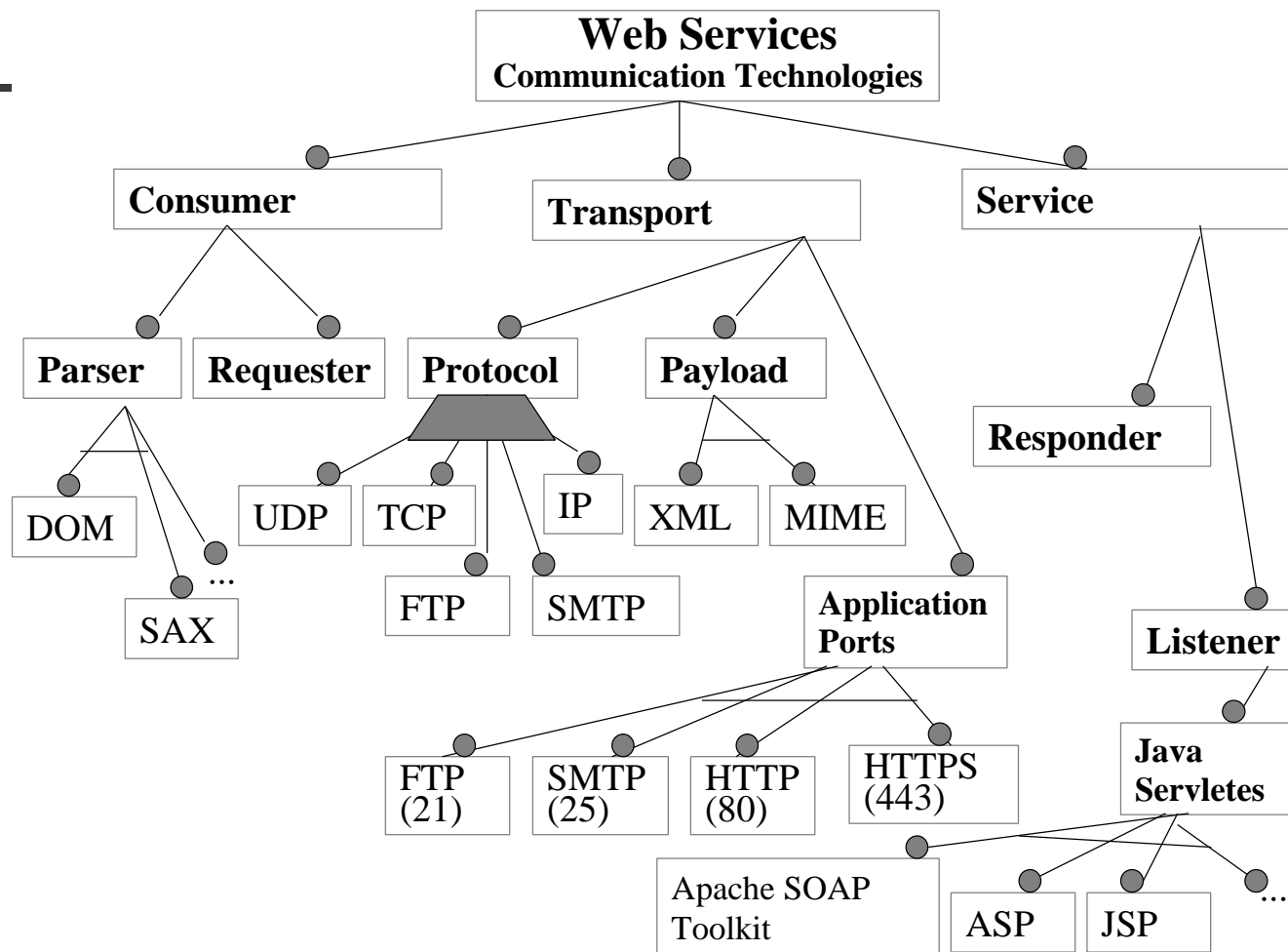
# Feature Diagram

- 
- ✎ Feature diagram forms a tree (graphical AND/OR hierarchy of features) and captures the relationships among the features
    - ✎ Concept node (root), features, sub-features
    - ✎ Feature types: mandatory, optional; alternative, or-features

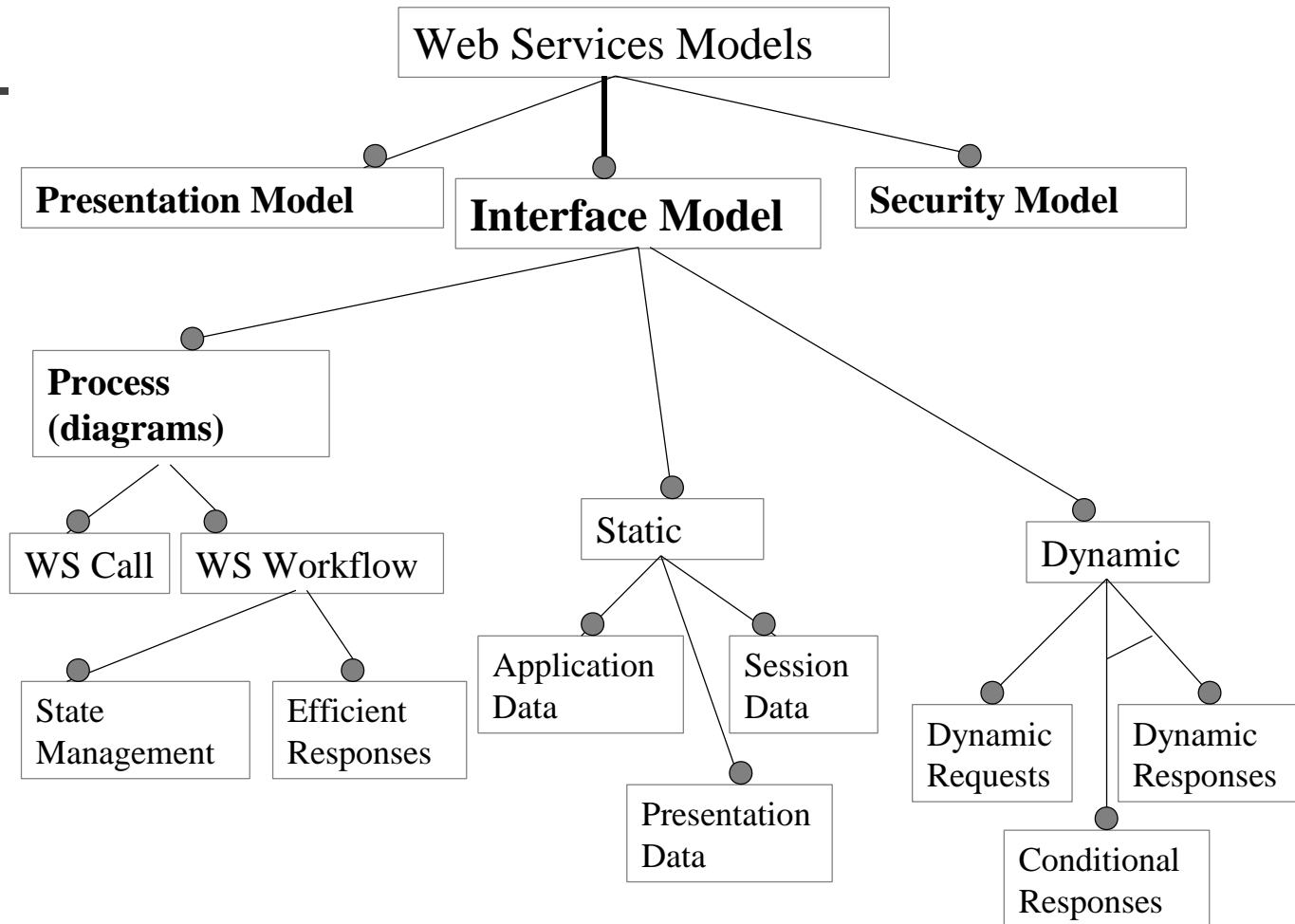
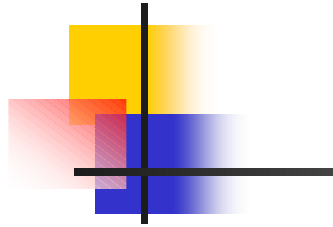
# Feature Diagram Notation in Generative Programming [CE00]

Symbol	Feature Type	Meaning
	mandatory	All included
	optional	Option (may be included or not)
	alternative	XOR-specialization
	Or-feature	OR-specialization

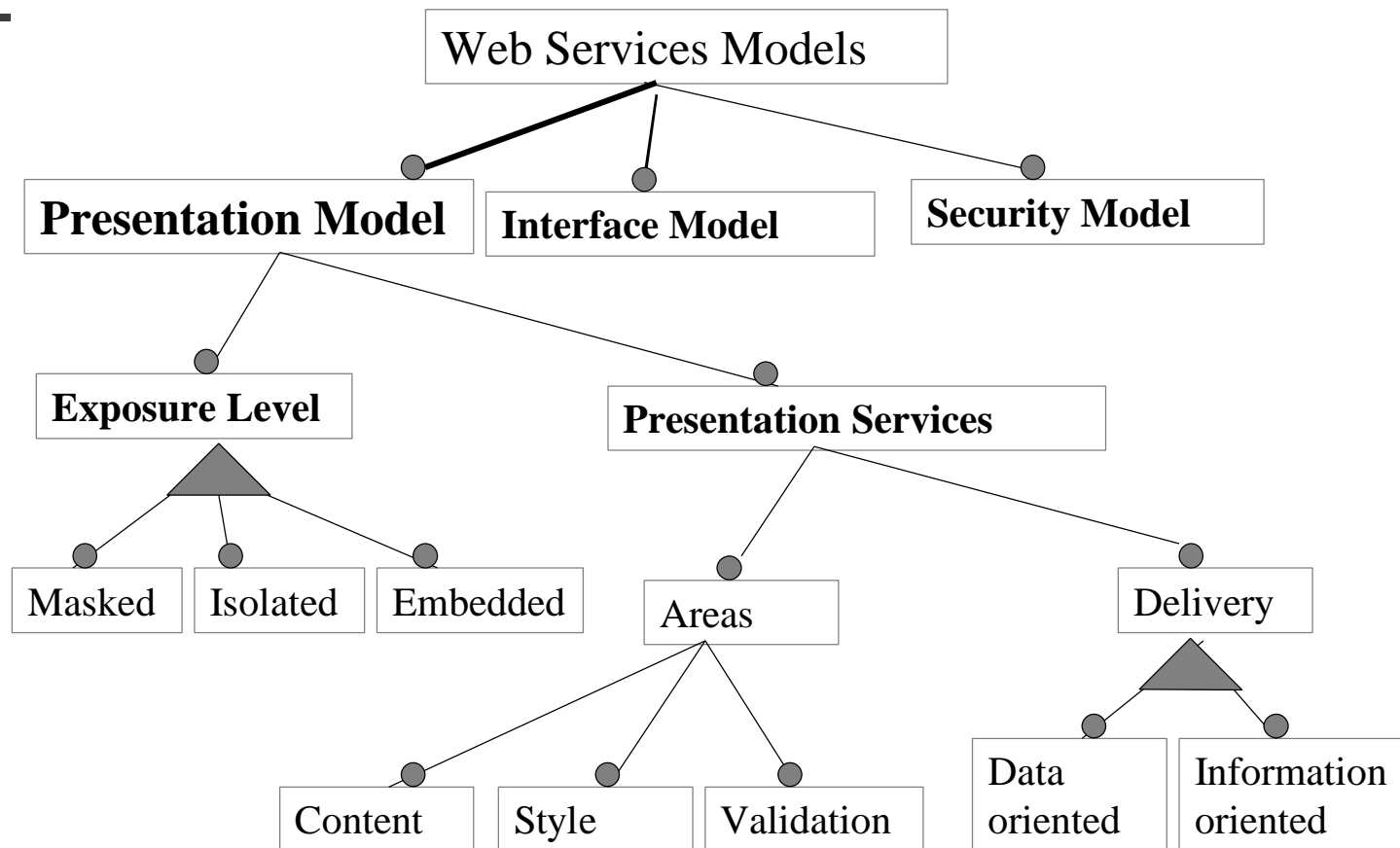
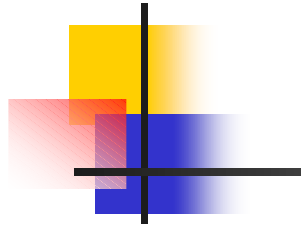
# WS Communication Technologies



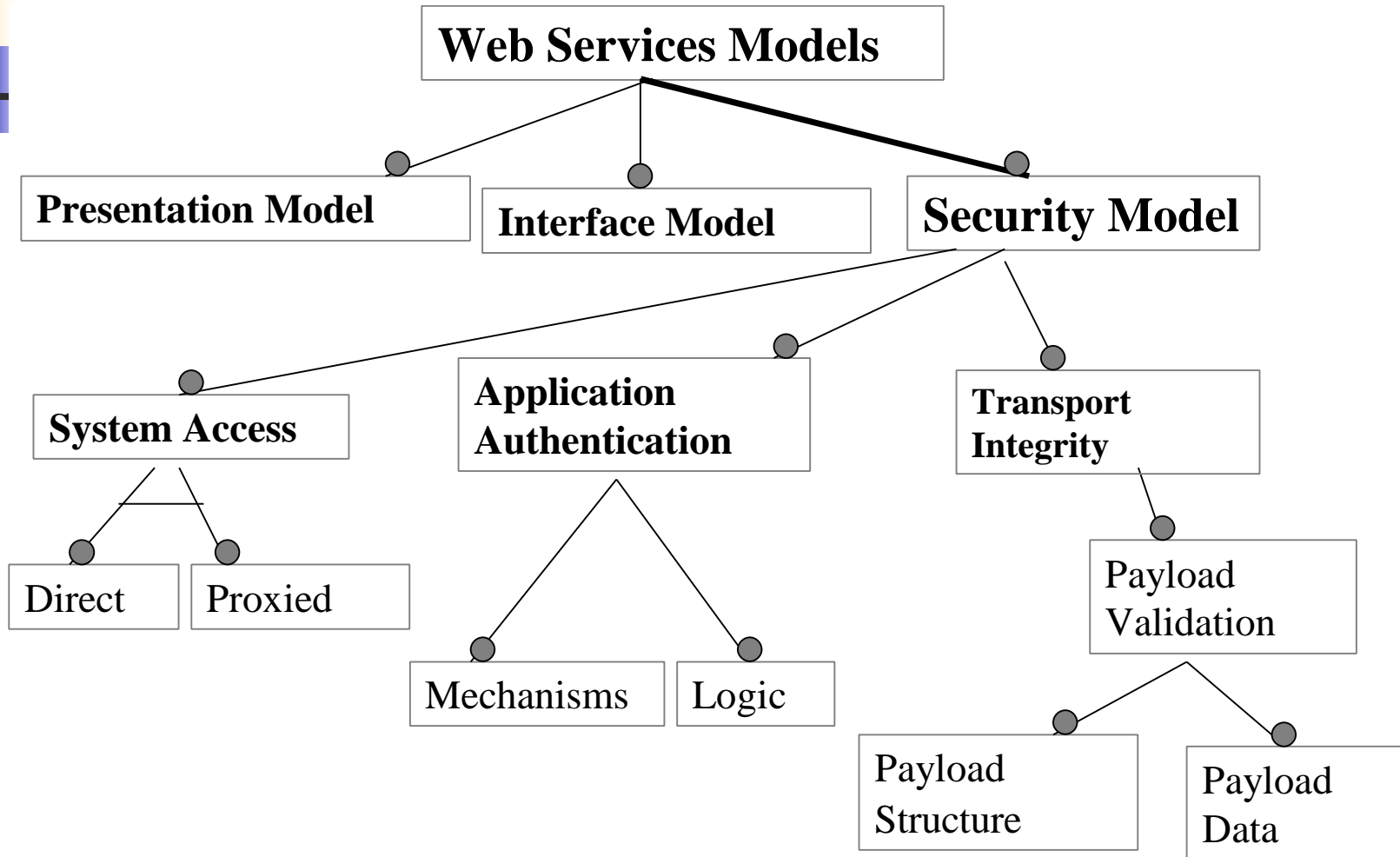
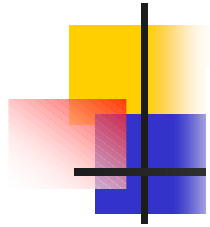
# WS Models – Interface Model



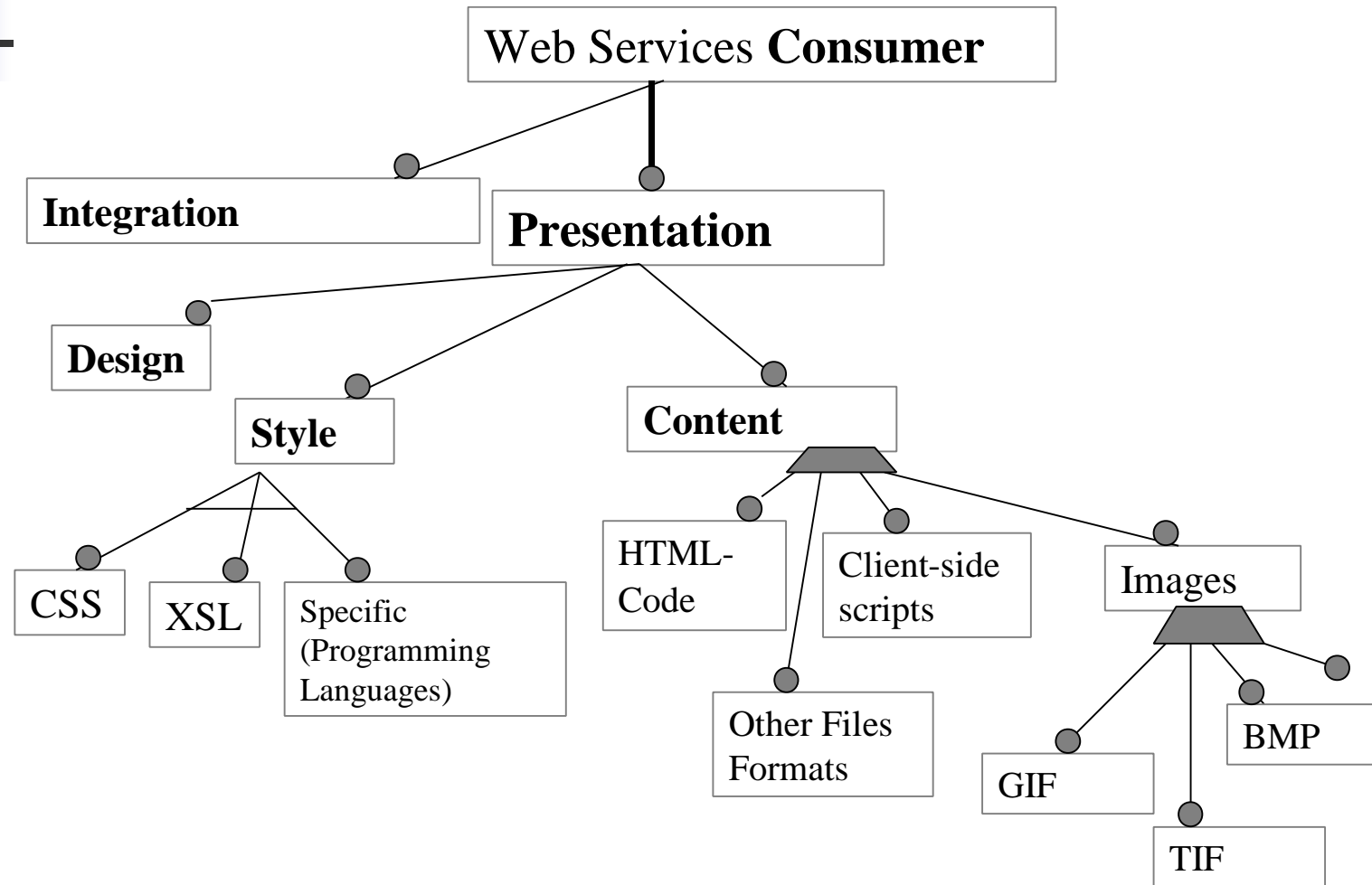
# WS Models – Presentation Model



# WS Models – Security Model



# WS Consumer – Presentation Model





# Conclusion

---

- WS are software components on Web-level
  - Services for Web, as also components in complex workflows
- Feature diagrams may be used for description of WS variability and dynamically instantiated for individual consumers profiles

## **Future work**

- Feature interactions problems (dependencies and traceability) as orthogonal problems to WS

# Modeling Web Services Variability with Feature Diagrams



---

Thank you!