Separating Semantics and Implementation

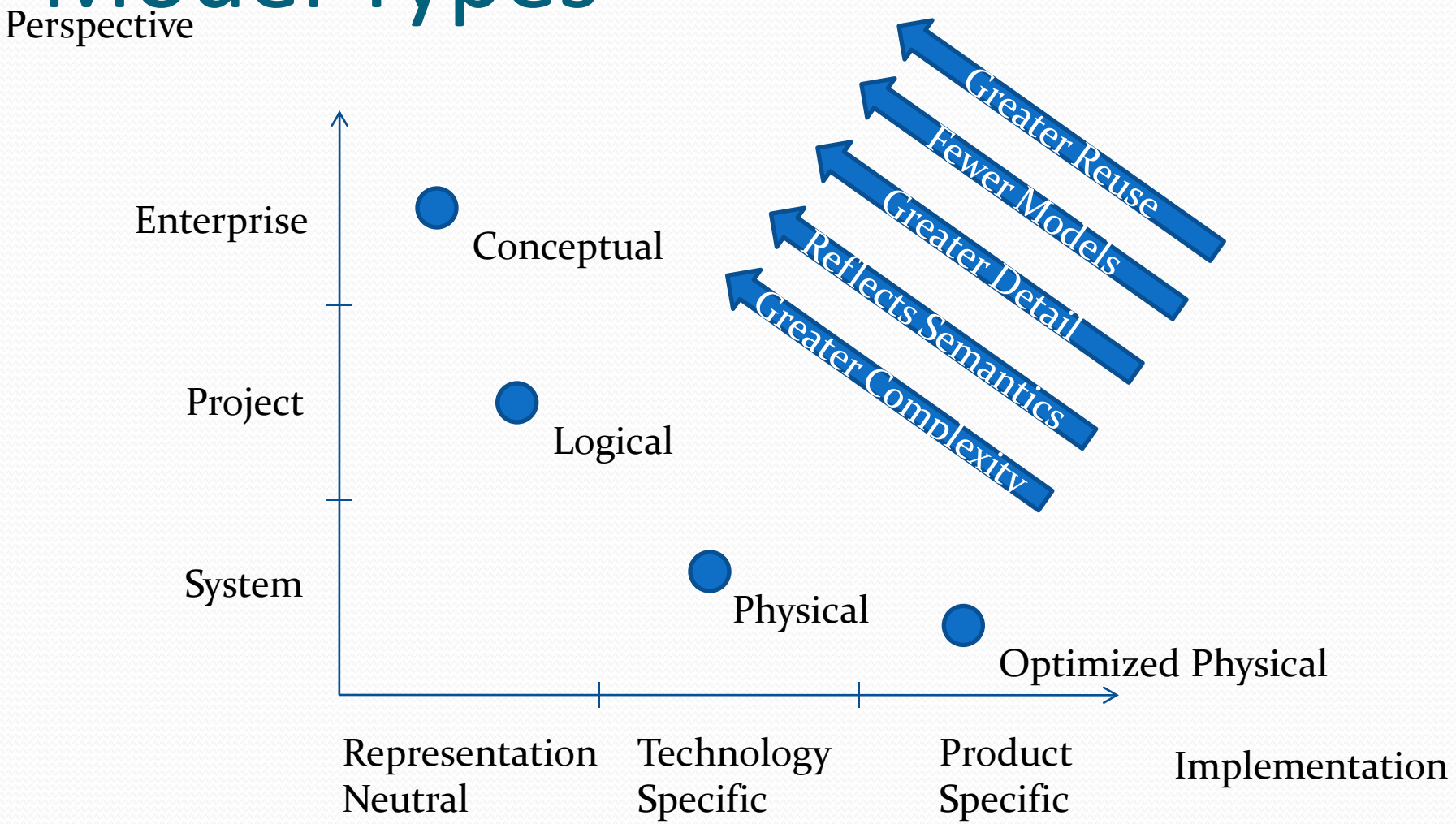
From a Single Ontologically Sound Conceptual Model to Multiple Physical Schema Languages

Bruce T. Bauman, U.S. DoD

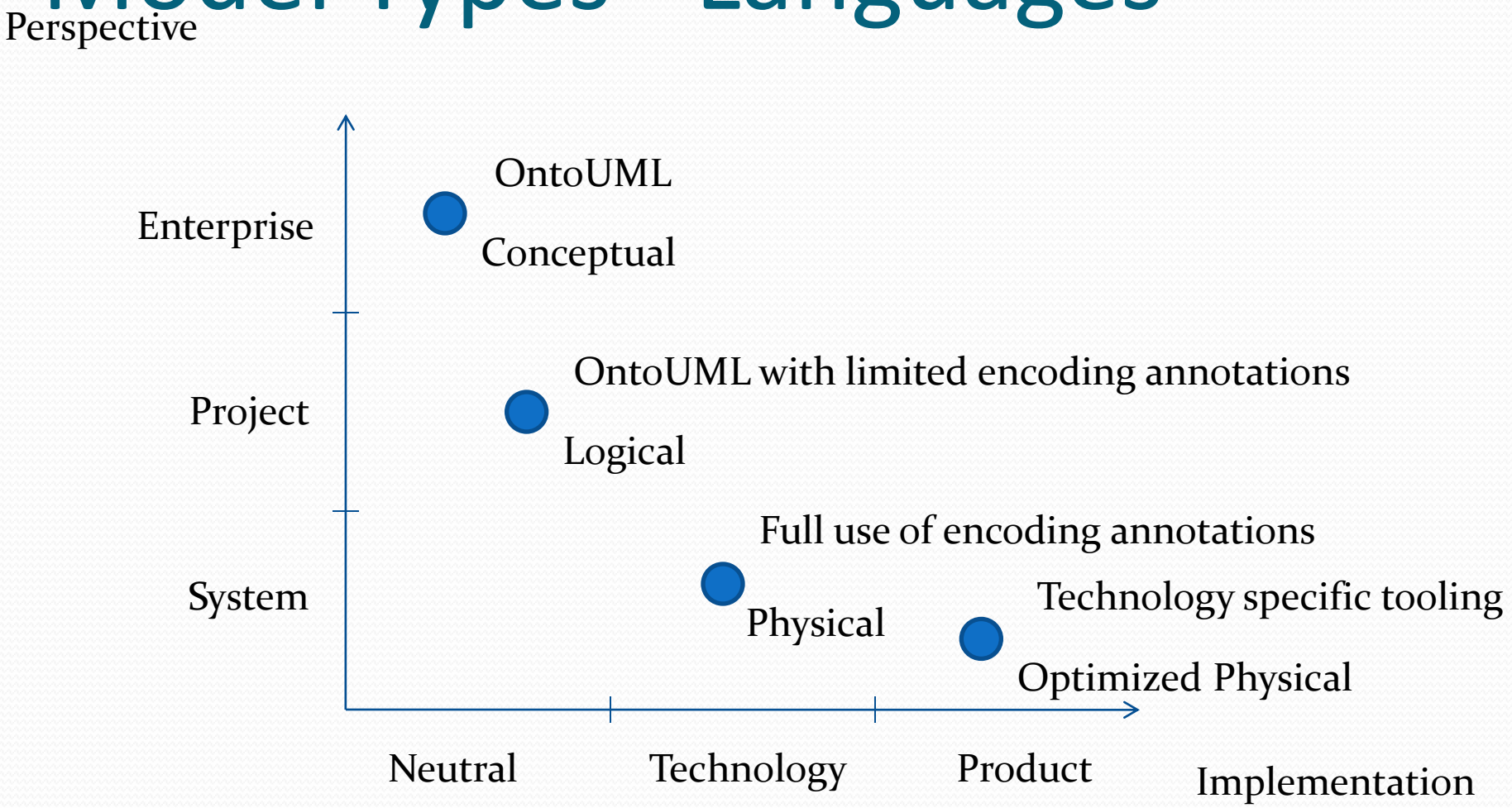
Agenda

- Motivation
- Challenge
- OntoUML
- Compiling
  - RDFS
  - XSD
  - SQL
- Summary
- Further Work
- Conclusion

Model Types


Perspective
Enterprise
Project
System
Conceptual
Logical
Physical
Optimized Physical
Greater Reuse
Fewer Models
Greater Detail
Reflects Semantics
Greater Complexity
Representation Neutral
Technology Specific
Product Specific
Implementation

Model Types - Languages



Why existing languages don't cut it

- Languages pitched at analysis and design (e.g., UML, ERD)
  - Optimized for design targeting specific technologies
  - Don't have a well defined semantic mapping
- Implementation languages (e.g., XML Schema, DDL, OWL, RDFS)
  - Clearly they have made implementation trades.
- Natural Language
  - Not precise enough.

Unified Foundational Ontology (UFO)

- Created by Giancarlo Guizzardi
- For Conceptual / Analysis modeling
- Ontologically Based
- UML class diagram notation

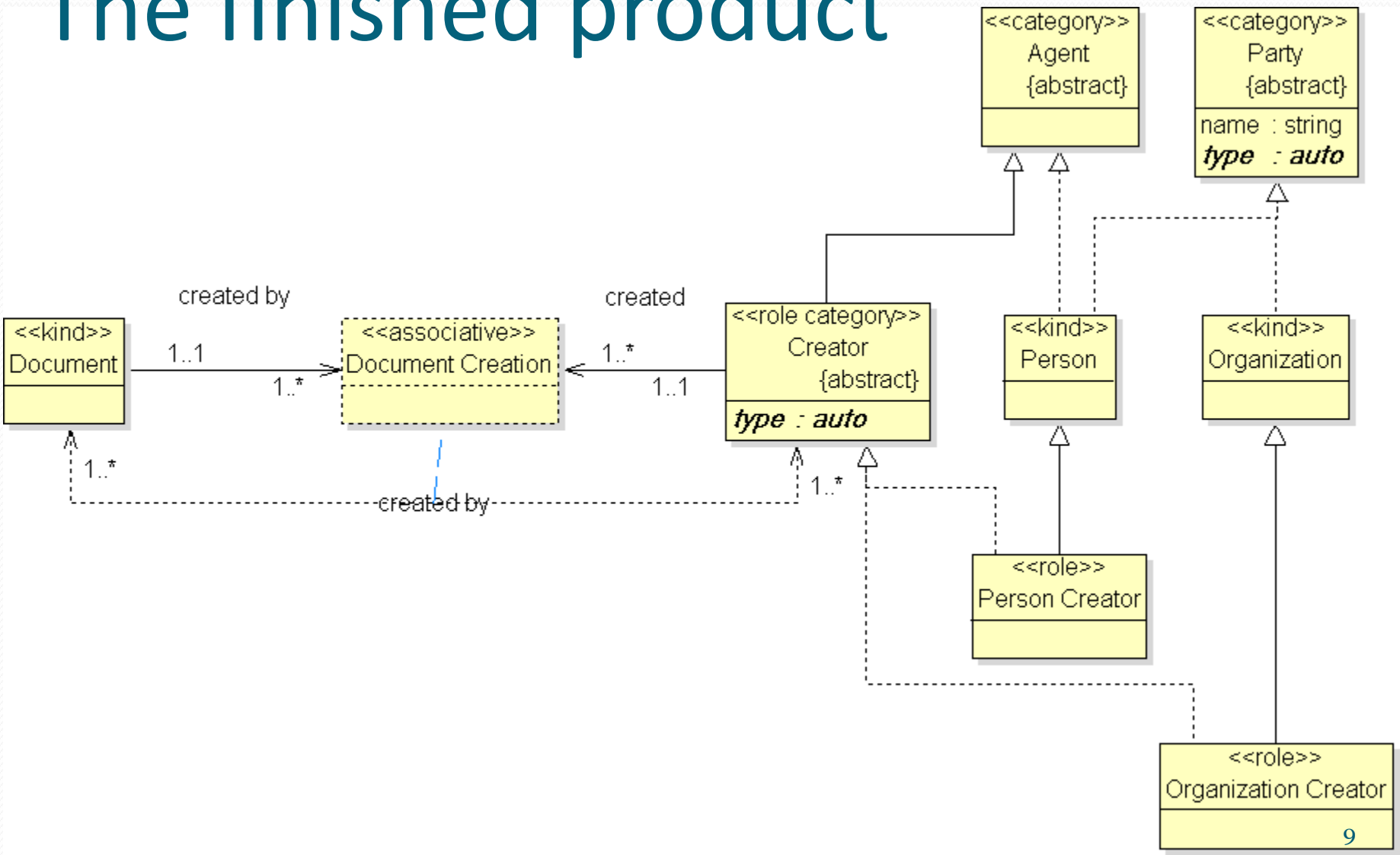
Our subset of OntoUML

- Classes
  - <<kind>> <<category>>
  - <<role>> <<roleCategory>>
  - <<dependent>> <<associative>>
  - <<event>>
- Associations
  - <<non-dependency>> <<dependency>>
  - <<composition>> <<aggregation>> (merelogical relations)
- Attributes
- Datatypes
  - <<primitive>> <<domain>>3 <<enumeration>>
  - <<structure>> <<union>>3

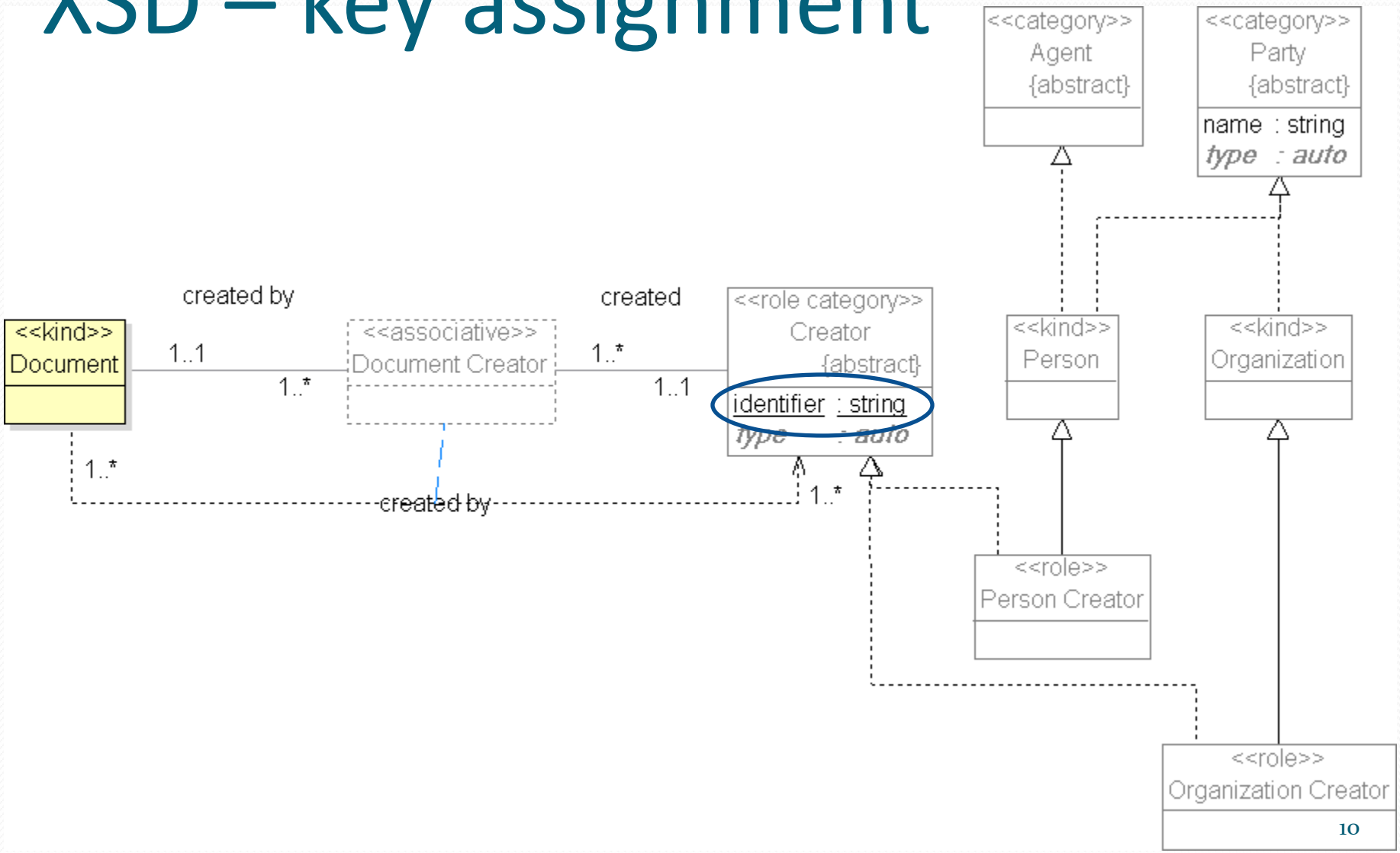
Classes

Concept	Rigidity	Identity	Existential Dependence	Example
<<kind>>	Rigid	Unified	Independent	Person
<<role>>	Anti-Rigid	Unified	Independent	Spouse
<<dependent>>	Rigid	Unified	Dependent	Mental State
<<associative>>	Rigid	Unified	Dependent	Marriage
<<category>>	Rigid	Dispersive	Independent	Party
<<roleCategory>>	Anti-Rigid	Dispersive	Independent	Customer

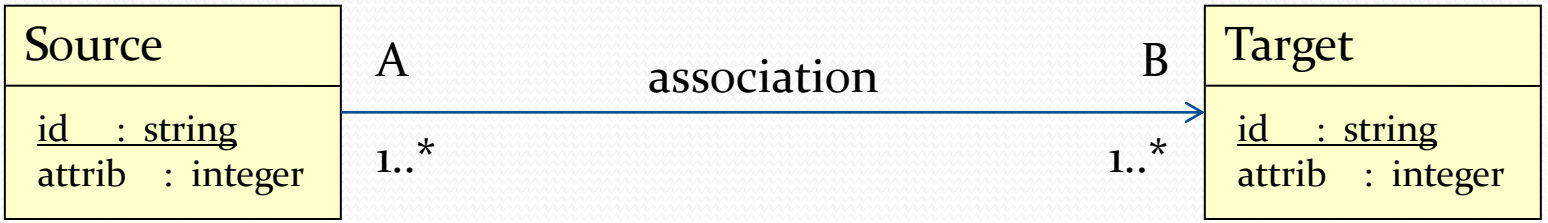
The finished product



XSD – key assignment



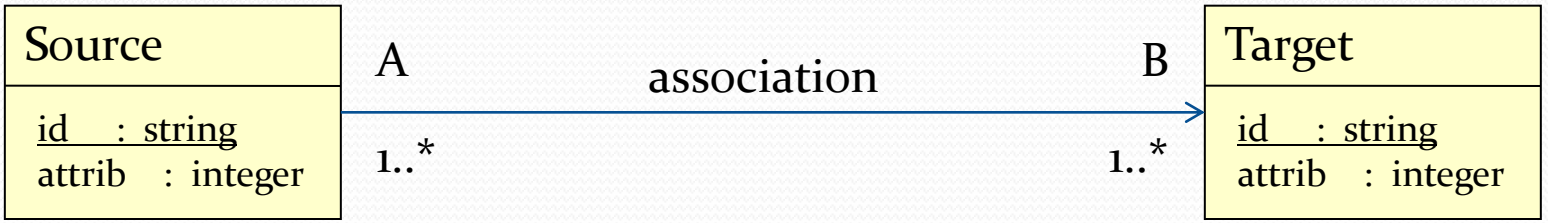
Association Encoding - embed



Include Association: true ☐ false ☒ Global ☐	
A N/A	B Navigable ☒ Include Endpoint true ☐ false ☒ Encoding : embed

```
<Source>
  <Id>Value</Id>
  <Attrib>Value</Attrib>
  <Target>
    <Id>Value</Id>
    <Attrib>Value</Attrib>
  </Target>
</Source>
```

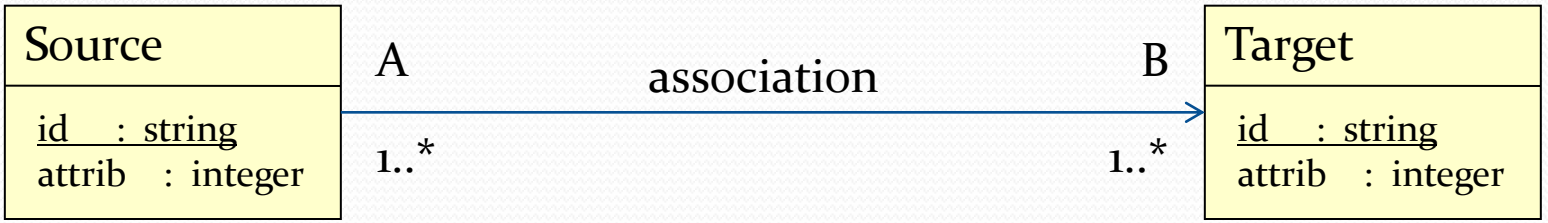
Association Encoding - reference



Include Association: true ☐ false ☒ Global ☐	
A N/A	B Navigable ☒ Include Endpoint true ☒ false ☐ Encoding : reference

```
<Source>
  <Target>
    <Id>FK Value</Id>
  </Target>
</Source>
```

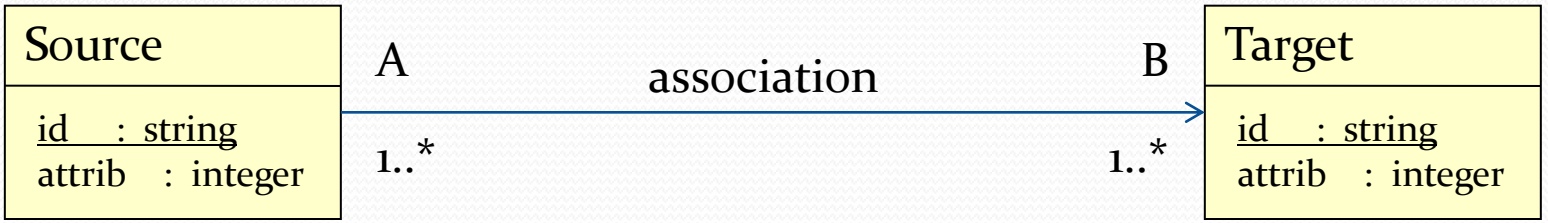
Association Encoding - link



Include Association: true ☐ false ☒ Global ☐	
A N/A	B Navigable ☒ Include Endpoint true ☒ false ☐ Encoding : link

```
<Source>
<Target href=“”/>
</Source>
```

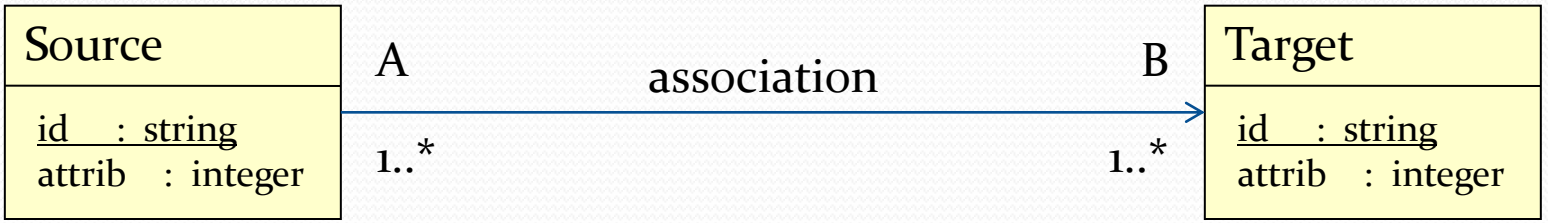
Association Encoding - parts



Include Association: true ☐ false ☒ Global ☐	
A N/A	B Navigable ☒ Include Endpoint true ☐ false ☒ Encoding : reference

```
<Source>
  <Target>FK Value</Target>
</Source>
```

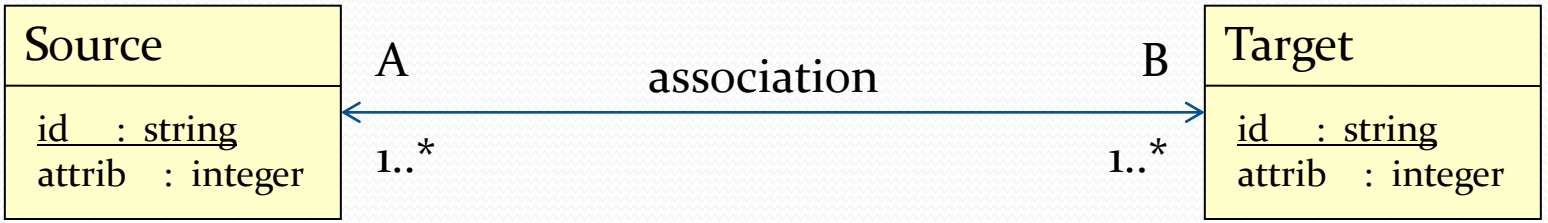
Association Encoding - parts



Include Association: true ☒ false ☐ Global ☐	
A N/A	B Navigable ☒ Include Endpoint true ☒ false ☐ Encoding : reference

```
<Source>
<Association>
   <Target>
    <Id>FK Value</Id>
   </Target>
 </Association>
</Source>
```

Association Encoding - parts



Include Association: true ☒ false ☐ Global ☒	
A N/A	B Navigable ☒ Include Endpoint true ☐ false ☒ Encoding : reference

```
<Association>
  <Source>FK Value</Source>
  <Target>FK Value</Target>
</Association>
```

Successes / Observations

- It actually does work
- Used on many projects
- Model reuse is occurring
- Encoding rules are sufficient however new encoding patterns are still being discovered
- Projects select very different encodings

Challenges

- Hard but not too hard
  - Physical Model Duality (encoding can seem like magic)
  - Tools
- Really Hard
  - Talent
  - Culture
- Things to consider
  - Return on Investment

Future Work

- Better visualization
- Change management
- Continue to improve documentation
- Prototype other implementations
- Explore semi-automatic translation

Questions

Bruce Bauman
Senior System Architect
U.S. Department of Defense

btbauma@gmail.com