WHITESTEIN
Technologies

Unified Modeling Language

UML Profiles

Radovan Cervenka

UML Profiles

- → **The mechanisms that allow metaclasses from existing metamodels to be extended to adapt them for different purposes, e.g., to tailor the UML metamodel for different platforms (such as J2EE or .NET) or domains (such as real-time systems or business process modeling).**
- It is a *conservative extension mechanism* which does not allow modifying the underlying metamodel (UML metamodel).
- Sometimes also referred to as “UML extension mechanisms”.

Metamodel and Metaclass

Metamodel

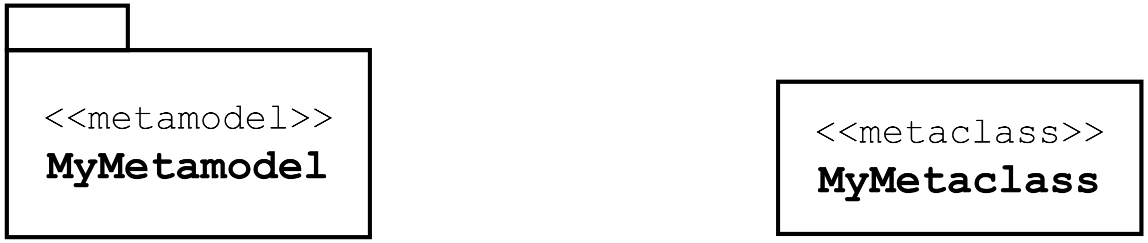
→ A model that defines the modeling concepts (their abstract syntax) of a modeling language.

■ A stereotyped model.

Metaclass

→ A class whose instances are modeling elements or their parts.

■ A specialized class.



<<metamodel>>
MyMetamodel

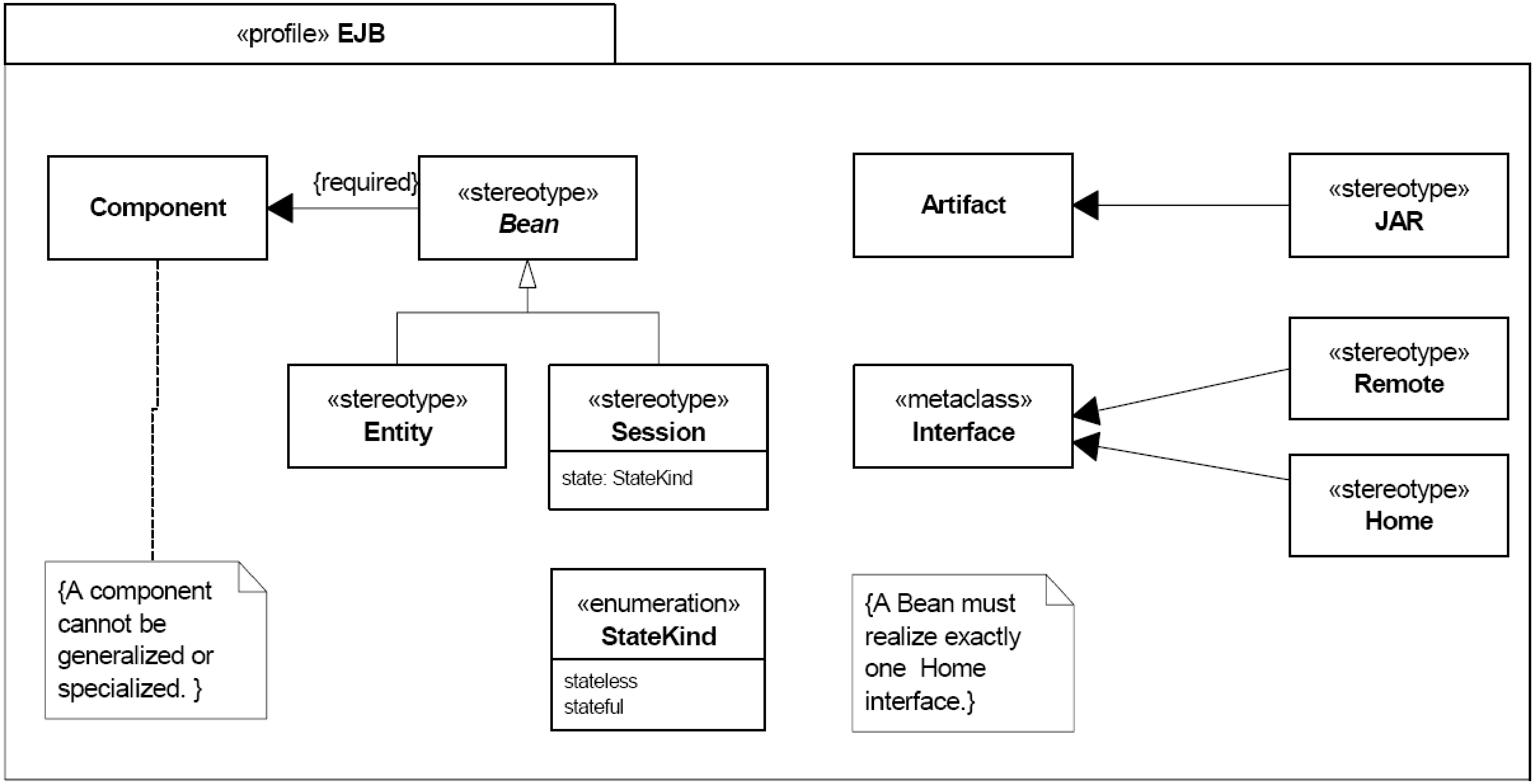
<<metaclass>>
MyMetaclass

Profile

→ Kind of package that extends a reference metamodel with the purpose of adapting the metamodel to a specific platform or domain.

- The primary extension construct is the stereotype.
- A profile is a restricted form of a metamodel that must always be related to a reference metamodel, such as UML.
  - A reference metamodel typically consists of metaclasses that are either imported or locally owned.
  - All metaclasses that are extended by a profile have to be members of the same reference metamodel.
  - Applying a profile does not change the underlying model in any way.
- By application a profile, the following elements are visible:
  1. referenced by an explicit metaclass reference, or
  2. contained (directly or transitively) in a package that is referenced by an explicit metamodel reference, or
  3. extended by a stereotype owned by the applied profile.
  - All other model elements are hidden (not visible) when the profile is applied.

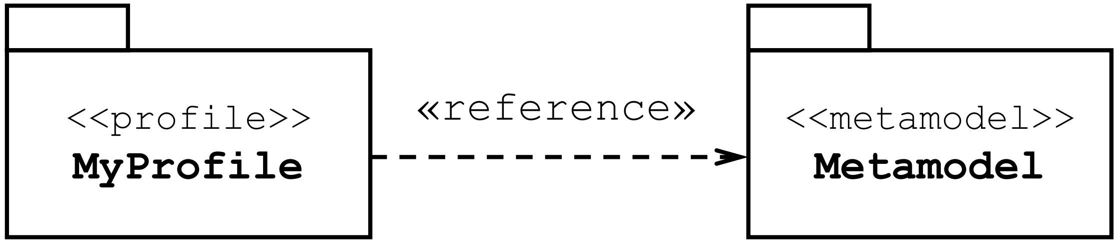
Examples of Profiles



Metamodel Reference

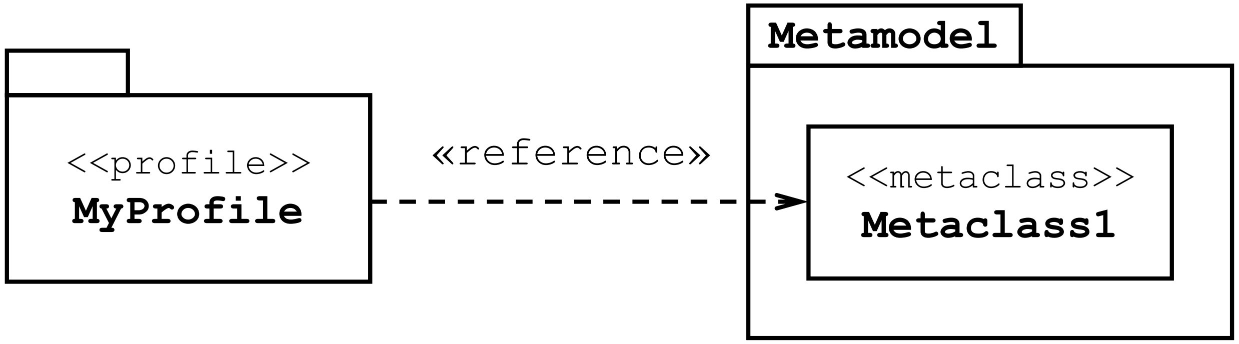
→ References a package containing (directly or indirectly) metaclasses that may be extended.

■ A specific package import.

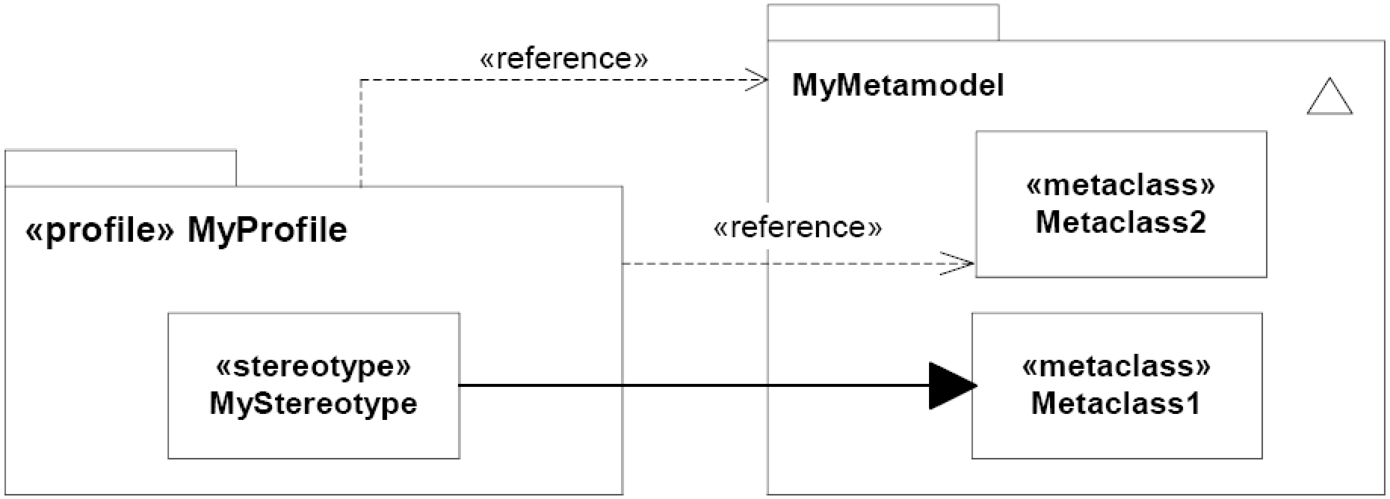


Metaclass Reference

- → References a metaclass that may be extended.
- ■ A specific element import.



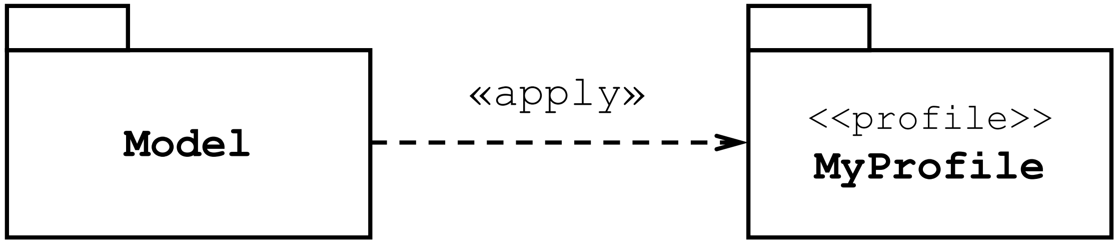
Example of References



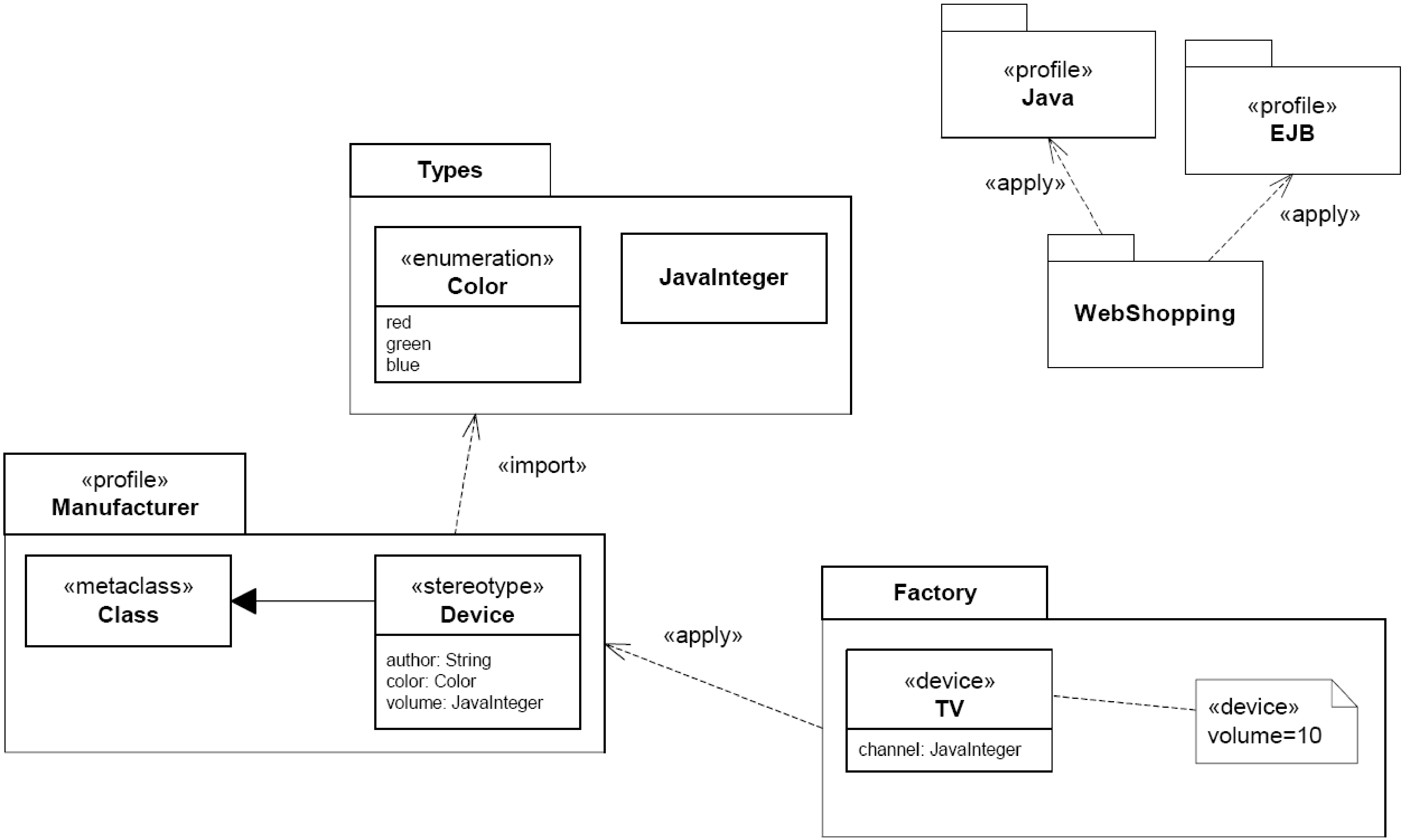
Profile Application

→ A relationship used to show which profiles have been applied to a package.

- One or more profiles may be applied to a package.
  - It is possible to apply multiple profiles to a package as long as they do not have conflicting constraints.
- Applying a profile means that it is allowed, but not necessarily required, to apply the stereotypes that are defined as part of the profile.
- If a profile that is being applied depends on other profiles, then those profiles must be applied first.
- When a profile is applied, instances of the appropriate stereotypes should be created for those elements that are instances of metaclasses with required extensions.



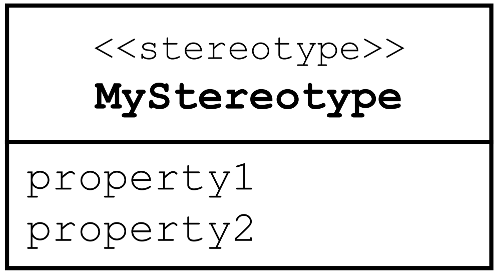
Examples of Profile Applications



Stereotype

→ Specification of how an existing metaclass may be extended, and enables the use of platform or domain specific terminology or notation in place of, or in addition to, the ones used for the extended metaclass.

- May have properties used as ***tag definitions***.
- When a stereotype is applied to a model element, the values of the properties may be referred to as ***tagged values***.
- Must extend one, or more, metaclasses.
  - Any model element from the reference metamodel can be extended by a stereotype.
  - For example in UML, states, transitions, activities, use cases, components, attributes, dependencies, etc. can all be extended.
- Can define one or more images.
- A specialized Class.

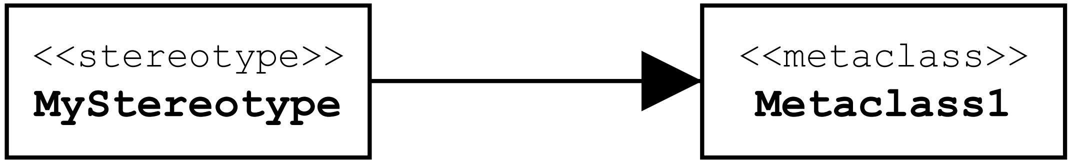


<<stereotype>>
MyStereotype

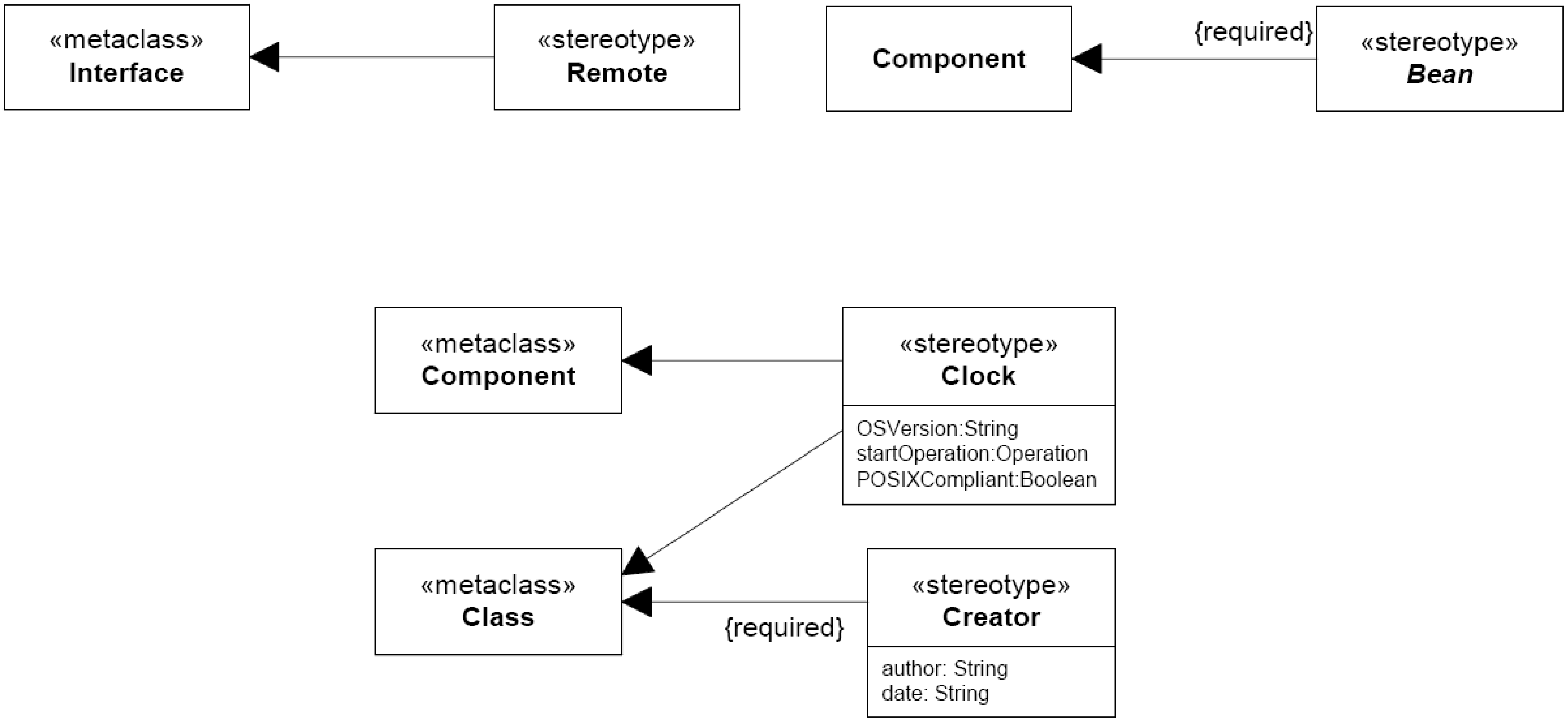
property1
property2

Extension

- → Is used to indicate that the properties of a metaclass are extended through a stereotype.
- {required} is used to indicate whether an instance of the extending stereotype must be created when an instance of the extended metaclass is created.
- A specialized association.
- Note: Stereotype (from UML) is the only kind of metaclass that cannot be extended by stereotypes.



Examples of Extensions



Application of a Stereotype

- Application of a stereotype in a modeling element.
- Format:

  ‘<<‘ *stereotype-name* ‘>>’ [‘<<‘ *stereotype-name* ‘>>’]*

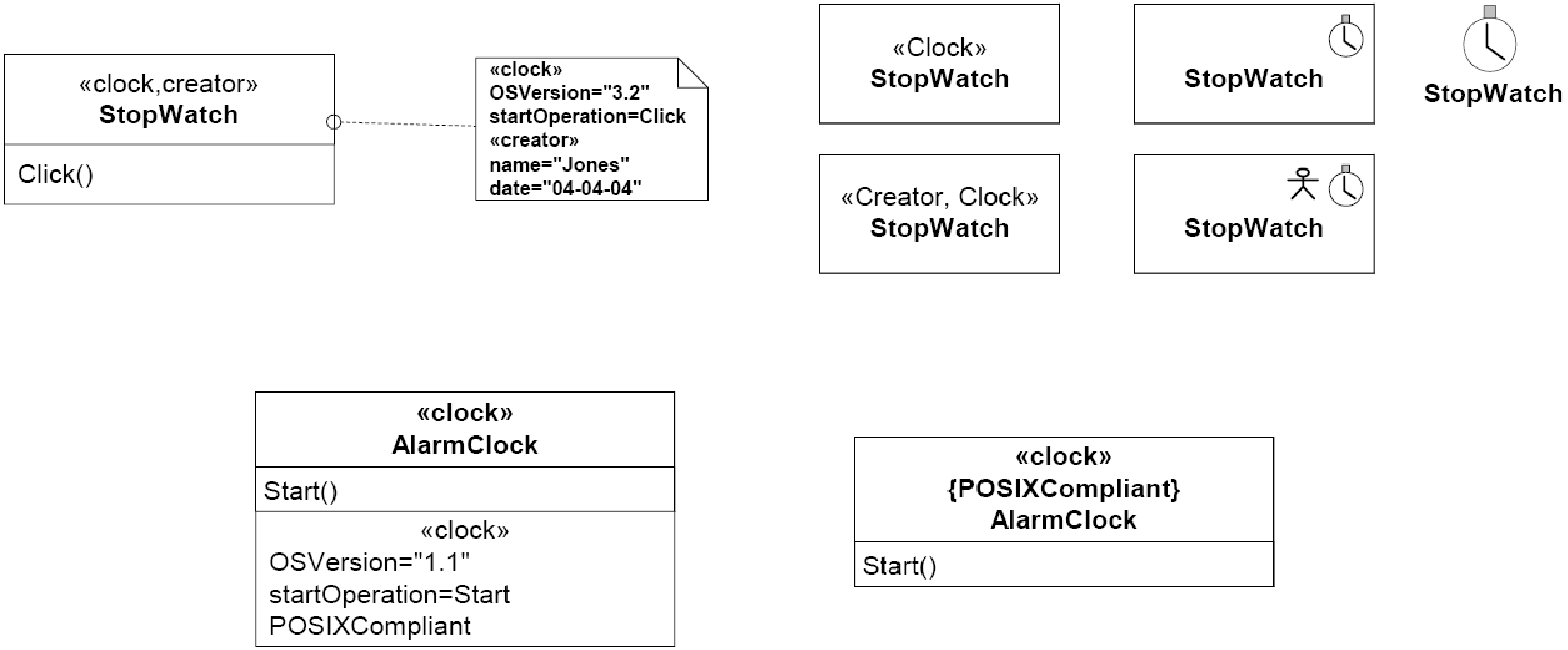
  - Placed above or before the name of the model element.
  - Alternatively (also), an icon (or multiple icons) can be used.
- Format of tagged values:

  namestring ‘=’ *valuestring*

  valuestring ::= *value* [‘,’ *value*]*

  - In a comment connected to the stereotyped element.
  - In a separate compartment.
  - Above the name strng.
  - If the type of property is Boolean and a value is omitted, the value is implicitly true.

Examples of Stereotype Applications



«clock,creator»
StopWatch

Click()

«clock»
OSVersion="3.2"
startOperation=Click
«creator»
name="Jones"
date="04-04-04"

«Clock»
StopWatch

StopWatch

StopWatch

«Creator, Clock»
StopWatch

StopWatch

«clock»
AlarmClock

Start()

«clock»
OSVersion="1.1"
startOperation=Start
POSIXCompliant

«clock»
{POSIXCompliant}
AlarmClock

Start()