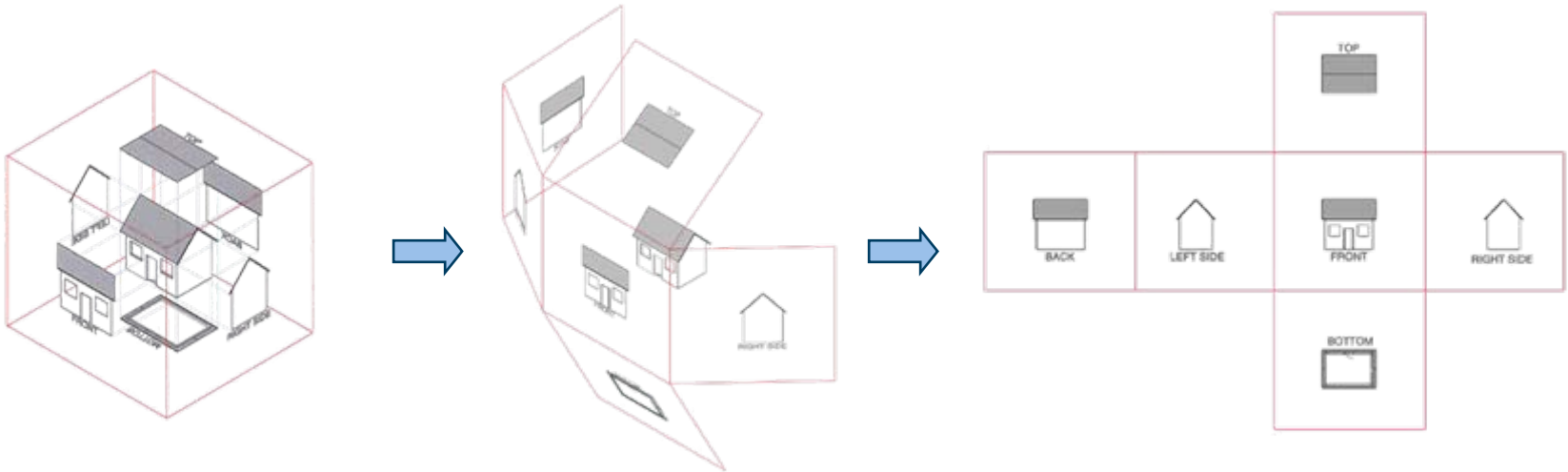
Orthographic Service Engineering

Contents

- Orthographic drawing
- Orthographic software modeling
- Multi-view manipulation
- Prototype tool
- Repository format
- Current status

A Simple "Modeling" Metaphor

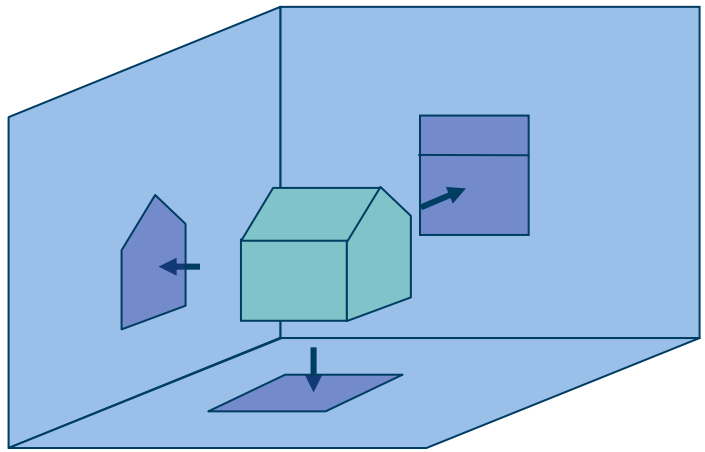
- other engineering disciplines have a long and successful tradition of technical drawing
  - orthographic projection



- so why don't we do this in software engineering?

Orthographic Projection

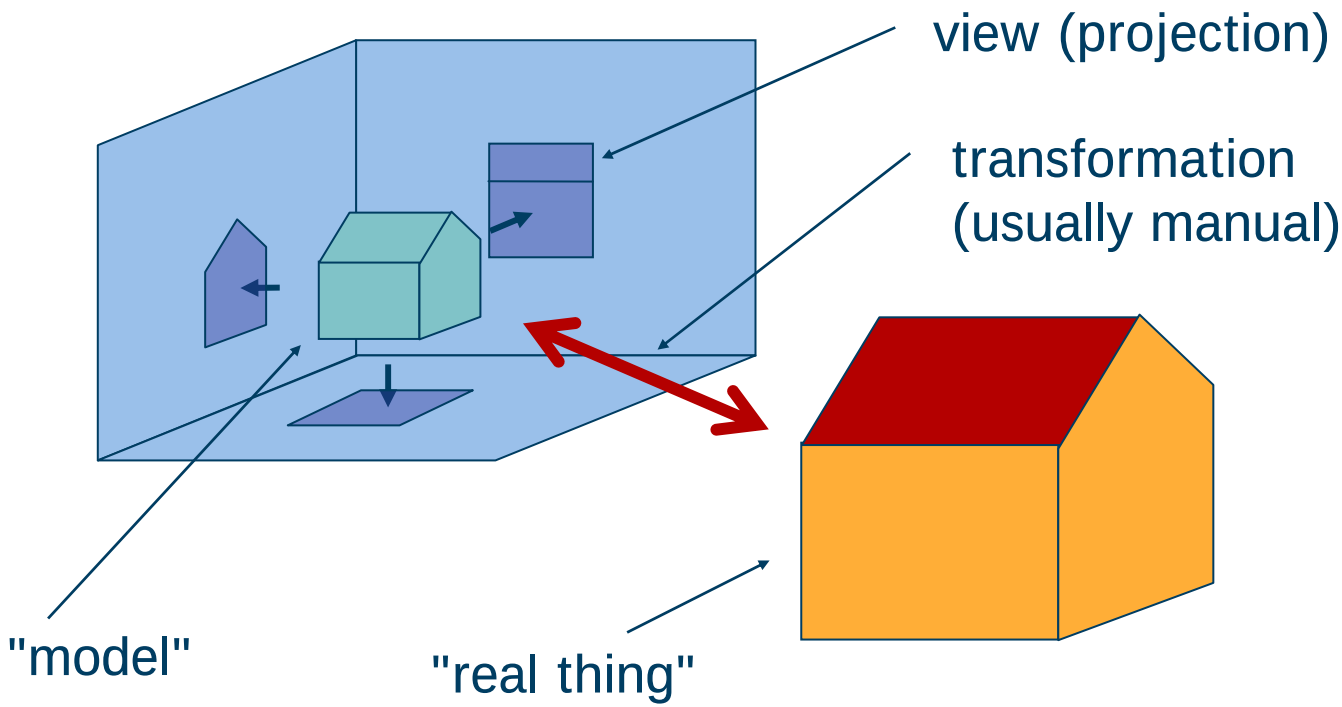
- a long-established technique for depicting physical artifacts in a way which is
  - concise
  - precise
  - intuitive
  - permanent
  - modular
  - scaleable
  - composition-based
  - standardized



- a suitable metaphor for model driven development?

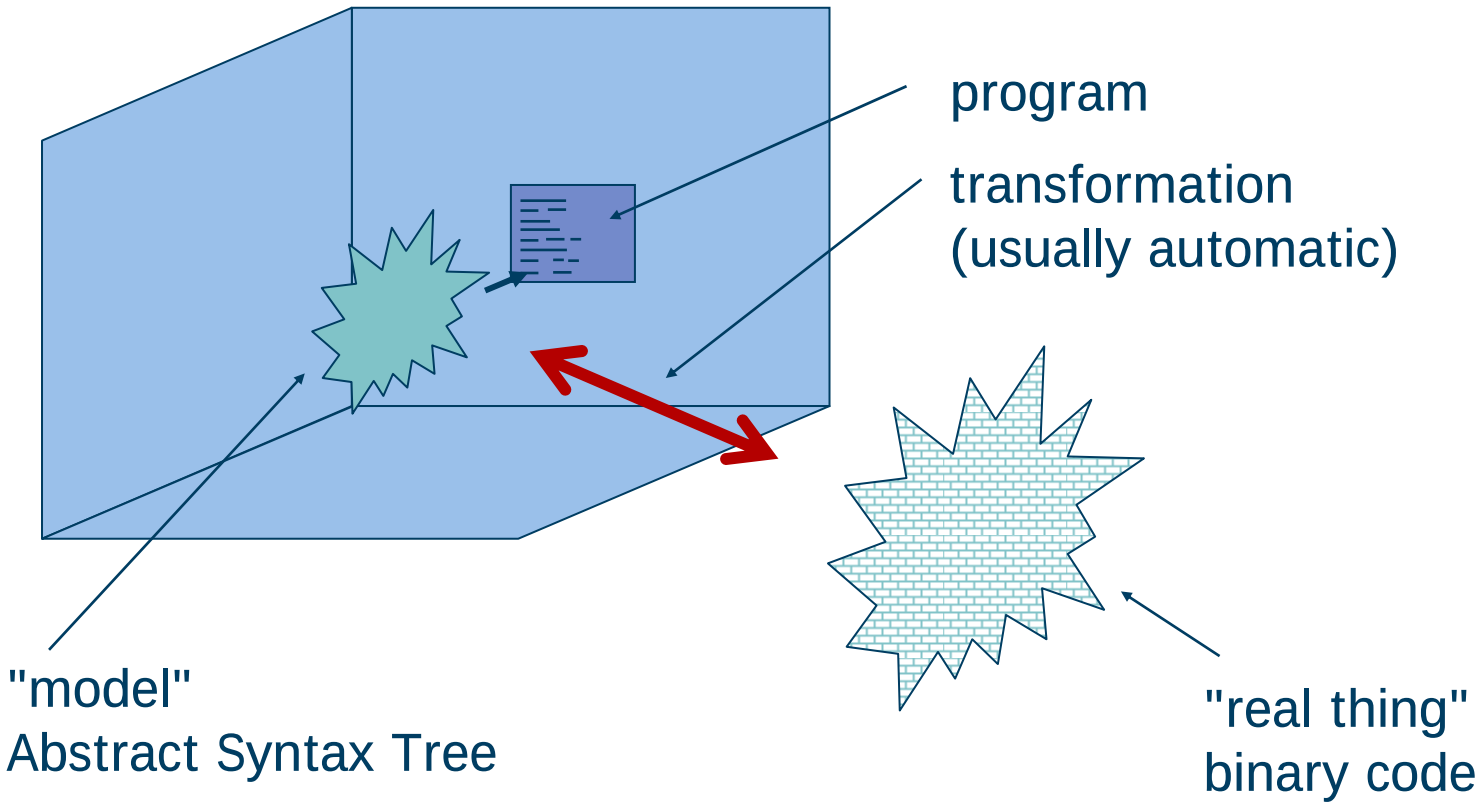
The Model versus the "Real Thing"

- distinction between the model and the "real thing" is clear in other engineering disciplines

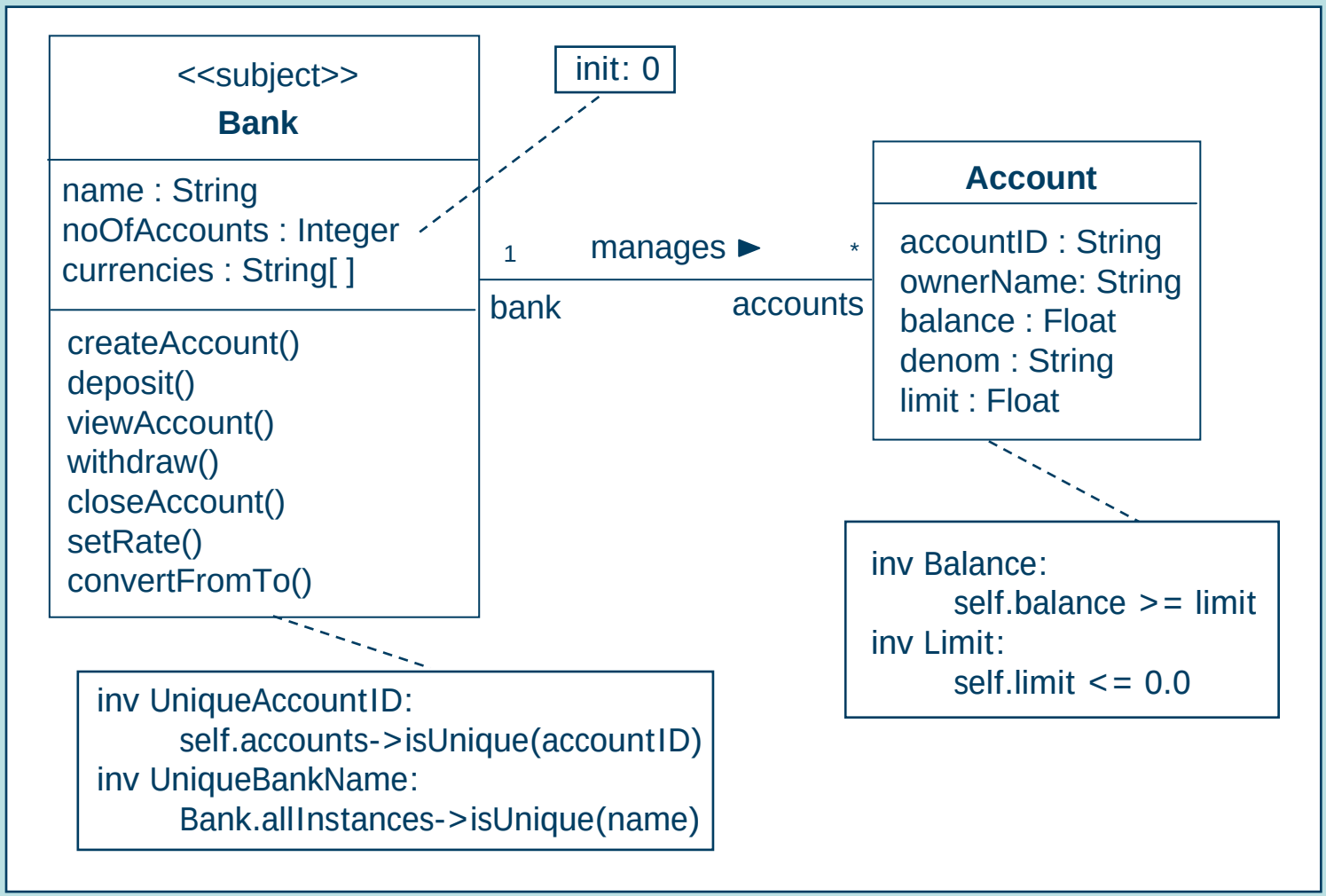


Traditional "Language" Technology

- in software, the binary code is the closest we can get to the "real thing"

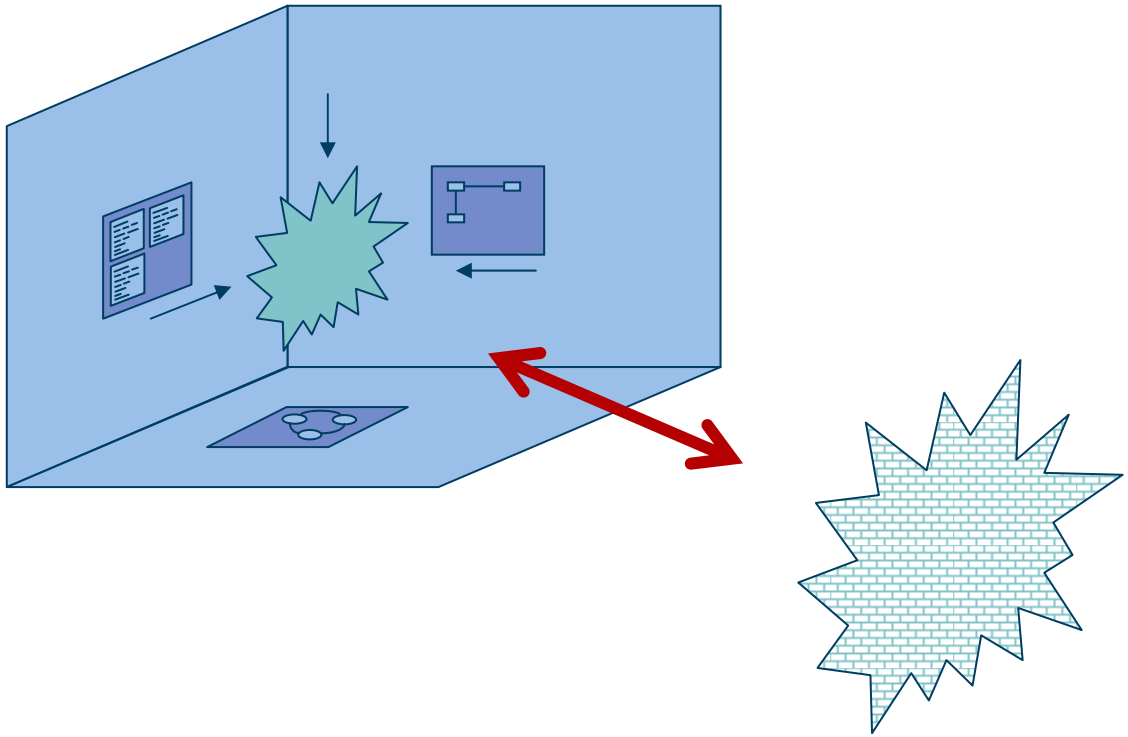


Orthographic Software Modeling

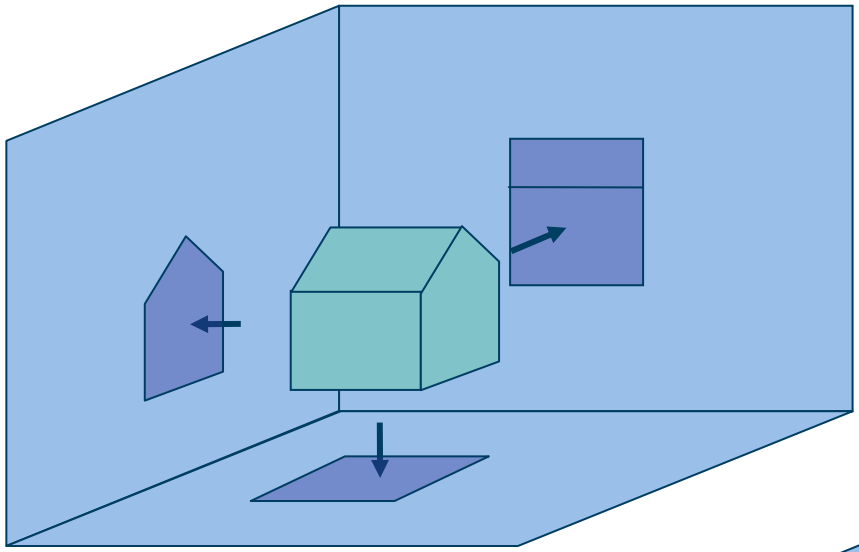


Orthographic Software Modeling

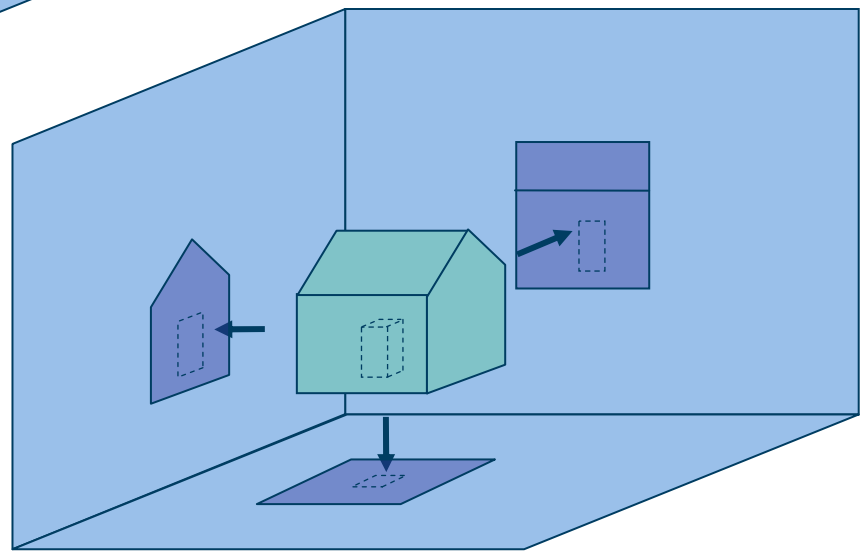
- mappings are driven by all three model



White Box vs. Black Box Views

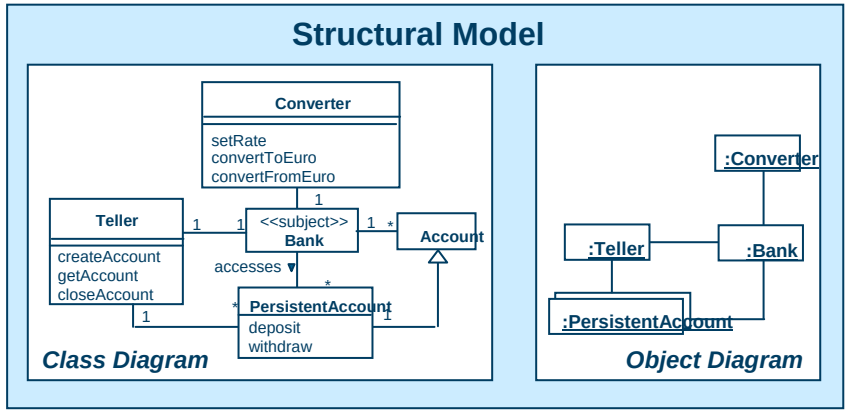


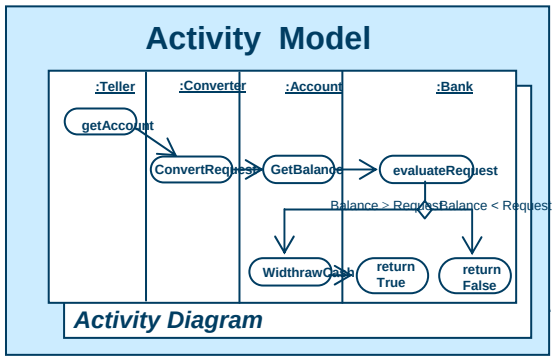
black box view

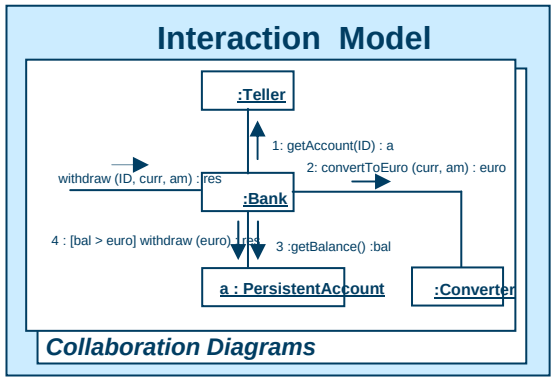


white box view

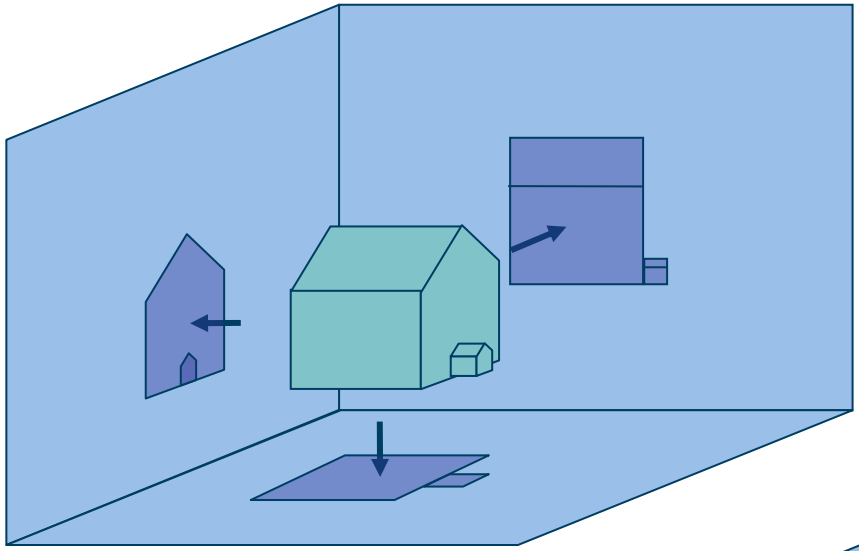
Bank Component Realization (KobrA)



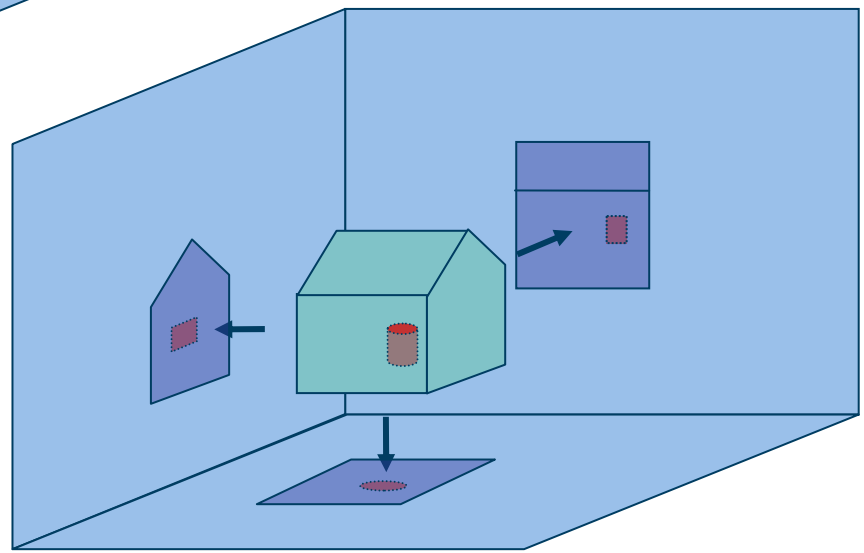




White Box vs. Black Box Components

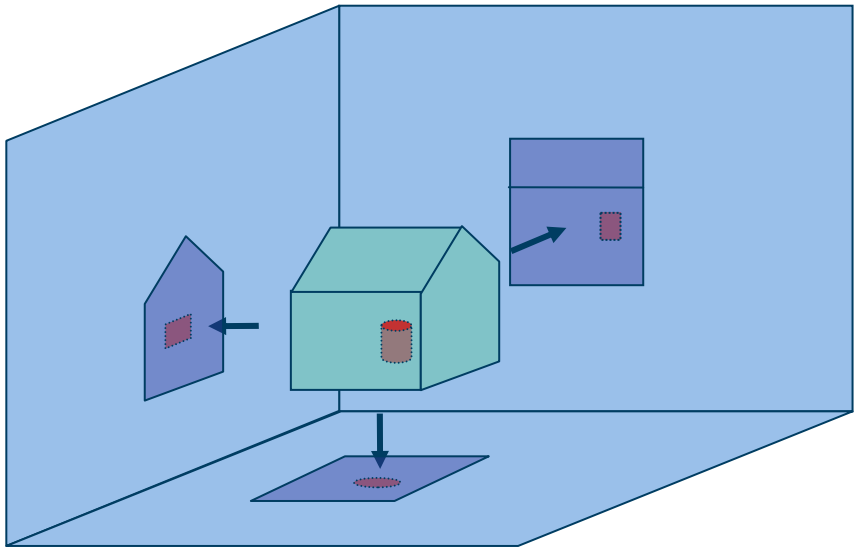


black box components

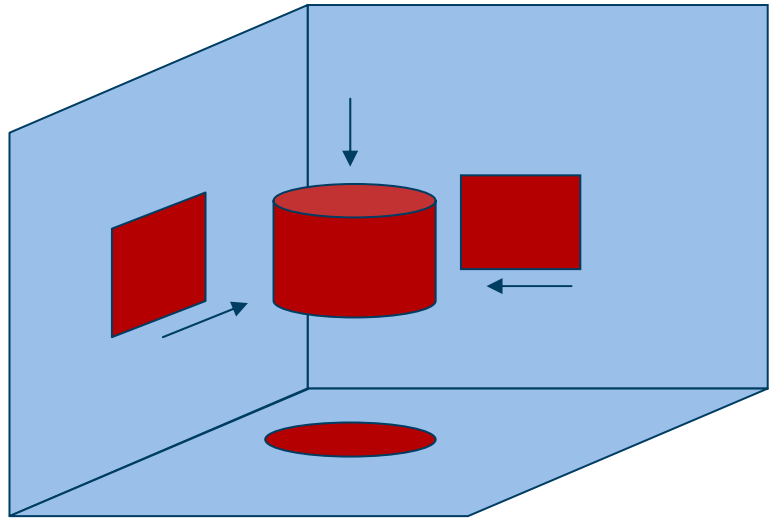


white box components

Orthographic Decomposition

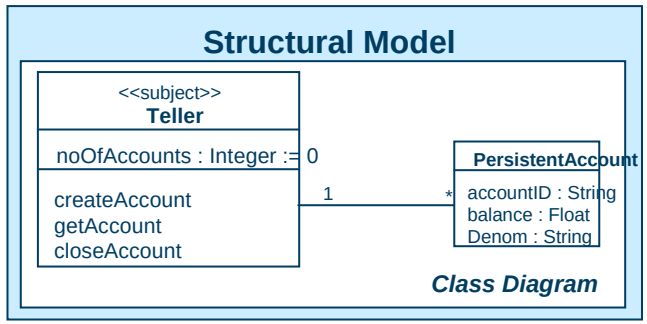


views of composite



views of component

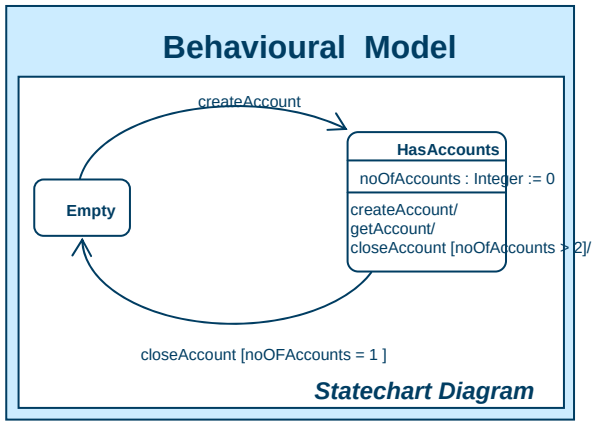
Teller Component Specification (KobrA)



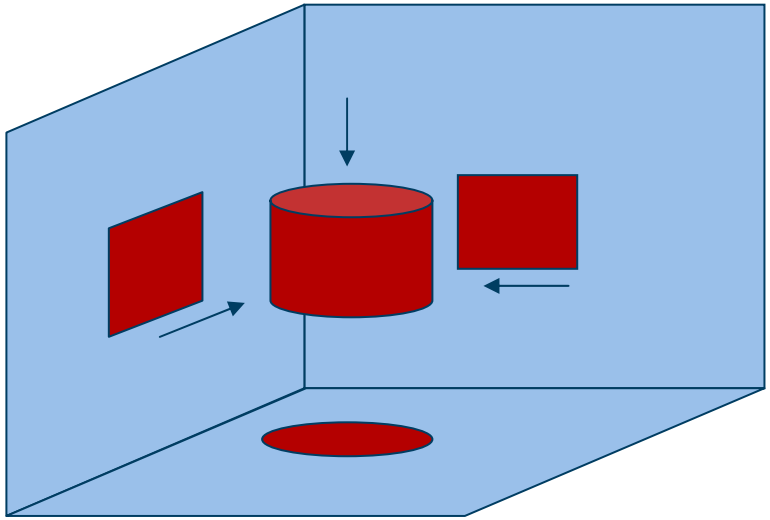
Functional Model

Name	createAccount
Informal Description	An account is opened in a particular currency for a customer with a particular name, and the Account ID is returned
Constraints	--
Receives	name : String currency:String
Returns	A String with the ID of the account
Changes	teller
Assumes	There is an exchange rate for the specified currency
Result	A new account with a unique ID in the denomination, currency, has been generated The name of the customer has been stored in account The account ID has been returned

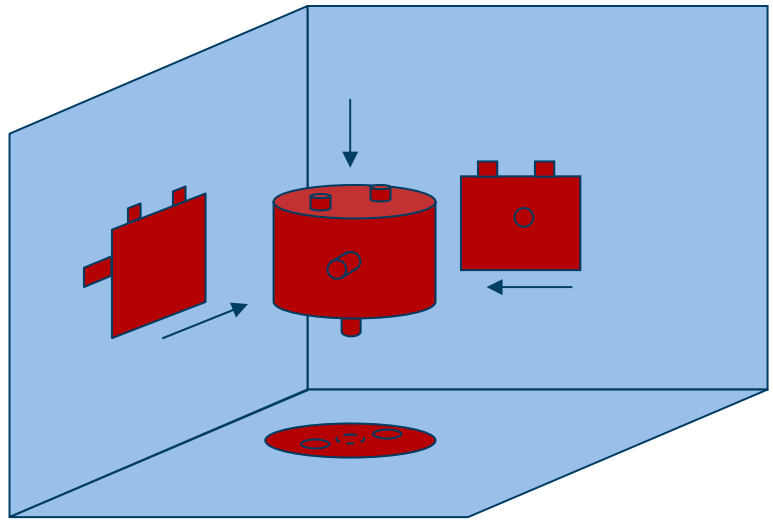
Operation Specifications



Abstraction

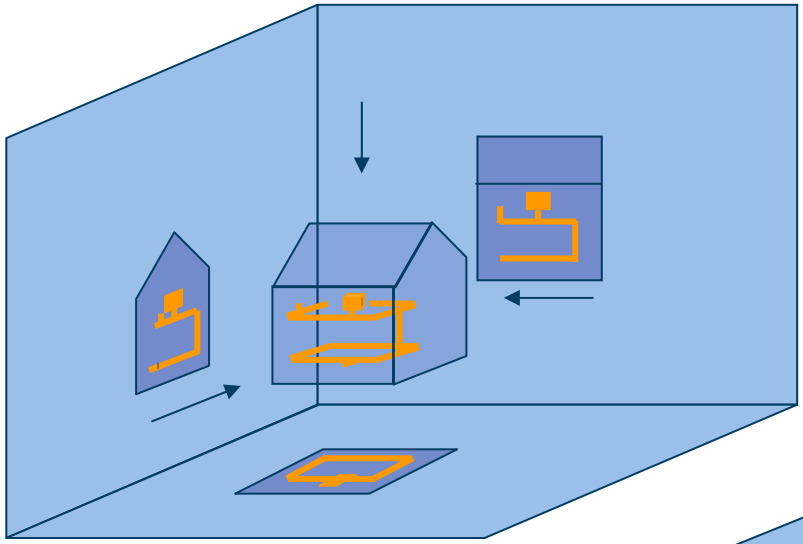


abstract view

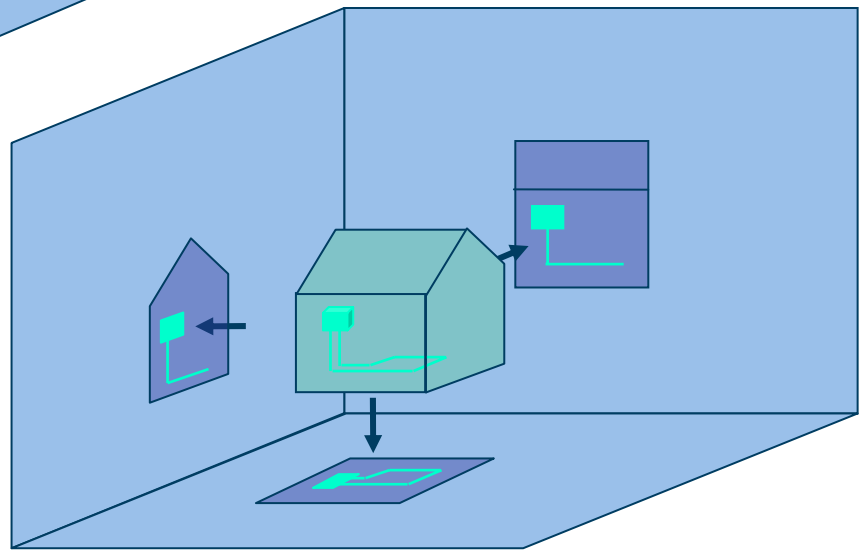


concrete (detailed) view

Aspect-Orientation

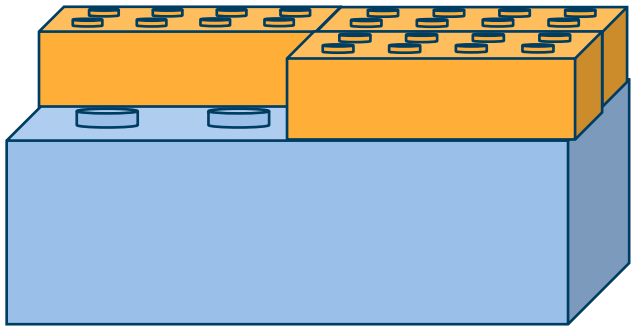


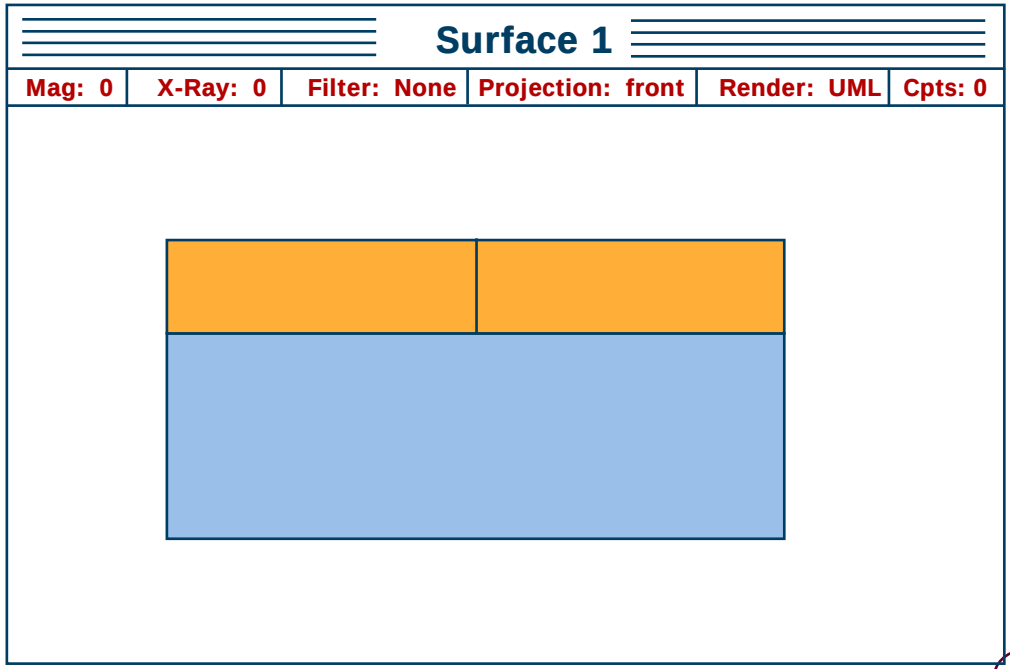
"orange" aspect elaboration



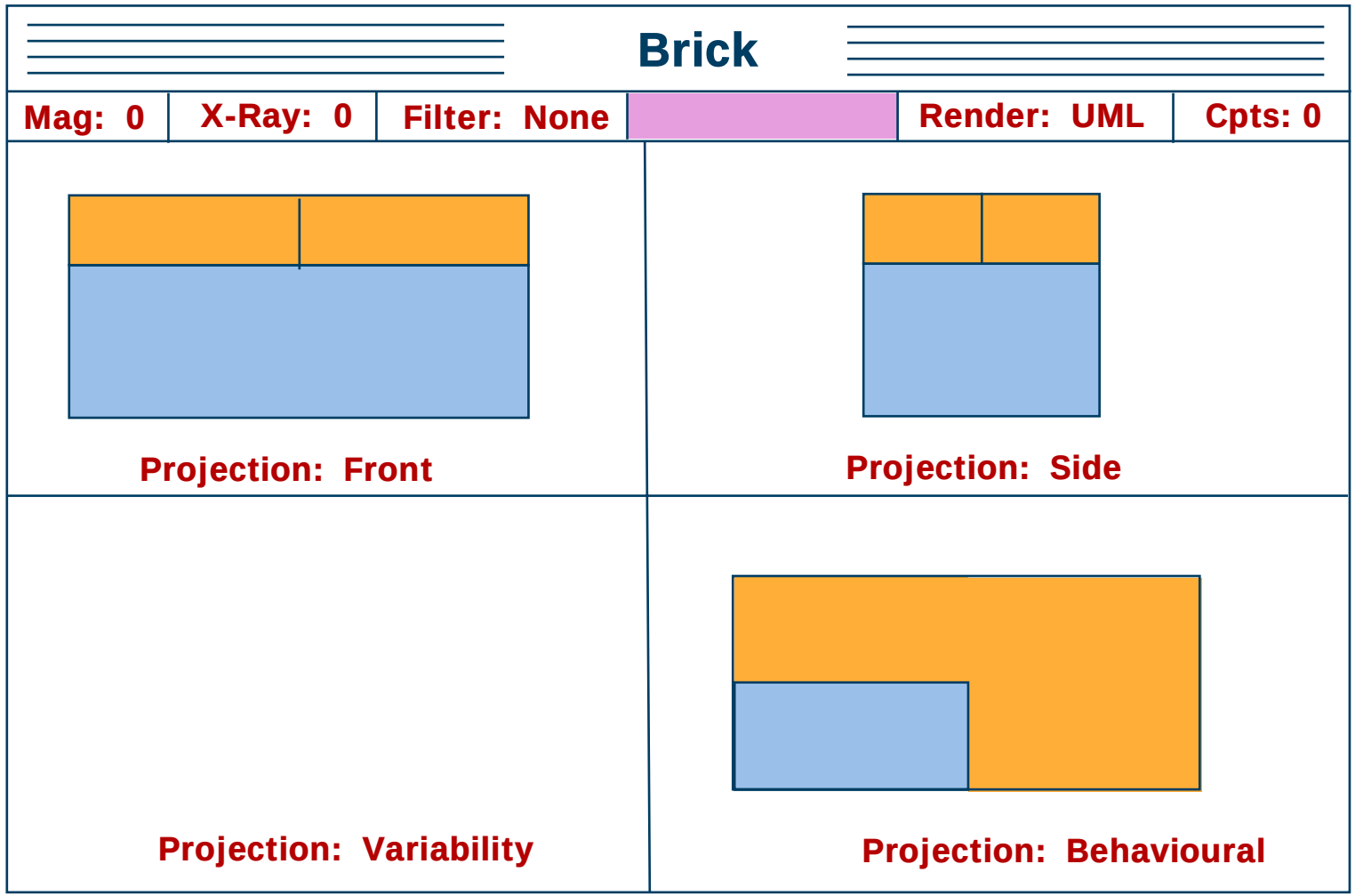
"blue" aspect elaboration

Projection Example





Projection Example: All Views Displayed



Bank Example

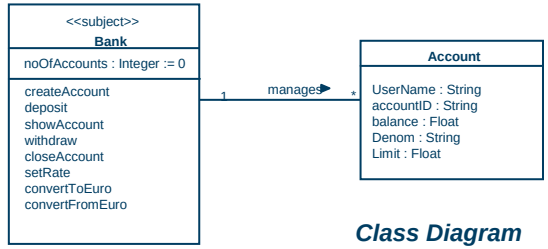
Bank

Mag: 0 | **X-Ray: 0** | **Filter: None** | **Render: UML** | **Cpts: 0**

Name	withdraw
Informal Description	An amount of money in a particular currency is withdrawn from an account.
Constrains	--
Receives	account:Account currency:String amount:Float
Returns	A boolean indicating whether the withdrawal was possible
Changes	account
Assumes	--
Result	If account or currency are not valid, the operation has been aborted and an exception has been raised. Else if account.balance + account limit >= amount, then account.balance := account.balance – amount and „true" has been returned Otherwise „false" has been returned.

Operation Specifications

Projection: Functional



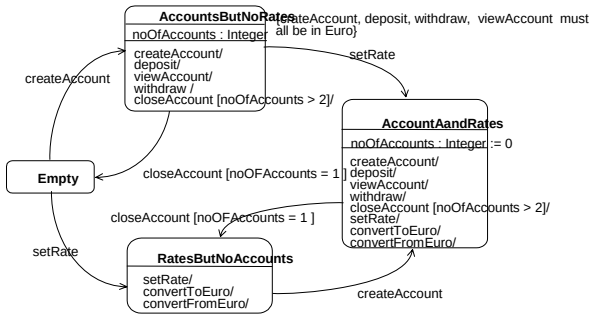
Class Diagram

Projection: Strutural

Decision Table

ID	Question	Subject	Resolution	Effect
1	Is a customer allowed to to overdraw his/her account up to a certain limit?	Overdraw Limit	yes (default)	Resolve structural diagram decision 1: yes Resolve operation spec. withdraw decision 1: yes ...
			no	Resolve structural diagram decision 1: no Resolve operation spec. withdraw decision 1: no ...
2	Is it an international bank that handles different currencies?	Currencies	yes (default)	Resolve structural diagram decision 2: yes Resolve operation spec. withdraw decision 2: yes Resolve operation spec. withdraw decision 3: yes ...
			no	Resolve structural diagram decision 2: no Resolve operation spec. withdraw decision 2: no Resolve operation spec. withdraw decision 3: no ...

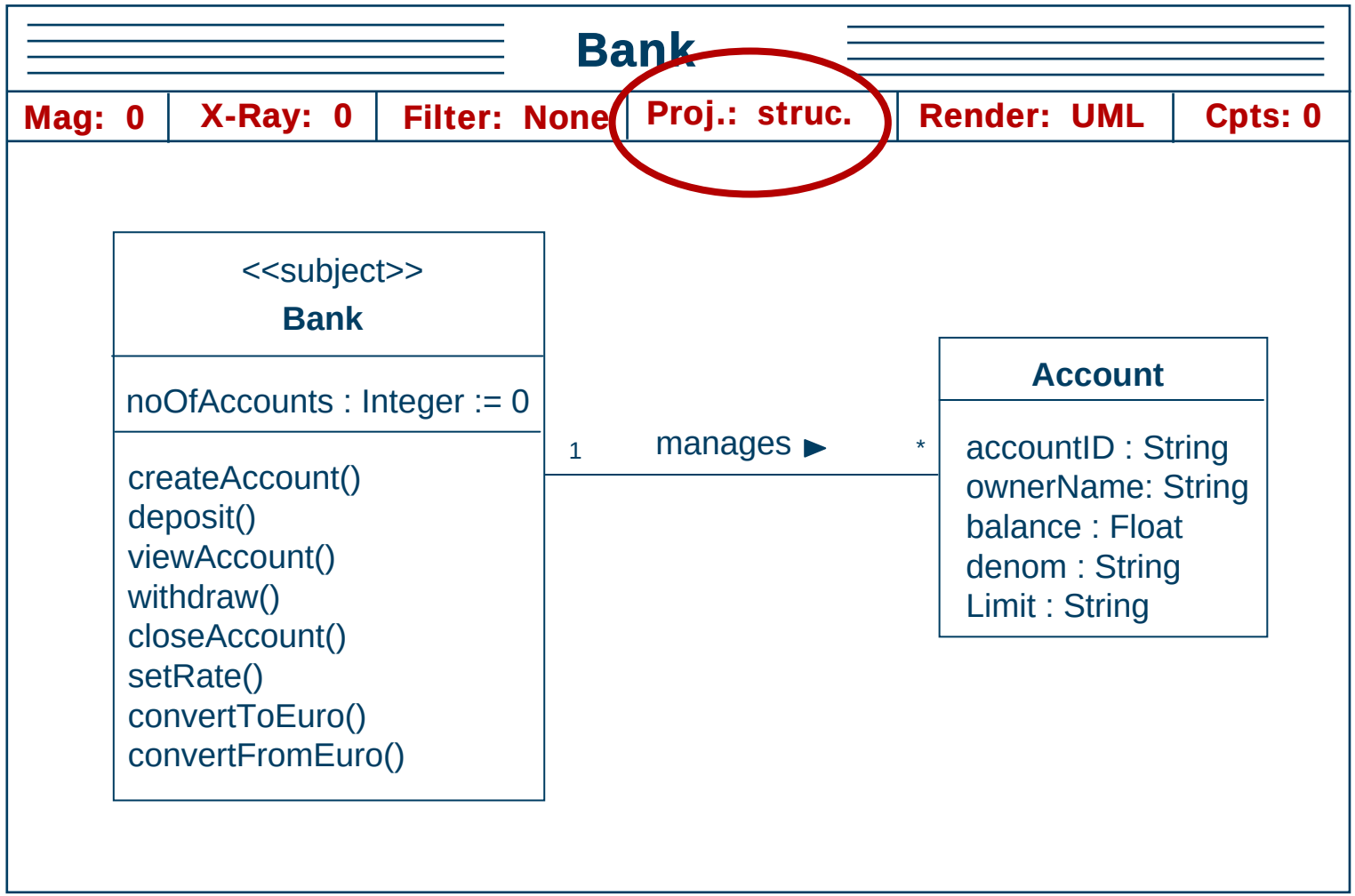
Projection: Variability



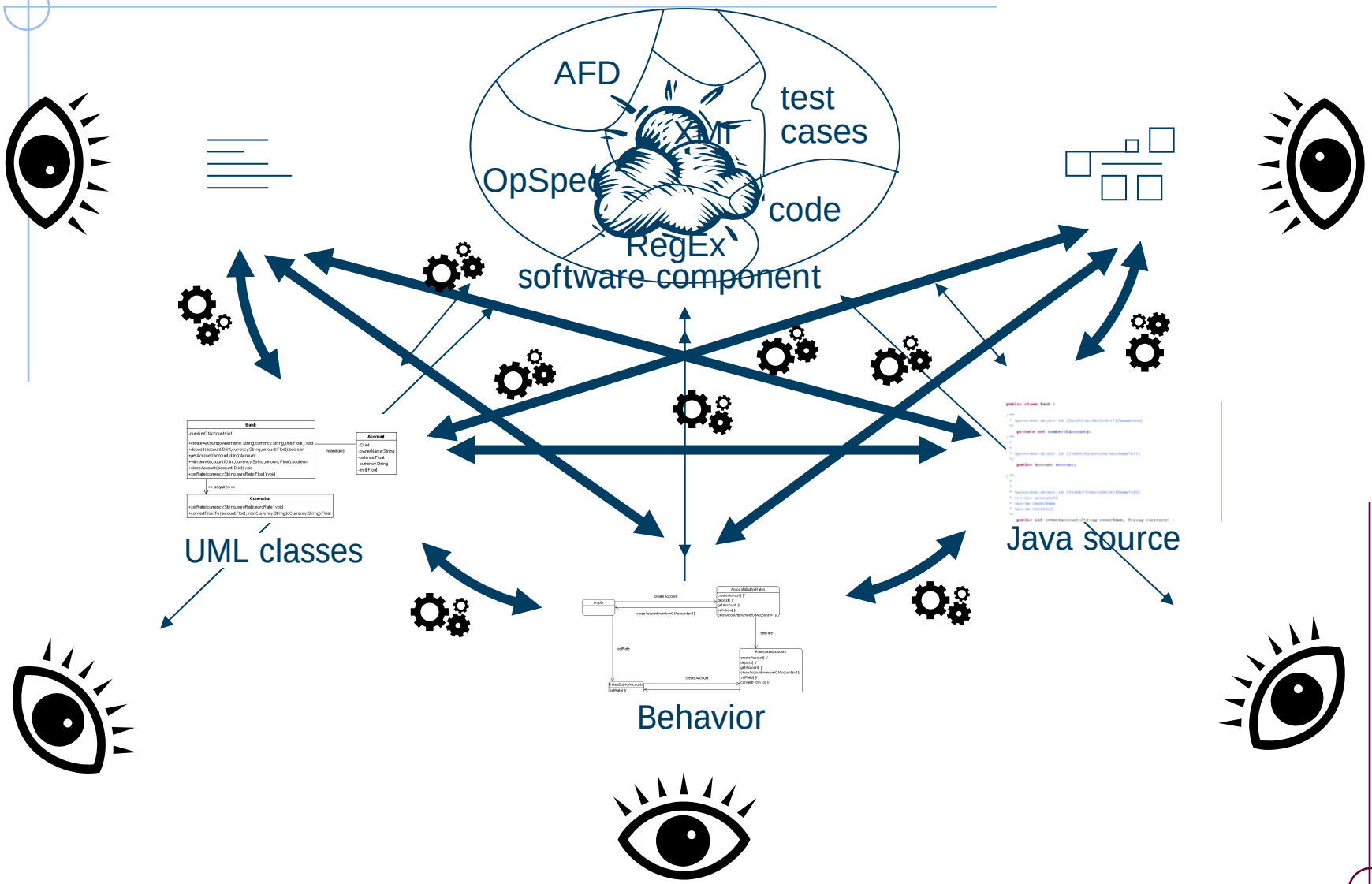
Statechart Diagram

Projection: Behavioural

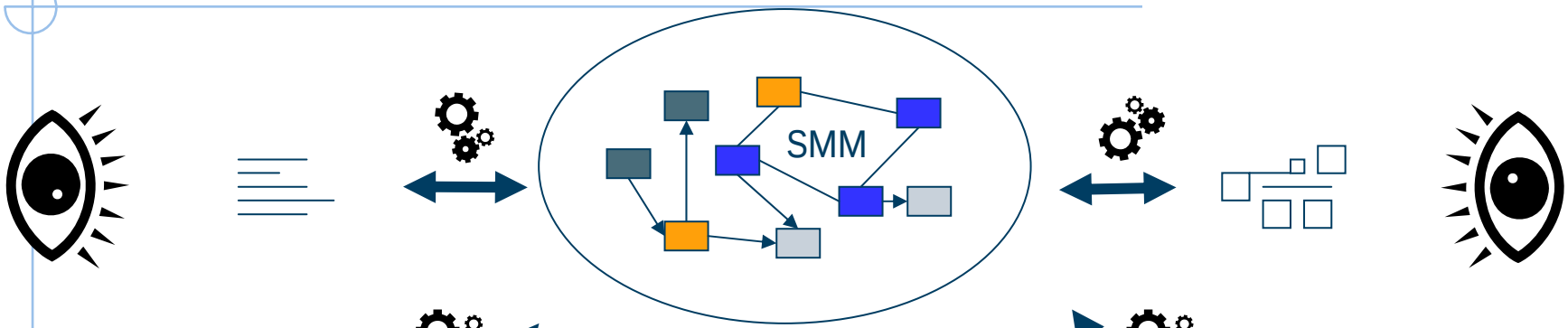
Bank Example: Structural View

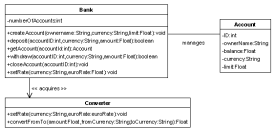


Component Modeling Tool Support



Ideal World



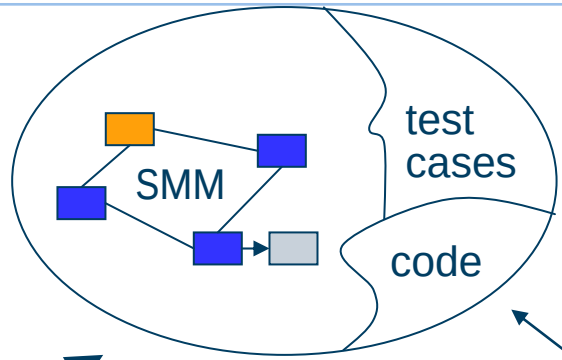


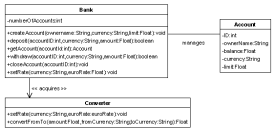
UML classes

Java source

Behavior

Pragmatic Solution

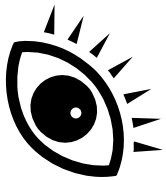


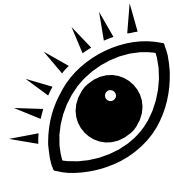


UML classes

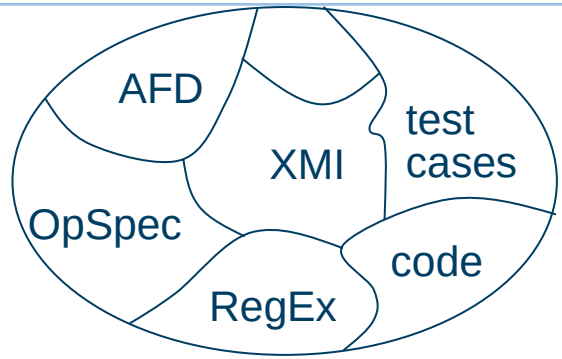
Java source

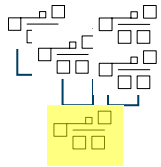
Behavior

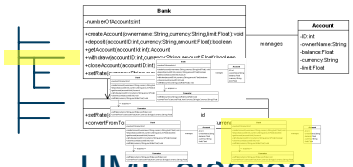




Artifact Navigation

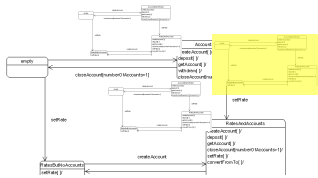






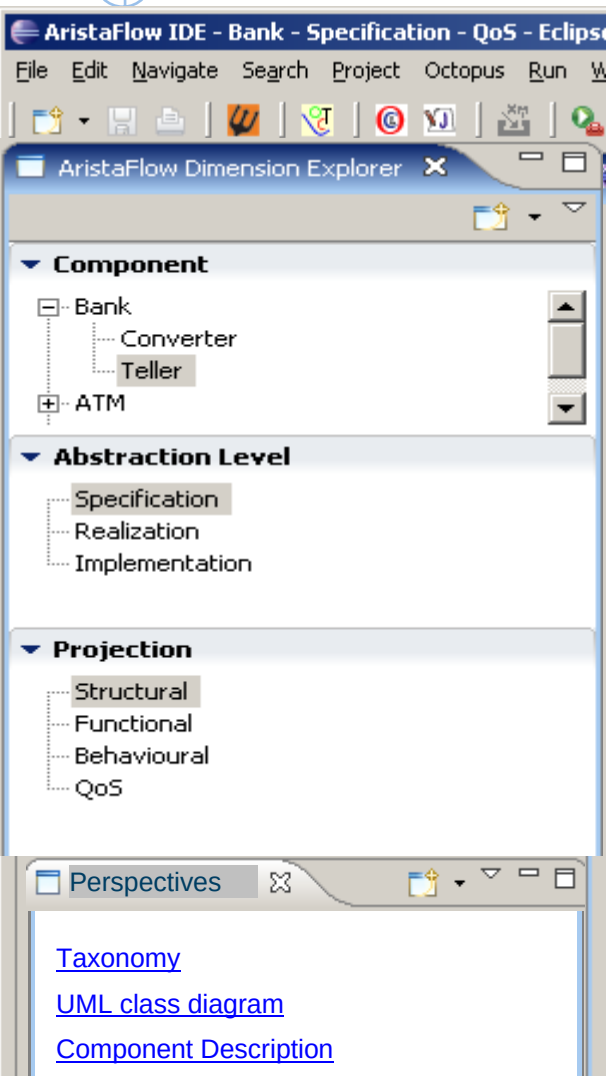
UML classes

Java s e



Behavior

Component Oriented Navigation



Component

Bank

Converter

change()

set()

Teller

….

Abstraction

Specification

Realization

Implementation

…

Projection

Structural

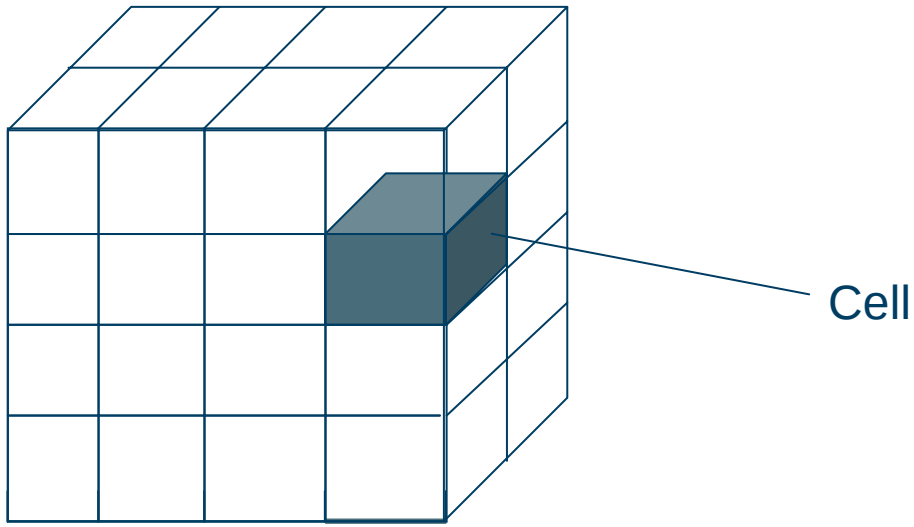
Functional

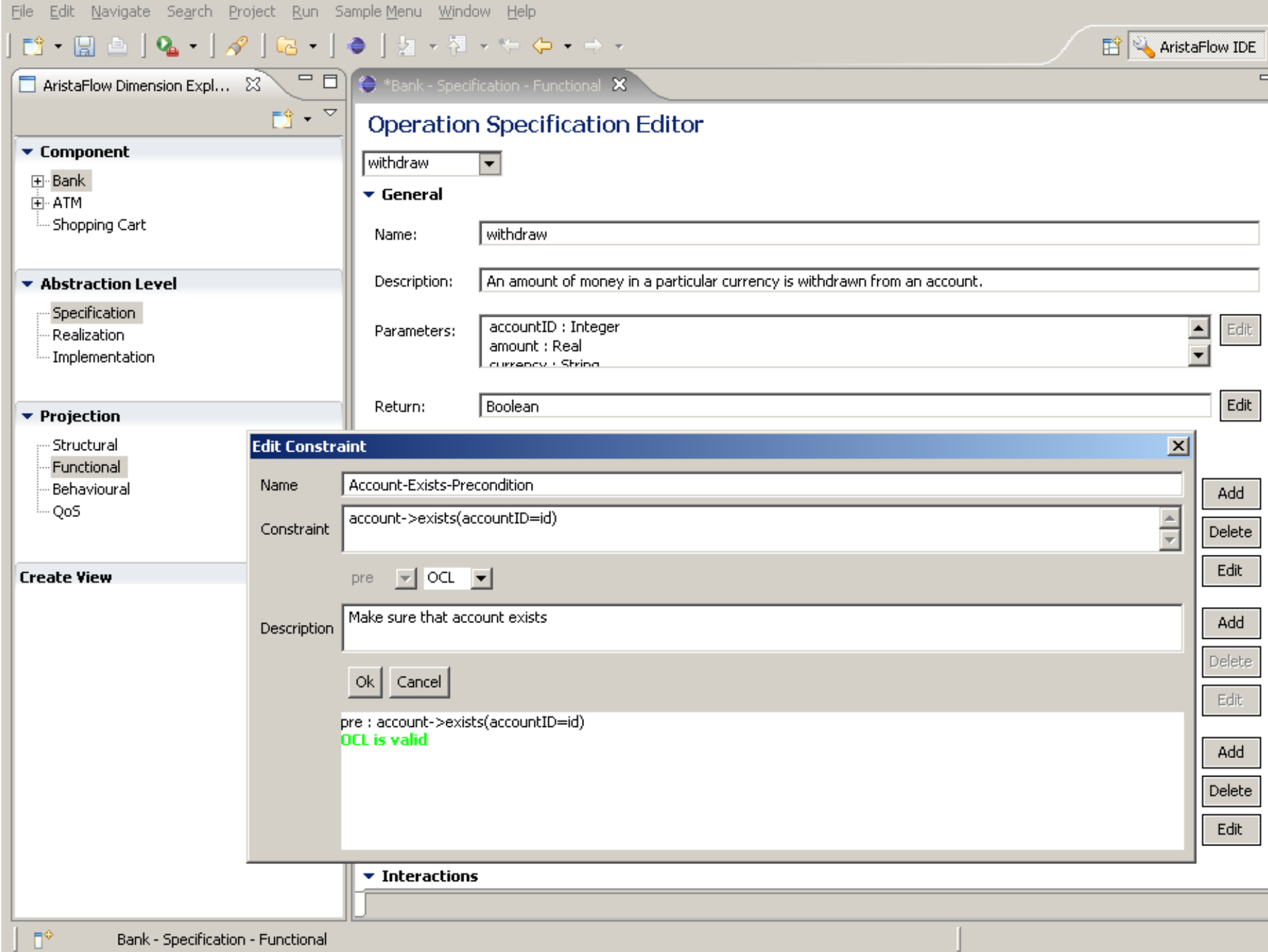
Behavioural

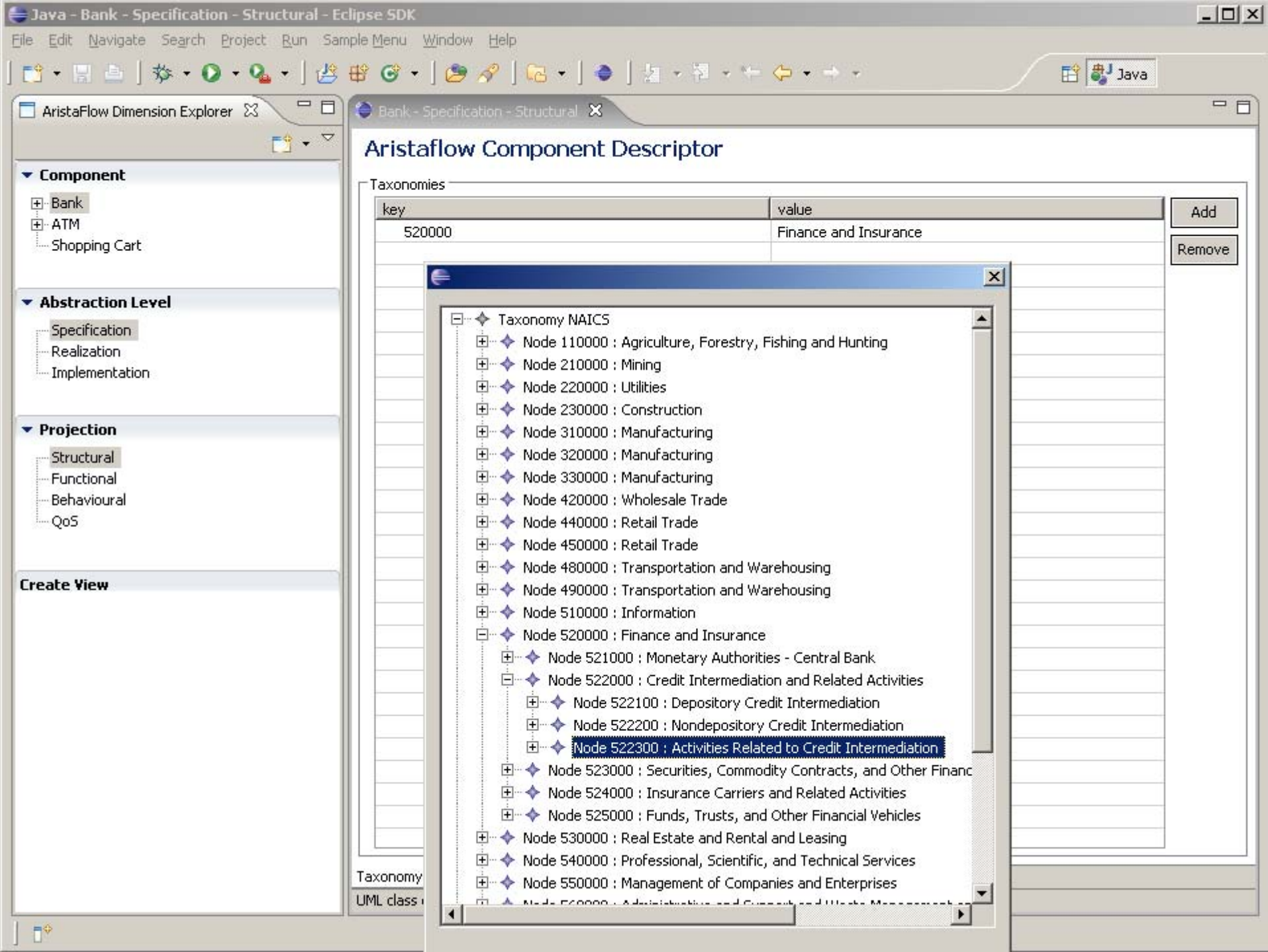
QoS

….

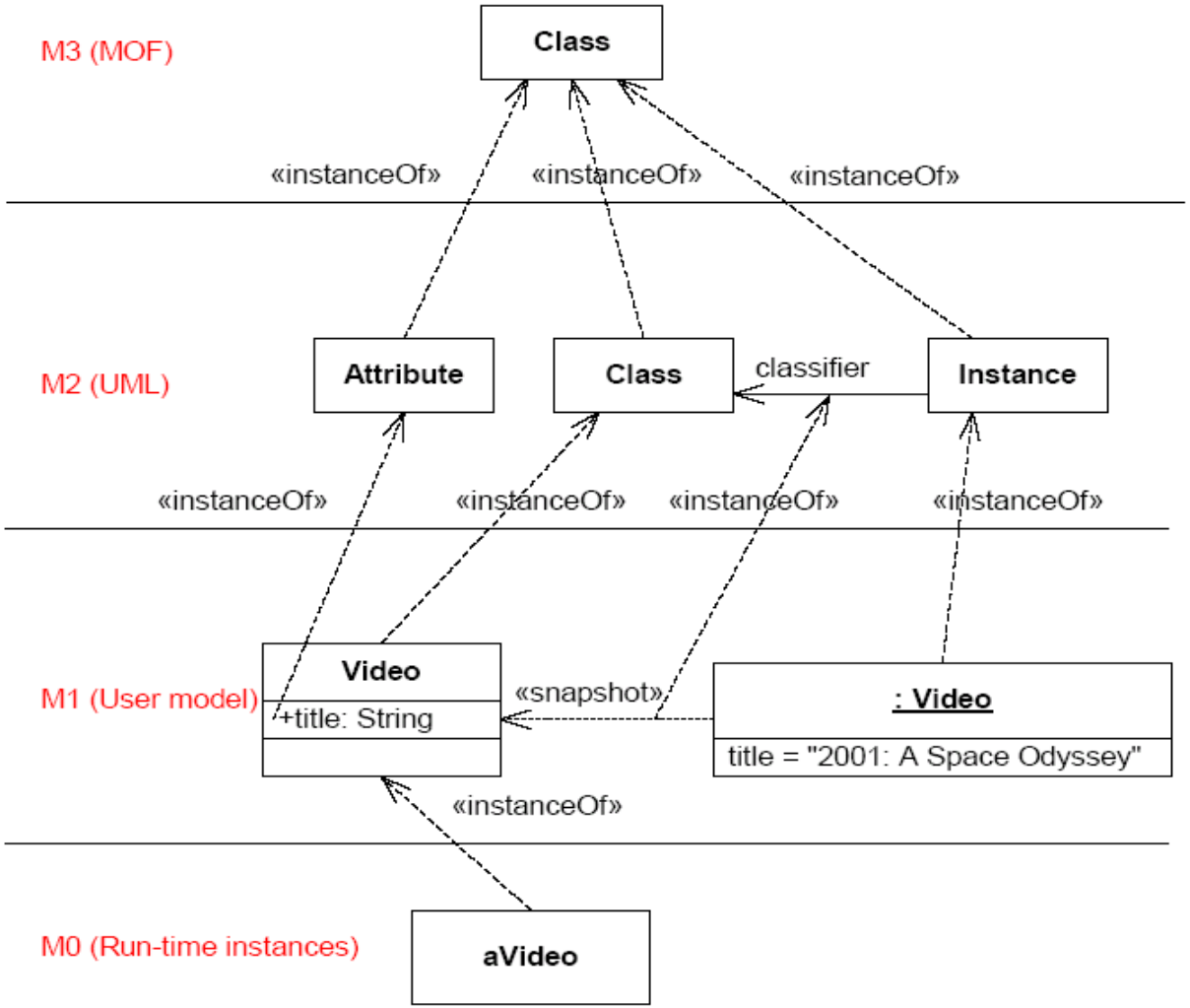
…



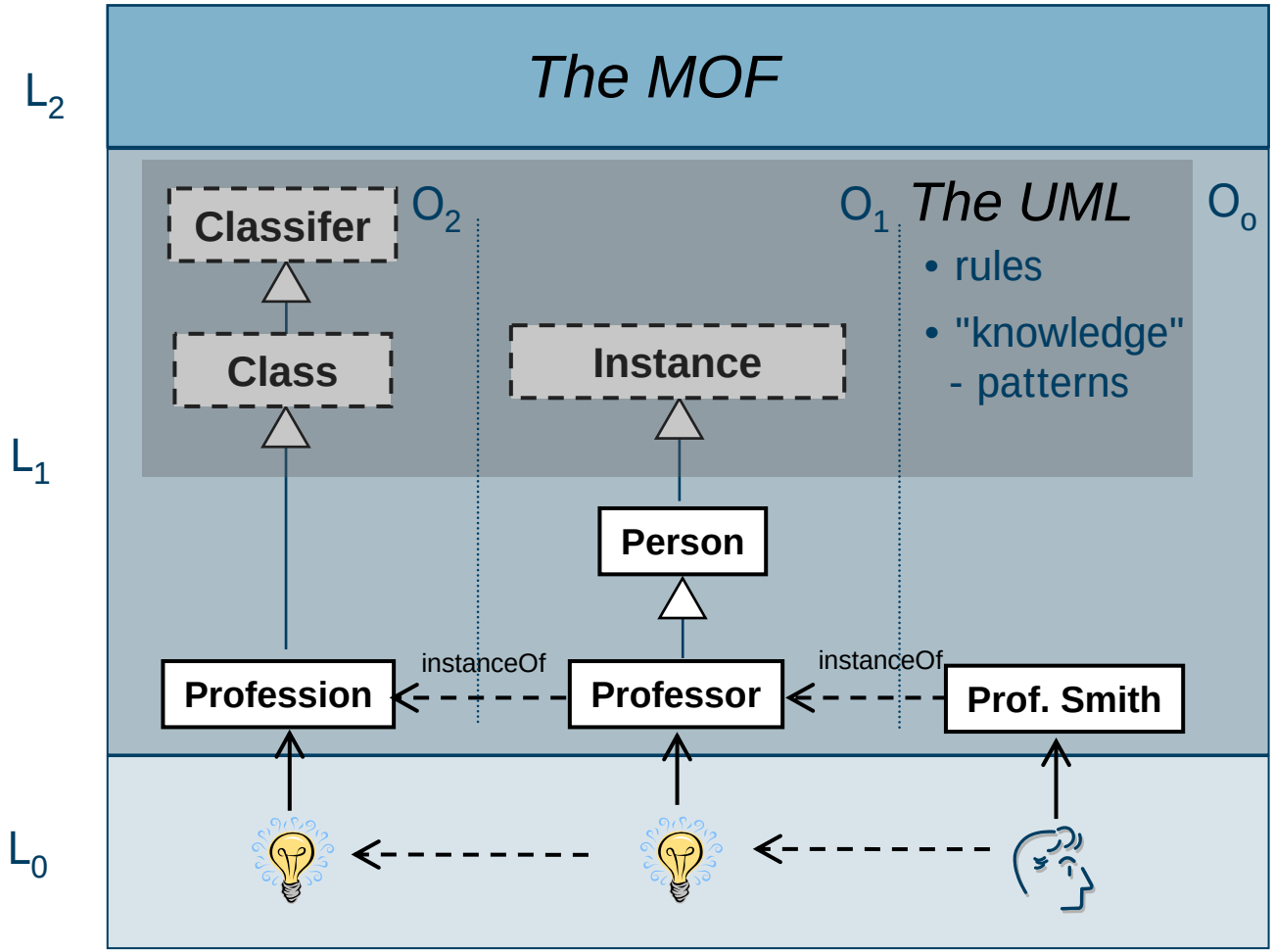

File Edit Navigate Search Project Run Sample Menu Window Help
AristaFlow IDE
AristaFlow Dimension Expl...
*Bank - Specification - Functional
Operation Specification Editor
withdraw
Component
Bank
ATM
Shopping Cart
Abstraction Level
Specification
Realization
Implementation
Projection
Structural
Functional
Behavioural
QoS
Create View
General
Name:
withdraw
Description:
An amount of money in a particular currency is withdrawn from an account.
Parameters:
accountID : Integer
amount : Real
Edit
Return:
Boolean
Edit
Edit Constraint
Name
Account-Exists-Precondition
Constraint
account->exists(accountID=id)
pre
OCL
Description
Make sure that account exists
Ok
Cancel
pre : account->exists(accountID=id)
OCL is valid
Add
Delete
Edit
Add
Delete
Edit
Add
Delete
Edit
Interactions
Bank - Specification - Functional


Java - Bank - Specification - Structural - Eclipse SDK
File Edit Navigate Search Project Run Sample Menu Window Help
Java
AristaFlow Dimension Explorer
Component
Bank
ATM
Shopping Cart
Abstraction Level
Specification
Realization
Implementation
Projection
Structural
Functional
Behavioural
QoS
Create View
Bank - Specification - Structural
Aristaflow Component Descriptor
Taxonomies
key
value
520000
Finance and Insurance
Add
Remove
Taxonomy NAICS
Node 110000 : Agriculture, Forestry, Fishing and Hunting
Node 210000 : Mining
Node 220000 : Utilities
Node 230000 : Construction
Node 310000 : Manufacturing
Node 320000 : Manufacturing
Node 330000 : Manufacturing
Node 420000 : Wholesale Trade
Node 440000 : Retail Trade
Node 450000 : Retail Trade
Node 480000 : Transportation and Warehousing
Node 490000 : Transportation and Warehousing
Node 510000 : Information
Node 520000 : Finance and Insurance
Node 521000 : Monetary Authorities - Central Bank
Node 522000 : Credit Intermediation and Related Activities
Node 522100 : Depository Credit Intermediation
Node 522200 : Nondepository Credit Intermediation
Node 522300 : Activities Related to Credit Intermediation
Node 523000 : Securities, Commodity Contracts, and Other Financ
Node 524000 : Insurance Carriers and Related Activities
Node 525000 : Funds, Trusts, and Other Financial Vehicles
Node 530000 : Real Estate and Rental and Leasing
Node 540000 : Professional, Scientific, and Technical Services
Node 550000 : Management of Companies and Enterprises
Taxonomy
UML class

OMG Four Layer Model



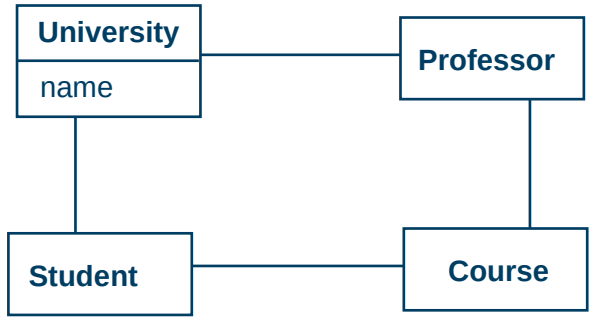
Orthographic Classification Architecture



[Atkinson and Kuehne, 1996 …]

Java OCL (JOCL)

A professor should only teach students at his university



OCL

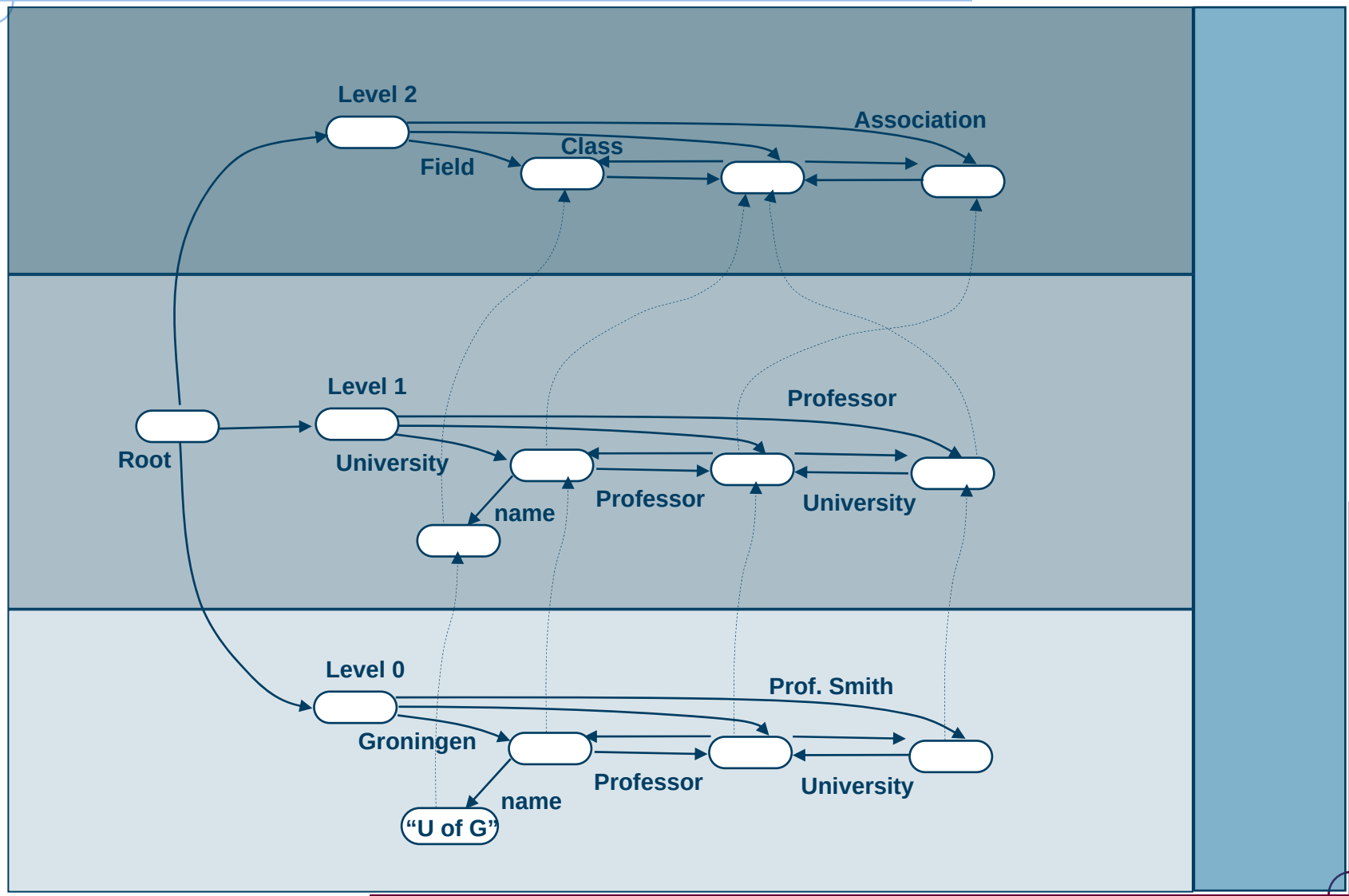
inv: self.course.student.university->asSet().
including(self.university)->asSet()->size() = 1

JOCL

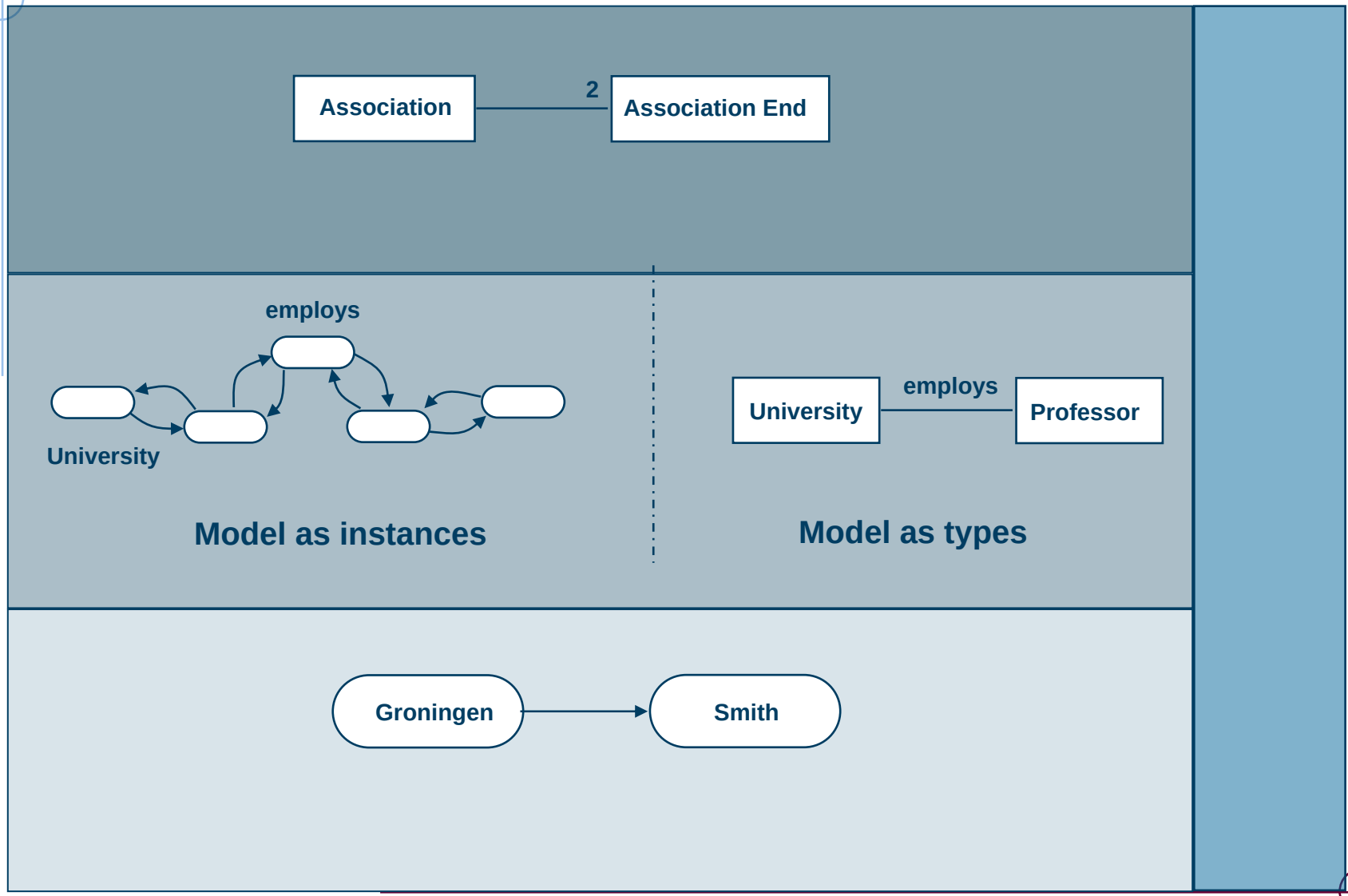
self.navigate("course").navigate("student").
navigate("university").asSet().including(self.
navigate("university")).asSet().size() == 1

S
M
M

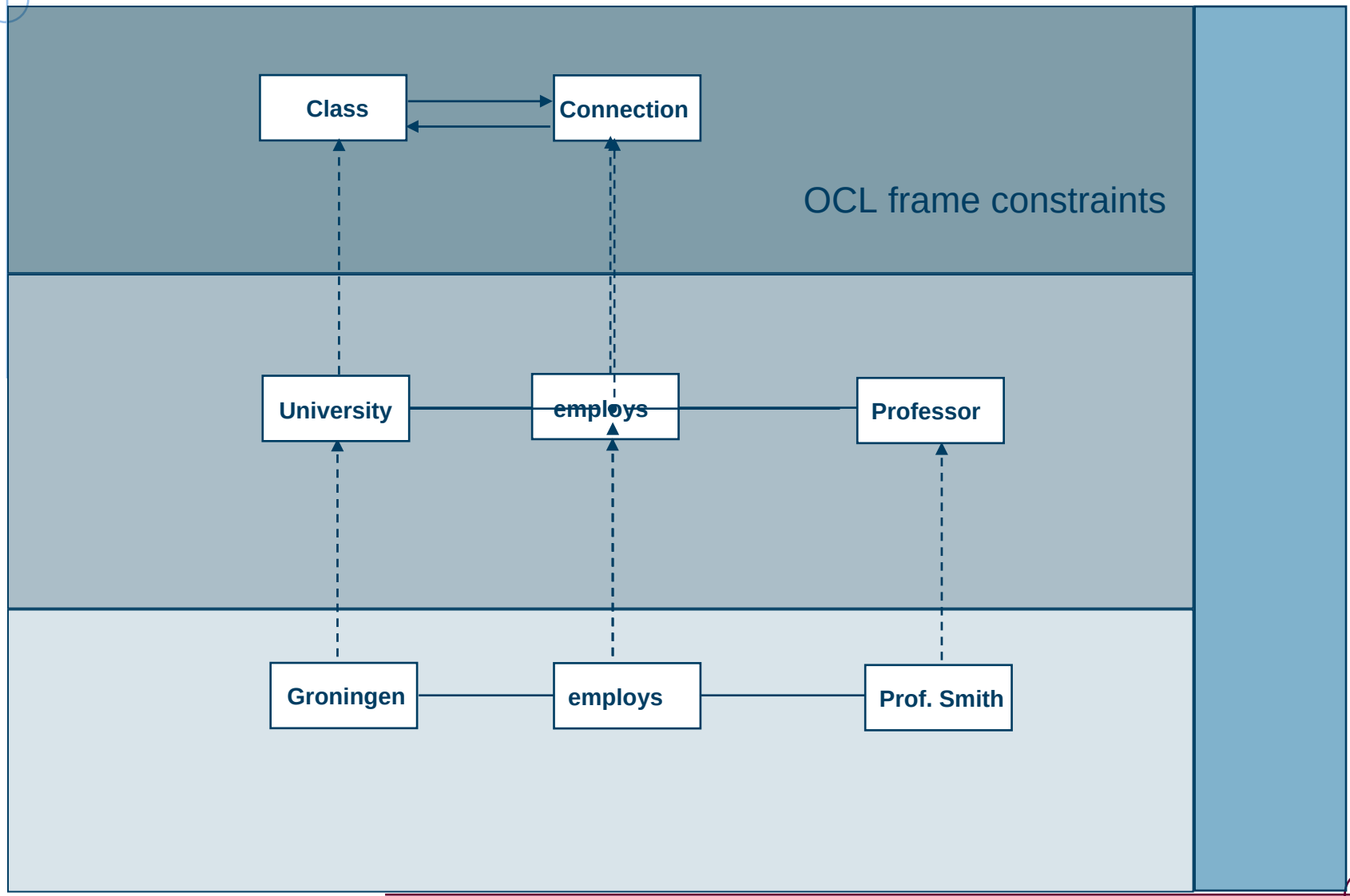
Repository Representation Format



AST and Surface Syntax Mismatch



Uniform Syntax



Current Editors and Visualizers

Projection / Abstraction	Structural	Functional	Behavioral	...
Specification	•UML class diagram •taxonomy •component description	•operation specification •activity template	•UML state chart •regular expression	...
Realization	•UML class diagram	•UML interaction diagram	•UML activity diagram	...
Implementation	•source code view	-	-	...
...	...	...	...	...

- Dimensions

  Composition, **Abstraction**, **Projection**, Version, Product Line, Metalevel, ...

IDE Status (Aristaflow Project)

- Completed
  - basic framework
  - calls editors
  - Component CRUD
- In Progress
  - persistence (bug fixes)
  - GUI improvements to ease navigation
- To Do
  - create dimensions, create perspectives dynamically
  - documentation
  - distributable packaging
  - synthesizers and checkers

Further Information

- Web Pages
  - swt.informatik.uni-mannheim.de
  - www.merobase.com
- Contact
  - atkinson@informatik.uni-mannheim.de