MDD in Practice

www.lukas-renggli.ch

Lukas Renggli

- **Software Engineer at Google**
  YouTube Video Analytics
- **SCG Alumni**
  Bachelor, Master and PhD
- **Open-Source Communities**
  Core-developer of Seaside
  Author of Magritte and Pier

Any sufficiently complicated program contains at least one meta-model.

Roadmap

1. **Protocol Buffers**
   Meta-data model for serialization
2. **OmniBrowser**
   Meta-model for tool building
3. **Google Web Toolkit**
   Model-driven web architecture
4. **Magritte**
   Generic meta-model

protobuf

Google's data interchange format

Protocol Buffers

Meta-data model for serialization

Describe your data format,
to share and evolve it.

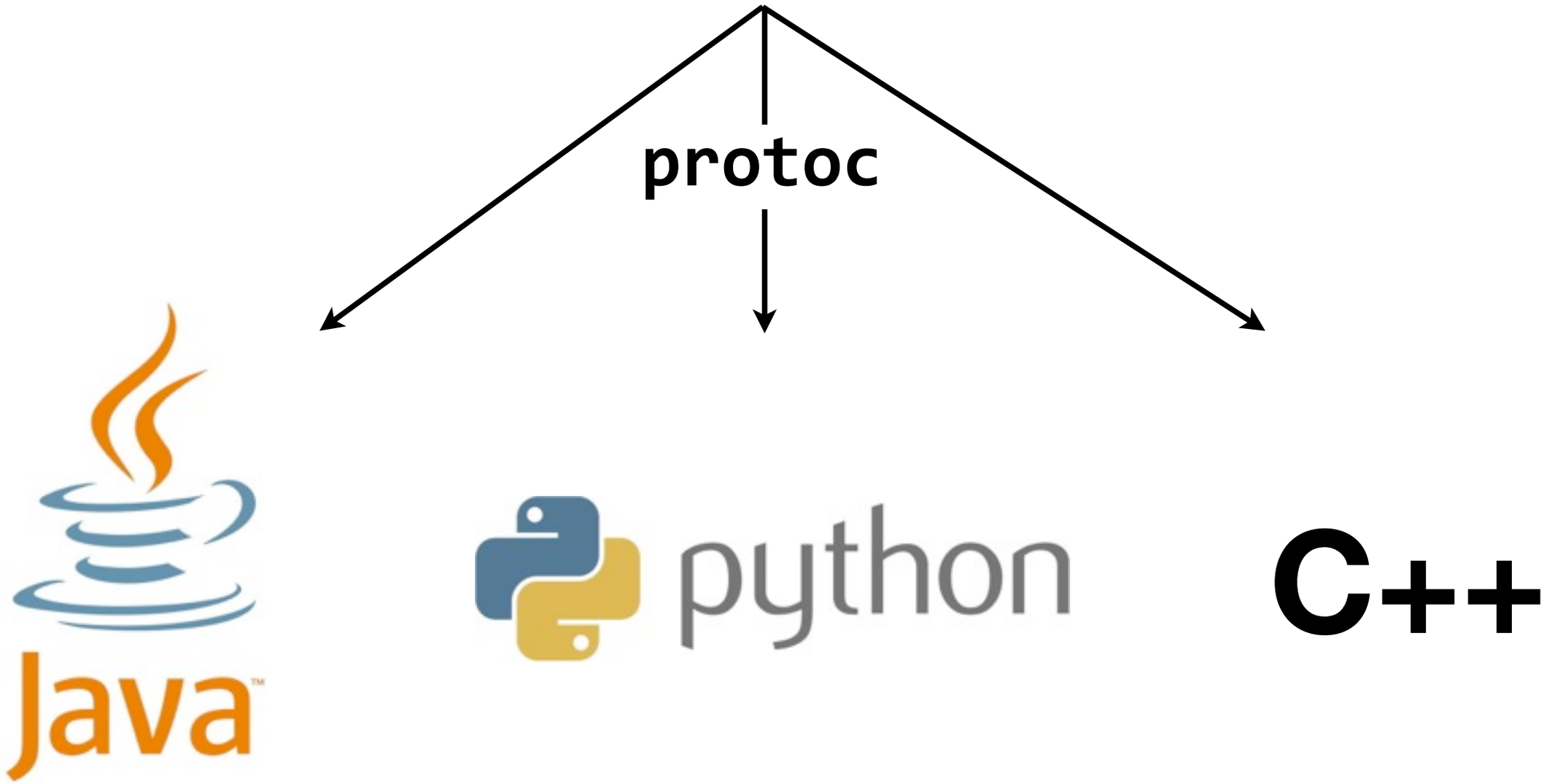
Protocol Buffers

- Encode structured data
- Language-neutral
- Platform-neutral
- Extensible
- Efficient

Protocol Buffer IDL

```
message Person {
    required int32 id = 1;
    required string name = 2;
    optional string email = 3;
}
```

```
message Person {
  required int32 id = 1;
  required string name = 2;
  optional string email = 3;
}
```



Serialize in C++

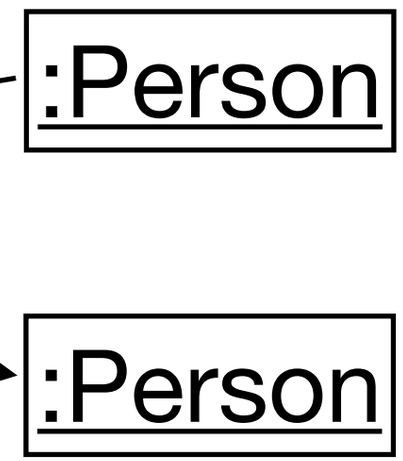
```
Person person;
person.set_name("John Doe");
person.set_id(1234);
person.set_email("jdoe@example.com");
fstream output("myfile", ios::out | ios::binary);
person.SerializeToOstream(&output);
```

Deserialize in Python

```
file = open("myfile", "rb")
person = Person()
person.ParseFromString(file.read())
file.close
print "Name: ", person.name
print "E-mail: ", person.email
```

```
message Person {
  required int32 id = 1;
  required string name = 2;
  optional string email = 3;
}
```


```
message Person {
  required int32 id = 1;
  required string name = 2;
  repeated string email = 3;
  repeated Phone phone = 4;
}
```



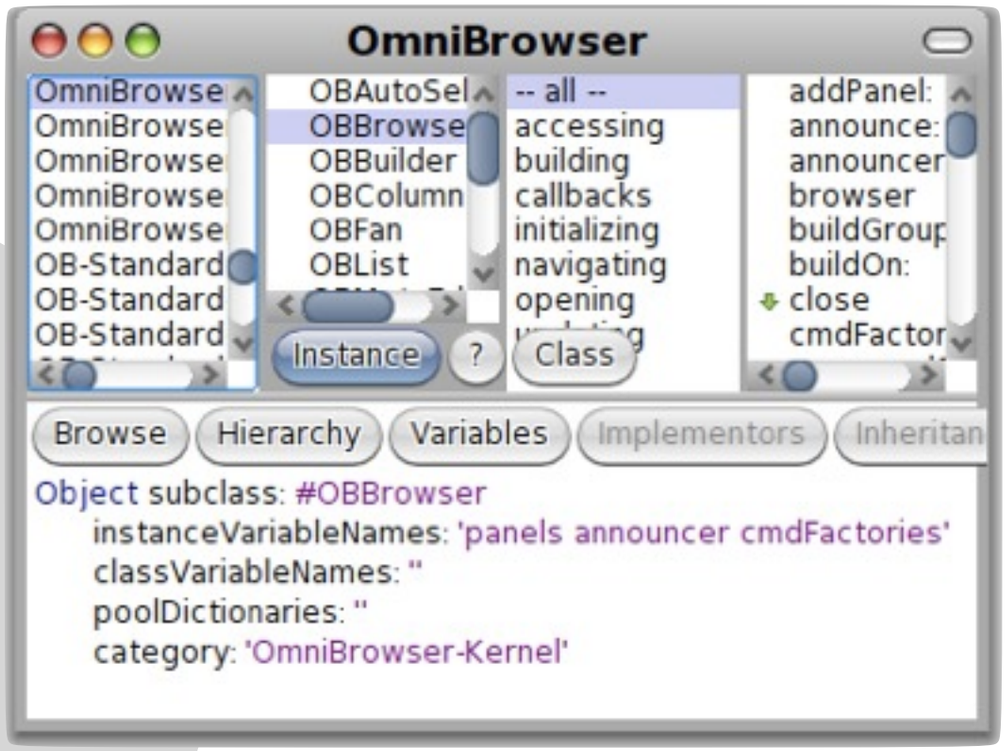
DEMO

Why not XML/JSON?

- Protocol Buffers are
  - 3—10 times smaller
  - 20—100 times faster
  - consistent code generators
  - less ambiguous
  - evolvable

Discussion

- Describe your data once, and use it across platforms and languages
- Evolution supported by design
- Used at Google for almost all
  - file formats (data storage)
  - remote procedure protocols (RPC)

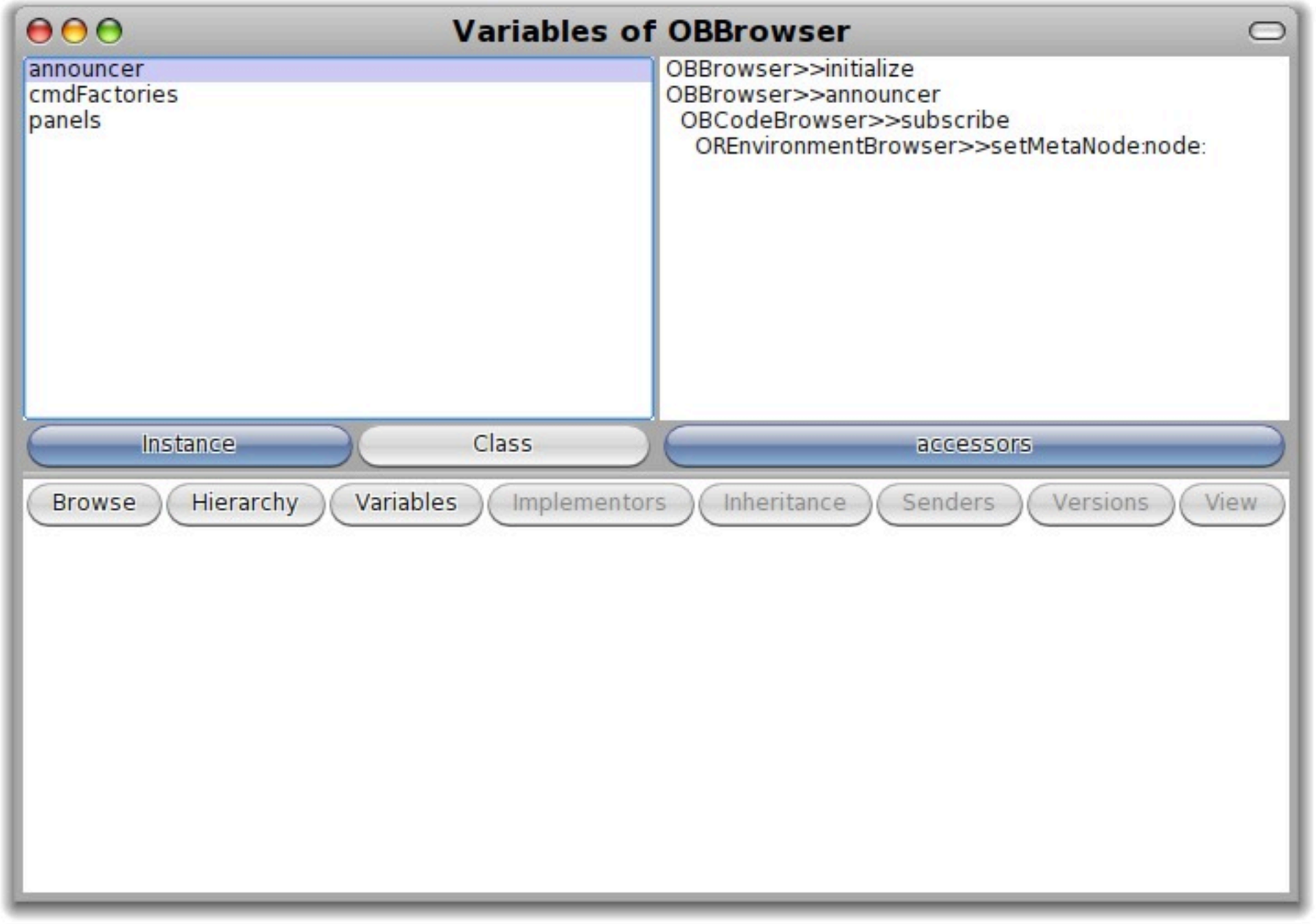


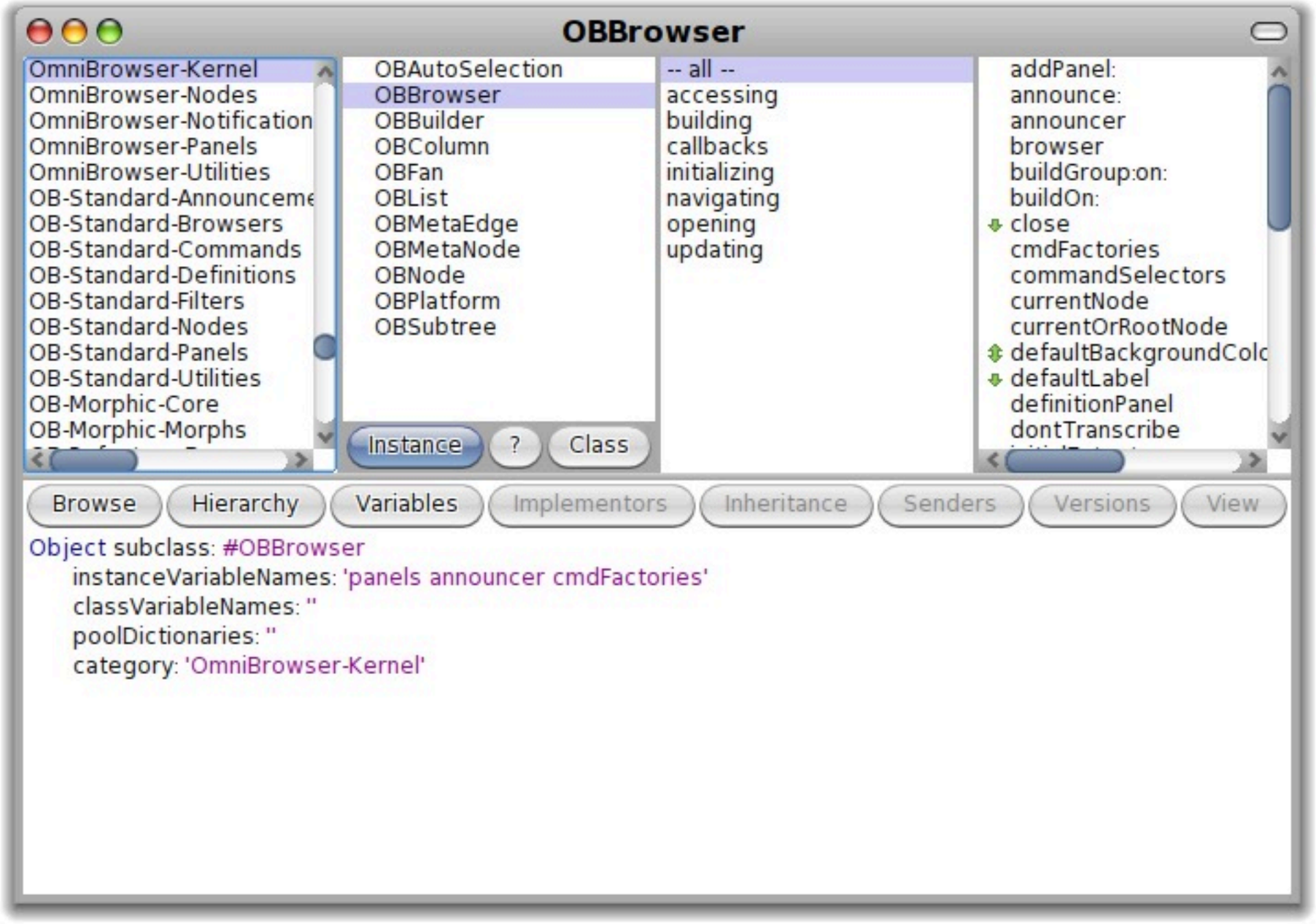
OmniBrowser

Meta-model for tool building

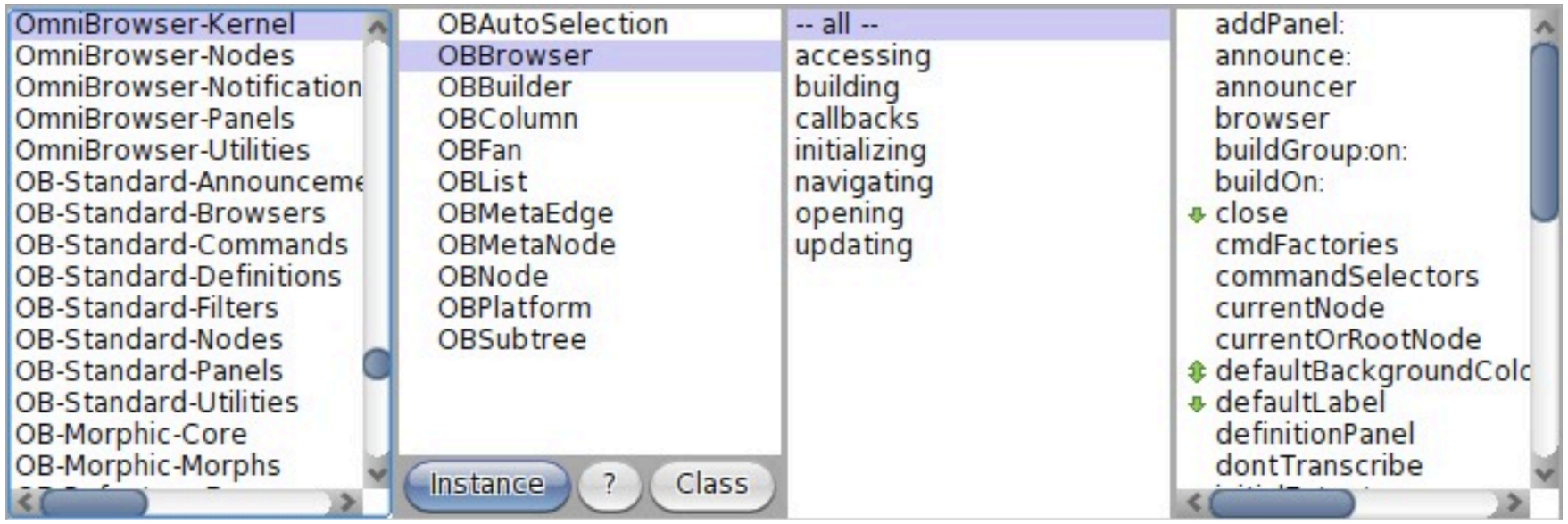
Create a first-class description of the domain.


OBBrowser Hierarchy
ProtoObject
Object
OBBrowser
OBCodeBrowser
OBHierarchyBrowser
OBInheritanceBrowser
OBListBrowser
OBImplementorsBrowser
OBHierarchyImplementor
OBMethodStringsBrowser
OBReferencesBrowser
OBSendersBrowser
OBHierarchySendersBro
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColor
defaultLabel
definitionPanel
dontTranscribe
Browse
Hierarchy
Variables
Implementors
Inheritance
Senders
Versions
View
Object subclass: #OBBrowser
instanceVariableNames: 'panels announcer cmdFactories'
classVariableNames: ''
poolDictionaries: ''
category: 'OmniBrowser-Kernel'

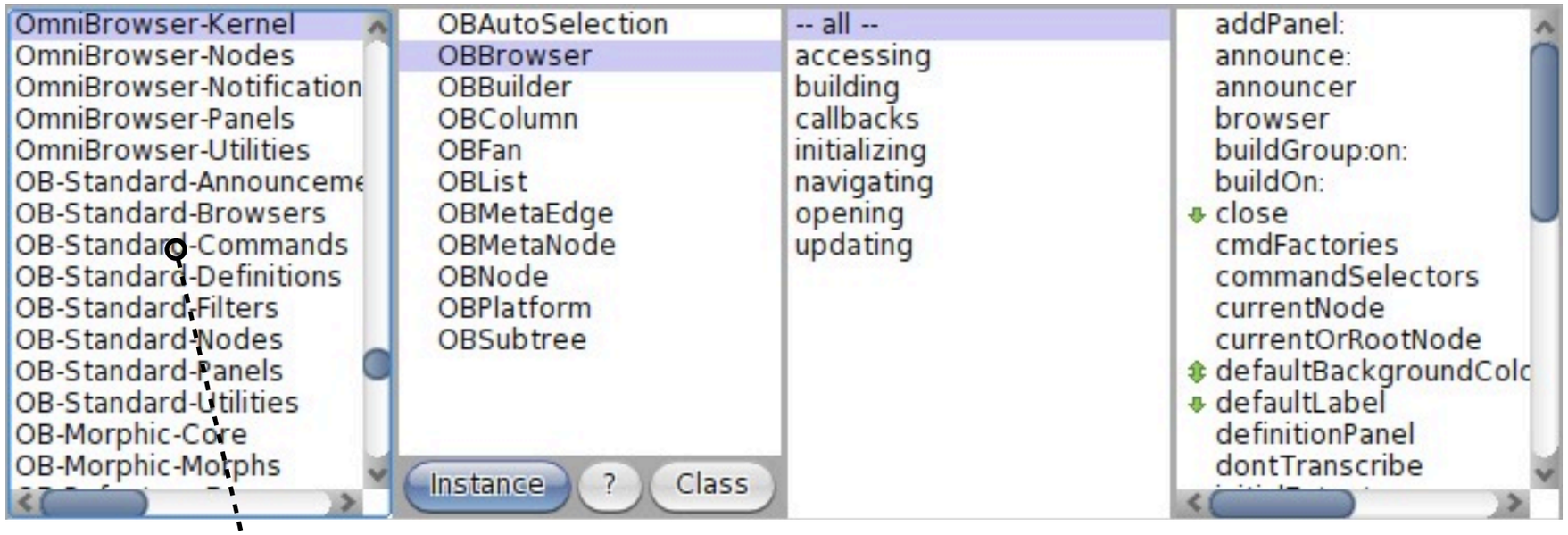

Variables of OBBrowser
announcer
cmdFactories
panels
OBBrowser>>initialize
OBBrowser>>announcer
OBCodeBrowser>>subscribe
ORENvironmentBrowser>>setMetaNode:node:
Instance
Class
accessors
Browse
Hierarchy
Variables
Implementors
Inheritance
Senders
Versions
View


OBBrowser
OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe
Browse
Hierarchy
Variables
Implementors
Inheritance
Senders
Versions
View
Object subclass: #OBBrowser
instanceVariableNames: 'panels announcer cmdFactories'
classVariableNames: ''
poolDictionaries: ''
category: 'OmniBrowser-Kernel'

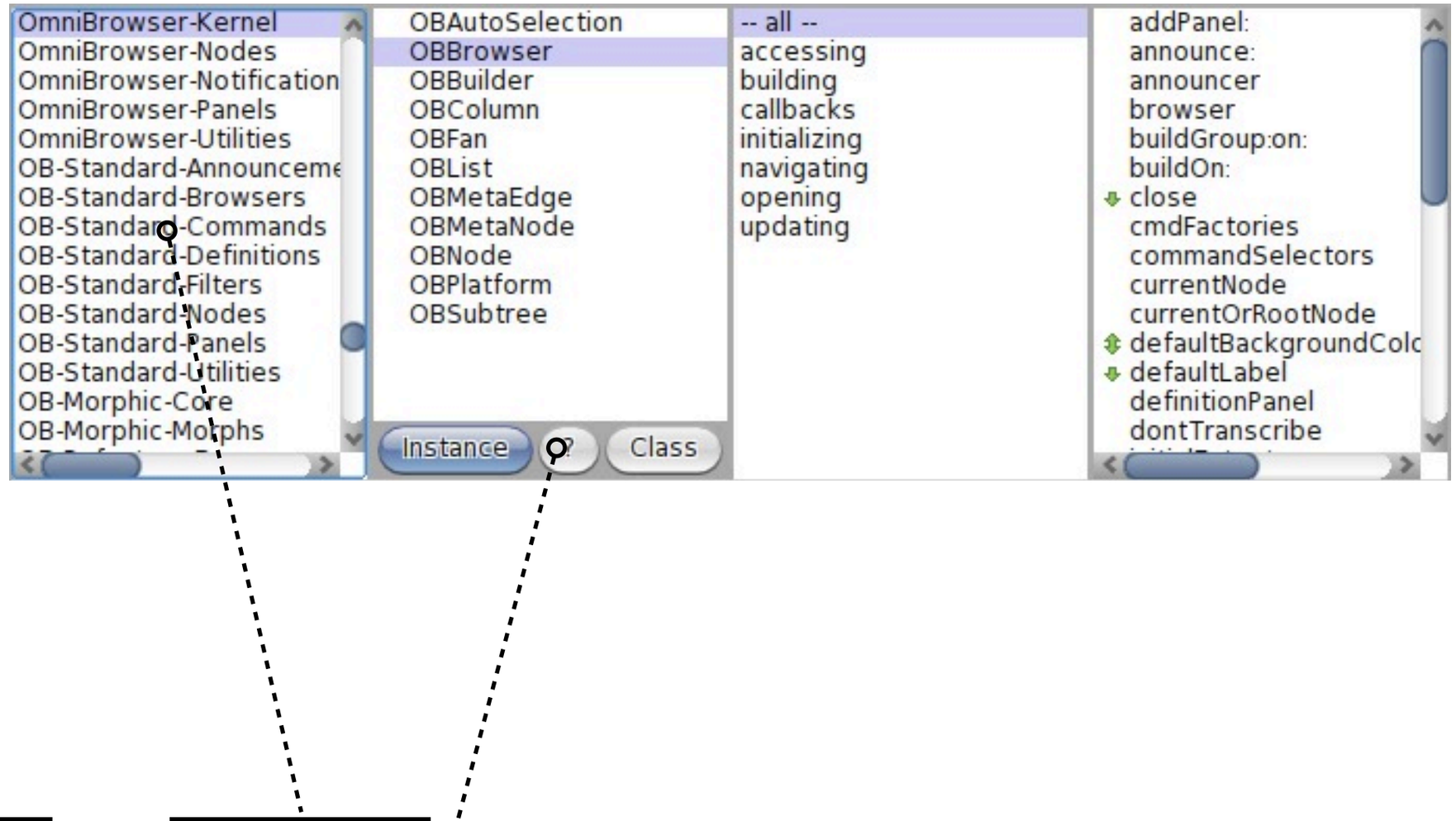
OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe

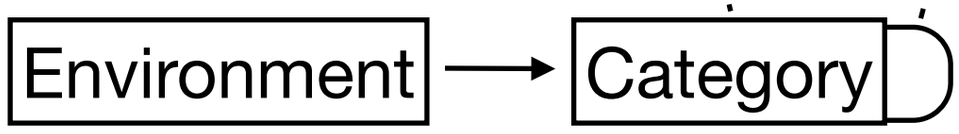


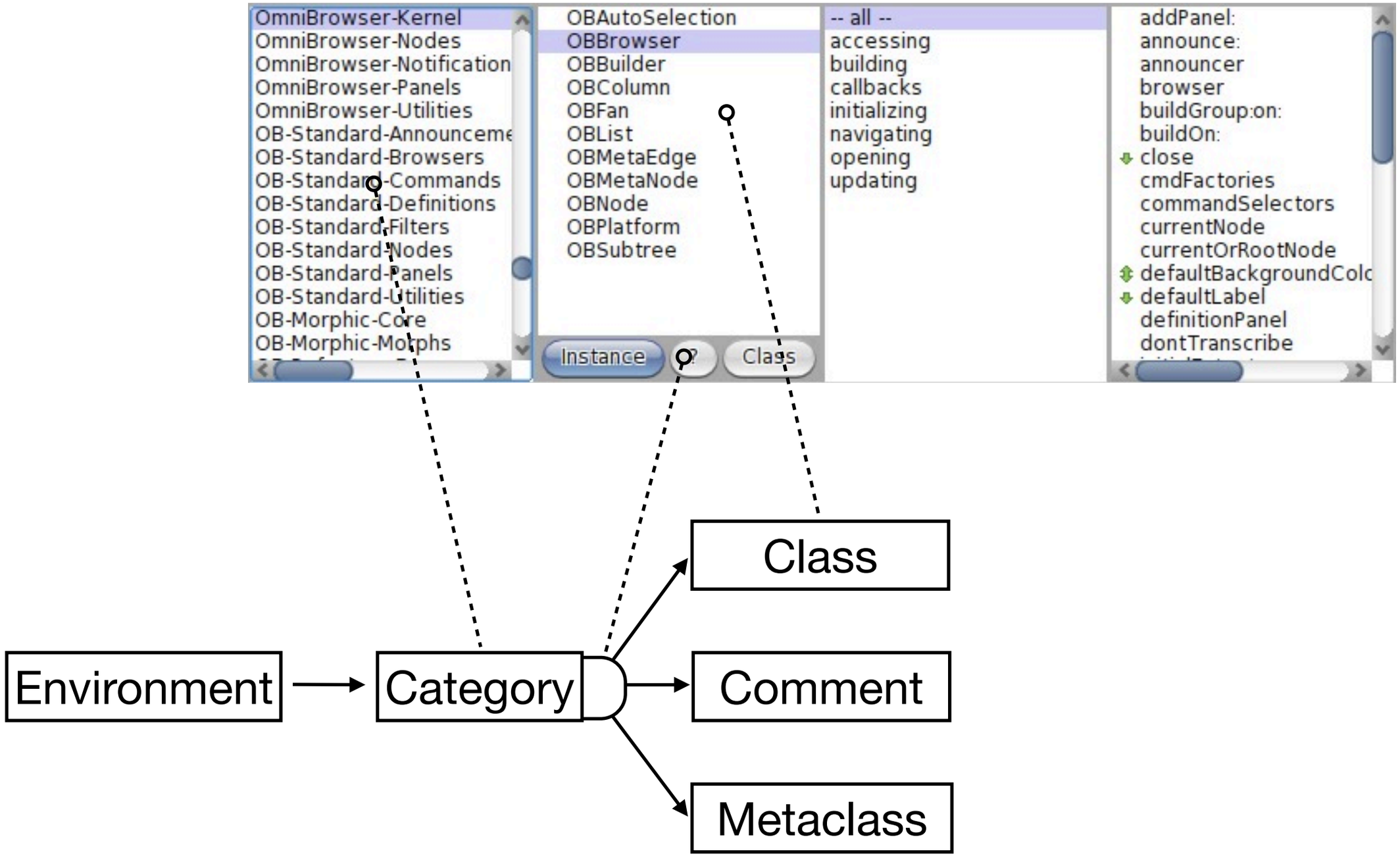
Environment

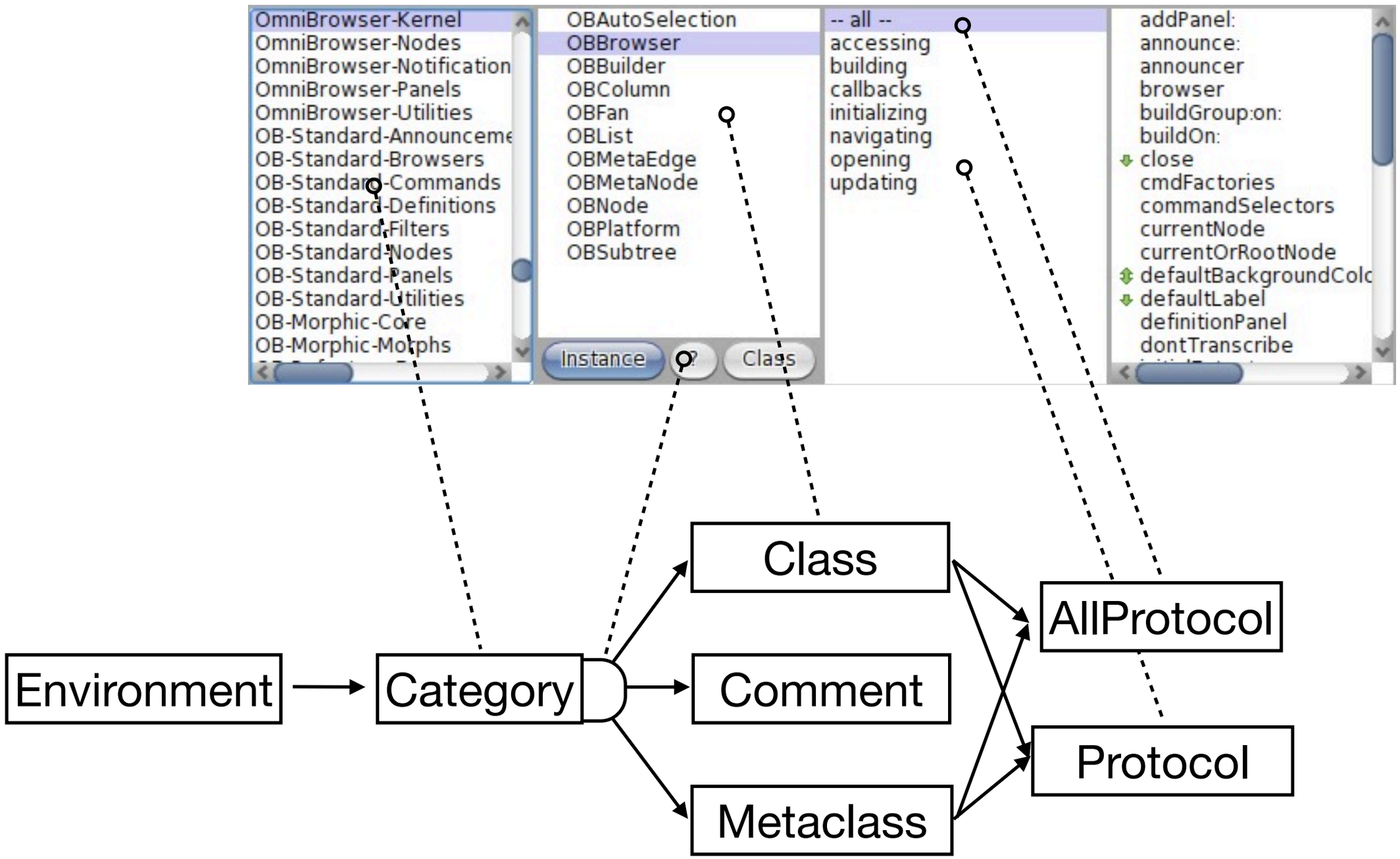


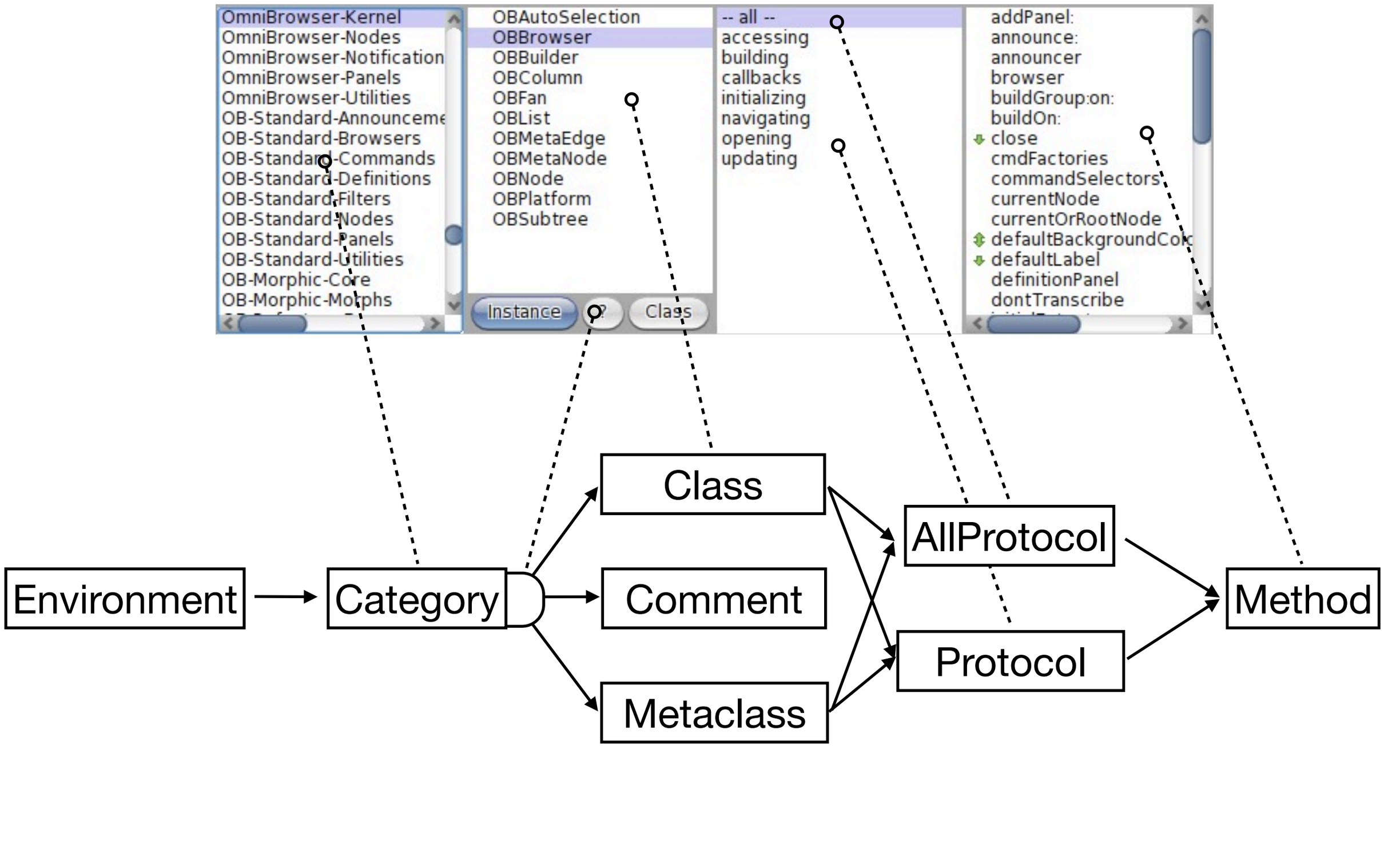
Environment → Category


OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe

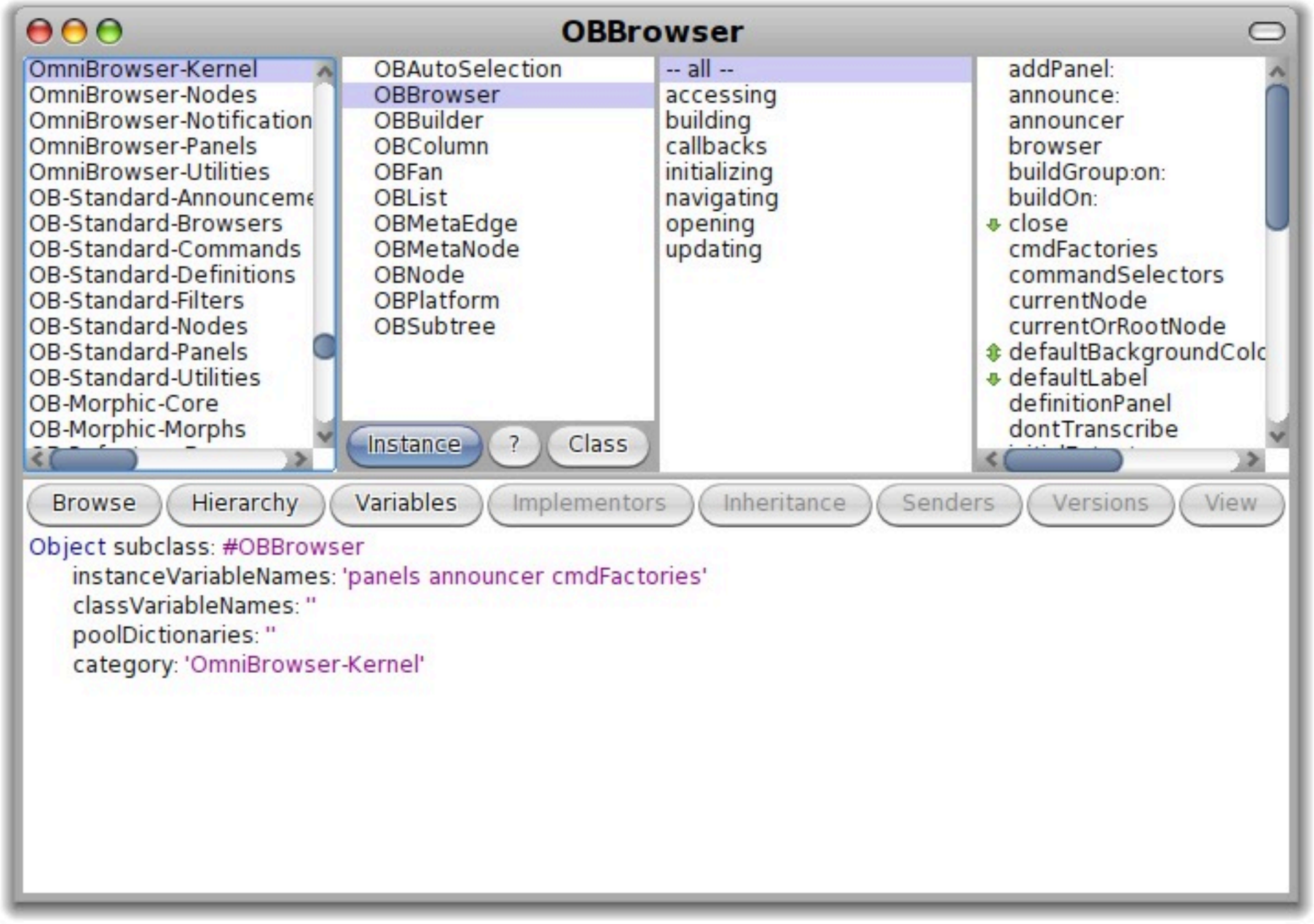

Environment
Category


OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe
Class
Environment
Category
Comment
Metaclass

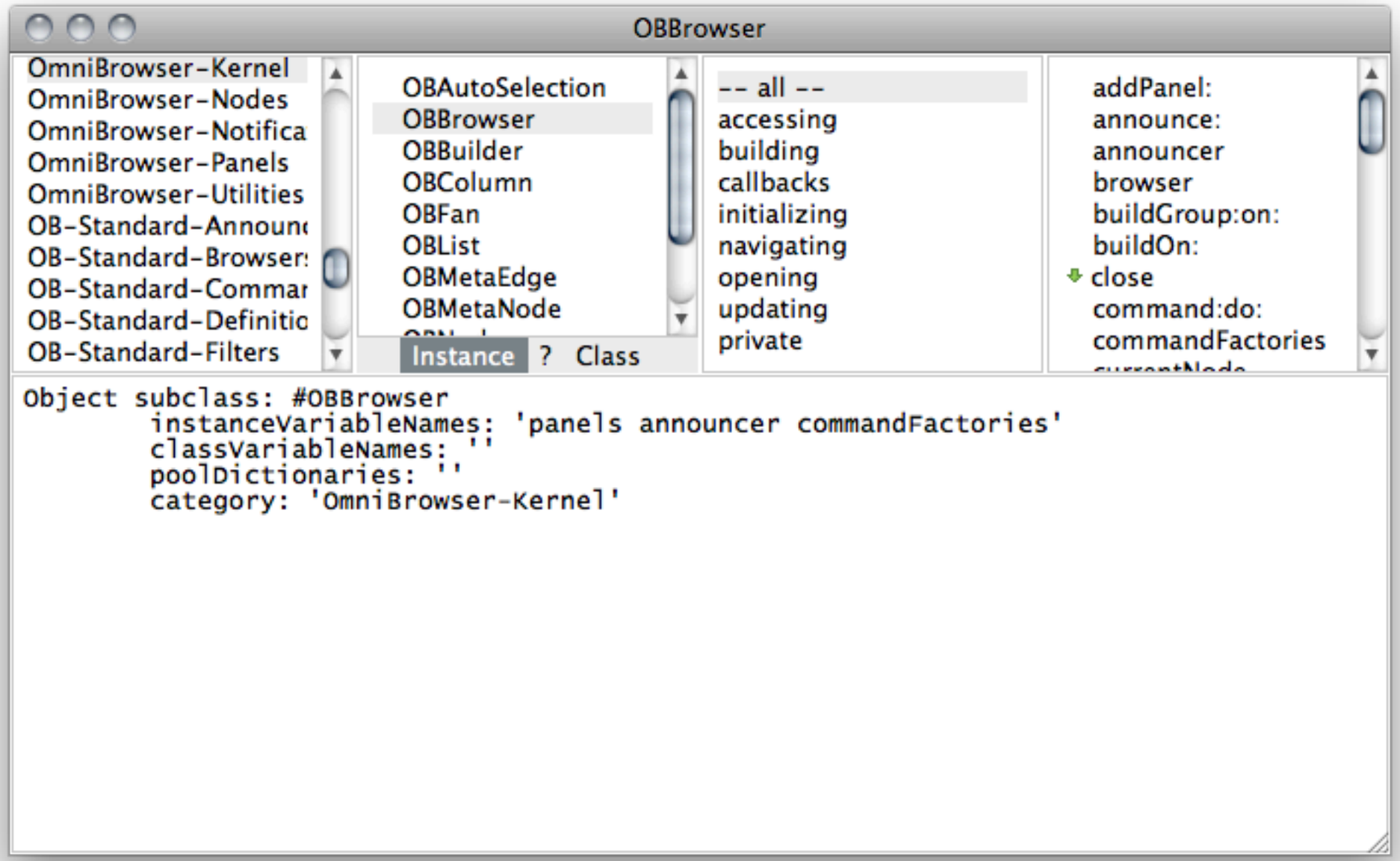

OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe
Environment
Category
Class
Comment
Metaclass
AllProtocol
Protocol


OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundCol
defaultLabel
definitionPanel
dontTranscribe
Environment
Category
Class
Comment
Metaclass
AllProtocol
Protocol
Method

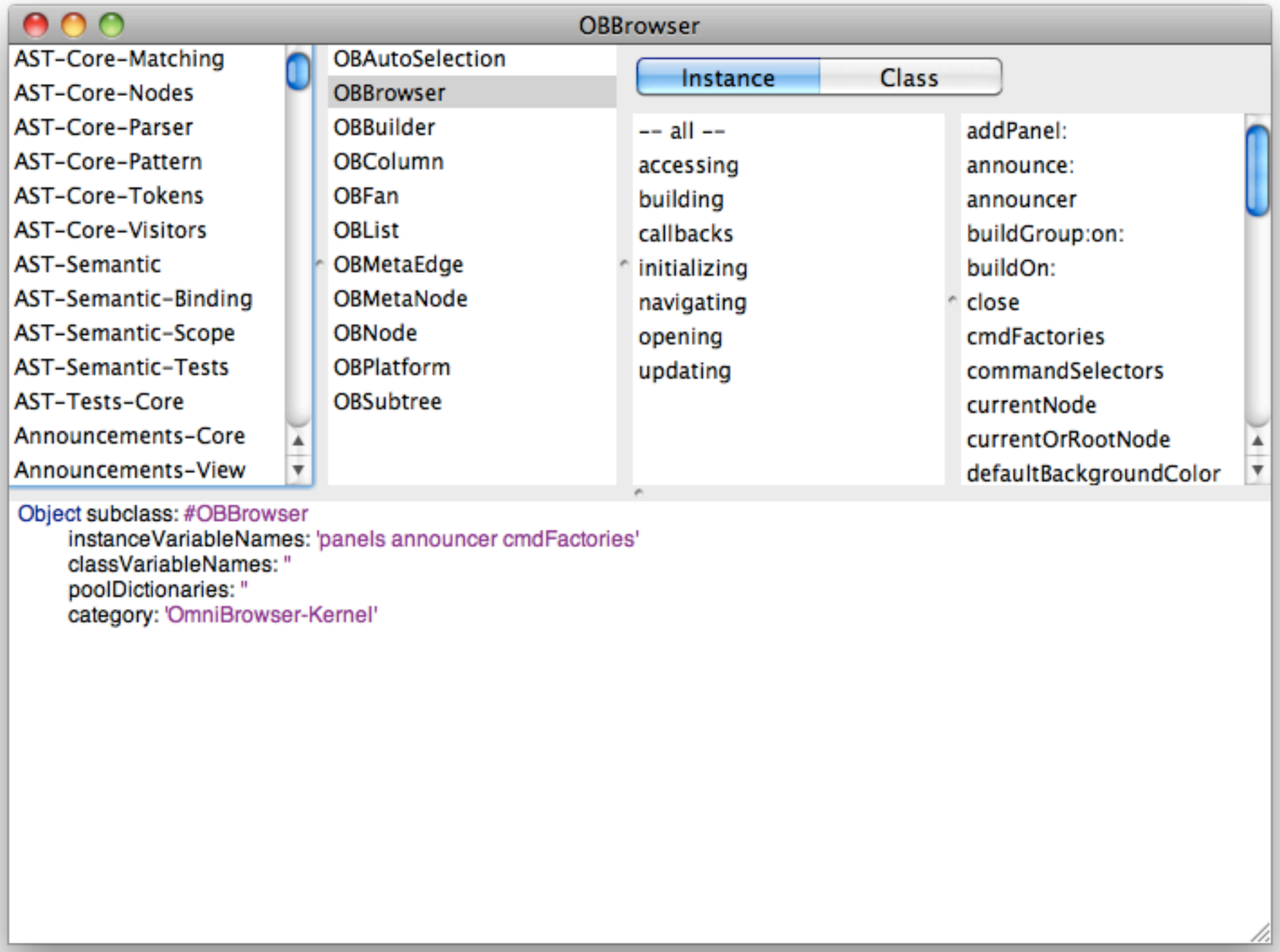
OB-Morph


OBBrowser
OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notification
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announceme
OB-Standard-Browsers
OB-Standard-Commands
OB-Standard-Definitions
OB-Standard-Filters
OB-Standard-Nodes
OB-Standard-Panels
OB-Standard-Utilities
OB-Morphic-Core
OB-Morphic-Morphs
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColo
defaultLabel
definitionPanel
dontTranscribe
Browse
Hierarchy
Variables
Implementors
Inheritance
Senders
Versions
View
Object subclass: #OBBrowser
instanceVariableNames: 'panels announcer cmdFactories'
classVariableNames: ''
poolDictionaries: ''
category: 'OmniBrowser-Kernel'

OB-Web


OBBrowser
OmniBrowser-Kernel
OmniBrowser-Nodes
OmniBrowser-Notifica
OmniBrowser-Panels
OmniBrowser-Utilities
OB-Standard-Announc
OB-Standard-Browser
OB-Standard-Commar
OB-Standard-Definitic
OB-Standard-Filters
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
Instance
?
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
private
addPanel:
announce:
announcer
browser
buildGroup:on:
buildOn:
close
command:do:
commandFactories
Object subclass: #OBBrowser
instanceVariableNames: 'panels announcer commandFactories'
classVariableNames: ''
poolDictionaries: ''
category: 'OmniBrowser-Kernel'

OB-Mars


OBBrowser
AST-Core-Matching
AST-Core-Nodes
AST-Core-Parser
AST-Core-Pattern
AST-Core-Tokens
AST-Core-Visitors
AST-Semantic
AST-Semantic-Binding
AST-Semantic-Scope
AST-Semantic-Tests
AST-Tests-Core
Announcements-Core
Announcements-View
OBAutoSelection
OBBrowser
OBBuilder
OBColumn
OBFan
OBList
OBMetaEdge
OBMetaNode
OBNode
OBPlatform
OBSubtree
Instance
Class
-- all --
accessing
building
callbacks
initializing
navigating
opening
updating
addPanel:
announce:
announcer
buildGroup:on:
buildOn:
close
cmdFactories
commandSelectors
currentNode
currentOrRootNode
defaultBackgroundColor
Object subclass: #OBBrowser
instanceVariableNames: 'panels announcer cmdFactories'
classVariableNames: ''
poolDictionaries: ''
category: 'OmniBrowser-Kernel'

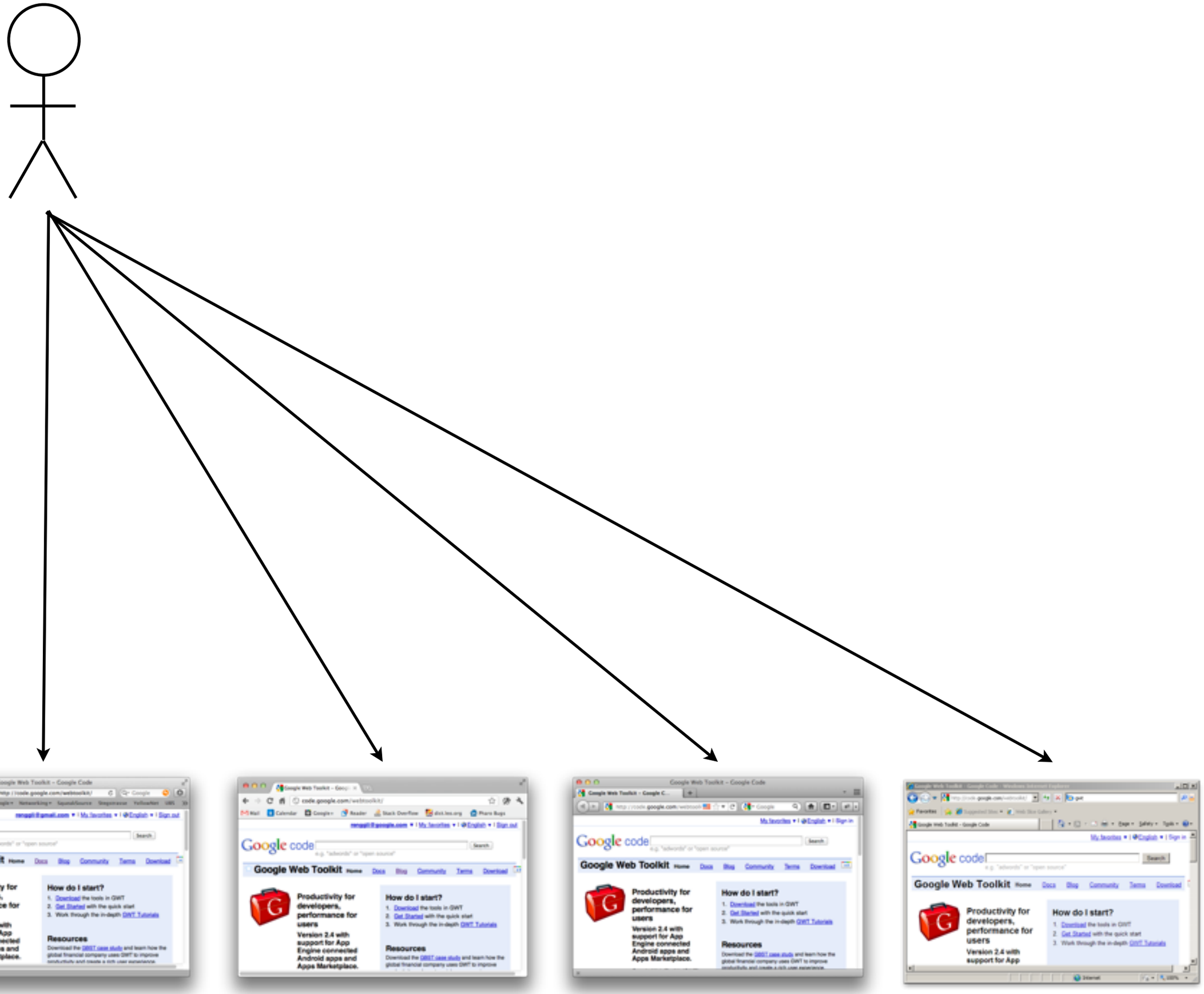
Discussion

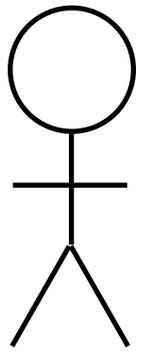
- Model the concepts of your domain
- Domain-specific reflection
- UI independent
- Can be limiting or slow
- People might get meta-confused

Google Web Toolkit

Model-driven web architecture

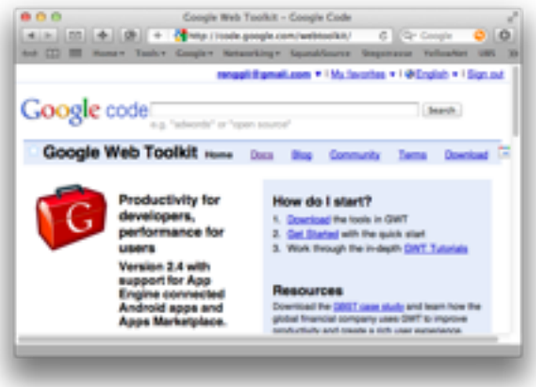
Write your application in Java,
run your application in JavaScript

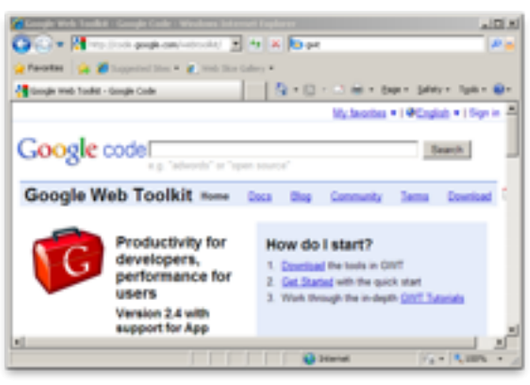

Google Web Toolkit – Google Code
Google code
Google Web Toolkit
Productivity for developers, performance for users
How do I start?
Resources



```
public class Main implements EntryPoint {
  public void onModuleLoad() {
    Label label = new Label("Hello World");
    RootPanel.add(label);
  }
}
```

platform independent model



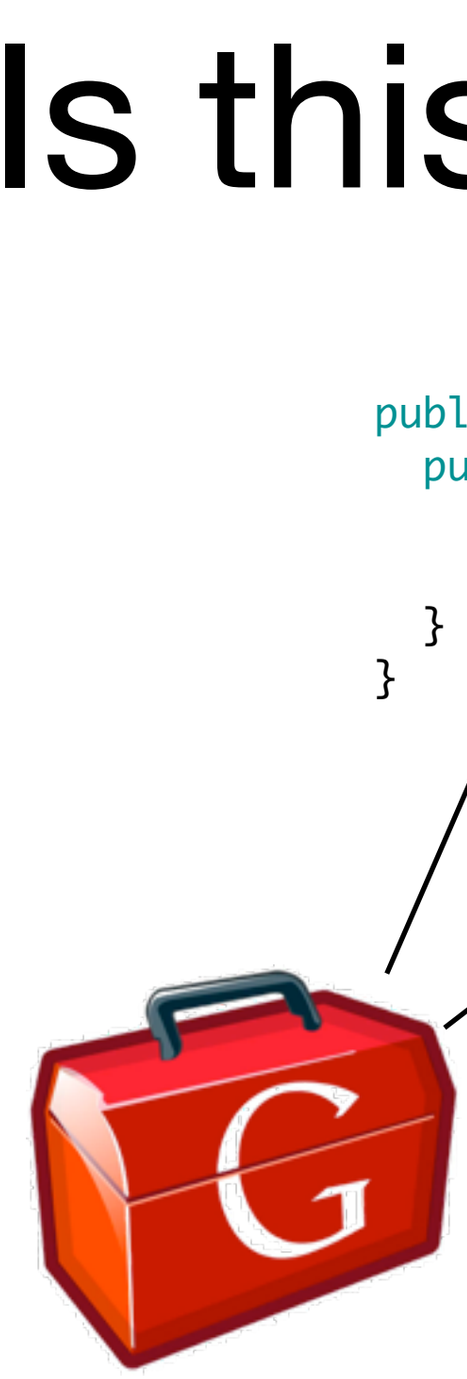


Is this really a model?

```
public class Main implements EntryPoint {
  public void onModuleLoad() {
    Label label = new Label(“Hello World”);
    RootPanel.add(label);
  }
}
```

Is this really a model?

```
public class Main implements EntryPoint {
  public void onModuleLoad() {
    Label label = new Label(“Hello World”);
    RootPanel.add(label);
  }
}
```



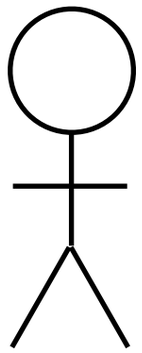
High-level
Widgets

Is this really a model?

```
public class Main implements EntryPoint {
  public void onModuleLoad() {
    Label label = new Label(“Hello World”);
    RootPanel.add(label);
  }
}
```

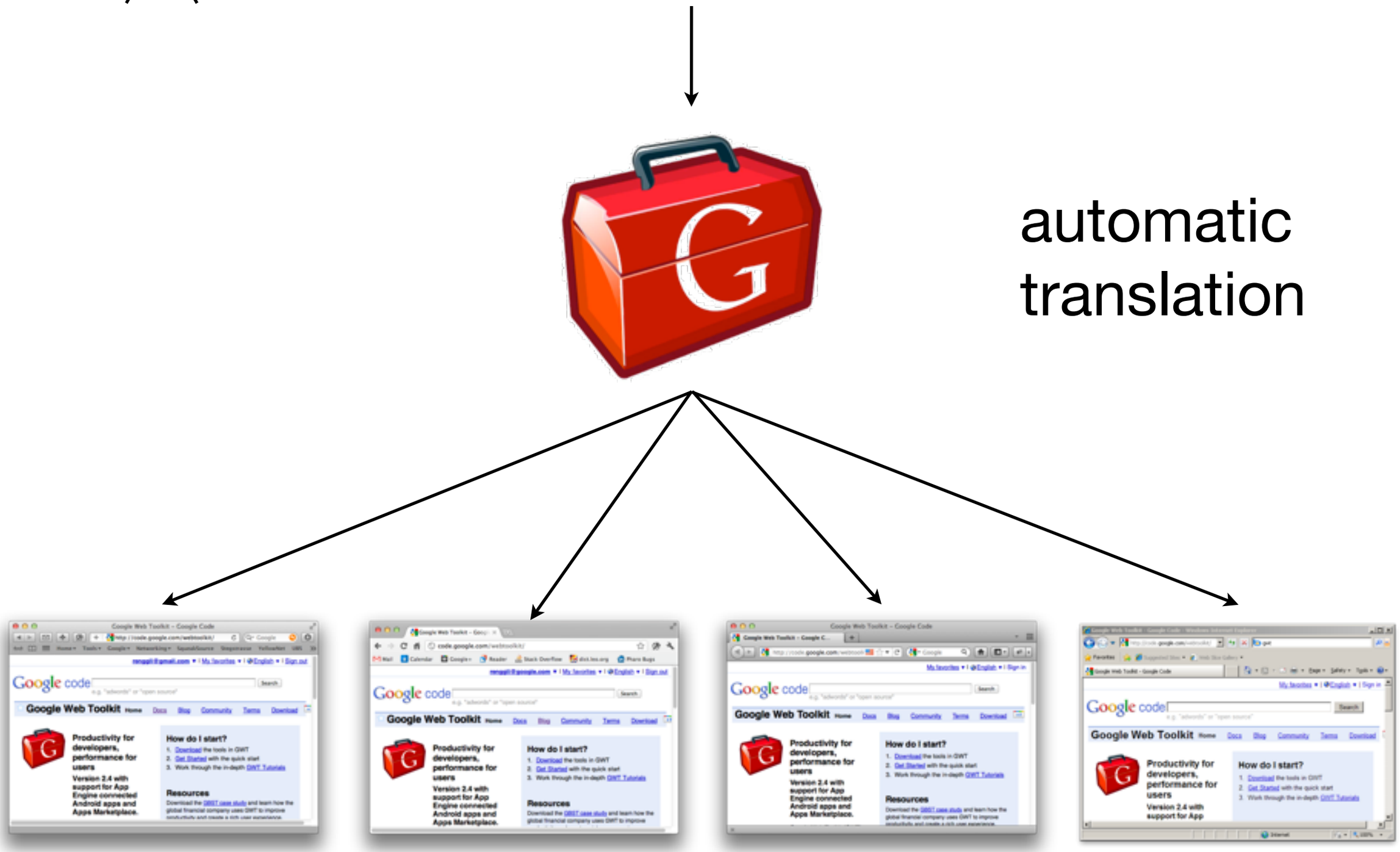
High-level Widgets

Description Language



```
public class Main implements EntryPoint {
  public void onModuleLoad() {
    Label label = new Label("Hello World");
    RootPanel.add(label);
  }
}
```

platform independent model



automatic translation

GWT Translation

	Mozilla	Chrome	Safari	IE	...
English					
French					
German					
...					

GWT Translation

	Mozilla	Chrome	Safari	IE	...
English					
French					
German					
...					

Discussion

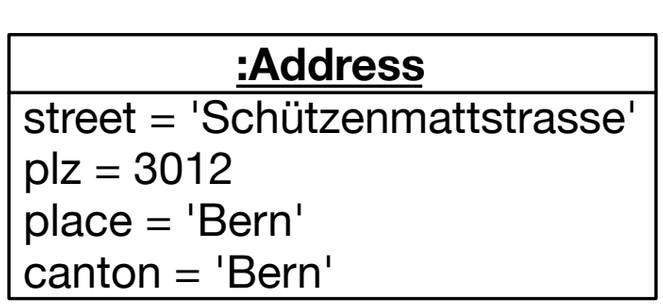
- Write JavaScript using a high-level widget library in a well defined (statically typed) language
- Translate and optimize code towards specific browsers
- Debugging, kind of works but you don’t want to know how

Magritte

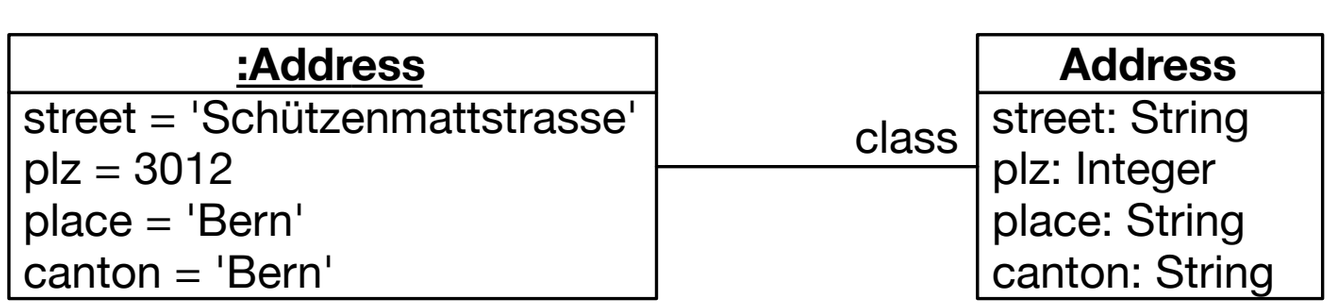
Generic Meta Meta-Model

Describe your objects once,
use the descriptions everywhere.

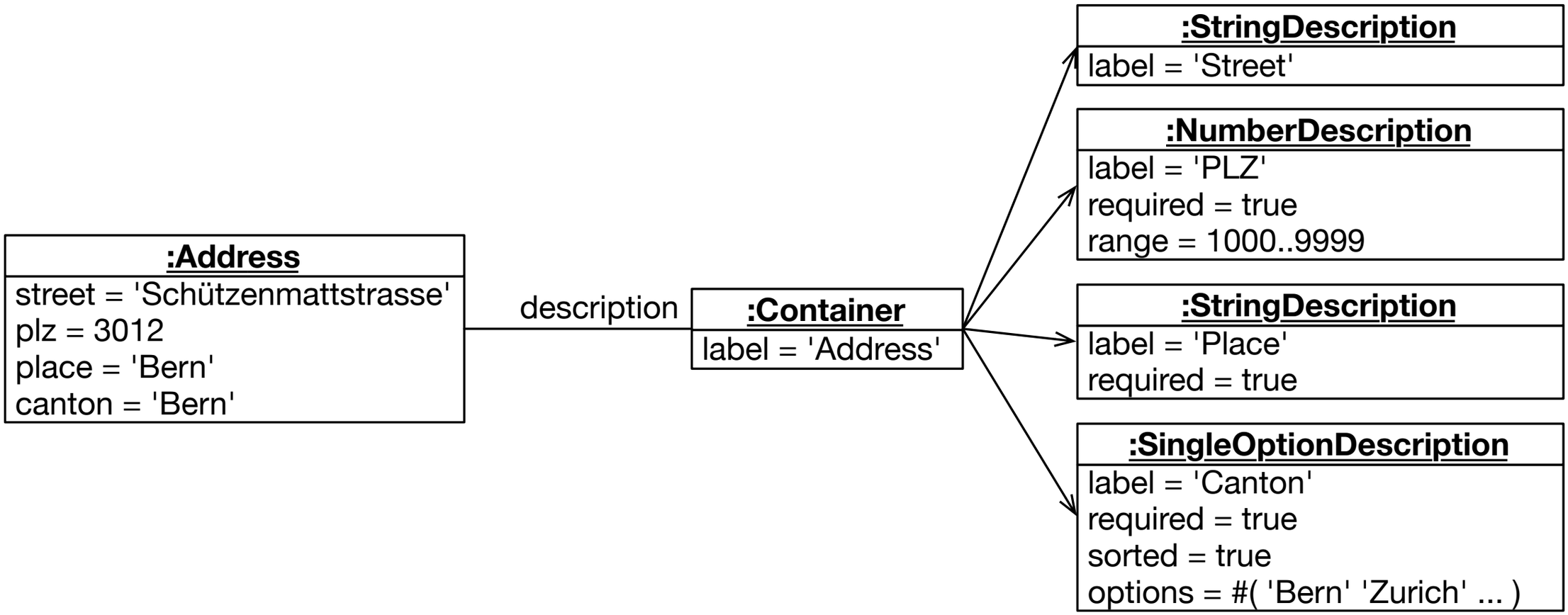
Address Object



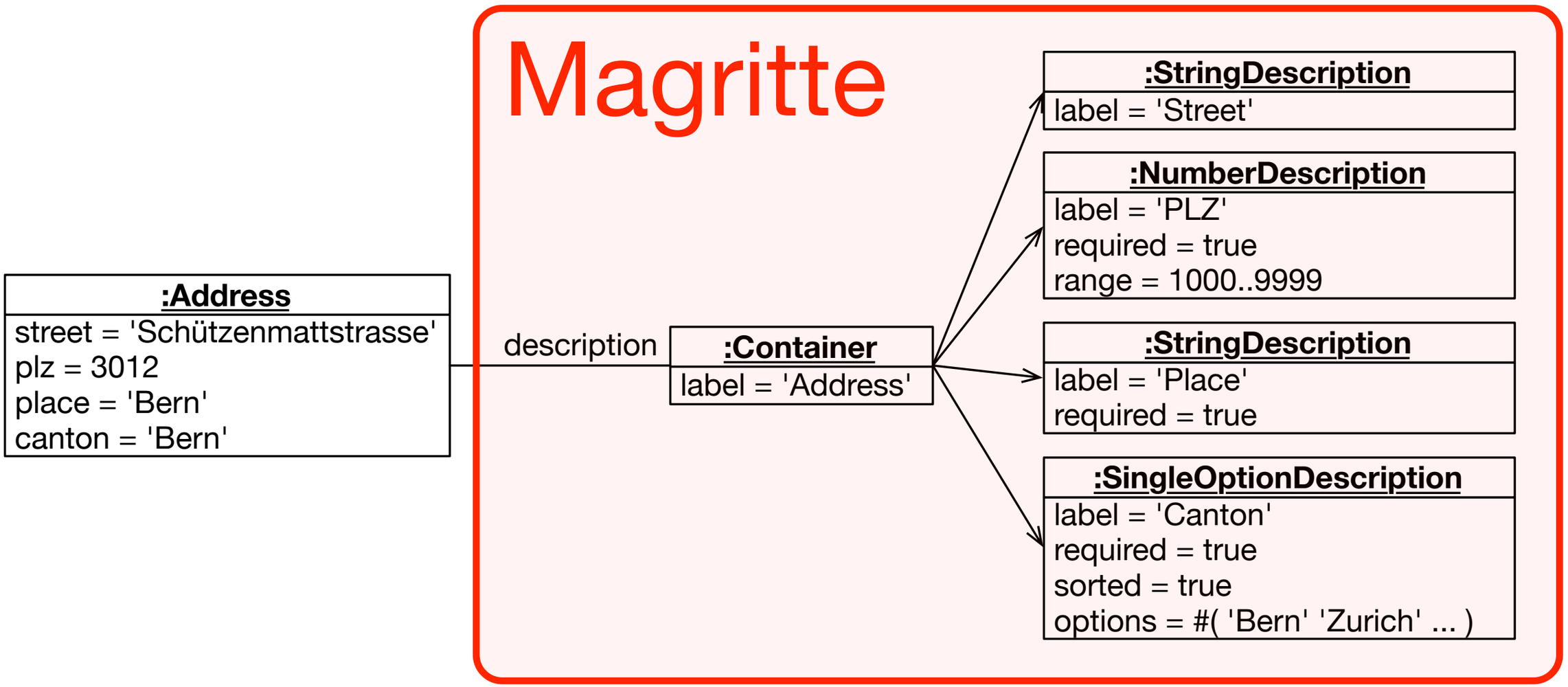
Address Class



Address Description

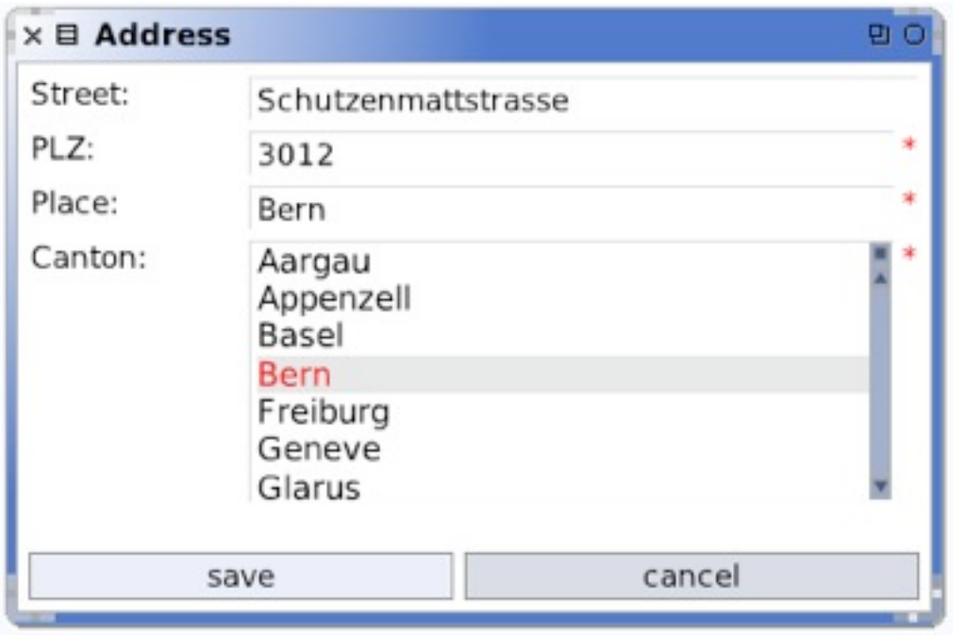


Address Description



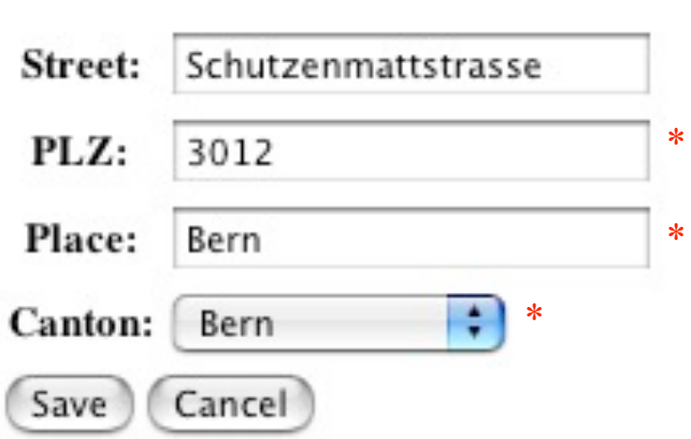
Morphic Rendering

```
result := anAddress asMorph
    addButtons;
    addWindow;
    callInWorld
```



Seaside Rendering

```
result := self call: (anAddress asComponent
    addValidatedForm;
    yourself)
```



Other Applications

- Viewer building
- Editor building
- Report building
- Documentation
- Data validation
- Query processing
- Object filtering
- Object serialization
- Object copying
- Object indexing
- Object initialization
- Object extension
- Object adaption
- Object customization
- Code generation

and much more ...

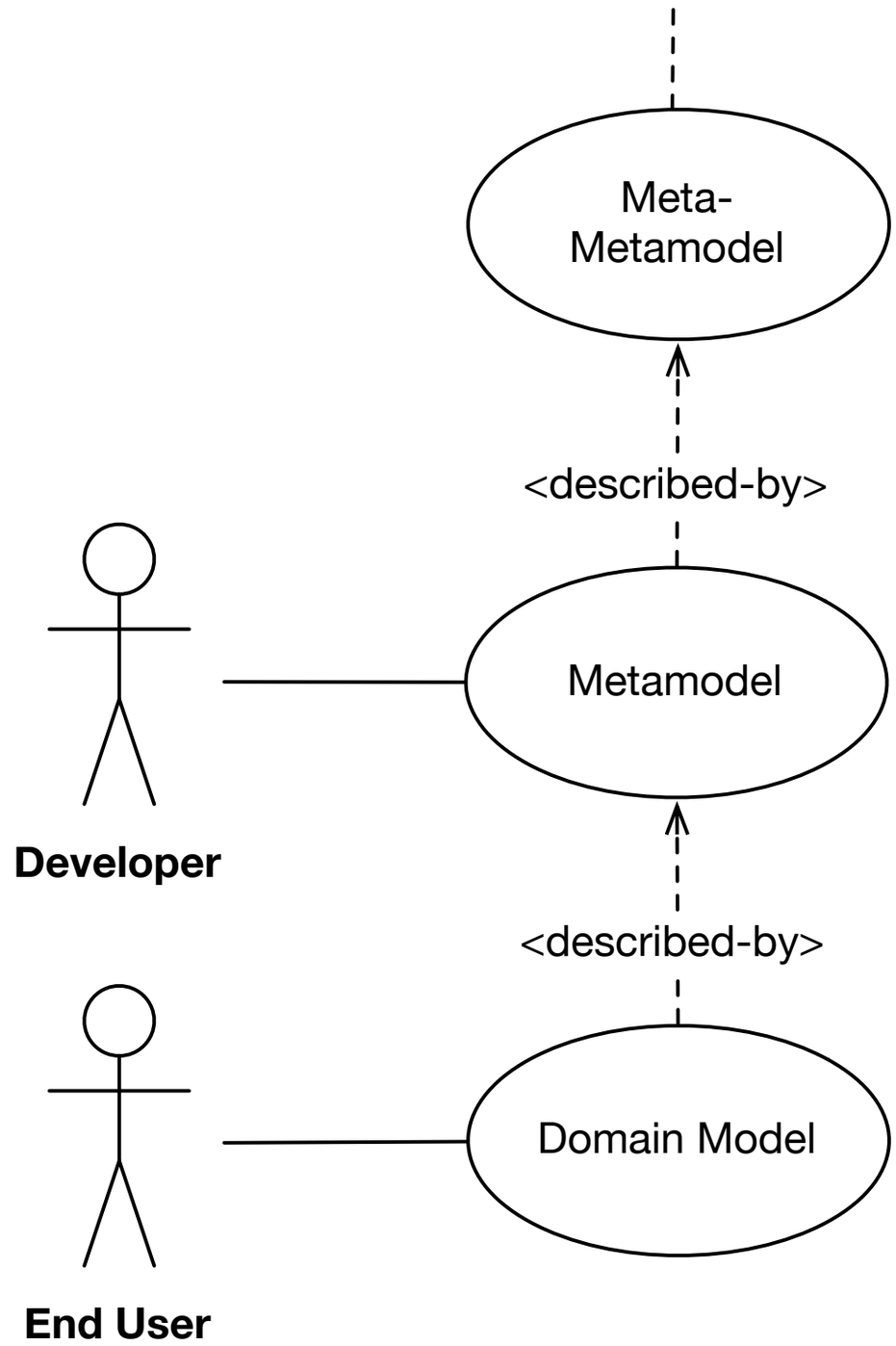
Why is it useful?

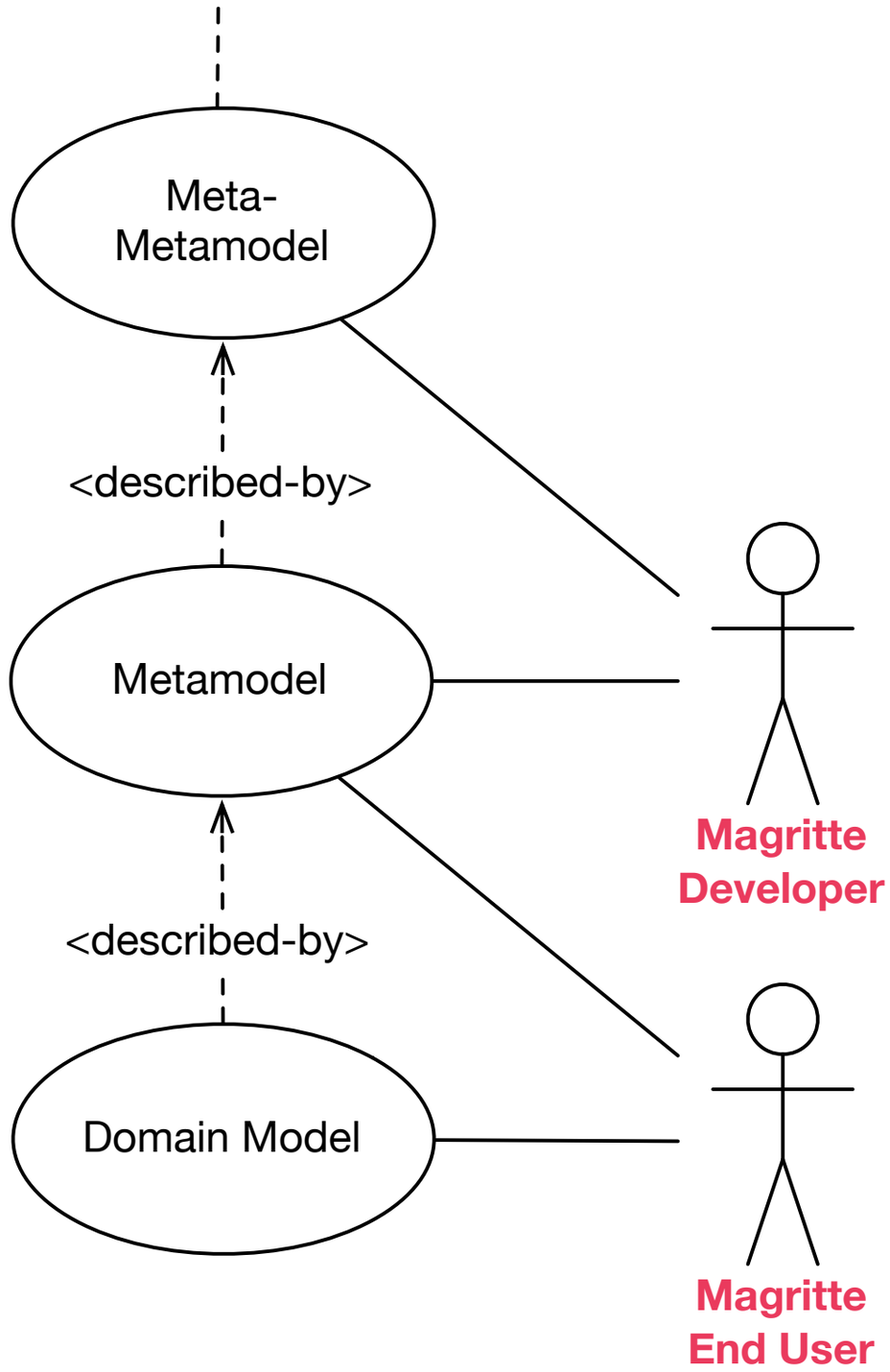
- Describe once, get everywhere.
- Extensibility of classes is ensured.
- Context dependent descriptions.
- End-user customizable.
- Developer configurable.

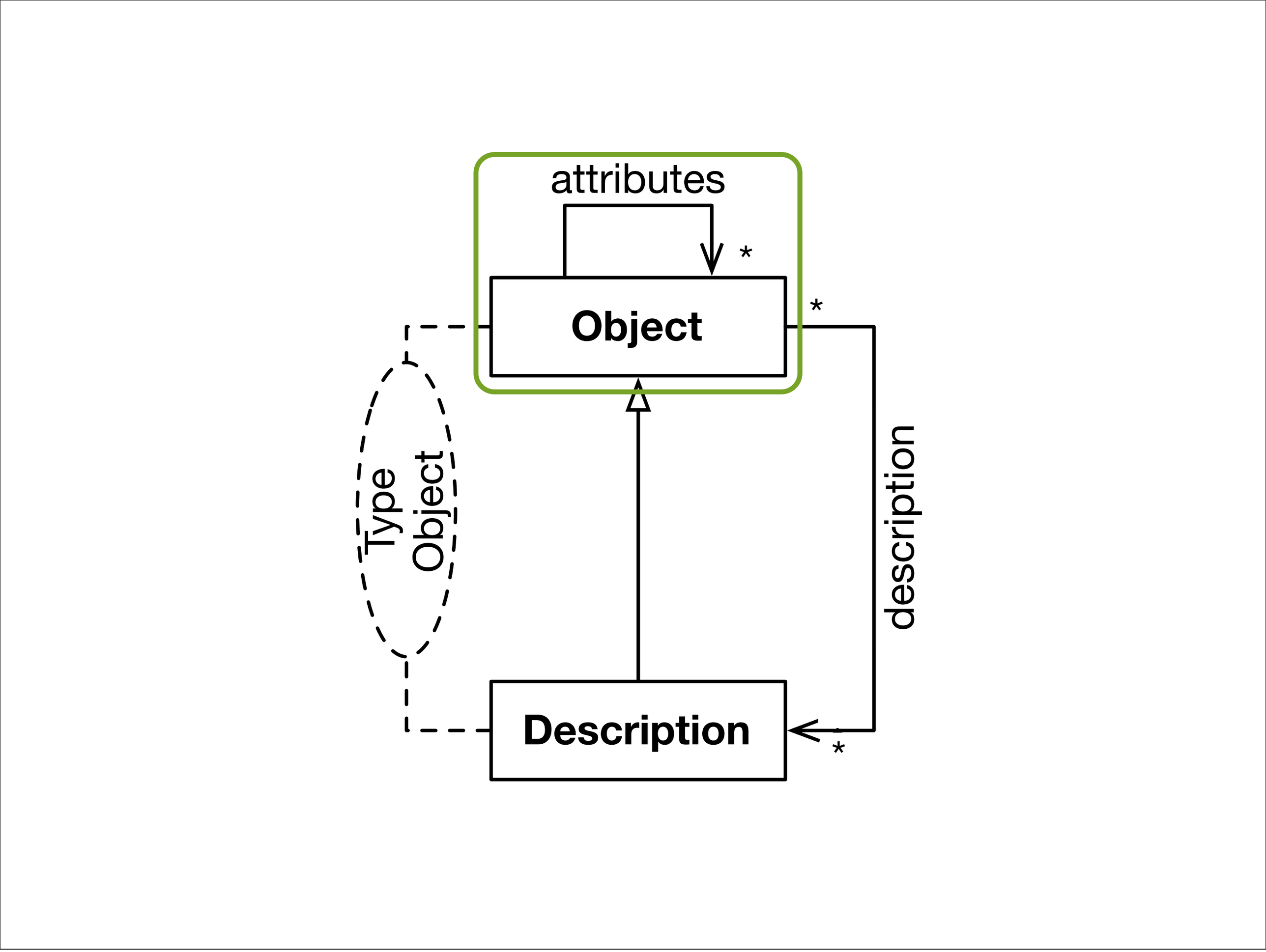
Why is it cool?

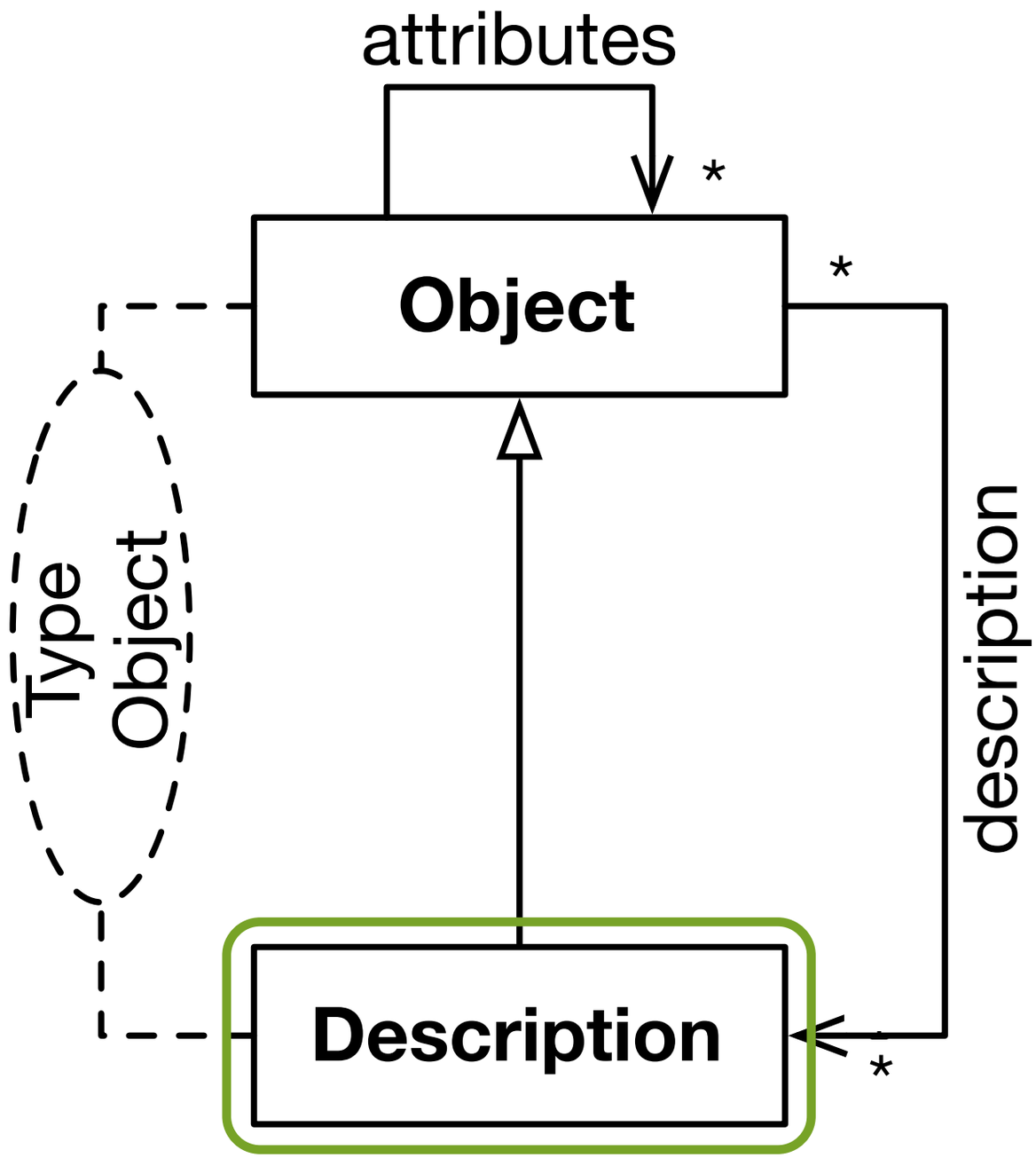
- Be more productive
- Lower coupling
- Do more, with less code
- Do more, with less hacking

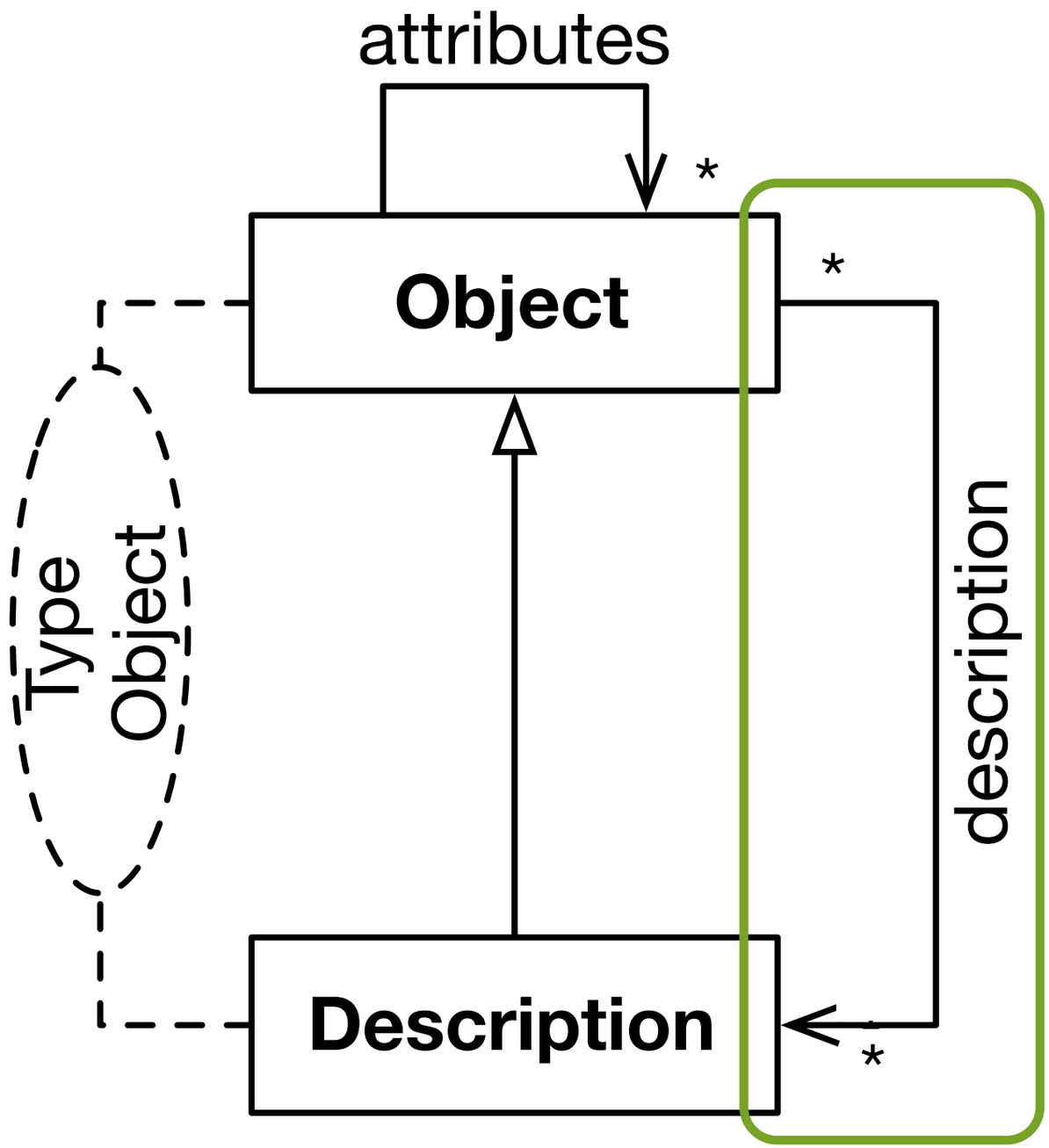
- **Empower your users**


Meta-
Metamodel
<described-by>
Metamodel
Developer
<described-by>
Domain Model
End User


Meta-
Metamodel
<described-by>
Metamodel
<described-by>
Domain Model
Magritte
Developer
Magritte
End User


attributes
*
Object
*
Type
Object
description
Description
*


attributes
*
Object
*
Type
Object
description
Description
*


attributes
*
Object
*
Type
Object
description
Description
*

Run-time

Adaptive Object Model

End users customizability

DEMO

Discussion

- Very powerful and flexible
- Runtime adaptive code
- End-user programmable code
- Can cause meta meta-confusion
- Can be slower than hardcoding

Any sufficiently complicated program contains at least one meta-model.