

*glamour*

<http://moose.unibe.ch/tools/glamour>

**Tudor Gîrba**

# Philipp Bunge

**Tudor Gîrba**

**Lukas Renggli**

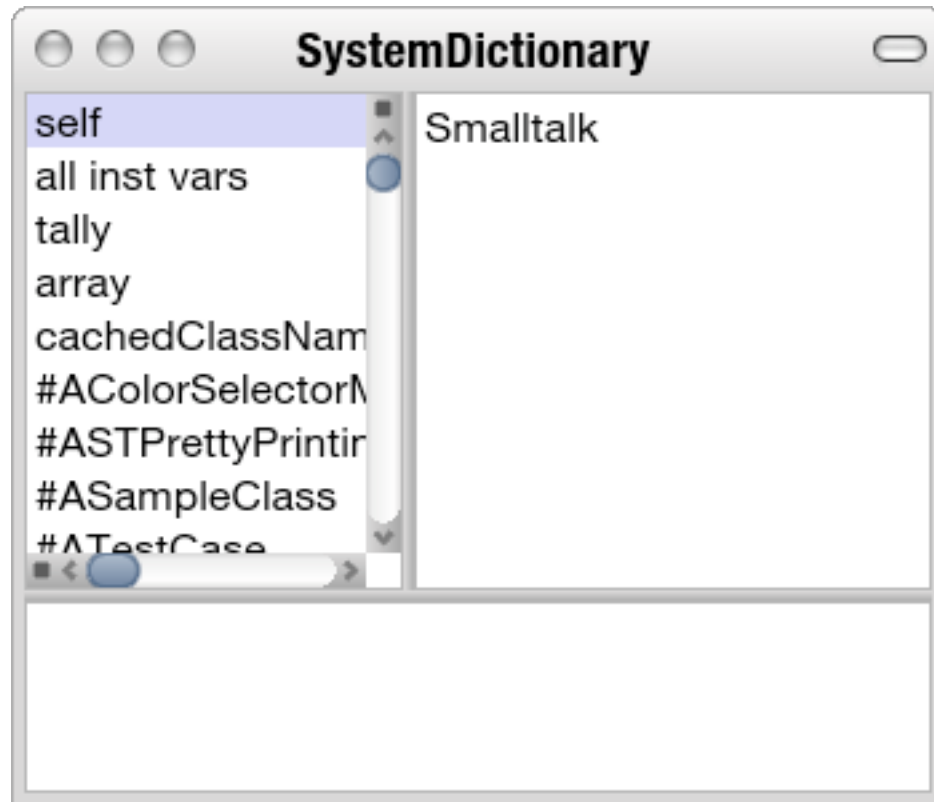
**Jorge Ressia**

**David Röthlisberger**



**Objects**

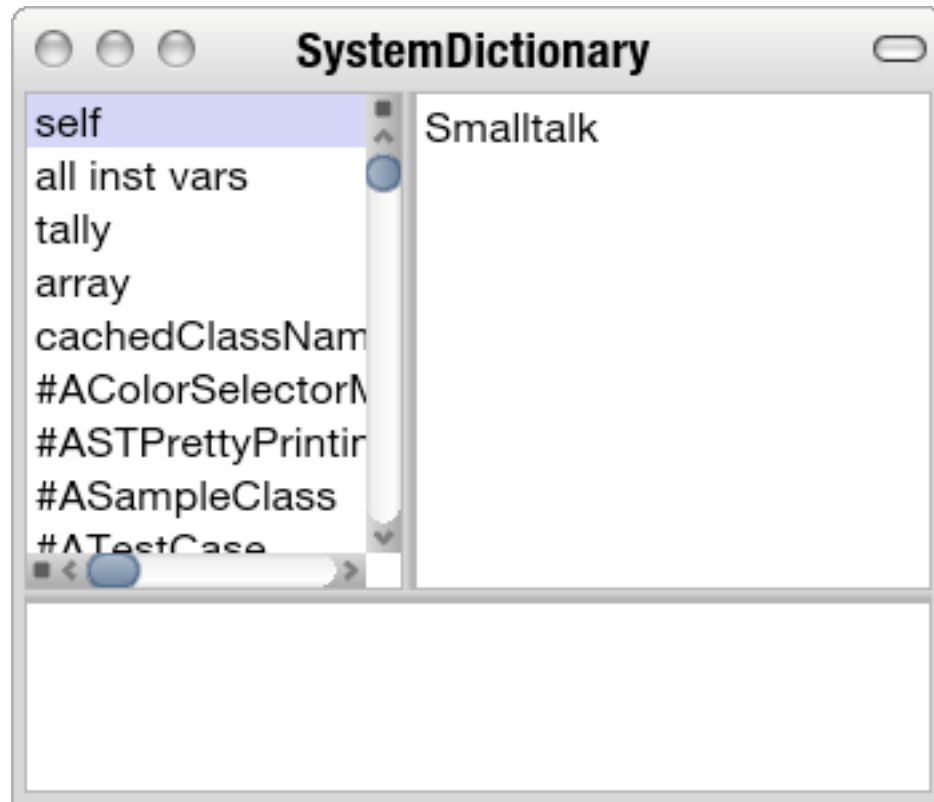
**Objects can be complex**



**Script your browser  
in 15 minutes with**

*gambler*





**System Browser: Float**

Kernel-Chronology  
Kernel-Classes  
Kernel-Methods  
Kernel-Numbers  
Kernel-Objects  
Kernel-Processes  
Kernel-ST80 Remnants  
KernelTests-Chronolog  
KernelTests-Classes

Complex  
Σ Magnitude  
Σ Number  
Σ Float  
Σ Fraction  
Σ ScaledDecimal  
Σ Integer

-- all --  
\*fame  
arithmetic  
comparing  
converting  
copying  
mathematical functions  
printing  
testing

/  
<  
<=  
=  
>  
>=  
abs  
absByteEncode:base  
absPrintExactlyOn:ba

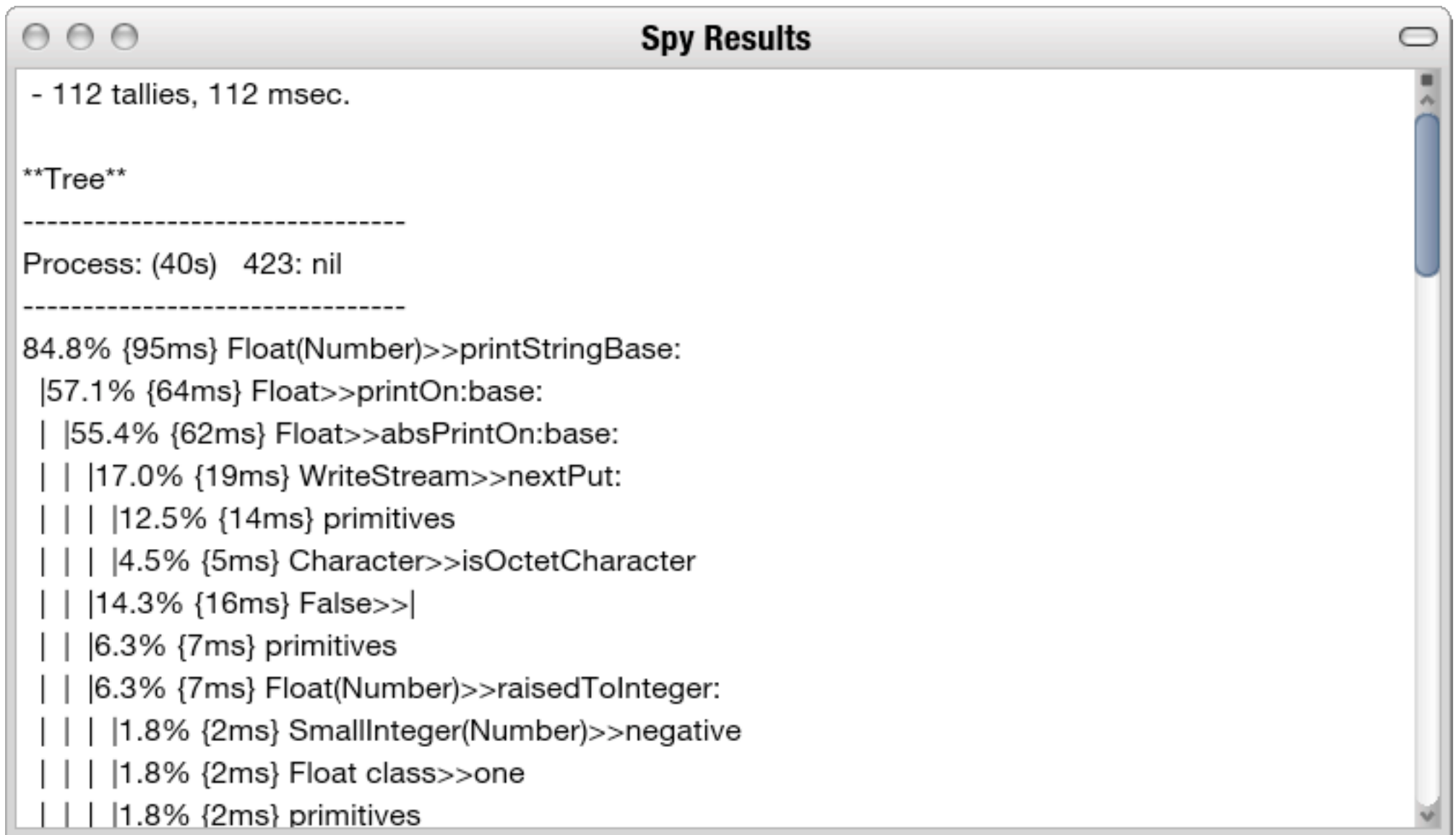
instance ? class

browse hierarchy variables implementors inheritance senders versions view

**abs**

```
"This is faster than using Number abs."  
self < 0.0  
  ifTrue: [^ 0.0 - self]  
  ifFalse: [^ self]
```

MessageTally spyOn: [10000 timesRepeat: [1.23 printString]]

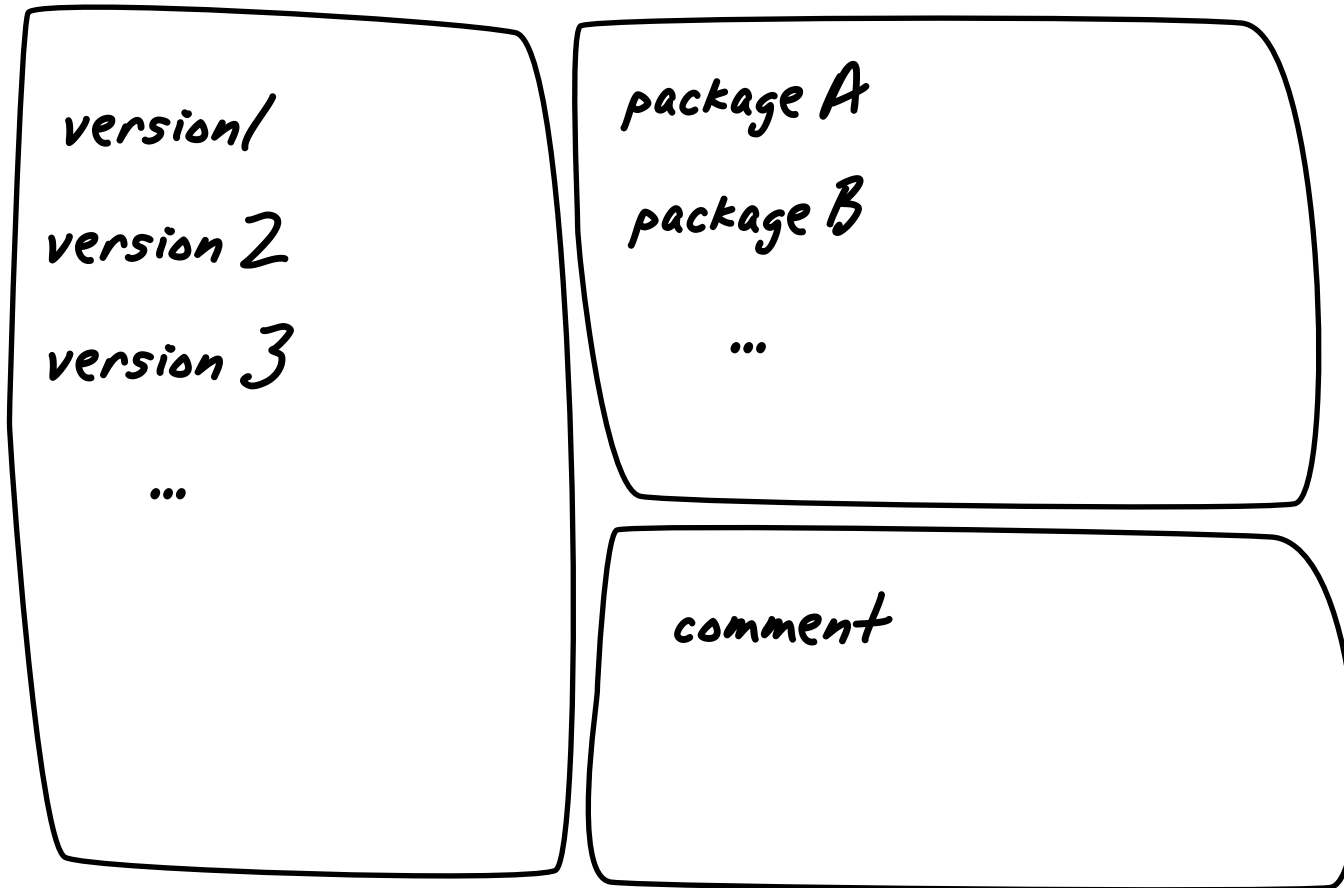


The image shows a window titled "Spy Results" with a scroll bar on the right. The content inside the window is as follows:

```
- 112 tallies, 112 msec.  
  
**Tree**  
-----  
Process: (40s) 423: nil  
-----  
84.8% {95ms} Float(Number)>>printStringBase:  
| 57.1% {64ms} Float>>printOn:base:  
| | 55.4% {62ms} Float>>absPrintOn:base:  
| | | 17.0% {19ms} WriteStream>>nextPut:  
| | | | 12.5% {14ms} primitives  
| | | | 4.5% {5ms} Character>>isOctetCharacter  
| | | 14.3% {16ms} False>>|  
| | | 6.3% {7ms} primitives  
| | | 6.3% {7ms} Float(Number)>>raisedToInteger:  
| | | | 1.8% {2ms} SmallInteger(Number)>>negative  
| | | | 1.8% {2ms} Float class>>one  
| | | | 1.8% {2ms} primitives
```

**Metacello?**

# Metacello?



**Script your browser  
in 15 minutes with**

*gambler*

The screenshot shows the Glamour IDE interface. At the top, the window title is "Glamour". The main area is divided into three panes:

- Left Pane:** A list of project folders including "Balloon-Geometry", "Balloon-Simulation", "Collections-Abstract" (highlighted), "Collections-Arrayed", "Collections-Sequenceable", "Collections-SkipLists", "Collections-Stack", "Collections-Streams", "Collections-Strings", and "Collections-Subsets".
- Middle Pane:** A list of classes including "ArrayedCollection", "Collection", and "SequenceableCollection" (highlighted).
- Right Pane:** A list of methods including "accessing", "comparing" (highlighted), "converting", "copying", "deprecated", "enumerating", "explorer", "removing", "testing", and "uniquing".

Below the panes, there are two tabs: "Method source" (selected) and "Class comment". The "Method source" tab displays the following code:

```
= otherCollection  
"Answer true if the receiver is equivalent to the otherCollection.  
First test for identity, then rule out different species and sizes of  
collections. As a last resort, examine each element of the receiver  
and the otherCollection."
```

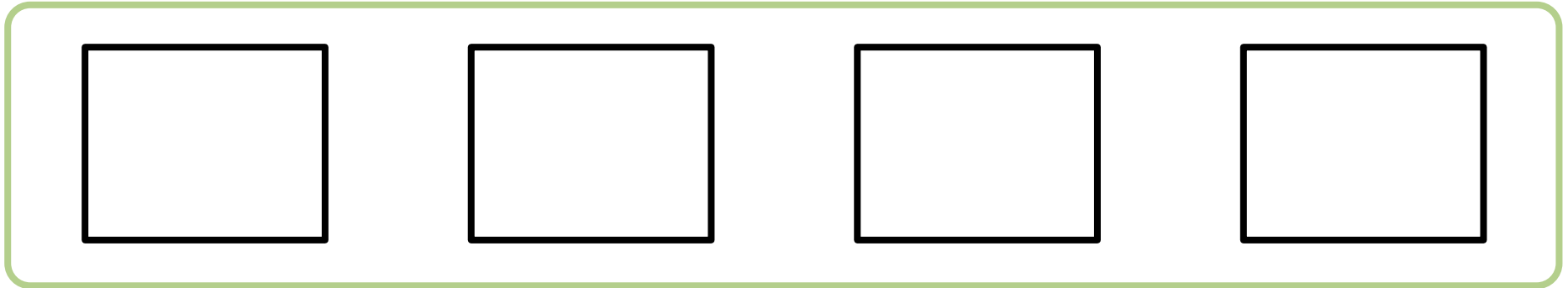
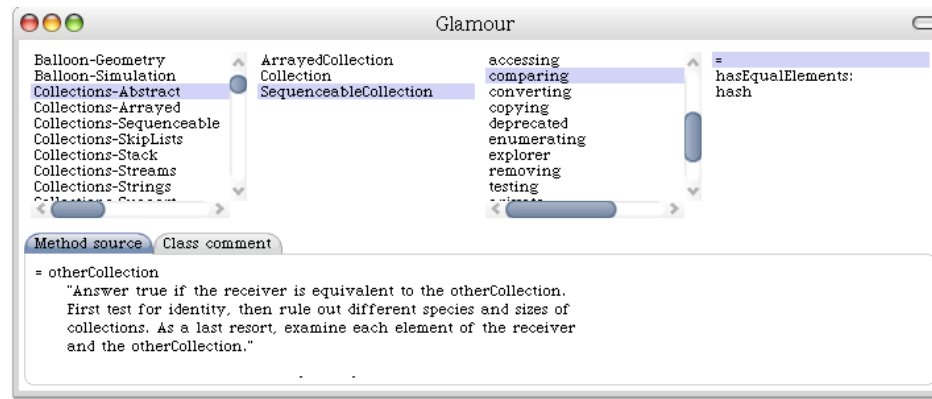
Glamour

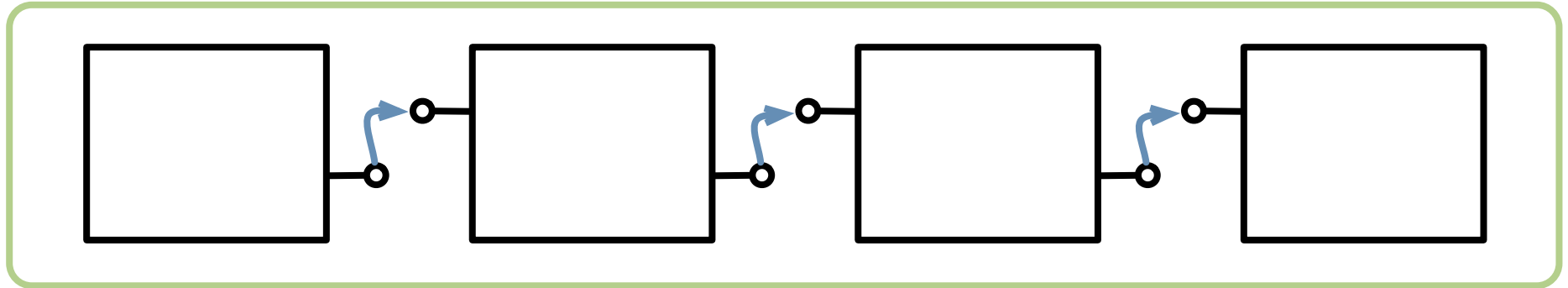
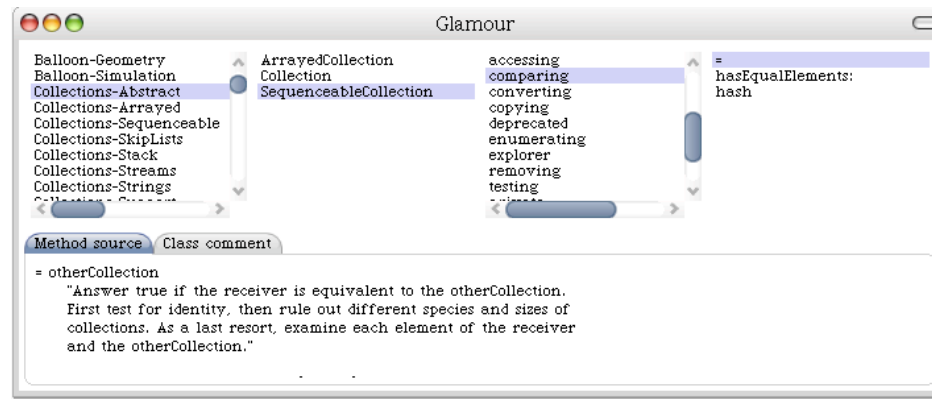
Balloon-Geometry	ArrayedCollection	accessing	=
Balloon-Simulation	Collection	comparing	hasEqualElements:
Collections-Abstract	SequenceableCollection	converting	hash
Collections-Arrayed		copying	
Collections-Sequenceable		deprecated	
Collections-SkipLists		enumerating	
Collections-Stack		explorer	
Collections-Streams		removing	
Collections-Strings		testing	
Collections-Summary		...	

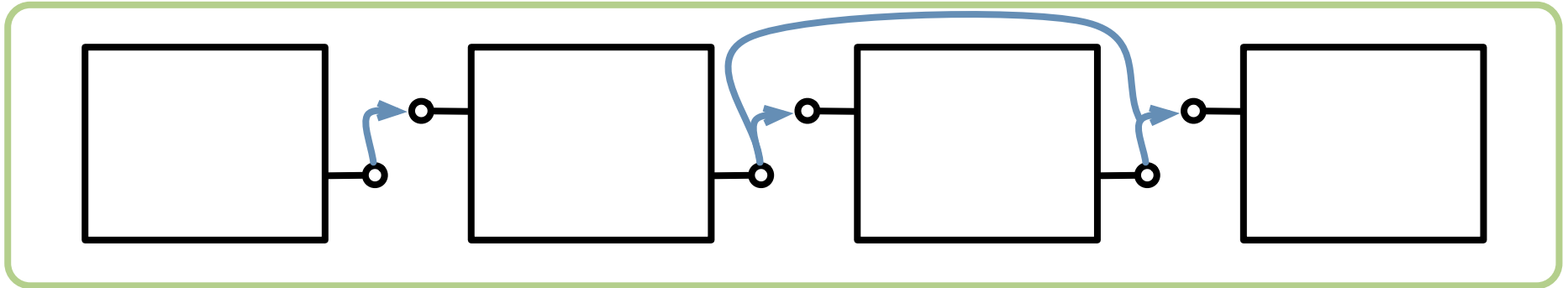
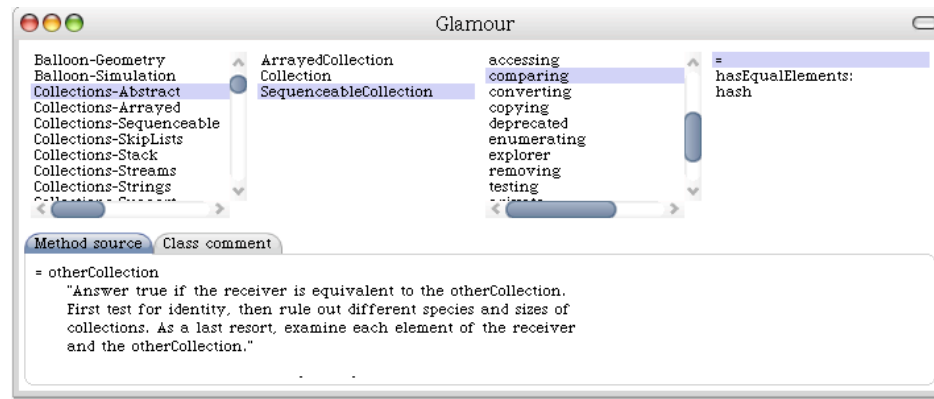
Method source    Class comment

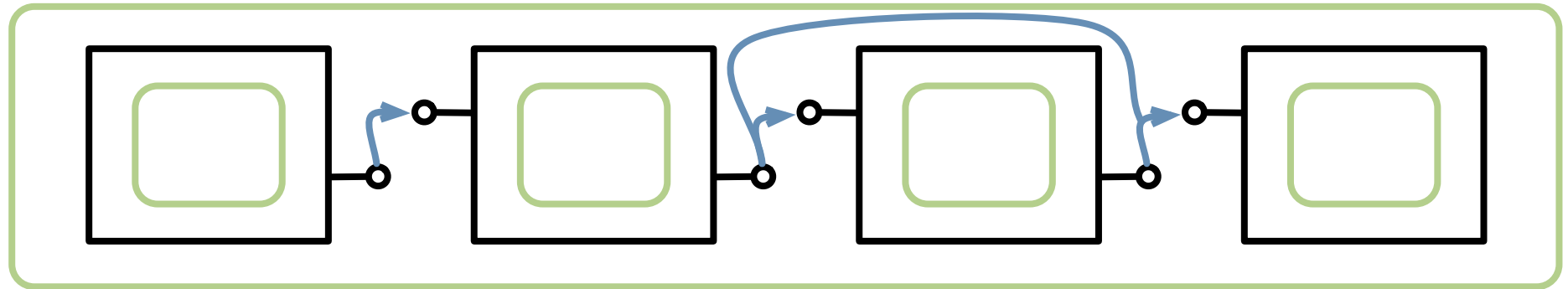
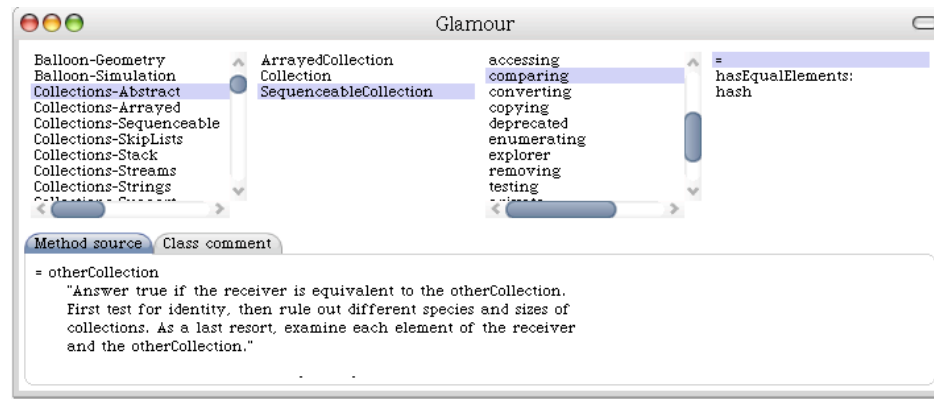
```
= otherCollection
"Answer true if the receiver is equivalent to the otherCollection.
First test for identity, then rule out different species and sizes of
collections. As a last resort, examine each element of the receiver
and the otherCollection."
```

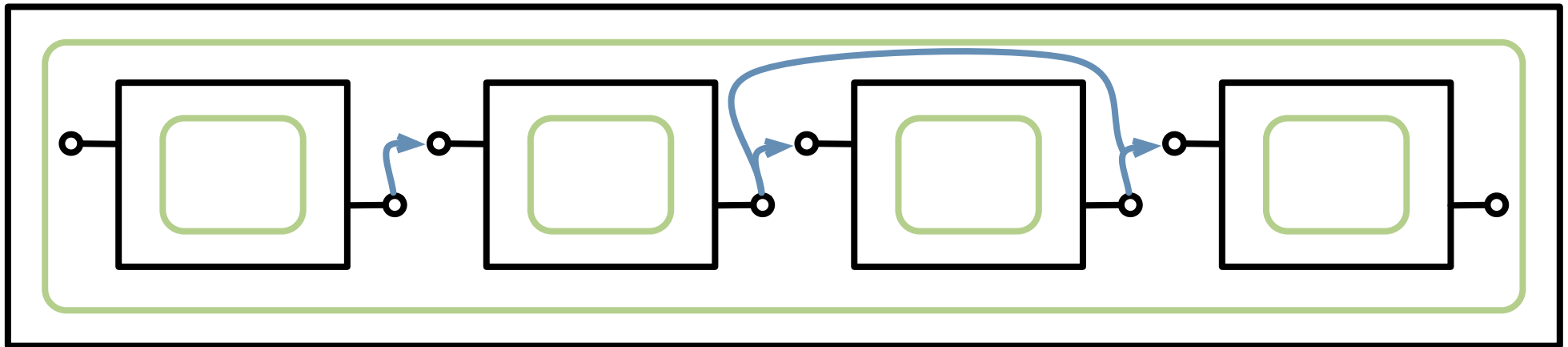
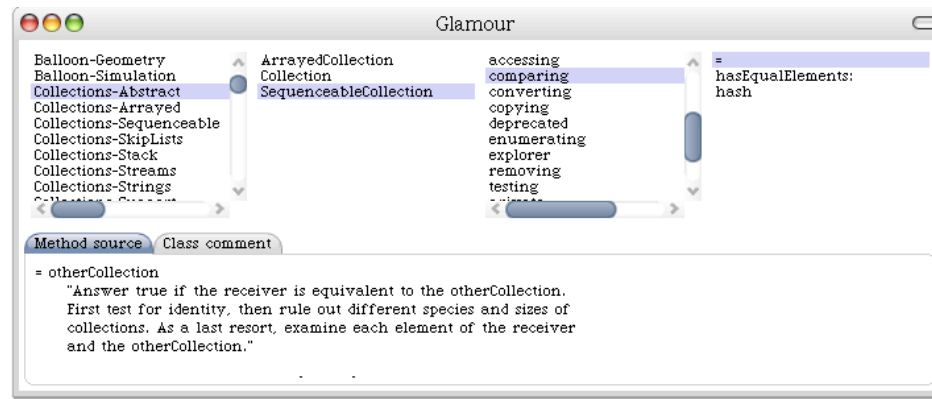


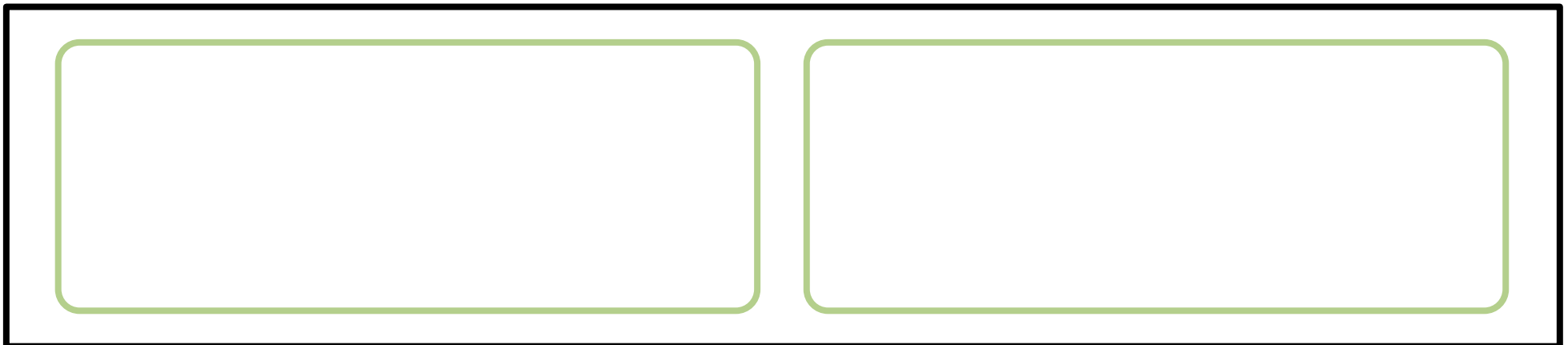
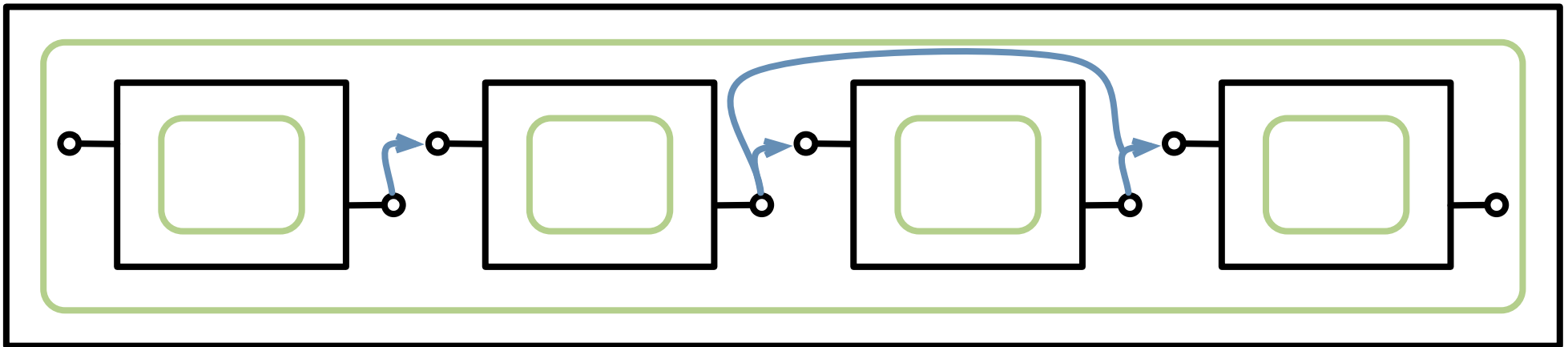
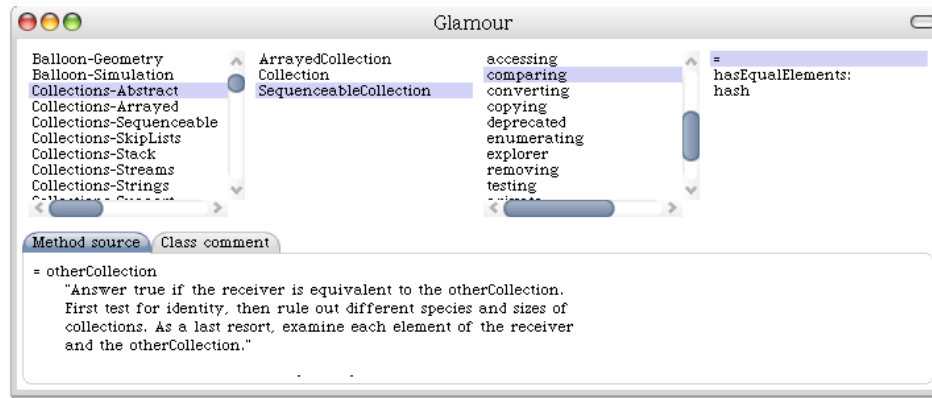


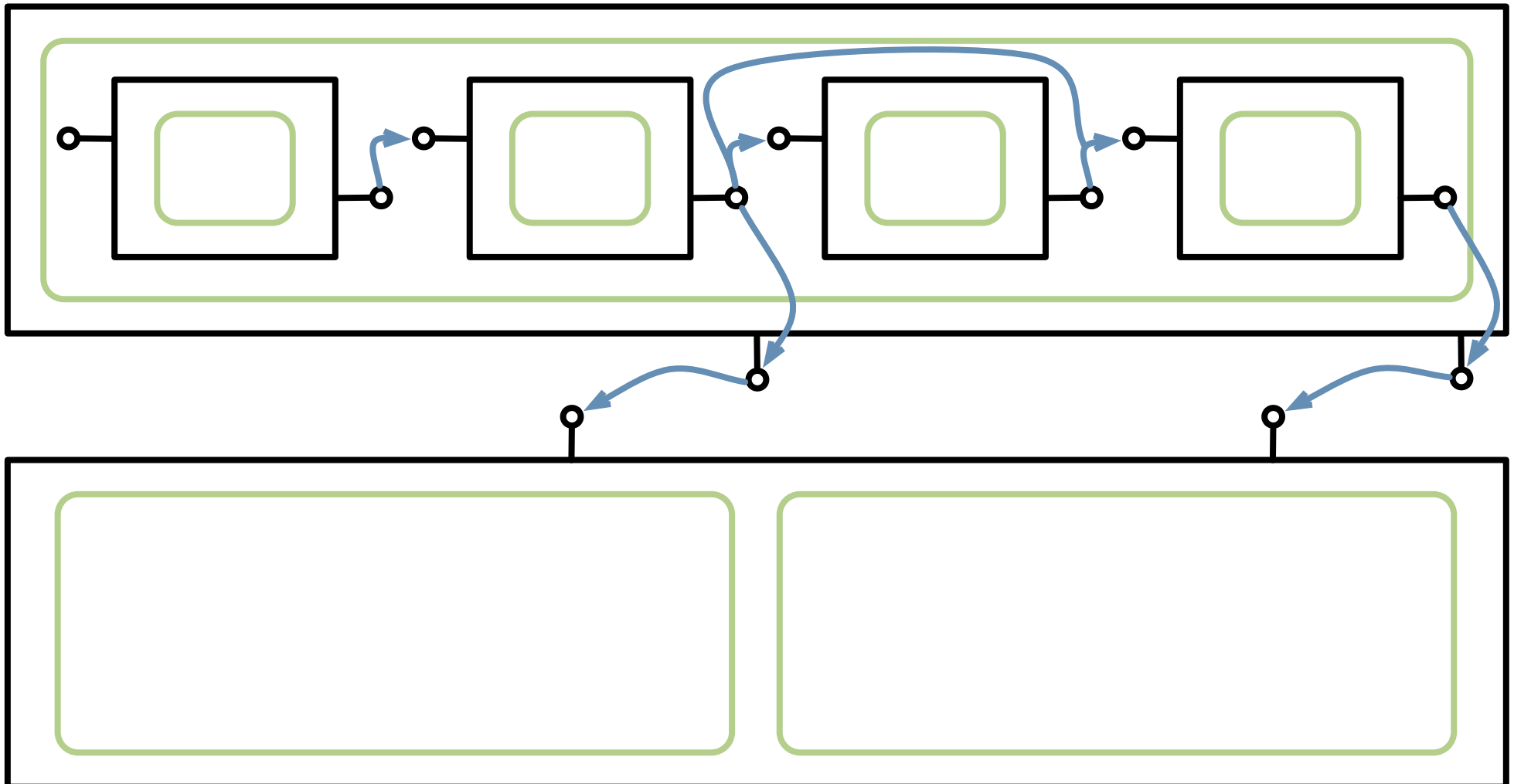
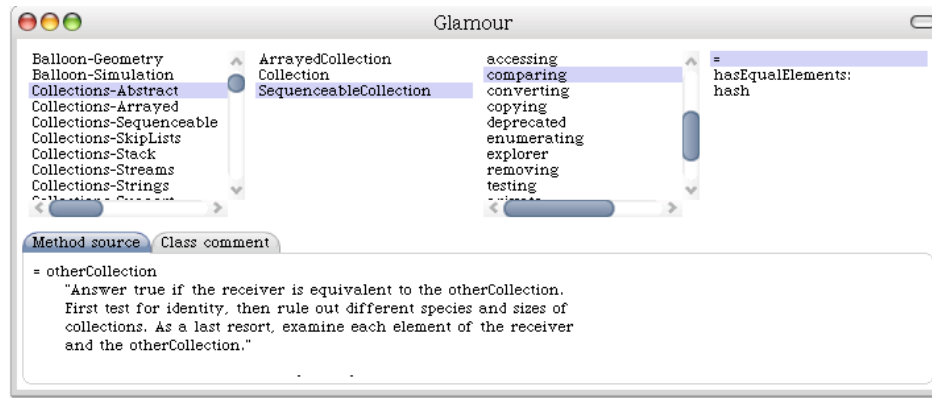






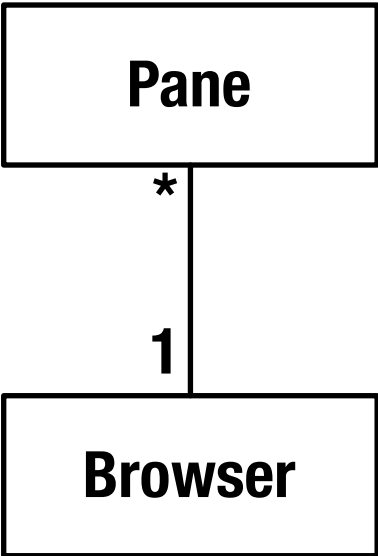


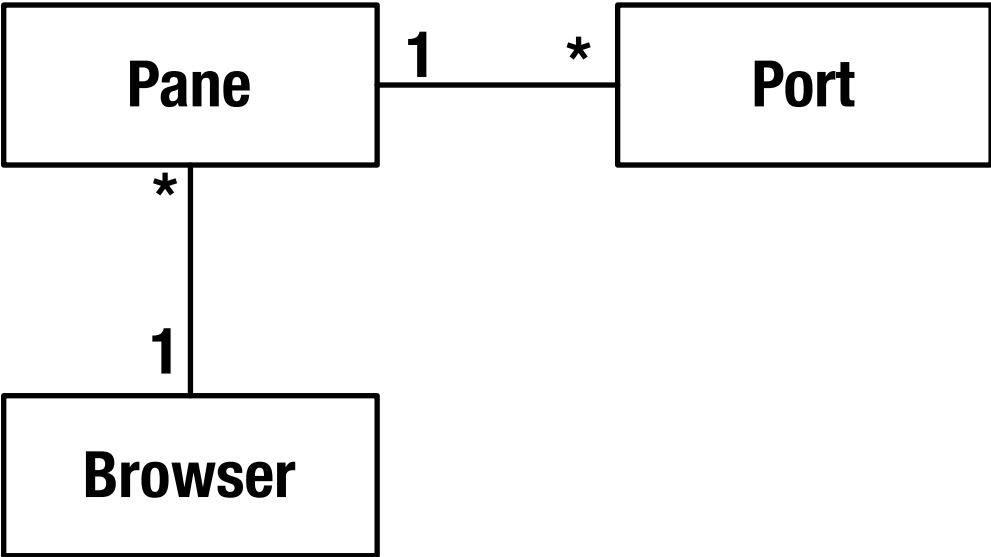


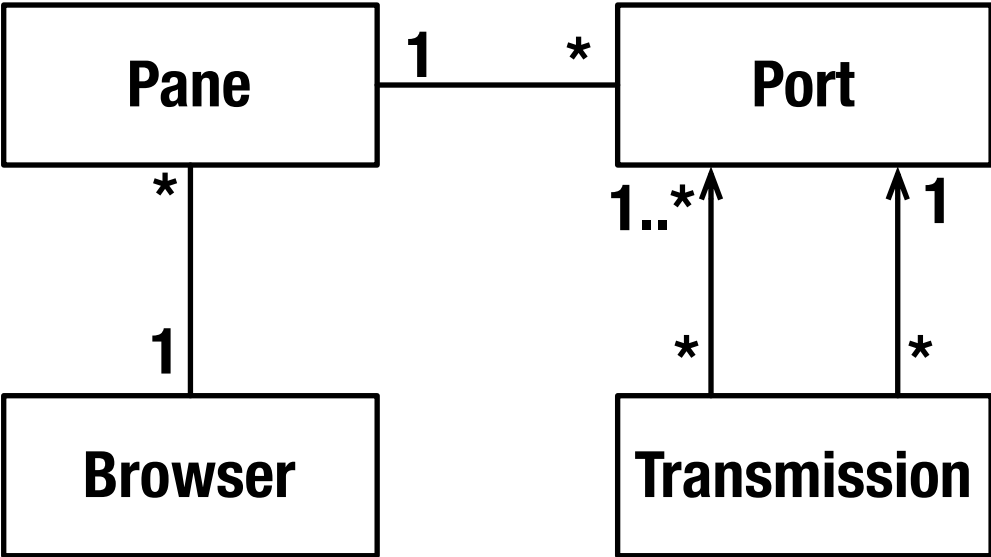


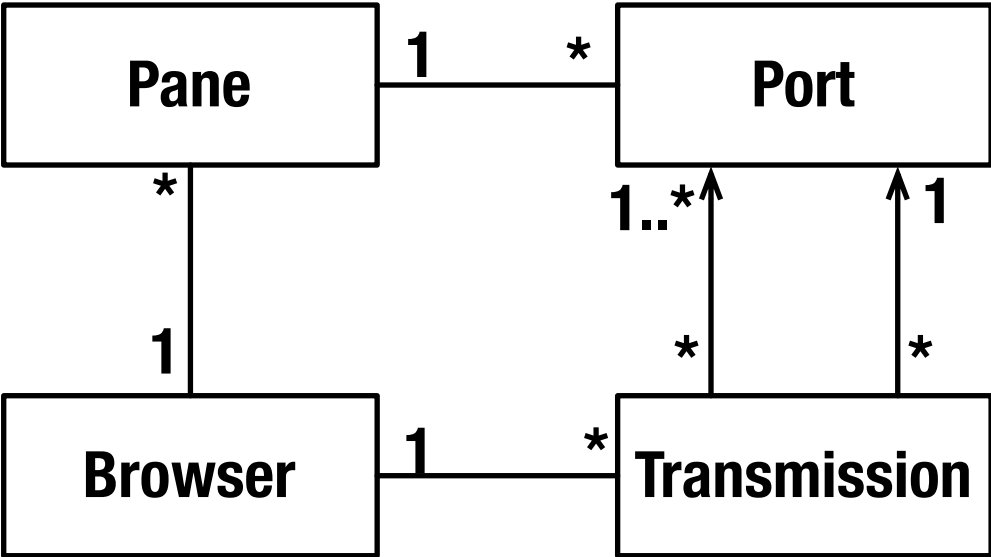
**Browser**

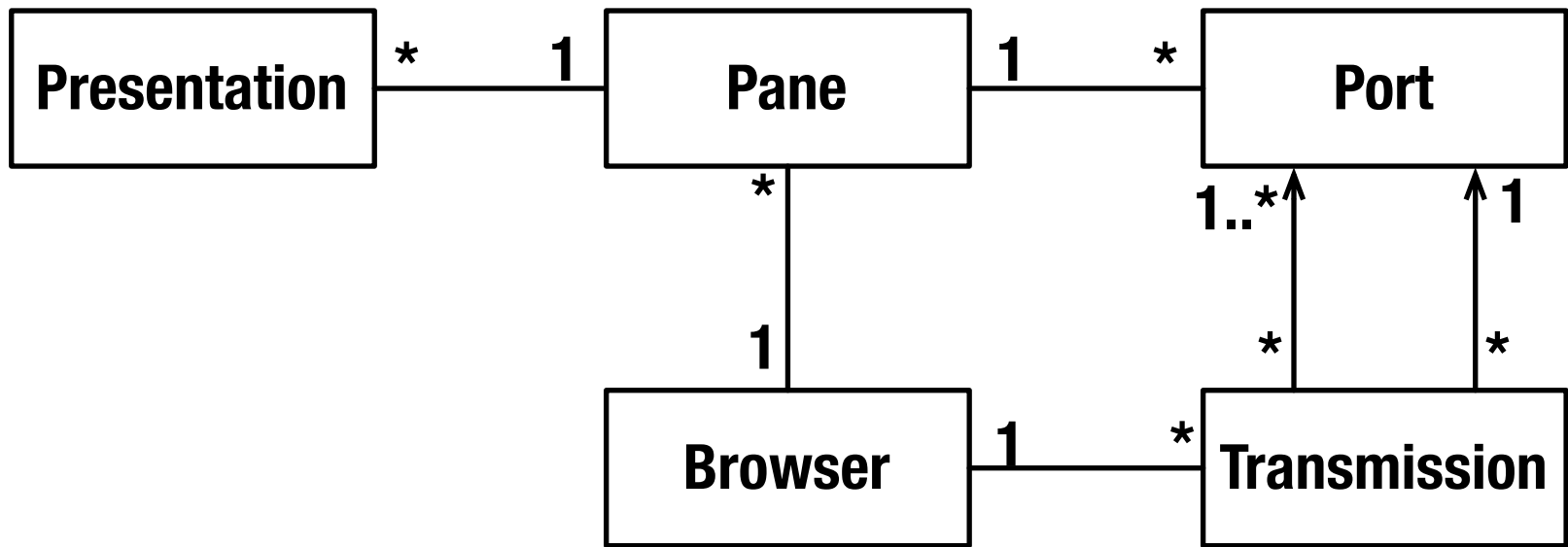


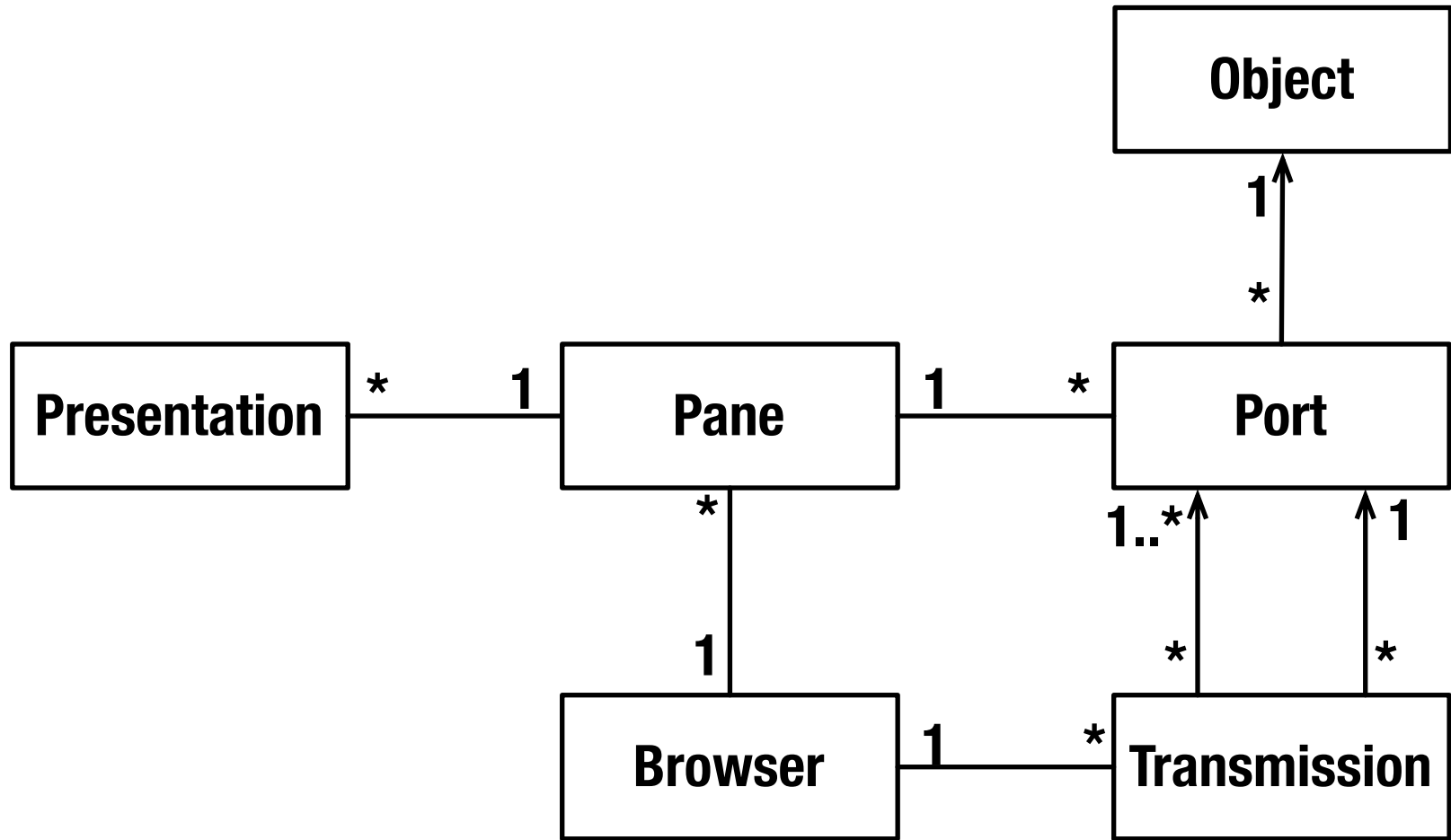


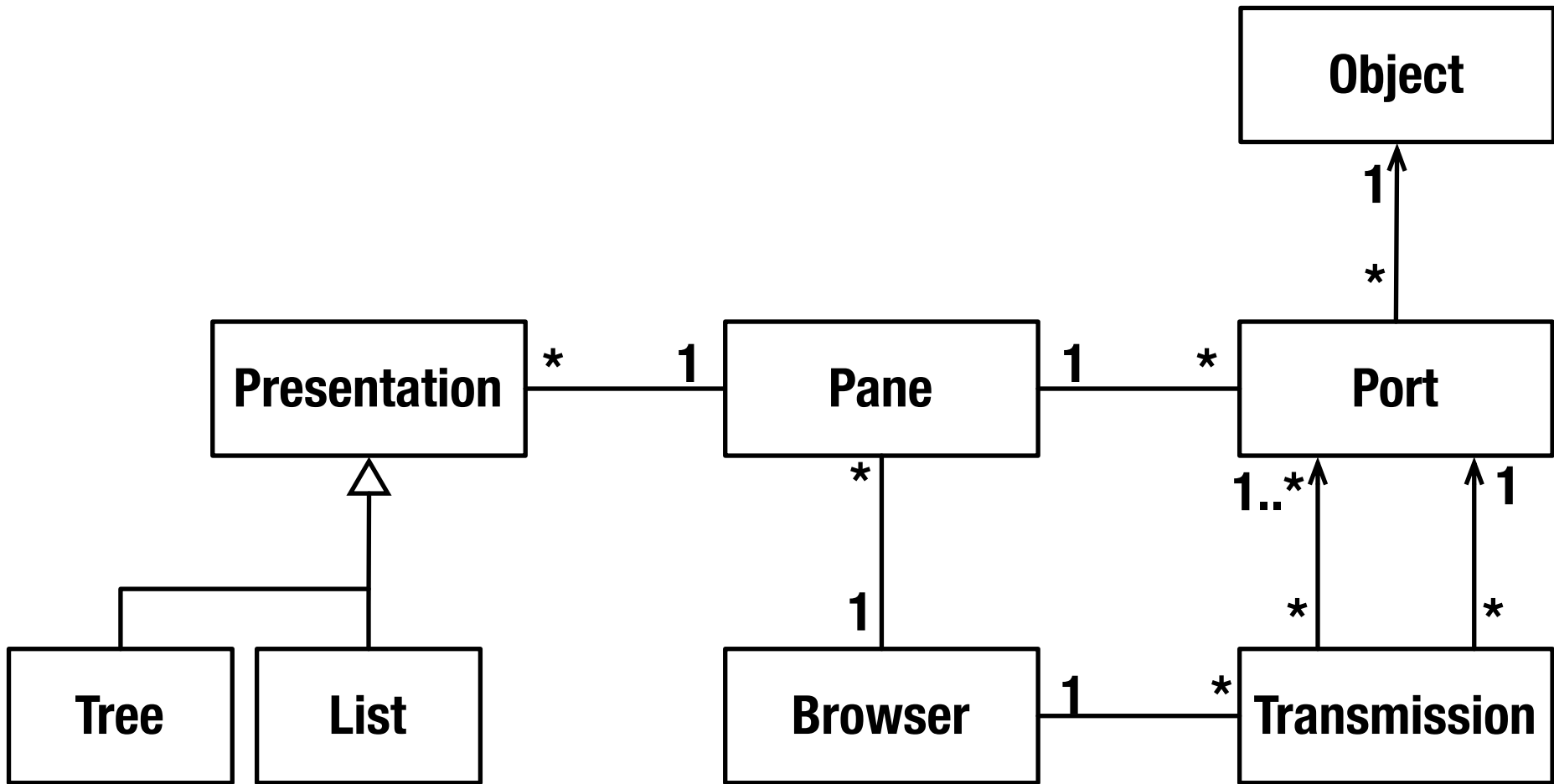


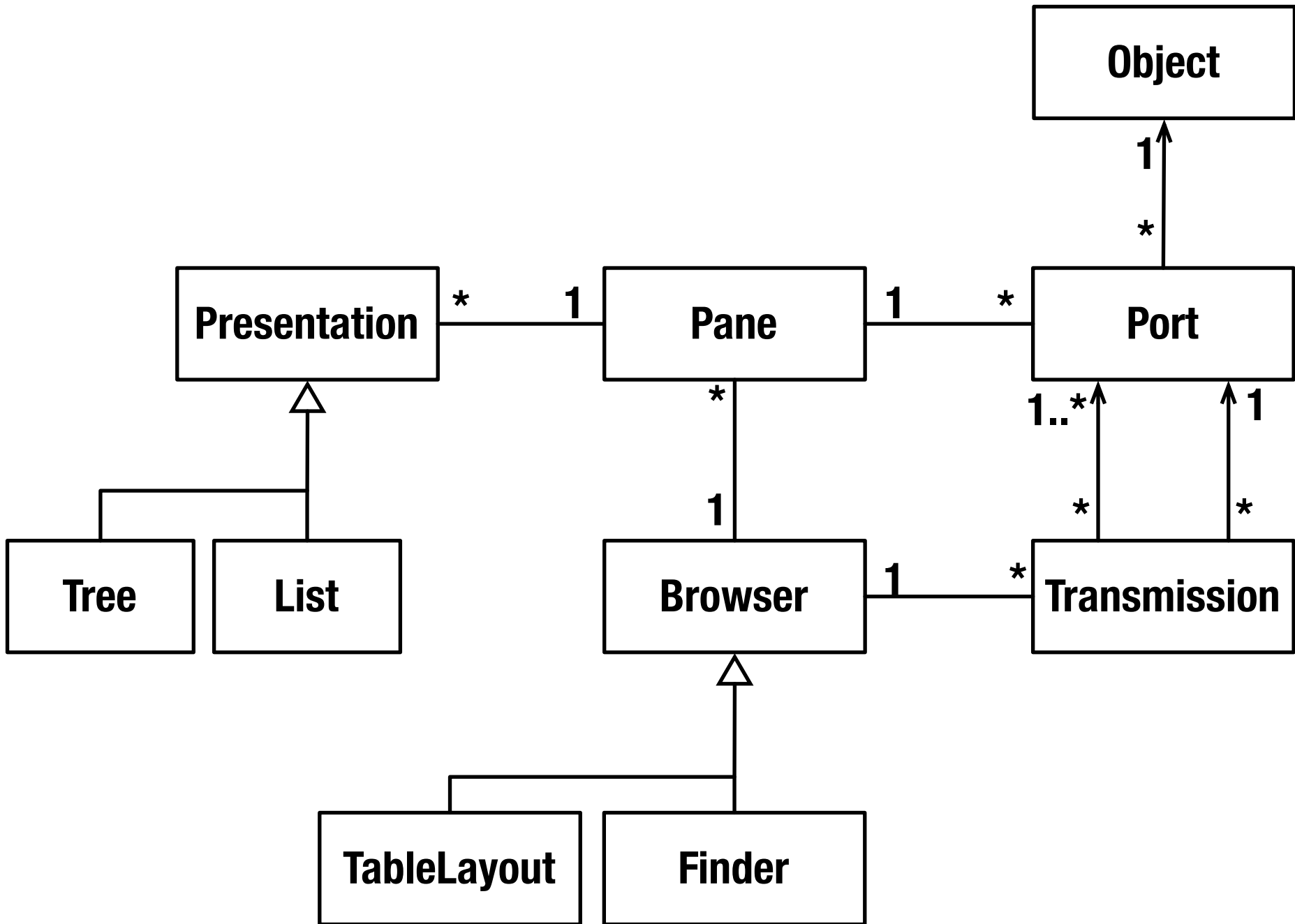




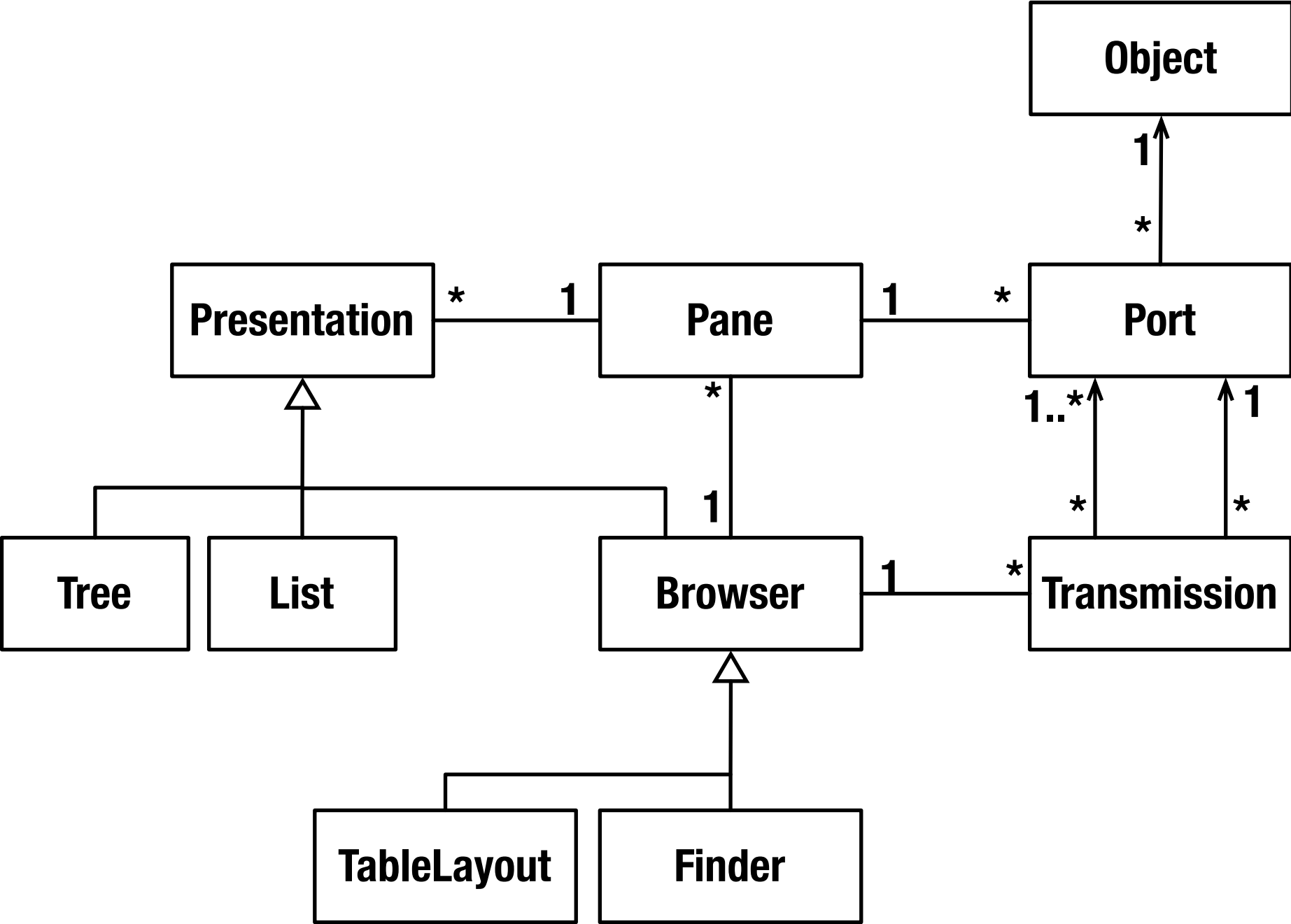


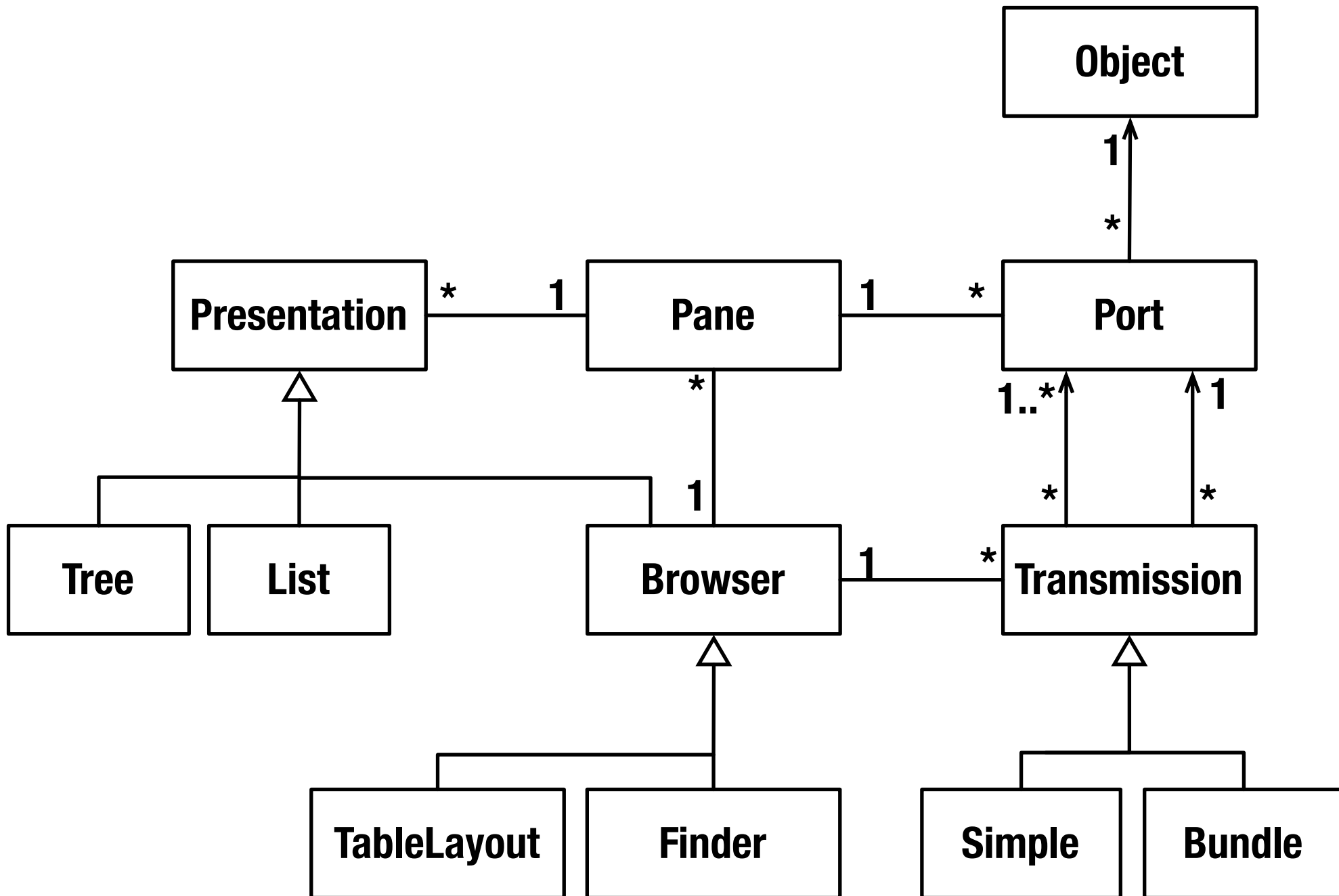


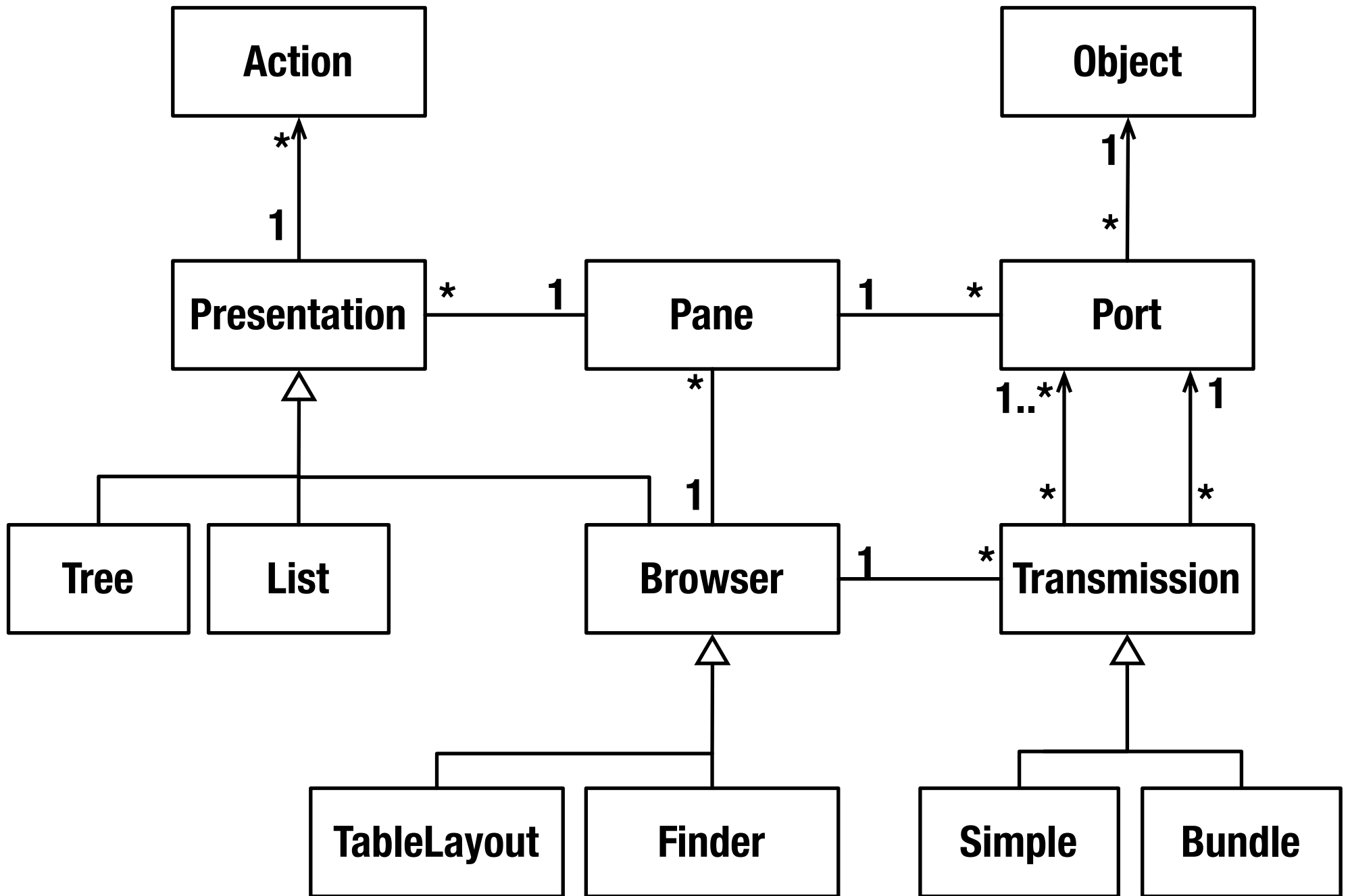


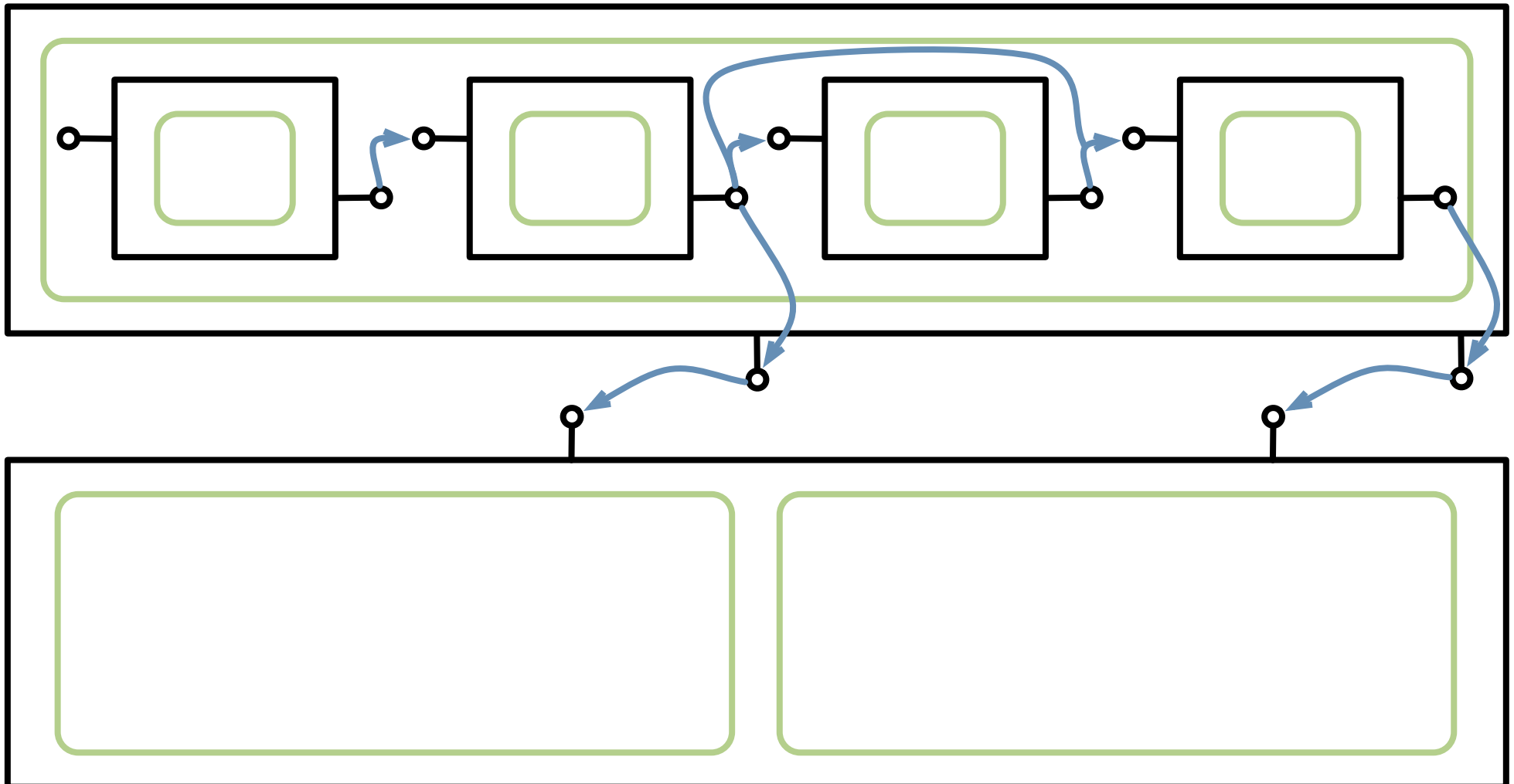
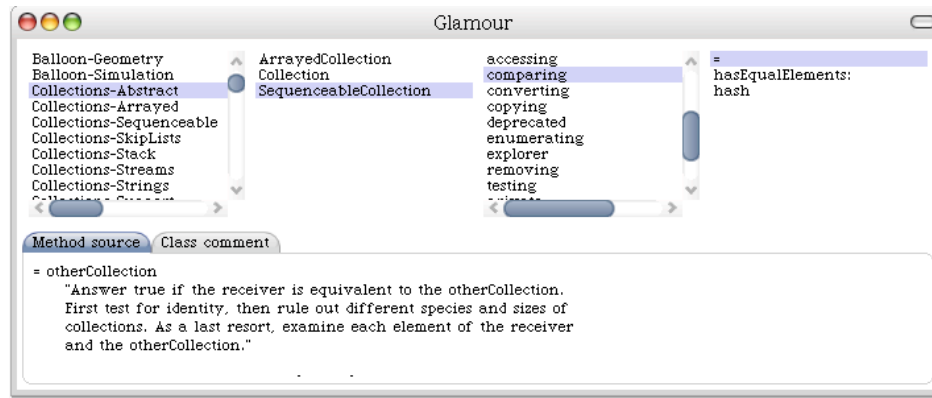


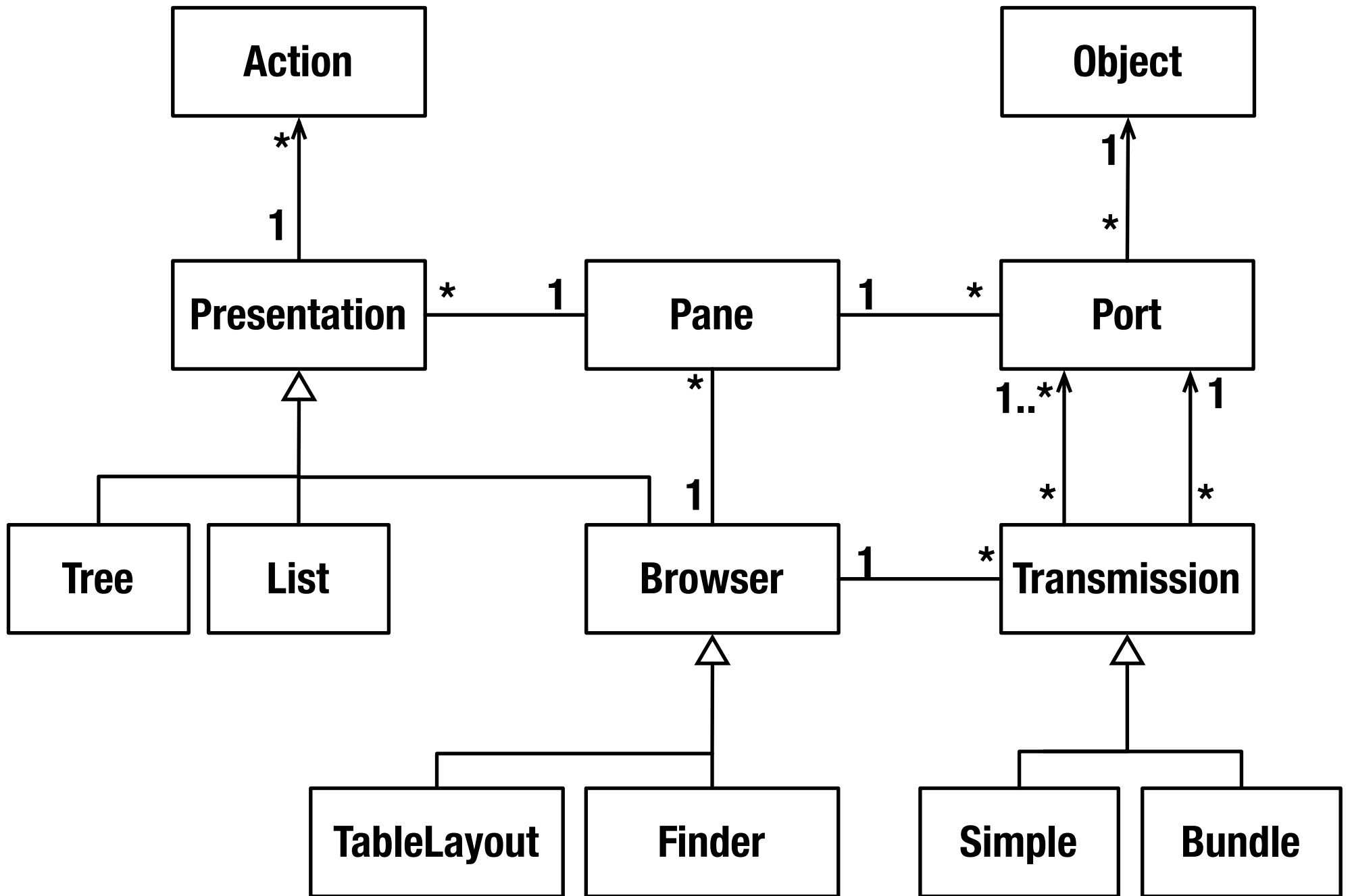


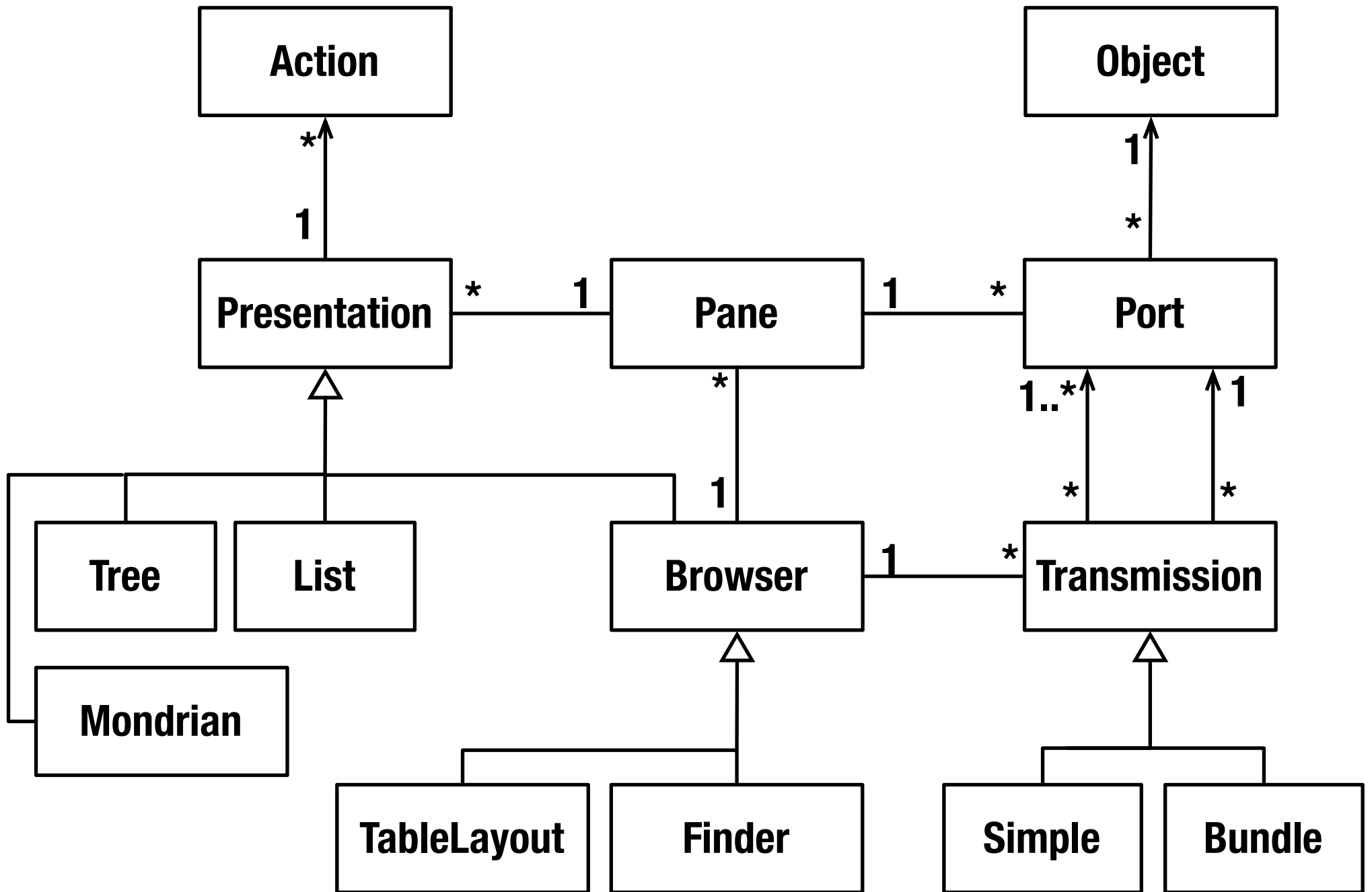




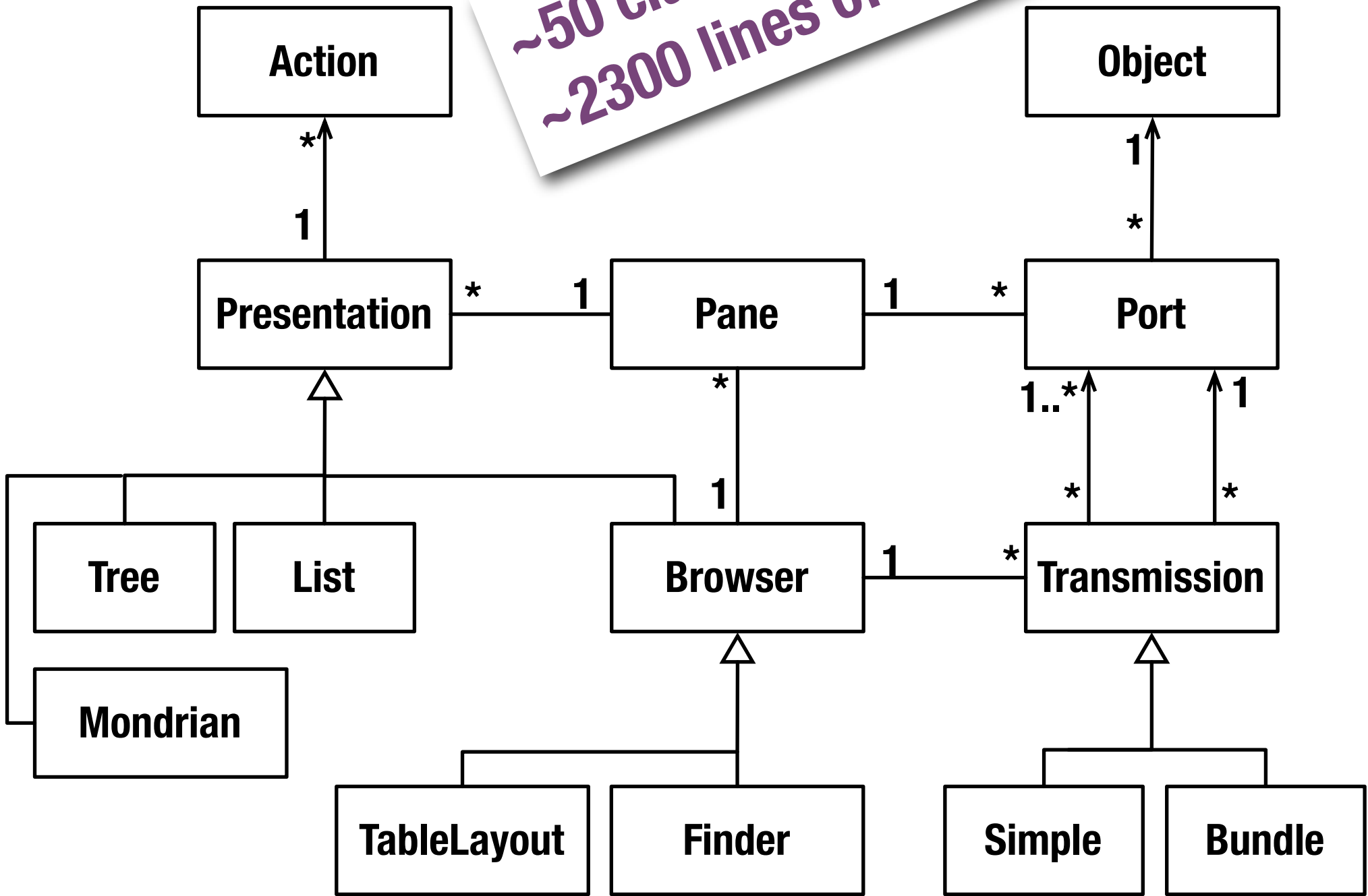








*~50 classes  
~2300 lines of code*



Glamour

- Balloon-Geometry
- Balloon-Simulation
- Collections-Abstract
- Collections-Arrayed
- Collections-Sequenceable
- Collections-SkipLists
- Collections-Stack
- Collections-Streams
- Collections-Strings
- Collections-Subscriptable

- ArrayedCollection
- Collection
- SequenceableCollection

- accessing
- comparing
- converting
- copying
- deprecated
- enumerating
- explorer
- removing
- testing
- visiting

=

hasEqualElements:  
hash

Method source    Class comment

= otherCollection  
"Answer true if the receiver is equivalent to the otherCollection. First test for identity, then rule out different species and sizes of collections. As a last resort, examine each element of the receiver and the otherCollection."

Seaside Glamorous Browser

http://localhost:8080/seaside/glamour/pier

Pier

- Information
- Introduction
- Syntax
- License

Name: Introduction

Tags:

Title: Introduction

Contents:

Pier is a powerful and extensible implementation of a meta-described content management and Wiki system, written with objects from top to bottom:

- "Object-Oriented Design:" Pier features a fully object oriented and meta-described domain model built using Magritte. The content of the pages is parsed and stored as a tree of different entities representing text, links, tables, lists, formattings, etc.
- "Extensibility:" Everything in Pier can be extended: page types, storage mechanism, actions, security mechanism, web-server, etc. Plug-ins can be shared within the community and loaded independently of each other into the system.
- "Open Source:" Pier is released under the MIT license which grants unrestricted rights to copy, modify, and redistribute as long as the original copyright and license terms are retained.

Save    Cancel

Glamorous Browser

- Work
  - Glamour-Paper
    - pictures
    - Glamour-Pharo
    - Glamour-VisualWorks

glamour.aux  
glamour.bbl  
glamour.log  
glamour.out  
glamour.pdf  
glamour.tex  
lincs.cls

% \$Id: glamour.tex 26800 2009-05-09 13:32:30Z osjcar \$  
% \$Date: 2009-05-09 15:32:30 +0200 (Sat, 09 May 2009) \$

\documentclass[runningheads]{lincs}

\usepackage{graphicx}

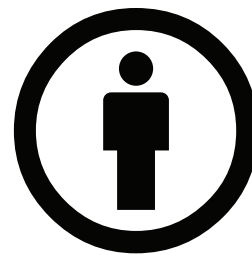


glamour

<http://moose.unibe.ch/tools/glamour>

# **Tudor Gîrba**

[www.tudorgirba.com](http://www.tudorgirba.com)



[creativecommons.org/licenses/by/3.0/](http://creativecommons.org/licenses/by/3.0/)