

Implementing Mondrian in Glamorous Toolkit

Bachelor thesis by
Cyrill Rohrbach

Supervised by
O. Nierstrasz & A. Bergel

Goal of the thesis

- Implement an easier to use Mondrian within GT
- Find out if GT is a better platform for Mondrian than normal Pharo

What is Mondrian?

- Developed in 2006 by Michael Meyer and Tudor Gîrba
- Lightweight tool to create interactive visualizations with simple scripts

The image displays three interconnected components of the Mondrian tool:

- Top Window:** A table titled "an OrderedCollection [9 items] (CRBoxBu" showing a list of builders. The table has columns for "Index" and "Item".
- Middle Window:** A code editor titled "a CompiledMethod (Presentation_1>>#methodForMondrianExplain)" showing the source code for the `methodForMondrianExplain` method. The code includes comments and uses `view` commands to interact with the visualization.
- Bottom Window:** A class diagram titled "a CRMondrian" showing a hierarchy of classes. `CRShapeBuilder` is the root class, and it has several subclasses: `CRBoxBuilder`, `CRCircleBuilder`, `CRClassBlueprintBuilder`, `CRCustomShapeBuilder`, `CRLabelBuilder`, `CRMondrianAsNode`, `CRPointBuilder`, and `CRPreBuiltNode`.

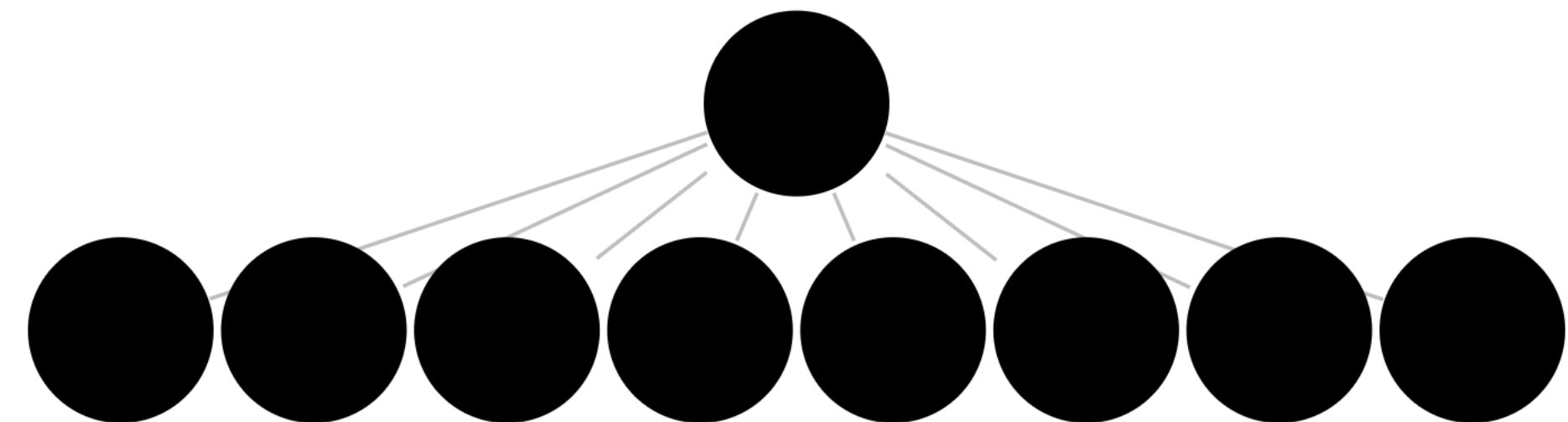
Mondrian key features

- Interactive visualizations
- Ability to accommodate any data model
- Small overhead
- Infinite nesting
- Ability to easily change the shape of the nodes
- Ability to add interactions to the nodes
- Ability to create polymetric views

How could GtMondrian be better?

- Changing the node shape should be easier
- Ability to add Actions
- Builders to make difficult graphs easier

```
gtMondrianPointLabelColor
<gtExample>
| m |
m := GtMondrian new .
m nodes shape: [ :x | BElement new
  geometry: BCircle new ;
  background: Color black ;
  aptitude: (BrGlamorousWithTooltipAptitude new
    showDelay: 0;
    hideDelay: 0;
    contentStencil: [BrLabel new aptitude:
      BrGlamorousLabelAptitude new
      glamorousRegularFontAndSize ;
      alignCenter;
      text: x])
  ];
  with: self classesAsCollection .
m edges connectFrom: #superclass .
m layout tree.
^m
```



CRMondrian

- Very similar to original Mondrian
- Ability to create new shapeBuilder to make complicated scripts easier
- Support for multiple shapeBuilders
- Support for Actions

CRMondrian Script

```
m := CRMondrian new. ← View Renderer  
m nodes box with: (1 to: 10). ← ShapeBuilder  
m edges connectFrom: [:x | x // 2]. ← EdgeBuilder  
m layout tree. ← Layout  
m
```

Feature Demo

Challenges

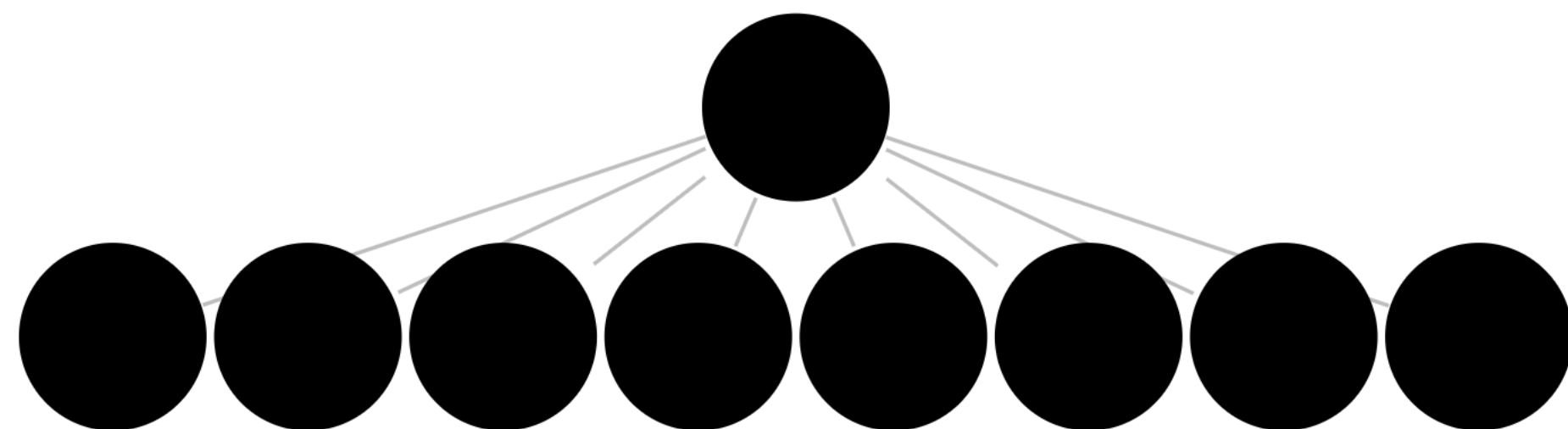
- Poor Documentation
- Changes in GT break implementation
- BIElements can't be copied

- GT is slow

Is it easier to use than GtMondrian

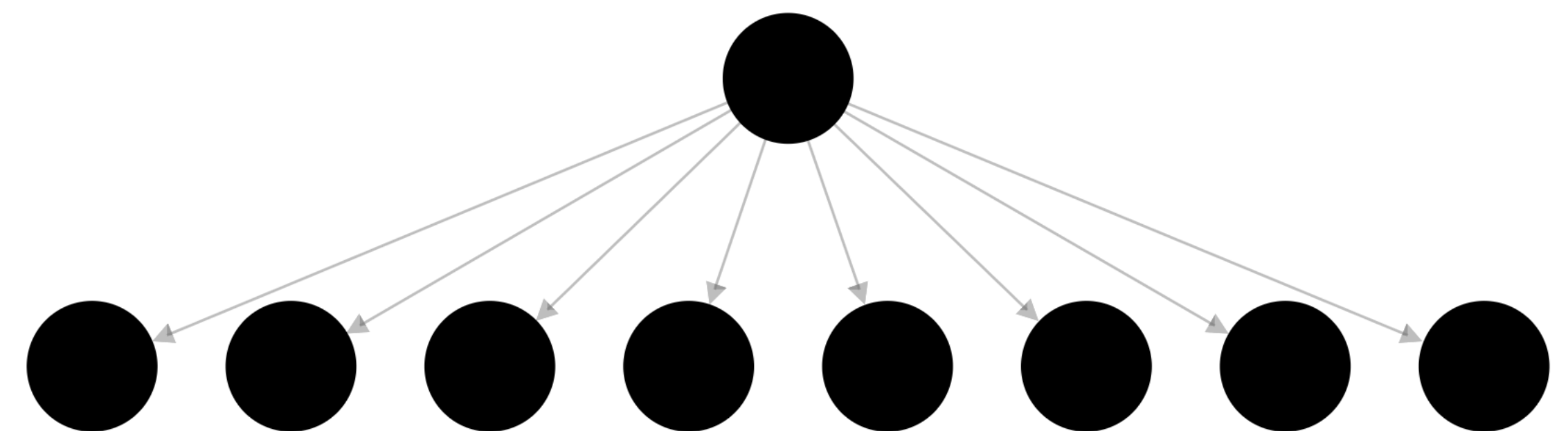
GtMondrian

```
gtMondrianPointLabelColor ●
<gtExample>
| m |
m := GtMondrian new ▶ .
m nodes shape: [ :x | BElement new ▶
  geometry: BCircle new ▶ ;
  background: Color black ▶ ■;
  aptitude: (BrGlamorousWithTooltipAptitude new ▶
    showDelay: 0;
    hideDelay: 0;
    contentStencil: [BrLabel new ▶ aptitude:
      BrGlamorousLabelAptitude new ▶
      glamorousRegularFontAndSize ▶ ;
      alignCenter;
      text: x])
];
  with: self classesAsCollection ▶ .
m edges connectFrom: #superclass.
m layout tree.
^m
```



CRMondrian

```
crMondrianPointLabelColor ●
<gtExample>
| m |
m := CRMondrian new ▶ .
m nodes
  point
  with: self classesAsCollection ▶ .
m edges arrow; connectFrom: #superclass.
m layout tree.
^m
```



Is GT better for Mondrian

PRO	CONTRA
Easy nesting because of One Rendering Tree	Graphic Elements can't be copied
Graph-logic is already implemented and ready for use	
Deep integration of visualization within the environment	

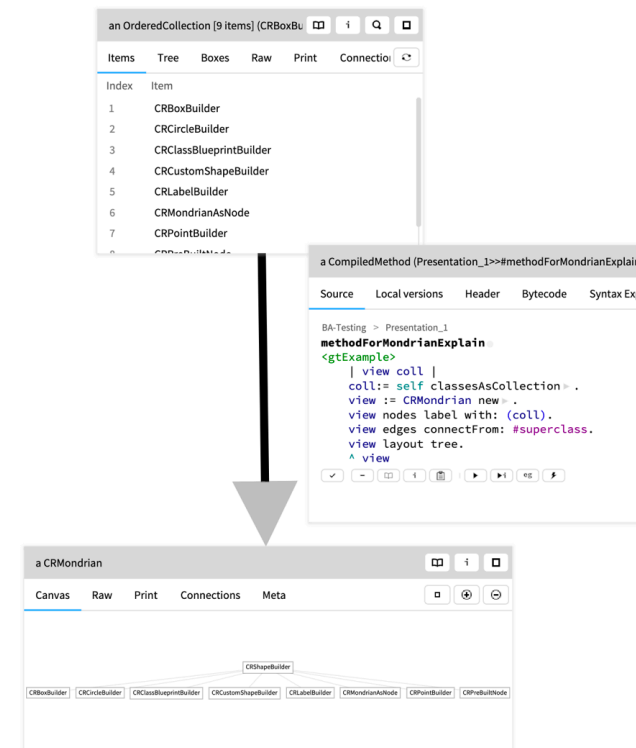
Is GT better for Mondrian

- Glamorous Toolkit is definitely a suitable environment to implement Mondrian within
- Since it is very visualization focused it is easier to implement certain parts of Mondrian

Summary

What is Mondrian?

- Developed in 2006 by Michael Meyer and Tudor Gîrba
- Lightweight tool to create interactive visualizations with simple scripts



3

CRMondrian

- Very similar to original Mondrian
- Ability to create new shapeBuilder to make complicated scripts easier
- Support for multiple shapeBuilders
- Support for Actions

6

Is GT better for Mondrian

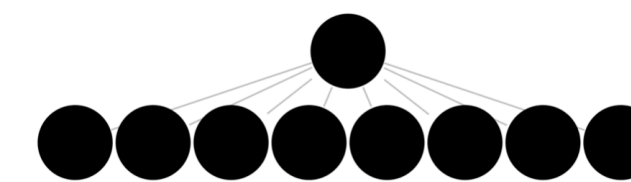
- Glamorous Toolkit is definitely a suitable environment to implement Mondrian within
- Since it is very visualization focused it is easier to implement certain parts of Mondrian

12

Is it easier to use than GtMondrian

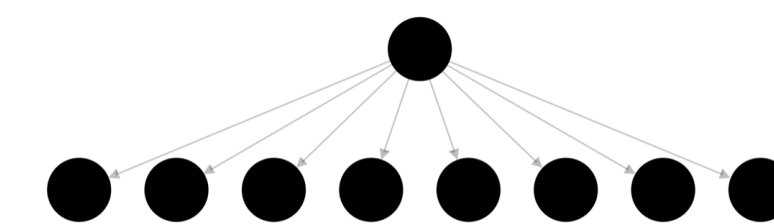
GtMondrian

```
gtMondrianPointLabelColor
<gtExample>
| m |
m := GtMondrian new .
m nodes shape: [ :x | BIElement new =
  geometry: BICircle new =
  background: Color black =
  aptitude: (BGLamorousWithTooltipAptitude new =
  showDelay: 0 =
  hideDelay: 0 =
  contentStencil: [BLabel new = aptitude:
    BGLamorousLabelAptitude new =
    glamorousRegularFontAndSize =
    alignCenter =
    text: x)
];
with: self classesAsCollection =
  glamorousRegularFontAndSize =
  layout tree.
```



CRMondrian

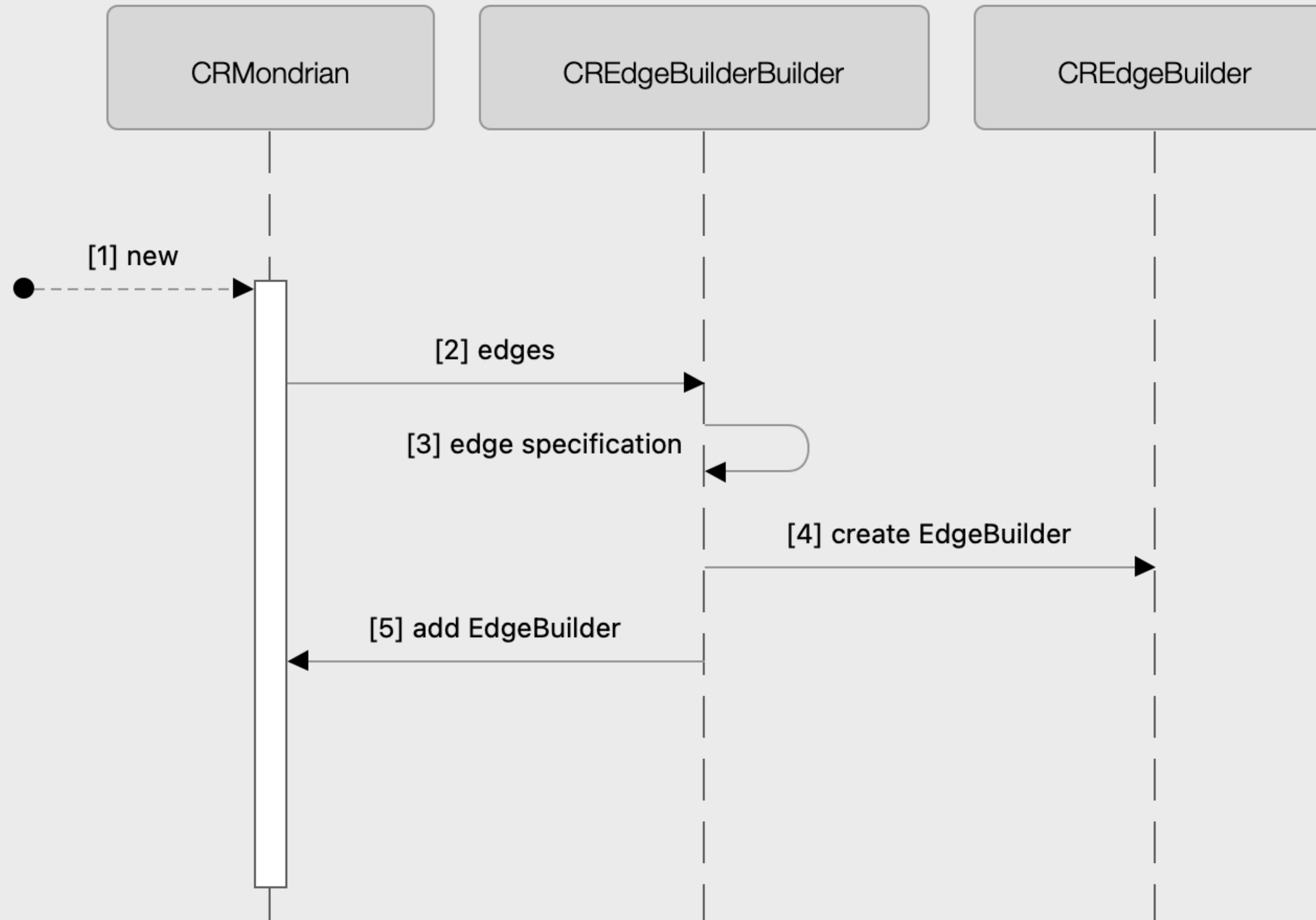
```
crMondrianPointLabelColor
<gtExample>
| m |
m := CRMondrian new .
m nodes
  point
  with: self classesAsCollection =
  glamorousRegularFontAndSize =
  edges arrow; connectFrom: #superclass.
  layout tree.
^m
```



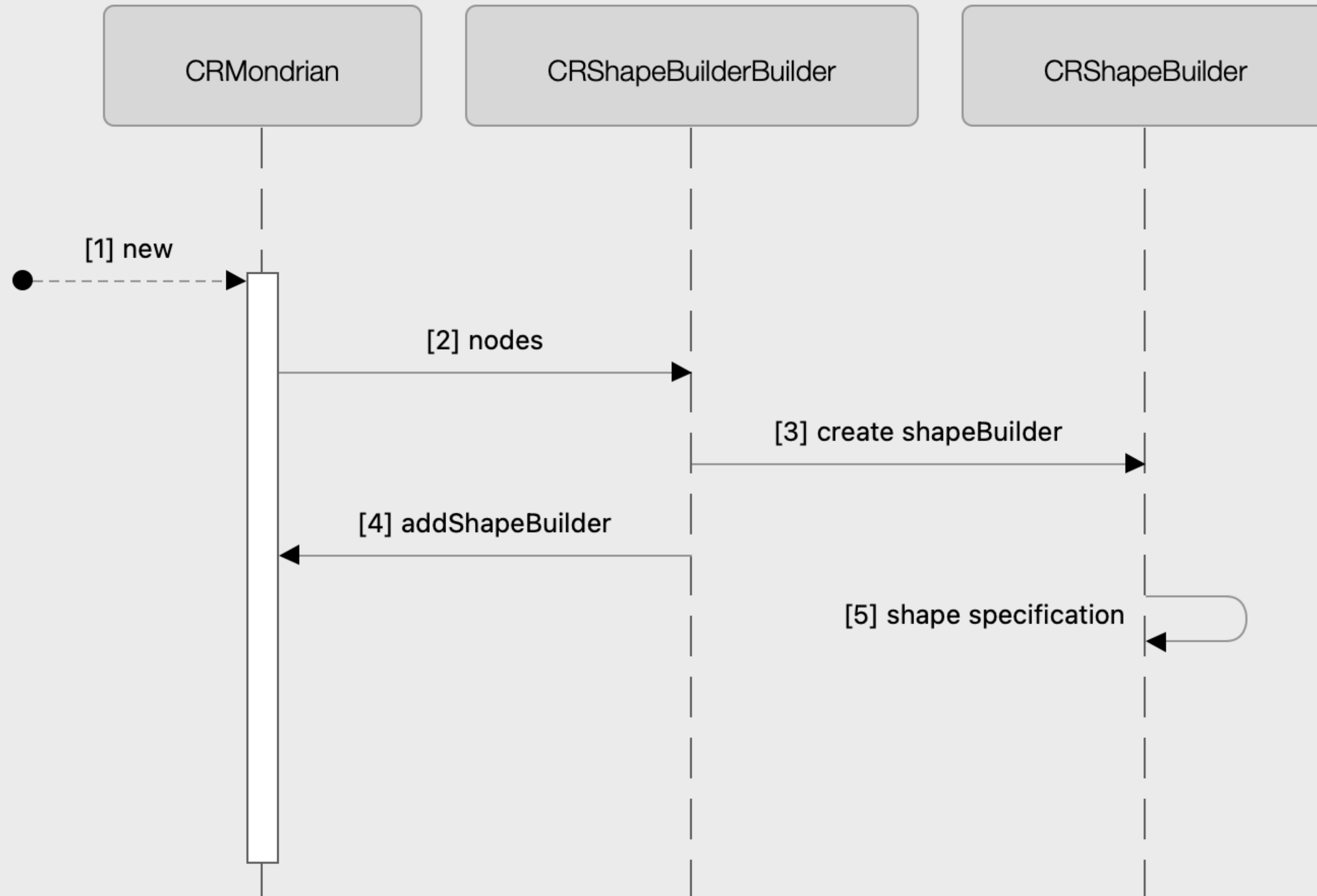
10

Backup

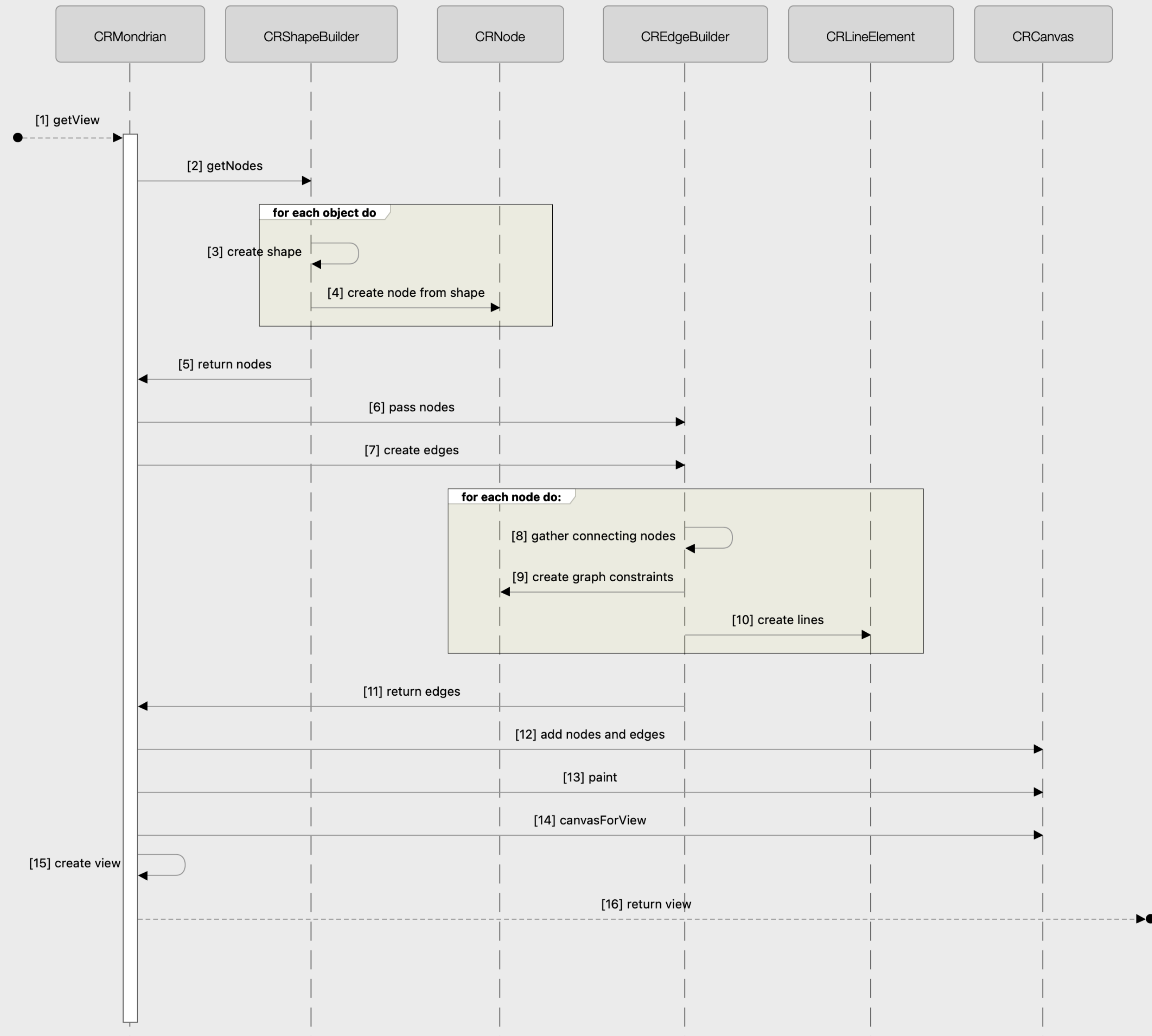
Edge Builder creation



Shape Builder creation



Create Graph



What is Glamorous Toolkit

- Entirely new Pharo environment
- Based on Moldable Development
- New graphical stack: One rendering tree

What is the one rendering tree

- Graphical stack
- All elements live within the same graphical stack
- No difference between graphs and UI
=> good interaction between the two