

SMA: Software Modeling and Analysis

Practical Session
Week 09

Assignment 09

Discussion

A09 - Exercise 01 | Roassal Visualization

Smalltalk *coding*.

Build a *sunburst visualization* to analyze test coverage of the Collection class hierarchy.



A09 - Exercise 01 | Roassal Visualization

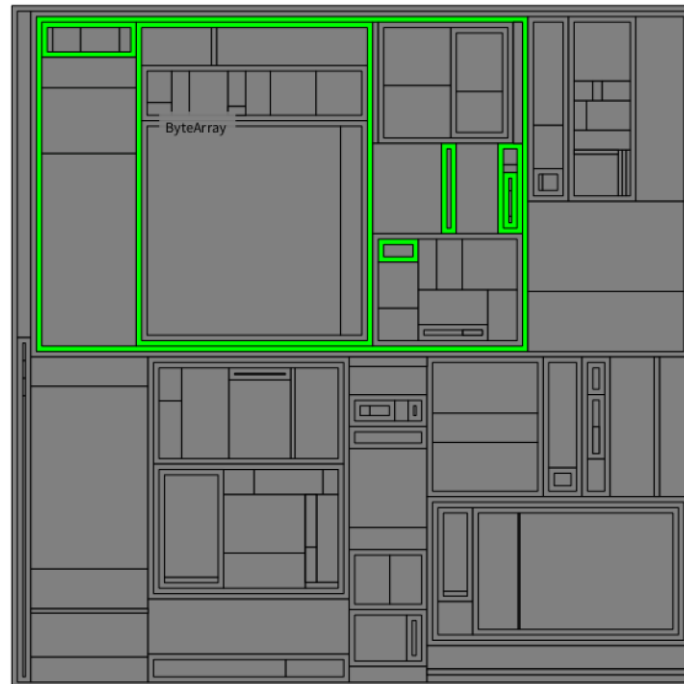


```
| b |  
b := RTSunburstBuilder new.  
b layout sunburstWithRadius: 100.  
b  
  angularSpacing: 1;  
  radialSpacing: 5.  
b shape  
  color: [ :cls |  
    (Smalltalk includesKey: (cls name , 'Test' asSymbol))  
      ifTrue: [ Color green ]  
      ifFalse: [ Color gray ] ].  
b leafWeight: [:e | e numberOfLinesOfCode].  
b explore: Collection using: #subclasses.  
b view elements  
  @ (RTLabeled new  
    color: Color black; fontSize: 20;  
    setAsShouldBeInTheFront;  
    center).  
^ b
```

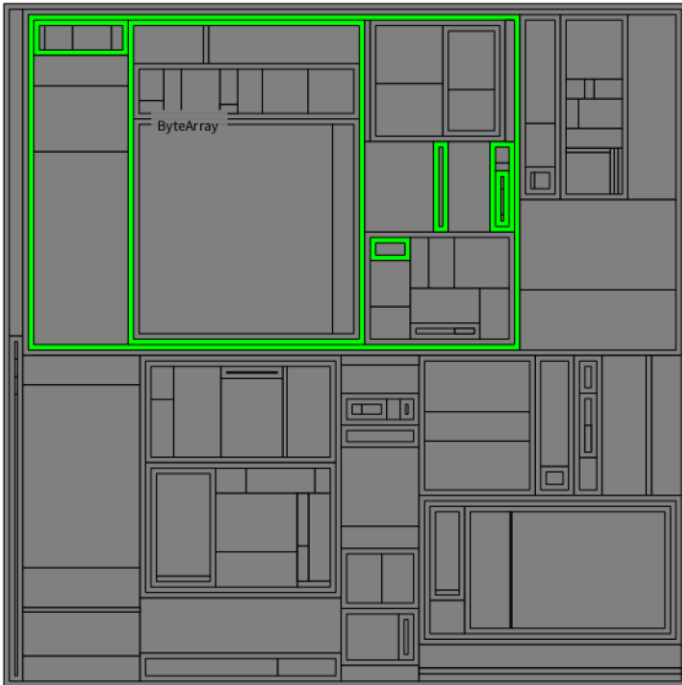
A09 - Exercise 02 | Roassal Visualization

Smalltalk *coding*.

Build a *treemap visualization* to gather an overview of classes that have subclasses, and contain the string Array in their names.



A09 - Exercise 02 | Roassal Visualization

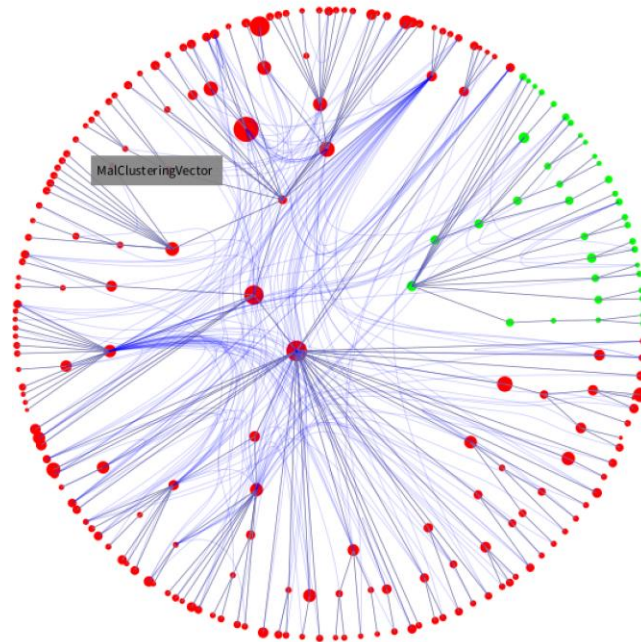


```
| b |  
b := RTTreeMapBuilder new.  
b shape  
  color: Color transparent;  
  borderColor: Color black;  
  if: [ :obj | obj isClass ]  
    color: [ :cls |  
      (cls subclasses notEmpty and:[('*Array*' match: cls name)])  
        ifTrue: [ Color green ]  
        ifFalse: [ Color gray ] ] .  
b  
  leafWeight: [:e | e methods size];  
  explore: Collection using: #subclasses.  
^ b
```

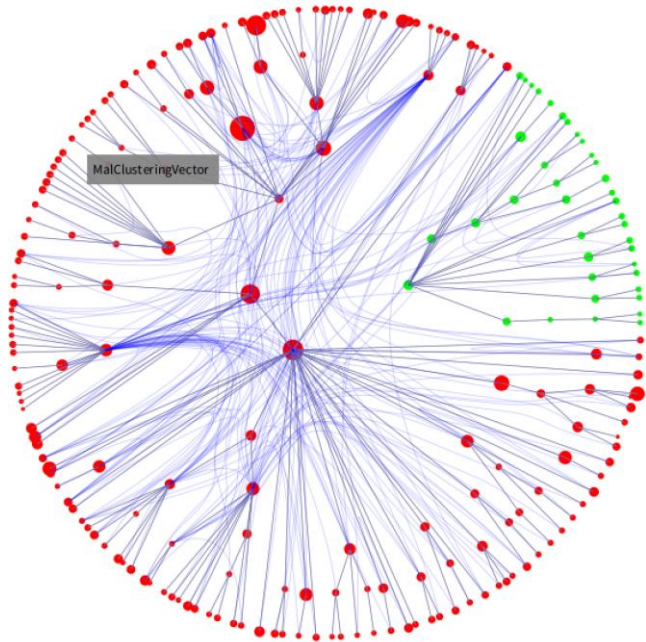
A09 - Exercise 03 | Roassal Visualization

Smalltalk *coding*.

Create a visualization using the Mondrian builder to analyze the class dependencies between the Collection class hierarchy and the RTLayout class hierarchy.



A09 - Exercise 03 | Roassal Visualization



```
| classes b |  
classes := RTLayout withAllSubclasses , Collection withAllSubclasses.  
b := RTMondrian new.  
b shape circle; color:[:cls | (RTLayout withAllSubclasses includes: cls)  
ifTrue:[Color green] ifFalse:[Color red]].  
b nodes: classes.  
b edges connectFrom: #superclass.  
b shape  
  bezierLineFollowing: #superclass;  
  color: (Color blue alpha: 0.2).  
b edges  
  notUseInLayout;  
  connectToAll: #dependentClasses.  
b normalizer  
  normalizeSize: #numberOfMethods using: #sqrt.  
b layout cluster.  
b build.  
^ b
```


A09 - Exercise 04 | Discussion

Visualization *reasoning*.

Comment on the strengths and limitations of each visualization you just created.

Sunburst visualization: Nice for hierarchies, tiles reveal relationship to parent tile, hard to evaluate manually (circular area needs is hard to estimate)

Tree map visualization: Advantages like sunburst + area easier to estimate, elements at same hierarchy-level hard to compare, exhausting manual inspection

Node-link visualization: Advantages from sunburst and tree map, supports multiple overlaid relation visualizations, very high resolution needed for further manual inspection

Assignment 10

Preview

You have to attend the lecture to reveal such slides.*



**Disclaimer:*

The content that has been shown on this slide is irrelevant for the exam.

A10 - Exercise 01 | General knowledge

Multiple-choice *crossing*.

- What is the main purpose of RMSE?
- Why are prediction error assessments important?
- What is the f-measure exactly?
- What is the difference between the terms accuracy and precision?
- Does the effort required to raise the test coverage increase linearly?
- What is true regarding precision and recall with respect to anti-virus software on a computer?

A10 - Exercise 02 | WEKA

Multiple-choice *crossing*.

- What are benefits of using the WEKA tool?

A10 - Exercise 03 | Machine learning specifics

Multiple-choice *crossing*.

- Which of these statements are correct?
- What are the effects of an algorithm stuck in a local minimum?
- Which statements are correct regarding underfitting and overfitting of a model?
- What were typical dataset partition sizes used for testing and training of the bug prediction models?