Artifact Description

Identifying Refactoring Opportunities for Replacing Type Code with Subclass and State

1 Testing Auto-SCST

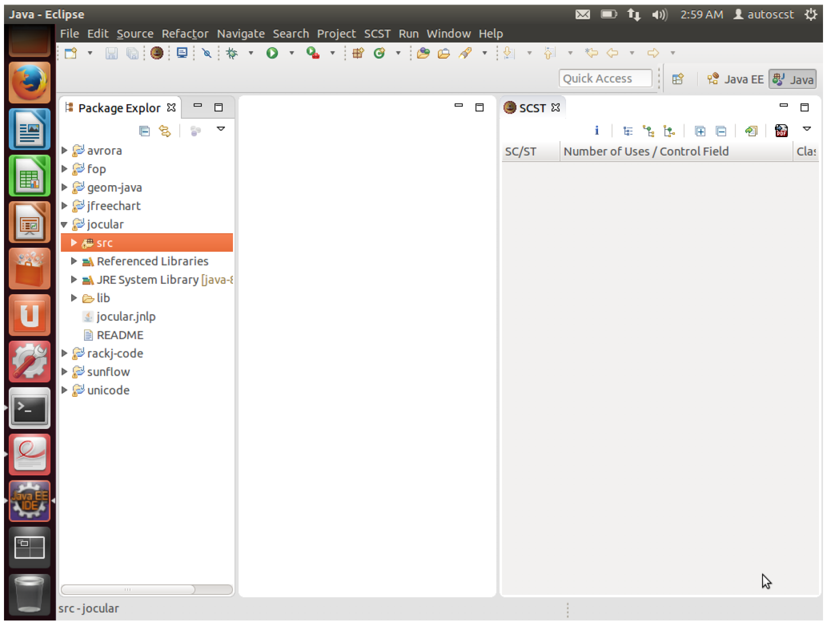
1. Download the plug-in (jar file) and move to the plugins folder of your Eclipse application. This will install the Auto-SCST plugin.

2. In the terminal, change to the Eclipse directory and run:
   ```
   $ ./eclipse -vmargs -Xmx4000m 2>/dev/null 1>/dev/null &
   ```
   This will start the Eclipse application.

3. A new entry SCST will be added to the menu bar of your Eclipse.

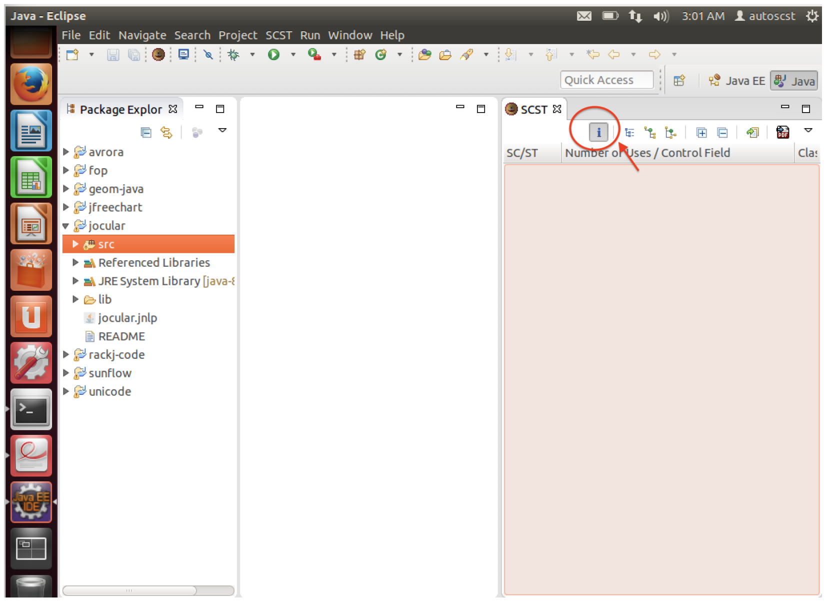
4. Click on SC/ST Refactoring item in the drop down list to see a new tab (view) on your editor.

5. Select the source folder of an input application (for example, double-click on `jOcular` application; next click on `src` folder). The applications used for evaluation in the paper can be downloaded from `https://github.com/anony-user/Auto-SCST`.

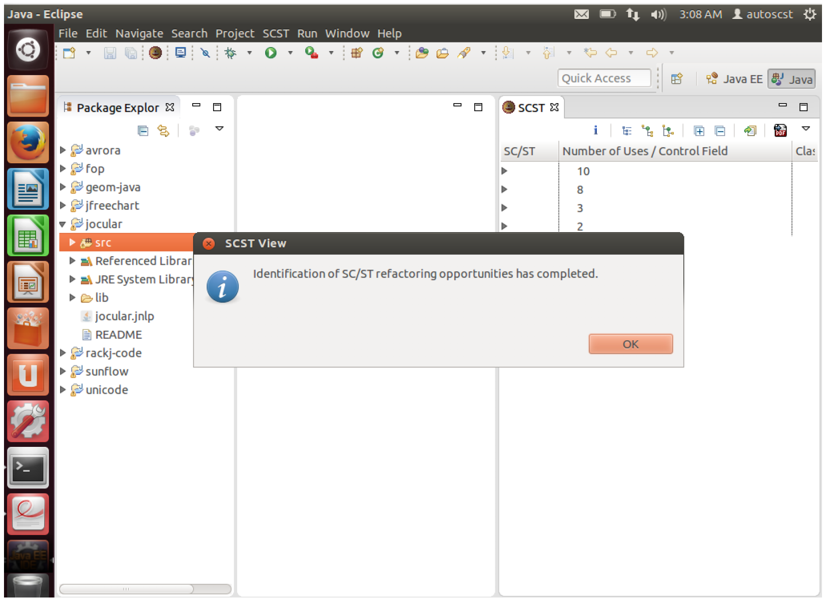


6. Invoke the identification step by clicking on the “i” button.

   “i” button remains pressed while the opportunity-identification is in process. This step may take approximately 1 to 5 minutes depending on the input application. Wait till the dialog box appears on the screen notifying the completion of this step.



7. Click on “OK” button.
   The identified opportunities are displayed in groups in the SCST tab. The numbers (for each group) displayed in the SCST tab represent the #uses = number of conditional-statements associated with a control-field.

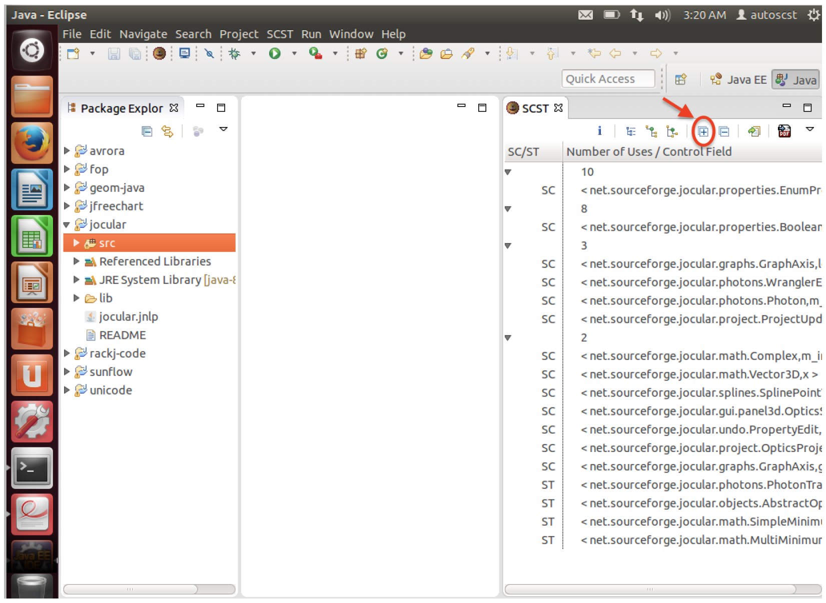


SCST View

Identification of SC/ST refactoring opportunities has completed.

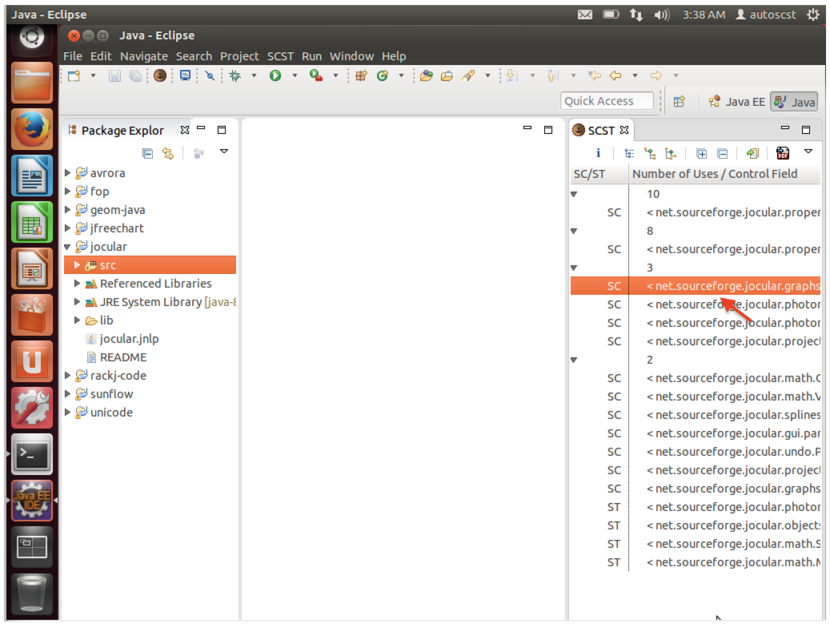
SC/ST	Number of Uses / Control Field
	10
	8
	3
	2

8. Click on “+” icon to expand the list of identified refactoring opportunities.



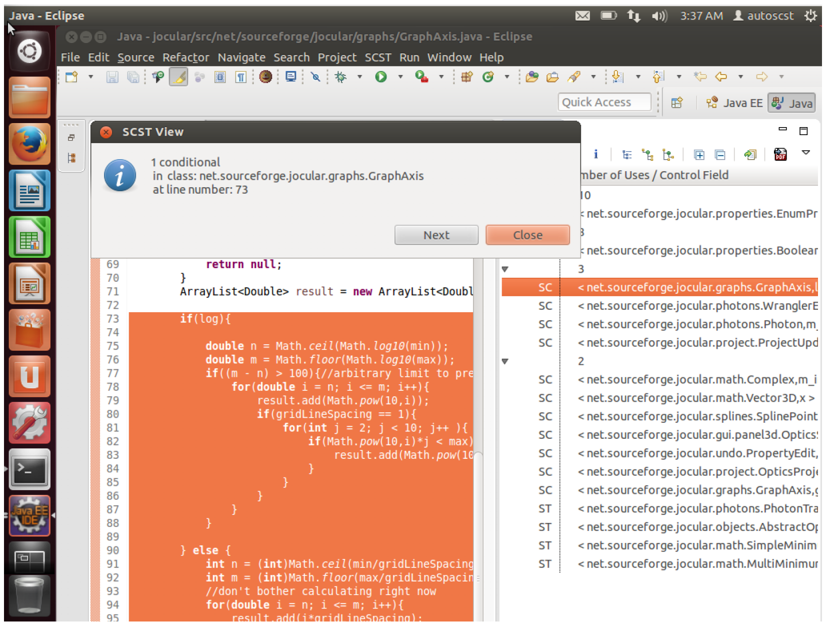
SC/ST	Number of Uses / Control Field
	10
SC	< net.sourceforge.jocular.properties.EnumPr
	8
SC	< net.sourceforge.jocular.properties.Boolear
	3
SC	< net.sourceforge.jocular.graphs.GraphAxis,l
SC	< net.sourceforge.jocular.photons.WranglerE
SC	< net.sourceforge.jocular.photons.Photon,m_
SC	< net.sourceforge.jocular.project.ProjectUpd
	2
SC	< net.sourceforge.jocular.math.Complex,m_i
SC	< net.sourceforge.jocular.math.Vector3D,x >
SC	< net.sourceforge.jocular.splines.SplinePoint
SC	< net.sourceforge.jocular.gui.panel3d.Optics
SC	< net.sourceforge.jocular.undo.PropertyEdit,
SC	< net.sourceforge.jocular.project.OpticsProje
SC	< net.sourceforge.jocular.graphs.GraphAxis,
ST	< net.sourceforge.jocular.photons.PhotonTra
ST	< net.sourceforge.jocular.objects.AbstractOp
ST	< net.sourceforge.jocular.math.SimpleMinim
ST	< net.sourceforge.jocular.math.MultiMinimur

9. Observe the identified refactoring opportunities in the SCST tab.
   Double-click on any one of identified refactoring opportunities to see its associated conditional-statements. For example, the third entry in the identified list of opportunities is double-clicked in the figure below. You may have to move the dialog box on the editor to see the code.



SC/ST	Number of Uses / Control Field
	10
SC	< net.sourceforge.jocular.proper
	8
SC	< net.sourceforge.jocular.proper
	3
SC	< net.sourceforge.jocular.graphs
SC	< net.sourceforge.jocular.photor
SC	< net.sourceforge.jocular.photor
SC	< net.sourceforge.jocular.project
	2
SC	< net.sourceforge.jocular.math.C
SC	< net.sourceforge.jocular.math.V
SC	< net.sourceforge.jocular.splines
SC	< net.sourceforge.jocular.gui.par
SC	< net.sourceforge.jocular.undo.P
SC	< net.sourceforge.jocular.project
SC	< net.sourceforge.jocular.graphs
ST	< net.sourceforge.jocular.photor
ST	< net.sourceforge.jocular.objects
ST	< net.sourceforge.jocular.math.S
ST	< net.sourceforge.jocular.math.M

Click on "Next" button to see next conditional-statement.
Note that each control-field may have multiple associated conditional-statements. For example, the one shown in the below figure shows the first of the three associated conditional-statements. And if refactoring is performed (outside the scope of the current paper) on this identified opportunity then these three conditional-statements can be replaced by polymorphic calls.

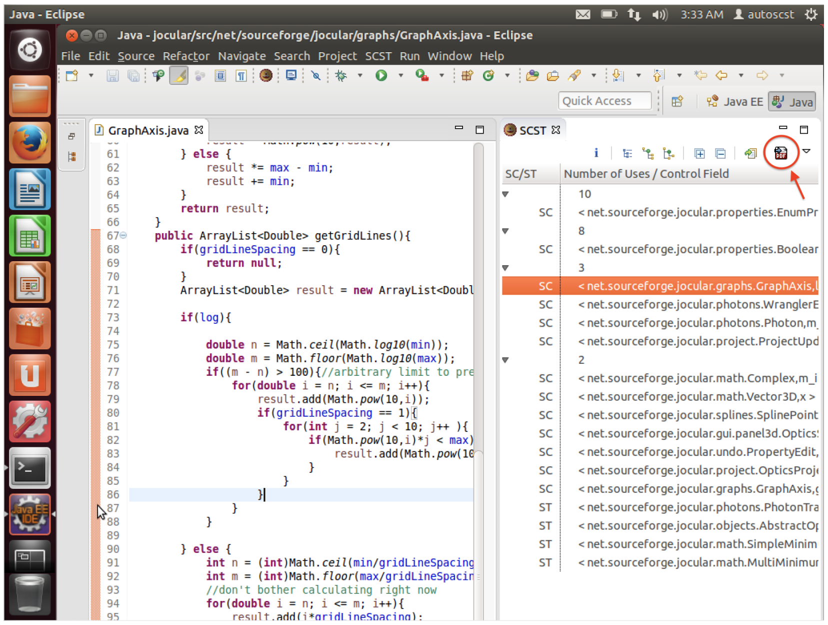

Java - Eclipse
3:37 AM
autoscst
Java - jocular/src/net/sourceforge/jocular/graphs/GraphAxis.java - Eclipse
File Edit Source Refactor Navigate Search Project SCST Run Window Help
Quick Access
Java EE
Java
SCST View
Next
Close

1 conditional
in class: net.sourceforge.jocular.graphs.GraphAxis
at line number: 73

```
69                return null;
70            }
71            ArrayList<Double> result = new ArrayList<Doubl
72
73            if(log){
74
75                double n = Math.ceil(Math.log10(min));
76                double m = Math.floor(Math.log10(max));
77                if((m - n) > 100){//arbitrary limit to pre
78                    for(double i = n; i <= m; i++){
79                        result.add(Math.pow(10,i));
80                        if(gridLineSpacing == 1){
81                            for(int j = 2; j < 10; j++ ){
82                                if(Math.pow(10,i)*j < max)
83                                    result.add(Math.pow(10
84                                }
85                            }
86                        }
87                    }
88                }
89
90            } else {
91                int n = (int)Math.ceil(min/gridLineSpacing
92                int m = (int)Math.floor(max/gridLineSpacin
93                //don't bother calculating right now
94                for(double i = n; i <= m; i++){
95                    result.add(i*gridLineSpacing);
```

SC/ST	Number of Uses / Control Field
	10
	< net.sourceforge.jocular.properties.EnumPr
	8
	< net.sourceforge.jocular.properties.Boolear
	3
SC	< net.sourceforge.jocular.graphs.GraphAxis,l
SC	< net.sourceforge.jocular.photons.WranglerE
SC	< net.sourceforge.jocular.photons.Photon,m_
SC	< net.sourceforge.jocular.project.ProjectUpd
	2
SC	< net.sourceforge.jocular.math.Complex,m_i
SC	< net.sourceforge.jocular.math.Vector3D,x >
SC	< net.sourceforge.jocular.splines.SplinePoint
SC	< net.sourceforge.jocular.gui.panel3d.Optics
SC	< net.sourceforge.jocular.undo.PropertyEdit,
SC	< net.sourceforge.jocular.project.OpticsProj
SC	< net.sourceforge.jocular.graphs.GraphAxis,
ST	< net.sourceforge.jocular.photons.PhotonTra
ST	< net.sourceforge.jocular.objects.AbstractO
ST	< net.sourceforge.jocular.math.SimpleMinim
ST	< net.sourceforge.jocular.math.MultiMinimur

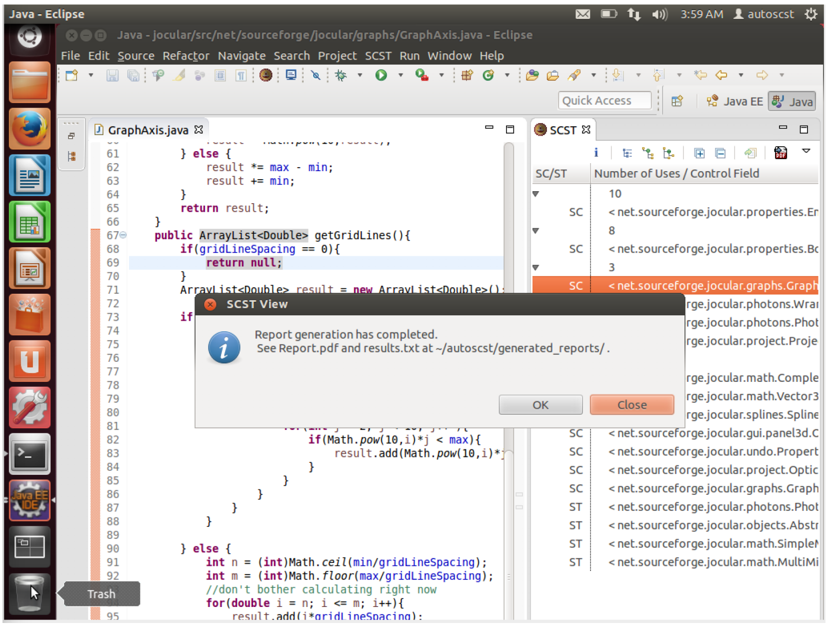
10. For a chosen application, we can generate reports on the identified refactoring opportunities. Generate reports by clicking on the "Generate PDF" icon. Two files are generated – $benchmarkName_Report.pdf and $benchamarkName_results.txt. PDF will contain some details about the input application and the identified refactoring opportunities. Text file will contain timing, cyclomatic complexity and details on the marked control-fields.


Java - Eclipse
3:33 AM autoscst
Java - jocular/src/net/sourceforge/jocular/graphs/GraphAxis.java - Eclipse
File Edit Source Refactor Navigate Search Project SCST Run Window Help
Quick Access
Java EE
Java
GraphAxis.java
SCST

```java
        } else {
            result *= max - min;
            result += min;
        }
        return result;
    }
    public ArrayList<Double> getGridLines(){
        if(gridLineSpacing == 0){
            return null;
        }
        ArrayList<Double> result = new ArrayList<Doubl

        if(log){

            double n = Math.ceil(Math.log10(min));
            double m = Math.floor(Math.log10(max));
            if((m - n) > 100){//arbitrary limit to pre
                for(double i = n; i <= m; i++){
                    result.add(Math.pow(10,i));
                    if(gridLineSpacing == 1){
                        for(int j = 2; j < 10; j++ ){
                            if(Math.pow(10,i)*j < max)
                                result.add(Math.pow(10
                            }
                        }
                    }
                }
            }

        } else {
            int n = (int)Math.ceil(min/gridLineSpacing
            int m = (int)Math.floor(max/gridLineSpacin
            //don't bother calculating right now
            for(double i = n; i <= m; i++){
                result.add(i*gridLineSpacing);
```

SC/ST	Number of Uses / Control Field
	10
SC	< net.sourceforge.jocular.properties.EnumPr
	8
SC	< net.sourceforge.jocular.properties.Boolear
	3
SC	< net.sourceforge.jocular.graphs.GraphAxis,l
SC	< net.sourceforge.jocular.photons.WranglerE
SC	< net.sourceforge.jocular.photons.Photon,m_
SC	< net.sourceforge.jocular.project.ProjectUpd
	2
SC	< net.sourceforge.jocular.math.Complex,m_i
SC	< net.sourceforge.jocular.math.Vector3D,x >
SC	< net.sourceforge.jocular.splines.SplinePoint
SC	< net.sourceforge.jocular.gui.panel3d.Optics:
SC	< net.sourceforge.jocular.undo.PropertyEdit,
SC	< net.sourceforge.jocular.project.OpticsProj
SC	< net.sourceforge.jocular.graphs.GraphAxis,(
ST	< net.sourceforge.jocular.photons.PhotonTra
ST	< net.sourceforge.jocular.objects.AbstractO
ST	< net.sourceforge.jocular.math.SimpleMinim
ST	< net.sourceforge.jocular.math.MultiMinimur

Reports will be generated at '∼/generated_reports'.


Java - Eclipse
3:59 AM autoscst
Java - jocular/src/net/sourceforge/jocular/graphs/GraphAxis.java - Eclipse
File Edit Source Refactor Navigate Search Project SCST Run Window Help
Quick Access
Java EE
Java
GraphAxis.java
SCST
SCST View
OK
Close
Trash

Report generation has completed.
See Report.pdf and results.txt at ~/autoscst/generated_reports/ .

11. One may repeat the steps 3-8 for each application used in the evaluation (javaGeom, jocular, Rackj, sunflow, UR, jfreechart, avrora, and fop) or any new application of their choice.

2 Steps to establish the results of Figure 13 (Page 18, Section 5) and Figure 14 (Page 20, Section 5.1.3)

Generate the reports for all the input applications.

(1) Perform steps 5-8 as shown in Section 1. Note that the applications used for evaluation in the paper can be downloaded from `https://github.com/anony-user/Auto-SCST`. The results in the paper are collected by generating the results on following source folders:

- `jocular/src/`
- `jfreechart/source/`
- `rackj-code/tags/trunk/`
- `geom-java/src/main/`
- `unicode/src/`
- `avrora/src/`
- `fop/src/`
- `sunflow/src/`

(2) Open the generated pdf on the command line.
For example, `$ evince ~/generated_reports/jocular_Report.pdf`.

- The PDF has two sections: input benchmark characteristics and the listing of the identified control fields.
- The input benchmark characteristics should match with that shown in Figure 13 of the paper. For example, the enumerated entries 1-12 correspond to the entries in Figure 13, columns 3-14.

(3) Open the generated text file on the command line.
For example, `$ vim ~/generated_reports/jocular_results.txt`.

- The text file has three sections: time taken for the identification step, improvement in the McCabe's cyclomatic complexity and the details on marked control-fields.
- Timing is the representative of column 17 in Figure 13. Cyclomatic complexity numbers should match the columns of Figure 14. Results on marked control-fields should match with column 15 (Marked control-fields) and column 16 (%MCF) of Figure 13.

3 Trouble shooting

1. Restart the eclipse application if it throws 'out of memory' error.
2. The directory `~/generated_reports` must be present for the reports to be generated. Otherwise, an error may be thrown.

4 Note

1. The tool may throw some errors when tested on new projects (other than tested benchmarks) if the projects use external libraries.
2. The tool is still a prototype implementation and can easily be extended to handle all cases.
3. Other supplementary material is present at `https://github.com/anony-user/Auto-SCST`.