

Building Curves and Charts with EyeSee

Alexandre Bergel
abergel@dcc.uchile.cl
11/09/2012

EyeSee

Charts and curves are effective in representing a collection of numerical values

EyeSee is an engine for scripting diagram drawing

It allows for drawing chart, diagrams, curves and histogram

EyeSee offers a small domain specific language, in the same spirit than Mondrian and Glamour

EyeSee by example

We will successively detail a number of examples that cover most of EyeSee feature.

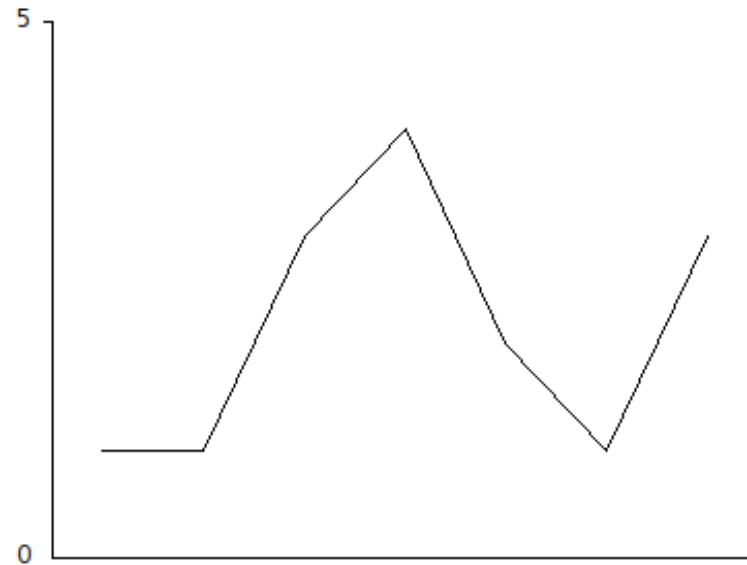
The class ESExample contains many examples, check them out!

Diagramming, simply

Plotting, nothing more

Simply sending `#openPlot` to a collection of numerical value do what you would expect

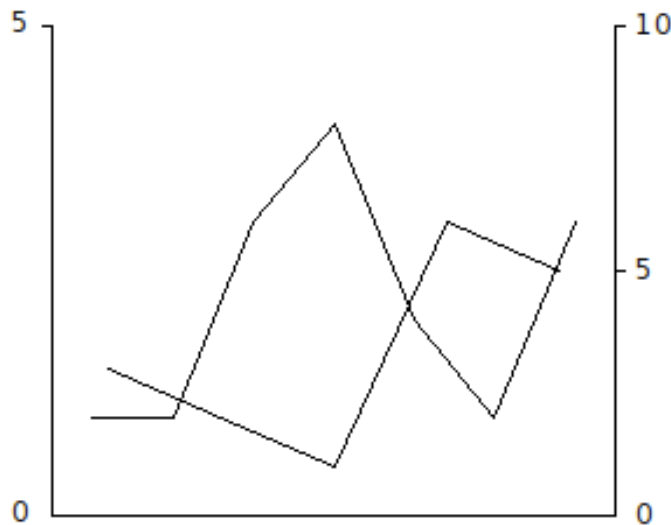
```
#(1 1 3 4 2 1 3) openPlot
```



Plotting twice more

Similarly, `#openPlotWith:` plot to ordered set of numerical values

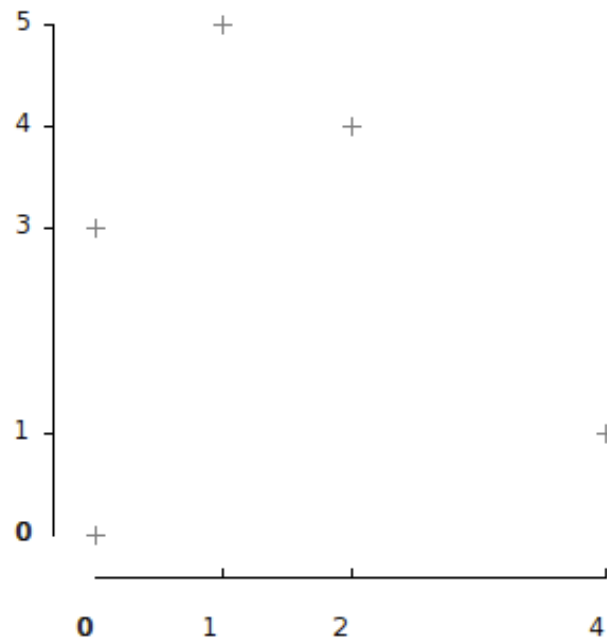
```
#(1 1 3 4 2 1 3) openPlotWith: #(3 2 1 6 5)
```



Scattering the World!

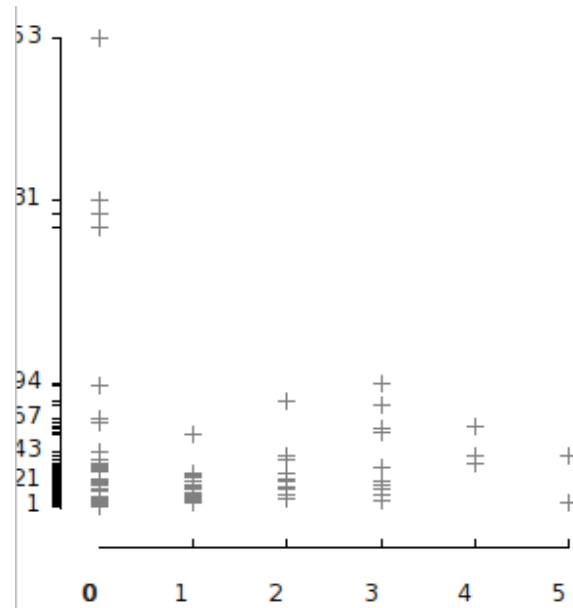
Sending `#openScatterPlot` to a collection of point will do what you would expect

```
{(0@0). (0@3). (1@5). (4@1). (2@4)}  
openScatterPlot
```



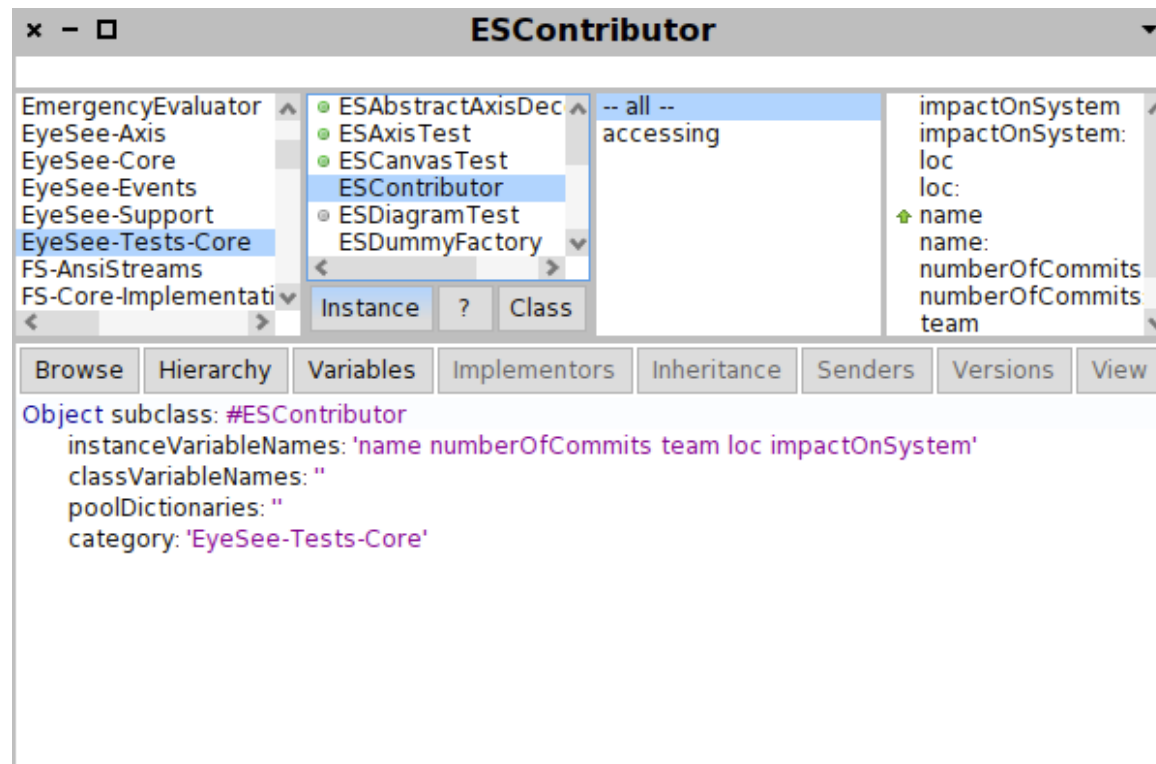
Scattering some classes

```
| data |  
data := Collection withAllSubclasses  
  collect: [:cls |  
    cls numberOfAttributes @  
    cls numberOfMethods ].  
data openScatterPlot
```



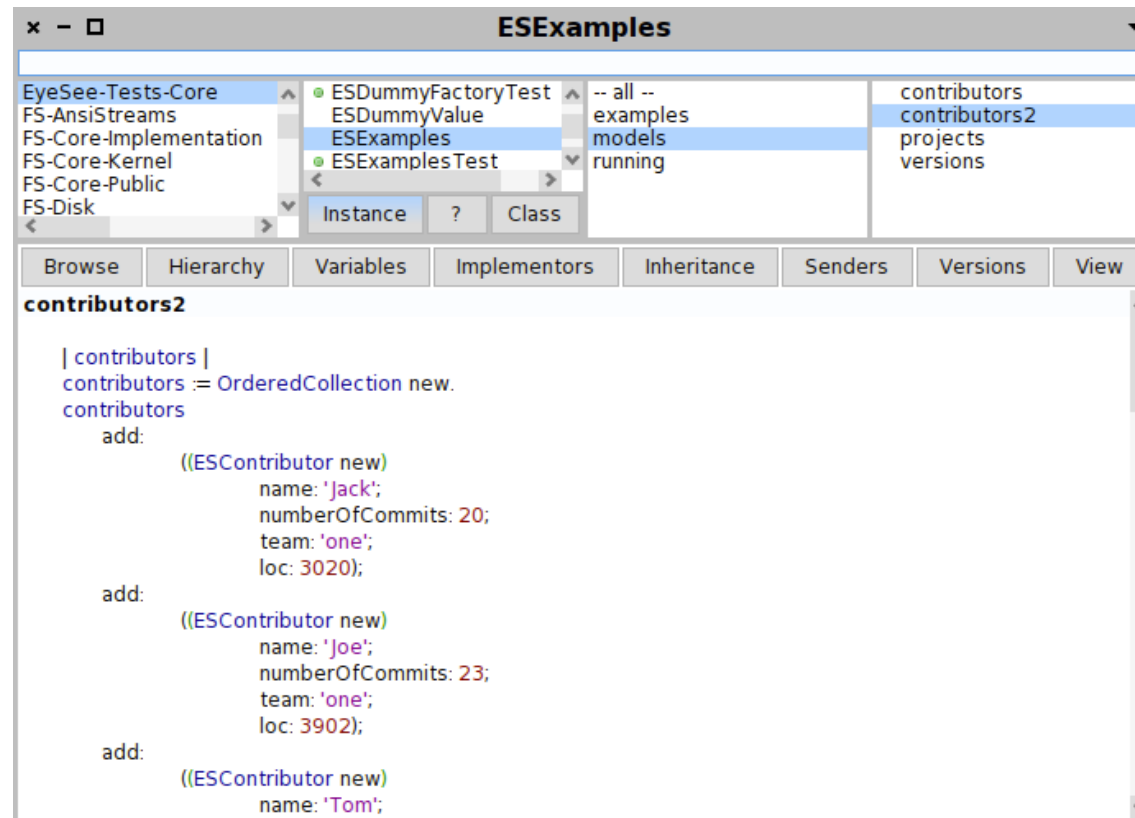
Modeling your values

Modeling contribution



ESContributor is simply a data structure used
to bound data together

Collecting the data



Instances of ESContributor are used as the visualized model in the next slides

Emphasis on differences with a deviation diagram

Try in a workspace

```
| diag |  
diag := ESDiagramRenderer new.  
diag deviationDiagram  
  y: #loc;  
  models: ESExamples new contributors2.  
diag open
```

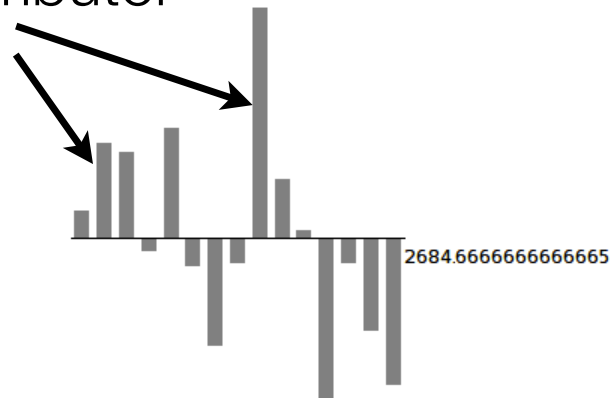


Emphasis on differences with a deviation diagram

Try in a workspace

```
| diag |  
diag := ESDiagramRenderer new.  
diag deviationDiagram  
  y: #loc;  
  models: ESExamples new contributors2.  
diag open
```

instances of ECContributor

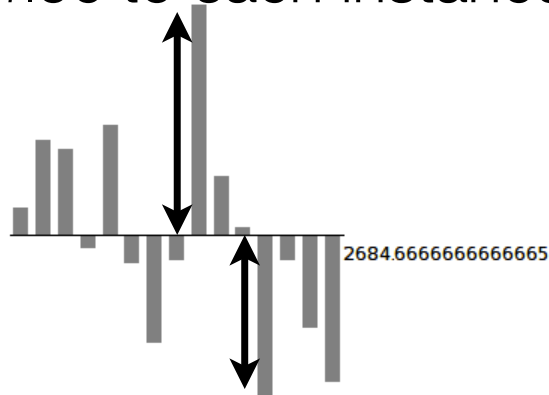


Emphasis on differences with a deviation diagram

Try in a workspace

```
| diag |  
diag := ESDiagramRenderer new.  
diag deviationDiagram  
  y: #loc;  
  models: ESExamples new contributors2.  
diag open
```

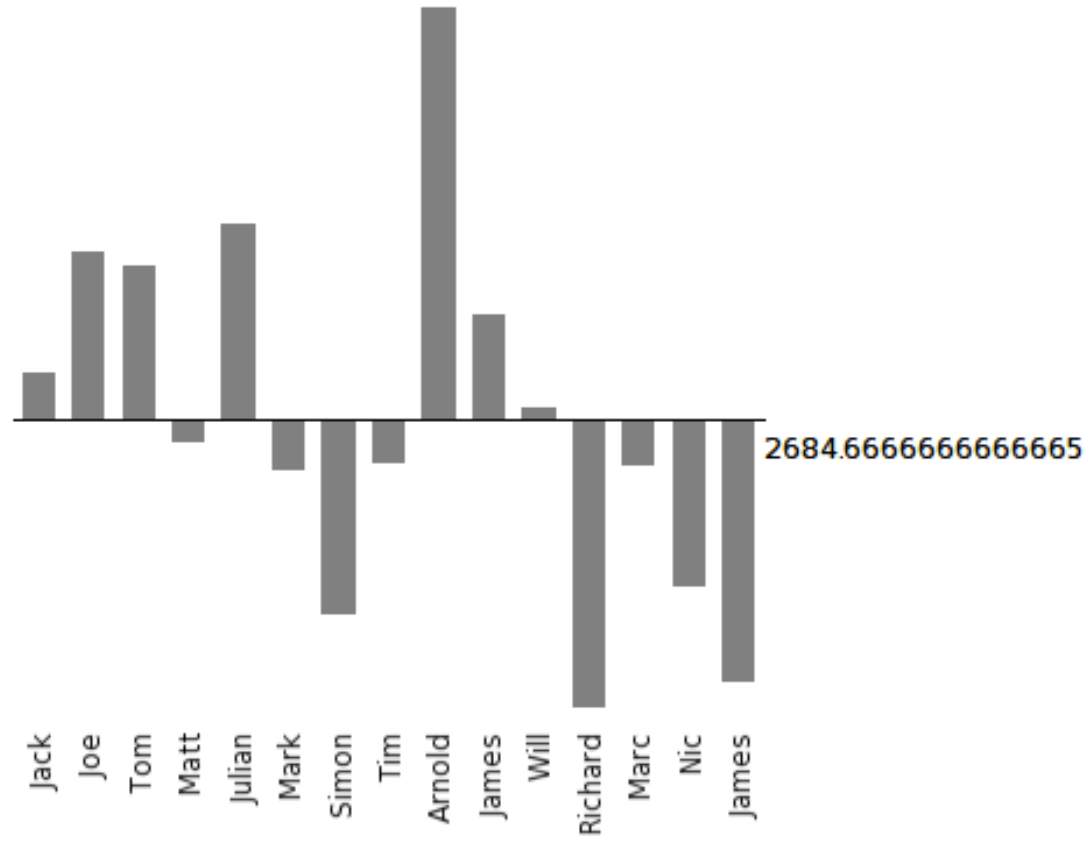
result of sending #loc to each instance



Emphasis on differences with a deviation diagram

Adding names and a size

```
| diag |  
diag := ESDiagramRenderer new.  
diag deviationDiagram  
  y: #loc;  
  identifier: #name;  
  width: 500;  
  height: 400;  
  rotatedLabels: true;  
  models: ESEExamples new contributors2.  
diag open
```



Emphasis on negative values

Adding names and a size

```
| diag |  
diag := ESDiagramRenderer new.  
diag deviationDiagram  
  y: #loc;  
  identifier: #name;  
  width: 800;  
  height: 400;  
  highlightAboveDeviation;  
  labelsInPercent;  
  valueAxis;  
  rotatedLabels: true;  
  models: ESExamples new contributors2.  
diag open.
```

+111.35000000000001%

+52.72%

+41.54%

+28.32%

+12.49%

+2.99%

-6.66%

-45.77%

-71.69%

-78.47%

Jack

Joe

Tom

Matt

Julian

Mark

Simon

Tim

Arnold

James

Will

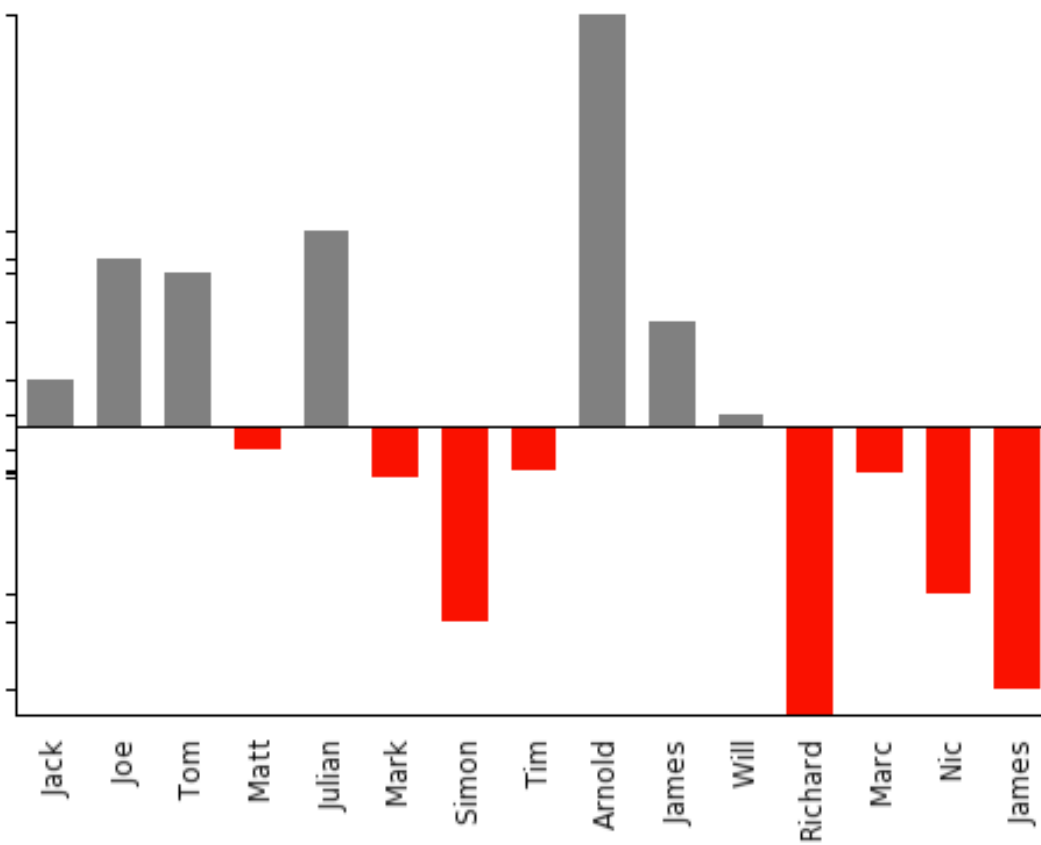
Richard

Marc

Nic

James

2684.6666666666665

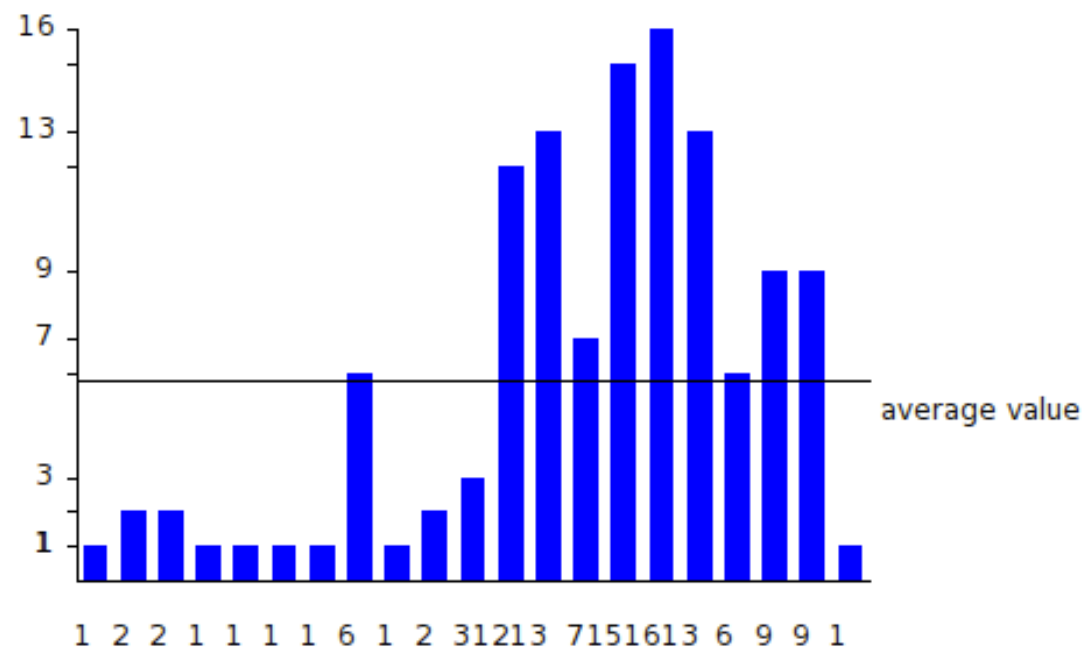


Adding a deviation line

It is often convenient to add a line to indicate a threshold or a particular value.

```
| diagram v |  
v := (1 to: 21) collect: #atRandom.  
diagram := ESDiagramRenderer new.  
(diagram verticalBarDiagram)  
  y: [:each | each];  
  labels: #yourself;  
  identifier: #yourself;  
  defaultColor: Color blue;  
  valueAxis;  
  width: 500;  
  deviationValue: v average;  
  deviationDescription: 'average value';  
  rotatedLabels: false;  
  models: v.  
diagram open
```

Adding a deviation line

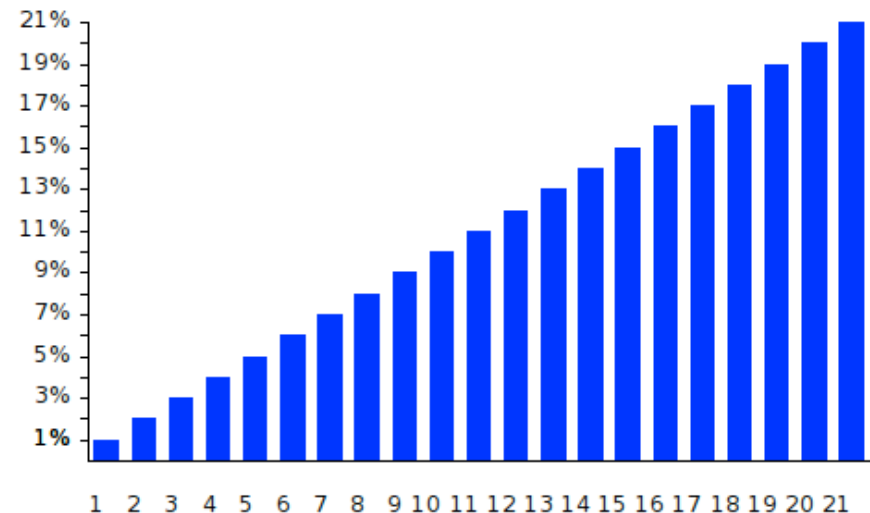


Shaping your curve

Labels on X-axis

This is what #labels: is used for!

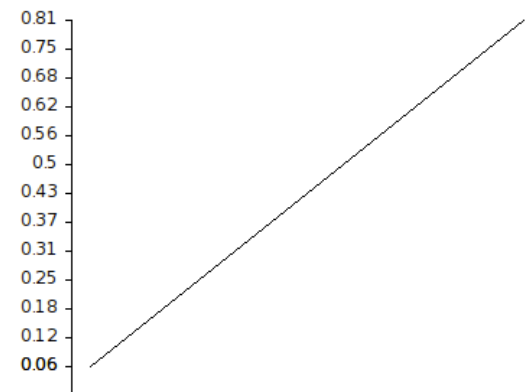
```
| diagram |  
diagram := ESDiagramRenderer new.  
(diagram verticalBarDiagram)  
  y: [:each | each];  
  labels: [ :each | each value asString, '%' ];  
  identifier: #yourself;  
  defaultColor: Color blue;  
  valueAxis;  
  width: 500;  
  rotatedLabels: false;  
  models: (1 to: 21).  
diagram open
```



Float with 2 digit decimal

Truncating a float is done via `#asTruncatedFloat`. This is quite handy if you want to have readable axis labels. Try to remove `#labels:` in the following

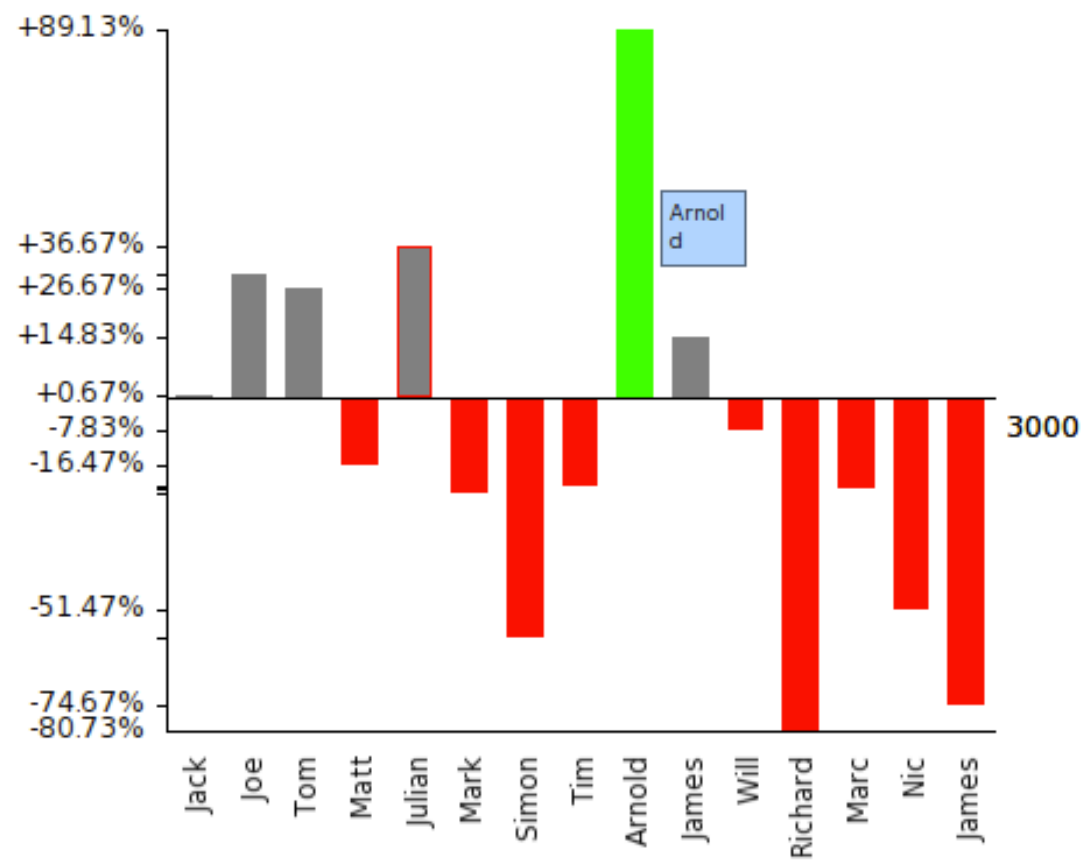
```
| diagram |  
diagram := ESDiagramRenderer new.  
(diagram lineDiagram)  
  y: #yourself;  
  labels: [ :each | each value asTruncatedFloat ];  
  valueAxis;  
  models: (1 to: 13) / 16.  
diagram open
```



Adding a bit of interaction

Interaction are easily defined

```
| diag |  
diag := ESDiagramRenderer new.  
(diag deviationDiagram)  
  y: #loc;  
  identifier: #name;  
  width: 500;  
  height: 400;  
  deviationValue: 3000;  
  highlightAboveDeviation;  
  labelsInPercent;  
  valueAxis;  
  rotatedLabels: true;  
  models: ESEExample new contributors2.  
  
diag interaction  
  strongHighlightWhenOver;  
  popupText: #name.  
  
diag open
```



Links and resources

Links

EyeSee was originally built by Matthias Junker and Markus Hofstetter in VW as part of their Bachelor work. The code was ported to Pharo by Andre Hora.

<http://www.moosetechnology.org/tools/eyesee>