Adaptable Visualisation based on user needs

Leonel Merino
merino @ iam.unibe.ch

Software Composition Group
University of Bern
Leonel Merino
Chilean

Who am I?

Bachelor from University of Chile
Working at industry

Master of Science (EMN - VUB)
Working at industry

March 2014 PhD student at SCG - U. of Bern
Agenda

- use cases
  - visualisations
    - demand expertise
      - taxonomy
      - DSL
    - some don't scale well
      - divide & conquer
  - survey user needs
Users need to see code in different ways

Use case: a researcher wants to analyse polymorphism and call sites at class level on a corpus.
Visualisations give context
Visualisations may simplify complexity
Visualisations not always make things simple
Another way to do the same

| aROView view windowSize := 2400 @ 1300. |
| aROView := R OV iew new. |
| view := ROM on di anViewBuilder view: aROView. |
| aView shape circle |
| if: [:aClass | self isPolyClass: aClass ] fillColor: Color green; |
| if: [:aClass | self isCallingPolyClass: aClass ] fillColor: Color red; |
| size: [:aClass | self getSize: aClass ]. |
| aView interaction |
| on: ROMouseClick |
| do: [:event | |
| | m browser |
| | m := event element model. |
| | (self isPolyClass: m) |
| | ifTrue: [ self viewMethodBuilderFocusedOnAPolyClass: m ] |
| | ifFalse: [ |
| | (self isCallingPolyClass: m) |
| | ifTrue: [ self viewMethodBuilderFocusedOnACallSite: m ] |
| | ifFalse: [ |
| | browser := MooseNamespacesCodeBrowser new browser. |
| | browser openOn: (m mooseModel allNamespaces select: #isRoot). |
| | (browser pane port: #focusOnClass) value: m ] ] ] ]; |
| popupText: [:c | |
| | poly |
| | c name , ' (', (self getSize: c) asInteger printString , 'x)' |
| | , |
| | (poly size > 0 |
| | ifTrue: [ String cr , '---calls---', String cr , (String cr join: poly) ] |
| | ifFalse: [ '' ] ]. |
| aView nodes: model allClasses. |
| aView edges: model allClasses from: [:node1 node ] to: [:node2 node superclass ]. |
| aView forceBasedLayout |
| ROEaselMorphic new populateMenuOn: view. |
| view openInWindowSized: windowSize. |
Another way to do the same

Another way to do the same

Another way to do the same
We provide adaptable visualisation
Revisiting users need

Use case: a researcher wants to analyse polymorphism and call sites at class level on a corpus.

❖ Which is the largest system?
❖ Which system has the most polymorphism?
❖ Which system has the most call sites?
DSL for specifying user needs

visualisations DSL
  on: corpus;
  metric: heuristic;
  granularity: class;
  keyword: most.
A visualisation that fits this need
Another need

Use case: a researcher wants to analyse polymorphism and call sites at class level on a corpus.

❖ How polymorphism is distributed among classes?
❖ How call sites are distributed among classes?
❖ What about classes that presents polymorphism and call sites?
A visualisation that fits this need
Memory usage doesn’t scale well

Model (MB) | Memory used (MB)
Scaling visualisations through D&C strategy

**DIVIDE**

- Dividing a large program into smaller blocks

**ANALYSE**

- Running analysis for sorting, colouring and sizing

**VISUALISE**

- Producing a visualisation of a block

**COMPOSE**

- Composing all visualisations together
Powered by

Pharo an object-oriented language

Moose a platform for software analysis

Roassal — Quicksilver visualisation engine

Pangea enables running language independent analysis on corpora of object-oriented software
D&C allowed us to scale

Implement | Update | Analise | Visualise | Compose

Quicksilver
Roassal
Pangea
Moose
Pharo
Browser

Qualitas Corpus (14,946 KLOC ~ 2 hrs.)
Adaptable Visualisation based on user needs

Visualisations aid software analysis

demand expertise

Survey

Taxonomy

some don’t scale well

DSL

visualisations on: corpus;
metric: heuristic;
granularity: class;
keyword: most.

Leonel Merino
merino @ iam.unibe.ch