Identifying Design Problems with Code Smell Agglomerations
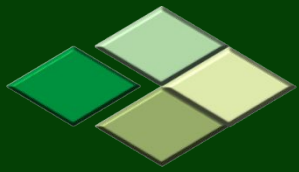
Graph approcah
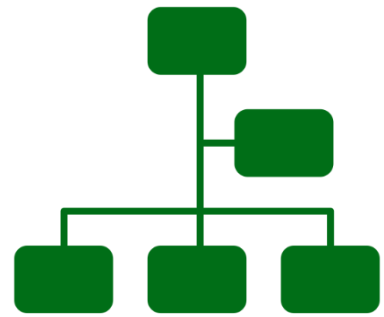
Benedicte Agbachi, Eduardo Fernandes, and Alessandro Garcia

OPUS Research Group
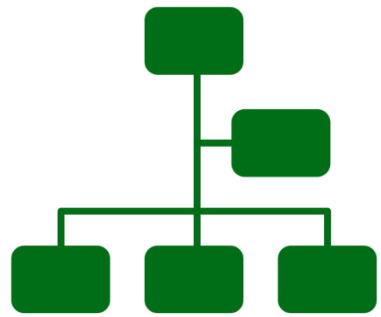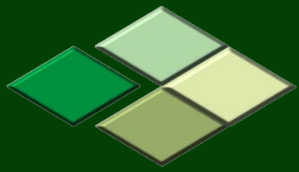
About
Graph-based

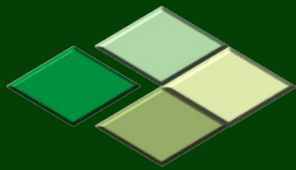**Visualizing** code smell agglomerations using graph notion

Available in an online form

The graphic visualization

+ description of code
smells

+ description of concerns

Graph-based main screen

Graph visualization

Description of code smells

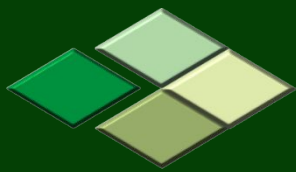Experimento   Sumários   Tipos de Code Smell   Problemas de Design   Logout

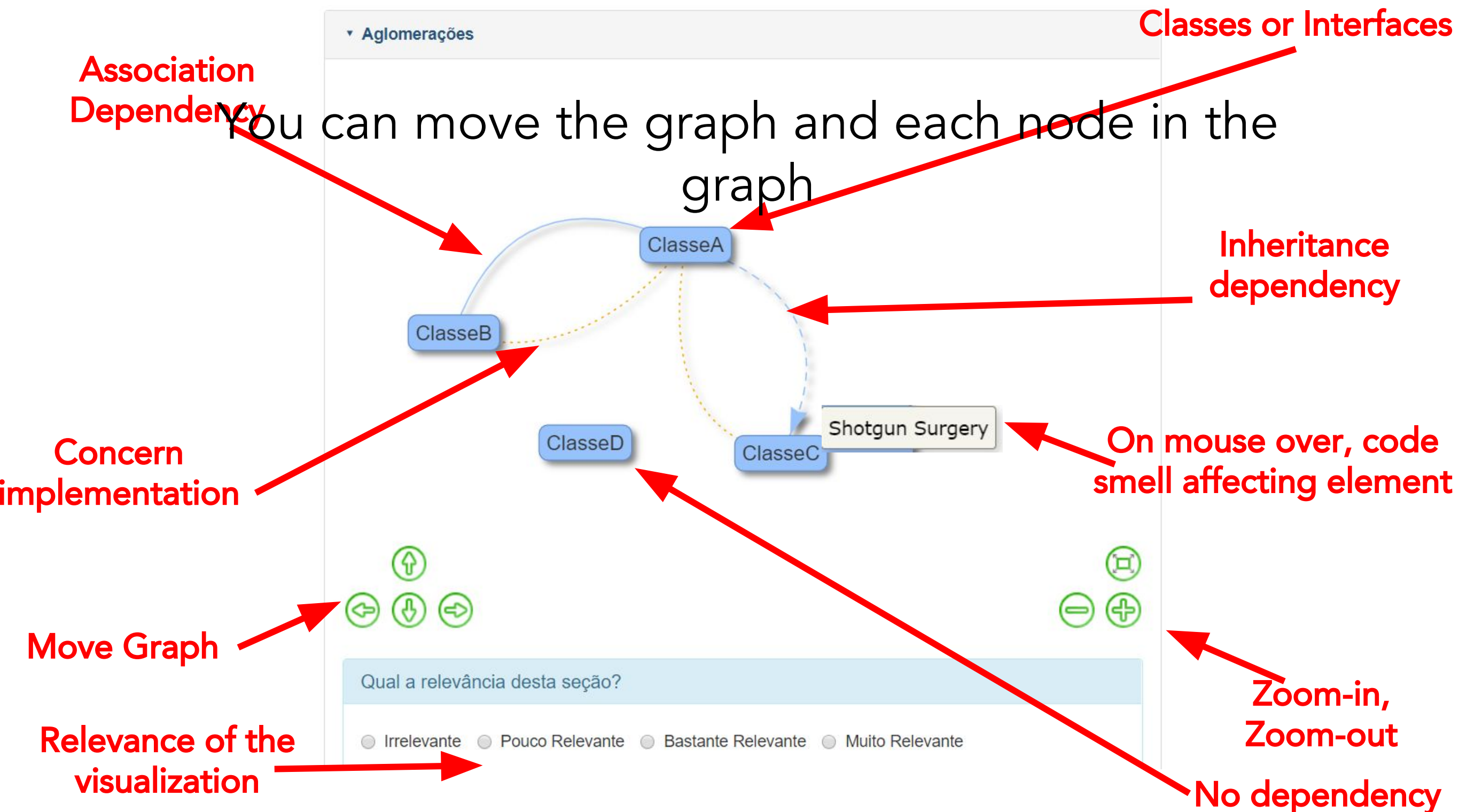## Sumário ExamplesGraph

Ordenar Seções ▾

▸ Aglomerações

▸ Anomalias de Código (Code Smells)

▸ Concerns

Salvar

Concerns

Agglomerations

You can move the graph and each node in the graph

Association Dependency

Classes or Interfaces

Inheritance dependency

Concern implementation

On mouse over, code smell affecting element

Move Graph

Zoom-in, Zoom-out

Relevance of the visualization

No dependency

▾ Aglomerações

ClasseA

ClasseB

ClasseD

ClasseC

Shotgun Surgery

Qual a relevância desta seção?

○ Irrelevante   ○ Pouco Relevante   ○ Bastante Relevante   ○ Muito Relevante

Code smells tab

Anomalias de Código (Code Smells)

|	
Anomalia	God Class
Descrição	Classe que executa muito trabalho por conta própria, delegando apenas pequenos detalhes a um conjunto de classes triviais e usando os dados de outras classes. Em resumo, é uma classe que faz ou que conhece muito do sistema.
Aglomerado?	Sim
Elemento	ClasseB
Razão	-

Description of each agglomerated smell

Anomalia	Feature Envy
Descrição	Um método que está interessado em outra classe diferente da classe a qual ele pertence.
Aglomerado?	Sim
Elemento	ClasseA.next
Razão	-

Anomalia	Shotgun Surgery
Descrição	Alterações em um método que dispara várias alterações no sistema.
Aglomerado?	Sim
Elemento	ClasseC.egent
Razão	-

Anomalia	Refused Bequest
Descrição	Uma classe que ignora a maioria das funcionalidades da super classe.
Aglomerado?	Não
Elemento	ClasseA
Razão	-

Description of no agglomerated smell
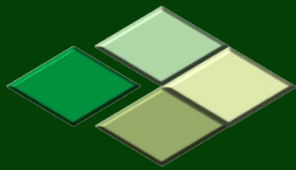
Concerns Tab

▾ **Concerns**

| Princípio	Descrição
Concern Principal	O principal concern (funcionalidade) implementado na aglomeração. **Razão:** Classes que implementam o concern principal: -ClasseA -ClasseB
Concern Secundário	Concern secundário que também é implementado na classe. **Razão:** Concern Secundário 1: [ClasseC] Concern Secundário 2: [ClasseB, ClasseA]

Salvar

{oagbachi, emfernandes, afgarcia}@inf.puc-rio.br

About the
Experiment with Graph-based
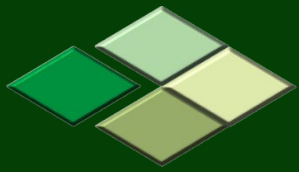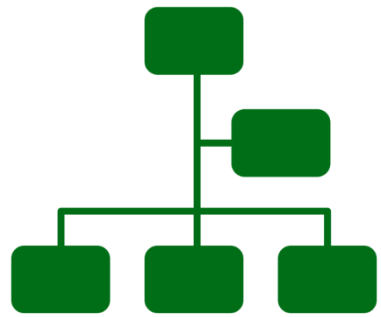
Reason about pre-selected code smell agglomerations

Identify (or not) **design problems** with the agglomerations

{oagbachi, emfernandes, afgarcia}@inf.puc-rio.br
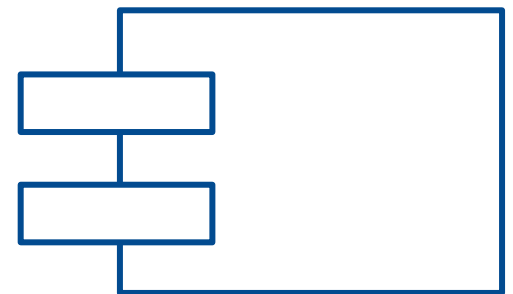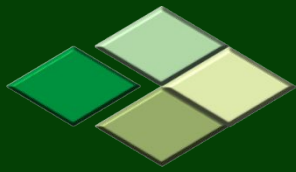
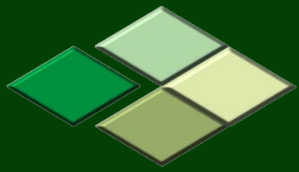Briefly **describe it**
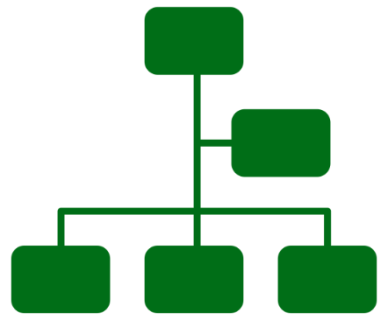
List the **visualization elements** that helped identifying it
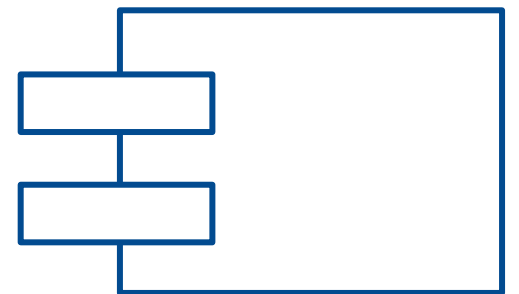
List the **code elements** involved in the design problem

You will have **90 minutes** to finish the experiment

Questions? Please **ask** the experiment instructor

{oagbachi, emfernandes, afgarcia}@inf.puc-rio.br

Identifying Design Problems with Code Smell Agglomerations

Graph-based

Benedicte Agbachi, Eduardo Fernandes, and Alessandro Garcia

OPUS Research Group

LES | DI | PUC-Rio - Brazil