



Ferramenta para Análise do Tratamento Excepcional de Objetos

Aluna: Simone Hanazumi

Orientadora: Ana Cristina Vieira de Melo (MAC - IME)

Apoio financeiro:





Motivação

- Exceções e Mecanismos de Tratamento de Exceções.
 - Linguagem Java: *try*, *catch*, *throw* e *finally*;
- Testes de programas orientados a objetos.
 - Análise dos caminhos percorridos pelas exceções:
 - Definição e Uso das Exceções;
 - Grafo de Fluxo de Controle.



Definição e Uso de Exceções

(S. Sinha e M. J. Harrold, 1999)

- Definição:
 - Atribuição de um valor a uma variável de exceção;
 - *Catch*;
 - *Throw*.
- Uso:
 - Acesso ao valor de uma variável de exceção;
 - *Catch*;
 - *Throw*.
- Par de definição-uso:
 - Definição de variável de exceção e uso com comando *throw*;
 - Definição de exceção em *catch* e uso no próprio bloco;
 - Definição em *throw* e uso em *catch*.



Grafo de Fluxo de Controle

(S. Sinha e M. J. Harrold, 1998)

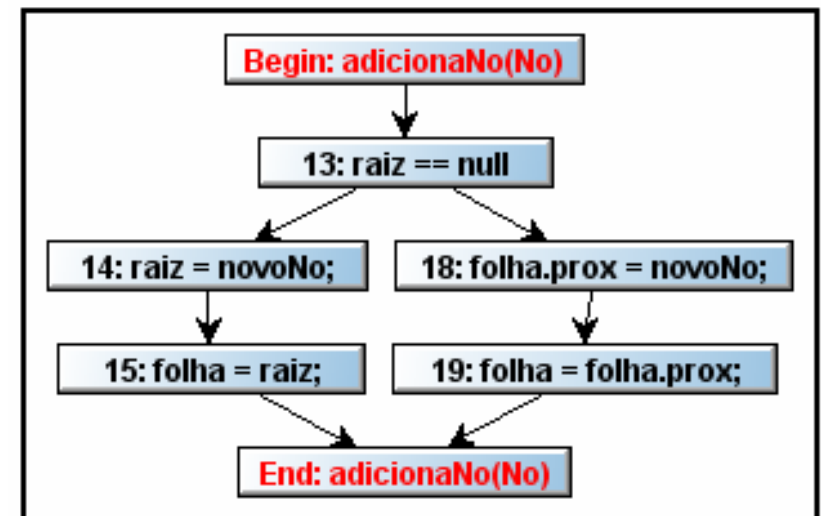
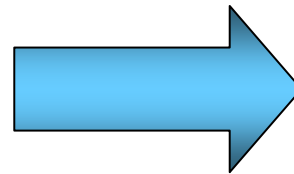
- Estrutura de dados utilizada para representar o fluxo de execução de um programa.
- Cada nó contém uma linha de código do programa. Eles são ligados por arestas, de acordo com a ordem de execução.

Grafo de Fluxo de Controle

(S. Sinha e M. J. Harrold, 1998)

- Exemplo*:

```
1 package exemplo;
2
3 public class SuperNo {
4     No raiz;
5     No folha;
6
7     public SuperNo() {
8         raiz = null;
9         folha = null;
10    }
11
12    public void adicionaNo (No novoNo) {
13        if (raiz == null) {
14            raiz = novoNo;
15            folha = raiz;
16        }
17        else {
18            folha.prox = novoNo;
19            folha = folha.prox;
20        }
21    }
22 }
```





Software desenvolvido: OConGraX

- Extensão da **OConGra** (*P. R. A. F. Nunes e A. C. V. de Melo, 2004*), software que obtém as definições e usos de objetos e gera o grafo de fluxo de controle de objetos;
- Escrito em **Java 6**, utilizando o ambiente de desenvolvimento **Eclipse 3.2.2***;
- Uso dos *frameworks*:
 - *Recoder* (versão 0.81)** ;
 - *JGraph* (versão 5.10.1.0)***.

* **Eclipse 3.2.2**: disponível em <http://www.eclipse.org>

** *Recoder*: disponível em <http://recoder.sourceforge.net>

*** *JGraph*: disponível em <http://www.jgraph.com>



Funcionalidades da Ferramenta

- Obtenção das linhas de definição e uso de objetos e exceções;
- Obtenção dos pares de definição e uso das exceções;
- Geração do grafo de fluxo de controle de objetos com informações sobre os mecanismos de tratamento de exceção utilizados no código recebido como entrada;
- Permite salvar o grafo como arquivo de imagem (jpg, jpeg, gif, bmp, png);
- Permite salvar as linhas de definições e usos de objetos e exceções num arquivo XML.

Exemplo de Uso da OConGraX

- Código Java recebido como entrada*:

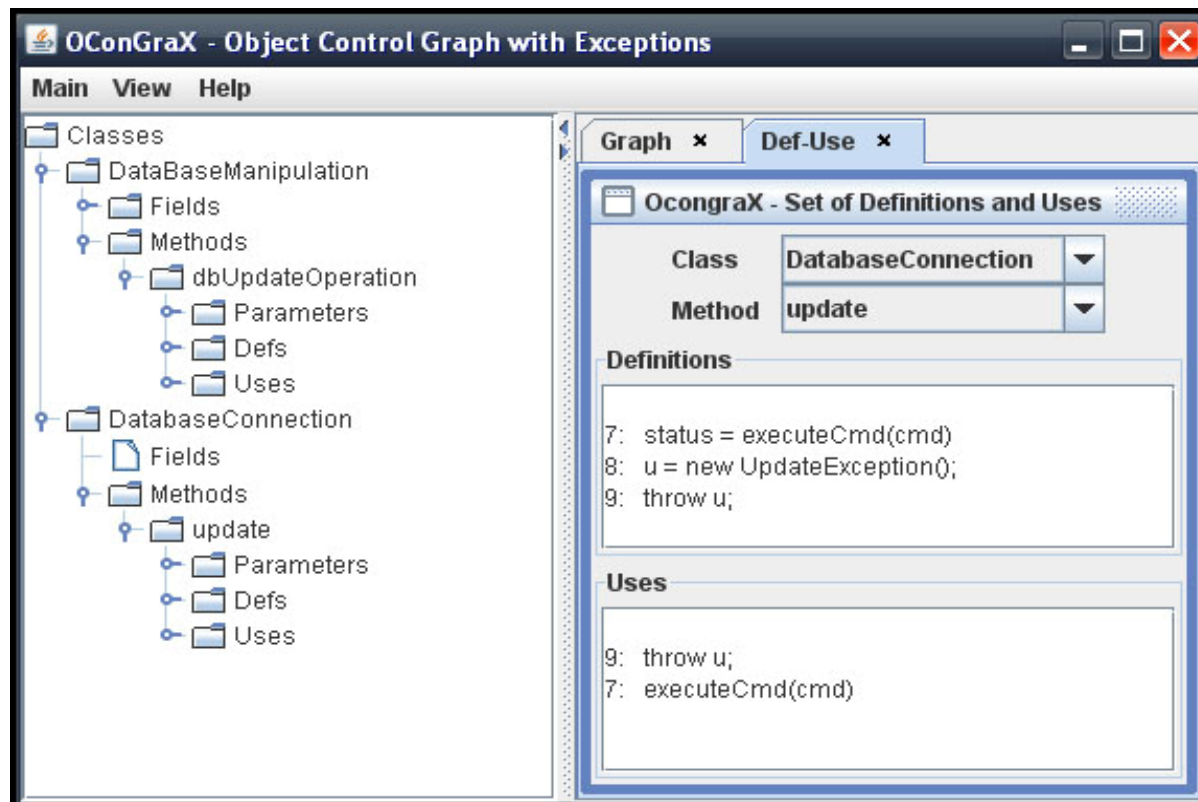
```
1 public class DataBaseManipulation {
2     DatabaseConnection dbConn =
3         new DatabaseConnection();
4     String sqlCmd;
5     Array row;
6     void dbUpdateOperation () {
7         dbConn.open();
8         try {
9             row = dbConn.select(sqlCmd);
10            sqlCmd = updateFields(row);
11            dbConn.update(sqlCmd);
12        }
13        catch (UpdateException ue) {
14            showMessage(ue);
15        }
16        finally {
17            dbConn.close();
18        }
19    }
20 }
```

```
1 public class DatabaseConnection {
2
3     void update(String cmd)
4         throws UpdateException {
5         UpdateException u;
6         int status;
7         if ((status = executeCmd(cmd)) == 0) {
8             u = new UpdateException();
9             throw u;
10        }
11    }
12 }
```

*Adaptado do exemplo do artigo: *Analysis of Programs with Exception-Handling Constructs*, de S. Sinha e M. J. Harrold, 1998.

Exemplo de Uso da OConGraX

- Definição e Uso:



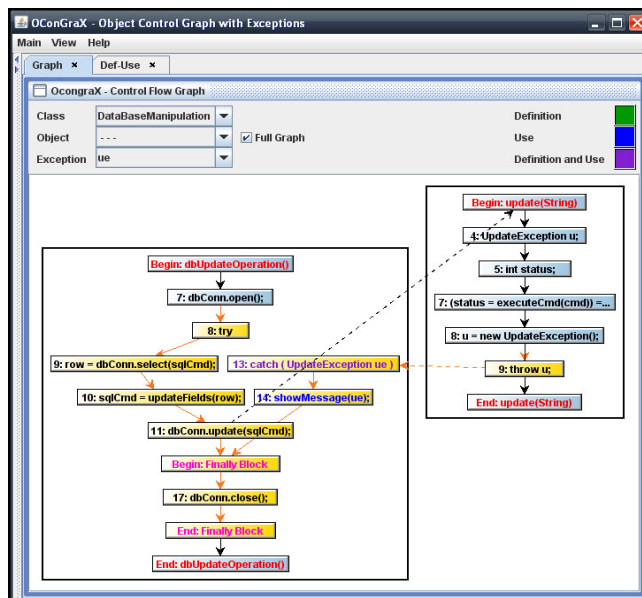
Tela de visualização das definições e usos de objetos e exceções.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<elements>
  <object>
    <classname>...</classname>
    <name>...</name>
    <def>...</def>
    .
    .
    <use>...</use>
    .
    .
  </object>
  .
  .
  <exception>
    <classname>DataBaseManipulation</classname>
    <name>ue</name>
    <pair>
      <def>13</def>
      <use>14</use>
    </pair>
    <def>13</def>
    <use>13</use>
    <use>14</use>
  </exception>
  .
  .
</elements>
```

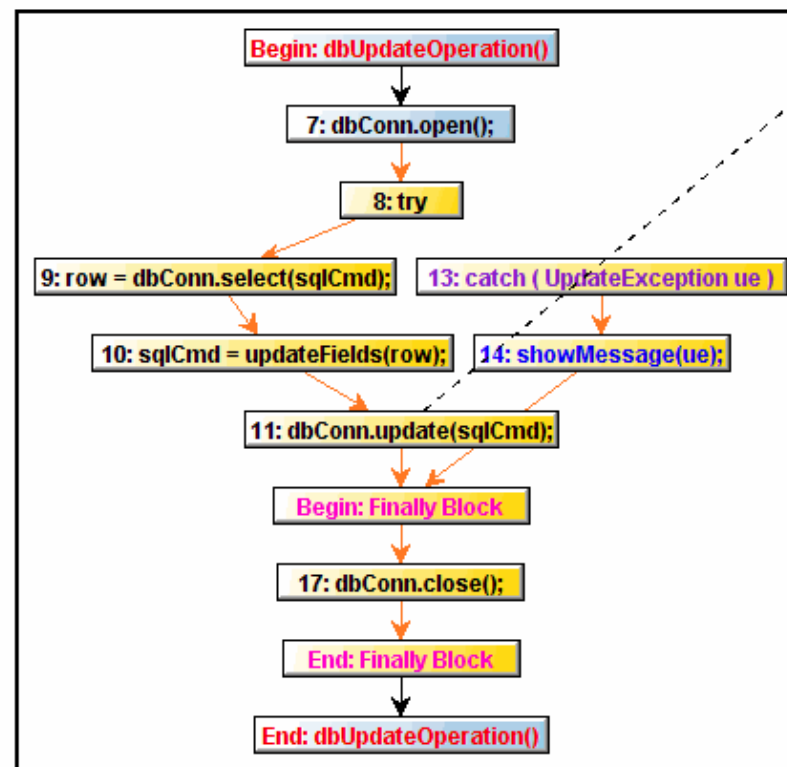
Exemplo parcial do arquivo XML gerado pela OConGraX.

Exemplo de Uso da OConGraX

- Grafo gerado:



Tela de visualização do grafo.



Arquivo GIF do grafo gerado pela OConGraX.



Conclusão

- O uso da ferramenta e dos dados gerados possibilitam:
 - Facilidade na geração de testes;
 - Testes mais completos e eficazes;
 - Análise do programa;
- Conseqüências:
 - Diminuição dos custos no desenvolvimento de testes e manutenção do software;
 - Programas menos propensos a erros e mais confiáveis.