

Detectando Códigos Clonados

Leopoldo Teixeira, Márcio Ribeiro e Paulo Borba

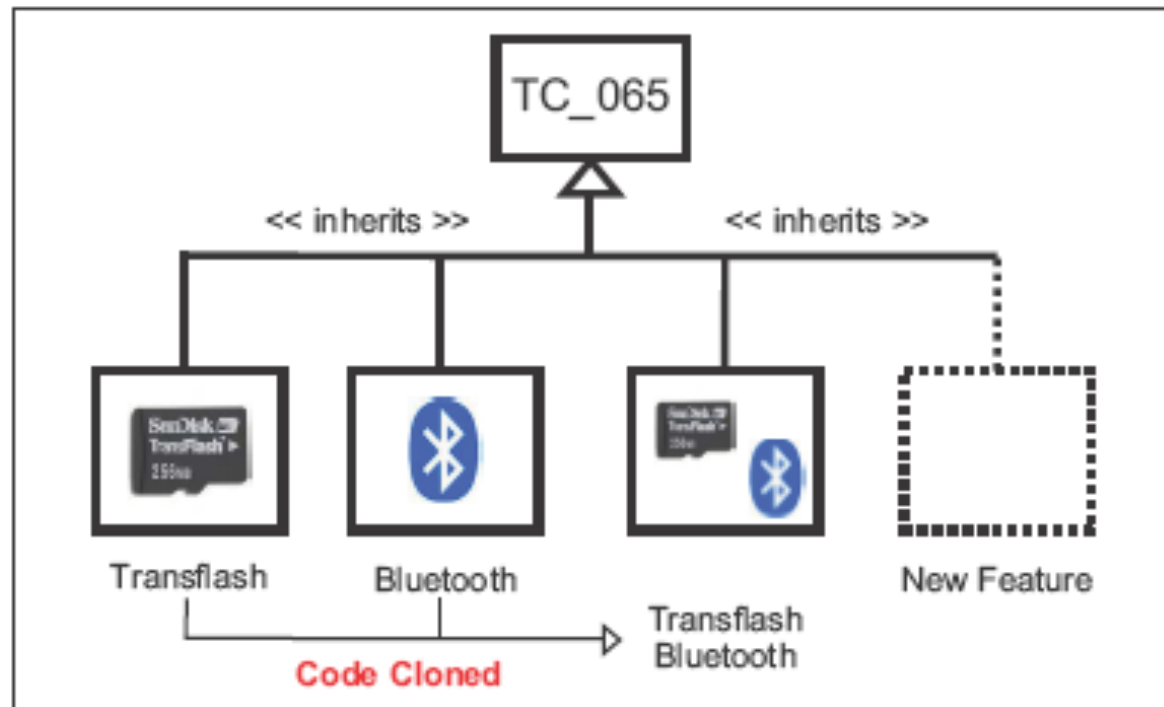
{lmt, mmr3, phmb}@cin.ufpe.br

Clones de código

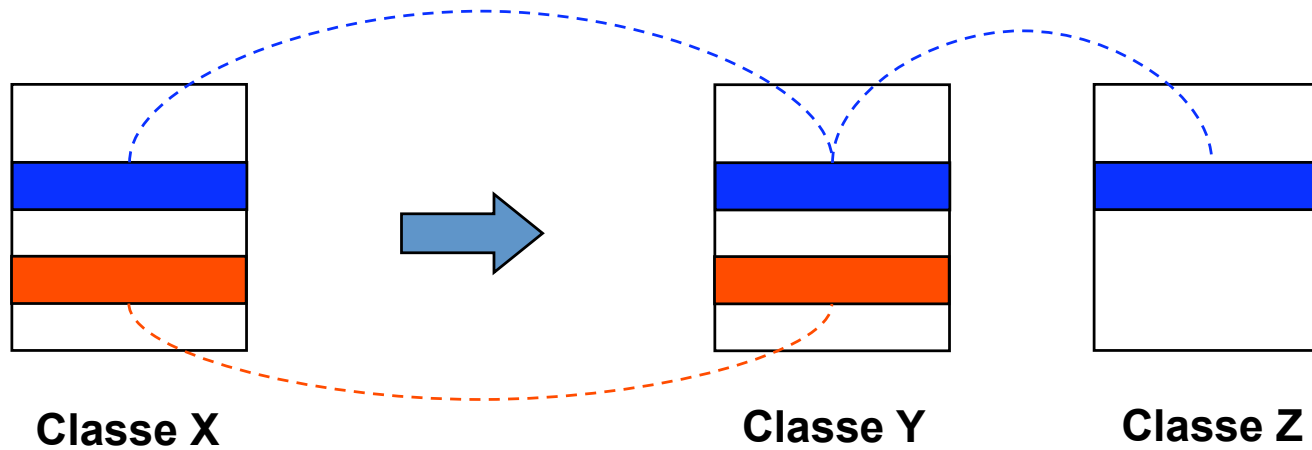
- Também conhecidos como códigos...



- Repetidos
- Replicados
- Duplicados
- Clonados



Por que detectar e remover clones?



Consome tempo!



Susceptível a erros!



Reduz a produtividade

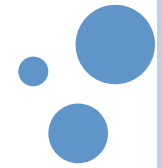


Motivação: identificação de *concerns*

```
public void updateHealthUnit(HealthUnit unit) throws RepositoryException, TransactionException {
    try {
        getPm().beginTransaction();
        healthUnitRecord.update(unit);
        getPm().commitTransaction();
    } catch (RepositoryException e) {
        getPm().rollbackTransaction();
        throw e;
    } catch (TransactionException e) {
        getPm().rollbackTransaction();
        throw e;
    } catch (Exception e) {
        getPm().rollbackTransaction();
    }
}
```

```
public void updateComplaint(Complaint complaint) throws RepositoryException, TransactionException {
    try {
        getPm().beginTransaction();
        complaintRecord.update(complaint);
        getPm().commitTransaction();
    } catch (RepositoryException e) {
        getPm().rollbackTransaction();
        throw e;
    } catch (TransactionException e) {
        getPm().rollbackTransaction();
        throw e;
    } catch (Exception e) {
        getPm().rollbackTransaction();
    }
}
```

Códigos similares!!!

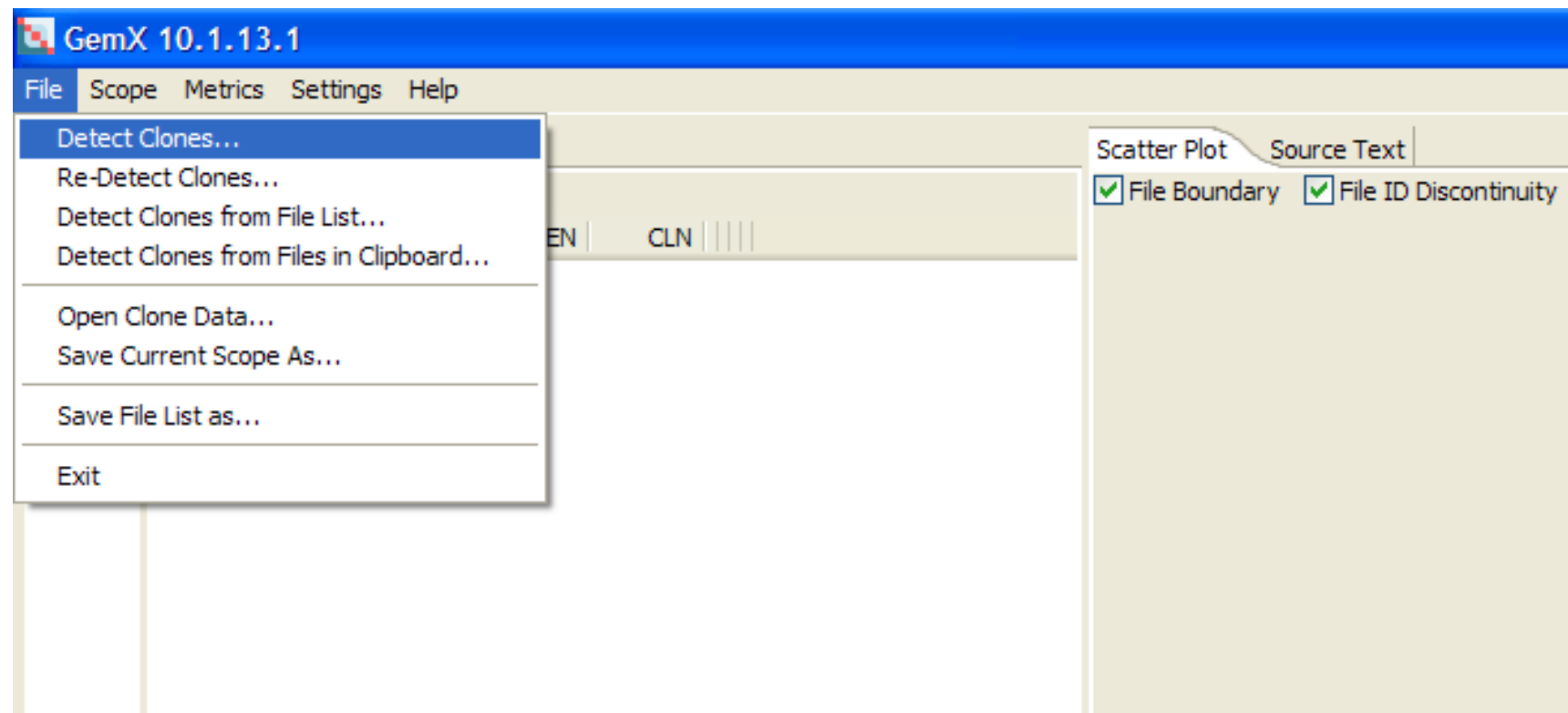


Como detectar clones?

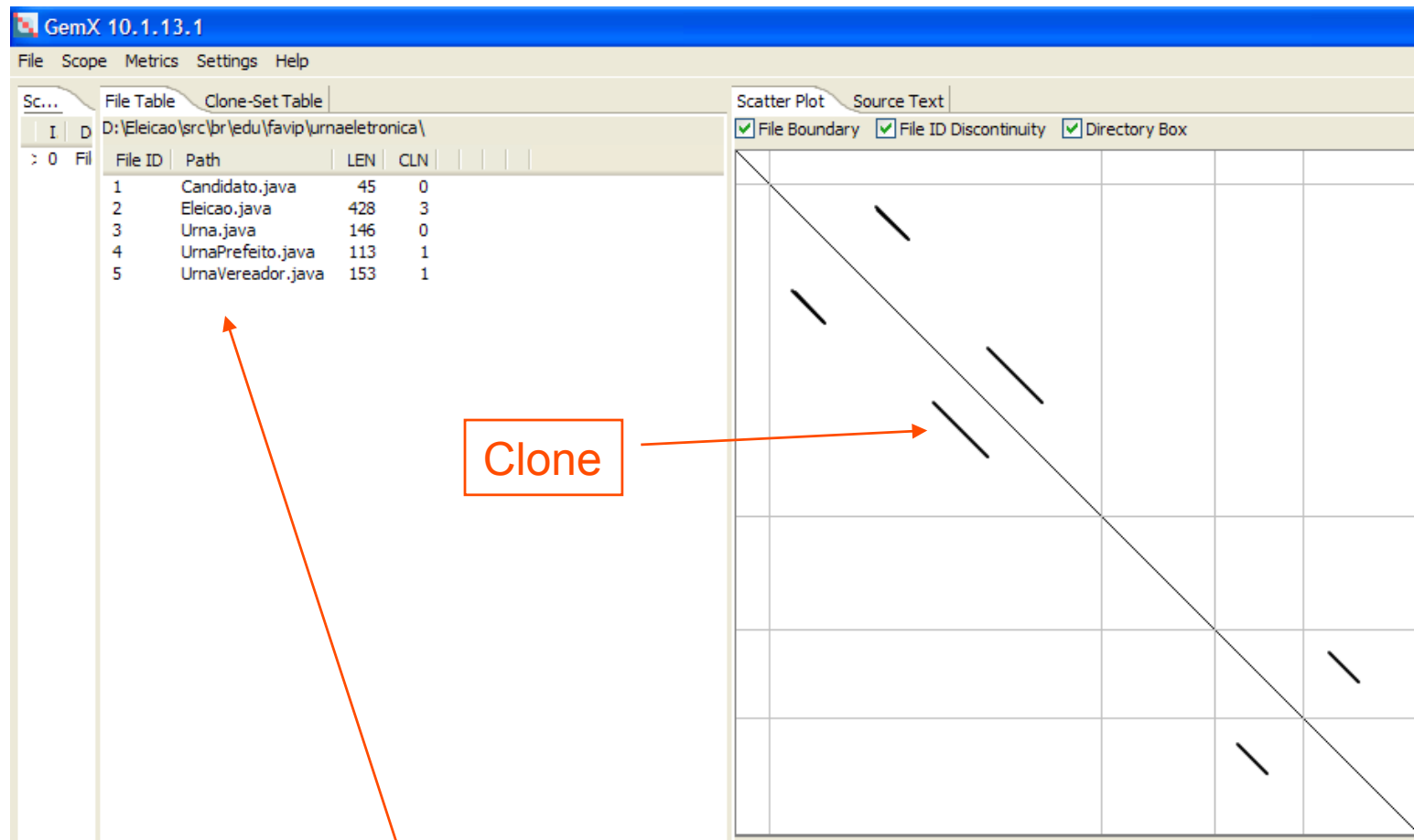
- Ferramenta CCFinder - “*Code Clone Finder*”
- Detecta clones nas seguintes linguagens
 - Java
 - C/C++
 - COBOL
 - VB
 - C#
- É baseada em *tokens*



Usando a ferramenta...



Usando a ferramenta...



Clone

Classes (arquivos)

Matriz de clones

Usando a ferramenta...

File ID	Path	LEN	CLN
1	Candidato.java	45	0
2	Eleicao.java	428	3
3	Urna.java	146	0
4	UrnaPrefeito.java	113	1
5	UrnaVereador.java	153	1

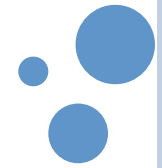
```
4 D: \Eleicao\src\br\edu\favip\urnaeletronica\UrnaPrefeito.java
1 package br.edu.favip.urnaeletronica;
2
3 public class UrnaPrefeito extends Urna {
4
5 public UrnaPrefeito(Candidato[] candidatos) {
6     super(candidatos);
7 }
8
9 public Candidato getVencedor() {
10
11     int maiorVoto = 0;
12     Candidato[] candidatos = getCandidatos();
13     Candidato vencedor = null;
14
15     for (int i = 0; i < candidatos.length; i++) {
16         Candidato candidato = candidatos[i];
17
18         if (candidato.getVotos() > maiorVoto) {
19             vencedor = candidato;
20             maiorVoto = candidato.getVotos();
21         }
22     }
23
24     return vencedor;
25 }
26
27 }
5 D: \Eleicao\src\br\edu\favip\urnaeletronica\UrnaVereador.java
1 package br.edu.favip.urnaeletronica;
2
3 public class UrnaVereador extends Urna {
4
5 public UrnaVereador(Candidato[] candidatos) {
6     super(candidatos);
7 }
8
9 public Candidato getVencedor() {
10
11     double calculoVoto = 0;
12     double maiorVoto = 0;
13     Candidato[] candidatos = getCandidatos();
14     Candidato vencedor = null;
15
16     for (int i = 0; i < candidatos.length; i++) {
17         Candidato candidato = candidatos[i];
18
19         if (candidato.getPartido() == Candidato.PARTII)
20             calculoVoto = candidato.getVotos() * 0.5;
21         else {
22             calculoVoto = candidato.getVotos();
23         }
24
25         System.out.println(calculoVoto);
26
27         if (calculoVoto > maiorVoto) {
28             vencedor = candidato;
29             maiorVoto = calculoVoto;
30         }
31     }
32
33     return vencedor;
34 }
35
36 }
```

Aba de Código

Duas classes com código clonado (blocos cinza)

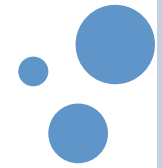
Relatório

```
C:\WINDOWS\system32\cmd.exe
807-937-1.tmp
exec: D:\Programas\CCFinder\bin\ccfx.exe F -e visualbasic -o gemxtemp-20080328-181807-937-1.tmp
81807-937-1.tmp
exec: D:\Programas\CCFinder\bin\ccfx.exe F java -o gemxtemp-20080328-181807-937-1.tmp D:\Eleicao\src
exec: D:\Programas\CCFinder\bin\ccfx.exe -i gemxtemp-20080328-181807-937-1.tmp -c i:char -b 30 -t 12 -s 2 -u + -m 60M -v
> preprocessing
0%:::20%:::40%:::60%:::80%:::100%
> count of files: 5
> count of preprocessed files: 0
> lap cpu time: 0.03125 seconds
> output file name: a.ccfxd
> counting chunks
> making hash table
> locating identical substrings
> count of detected clone pairs: 4
> lap cpu time: 0.0625 seconds
> sorting
> applying soft shaper
> count of clone pairs removed by soft shaper: 0
> done.
> peak used memory: 242.78 mibi bytes
> used cpu time: 0.5625 seconds
```



Projetos

- Instalar o CCFinder
- Utilizá-lo nos seus respectivos projetos
 - Gráfico de clones de código, gerado pela ferramenta
 - Exemplos de clones encontrados
 - Parâmetros usados para configuração da ferramenta
 - Quantidade de clones encontrados
 - Lista de *concerns* associados a estes clones
 - Quantidade de clones por *concern*
- Preparar a apresentação, segundo modelo no site da disciplina



Demonstração

- Projeto de exemplo
- Calibrar parâmetros
- Para mais detalhes em como usar o CCFinder, acessar <http://www.ccfinder.net/doc/10.2/en/tutorial-gemx.html> (Último acesso em: 04/09/2008)