



# Uma Ferramenta para Teste Estrutural de Regras Ativas

---

Ms. Virgínia Mara Cardoso

Dr. Plínio de Sá Leitão Júnior

Prof. Dr. Mario Jino

**DCA – FEEC – UNICAMP**

Abril/2008



# Teste de Banco de Dados

---

- Geração automática de dados de teste.
- Teste de consultas.
- Teste de esquema da base de dados.
- Teste de programa de aplicação de banco de dados:
  - programas de aplicação;
  - regras ativas (banco de dados ativo).

Proposta: construção de uma ferramenta –  
ART-TOOL (*Active Rule Testing Tool*), para o  
suporte ao teste de regras ativas.

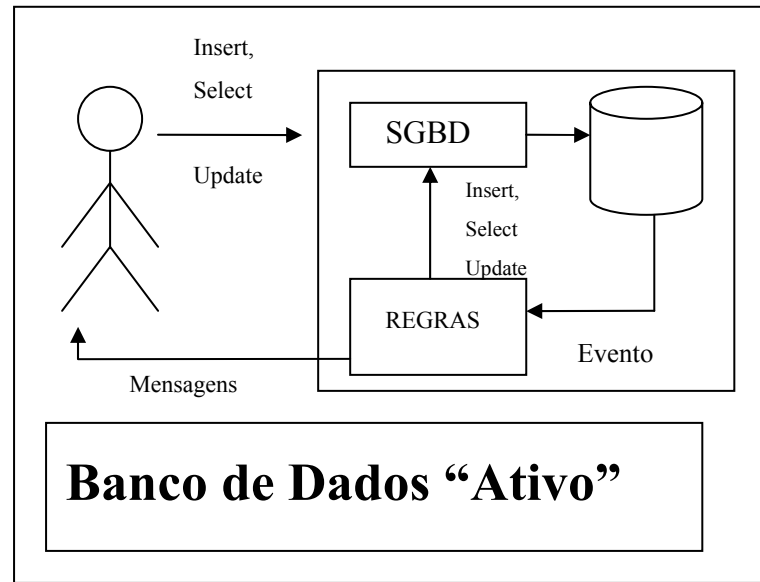
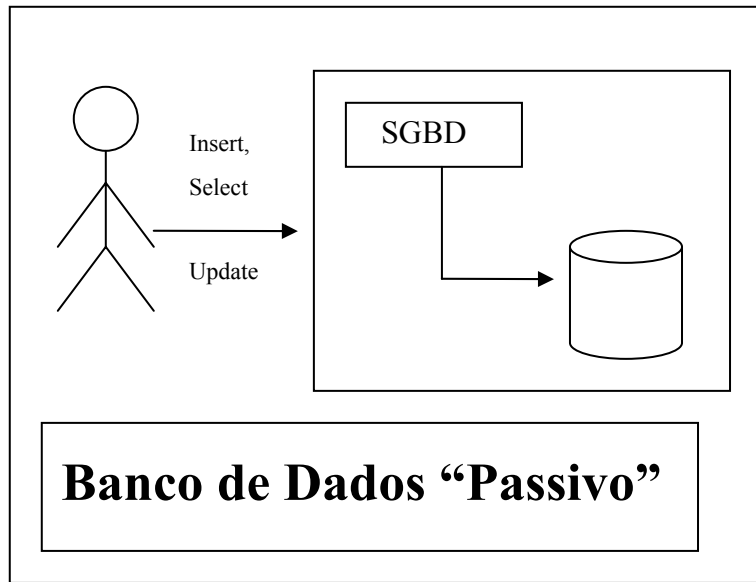


# Tópicos

---

- Banco de Dados Ativo - Regras Ativas.
- Teste Estrutural.
- Modelo de Fluxo de Controle.
- Modelo de Fluxo de Dados.
- Modelo de Instrumentação.
- A Ferramenta ART-TOOL.
- Aplicação da Ferramenta.
- Conclusões.
- Contribuições.
- Trabalhos Futuros.

# Banco de Dados Ativo - Regras Ativas



“Sistema de banco de dados ativo é uma extensão de um modelo de banco de dados pré-existente (relacional, relacional estendido, funcional ou orientado a objetos) com variantes de um paradigma comum de computação, as regras” (Campin *et al.*, 1994).



# Regras Ativas

---

Implementação de várias aplicações:

- internas

{ gerenciamento de restrição de integridade;  
manutenção de dados derivados;

- externas

{ gerenciamento de fluxo de trabalho;  
regras de negócio.

# Regras Ativas – SQL-3

R1 - registra que o salário de um empregado foi modificado em mais de 10%.

```
CREATE TRIGGER tgEmpl_UpdateSalary
```

```
  BEFORE UPDATE OF salary
```

```
  ON employee
```

```
  FOR EACH ROW
```

```
  WHEN (:new.salary > 200)
```

```
BEGIN
```

```
  IF ABS(:new.salary - :old.salary) > (:old.salary * 0.1) THEN
```

```
    INSERT INTO log
```

```
      (timetamp, description, tableid, tablekey, alert)
```

```
      VALUES (SYSDATE, 'update salary', 'employee', 'new value > 10%');
```

```
  END IF;
```

```
END;
```

**EVENTO**

**CONDIÇÃO**

**AÇÃO**



# Teste Estrutural

---

- O teste deve ser planejado e projetado para definir formas de execução do programa com a maior probabilidade de revelar defeitos no software.
- Focaliza a estrutura de controle e de dados do programa.
- Casos de teste são derivados da implementação e visam o exercício de elementos do programa.
- Critérios de teste estrutural:
  - exercício de caminhos, nós, arcos;
  - potenciais usos: requerem o exercício de associações entre a definição de uma variável e seu potencial uso.



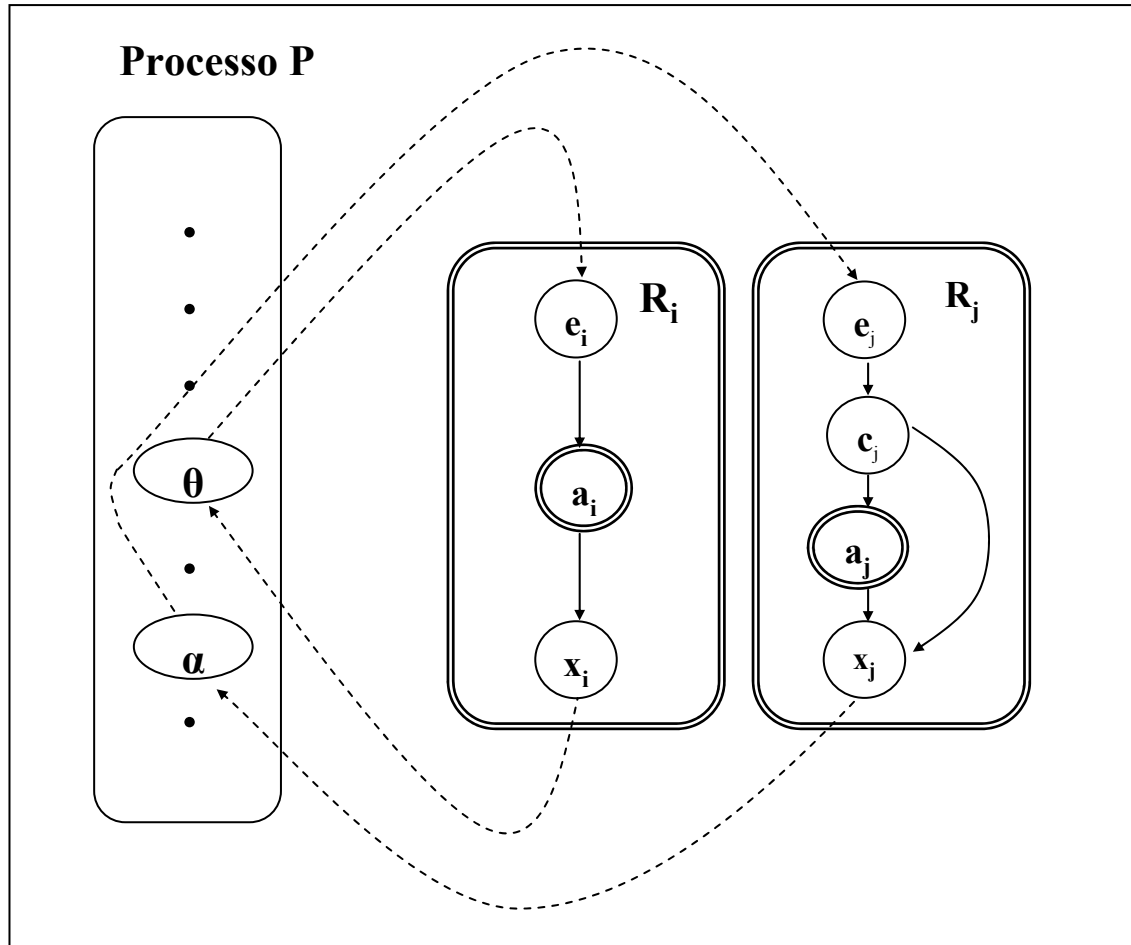
# Técnicas de Teste Estrutural

---

Modelos necessários para a aplicação dos critérios selecionados:

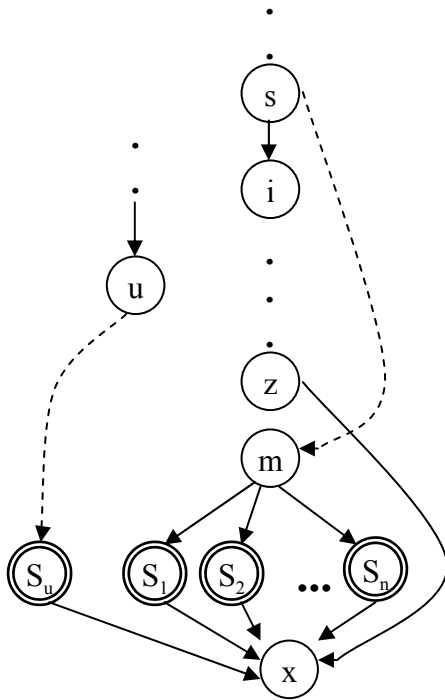
- modelo de fluxo de controle;
- modelo de fluxo de dados;
- modelo de instrumentação:
  - identificação de tuplas.

# Modelo de Fluxo de Controle para Regras Ativas



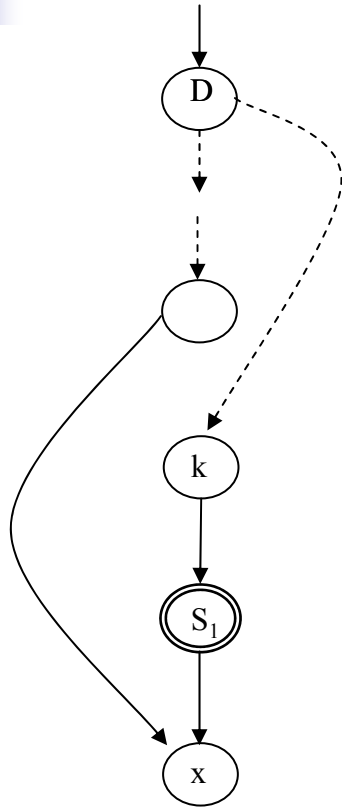
# Modelo de Fluxo de Controle para Regras Ativas

Estrutura de controle para o tratamento de exceções.



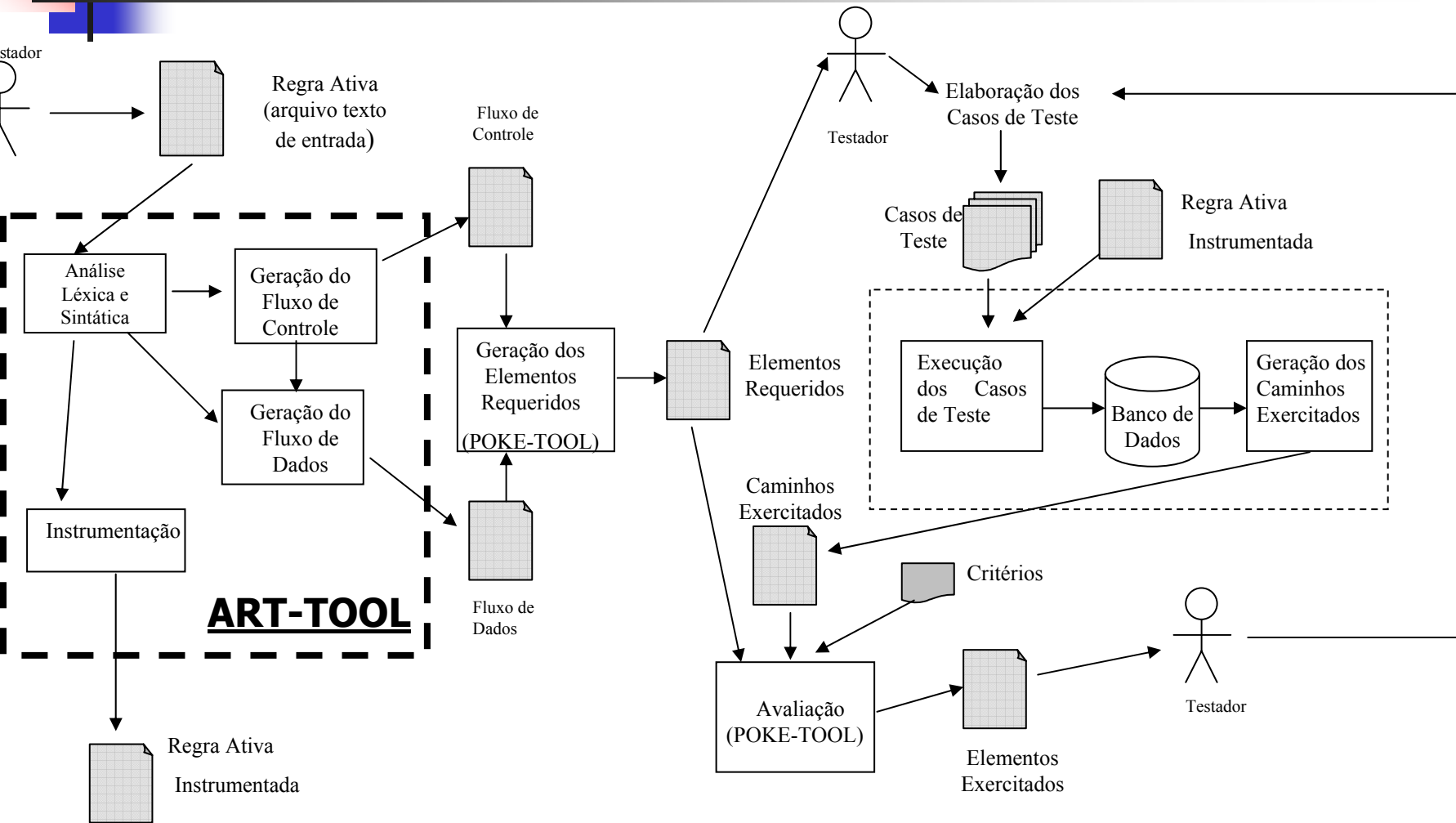
```
DECLARE USER_EXC EXCEPTION;
.....
BEGIN
  .....
  .....
  EXCEPTION
    WHEN TOO_MANY_ROWS THEN
    .....
    .....
    WHEN USER_EXC THEN
    .....
    .....
    WHEN OTHERS THEN
    .....
    .....
END;
```

# Modelo de Instrumentação



```
.....  
Ponta_de_prova(D);  
DELETE FROM tempempl;  
.....  
Ponta_de_prova(x);  
  
EXCEPTION  
  
WHEN OTHERS THEN  
    Ponta_de_prova(k);  
    Ponta_de_prova(x);  
  
END;
```

# ART-TOOL (Active Rule Testing Tool)

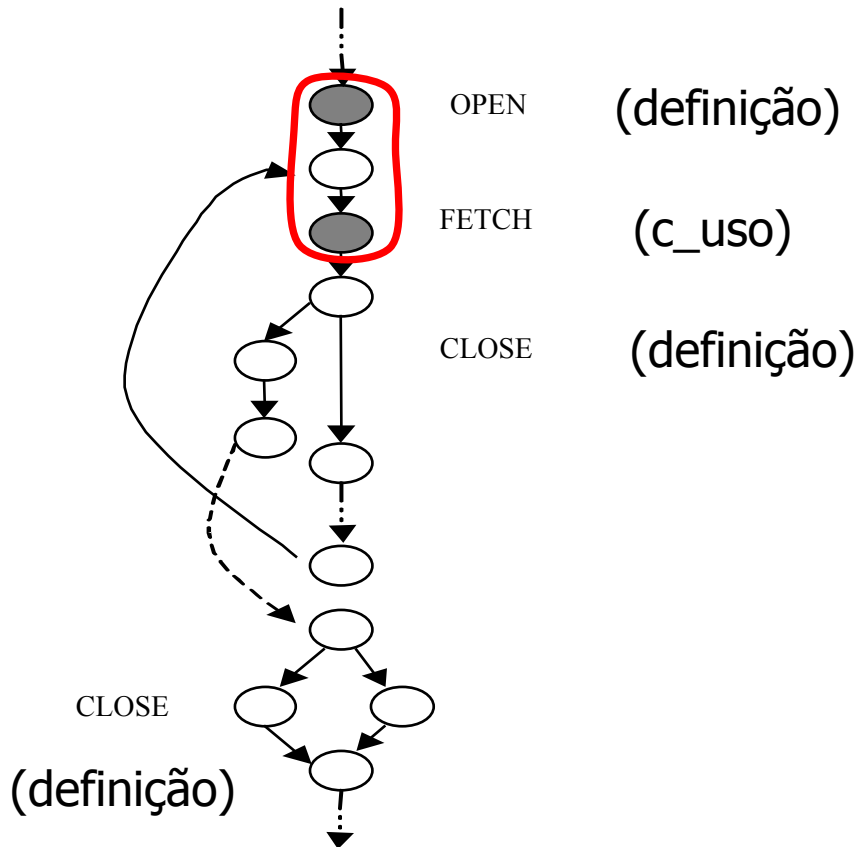




# Teste – Comandos de Cursores

Associações requeridas na aplicação de critérios Potenciais Usos

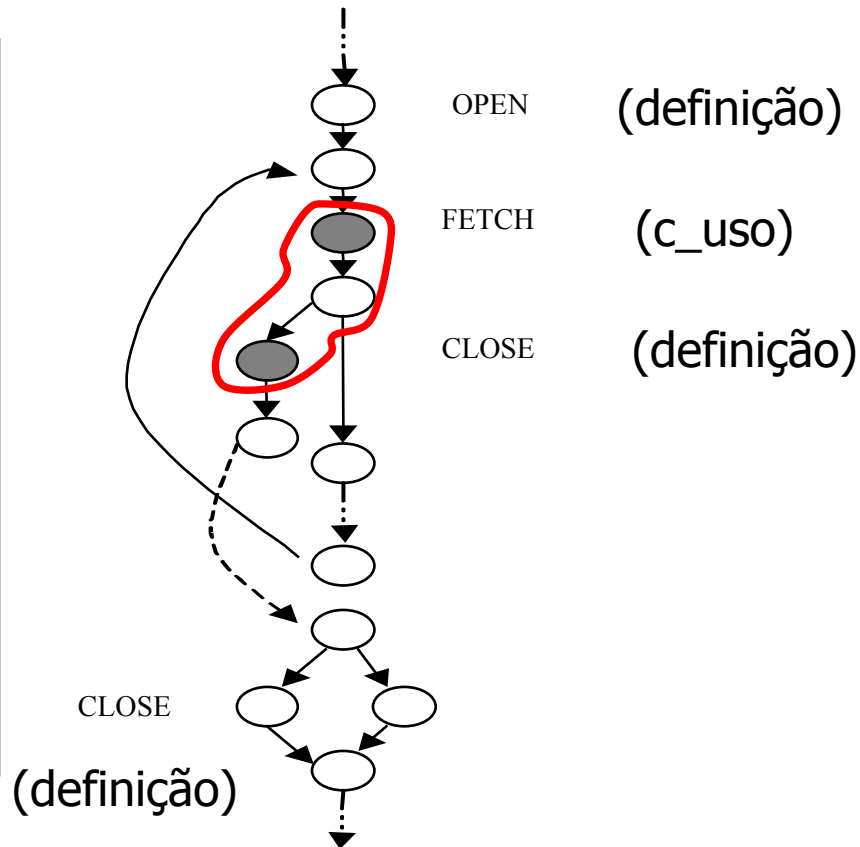
```
.....  
Open cr_control;  
.....  
Fetch cr_control  
  into v_nome, v_conta;  
.....  
Close cr_control;  
.....  
.....  
Close cr_control;  
.....
```



# Teste – Comandos de Cursores

Associações requeridas na aplicação de critérios Potenciais Usos

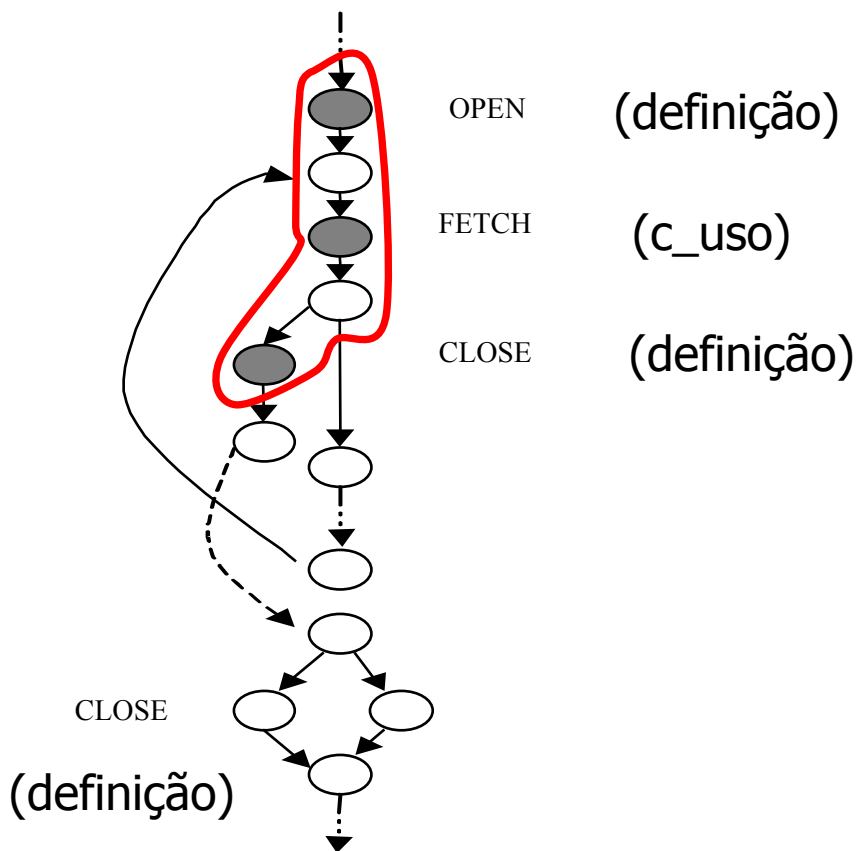
```
.....  
Open cr_control;  
.....  
Fetch cr_control  
  into v_nome, v_conta;  
.....  
Close cr_control;  
.....  
.....  
Close cr_control;  
.....
```



# Teste – Comandos de Cursores

Associações requeridas na aplicação de critérios Potenciais Usos

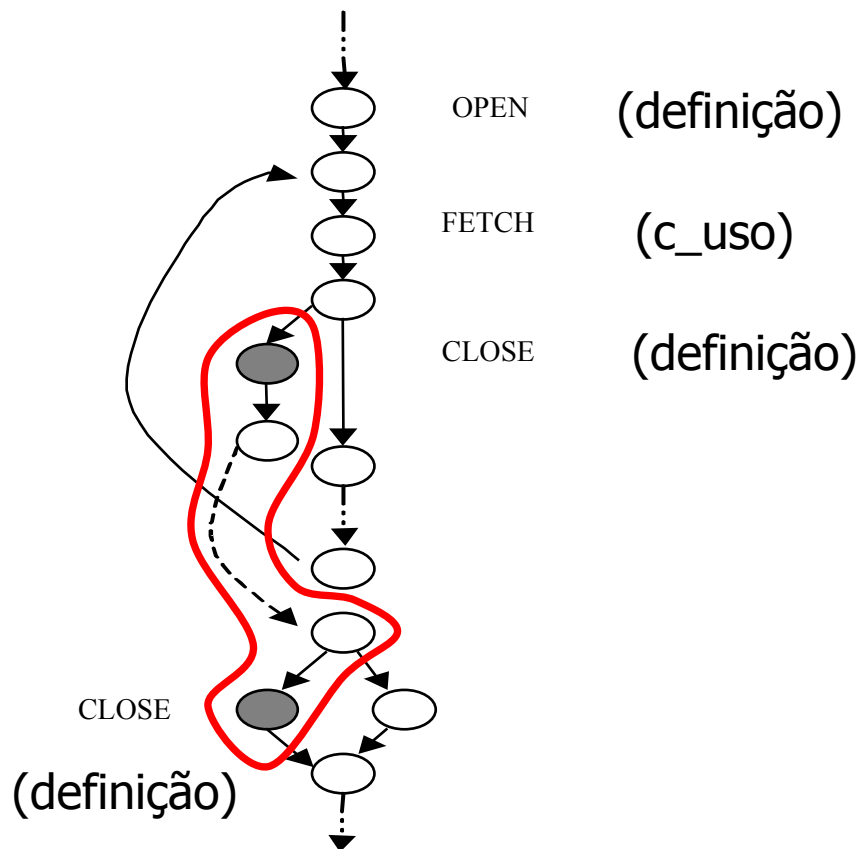
```
.....  
Open cr_control;  
.....  
Fetch cr_control  
    into v_nome, v_conta;  
.....  
Close cr_control;  
.....  
.....  
Close cr_control;  
.....
```



# Teste – Comandos de Cursores

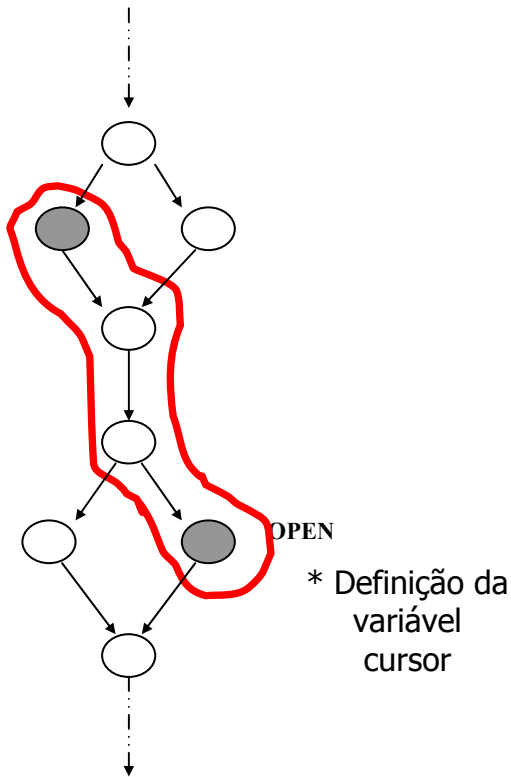
Associações requeridas na aplicação de critérios Potenciais Usos

```
.....  
Open cr_control;  
.....  
Fetch cr_control  
  into v_nome, v_conta;  
.....  
Close cr_control;  
.....  
.....  
Close cr_control;  
.....
```



# Teste – Comandos de Cursores

**OPEN**  
\* Definição da  
variável  
cursor



OBS: com a aplicação dos critérios Potenciais Usos foi possível detectar esse defeito.

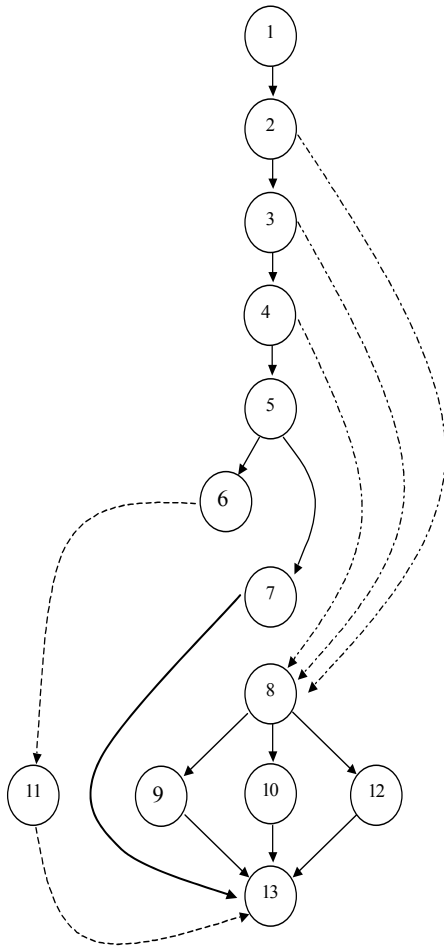


# Aplicação da Ferramenta

---

- 15 regras testadas:
  - 5 regras, exemplo real;
  - 10 regras, exemplos elaborados.
- Caso de teste:
  - Operação do evento (Insert/Delete/Update);
  - Estado da base de dados (antes do disparo da regra);
  - Estado final da base esperado.

# Análise de Cobertura



Casos de Teste-R1	Critérios							
	Todos Nós	Todos Arcos	Todos Usos	Todos P-Usos	Todos Pot-Usos	Pot-Du Caminhos	Pot – Usos /Du	Du Caminhos
1	38 %	25 %	11 %	10 %	14 %	8 %	14 %	12 %
2	38 %	25 %	11 %	10 %	14 %	8 %	14 %	12 %
1 e 2	46 %	38 %	15 %	14 %	24 %	15 %	24 %	16 %
3	54 %	13 %	41 %	29%	14 %	10 %	14 %	36 %
1,2 e 3	77 %	50 %	52 %	43 %	38 %	25 %	38 %	48 %
4	62 %	13 %	44 %	29 %	20 %	15 %	20 %	40 %
1,2,3 e 4	92 %	63 %	59 %	48 %	52 %	40 %	52 %	56 %



# Conclusões

---

- Os modelos propostos e implementados mostraram-se úteis à automação do processo de teste.
- A aplicação dos modelos à abstração de elementos requeridos pelos critérios utilizados foi capaz de revelar defeitos típicos em comandos de manipulação de dados, inclusive relacionados ao mecanismo de cursor.
- Ficou evidente a necessidade de um bloco de tratamento de exceção quando há comandos de manipulação no código da regra.
- Os resultados obtidos pelo uso da ferramenta ART-TOOL indicam que os critérios baseados em fluxo de dados, especificamente os critérios Potenciais Usos, representam um valioso recurso no teste sistemático de regras ativas.



# Contribuições

---

- Esforço inicial para o teste de regras ativas escritas em SQL.
- Elaboração de modelos úteis à automação do teste estrutural de regras ativas.
- Exemplos de aplicação de critérios estruturais baseados em fluxo de dados a regras ativas.
- Modelo de instrumentação para a identificação de *tuplas*.
- Análise do mecanismo de cursor.



# Trabalhos Futuros

---

- Aplicação da ferramenta em novos conjuntos de regras.
- Extensão dos modelos de implementação ao tratamento de exceção em blocos aninhados.
- Implementação do modelo de instrumentação para identificação de *tuplas* e elaboração de critérios específicos.
- Automação do teste de regras ativas para outros sistemas gerenciadores de banco de dados.
- A ferramenta ART-TOOL como recurso para apoiar outras atividades de teste.