

Orientando: Marilan Ricardo Tagliari

Orientador: Everaldo Artur Grahl



**UNIVERSIDADE**  
REGIONAL DE BLUMENAU

***FERRAMENTA DE APOIO AO  
MAPEAMENTO DE ESPECIFICAÇÃO  
ESTRUTURADA PARA ESPECIFICAÇÃO  
ORIENTADA A OBJETOS***

# ***Roteiro***

- **Introdução**
- **Objetivos**
- **Especificação Estruturada**
- **Especificação Orientada a Objetos**
- **Estratégia de Mapeamento**
  - Diagrama de Caso de Uso
  - Diagrama de Classes
  - Diagrama de Seqüência
- **Desenvolvimento do Protótipo**
  - Especificação
  - Implementação
- **Resultados**
- **Conclusões**
- **Extensões**
- **Protótipo**
- **Questionamentos**

# *Introdução*

- Softwares mais complexos, reduzir tempo de desenvolvimento, fácil manutenção e custos mais baixos
- Novas técnicas de desenvolvimento (Orientação a Objetos)
- Problemas em migrar sistemas antigos desenvolvidos de forma estruturada em novos sistemas orientados a objetos
- Uma solução seria adotar um processo de mapeamento
- George (1996) elaborou estratégia de mapeamento

# Objetivos

O objetivo principal deste trabalho é construção de uma ferramenta para apoiar o engenheiro de software no processo de mapeamento de especificação estruturada para especificação orientada a objetos.

Objetivos específicos:

- Ferramenta CASE – Diagramas estruturados e UML;
- Importar diagrama da ferramenta CASE *Power Designer*

# ***Especificação Estruturada***

Alguns componentes da especificação estruturada:

- Diagrama de Entidade Relacionamento
- Diagrama de Contexto
- Diagrama de Fluxo de Dados
- Dicionário de Dados

# Especificação Orientada a Objetos

Utilização da UML (*Unified Modeling Language*) para representar as especificações Orientadas a Objetos.

Alguns diagramas da UML:

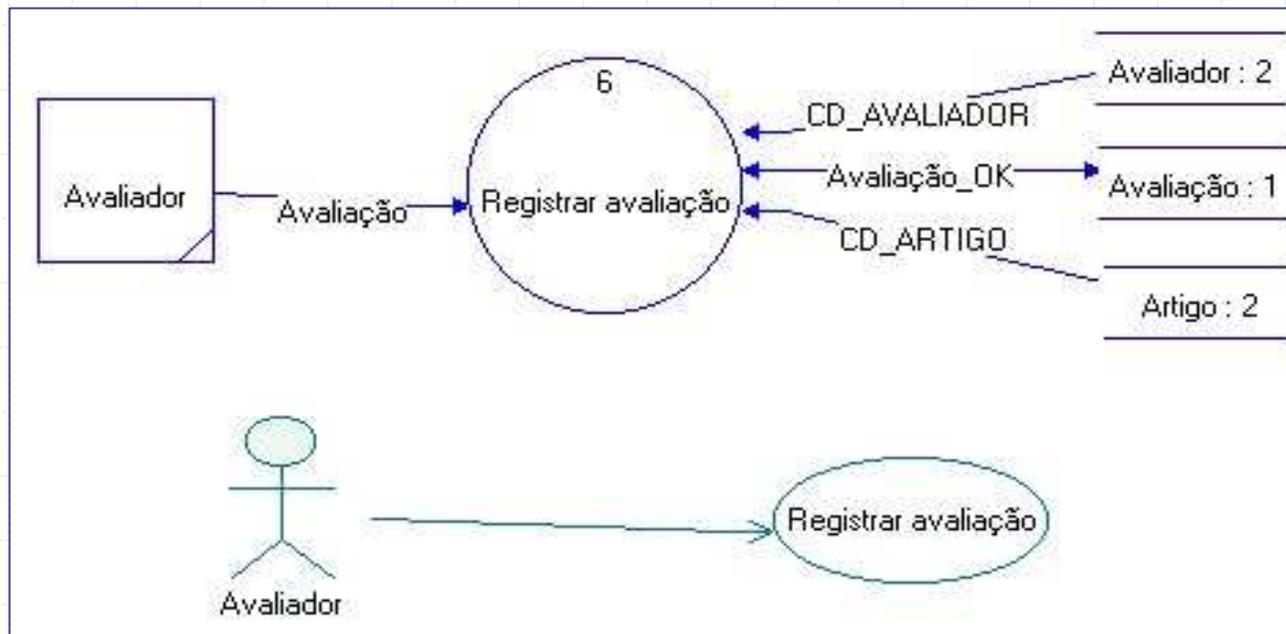
- Diagrama de Caso de Uso
- Diagrama de Classes
- Diagrama de Seqüência

# *Mapeamento*

- Estratégia de George (1996)
- Trabalho de Zibell (1996)
- Trabalho de Saldanha (1998)
- Melhoramento da estratégia de George (1996)
- Adição de dois novos diagramas na estratégia
  - Diagrama de Caso de Uso
  - Diagrama de Seqüência

# Mapear p/ Diag. Caso de Uso

Exemplo de mapeamento do diagrama de Caso de Uso

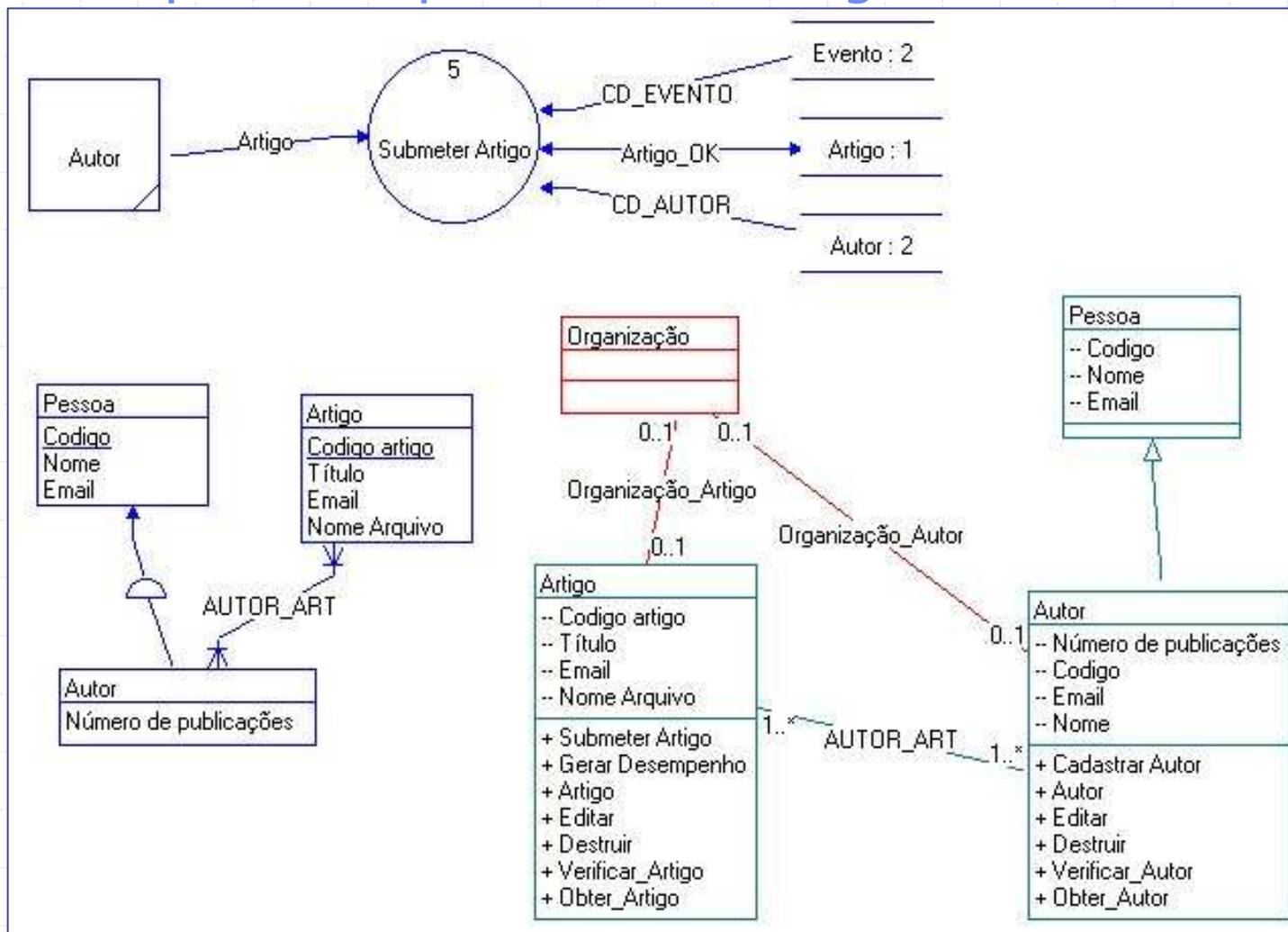


Exemplo da descrição do Caso de Uso

- Receber dados de "Avaliador"
- Verificar Avaliação
- Verificar Artigo
- Verificar Avaliador
- Cadastrar Avaliação

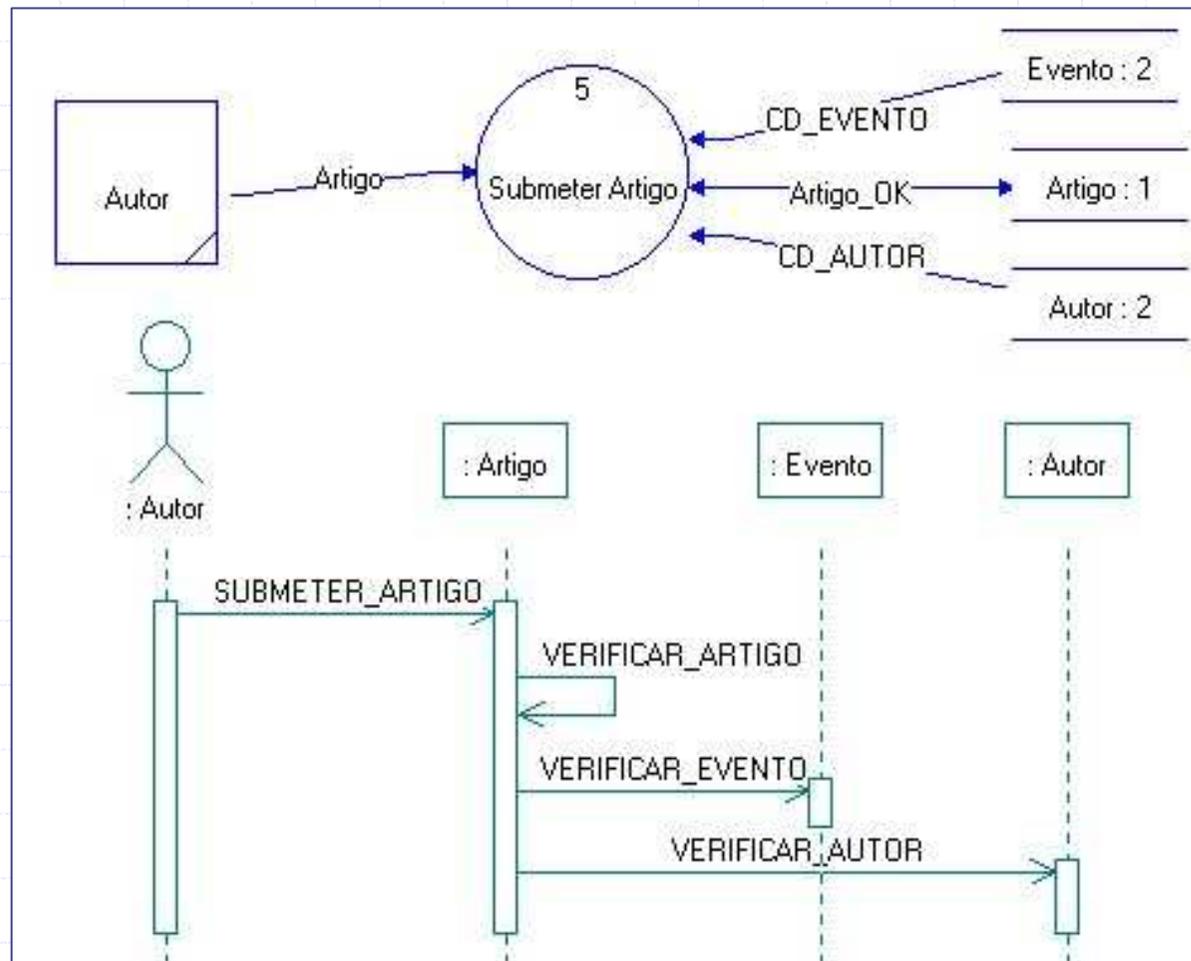
# Mapear p/ Diag. Classes

Exemplo de mapeamento de diagrama de classes



# Mapear p/ Diag. Seqüência

Exemplo de mapeamento de diagrama de seqüência



# *Desenvolvimento*

## Requisitos do Problema

- Estratégia de mapeamento

## Requisitos da Ferramenta

- permitir editoração dos diagramas estruturados e OO
- permitir importar os diagramas do Power Designer
- permitir editar os diagramas importados
- gerar diagramas OO através da utilização da estratégia
- permitir editar os diagramas gerados através do mapeamento

# *Especificação*

Ferramenta para especificação: *Rational Rose 2000*

Linguagem de especificação: UML (*Unified Modeling Language*)

Diagramas utilizados:

- Caso de Uso
- Classes
- Seqüência

# *Especificação*

## Diagrama de Casos de Uso (Primários)

- Importar DER
- Importar DFD
- Mapear para Caso de Uso
- Mapear para Classes
- Mapear para Seqüência

( Visualização na ferramenta *Rational Rose 2000* )

# *Especificação*

## Diagrama de Classes

- Pacotes
- Dicionário
- Objetos de Modelo
- Objetos de Modelo Estruturado
- Objetos de Modelo Orientado a Objetos
- Importação
- Mapeamento

( Visualização na ferramenta *Rational Rose 2000* )

# *Especificação*

## Diagramas de Seqüência (Casos de Uso Primários)

- Importar DER
- Importar DFD
- Mapear para Caso de Uso
- Mapear para Classes
- Mapear para Seqüência

( Visualização na ferramenta *Rational Rose 2000* )

# Implementação

## Exemplo do mapeamento de Diagrama de Caso de Uso

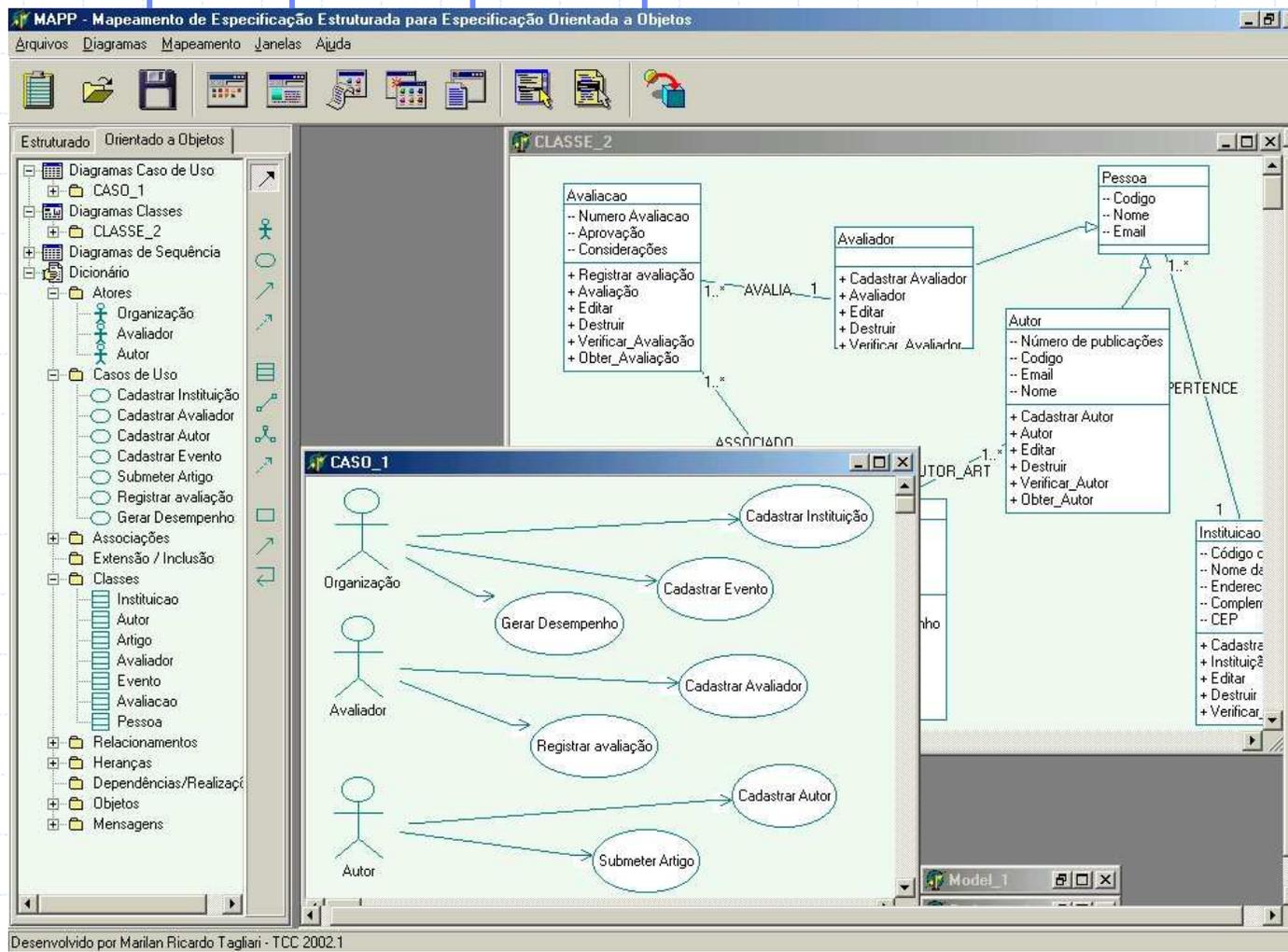
```
procedure TMapear.Est_OO_CASO;  
begin  
    // cria formulario de caso de uso  
    FormCASO := TFrmCASO(Dic_OO.NovoCASO);  
    // chama metodo para mapear atores  
    Map_Atores;  
    // chama metodo para mapear casos  
    Map_Casos;  
    // chama metodo para mapear associacoes  
    Map_Assoc;  
end;
```

## Exemplo do mapeamento de Atores

```
procedure TMapear.Map_Atores;
var
  x : integer;
  Ent : TObj_Ent_Ext;
  Ator : TObj_Ator;
  Graf : TObj_Grafico;
  prox : integer;
begin
  prox := 10;
  // percorrer entidades externas
  for x := 0 to Dic_Est.Ent_Ext.Count-1 do begin
    Ent := TObj_Ent_Ext(Dic_Est.Ent_Ext.item[x].Data);
    // criar ator
    Ator := TObj_Ator.Create;
    Ator.Nome := Ent.Nome;
    Ator.Code := Ent.Code;
    // adicionar no dicionario
    Dic_00.adic_Ator(Ator);
    // adicionar objeto grafico
    graf := Dic_00.adic_graf_Ator(FormCASO,Ator);
    // posicionar objeto grafico
    graf.top := prox;
    prox := prox + 100;
    graf.left := 10;
    // adicionar na lista de controle
    Hash_CASO.put(inttostr(Ent.ID),Graf);
  end;
end;
```

# Operacionalidade

## Tela principal do protótipo



# Resultados e Discussão

Três estudos de caso foram submetidos ao mapeamento:

- Eletrotécnica;
- Turismo;
- Escola Informática.

Componentes	Caso 1	Caso 2	Caso 3
	Mapeados/Desejáveis	Mapeados/Desejáveis	Mapeados/Desejáveis
<b>Diagrama de Caso de Uso</b>			
Atores	3/3	3/3	4/4
Casos de uso	5/5	8/8	12/12
Associações	5/5	8/8	12/12
<b>Diagrama de Classes</b>			
Classes	6/4	8/6	12/11
Atributos	11/11	28 / 28	52 / 52
Métodos	18/12	30 / 30	48 / 44
Relacionamentos	4/2	12/5	21/10
Heranças	2/2	0 / 0	3/3
<b>Diagrama de Seqüência</b>			
Objetos	12/12	23 / 23	32 / 31
Mensagens	20 / 20	37 / 37	54 / 50

# Conclusões

Principais conclusões:

- Trabalho atendeu aos objetivos;
- Mapeamento foi possível e é viável;
- Importação de especificações possibilita o reuso;
- Ferramenta apresenta condições de uso acadêmico;
- Diag. caso de uso apresenta idéia inicial;
- Diag. classes é necessário refinamento manual;
- Diag. seqüência é necessário refinamento manual;
- Qualidade do mapeamento depende dos modelos origem;
- Ferramenta possibilitou geração de código Java e Delphi;
- Ferramenta possibilitou geração de arquivos XML;

# *Limitações*

Principais limitações da ferramenta:

- Importação exclusiva do *Power Designer 6*;
- Geração de código apenas para Java e Delphi.

# *Extensões*

Sugestões para trabalhos futuros:

- Leitura de outros repositórios de ferramentas CASE;
- Leitura de outras versões do *Power Designer*;
- Aperfeiçoar estratégia de mapeamento;
- Permitir editoração de outros diagramas da UML;
- Estratégia de mapeamento para especificações WEB;
- Gerar código para outras linguagens.

# Referências Bibliográficas

DEMARCO, Tom. **Análise estruturada e especificação de sistemas.**

Rio de Janeiro: Campus, 1989.

FURLAN, José David. **Modelagem de objetos através da UML.** São

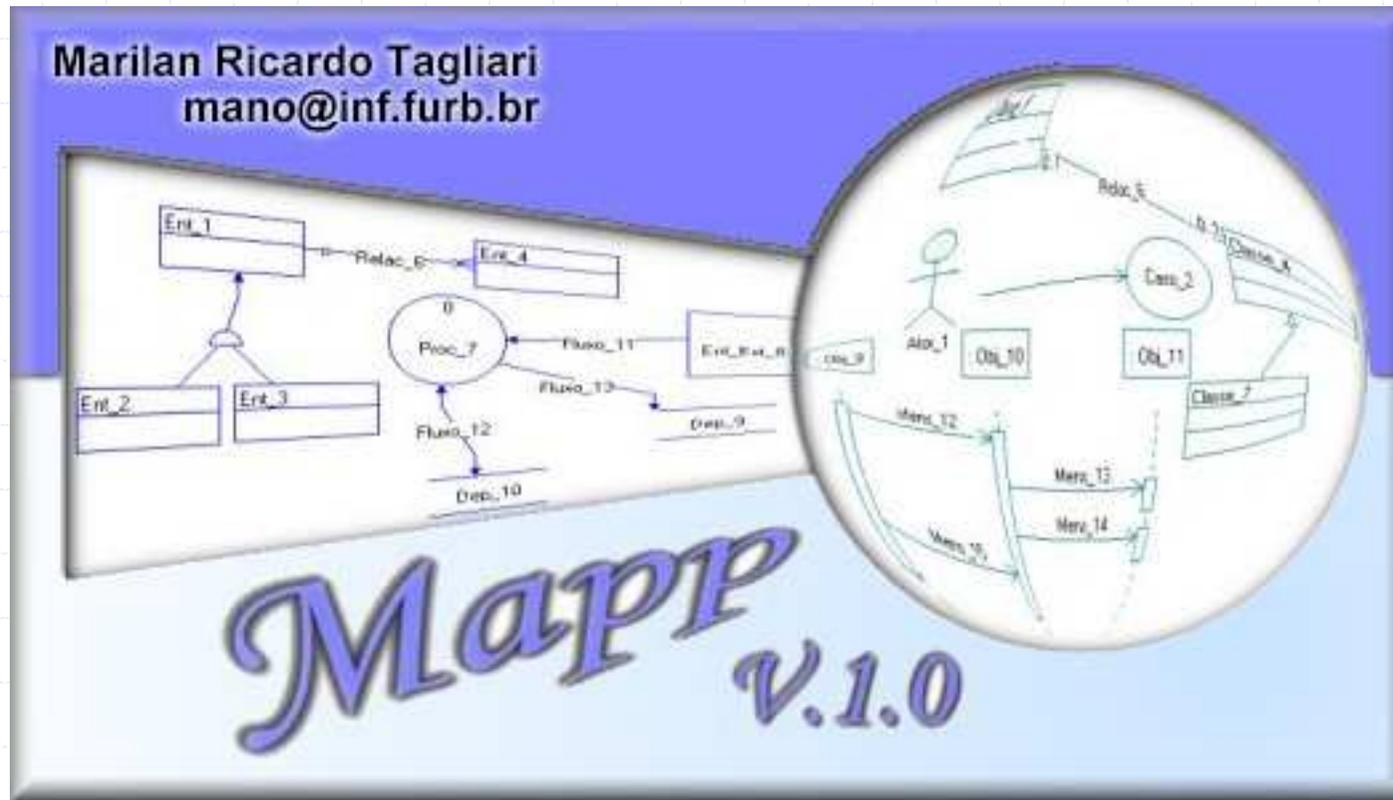
Paulo: Makron Books, 1998.

GEORGE, Joseph. A strategy for mapping from function-oriented software models to object-oriented models. **Software Engineering Notes (ACM SIGSOFT)**, v. 21, n. 2, p. 56-63, Março de 1996.

SALDANHA, Emerson B. **Protótipo de uma ferramenta de apoio à migração de especificação estruturada para especificação orientada por objetos.** 1999. 46 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

ZIBELL, Dílson. **Protótipo de auxílio ao mapeamento de especificações estruturadas para especificações orientadas a objetos.** 1996. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

# Protótipo



# ***Questionamentos***

