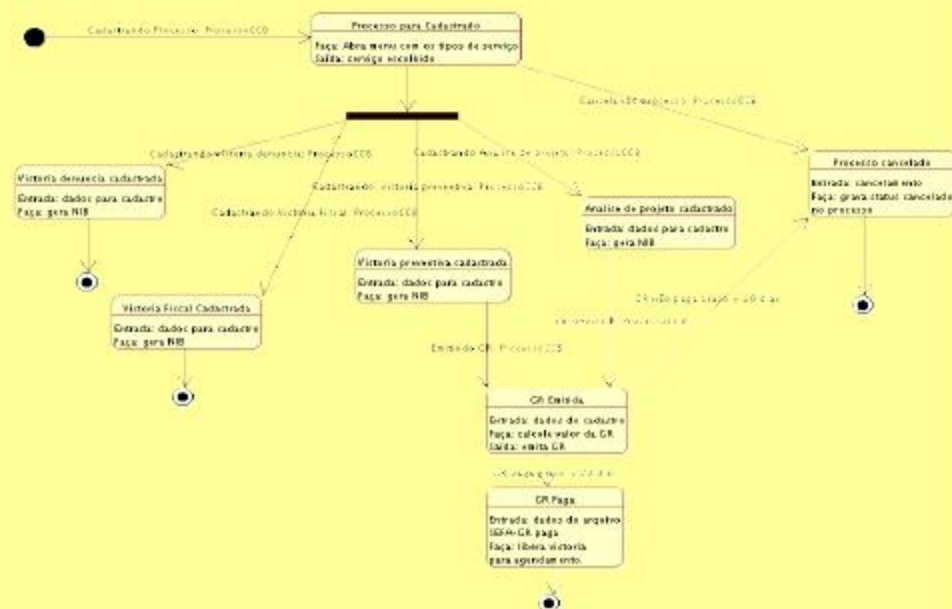
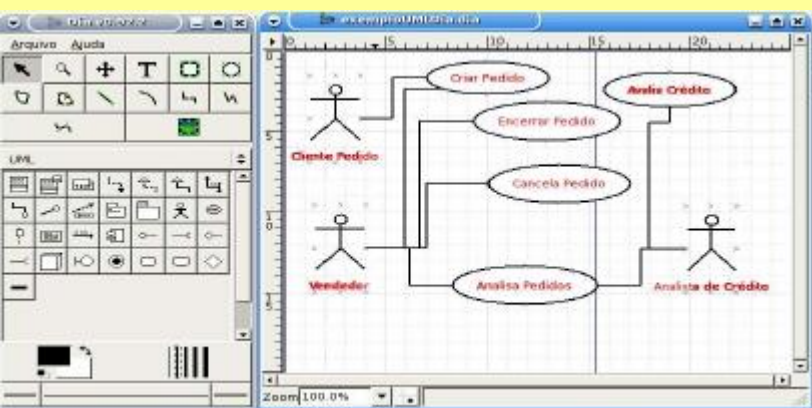


Ferramentas para UML



Introdução

- A Linguagem UML.
- Os Software Livres para UML e suas características (Dia, Umbrello, ArgoUML).

Uma breve introdução a utilização dos Softwares.

- Do que estes softwares são capazes.
- Quais os suportes a UML.
- Licenças, Desenvolvedores, Dicas, Versões,
- Projetos relacionados e Plataformas suportadas.

Muitos Exemplos de Diagramas UML.



A Linguagem UML

- Surgiu no final dos anos 80 e no início dos anos 90.
- É uma linguagem de modelagem de projetos Orientados a Objetos, padrão da OMG.
- É a notação (principalmente gráfica) utilizada por métodos para expressar projetos.
- Utilizado para modelagem de sistemas na maioria das empresas que desenvolvem Software OO.

Ferramentas para UML

Comerciais e "Free Editions"

- MagicDraw
- Togheter
- Poseidon
- Enterprise Architect
- Rational Rose
- Jude
- Omondo Eclipse UML

Livres (BSD e GPL)

- Umbrello
- ArgoUML
- Dia
- BOUML
- Fajuba
- StarUML

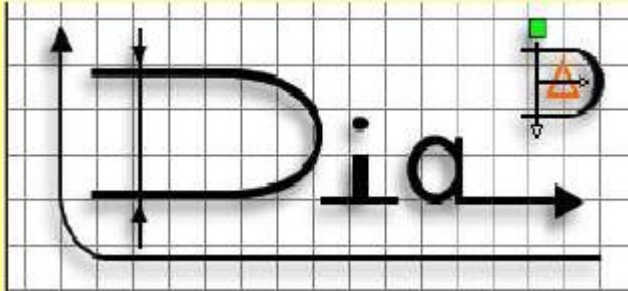
Ferramentas para UML Tabela

Comerciais e "Free Editions"

- MagicDraw (\$ 1,599,00)
- Together (\$ 11.500,00)
- Poseidon (\$ 1.549,00)
- Enterprise Architect
(\$ 2.500,00)
- Rational Rose (\$ 5.995,00)
- Jude(\$280,00 1usuário/1ano)
- Omondo Eclipse UML
(\$ 18.136,31 / 5 usuários)

Livres (BSD e GPL)

- Umbrello +
- ArgoUML +
- Dia = (\$ 0,00 e ainda
leva os fontes)
- BOUML
- Fajuba



Dia UML

- Dia é um programa baseado em gtk+ para criação do diagrama, liberado sob a licença do GPL.
- É parte do projeto Gnome (Linux).
- Atualmente tem objetos especiais de Lógica, entidade e relacionamento, diagramas UML, fluxogramas, diagramas da rede, e circuitos simples entre outros.



Plataformas Suportadas

- Gnu/Linux.
- Variações de Unix.
- Foi “portada” para Windows (Win32).

Do que o Dia e capaz em relação a UML?



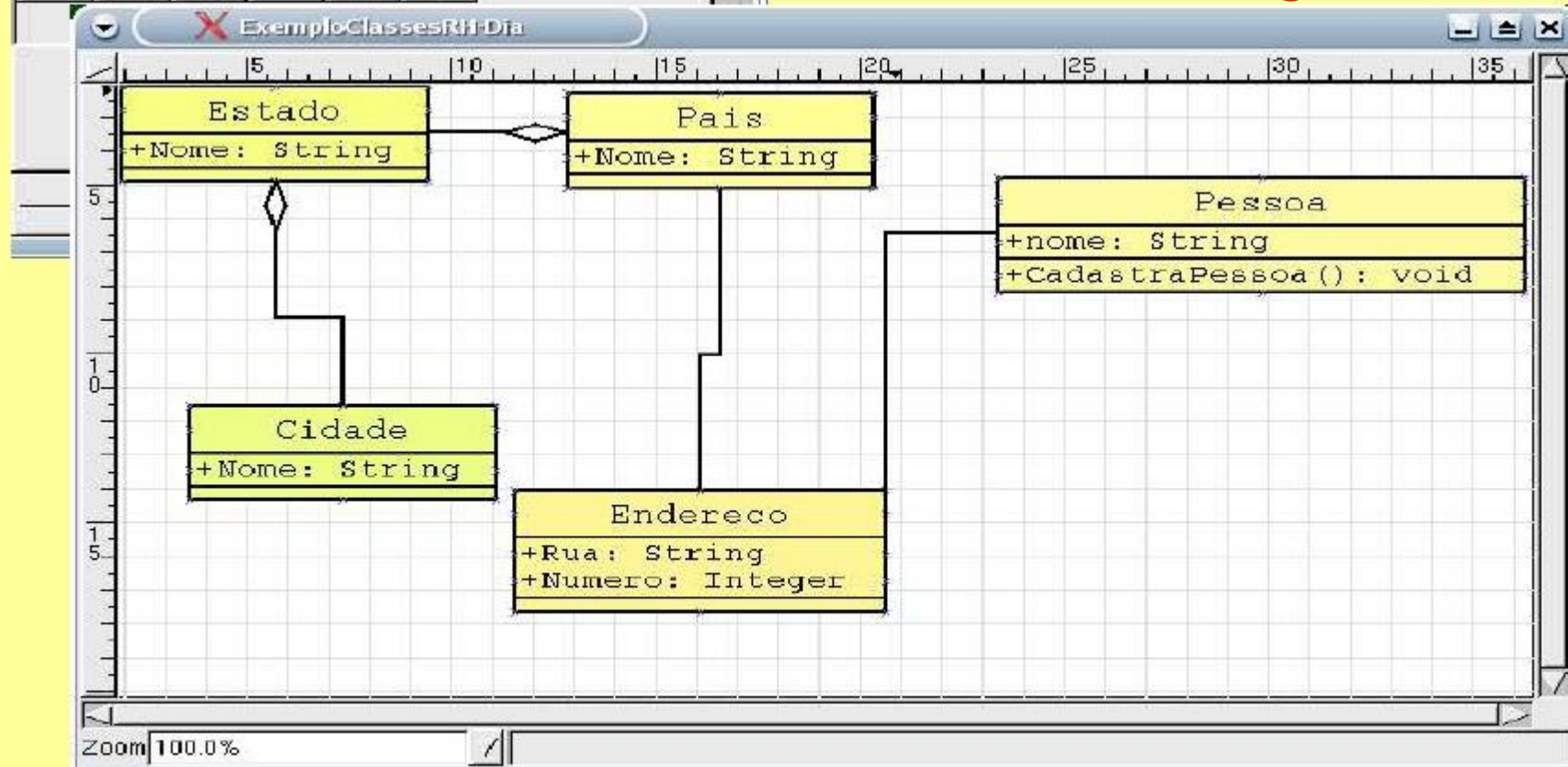
- Com o Dia podem ser criados a maioria dos Diagramas da UML , sua estrutura e simplesmente a união de componentes/objetos utilizados na Modelagem UML.
- Podem ser exportados para arquivos gráficos.
- Os diagramas podem ter componentes/Objetos de outras estruturas como ER, Fluxograma, Rede, Lógica etc.
- Personalização de Barras de Elementos.



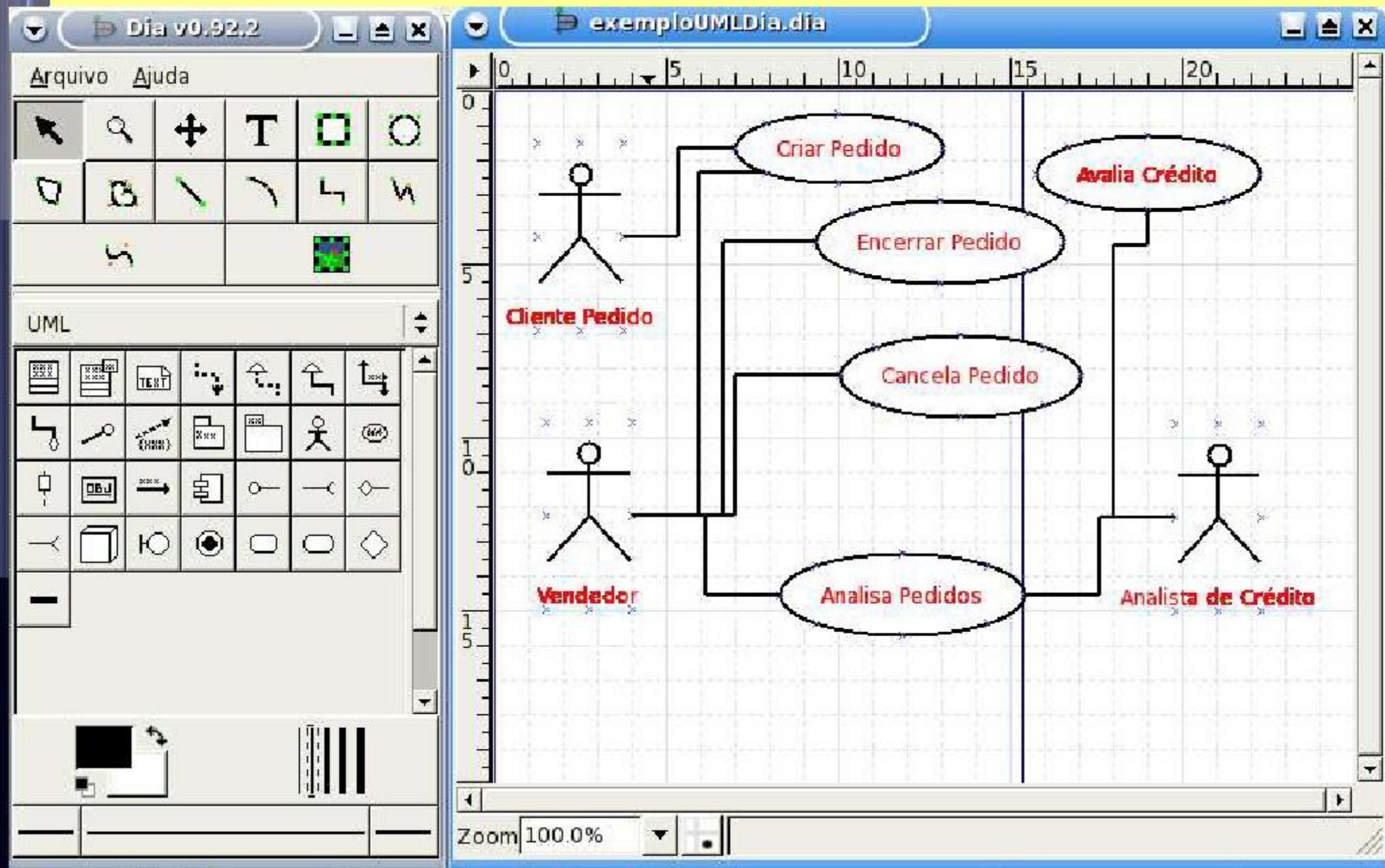
Dia UML

Tipo de Diagramas
< Objetos UML

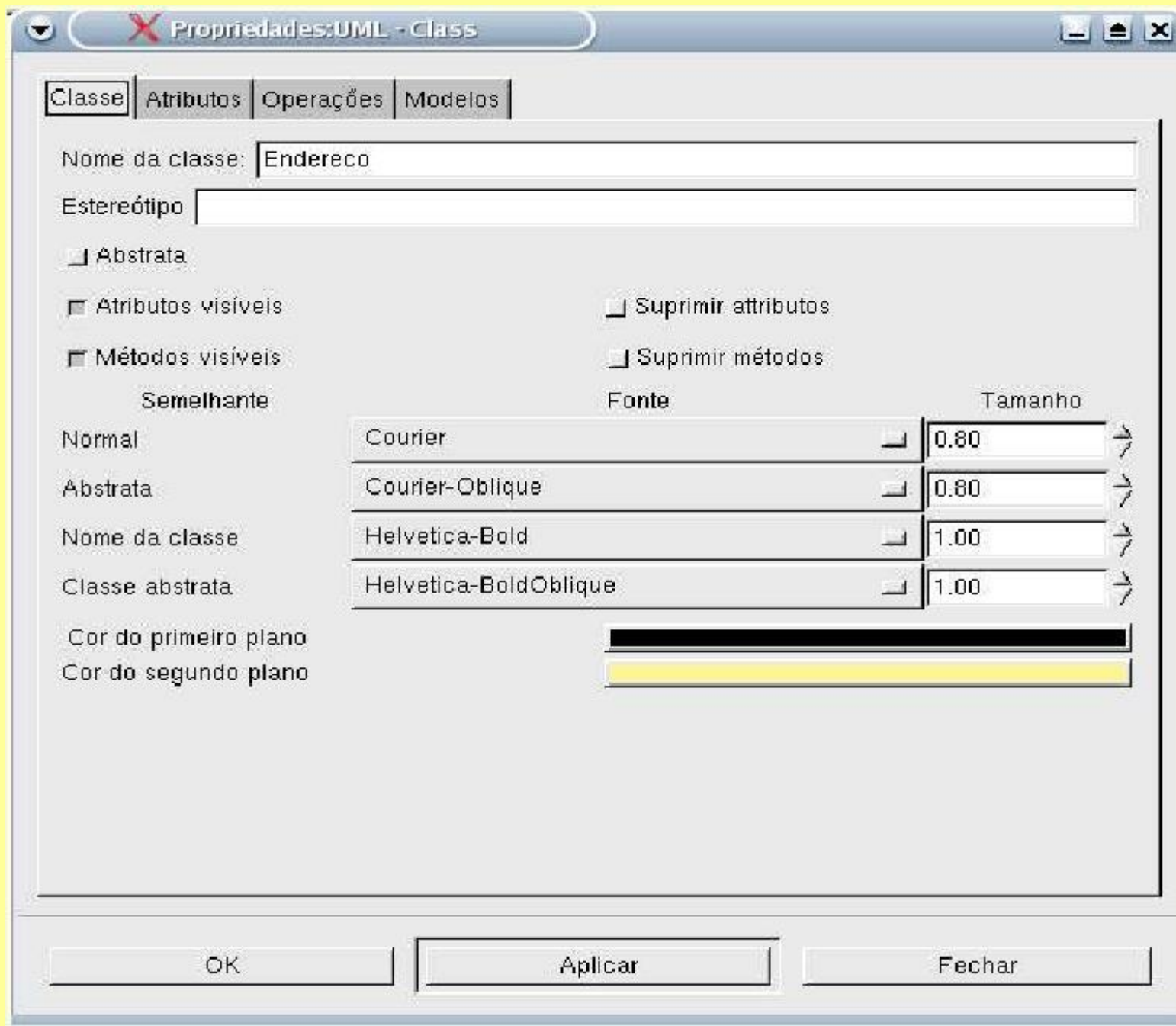
Editor de diagramas



Dia UML



Propriedades das Classes



Atributos de Classes

Propriedades:UML - Class

Classe | Atributos | Operações | Modelos

+Rua: String
+Numero: Integer

Novo
Deletar
Subir
Descer

Dados do atributo

Nome: Numero

Tipo: Integer

Valor:

Visibilidade: Publico

☐ Escopo da classe

OK Aplicar Fechar

Operações das Classes

Propriedades:UML - Class

Classe | Atributos | Operações | Modelos

+CadastraPessoa(): void

Novo
Deletar
Subir
Descer

Dados do método

Nome: CadastraPessoa

Tipo: void

Esteréotipo:

Visibilidade: Público

☐ Escopo da classe

Tipo de herança: Polimórfica (virtual)

☐ Query

Parâmetros:

Novo
Deletar
Subir
Descer

Dados dos parâmetros

Nome:

Tipo:

Valor padrão:

Direção: Indefinido

OK Aplicar Fechar

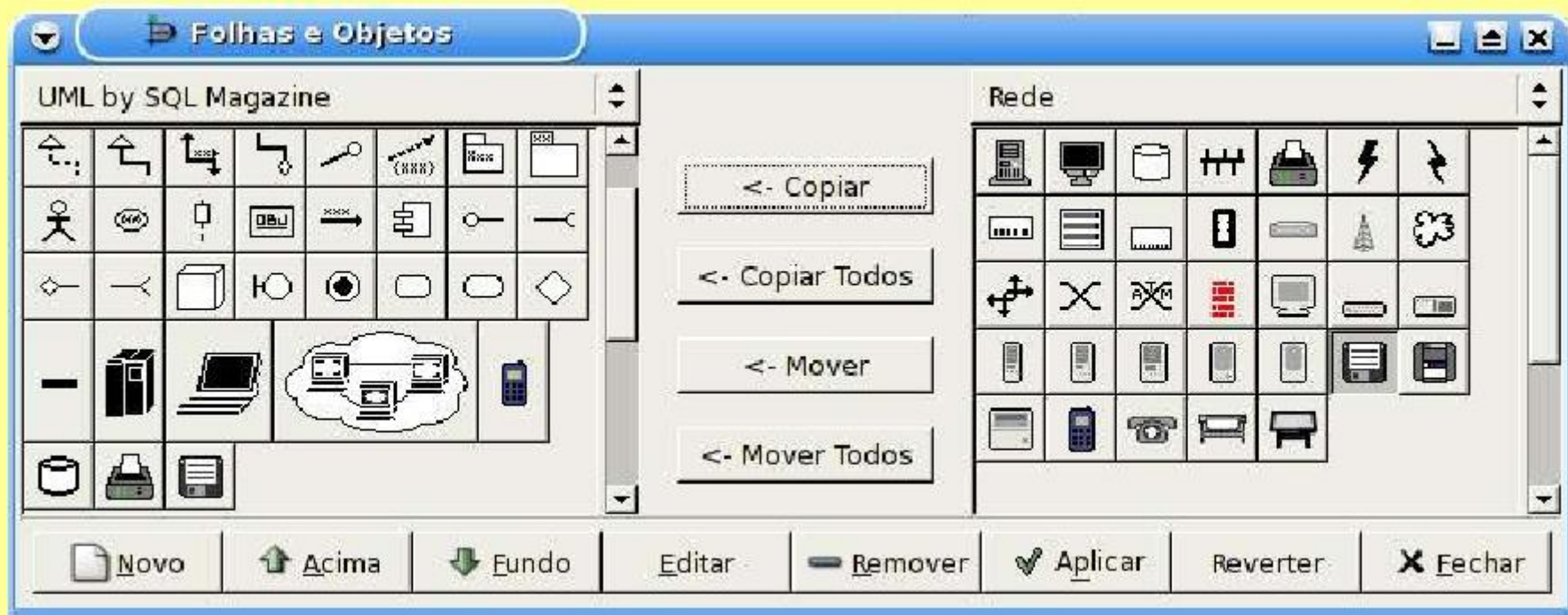
Propriedades de Mensagens e Nota





Personalização de Barra de Componentes

- Podemos fazer um mix de componentes da UML com os demais componentes suportados pela ferramenta.
- A ferramenta permite também que seja criada uma folha de objetos (UML , UML by Ambiente Livre , UML c/ ER ,etc)



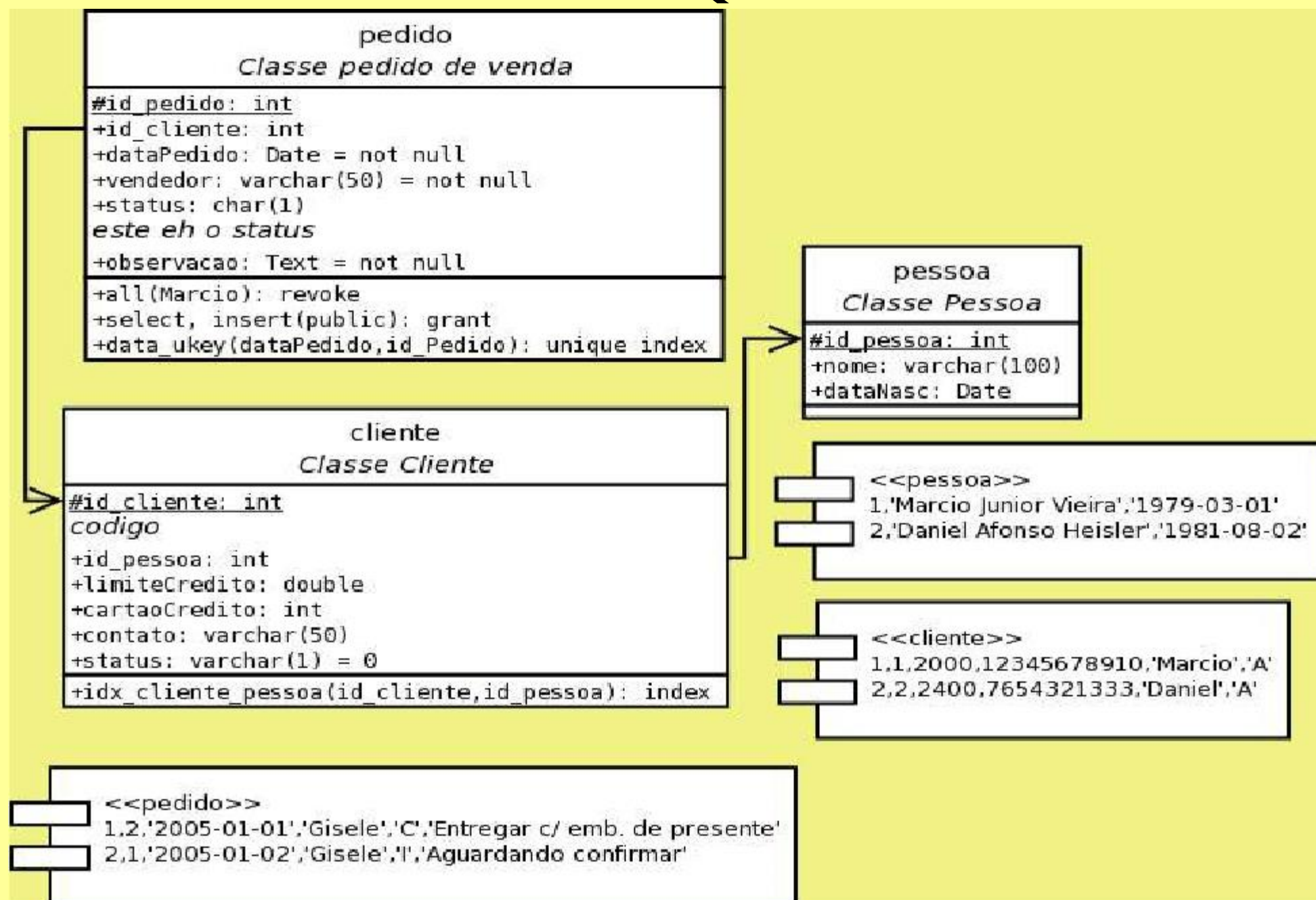


Informações Adicionais



- A versão atual do Dia é a 0.97.3 (desde 09/2014) .
- ArgoUML2Dia converte-se do formato de ArgoUML ao Dia, embora todos os tipos do objeto não são suportados ainda.
- Agata (gerador de relatório). Pode gerar diagramas do Dia através de Bases SQL.
- TeDia2SQL e Dia2SQL Converter diagramas do Dia para bases de dados do SQL (Sybase, PostgreSQL, oracle, DB/2, Mssql, MySQL)
- A maioria das distribuições Gnu/Linux já vem com o Dia nos CDs de Instalação.

Dia2SQL



Executando o Dia2SQL Script PHP

```
php dia2sql.php -f <formato> -i <arquivoDiaUML.dia> -o meuarquivogerado.sql  
pgsql – PostgreSQL  
mysql – MySQL
```

site do dia2SQL - <http://codigolivres.org.br/projects/dia2sql/>

Table: cliente

Purpose: Classe Cliente

CREATE TABLE cliente

```
(  
  id_cliente    int PRIMARY KEY AUTO_INCREMENT,  
  id_pessoa     int,  
  limiteCredito double,  
  cartaoCredito int,  
  contato       varchar(50),  
  status        varchar(1)  
);
```

ALTER TABLE cliente COMMENT = 'Classe Cliente';

ALTER TABLE cliente MODIFY id_cliente int NOT NULL;

ALTER TABLE cliente DROP PRIMARY KEY;

ALTER TABLE cliente ADD PRIMARY KEY (id_cliente);

ALTER TABLE cliente ADD INDEX idx_cliente_pessoa (id_cliente,id_pessoa);

ALTER TABLE cliente ADD FOREIGN KEY (id_pessoa) REFERENCES pessoa(id_pessoa);

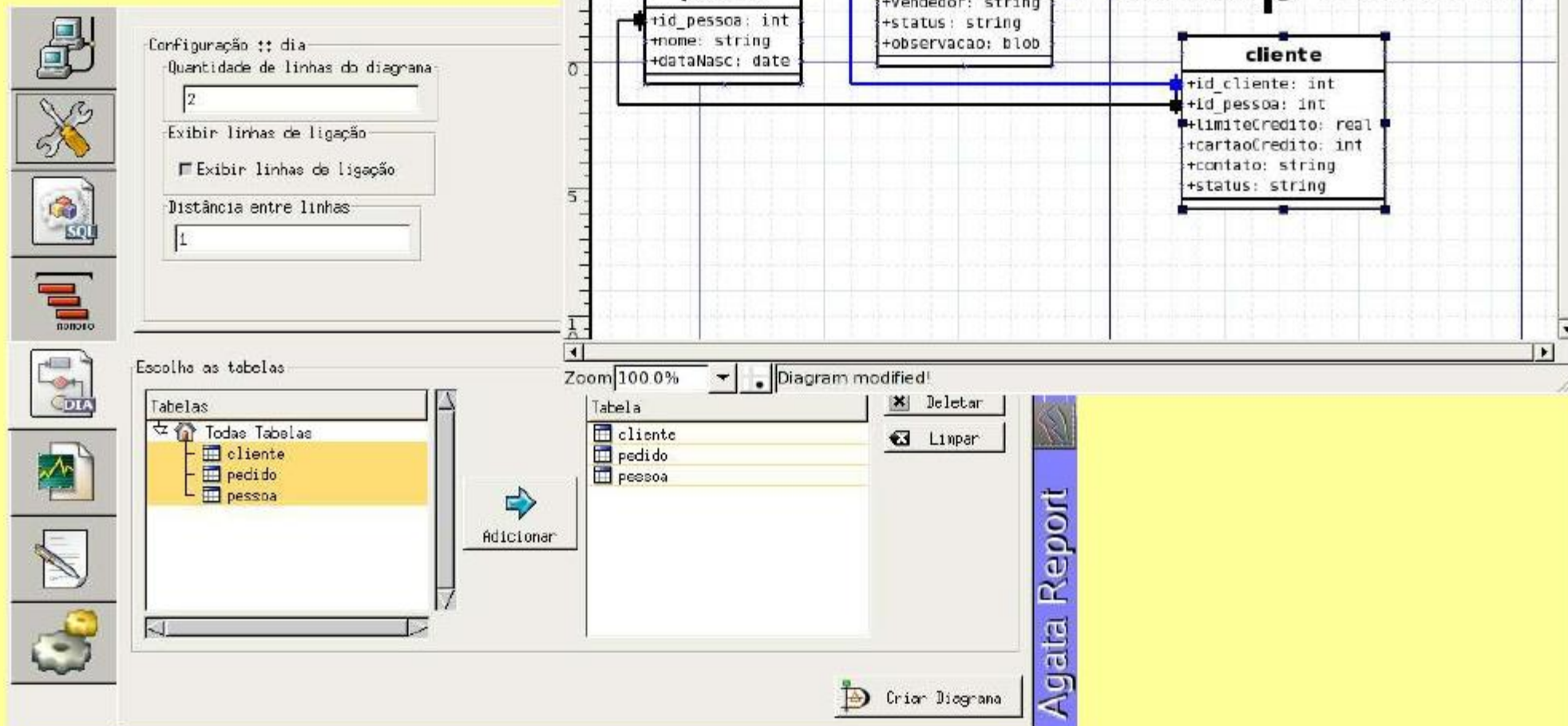
INSERT INTO cliente VALUES (1,1,2000,12345678910,'Marcio','A');

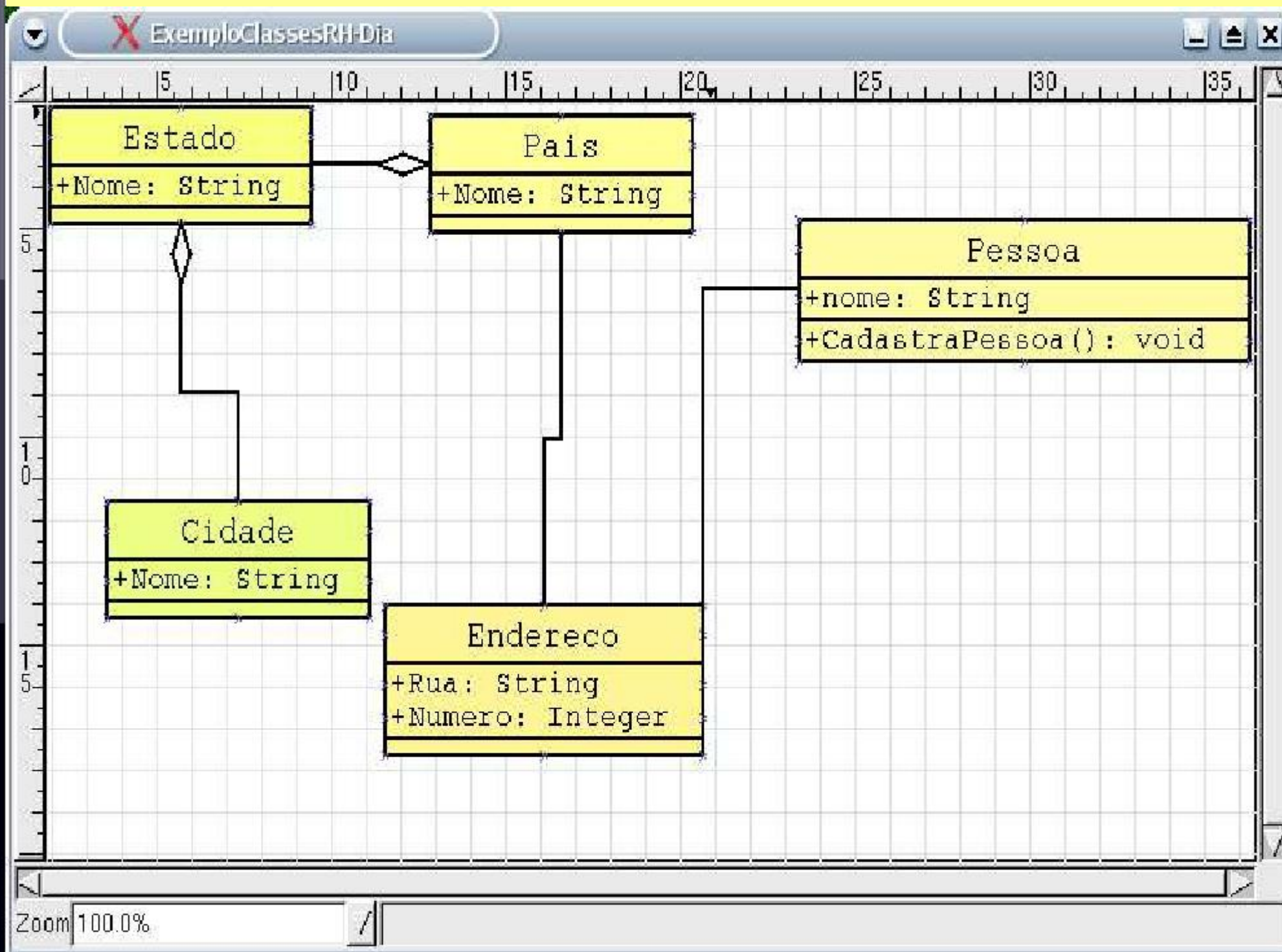
INSERT INTO cliente VALUES (2,2,2400,7654321333,'Daniel','A');

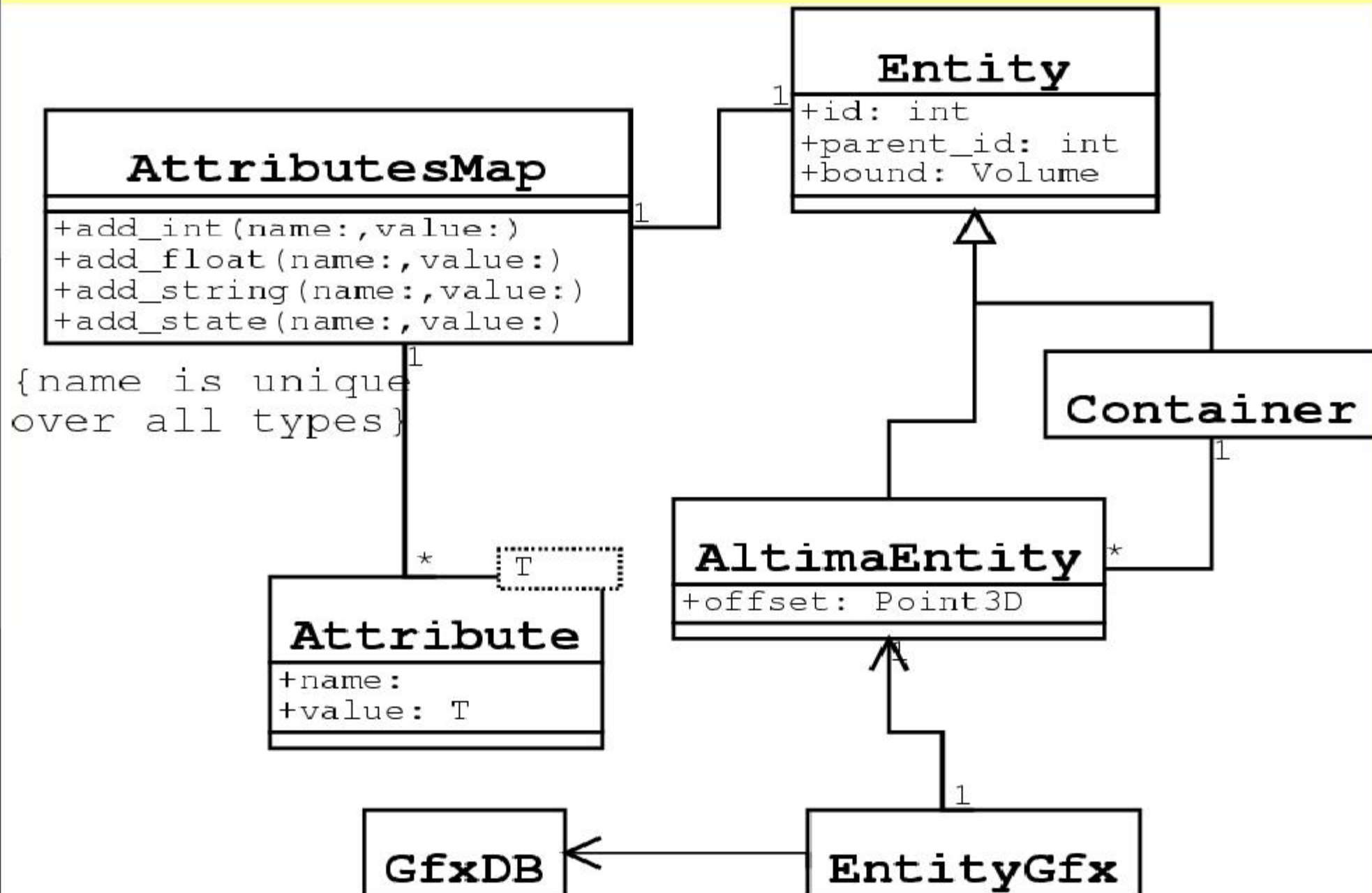
Engenharia Reversa da Bases SQL: Dia+AgataReport

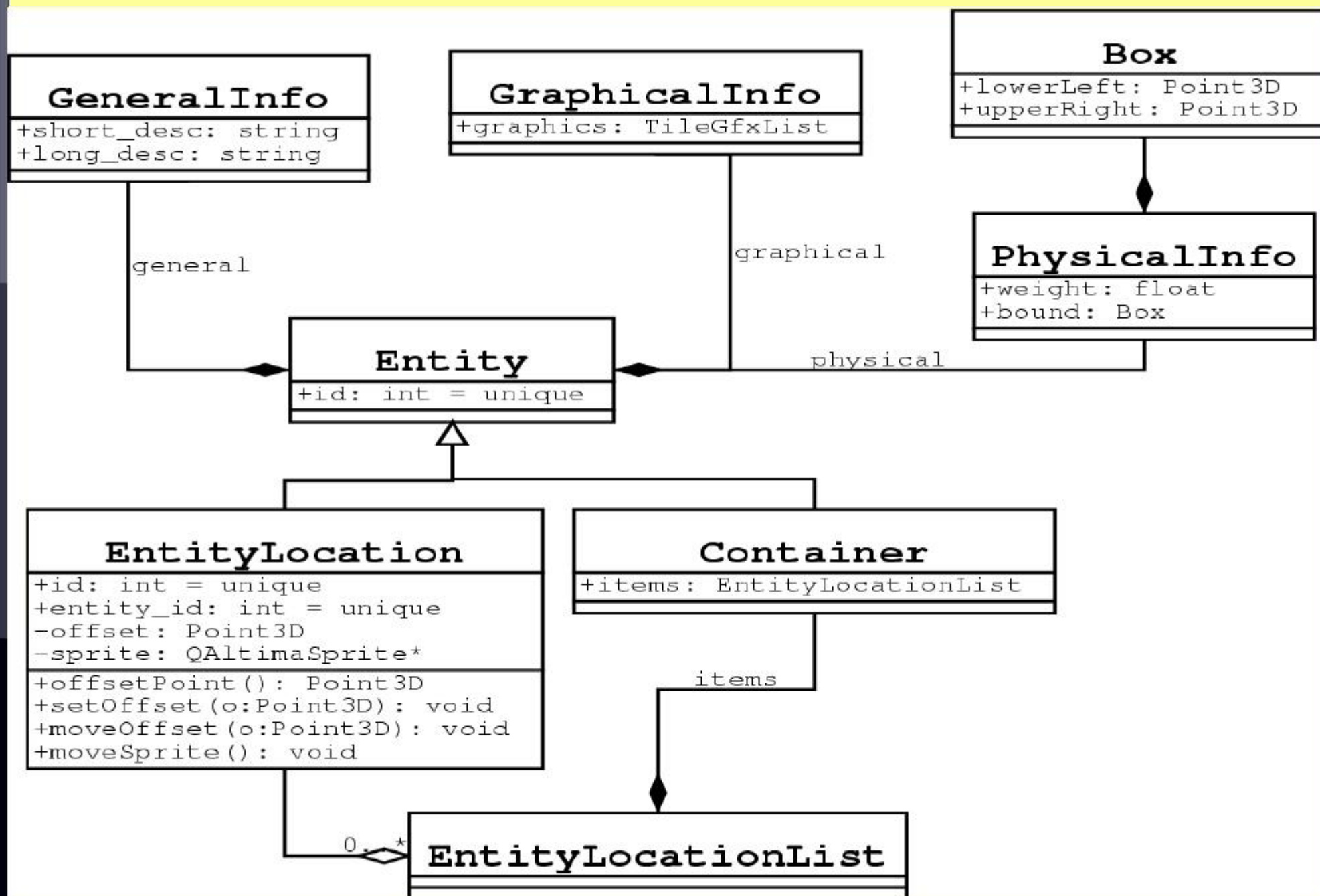


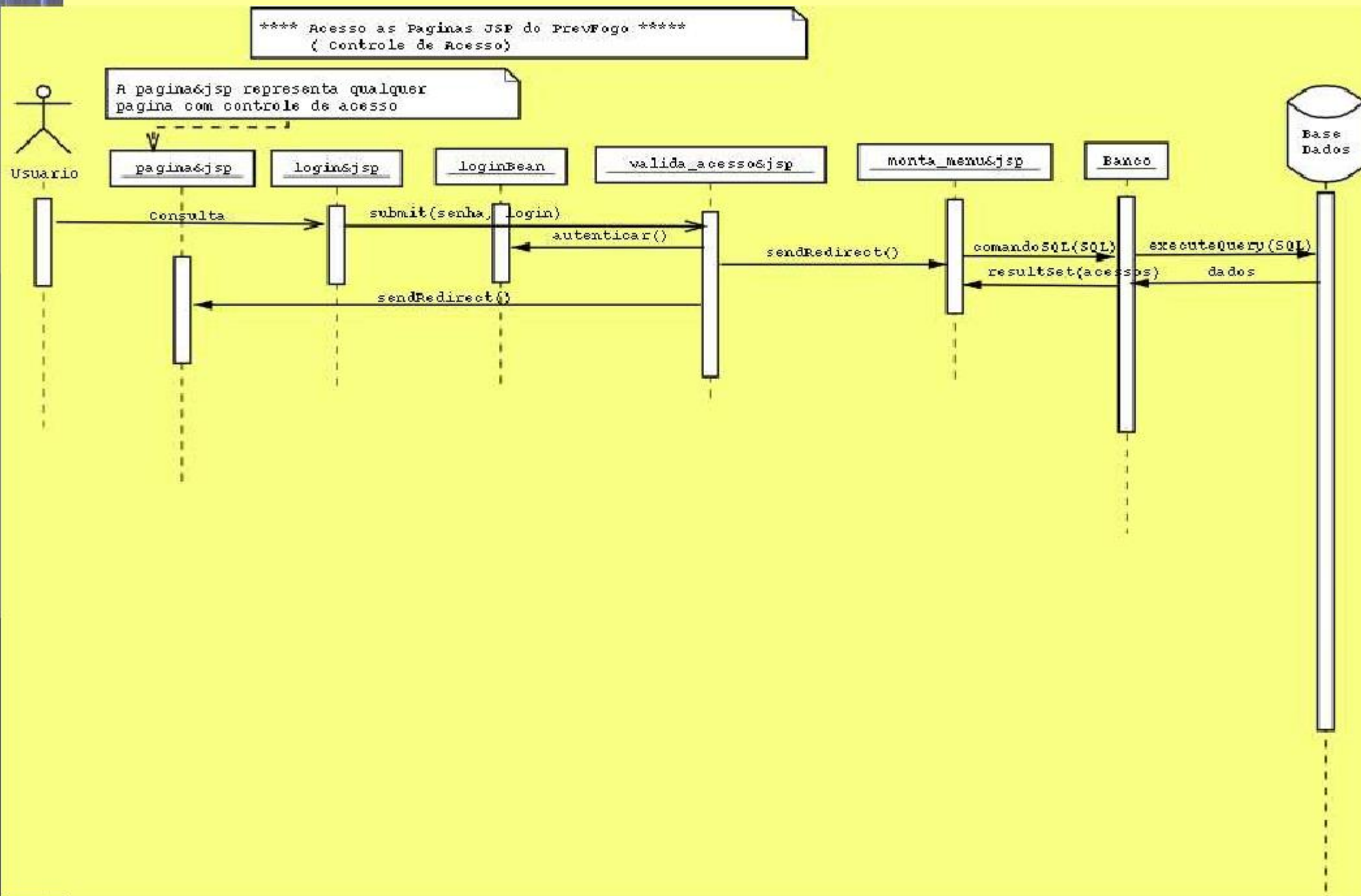
Dia+AgataReport

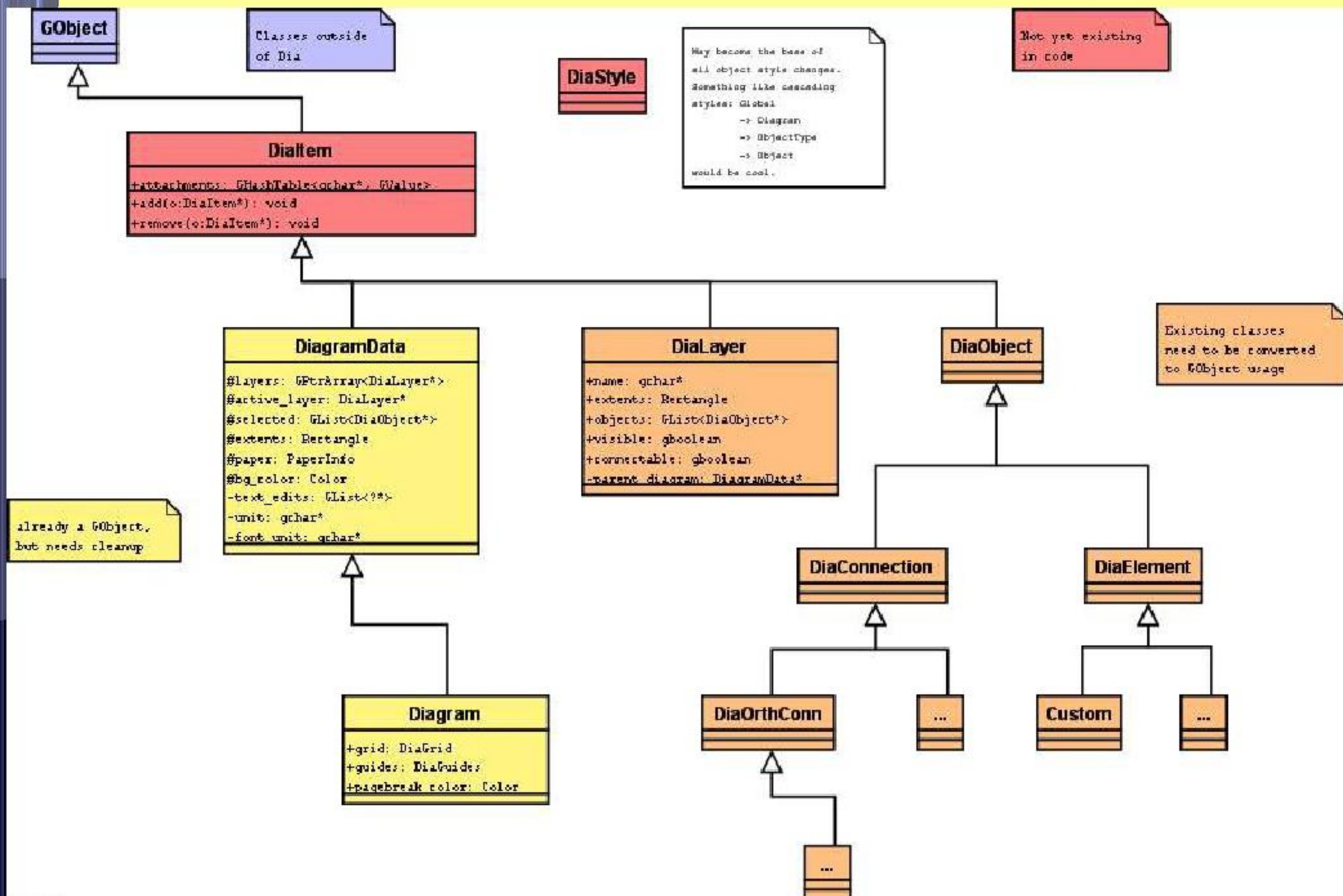


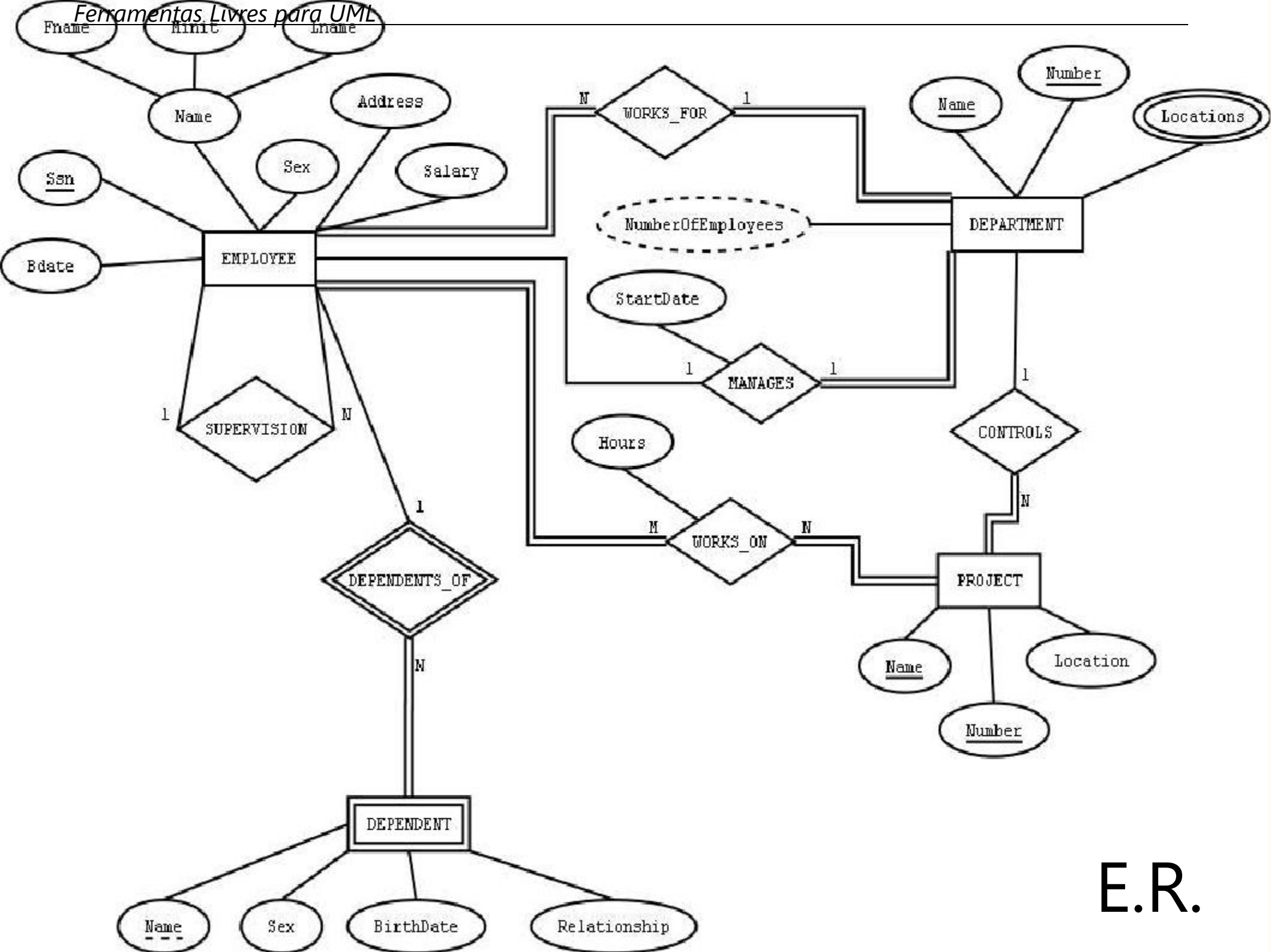




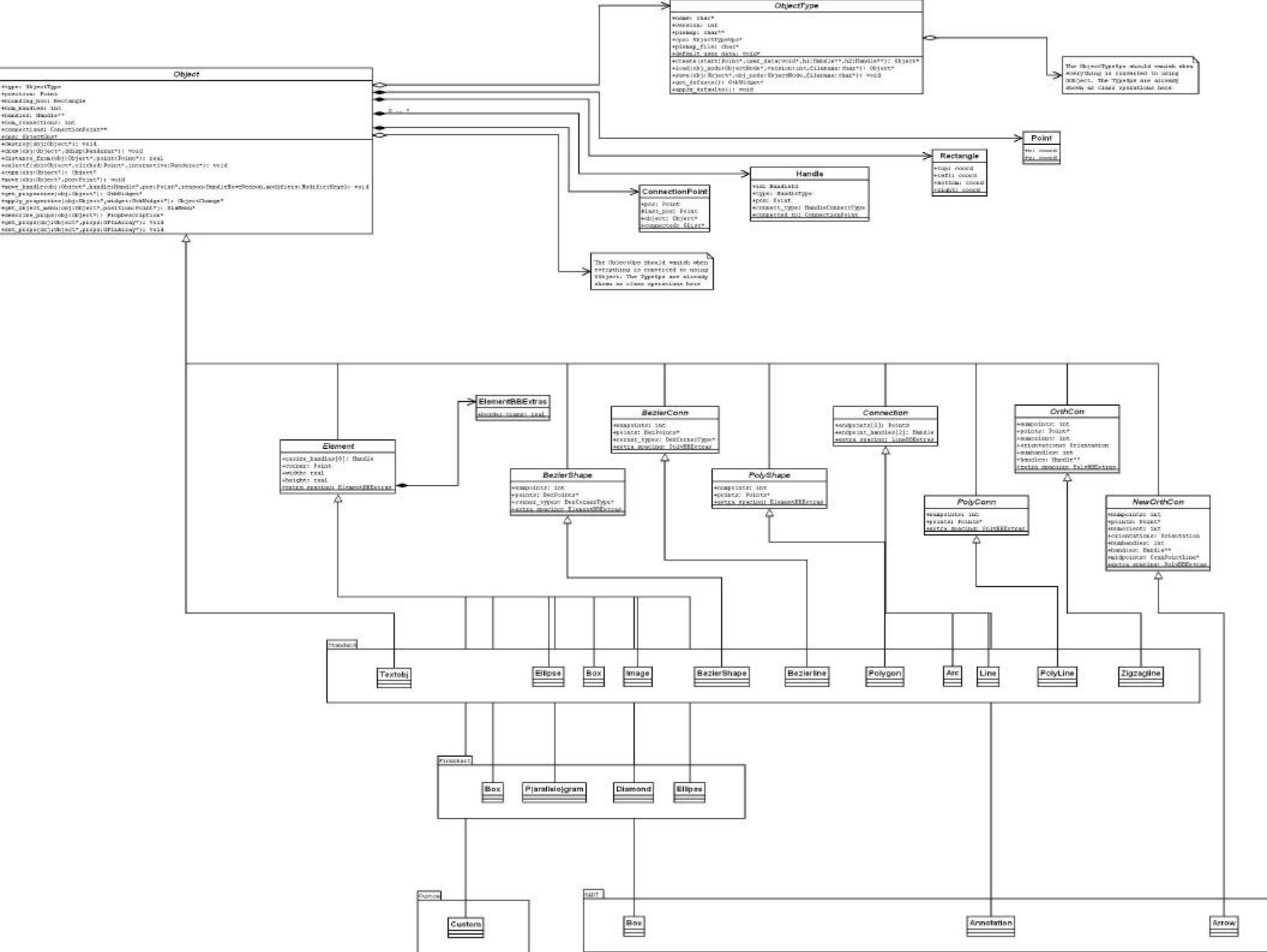


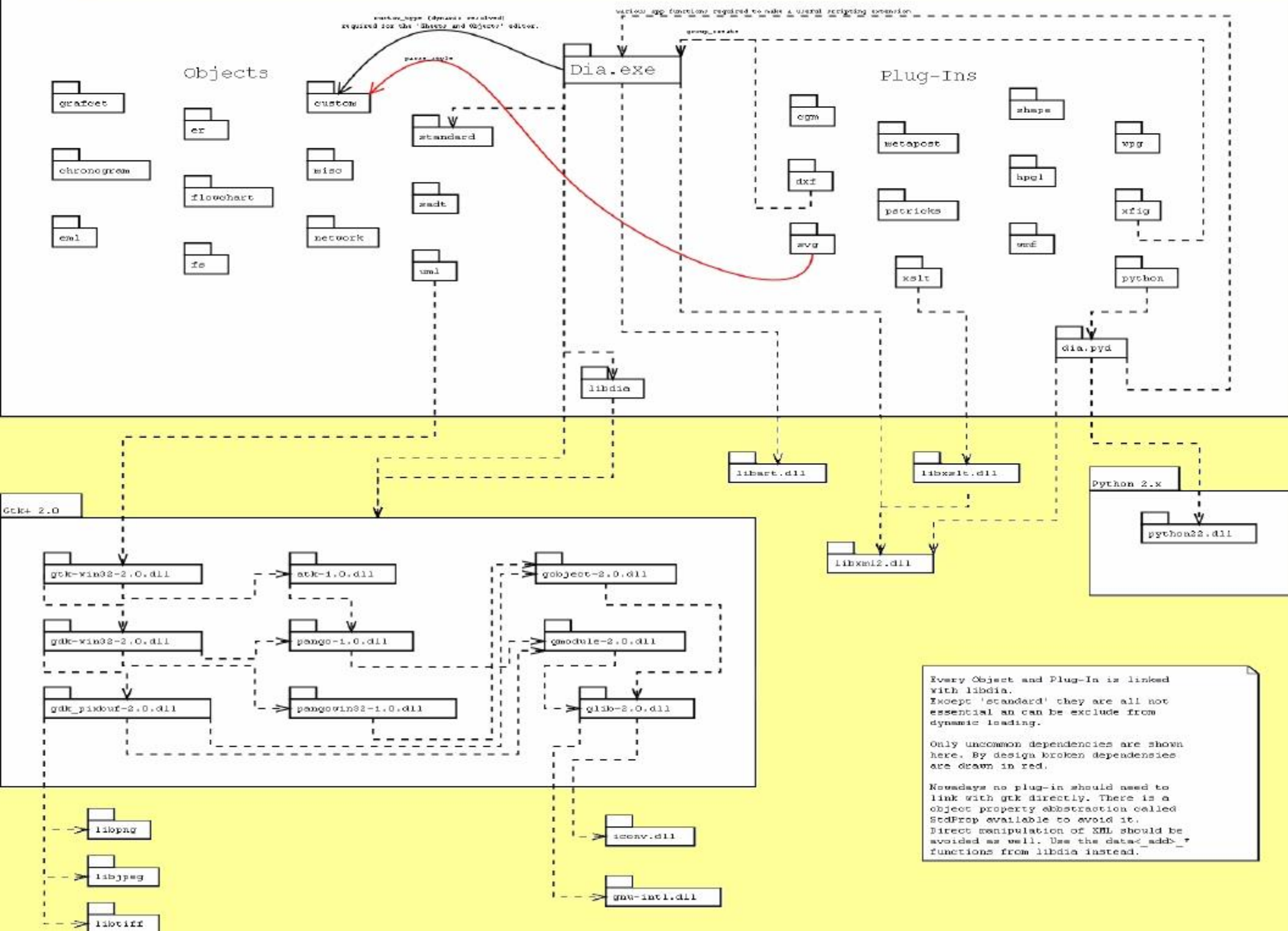






E.R.





ArgoUML

ArgoUML é uma ferramenta CASE baseada na notação UML (Unified Modeling Language).

Foi desenvolvido pela comunidade de desenvolvedores de código livre Tigris vinculada a Universidade da Califórnia , Berkeley.

Sua interface é bem completa o que a torna um pouco complexa de manipular.

Do que ele é capaz ?

- Desenhar e imprimir diagramas UML.
- Gerar declarações de classes Java.
- Exportar documentação para páginas Web em Java.
- Gerar arquivos Gráficos (gif).
- Com auxilio de software de terceiros e possível gerar comandos SQL.
- Engenharia Reversa: fornece uma estrutura modular da engenharia reversa de classes Java.
- Exportar dados para o padrão XMI (baseado no formato XML)

Diagramas UML Suportados

- Diagrama de classe
- Diagrama de estados
- Diagrama de Atividades
- Diagrama de casos de uso
- Diagramas de colaboração
- Diagrama de Utilização/Componentes
- Diagrama de seqüência

Padrão UML Utilizado

- O MetaModelo Padrão de UML implementado é O UML 1.4 da OMG (Object management Group)
- Atualmente a OMG está na versão 0.34 (12/2011)
Suporta plataformas Linux e Windows



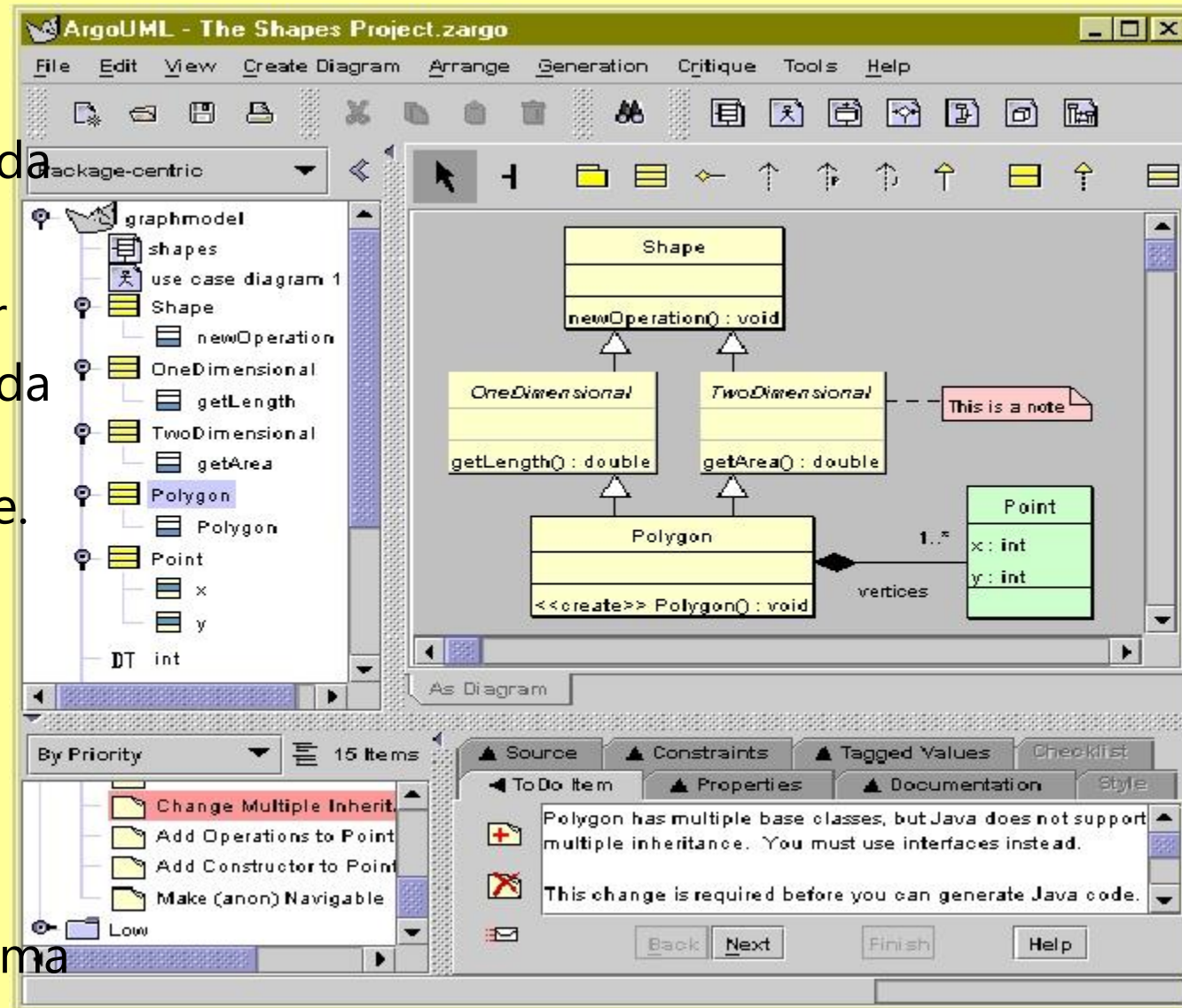


Requisitos para instalação

- Java 1.4 ou superior (JavaWebStart)
- ArgoUML é escrito inteiramente em Java e usa as classes padrões do Java . Isto permite que ArgoUML funcione virtualmente em toda a plataforma que suporte uma JVM (Java Virtual Machine).
- Internacionalização: inglês americano, inglês britânico, francês, alemão, espanhol, Russo, Norueguês, chinês.

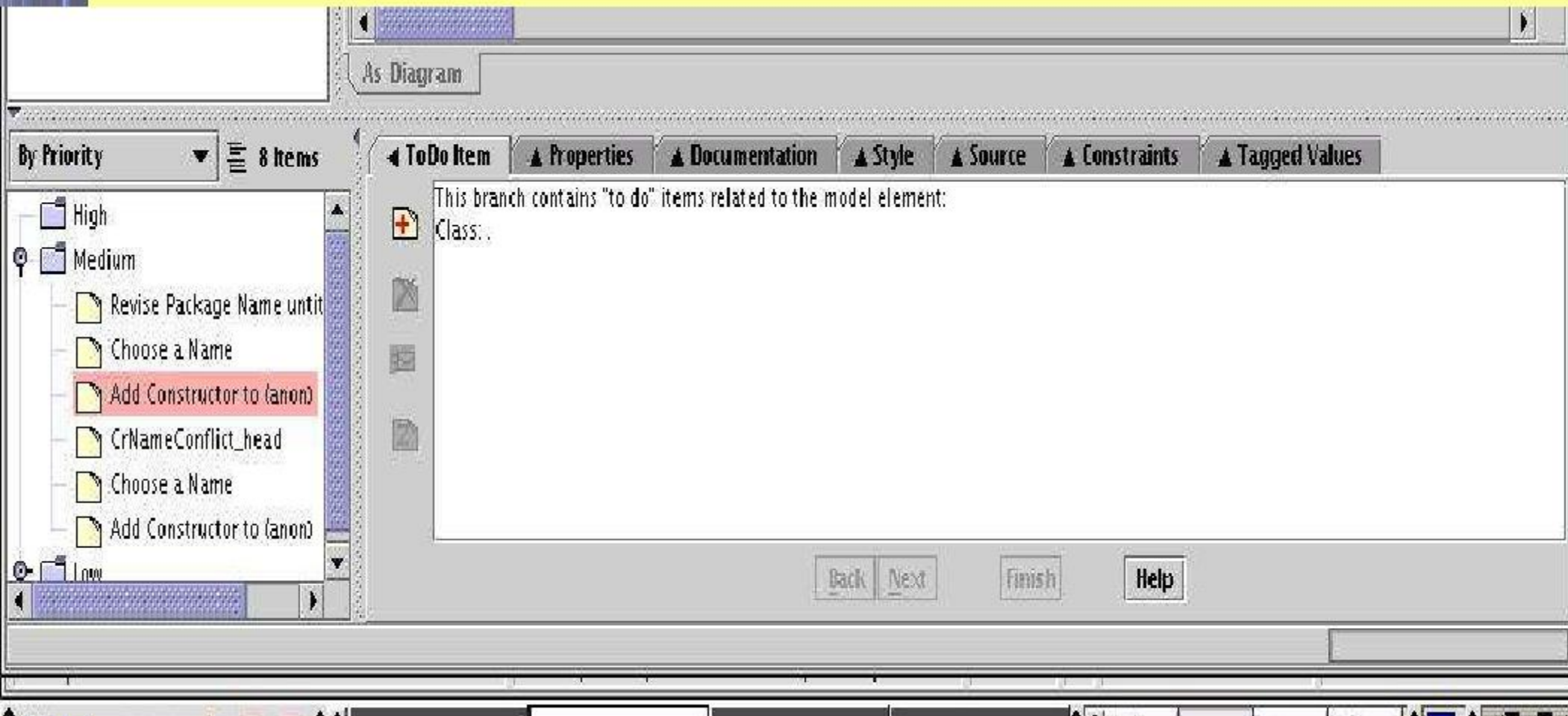
ArgoUML

- A tela de ArgoUML é dividida em 4 painéis:
- Esquerda superior : uma vista hierárquica do projeto atual.
- Direita superior : editor para a parte selecionada do projeto, neste caso um diagrama da classe.
- Esquerda inferior : desenhador de Lista ToDo (para fazer) .
- Direita inferior : detalhes do objeto selecionado no diagrama ToDo (Para fazer) .



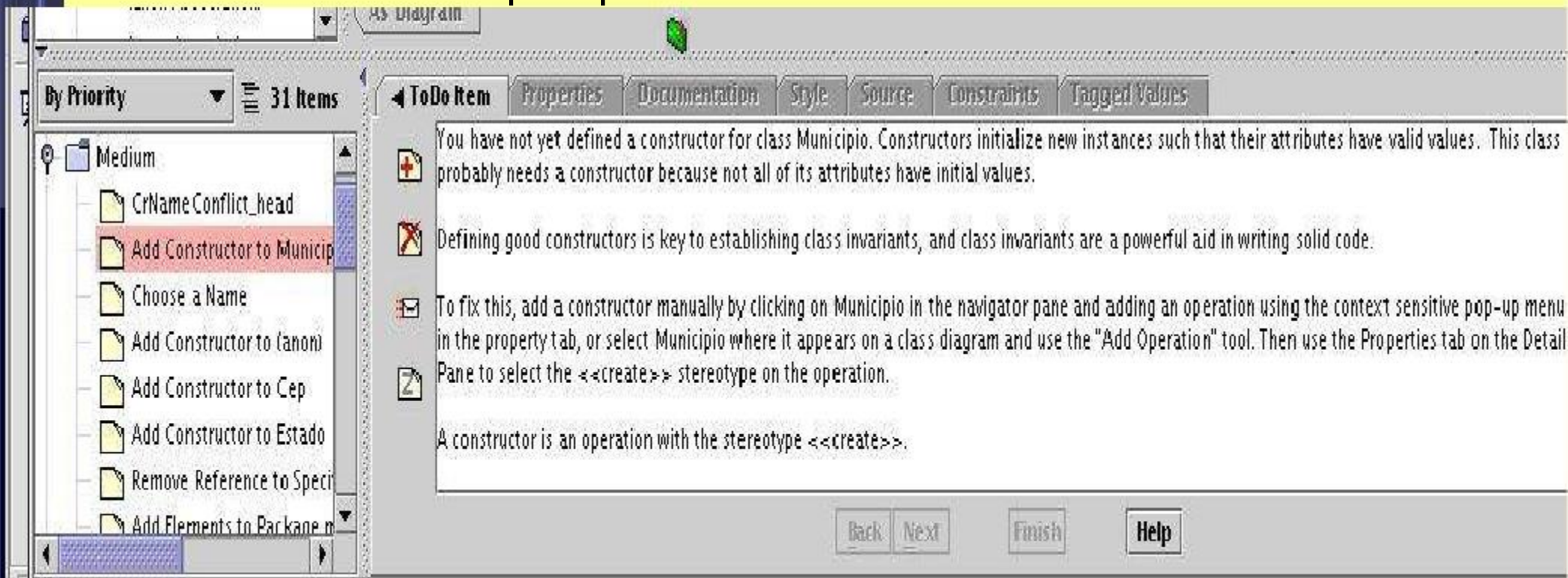
Placa de Detalhes

- Permite que você edite detalhes do elemento selecionado do projeto.



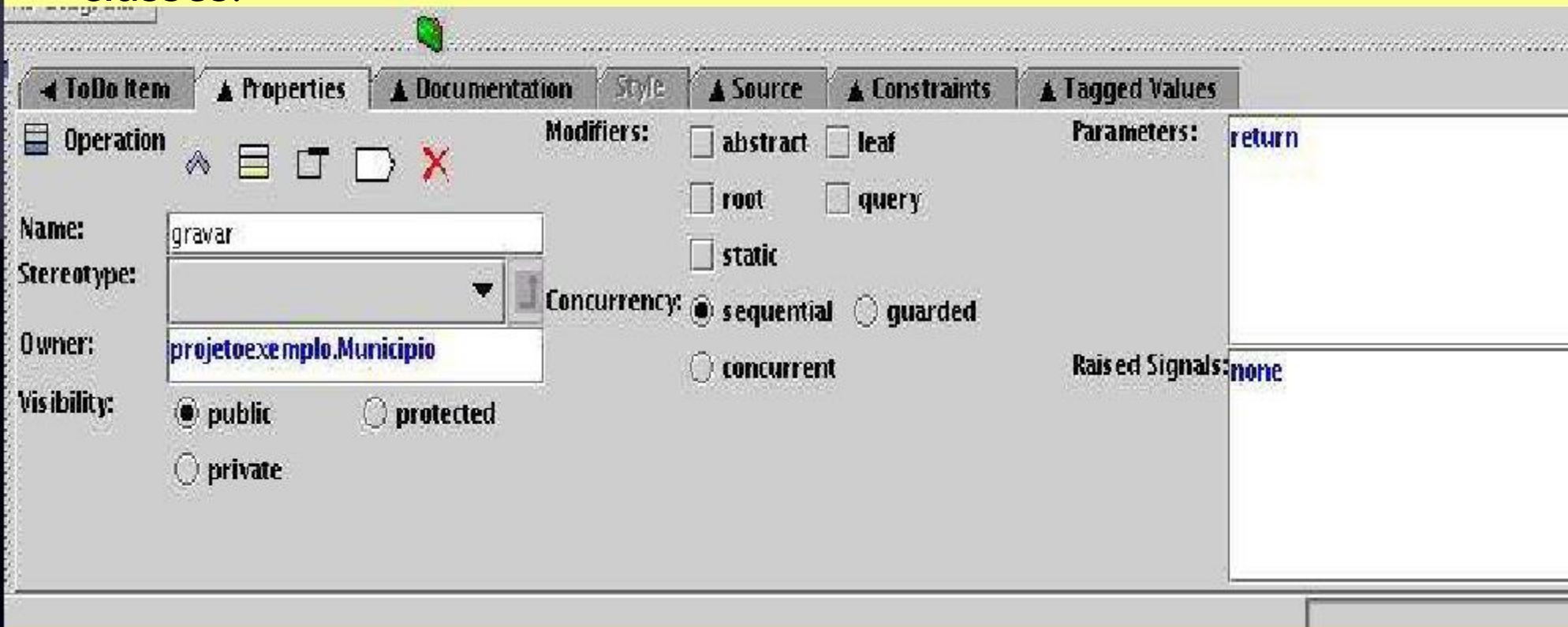
ToDoItem

- O software interage um o usuário através de uma lista de procedimentos que ele deve executar, criando pendências para o termino da elaboração de um diagrama. Mostra a descrição do item selecionado, consistindo com 3 parágrafos curtos sobre prováveis problemas e etapas de como resolver. (Ex. Método Construtor)
- Diferencial de qualquer outra ferramenta do mercado.



Propriedades

- Mostra a propriedade do elemento selecionado do projeto, as propriedades variam de acordo com o elemento.
- propriedades como: nome da classe, visibilidade, generalizações, especializações, atributos, métodos e ainda alguns botões que auxiliam a excluir a classe ou criar novos atributos, métodos ou classes.



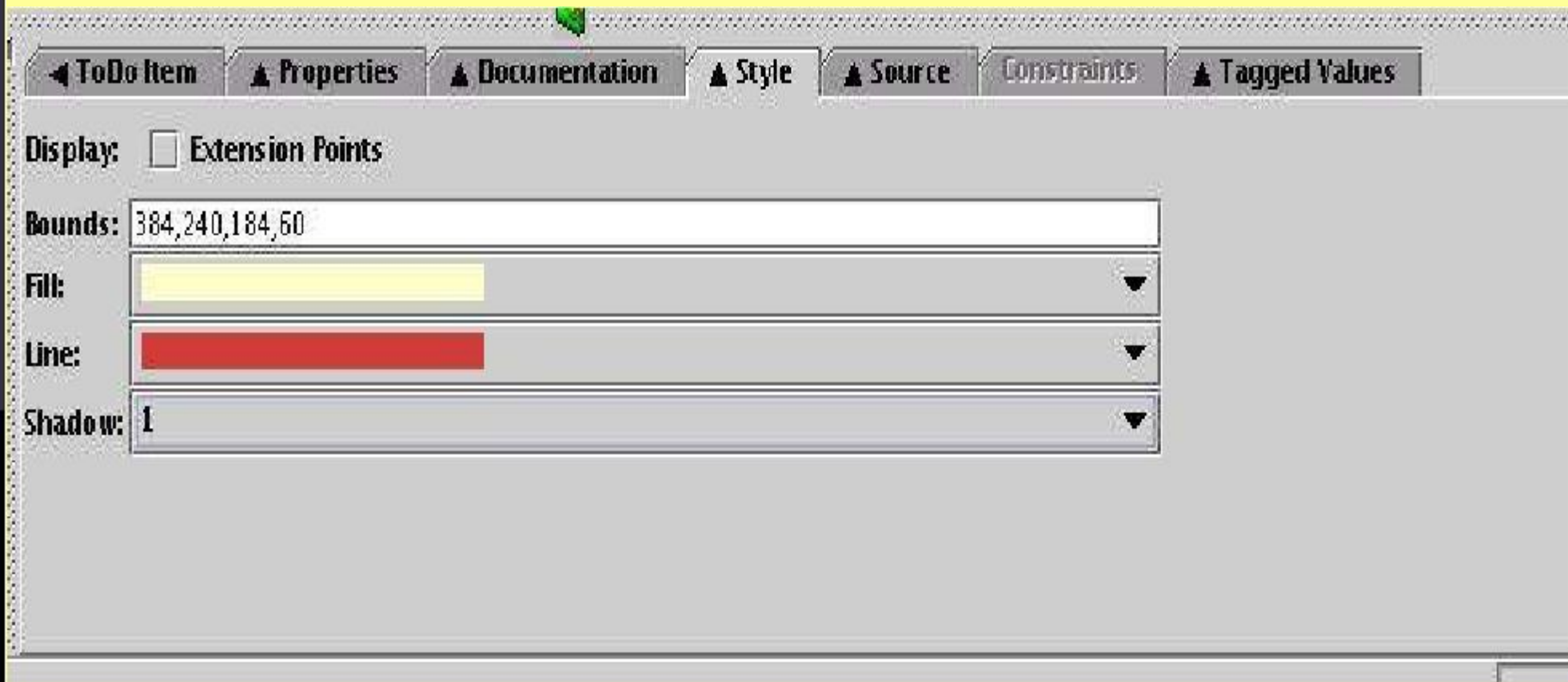
Documentação

- Documentação do elementos selecionados : autor , versão e descritivo etc.

The screenshot displays a software interface for UML documentation. At the top, there is a horizontal tab bar with six tabs: 'ToDo Item', 'Properties', 'Documentation' (which is currently selected), 'Style', 'Source', and 'Constraints'. Below the tabs, on the left side, is a section titled 'Documentation' with a radio button icon. This section contains several input fields: 'Author:' with the text 'Marcio Junior Vieira', 'Version:' with 'V1.0', 'Since:' with '10/09/2003', 'Deprecated:' with an unchecked checkbox, and 'See:' with a large empty text area. To the right of these fields is a large text area labeled 'Documentation:' containing the text 'Somente o Super usuário terá o direito de eliminar dados do CEP.'

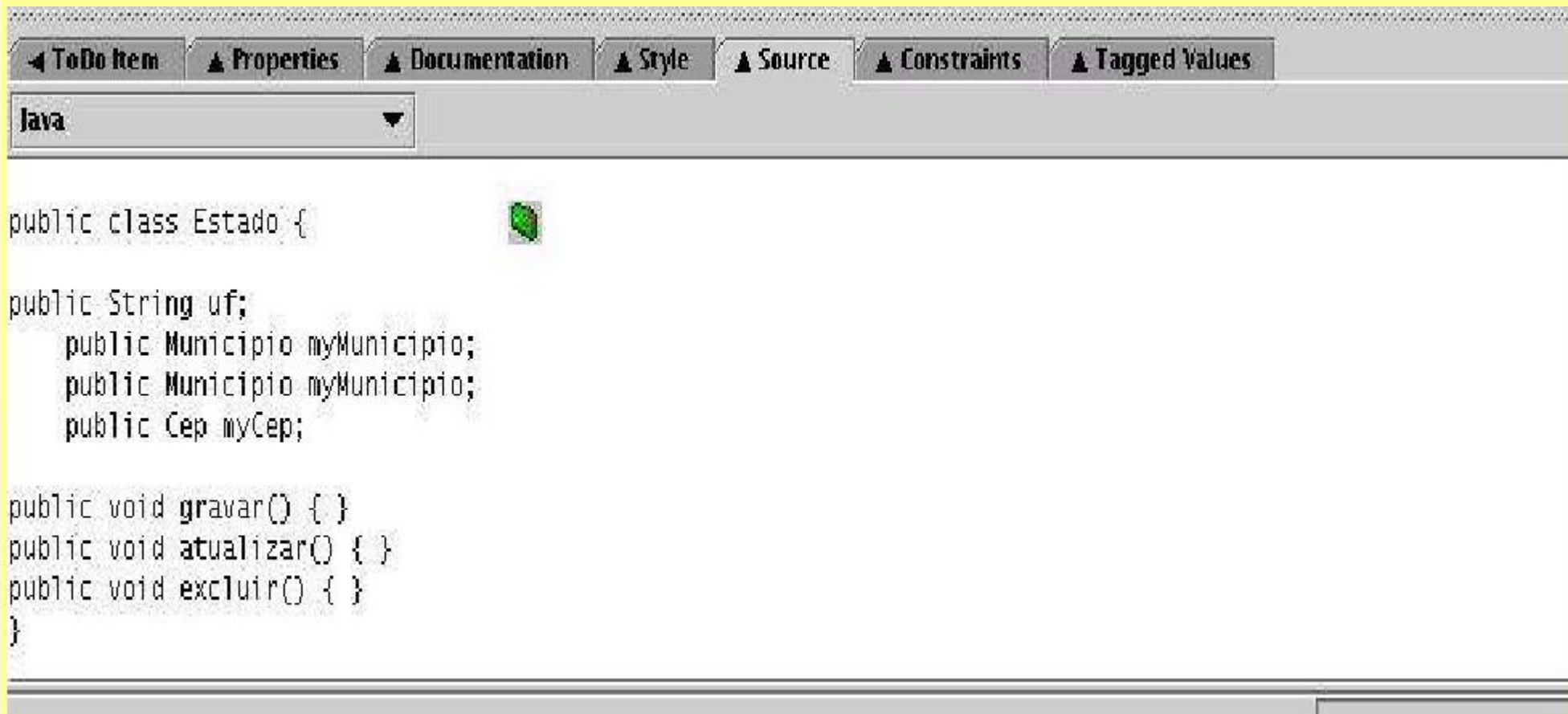
Estilo

- Propriedades visuais do elementos selecionados no projeto (cor , tamanho, formas gráficas)



Fontes

- Visualização do código fonte Java que será gerado.
- A ferramenta não permite que seja alterado o código gerado, apenas após sua exportação.



The screenshot shows a software interface with a tabbed menu at the top containing: 'ToDo Item', 'Properties', 'Documentation', 'Style', 'Source', 'Constraints', and 'Tagged Values'. The 'Source' tab is selected. Below the tabs is a dropdown menu showing 'Java'. The main area displays the following Java code:

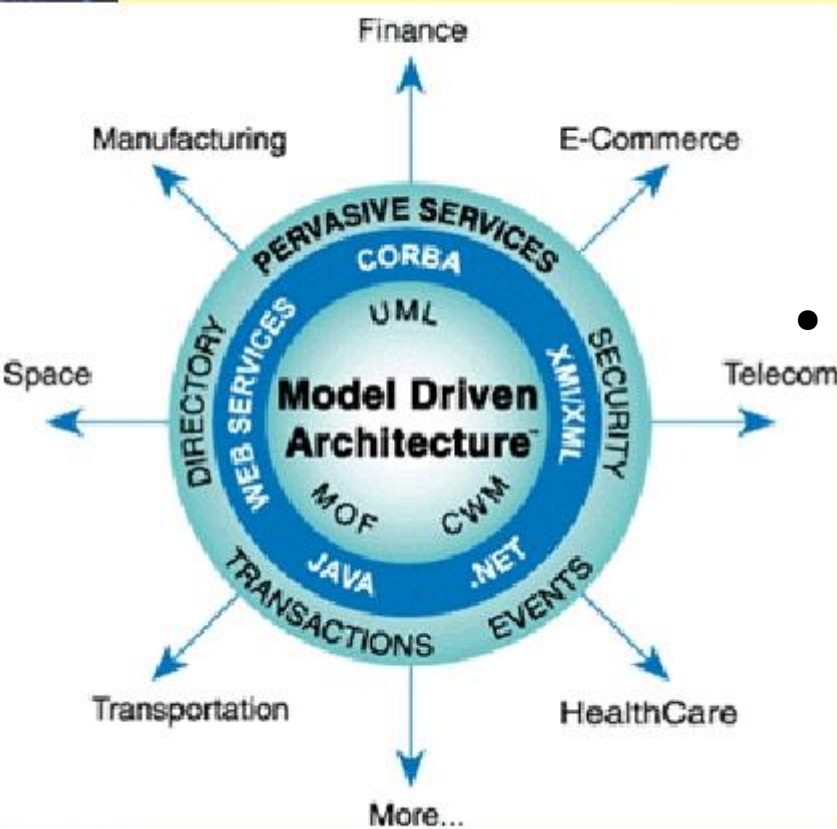
```
public class Estado {  
  
    public String uf;  
    public Municipio myMunicipio;  
    public Municipio myMunicipio;  
    public Cep myCep;  
  
    public void gravar() { }  
    public void atualizar() { }  
    public void excluir() { }  
}
```

ChaveValor (TaggedValues)

- Permite a entrada de uma chave no elemento e um valor correspondente , pode ser utilizado como histórico, ou discriminar dados úteis para o elemento.

Target: ActionState (Selecionar Município e UF do CEP)	
Tag	Value
homologado	NÃO
desenvolvido	SIM
disponível para instalação	NÃO

Integração com AndroMDA

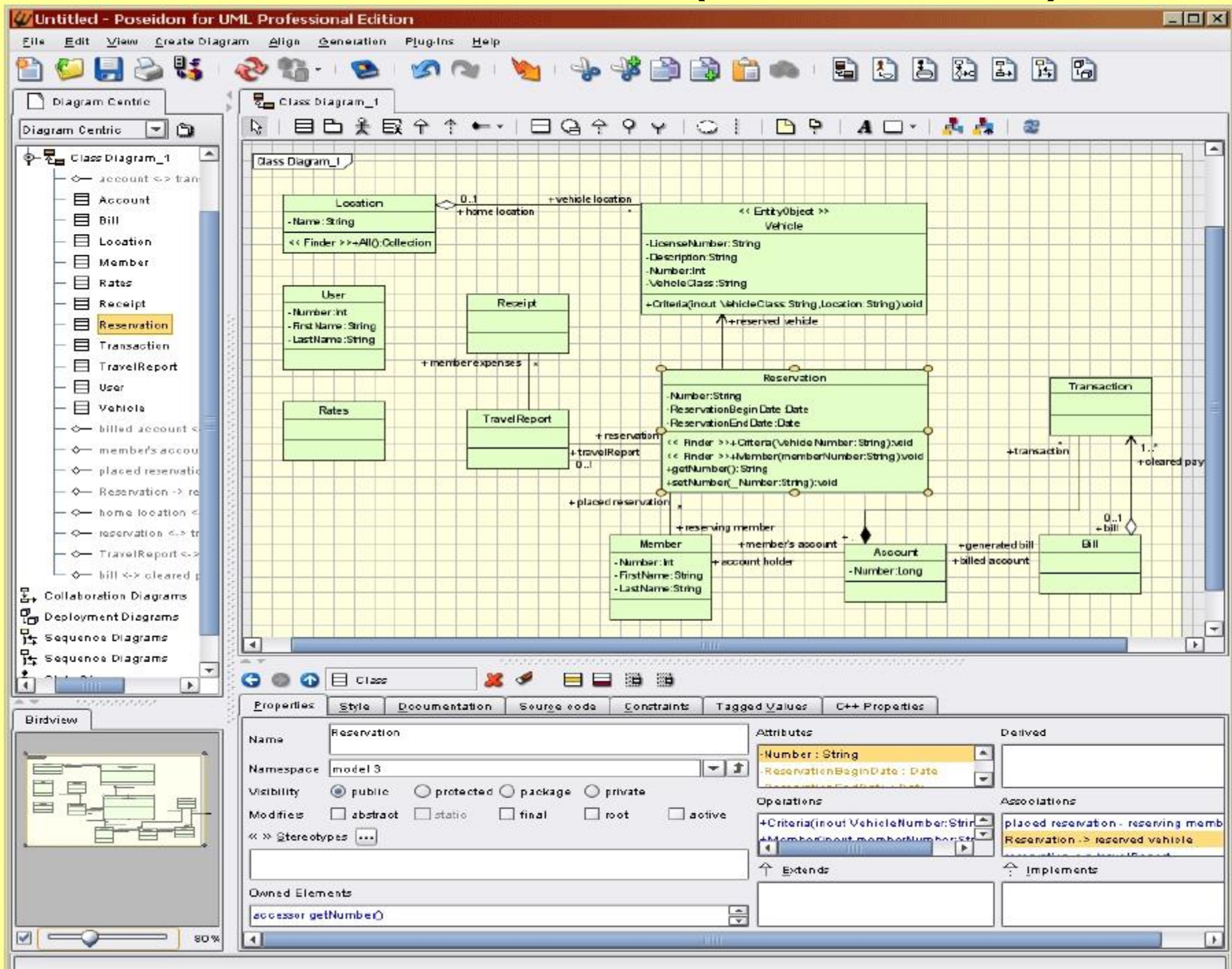


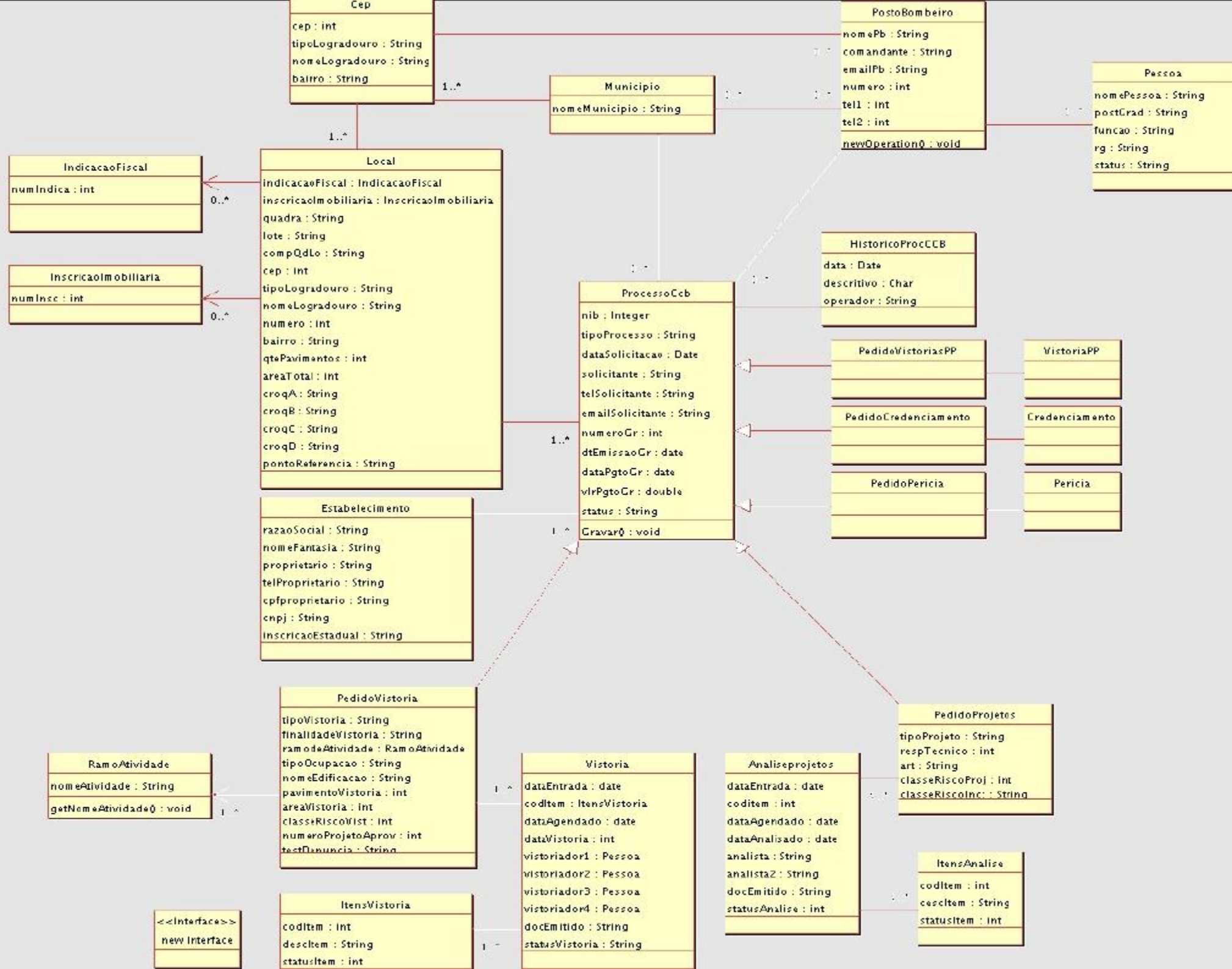
- O AndroMDA é um framework open source baseado em MDA (Model Driven Architecture). Ele utiliza modelos UML gerados por ferramentas CASE (padrão XML) e uma série de plugins, chamados de cartuchos (cartridges) para realizar a geração de componentes customizados, ou seja, o código-fonte do sistema.
- Os componentes podem ser gerados para as plataformas padrão de mercado: J2EE (Java), .net (C#), PHP, HTML, etc. Aos plugins já disponíveis somase a capacidade de customização do AndroMDA. Basta escrever um novo plugin (ou modificar um já existente) para que uma nova linguagem ou plataforma possa ser suportada.

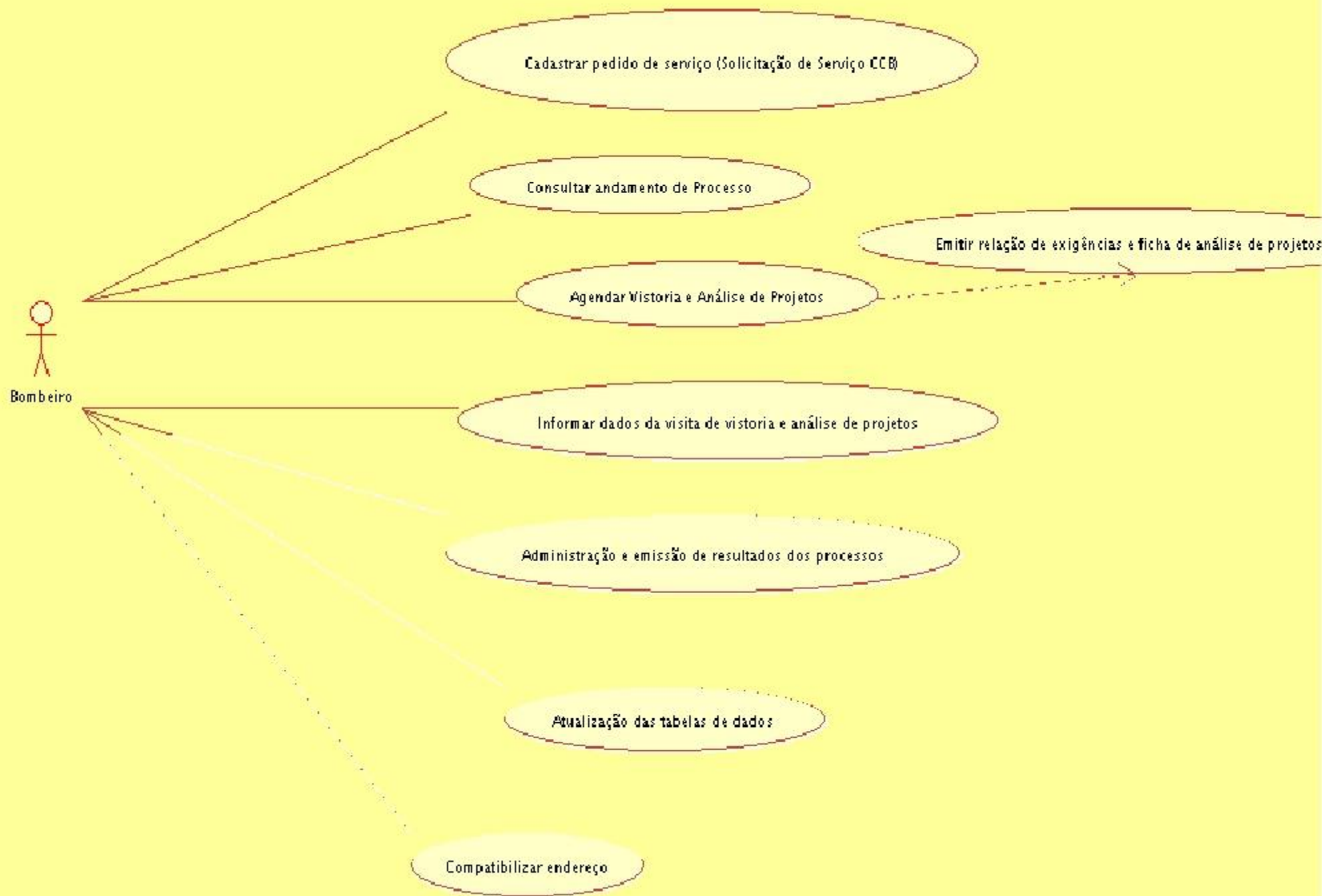
Informações adicionais

- Suporte e compatibilidade ao AndroMDA.
- Os comandos "Undo" e "Redo" (desfazer e refazer) ainda não foram implementados na versão atual (apenas reverter para arq. salvo).
- O início do projeto foi no ano de 1998 na Universidade da California, Berkeley
- Licença BSD Open Source.
- Mais de 1.000.000 downloads do site da Tigris.
- Software comercial baseado (Poseidon for UML)

Poseidon UML (comercial)

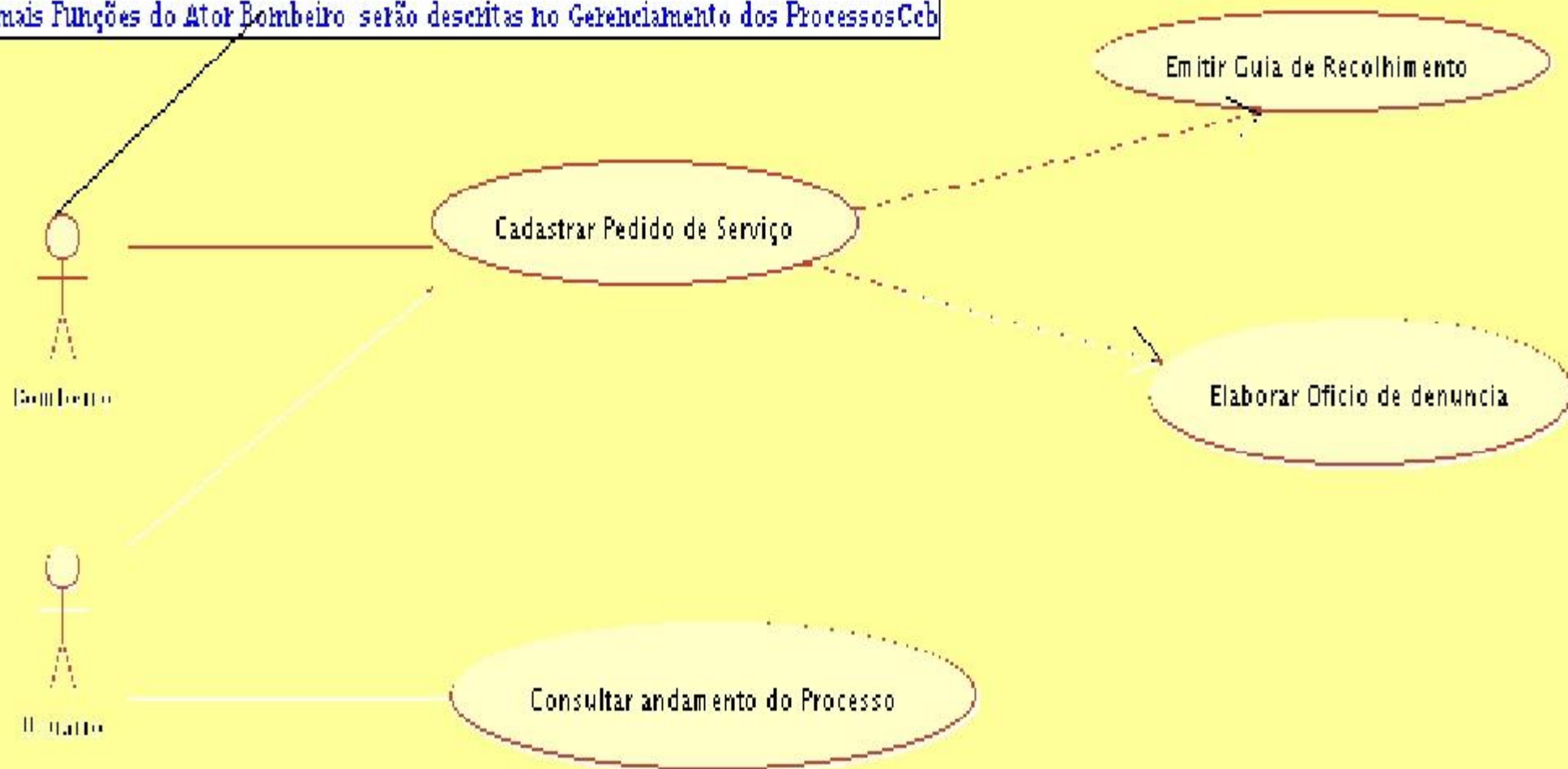


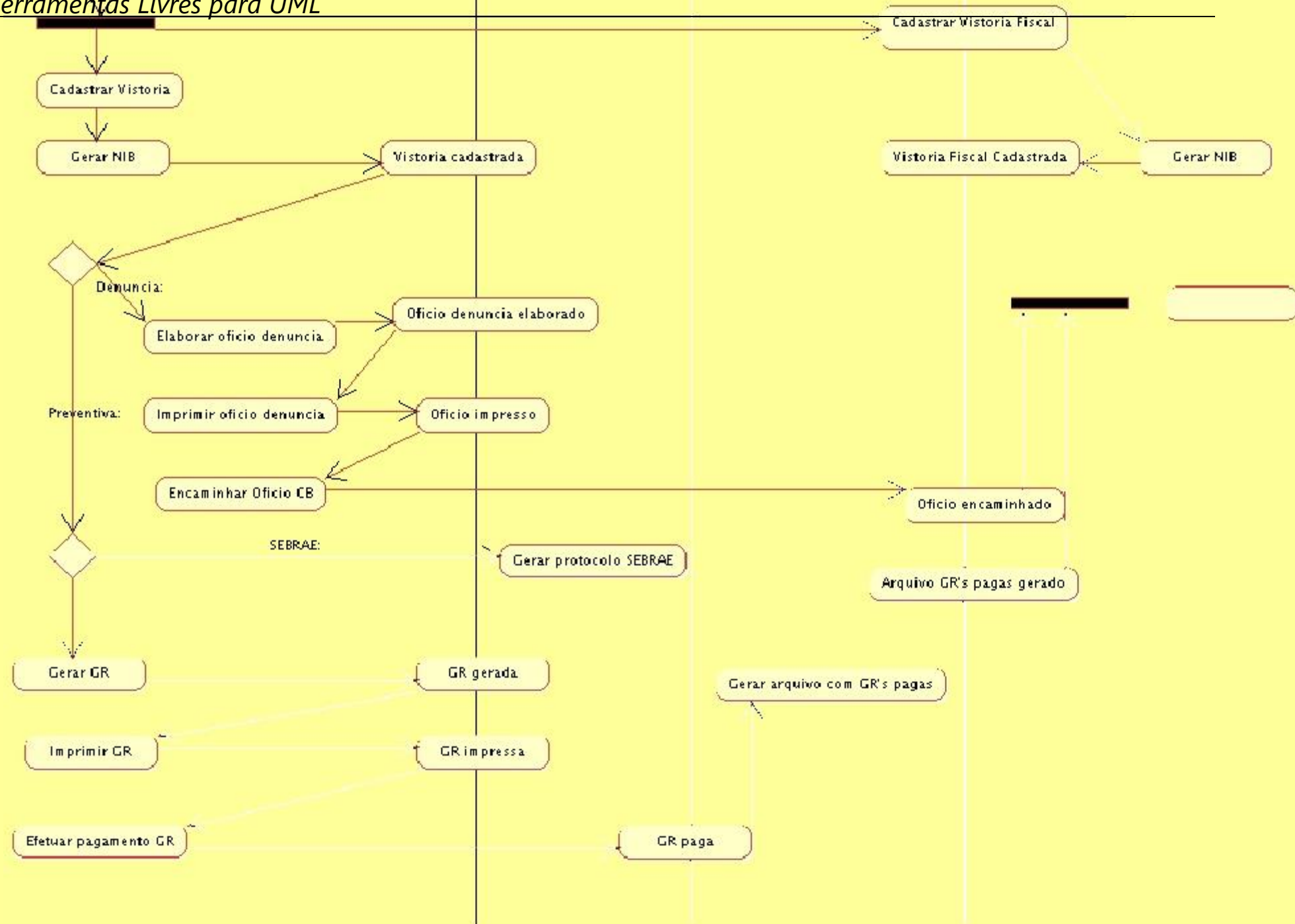


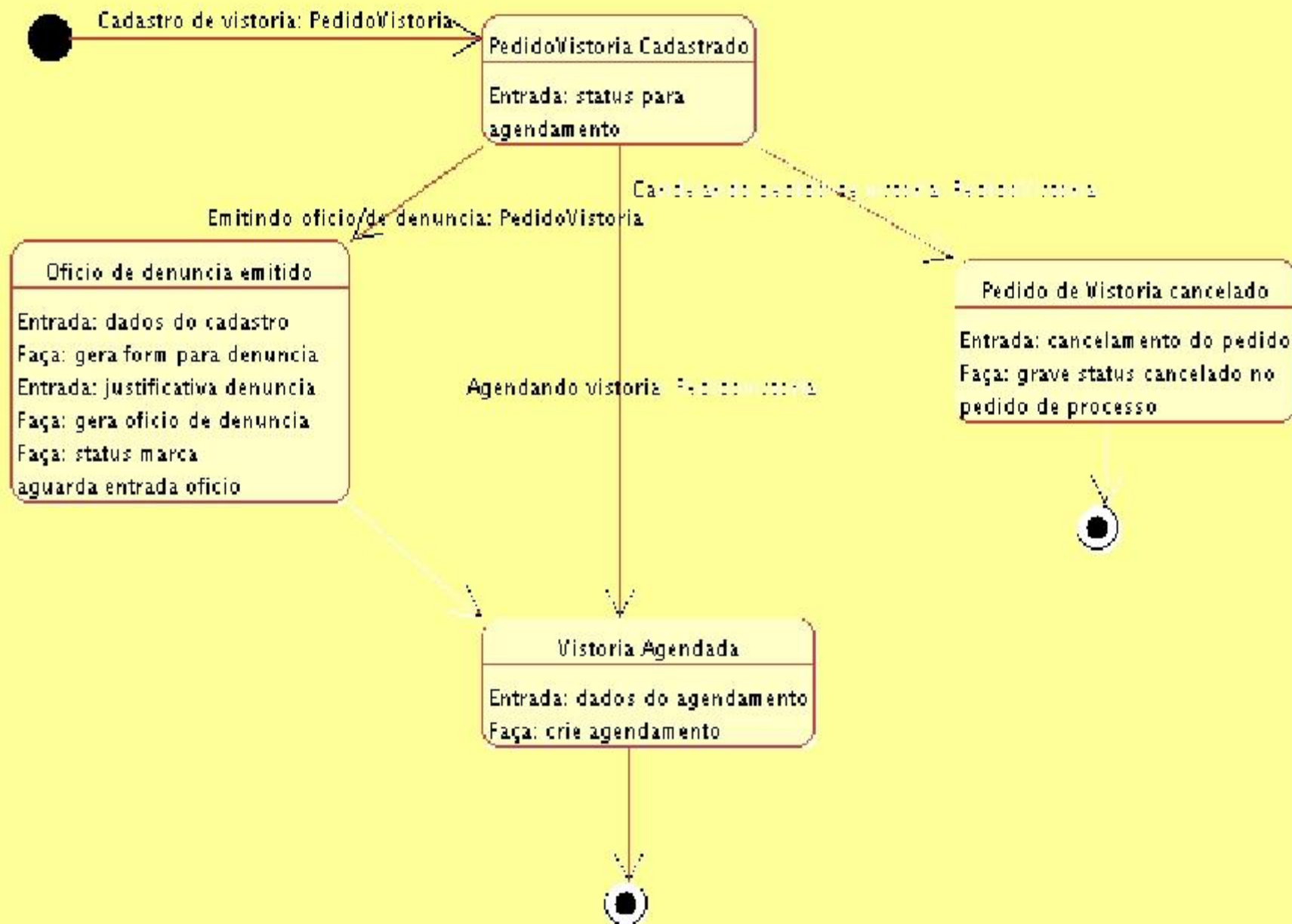


SOLICITAÇÃO DE SERVIÇO CCB

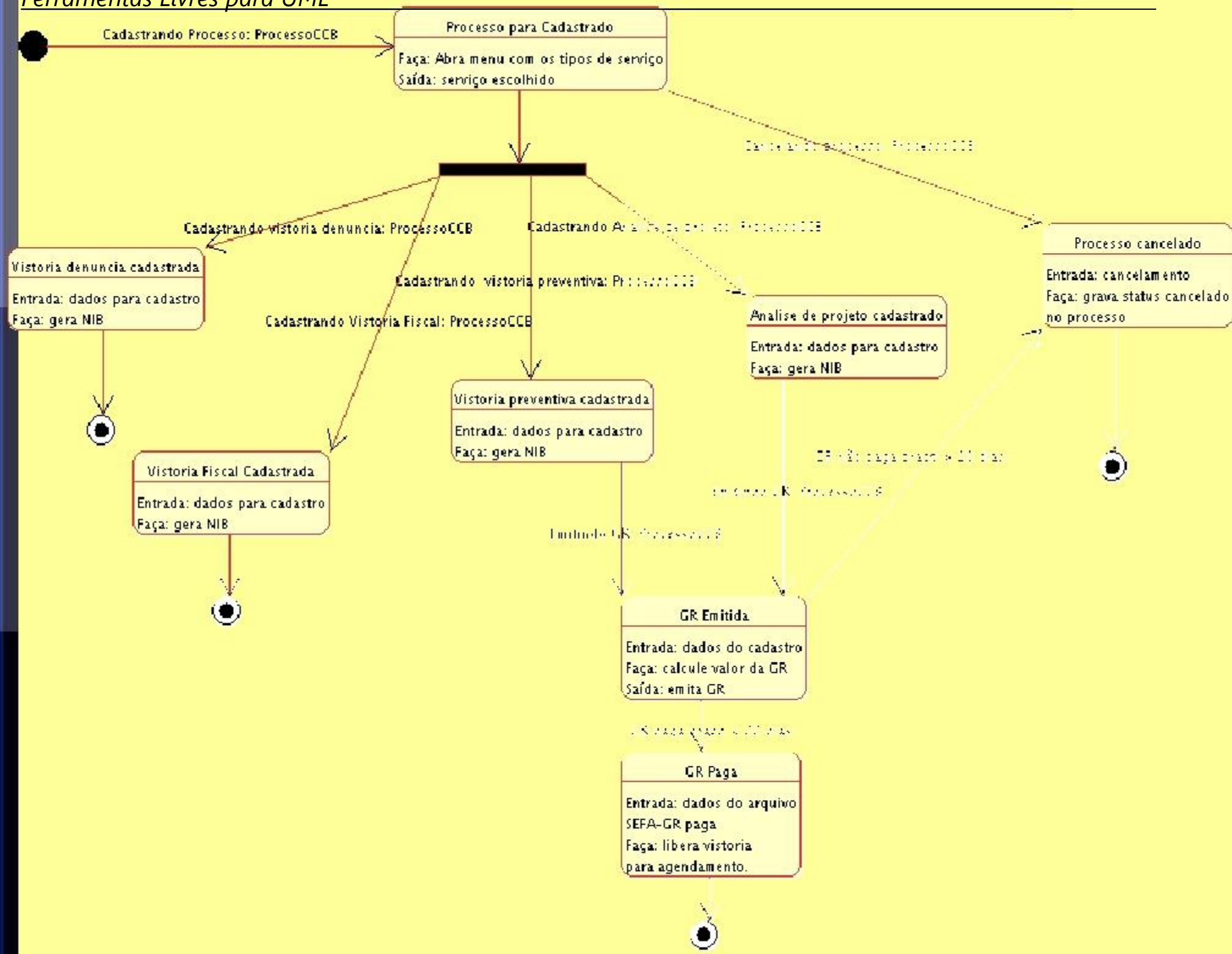
As Demais Funções do Ator Bombeiro serão descritas no Gerenciamento dos Processos Ccb







Ferramentas Livres para UML





Umbrello

Umbrello é um Software de Modelagem UML, que é integrado ao projeto KDE.

Este Software é utilizado para modelar o próprio projeto do KDE por a grande de seus desenvolvedores que utilizam UML.

Foi "portado" para plataforma Windows.



Plataformas Suportadas

- Gnu/Linux.
- FreeBSD
- Solaris
- Windows



Do que o Umbrello é Capaz

- Desenhar e imprimir diagramas UML.
- Gerar declarações de classes Java, PHP, javaScript, ActionScript C++ C# Pascal, SQL, Ada, IDL, XMLSchema, Python, Perl e Ruby.
- Gerar arquivos Gráficos (png).
- Engenharia Reversa de Classes .
- Arquivos no padrão XMI (baseado no formato XML)
- Wizard para Criação de Classes e geração de Código Fonte e configuração de cabeçalhos do Código de Fonte
- Visualizador de Código Fonte.
- Refatoração (refactoring).



Diagramas suportados

- Diagramas de classe.
- Diagrama de seqüência.
- Diagramas de Colaboração.
- Diagramas de caso de Uso.
- Diagrama de Estado.
- Diagrama de Atividade.
- Diagrama de componentes.
- Diagrama de Utilização.



Suporte para criação de diagramas.

- Multiplicidade.
- Atributos.
- Operações.
- Modelos.
- Associações.
- Generalização.
- Dependência.
- Composição.
- Agregação.
- Refatoração.

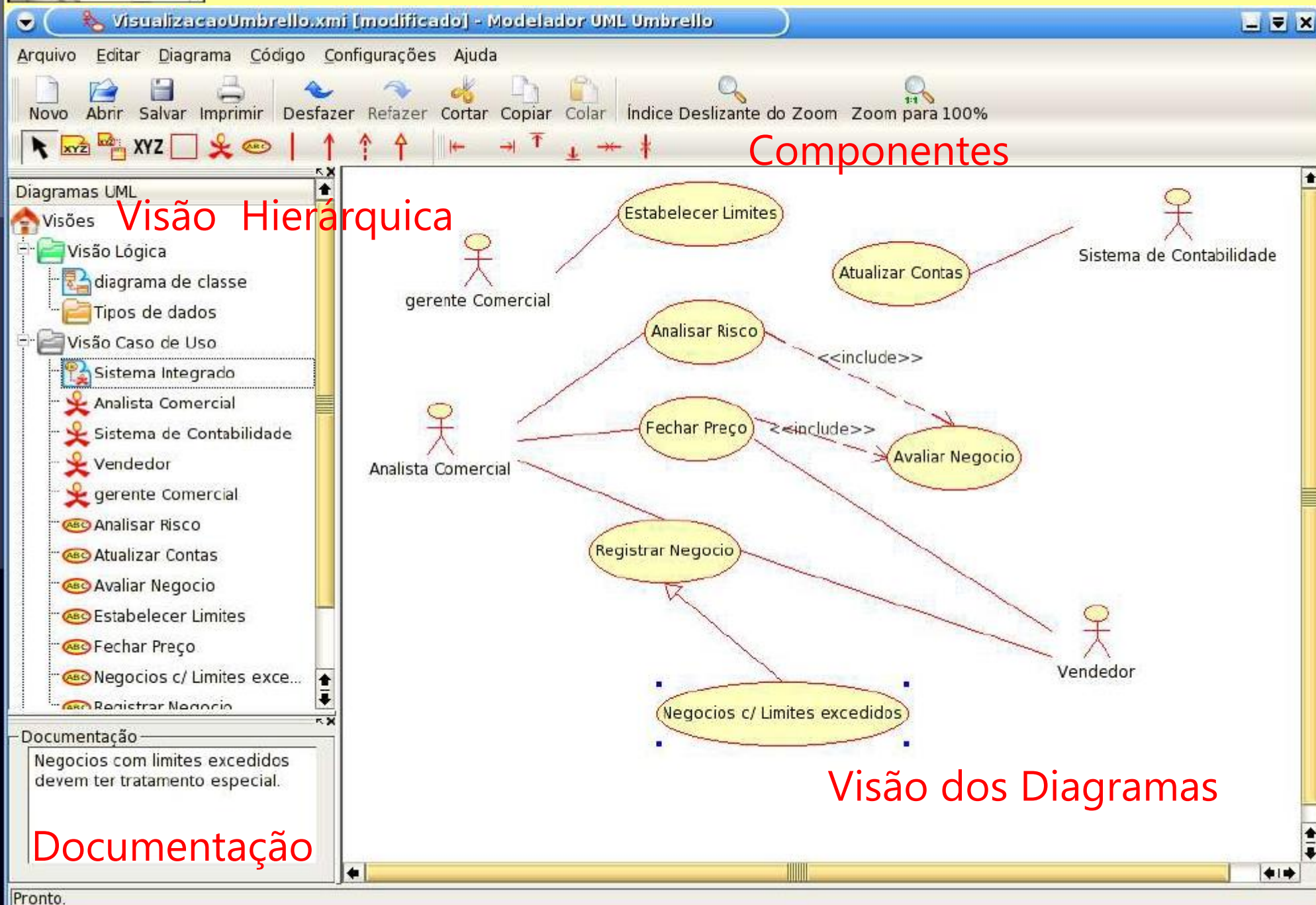


O Umbrello e sua utilização!

- O Umbrello atualmente é recomendado para pequenos projetos.
- Diversos desenvolvedores de Softwares que trabalham no projeto KDE utilizam o Umbrello para Modelagem UML.
- Seus desenvolvedores vêm trabalhando para que o software tenha um suporte de desenvolvimentos para médios e grandes projetos.
- Alguns Usuário :
- **Projeto Quanta** : [http:// quanta.kdewebdev.org/](http://quanta.kdewebdev.org/)
- **GnuMed** - [http:// www.gnumed.org/](http://www.gnumed.org/)
- **OMG** - [http:// www.omg.org/](http://www.omg.org/)
- **OpenCity** - [http:// www.opencity.info/](http://www.opencity.info/)
- Mais usuários em www.umbrello.org (users)



Umbrello UML Modeller





Visões dos diagramas UML.

Visão Lógica

Diagramas de Classes.

Diagramas de Colaboração.

Diagramas de Estado.

Diagramas de Atividades.

Diagramas de Seqüência.

Visão Caso de Uso.

Diagramas de Caso de Uso.

Visão do componente.

Diagramas de Componentes.

Visão de Distribuição.

Diagramas de Distribuição.





Propriedades Caso de Uso

- Nomes
- Visibilidade Obj.
- Documentação
- Cores
- Fontes





Refactoring

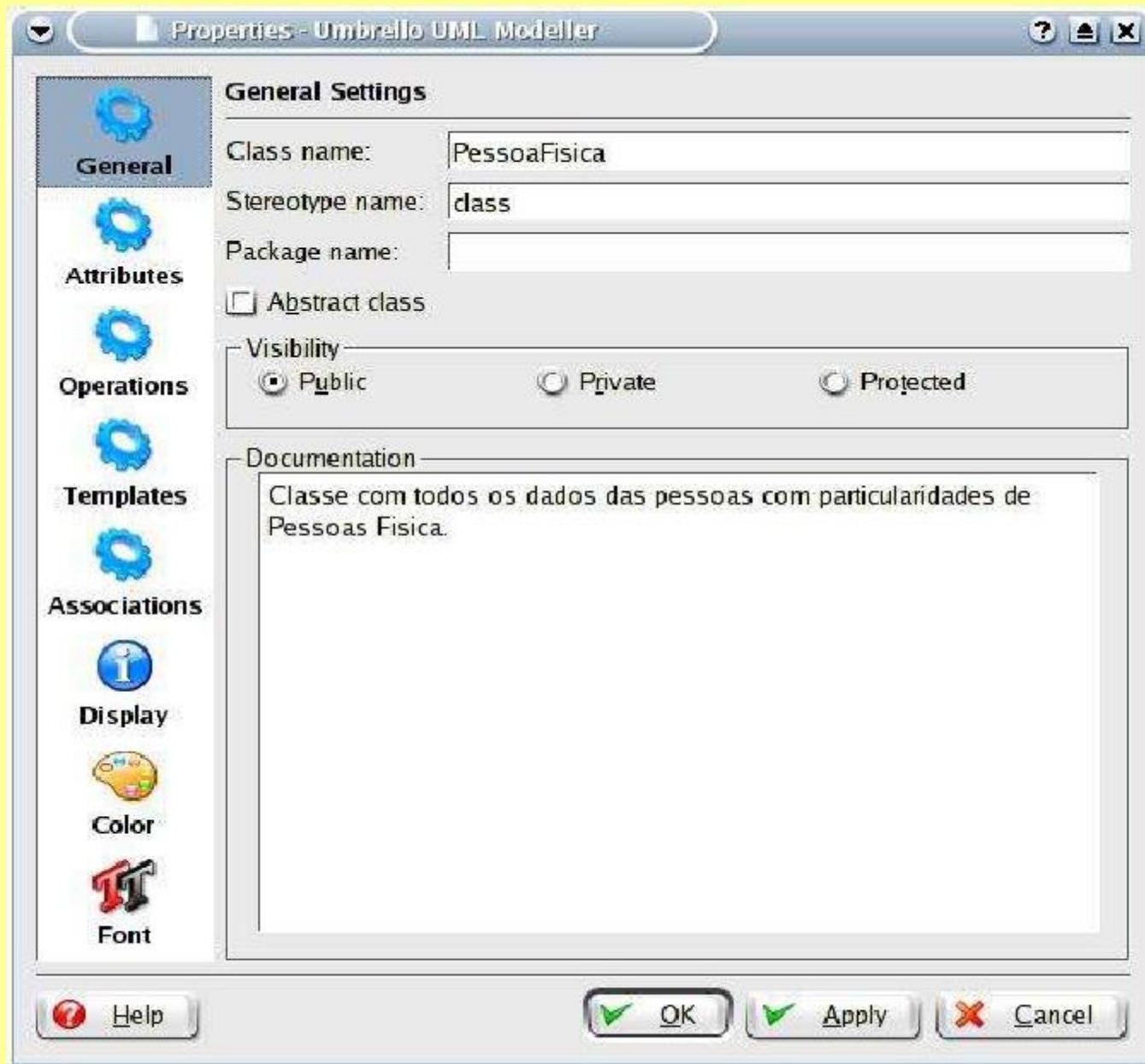


- Segundo Martin Fowler significa: “Mudança interna de estrutura de um software, que o torna mais fácil de se entender e mais barato de modificar, sem causar alterações em seu comportamento observável”.



Propriedades Classes

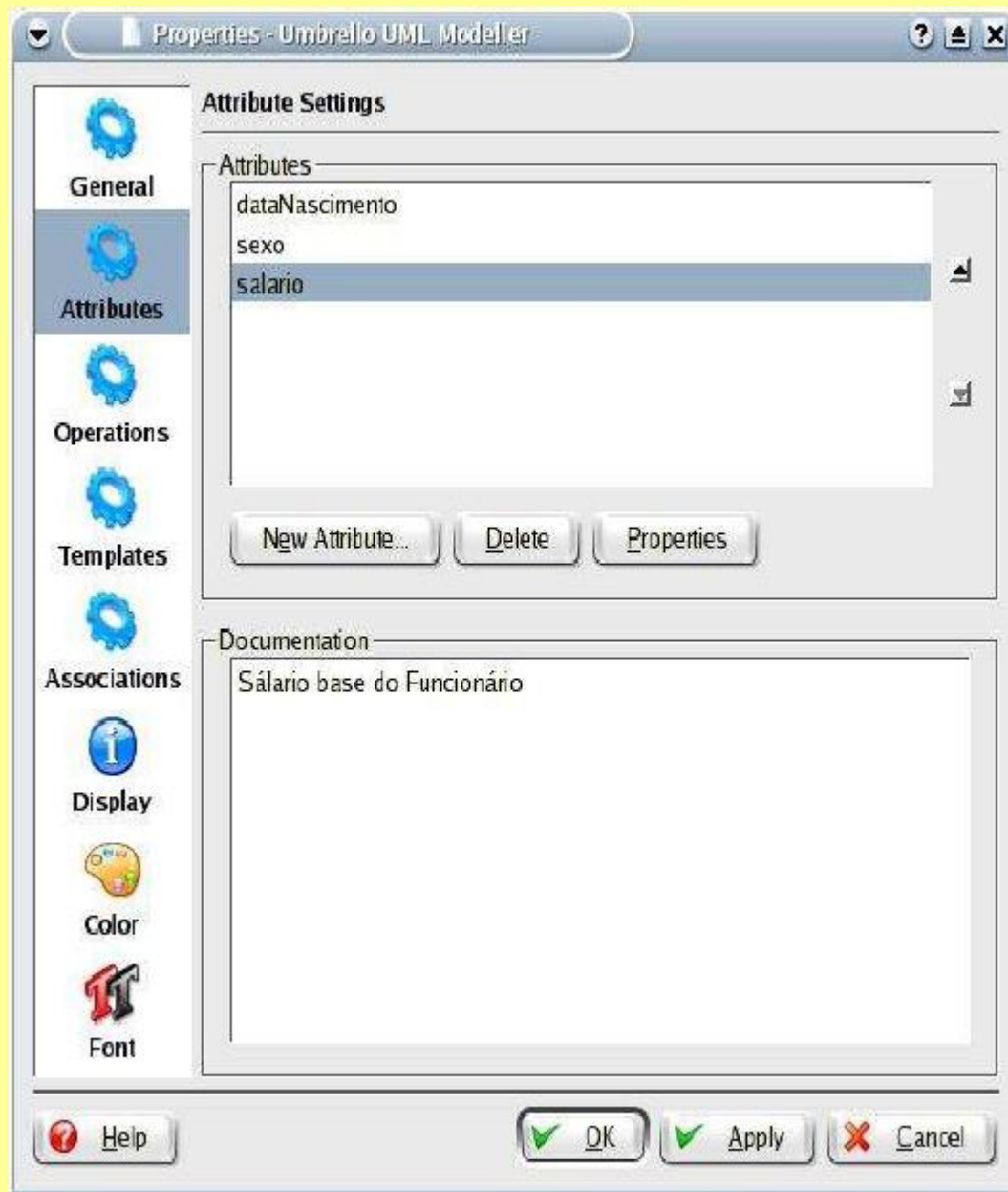
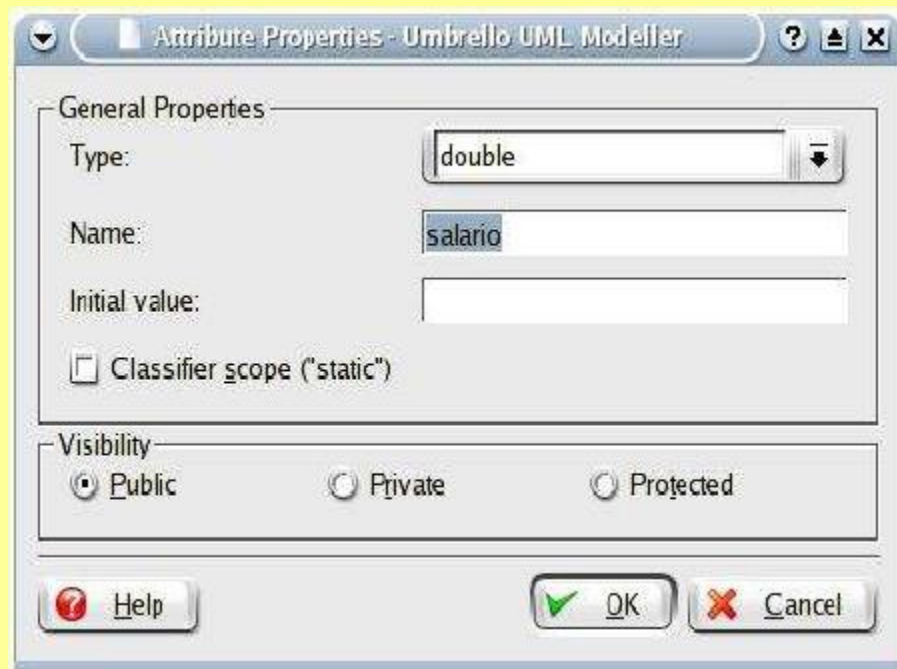
- Atributos
- Operações
- Associações
- Templates
- Cores
- Fontes





Propriedades Atributos de Classes

- Documentação
- Propriedades





Visualizador de Fontes

The screenshot shows the "Visualizador de Código" (Source Viewer) window in Umbrello. The window title is "Projeto-LightCMS.xml [modificado] - Modelador UML Umbrello". The window has a toolbar with "Mostrar tipo de bloco" (Show block type) and "Mostrar blocos ocultos" (Show hidden blocks) checkboxes, both of which are checked. There is also a "Fechar" (Close) button. The main area displays the source code for "Module.java". The code is color-coded: imports are pink, comments are yellow, and code blocks are grey. The code defines a "Module" class with a private integer field "codeModule", a constructor, and two methods: "getCodeModule()" and "setCodeModule()".

```
Module.java

import DataBase;
import int;
// Class Module
//
public class Module {
    // Fields
    //
    private int codeModule;
    // Methods
    // Constructors
    // Empty Constructor
    public Module ( ) { }
    // Accessor Methods
    /**
     * Get the value of codeModule
     *
     * @return the value of codeModule
     */
    private int getCodeModule ( ) {
        return codeModule;
    }
    /**
     * Set the value of codeModule
     *
     *
     */
    private void setCodeModule ( int value ) {
        codeModule = value;
    }
    // Operations
}
```

Código fonte Java gerado e Headings

/*****

Pedido.java - Copyleft marcio

GNU Public License Aqui você pode escrever a licença para seu código, comentários e outras informações para isso basta configurar os "headings"

os arquivos estão alocados em /usr/lib/kde3/share/apps/umbrello/headings/heading.java

Este arquivo foi gerado : Sáb Jan 29 2005 as 13:29:43

*****/

```
import java.util.Date;
```

```
import java.lang.String;
```

```
import Boolean;
```

```
public class Pedido {
```

```
    // Fields
```

```
    private int id_pedido;
```

```
    private java.util.Date dataPedido;
```

```
    private java.lang.String observacoes;
```

```
    // Methods
```

```
    // Constructors
```

```
    // Accessor Methods
```

```
    /**
```

```
     * Get the value of id_pedido
```

```
     *
```

```
     * @return the value of id_pedido
```

```
     */
```

```
    private int getId_pedido ( ) {
```

```
        return id_pedido;
```

```
    }
```

```
    // Operations
```

```
    public void encerrarPedido ( ) {
```

```
    }
```

```
}
```




SQL gerado a partir de uma classe

pedido Copyleft marcio

GNU Public License Aqui você pode escrever a licença para seu código, comentários e outras

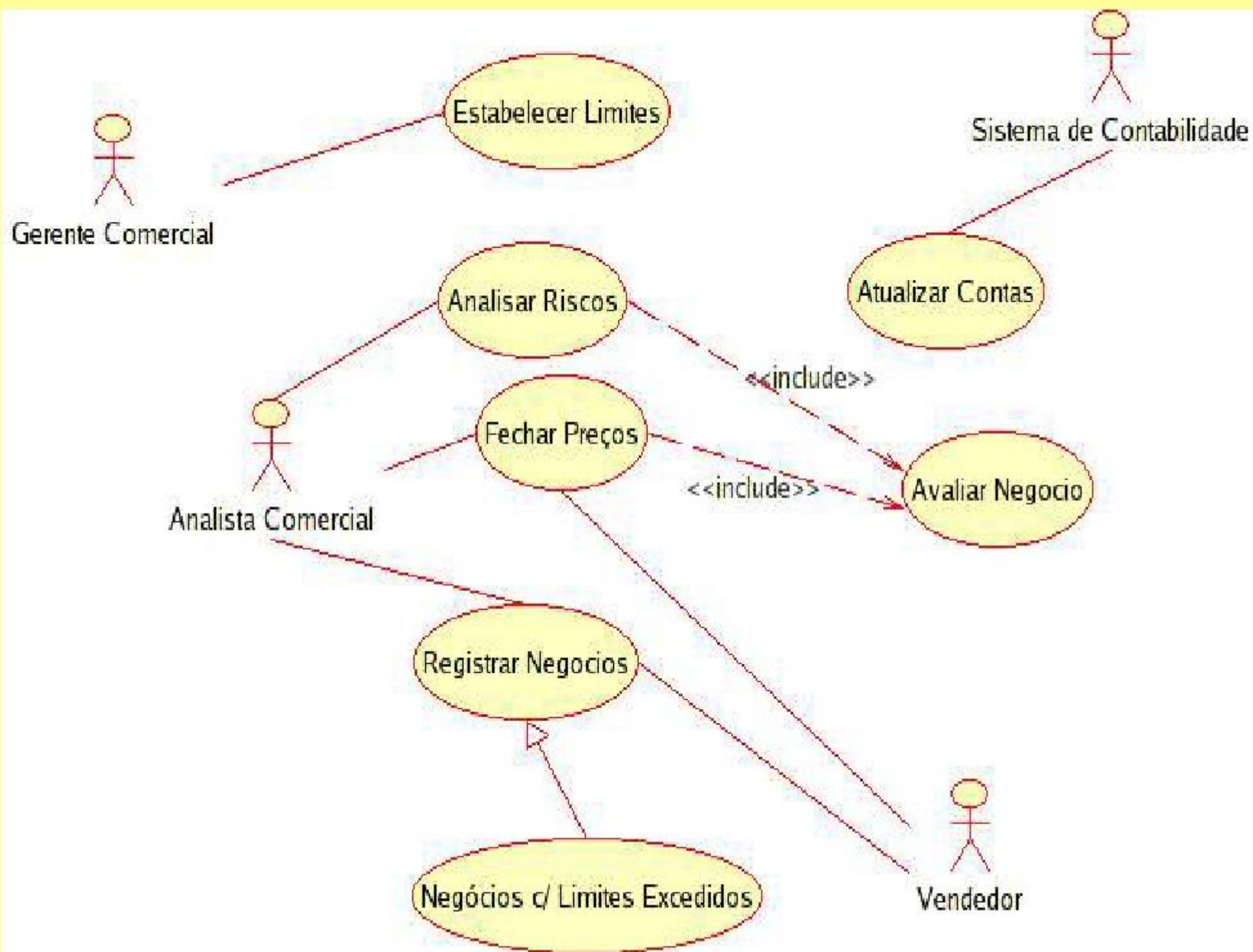
informações para isso basta configurar os "headings"

os arquivos estão alocados em /usr/lib/kde3/share/apps/umbrello/headings/heading.sql

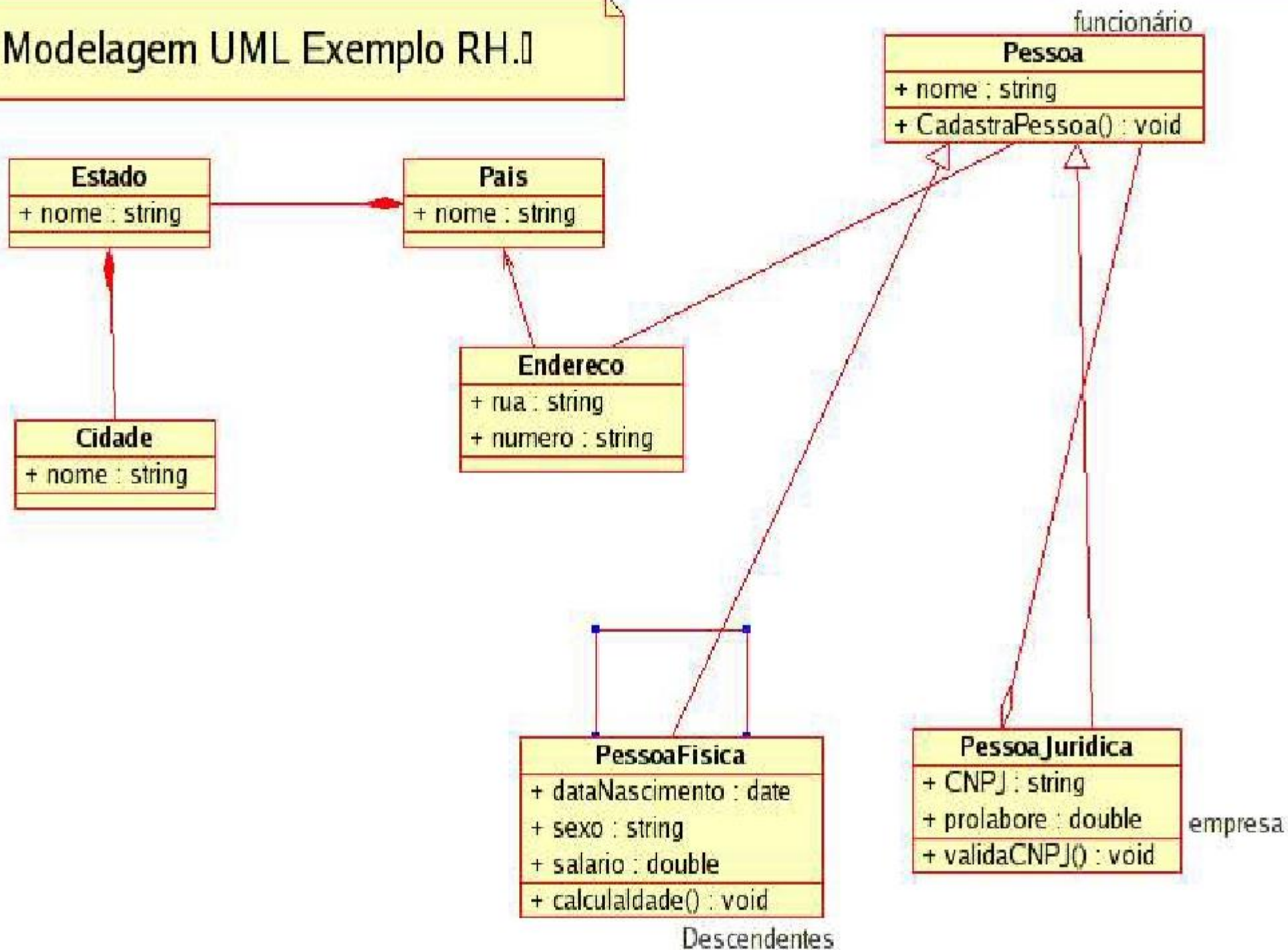
Este arquivo foi gerado : Sáb Jan 29 2005 as 13:33:43

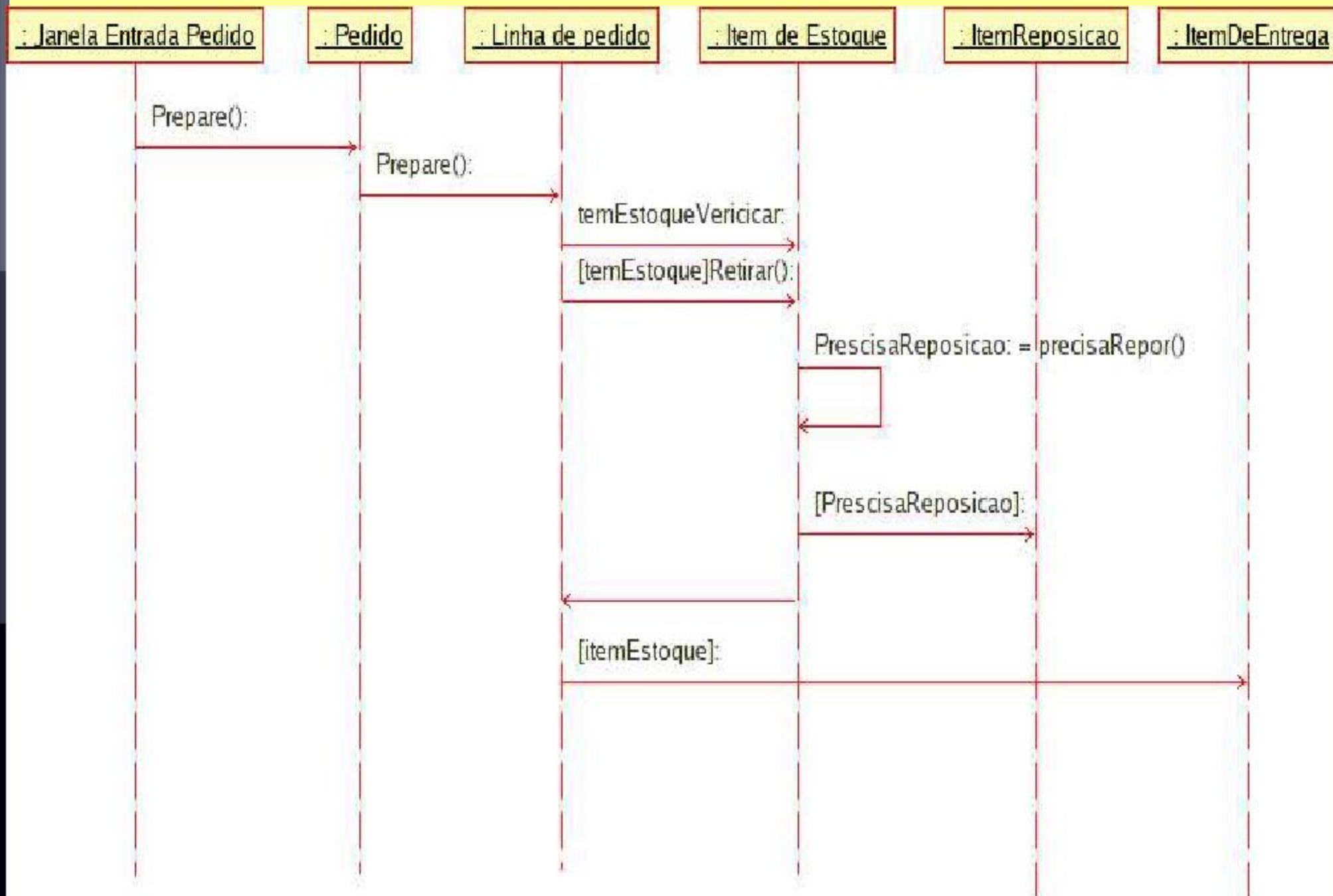
TABLE: Pedido

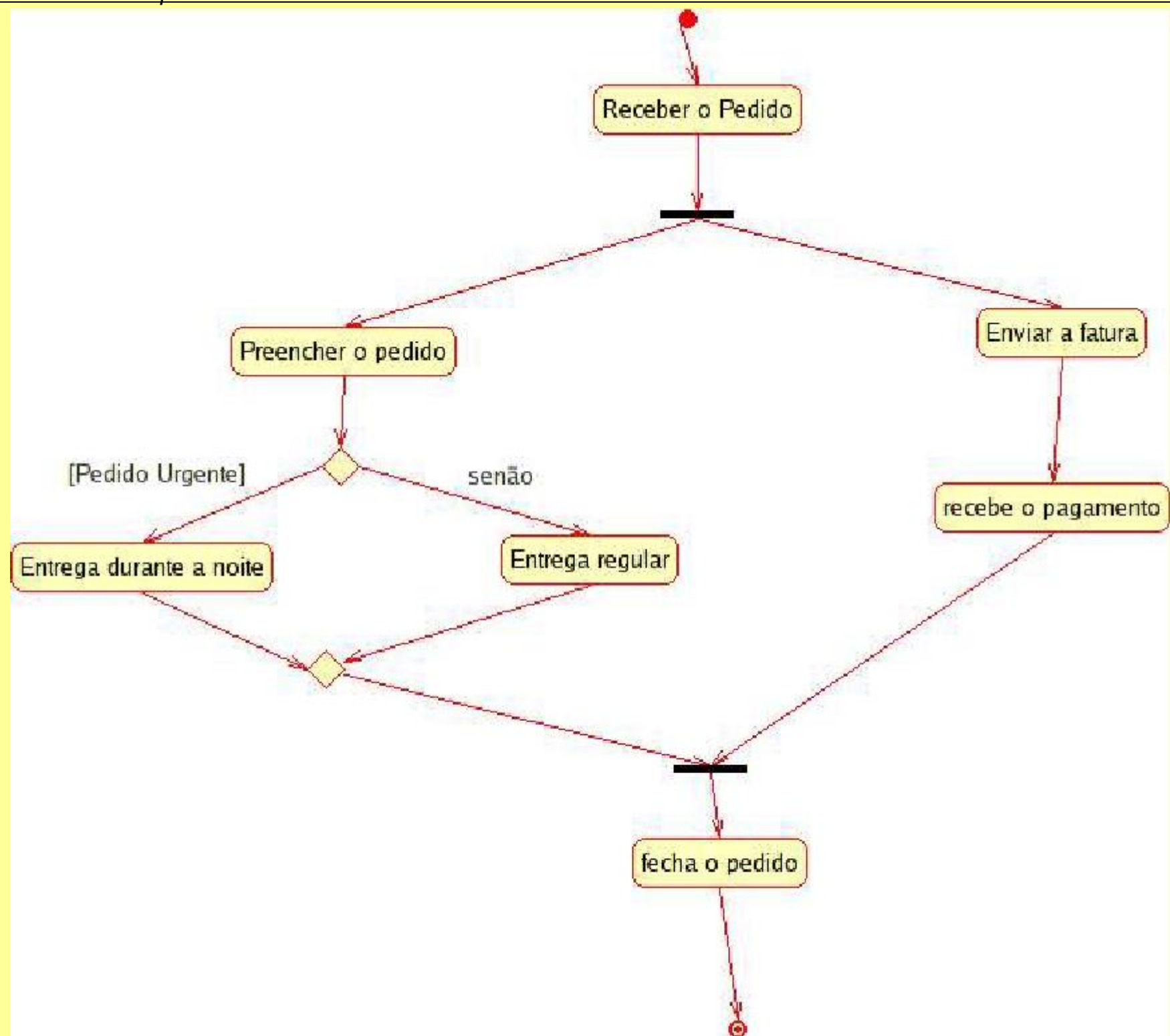
```
CREATE TABLE Pedido (  
    id_pedido int ,  
    dataPedido Date ,  
    vendedor Char(50) ,  
    status Char(1) ,  
    observacoes Text  
);
```

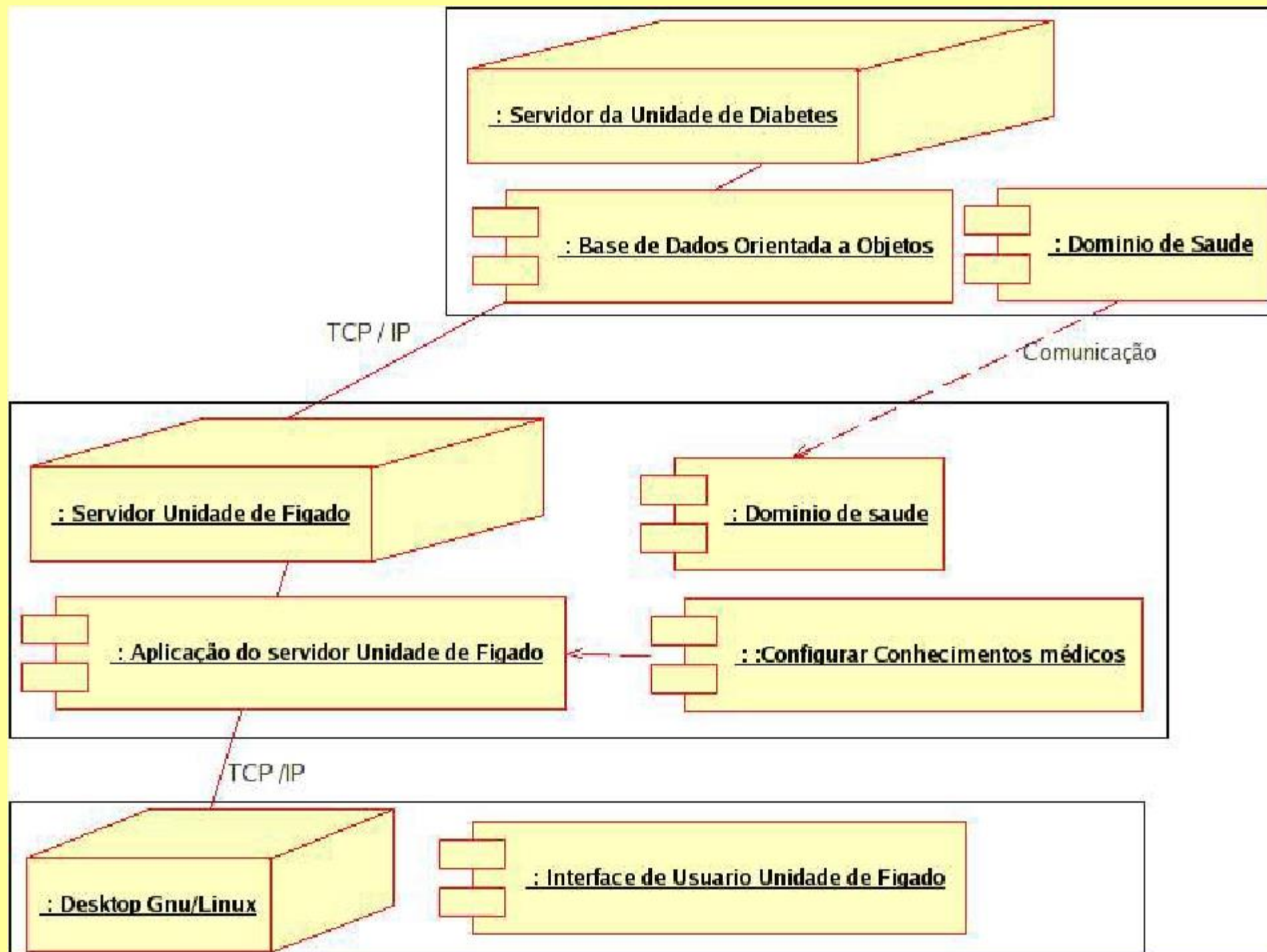


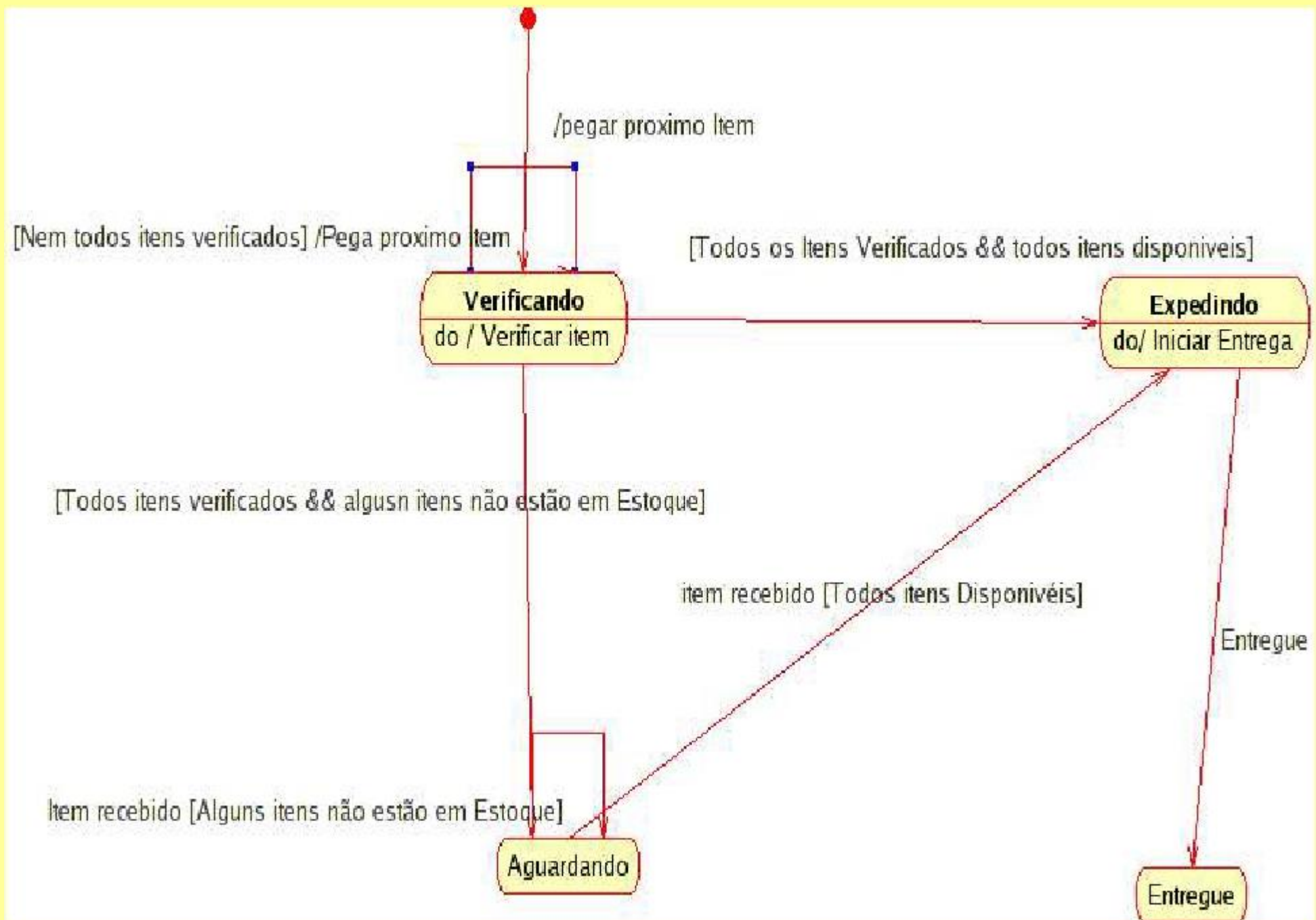
Modelagem UML Exemplo RH.▯

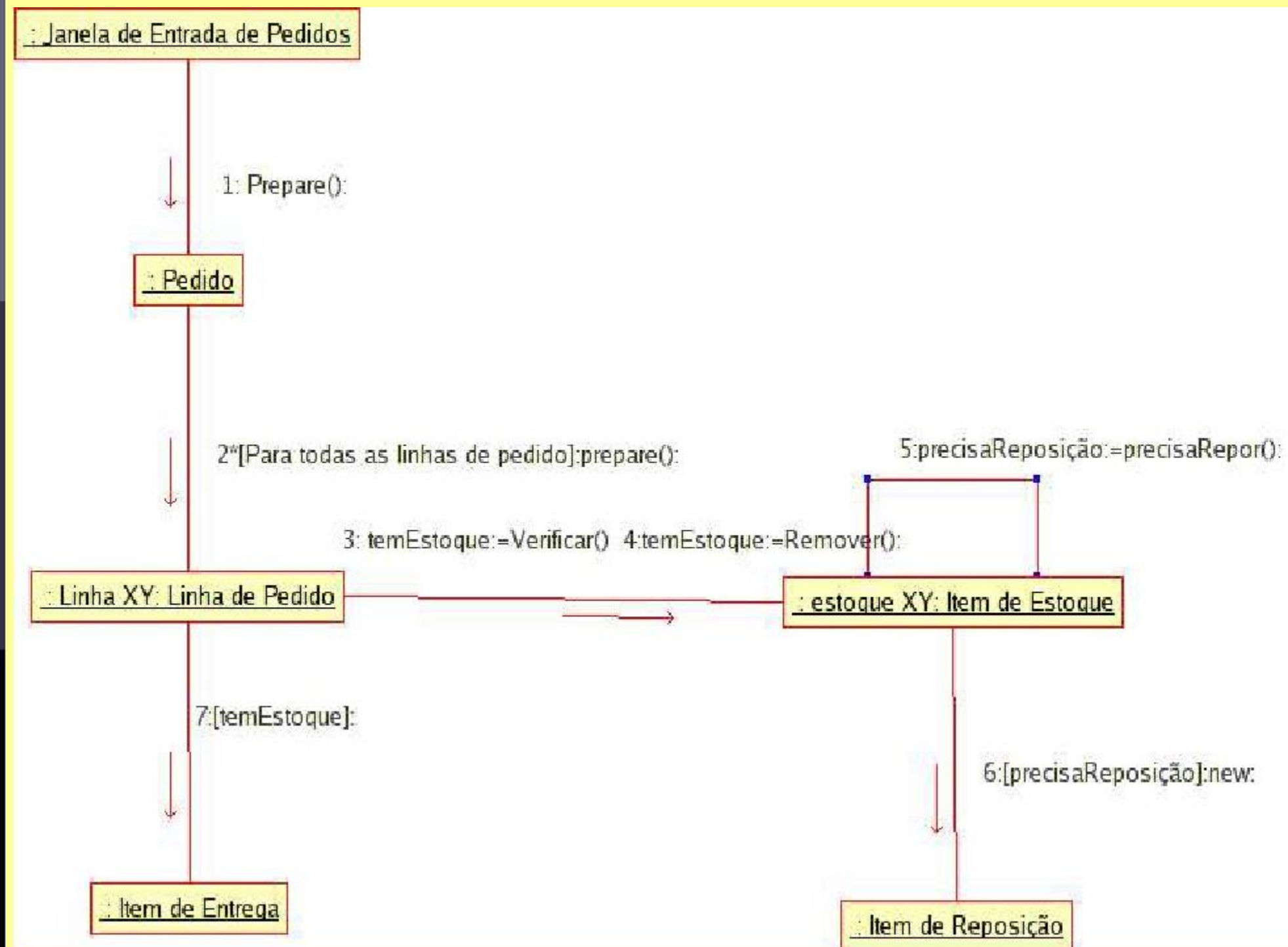


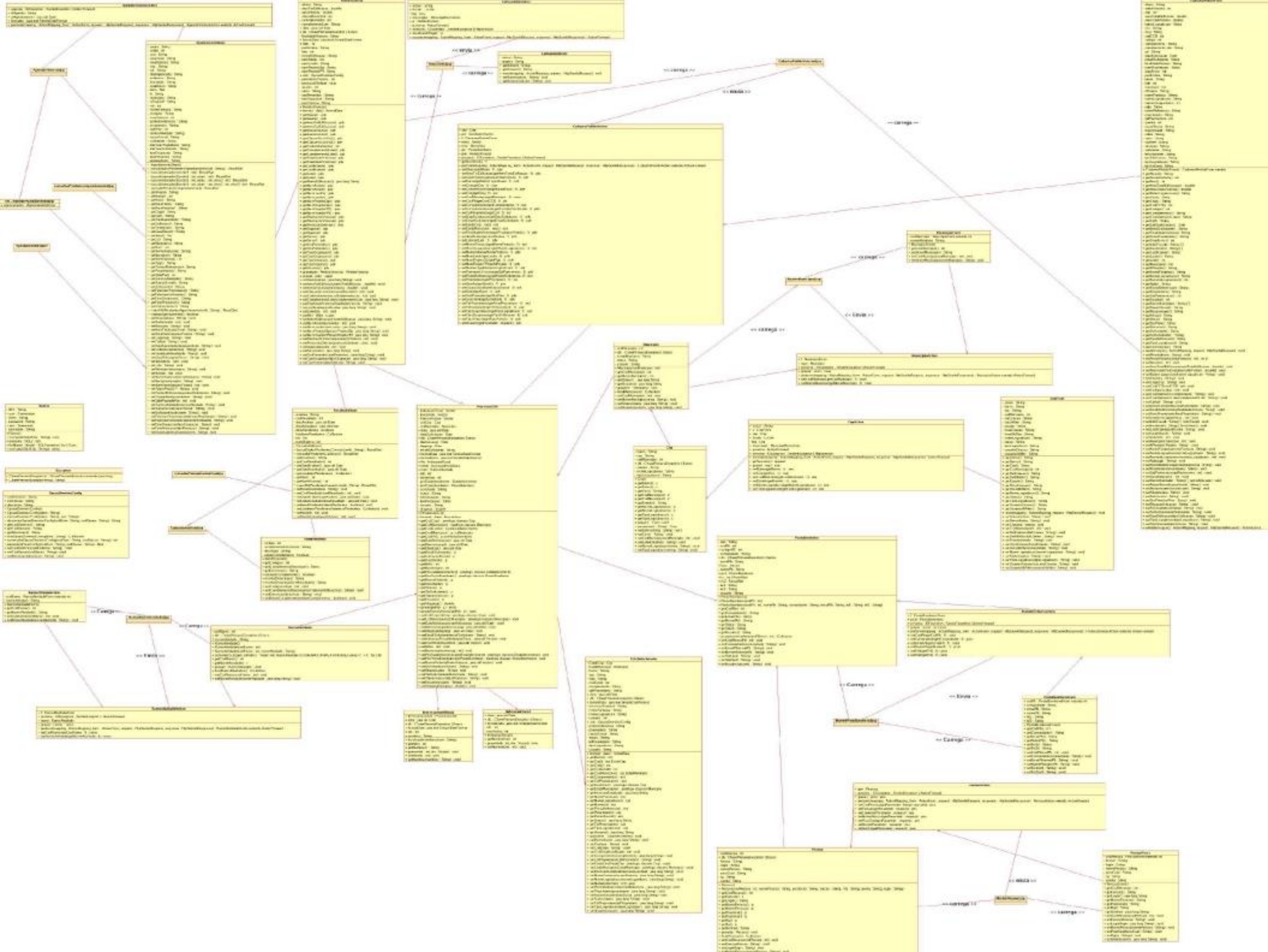












Umbrello UML Modeller 1.5

Umbrello UML Modeller 1.5 interface showing a UML class diagram.

The interface includes a menu bar (File, Edit, Diagram, Code, Settings, Help), a toolbar with various UML diagram tools, and a left sidebar with a "UML Diagrams" tree and a "Documentation" pane.

The main diagram area displays a UML class diagram with the following classes and relationships:

- installation_report**: Attributes: - installer : programmer, - d : date, - c : contact man. Methods: + install([c : order] : installation_report), + close_malfunction([m : malfunction] : problem_report).
- head_programmer**: Methods: + separate(), + setLeadProgrammer().
- programmer**: Methods: + install([c : order] : installation_report), + close_malfunction([m : malfunction] : problem_report).
- problem_report**: Attributes: - start_fix : date, - end_fix : date, - release : int, - description : string, - p : programmer.
- SWH_manager**: Methods: + setLevelTeam(), + setHeadProgrammer(), + setMarketingTeam().
- product**: Attributes: - release : int, - headProgrammer : lead_programmer, - productManager : product_manager.
- product_manager**: Methods: + sell().
- marketing_staff**: Methods: + sell().
- release**: Attributes: - start_reselling, - end_reselling.
- order**: Attributes: - orderStart : date, - guaranteeTime : date, - c : cellular_company, - guarantee_start : date, - contact : contact_man.
- malfunction**: Attributes: - description : string, - open : date.
- cellular_company**: Attributes: - name : string, - address : string.
- contact_man**: Attributes: - c : cellular_company, - name : string, - address : string, - job : int, - problem().
- software_entity**: Attributes: - difficulty : int, - price : int, - release : int, - function : int, - complexity : int, - leadProgrammer : lead_programmer, - setEndReselling().
- function**: Methods: + setEndReselling().

The diagram shows various associations and generalizations between these classes, including relationships like "separate" between head_programmer and programmer, "install" between programmer and installation_report, and "sell" between product_manager and marketing_staff.

Please send contributions of screenshots to jr@jridgell.org.

Referência

- **Web Sites**

- [http:// www.umbrello.org](http://www.umbrello.org)
- [http:// codigolivre.org.br/ projects/ dia2sql/](http://codigolivre.org.br/projects/dia2sql/)
- [http:// argouml.tigris.org](http://argouml.tigris.org)
- [http:// www.gnome.org/ projects/ dia/](http://www.gnome.org/projects/dia/)
- [http:// www.omg.com.br](http://www.omg.com.br)
- [http:// www.andromda.com.br](http://www.andromda.com.br)
- [http:// www.andromda.org](http://www.andromda.org)
- [http:// www.agata.org.br/](http://www.agata.org.br/)

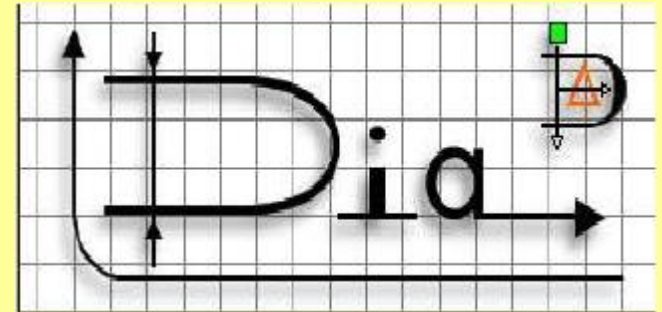
- **Revistas e Livros**

- SQL Magazine Edições 19,22,26 (Marcio Junior Vieira)
- Fowler, Martin
UML essencial: Um breve guia para a linguagem- padrão de modelagem de objetos/ Martin Fowler e Kendal Scott- 2.ed. - Porto Alegre:Bookman, 2000.
- Fowler, Martin
Refatoração: Aperfeiçoando o projeto de código existente.
Bookman, 2004.
- Lozano, Fernando
JAVA em GNU/ Linux – Rio de Janeiro: Alta Books, 2002

Links

Dia

- www.gnome.org/projects/dia/ Projeto Oficial



Umbrello

- www.umbrello.org Projeto Oficial.



ArgoUML

- www.argouml.org Projeto Oficial.
- www.tigris.org Tigris open source.



UML

- www.uml.org Unified Modeling Language.
- www.omg.org Object management Group.

