

**Guia para Estruturação de Modelagem  
Metodologia Celepar**

**Agosto 2009**

### Sumário de Informações do Documento

<b>Documento: guiaEstruturacaoModelagem.odt</b>		<b>Número de páginas: 17</b>	
<b>Versão</b>	<b>Data</b>	<b>Mudanças</b>	<b>Autor</b>
1.0	11/03/2008	Criação.	Marcos Chiarello
1.0	3/06/2009	Complementação	Danielle Mayer
1.0	19/08/2009	Revisão	Marcos, Danielle
1.1	14/01/2010	Revisão – Complementando o tópico Compartilhamento	Danielle Mayer

## Sumário

1	Introdução.....	4
1.1	Visão Geral .....	4
2	organização da modelagem.....	4
2.1	Modelo de Caso de Uso.....	4
2.2	Classes.....	5
2.2.1	Domínio.....	5
2.2.2	Serviço.....	5
2.2.3	Persistência.....	5
2.2.4	Controle.....	5
2.2.5	Telas.....	6
2.3	Diagrama de Seqüência.....	6
2.4	Ilustração.....	6
3	Compartilhamento.....	8
3.1	Configuração da localização do repositório de versões.....	8
3.2	Compartilhando um projeto no controle de versões – CVS.....	8
3.3	Recuperando um projeto existente no controle de versões – CVS.....	10
4	trabalhado em equipe.....	12
4.1	Comparando versões de um modelo:.....	13
4.2	Fluxo clássico para o desenvolvimento paralelo.....	14
4.3	Identificando conflitos.....	14
4.4	Sincronizando artefatos.....	15
4.5	Resolvendo os conflitos e diferenças.....	16
4.6	Resolvendo os conflitos específicos.....	17

# 1 INTRODUÇÃO

Este guia tem por objetivo de orientar a atuação do Analista de Sistemas na forma de estruturação e compartilhamento(trabalho em equipe) da modelagem dos projetos de software utilizando ferramenta homologada pela CELEPAR.

## 1.1 Visão Geral

Durante o desenvolvimento de um projeto de software, diversos modelos em notação UML(Unified Modeling Language) devem ser criados, conforme prescrito pela Metodologia de Desenvolvimento vigente, e para tanto é necessário o uso de uma ferramenta de modelagem. A ferramenta homologada pela CELEPAR é o Rational Software Modeler (RSM).

Os modelos prescritos pela Metodologia, independente de plataforma(java, php, etc..) basicamente indicam a confecção dos Modelo de Casos de Uso, Modelo de Domínio, Modelo de Projeto (Classes de Controle, Classes de Serviços e Classes de Persistência, Classes de Domínio, Classes DTO) e Diagrama de Seqüência.

# 2 ORGANIZAÇÃO DA MODELAGEM

## 2.1 Modelo de Caso de Uso

O Modelo de Casos de Uso é um modelo das funções pretendidas do sistema e seu ambiente. Este modelo é organizado por um ou mais Diagramas de Caso de Uso que apresentam atores, casos de uso e seus relacionamentos.

### Organização:

- Pacote “**Modelo\_Casos\_Uso**”: Conterá todos os elementos relacionados ao Modelo de Caso de Uso. Pode ser organizado em pacotes funcionais.
  - **Pacote Funcional:** Para projetos de software maiores ou mesmo com o objetivo de melhorar a compreensão do domínio do problema o Modelo de Caso de Uso pode ser divididos em pacotes funcionais, separando projeto por domínio de problema. Os elementos (casos de uso e atores) dentro de um pacote podem ter relacionamentos com elementos de outros pacotes. Quando um Ator ou Caso de Uso aparecer em

mais de um pacote, define-se este Ator ou Caso de Uso em um pacote e apenas arrasta-o para o diagrama do(s) outro(s) pacotes que o utilizarem, **sem efetuar cópia**.

## 2.2 Classes

### 2.2.1 Domínio

O Modelo de Domínio tem por objetivo a identificação dos conceitos relacionados com os requisitos do sistema e as regras de negócio, analisando o problema sob uma perspectiva conceitual para melhor compreensão.

#### Organização:

- **Pacote Classes / Pacote Dominio**

### 2.2.2 Serviço

Classes que objetivam minimizar as dependências entre camadas, fornecendo uma interface de alto nível para executar as regras de negócios de uma aplicação.

#### Organização:

- **Pacote Classes / Pacote Servico**

### 2.2.3 Persistência

Classes que encapsulam o acesso à fonte de dados.

#### Organização:

- **Pacote Classes / Pacote Persistencia**

### 2.2.4 Controle

Classes que recebem as **solicitações da camada de visão** e as transferem para a camada de modelo.

#### Organização:

- **Pacote Classes / Pacote Controle**

### 2.2.5 Telas

Classes que recebem o evento do ator externo e repassam para a camada de controle.

#### Organização:

- **Pacote Classes / Pacote Telas**

### 2.3 Diagrama de Seqüência

Descreve como um Caso de Uso é realizado no modelo de projeto, em termos de objetos que colaboram entre si para executar as tarefas requeridas. Representa a conexão existente entre os requisitos expressos como Casos de Uso e o projeto de objetos que atende a esses requisitos.

#### Organização:

- **Pacote Modelo de Casos de Uso / Pacote Funcional(se existir) / Caso de Uso(elemento) / cenário (interação)**

**Observação:** Na ferramenta RSM, para adicionar uma interação ao elemento Caso de Uso clique com o botão direito sobre o caso de uso e selecione opção Propriedades, nesta opção inclua uma nova interação na propriedade “OwnedBehavior”.

### 2.4 Ilustração



Figura 1: Estruturação da Modelagem

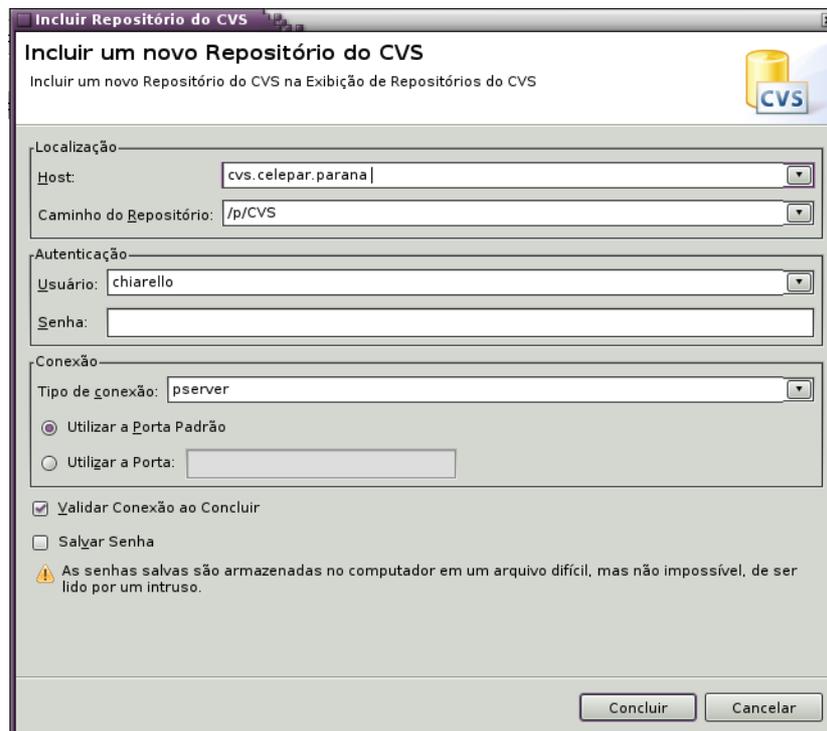
**Observação 1:** Todos os pacotes criados na ferramenta RSM podem ser divididos em mais arquivos de modelagem(.emx), através da opção “Criar modelo a partir do pacote” utilizando o botão direito do mouse sobre o nome do pacote. A separação em vários arquivos de modelagem é útil para diminuir a concorrência de uso(divisão de trabalho entre pessoas).

**Observação 2:** A organização citada neste documento refere-se a classes de Domínio, Serviço e Persistência. Caso seja necessário guardar outros tipos de classe(DTO, VO, etc...) pode-se criar um novo pacote abaixo do pacote raiz **Classe**.

### 3 COMPARTILHAMENTO

#### 3.1 Configuração da localização do repositório de versões

Através da perspectiva, Exploração de Repositórios do CVS, clique com o botão da direita do mouse e em seguida clique em nova localização de repositório e a tela “Incluir repositório no CVS”, irá abrir, conforme Figura 2.



The screenshot shows a dialog box titled "Incluir Repositório do CVS" with the subtitle "Incluir um novo Repositório do CVS na Exibição de Repositórios do CVS". The dialog is divided into several sections:

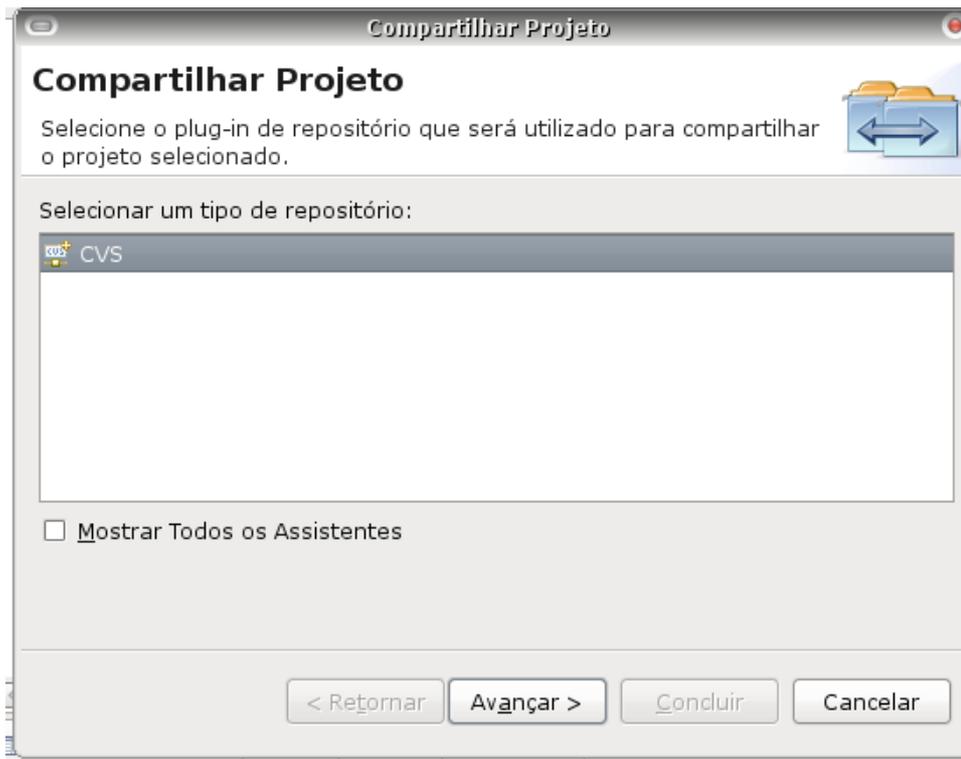
- Localização:** Host: cvs.celepar.parana; Caminho do Repositório: /p/ CVS.
- Autenticação:** Usuário: chiarello; Senha: (empty field).
- Conexão:** Tipo de conexão: pserver; Utilizar a Porta Padrão (selected); Utilizar a Porta: (empty field).
- Checkboxes: Validar Conexão ao Concluir (checked); Salvar Senha (unchecked).
- Warning: As senhas salvas são armazenadas no computador em um arquivo difícil, mas não impossível, de ser lido por um intruso.
- Buttons: Concluir, Cancelar.

Figura 2 - Novo repositório.

#### 3.2 Compartilhando um projeto no controle de versões – CVS

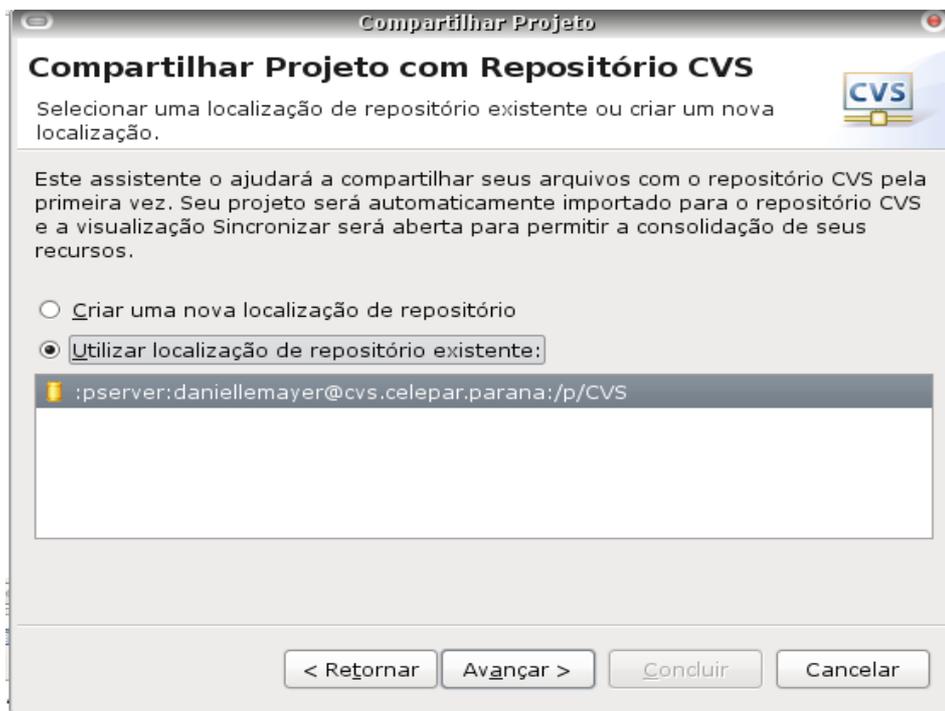
Compartilhe um projeto no CVS através dos seguintes passos:

1 – Clique com o botão direito no projeto desejado, selecione a opção **Equipe->Compartilhar Projeto**, conforme tela abaixo. Após clique em avançar.



*Figura 3: Compartilhar Projeto*

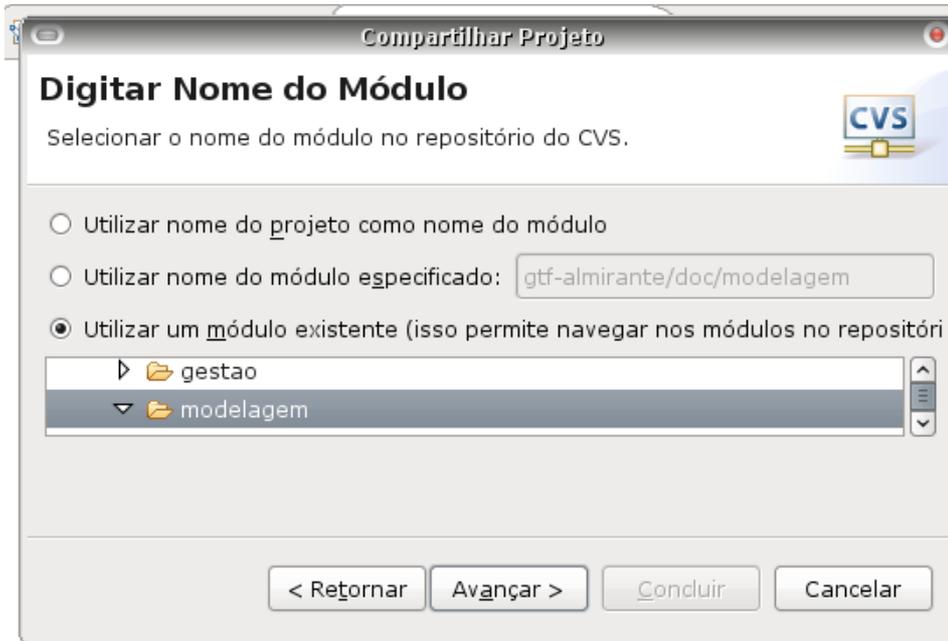
2 – Selecione a opção “Utilizar localização de repositório existente”. Feito isto, selecione a pasta do cvs em que ficará o projeto de modelagem criado.



*Figura 4: Compartilhar Projeto*

3 - Selecione a opção “Utilizar um módulo existente....” e navegue pelas pastas do cvs até

encontrar a pasta destino para este novo projeto.



*Figura 5: Compartilhar Projeto*

4 - Clique em avançar até a conclusão do compartilhamento.

### **3.3 Recuperando um projeto existente no controle de versões – CVS**

A recuperação de um projeto de modelagem disponível no CVS é realizado através do menu -> arquivo -> importar -> registrar saída de projeto do CVS, seguindo os seguintes passos:

1 – Selecione a opção “Utilizar localização de repositório existente”, conforme figura abaixo, e clique em avançar.

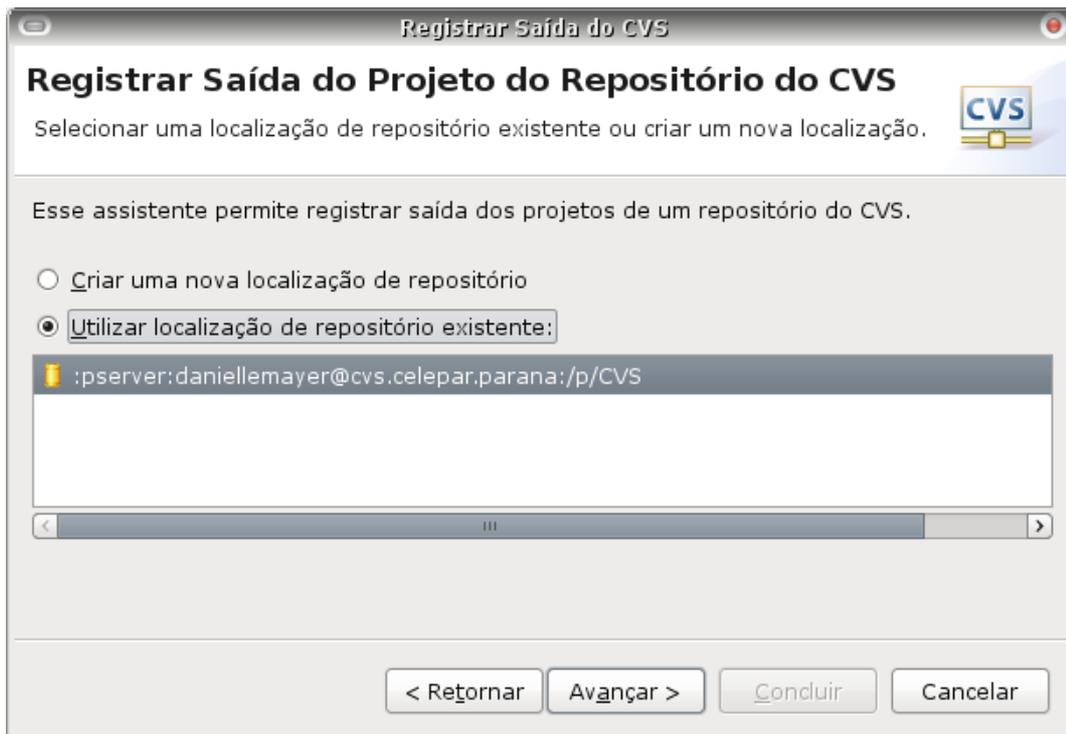


Figura 6: Importando Projeto do CVS

2 – Selecione a opção “Utilizar um módulo existente.....”, navegue pelos módulos e escolha a pasta do cvs criada para armazenar o projeto em questão(**pode ser o projeto todo ou apenas a pasta aonde está armazenado o .emx, a critério do usuário**) e clique em avançar:

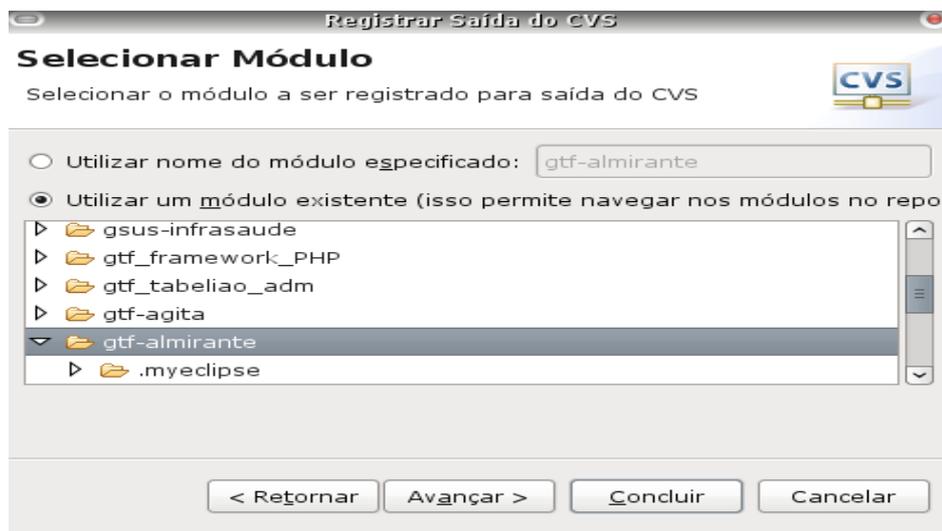


Figura 7: Importando Projeto do CVS

3 – Selecione o método de registro da saída **aceitando o nome** da pasta sugerido que contém o projeto do cvs (**nome do projeto todo ou apenas o nome a pasta aonde está armazenado o .emx, conforme escolhido no item anterior – item 2** ) e clique em avançar até a conclusão da importação.

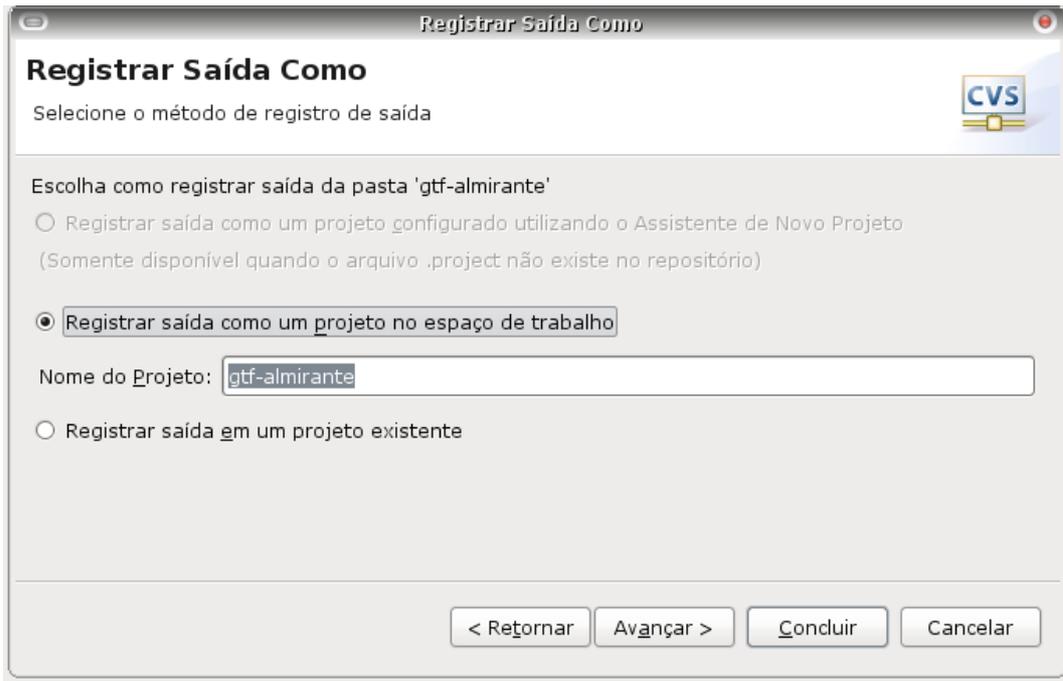


Figura 8: Importando Projeto do CVS

## 4 TRABALHADO EM EQUIPE

Quando equipes desenvolvem modelos em paralelo, seus integrantes comparam e mesclam modelos para resolver conflitos ou diferenças entre versões de um modelo. Equipes que trabalham com modelos em um sistema de gerenciamento de configuração podem comparar versões de arquivos para entender o histórico e o desenvolvimento. Os participantes da equipe precisam mesclar os arquivos freqüentemente quando desenvolvem modelos em paralelo. Equipes pequenas geralmente tem poucos problemas quando trabalham em paralelo, mas em equipes grandes a sobreposição de artefatos e responsabilidades se tornam comum gerando riscos para o projeto:

- Risco de sobrescrever artefatos de outros membros da equipe;
- Comunicações gerais para a coordenação de atividades paralelas;
- Obrigação de juntar todos os artefatos produzidos em paralelos que geraram mudanças em uma versão final.

Desenvolvimento paralelo de modelos é consideravelmente mais complexo do que desenvolvimento paralelo de artefatos texto como código Java. Quando um erro ocorrer no código

(texto) durante uma junção, a correção é geralmente evidente ao inspecionar o arquivo resultante da fusão. Muitas vezes, o compilador irá sinalizar o erro com precisão. Na modelagem de artefatos UML, mesmo utilizando o Eclipse Modeling Framework (EMF), leva a uma grande complexidade interna no arquivo de sintaxe e semântica de conteúdo. Esta complicação adicional torna ainda mais importante a identificação clara das junções de inconformidades e gerir as interações nas modelagens em paralelo.

#### **4.1 Comparando versões de um modelo:**

Dentro do RSM é possível a comparação entre versões do modelo com o seu histórico, tendo como finalidade visualizar possíveis diferenças.

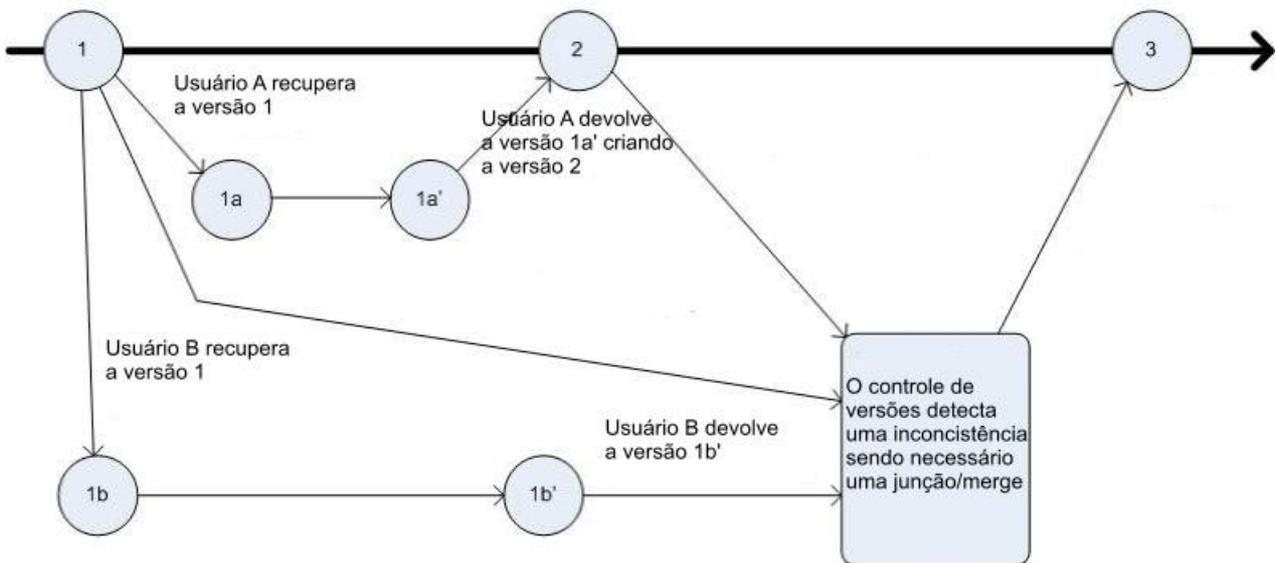
Para realizar uma comparação nos modelos basta:

1º – Na guia de visualização “Explorar Modelos”, clique com botão direito do mouse sobre o modelo, em seguida em equipe e em mostrar no histórico de recursos;

2º – Na guia “visualização histórico de recurso”, selecione uma das versões existentes na lista de revisão;

3º – Clique com o botão da direita do mouse sobre a seleção, em seguida clique em comparar. O sistema irá abrir uma guia como as possíveis diferenças.

## 4.2 Fluxo clássico para o desenvolvimento paralelo.



*Figura 9 - Fluxo clássico de trabalho paralelo.*

A Figura 9 mostra dois usuários (A e B) utilizando a mesma versão um (1) de um artefato. Os usuários fazem alterações em separado no mesmo artefato, quando o usuário A submeter, em primeiro lugar o artefato alterado, cria a versão 2. Quando o usuário B realizar novas alterações e submete-las o controle de versões detecta a mudança e marca o arquivo como um conflito. Usuário B poderá aceitar as alterações, de forma automática ou manual, e completa a junção do arquivo para envio para o repositório, criando versão 3.

## 4.3 Identificando conflitos

A fórmula para a detecção de mudanças em trabalhos paralelos é, na verdade, muito simples, artefatos recebidos, Figura 10, são comparados com as versões mais recentes existentes no controle de versões (também chamada de versão remota). Se a versão remota é o antepassado direto da próxima versão, então o *check-in* pode proceder. Se assim não for, então um conflito é sinalizado, conforme Figura 11, e uma junção deve ser realizada. Através desta manobra, reduz drasticamente o potencial de perda de dados

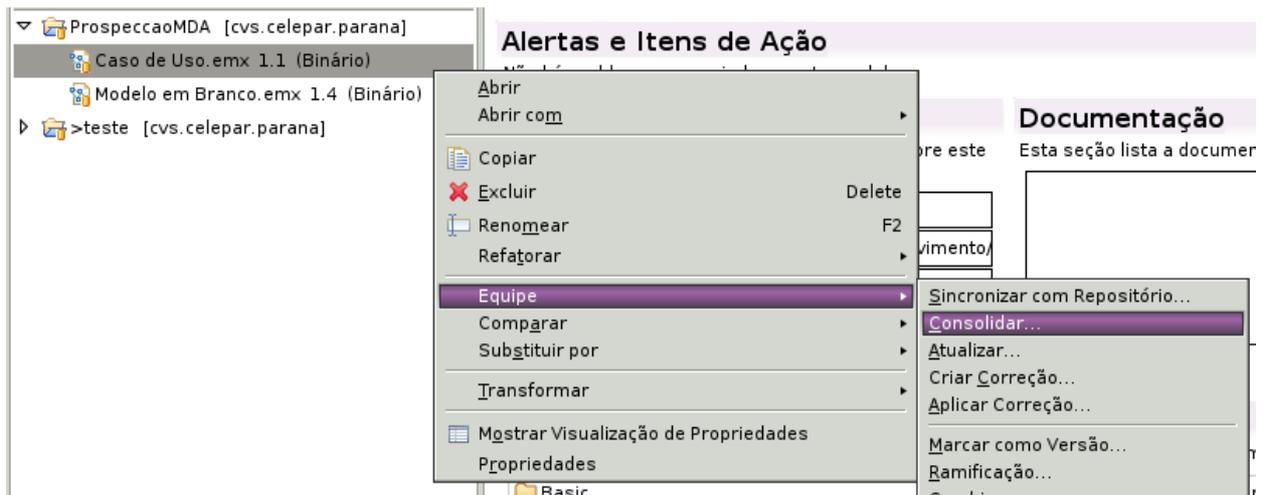


Figura 10 - Consolidação de alterações.

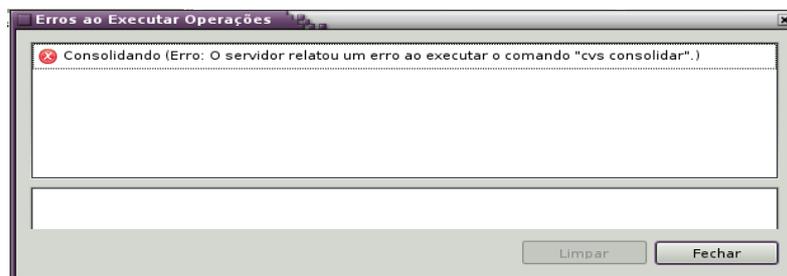


Figura 11 - Sinalização de inconformidade.

#### 4.4 Sincronizando artefatos

Para a realização das junções, indicadas pela existência de conflitos, é necessário sincronizar as diferentes versões de artefatos existentes (local e remota), conforme a Figura 12.

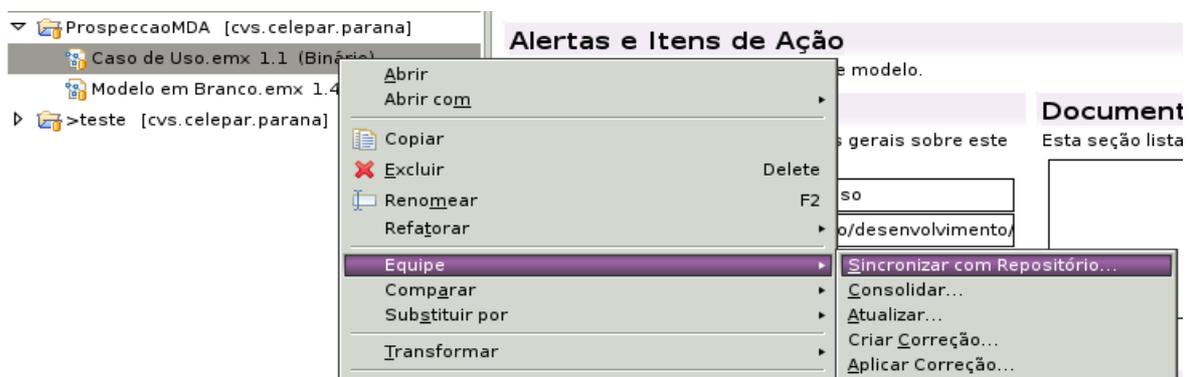


Figura 12 - Sincronização de artefatos.

A Figura 13, mostra uma junção de três vias com 2 usuários um **local** a esquerda, identificado com o círculo três (3) e um **remoto** a direita, identificado com o círculo cinco (5). Um possível resultado é representado, após a junção/merge, sob o círculo (2). Sob o círculo (4) é

apresentado o conjunto básico que originou os conflitos e sob o círculo (1) é apresentado as diferenças entre as estruturas, **local** e **remota**. No RSM a funcionalidade de merge é muito semelhante ao Eclipse para comparar uns com os outros e mesclar. Esta técnica é descrita em detalhes mais adiante.

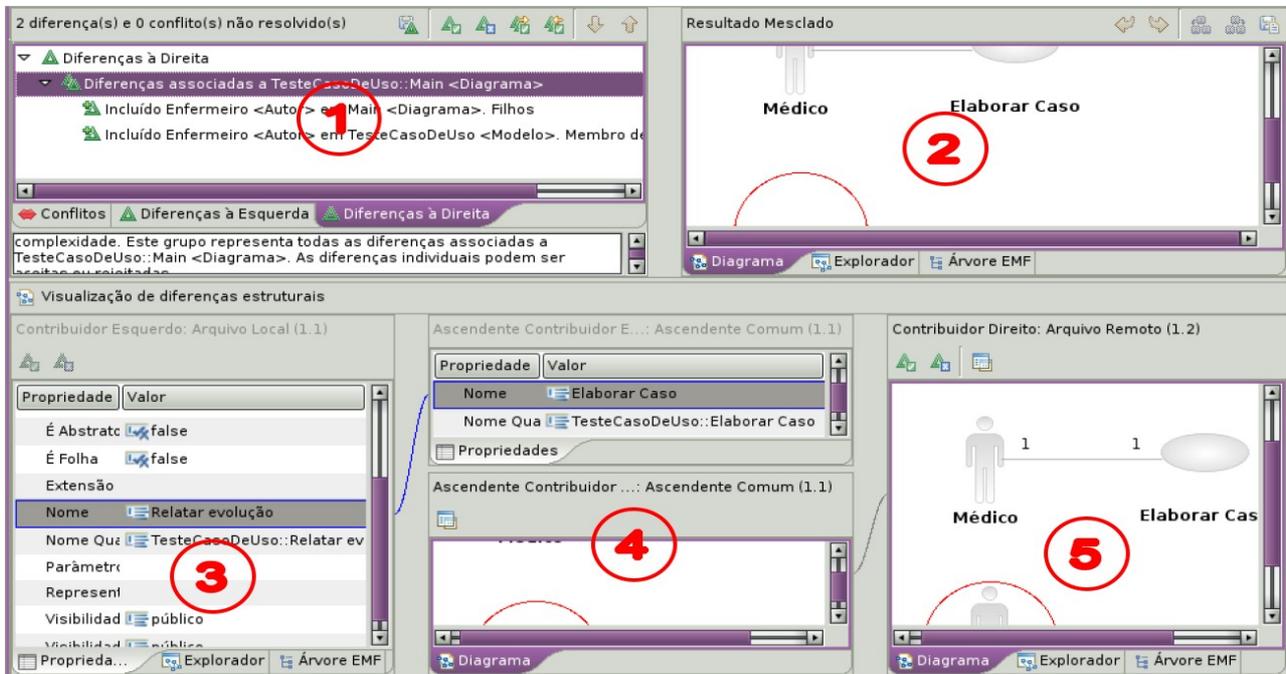


Figura 13 - Observador e a identificação das mudanças.

## 4.5 Resolvendo os conflitos e diferenças

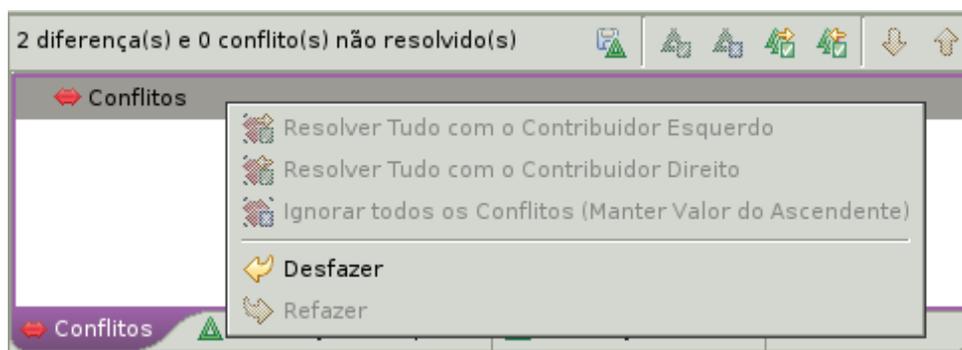


Figura 14 - Diferenças estruturais.

A Figura 14 representa a página de conflitos e mostra 2 (duas) diferenças. Para resolver todos os conflitos, na página conflitos, clique com o botão direito do mouse em conflitos; em seguida, clique em um dos itens a seguir:

- Para aceitar todas as alterações da página Diferenças à Esquerda, clique em Resolver Tudo com Contribuidor Esquerdo;

- Para aceitar todas as alterações da página Diferenças à Direita, clique em Resolver Tudo com Contribuidor Direito;
- Para manter os valores do modelo base, clique em Ignorar Todos os Conflitos (Manter Valor do Ascendente).

## 4.6 Resolvendo os conflitos específicos

Ao mesclar versões de um modelo, é possível resolver conflitos individualmente, aceitando todas as alterações de diferentes versões do modelo.

**Atenção:** Para evitar danos aos dados no modelo, não mescle modelos fora de RSM (Rational Software Modeler).

Para resolver um conflito específico, na página conflitos, clique com o botão direito do mouse em um conflito; em seguida, clique em um dos itens a seguir:

- Para aceitar as alterações da página Diferenças à Esquerda, clique em “Resolver com Contribuidor Esquerdo”, conforme representado na Figura 15 onde foi alterado nome do Caso de Uso.

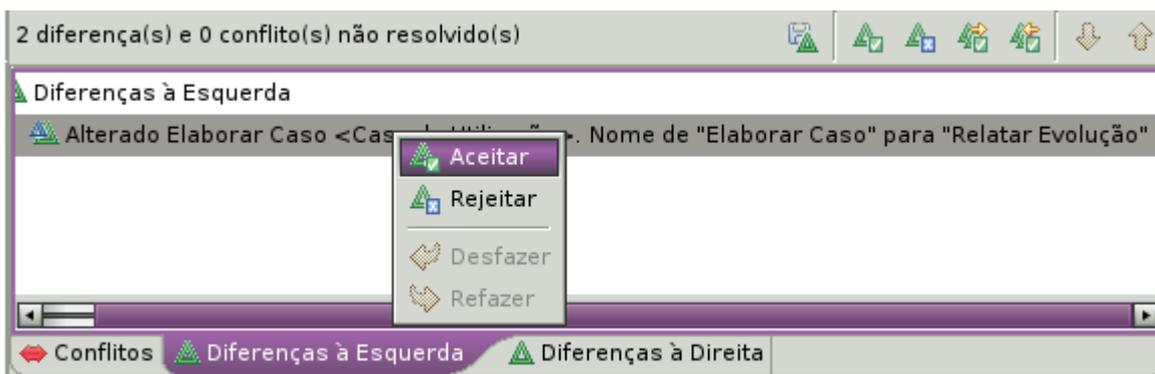
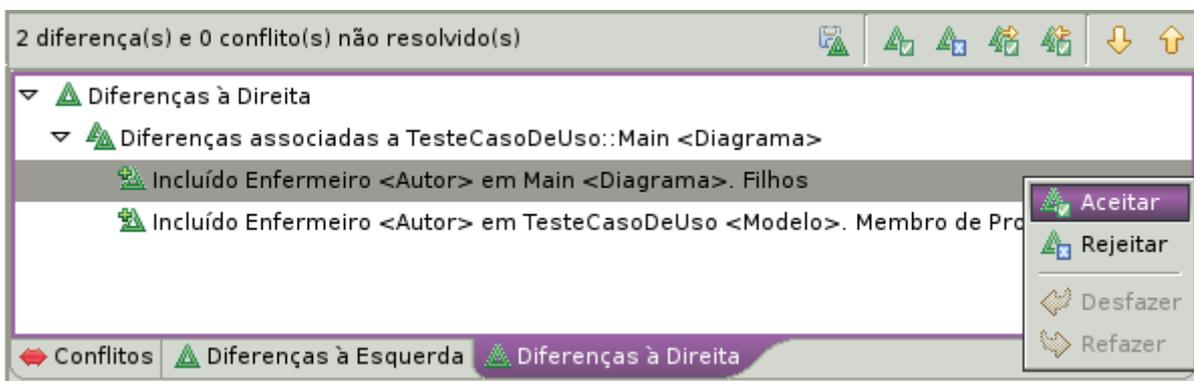


Figura 15 - Diferença local.

- Para aceitar as alterações da página Diferenças à Direita, clique em “Resolver com Contribuidor Direito”, conforme representado na '1"Figura 16 onde foi criado um novo Ator “Enfermeiro”.



'1"Figura 16 - Diferença no servidor.

- Para manter os valores do modelo base, clique em Rejeitar (Manter Valor do Ascendente).