



Universidade Federal de Pernambuco



Graduação em Bacharelado em Ciências da Computação

Centro de Informática

**Promodeller: Ferramenta de Modelagem de Processo de
Software a partir de arquivos no padrão BPMN utilizando
Ontologia de Processo**

Trabalho de Graduação

Aluno: Felipe Ebert <felipe.ebert@gmail.com>

Orientador: Professor Ph.D. Alexandre Marcos Lins de Vasconcelos <amlv@cin.ufpe.br>

Universidade Federal de Pernambuco

Centro de Informática

FELIPE EBERT

**Promodeller: Ferramenta de Modelagem de Processo de Software
a partir de arquivos no padrão BPMN utilizando Ontologia de
Processo**

Monografia apresentada ao Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do Grau de Bacharel em Ciência da Computação sob orientação do Professor Ph.D. Alexandre.

Recife, Julho de 2009

Dedicado a minha família e meus amigos por serem os responsáveis por tudo que me tornei e conquistei.

Agradecimentos

Em primeiro lugar, agradeço à minha família por ter me proporcionado uma educação de qualidade e ter me mostrado os verdadeiros princípios e valores que uma pessoa dever ter. Agradeço também por sempre me apoiarem em busca dos meus sonhos.

À minha namorada, Camila, sempre demonstrando carinho e compreensão me incentivou e deu forças a realizar esse trabalho.

Aos meus amigos, que há muito tempo celebramos nossas amizades nos *Gravatá's*. Com certeza ficarão marcados nossos encontros.

Ao meu orientador professor Alexandre Vasconcelos por ajuda e orientação no desenvolvimento deste trabalho.

Muito Obrigado.

Resumo

A modelagem de processos ajuda empresas a conseguirem um melhor controle e gerenciamento dos processos. Para empresas de software, a modelagem de processos de software é de extrema importância para a qualidade do produto final. E esse processo deve ser modelado em uma linguagem que possa ser entendida facilmente por todas as pessoas envolvidas no desenvolvimento do software. Nesse contexto surgiu o SPEM (*Software Process Engineering Metamodel*) que é um dos padrões de modelagem de processos de software mais difundidos e aceitos na comunidade de Engenharia de Software.

De forma parecida, empresas, não necessariamente de software, precisam controlar seus processos de negócio para que todas suas atividades e tarefas estejam de acordo com as metas e estratégias da organização. Para tanto, foi desenvolvido o BPMN (*Business Process Modeling Notation*) que é um padrão de modelagem para processos de negócio.

Assim, este trabalho visa somar à ferramenta de modelagem processo ProModeller, que utiliza a notação SPEM, a opção de mapear para o padrão BPMN a modelagem de um processo de software específico, visto que o processo de software é também um processo de negócio da empresa.

Palavras-chave: Processos de Software, Processos de Negócio, Modelagem, SPEM, BPMN, XPDL.

Lista de Figuras

Figura 2.1- Seqüência Básica de um Processo.	13
Figura 2.2- Arquitetura do SPEM definida pela OMG.	14
Figura 2.3- Estrutura de Pacotes do SPEM.	15
Figura 2.4- Detalhamento dos Sub-Pacotes do SPEM.	16
Figura 3.1- Diagrama de Pacotes do ProModeller.	25
Figura 3.2- Diagrama de Atividades do ProModeller.	26
Figura 3.3- Diagrama de Pacotes do ProModeller.	28
Figura 3.4- Diagrama de Pacotes do Pacote GUI do RroModeller.	29
Figura 4.1- Novo Diagrama de Sub-Pacotes do Pacote GUI.	31
Figura 4.2- Implementação da Funcionalidade Salvar Formato BPMN.	32
Figura 4.3- Exemplo de Modelagem BPMN no ProModeller.	33
Figura 4.4- Arquivo XPDL do Processo da Figura 3.4.	38
Figura 5.1- Modelagem do Processo em SPEM.	44
Figura 5.2- Modelagem do Processo em BPMN.	47
Figura 5.3- Modelagem do Processo BPMN na Ferramenta BizAgi.	52

Lista de Tabelas

Tabela 2.1- Estereótipos do SPEM Profile.....	18
Tabela 2.2- Categoria Flow Objects o BPMN.....	20
Tabela 2.3- Categoria Connecting Objects do BPMN.....	21
Tabela 2.4- Categoria Swimlanes do BPMN.....	21
Tabela 2.5- Categoria Artifacts do BPMN.....	22
Tabela 2.6- Tabela Mapeamento SPEM - BPMN.....	23
Tabela 5.1- Critério da Modelagem do Processo.....	40
Tabela 5.2- Relação dos Artefatos das Atividades do Processo.....	41

Sumário

Sumário	8
1. Introdução	10
1.1. Motivação	10
1.2. Objetivos.....	11
1.3. Metodologia do Trabalho	11
1.4. Estrutura do Trabalho	11
2. Modelagem de Processos.....	13
2.1. SPEM.....	13
2.1.1. Estrutura de Pacotes	14
2.1.2. Estereótipos do SPEM.....	18
2.2. BPMN.....	19
2.2.1. Elementos Gráficos.....	19
2.3. Mapeamento entre SPEM e BPPMN	22
3. ProModeller: Ferramenta para Modelagem de Processo de Software	25
3.1. Serviços do ProModeller	26
3.2. Estrutura Interna do ProModeller.....	27
4. Implementação da Visão BPMN no ProModeller.....	30
4.1. Funcionalidade de Salvar no Formato BPMN.....	30
4.2. Modelagem da Visão BPMN e Salvar Formato BPMN.....	31
4.3. Exemplo de Uso da Visão BPMN.....	33
5. Exemplo de Uso do ProModeller: Mapeamento de SPEM em BPMN.....	39
5.1. O Processo – Garantia de Qualidade de Processo e Produto	39
5.2. Modelagem do Processo em SPEM	40
5.3. Modelagem do Processo em BPMN	45
6. Conclusão.....	53

6.1. Trabalhos Futuros.....	53
6.2. Considerações Finais	54
Referências Bibliográficas.....	55

1. Introdução

A crescente exigência de qualidade dos mercados tem influenciado as organizações a buscarem novos métodos e ações que satisfaçam essa demanda. Nesse contexto, tanto empresas de TI (Tecnologia da Informação) quanto de outras áreas tentam se adaptar utilizando padrões de qualidade. Nessa adaptação é de extrema importância que os processos dessas empresas sejam bem planejados, obedeçam a algumas regras e gerem o resultado esperado [SÁ FILHO, 2007].

Nesse contexto, as organizações que pretendem melhorar e controlar os seus processos têm utilizado técnicas de Modelagem de Processos (como por exemplo: SPEM (*Software Process Engeneering Metamodel*) [SPEM, 2008] para modelagem de processos de software e BPMN (*Business Process Modeling Notation*) [BPMN, 2009] para a modelagem de processos de negócio) de forma a planejar, controlar e verificar os processos.

Porém, para uma empresa que utiliza SPEM, fica mais difícil ou quase impossível realizar a integração e interação da modelagem dos processos de software com a modelagem de processos de outras áreas da organização, dificultando assim o alinhamento dos processos com os objetivos globais da organização. Se estas empresas pudessem mapear seus processos modelados em SPEM para BPMN, seria possível realizar essa integração e interação.

1.1. Motivação

O uso de linguagens de modelagem de processos está cada vez mais em evidência devido à sua contribuição para a definição, acompanhamento e evolução dos processos.

A grande utilização dos padrões SPEM e BPMN reflete diretamente a busca das empresas pela qualidade dos seus produtos e serviços. Uma análise profunda entre esses padrões como também a implementação de uma ferramenta que faça um mapeamento entre essas duas notações pode resultar numa maior produtividade e qualidade para a atividade de modelagem de processos. O mapeamento de SPEM para BPMN implica que a empresa terá uma associação direta entre os padrões possibilitando: um mapeamento em linguagens de execução de processo, podendo assim avaliar e verificar os seus processos em ferramentas de execução de processo; uma compreensão da modelagem do processo por todos da

organização devido à simplicidade gráfica do BPMN (visto que o SPEM é uma linguagem de domínio específico para processos de software) e facilidade para migrar de SPEM para BPMN.

1.2. Objetivos

Este Trabalho de Graduação tem como objetivo apresentar um estudo comparativo entre as notações para modelagem de processos SPEM e BPMN, bem como propor e implementar uma ferramenta que faça o mapeamento do padrão SPEM para o BPMN. Este mapeamento é de grande importância para as empresas que utilizam o padrão SPEM, pois facilita quando for necessário utilizar o padrão BPMN para realizar a integração com os demais processos organizacionais de modo a alinhar os seus objetivos com as metas globais da instituição

1.3. Metodologia do Trabalho

Neste trabalho será utilizado o estudo bibliográfico da literatura relacionada aos padrões SPEM e BPMN para modelagem e definição de processos, dando um foco maior aos processos de desenvolvimento de software. Em seguida será feita uma análise comparativa e um mapeamento entre esses dois padrões. Esse mapeamento servirá de base para o desenvolvimento de uma ferramenta que realizará o mapeamento de SPEM para BPMN.

1.4. Estrutura do Trabalho

Além deste capítulo introdutório, este trabalho contém mais cinco capítulos:

- Capítulo 2 – Modelagem de Processos, esse capítulo apresenta conceitos e fundamentos relacionados à modelagem de processos de software e de negócios, os seus respectivos padrões de modelagem SPEM e BPMN como também uma comparação entre eles;
- Capítulo 3 – ProModeller: Ferramenta para Modelagem de Processo de Software. Este capítulo apresenta a ferramenta ProModeller, utilizada para modelar processos em SPEM, e suas principais características;

- Capítulo 4 – Implementação da Visão BPMN no ProModeller. Este capítulo descreve como foi desenvolvida a nova funcionalidade de mapeamento de SPEM para BPMN no ProModeller;
- Capítulo 5 – Exemplo de Uso do ProModeller. Neste capítulo será apresentado um exemplo de uso com o intuito de avaliar o desempenho da ferramenta desenvolvida;
- Capítulo 6 – Conclusões e Trabalhos Futuros. Este capítulo encerra o trabalho, trazendo conclusões acerca da contribuição do mesmo e apresentando também sugestões de continuidade através de trabalhos futuros.

2. Modelagem de Processos

Um dos fatores apontados como imprescindíveis para o sucesso de empresas, tanto de desenvolvimento de software quanto do demais ramos, é a utilização de um processo bem definido. Processo, sob o ponto de vista de desenvolvimento de software, pode ser definido como “*um conjunto de atividades e resultados associados que levam à produção de um produto de software*” segundo [SOMMERVILLE, 2003]. Essas atividades são organizadas e estruturadas, podendo ter como entrada algum artefato, e devem gerar pelo menos um artefato, sendo executadas por um agente humano ou computacional. Em geral devem-se considerar os seguintes pontos sobre um processo de software:

- Recursos - que são necessários para a execução das atividades;
- Atividades – que são executadas pelos recursos e geram algum artefato;
- Artefatos – que são produtos gerados pelas atividades, e que podem ser também matéria-prima para outras atividades do processo.

Já sob o ponto de vista de processo de negócio, tem-se uma definição bastante semelhante a de processo de software: é uma atividade ou um conjunto de atividades realizadas por uma organização criar ou adicionar algum valor no seu produto para os clientes, ou seja, qualquer atividade ou conjunto de atividades em que há uma entrada, uma transformação e uma saída [SÁ FILHO, 2007] com mostra a figura 2.1:

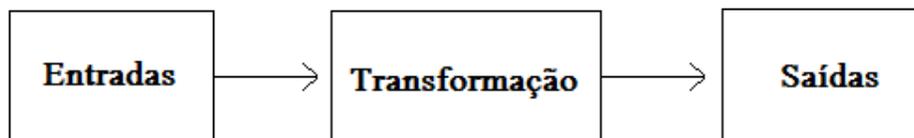


Figura 2.1- Sequência Básica de um Processo.

2.1. SPEM

O SPEM é um meta-modelo utilizado para definir e modelar processos de software e é mantido pela *Object Management Group* (OMG) desde 2002. Atualmente o SPEM está na versão 2.0 de Abril de 2008. O SPEM é baseado em uma arquitetura de quatro camadas, como mostra a figura 2.2.

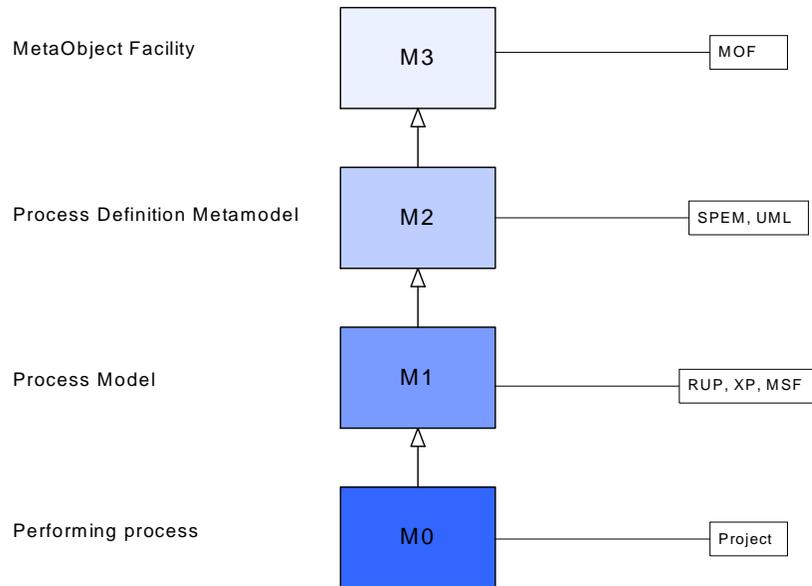


Figura 2.2- Arquitetura do SPEM definida pela OMG.

O nível M0 é definido por processos instanciados, ou seja, processos que estão em execução em um projeto. A definição dos processos abstratos, estão no nível M1, como por exemplo o *Rational Unified Process* (RUP). O SPEM se encontra no nível M2 servindo de base para os processos do nível M1. Por fim, no nível M3 está definido o *Meta-Object Facility* (MOF) que é uma tecnologia adotada pela OMG para definir meta-dados.

2.1.1. Estrutura de Pacotes

O modelo SPEM é estruturado em dois pacotes, como mostrado na figura 2.3.

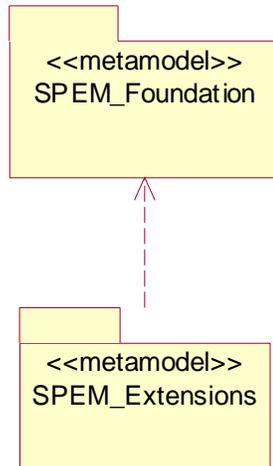


Figura 2.3- Estrutura de Pacotes do SPEM.

O primeiro pacote, chamado *SPEM_Foundation*, representa uma extensão da especificação *Unified Modeling Language* (UML) e contém os elementos básicos que dão suporte ao SPEM e ao segundo pacote, o *SPEM_Extensions*, que contempla abstrações necessárias e as semânticas requeridas para a engenharia de processo de software [SPEM, 2008].

O pacote *SPEM_Foundation* é formado por seis sub-pacotes: *Core*, *Data_Types*, *Activity_Graphs*, *Model_Management*, *Actions* e *State_Machines*. Dentro deles estão contidos elementos básicos para modelagem de processo, como tipos de dados: *Boolean*, *Integer*, *String*, *AggregationKinf*, etc.; elementos que representam ações: *Action*, *CallAction*, etc.; elementos de estados: *State*, *Transiction*, etc.; elementos que representam diagramas de atividades: *ActionState*, *ActivityGraph*, etc.; e um elemento para gerenciamento dos modelos: *Package*.

O pacote *SPEM_Extensions* também está estruturado em sub-pacotes: *Basic Elements*, *Dependencies*, *Process Structue*, *Process Components* e *Process Lifecycle*. A figura 2.4 mostra essas relações de dependências.

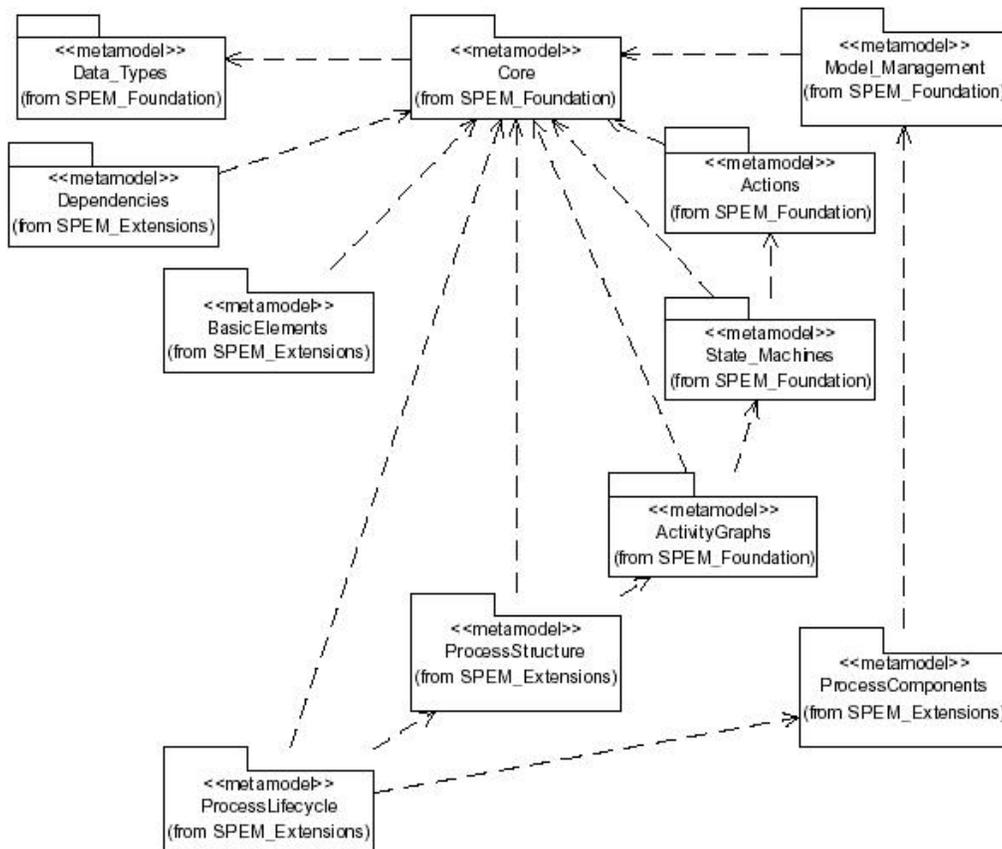


Figura 2.4- Detalhamento dos Sub-Pacotes do SPEM.

O pacote *Basic Elements* (Elementos Básicos) contém elementos básicos para a descrição do processo. São elementos desse pacote:

- **Content Description:** contém a descrição de um elemento do processo;
- **Guidance:** responsável por fornecer informações mais detalhadas sobre os elementos do processo. Existem quatro tipos de *Guidance*: *Technique*, *Checklist*, *Guideline* e *Template*.

O pacote *Dependencies* contém elementos que representam dependências. São elementos desse pacote:

- **Category:** serve para associar um pacote a um elemento de outro pacote;

- **Import:** serve para indicar que todo o conteúdo do pacote em questão é adicionado ao *namespace* do pacote origem;

O pacote *Process Structure* contém os elementos principais para a construção da modelagem. São elementos desse pacote:

- **Work Product and Kind:** Um *Work Product* está relacionado a uma classe de produto e um *Kind* está relacionado ao tipo de produto.
- **Work Definition and Work Definition Parameter:** é um tipo de operação que representa o tipo de trabalho desenvolvido no processo, que pode ser uma *Task*, *Step* ou *Activity*.
- **Activity and Step:** é a principal subclasse de *Work Definition* e representa o trabalho realizado por um *Role Definition*. Uma atividade pode ser constituída por *Steps* que são elementos atômicos.
- **Role Use and Role Definition:** um *Role Use* representa o executor de uma ou mais definições de trabalho (*Work Definitions*) no processo. *Role Definition* representa papéis e responsabilidades em atividades (*Activity*).

O pacote *Process Components* contém elementos que têm por objetivo dividir o projeto em definições para que, assim, possam ser colocadas sob gerência de configuração e versão. São elementos desse pacote:

- **Method Package and Process Package:** é um “recipiente” que pode possuir elementos de definição de métodos e de processo.
- **Process Component:** representa uma descrição do processo, sendo consistente e podendo ser reusado com outros componentes para criar um processo maior e mais complexo.
- **Process:** é definido a partir de um *Process Component*. Ele representa uma família de processos, dentro dos quais, mais processos podem ser definidos.
- **Category (instance):** é uma especialização de *Category* e divide as atividades do processo de acordo com um tema.

O pacote *Process Lifecycle* contém elementos que ajudam a definir a execução do processo, ou seja, como ele será executado. São elementos desse pacote:

- **Kind:** qualifica as entidades básicas. Por exemplo, um *Guidance* pode ser do *Kind: White Paper, Guideline, Checklist* ou *Templates*, e uma *Activity* como uma *Activity* (atividade) ou *Phase* (fase).
- **Precondition:** a cada *Work Definition* pode ser associada a uma pré-condição.
- **Postcondition:** semelhante a um *Precondition*, uma pré-condição pode ser associada a uma *Work Definition*.

2.1.2. Estereótipos do SPEM

Na tabela 2.1 a seguir são mostrados os principais elementos gráficos de SPEM [SPEM, 2008].

Tabela 2.1- Estereótipos do SPEM Profile.

Estereótipo	Descrição	Notação
Work Product	Representa elementos consumidos ou gerados por uma atividade.	
Work Definition	Representa elementos que podem ser decompostos em outros elementos.	
Guidance	Representa itens que podem ajudar outros elementos a realizar tarefas.	
Activity	Representa uma tarefa que um Role Definition / Role Use realiza no processo.	
Role Definition	Representa o papel dos recursos dentro do Processo. Como por exemplo, arquiteto de software, gerente de projeto, etc.	

Role Use	Representa os executores das atividades, isto é, os recursos.	
Process Package	Representa um <i>container</i> que separa os processos por conteúdo.	
Kind	Representa as fases do processo	
Process	Representa um conjunto de atividades e/ou processos.	
Document	Representa os diversos tipos de documentos do processo.	

2.2. BPMN

O BPMN (*Business Process Modeling Notation*) é uma notação padrão para a modelagem de processos de negócios. Ele permite mapear em detalhes todos os processos de negócio da organização como também representar relações entre a organização e os outros *stakeholders* (por exemplo, fornecedores e clientes) [BPMN, 2009].

O objetivo do BPMN é facilitar o entendimento dos processos da empresa por todos os envolvidos no negócio, desde os desenvolvedores até os analistas de negócio. Além disso, também tem o objetivo de garantir que linguagens de execução de processo de negócio, como BPEL4WS (*Business Process Execution Language for Web Services*) possam ser visualizadas como uma notação orientada a negócios.

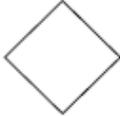
2.2.1. Elementos Gráficos

Os diagramas de processos de negócio – BDP – *Business Process Diagram*, modelados a partir do BPMN são baseados em um conjunto de elementos gráficos. Esse conjunto

permite a modelagem e compreensão de processos de negócio desde os mais simples até os mais complexos.

Os diagramas estão estruturados em quatro categorias: *Flow Objects* (Objetos de Fluxo), *Connecting Objects* (Objetos de Conexão), *Swimlanes* (Raias) e *Artifacts* (Artefatos) que são mostrados nas tabelas 2.2, 2.3, 2.4 e 2.5 a seguir.

Tabela 2.2- Categoria Flow Objects o BPMN.

Flow Objects		
Elemento	Descrição	Notação
Event	Um evento é algo que acontece durante a execução do processo. Ele afeta a execução do processo e, geralmente, possui uma causa (<i>trigger</i>) e um impacto (<i>result</i>). Eventos podem ser de três tipos, com base no momento em que afetam o processo: <i>Start</i> (Início), <i>Intermediate</i> (Intermediário) e <i>End</i> (Fim). Eventos do tipo início e intermediário também podem ser divididos quanto ao seu <i>trigger</i> (gatilho) em: <i>Message</i> , <i>Timer</i> , <i>Error</i> , <i>Cancel</i> , <i>Compensation</i> , <i>Rule</i> , <i>Link</i> , <i>Multiple</i> , <i>Terminate</i> .	
Activity	Atividade é um termo genérico para um trabalho desempenhado por uma organização. Uma atividade pode ser atômica ou não-atômica (composta). Os tipos de atividades que são parte de um modelo de processo são: <i>Process</i> (Processo), <i>Sub-Process</i> (Sub-Processo) e <i>Task</i> (Tarefa). <i>Task</i> e <i>Sub-Process</i> são representados por retângulos arredondados. <i>Process</i> não possui notação gráfica associada (é o modelo em si) ou é o conteúdo de um <i>Pool</i> (será visto mais adiante).	
Gateway	Um <i>Gateway</i> é usado para controlar a divergência ou convergência de fluxos de seqüência. Dessa forma, ele irá determinar ramificação, bifurcação, ligação e junção de caminhos. Símbolos internos à notação gráfica irão indicar o	

	tipo de controle.
--	-------------------

Tabela 2.3- Categoria Connecting Objects do BPMN.

Connecting Objects		
Elemento	Descrição	Notação
Sequence Flow	Um fluxo de seqüência é usado para indicar a ordem em que atividades serão executadas em um processo. Os fluxos de seqüência podem ser de quatro tipos: <i>Uncontrolled Flow</i> (Fluxo sem Controle), <i>Conditional Flow</i> (Fluxo condicional), <i>Default Flow</i> (Fluxo Padrão) e <i>Exception Flow</i> (Fluxo de Exceção).	
Message Flow	Um fluxo de mensagem é usado para mostrar a troca de mensagens entre dois participantes do processo. Em BPMN, dois <i>Pools</i> separados no diagrama representam dois participantes (uma entidade de negócio ou um papel do processo, por exemplo).	
Association	Usada para associar informação a objetos de fluxo. Objetos gráficos ou textuais, que não são de fluxos podem ser associados a objetos de fluxo.	

Tabela 2.4- Categoria Swimlanes do BPMN.

Swimlanes		
Elemento	Descrição	Notação
Pool	Um <i>Pool</i> representa um participante do processo.	
Lane	Uma <i>Lane</i> é uma sub-partição de um <i>Pool</i> . <i>Lanes</i> são usados para	

organizar e categorizar as atividades.

Tabela 2.5- Categoria Artifacts do BPMN

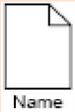
Artifacts

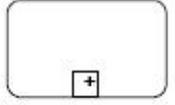
Elemento	Descrição	Notação
Data Object	Objetos de dados são considerados artefatos porque eles não têm efeito direto no fluxo de seqüência ou fluxo de mensagens do processo, mas eles provêem informação acerca do que as atividades precisam para serem executadas e o que elas produzem.	
Group	É um grupo de atividades que não afeta o fluxo de seqüência. O agrupamento pode ser realizado para fins de análise ou documentação. Grupos também podem ser usados para identificar atividades de uma transação distribuída que é mostrada entre <i>Pools</i> .	
Text Annotation	Anotações de texto são mecanismos utilizados pelos modeladores para prover informação adicional aos leitores do diagrama de processo de negócio.	

2.3.Mapeamento entre SPEM e BPPMN

Depois de apresentadas as duas notações utilizadas pra modelagem de processo, faz-se necessário realizar o mapeamento entre elas. O interesse desta seção é confrontar as duas tecnologias visando o objetivo de modelagem de processo de software, assim tomaremos como base o meta-modelo SPEM. Na tabela 2.6 consta a proposta de mapeamento dos principais elementos do SPEM para os elementos correspondentes do BPMN [SPEM, 2008] [BPMN, 2009].

Tabela 2.6- Tabela Mapeamento SPEM - BPMN.

Mapeamento SPEM - BPMN		
SPEM	BPMN	Justificativa
 <p><i>Work Product</i></p>	 <p><i>Data Object</i></p>	<p>Eles representam todo tipo de artefato gerado ou consumido durante o processo. Eles também têm o objetivo de adicionar informações às atividades.</p>
 <p><i>Work Definition</i></p>	 <p><i>Embedded Sub-Process</i></p>	<p><i>Work Definition</i> representa (agrupa) um conjunto de atividades do processo, assim como o <i>Embedded Sub-Process</i> contém várias atividades contidas nele.</p>
 <p><i>Guidance</i></p>	 <p><i>Text Annotation</i></p>	<p>O <i>Guidance</i> representa itens que provêm informações mais detalhadas sobre os elementos, assim como é a função do <i>Text Annotation</i>.</p>
 <p><i>Activity</i></p>	 <p><i>Task</i></p>	<p>Uma <i>Activiy</i> representa uma atividade atômica, que não pode ser dividida em outras atividades, assim como uma <i>Task</i>.</p>
 <p><i>Role Use</i></p>	 <p><i>Pool</i></p>  <p><i>Lane</i></p>	<p>O <i>Role Use</i> representa a entidade executora das atividades do processo, e o <i>Pool</i> engloba várias <i>Tasks</i>, representando assim um participante do processo. O <i>Lane</i> é simplesmente uma subdivisão do <i>Pool</i>.</p>

 <i>Role Definition</i>	 <i>Pool</i>  <i>Lane</i>	<p>O <i>Role Definition</i> representa o papel do processo que executa as atividades, e o <i>Pool</i> engloba várias <i>Tasks</i>, representando assim um participante do processo. O <i>Lane</i> é simplesmente uma subdivisão do <i>Pool</i>.</p>
 <i>Process Package</i>	 <i>Independent Sub-Process</i>	<p>O <i>Process Package</i> representa um <i>container</i> contendo elementos de definição do processo, como artefatos, papéis e atividades. E o <i>Independent Sub-Process</i> também pode englobar esses elementos de definição de processo.</p>
 <i>Kind (instance)</i>	 <i>Embedded Sub-Process</i>	<p><i>Kind</i> representa um conjunto de atividades qualificando-as como uma fase do processo, portanto também pode ser mapeada para <i>Sub-Process</i>.</p>
 <i>Process</i>	<p>Não se aplica.</p>	<p>O <i>Process</i> representa um processo completo, independente e auto-contido. Não existe nenhuma notação para representá-lo no BPMN, ele é representado pelo diagrama de processo como um todo.</p>
 <i>Document</i>	 <i>Data Object</i>	<p>O <i>Document</i> representa um artefato produzido ou consumido por uma atividade durante o processo, assim como o <i>Data Object</i> no BPMN representa artefatos do processo.</p>

3. ProModeller: Ferramenta para Modelagem de Processo de Software

O ProModeller é uma ferramenta de apoio à modelagem de processos de software que foi desenvolvida por alunos do CIn, Centro de Informática da UFPE, no ano de 2008. Ela foi implementada utilizando a linguagem de programação JAVA com as ferramentas Eclipse e NetBeans. Ela faz o uso de ontologia de definição de processos de software definida por [FALBO, 1998] para realizar a modelagem. Para poder desenhar os elementos na tela foi utilizado a API (*Application Programming Interface*) JGraph – Java Graph – que permite a modelagem de elementos em grafos.

Esta ferramenta contribui também com outra ferramenta chamada ImPPros – Ambiente de Implantação Progressiva de Processo de Software – que tem como objetivo fornecer um apoio automatizado por meio de um ambiente capaz de suportar as fases de Definição, Simulação, Execução e Avaliação do projeto de software. O ProModeller dá apoio ao ImPPros com a modelagem visual do processo de software.

O ProModeller é capaz de modelar e apresentar dois tipos de diagramas referentes ao processo, o diagrama de pacotes (apresentado na figura 3.1) e o diagrama de seqüência (apresentado na figura 3.2).



Figura 3.1- Diagrama de Pacotes do ProModeller.

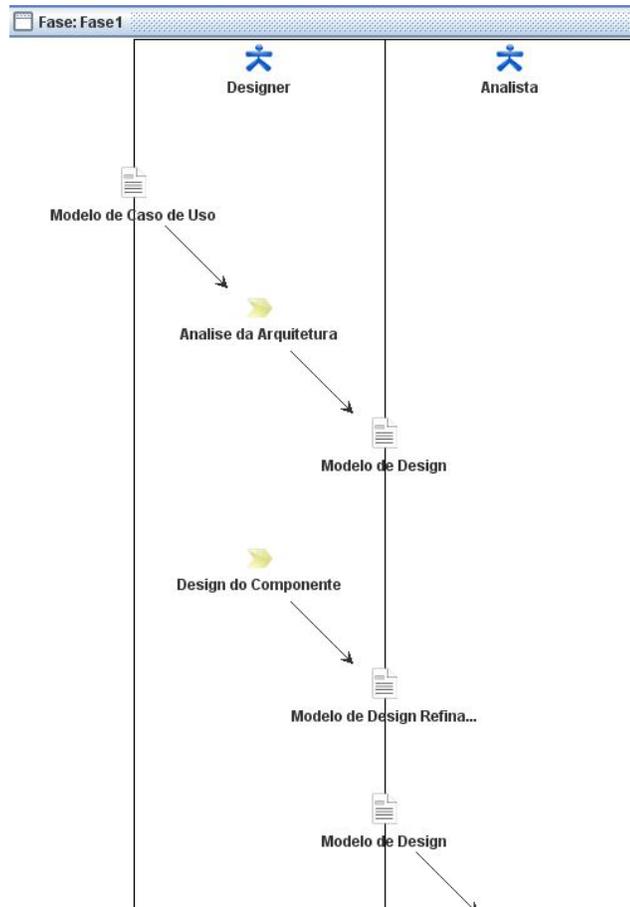


Figura 3.2- Diagrama de Atividades do ProModeller.

3.1. Serviços do ProModeller

O ProModeller fornece vários serviços que apoiam a criação, manutenção e execução do processo. Dentro das suas principais funcionalidades podemos citar:

- **Criar Modelagem:** permite criar uma modelagem do início, ou seja, partindo do “zero” o usuário pode construir todo o processo na ferramenta.
- **Salvar Modelagem:** depois de criada a modelagem do processo na ferramenta, é possível salvar essa modelagem em um arquivo texto utilizando a ontologia de processo.
- **Abrir Modelagem:** a ferramenta também oferece a função de carregar uma modelagem salva e assim poder visualizá-la e/ou editá-la.

- **Merge de Macro-Atividades:** essa funcionalidade proporciona ao usuário a possibilidade dele realizar uma junção, na verdade um merge, de duas *Work Definitions*, que aqui foi denominada de macro-atividade (pois engloba outras atividades dentro dela). Na hora do merge, o usuário escolhe o nome da nova atividade resultante, os insumos e produtos (*Document*), os recursos (*Role Definition*) e o novo guideline (*Guidance*).
- **Abrir Processos do ImPPros:** o ProModeller, como dito anteriormente, apóia o ImPPros. Assim os processos que o ImPPros gera podem ser carregados pelo ProModeller permitindo ao usuário a sua visualização e edição.
- **Salvar Meta-Arquivo do Processo:** a ferramenta foi projetada para estender a ontologia de definição de processo definida por [FALBO, 1998], pois esta não dá suporte para adicionar informações mais detalhadas sobre cada entidade do processo. Assim, é possível adicionar informações nas seguintes entidades:
 - **Atividade (*Activity*) e Macro-Atividade (*Work Definition*):** pode-se definir o Propósito da atividade e uma Descrição da mesma;
 - **Recurso (*Role Definition*):** pode-se definir as Habilidades do recurso e uma Descrição do mesmo;
 - **Artefato (*Document*):** pode-se definir o Propósito do artefato e uma Descrição do mesmo.
- **Visualizar e Verificar Ontologia:** e por último, com o ProModeller é possível visualizar a ontologia do processo assim como fazer uma verificação da consistência desta ontologia.

3.2. Estrutura Interna do ProModeller

O ProModeller foi desenvolvido baseado numa arquitetura dividida em camadas e foram utilizados os seguintes padrões de projeto: *Singleton* e *Bridge*.

O padrão *Singleton* define que uma classe só pode ter uma instancia no sistema, ou seja, depois que a classe foi criada não se pode criar outra instância dela. Assim, foi utilizado esse padrão nas classes de controle do pacote de negócios para garantir que todo o sistema esteja sob o controle de um único objeto.

O padrão *Bridge* define que uma interface possa variar independente das suas implementações, ou seja, essa interface desaclopa a abstração da implementação de modo que elas possam variar. Este padrão foi utilizado nas classes dos pacotes GUI, negócios e persistência para facilitar o desenvolvimento e manutenção da ferramenta.

A figura 3.3 abaixo mostra o diagrama de pacotes do ProModeller. O pacote GUI contém as classes de interface com o usuário e pela sua complexidade e extensão foi dividido em sub-pacotes que serão apresentados mais a frente. O pacote negócios contém as classes de controle de funções básicas do sistema como Criar, Abrir, Salvar e Validar Modelagem. O pacote negócios.exceções contém as classes de exceções do pacote negócios. O pacote persistência contém as classes que realizam funções de escrever e ler o processo em arquivos textos. O pacote persistência.exceções contém as classes de exceções do pacote persistência. E por fim, o pacote básicas contém as classes de entidade do sistema, que modelam os objetos utilizados na implementação.

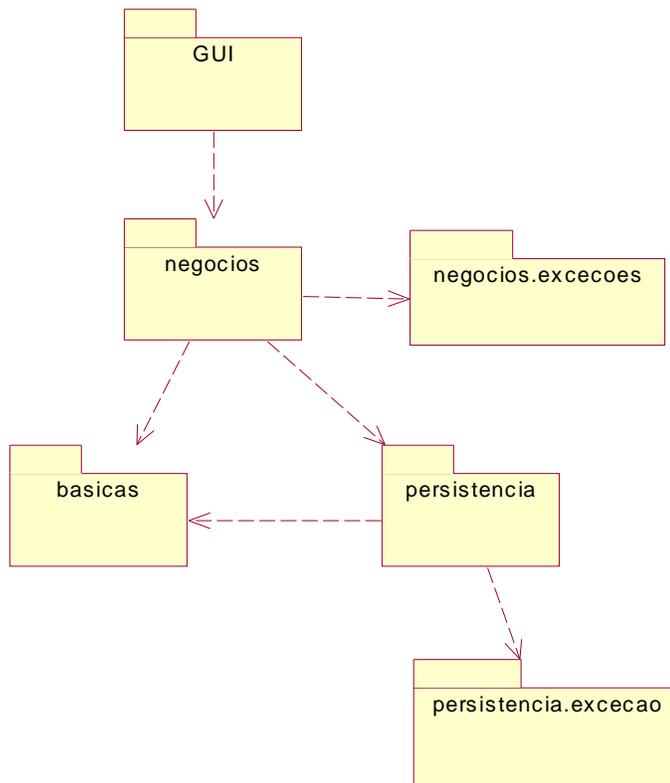


Figura 3.3- Diagrama de Pacotes do ProModeller.

A figura 3.4 abaixo apresenta os sub-pacotes do pacote GUI. O pacote `gui.merge` contém as classes de interface para realizar a funcionalidade de realizar merge de macro-atividades. O pacote `gui.novamod` contém as classes de interface que guiam o usuário para criar uma nova modelagem. O pacote `gui.pe` contém as classes de interface que apresentam a modelagem dos processos especializados. E por último, o pacote `gui.pi` contém as classes de interface que apresentam a modelagem dos processos instanciados e está dividido em dois pacotes: o `gui.pi.pacote` que contém as classes que apresentam o diagrama de pacotes do processo e o `gui.pi.ativ` que contém as classes que apresentam o diagrama de atividade.

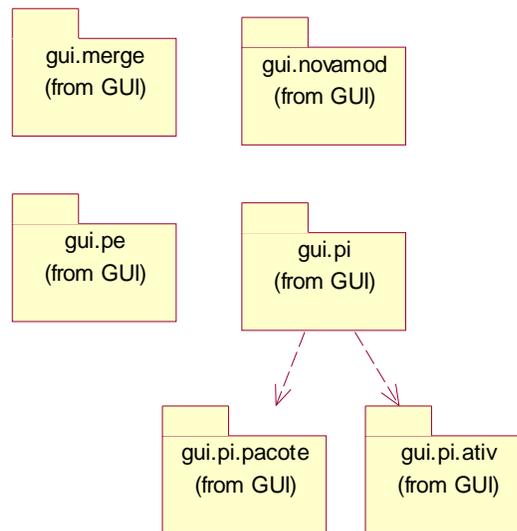


Figura 3.4- Diagrama de Pacotes do Pacote GUI do RroModeller.

4. Implementação da Visão BPMN no ProModeller

O ProModeller somente fornece suporte à modelagem de processos de software, não permitindo a modelagem de processos de negócios, assim foi adicionada a ele a funcionalidade de Visão BPMN. Essa funcionalidade permite que o usuário faça a modelagem toda em SPEM e a partir dela gere a modelagem na notação BPMN.

Esta nova funcionalidade traz as seguintes vantagens para a empresa:

- O ganho de expressividade obtido por SPEM na modelagem de processos de software tem como efeito colateral uma diminuição na capacidade de prover entendimento do modelo para pessoas que não são da área de engenharia de software. Isso não acontece com os modelos BPMN, visto que é um padrão para modelagem de processos de negócio em geral e tem como um dos principais objetivos fornecer uma notação compreensível por todos os envolvidos no processo.
- O BPMN permite o mapeamento dos seus modelos para linguagens de execução de processos de negócios, assim a organização pode fazer simulações do seu desempenho.
- A especificidade dos elementos de processo fornecido por SPEM dificulta ou até mesmo impede que se consiga integrar os modelos de processos de software com os demais modelos de processo de negócio. Já processos de software modelados através do BPMN podem ser integrados com os demais modelos de processo de negócio de uma organização permitindo assim uma integração organizacional e o alinhamento dos objetivos com as metas globais da instituição.

4.1. Funcionalidade de Salvar no Formato BPMN

Além de gerar a modelagem no padrão BPMN, também foi implementada a funcionalidade de Salvar Formato BPMN. Essa funcionalidade utiliza o padrão XPDL (*XML Process Definition Language*) que é gerenciado pela XfMC (*Workflow Management*

Coalition) [WfMC, 2008] que é uma associação internacional com mais de 300 associados que trabalham em cima do seguimento de *Workflow* desde 1993.

O XPDL é um padrão utilizado por muitas ferramentas de modelagem de processos de negócio e isso permite que haja uma comunicação entre essas ferramentas com o XPDL. Ele é um padrão XML que descreve regras de processos de negócios e está baseado em um conjunto de atividades relacionadas entre si através de transições.

Assim, foi utilizado esse padrão para realizar o intercâmbio de processos entre o ProModeller e outras ferramentas que suportem o formato XPDL.

4.2. Modelagem da Visão BPMN e Salvar Formato BPMN

A Visão BPMN foi modelada em dois novos pacotes no sistema: o pacote `gui.pi.bpmn` e o pacote `gui.pi.bpmn.util` como mostra a figura 4.1 abaixo. No pacote `gui.pi.bpmn` estão contidas as classes de interface que apresentam ao usuário a modelagem no formato BPMN. O pacote `gui.pi.bpmn.util` contém as classes que dão apoio à maneira como os elementos são plotados na tela.

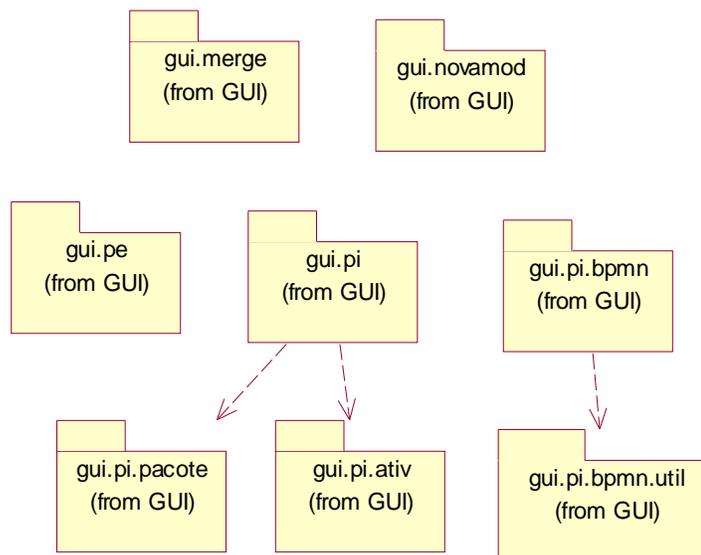


Figura 4.1- Novo Diagrama de Sub-Pacotes do Pacote GUI.

A funcionalidade de Salvar no Formato BPMN foi implementada adicionando as seguintes classes como mostra a figura 4.2.

- **ProcessoBPMN** no pacote *basicas* - modela os elementos do processo (atividade, recurso, artefato e procedimento) e contém atributos como *tipo*, *nome*, *id*, *coordenadaX*, *coordenadaY*, *comprimento*, *altura*, etc.;
- **PersistenciaBPMN** no pacote *persistência* – realiza uma varredura no processo para escrever em um novo arquivo texto a modelagem no formato XPDL;
- **ControladorModelagemBPMN** no pacote *negócios* – controla a árvore do processo fornecendo opções de adicionar e remover elementos do processo.

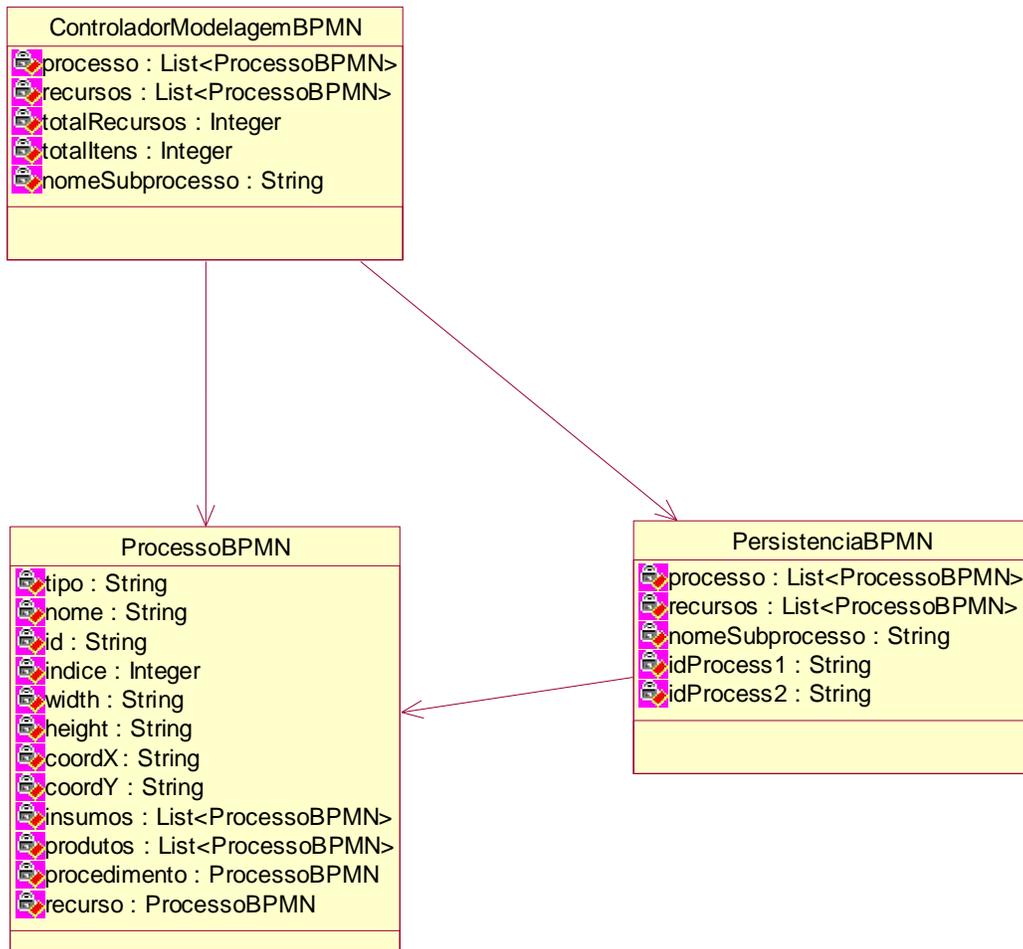


Figura 4.2- Implementação da Funcionalidade Salvar Formato BPMN.

4.3. Exemplo de Uso da Visão BPMN

Como o intuito dessa funcionalidade é somente gerar a modelagem BPMN a partir do SPEM, não é permitido ao usuário modificar a modelagem BPMN, ou seja, ele somente pode visualizá-la. A figura 4.3 mostra um exemplo da modelagem BPMN gerada pelo ProModeller.

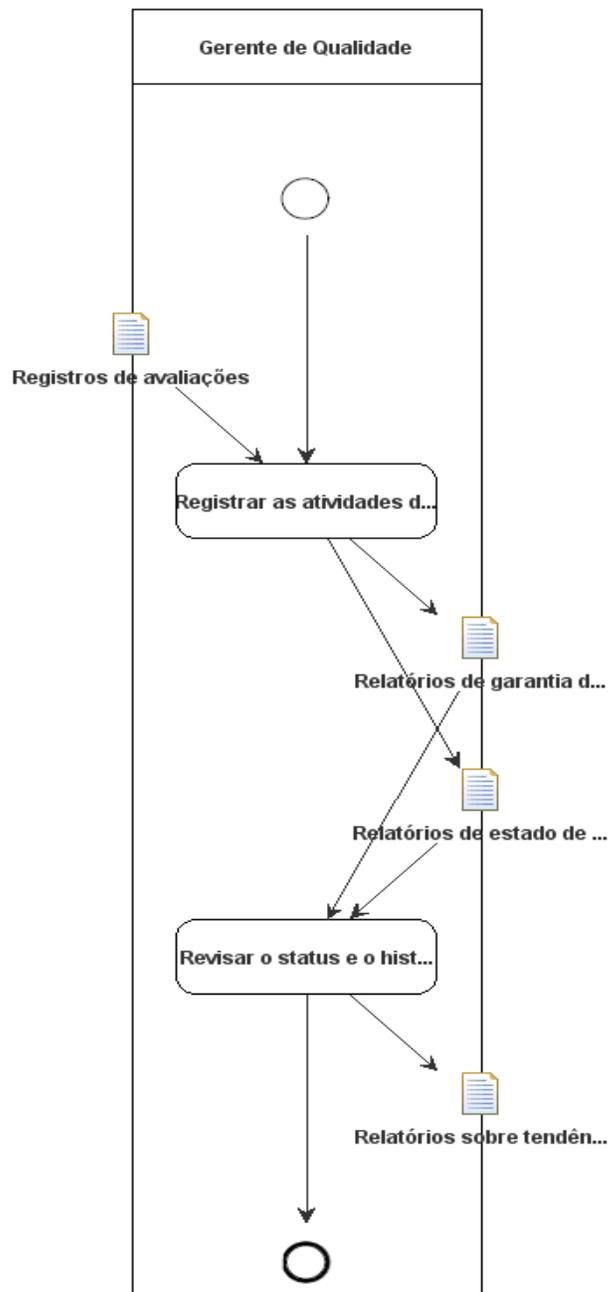


Figura 4.3- Exemplo de Modelagem BPMN no ProModeller.

A figura 4.4 mostra o arquivo XPDL gerado a partir do processo da figura 4.3.

```
<?xml version="1.0" encoding="utf-8"?>
<Package xmlns="http://www.wfmc.org/2008/XPDL2.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" Id="bdd3138d-724f-44d0-8d79-f549ad6e7010" Name="Diagram 1"
OnlyOneProcess="false">
<PackageHeader>
<XPDLVersion>2.1</XPDLVersion>
<Vendor>BizAgi Process Modeler.</Vendor>
<Created>Thu Jun 11 18:54:09 GMT-03:00 2009</Created>
<Description>Diagram 1</Description>
<Documentation />
</PackageHeader>
<RedefinableHeader>
<Author />
<Version />
<Countrykey>PE</Countrykey>
</RedefinableHeader>
<ExternalPackages />
<Participants />
<Pools>
<Pool Id="bfb7c5f8-12ae-4f7d-a4bb-59d41fd19e9c" Process="processo01" BoundaryVisible="false">
<Lanes />
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="0" Width="0" BorderColor="-16777216" FillColor="-1">
<Coordinates XCoordinate="0" YCoordinate="0" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
</Pool>
<Pool Id="processo01-pool" Name="Estabelecer Registros" Process="processo02" BoundaryVisible="true">
<Lanes>
<Lane Id="Gerente de Qualidade1" Name="Gerente de Qualidade" ParentPool="processo01-pool">
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="260" Width="1600" BorderColor="-11513776" FillColor="-1315861">
<Coordinates XCoordinate="50" YCoordinate="0" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<Documentation />
<ExtendedAttributes />
</Lane>
</Lanes>
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="260" Width="1650" BorderColor="-16777216" FillColor="-1">
<Coordinates XCoordinate="37" YCoordinate="30" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
</Pool>
</Pools>
<MessageFlows />
<Associations>
<Association Id="association2" Source="Registros de avaliaco3" Target="Registrar as atividades de garantia da qualidade do processo
e do produto4" Name="">
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="500" YCoordinate="175" />
<Coordinates XCoordinate="650" YCoordinate="205" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
<ExtendedAttributes />
</Association>
<Association Id="association4" Source="Registrar as atividades de garantia da qualidade do processo e do produto4" Target="Relatorios
de garantia da qualidade5" Name="">
<ConnectorGraphicsInfos>
```

```

<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="740" YCoordinate="205" />
<Coordinates XCoordinate="850" YCoordinate="175" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
<ExtendedAttributes />
</Association>
<Association Id="association5" Source="Registrar as atividades de garantia da qualidade do processo e do produto4" Target="Relatorios
de estado de acoes corretivas6" Name="">
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="740" YCoordinate="205" />
<Coordinates XCoordinate="1050" YCoordinate="175" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
<ExtendedAttributes />
</Association>
<Association Id="association7" Source="Revisar o status e o historico das atividades7" Target="Relatorios sobre tendencias em relacao a
qualidade8" Name="">
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="1340" YCoordinate="205" />
<Coordinates XCoordinate="1450" YCoordinate="175" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
<ExtendedAttributes />
</Association>
</Associations>
<Artifacts>
<Artifact Id="Registros de avaliacoes3" ArtifactType="DataObject">
<DataObject Id="Registros de avaliacoes3" Name="Registros de avaliacoes" State="">
<RequiredForStartSpecified>false</RequiredForStartSpecified>
<ProducedAtCompletionSpecified>false</ProducedAtCompletionSpecified>
</DataObject>
</NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="40" Width="50" BorderColor="-10066330" FillColor="-986896">
<Coordinates XCoordinate="450" YCoordinate="150" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<Documentation />
</Artifact>
<Artifact Id="Relatorios de garantia da qualidade5" ArtifactType="DataObject">
<DataObject Id="Relatorios de garantia da qualidade5" Name="Relatorios de garantia da qualidade" State="">
<RequiredForStartSpecified>false</RequiredForStartSpecified>
<ProducedAtCompletionSpecified>false</ProducedAtCompletionSpecified>
</DataObject>
</NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="40" Width="50" BorderColor="-10066330" FillColor="-986896">
<Coordinates XCoordinate="850" YCoordinate="150" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<Documentation />
</Artifact>
<Artifact Id="Relatorios de estado de acoes corretivas6" ArtifactType="DataObject">
<DataObject Id="Relatorios de estado de acoes corretivas6" Name="Relatorios de estado de acoes corretivas" State="">
<RequiredForStartSpecified>false</RequiredForStartSpecified>
<ProducedAtCompletionSpecified>false</ProducedAtCompletionSpecified>
</DataObject>
</NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="40" Width="50" BorderColor="-10066330" FillColor="-986896">
<Coordinates XCoordinate="1050" YCoordinate="150" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<Documentation />
</Artifact>

```

```

<Artifact Id="Relatorios sobre tendencias em relacao a qualidade8" ArtifactType="DataObject">
<DataObject Id="Relatorios sobre tendencias em relacao a qualidade8" Name="Relatorios sobre tendencias em relacao a qualidade"
State="">
<RequiredForStartSpecified>>false</RequiredForStartSpecified>
<ProducedAtCompletionSpecified>>false</ProducedAtCompletionSpecified>
</DataObject>
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="40" Width="50" BorderColor="-10066330" FillColor="-986896">
<Coordinates XCoordinate="1450" YCoordinate="150" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<Documentation />
</Artifact>
</Artifacts>
<WorkflowProcesses>
<WorkflowProcess Id="processo01" Name="Main Process">
<ProcessHeader>
<Created>Thu Jun 11 18:54:09 GMT-03:00 2009</Created>
<Description />
</ProcessHeader>
<RedefinableHeader>
<Author>BizAgi Process Modeler.</Author>
<Version>1.0</Version>
<Countrykey>PE</Countrykey>
</RedefinableHeader>
<ActivitySets />
<Activities />
<Transitions />
<ExtendedAttributes />
</WorkflowProcess>
<WorkflowProcess Id="processo02" Name="Estabelecer Registros">
<ProcessHeader>
<Created>Thu Jun 11 18:54:09 GMT-03:00 2009</Created>
<Description />
</ProcessHeader>
<RedefinableHeader>
<Author>BizAgi Process Modeler.</Author>
<Version>1.0</Version>
<Countrykey>PE</Countrykey>
</RedefinableHeader>
<ActivitySets />
<Activities>
<Activity Id="inicio2">
<Event>
<StartEvent Trigger="None" />
</Event>
<Documentation />
<ExtendedAttributes />
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="30" Width="30" BorderColor="-10311914" FillColor="-1638505">
<Coordinates XCoordinate="250" YCoordinate="145" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<IsForCompensationSpecified>>false</IsForCompensationSpecified>
</Activity>
<Activity Id="Registrar as atividades de garantia da qualidade do processo e do produto4" Name="Registrar as atividades de garantia da
qualidade do processo e do produto">
<Implementation>
<Task />
</Implementation>
<Performers />
<Documentation />
<ExtendedAttributes />
<OutputSets>
<OutputSet>

```

```

<Output ArtifactId="Registros de avaliacoes3" />
<Output ArtifactId="Relatorios de garantia da qualidade5" />
<Output ArtifactId="Relatorios de estado de acoes corretivas6" />
</OutputSet>
</OutputSets>
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="60" Width="90" BorderColor="-16553830" FillColor="-1249281">
<Coordinates XCoordinate="650" YCoordinate="160" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<IsForCompensationSpecified>>false</IsForCompensationSpecified>
</Activity>
<Activity Id="Revisar o status e o historico das atividades7" Name="Revisar o status e o historico das atividades">
<Implementation>
<Task />
</Implementation>
<Performers />
<Documentation />
<ExtendedAttributes />
<OutputSets>
<OutputSet>
<Output ArtifactId="Relatorios sobre tendencias em relacao a qualidade8" />
</OutputSet>
</OutputSets>
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="60" Width="90" BorderColor="-16553830" FillColor="-1249281">
<Coordinates XCoordinate="1250" YCoordinate="160" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<IsForCompensationSpecified>>false</IsForCompensationSpecified>
</Activity>
<Activity Id="final9">
<Event>
<EndEvent />
</Event>
<Documentation />
<ExtendedAttributes />
<NodeGraphicsInfos>
<NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="30" Width="30" BorderColor="-6750208" FillColor="-1135958">
<Coordinates XCoordinate="1650" YCoordinate="145" />
</NodeGraphicsInfo>
</NodeGraphicsInfos>
<IsForCompensationSpecified>>false</IsForCompensationSpecified>
</Activity>
</Activities>
<Transitions>
<Transition Id="transition0" From="inicio2" To="Registrar as atividades de garantia da qualidade do processo e do produto4" Name="">
<Condition />
<ExtendedAttributes />
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="280" YCoordinate="160" />
<Coordinates XCoordinate="650" YCoordinate="205" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
</Transition>
<Transition Id="transition1" From="Registrar as atividades de garantia da qualidade do processo e do produto4" To="Revisar o status e o historico das atividades7" Name="">
<Condition />
<ExtendedAttributes />
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="740" YCoordinate="205" />
<Coordinates XCoordinate="1250" YCoordinate="205" />
</ConnectorGraphicsInfo>

```

```
</ConnectorGraphicsInfos>
</Transition>
<Transition Id="transition2" From="Revisar o status e o historico das atividades7" To="final9" Name="">
<Condition />
<ExtendedAttributes />
<ConnectorGraphicsInfos>
<ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="0">
<Coordinates XCoordinate="1340" YCoordinate="205" />
<Coordinates XCoordinate="1650" YCoordinate="160" />
</ConnectorGraphicsInfo>
</ConnectorGraphicsInfos>
</Transition>
</Transitions>
<ExtendedAttributes />
</WorkflowProcess>
</WorkflowProcesses>
<ExtendedAttributes />
</Package>
```

Figura 4.4- Arquivo XPDL do Processo da Figura 3.4.

5. Exemplo de Uso do ProModeller: Mapeamento de SPEM em BPMN

Neste capítulo apresentaremos um exemplo de uso da funcionalidade de conversão de um processo de software modelado em SPEM na visão correspondente em BPMN. Tendo como base o mapeamento realizado no capítulo 2, mostraremos um caso real de modelagem de processo onde será utilizado o ProModeller para fazer o mapeamento de SPEM para o BPMN.

Para isso foi escolhida a área de processo Garantia de Qualidade de Processo e Produto do CMMI (*Capability Maturity Model Integration*) [CMMI, 2008]. Assim, apresentaremos a seguir utilização do ProModeller para modelar esse processo.

5.1. O Processo – Garantia de Qualidade de Processo e Produto

O propósito da Garantia de Qualidade de Processo e Produto é fornecer aos envolvidos no projeto, tanto gerentes quanto desenvolvedores, uma visão clara sobre o uso dos processos e seus produtos de trabalhos associados [CMMI, 2008]. Dentre os principais objetivos desse processo podemos citar:

- Avaliar objetivamente os processos, produtos de trabalho e serviços executados em relação às descrições de padrões, processos e procedimentos aplicáveis.
- Garantir que possíveis não-conformidades do processo sejam identificadas, documentadas e tratadas;
- Fornecer um *feedback* sobre os resultados das atividades de garantia da qualidade para a equipe de projeto e os gerentes.

O CMMI trabalha com Metas Específicas e Práticas Específicas. Ambas se aplicam a uma área de processo particular e estão relacionadas à dimensão do processo. As Metas Específicas representam os principais objetivos da área de processo, são definições de alto nível que englobam as Práticas Específicas. E estas, representam os passos para atender as Metas Específicas.

Além disso, as Práticas Específicas ainda são divididas em subpráticas que geram e consomem os artefatos do processo. A Garantia de Qualidade de Processo e Produto está estruturada da seguinte forma:

- Meta Específica 1 – Avaliar Objetivamente Processos e Produtos de Trabalho:
 - Prática Específica 1.1 – Avaliar Objetivamente os Processos;
 - Prática Específica 1.2 – Avaliar Objetivamente Produtos de Trabalho e Serviços;
- Meta Específica 2 – Fornecer um Entendimento Objetivo:
 - Prática Específica 2.1 – Comunicar e Garantir a Solução de Não-Conformidades;
 - Prática Específica 2.2 – Estabelecer Registros.

5.2. Modelagem do Processo em SPEM

Para demonstração do uso da ferramenta somente foi modelada a Prática Específica 1.1 por questões de simplicidade e visto que as outras práticas seguem o mesmo processo de construção. A Prática Específica 1.1 foi modelada como mostra a tabela 5.1.

Tabela 5.1- Critério da Modelagem do Processo.

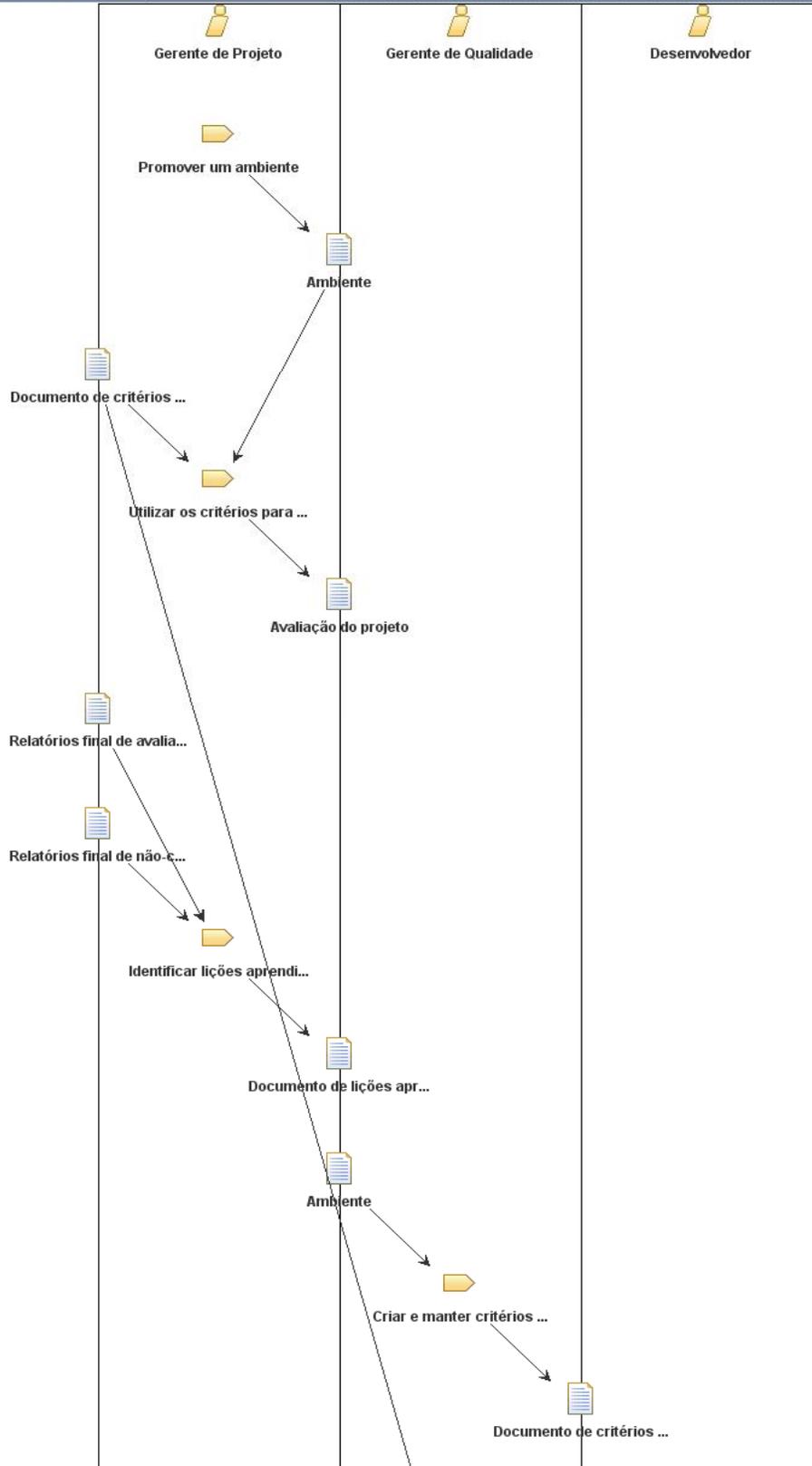
Avaliar Objetivamente os Processos – Critérios de Modelagem	
CMMI	SPEM
Meta Específica 1 – Avaliar Objetivamente Processos e Produtos de Trabalho	Fase
Prática Específica 1.1 – Avaliar Objetivamente os Processos	Macro-Atividade
Subprática 1.1.1 – Promover um Ambiente	Atividade
Subprática 1.1.2 – Criar e Manter Critérios de Avaliação	Atividade
Subprática 1.1.3 – Utilizar os Critérios para Avaliar a Aderência do Projeto	Atividade
Subprática 1.1.4 – Identificar Não-Conformidades na Avaliação	Atividade
Subprática 1.1.5 – Identificar Lições Aprendidas	Atividade

A especificação do CMMI não deixa claro todos os artefatos produzidos e utilizados durante essas atividades, ele só sugere os principais artefatos que cada Prática Específica deve gerar. Assim, mostraremos na tabela 5.2 todas as atividades relacionadas com seus produtos específicos.

Tabela 5.2- Relação dos Artefatos das Atividades do Processo.

Atividade	Insumo	Produto
Subprática 1.1.1 – Promover um Ambiente	X	- Ambiente
Subprática 1.1.2 – Criar e Manter Critérios de Avaliação	- Ambiente	- Documento de Critérios de Avaliação
Subprática 1.1.3 – Utilizar os Critérios para Avaliar a Aderência do Projeto	- Documento de Critérios de Avaliação	- Avaliação do Projeto
Subprática 1.1.4 – Identificar Não-Confirmandas na Avaliação	- Documento de Critérios de Avaliação	- Relatório de Não-Conformidades dos Processos
Subprática 1.1.5 – Identificar Lições Aprendidas	- Relatório Final de Avaliação dos Processos - Relatório final de não-conformidades dos processos	- Documento de Lições Aprendidas

Agora que já está decidido como será realizada a modelagem da área de processo Garantia de Qualidade de Processo e Produto, mostraremos na figura 5.1 essa modelagem no ProModeller.



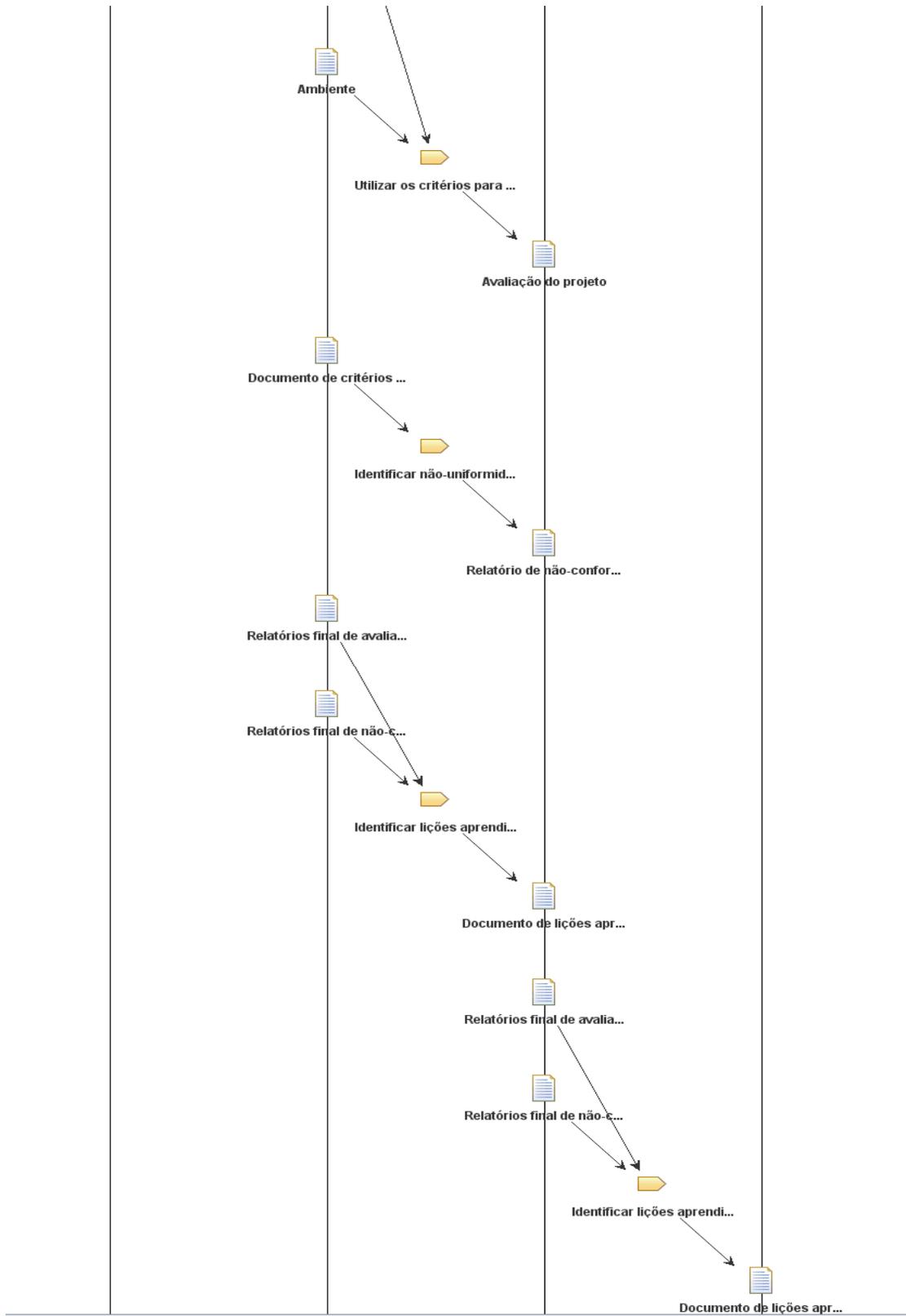
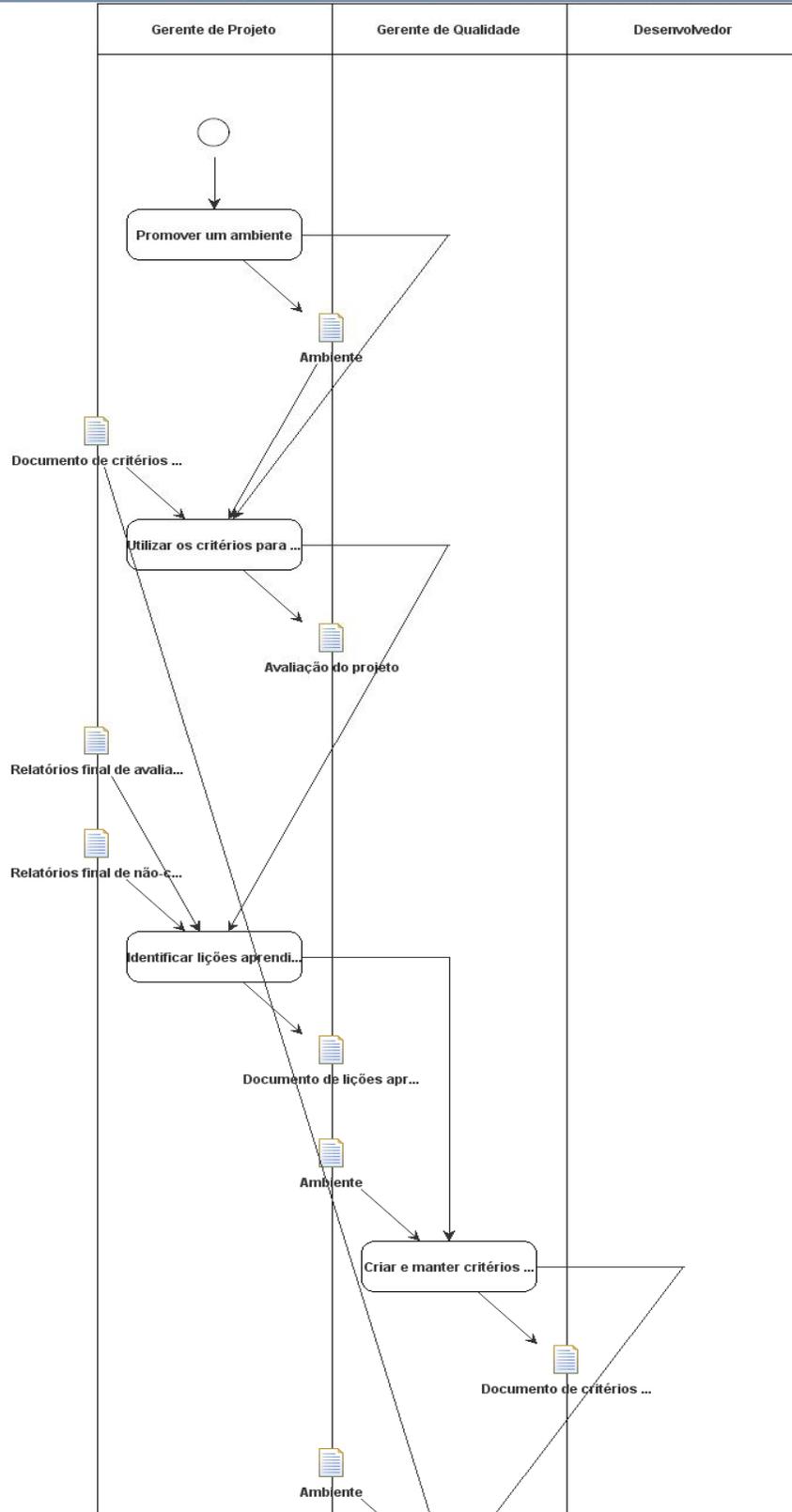


Figura 5.1- Modelagem do Processo em SPEM.

5.3. Modelagem do Processo em BPMN

Agora mostraremos o resultado, na figura 5.2, da Visão BPMN do ProModeller aplicada no processo Garantia de Qualidade de Processo e Produto.



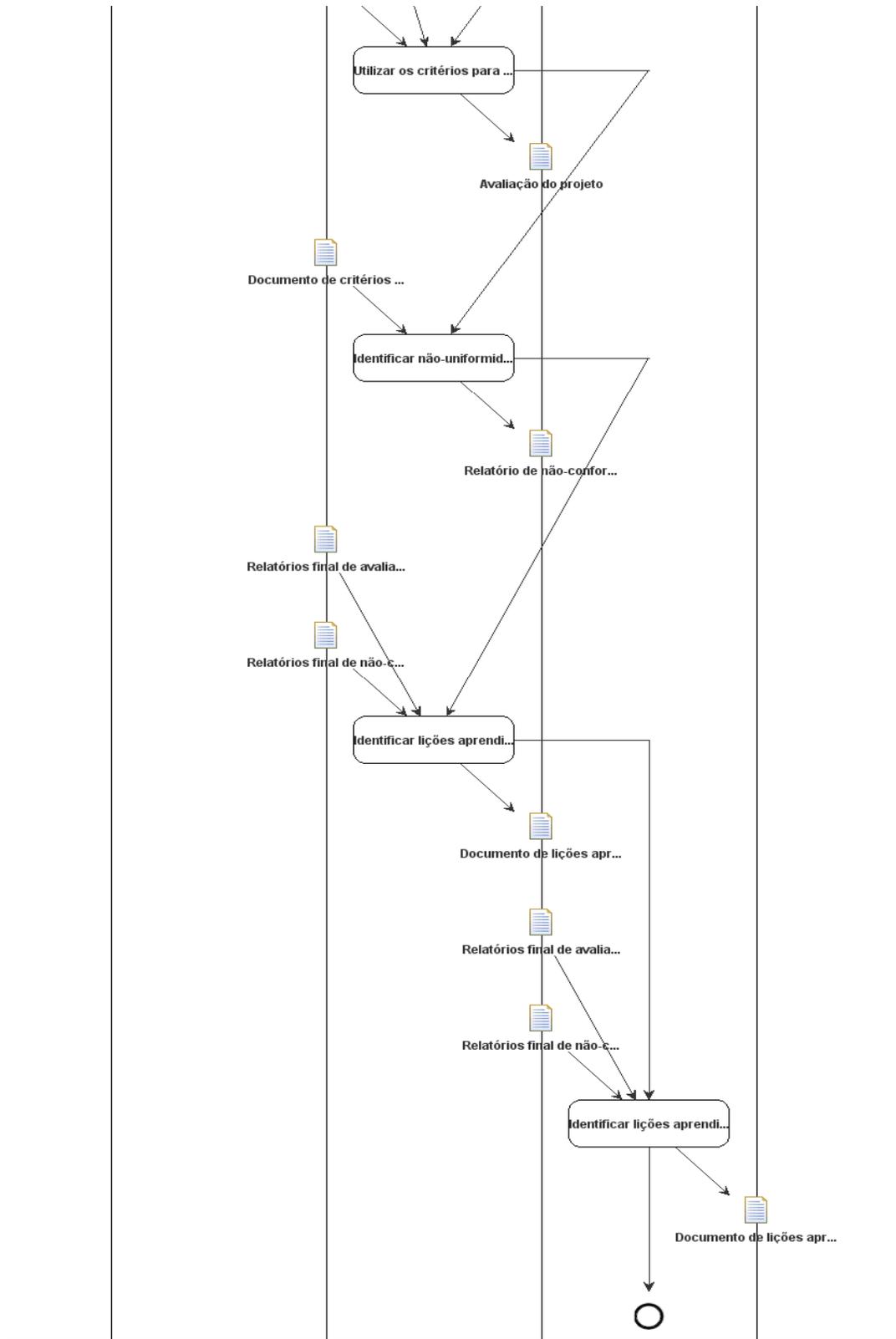
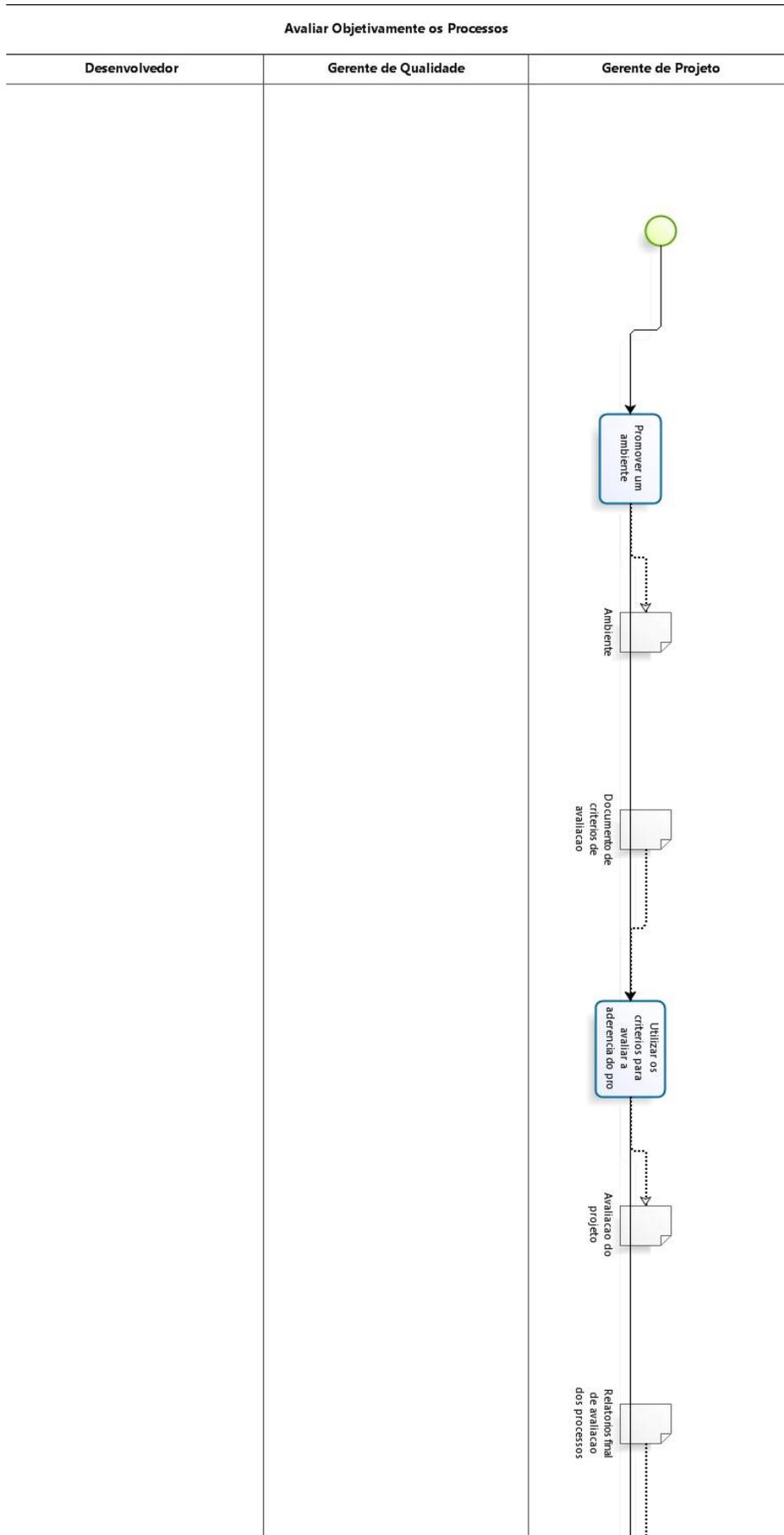
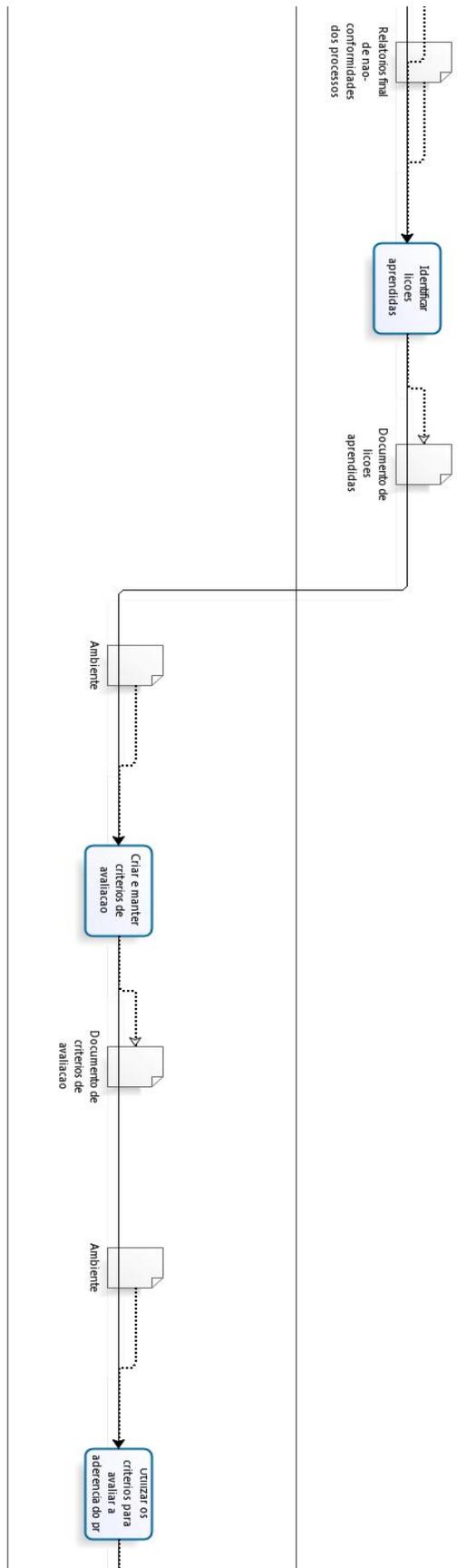


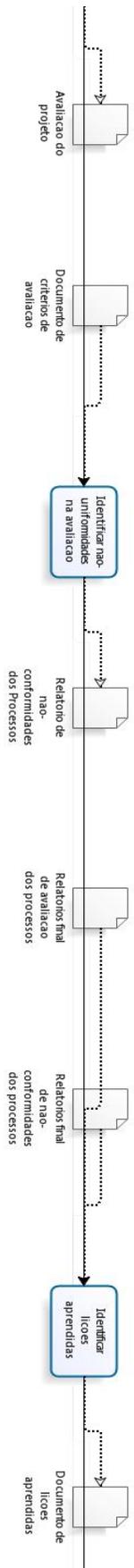
Figura 5.2- Modelagem do Processo em BPMN.

Depois de gerada a modelagem no formato BPMN no ProModeller, utilizaremos uma ferramenta auxiliar para na verificação do processo BPMN. A ferramenta *BizAgi Process Modeler* é uma ferramenta gratuita, muito simples e de fácil utilização [BizAgi, 2009]. Ela oferece as funcionalidades de importar e exportar arquivos XPDL, assim podemos verificar nosso processo.

Após salvar a modelagem no Formato BPMN no ProModeller, importamos o arquivo XPDL na ferramenta BizAgi e o resultado, semelhante ao que o ProModeller gerou, é mostrado na figura 5.3 abaixo.







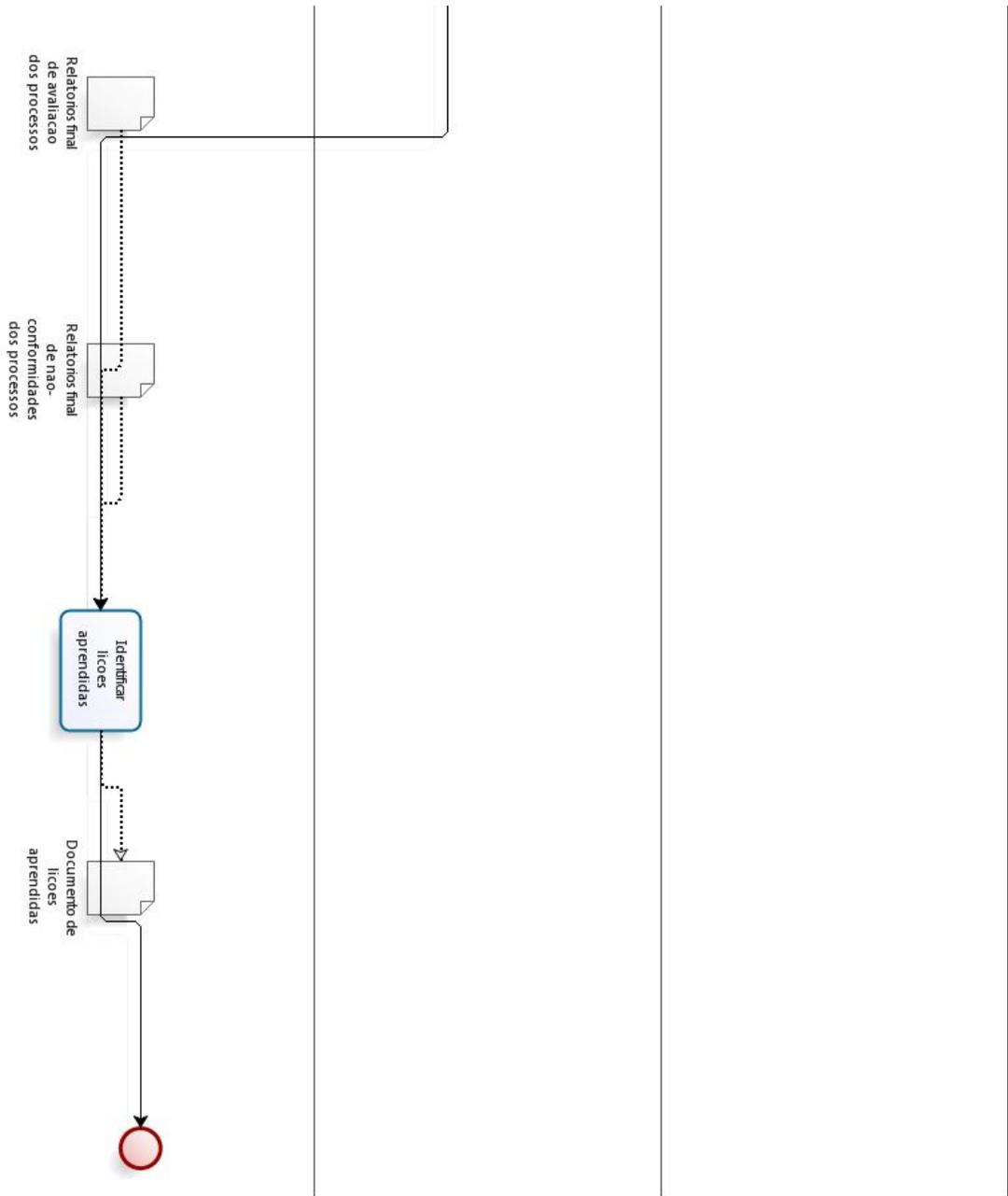


Figura 5.3- Modelagem do Processo BPMN na Ferramenta BizAgi.

6. Conclusão

Foi apresentada neste trabalho uma comparação entre dois importantes padrões para modelagem de processos, o SPEM e o BPMN, aplicados para modelagem de processos de software. Para isso, foi fundamentada uma abordagem que representa o processo de software como um processo de negócio dentro da organização. Isso possibilita a integração do processo de software com os demais processos organizacionais e um alinhamento com os objetivos e metas globais da empresa.

Para sustentar essa abordagem um pouco diferente em que se modela processo de software como processo de negócio, primeiramente apresentamos a definição de modelagem de processos assim como os padrões SPEM e BPMN. Além disso, fizemos o mapeamento dos elementos da modelagem do SPEM para os do BPMN.

Em seguida, apresentamos a ferramenta ProModeller. Com ela o usuário realiza modelagem de processos de software com SPEM. Essa ferramenta fornece dois tipos de diagramas, o de atividades e o de pacotes, que são baseados em ontologia de processo de software.

Depois foi implementada uma nova funcionalidade no ProModeller para mapear os diagramas de SPEM para BPMN. Com essa nova funcionalidade é possível realizar toda modelagem do processo em SPEM e no final obter o resultado em BPMN.

E por último, apresentamos um exemplo de uso do ProModeller através da modelagem de um subprocesso do CMMI. Assim foi realizada toda a modelagem em SPEM e com a nova funcionalidade do ProModeller foi possível ver o mesmo subprocesso em BPMN como também importar o subprocesso em outra ferramenta utilizando o padrão XPDL.

6.1. Trabalhos Futuros

Após a conclusão deste trabalho, é possível identificar algumas possibilidades de trabalhos futuros. Uma delas seria a implementação do mapeamento inverso, ou seja, mapear elementos do BPMN para SPEM. Assim seria possível às organizações que já trabalham com BPMN a visualização dos seus processos no formato do SPEM.

Outra opção seria complementar a ferramenta ProModeller com a funcionalidade de poder criar e editar modelos BPMN. Assim seria possível realizar correções no modelo diretamente no formato BPMN.

6.2. Considerações Finais

Este trabalho teve como objetivo apresentar à comunidade de engenharia de software informações sobre dois importantes padrões de modelagem de processos como também apresentar uma nova ferramenta de modelagem. Espera-se que com o estudo e a ferramenta apresentada seja possível melhorar o ambiente das organizações no que tange a modelagem de processos. E assim elas possam trabalhar com os dois padrões sem problemas tendo em mãos uma ferramenta de modelagem e mapeamento de padrões que lhes dê esse suporte.

Referências Bibliográficas

- [SÁ FILHO, 2007] SÁ FILHO, MANOEL G. C. A. Um estudo comparativo entre SPEM e BPMN como padrões para modelagem de Processos de Software. Trabalho de Graduação. Universidade Federal de Pernambuco. Agosto 2007.
- [FALBO, 1998] FALBO, RICARDO DE A. – Integração de Conhecimento em um Ambiente de Desenvolvimento de Software [Rio de Janeiro] 1998.
- [SPEM, 2008] THE OBJECT MANAGEMENT GROUP, INC. Software Process Engineering Metamodel Specification (SPEM). Abril 2008.
- [BPMN, 2009] BPMI - Business Process Modeling Notation (BPMN) versão 1.2, Janeiro 2009.
- [SOMMERVILLE, 2003] SOMMERVILLE, IAN. Engenharia de Software. 2003.
- [CMMI, 2008] CMMI® para Desenvolvimento, Versão 1.2. Abril 2008.
- [WfMC, 2008] Workflow Management Coalition, Workflow Standard Process Definition Interface - XML Process Definition Language. Versão 2.1a. Outubro 2008.
- [BizAgi, 2009] BPM BizAgi - <http://www.bizagi.com/esp>. Último acesso em: 22/06/2009.