

4. Método Proposto

O objetivo deste capítulo é apresentar e descrever os passos do método proposto para integrar o conhecimento obtido pelo modelo de processo de negócio, e pela ontologia do Udl, com intuito de facilitar a fase de elicitação de requisitos. Este método tem como objetivo principal auxiliar o engenheiro de requisitos na tarefa de elicitar requisitos mais aderentes ao negócio da organização e, desta forma, prover melhores artefatos para o desenvolvimento de sistemas de informação mais eficientes e que atendam às necessidades da empresa.

O método é dividido em cinco fases, que são apresentadas respectivamente nas seções 4.1, 4.2, 4.3, 4.4 e 4.5 deste capítulo. São elas: Conhecendo o Udl, Produzindo o conhecimento, Integrando o conhecimento e Exibindo o conhecimento integrado e Verificação entre modelos gerados.

4.1. Conhecendo o Udl

Esta é a primeira fase do método, na qual o engenheiro de requisitos tem seu primeiro contato com o Udl; é quando começa a construir seu entendimento sobre o Udl da aplicação e a perceber as necessidades da organização. Nesta fase, o engenheiro consulta as fontes de informações do Udl, como documentos, pessoas da organização responsáveis por áreas chaves, sistemas existentes na organização, entre outros.

Por esta ser uma fase de contato inicial, necessária para o entrosamento do engenheiro no Udl, não possui passos determinados a serem seguidos. O engenheiro de requisitos deve realizar estes primeiros contatos da maneira que julgar mais adequada para obtenção de um breve conhecimento sobre o Universo de Informações.

4.2. Produzindo o Conhecimento

Esta é a segunda fase do método, onde é apresentado como o “conhecimento” do modelo de processo de negócio e da ontologia pode ser produzido para servir de insumo para a terceira fase do método, na qual estes conhecimentos serão integrados. Para isso, mostraremos, na subseção 4.2.1, as maneiras utilizadas para obtenção do conhecimento e em quais situações devem ser aplicadas. Na subseção 4.2.2, exibiremos alguns itens que auxiliam na construção estruturada dos modelos. Na subseção 4.2.3, mostraremos como deve ser construído o modelo de processo de negócio e, na subseção 4.2.4, como deve ser construída a ontologia do Udl. Finalmente, na subseção 4.2.5, exibiremos como realizar a validação dos modelos construídos.

4.2.1. Formas de Obtenção do Conhecimento

A proposta deste trabalho é integrar, através de um método, os conhecimentos obtidos do modelo de processo de negócio e da ontologia, visando facilitar a elicitação de requisitos. Para atingir este objetivo, temos que obter estes “conhecimentos”. Esta obtenção pode ser feita de três formas:

1. **Criando “os conhecimentos”, ou seja, criando os modelos de onde provêm estes conhecimentos.**

Neste caso, produziremos os modelos de processo e de ontologia para utilizarmos como fonte de conhecimento para o método proposto.

2. **Utilizando modelos já desenvolvidos.**

Neste caso, devemos verificar os modelos existentes e definir se os mesmos podem ser utilizados como fonte de conhecimento, ou seja, se são equivalentes ao modelo de processo e de ontologia necessários para o método proposto.

3. Evoluindo modelos já desenvolvidos.

Neste caso, devemos verificar os modelos existentes e definir as partes destes modelos que podem ser utilizados como fonte de conhecimento, ou seja, que modelos ou partes de modelos poderão ser evoluídos para se tornarem equivalentes ao modelo de processo e de ontologia necessários para o método proposto.

Para as organizações que não possuem o mapeamento de seus processos e/ou a especificação formal do seu Udl (ontologia), será necessário construí-los. Portanto, a forma mais indicada é a primeira.

É importante notar que, após esta construção, se faz necessária uma contínua atualização dos modelos, em paralelo às modificações que ocorrem no Udl do negócio, para que em futuras utilizações eles estejam aderentes ao negócio.

Para organizações que já construíram seus modelos de processo e ontologia, a segunda e a terceira forma são as mais adequadas. A decisão depende da avaliação dos modelos que as organizações possuem. Se forem modelos compatíveis com os necessários à execução do método, a segunda forma é a mais adequada; caso os modelos não sejam compatíveis e necessitem de uma evolução para se compatibilizarem com os utilizados no método, então a terceira forma é a mais indicada. Mas vale salientar que é de extrema importância verificar o controle de atualização dos modelos, para evitar extrair conhecimentos que não estão aderentes ao negócio.

Neste trabalho, utilizaremos a primeira forma para obtenção dos “conhecimentos”, a fim de auxiliar as organizações que não possuem seus processos e ontologias modeladas.

4.2.2. Iniciando a Construção dos Modelos

Para uma construção estruturada dos modelos de processo e de ontologia é necessário estabelecer, para cada modelo, os quatro itens listados abaixo:

- O método: responsável por estabelecer a seqüência de passos necessária para o levantamento e modelagem das informações.
- O meta-modelo: responsável por definir as informações que deve ser modeladas.
- A linguagem e sua notação: responsável por definir a forma de comunicação das informações modeladas, os símbolos e regras para representá-las.
- A ferramenta: responsável por dar apoio para documentação das informações.

Definimos estes itens como partes de uma construção estruturada do modelo. Porque, seguindo estes itens, somos direcionados na construção do modelo, afinal temos um guia para construí-lo (método), um conjunto de informações que deve ser modelada (meta-modelo), uma linguagem na qual podemos representá-lo (linguagem e notação) e um apoio para documentá-lo (ferramentas).

4.2.3. Construção do Modelo de Processo

Para o desenvolvimento do Modelo do Processo de Negócio, conforme explicitado na seção anterior, é necessário utilizarmos um método, ou seja, o passo-a-passo que conduz a atingir o objetivo de produzir um modelo de negócio. Para tal, utilizaremos uma simplificação do método de modelagem de processo de negócio de [Sharp 00], selecionado no capítulo 2. Este método sofreu uma simplificação, pois o mesmo se estende até a criação de casos de uso para o sistema, mas não é objetivo desta modelagem derivar os casos de uso; nosso objetivo é conhecer o processo de negócio de uma organização. Por isso, fez-se necessária a simplificação do método.

Os passos do método simplificado, utilizado na modelagem de processos [Sharp 00], estão descritos abaixo:

1. Emoldurar o Processo

- Identificar o conjunto de processos, que inclui tanto os processos-alvo, que são o foco da modelagem, quanto os processos que os rodeiam. Esta atividade ajuda a clarificar os processos que estão dentro e fora do escopo da modelagem.
- Identificar os processos-alvo e seus limites (nome evento iniciados, atores, resultados, entre outros)
- Revisar ou documentar a missão, estratégia e objetivos da organização.
- Mapear uma prévia estimada do processo.
- Mapear uma prévia do processo aperfeiçoado.
- Desenvolver um glossário de termos do Udl.
- Opcionalmente, iniciar uma documentação sobre as observações da cultura, competências.

2. Entender o Processo Atual (AS-IS)

- Mapear o *workflow* do processo corrente. Visando identificar “quem faz o que”, o autor sugere utilizar os diagramas “*swimlanes*”.
- Investigar os “outros objetivos” da organização, como por exemplo, utilização de algum sistema já existente na organização, políticas a seguir, metas de medições para as atividades.
- Identificar pontos fracos do processo.
- Mapear uma avaliação final dos “outros objetivos” da organização, como suas regras, políticas, sistemas, motivações.
- Documentar aspectos importantes de cultura, competências e sistemas de gerência.

3. Projetar o Novo Processo (TO-BE)

Esta fase está subdividida em dois estágios:

3.1. Caracterizar o Novo Processo (TO-BE)

- Decidir a direção do novo processo (Continuar como está? Realizar melhorias? Refazer o processo?).
- Desenvolver idéias de novas funcionalidades para o novo processo, baseadas nos objetivos em conjunto com os “outros objetivos”.
- Montar uma matriz com os objetivos e “outros objetivos” e mapear as idéias das novas funcionalidades que o atendem.
- Avaliando a matriz, selecionar novas funcionalidades chaves para o novo processo.

3.2. Projetar o *Workflow* do Novo Processo (TO-BE)

Neste estágio é desenvolvido o *workflow* do novo processo.

Este passo 3 só é necessário ser executado caso perceba-se que o processo sofrerá evolução para atingir novos objetivos. Ou o usuário do processo está com problemas em algumas atividades, ou para melhorar a execução de processo e/ou atividades.

Selecionado o método para construção do modelo de processo de negócio, precisamos estabelecer o meta-modelo no qual se baseará o modelo de processo. O meta-modelo selecionado foi o *usual process* [Fiorini 01], conforme mostrado no capítulo 2, por, entre outras vantagens, ser um meta-modelo com foco na reutilização dos processos, ou seja, um meta-modelo desenvolvido para permitir a reutilização dos processos.

Escolhidos o método e o meta-modelo, partimos para selecionar a linguagem que, conforme mostrado no capítulo 2, foi a linguagem do *usual process* [Fiorini 01] que se mostrou mais adequada. Esta linguagem, porém,

sofreu uma re-interpretação em alguns termos, para adequar suas informações com as informações esperadas pelo método proposto neste trabalho. Abaixo mostramos as re-interpretações feitas:

Políticas – Este termo, segundo [Leite 98], também deve conter as informações de regras de negócio, bem como as informações de tempo estimado para duração das atividades.

Restrição – Este termo deverá conter as associações das regras de negócio com as atividades.

Pré-condição – Este termo deverá conter, segundo [Maffeo 92], as informações de tempo (evento temporal) e de eventos de entrada.

Pós-condição – Este termo poderá representar um fluxo de decisão, ao abrir fluxos de atividades distintos, além de representar as informações de eventos de saídas.

Duração do projeto – Este termo deverá conter a informação da duração de implantação do processo, ao invés da duração do projeto conforme definido pela autora [Fiorini 01].

Como mostrado anteriormente, no capítulo 2, a linguagem proposta por [Fiorini 01] é textual, não possui uma notação gráfica própria para representação dos elementos do meta-modelo. A fim de facilitar a visualização, bem como a comunicação do processo, organização e sequenciamento de suas atividades, propomos, como complemento da linguagem textual, uma notação gráfica própria da linguagem para representar graficamente alguns dos principais elementos do processo (Figura 9):

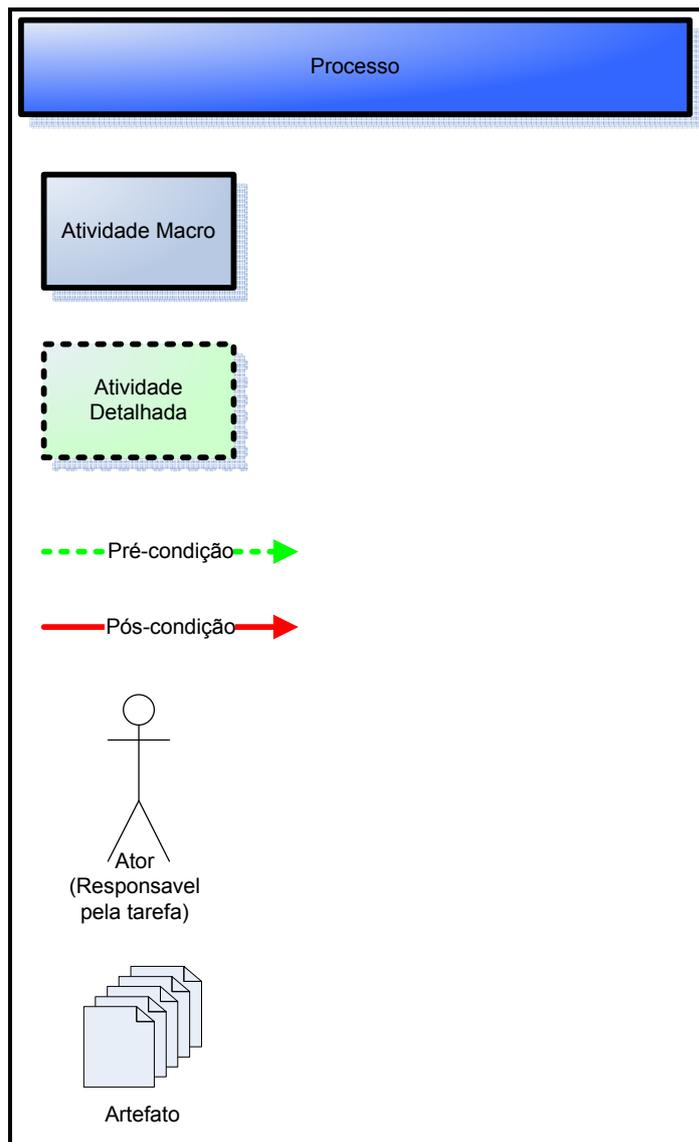


Figura 9 - Notação para os principais elementos do *usual process*.

Por fim, para apoiar a documentação do modelo do processo, segundo o meta-modelo e a linguagem proposta por [Fiorini 01], foi selecionada a ferramenta SRP – Sistema de Cadastro e Reutilização de Processos de Negócio desenvolvida neste trabalho, conforme foi mostrado no capítulo 2.

Ao final desta etapa, teremos o modelo de processo de negócio construído.

4.2.4. Construção da Ontologia

Assim como na modelagem do processo, e conforme mostrado na subseção 4.2.2, para modelar a ontologia do Udl é necessário estabelecer o método a ser seguido. O método selecionado para a construção da ontologia, conforme mostrado no capítulo 2, foi o processo semi-automático para construção de ontologia [Breitman 04], que utiliza o léxico como entrada para iniciar sua execução.

Durante a aplicação do método selecionado, percebeu-se a necessidade de realizar algumas alterações com intuito de facilitar sua execução. Portanto, propomos o uso de algumas heurísticas adicionais, que foram desenvolvidas neste trabalho, para complementá-lo. Estas heurísticas estão descritas abaixo e, para facilitar seu entendimento, utilizaremos exemplos do estudo de caso proposto no capítulo 5:

- a. Caso um termo, que não está presente no léxico, apareça algumas vezes como o objeto direto/indireto dos verbos nos relacionamentos das classes da ontologia: devemos considerar que o mesmo pode ser um novo termo para o léxico e, portanto, uma nova classe ou propriedade para a ontologia.

Ex.: O termo “pós-graduação”.

- b. Uma descrição de impacto do léxico, com mais de um objeto direto e/ou indireto: deve ser representado na ontologia por mais de um relacionamento, ou seja, um relacionamento para cada objeto. Depois deve ser verificada a consistência destes relacionamentos.

Ex.: O impacto do termo “agência de fomento”: Agência de fomento oferece bolsa de fomento para o DI.

Relacionamentos na ontologia: 1- Oferece bolsa de fomento; 2- Oferece para o DI.

- c. Uma descrição do impacto no léxico com dois ou mais verbos diferentes também geram dois ou mais relacionamentos para o termo.

Ex.: O impacto do termo “formulário de inscrição”: Formulário de inscrição é entregue pelo candidato na inscrição da pós para ser avaliado pela comissão de pós-graduação.

Relacionamentos na ontologia: 1- É entregue pelo candidato; 2- É avaliado pela comissão de pós-graduação.

- d. Em relação aos termos disjuntos, acrescentar, na busca realizada no vocabulário do léxico (passo 3.1.1.4 e 5.1.2), a procura de indícios de alternativa (ou exclusivo), além dos indicativos de negação já sugeridos pelo método.

Ex.: Ou candidato de mestrado ou candidato de doutorado.

- e. Os verbos do léxico, assim como seus impactos, devem fazer parte dos relacionamentos da ontologia. Portanto, deve-se verificar se na ontologia já existem classes que tenham relacionamento que represente o verbo do léxico e cada um de seus impactos. Caso não exista acrescentar o(s) relacionamento(s).

Ex.: O verbo “atribuir bolsa”, tem um impacto que não está relatado na ontologia como relacionamento: Aluno deve cumprir os requisitos do novo regime de financiamento.

- f. A heurística anterior pode se tornar um forte indício para auxiliar na execução do passo 5.1.1 no método [Breitman 04], caso o ator do verbo seja um termo estado do léxico.

Ex.: O verbo “confirmar vaga” tem como ator o “candidato aprovado” (termo classificado como estado no léxico). Portanto, “candidato aprovado” passa a ser uma classe, visto que o impacto do verbo se tornou um relacionamento na ontologia (heurística e).

- g. Os estados que se transformarem em propriedades também tem que ter seus impactos transformados em relacionamentos na ontologia, caso os relacionamentos ainda não existam, semelhante à heurística e.

Ex.: O estado “Matrícula em admissão”.

- h. No caso de um impacto no qual o objeto direto/ indireto é um verbo (ou um sinônimo do verbo), verificar se já não existe algum relacionamento na ontologia que mapeou este relacionamento. Afinal, segundo a heurística e, todos os impactos dos verbos são mapeados como relacionamentos, caso não exista “acrescentar o relacionamento”.
- Ex.: “DAR efetiva inscrição do candidato”, neste caso o objeto direto do relacionamento é um sinônimo de um termo do tipo verbo do léxico (“inscrição” é sinônimo de “inscrever na pós-graduação”). Porém, este relacionamento (“inscrever na DAR”) já foi mapeado na ontologia.
- i. A heurística anterior também vale para os termos do tipo estado.
- Ex.: “DAR cria matrícula em admissão”.
- j. Colocar os sinônimos dos termos na documentação da ontologia. Heurísticas para definição de sinônimos podem ser encontradas em [Franco 92].

A aplicação do método semi-automático para construção de ontologias, que transforma o léxico em ontologia [Breitman 04], complementado pelas heurísticas acima descritas, contribuiu para a eliciação das seguintes primitivas básicas da ontologia:

- Classes e sua descrição;
- propriedade das classes (“*slots*”): do tipo relacionamento com outras classes;
- conjunto de termos que possivelmente formarão uma hierarquia;
- axiomas.

O meta-modelo selecionado, o *Frame Ontology*, é composto pelo conjunto de informações listadas na Figura 10 e, como podemos notar, na comparação entre os itens do meta-modelo e itens elicitados com a aplicação do método, existem várias destas informações que não foram identificadas pelo método acima, são as “informações que faltaram”.

- Classes e suas hierarquias, inclusive herança múltipla.
 - Classe concreta ou abstrata
- Propriedades das classes (slots)
 - Atributos das classes (intrínsecos e extrínsecos)
 - Relacionamentos do tipo parte
 - Relacionamentos com outras classes
- Facetas dos slots
 - Cardinalidade
 - Simples/Múltipla
 - Mínima e Máxima
 - Tipo de valor
 - String
 - Number
 - Boolean
 - Enumerado
 - Instancia
 - Domínio e range do slot
 - Slots inversos
 - Valores padrão
- Axiomas
- Instancias

Figura 10 – Conjunto de informação do meta modelo *Frame Ontology*.

Para auxiliar na identificação das “informações que faltaram”, propomos, abaixo, algumas heurísticas que visam auxiliar na descoberta de algumas destas informações:

1. Hierarquia

O método [Breitman 04] nos fornece o conjunto de termos que tem relacionamentos idênticos e, possivelmente, podem fazer parte de uma hierarquia (item 6 do método [Breitman 04] (Figura 8)). Portanto, podemos nos utilizar das seguintes heurísticas para organizá-las hierarquicamente:

1. Na posição de cima deve ficar o termo que tem apenas os relacionamentos comuns (superclasse), e embaixo os termos que tem ,além destes relacionamentos, mais outros relacionamentos (sub-classes).
2. Verificar a relação de “*is a*” e/ou “é um tipo de” entre os termos que estão embaixo (sub-classes) em relação ao que está em cima (superclasse). Esta verificação pode ser feita observando se existe, no impacto ou na noção, algum relacionamento que indique esta relação, ou mesmo em entrevistas e/ou documentação do Udl.
3. Verificar consistência da ontologia.

Existem alguns casos em que pode haver confusão na definição da hierarquia entre termos, abaixo listamos dois casos onde verificamos a possibilidade de ocorrência de equívocos na definição da hierarquia entre termos da ontologia:

Caso 1. Hierarquia ou Relacionamento “Parte de”?

Deve-se verificar nos conjuntos de termos, que possivelmente formaram uma hierarquia (resultado da aplicação do método [Breitman 04]), os seguintes pontos que procuram clarear as diferenças entre estes dois conceitos:

- Na hierarquia, as “sub-classes” são “um tipo da” “superclasse”. No relacionamento “Parte de”, as “sub-classes” são partes da, ou melhor, “são-uma” “super-classe” e, como partes de um conjunto, não possuem a relação de herança.

Estamos utilizando os termos “sub-classe” e “superclasse” num relacionamento “partes de” apenas para fins de prover um melhor entendimento de que a “sub-classe” está contida na “super-classe”, por isso a aplicação das aspas, porém, conforme dito acima, não existe relacionamento hierárquico entre estas classes.

- As classes que possuem um relacionamento “Parte de” podem ter relacionamento forte, no qual depende da “superclasse” para existir (ex.: “membros da comissão” é parte da “comissão de pós-graduação” e, neste caso, “membros da comissão” não existe se não existir a comissão de “pós-graduação”); ou relacionamento fraco, no qual as “sub-classes” têm existência independente da “superclasse” (ex.: “histórico escolar” é parte da “documentação do candidato” e, neste caso, o “histórico escolar” tem existência independente da “documentação do candidato”).

Estas verificações podem ser feitas observando se existe na descrição ou no relacionamento das classes, ou mesmo em entrevistas e/ou documentação do Udl algum indício que comprove o descrito nos pontos acima. Caso seja comprovado, ao classificar os relacionamentos dos termos como “Partes de” ou

“herança”, deve-se verificar a consistência destes relacionamentos no contexto da ontologia.

Caso 2. Classe ou Instância?

Deve-se verificar, para cada classe da ontologia, se atende aos pontos listados abaixo. Caso atendam, devem ser classificadas com instâncias:

- Se existe um conjunto de classes com atributos e relacionamentos idênticos (provavelmente já faz parte do conjunto de termos que, possivelmente, formarão uma hierarquia). Em seguida, verificar quais destas classes só tem sentido no Udl se tiverem seus atributos “preenchidos” com valores.
- Observar qual(is) classes não podem ser instanciadas.

As classes que obedeceram às heurísticas acima, provavelmente são instâncias da classe que não obedeceu às heurísticas, como exemplo temos: Agência de fomento, Capes, Cnpq, Faperj (conjunto de termos que possivelmente formaram uma hierarquia). As três últimas não podem ser instanciadas e só fazem sentido no Udl se estiverem com seus atributos preenchidos. Logo, são instâncias de Agência de Fomento. Após a classificação como classes ou instâncias deve-se verificar a consistência no contexto da ontologia.

2. Atributos das Classes

- Os termos que estão classificados como “estados no léxico” são candidatos a serem atributos do termo ao qual o estado é atribuído, como, por exemplo, o termo “matriculado” é um estado atribuído ao candidato e pode ser considerado como um atributo do mesmo.
- Outros atributos também podem ser descobertos através de entrevistas e/ou documentação do Udl.

Após a classificação dos termos como atributo, deve-se verificar a consistência no contexto da ontologia.

Nota-se que nem todas as informações foram identificadas, ou pelo menos tiveram sua identificação auxiliada, através de heurísticas. Para estas informações recomenda-se a utilização de entrevistas e/ou documentação do Udl para identificá-las.

- Cardinalidade
 - Simples/Múltipla
 - Mínima e Máxima
- Tipo de valor
 - Texto
 - Numérico
 - Boleano
 - Enumerado
 - Instância
- Domínio e range do slot
- Slots inversos
- Valores padrão
- Constraints – axiomas e regras de negócio
- Classe concreta e abstrata

Figura 11 - Informações não identificadas pelas heurísticas.

Para documentar a ontologia produzida, selecionamos, conforme mostrado no capítulo 2, uma ferramenta que é bastante referenciada na literatura e de livre distribuição, o Protege. Esta ferramenta foi desenvolvida com base no meta-modelo *Frame Ontology* [Gruber 93] e possui *plugins* para permitirem converter a ontologia modelada em diversas linguagens, como RDF, OWL, XML.

Em resumo, ao final desta etapa, teremos a ontologia construída a partir da aplicação do método semi-automático de construção da ontologia [Breitman 04], complementado pelas heurísticas apresentadas anteriormente e por entrevistas com atores do Udl e documentação do Udl.

4.2.5. Validação dos Modelos Construídos

Os modelos construídos devem ser validados pelos usuários responsáveis para, desta forma, validarmos a consistência dos modelos com o Udl. Esta validação é realizada através da demonstração dos modelos construídos para os usuários e coleta dos “*feedbacks*” fornecidos pelos mesmos durante a demonstração dos modelos. A partir destes “*feedbacks*” os modelos são refinados e retro-alimentados.

Com o processo de negócio modelado, a ontologia do negócio construída e ambos os modelos validados temos os insumos necessários para integrar estes conhecimentos na terceira fase do método, descrita na próxima seção.

4.3. Integrando o Conhecimento

O objetivo desta seção é apresentar a terceira fase do método proposto para integrar os conhecimentos, obtidos com o modelo de processo e ontologia gerados na seção anterior, com intuito de auxiliar o engenheiro de requisitos na sua tarefa de elicitação de requisitos de sistemas de informação para a organização.

Na subseção 4.3.1 apresentaremos o embasamento teórico que foi utilizado como fonte para criação do método de integração dos conhecimentos. Na subseção 4.3.2 mostraremos a preparação para o método de integração de conhecimento e, finalmente, na subseção 4.3.3 mostraremos o método de integração de conhecimento.

4.3.1. Embasamento Teórico

A integração dos conhecimentos teve sua concepção baseada na idéia do EM [Bubenko 99] que, segundo o autor, define-se como:

“Enterprise Modelling consists of several submodels of knowledge, that are developed in group activities, by users, stakeholders, and requirements engineers, during particular knowledge acquisition processes. The submodels are: Objectives Model (OM), Activities & Usage Model (AUM), Actors Model (AM), Concept Model (CM) and Information System Requirements Model (ISRSM). All the submodels are interrelated by the intermodel links”

O EM mostra que, para produção de modelos organizacionais mais completos, as ligações entre modelos são essenciais porque tornam o conhecimento mais disponível, sendo possível ver porque certos processos e requisitos do sistema de informação têm que ser introduzidos [Pádua 04].

As ligações propostas pelo EM são ilustradas na figura abaixo:

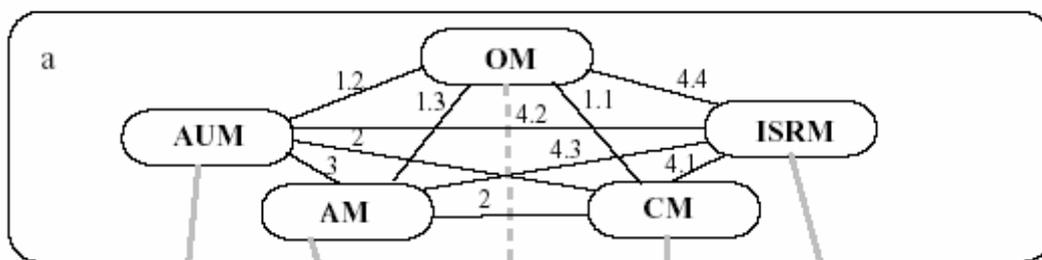


Figura 12 - Ligações entre os modelos organizacionais no EM [Bubenko 94].

Seguindo a linha das ligações entre modelos propomos, no método de integração de conhecimento, criar ligações entre os modelos de processo e ontologia. Algumas dessas ligações são derivadas do EM (conforme pode ser verificado nos passos do método de integração de conhecimento), apesar do EM ter 5 sub-modelos (OM, AUM, AM, CM e ISRSM), enquanto nosso método tem 2 modelos, porém podemos considerar que OM, AUM e AM são, aproximadamente, equivalentes ao modelo de processo, enquanto o CM é, aproximadamente, equivalente à ontologia.

Quando dizemos que um modelo do EM é aproximadamente equivalente a um modelo do método, estamos querendo mostrar que estes modelos possuem informações diferentes, apesar de, em suas essências, modelarem o mesmo tipo de informação. Um exemplo disto é o CM e a ontologia, ambos, em sua essência, modelam os conceitos e relacionamentos de um Udl, ou seja, especificam o Udl, porém possuem informações diferentes: a ontologia modela os axiomas entre os conceitos enquanto o CM não modela.

Nas próximas seções apresentamos, respectivamente, os passos da preparação para método de integração do conhecimento e o próprio método. Este visa auxiliar a fase de elicitación de requisitos, facilitando a busca, ou melhor, a elicitación de informação, através da produção de informação de maneira estruturada, permitindo rastrear as fontes da qual provieram a informação e apresentando informações consideradas essenciais para a criação do documento de requisitos.

4.3.2.Preparação para o Método de Integração dos Conhecimentos

Passo 1. Organizar Entradas para o Método:

1. A modelagem de processo do negócio

A modelagem selecionada para ser utilizada neste método foi a da [Fiorini 01], conforme apresentado na subseção 4.2.3.

2. A ontologia da linguagem utilizada na modelagem de processo

A ontologia foi desenvolvida com base na documentação da linguagem de modelagem de processo da [Fiorini 01].

3. A ontologia do Udl

A ontologia foi desenvolvida utilizando o processo semi-automático para construção de ontologia a partir do léxico [Breitman 04], e complementada por algumas heurísticas, conforme apresentado na subseção 4.2.4.

4. Solicitação do cliente + necessidades

Estas informações foram obtidas na etapa de conhecimento do Udl (seção 4.1).

Passo 2. Conhecer o Udl Atual do Negócio:

O engenheiro de requisitos deverá navegar pelas entradas 1, 2 e 3 do método, da seguinte maneira:

1. Compreensão da linguagem na qual o processo está representado. Para tal, o engenheiro de requisitos deve:

1.1. Conhecer a notação utilizada na linguagem de modelagem do processo.

A especificação desta notação deve ser encontrada na ontologia da linguagem, assim como na documentação da linguagem de modelagem de processo.

1.2. Conhecer o significado de cada notação. Estes significados devem ser encontrados na ontologia da linguagem.

O engenheiro de requisitos deve buscar, na documentação, classes da ontologia, seus significados e seus relacionamentos com outras classes.

2. Compreensão do processo atual do negócio. Para tal, o engenheiro de requisitos deve:

2.1. Conhecer o processo de negócio. A especificação deste processo deve ser encontrada nos modelos do processo do negócio.

2.2. Conhecer o Udl. A especificação deste Udl deve ser encontrada na ontologia do Udl.

3. Compreensão inter-relacionada das entradas. Para realizar este passo, o engenheiro de requisitos deve navegar pelo modelo de processo e, à medida que for encontrando:

3.1. Termos específicos do Udl do negócio: deve buscar na documentação das classes, na ontologia do Udl, seus significados.

3.2. Notações próprias da linguagem: deve buscar, na documentação das classes, na ontologia da linguagem, seus significados.

Passo 3. Focar no Processo que envolve a Solicitação do Usuário e o(s) Processo(s) Relacionado(s):

O engenheiro de requisitos, após ter obtido um breve conhecimento da situação atual do Udl, deve selecionar e focar no processo que está relacionado com a solicitação e as necessidades do usuário. Neste momento, deve atentar também para pontos do processo-foco que tem ligação com outros processos do Udl.

4.3.3.Método de Integração dos Conhecimentos

Passo 1. Gerar lista de objetivos com atividades macro do processo-foco e com atividades macro dos processos relacionados:

O engenheiro de requisitos deve listar todas as atividades macro e o objetivo do processo-foco, bem como as dos processos relacionados selecionadas no passo 3 da preparação para o método.

Onde encontrar as informações:

- Processo e sua descrição e objetivo: No SRP → na visualização do processo, respectivamente os itens “nome”, “descrição” e “objetivo”.

Processos	<input type="text" value="Nome"/>	Processo Admissão na pós-graduação PUC-RJ
Visualizar	Autor	Ana Luiza
Cadastrar	Natureza	Usual
Reutilização	Classificação	Fundamental
Tabelas Auxiliares	Conformidade	CMM
Visualizar	<input type="text" value="Objetivo"/>	Este processo tem como objetivo permitir o ingresso na pós-graduação do DI, através de processo de seleção
Cadastrar	<input type="text" value="Descrição"/>	Este processo contempla desde a divulgação das inscrições, o processo de seleção dos candidatos até a divulgação dos selecionados com suas respectivas bolsas.
Admin	Pré-condição	Início do ano letivo
Logout		

Figura 13 - Visualização do processo – Ferramenta SRP.

- Atividades macro: No SRP → na visualização da árvore do processo temos todas as atividades do processo e, “clikando” em cada uma das atividades, temos suas descrições, no item “descrição”.

Processos	Árvore de Atividades	Nova Atividade Macro
Visualizar	Processo Admissão na pós-graduação PUC-RJ alterar	
Cadastrar	<ul style="list-style-type: none"> ▽ Divulgar edital de seleção para pós-graduação alterar excluir <ul style="list-style-type: none"> ▽ Preparar edital de seleção p/ pos alterar excluir △ ▽ Disponibilizar modelos de documentos pedidos no edital alterar △ Disponibilizar edital alterar excluir △ ▽ Realizar inscrição para pós-graduação alterar excluir <ul style="list-style-type: none"> ▽ Preparar inscrição alterar excluir △ ▽ Receber inscrição alterar excluir △ Verificar documentação do candidato alterar excluir 	
Reutilização		
Tabelas Auxiliares		
Visualizar		
Cadastrar		
Admin		
Logout		

Figura 14 - Árvore do processo – Ferramenta SRP.

Atividade Macro	
<input type="text" value="Nome"/>	Ação: Divulgar Objeto: edital de seleção para pós-graduação
Sinônimos (separados por vírgula)	Ação: Objeto:
<input type="text" value="Descrição"/>	Esta atividade objetiva realizar a divulgação do edital de seleção para pós-graduação, onde constam as informações necessárias para os candidatos efetuarem as inscrições.

Figura 15 - Visualização da Atividade Macro – Ferramenta SRP.

Contribuição:

- Mostra ao engenheiro de requisitos uma visão geral dos objetivos dos processos (foco e os relacionados). Estes objetivos são úteis na elicitação dos requisitos, visto que os requisitos devem estar aderentes aos objetivos do processo.
- Separação das atividades macro de cada processo estão contribuindo para atingir os objetivos das que estão desvinculadas do objetivo.
 - Estas atividades podem ser fontes de requisitos
- Verificar se os processos com suas atividades estão descritos de maneira coerente com seus objetivos.
- Pode servir de base para extração dos objetivos do sistema.

Passo 1.1. Gerar Lista de Atividades Macro do Processo Foco Relacionadas com outros Processos:

O engenheiro de requisitos deve analisar todas as atividades do processo-foco, através de suas descrições, para selecionar as atividades que tem algum relacionamento com os outros processos listados no passo 3.

Contribuição:

- Mostrar ao engenheiro de requisitos uma visão de quais atividades podem causar impactos em outros processos ou ser afetadas pelos mesmos.

Passo 2. Gerar Lista de Atividades x Responsáveis:

Para o processo–foco, o engenheiro de requisitos deve extrair todas as suas atividades macro e detalhadas, com suas respectivas descrições, bem como, suas relações de nível hierárquico, caso existam. Para cada uma destas atividades, devem ser exibidos seus responsáveis com as descrições de suas funções.

Onde encontrar as informações:

- Atividades macro, descrição e responsáveis: No SRP → na visualização da árvore do processo temos todas as atividades do processo e, “clizando” em cada uma das atividades, temos suas descrições, no item descrição e no item função temos o responsável por executá-la.
- Atividade detalhada, descrição e responsáveis: No mesmo caminho da atividade macro, sendo que, na árvore de processo, a atividade detalhada encontra-se “dentro”, ou seja, no nível hierárquico abaixo da atividade macro.

Processos	Árvore de Atividades	Nova Atividade Macro	Nova
Visualizar	Processo Admissão na pós-graduação PUC-RJ alterar		
Cadastrar	<ul style="list-style-type: none"> ▽ Divulgar edital de seleção para pós-graduação alterar excluir <ul style="list-style-type: none"> ▽ Preparar edital de seleção p/ pos alterar excluir △ ▽ Disponibilizar modelos de documentos pedidos no edital alterar excluir △ Disponibilizar edital alterar excluir 		
Reutilização	<ul style="list-style-type: none"> △ ▽ Realizar inscrição para pós-graduação alterar excluir 		
Tabelas Auxiliares			
Visualizar			
Cadastrar			

Figura 16 - Atividade detalhada na árvore do processo - Ferramenta SRP.

Atividade Macro	
Nome	Ação: Divulgar Objeto: edital de seleção para pós-graduação
Sinônimos (separados por vírgula)	Ação: Objeto:
Descrição	Esta atividade objetiva realizar a divulgação do edital de seleção para pós-graduação, onde constam as informações necessárias para os candidatos efetuarem as inscrições.
Restrições	
Pre-condição	Início do ano letivo
Pos-Condição	Edital divulgado
Tabelas Auxiliares	
Ferramentas:	Não há ferramenta associada
Artefatos de Entrada:	Não há artefato associado
Artefatos de Saída:	Edital de seleção
Funções:	Coordenador
Métodos:	Não há método associado

Figura 17 – Item Funções: corresponde ao responsável por executar a atividade macro - Ferramenta SRP.

- Relações de nível hierárquico: No SRP → na visualização da hierarquia da árvore do processo.
- Descrição da função dos responsáveis: No SRP → na visualização da tabela auxiliar Funções, no item “descrição”.

Sistema de Reutilização de Processos	
Processos	Artefatos Conceitos Ferramentas Formulários Funções Medições Métodos Políticas Treinamentos
Visualizar	
Cadastrar	Função
Reutilização	
Tabelas Auxiliares	
Visualizar	Nome Coordenador
Cadastrar	Descrição Professor da pós-graduação responsável por gerir o DI, implementar as regras determinadas pela comissão, bem como as atividades rotineiras de coordenação de um departamento de informática
Admin	
Logout	

Figura 18 - Tabela auxiliar Função exibindo a descrição da função – Ferramenta SRP.

Contribuição:

- Conhecer os responsáveis pelas atividades do processo e, desta forma, saber, de maneira simples, quem entrevistar para obter mais informação sobre o processo e/ou atividade.
- Os responsáveis são funções que possuem descrição. Logo, é possível verificar se a atribuição das atividades aos responsáveis está coerente com a competência descrita na função.
- Pode servir de base para extração dos futuros requisitos e seus atores.

Passo 3. Gerar Lista de Atividades x Entradas x Saídas:

Reproduzir a lista de atividades geradas no passo anterior, adicionando as informações de entradas (pré-condição) e saídas (pós-condição) de todas as atividades, identificando as que são artefatos, atividades e eventos.

Onde encontrar as informações:

- Entradas e saídas das atividades com suas respectivas identificações: No SRP → na visualização da árvore do processo temos todas as atividades do processo e, “clikando” em cada uma das atividades, encontramos suas entradas e saídas.
- Artefatos: São encontrados no item “artefatos de entrada” e “artefatos de saída”.
- Atividades: São encontrados na visualização da hierarquia da árvore de processos.
- Eventos: São encontrados nos itens “pré-condição” e “pós-condição”.

Nome	Ação: Divulgar Objeto: edital de seleção para pós-graduação
Sinônimos (separados por vírgula)	Ação: Objeto:
Descrição	Esta atividade objetiva realizar a divulgação do edital de seleção para pós-graduação, onde constam as informações necessárias para os candidatos efetuarem as inscrições.
Restrições	
Pre-condição	Início do ano letivo
Pos-Condição	Edital divulgado
Tabelas Auxiliares	
Ferramentas:	Não há ferramenta associada
Artefatos de Entrada:	Não há artefato associado
Artefatos de Saída:	Edital de seleção
Funções:	Não há função associada
Métodos:	Não há método associado

Figura 19 - Entradas das atividades – Ferramenta SRP.

Contribuição:

- Pode auxiliar na definição da ordem de prioridade para desenvolvimento das funcionalidades.
- Visualização consolidada das atividades, eventos e artefatos, que são necessários para iniciar uma atividade.
- Visualização consolidada das atividades, eventos e artefatos, que são esperados como resultados da execução de uma atividade.
- Pode servir de base para extração das “pré-condições” e “pós-condições” das funcionalidades.

Passo 4. Gerar Lista de Atividades x Relacionamentos:

Reproduzir a lista de atividades gerada no passo 2. O nome das atividades no processo é composto por ação e objeto. O engenheiro de requisitos deve procurar no objeto de cada atividade termos que sejam mencionados na ontologia, sejam estes termos classificados como classes ou “*slots*” da ontologia. Encontrados estes termos, o engenheiro deve adicioná-los à lista, associando a atividade que o mencionou com sua descrição e todos os relacionamentos que o envolvem.

Onde encontrar as informações:

- Termos – Classes: No Protege → na visualização das classes.
- Termos – “*Slots*”: No Protege → na visualização dos “*slots*”.

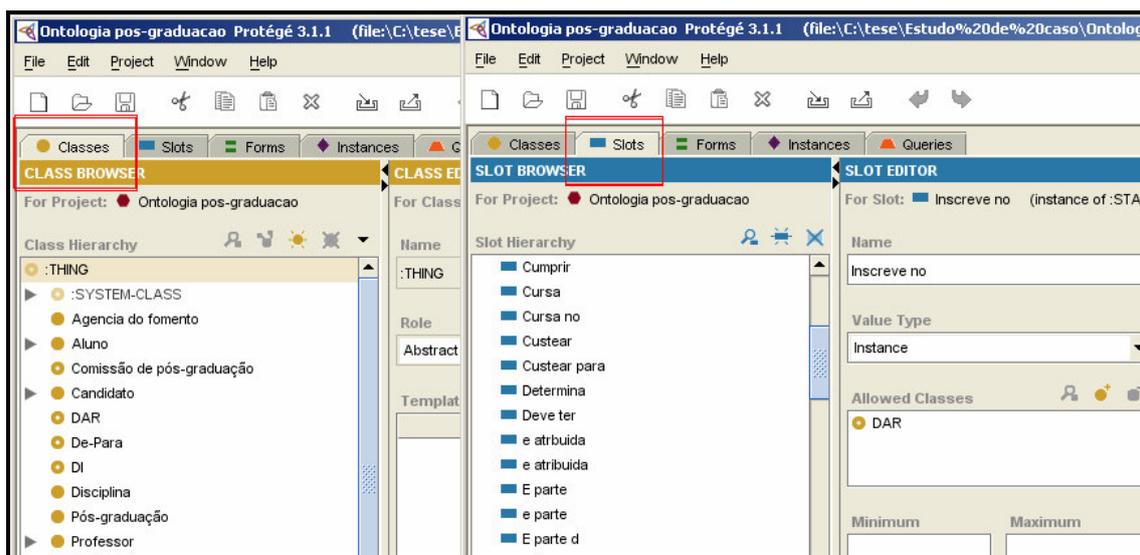


Figura 20 - Classes e “*Slots*” – Ferramenta Protege.

- Descrição dos termos: No Protege → no item de documentação da classe e do “*slot*”.
- Relacionamentos: No Protege → no “*slot*” da classe classificadas como *instance-type*.

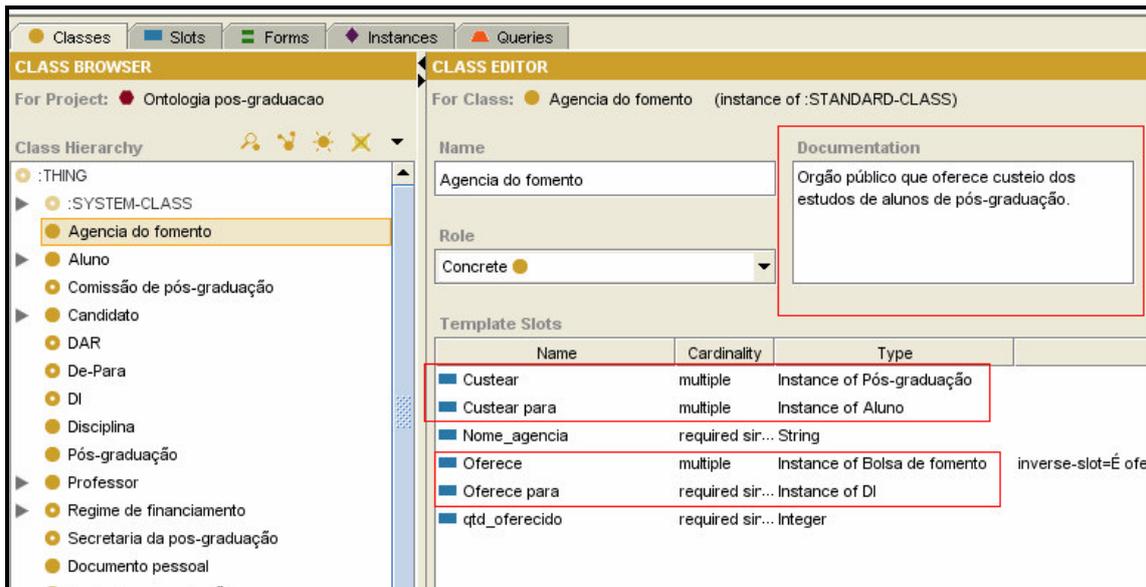


Figura 21 - Descrição e relacionamento – Ferramenta Protege.

Contribuição:

- Os relacionamentos referentes aos termos da atividade podem indicar novas atividades que não apareceram na modelagem de processo, e que são necessárias para desenvolvimento do sistema.
- Conhecendo outros relacionamentos dos termos utilizados na atividade, podemos verificar possíveis impactos que esta atividade pode causar em outras.

Passo 5. Gerar Lista de Atividades x Axiomas/ Regras de Negócio:

Reproduzir a lista de atividades, gerada no passo 2, associando a cada atividade suas regras de negócio e axiomas. As regras de negócio podem ser provenientes do processo ou da ontologia. As provenientes do processo estão diretamente associadas às atividades, porém, para descobrir as regras de negócio e axiomas, provenientes da ontologia é necessário procurar, no objeto de cada atividade, termos que sejam mencionados na ontologia. Encontrados estes termos, o engenheiro deve adicionar à lista, as regras de negócio e axiomas dos termos das atividades que o mencionaram.

Onde encontrar as informações:

- Regras de negócio: No SRP → na visualização da árvore do processo, temos todas as atividades do processo e, “clitando” em cada uma das atividades, encontramos o item “restrições” onde está descrita a regra de negócio associada à atividade.
- Regras de negócio e axiomas: No Protege → na visualização da classe no item “*constraint*”.

Contribuições:

- Visualização consolidada de todas as regras que as atividades devem seguir.
- Pode servir de base para as regras de negócio do sistema.

Passo 6. Gerar Lista de Atividades x Restrições:

Reproduzir a lista de atividades, gerada no passo 2, adicionando as restrições (cardinalidade) de cada termo, que aparecem no objeto da atividade, e é mencionado na ontologia.

Onde encontrar as informações:

- Restrições-Cardinalidade: No Protege → na visualização dos “*slots*” (classificados como *instance-type*) de cada classe.

Contribuições:

- As restrições das atividades (cardinalidade) são um tipo de informação utilizada na elicitação de requisitos, como restrições do negócio, que devem ser seguidas para que os requisitos fiquem aderentes ao negócio. Exemplo: Temos na ontologia, exibida no estudo de caso, o “*slot*” classificado como *instance type*, “atribui”. Este possui cardinalidade 1 → n entre as classes “Comissão de pós-graduação” e “Bolsa”. Traduzindo para uma restrição de negócio, temos que a “comissão de pós-graduação” pode atribuir de 1 a n bolsas.

Passo 7. Gerar Lista de Atividades Apoiadas por Sistemas:

Reproduzir a lista de atividades, gerada no passo 2, adicionando à informação se a atividade é ou não apoiada por sistemas, e sendo apoiada, dizendo por qual sistema.

Onde encontrar as informações:

- Sistema: No SRP → na visualização da árvore do processo temos todas as atividades do processo e, “clizando” em cada uma das atividades, encontramos o item “ferramentas”, que informa qual sistema apóia a atividade.

Contribuições:

- Conhecer as atividades que possuem sistemas, para uma futura integração do mesmo com o sistema que será desenvolvido.
- Verificar se os sistemas estão atendendo à realização das atividades, com intuito de verificar a necessidade de modificar ou não o sistema.

Passo 8. Exibir os Atributos dos Termos do Udl com suas Restrições (cardinalidade, valores, tipo de dados):

Montar uma lista com todas as classes da ontologia do Udl e, para cada classe, exibir todos os seus atributos; para cada atributo, exibir suas restrições.

Onde encontrar as informações:

- Tipo da classe: No protege → na visualização da classe no item “*Role*”.
- Atributos da classe: No Protege → na visualização dos “*slots*” (classificado como *string*, *float*, *integer*, *boolean* ou *symbol*) de cada classe.

- Tipos do atributo: No Protege → na visualização dos “*slots*” (classificado como *string*, *float*, *integer*, *boolean* ou *symbol*) de cada classe, verificando a classificação do mesmo.
- Valores padrão do atributo: No Protege → na visualização dos “*slots*” (classificado como *string*, *float*, *integer*, *boolean* ou *symbol*) de cada classe, no item “*default value*”.
- *Template* do valor do atributo: No Protege → na visualização dos “*slots*” (classificado como *string*, *float*, *integer*, *boolean* ou *symbol*) de cada classe no item “*template value*”.

Contribuição:

- Prévia o modelo de dados.

Passo 9. Exibir a Organização Hierárquica dos Termos do Udl:

Montar uma visão como todas as classes da ontologia e suas subclasses.

Onde encontrar as informações:

- Hierarquia: No Protege → na visualização das classes.

Contribuição:

- Conhecer a informação de que os termos “sub-classes”, na ontologia, possuem todas as características do termo “superclasse”, possibilitando descrições de requisitos mais sucintas. Por exemplo: numa especificação dos atributos dos termos para construção de um cadastro, mostrando que os termos são hierarquicamente relacionados, não é necessário descrever, para ambos os termos, todos os atributos comuns.

4.4. Exibindo o Conhecimento Integrado

O objetivo desta seção é apresentar a quarta fase do método, na qual exibiremos um mapeamento entre as informações geradas na integração, ou seja, o conhecimento integrado, e as informações necessárias para o documento de requisitos do IEEE.

O resultado do trabalho de um engenheiro de requisitos são os requisitos para o desenvolvimento de um sistema. A maneira utilizada para apresentação destes requisitos é um documento de especificação de requisitos, e um dos mais utilizados é o documento do padrão IEEE 830 1998 de Especificação de Requisitos de Software.

O intuito deste trabalho é auxiliar o trabalho do engenheiro de requisitos, portanto, depois de geradas as informações resultantes da aplicação do método de integração de conhecimentos, realizamos o mapeamento de correspondência entre as informações resultantes da aplicação do método e os itens do documento do IEEE.

É importante salientar que as informações resultantes da aplicação do método não “preenchem” diretamente os itens do documento do IEEE. Estas informações não têm o intuito de gerar os requisitos para preencher o documento do IEEE. Seu intuito é facilitar o trabalho de engenheiro fornecendo, de maneira estruturada, as informações necessárias para que o mesmo desenvolva os requisitos ou, até mesmo, saiba onde buscar mais informações para descrever os requisitos do sistema.

Abaixo é exibida a tabela que contém o mapeamento da correspondência entre as informações resultantes da aplicação do método e os itens do documento do IEEE:

Resultado do método	Documento IEEE
Passo1(item 1) da Preparação para o método	1.4 - Escopo do Produto
Passo 1 da Preparação do Método	1.5 – Referências
Passo 3 da Preparação para o método e Passo1(item 4) da Preparação para o método	1.1 – Propósito
Resultado do Passo1 do Método	2.2 - Funções do Produto
Resultado do Passo2 do Método	2.3 - Classes de Usuários e Características
Resultado do Passo7 do Método	3.3 - Interfaces com outros Sistemas
Resultado do Passo5 do Método	5.5 – Regras de Negócio
Resultado do Passo3 da Preparação para o método e Resultado do Passo1.1 do Método e Resultado do Passo4 do Método	2.6 - Premissas e Dependências
Resultado do Passo2 do Método	4.1 - Funcionalidades do sistema
Resultado do Passo3 do Método	4.2 - Descrição e Prioridade
Resultado do Passo3 do Método	4.3 - Seqüências de Estímulos e Respostas
Resultado do Passo2 do Método e Resultado do Passo4 do Método e Resultado do Passo6 do Método	4.4.Requisitos Funcionais
Resultado do Passo1 (item 3) da Preparação para o método	6.6 – Apêndice A:Glossário
Resultado do Passo8 do Método	Apêndice B: Modelos de Análise

Tabela 3 - Mapeamento da correspondência entre as informações resultantes da aplicação do método e os itens do documento do IEEE.

Como podemos notar, alguns itens do documento gerado não foram diretamente mapeados para o documento do IEEE. Porém, estes itens serviram como fonte de informação para o engenheiro de requisitos para conhecer melhor o Udl e conseguir gerar requisitos mais aderentes ao negócio.

4.5.Verificando os Modelos Gerados

Além do auxílio no desenvolvimento dos requisitos, a proposta de junção do processo de negócio com a ontologia também proporciona a verificação de consistência entre eles. Por estes modelos possuírem informações semelhantes, é possível encontrar dados que se apresentem em um modelo e não constem no outro. Assim, durante a análise dos modelos, consegue-se fazer uma verificação destas informações inconsistentes entre os modelos de processos e da ontologia do negócio. E, desta forma, completá-los, ou seja, re-alimentar os modelos com o feedback obtido nesta verificação, para que os modelos fiquem mais completos e consistentes entre si.

4.6.Conclusão

Neste capítulo, apresentamos o método proposto neste trabalho para auxiliar na elicitação dos requisitos de sistemas de informação. Este método tem suas etapas resumidas na figura abaixo:

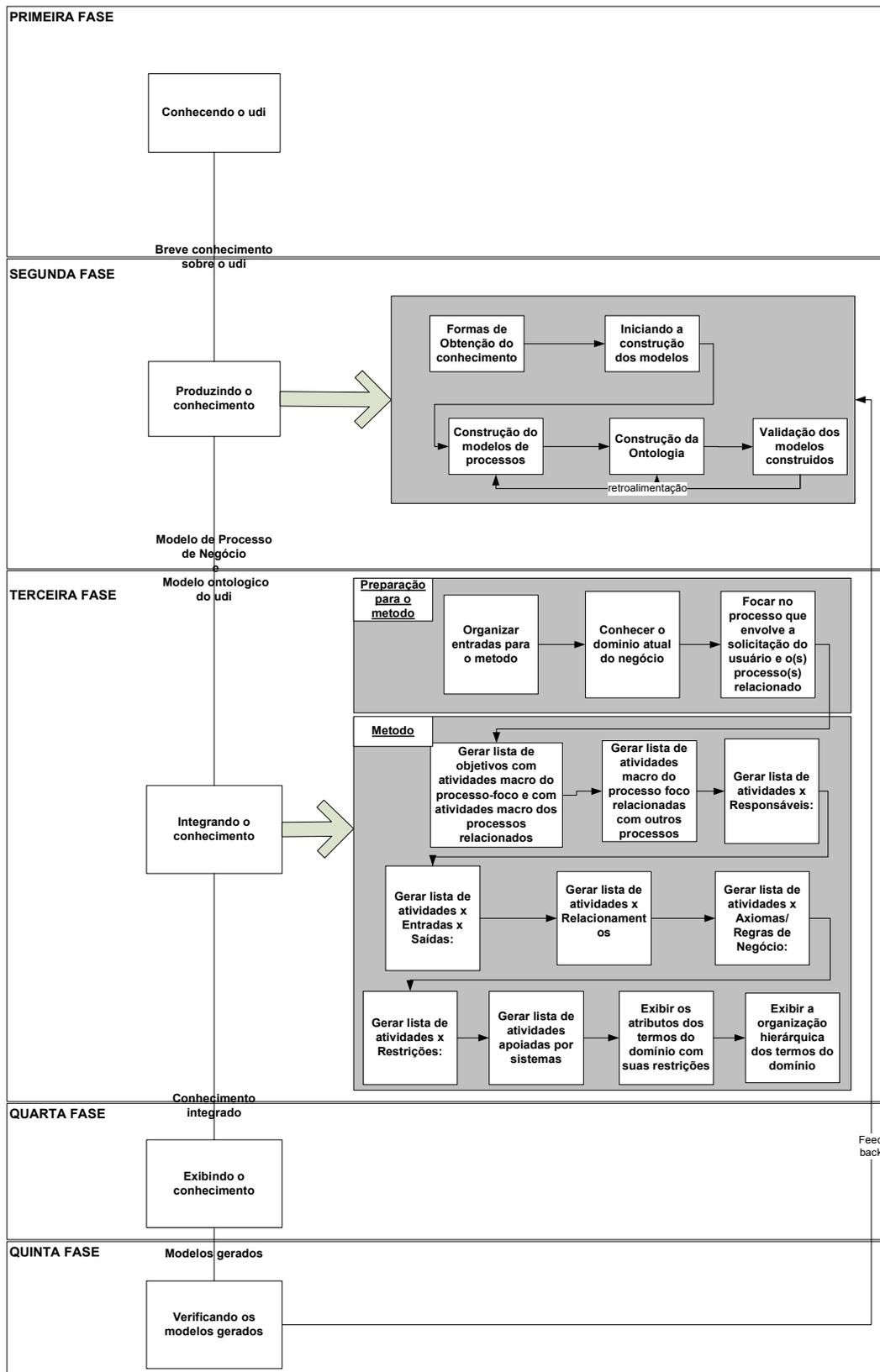


Figura 22 - Resumo do método proposto.

Salientamos que este método foi desenvolvido para as entradas especificadas no passo 1 da preparação para método de integração do conhecimento (subseção 4.3.2). Para utilizar qualquer outra entrada, é necessário realizar adaptações no método.

No próximo capítulo mostraremos, num estudo de caso realizado numa organização real, o passo-a-passo da aplicação do método proposto.