

# Definição de requisitos de *software* baseada numa arquitetura de modelagem de negócios

**DELMIR PEIXOTO DE AZEVEDO JUNIOR**  
Universidade Petrobras

**RENATO DE CAMPOS**  
UNESP

## Resumo

Não é uma tarefa fácil definir requisitos para os sistemas de software que darão suporte a um negócio, dada a dinâmica de mudanças nos processos. O levantamento de requisitos tem sido feito de forma empírica, sem o apoio de métodos sistematizados que garantam o desenvolvimento baseado nos reais objetivos do negócio. A engenharia de software carece de métodos que tornem mais ordenadas e metódicas as etapas de modelagem de negócios e de levantamento de requisitos de um sistema. Neste artigo é apresentada uma metodologia de desenvolvimento de software resultante da incorporação de atividades propostas para modelagem de negócios e levantamento de requisitos, baseadas em uma arquitetura de modelagem de negócios. Essas atividades tornam o desenvolvimento de software mais sistemático e alinhado aos objetivos da organização, e podem ser incorporadas em qualquer metodologia de desenvolvimento baseada no UP (Unified Process - Processo Unificado).

## Palavras-chave

Desenvolvimento de software, modelagem de negócios, processo unificado, modelagem de requisitos.

## *Software requirements definition based on a business modeling architecture*

## Abstract

*It is not an easy task to define the requirements to software systems that support businesses by reason of the dynamic of changes in business processes. The activity of finding systems requirements has been performed empirically, without systematic methods that can fulfill business objectives. The software engineering needs methods that become the activity of finding systems requirements, in a software development process, a more systematic activity. In this article is presented a software development methodology that integrates activities proposed to business modeling and requirements definition, based on a business modeling architecture. The activities proposed become the software development more systematic and focused on the organization objectives, and can be incorporated in any methodology of development based on the UP (Unified Process).*

## Key words

*Software development, business modeling, unified process, requirements definition.*

## INTRODUÇÃO

Quanto mais rápido um negócio puder alterar seus processos e os sistemas de informação que lhe dão suporte, mais preparado estará para reagir a eventos de concorrência no mercado. O levantamento de requisitos é a etapa do desenvolvimento de sistemas de informação responsável por identificar e modelar as necessidades do negócio a serem atendidas pelos sistemas de informação, e é, portanto, uma atividade cada vez mais relevante em um dinâmico cenário.

As organizações empresariais modernas precisam estar em constante evolução para manterem-se competitivas.

São necessárias frequentes reformulações e inovações nos processos de negócio e, conseqüentemente, nos sistemas de informação que lhes dão suporte. A integração entre os objetivos do negócio, os processos de negócio e sistemas de informação é um fator determinante da dinâmica necessária à organização e também um desafio aos gerentes. Nesse cenário, os sistemas de informações são os habilitadores do negócio e, portanto, precisam estar alinhados com os reais objetivos deste negócio. Existem vários métodos, técnicas e ferramentas de modelagem para facilitar o entendimento e a análise da complexidade das organizações modernas (KALPIC; BERNUS, 2002; LI; WILLIAMS, 2002; VERNADAT, 2002). Essas facilidades são utilizadas na tentativa de tornar a realidade organizacional, complexa e abstrata, mais compreensível. Também existem várias metodologias para o desenvolvimento de sistemas de informação. O que se observa, no entanto, é a falta de integração da análise nos dois domínios, o do negócio e o dos sistemas que lhe fornecerão suporte (ODEH; KAMM, 2003; SHEN, 2004).

O UP (Unified Process – Processo Unificado) é uma das metodologias de desenvolvimento de sistemas de software que vem obtendo destaque entre as demais. No entanto, mesmo no UP, a atividade de levantamento de requisitos ainda é um processo empírico, não considerando de forma sistemática a importância do foco nos objetivos do negócio. Neste contexto, evidencia-se a necessidade de maior aproximação entre o levantamento de requisitos de sistemas de softwares e as reais necessidades do negócio em processos ou metodologias de desenvolvimento. No paradigma da orientação a objeto, a análise de requisitos tem sido feita com base num elemento de modelagem da UML (Unified Modeling Language) chamado Caso de Uso. Embora existam algumas heurísticas propostas para identificação de casos de uso, como as apresentadas em Schneider e Winters

(1998), em Jacobson, Booch e Rumbaugh (1999), e em Lilly (1999), não existem métodos estabelecidos que tornem esta atividade mais sistemática.

O alinhamento entre requisitos de software e as reais necessidades de informatização da empresa pode ser melhorado e sistematizado através de técnicas de modelagem de negócios (ou de empresa). A tecnologia da orientação a objeto, através da UML, permite a integração da representação de modelos nos dois domínios, negócio e software. Porém, existe a falta de metodologias completas que alinhem de forma sistemática o levantamento de requisitos de software às reais necessidades de um negócio, em função de seus objetivos (AZEVEDO JR., 2003).

**S**istemas de informações são os habilitadores do negócio e, portanto, precisam estar alinhados com os reais objetivos deste negócio.

Pode-se dizer que o UP como a união das boas práticas do desenvolvimento de software é base para a definição de várias metodologias encontradas no mercado. Este trabalho descreve uma metodologia resultante da incorporação de algumas atividades para o levantamento de requisitos baseadas na arquitetura de modelagem de negócios de Eriksson e Penker (2000), a fim de tornar mais sistemática esta etapa do desenvolvimento de sistemas. As atividades propostas podem ser aplicadas em qualquer metodologia de desenvolvimento de sistemas que se fundamente nos mesmos princípios estabelecidos no UP.

Em relação à metodologia de pesquisa, fundamentou-se em extensa revisão bibliográfica; na elaboração e descrição de atividades baseadas em uma arquitetura de modelagem de negócios para metodologias de desenvolvimento de software; na aplicação das atividades propostas na metodologia de desenvolvimento de sistemas de uma empresa; e em um teste com o desenvolvimento de parte de um sistema de informação, visando a avaliação da proposta.

Ao longo deste artigo serão propostas atividades de modelagem baseadas em uma arquitetura de modelagem de negócios, visando inseri-las de forma integrada em metodologias de desenvolvimento de sistemas de informações baseadas no Processo Unificado (UP), e assim torná-las capazes de melhor definir e alinhar, de forma sistemática, os requisitos do sistema aos reais objetivos do negócio. Desta forma, na próxima seção são apresentadas definições relativas à Engenharia de Requisitos e ao UP, exibidos conceitos relacionados à modelagem de processos de negócios com a UML e questões relacionadas com a identificação de casos

de uso de negócio. Na seção seguinte são descritas as atividades propostas a serem inseridas em metodologias baseadas no UP. Após, são apresentados os fluxogramas e descritas as atividades de toda uma metodologia de desenvolvimento de software com as atividades propostas, assim como os artefatos resultantes, e as considerações finais.

## ENGENHARIA DE REQUISITOS E O UP

A engenharia de software se produz através de um conjunto de fases. Cada uma das fases pode envolver métodos, ferramentas e procedimentos, cujas formas de estruturação são citadas como modelo de engenharia de software (PRESSMAN, 2002). Ainda segundo Pressman (2002), independentemente do modelo de desenvolvimento de software, o processo contém três fases genéricas: definição, desenvolvimento e manutenção.

Quatro modelos de engenharia de software têm sido amplamente discutidos: o ciclo de vida clássico (ou cascata), a prototipação, o modelo espiral e as técnicas de Quarta Geração (PRESSMAN, 2002). Atualmente um novo modelo tem sido bastante usado: o modelo iterativo e incremental (JACOBSON; BOOCH; RUMBAUGH, 1999; PAULA FILHO, 2001).

A análise de requisitos é uma etapa presente na fase de definição do software, independentemente do modelo de engenharia de software adotado. Paula Filho (2001) afirma que a engenharia de requisitos é formada por um conjunto de técnicas empregadas para levantar, detalhar, documentar e validar os requisitos de um produto de software. Assim, é possível definir a engenharia de requisitos como um campo da engenharia de software que visa a aplicação de técnicas de engenharia em métodos de análise de requisitos, que efetua a ligação entre a necessidade de informatização de processos e o projeto do software que atenderá a tais necessidades.

No decorrer das duas últimas décadas, uma série de métodos de análise e especificação de requisitos foi desenvolvida, sendo poucas as propostas que visam a sistematização da definição de requisitos de forma menos subjetiva (SANTANDER, 2002).

O UP (Processo Unificado) é um processo estabelecido para o desenvolvimento de software resultado de três décadas de desenvolvimento e de uso prático. Jacobson, Booch e Rumbaugh (1999) apresentam as origens do UP no processo Objectory, que teve sua primeira versão apresentada em 1987, passando pelas contribuições do Processo Rational Objectory (1997), até chegar ao Processo Unificado da Rational – RUP (KRUCHTEN, 2003). O propósito do UP, como qualquer outro processo de desenvolvimento, é determinar um conjunto de atividades necessárias para transformar requisitos em sistemas de software. Neste

caso, utiliza a UML como linguagem para a modelagem dos artefatos de software produzidos ao longo do processo de desenvolvimento. O UP é fundamentado em três boas práticas: dirigido por caso de uso, centrado em arquitetura, iterativo e incremental. Os casos de uso definidos para um sistema são a base de todo o processo de desenvolvimento. A partir de uma perspectiva de gerenciamento, o ciclo de vida de software do UP é dividido em quatro fases seqüenciais (Concepção, Elaboração, Construção e Transição), sendo que cada fase refere-se a uma determinada meta a ser atingida ao longo do desenvolvimento.

Cada uma das quatro fases do UP é adicionalmente dividida em iterações e finalizada com um ponto de checagem que verifica se os objetivos daquela fase foram alcançados. Toda iteração é organizada em termos de *workflows* de processo, que são conjuntos de atividades realizadas pelos responsáveis pela produção dos artefatos. Os principais *workflows* de processo do UP são Levantamento de Requisitos, Análise, Projeto, Implementação, Teste e Implantação, sendo que o RUP se diferencia, principalmente, em relação ao *workflow* de Modelagem de Negócio.

A Figura 1, conhecida como “Gráfico das Baleias”, exemplifica como os *workflows* podem ser utilizados, com maior ou menor ênfase, em cada fase de desenvolvimento. Nela podem ser observadas duas dimensões:

- O eixo horizontal representa o tempo e mostra os aspectos do ciclo de vida do processo à medida que se desenvolve;
- O eixo vertical representa os *workflows*, que agrupam as atividades de maneira lógica, por natureza.

A UML foi adotada pela OMG (Object Management Group) em 1997 como linguagem padrão para a modelagem de sistemas orientados a objeto. É uma linguagem para especificação, visualização, construção e documentação de artefatos de sistemas de software, como também para a modelagem de negócios e outros sistemas não necessariamente relacionados a software, e representa uma coleção das melhores práticas de engenharia que comprovaram bom êxito na modelagem de sistemas grandes e complexos (OMG, 1997). Os principais diagramas da UML são: Diagrama de Classes, Diagrama de Objetos, Diagrama de Casos de Uso, Diagrama de Seqüência, Diagrama de Colaborações, Diagrama de Gráficos de Estados, Diagrama de Atividades, Diagrama de Componentes, Diagrama de Implantação.

A UML também permite a ampliação da linguagem de maneira controlada, através de mecanismos de extensão, de forma a expressar todas as nuances possíveis de todos os modelos em qualquer domínio de análise (exemplo: análise de negócios), fornecendo alguma flexibilidade.

## MODELAGEM DE PROCESSOS E CASOS DE USO DE NEGÓCIOS

Processo de negócio pode ser definido como um conjunto de atividades conexas que toma um insumo de entrada e o transforma para criar um resultado de saída. Teoricamente, a transformação que nele ocorre deve adicionar valor e criar um resultado que seja mais útil e eficaz ao cliente que recebe o serviço ou produto (JOHANSSON *et al.*, 1995). Feliciano Neto (1996) observa que a grande dificuldade encontrada pelos gerenciadores de projetos de implantação de sistemas de informação para cumprir cronogramas e levantamento de custos está relacionada à dificuldade de delimitação de contexto do sistema. Decompor a empresa em funções de negócio, sem se preocupar com uma visão organizacional, facilita a definição do contexto onde o sistema de informação irá atuar.

As funções do negócio propostas por Feliciano Neto (1996) mostram-se, na prática, como a descrição de processos de negócios. Assim sendo, é necessário que as metodologias de modelagem de negócios (ou de empresas) atuais tenham em sua essência o tratamento baseado em processos.

Várias são as técnicas, metodologias e notações existentes para a modelagem dos processos de empresas (KOSANKE *et al.*, 1999; KOSANKE; NELL, 1999; SCHEER, 2000; VERNADAT, 1996). No entanto, definir modelos de negócio é uma atividade complexa porque os usuários têm diferentes necessidades e estas mudam constantemente (KALPIC; BERNUS, 2002; KIRIKOVA, 2000). Para uma empresa ser adaptável a mudanças, é necessário que tenha uma descrição simples e unificada de suas entidades e, em-

bora este seja o objetivo de muitos esforços para modelagem, o que se tem ainda hoje é uma descrição tipicamente extensa, inflexível e frágil (MARSHALL, 1999).

A UML (Unified Modeling Language) encontra-se atualmente consagrada para a modelagem de sistemas de software e tem sido proposta para a modelagem de empresas através de seus mecanismos de extensão e, segundo a OMG (1997), esses mecanismos permitem adequá-la a novidades e domínios específicos. As extensões definidas pelos usuários na UML se dão através de estereótipos, valores associados e restrições que coletivamente estendem-na e adaptam-na a um domínio específico. Nas subseções seguintes são apresentadas três propostas de extensões para a modelagem de negócios com a UML, com maior ênfase na descrição da proposta de Eriksson e Penker (2000), já que apresentam um método mais sistemático e completo para definir e alinhar os requisitos de sistemas aos objetivos do negócio.

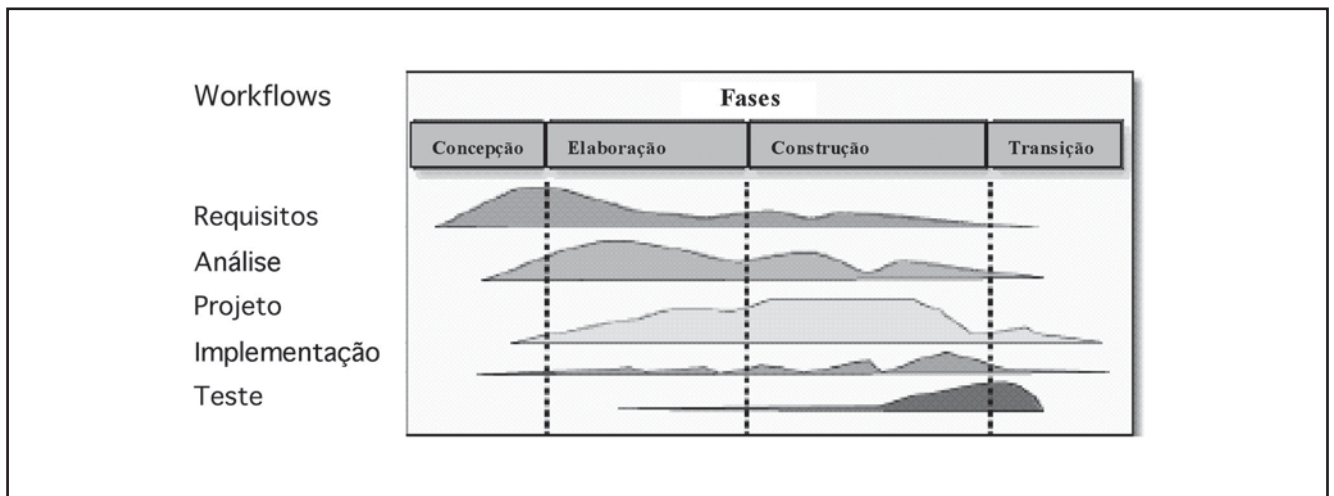
### Proposta da OMG

A OMG (1997), em sua publicação “UML extension for business modeling”, descreve uma extensão da UML para a modelagem de negócios em termos de seus mecanismos de extensão. O documento, porém, não é uma tentativa de descrever completamente os novos conceitos e notações para a modelagem de negócios, apenas expõe os *stereotypes* que podem ser usados para adaptar o uso da UML. Já a UML 2.0 apresenta algumas inovações em relação ao mecanismo de extensão e à modelagem de processos de negócios.

### Proposta de Marshall

Marshall (1999) apresenta uma proposta de extensão da UML para a modelagem de negócios em que sugere um

Figura 1: Relação entre workflows e fases.



Fonte: Adaptado de KRUCHTEN, 2003.



metamodelo para identificar e descrever conceitos através dos quais sistemas de negócios são delineados, utilizando a UML para ilustrar tais conceitos, e propõe uma modelagem baseada em quatro elementos centrais: propósito, processo, entidade e organização.

### Proposta de Eriksson e Penker

Eriksson e Penker (2000) propõem uma técnica que estende a UML fundamentada em processos e orientação a objetos para construir arquiteturas de negócios, baseando-se no uso de extensões da UML para representar processos, recursos, regras e objetivos. Afirmam que sua técnica não deve ser vista como um conjunto definido de extensões para negócios, mas como uma base para que desenvolvimentos e adaptações possam ser feitos (por arquitetos de negócios) para situações específicas de modelagem.

**O**UP é fundamentado em três boas práticas: dirigido por caso de uso, centrado em arquitetura, iterativo e incremental.

As propostas de Eriksson e Penker (2000) formam uma Arquitetura baseada na linguagem UML para a modelagem de negócios onde podem ser adicionados *stereotypes*, *tagged values* e *constraints* convenientes para cada linha de negócios. Fundamentam-se na hipótese de que um negócio pode ser modelado através de objetos e dos relacionamentos entre estes, que a arquitetura de modelagem fornece vistas para a modelagem com foco em aspectos significativos e cada vista pode ser modelada por um ou mais tipos de diagramas. Oferecem as seguintes vistas para a Arquitetura:

- **Visão do Negócio (*Business vision*):** modela conceitos e objetivos a serem seguidos de acordo com a estratégia do negócio;
- **Processo do Negócio (*Business process*):** modela os processos de negócio e seus relacionamentos com os recursos a serem seguidos para atingir os objetivos;
- **Estrutura do Negócio (*Business structure*):** modela a estrutura dos recursos (físicos, informacionais, humanos);
- **Comportamento do Negócio (*Business behavior*):** modela o comportamento e a interação entre recursos e entre processos.

O Diagrama de Processos de Negócio e o Diagrama de Linha de Montagem têm um papel importante na ligação entre requisitos de negócios e casos de uso. O primeiro descreve os processos de negócio através de suas relações com Objetos (Objetivos, Entradas, Saídas, Fornecedores e Controles). Já o Diagrama de Processos de Negócio

mostra no seu topo um processo de negócio e abaixo um número de pacotes horizontais chamados pacotes de linha de montagem, cada um representando um grupo de objetos, que podem ser de uma classe específica ou de diferentes classes.

Um pacote de linha de montagem é um item pacote da UML estereotipado para <<linha de montagem>> e desenhado como um longo retângulo horizontal, e o diagrama de linha de montagem pode ser usado como técnica para levantamento de casos de uso do sistema ou de sistemas que darão suporte aos processos de negócio. Um exemplo de um diagrama de linha de montagem é apresentado na Figura 2 em que é mostrada a identificação de necessidades de leituras e gravações em sistemas existentes e a serem desenvolvidos, e a conseqüente identificação de casos de uso relativos ao processo de negócio “Verificação de Documentação”. O

propósito deste diagrama é mostrar como o processo (na parte superior do diagrama) utiliza e gera objetos na linha de montagem. A referência de um objeto a uma linha de montagem é indicada por um fluxo de objeto (representado por uma linha tracejada na UML) entre o processo

e o objeto dentro do pacote na linha de montagem. A identificação dos casos de uso através desta técnica mostra-se mais adequada, pois faz com que os objetivos dos atores, e conseqüentemente os requisitos do sistema em forma de casos de uso, estejam alinhados aos objetivos globais do negócio, já que são analisados com base nos processos de negócio e estes, por sua vez, foram definidos em função dos objetivos do negócio.

### Identificação de Casos de Uso do Negócio

A forma de representação e concepção dos processos de negócio não é um ponto comum nas propostas de modelagem de negócio com UML. São observadas duas linhas distintas: a que defende a representação e definição dos requisitos dos negócios através de casos de uso, que corresponderia a um processo; e a que discorda de tal representação, propondo que se faça primeiramente a modelagem dos processos para definição dos requisitos de negócios, e só depois se definam os requisitos de sistemas de informações através de casos de uso.

A primeira linha foi introduzida pela OMG em 1997 na primeira versão da especificação da UML, posteriormente aprimorada no RUP, e defende a modelagem de processos de negócio através de modelos de casos de uso de negócio. Assim como o caso de uso para um sistema de software, o modelo de caso de uso de negócio representa o sistema (agora, o negócio) na perspectiva do usuário, e esboça como ele agrega valor a quem o utiliza.

Um modelo de caso de uso de negócio descreve os processos de negócio de uma organização em termos de casos de uso e atores de negócio, correspondentes a processos de negócio e clientes, respectivamente (JACOBSON; BOOCH; RUMBAUGH, 1999; OMG, 1997).

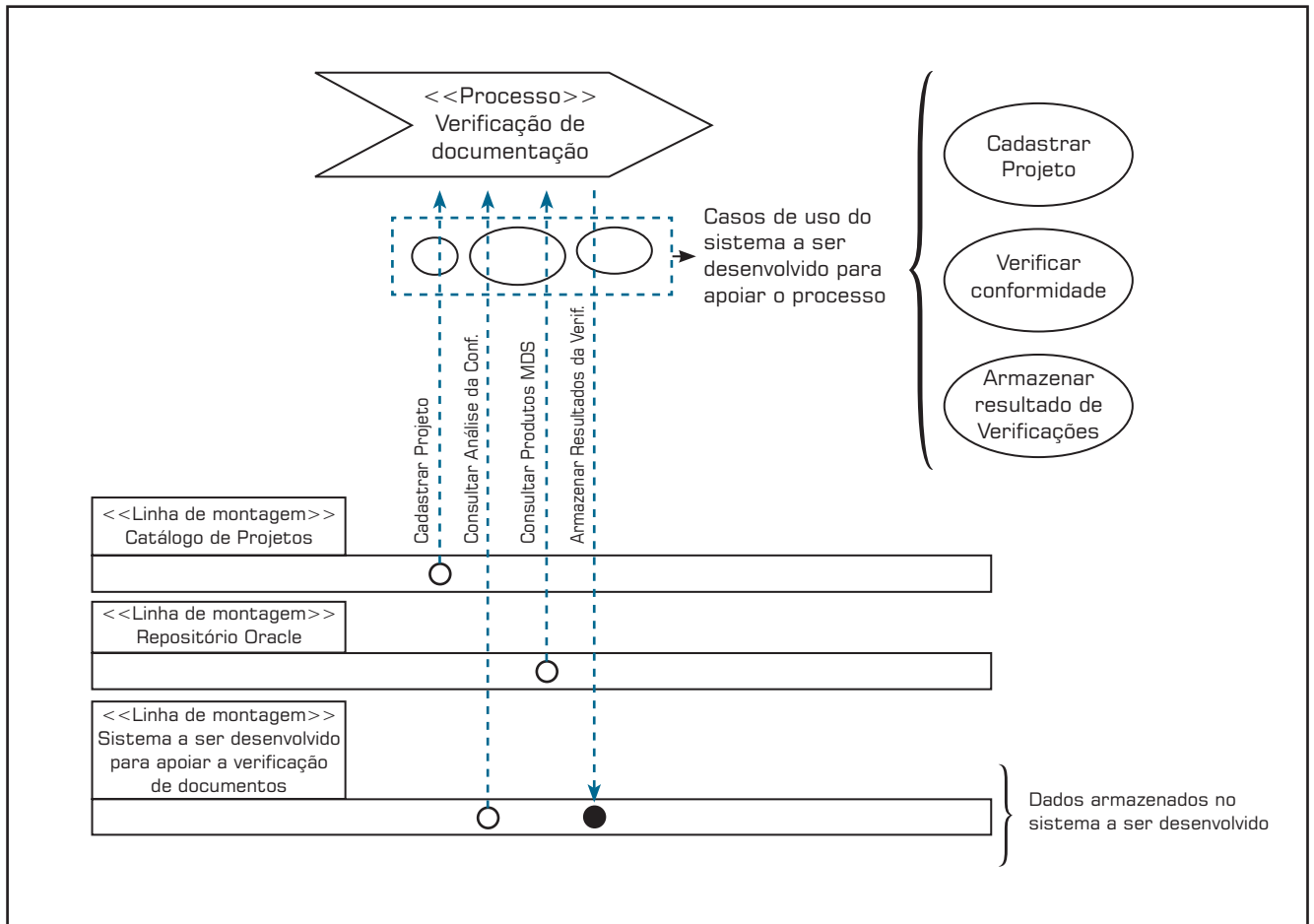
A segunda linha corresponde às iniciativas de Marshall (1999) e Eriksson e Penker (2000), que sustentam não serem os processos de negócio bem representados pelos casos de uso, pois estes servem para representar um domínio fechado correspondente a determinados requisitos de usuários e não podem ser vistos simplificarmente como requisitos de clientes.

Isto significa que um caso de uso nem sempre é equivalente a um processo, pois fornece um serviço que é requerido como parte de um processo exterior ao sistema de software. Um caso de uso é completamente implementado no software, enquanto um processo é, em geral, apenas parcialmente implementado no software (o termo caso de uso é uma abstração para definir comunicação entre atores

e um sistema de software). Os casos de uso podem ser considerados como as especificações dos serviços que o sistema de software fornece ao processo de negócio (ERIKSSON; PENKER, 2000), conforme ilustrado na Figura 2.

É necessário, portanto, que a modelagem dos processos de negócio dê ênfase ao fluxo de informações entre os processos ao longo da cadeia de valores que busca atingir os objetivos globais do negócio. Atentando para isto, a segunda linha propõe a utilização de diagramas de atividades para a representação dos processos de negócio no domínio da modelagem de negócio, exibidos através do diagrama de atividade da UML, em que os itens atividade são estereotipados como processos, conforme o proposto por Eriksson e Penker (2000). Faltam, porém, métodos para a sistematização das etapas de definição dos processos de negócios e de definição dos requisitos de software em processos de desenvolvimento de software (tal como o UP), que considerem o alinhamento entre os objetivos do negócio e os requisitos do software.

Figura 2: Exemplo de identificação de casos de uso no Diagrama de Linha de Montagem.



## ATIVIDADES PARA SISTEMATIZAÇÃO DO LEVANTAMENTO DE REQUISITOS

O UP une as boas práticas no desenvolvimento de software e é base para a definição de várias metodologias encontradas no mercado, porém sente-se a falta de métodos e ferramentas adequadas para o levantamento dos requisitos do negócio. Assim, procura-se inserir nesse processo atividades para o levantamento de requisitos de sistemas baseados em uma arquitetura modelagem de negócio, com o intuito de tornar mais sistemática esta etapa do desenvolvimento. Para isso, é proposto um conjunto de atividades a ser inserido no UP para a modelagem de negócio, com base na técnica de modelagem proposta por Eriksson e Penker (2000), incorporando em processos de desenvolvimento (neste caso, o UP) as características e vantagens destacadas no final da seção anterior. Também são propostas atualizações em algumas atividades preestabelecidas no UP, de forma a poderem ser aplicadas a qualquer metodologia que nele se baseie.

**A** forma de representação e concepção dos processos de negócio não é um ponto comum nas propostas de modelagem de negócio com UML.

A técnica de construção de arquiteturas de negócio proposta por Eriksson e Penker (2000) é, dentre as propostas de modelagem de negócio com UML pesquisadas, a única que aborda de forma sistemática a passagem da arquitetura de negócio para uma arquitetura de software que dê suporte à primeira. Porém, os autores não exploram a sistematização desta passagem no contexto de um processo ou metodologia de desenvolvimento de sistemas. O UP é dividido em quatro fases (levantamento de requisitos, análise, projeto, implementação e teste) e possui *workflows* que devem ser executados em todas elas, com uma abordagem específica das atividades do fluxo de trabalho para cada uma. As atividades do *workflow* de levantamento de requisitos existem em todas as fases do desenvolvimento, com ênfase nas fases de Concepção e de Elaboração. Na concepção, há destaque para a identificação dos requisitos, mas não para a especificação detalhada dos mesmos. A fase Elaboração (ver Figura 1) é a que requer maior concentração de esforços e detalhes na atividade de especificação de requisitos.

Um método de levantamento de requisitos que derive dos casos de uso de uma arquitetura de software no UP

deve definir atividades e seus fluxos, bem como o estado esperado dos artefatos gerados por estas atividades em cada fase do processo (Concepção, Elaboração, Construção, e Transição), considerando-se uma estrutura iterativa e incremental, bem como as atividades já definidas nesta estrutura. A aplicação da técnica de Eriksson e Penker (2000) ao UP se realiza, portanto, através da criação de um *workflow* completo para a modelagem de negócio (que não existe originalmente no UP), da criação de novas atividades, e da atualização de algumas atividades anteriormente já estabelecidas nos *workflows* do UP, com a preocupação de integrá-los. São definidas também as abordagens que cada atividade proposta deve ter nas fases de Concepção e de Elaboração. As demais atividades do UP são consideradas inalteradas, como originalmente proposto (JACOBSON; BOOCH; RUMBAUGH, 1999). A seguir, são descritas as atividades propostas para os três *workflows* (Modelagem de Negócio, Levantamento de Requisitos e Análise) e as abordagens das fases de Concepção e de Elaboração de um processo de desenvolvimento baseado no UP (AZEVEDO JR.; CAMPOS, 2004).

### Workflow para Modelagem de Negócio

O *workflow* definido para a modelagem de negócio é apresentado na Figura 3. A seguir, as descrições de cada atividade proposta e seus respectivos produtos:

- **Modelar os Objetivos do Negócio:** a modelagem dos objetivos deve identificar os principais objetivos e subobjetivos do negócio em uma estrutura hierárquica que permita a visualização de dependência entre tais objetivos. Este modelo servirá de base para a definição dos processos de negócio. A modelagem dos objetivos do negócio deve ser feita com base em entrevistas realizadas com os conhecedores do negócio. Produto resultante: Diagrama de Modelo de Objetivos.
- **Modelar os Processos de Negócio:** os processos de negócio devem ser definidos buscando-se a realização dos objetivos identificados no Modelo de Objetivos do Negócio. Porém, não é necessário haver uma relação um para um entre processos de negócios e objetivos do negócio, pois muitos processos auxiliares não estarão necessariamente relacionados a um objetivo do Modelo de Objetivos do Negócio. Entrevistas com os envolvidos no negócio também devem ser realizadas para fornecer subsídios à definição dos processos de negócio. Produto resultante: Diagrama de Processos de Negócio.
- **Modelar os Recursos Envolvidos:** os recursos, informações e unidades organizacionais devem ser modelados através dos diagramas da Vista de Estrutura do Negócio. A modelagem destes elementos deve ser

feita paralelamente às atividades de Modelagem de Processos de Negócio, a fim de melhor entender os termos relacionados ao negócio e, conseqüentemente, ter maior consistência na modelagem do mesmo. Produtos resultantes: Diagramas de Modelos de Recursos, de Informações e de Organização.

- **Modelar Comportamento dos Recursos:** um Diagrama de Estados de Recurso pode ser criado para facilitar a determinação dos processos de negócio quando este se caracteriza por refinamentos de um mesmo objeto ao longo da cadeia de valor. Por exemplo, considerando-se um negócio de vendas, o pedido pode ser abordado como um objeto cujo estado vai sendo alterado (refinado) ao longo de toda a cadeia de valor, desde a abertura do pedido até a confirmação do pedido entregue ao cliente. Em um caso como este, a identificação dos estados possíveis de tal objeto (como pedido solicitado, pedido em verificação de estoque, pedido em produção, pedido em expedição e pedido entregue) pode facilitar a identificação dos processos de negócio necessários ao cumprimento das mudanças de estado do produto. Produto resultante: Diagrama de

Estado de Recurso e Diagramas de Interação de Recursos e de Estados.

- **Definir Papéis e Responsabilidades:** cada processo de negócio deve possuir um responsável, uma vez que geralmente não estará ligado a uma única unidade organizacional, mas passando por mais de uma delas. Cada processo, por sua vez, define um fluxo de eventos que pode envolver um ou mais atores. É necessário definir quais atores agem em cada um dos processos. Isto pode ser feito através de uma análise do fluxo de eventos e associação destes aos atores envolvidos no processo. Produto resultante: Tabela de Papéis e Responsabilidades.

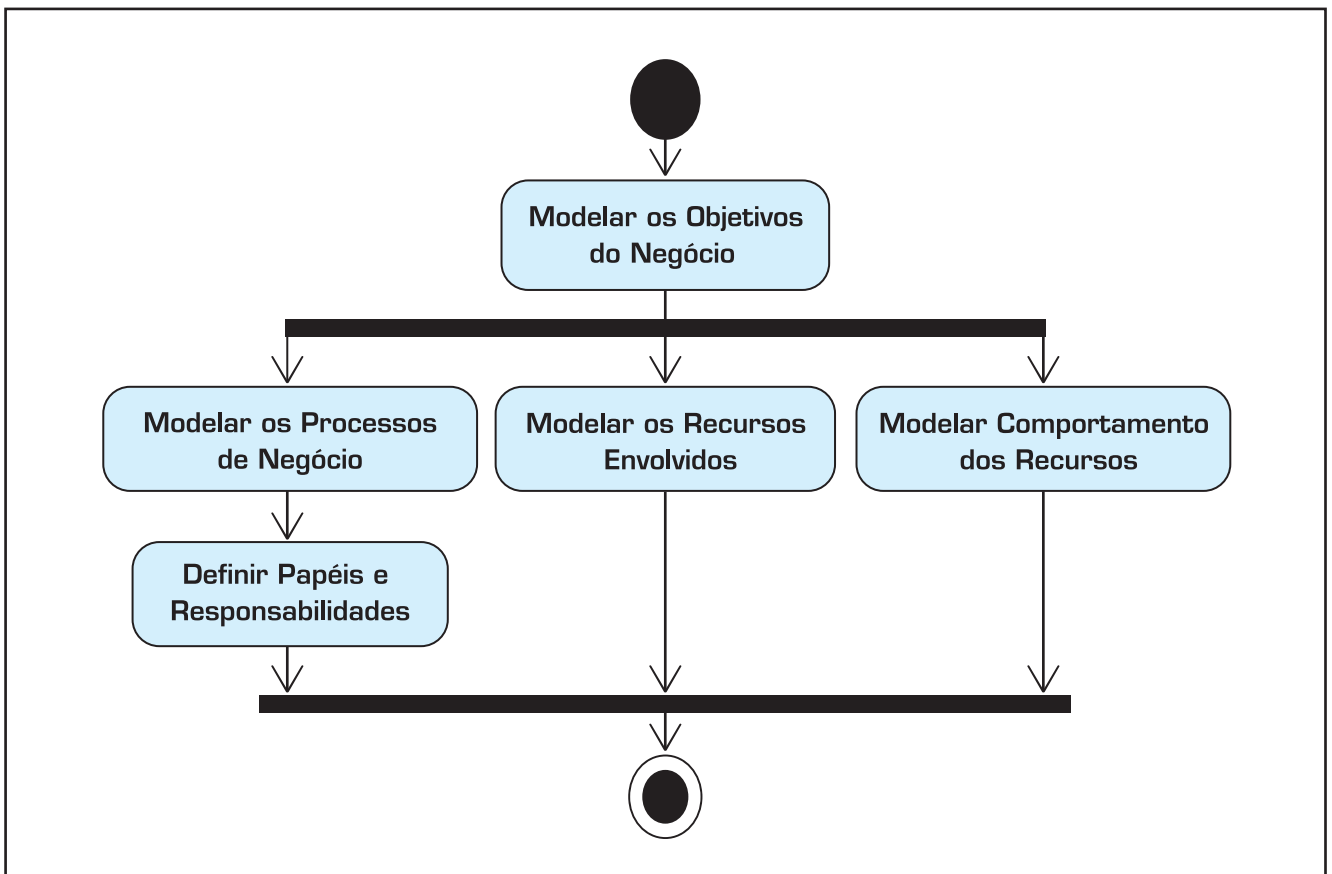
#### Abordagens das atividades propostas para Modelagem de Negócios

As abordagens das atividades propostas em cada fase de desenvolvimento estão descritas a seguir:

##### *Modelar os Objetivos do Negócio*

Na fase de Concepção – o Modelo de Objetivos deve abordar todos os objetivos relevantes ao projeto, desde os

Figura 3: *Workflow* para a Modelagem de Negócio.





de nível mais estratégico até os que estejam ao nível dos objetivos de processos de negócio.

Na fase de Elaboração – deve-se atualizar o modelo de objetivos em função de possíveis esclarecimentos posteriores.

#### **Modelar os Processos de Negócio**

Na fase de Concepção – deve-se identificar os principais processos de negócio, suas relações com os recursos (entradas, saídas, fornecedores, controles e objetivo), e a seqüência de execução dos mesmos. Porém, não é necessária a descrição detalhada do fluxo de eventos ocorrido internamente no processo.

Na fase de Elaboração – detalhar o fluxo de eventos dos processos que serão abordados na iteração atual.

#### **Modelar os Recursos Envolvidos**

Na fase de Concepção – devem ser modelados todos os recursos significativos identificados no Modelo de Processo de Negócio definido na fase Concepção, de forma a analisar a dependência entre tais recursos e suas propriedades.

Na fase de Elaboração – modelar todos os recursos significativos identificados durante o detalhamento dos fluxos de eventos de cada processo de negócio.

#### **Modelar Comportamento dos Recursos**

Na fase de Concepção – modelar o comportamento de recursos nos casos em que ocorram várias alterações ao longo dos processos de negócio, já que esta dinâmica de alterações precisa ser melhor entendida.

Na fase de Elaboração – detalhar os Diagramas de Estado de Recursos, caso tenham sido criados na fase Concepção, com base no detalhamento dos fluxos de evento dos processos.

#### **Definir Papéis e Responsabilidades**

Na fase de Concepção – definir apenas os responsáveis por cada processo de negócio, sejam eles unidades organizacionais ou funções.

Na fase de Elaboração – definir os papéis (atores) associados aos eventos que ocorrem no fluxo de evento de cada processo de negócio.

#### **Workflow de Levantamento de Requisitos**

A seguinte atividade foi adicionada ao *Workflow* de Levantamento de Requisitos:

– **Identificar Necessidades de Informatização:** nesta atividade é necessário associar os processos de negócio aos sistemas de informação que lhes dão suporte e, assim, identificar a possível necessidade de novos sistemas de informação, com a caracterização de carências de suporte automatizado de informação e operações aos processos.

Sugere-se a utilização do Diagrama de Linha de Montagem como base para a realização desta atividade. Produto resultante: Diagrama de Linha de Montagem com os pacotes de linha de montagem identificados.

A atividade Encontrar Atores e Casos de Uso, já existente no UP, foi atualizada:

– **Derivar Casos de Uso dos Processos de Negócio:** os casos de uso devem ser identificados com base nos processos de negócio. Esta atividade deve resultar em uma Relação de Casos de Uso na qual deve-se associar cada caso de uso identificado ao processo (ou processos) de negócio a que este atende. Sugere-se a utilização do Diagrama de Linha de Montagem como base para a realização desta atividade (exemplo na Figura 2). A identificação dos casos de uso no Diagrama de Linha de Montagem se dá através do agrupamento de referências (entre o processo e os sistemas) de mesma natureza. Produto resultante: Diagrama de Linha de Montagem com casos de uso identificados.

#### **Abordagens das atividades propostas para Levantamento de Requisitos**

As abordagens destas atividades em cada fase de desenvolvimento considerada são descritas a seguir:

##### **Identificar Necessidades de Informatização**

Na fase de Concepção – identificar sistemas de software que dão suporte aos processos de negócio, bem como identificar a necessidade de novos sistemas e subsistemas. Utilizar o Diagrama de Linha de Montagem como recurso de apoio ao desenvolvimento desta atividade. Deve-se começar com os pacotes em um alto nível de abstração, representando os sistemas já existentes e a natureza das informações das referências que estes fazem a cada processo de negócio analisado. Aplica-se, então, uma primeira avaliação quanto à natureza das informações e às operações necessárias ao processo e o atendimento destas pelos sistemas existentes, de forma a identificar tipos de informações e operações que não estão sendo mantidas pelos sistemas de software disponíveis. Tais necessidades de informação e de operações devem ser referenciadas a um outro pacote representativo do sistema (ou sistemas) a ser construído para atender a tais requisitos.

Na fase de Elaboração – deve-se atualizar e aprofundar a análise iniciada na Concepção, com base na descrição do fluxo de eventos dos processos. É necessário avaliar cada fluxo de evento e identificar eventos que podem ser auxiliados por sistemas de informação, mas que ainda não são. Tais auxílios devem ser representados como referências do processo aos sistemas que os realizam. Considerando

o escopo de um sistema identificado na concepção, deve-se representar cada linha de montagem como uma classe do sistema e distribuir a responsabilidade entre as classes através das referências feitas a cada uma delas pelos processos. Cada evento a ser informatizado deve resultar em uma referência à classe que o realizará e quando esta não existir, deverá ser criada como uma nova linha de montagem. Este processo deve ser feito respeitando-se o conceito de encapsulamento.

### **Derivar Casos de Uso dos Processos de Negócio**

Na fase de Concepção – a atividade deve objetivar a identificação dos casos de uso arquiteturalmente significativos. Estes casos de uso representam funcionalidades num alto nível de abstração e servem como base para a definição da vista lógica da arquitetura do software que os realizará.

Na fase de Elaboração – a atividade visa identificar todos os casos de uso do sistema com base na análise das referências entre os processos detalhados e os sistemas de software que os apoiarão.

### **Workflow para análise**

A atividade Realização de Casos de Uso, já existente no UP, foi atualizada:

- **Identificar Classes a partir da Arquitetura de Negócio:** esta atividade consiste na identificação de Classes a partir de modelos da Vista de Estrutura do Negócio e da Vista de Processos de Negócio. Produto resultante: Diagrama de Classes.

### **Abordagem da atividade proposta para análise**

As abordagens da atividade proposta nas fases de desenvolvimento consideradas são descritas a seguir:

Na fase de Concepção – busca-se a identificação das principais Classes do sistema, com base na análise dos Modelos de Recursos e de Informações.

Na fase de Elaboração – deve ser feita uma reavaliação das Classes identificadas, com base nas referências do Diagrama de Linha de Montagem desenvolvido nesta fase. Através da análise das referências deve-se identificar que classes serão responsáveis pela realização dos casos de uso identificados no Diagrama de Linha de Montagem.

## **INCORPORAÇÃO DAS ATIVIDADES EM UMA METODOLOGIA**

Cada empresa que desenvolve *software* tem suas particularidades e busca desenvolver suas próprias metodologias ou adaptar alguma existente no mercado. Neste trabalho utilizouse para teste uma metodologia de desenvolvimento de *software* de uma empresa que é baseada no UP e guia o desenvolvimento dos sistemas concebidos no paradigma da orientação a objeto.

Por ser iterativa, cada fase percorre todo o conjunto de *workflows*. Por ser incremental, cada iteração atualiza os artefatos gerados nas iterações anteriores. Cada Artefato corresponde a uma documentação (como um modelo) ou outro objeto de valor a ser criado no desenvolvimento (como um componente de *software*). A metodologia da empresa também estabelece os estados esperados dos artefatos ao final de cada fase. Porém ela não apresenta, como o UP, atividades para o adequado levantamento dos requisitos dos negócios.

Nesta seção são descritos os fluxogramas de atividades originados da incorporação das atividades propostas na metodologia (principalmente as fases de Concepção e Elaboração). Nas figuras dos fluxogramas as atividades adicionadas à metodologia apresentam-se em cor cinza escuro e as atividades já constantes, mas que foram atualizadas, apresentam-se com listras. Também são apresentadas as descrições de cada atividade e os estados esperados para cada artefato ao final de cada fase. Esta metodologia resultante foi utilizada no contexto do desenvolvimento de um sistema de Controle de Expedições da empresa, sendo apresentados exemplos de modelos gerados neste teste.

### **Descrição do Fluxograma da Fase Concepção**

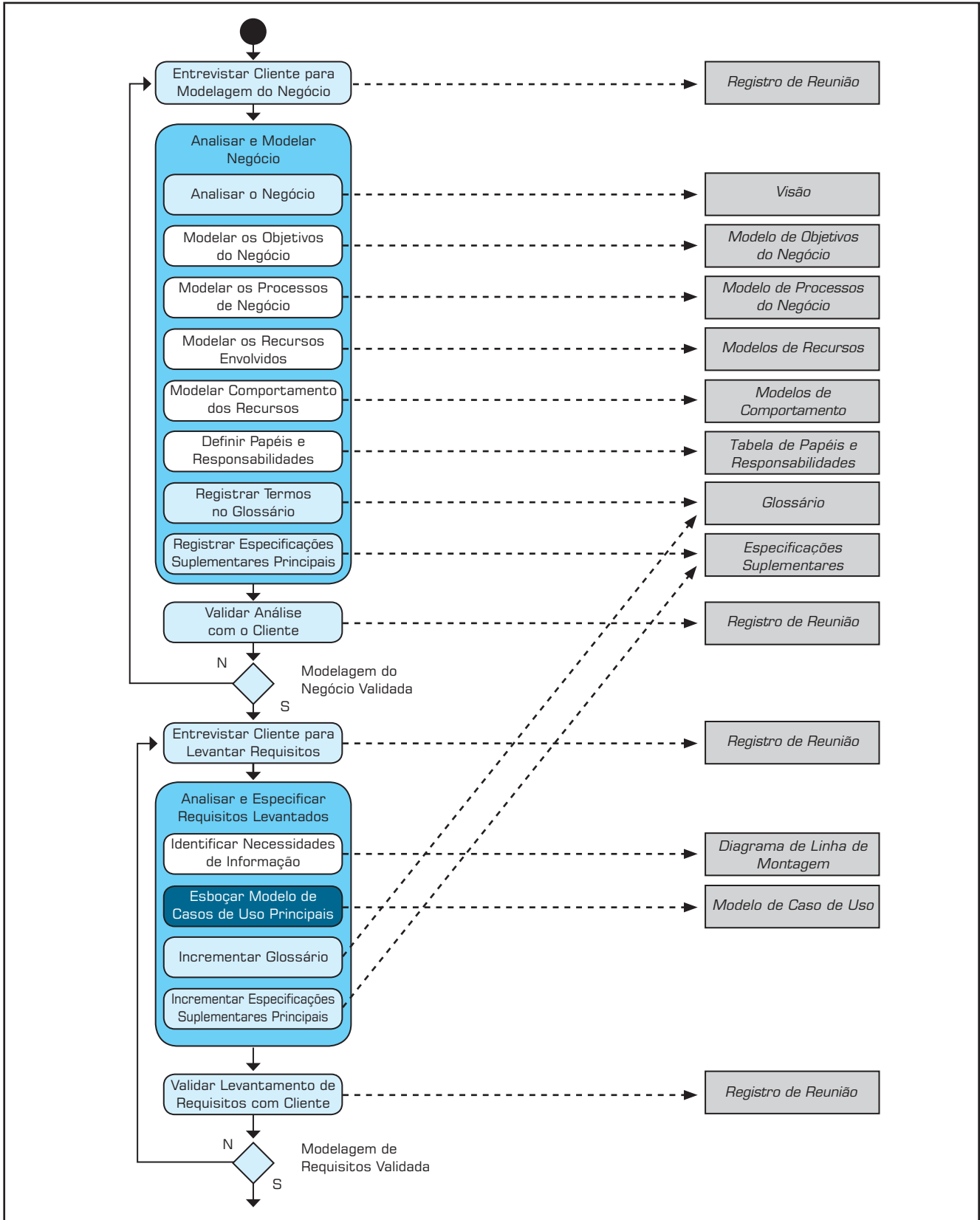
Na Figura 3 é apresentada a parte inicial do fluxograma de atividades resultante para a fase Concepção. A Figura 4 apresenta um modelo gerado nesta fase, na atividade de Modelagem de Processos de Negócios. O Quadro 1 apresenta o estado esperado dos artefatos ao final da fase de Concepção. Descrevem-se a seguir as atividades do fluxograma desta fase.

A.1) Entrevistar Cliente para Modelagem do Negócio: entendimento do negócio onde o futuro Sistema atuará, com registro das informações levantadas num *Registro de Reunião*, contendo um esboço do *Diagrama de Contexto* e do *Modelo de Processos de Negócio*;

A.2) Analisar e Modelar o Negócio

- Analisar o Negócio: Documentar a *Descrição do Negócio* com base nas informações levantadas nas reuniões;
- Modelar os Objetivos do Negócio, Modelar os Processos de Negócio (ver Figura 4), Modelar os Recursos Envolvidos, Modelar Comportamento dos Recursos e Definir Papéis e Responsabilidades são atividades que devem ser feitas conforme descritas na seção “Workflow para Modelagem de Negócio” com a abordagem para a fase de Concepção (seção “Abordagens das atividades propostas para Modelagem de Negócios”);
- Registrar Termos no Glossário: para expressar os conceitos presentes no negócio, documentando no *Glossário*;
- Registrar Especificações Suplementares Principais: podem ser previstos alguns requisitos não funcionais como relativos à segurança e à performance por exemplo, registrando em *Especificações Suplementares*;

Figura 3: Parte inicial do fluxograma referente à fase de Concepção.



A.3) Validar Análise com o Cliente: realizar reunião com o Cliente para apresentar e validar a *Análise do Negócio*, *Diagrama de Contexto*, *Modelo de Processos de Negócio*, *Glossário* e *Especificações Suplementares*;

A.4) Entrevistar Cliente para Levantar Requisitos: levantamento das principais funcionalidades do futuro Sistema visando a identificação dos principais casos de uso e requerimentos não funcionais, registrando as informações num *Registro de Reunião*;

A.5) Analisar e Especificar Requisitos Levantados

- Identificar Necessidades de Informatização: conforme seção “Workflow de Levantamento de Requisitos” e “Abordagens das atividades propostas para Levantamento de Requisitos” (abordagem para a fase de concepção);
- Esboçar Modelo de Casos de Uso Principais: conforme seção “Workflow de Levantamento de Requisitos” e “Abordagens das atividades propostas para Levantamento de Requisitos” (abordagem para a fase de concepção);
- Incrementar Glossário: atualizar o *Glossário* com os novos termos definidos nas reuniões de Levantamento de Requisitos;
- Incrementar Especificações Suplementares Principais: Atualizar as *Especificações Suplementares* com os novos requisitos não funcionais identificados nas reuniões de Levantamento de Requisitos;

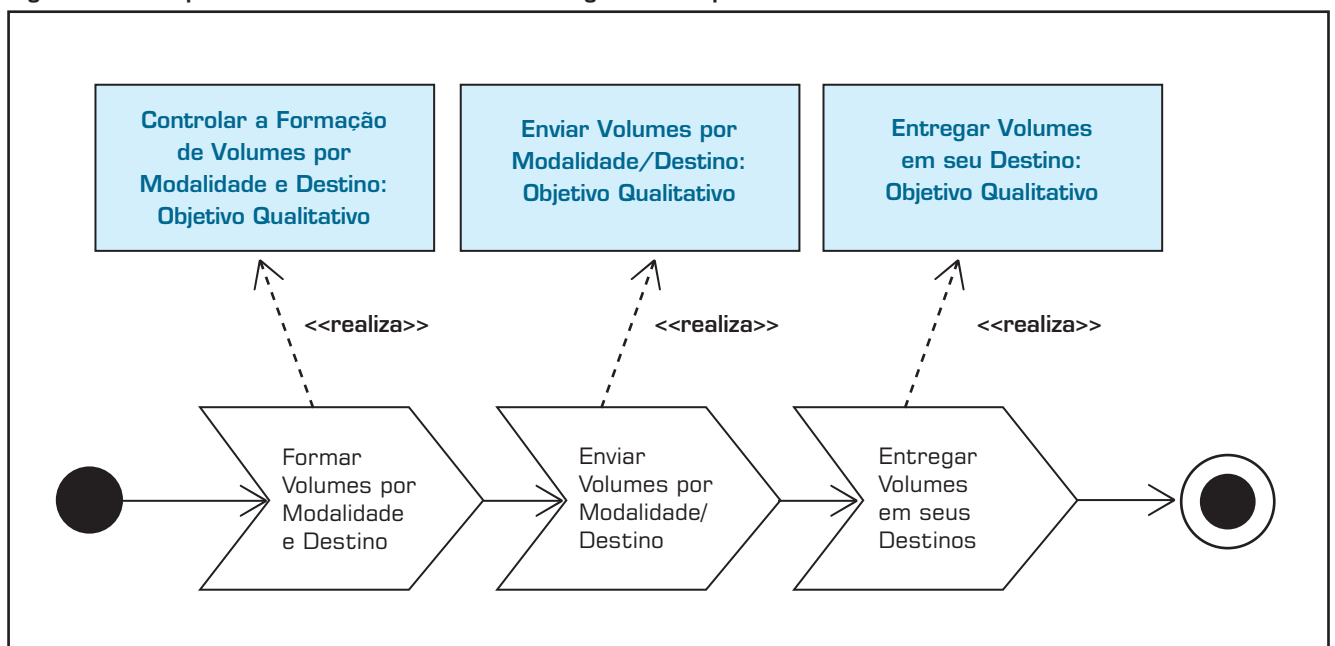
A.6) Validar Levantamento de Requisitos com Cliente: reunião para apresentar e validar o *Modelo de Caso de Uso* e as alterações no *Modelo de Processo de Negócio*, no *Glossário* e nas *Especificações Suplementares*;

A.7) Entrevistar Cliente para Definir Arquitetura: levantar informações e esboçar alternativas de Modelos de Classe e Pacotes, Modelo de Componentes e de Implantação em alto nível (procurando definir a arquitetura de hardware e *software* em linhas gerais).

A.8) Definir Arquitetura do *Software*

- Desenvolver Modelo de Pacotes: analisar e definir um *Modelo de Pacotes* em alto nível buscando identificar as relações entre eles e a possível reutilização de arquiteturas de referência ou padrões;
- Desenvolver Modelo de Classes: com as principais classes do sistema e seus relacionamentos, verificando a possibilidade de reutilização de Classes já existentes, utilizando a proposta da seção “Workflow para análise”, abordagem para a fase de concepção (“Abordagem da atividade proposta para análise”);
- Desenvolver Modelos de Componentes: uma visão dos subsistemas de informação e seus relacionamentos;
- Desenvolver Modelos de Implantação: uma visão da arquitetura de hardware do novo sistema;
- Incrementar Glossário: atualizar o *Glossário* com os novos termos definidos na definição da Arquitetura do *Software*;
- Incrementar Especificações Suplementares: atualizar as *Especificações Suplementares* com os novos termos definidos na definição da Arquitetura do *Software*;
- Incrementar Modelo de Casos de Uso Principais: atualizar o *Modelo de Casos de Uso* com os novos termos definidos na definição da Arquitetura do *Software*;

Figura 4: Exemplo de Modelo de Processo de Negócio — Expedição.



A.9) Validar Arquitetura com Cliente: validar a arquitetura geral do sistema com o cliente.

A.10) Elaborar Cronograma Geral: para Desenvolvimento do Projeto com base nos requisitos levantados;

A.11) Elaborar Orçamento: com base no *Cronograma Geral* e requisitos levantados;

A.12) Formalizar Prestação de Serviço: consiste na elaboração e assinatura de um contrato junto ao cliente.

**Descrição do Fluxograma da Fase de Elaboração**

O fluxograma de atividades resultante para a fase Elaboração é apresentado na Figura 5. A Figura 6 apresenta um modelo gerado nesta fase, na atividade Identificar Necessidades de Informatização, sendo que o Quadro 2 apresenta o estado esperado dos artefatos ao final desta fase.

B.1) Definir Iterações da Elaboração: definir subdomínios para a fase de Elaboração com base na *Arquitetura do Software* e nas *Especificações Suplementares* definidas em cada iteração (deve conter o detalhamento do cronograma para a fase de Elaboração, programando o desenvolvimento dos Casos de Uso por iterações);

B.2) Entrevistar Cliente para Levantar Requisitos: detalhamento dos *Casos de Uso* já identificados bem como a

identificação e detalhamento de outros Casos de Uso (registrar o levantamento num *Registro de Reunião*);

B.3) Analisar e Modelar o Negócio: Modelar os Objetivos do Negócio; Modelar os Processos de Negócio; Modelar os Recursos Envolvidos; Modelar Comportamento dos Recursos; e Definir Papéis e Responsabilidades, conforme descritas na seção “*Workflow* para Modelagem de Negócio” com a abordagem para a fase de Elaboração (“Abordagens das atividades propostas para Modelagem de Negócios”);

B.4) Analisar e Especificar Requisitos Levantados: consiste na realização em colaboração das atividades: Identificar Necessidades de Informatização (ver Figura 6); Incrementar Modelo de Casos de Uso (conforme seção “*Workflow* de Levantamento de Requisitos” com abordagem para a fase de Elaboração, seção “Abordagens das atividades propostas para Levantamento de Requisitos”); Incrementar (atualizar) Glossário; e Incrementar (atualizar) Especificações Suplementares;

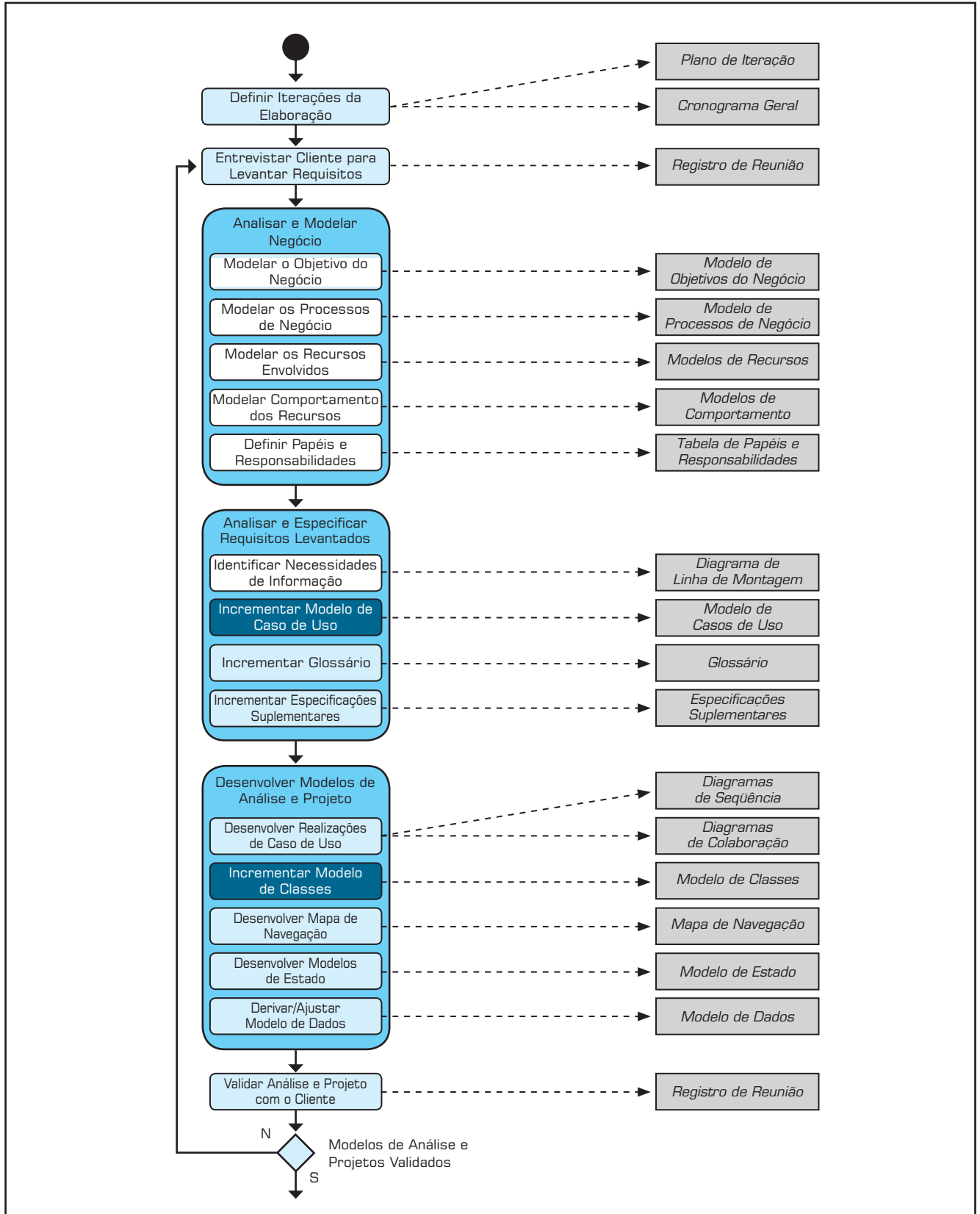
- B.5) Desenvolver Modelos de Análise e Projeto
- Desenvolver Realizações de Caso de Uso: para cada Caso de Uso, mostrar as interações realizadas pelos objetos (Classes) necessárias à realização do caso de uso;
  - Incrementar Modelo de Classes: através da análise das

**Quadro 1: Estados dos Artefatos ao final da Concepção.**

| ARTEFATO                             | ESTADO ESPERADO AO FINAL DA CONCEPÇÃO   |
|--------------------------------------|---|
| Cronograma Geral                     | Definido.   |
| Orçamento                            | Definido.   |
| Formalização da Prestação de Serviço | Realizada.  |
| Registro de Reunião                  | Criado para cada reunião acontecida durante a fase.   |
| Descrição do Negócio                 | Cerca de 90% definida (é revisado na fase seguinte).  |
| Modelo de Objetivos do Negócio       | Principais objetivos identificados.   |
| Modelo de Processos de Negócio       | Principais processos de negócio identificados.  |
| Modelos de Recursos                  | Principais recursos identificados e modelados.  |
| Modelos de Comportamento             | Criado para recursos importantes e complexos.   |
| Tabela de Papéis e Responsabilidades | Com responsáveis pelos processos identificados.   |
| Diagrama de Linha de Montagem        | Criado para os processos macros a serem informatizados.   |
| Glossário                            | Termos definidos e documentados.  |
| Especificações Suplementares         | Principais requerimentos não funcionais documentados.   |
| Modelo de Casos de Uso               | Esboçado. Atores e Casos de Uso importantes definidos; Fluxo de eventos definidos em linhas gerais para os Casos de Uso.  |
| Arquitetura do <i>Software</i>       | <i>Modelo de Casos de Uso, Modelo de Pacotes, Modelo de Classes, Modelo de Componentes, Diagramas de Seqüências e de Colaboração, e Modelo de Implantação</i> descritos de uma maneira macro (principais aspectos). |



Figura 5: Fluxograma referente à fase de Elaboração.



Realizações de Caso de Uso e Modelo de Classes, deve-se identificar a necessidade de novas classes e incrementá-las no *Modelo de Classes* (os *Diagramas de Linha de Montagem* podem ser usados como base para a identificação de novas classes, conforme seção “*Workflow para Análise*”, abordagem para a fase de *Elaboração*, seção “*Abordagem da atividade proposta para Análise*”);

- Desenvolver Mapa de Navegação: identificar no Modelo de Classes as Classes de Interface, e dentre estas as que serão páginas Web, desenvolvendo o *Mapa de Navegação*;
- Desenvolver Modelos de Estado: para analisar e explicitar a mudança de estados de objetos ao longo da execução dos processos ou eventos, desenvolvendo *Modelos de Estado*;
- Derivar / Ajustar Modelo de Dados: com base no *Modelo*

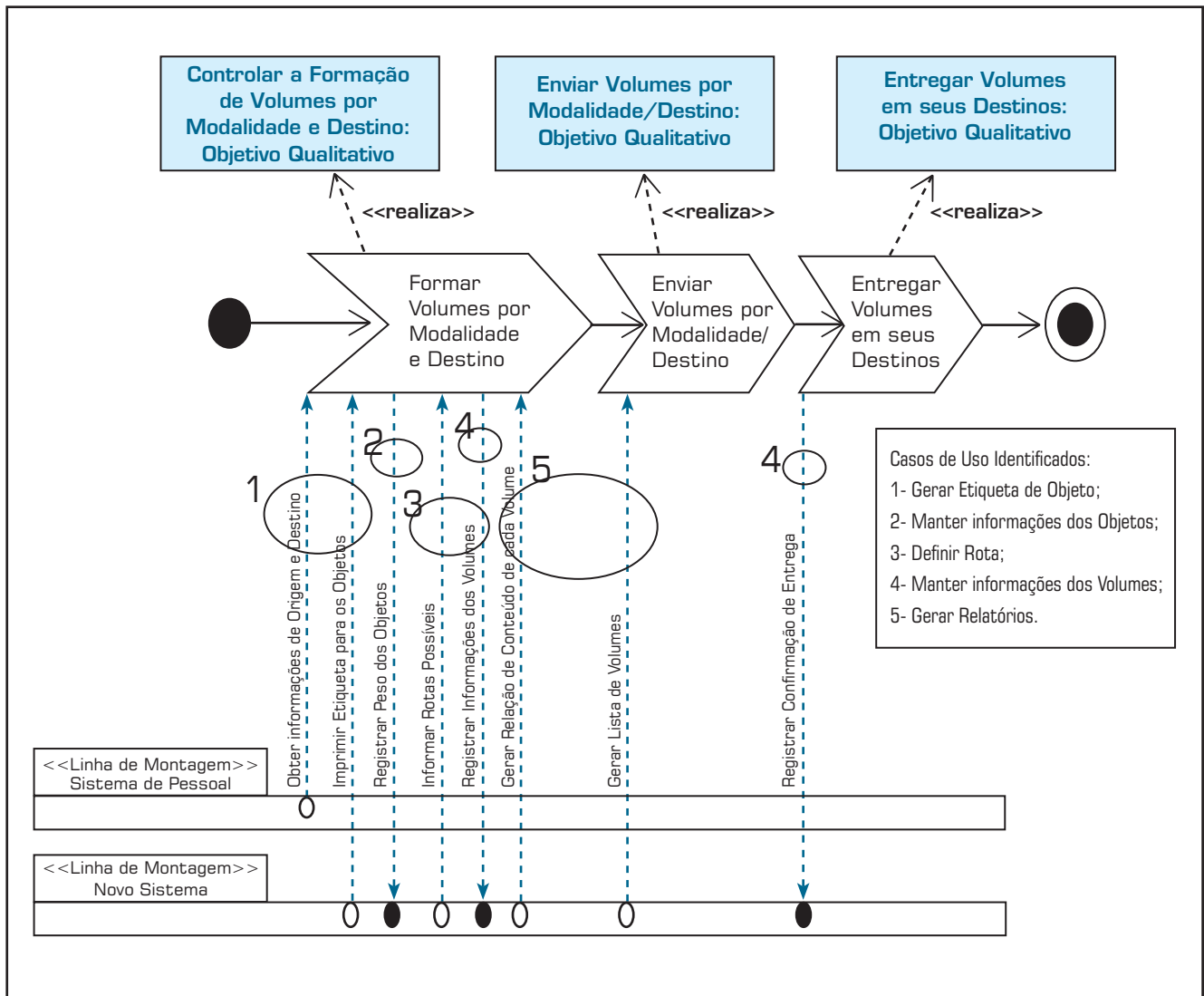
*de Classes* deve-se desenvolver o *Modelo de Dados*, fazendo a correspondência das classes para o modelo relacional;

B.6) Validar Análise e Projeto com o Cliente: entrevista com Cliente para validar os modelos de Análise e Projeto (gerar um *Registro de Reunião*);

B.7) Atualizar Arquitetura do Negócio: atualizar todos os modelos da Arquitetura do Negócio com as novas informações da Análise e Projeto.

As atividades propostas neste trabalho abrangem principalmente as fases de *Concepção* e *Elaboração*, pois é durante estas que os casos de uso são normalmente identificados. Portanto, as fases de *Construção* e *Transição* não tiveram suas atividades atualizadas ou modificadas. Para uma compreensão mais completa da metodologia resultante, nas

Figura 6: Exemplo de modelo de Linha de Montagem: Processo de Expedição.



próximas seções são apresentados os fluxogramas relativos às fases de Construção e Transição.

### Descrição do Fluxograma da Fase de Construção

As atividades da fase de Construção são apresentadas na Figura 7. O Quadro 3 apresenta o estado esperado dos artefatos ao final da fase Construção.

C.1) Definir Iterações da Construção: planejar a implementação de cada componente nas iterações respeitando-se o prazo estabelecido no *Cronograma Geral* (deve-se identificar a prioridade de implementação dos componentes);

C.2) Elaborar Plano de Testes: deve conter a descrição de todos os testes que serão realizados durante o projeto, bem como o momento em que acontecerão;

C.3) Implementar: consiste das atividades Implementar Componentes (Codificação dos componentes, gerando os *Componentes Implementados*) e Implementar Banco de Dados (Geração do *Banco de Dados* com base no *Modelo de Dados*);

C.4) Testar Componentes: realizar testes dos componentes implementados com base no *Plano de Testes*;

C.5) Incrementar Modelos de Requisitos e de Análise e Projeto: Atualizar *Glossário* com novos termos definidos; Atualizar *Especificações Suplementares* com novos requerimentos não funcionais identificados na Construção; Atualizar *as Realizações de Caso de Uso* devido a possíveis necessidades de mudanças de projeto identificadas durante a implementação; Atualizar o *Modelo de Classes*; Atualizar *Mapa de Navegação*; Atualizar *Modelo de Estados* com possíveis mudanças ocorridas na Implementação; Atualizar *Modelo de Dados* com possíveis mudanças ocorridas na construção do Banco;

C.6) Integrar *Componentes Implementados* de acordo com o *Plano de Integração* definido;

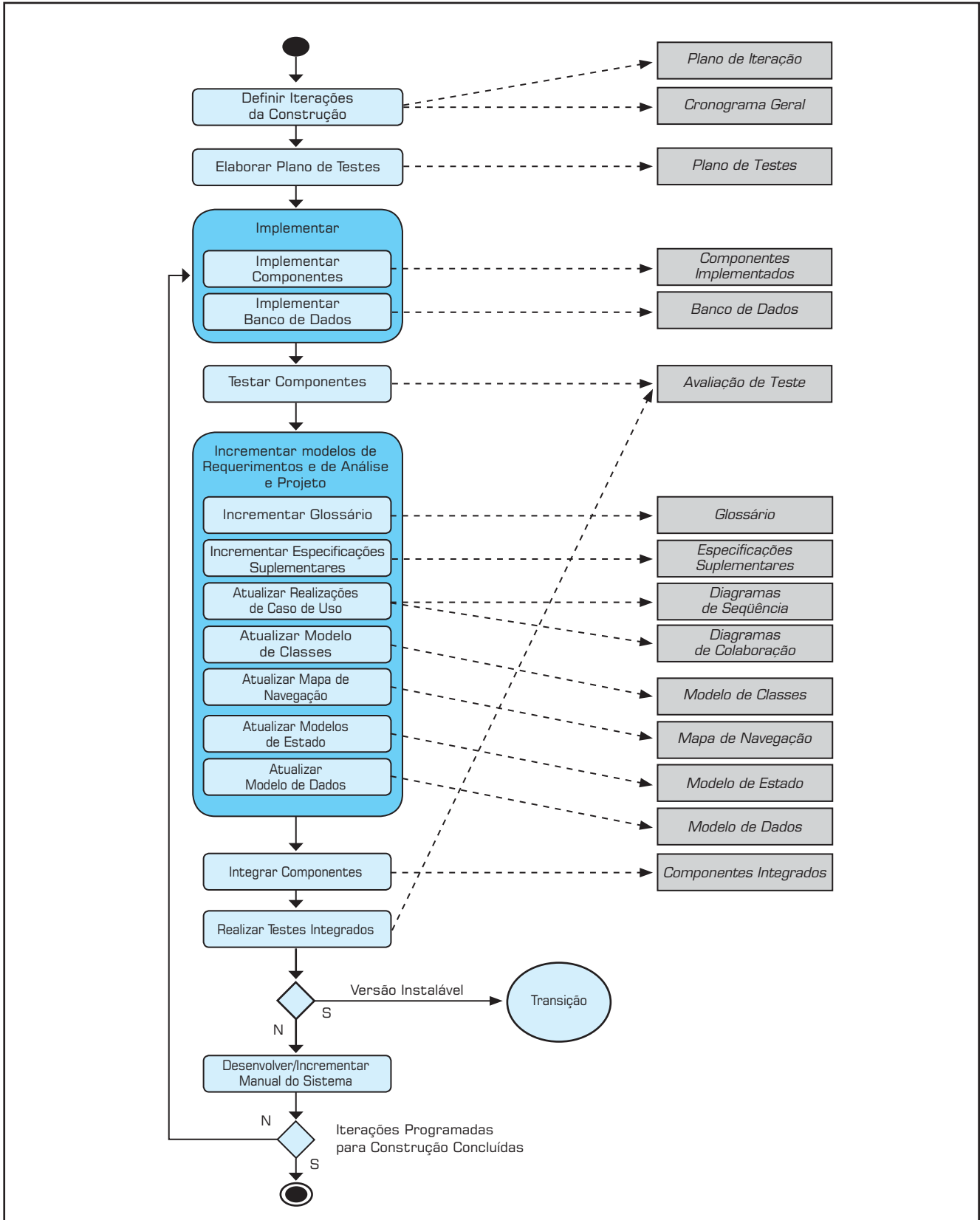
C.7) Realizar Testes Integrados com os componentes integrados, verificando as interações entre eles;

C.8) Desenvolver/Incrementar Manual do Sistema para os usuários com explicações operacionais para realização das funcionalidades requeridas para o Sistema.

**Quadro 2: Estado dos Artefatos ao final da fase de Elaboração.**

| ARTEFATO                             | ESTADO ESPERADO AO FINAL DA ELABORAÇÃO                                       |
|--------------------------------------|--|
| Plano de Iteração                    | Subdomínios do sistema e seus respectivos <i>Casos de Uso</i> identificados. |
| Cronograma Geral                     | Detalhado para a fase de Elaboração desde o início da fase.                  |
| Registro de Reunião                  | Criado para cada reunião.  |
| Modelo de Casos de Uso               | Definido.  |
| Modelo de Objetivos                  | Definido.  |
| Modelo de Processos de Negócio       | Revisado.  |
| Modelos de Recursos                  | Definido.  |
| Modelos de Comport. de Recursos      | Definido.  |
| Tabela de Papéis e Responsabilidades | Definida.  |
| Diagrama de Linha de Montagem        | Criado para os processos detalhados a serem informatizados.                  |
| Glossário                            | Atualizado com novos termos.   |
| Especificações Suplementares         | Atualizado, capturando todos os requerimentos não funcionais.                |
| Diagramas de Seqüência               | <i>Diagramas de Seqüência</i> descrevendo as realizações de Caso de Uso.     |
| Diagramas de Colaboração             | <i>Diagramas de Colaboração</i> descrevendo as realizações de Caso de Uso.   |
| Modelo de Classes                    | Definido.  |
| Mapa de Navegação                    | Definido.  |
| Modelo de Estado                     | Desenvolvido para os principais objetos do sistema.                          |
| Modelo de Dados                      | Definido no nível lógico.  |
| Arquitetura do Negócio               | Revisada, com um maior número de detalhes.                                   |

Figura 7: Fluxograma referente a fase de Construção.



### Descrição do Fluxograma da Fase de Transição

As atividades da fase Transição são apresentadas no fluxograma da Figura 8 e o Quadro 4 apresenta o estado esperado dos artefatos ao final da fase Transição.

D1) Documentar Notas de Versão: descrição das funcionalidades implementadas (quais os casos de uso que a versão realiza), da plataforma em que foi testada, de como realizar sua instalação, *Bugs*, limitações e soluções conhecidas nos testes;

D2) Desenvolver Estratégia de Implantação: planejamento para a implantação do sistema construído;

D3) Implantar

- Instalação do Ambiente de *Hardware* e *Software*: preparação do ambiente de *hardware* e *software* do cliente para receber o novo sistema (preparação da infra-estrutura tecnológica do cliente) ;
- Treinar Usuários: atividades de planejamento e execução de treinamento dos usuários no novo sistema;
- Implantar Sistema: preparação do ambiente organizacional para implantação do novo sistema;

D4) Realizar Testes no Ambiente de Produção: conforme *Plano de Testes*;

D5) Avaliar Manutenção: planejar a manutenção, podendo ser necessário voltar a uma das quatro fases do desenvolvimento: Concepção, Elaboração, Construção ou Transição.

### CONSIDERAÇÕES FINAIS

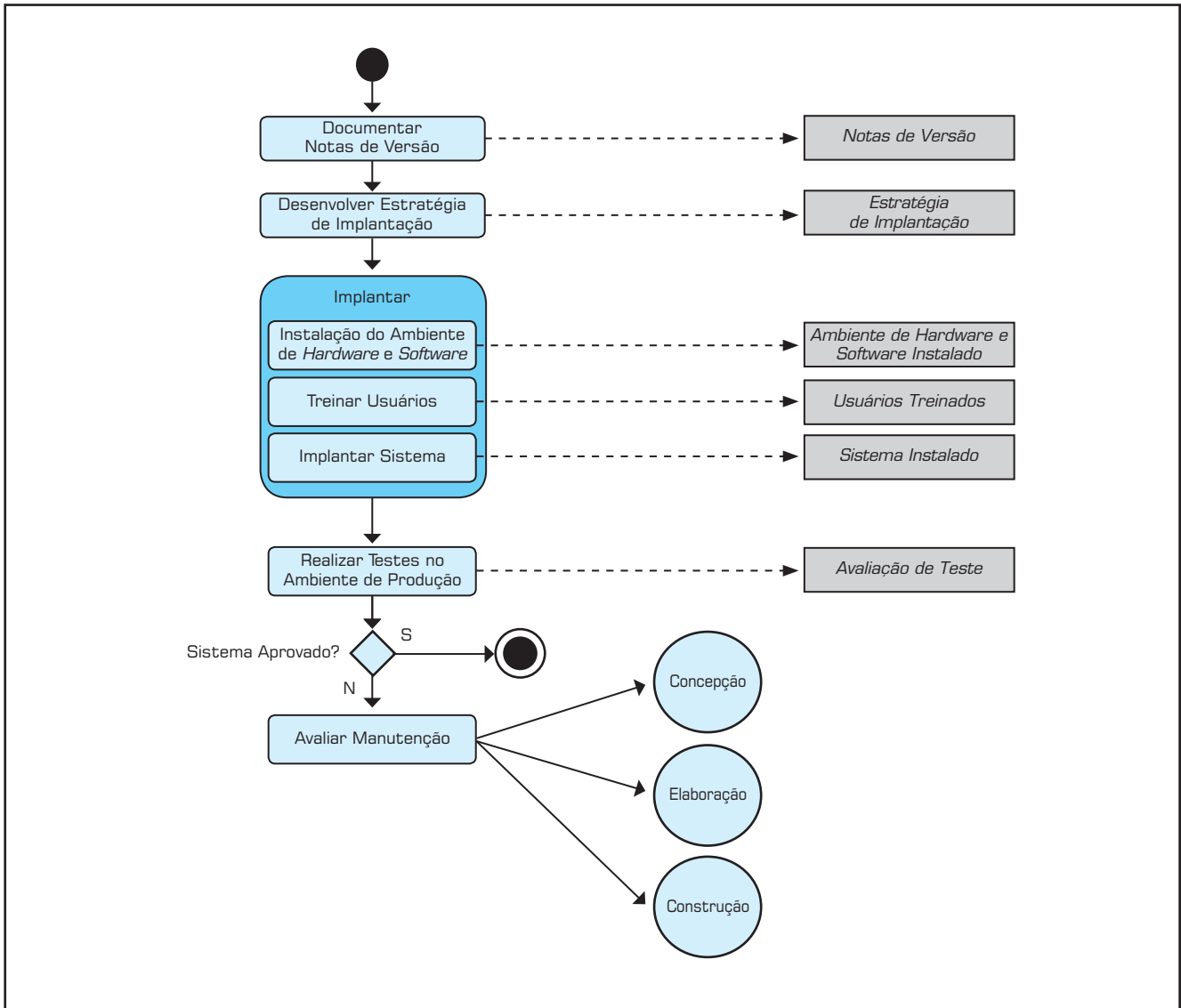
Neste artigo enfatizou-se a necessidade de o desenvolvimento de sistemas complexos de software ser guiado por uma metodologia ou um processo de desenvolvimento que organize e controle a produção das várias partes (artefatos) constituintes de um sistema. O UP contempla boas práticas de engenharia de software, mas não define adequadamente atividades para a modelagem de negócio. O RUP oferece uma proposta de modelagem de negócio, porém, apresenta limitações quanto à modelagem e quanto ao alinhamento dos casos de uso identificados aos reais objetivos do negócio. A técnica de construção de arquiteturas de negócio proposta por Eriksson e Penker (2000) é, dentre as propostas de modelagem de negócio com UML pesquisadas neste trabalho, a única que aborda de forma sistemática a passagem da arquitetura de negócio para uma arquitetura de software.

**Quadro 3: Estado dos Artefatos ao final da fase de Construção.**

| ARTEFATO                     | ESTADO ESPERADO AO FINAL DA CONSTRUÇÃO   |
|------------------------------|--|
| Plano de Iteração            | Estimativa para a implementação de cada componente identificado no Modelo de Componentes, por iteração.                  |
| Cronograma Geral             | Detalhado para a fase de Construção desde o início da fase.  |
| Plano de Testes              | Elaborado, contendo a descrição dos testes que serão aplicados nos componentes, bem como o cronograma de sua realização. |
| Componentes Implementados    | Componentes implementados e testados. Com a maioria dos componentes já integrados.                                       |
| Banco de Dados               | Construído.  |
| Avaliação de Teste           | Realizadas para cada componente e para integração de conjuntos deles.  |
| Glossário                    | Atualizado e completamente revisado.   |
| Especificações Suplementares | Atualizado e completamente revisado.   |
| Diagramas de Seqüência       | Revisados.   |
| Diagramas de Colaboração     | Revisados.   |
| Modelo de Classes            | Revisado.  |
| Mapa de Navegação            | Revisado.  |
| Modelos de Estado            | Revisados.   |
| Modelo de Dados              | Revisado.  |
| Componentes Integrados       | Cerca de 90% integrados.   |



Figura 8: Fluxograma referente à fase de Transição.



Quadro 4: Estado dos Artefatos ao final da Transição.

| ARTEFATO                               | ESTADO ESPERADO AO FINAL DA TRANSIÇÃO   |
|--|---|
| Notas de Versão                        | Notas de Versão geradas para cada versão entregue ao usuário.                           |
| Estratégia de Implantação              | Definida no início da fase.   |
| Ambiente de <i>Hardware e Software</i> | Infra-estrutura do cliente completamente instalada e operando.                          |
| Usuários Treinados                     | Usuários treinados para o sistema.  |
| Sistema Instalado                      | Sistema Instalado e operando em ambiente de produção.                                   |
| Avaliação de Teste                     | Realizada para cada componente construído e para cada módulo de componentes integrados. |

Os autores, porém, não exploram a sistematização desta passagem no contexto de um processo ou metodologia de desenvolvimento de sistemas.

O objetivo deste artigo foi descrever toda uma metodologia de desenvolvimento de software com atividades para a modelagem de negócio trazendo vantagens como: (i) identificação sistemática de necessidades de informatização a partir do fluxo de evento dos processos estabelecido na atividade, conforme os objetivos do negócio; (ii) identificação sistemática dos casos de uso numa abordagem iterativa estabelecida na atividade Derivar Casos de Uso dos Processos de Negócio; (iii) e também incorporação de atividades de forma consistente com o modelo incremental, com interfaces bem estabelecidas com as atividades preestabelecidas do UP.

A metodologia resultante apresentada foi aplicada a um projeto piloto que correspondeu ao desenvolvimento de um sistema para controle de expedição de uma empresa. Percebeu-se por desenvolvedores, que usavam a metodologia da empresa e que também participaram do projeto piloto, que a metodologia proposta permitiu identificar requisitos e artefatos no projeto do sistema (tais como

objetivos do negócio, atividades, classes de objetos e necessidades de informatização do processo de expedição) que não seriam (ou não seriam facilmente) identificados com a metodologia normalmente usada na empresa. Isto indica que a metodologia resultante apresentou melhorias com relação a uma metodologia original, puramente baseada no UP.

## A metodologia resultante apresentou melhorias com relação a uma metodologia original, puramente baseada no UP.

Atualmente, parte dos conceitos apresentados neste trabalho está sendo incorporada a uma metodologia de desenvolvimento de ERPs (Enterprise Resources Planning) de código aberto (SILVA *et al.*, 2006), dentro do projeto ERP5 ([www.erp5.org](http://www.erp5.org)). Como proposta para trabalhos futuros sugere-se a comparação desta técnica com demais técnicas de identificação de requisitos, e a construção de uma ferramenta CASE que permita maior automação das atividades definidas neste trabalho.

Artigo recebido em 12/07/2005

Aprovado para publicação em 27/09/2007

### Referências

- AZEVEDO JR., D. P. *Aplicação da técnica de Modelagem de Negócio com UML a processos iterativos de desenvolvimento de software*. 2003. 127 f. Dissertação (Mestrado – Ciências de Engenharia, área Engenharia de Produção) – Universidade Estadual do Norte Fluminense Darcy Ribeiro, Centro de Ciências e Tecnologia, Campos dos Goytacazes, 2003.
- AZEVEDO JR., D. P.; CAMPOS, R. *Aplicação de uma Arquitetura de Modelagem de Processos de Negócios no Desenvolvimento de Software*. *Vértices*, v. 6, n. 3, p. 147-175, 2004.
- ERIKSSON, H. E.; PENKER, M. *Business modeling with UML: business patterns at work*. New York: John Wiley, 2000.
- FELICIANO NETO, A. *Sistemas flexíveis de informações*. São Paulo: Makron, 1996.
- JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. *The unified software development process*. Reading: Addison-Wesley, 1999.
- JOHANSSON, H. J. *et al. Processos de negócios*. São Paulo: Pioneira, 1995.
- KALPIC, B.; BERNUS, P. Business process modelling in industry: the powerful tool in enterprise management. *Computers in Industry*, v. 47, n. 3, p. 299-318, 2002.
- KIRIKOVA, M. Explanatory capability of enterprise models. *Data and Knowledge Engineering*, v. 33, n. 2, p. 119-136, May 2000.
- KOSANKE, K.; VERNADAT, F.; ZELM, M. CIMOSA: enterprise engineering and integration. *Computers in Industry*, v. 40, n. 2, p. 83-97, Nov. 1999.
- KOSANKE, K.; NELL, J. G. Standardisation in ISO for enterprise engineering and integration. *Computers in Industry*, v. 40, n. 2, p. 311-319, Nov. 1999.
- KRUCHTEN, P. *Introdução ao RUP: Rational Unified Process*. Rio de Janeiro: Ciência Moderna, 2003.
- LI, H.; WILLIAMS, T. J. Management of complexity in enterprise integration projects by the PERA methodology. *Journal of Intelligent Manufacturing*, v. 13, n. 6, p. 417-427, Dec. 2002.
- LILLY, S. Use case pitfalls: top 10 problems from real projects using use cases. In: TECHNOLOGY OF OBJECT-ORIENTED LANGUAGES AND SYSTEMS - TOOLS, 30., 1999, Santa Barbara. *Proceedings...* Santa Barbara: IEEE, p. 174-183, 1999.
- MARSHALL, C. *Enterprise modeling with UML: designing successful software through business analysis*. Reading: Addison-Wesley, 1999.
- ODEH, M.; KAMM, R. Bridging the gap between business models and system models. *Information and Software Technology*, v. 45, n. 15, p. 1053-1060, 2003.
- OMG – Object Management Group. *UML extension for business modeling*. v. 1.1, 1997. Disponível em: <<ftp://ftp.omg.org/pub/docs/ad/97-08-07.pdf>>. Acesso em: 12 nov. 2007.

■ **Referências**

- PAULA FILHO, W. P. *Engenharia de software: fundamentos, métodos e padrões*. Rio de Janeiro: Livros Técnicos e Científicos, 2001.
- DE ENGENHARIA DE SOFTWARE, 16. 2002, Gramado. *Anais...* Disponível em: <<http://www.lbd.dcc.ufmg.br:8080/colecoes/sbes/2002/013.pdf>>. Acesso em: 10 nov. 2003.
- SHEN, H. *et al.* Integration of business modelling methods for enterprise information system analysis and user requirements gathering. *Computers in Industry*, v. 54, n. 3, p. 307-323, Aug. 2004.
- VERNADAT, F. B. *Enterprise modeling and integration: principles and application*. London: Chapman & Hall, 1996.
- PRESSMAN, R. S. *Engenharia de software*. 5. ed. Rio de Janeiro: McGraw-Hill, 2002.
- SCHEER, A. *ARIS – Business process modeling*. 3. ed. New York: Springer, 2000.
- SILVA, C. M. F. et al. GERAM como arquitetura de referência para um ERP livre de código aberto. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 26. 2006, Fortaleza. *Anais eletrônicos...* Rio de Janeiro: ABEPRO, 2006. 1 CD-ROM.
- VERNADAT, F. B. Enterprise modeling and integration (EMI): current status and research perspectives. *Annual Reviews in Control*, v. 26, n. 1, p. 15-25, 2002.
- SANTANDER, V. F. A.; CASTRO, J. F. B. Integrating use cases and organizational modeling. In: SIMPÓSIO BRASILEIRO
- SCHNEIDER, G.; WINTERS, J. P. *Applying use cases: a practical guide*. Boston: Addison-Wesley, 1998.

■ **Sobre os autores**

**Delmir Peixoto de Azevedo Junior**

Universidade Petrobras  
 Analista de Sistema  
 E-mail: delmirjunior@yahoo.com.br

**Renato de Campos**

Universidade Estadual Paulista – UNESP  
 Professor  
 End.: Rua Alberto Segalla, 1-117, ap. 91A – Bairro Infante Dom Henrique – Bauru – SP – CEP 17012-634  
 Tel./Fax: (14) 3103-6122  
 E-mail: rcampos@feb.unesp.br