



**Universidade Federal de Pernambuco  
Centro de Informática  
Pós-Graduação em Ciência da Computação**

# **Uma Extensão do Rational Unified Process baseada na Criação de Valor**

por

**Gustavo Tibério d'Anuniação**

Dissertação de Mestrado

Orientador: Dr. Hermano Perrelli de Moura



Universidade Federal de  
Pernambuco  
[posgraduacao@cin.ufpe.br](mailto:posgraduacao@cin.ufpe.br)  
[www.cin.ufpe.br/~posgraduacao](http://www.cin.ufpe.br/~posgraduacao)

Recife, junho/2009





Universidade Federal de Pernambuco  
**CENTRO DE INFORMÁTICA**  
**PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**GUSTAVO TIBÉRIO D'ANUNCIAÇÃO**

# **Uma Extensão do Rational Unified Process baseada na Criação de Valor**

***ESTE TRABALHO FOI APRESENTADO À PÓS-GRADUAÇÃO EM CIÊNCIA  
DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA UNIVERSIDADE  
FEDERAL DE PERNAMBUCO COMO REQUISITO PARCIAL PARA  
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA DA COMPUTAÇÃO.***

***ORIENTADOR: PROF. DR. HERMANO PERRELLI DE MOURA***

RECIFE, JUNHO/2009



Àqueles que sempre me ensinam muito sobre o que há de maior valor na vida:  
Carmen, Clovis e Henrique – minha esposa, meu pai e meu filho.

À minha mãe (*in memoriam*).

## Agradecimentos

A Deus, por ter me permitido iniciar e concluir esta jornada, tendo durante o período me dado o melhor presente de todos: meu filho Henrique.

À minha família, pelo incentivo e injeções de ânimo nos momentos difíceis.

À minha esposa, Carmen, especialmente pelo apoio, e pela paciência durante o período do Mestrado.

Aos meus colegas do Tribunal de Contas de Pernambuco, em especial a Teresa Moura, Saulo Malinconico e Ana Carolina Machado, pelo apoio e compreensão, especialmente quanto às atribuições de tarefas e flexibilização dos horários de trabalho durante o Mestrado.

Aos profissionais que participaram da avaliação desta dissertação, pela colaboração, sugestões e críticas sempre construtivas.

A Luciana Leal e Marcelo Renê, colegas de Mestrado que me deram dicas e sugestões essenciais quanto à condução da avaliação e à escrita da dissertação.

Ao meu orientador, Hermano Perrelli, por todo o trabalho de orientação, em especial pelas críticas e sugestões sempre muito importantes, e pela atenção nas revisões do texto.

## Resumo

Uma das tendências da pesquisa em gerenciamento de projetos para os próximos anos é o estudo da ênfase na criação de valor pelos projetos. Este tema envolve uma maior preocupação com o alinhamento dos projetos aos objetivos estratégicos das organizações. No contexto dos projetos de desenvolvimento de software, tal preocupação passa pela revisão das metodologias de desenvolvimento de software que os suportam. A Engenharia de Software Baseada em Valor (VBSE) é uma abordagem que adiciona considerações de valor aos aspectos técnicos da Engenharia de Software, estabelecendo um conjunto de elementos-chave necessários à evolução das técnicas e práticas atuais para que enfatizem a criação de valor.

Este trabalho faz uma análise do Rational Unified Process, um *framework* de processos de software, quanto à ênfase em criação de valor que ele possibilita, usando os elementos-chave da VBSE como diretriz para a análise. Pontos de convergência entre o RUP e os elementos-chave são identificados, e uma extensão do processo é proposta para suprir as lacunas encontradas. Uma avaliação do novo processo é feita e oportunidades para trabalhos futuros apresentadas.

**Palavras-chave:** Engenharia de Software, Economia de Software, Gerenciamento de Projetos, Criação de Valor, Engenharia de Software Baseada em Valor – VBSE, Rational Unified Process – RUP.

## Abstract

Among the trends of research in project management for the next years is the study of the focus in value creation in projects. The field poses a greater concern with the alignment of projects with the strategic goals of the organizations. Such concern, in the context of software development projects, requires reviewing the software development methodologies which support these projects. The Value-Based Software Engineering (VBSE) approach adds value considerations to the technical aspects of Software Engineering, introducing a set of key elements needed to the evolution of current practices and techniques with focus on value creation.

This work analyzes Rational Unified Process, a software process framework, in respect of the focus on value creation it enables, using the VBSE key elements as a guideline to the analysis. RUP's adherence to the key elements is analyzed, and an extension to the process is proposed to fill the gaps found. A validation of the new process is conducted and opportunities of future works are presented.



# Sumário

Agradecimentos .....	iv
Resumo .....	v
Abstract .....	vi
Sumário .....	vii
Lista de Figuras .....	ix
Lista de Tabelas .....	x
Lista de Tabelas .....	x
1 Introdução .....	1
1.1 Contexto e Motivação .....	1
1.2 Objetivos .....	3
1.3 Metodologia .....	5
1.4 Estrutura da Dissertação .....	5
2 Processos, Projetos e Criação de Valor em Engenharia de Software .....	8
2.1 Definição de Valor .....	8
2.2 Processos de Software .....	9
2.2.1 Processos Ágeis e Processos Disciplinados .....	10
2.3 Rational Unified Process (RUP) .....	11
2.3.1 Elementos Essenciais do RUP .....	12
2.3.2 Estrutura Estática – A Descrição do Processo .....	13
2.3.3 Estrutura Dinâmica – O Ciclo de Vida do Processo .....	16
2.4 Gestão de Projetos de Software .....	18
2.5 Estratégia de Projetos de Software e de TI .....	20
2.5.1 Alinhamento Estratégico da Gestão de Projetos .....	21
2.5.2 Gestão de Programas e Portfólios de Projetos .....	22
2.6 Economia de Software .....	25
2.7 Engenharia de Software Baseada em Valor .....	27
2.7.1 Teoria Inicial da VBSE .....	27
2.7.2 Framework de Processo da VBSE .....	29
2.7.3 Elementos-chave .....	31
2.7.4 Análise de Realização de Benefícios .....	32
2.7.5 Identificação e Conciliação de Proposições de Valor dos Interessados .....	33
2.7.6 Análise de Caso de Negócio .....	36
2.7.7 Gerenciamento Contínuo de Riscos e Oportunidades .....	36
2.7.8 Engenharia Simultânea de Sistema e de Software .....	37
2.7.9 Monitoramento e Controle Baseados em Valor .....	38
2.7.10 Mudança como Oportunidade .....	40
2.7.11 Relacionamento entre os Elementos-chave e o <i>Framework</i> de Processo da VBSE .....	41
2.8 Discussão .....	41
3 Criação de Valor no RUP .....	44
3.1 O RUP e a Criação de Valor .....	44
3.2 Convergência entre RUP e VBSE quanto a Criação de Valor .....	46

3.2.1	Análise de Realização de Benefícios .....	47
3.2.2	Identificação e Conciliação de Proposições de Valor das Partes Interessadas .....	47
3.2.3	Análise de Caso de Negócio .....	49
3.2.4	Gerenciamento Contínuo de Riscos e Oportunidades .....	49
3.2.5	Engenharia Simultânea de Software e de Sistema .....	50
3.2.6	Monitoramento e Controle Baseados em Valor .....	52
3.2.7	Mudança como Oportunidade .....	52
3.2.8	Resumo da Identificação de Convergência .....	54
3.3	RUP-VC: Uma Extensão do RUP para Criação de Valor .....	55
3.3.1	Análise de Realização de Benefícios .....	56
3.3.2	Identificação e Conciliação de Proposições de Valor das Partes Interessadas .....	60
3.3.3	Análise de Caso de Negócio .....	64
3.3.4	Gerenciamento Contínuo de Riscos e Oportunidades .....	66
3.3.5	Engenharia Simultânea de Sistemas e de Software .....	68
3.3.6	Monitoramento e Controle Baseados em Valor .....	69
3.3.7	Mudança como Oportunidade .....	70
3.4	O Site da Extensão RUP-VC .....	71
3.5	Considerações sobre a Análise .....	73
4	Avaliação da Extensão RUP-VC .....	76
4.1	O Processo de Avaliação .....	76
4.2	Definição dos Objetivos .....	76
4.3	Planejamento .....	77
4.3.1	Definir a Abordagem para a Avaliação .....	78
4.3.2	Elaboração do Questionário .....	78
4.3.3	Preparação do Resumo da Proposta .....	79
4.3.4	Definição de Data e Local do Workshop .....	79
4.3.5	Convocação do Público-alvo .....	79
4.3.6	Preparação do Material para o Workshop .....	79
4.4	Execução .....	80
4.5	Análise e Interpretação dos Resultados .....	80
4.5.1	Conclusões da Análise .....	96
4.6	Implantação das Melhorias .....	97
4.7	Considerações Finais .....	98
5	Conclusões e Trabalhos Futuros .....	99
5.1	Principais Contribuições .....	99
5.2	Trabalhos relacionados .....	100
5.3	Trabalhos Futuros .....	102
5.4	Considerações Finais .....	104
	Referências .....	106
	Apêndice A - Pontos de Extensão do RUP-VC .....	113
	Apêndice B - Exemplo de Elemento Estendido pelo RUP-VC .....	115
	Apêndice C - Questionário de Avaliação do RUP-VC .....	119

## Lista de Figuras

4	
Figura 2.1 - Visão bidimensional do Rational Unified Process (IBM, 2006).....	13
Figura 2.2 - Exemplo das fases e iterações do RUP. Fonte: IBM (2006).....	18
Figura 2.3 – Resultados da resolução de projetos de software. Fonte: (STANDISH GROUP, 2008).....	19
Figura 2.4 – Modelo empresarial de negócios. Fonte: (MORRIS; JAMIESON, 2004) .....	21
Figura 2.5 - O contexto da governança de projetos. Adaptado de: (WEAVER, 2007) .....	24
Figura 2.6 – Estrutura da teoria "4+1" da VBSE. Adaptado de (BOEHM; JAIN, 2006).....	28
Figura 2.7 – Teoria da VBSE como um <i>framework</i> de processo. Adaptado de (BOEHM; JAIN, 2006).....	30
Figura 2.8 – Cadeia de Resultados. Adaptado de (BOEHM, 2003) .....	33
Figura 2.9 – Diagrama de conflito de modelos. Adaptado de Boehm (2003) .....	35
Figura 2.10 – Processo de Feedback de Realização de Valor. Adaptado de (BOEHM, 2003).....	39
Figura 3.1 - Tarefas e papéis do fluxo Gerenciar Requisitos Variáveis do RUP. Fonte: (IBM,2006).....	54
Figura 3.2 - Fluxo da disciplina Gerenciamento de Portfólio de Projetos do RUP-VC .....	59
Figura 3.3 - Home page do site do RUP-VC.....	71
Figura 3.4 - Detalhe da home page do RUP-VC (link para o conteúdo da extensão).....	72

## Lista de Tabelas

Tabela 1.1 - Metodologia da Pesquisa (definida pelo autor).....	4
Tabela 2.1 - Mapeamento dos elementos-chave da VBSE com o <i>framework</i> de processo _____ (resultado obtido pelo autor).....	42
Tabela 3.1 – Pontos de Convergência por Elemento-chave (resultado obtido pelo autor).....	54
Tabela 4.1 - Caracterização da avaliação como estudo experimental _____ (resultado obtido pelo autor).....	77
Tabela 4.2 - Perfil dos respondentes.....	81

# 1 Introdução

Este capítulo descreve as principais motivações para a realização deste trabalho, apresenta os objetivos e a abordagem utilizada. Será apresentado também o escopo do trabalho, bem como a estrutura da dissertação.

## 1.1 Contexto e Motivação

De um ponto de vista econômico, considerando o contexto das organizações, Bowman e Ambrosini (2000) definem dois tipos de valor:

- *Valor de uso*: refere-se às qualidades específicas de um produto, serviço ou tarefa, tal como percebidas pelo cliente a partir das suas necessidades. O valor de uso é, portanto, um julgamento subjetivo que varia de indivíduo a indivíduo.
- *Valor de troca*: refere-se a preço. É a quantia monetária realizada em um determinado ponto no tempo, no qual se dá a troca por mercadorias ou serviços.

A partir dos conceitos acima, Lepak et al. (2007) constroem a definição de *criação de valor* como sendo a geração de uma quantidade relativa de valor – avaliada subjetivamente, pois se trata de valor de uso – que é realizada por um indivíduo ou organização (o usuário) quando este investe uma quantia monetária na aquisição ou construção de um produto, serviço ou atividade. A criação de valor ocorre porque a quantia monetária trocada excede os custos do produtor, ao mesmo tempo em que compensará a diferença de desempenho entre a opção atual (produto ou serviço atual) e o novo produto ou serviço adquirido pelo usuário.

Uma das conclusões da pesquisa promovida pelo Engineering and Physical Sciences Research Council (EPSRC) britânico e apresentada em (WINTER *et al.*, 2006), quanto aos rumos que a pesquisa em gestão de projetos deverá tomar para além dos fundamentos conceituais atuais, é o estudo da *criação de valor* pelos projetos como foco principal da gestão de projetos nas organizações, em vez da atual ênfase na criação de *produtos*. O estudo identificou que, para muitas organizações, a principal preocupação não está mais nos ativos de capital, sistemas e instalações, *mas cada vez mais está no desafio de vincular a estratégia de negócios aos projetos*, maximizando a geração de receitas, e gerindo a entrega de benefícios

em relação aos diferentes grupos de interessados.

Nesse sentido, os autores sugerem, por exemplo, um maior esforço de pesquisa em relação aos processos de seleção estratégica de projetos e programas nas organizações, de forma a *reconhecer diferentes formas de valor* além das representações convencionais da produção e da manufatura. Considerando que uma organização é qualquer combinação de esforços individuais que tem por finalidade realizar propósitos coletivos (MAXIMIANO, 1992), um outro aspecto levantado é a noção de valor como tendo múltiplos significados, dependendo dos propósitos individuais ou organizacionais associados. A complexidade deste caráter múltiplo leva por vezes à percepção de que a criação de valor, dependendo do escopo do projeto, estende-se por um prazo longo, não podendo ser limitada ao período entre a iniciação e o encerramento dos projetos.

Relacionado à noção mais ampla de valor em projetos está o conceito de objetivo estratégico de uma organização. Morris e Jamieson (2004) constatam que a estratégia corporativa é em última instância traduzida em coleções – ou portfólios – de programas ou projetos, que se tornam os veículos para a implementação das iniciativas estratégicas aprovadas. Portanto, a gestão de projetos dá suporte aos processos de entrega de valor, no sentido mais amplo, aos clientes.

Com relação aos projetos de Tecnologia da Informação (TI), dentre as evidências de que a criação de valor é um fator de grande importância a ser focado está o estudo de Thorp (2007), que sustenta que ocorre um problema gerencial, com relação às iniciativas de TI executadas para maximizar o valor das informações produzidas em decorrência da execução dos processos de negócios; Thorp denominou tal problema o *paradoxo da informação*. Por este paradoxo, tais iniciativas de TI, quando implementadas, findam por produzir muito mais informação ainda; com isso, não cumprem o que prometem com relação a dar resultados de negócios para as empresas e outras organizações nas quais são executadas.

No que se refere aos resultados dos projetos de TI, merecem um estudo mais aprofundado os projetos de desenvolvimento de software, devido aos reconhecidos problemas em sua efetividade (STANDISH GROUP, 1995; STANDISH GROUP, 2008). A Engenharia de Software têm evoluído no sentido de prover métodos e processos para aumentar a qualidade dos produtos e a probabilidade de sucesso dos projetos de desenvolvimento. A gestão de projetos tem sido uma das competências consideradas essenciais pelos principais

padrões de melhoria de processo de software (SEI, 2006; SOFTEX, 2007).

Um dos campos de estudo da Engenharia de Software é a Economia de Software, que lida com o aperfeiçoamento do valor criado pelos investimentos de software. Uma abordagem ligada à Economia de Software é a Engenharia de Software Baseada em Valor (VBSE), que propõe integrar considerações de valor às técnicas atuais da Engenharia de Software (BOEHM, 2003).

Do exposto acima, surgiu a escolha do tema da dissertação de Mestrado aqui proposta: a criação de valor em projetos. Dentro deste tema, há interesse em descobrir como a criação de valor acontece especificamente nos projetos de software. Para este trabalho, o contexto limita-se às empresas ou organizações desenvolvedoras de software, isto é, aquelas cujo negócio principal é entregar a seus clientes produtos de software. Neste contexto incluem-se as *software houses*, bem como as unidades de negócio responsáveis pelo desenvolvimento de sistemas de software para uso pelas organizações. Da mesma forma, o valor, cuja criação se busca verificar neste trabalho, refere-se aos ganhos e benefícios desejados pelas principais partes interessadas na dinâmica do desenvolvimento de software nessas organizações: os clientes, os usuários, a equipe de desenvolvedores, os gerentes dos projetos de desenvolvimento, a gerência da organização, e seus acionistas.

Mais especificamente, o tópico escolhido dentro do tema suscita a seguinte pergunta: “como fazer com que os projetos de desenvolvimento de software, que seguem práticas e princípios da Engenharia de Software, deem mais ênfase à criação de valor?”. Para responder a essa pergunta decidiu-se, como trabalho a ser desenvolvido nesta dissertação de Mestrado, pela elaboração de *uma análise, e consequente extensão, de um framework de processo de software, com base na criação de valor.*

## **1.2 Objetivos**

Os principais objetivos deste trabalho são, portanto:

- Identificar pontos de convergência entre o RUP e os elementos-chave da Engenharia de Software Baseada em Valor;
- Propor uma extensão ao RUP, nos pontos em que ele não converge à VBSE.

As razões para a escolha do RUP como processo a ser analisado e estendido foram:

1. O RUP é um processo de ampla aceitação de mercado (DAVIDSON, 2008; LAMRI, 2004).
2. O RUP traz princípios e boas práticas de desenvolvimento de software, dos quais é possível avaliar se apresentam convergência aos elementos-chave da Engenharia de Software baseada em Valor. Da análise, será possível tirar conclusões sobre o quanto o RUP se presta à criação de valor no desenvolvimento de software.

As razões para a escolha da VBSE como fio condutor da análise de valor foram:

1. É uma iniciativa estruturada, com o claro objetivo de integrar considerações de valor para as práticas e técnicas da Engenharia de Software (BOEHM, 2003);
2. Sua abrangência de pesquisa é suficientemente grande de modo a se adequar aos diversos tipos de processos e metodologias, desde os consagrados quanto os emergentes, como se pode comprovar na compilação de trabalhos artigos presente em Biffel et al. (2006).

**Tabela 1.1** - Metodologia da Pesquisa (definida pelo autor)

<b>Fases da Pesquisa</b>	<b>Metodologia Adotada</b>
Fase 1: Definição do Escopo	<ul style="list-style-type: none"><li>• Selecionar um tema associado à área de interesse;</li><li>• Escolher o tópico a pesquisar dentro do tema;</li><li>• Determinar, no contexto do tópico, a pergunta-problema de pesquisa a ser respondida.</li></ul>
Fase 2: Pesquisa Bibliográfica	<p>Pesquisar na literatura os assuntos relacionados à pergunta-problema:</p> <ul style="list-style-type: none"><li>• Rational Unified Process – RUP;</li><li>• Alinhamento estratégico e criação de valor em projetos;</li><li>• Engenharia de Software e criação de valor.</li></ul>



Fases da Pesquisa	Metodologia Adotada
Fase 3: Análise e Extensão	<ul style="list-style-type: none"> <li>• Analisar pontos de convergência do RUP à VBSE;</li> <li>• Identificar pontos de extensão;</li> <li>• Criar uma extensão ao RUP com os pontos identificados.</li> </ul>
Fase 4: Avaliação da Extensão	<p>Avaliar qualitativamente a extensão proposta na prática:</p> <ul style="list-style-type: none"> <li>• Realizar workshops para apresentação do tema;</li> <li>• Aplicar questionários;</li> <li>• Analisar as respostas obtidas;</li> <li>• Caso necessário, aperfeiçoar a extensão após análise, a partir do feedback das respostas.</li> </ul>

### 1.3 Metodologia

O desenvolvimento deste trabalho adotou uma estratégia de evolução contínua e não de etapas linearmente subsequentes. A pesquisa apresenta basicamente dois momentos com focos distintos. No primeiro, de caráter eminentemente exploratório, foram feitas a revisão bibliográfica e a coleta das informações necessárias que foram utilizadas para a concepção do modelo. Já no segundo momento da pesquisa, o foco passou a ser a descrição e avaliação da análise e da extensão propostas. A metodologia utilizada neste trabalho de pesquisa é mostrada de forma detalhada na Tabela 1.1.

### 1.4 Estrutura da Dissertação

No Capítulo 2 são apresentados os conceitos de Projeto de Software e Processo de Software. São discutidos vários aspectos sobre esses conceitos, como a gestão de projetos e seu alinhamento em relação à estratégia empresarial e qualidade de software, e possíveis caminhos sobre como se pode proceder à criação de valor em projetos de software. Também aborda a Engenharia de Software Baseada em Valor, que é um *framework* de práticas e métodos para a ênfase na criação de valor em projetos de software.

O Capítulo 3 dedica-se a apresentar a contribuição principal deste trabalho, composta de duas partes:

- A análise do Rational Unified Process à luz da Engenharia de Software Baseada em Valor, apontando pontos de convergência do processo aos métodos e práticas definidas naquele *framework*;
- Uma extensão do RUP com atividades, diretrizes e outros elementos que proporcionem uma maior ênfase na criação de valor.

O Capítulo 4 detalha o processo de avaliação da extensão elaborada, explicando as atividades e os passos da metodologia, e analisando e os resultados obtidos.

Finalmente, o Capítulo 5 apresenta as conclusões sobre a pesquisa a partir dos resultados da avaliação, e sugere trabalhos futuros.



## 2 Processos, Projetos e Criação de Valor em Engenharia de Software

Este capítulo apresenta alguns conceitos necessários à compreensão dos assuntos que serão tratados no próximo capítulo. Na primeira seção, é apresentado o conceito de valor que será adotado neste trabalho. Na Seção 2.2 é feita uma introdução aos processos de software. Na Seção 2.3 é apresentado o Rational Unified Process. Em seguida, a Seção 2.4 apresenta alguns aspectos da gestão de projetos de software. A Seção 2.5 trata de aspectos do alinhamento da estratégia de negócios com a estratégia de projetos de software, falando da gestão de projetos, programas e portfólios.

A Seção 2.6 traz uma apresentação do campo de estudo conhecido por Economia de Software, seguida da apresentação de uma abordagem relacionada, a Engenharia de Software Baseada em Valor, na Seção 2.7. O capítulo finaliza com uma discussão dos assuntos apresentados, na Seção 2.8.

### 2.1 Definição de Valor

A fim de trabalhar com a criação de valor como objetivo perseguido, é preciso que se tenha uma noção clara da abrangência do valor que se quer constatar como possível de ser criado. Como afirmam BOEHM e SULLIVAN (2000), há muitas dimensões nas quais o valor pode ser avaliado, do lucro monetário à solução de problemas sociais. Assim, como o problema de pesquisa deste trabalho envolve um contexto organizacional onde são executados projetos de desenvolvimento de software, o conceito de valor a ser adotado deve estar ligado ao conceito de estratégia de negócios, mas também ligado ao coletivo (dado que se lida com equipes desenvolvendo software para organizações), e ainda ao aspecto tradicional de valor financeiro. Portanto, para os fins deste trabalho entende-se que o *valor* entregue por um projeto refere-se aos seguintes aspectos:

- Valor em termos financeiros, na forma do retorno sobre o investimento no projeto;
- Benefícios intangíveis trazidos pelo projeto, isto é, aqueles difíceis de quantificar financeiramente, como por exemplo, melhoria da imagem da organização junto aos

clientes;

- Aproveitamento de oportunidades de negócio propiciadas pelo projeto, ou potencializadas pelos resultados do projeto.

Nesse mesmo sentido, considera-se neste trabalho que *criação de valor* significa a produção de valor tal qual definido acima, contribuindo para os objetivos estratégicos de negócio da organização.

A esta altura é conveniente introduzir o conceito, oriundo do marketing, de “proposição de valor”, também conhecido como “proposta de valor”. Segundo Kotler (2003), a proposição de valor de um produto é a experiência total prometida pelo produto, respaldada por seu funcionamento efetivo e confiável. Matos (2004) vai além ao definir proposição de valor como o elemento básico da estratégia. Nesse sentido, a empresa pode partir de sua proposição de valor básica (isto é, a combinação de produto, serviço, preço e canal de distribuição que a empresa oferece aos seus clientes), focalizar seus componentes essenciais e criar uma maneira singular de proporcionar esse valor aos clientes – a estratégia. Esta visão de proposta de valor como elemento estratégico é compartilhada pela metodologia *Balanced Scorecard* (KAPLAN; NORTON, 1997), que inclui os atributos das propostas de valor como medidas fundamentais de desempenho da organização, relativas à perspectiva estratégica dos clientes.

## **2.2 Processos de Software**

Sommerville (2001) define *processo de software* como um conjunto de atividades e resultados associados que levam à produção de um produto de software, seja ele construído partindo-se do zero ou modificando sistemas existentes. Pressman, por sua vez, define processo de software como um arcabouço para as tarefas necessárias à construção de software de alta qualidade (PRESSMAN, 2006 p.16). Tal definição leva à discussão do conceito de qualidade em software. Também de acordo com Pressman, qualidade de software é “a satisfação de requisitos funcionais e de desempenho explicitamente declarados, normas de desenvolvimento explicitamente documentadas e características implícitas que são esperadas em todo o software desenvolvido profissionalmente”.

Do exposto acima se conclui que um processo de software, ou seja, o conjunto das

normas de desenvolvimento sobre as quais fala a definição de qualidade, deve ser explicitamente declarado e seguido. Também se conclui que os requisitos de um software, tanto os explicitamente declarados quanto os implícitos, são a fundação a partir da qual a qualidade é medida. É importante lembrar que o conceito de valor está intimamente ligado aos requisitos, em especial os implícitos, que estão embutidos nas demandas, nas expectativas e nos desejos dos usuários e outras partes interessadas com relação ao software.

### 2.2.1 Processos Ágeis e Processos Disciplinados

Segundo Boehm e Turner (2004), é possível diferenciar os processos de software a partir de duas abordagens básicas. Uma delas concentra-se no conceito de *disciplina*, onde o conjunto de atividades de planejamento do processo guia o projeto e determina em grande parte a consecução dos objetivos. A outra abordagem enfatiza a *agilidade*, no sentido de adaptar o projeto às mudanças – organizacionais, de escopo, de prazos, entre outras – demandadas pelas suas partes interessadas.

A abordagem disciplinada (ou *orientada a planejamento*<sup>1</sup>), geralmente considerada a forma tradicional de desenvolver software, origina-se nas disciplinas de engenharia de sistemas e de qualidade, e em sua aplicação nos primeiros grandes sistemas aeroespaciais, formados por hardware e software. Com o objetivo de integrar-se facilmente ao ambiente de desenvolvimento de sistemas como um todo, esses métodos refletiam as necessidades dos processos de engenharia pré-existentes. Devido a essa origem, os processos orientados a planejamento caracterizam-se por uma sistematização do desenvolvimento, preocupação com a completude da documentação a cada passo do processo, concretizada através de verificações e revisões. Seguem a abordagem disciplinada os *frameworks* de processos PSP (HUMPHREY, 1994) e TSP (HUMPHREY, 2006), RUP (IBM, 2006) e o *framework* de melhoria de processos CMMI-DEV (SEI, 2006).

A abordagem ágil, por sua vez, é fruto das experiências com prototipagem rápida e desenvolvimento rápido, bem como do ressurgimento da filosofia de que programação é mais uma arte do que um processo industrial (BOEHM e TURNER, 2004 p.16). O termo

---

<sup>1</sup> Tradução livre, do inglês *plan-driven approach*.

“caórdico”<sup>2</sup> foi cunhado como significando o trabalho que unifica o caos e a ordem, de forma a desafiar o gerenciamento tradicional de projetos de software que se dá através de processos lineares e orientados a planejamento. Os processos ágeis vêm ao encontro do trabalho “caórdico”. Como características marcantes, os processos ágeis são leves, empregam ciclos iterativos curtos, envolvem ativamente os usuários na definição e priorização de requisitos, e baseiam-se no conhecimento tácito da equipe de desenvolvimento ao invés de na documentação. Dentre os métodos ágeis, podemos destacar eXtreme Programming (XP) (BECK, 2004), Crystal (COCKBURN, 2001) e Scrum (SCHWABER, 2004).

Como objeto da pesquisa apresentada neste trabalho, foi escolhido um adepto da abordagem disciplinada, o Rational Unified Process, descrito a seguir.

### **2.3 Rational Unified Process (RUP)**

O RUP é um processo de software que adere à abordagem disciplinada na atribuição de tarefas e responsabilidades dentro de uma organização de desenvolvimento de software (KRUCHTEN, 2003, p.19).

O RUP, na sua versão 7.0.1, incorpora três elementos centrais (IBM, 2006):

- Um conjunto fundamental de filosofias e princípios para o desenvolvimento de software bem sucedido: esse conjunto está expresso nos *princípios-chave* e nos *elementos essenciais do processo*<sup>3</sup>;
- Um framework de conteúdos de método reutilizáveis e componentes para construção de processo: consiste em uma família de plug-ins que permite criar configurações de processo personalizadas ou processos adaptados de acordo com as necessidades;
- Um método e uma linguagem subjacentes para definição de processos: o RUP é descrito na linguagem SPEM<sup>4</sup> versão 2.0, que é uma unificação de diferentes

---

<sup>2</sup> Tradução livre da expressão em inglês “chaordic”.

<sup>3</sup> Tradução livre das expressões originais em inglês *Key Principles e Process Essentials*.

<sup>4</sup> Acrônimo para *Software Process Engineering Metamodel*.

linguagens de engenharia de processo, tais como SPEM versão 1.0 e a linguagem usada por versões anteriores do RUP.

### **2.3.1 Elementos Essenciais do RUP**

O RUP baseia-se em boas práticas para desenvolvimento de software, expressas pelos dez elementos essenciais:

1. Visão: desenvolver uma visão compartilhada que capture as necessidades reais das partes interessadas;
2. Plano: gerenciar de acordo com um plano abrangente, que envolva a concepção, o monitoramento e o controle, e a avaliação de diversos aspectos do projeto;
3. Riscos: é essencial identificar e mitigar riscos e rastrear problemas relacionados;
4. Caso de Negócios: examinar as justificativas econômicas, que determinam se compensa, ou não, investir no projeto;
5. Arquitetura: projetar a estrutura dos componentes mais significativos do sistema, e suas interfaces;
6. Protótipo: construir o sistema progressivamente e testar o produto também de forma incremental;
7. Avaliação: avaliar resultados (status, progresso) do projeto regularmente, a cada iteração;
8. Controle de Mudanças: gerenciar e controlar as solicitações de mudança dos usuários, e outras alterações;
9. Suporte ao Usuário: implementar um produto utilizável, prestando atenção em treinamento e documentação de uso e instalação;
10. Processo: adotar um processo que se ajuste ao projeto e ao produto.



A seguir, será apresentada a estrutura do RUP, nos seus dois aspectos principais: a *estrutura estática*, que descreve o conteúdo do processo através de vários tipos de elementos de conteúdo; e a *estrutura dinâmica*, que descreve os fluxos e sequências de atividades a executar em um projeto. A Figura 2.1, conhecida como “gráfico das baleias”, sumariza as duas visões da estrutura.

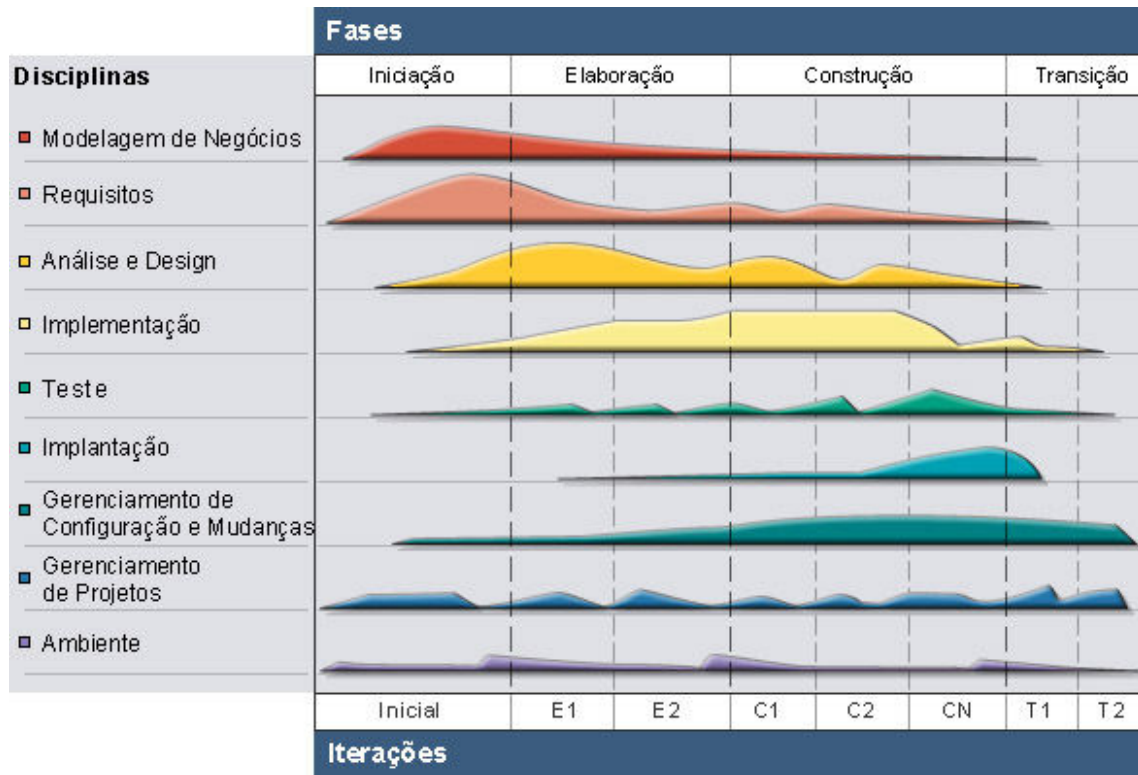


Figura 2.1 - Visão bidimensional do Rational Unified Process (IBM, 2006)

### 2.3.2 Estrutura Estática – A Descrição do Processo

O conteúdo do RUP é estruturado em elementos de conteúdo chamados *disciplinas*. Estas são coleções de atividades, denominadas *tarefas*, relacionadas em uma área de interesse maior dentro do projeto de software. O RUP define nove disciplinas (KRUCHTEN, 2003):

- Modelagem de Negócios: envolve tarefas que permitem modelar a dinâmica, a estrutura e o modelo de gestão da organização na qual o sistema de software será implantado. Permite levantar e visualizar problemas atuais da organização, como gargalos e redundâncias de atividades nos processos de negócio, e identificar possibilidades de melhoria.

- Requisitos: tem por objetivo definir, em consenso por todos os interessados pelo sistema, uma visão do que o sistema deve fazer, especificando suas características técnicas e funcionais (requisitos), e definir as fronteiras do sistema. Essa visão deve ser compreendida pela equipe de desenvolvimento, e deve fornecer uma base para a estimativa de custo e tempo de desenvolvimento e para o planejamento das iterações (definidas adiante nesta seção).
- Análise e Design: disciplina que visa traduzir os requisitos em uma especificação que guiará a implementação; visa também desenvolver a arquitetura do sistema e adequar o design para o ambiente de implantação.
- Implementação: reúne as atividades de implementação das classes e objetos em termos de componentes (código-fonte, arquivos binários, executáveis e outros), testes de unidade nesses componentes e integração dos mesmos ao sistema executável.
- Teste: realiza atividades centradas na avaliação do produto de software, abrangendo a localização e registro de falhas que comprometam a qualidade do software, e validação das funcionalidades do software confrontando com os requisitos e com o design.
- Implantação: tem o objetivo de colocar o sistema em produção. Inclui testar o sistema no ambiente de produção, empacotar o software para distribuição (*shrink-wrapped software*), distribuir e/ou instalar o software, realizar treinamento de usuários e migração de dados.
- Gerenciamento de Configuração e Mudanças: disciplina que tem a finalidade de registrar e manter as trilhas das alterações e evoluções dos artefatos que são produzidos no projeto, através da definição de políticas de gerenciamento da configuração. O RUP é um processo iterativo, portanto mudanças são frequentes; esta disciplina trata de acompanhá-las, através de solicitações de mudanças que devem ser registradas, avaliadas, e executadas caso aprovadas.
- Gerenciamento de Projetos: O objetivo desta disciplina é fornecer um *framework* para gerenciar projetos de software, com diretrizes práticas para planejar, montar a

equipe, executar e monitorar os projetos. Também deve dar subsídios para a realização do gerenciamento de riscos do projeto.

- Ambiente: disciplina que tem como meta fornecer à organização o ambiente para o desenvolvimento de software, com processos e ferramentas que dão suporte à equipe de desenvolvimento. Concentra-se na execução das atividades necessárias à configuração do processo de desenvolvimento de software.

Uma *tarefa*, por sua vez, descreve uma unidade de trabalho dentro do processo; uma tarefa possui uma finalidade clara, geralmente expressa em termos da criação ou atualização de alguns *produtos de trabalho*, que são os resultados do processo. Dentre os produtos de trabalho estão as entregas como os *releases* executáveis, ou os artefatos como modelos, classes ou planos, por exemplo. As tarefas podem ser divididas em seções de *etapas* (ou *passos*). Uma etapa descreve uma parte significativa e consistente do trabalho geral descrito para uma tarefa.

Outro elemento de conteúdo importante do RUP são as *funções*. Uma função é um elemento que define um conjunto de habilidades, competências e responsabilidades relacionadas. As funções são utilizadas pelas tarefas para especificar quem as desempenha, bem como definir um conjunto de produtos de trabalho, pelos quais são responsáveis. Eis algumas das funções mais importantes definidas no RUP:

- Gerente do Projeto: tem a responsabilidade de alocar recursos, definir prioridades, coordenar interações com cliente e usuários, e manter a equipe concentrada nas metas do projeto. Também estabelece as práticas que asseguram a integridade e a qualidade dos artefatos do projeto.
- Analista de Sistemas: lidera e coordena o levantamento de requisitos e a modelagem de casos de uso através da definição das funcionalidades e da delimitação do sistema, identificando quais atores existem e quais casos de uso são necessários para as interações com o sistema.
- Arquiteto de Software: tem a responsabilidade de conduzir as principais decisões técnicas do desenvolvimento, expressas na arquitetura do software. Isto inclui, entre outras atividades, identificar e documentar os aspectos significativos do

ponto de vista da arquitetura, inclusive as visões de requisitos, design, implementação e implantação do sistema, e equilibrar as preocupações dos vários interessados, lidando com os riscos técnicos envolvidos.

- Implementador: responsável por desenvolver e testar componentes, de acordo com os padrões adotados no projeto, para integração em subsistemas maiores. Também deve criar os componentes de testes necessários para apoiar as atividades de testes.
- Testador: responsável pelas atividades centrais do esforço de testes, o que envolve conduzir os testes necessários e registrar os resultados.

Por fim, há uma série de elementos de conteúdo destinados a fornecer orientação ao usuário do processo. Dentre estes elementos, encontram-se as *diretrizes* (ou *guias*), que fornecem detalhes adicionais sobre como tratar de um elemento de conteúdo específico; os *conceitos*, que destacam ideias principais e princípios básicos dando suporte a um tópico central do RUP; e *checklists*, que identificam uma série de itens que precisam ser concluídos ou verificados durante a realização das tarefas.

### 2.3.3 Estrutura Dinâmica – O Ciclo de Vida do Processo

O ciclo de vida de desenvolvimento no RUP é composto por quatro *fases* distintas, com objetivos diferentes, e foco na entrega de artefatos diferentes. São elas:

- Iniciação<sup>5</sup>: é a primeira fase do RUP, na qual a motivação inicial para o desenvolvimento do software é conduzida até o ponto de ser patrocinada (ao menos dentro da organização) para entrar na próxima fase, de Elaboração. Seus objetivos principais são: especificar a visão do produto final e seu caso de negócio; e definir o escopo do projeto.
- Elaboração: na segunda fase do ciclo de vida do processo, a visão do produto e sua arquitetura são amadurecidas. Os objetivos: planejar as atividades necessárias e os recursos exigidos para tanto; especificar as funcionalidades; e projetar a arquitetura.

---

<sup>5</sup> Neste trabalho, preferiu-se traduzir a denominação em inglês *Inception* para o termo presente na documentação do RUP, ao invés da tradução já consagrada *Concepção*.

- Construção: nesta terceira fase, o software é transformado, a partir de uma base arquitetural executável até o ponto no qual ele estará pronto para ser implantado na comunidade de usuários. Objetivos: construir o produto e evoluir a visão, a arquitetura e os planos até que o produto esteja pronto para entrega aos usuários.
- Transição: na quarta e última fase do ciclo de vida, o software é entregue à comunidade de usuários. Objetivos: transicionar o produto aos seus usuários, o que inclui fabricar, entregar, treinar, dar suporte e manutenção ao produto buscando a satisfação dos usuários.

As fases são delimitadas no tempo por marcos de ciclo de vida, cada um deles ao final de uma das fases. São eles:

- LCA (*Life Cycle Objectives*, ou Marco dos Objetivos do Ciclo de Vida), ao final da fase de Iniciação;
- LCO (*Life Cycle Architecture*, ou Marco da Arquitetura), ao fim da fase de Elaboração;
- IOC (*Initial Operational Capability*, ou Marco de Recurso Operacional Inicial), ao final da fase de Construção;
- PR (*Product Release*, ou Marco de Liberação do Produto), ao final da fase de Transição.

As fases também podem ser subdivididas em *iterações*, que são agrupamentos de atividades que levam a um release do produto, como mostra a Figura 2.2. Uma iteração de desenvolvimento é, de alguma forma, uma passagem completa por pelo menos as disciplinas de Requisitos, Análise e Design, Implementação e Teste. É como um pequeno projeto cascata em si mesmo. Tal como as fases, as iterações devem ter elas próprias seus marcos de ciclo de vida, onde os critérios de sucesso planejados para elas devem ser avaliados.

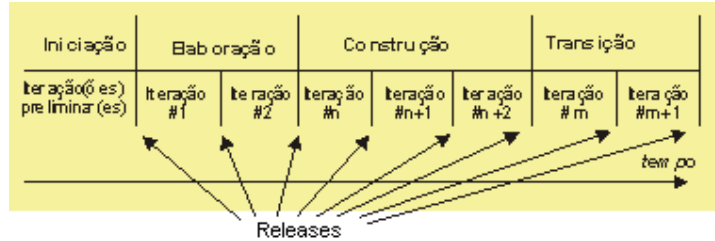


Figura 2.2 - Exemplo das fases e iterações do RUP. Fonte: IBM (2006)

## 2.4 Gestão de Projetos de Software

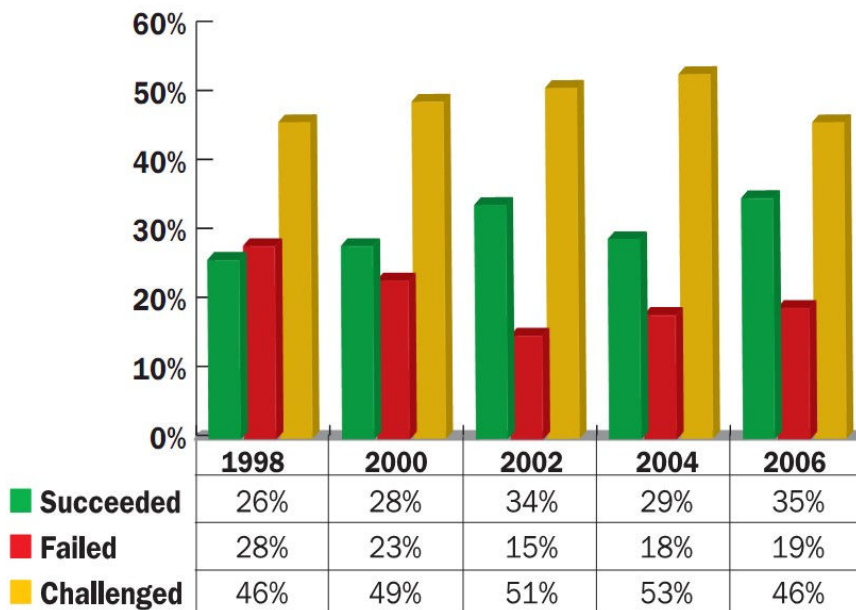
A gestão de projetos, ou gerenciamento de projetos, é uma área de conhecimento em franca expansão atualmente. Baseia-se no conceito de *projeto* que, segundo o Guia PMBOK, do Project Management Institute (PMI), é um empreendimento temporário – isto é, com início e final definidos – cujo objetivo é entregar um produto, serviço ou resultado exclusivo (PMI, 2004 p.5). O mesmo PMBOK define gerenciamento de projetos como “a aplicação de conhecimento, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender aos seus requisitos” (PMI, 2004 p.8).

A partir do conceito de projeto, é possível derivar o conceito de *projeto de TI*, como sendo um projeto cuja entrega para a organização cliente é composta de um conjunto de produtos e serviços relacionados à tecnologia da informação. Dentre os projetos de TI, é importante destacar os *projetos de software*, que se encarregam do desenvolvimento – isto é, da definição, construção e implantação – de sistemas de software.

A gestão de projetos de software tem características singulares devido à peculiaridade do produto resultante desses projetos. Segundo Royce (1998), a melhor coisa com relação a software também é a pior: sua flexibilidade. A mesma característica que permite ao software transformar-se em “quase qualquer coisa” também torna difícil planejar, monitorar e controlar seu desenvolvimento. Esta característica eleva os riscos de insucesso nos projetos de software. A Figura 2.3 mostra um gráfico que sumariza os resultados do estudo do Standish Group que apontam que, nos últimos anos, a grande maioria dos projetos de desenvolvimento de novos sistemas fracassaram (*failed*) ou tiveram seu sucesso comprometido (*challenged*) (STANDISH GROUP, 2008).

Stepanek (2005) relaciona nada menos que doze diferenças inter-relacionadas entre desenvolvimento de software e outros esforços de negócio. Dentre as diferenças, quatro delas são apontadas como causas-raiz dos insucessos dos projetos de software, a saber:

- A complexidade dos produtos de software, como consequência da complexidade dos domínios de aplicação que se pretende abranger com software;
- A crescente complexidade da tecnologia envolvida para desenvolver software, o que demanda cada vez mais esforço de atualização dos recursos humanos;



**Figura 2.3** – Resultados da resolução de projetos de software. Fonte: (STANDISH GROUP, 2008)

- O caráter abstrato do software que, aliado à complexidade, dificulta sua “materialização” para efeito de compreensão, *design* e manutenção;
- O fato de a tecnologia de construção e gestão de software mudar constante e rapidamente, indicando que desenvolvimento de software é ainda uma disciplina imatura.

Em decorrência do panorama descrito acima, as metodologias, modelos de desenvolvimento e processos de software vêm dando ênfase cada vez maior à gestão de projetos. Dentre os vários representantes da abordagem disciplinada, percebe-se esta ênfase

nos seguintes exemplos:

- No CMMI-DEV estagiado, dentre as áreas de processo necessárias para atingir o nível 2 de maturidade – o nível inicial da maturidade, abaixo do qual a organização está “no caos” – estão o planejamento de projetos, e o monitoramento e controle dos mesmos;
- O RUP contém toda uma disciplina de Gerenciamento de Projetos, contemplando diversas tarefas de planejamento e monitoramento, além de uma função específica de gerente de projeto, como visto na Seção 2.3.

A importância da gestão de projetos dentre os métodos da abordagem ágil, destaca-se, dentre outras, nas iniciativas:

- Na metodologia XP, diversas práticas são definidas para a gestão do projeto, tais como: o uso de métricas como ferramenta básica de gestão; a definição dos papéis gerenciais de *coach* e *tracker*; o planejamento em dois níveis (macro e por iteração); e o “jogo do planejamento”, no qual usuários e equipe de desenvolvimento planejam em conjunto (BECK, 2004);
- O processo Scrum é um método criado especificamente para planejamento, controle e monitoramento do desenvolvimento (SCHWABER, 2004, p.xvii).

## **2.5 Estratégia de Projetos de Software e de TI**

Neste ponto, é importante introduzir um assunto que será mencionado nos capítulos 3 e 4: a estratégia de projetos de Tecnologia da Informação. Este assunto é atual e importante, pois, como ressalta Albertin (2001), todo projeto deve considerar seu valor perante a organização para a qual está sendo executado para o seu planejamento, aprovação e execução, o que torna a determinação efetiva do valor estratégico da TI imprescindível.

Primeiramente, cabe definir o que é estratégia de uma organização. Uma definição típica refere-se à estratégia como sendo os planos da alta gerência para atingir resultados consistentes com a missão e as metas da organização (WRIGHT et al., 1992). Outra definição é dada por Porter (1996), segundo o qual estratégia é a criação de uma posição exclusiva e de valor para a organização, envolvendo um conjunto diferente de atividades. A seguir será



abordado como alinhar a estratégia organizacional à gestão de projetos.

### 2.5.1 Alinhamento Estratégico da Gestão de Projetos

Shenhar *et al.* (2007) definem o alinhamento estratégico da gestão de projetos da seguinte maneira: o alinhamento da gestão de projetos e da estratégia de negócios é um estado interno colaborativo onde as atividades de projeto apoiam continuamente o alcance das metas estratégicas corporativas.

A colaboração entre a estratégia corporativa e a estratégia dos projetos pode ser feita de diversas maneiras. Algumas organizações utilizam modelos de gestão estruturados para a tarefa. Tais modelos identificam os principais processos de entrega de valor (*value delivery processes*) ou processos-chave de negócios, necessários para apoiar a entrega de valor aos clientes e acionistas.

Morris e Jamieson (2004) acreditam que a gerência de projetos é um processo-chave de negócios em uma organização, sendo, portanto, digno de ser incluído em um modelo empresarial de negócios. A Figura 2.4 mostra esse modelo, que retrata a estrutura de uma organização em termos de seus processos, e identifica quais são os principais processos do sistema de entrega de valor.

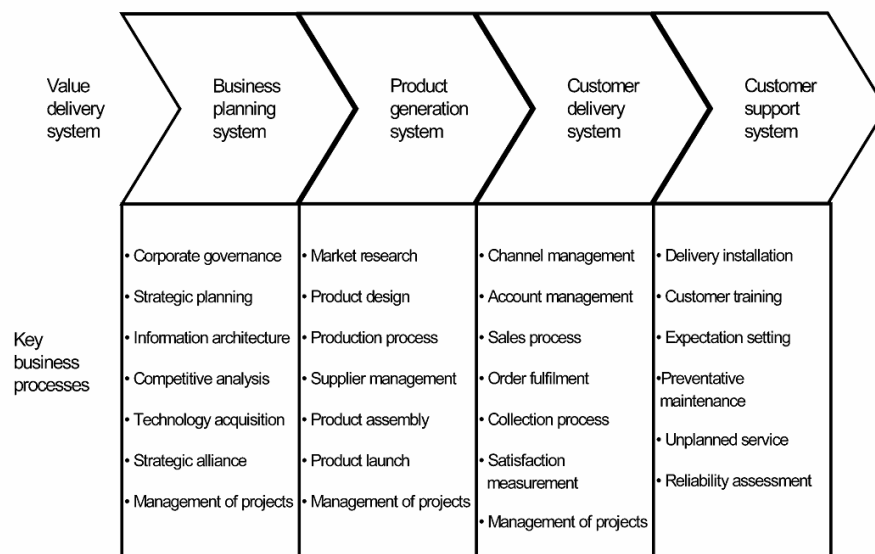


Figura 2.4 – Modelo empresarial de negócios. Fonte: (MORRIS; JAMIESON, 2004)

## 2.5.2 Gestão de Programas e Portfólios de Projetos

Segundo Morris e Jamieson (2004), a estratégia corporativa é tipicamente cascadeada através de unidades estratégicas de negócios (*strategic business units*, SBUs), e finda por ser representada como coleções – ou *portfólios* – de *programas* ou projetos. Estas se tornam os veículos para a implementação das iniciativas estratégicas aprovadas. Thorp (2007) define a gestão de programas e a gestão de portfólios como fundamentais para a realização de benefícios de negócios pelas iniciativas de TI das empresas contemporâneas.

### *Gestão de Programas*

Um programa, segundo o Guia PMBOK, é um grupo de projetos relacionados, gerenciados de modo coordenado para obtenção de benefícios e controle que não estariam disponíveis se eles fossem gerenciados individualmente (PMI, 2004, p.16). Ainda segundo o PMI, a gestão de programas é o gerenciamento centralizado e coordenado de um grupo de projetos para atingir os objetivos e benefícios estratégicos do programa.

O Office of Government Commerce (OGC) britânico, criador do método PRINCE2 de gestão de projetos, considera que o alinhamento entre estratégia e projetos seja um dos principais benefícios da gestão de programas (OGC, 1999).

Com relação a programas envolvendo projetos de TI, Thorp define um *programa consolidado de investimentos*<sup>6</sup> como um programa que congrega todas as iniciativas necessárias para a entrega de resultados de negócios. Programas dessa natureza incluem muitos tipos de projetos: implementações em TI, treinamento, marketing, mudança organizacional e redesenho de processos de negócios.

### *Gestão de Portfólio de Projetos*

O Guia PMBOK define portfólio como sendo um conjunto de projetos ou programas e outros trabalhos agrupados para facilitar o gerenciamento eficaz desse trabalho a fim de atender aos objetivos de negócio estratégicos (PMI, 2004, p.16).

---

<sup>6</sup> Tradução livre do original em inglês *blended investment program*.

Morris e Jamieson (2004) apresentam a seguinte definição de gestão de portfólio de projetos: é a atividade de alinhar as demandas por recursos à disponibilidade desses recursos a fim de se alcançar um conjunto de objetivos estratégicos. Enquanto a gestão de projetos tem a ver com “realizar da forma certa os projetos”, a gestão de portfólios de projetos tem a ver com “realizar os projetos certos”.

A gestão de portfólios de projetos se encaixa muito bem nos contextos onde há a necessidade de gerir opções de investimento numerosas, complexas e sujeitas às flutuações do ambiente e do mercado; por isso vem sendo utilizada há muito tempo no contexto das aplicações financeiras, e atualmente no ambiente de TI, que possui características análogas.

### *Governança de Projetos e Governança de TI*

Uma outra perspectiva integrante da gestão estratégica de projetos é dada pela implantação da *governança de projetos*. Para entender o que significa este conceito, torna-se antes necessário definir o que é governança corporativa. Segundo o PMI, governança corporativa é o processo pelo qual uma organização direciona e controla suas atividades operacionais e estratégicas, e pelo qual a organização responde aos direitos, expectativas e desejos legítimos das suas partes interessadas (PMI, 2006 p.106). Em suma, governança corporativa diz respeito a como uma organização toma suas decisões estratégicas de forma a criar valor aos interessados.

A governança de projetos é, então, um subconjunto da governança corporativa que se concentra nas áreas relacionadas às atividades de projeto, incluindo o direcionamento do portfólio de projetos, o patrocínio de projetos pela organização, o gerenciamento e eficiência de projetos e programas, e a produção e divulgação de informações (WEAVER, 2007). A Figura 2.5 ilustra a interseção entre governança corporativa e a disciplina de gerência de projetos, onde se situa a governança de projetos.



**Figura 2.5** - O contexto da governança de projetos. Adaptado de: (WEAVER, 2007)

Encontramos em Weill e Ross (2004) a seguinte definição para *governança de TI*: trata-se de especificar os direitos de decisão e o arcabouço de prestação de contas<sup>7</sup> a fim de encorajar um comportamento desejável no uso da TI. Esta definição, tal como as outras definições apresentadas acima, relaciona-se com as tomadas de decisão e com a responsabilidade na criação de valor que decorre delas. O “comportamento desejável” de que fala a definição é pautado pela cultura e as crenças da organização, tais como definidas e disseminadas pela estratégia, pela declaração de missão e visão, princípios de negócios e estrutura organizacional. De tudo isso se conclui que a governança de TI refere-se à criação de valor no ambiente organizacional, através da gestão responsável dos ativos e projetos de TI.

---

<sup>7</sup> Optou-se por traduzir o termo em inglês *accountability framework* para a expressão “arcabouço de prestação de contas”.

## 2.6 Economia de Software

A Economia de Software<sup>8</sup> é um campo de estudos situado na interseção entre a Economia da Informação e a Engenharia de Software. Sua preocupação principal é aperfeiçoar o valor criado pelos investimentos em software (BOEHM; SULLIVAN, 2000).

Os estudos em Economia de Software tiveram forte influência de Barry W. Boehm com seu seminal livro *Software Engineering Economics* (BOEHM, 1981), que definiu as bases para se determinar e gerenciar o valor econômico agregado pelas iniciativas de desenvolvimento de sistemas de software às organizações clientes. A obra apresenta vários conceitos e técnicas de microeconomia – tais como funções de produção, economias de escala, valor líquido, análise marginal, valor presente e teoria da decisão estatística – com exemplos de aplicação às situações de decisão a respeito de software. Na mesma obra foi introduzido o método COCOMO de estimativa de esforço para software.

Do mesmo período, são dignos de nota os trabalhos de Parnas, que trazem a noção de “projeto para mudança”<sup>9</sup>, a qual reconhece que grande parte do custo do ciclo de vida total de um sistema é resultante de sua evolução, e que um sistema que não foi projetado para evolução incorrerá em custos enormes (PARNAS, 1979).

O campo de estudos na Economia de Software evoluiu, com importantes contribuições como:

- Modelos de processos de software evolutivos, iterativos e em espiral, que utilizam gerenciamento de riscos e considerações de valor de produto para priorizar e sequenciar os incrementos de funcionalidade no sistema (BOEHM, 1988);
- O conceito de reuso de software, com seus impactos econômicos e organizacionais positivos, sobretudo na melhoria da qualidade do software e na redução do esforço de desenvolvimento (JACOBSON *et al.*, 1997);

---

<sup>8</sup> Tradução da expressão em inglês *software economics*.

<sup>9</sup> Tradução livre para o termo original em inglês *design for change*.

- A incorporação, à Engenharia de Software, de conceitos econômicos relativos à satisfação dos critérios e funções de utilidade de múltiplas partes interessadas, como nas técnicas *Participatory Design* (PD) e *Joint Application Design* (JAD) (CARMEL *et al.*, 1993), e na engenharia de requisitos baseada na *Teoria W* (BOEHM; ROSS, 1989).

Mais recentemente, Boehm e Sullivan (2000) apresentaram um roteiro de ideias para o futuro desenvolvimento da pesquisa em Economia de Software. Dentre as principais, estão:

- Engenharia de software como atividade de criação de valor: os autores constataam que há uma “desconexão” entre os critérios de decisão que tendem a guiar os engenheiros de software, e os critérios de criação de valor das organizações onde o software é desenvolvido. Assim, um ajuste importante precisa ser feito: os tomadores das decisões em relação a software precisam começar a pensar mais estrategicamente;
- Falta de combinação entre aspectos técnicos e de valor (e suas causas): a maioria dos designers e engenheiros de software não é ensinada a raciocinar tendo a criação de valor como um objetivo. Um exemplo comum de não pensar em otimização de valor em decisões sobre projetos de software é não cancelar um projeto imediatamente quando chegam novas informações que mostram que ele está fadado ao fracasso.
- Entendimento mais profundo do papel da estratégia na criação de valor: é preciso compreender que a ideia de valor é complexa e sutil, identificar novas *fontes* de valor para as decisões em software – como, por exemplo, a opção estratégica de entrar em novos mercados – e esclarecer conexões entre as dimensões técnica e econômica. Outro aspecto é entender outras *medidas* para o valor, além da financeira. A administração pública e as organizações não-governamentais, por exemplo, valorizam resultados não monetários.

Os autores apresentam um roteiro para o desenvolvimento dessas ideias, propondo diretrizes para a pesquisa em Economia de Software; tais diretrizes deram origem à abordagem chamada *Engenharia de Software Baseada em Valor*, que será apresentada a seguir.

## 2.7 Engenharia de Software Baseada em Valor

A Engenharia de Software Baseada em Valor (VBSE<sup>10</sup>), apresentada em (BOEHM, 2003), é proposta como uma abordagem que integra considerações de valor aos princípios e práticas da Engenharia de Software. Quanto à definição de valor entregue por um projeto, utilizada para esta introdução à VBSE, será utilizada a definição apresentada na Seção 2.5.2. A VBSE vem ao encontro de uma necessidade importante dos ambientes empresariais, onde considerações de valor tornam-se fator determinante para a seleção de projetos.

A estratégia da VBSE para integrar valor aos princípios e práticas da Engenharia de Software é desenvolver um *framework* abrangente, no qual tais práticas e princípios se reforcem uns aos outros. Dentre os elementos desse *framework*, encontram-se, entre outros:

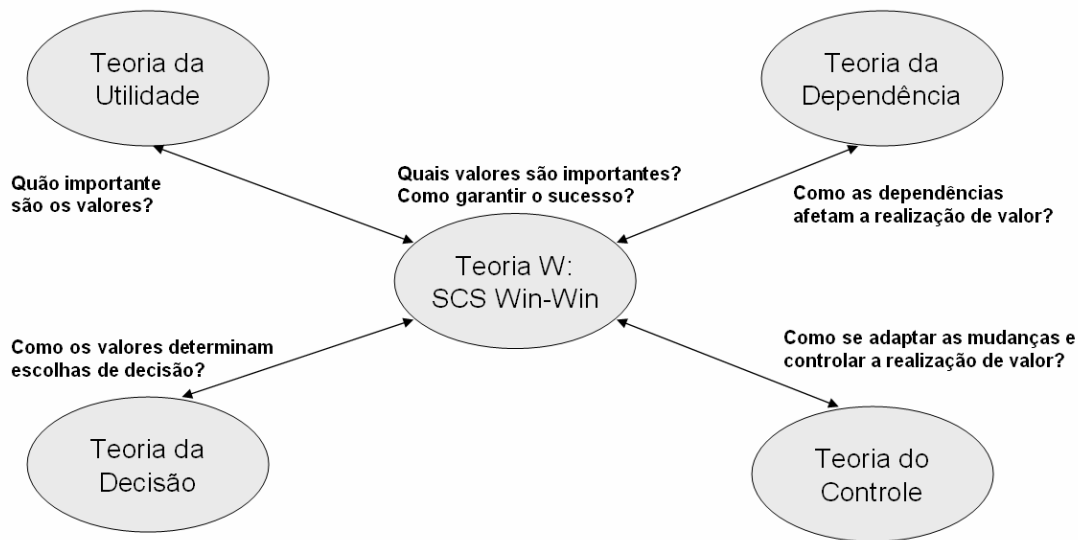
- Engenharia de requisitos baseada em valor, incluindo princípios e práticas para identificar partes interessadas críticas, fazer levantamento de suas proposições de valor e conciliá-las de forma a encontrar um conjunto de objetivos satisfatório a todos.
- Planejamento e controle baseados em valor, que incluem princípios e práticas para estender as técnicas tradicionais de planejamento e controle de custos, cronograma e produto, de modo a incluir o planejamento e o controle do valor entregue às partes interessadas.
- Gerência da qualidade baseada em valor, incluindo a priorização de fatores desejados de qualidade no que diz respeito às proposições de valor das partes interessadas.
- Gestão de pessoas baseada em valor, incluindo formação de equipes de interessados e gerência de expectativas; gestão da acomodação das proposições de valor de todas as partes interessadas durante o ciclo de vida do projeto; e integração de considerações éticas na prática diária do projeto.

### 2.7.1 Teoria Inicial da VBSE

A teoria que dá fundamentação à VBSE é apresentada por Boehm e Jain (2006), e é baseada no modelo “4+1” de visões de arquitetura de software de Kruchten (2003).

---

<sup>10</sup> Acrônimo para o termo original em inglês *Value-Based Software Engineering*.



**Figura 2.6** – Estrutura da teoria "4+1" da VBSE. Adaptado de (BOEHM; JAIN, 2006)

A Figura 2.6 ilustra as visões e o relacionamento entre elas. No centro, guiando os outros elementos da teoria da VBSE, está a Teoria W de Boehm (BOEHM; ROSS, 1989), que traz o conceito de parte interessada crítica para o sucesso do projeto (SCS)<sup>11</sup> e o Teorema do Sucesso da Empresa. Este último diz: “Sua empresa será bem sucedida se e somente se ela fizer com que os seus interessados críticos para o sucesso sejam vencedores”. Isso significa dizer que o cenário ideal de sucesso é uma situação de ganha-ganha, ou seja, uma situação em que todos os SCSs se percebem vencedores.

O Teorema da Realização do Ganha-Ganha<sup>12</sup> tenta responder como conseguir e manter uma situação de ganha-ganha:

“Fazer dos seus interessados críticos para o sucesso vencedores requer:

1. Identificar todos os interessados críticos para o sucesso (SCSs);
2. Entender como os SCSs querem vencer;
3. Levar os SCSs a negociar um conjunto ganha-ganha de planos de produto e projeto;
4. Controlar o progresso rumo à realização do ganha-ganha dos SCS, incluindo adaptação a mudanças” (BOEHM e JAIN, 2006. Citação traduzida pelo autor).

<sup>11</sup> SCS é o acrônimo para o original em inglês *Success-critical Stakeholder*.

<sup>12</sup> Tradução livre para *Win-Win Achievement Theorem*.

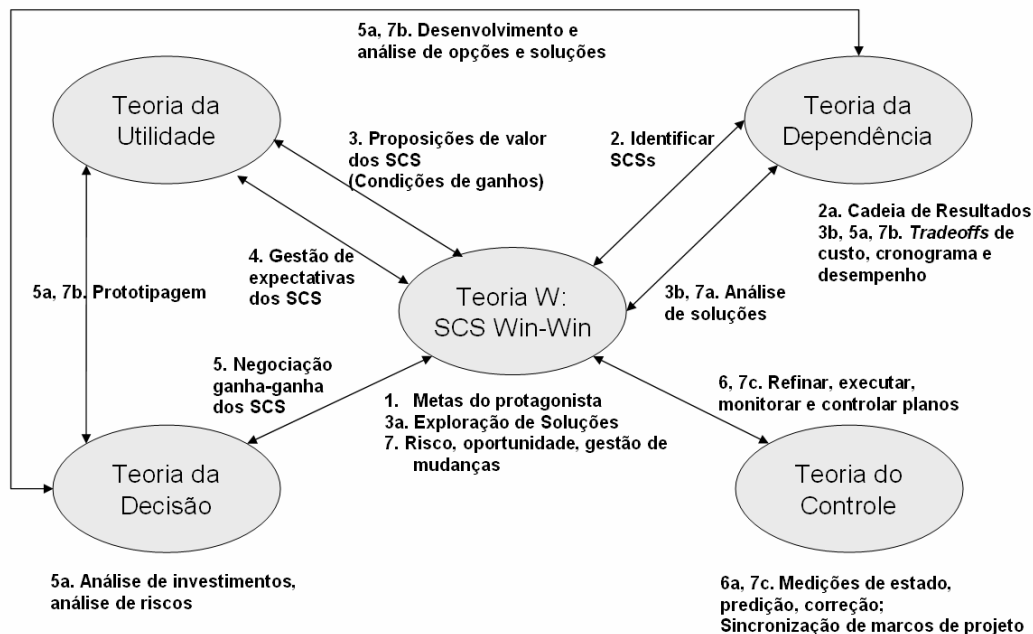


Do Teorema da Realização Ganha-Ganha surgem as outras visões da Figura 2.6:

1. A identificação dos SCSs associa-se à *Teoria da Dependência*, da qual uma técnica-chave é a Cadeia de Resultados, apresentada na Seção 2.7.4 deste trabalho.
2. O entendimento de como os SCSs pretendem vencer ou, em outras palavras, a compreensão das suas funções de utilidade<sup>13</sup>, é assunto estudado pela *Teoria da Utilidade* (DEBREU, 1959). Desta teoria derivam os conceitos de utilidade, já visto na Seção 2.5, e o caráter relativo e até certo ponto subjetivo do conceito de valor, também visto na Seção 2.5.
3. Levar os SCSs a negociar planos ganha-ganha é assunto da *Teoria da Decisão*; esta apresenta muitos aspectos, como a teoria da negociação (FISCHER, 2005), teoria dos jogos, teoria estatística da decisão (BLACKWELL; GIRSHICK, 1954) e teoria das opções reais (ERDOGMUS *et al.*, 2006);
4. Controlar o progresso rumo à realização do ganha-ganha dos SCS é domínio da *Teoria do Controle*. Na VBSE, os princípios da teoria do controle são melhor aplicados ao valor esperado a ser realizado, do que apenas ao progresso do projeto em relação ao planejado. Isso envolve algumas técnicas como: gestão de riscos; mecanismos de controle multicritérios como o *Balanced Scorecard* (KAPLAN; NORTON, 1997); entre outras.

### 2.7.2 Framework de Processo da VBSE

Boehm e Jain (2006) propõem uma versão orientada a processo da Teoria 4+1 da VBSE, mostrada na Figura 2.7. É importante notar que, no passo 1 do fluxo do processo, um *protagonista* ou agente de mudanças fornece o impulso para que se inicie o projeto (ou iniciativa). Dependendo do contexto, protagonistas podem ser líderes organizacionais com autoridade e recursos, empreendedores com metas e recursos, inventores com metas e ideias, ou consórcios (de indivíduos ou organizações) com metas compartilhadas e liderança e recursos distribuídos.



**Figura 2.7** – Teoria da VBSE como um *framework* de processo. Adaptado de (BOEHM; JAIN, 2006)

Os passos de 1 a 7 do fluxo refletem as atividades principais e os passos secundários (2a, 3b e 7c, por exemplo) sugerem ferramentas para realizar as atividades principais. Organizando os passos mostrados numa lista ordenada, obtém-se um esboço de *framework* de atividades genérico, proposto para um processo de software que contempla a criação de valor tal qual defendida pela VBSE:

1. Metas do protagonista (ou agente de mudança)
2. Identificar interessados críticos de sucesso (SCS)
  - 2a. Cadeias de Resultados
3. Entender as proposições de valor dos SCS ("win conditions")
  - 3a. Exploração da solução "possível"
  - 3b. Análise da solução; Trade-offs de custo, cronograma e desempenho
4. Gestão das expectativas dos SCS

<sup>13</sup> Função de utilidade: tradução para o inglês *utility function*, é relacionada ao conceito de utilidade. Significa a forma de colocar em ordem de preferência os benefícios que estão disponíveis a uma pessoa, de acordo com a satisfação trazida por eles.

5. Negociação ganha-ganha com os SCS
  - 5a. Análise de investimento, análise de riscos; prototipação; desenvolvimento e análise de opções e da solução; trade-offs de custo, cronograma e desempenho
6. Refinar, executar, monitorar e controlar planos:
  - 6a. Medições, predição e correção de estado; sincronização de marcos de projeto.
7. Gestão de riscos, oportunidades e mudança:
  - 7a. Análise da solução
  - 7b. Prototipação; Trade-offs de custo, cronograma e desempenho
  - 7c. Medições, predição e correção de estado; sincronização de marcos de projeto.

Nas próximas seções, serão apresentados os elementos-chave, e mapeado um relacionamento entre eles e os passos do *framework* acima.

### **2.7.3 Elementos-chave**

Boehm (2003) apresenta sete elementos-chave que formam a base para a VBSE:

1. Análise de Realização de Benefícios
2. Identificação e Conciliação de Proposições de Valor das Partes Interessadas<sup>14</sup>
3. Análise de Caso de Negócio
4. Gerenciamento Contínuo de Riscos e Oportunidades
5. Engenharia Simultânea de Sistema e de Software
6. Monitoramento e Controle baseados em Valor
7. Mudança como Oportunidade

Esses elementos serão apresentados de forma mais aprofundada nas seções seguintes.

---

<sup>14</sup> Usaremos neste trabalho o termo “parte interessada” como tradução para o inglês *stakeholder*.

## 2.7.4 Análise de Realização de Benefícios

O primeiro elemento-chave proposto pela VBSE deriva da Abordagem de Realização de Benefícios<sup>15</sup> (BRA), um método recente desenvolvido por John Thorp e a DMR Consulting Group (subsidiária da Fujitsu Consulting), e descrito em (THORP, 2007). Seu objetivo é aproximar as ações em busca de benefícios de negócios de uma empresa, das ações e investimentos em TI necessários para alcançar os benefícios esperados.

A BRA foi criada para ser uma nova abordagem para o uso de tecnologia da informação nas organizações de modo a produzir resultados de negócios de forma mais consistente e previsível. Sua proposta é substituir a tendência a ver as mudanças habilitadas pela TI como “balas de prata”, tal como definidas por Brooks (1987). Ao invés disso, as iniciativas de TI passam a ser vistas de forma integrada ao sistema de negócios, atrelando o uso de tecnologia à produção de benefícios de negócios. Segundo Thorp, a BRA é definida por três fundamentos:

- Gerenciamento de programas: as organizações precisam fazer a transição entre a gestão de projetos de software individuais e o gerenciamento de programas consolidados de negócios, incluindo todos os elementos do seu sistema de negócios.
- Gerenciamento de portfólio: as organizações precisam fazer a transição entre a competição desordenada de recursos entre projetos e um gerenciamento disciplinado de portfólio de projetos e programas.
- Governança de ciclo completo: as organizações precisam ampliar o escopo da gestão dos projetos, programas e portfólios, para abranger o ciclo de vida desde a concepção do projeto ou programa à realização dos benefícios de longo prazo.

A análise da realização dos benefícios na BRA tem como principal ferramenta a Cadeia de Resultados<sup>16</sup>, que é um *framework* que vincula as Iniciativas que consomem os recursos da organização, às Contribuições (não em termos dos sistemas entregues, mas de

---

<sup>15</sup> Tradução do original em inglês *Benefits Realization Approach*.

<sup>16</sup> Tradução do original em inglês *Result Chain*.

seus efeitos nas operações da empresa) e aos Resultados: estes podem levar a outras contribuições ou a valor agregado ao negócio. A Cadeia de Resultados também vincula os Resultados a Premissas, que condicionam a realização dos Resultados. Em projetos de software ou de TI, a Cadeia de Resultados pode ser utilizada para identificar:

- Outras iniciativas, além do desenvolvimento de software, necessárias para a realização dos benefícios do projeto. O conjunto de iniciativas resultante poderá ser gerido como um programa consolidado de investimentos, tal como definido na Seção 2.5.2;
- Partes interessadas adicionais, críticas para o sucesso do projeto, que precisam estar representadas no processo de construção e aprovação da visão compartilhada do produto final ou sistema.

A Figura 2.8 mostra um exemplo de uma Cadeia de Resultados simples, referente a um projeto de implantação de um sistema de entrada de pedidos, na qual o tempo desde o pedido até a entrega é a premissa que vincula as contribuições da iniciativa tomada ao resultado esperado, de forma que, em não sendo esta premissa verdadeira, toda a cadeia de resultados seria comprometida.

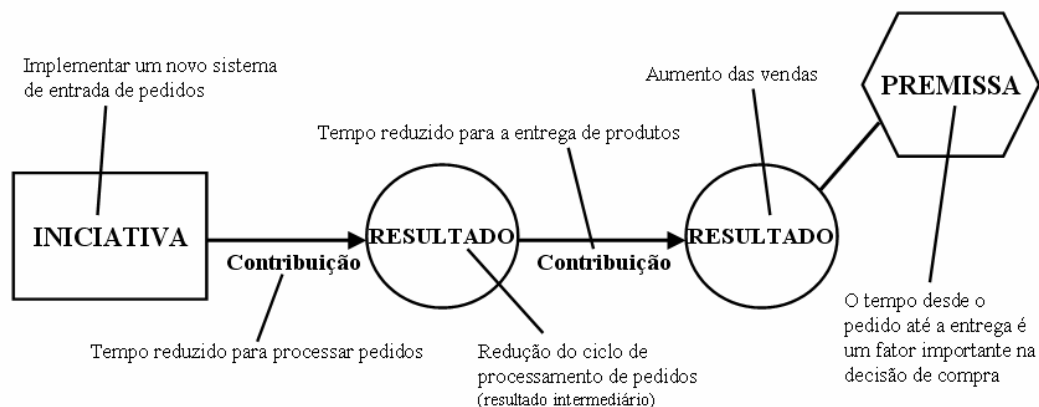


Figura 2.8 – Cadeia de Resultados. Adaptado de (BOEHM, 2003)

### 2.7.5 Identificação e Conciliação de Proposições de Valor dos Interessados

Identificadas as iniciativas (envolvendo TI ou não) que, baseadas nas premissas

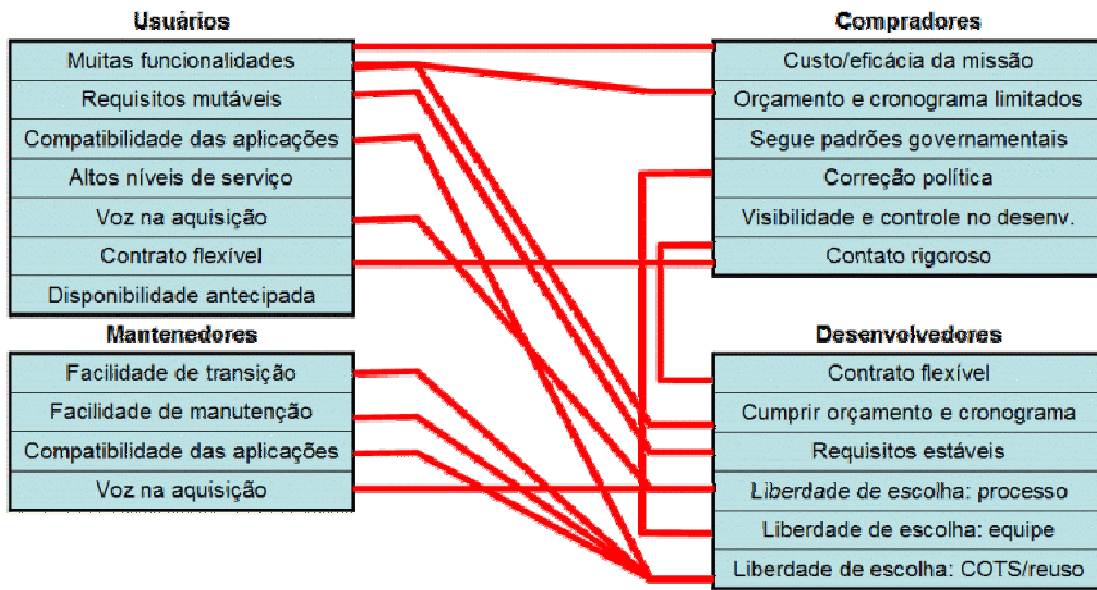
adotadas, levarão aos resultados esperados, é conveniente que os interessados expressem suas proposições de valor quanto a tais soluções, para ajudar a definir melhor quais serão seus objetivos. Nesse caso, é preciso também que tais proposições sejam compatíveis entre si. Uma das maneiras de se identificar incompatibilidades entre as proposições de valor dos interessados, de forma a permitir uma conciliação, é através da elaboração do diagrama de conflito de modelos<sup>17</sup>, mostrado na Figura 2.6. Neste tipo de diagrama, proposto por Boehm *et al.* (2000), identificamos as proposições de valor de cada grupo crítico de interessados e, confrontando com as proposições dos outros grupos, identificamos possíveis conflitos, mostrados graficamente como vínculos entre as proposições conflitantes.

A VBSE sugere diversas abordagens para conciliação das proposições de valor dos interessados:

- Gerenciamento de expectativas: gerenciar expectativas significa tornar os principais interessados cientes dos potenciais conflitos entre as proposições de valor uns dos outros. Mesmo a simples divulgação desses conflitos, para que os envolvidos saibam o esforço necessário para resolvê-los, pode ajudar no relaxamento das exigências menos críticas, e muitas vezes mais custosas, dos interessados.

---

<sup>17</sup> Tradução do original em inglês *model clash diagram*.



**Figura 2.9** – Diagrama de conflito de modelos. Adaptado de Boehm (2003)

- Técnicas de visualização e análise de *trade-off*: Protótipos, cenários e modelos de estimativas ajudam os interessados a ter entendimento mútuo dos aspectos que cada um considera mais importantes ou mais exequíveis em um sistema.
- Priorização: quando os interessados categorizam ou classificam por prioridade suas funcionalidades desejadas, temos condições de determinar a combinação ótima de funcionalidades exequíveis com os recursos existentes, os quais muitas vezes são limitados ou apresentam restrições de uso.
- Groupware: são as ferramentas de colaboração que apoiam *brainstorming*, discussões e negociações de situações de conflito, inclusive no que se refere às proposições de valor.
- Análise de Caso de Negócio: um *Caso de Negócio* pode ser definido como um documento que descreve a justificativa para o início ou a continuação de um projeto, fornecendo as razões para a execução de um novo projeto ou da próxima fase de um projeto em andamento (OGC, 2002). Com este documento é possível determinar quais funcionalidades trazem o melhor retorno sobre o investimento, permitindo melhor conciliar as proposições de valor dos interessados. Na próxima Seção será discutida de forma detalhada a Análise do Caso de Negócio.

### **2.7.6 Análise de Caso de Negócio**

Uma definição simples para Análise de Caso de Negócio é: determinar custos financeiros, benefícios, e retorno sobre o investimento (ROI) de todo o ciclo de vida de um sistema. Tal análise é feita para que possamos comparar duas ou mais opções – entre iniciativas de TI a serem implantadas, ou entre componentes de software a serem adquiridos, por exemplo – de forma a decidirmos escolher a opção que traga mais valor ao negócio.

A VBSE explora um conceito de análise de Caso de Negócio mais abrangente, envolvendo benefícios intangíveis e riscos. Muitas vezes, é preferível uma opção com ROI de longo prazo, sobre outra cujo retorno se dê em prazo mais curto, por envolver também benefícios adicionais difíceis de quantificar, tais como maior facilidade de envolvimento dos interessados, ou algum tipo de benefício político. Também pode acontecer de não ser facilmente determinado o potencial de retorno das opções, o que introduz o fator risco à análise econômica. Nesse caso, pode haver valor em se conduzir uma análise aprofundada do caso de negócio, ou seja, investir na aquisição de informação de forma a reduzir riscos.

### **2.7.7 Gerenciamento Contínuo de Riscos e Oportunidades**

O PMI define risco de um projeto da seguinte forma:

“O risco de um projeto é um evento ou condição incerta que, se ocorrer, terá um efeito positivo ou negativo sobre pelo menos um objetivo do projeto, como tempo, custo, escopo ou qualidade (ou seja, em que o objetivo de tempo do projeto é a entrega de acordo com o cronograma acordado; em que o objetivo de custo do projeto é a entrega de acordo com o custo acordado, etc.)” (PMI, 2004)

Ainda segundo o PMI, as organizações percebem os riscos quando eles estão relacionados a ameaças ao sucesso do projeto (riscos negativos) ou a oportunidades para aumentar as chances de sucesso do projeto (riscos positivos).

Na visão da VBSE, o gerenciamento de ameaças e oportunidades não deve se limitar ao momento da definição do caso de negócio do projeto, e das avaliações dos marcos de controle. Em vez disso, deve permear todo o ciclo de vida da solução. A análise de risco deve sempre introduzir o “fator gente” nas tomadas de decisão de caráter econômico em um projeto, dado que a aversão ao risco que as pessoas têm determinam em grande parte as funções de utilidade que elas atribuem às funcionalidades cogitadas para o software. As funções de utilidade são fatores importantes para tomar decisões críticas, e modificá-las de



acordo com a conjuntura.

O estudo das funções de utilidade passa por analisar as opções de benefícios (proposições de valor) envolvidas nas decisões de projeto críticas e avaliar quais são as preferências dos envolvidos (as partes interessadas) em relação a elas. Um exemplo: um projeto de software tem como possíveis opções de implementação o reuso de componentes, cuja duração foi estimada entre duas e seis semanas, e a codificação de funcionalidades equivalentes pela própria equipe, com duração estimada em quatro semanas. Os implementadores podem atribuir uma utilidade maior à opção de codificar, pois ela traz menos incertezas quanto à duração e ao esforço esperados das atividades envolvidas. Por outro lado, o gerente do projeto pode atribuir maior utilidade à opção de reuso, pois ela tem uma probabilidade real de durar duas semanas, configurando um risco positivo de avanço no cronograma, e consequentes economias de esforço e custo. Para isso, o gerente pode tomar providências para maximizar o risco de a atividade durar apenas duas semanas.

A VBSE propõe o uso do conceito de *exposição ao risco* como balizador do tipo de adaptação (*tailoring*) a ser feito na metodologia de desenvolvimento de software a ser utilizada por uma equipe. A exposição ao risco (ER) relacionada a uma determinada ação é obtida avaliando-se a probabilidade de uma perda  $P(p)$  associada à ação e o correspondente tamanho da perda  $T(p)$ , e calculando-se o produto de ambos:  $ER = P(p) * T(p)$ . No âmbito da VBSE, entende-se por “perda” uma redução de lucros, reputação, qualidade de vida ou outros atributos relacionados a valor; no caso de oportunidades, a entidade “perda” é substituída por um ganho de valor, refletido nos mesmos atributos.

No que se refere ao *tailoring* dos métodos de desenvolvimento, a exposição aos riscos de um planejamento inadequado é maior quanto maior for a “agilidade” do processo adotado. Por outro lado, em um processo muito “disciplinado”, ou “burocrático”, é maior a exposição ao risco de erosão do *market share* devido a atrasos no lançamento do produto no mercado.

### **2.7.8 Engenharia Simultânea de Sistema e de Software**

O CMMI assim define engenharia de sistemas:

“... é a abordagem interdisciplinar que governa o esforço técnico e gerencial total requerido para transformar um conjunto de necessidades, expectativas e restrições dos clientes em uma solução de produto, e apoiar essa solução ao longo da vida do produto” (SEI, 2006 p.557).

O mesmo CMMI define a Engenharia de Software como “a aplicação de uma abordagem sistemática, disciplinada e quantificável ao desenvolvimento, operação e manutenção de software” (p.555). Segundo Pressman (2006 p.99), a engenharia de sistemas, no ambiente dos sistemas de apoio aos negócios de uma empresa, toma a forma de *engenharia de processo de negócios*.

A conjuntura atual dos mercados leva as organizações a aumentar sua agilidade no desenvolvimento de software, ao mesmo tempo em que seus produtos e serviços tornam-se mais e mais integrados a software. A VBSE afirma que, para atender a essas exigências, é preferível que os engenheiros de sistema e os engenheiros de software desenvolvam de forma concomitante os conceitos, requisitos, arquitetura, planos de ciclo de vida e partes importantes do código. Tal prática, denominada *engenharia simultânea*, visa reduzir o tempo de desenvolvimento de um produto através da simultaneidade das atividades, ao mesmo tempo em que exige melhor planejamento, controles e comunicação no projeto, com o objetivo de reduzir riscos associados à execução de tarefas em paralelo (KERZNER, 2006).

Uma característica que a VBSE prega ser essencial à simultaneidade de atividades de sistemas e software é a definição de marcos de projeto que possuem critérios de avaliação da consistência entre os diversos artefatos produzidos, bem como da sua viabilidade. Essa característica é verificada naturalmente nas metodologias e processos de software evolutivos, incrementais, ou do tipo espiral. Um exemplo é o modelo de maturidade CMMI-DEV (SEI, 2006), que nasceu da junção dos modelos CMMI-SW (centrado em desenvolvimento de software) e CMMI-SE (para engenharia de sistemas).

### **2.7.9 Monitoramento e Controle Baseados em Valor**

Uma técnica frequentemente usada para monitorar e controlar projetos de software é o *Earned Value Management* (EVM). Na Engenharia de Software, o EVM pode ser integrado, por exemplo, aos projetos de implantação do modelo CMMI (SOLOMON, 2002).

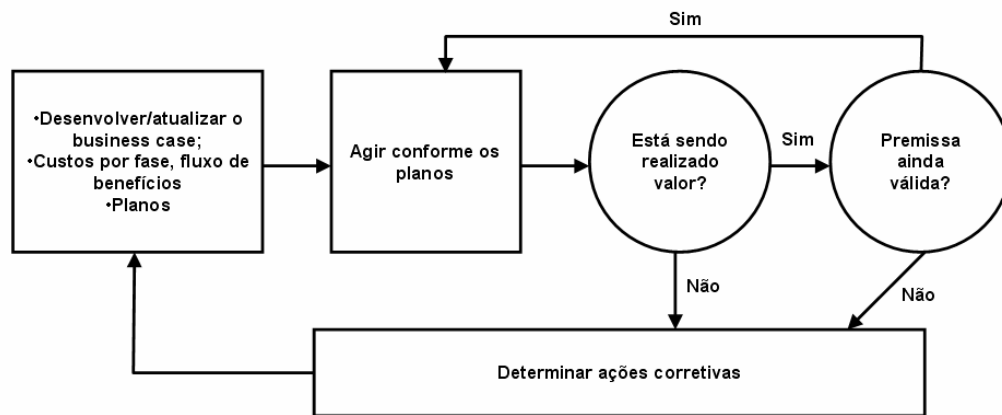
Embora a técnica EVM seja eficaz para acompanhar se o projeto está seguindo o plano original, ela tem algumas desvantagens:

- Torna-se de difícil aplicação se o plano do projeto sofre muitas ou frequentes alterações;

- Não nos diz nada sobre o real valor que está sendo criado pelo projeto – um projeto pode ser extremamente bem sucedido em relação ao valor agregado comparado ao custo, mas um total desastre em termos do valor agregado à organização.

Sendo assim, tornam-se necessárias técnicas que apoiem monitoramento e o controle do real valor a ser agregado pelos resultados do projeto. A primeira delas é o uso do *caso de negócio* (discutido na Seção 2.7.6) como forma de monitorar o real valor das funcionalidades que serão entregues; para isto, o caso de negócio precisa ser atualizado para refletir as mudanças – nas premissas do negócio, no mercado, na organização, no progresso do desenvolvimento do sistema – que aconteçam. Dado que é difícil monitorar o valor criado por funcionalidades ainda não entregues, esta abordagem funciona melhor quando o projeto é gerenciado de forma a produzir incrementos frequentes de funcionalidades.

Outra técnica é o monitoramento de premissas e de progresso com respeito a todas as Iniciativas e Resultados envolvidos na Cadeia de Resultados do projeto (discutida na Seção 2.7.4). Um processo de *feedback* de realização de valor, baseado no EVM com a introdução dos Resultados e Premissas identificados na Cadeia de Resultados, é mostrado na Figura 2.6.



**Figura 2.10** – Processo de Feedback de Realização de Valor. Adaptado de (BOEHM, 2003)

Enquanto as técnicas acima focam o controle em nível de projeto, outras abordagens são necessárias para o monitoramento em nível organizacional. A VBSE aponta a *Experience Factory* (EF) (BASILI *et al.*, 1994) e, associada a esta, a *Goal-Question-Metric* (GQM) (BASILI *et al.*, 1994a) como possíveis abordagens de monitoramento baseadas em valor. A EF estabelece uma infraestrutura para a capitalização e reuso das experiências e dos

produtos de um ciclo de vida de projeto, permitindo que a equipe de gestão de portfólio de projetos tenha bases para comparar dados de desempenho e objetivos de diversos projetos. A GQM estabelece uma hierarquia que forma um sistema de medição: as metas definidas (*goals*) determinam questões (*questions*) que devem ser respondidas para avaliação do cumprimento dos objetivos. Para cada questão é associada uma métrica (*metric*), que é um conjunto de dados que responde à métrica de forma quantitativa.

Uma outra abordagem de monitoramento e controle de caráter multidimensional, isto é, que trabalha tanto no nível dos projetos individuais, como no nível dos portfólios de projetos e também na organização como um todo, é o *Balanced Scorecard* (KAPLAN; NORTON, 1997), que organiza as metas, as estratégias e iniciativas dentro de quatro perspectivas: financeira, clientes, processos internos de negócio, e aprendizado e crescimento.

### **2.7.10 Mudança como Oportunidade**

Atualmente, mudanças acontecem continuamente nas tecnologias, no mercado, nas organizações, e conseqüentemente nas prioridades e proposições de valor das partes interessadas dos projetos; além disso, a velocidade das mudanças é crescente. Devido a isso, é cada vez mais verdade que as organizações que conseguem se adaptar às mudanças mais rápido que os concorrentes terão mais sucesso no mercado. Deste fato, pode-se concluir que a capacidade de se adaptar às mudanças tem valor de negócio. Se considerarmos software como a principal tecnologia para adaptação às mudanças, tem-se que o conceito de mudança como oportunidade de negócios é fundamental, a ponto de ter sido eleito como um elemento-chave na VBSE.

A discussão deste elemento-chave identifica dois tipos de técnicas para aumentar a adaptabilidade de um projeto de desenvolvimento de software às mudanças. Tais tipos são:

- *Técnicas centradas em arquitetura*, que têm dois focos principais:
  - A identificação das fontes mais prováveis de mudança ou evolução de requisitos, de forma a analisar seu potencial de riscos negativos ou positivos, e assim planejar as ações para lidar com os riscos;
  - O uso de modularização para isolar as fontes de mudança dentro de módulos arquiteturais. Tal isolamento visa limitar o tamanho do impacto em termos de

retrabalho em componentes já desenvolvidos caso uma mudança seja necessária, otimizando os fatores custo e tempo de desenvolvimento;

- *Técnicas centradas em refatoração*<sup>18</sup>, que focam em manter o produto o mais simples possível, e em reorganizar o produto por demanda, para acomodar o próximo conjunto de mudanças.

Segundo Boehm (2003), enquanto os métodos ágeis de desenvolvimento baseiam-se em refatoração para acomodar mudanças, os processos orientados a planejamento utilizam as técnicas centradas arquitetura.

### **2.7.11 Relacionamento entre os Elementos-chave e o *Framework* de Processo da VBSE**

Como primeiro passo para a identificação de pontos de convergência entre os elementos-chave da VBSE e um framework de processo específico (o RUP), buscou-se primeiro definir o relacionamento entre os elementos-chave e o *framework* genérico de processo da VBSE, com o objetivo de identificar quais elementos-chave são envolvidos na execução de cada passo. O mapeamento resultante desta atividade, mostrado na Tabela 2.1, ajuda a compreender quais os objetivos de cada elemento-chave, em termos das atividades essenciais em um processo genérico de desenvolvimento de software. A partir do mapeamento com o *framework* genérico, torna-se mais fácil localizar quais elementos de conteúdo de um *framework* específico (o RUP) mais se aproximam de atender aos elementos-chave da VBSE.

## **2.8 Discussão**

Neste Capítulo foram apresentadas as bases teóricas que nortearão a análise e a proposta de extensão definidas neste trabalho: processos de software; gestão de projetos de software; estratégia de projetos de software, uma introdução ao conceito de valor; Economia de Software (uma disciplina que une os assuntos acima); e VBSE, proposta de Barry Boehm para ênfase na criação de valor na Engenharia de Software.

---

<sup>18</sup> Aqui traduzimos o termo em inglês *refactoring*.

**Tabela 2.1** - Mapeamento dos elementos-chave da VBSE com o *framework* de processo  
(resultado obtido pelo autor)

<i>Elemento-chave</i>	<i>Passos do Framework de Processo realizados</i>
Análise de Realização de Benefícios	1. Metas do protagonista 2. Identificar SCSs 2a. Cadeia de Resultados
Identificação e Conciliação de Proposições de Valor dos Interessados	3. Proposições de valor dos SCSs 3a. Exploração da solução “possível” 4. Gestão de expectativas dos SCSs 5. Negociação ganha-ganha com os SCSs 5a. Prototipação
Análise de Caso de Negócio	5. Negociação ganha-ganha com os SCSs 5a. Análise de investimento; trade-offs de custo, cronograma e desempenho; desenvolvimento e análise de opções e da solução
Gerenciamento Contínuo de Riscos e Oportunidades	7. Gestão de riscos, oportunidades e mudança
Engenharia Simultânea de Sistema e de Software	7. Gestão de riscos, oportunidades e mudança 7c. Sincronização de marcos de projeto
Monitoramento e Controle baseados em Valor	6. Refinar, executar, monitorar e controlar planos
Mudança como Oportunidade	7. Gestão de riscos, oportunidades e mudança

Os processos de software, como ferramentas de qualidade para o desenvolvimento, desempenham papel bastante importante na Engenharia de Software. Com eles as equipes de desenvolvimento podem estabelecer melhor controle nos diversos aspectos do desenvolvimento, permitindo obter de forma consistente os resultados esperados. Pode-se classificar os processos de software em disciplinados e ágeis. Enquanto os últimos concentram-se em adaptabilidade às mudanças (de requisitos, prazo, qualidade) do projeto,

enfatizando a produção de código e testes de código, os primeiros têm uma abordagem mais orientada ao planejamento dos projetos, com forte ênfase em documentação. Dentre os processos disciplinados, o Rational Unified Process (RUP) foi descrito em maiores detalhes neste capítulo.

A gestão de projetos de software tem peculiaridades que diminuem o risco de sucesso dos projetos, se gerenciados usando-se práticas tradicionais de gestão de projetos. Práticas e métodos específicos se fazem necessários, e os processos de software mais conhecidos contemplam opções de gestão dos projetos de software.

O alinhamento estratégico de projetos, que visa alinhar os objetivos estratégicos das organizações às iniciativas operacionais (projetos) que implementam tais objetivos, é realizado através do gerenciamento de programas, portfólios e pela governança de projetos.

Dentro da Engenharia de Software, a Economia de Software estuda a aplicação de critérios econômicos em todos os níveis de tomada de decisão no desenvolvimento de software. Uma proposta de Barry Boehm, a Engenharia de Software Baseada em Valor (VBSE), pretende incorporar a ênfase na criação de valor econômico aos métodos, práticas da Engenharia de Software, a partir da adoção de sete elementos-chave para a criação de valor.

Tendo em vista as considerações acima, este trabalho se propõe a desenvolver uma análise do quanto o RUP enfatiza a criação de valor nos projetos onde é aplicado, tomando como referência para esta análise os princípios, métodos e práticas sugeridos na definição dos elementos-chave da VBSE. Esta análise será descrita em maiores detalhes no próximo capítulo.

## 3 Criação de Valor no RUP

Para o objetivo desta dissertação, o suporte dado por um processo de software “de mercado” à criação de valor será analisada do ponto de vista das suas atividades, práticas e princípios. Para tanto, foi escolhido o Rational Unified Process (RUP).

Nas seções seguintes apresentaremos uma análise do RUP perante a Engenharia de Software Baseada em Valor (VBSE), e uma proposta de extensão do RUP relativa à criação de valor.

### 3.1 O RUP e a Criação de Valor

Vale a pena investigar que noção de valor está embutida no RUP, a partir do que é encontrado nas definições e conceitos básicos deste processo.

É pertinente iniciar a investigação da noção de valor embutida no RUP pela declaração dos princípios-chave do processo. Estes representam boas práticas que visam o sucesso no desenvolvimento de software (IBM, 2006). Os princípios-chave são:

- Adaptar o Processo: ressalta a necessidade de dimensionar corretamente o processo de desenvolvimento para as necessidades do projeto.
- Equilibrar Prioridades das Partes Interessadas em Competição: reforça a importância de equilibrar as necessidades, algumas conflitantes entre si, das partes interessadas no sistema.
- Colaborar Através de Equipes: recomenda o fomento à efetiva colaboração entre os membros das equipes de desenvolvimento e de negócios.
- Demonstrar Valor Iterativamente: enfatiza os benefícios de desenvolver software em ciclos de desenvolvimento repetidos, de modo a acomodar mudanças e reduzir riscos.
- Elevar Nível de Abstração: descreve como reduzir a complexidade elevando o nível de abstração.



- Focalizar Continuamente na Qualidade: enfatiza que, para obter qualidade, é preciso um acompanhamento por todo o ciclo de vida do processo.

Na declaração de cada princípio-chave são mencionados os benefícios que cada um dos princípios-chave pode trazer ao desenvolvimento de software. Dentre os principais benefícios listados, destacam-se:

- Planos e estimativas mais realistas (associados ao princípio-chave “Adaptar o processo”);
- Maior produtividade das equipes de desenvolvimento (associado aos princípios-chave “Trabalhar em conjunto com equipes” e “Elevar o nível de abstração”);
- Produtos com maior qualidade (associa-se a “Focalizar continuamente na qualidade”);
- Otimização do valor de negócios do software (associa-se a “Equilibrar prioridades das partes interessadas em competição”);
- Alinhamento das aplicações com as necessidades de negócio e dos usuários (associado a “Equilibrar prioridades das partes interessadas em competição”).

Pela lista acima se percebe que, além dos benefícios tradicionalmente associados aos processos de software – maior qualidade na gestão dos projetos de software e em seus produtos – o RUP preocupa-se com outros benefícios, como valor de negócios e alinhamento com o negócio.

Outro ponto a investigar é a disciplina de Requisitos, que possui, dentre suas finalidades, estabelecer e manter acordo entre clientes e partes interessadas a respeito do que o sistema deverá fazer (IBM, 2006). Esta preocupação com a definição pactuada do produto de software gerado com o uso do RUP é essencial para a determinação do valor criado pelo produto.

Ainda na disciplina de Requisitos, um ponto a discutir é a associação do conceito de valor ao conceito de caso de uso. Segundo Kruchten (2003), no RUP o caso de uso é um agrupamento sequenciado de ações que um sistema executa para fornecer resultados que tenham valor ao ator do caso de uso. Dessa definição conclui-se que a criação de

valor no RUP também deriva do levantamento de requisitos e da definição de casos de uso eficazes em capturar o que é valor para o cliente em relação ao software a ser construído.

### **3.2 Convergência entre RUP e VBSE quanto a Criação de Valor**

Esta seção apresenta pontos de convergência identificados entre o RUP e os elementos-chave da VBSE. O critério aplicado para determinar se há convergência é a existência de tarefas, práticas ou orientações no RUP padrão que sejam previstas na VBSE. A intenção é mostrar quais elementos-chave são atendidos, pelo menos em parte, pelo conteúdo presente na configuração-padrão do RUP. Neste trabalho, dar-se-á o nome de *ponto de convergência* a cada prática, técnica ou ferramenta sugerida pela VBSE e identificada como presente no RUP. O conteúdo do RUP utilizado para esta análise é a presente na *Method Library* versão 7.0.1 em português (IBM, 2006).

Para identificar os pontos nos quais o RUP se alinha com (isto é, converge para) um determinado elemento-chave, foi utilizado o seguinte procedimento:

1. A partir do mapeamento delineado na Seção 2.7.11 e apresentado na Tabela 2.1, selecionou-se um elemento-chave, e os passos do framework de processo genérico relacionados;
2. A seguir, foram identificados os fluxos de trabalho e tarefas do RUP que realizam os passos encontrados no passo 1 acima;
3. Para cada fluxo ou tarefa identificado, seu conteúdo foi analisado, buscando-se a implementação ou sugestão de uso de algum dos princípios, práticas ou técnicas sugeridas por Boehm (2003). Caso a análise encontre pelo menos uma implementação, considerou-se que o RUP é *convergente* à VBSE para efeito da análise desta dissertação.
4. Para cada prática ou técnica encontrada no RUP no passo anterior, um ponto de convergência foi criado.
5. Para cada elemento-chave, repetem-se os passos acima.

As próximas seções discorrerão sobre os pontos de convergência identificados na

pesquisa.

### 3.2.1 Análise de Realização de Benefícios

O primeiro elemento-chave a analisar no RUP quanto à convergência à VBSE é a Análise de Realização de Benefícios que, como visto no Capítulo 2, procura aproximar as iniciativas de TI da organização de outras ações e projetos que tenham os mesmos objetivos de negócio. Da definição do elemento-chave conclui-se que:

- O escopo da Análise de Realização de Benefícios vai além do escopo de projetos individuais, abrangendo em programas a coordenação de diversos projetos com objetivos relacionados;
- A técnica terá eficácia se o conjunto das iniciativas analisadas, juntamente com outras conduzidas pela organização relativas a outros objetivos estratégicos, for gerenciado de forma a utilizar os recursos destinados aos projetos da forma mais eficiente e alinhada à estratégia possível.

O foco do RUP é disciplinar o esforço em torno de um único projeto de software, durante o qual a equipe de desenvolvimento interage com outros agentes do contexto organizacional. Para que o RUP fosse aderente à Análise de Realização de Benefícios, deveria abrigar, dentre seus fluxos de trabalho, a análise de diversos projetos interrelacionados pelos mesmos objetivos de negócio, de forma a mapear clara e objetivamente os benefícios trazidos por tais projetos. Como tal característica é ausente do RUP, neste aspecto, portanto, o *framework* de processo não provê convergência à VBSE.

### 3.2.2 Identificação e Conciliação de Proposições de Valor das Partes Interessadas

Como visto na Seção 2.7.5, um dos princípios-chave do RUP é “Equilibrar prioridades das partes interessadas em competição”, o que significa que o processo dá importância ao equilíbrio das necessidades, frequentemente conflitantes, dos envolvidos nos projetos.

O RUP contempla, dentro da disciplina de Requisitos, uma tarefa denominada *Identificar Pedidos dos Envolvidos*, cujos objetivos são:

- Compreender quem são as partes interessadas do projeto;

- Coletar demandas sobre quais necessidades o sistema deve atender;
- Priorizar demandas das partes interessadas.

Ao final da tarefa, a equipe de desenvolvimento deverá obter, entre outros artefatos, uma lista de Demandas dos Interessados, contendo todas as necessidades e expectativas dos interessados para com o sistema, bem como quais dessas demandas serão atacadas durante o desenvolvimento.

Tais objetivos gerais e produtos finais se alinham aos princípios do elemento-chave de Identificação e Conciliação de Proposições de Valor. Apesar disto, será visto mais adiante que a tarefa carece de definições de ferramentas e práticas para realizar tais objetivos.

#### *Ponto de Convergência nº 1: Técnicas de Visualização e Análise de Trade-off*

Como visto no Capítulo 2, dentre as diversas abordagens que a VBSE propõe como ferramentas para a conciliação das proposições de valor entre as partes interessadas do projeto estão as técnicas de visualização e análise de trade-off, tais como prototipação e *storyboarding*.

As proposições de valor das partes interessadas são extraídas das suas demandas. Como visto logo acima, no RUP estas demandas devem ser levantadas executando-se a tarefa *Identificar Pedidos dos Envolvidos*. Ao longo da tarefa, o RUP recomenda aplicar as seguintes técnicas:

- *Storyboarding*: técnica para captura das descrições lógica e conceitual de uma funcionalidade de sistema para um cenário específico. Em geral significa o uso de alguma ferramenta para ilustrar aos usuários (às vezes usando animação) como o sistema irá se adequar à organização, e indicar o comportamento do sistema.
- Protótipo: é uma versão inicial reduzida de uma solução de sistema ou de uma porção de tal solução. É construído em curto espaço de tempo e evoluído durante algumas iterações, e serve para testar e avaliar a eficácia geral do design utilizado para resolver o problema em questão. O uso de protótipos na identificação de requisitos pode reduzir as incertezas quanto à compreensão dos requisitos pelos usuários e outras partes interessadas.

A convergência do RUP ao elemento-chave dá-se, portanto, pela aplicação das técnicas acima de forma intrínseca à definição da tarefa.

### **3.2.3 Análise de Caso de Negócio**

Outra ferramenta indicada pela VBSE é a análise de Caso de Negócio. Trata-se da justificativa econômica do projeto.

#### *Ponto de Convergência nº 2: Desenvolvimento do Caso de Negócio*

Um dos elementos essenciais do RUP é o caso de negócio. Segundo a definição desse elemento, "... [o caso de negócios] fornece informações para os tomadores de decisão econômica sobre o valor do projeto e é utilizado para determinar se o projeto deve continuar" (IBM, 2006).

Outra evidência da convergência com a VBSE é a definição, na disciplina de Gerenciamento de Projetos, da tarefa *Desenvolver Caso de Negócio*, cujo objetivo é identificar os aspectos de produto, de negócios e financeiro fundamentais do desenvolvimento de um novo produto, bem como suas maiores restrições e opções a considerar.

### **3.2.4 Gerenciamento Contínuo de Riscos e Oportunidades**

Para o RUP, o gerenciamento de riscos é fator crítico para o sucesso de um projeto de software, sendo um dos seus elementos essenciais. O elemento defende que sejam atacados logo no início do projeto os riscos mais graves identificados no planejamento.

O RUP tem, na sua disciplina de Gerenciamento de Projetos, diversas tarefas e artefatos que visam atender ao princípio-chave citado.

#### *Ponto de Convergência nº 3: Gerenciamento de Riscos do Projeto*

Na disciplina de Gerenciamento de Projeto, as principais tarefas do gerenciamento de riscos estão contempladas:

- *Desenvolver Plano de Gerenciamento de Riscos*, onde se planeja de que forma serão tratados os riscos durante o projeto, possibilitando que se estabeleça um gerenciamento contínuo;

- *Identificar e Avaliar Riscos* trata da identificação e avaliação dos riscos durante o planejamento das iterações;
- *Monitorar Status do Projeto* cuida do monitoramento dos riscos e dos problemas que podem causar sua realização.

### **3.2.5 Engenharia Simultânea de Software e de Sistema**

A VBSE aponta a importância do desenvolvimento concomitante dos vários aspectos dos produtos ou serviços que contêm outros componentes além de software. O RUP apresenta pontos em comum com este elemento-chave da VBSE com relação a dois aspectos principais:

- É um processo iterativo e incremental, como mencionado no princípio-chave do RUP “Demonstrar o valor iterativamente”. Isto significa que o RUP possui marcos definidos de iteração e de projeto, e tarefas para avaliar, nesses marcos, se o projeto está entregando valor;
- Dá suporte à engenharia de sistemas, em especial à modelagem de negócios de sistemas de informação, através da disciplina de Modelagem de Negócios.

Os pontos de convergência acima serão detalhados a seguir. Como nota adicional, o RUP vem sendo utilizado como uma das bases para definição de processos de desenvolvimento na engenharia de sistemas, em áreas como o design de componentes eletrônicos digitais de alta integração. Um exemplo é o processo ipPROCESS (LIMA *et al.*, 2005), definido para desenvolvimento de blocos de Intellectual Property (IP-cores). Iniciativas desse tipo permitem uma melhor integração no desenvolvimento de soluções com software e hardware.

#### ***Ponto de Convergência nº 4: Marcos de Projeto bem definidos***

Para que um processo de software atenda ao elemento-chave, a Engenharia de Sistemas e a Engenharia de Software dos produtos elaborados seguindo o processo devem atuar de forma concomitante e convergente, ajudando a criar valor ao produto. Para tanto, *pontos de controle* devem ser definidos para que, periodicamente, os artefatos de software e de sistema (ou de processo de negócio) produzidos sejam avaliados, segundo critérios de consistência e viabilidade.

Como visto no Capítulo 2, o RUP define marcos de projeto entre as suas fases e, dentro destas, também são definidos marcos ao final de cada iteração. O objetivo dos marcos no processo é avaliar se o desenvolvimento está alcançando os objetivos estabelecidos no planejamento do projeto.

Na disciplina Gerenciamento de Projetos, há a tarefa *Revisão de Marco de Ciclo de Vida*, na qual é executado o processo de revisão do estado do projeto ao final de cada fase, de forma a determinar se o projeto está apto a passar para a próxima fase ou se ainda há pendências. Segundo a descrição da tarefa, ela deve marcar um ponto no qual as expectativas técnicas e de gestão do projeto devem ser ressincronizadas. Ressalte-se que as questões a serem consideradas devem relacionar-se principalmente à gestão do projeto: no caso de um processo com ênfase à criação de valor, as questões relacionadas a esse tema deverão ser avaliadas, e suas soluções encaminhadas.

Os marcos de final de iteração cumprem papel semelhante ao dos marcos de projeto, apenas com um escopo mais limitado, o escopo da iteração. A tarefa *Revisão de Aceitação da Iteração*, executada ao final de cada iteração, tem a finalidade de revisar e chegar a um acordo sobre se a iteração cumpriu seus objetivos.

Do exposto acima, conclui-se que o RUP provê os pontos de controle e suas respectivas tarefas de revisão, necessários para a simultaneidade entre as tarefas de Engenharia de Sistemas e de Engenharia de Software de um projeto de desenvolvimento. Daí a razão para identificar isto como ponto de convergência do RUP em relação ao elemento-chave objeto desta Seção.

#### *Ponto de Convergência nº 5: Integração da Modelagem de Negócios ao Processo de Software*

Os sistemas de informação que apoiam as tomadas de decisões e coordenam os processos de negócios das organizações contemporâneas são fortemente apoiados por software (LAUDON; LAUDON, 2007). As iniciativas de aperfeiçoamento de processos de negócios dessas organizações envolvem projetos de desenvolvimento ou evolução do software que apoia seus sistemas de informação. Tem-se, nesse cenário, que a “engenharia de sistemas” de tais iniciativas envolve a modelagem e reengenharia dos processos de negócio. Segundo a VBSE, esta engenharia de sistemas deve evoluir de forma concomitante com a

engenharia de software empregada no desenvolvimento.

Analisando o RUP sob a perspectiva apresentada acima, vê-se que uma forma de convergência entre o RUP e a VBSE quanto ao elemento-chave mencionado, neste caso específica para sistemas de informação, é a própria presença de uma disciplina de Modelagem de Negócios. A descrição dessa disciplina no RUP diz:

“Os sistemas de informações devem ser projetados para assegurar que as informações fornecidas sejam oportunas, exatas, suficientes e relevantes. Podemos assegurar que os sistemas de informações suportem decisões de negócios dessa maneira somente se entendermos o contexto no qual essas decisões são tomadas” (IBM, 2006 – “Disciplina: Modelagem de Negócios”).

Sendo incorporada como disciplina integrante do *framework* de processo, a Modelagem de Negócios submete-se, no âmbito de um projeto de desenvolvimento, aos pontos de controle definidos nos marcos de ciclo de vida. Essa característica favorece a simultaneidade das engenharias de sistemas e de software do projeto, e indica uma preocupação do RUP com a criação de valor no desenvolvimento de sistemas de informação.

### **3.2.6 Monitoramento e Controle Baseados em Valor**

Este elemento-chave traz a ideia do uso de ferramentas baseadas em valor para monitorar o progresso do desenvolvimento de produtos de software, e agir corretivamente caso os resultados não estejam criando o valor esperado (ver Seção 2.7.9).

*Ponto de Convergência nº 6: Caso de Negócio como Ferramenta de Monitoramento*

Dentre as técnicas sugeridas (ver Seção 2.7.9), está o monitoramento através dos indicadores e funcionalidades descritos no Caso de Negócio. O RUP alinha-se a este elemento-chave por mencionar, na tarefa *Desenvolver Plano de Medidas* da disciplina de Gerenciamento de Projeto, o Caso de Negócio como fonte de informações para a definição dos objetivos primários de gerenciamento, isto é, as restrições, requisitos e funcionalidades esperadas e desejadas que precisam ser monitorados e controlados.

### **3.2.7 Mudança como Oportunidade**

Este elemento-chave trata da criação de valor nos projetos de software através do rápido aproveitamento das oportunidades trazidas com as mudanças no ambiente do projeto. Vimos no Capítulo 2 que a VBSE propõe dois tipos de técnicas para aumentar a



adaptabilidade às mudanças de uma organização que desenvolve software: técnicas centradas em arquitetura e técnicas centradas em refatoração.

Os processos de software orientados a planejamento, como o RUP, preferem uma abordagem centrada em arquitetura, a qual baseia-se na identificação das fontes de mudanças mais prováveis, e na modularização do produto de software como forma de isolar essas fontes, protegendo-as de interferências umas das outras.

#### *Ponto de Convergência nº 7: Processo Centrado na Arquitetura*

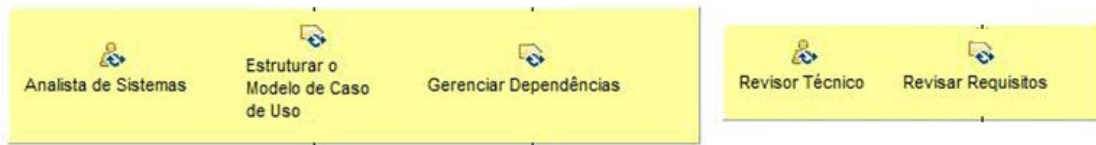
O princípio-chave “Elevar o nível de abstração” do RUP traduz-se na redução da complexidade e no aumento da produtividade no desenvolvimento através do reuso de ativos de software – código-fonte, componentes, processos de negócio, sistemas legados e outros – e no foco em uma arquitetura desde as iterações iniciais do projeto.

Além disso, um dos elementos essenciais do RUP é a arquitetura; o RUP define que a arquitetura do sistema a ser construído é a estrutura, ou organização, dos componentes mais significativos do sistema, que interagem através de interfaces. Os componentes e interfaces da arquitetura são compostos de outros componentes e interfaces sucessivamente menores.

Projetar uma arquitetura de componentes, trabalhando com alto nível de abstração, permite criar uma organização do sistema em componentes que permite o isolamento das fontes identificadas de mudança. No caso de uma mudança exigir que um componente seja alterado, esta alteração terá consequências limitadas no custo e no esforço de desenvolvimento ou implantação dos outros componentes. Assim torna-se mais fácil implantar as mudanças e aproveitar suas oportunidades.

#### *Ponto de Convergência nº 8: Controle das Mudanças nos Requisitos*

O suporte à identificação de fontes de mudança no RUP se dá através do fluxo *Gerenciar Requisitos Variáveis* da disciplina de Requisitos, cujos papéis e atividades são apresentados na Figura 3.1. Esse fluxo visa dar ao desenvolvimento meios de prever e acompanhar as constantes mudanças nos requisitos, usando rastreamento e monitoramento de requisitos de acordo com o levantamento de atributos como benefício para as partes interessadas, quantidade de esforço de implementação, risco de implementação e estabilidade (probabilidade de sofrer futuras mudanças) do requisito.



**Figura 3.1** - Tarefas e papéis do fluxo Gerenciar Requisitos Variáveis do RUP. Fonte: (IBM,2006)

O outro ponto de melhoria da adaptabilidade a mudanças é o isolamento das fontes de mudança através da modularização do software. O RUP dá ênfase à prática da modularização dado que o processo é declaradamente *centrado na arquitetura*. Isso quer dizer que as decisões de *design* do software devem ser coerentes com a arquitetura do software, sendo esta definida pelo próprio RUP como “a organização ou a estrutura dos componentes significativos do sistema que interagem por meio de interfaces, com elementos constituídos de componentes e interfaces sucessivamente menores” (IBM, 2006). Portanto, a modularização assume a forma de uma clara definição de componentes de software, que se comunicam por interfaces que isolam os detalhes de implementação, isolando, portanto, os efeitos de mudanças de implementação.

### 3.2.8 Resumo da Identificação de Convergência

A Tabela 3.1 sumariza os pontos de convergência do RUP com os elementos-chave da VBSE identificados na pesquisa deste trabalho.

**Tabela 3.1** – Pontos de Convergência por Elemento-chave (resultado obtido pelo autor)

<i>Elemento-chave</i>	<i>Pontos de Convergência encontrados no RUP</i>
Análise de Realização de Benefícios	Nenhum
Identificação e Conciliação de Proposições de Valor dos Interessados	1. Técnicas de Visualização e Análise de Trade-off
Análise de Caso de Negócio	2. Desenvolvimento do Caso de Negócio
Gerenciamento Contínuo de Riscos e Oportunidades	3. Gerenciamento de Riscos do Projeto

Engenharia Simultânea de Sistema e de Software	4. Marcos de Projeto bem Definidos 5. Integração da Modelagem de Negócios ao Processo de Software
Monitoramento e Controle baseados em Valor	6. Caso de Negócio como Fonte para o Monitoramento
Mudança como Oportunidade	7. Processo Centrado na Arquitetura 8. Controle das Mudanças nos Requisitos

### 3.3 RUP-VC: Uma Extensão do RUP para Criação de Valor

Apesar de todos os pontos de convergência do RUP à VBSE verificados na análise realizada, os elementos-chave ainda não são contemplados em sua totalidade, ou seja, nem todas as práticas sugeridas por Boehm estão contempladas na definição do processo. Assim, algumas contribuições ao processo são desejáveis, mesmo naqueles elementos-chave para os quais foram identificados pontos convergentes, no sentido de estender o RUP visando atender melhor às recomendações.

Assim, após identificar pontos de convergência entre RUP e VBSE, propõe-se estender o *framework* de processo original. O objetivo dessa extensão é, parafraseando Boehm, acrescentar considerações de valor às práticas, princípios e técnicas do RUP original. A extensão foi produzida como resultado da execução dos seguintes passos:

1. Identificação de práticas, técnicas e princípios, preconizados através dos elementos-chave da VBSE, mas não contemplados no RUP;
2. Identificação de elementos do processo (disciplinas, tarefas, artefatos, diretrizes) que pudessem abrigar *pontos de extensão* ao processo;
3. Implementação dos pontos de extensão identificados.

O termo *ponto de extensão* refere-se a qualquer intervenção em elementos do conteúdo original do RUP, para estendê-los de forma a dar-lhes uma maior ênfase na criação de valor.

O restante desta seção discute as novas possibilidades de convergência identificadas,

relativas a cada elemento-chave, e apresenta os pontos de extensão criados para atendê-las. Em seguida, a Seção 3.4 trata da publicação do *website* do *framework* de processo estendido resultante, denominado **RUP-VC** (acrônimo para *Rational Unified Process for Value Creation*).

### 3.3.1 Análise de Realização de Benefícios

A Cadeia de Resultados, descrita no Capítulo 2 como uma ferramenta útil, segundo a VBSE, para a avaliação e criação de valor em projetos de software, não é uma ferramenta adotada no RUP originalmente. Propõe-se nesta dissertação a utilização da Cadeia de Resultados como uma forma de a equipe de desenvolvimento situar de forma apropriada o projeto do software a ser desenvolvido dentro de um programa que contenha outras iniciativas relacionadas. A Cadeia de Resultados fornece um modelo pelo qual identificamos vínculos (de causa-efeito, de priorização, de requisitos) entre a iniciativa de desenvolvimento de software e outras iniciativas de negócios da organização, bem como vínculos entre o conjunto das iniciativas e os benefícios pretendidos.

#### *Ponto de Extensão nº 1: Adição de uma Disciplina de Gerenciamento de Portfólio*

Por definição, a Abordagem de Realização de Benefícios (BRA), como prática de fomento à criação de valor, é formada por três fundamentos: gestão de programas, gestão de portfólios e governança de ciclo completo. Assim, os projetos de software que se beneficiarão da BRA precisam estar inseridos em um contexto em que sejam coordenados, junto com outros projetos e iniciativas, em portfólios e programas.

O RUP, em sua forma clássica, não abrange os conceitos de portfólio e programa; em vez disso, o escopo do processo é um projeto único. Assim, para uma organização com ambiente de múltiplos projetos, isto é, com diversos projetos a gerenciar simultaneamente, as ferramentas oferecidas pelo RUP original permitem apenas tratar desses projetos em nível tático, pelos gerentes dos projetos individuais, e com horizonte de planejamento no curto prazo. São necessárias ferramentas complementares para contemplar tanto o relacionamento coordenado de projetos, através de programas, quanto o compartilhamento racional de recursos e o controle da prioridade entre projetos e programas, através de portfólios, visando os resultados de negócios numa perspectiva estratégica (DYE; PENNYPACKER, 2000).

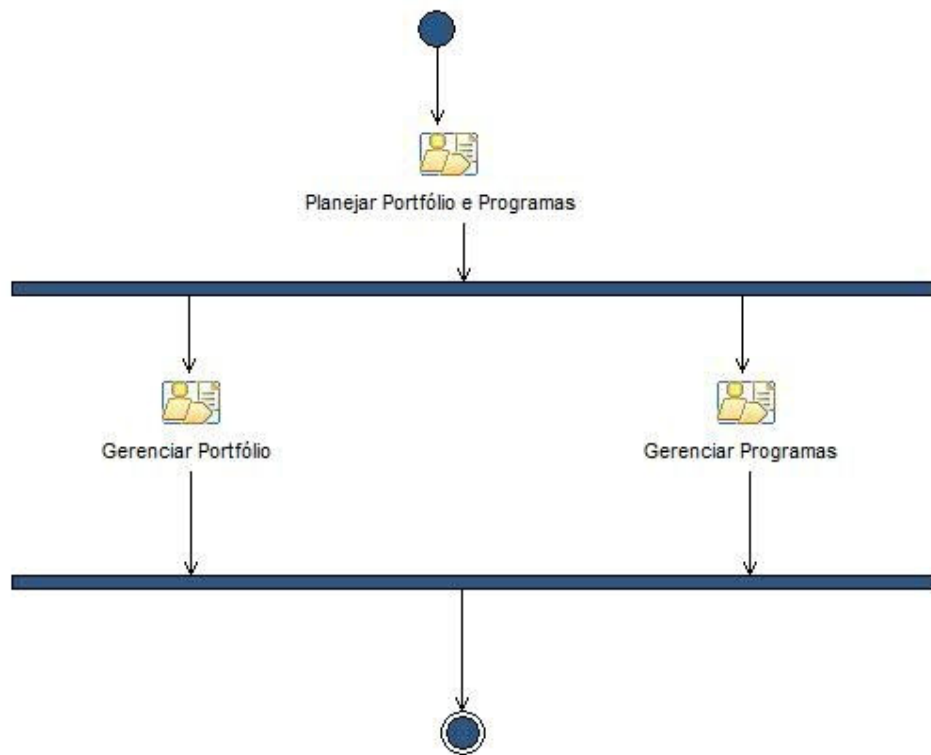
Neste sentido, a adição ao RUP de uma disciplina de Gestão de Portfólio atende a essa necessidade. Com base na definição de uma disciplina de gestão de portfólio, feita por Ambler *et al.* (2001) em sua extensão ao RUP denominada Enterprise Unified Process (EUP), e dos conceitos e boas práticas definidas pelo PMI no seu *Standard for Portfolio Management* (PMI, 2008), foi definida para este trabalho a disciplina de Gerenciamento de Portfólio de Projetos, cujo fluxo de trabalho é mostrado na Figura 3.2. A disciplina pretende fornecer os elementos essenciais para a gestão de portfólios que incluam projetos de software.

A partir da Figura 3.2 observam-se os principais fluxos de trabalho da disciplina Gerenciamento de Portfólio de Projetos:

- Planejar Portfólios e Programas: trata da avaliação, seleção, priorização, identificação de interdependências e planejamento de programas e projetos dentro de um portfólio. Inclui as seguintes tarefas:
  - *Avaliar Proposta de Projeto*: cabe às pessoas com a atribuição de Gerente do Portfólio reunir-se periodicamente – em geral a cada trimestre ou semestre – para avaliar as propostas de projetos submetidas pelas várias unidades de negócios da organização;
  - *Priorizar Projetos*: a cada novo conjunto de propostas de projeto aprovadas, a equipe de gerência de portfólio realiza a priorização dos projetos do portfólio. Projetos existentes podem ter sua prioridade revista ou mesmo serem cancelados para acomodar os novos projetos recém-aprovados.
  - *Identificar Interdependências*: os projetos e programas componentes de um portfólio podem depender dos mesmos recursos organizacionais; esta tarefa da gestão de portfólio cuida de identificar tais interdependências de forma a otimizar o uso desses recursos na gestão do portfólio.
  - *Definir Programa*: nesta tarefa, projetos com objetivos de negócio relacionados são agrupados em programas, de modo a permitir consolidar investimentos naqueles objetivos. A *Cadeia de Resultados*, descrita no ponto de extensão a seguir, é uma ferramenta importante para a execução

desta tarefa. Depois de definidos, os programas então são formalmente iniciados;

- *Desenvolver Plano de Programa*: depois de definido, um programa tem seu planejamento elaborado, onde são previstos aspectos como interfaces entre projetos do programa, planejamento dos recursos alocados, do escopo, do cronograma geral, das comunicações, dos riscos e de outras áreas do programa;
- *Desenvolver Plano de Portfólio*: o planejamento de portfólio é realizado e atualizado nesta tarefa. Nela, os programas e projetos que farão parte do portfólio são selecionados, os riscos do portfólio são analisados, e ações para manter o equilíbrio entre os componentes são definidas.
- **Gerenciar Programas**: apresenta as tarefas relacionadas ao monitoramento de cada um dos programas mantidos pela organização, alinhados aos seus objetivos de negócio estratégicos. Inclui as tarefas:
  - *Revisão de Aprovação de Projeto*: originalmente pertencente à disciplina de Gerenciamento de Projeto do RUP, esta tarefa foi trazida para a disciplina de Gerenciamento de Portfólio para refletir a atribuição de aprovação de projetos baseados em critérios estratégicos de negócios;
  - *Iniciar Projeto*: outra tarefa movida da disciplina de Gerenciamento de Projeto, esta trata dos passos iniciais para o início de um projeto, tais como designar o gerente do projeto e formalizar os parâmetros do projeto, como restrições, premissas, orçamento, data final do projeto etc.;
  - *Monitorar Programa*: trata de monitorar e controlar o programa com relação às métricas de negócios definidas no plano do programa.



**Figura 3.2** - Fluxo da disciplina Gerenciamento de Portfólio de Projetos do RUP-VC

- Gerenciar Portfólio: apresenta as tarefas relacionadas ao monitoramento do portfólio de projetos da organização como um todo. Inclui as tarefas:
  - *Revisão de Aprovação de Projeto*: esta tarefa é a mesma descrita no fluxo Gerenciar Programas, porém no contexto do portfólio ela se aplica apenas aos projetos que não estão inseridos em nenhum programa;
  - *Iniciar Projeto*: idem para a tarefa acima;
  - *Monitorar Portfólio*: monitorar o portfólio com base nos indicadores estratégicos da organização, monitorando riscos e mudanças no negócio, e revisando o desempenho organizacional.

*Ponto de Extensão nº 2: Uso da Cadeia de Resultados na Definição de Programas e Portfólio*

Tendo sido introduzida no RUP a disciplina de Gerenciamento de Portfólio de

Projetos, outra técnica foi inserida visando atender à sugestão do elemento-chave Análise de Realização de Benefícios: a utilização da Cadeia de Resultados para a definição dos programas a serem conduzidos pela organização e do portfólio de projetos que será administrado a cada momento.

Sendo a Cadeia de Resultados, apresentada na Seção 2.7.4, uma técnica de modelagem com a finalidade de permitir identificar vínculos entre negócio e TI, é natural inseri-la, dentro do escopo do fluxo de trabalho do Gerenciamento de Portfólio de projetos, nas tarefas relativas à identificação dos objetivos e resultados de negócios. Assim, foi criada uma diretriz no RUP apresentando a técnica da Cadeia de Resultados. Também foi incluída uma referência a esta diretriz na tarefa *Definir Programa*, pois a ferramenta ajuda na identificação de processos relacionados por objetivos de negócio em comum.

### **3.3.2 Identificação e Conciliação de Proposições de Valor das Partes Interessadas**

Conforme apresentado na Seção 3.1.1, o RUP recomenda a utilização de técnicas de visualização e análise de trade-off, como *storyboarding* e prototipação, na abordagem relativa ao elemento-chave nº 2, a conciliação das proposições de valor das partes interessadas a respeito do software. Por outro lado, outras ferramentas importantes sugeridas pelo elemento-chave não estão contempladas de forma explícita no processo de software. Nesta seção, propõem-se formas de preencher tais lacunas.

#### ***Ponto de Extensão nº 3: Uso de Groupware***

Conforme visto no Capítulo 2, uma abordagem sugerida pela VBSE como efetiva para identificação e conciliação de proposições de valor é o uso de software colaborativo (ou groupware) para auxiliar nas tarefas de priorização de requisitos, *brainstorming* e negociação de conflitos. O uso de groupware permite que a participação de envolvidos seja ampliada, pois a eliminação da necessidade da interação presencial, promovida pelas ferramentas colaborativas, permite expandir a quantidade de grupos de interessados, e organizar melhor suas demandas.

Na disciplina de Requisitos do RUP, a tarefa *Identificar Pedidos dos Envolvidos* tem por objetivo levantar as demandas dos interessados a respeito do que o sistema deve oferecer.



No fluxo de trabalho desta tarefa, um dos passos é “Conduzir Workshops de Requisitos”, que está associado a várias diretrizes referentes à condução dos workshops para levantamento de requisitos. Foi identificada uma lacuna exatamente neste ponto: o elemento-chave sugere que workshops de levantamento de requisitos sejam, quando possível, auxiliados por software colaborativo, ou groupware, sobretudo quando há colaboradores distribuídos geograficamente, participando de forma remota. Identificamos aí um ponto de extensão, exatamente pela inclusão no RUP de uma diretriz para o uso de ferramentas colaborativas. Esta diretriz foi associada ao mencionado passo da tarefa *Identificar Pedidos dos Envolvidos*.

#### ***Ponto de Extensão nº 4: Gerenciamento das Expectativas dos Interessados***

A gestão das expectativas dos interessados com relação ao software a ser desenvolvido é uma tarefa essencial para o maximizar o valor do produto final. Entende-se que as expectativas são importantes pois nelas os interessados baseiam suas avaliações das funcionalidades e características demandadas. Portanto, conciliar expectativas dos interessados quanto ao projeto e ao produto de software, de modo a chegar a um conjunto atingível de funcionalidades, prazo, custo e critérios de qualidade, contribui positivamente para estabelecer parâmetros razoáveis de custo e esforço de desenvolvimento, e influencia as decisões quanto a planejamento, *design* e cronogramas.

Para o RUP, a disciplina de Requisitos explica como transformar demandas dos interessados em produtos que formam o escopo do sistema. Entretanto, o RUP não declara explicitamente a preocupação com a gestão das expectativas. Assim, o RUP-VC estende a tarefa *Desenvolver a Visão* da disciplina de Requisitos, visando contemplar a gestão de expectativas dos interessados.

A tarefa *Desenvolver a Visão*, tal como originalmente definida no RUP, tem como principal artefato de saída o Documento de Visão, que define a visão dos interessados sobre o produto a ser desenvolvido, em termos de suas necessidades e recursos mais importantes. A tarefa é desempenhada através dos seguintes passos:

1. Chegar a um acordo quanto ao problema a ser resolvido
2. Identificar partes interessadas
3. Definir os limites do sistema

4. Identificar restrições a serem impostas ao sistema
5. Formular a declaração do problema
6. Definir os recursos do sistema
7. Avaliar seus resultados

Gerenciar expectativas dos interessados significa, dentro do âmbito da identificação e conciliação de proposições de valor, capturar as expectativas das partes interessadas identificadas no projeto, e realizar atividades de conciliação dessas expectativas. Assim, estendendo a tarefa a fim de trazer essas atividades para o desenvolvimento do Documento de Visão, foram incluídos dois passos a mais no fluxo da tarefa:

- “Capturar expectativas das partes interessadas”: trata-se de usar a Cadeia de Resultados para entrevistar as partes interessadas críticas para o sucesso do projeto, de maneira a validá-la e identificar suas premissas, iniciativas e resultados adicionais de alta prioridade.
- “Conciliar as expectativas das partes interessadas”: consiste em identificar e conciliar os conflitos de modelos entre os interessados, sumarizar os resultados obtidos e coordená-los, com a participação das partes interessadas, no Documento de Visão.

Após a extensão, o fluxo proposto de atividades para a tarefa fica como abaixo:

1. Chegar a um acordo quanto ao problema a ser resolvido
2. Identificar partes interessadas
3. Capturar as expectativas das partes interessadas
4. Definir os limites do sistema
5. Identificar as restrições a serem impostas ao sistema
6. Conciliar as expectativas da partes interessadas
7. Formular a declaração do problema

8. Definir os recursos do sistema
9. Avaliar seus resultados

*Ponto de Extensão nº 5: Proposições de Valor como Critérios de Priorização de Casos de Uso*

A VBSE aponta, como abordagem efetiva na conciliação das proposições de valor dos interessados, fazê-los priorizar ou categorizar as funcionalidades do software a ser desenvolvido a partir das suas proposições de valor. Ter a definição de quais são as funcionalidades mais críticas sob o ponto de vista dos envolvidos ajuda a equipe de desenvolvimento a alocar de forma mais otimizada os recursos disponíveis ao desenvolvimento, possibilitando uma consequente otimização do valor produzido.

Um dos princípios-chave do RUP é “Demonstrar Valor Iterativamente”, que diz que o desenvolvimento do software deve ser iterativo, onde cada iteração produz um incremento ao produto de software que agrega valor (do ponto de vista do usuário) ao produto. O RUP encarrega a disciplina de Gerenciamento de Projeto de planejar, executar e monitorar o desenvolvimento por meio de iterações sucessivas. Dentro da disciplina, a tarefa *Desenvolver Plano de Iteração* trata de planejar cada iteração, iniciando pela definição do escopo da iteração. Nesta tarefa, o escopo da iteração é definido em conjunto pela equipe e pelo cliente, a partir de critérios como riscos envolvidos, funcionalidade necessária (definida pelo cliente), o tempo alocado para a iteração e os objetivos específicos da fase em que a iteração se encontra. Mais especificamente, o processo define o passo denominado “Determinar o escopo da iteração”, onde são apresentadas as ações necessárias para a priorização dos casos de uso que poderão entrar no escopo da iteração. Este passo foi estendido no RUP-VC para mencionar que as proposições de valor devem ser levadas em consideração na priorização.

A tarefa de planejamento da iteração acima descrita é municiada pela priorização técnica dos casos de uso, que o RUP sugere ser conduzida pelo arquiteto de software, e que está contida no Documento de Arquitetura de Software. Trata-se da tarefa *Priorizar Casos de Uso*, da disciplina de Requisitos. O RUP-VC estende a tarefa inserindo uma ferramenta baseada em valor para a priorização de casos de uso: o diagrama de conflito de modelos, ou teia de conflito de modelos, apresentado na Seção 2.7.5. A ideia é construir um diagrama de conflito de modelos para identificar conflitos existentes entre a relevância

atribuída por cada grupo de interessados aos diversos aspectos do produto de software: requisitos funcionais e não funcionais, decisões de design, restrições de gerenciamento do projeto etc.

### **3.3.3 Análise de Caso de Negócio**

Um caso de negócio tradicional de um projeto apresenta as opções de solução para o projeto, comparando o retorno sobre o investimento (ROI) entre as opções, e sugerindo como solução economicamente mais atrativa aquela que tiver melhor ROI.

Quanto à criação de valor, a VBSE adiciona dois fatores além do ROI, que podem ser importantes: benefícios intangíveis e riscos.

#### *Ponto de Extensão nº 6: Análise de Opções considerando Riscos e Benefícios Intangíveis*

Como dito no Capítulo 2, a análise de caso de negócio é contemplada no RUP. Entretanto, a metodologia sugerida para tal análise é a tradicional, visando obter o ROI com base nos custos financeiros e nos benefícios quantificáveis financeiramente. A VBSE sugere considerar, na análise da viabilidade econômica de um projeto de software, os riscos envolvidos e os benefícios cuja quantificação monetária não seja possível ou não tenha precisão aceitável.

A tarefa *Desenvolver Caso de Negócio* da disciplina de Gerenciamento de Projetos do RUP é executada durante o planejamento de cada iteração ou fase do projeto. Propõe-se no RUP-VC estender a tarefa, incluindo análise de opções baseada nos riscos. Isto se traduz em analisar as opções de realização do trabalho de desenvolvimento a ser executado, considerando na decisão fatores outros além do retorno financeiro.

Em um contexto de análise de caso de negócio baseado em valor, cada fase do ciclo de vida apresenta diversos trade-offs de negócios, determinando opções de abordagem do projeto. Reifer (2002) dá alguns exemplos de trade-offs:

Durante a fase de Concepção:

- Analisar, a partir dos objetivos iniciais do projeto, trade-offs de custo, cronograma e funcionalidade, usando técnicas de análise de custo como variável independente;

- Determinar o preço do produto de software (ou outra atividade do projeto que gere receita) usando análise de tendências para prever o volume de vendas necessário;
- Requisitar orçamentos usando análise de custo/benefício para justificar as despesas necessárias.

Durante a fase de Elaboração:

- Conduzir análise entre "fazer" e "adquirir" (*make-or-buy*) para determinar quais escolhas de componentes melhor preenchem a arquitetura, usando técnicas de análise de retorno e de oportunidade de investimento. Uma perspectiva de opções reais pode ser utilizada para tal (SULLIVAN et al., 1999);
- Desenvolver orçamentos para o custo do software e cronograma usando modelos como o COCOMO II (BOEHM *et al.*, 2000b) para estimar o esforço e a duração necessária;
- Priorizar riscos (positivos e negativos) através de uma avaliação quantitativa dos impactos dos itens de risco.

Durante a fase de Construção:

- Decidir quanto ao lançamento ou não de uma versão ou incremento do produto baseando-se em análise de tendências (por exemplo, olhando para defeitos e níveis de qualidade associados);
- Desenvolver estimativas de custo e cronograma de conclusão, usando modelos para determinar se é possível ou não entregar como prometido, dependendo da taxa de progresso atual;

Durante a fase de Transição:

- Conduzir trade-offs entre "consertar" ou "substituir" (*replace/repair*) usando técnicas como análise de Pareto, para determinar quais módulos do software estão mais sujeitos a defeitos;
- Desenvolver estimativas quanto à manutenção, usando modelos de custo para prever esforço e duração baseado em projeções estatísticas de mudanças.

### 3.3.4 Gerenciamento Contínuo de Riscos e Oportunidades

Como visto no Capítulo 2, a análise e o gerenciamento de riscos e de oportunidades em um projeto de software estende a análise prévia de viabilidade, feita no caso de negócio, para todo o ciclo de vida do sistema. Mais importante: no que se refere à criação de valor, a gestão dos riscos introduz o “fator gente” na tomada de decisões de caráter econômico, através da análise da utilidade, relacionada a riscos, atribuída pelos indivíduos.

O RUP apresenta um tratamento consistente dos riscos, embutido principalmente na disciplina de Gerenciamento de Projeto, mas também presente em preocupações quanto à arquitetura e a disciplina de testes. No entanto, na análise do processo com relação a este elemento-chave, foram descobertas algumas deficiências. São elas:

- O conceito de oportunidade – significando “riscos positivos”, como descritos no Capítulo 2 – não é tratado pelo RUP. Em vez disso, apenas os riscos que representam ameaças ao sucesso do projeto (riscos negativos) são considerados;
- A aversão ao risco, decorrente das utilidades que as partes interessadas atribuem às funcionalidades, não é explicitamente tratada no RUP;
- Os riscos associados à adaptação do processo de software ao projeto não são explicitamente tratados.

Abaixo são propostos três pontos de extensão à gerência de riscos prevista no RUP, de modo a cobrir essas deficiências.

#### *Ponto de Extensão nº 7: Gerenciamento de Oportunidades*

O gerenciamento de oportunidades difere do gerenciamento de ameaças de ameaças nas estratégias de ação que são utilizadas basicamente na resposta a esse tipo positivo de risco. Enquanto o RUP define que as estratégias de resposta aos riscos negativos tratam de preveni-los, transferi-los para terceiros ou mitigá-los, isto é, reduzir sua probabilidade ou seu impacto, as estratégias para lidar com oportunidades incluem, segundo o Guia PMBOK (PMI, 2004):

- *Explorar*, que consiste em eliminar a incerteza associada a um risco positivo, fazendo com que a oportunidade efetivamente aconteça;

- *Compartilhar*, que envolve atribuir parte da propriedade do risco a terceiros que possam capturar melhor a oportunidade em benefício do projeto;
- *Melhorar*, que visa aumentar a exposição ao risco positivo, através do aumento da probabilidade e/ou do impacto caso ocorra. Isso se dá pela identificação e maximização dos acionadores dos riscos de impacto positivo.

Há também uma estratégia comum a riscos positivos e negativos. Trata-se da *aceitação* do risco, que significa que a equipe de projeto resolveu não mudar o planejamento para tratar o risco. Esta estratégia pode ser *passiva*, na qual a equipe só tratará o risco aceito à medida que ele ocorra, ou *ativa*, quando é planejada uma reserva de contingência para tratar as ameaças ou oportunidades.

Este trabalho propõe estender o RUP acrescentando à diretriz *Lista de Riscos*, que descreve como identificar e gerenciar riscos de projeto, as estratégias para resposta aos riscos positivos, conforme as orientações do Guia PMBOK. Também acrescenta, à tarefa *Identificar e Avaliar Riscos*, um passo específico sobre como identificar estratégias de exploração de riscos positivos.

#### *Ponto de Extensão nº 8: Avaliação de Riscos a Partir das Funções de Utilidade dos Interessados*

A gerência de riscos orientada a valor proposta na VBSE leva em consideração a análise da utilidade das opções de implementação do projeto para os interessados. Estudando como as funções de utilidade contribuem para a aversão aos riscos por parte dos interessados, torna-se possível propor a conciliação das funções de utilidade, de forma análoga à conciliação das proposições de valor discutida na Seção 2.7.5, como uma das estratégias para lidar com os riscos.

Neste ponto de extensão, propõe-se estender a tarefa *Identificar e Avaliar Riscos*, em especial o passo “Identificar riscos potenciais”, de forma a levantar os riscos decorrentes das incompatibilidades entre as funções de utilidade. Para a efetividade dessa extensão, propõe-se também adequar, na tarefa *Desenvolver Plano de Gerenciamento de Riscos*, o planejamento das respostas aos riscos de forma a lidar com os riscos (e oportunidades) relativos às funções de utilidade percebidas pelos usuários.

*Ponto de Extensão nº 9: Adaptar o Processo de Software Considerando a Exposição ao Risco*

Um assunto bastante discutido nos últimos anos, como apontado por Boehm e Turner (2004), é o uso de processos orientados a planejamento, como o RUP, em contraposição ao uso de métodos ágeis, dentre eles Extreme Programming (XP) e Scrum. Em relação a este assunto, a VBSE propõe o uso do nível de exposição a riscos relativos à quantidade de “burocracia” do processo de software utilizado no projeto (a variável ER definida na Seção 2.7.7), como balizador das adaptações (*tailoring*) ao processo utilizado no desenvolvimento.

A disciplina Ambiente do RUP possui a tarefa *Adaptar o Processo de Desenvolvimento para o Projeto*, dedicada à adaptação do processo de desenvolvimento ao projeto em questão. Como extensão ao processo, no passo “Analisar o Projeto” desta tarefa, foi feita a inclusão de uma avaliação da exposição aos riscos inerentes à adoção das principais disciplinas, fluxos de trabalho, tarefas e artefatos do RUP. Esse fator é determinante na definição de quão “ágil” será o processo a ser adotado.

### **3.3.5 Engenharia Simultânea de Sistemas e de Software**

Do exposto na Seção 3.2.5, vimos que o RUP apresenta um bom nível de convergência ao elemento-chave que preconiza a simultaneidade entre o desenvolvimento da solução (engenharia de sistemas) e o desenvolvimento do software (engenharia de software). Para aumentar ainda mais a convergência a este elemento-chave, estendeu-se a tarefa de revisão dos marcos de ciclo de vida, como descrito a seguir.

*Ponto de Extensão nº 10: Revisão de Marco de Ciclo de Vida Baseada em Valor*

A VBSE considera, como uma característica importante do elemento-chave, o uso de múltiplos indicadores de consistência e viabilidade como critérios de revisão e decisão dos marcos de ciclo de vida. Reifer (2002) propõe que o Caso de Negócio do projeto de desenvolvimento seja utilizado como um dos balizadores da revisão. Esse artefato deve ser examinado para revalidar suas suposições e conclusões, à luz de qualquer alteração no contexto geral do negócio ou dos sistemas do projeto.



A avaliação dos resultados do desenvolvimento do software (atividade sob a abrangência da Engenharia de Software) e das outras atividades necessárias à conclusão do projeto ou iniciativa de negócios (área abrangida pela Engenharia de Sistemas) é prevista no RUP dentro da tarefa *Revisão de Marco de Ciclo de Vida*, da disciplina de Gerenciamento de projeto. Durante a tarefa, os trade-offs analisados e descritos no Caso de Negócio, descritos na Seção 3.3.3, devem ser reavaliados; analogamente, mudanças nas premissas, contribuições e resultados da Cadeia de Resultados também devem ser analisadas na revisão. Os principais fatores para avaliação foram incluídos como extensão do RUP-VC à tarefa acima mencionada, na seção “Problemas para Consideração” da descrição principal da tarefa.

### **3.3.6 Monitoramento e Controle Baseados em Valor**

Como visto no Capítulo 2, a VBSE questiona o método *Earned Value Management* (EVM) para o monitoramento de projetos de software baseados em valor, tanto pelo fato de o EVM se tornar difícil de implementar se o projeto sofre muitas e frequentes mudanças, quanto por não se conseguir obter qual o real valor que está sendo criado pelo projeto. Assim, a VBSE propõe outras técnicas que devem ser utilizadas para o controle baseado em valor.

Analisando o RUP no que diz respeito a essas técnicas, propõe-se que algumas tarefas da disciplina de Gerenciamento de Projeto sejam modificadas para atender ao monitoramento baseado em valor.

#### *Ponto de Extensão nº 11: Monitoramento de Valor no Nível do Projeto*

A VBSE propõe que o Caso de Negócio forneça indicadores do valor que deverá ser criado pelas funcionalidades desejadas pelo software. Estes indicadores devem ser utilizados no monitoramento do valor criado no projeto para a organização.

No RUP, a diretriz de Métricas descreve princípios relativos a medidas que devem ser coletadas e analisadas no monitoramento dos projetos que usam o processo. Também descreve um conjunto mínimo de métricas a ser aplicado no projeto. A diretriz é um dos insumos para a tarefa *Desenvolver Plano de Medidas* da disciplina de Gerenciamento de Projeto. O cumprimento bem sucedido do projeto será determinado pelas metas descritas no Plano de Medidas.

No RUP-VC, a diretriz de Métricas foi estendida para incluir os *indicadores de valor*

*de negócio* que devem ser monitorados, tendo sido adicionados alguns exemplos de tais indicadores, basicamente extraídos de duas fontes:

- Fatores de trade-off, contidos no Caso de Negócio e apresentados na Seção 3.3.3;
- Iniciativas e resultados da Cadeia de Resultados nos quais o projeto está envolvido.

Além disso, foi estendido o passo “Desenvolver os objetivos primários de gerenciamento” da tarefa supracitada, que é onde são determinadas e registradas as restrições e os requisitos funcionais, não-funcionais, orçamentários e de programação importantes que precisam ser controlados. Nesse passo foi incluída a menção à Cadeia de Resultados como fonte de indicadores de valor de negócio.

#### *Ponto de Extensão nº 12: Monitoramento de Valor no Nível da Organização*

O RUP é um processo cujo foco é o projeto de software individual. A extensão RUP-VC adicionou a disciplina de Gerenciamento de Portfólio, que permite criar, conduzir e monitorar diversos projetos de software, agrupados ou não em programas. Na disciplina, a tarefa Monitorar Portfólio trata de fazer o monitoramento do portfólio baseado em valor. Utilizando a Cadeia de Resultados e os Casos de Negócio individuais de cada projeto, é possível derivar e monitorar indicadores dos objetivos estratégicos de negócio relativos aos projetos e programas. Uma abordagem para alcançar esse cenário é a implantação da Fábrica de Experiências baseada em valor (BOEHM *et al.*, 2002). Uma diretriz foi criada no RUP-VC descrevendo a abordagem.

### **3.3.7 Mudança como Oportunidade**

Com o sentido de prover o RUP com ferramentas para responder às mudanças que trazem consigo oportunidades, propõe-se neste trabalho fazer uma adaptação (*tailoring*) do RUP ao projeto de desenvolvimento, de maneira baseada em valor, de forma que o processo adaptado torne-se mais “ágil”. A decisão para a adaptação baseia-se na natureza do projeto: caso seja de tamanho pequeno para a organização, e com requisitos que mudam rapidamente, dar agilidade ao processo é fundamental para o aproveitamento das oportunidades que as mudanças podem trazer. Caso o tamanho e a criticidade do projeto exijam um grande esforço para incorporar as demandas frequentes por mudanças, é mais indicada a abordagem tradicional do RUP, de controle de mudanças de requisitos associada à modularização

arquitetural das funcionalidades mais passíveis de ser fontes de mudança.

**Ponto de Extensão nº 13: Permitir Adaptações Visando Agilidade do Processo**

O RUP contempla o *tailoring* do processo na tarefa *Adaptar o Processo de Desenvolvimento para o Projeto*, da disciplina Ambiente. Nela são apresentadas as considerações a serem avaliadas para a definição do escopo da adaptação. A extensão RUP-VC acrescenta a essas considerações a exposição aos riscos decorrentes de se adicionar ou retirar “burocracia”, ao processo, deixando-o mais pesado ou mais leve. Também foi estendida a diretriz “Discriminadores do Processo”, que apresenta fatores que afetam a customização do RUP. A diretriz recebeu a adição do fator adaptabilidade a mudanças. Também foi acrescentada uma referência a Lines *et al.* (2008), que oferece sugestões de aplicação de estratégias ágeis no RUP.

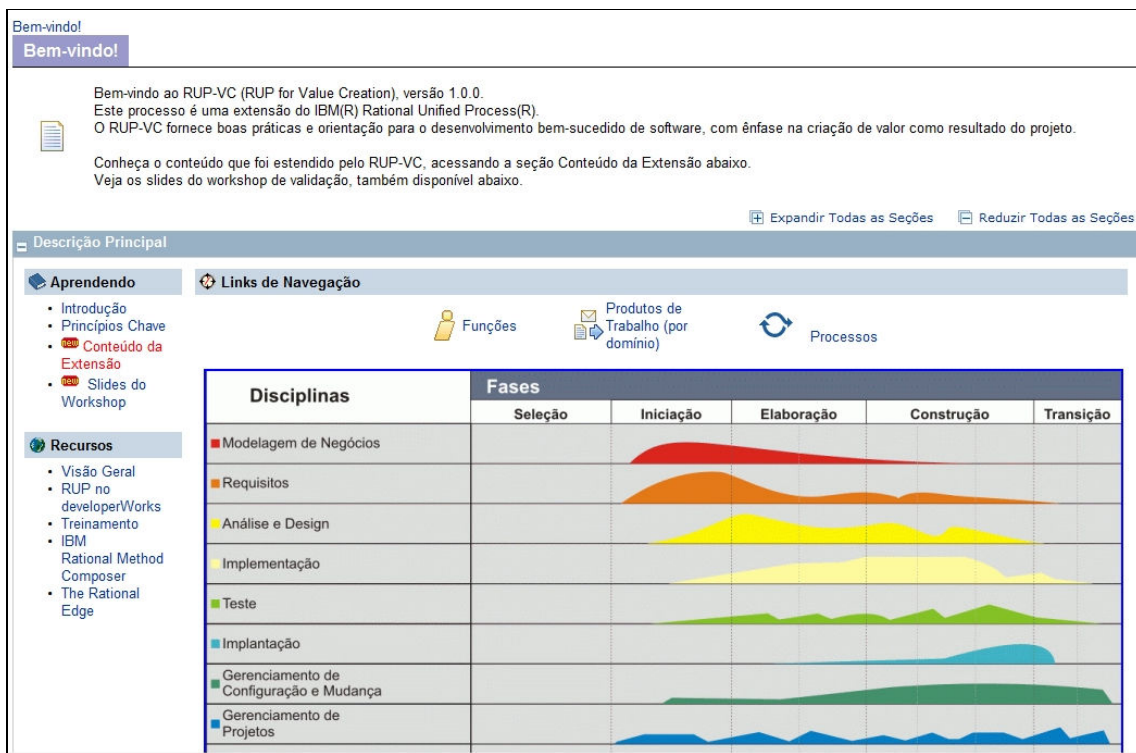


Figura 3.3 - Home page do site do RUP-VC

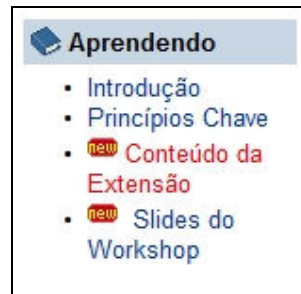
**3.4 O Site da Extensão RUP-VC**

O produto final da extensão é a catalogação de todas as intervenções feitas, compiladas no formato de apresentação utilizado pelo RUP e publicadas em um

*website*. A URL do site publicado é <http://www.cin.ufpe.br/~gta/rup-vc>. Esta foi a URL disponibilizada para o processo de avaliação, a ser apresentado no Capítulo 4. A Figura 3.3 mostra a home page do website.

Para a produção do material do site, tomou-se como fonte a biblioteca do *framework* de processo RUP, publicada pela IBM (2006). Os artefatos do processo alterados ou adicionados foram produzidos com a ferramenta Eclipse Process Framework Composer (Eclipse Foundation, s.d.). A ferramenta também foi utilizada para a publicação do website com os resultados.

Devido ao número considerável de pontos de extensão, optou-se por confeccionar uma página do site que funcionasse como um índice das intervenções feitas. A página, intitulada “Conteúdo da Extensão”, é referenciada em destaque na home page do site do RUP-VC; a Figura 3.4 mostra o link em detalhe. O resumo de intervenções por elemento-chave da VBSE mostrado na página “Conteúdo da Extensão” está reproduzido no Apêndice A.



**Figura 3.4** - Detalhe da home page do RUP-VC (link para o conteúdo da extensão)

É importante explicar algumas notações utilizadas na extensão, que ajudam a identificar o conteúdo adicionado ou alterado:

- Os elementos do RUP que tiveram seu conteúdo original alterado estão indicados como “estendidos”; os elementos adicionados como resultado da extensão estão indicados como “adicionados”. Todos estão listados no Apêndice A.
- Nos fluxos das tarefas que foram objeto de intervenção, os passos alterados ou adicionados são indicados conforme o elemento-chave ao qual se referem, pela inscrição “(VBSE EC $n$ )”, onde EC significa “elemento-chave”, e  $n$  é um número de 1 a 7, ao final da sua descrição.

- Texto acrescentado ou modificado dentro do conteúdo foi identificado com a formatação *sublinhado itálico*.

O Apêndice B apresenta, como exemplo de conteúdo estendido publicado no site, as intervenções realizadas no RUP, relativas ao elemento-chave Análise de Caso de Negócio.

### **3.5 Considerações sobre a Análise**

Os pontos de convergência encontrados no RUP, em um total de oito, traduzem o nível de aproximação entre os conceitos de criação de valor sugeridos pela VBSE e a prática de um *framework* de processo de software moderno. Já os pontos de extensão que compõem o RUP-VC, em um total de treze, são intervenções que se destinam a fornecer novos pontos de convergência, no que diz respeito ao atendimento das propostas dos elementos-chave pelo RUP.

É importante frisar que o estudo da VBSE se desenvolve sobre os elementos-chave apresentados neste trabalho. A intenção desta pesquisa não foi produzir um conjunto de aperfeiçoamentos ao RUP que contemplasse todas as práticas estudadas pela bibliografia existente sobre VBSE. Em vez disso, apenas os conceitos básicos expressos por Boehm (2003) nos sete elementos-chave foram considerados; foram identificados pontos de extensão para todos os elementos-chave. Na definição de cada ponto de extensão, houve o cuidado de incluir na intervenção técnicas utilizadas em outros pontos de extensão, caso também se adequassem ao ponto em questão. Com isso, procurou-se manter a consistência entre o que preconizam os elementos-chave da VBSE e a utilização dos mesmos no conteúdo do RUP-VC.

Uma questão relacionada ao ponto anterior é o porquê da decisão em propor a extensão RUP-VC considerando apenas seus fundamentos, cristalizados nos elementos-chave da VBSE, não agregando outros desenvolvimentos como, por exemplo, o conjunto de práticas e aplicações reunidas nas Partes 2 e 3 de BIFFL *et al.* (2006). A opção foi pela análise em amplitude de um *framework* de processo de software, de modo a abranger os mais variados conceitos relacionados à VBSE. Além disso, o esforço de tempo em se investigar a introdução em profundidade na extensão de trabalhos derivados dos princípios não seria exequível no espaço de tempo de uma Dissertação de Mestrado. Como consequência, dependendo do ponto de vista a extensão possivelmente sofre de falta de profundidade; no entanto, ela explora toda

a abrangência dos fundamentos da VBSE.

Outro ponto a destacar é a preocupação em refletir na análise as diferentes formas de valor que fazem parte da definição de valor adotada (apresentada na Seção 2.5). Além do tratamento do valor financeiro – presente no RUP original (caso de negócio, monitoramento de custos etc.), mas também enriquecido com a inserção do gerenciamento de portfólio (ponto de extensão nº1) – foram integradas no RUP-VC as preocupações com benefícios intangíveis – sobretudo nos pontos de extensão 2, 4, 5, 6, 11 e 12 – e com a gestão do valor das oportunidades – em especial nos pontos de extensão 7, 8 e 9.

Cabe aqui uma breve análise do custo da inclusão dos pontos de extensão ao RUP. Sem dúvida, a intervenção mais custosa em termos de esforço para realizar é a disciplina de gerenciamento de portfólio (ponto de extensão nº1). A disciplina adiciona complexidade ao processo, pois expande o escopo do fluxo de gerenciamento do RUP original para abranger múltiplos projetos, agregando novas responsabilidades, atividades e ferramentas, e tornando as equipes de gestão de cada projeto individual partes integrantes de uma equipe organizacional mais abrangente. Outros pontos de extensão dependem de alguma forma do ponto nº1 para serem implantados, em especial os pontos nº2 (uso da Cadeia de Resultados) e nº12 (monitoramento no nível organizacional). As outras intervenções estão em um nível de implantação mais imediato, estendendo tarefas já existentes para dar um enfoque diferente aos objetivos das mesmas.

Como última consideração, é preciso deixar claro que o trabalho de criação da extensão RUP-VC limitou-se à proposição de evoluções aos elementos do RUP, no intuito de provê-los de maior ênfase na criação de valor. Está fora do escopo da pesquisa a definição de metodologias de implantação do RUP-VC em ambientes organizacionais, embora tal atividade fosse necessária para um futuro estudo experimental da extensão (mais sobre trabalhos futuros no Capítulo 5). Apenas à guisa de informação, ressalte-se que Boehm (2003) fornece um conjunto de diretrizes para implantar os elementos-chave da VBSE em nível organizacional, bem como em nível nacional ou global.



## 4 Avaliação da Extensão RUP-VC

Como metodologia para avaliação do RUP-VC, optou-se por uma abordagem qualitativa, que consistiu em apresentar a proposta desta dissertação em um workshop a gerentes de projetos de desenvolvimento de software e desenvolvedores de software, com experiência com o RUP e indagar sobre suas opiniões em relação à proposta apresentada. Este capítulo descreve o processo de avaliação.

### 4.1 O Processo de Avaliação

O trabalho de avaliação foi estruturado nas seguintes fases:

- Definição de Objetivos: definir os objetivos da avaliação
- Planejamento da Avaliação: definição das etapas para execução da avaliação
- Execução da Avaliação: pôr em prática o planejamento realizado
- Análise e Interpretação dos Resultados: produção de uma análise a partir dos resultados da avaliação
- Implantação das Correções e Melhorias: voltar ao RUP-VC para aplicar melhorias resultantes da análise.

As seções seguintes apresentam cada fase com maiores detalhes.

### 4.2 Definição dos Objetivos

Uma avaliação para a extensão RUP-VC deve relacionar-se com a pergunta-problema da pesquisa. Uma forma de sondar se os projetos de desenvolvimento criam valor é investigar se o processo de desenvolvimento utilizado nos projetos contém técnicas e práticas que enfatizem a criação de valor pelo projeto. Assim, decidiu-se por avaliar qualitativamente a extensão RUP-VC proposta neste trabalho, do ponto de vista de gerentes de projetos e de desenvolvedores de software que utilizam o RUP, buscando-se identificar o grau de concordância (aprovação) ou discordância (reprovação) desses profissionais com a extensão, como meio de verificar se ela oferece suporte à criação de valor em projetos de



software em termos das práticas e atividades utilizadas para tal. Esta foi considerada, para a pesquisa, uma forma adequada de responder a pergunta-problema.

Em uma perspectiva de estudo experimental é possível resumir os objetivos da avaliação do RUP-VC conforme apresenta a Tabela 4.1, que detalha os atributos do estudo realizado.

**Tabela 4.1** - Caracterização da avaliação como estudo experimental  
(resultado obtido pelo autor)

<i>Atributo</i>	<i>Valor para o Experimento</i>
Analisar (objeto do estudo)	O conjunto das intervenções reunidas na extensão RUP-VC
Com o propósito de (objetivo)	Avaliar e melhorar a referida extensão
Com respeito a (foco da qualidade)	Ênfase na criação de valor em projetos de software
Do ponto de vista (perspectiva)	Do gerente de projetos de software e do desenvolvedor de software que utilizam o RUP
No contexto de (amostra da pesquisa)	Profissionais com formação em Ciência da Computação e áreas afins, e com experiência em gerenciamento de projetos de software ou em desenvolvimento de software em projetos com o RUP.

### **4.3 Planejamento**

Após a definição dos objetivos, na fase de planejamento as seguintes atividades foram executadas:

1. Definir a abordagem para a avaliação;
2. Elaborar questionário;
3. Preparar o resumo da proposta para acompanhar o questionário;
4. Definir data e local do workshop;

5. Elaborar convocação e encaminhar;
6. Preparar material a ser distribuído conforme quantitativo de convites aceitos.

As sub-seções seguintes trazem o detalhamento das atividades de planejamento.

#### **4.3.1 Definir a Abordagem para a Avaliação**

Considerando que se tem como objetivo identificar aspectos que conduzam à concordância (aprovação) ou discordância (reprovação) de profissionais da área com a proposta desta dissertação, optou-se por construir uma escala de sete posições, que vai da atitude mais favorável (concordância), em relação a uma determinada proposição, até a mais desfavorável (discordância), de forma que o participante possa externar a sua posição sobre as proposições constantes do questionário aplicado (vide Apêndice C).

Cada proposição conta com um conjunto de respostas dentro da escala definida, a saber: concordo totalmente (C+); concordo (C<sub>0</sub>); concordo parcialmente (C-); discordo parcialmente (D-); discordo (D); discordo totalmente (D+); sem opinião (SO).

Após esta definição, partiu-se para a seleção dos pontos relevantes que representam a proposta da extensão RUP-VC, a partir da consideração de questões previamente definidas. Os seguintes pontos foram selecionados como importantes para se avaliar na extensão:

- A importância e a oportunidade do tema escolhido;
- O conceito de criação de valor em projetos adotado no trabalho;
- A escolha do RUP como *framework* de processo a ser analisado e estendido;
- A escolha da Engenharia de Software Baseada em Valor (VBSE) como diretriz da análise;
- A adequação da implementação dos elementos-chave na extensão, aos objetivos de cada elemento-chave.

#### **4.3.2 Elaboração do Questionário**

Dado o escopo da avaliação, e as restrições de disponibilidade de tempo dos

participantes para participar da mesma, considerou-se que um questionário seria um instrumento adequado para interagir formalmente com os participantes da avaliação. Através das respostas ao questionário torna-se possível identificar motivos para concordância ou discordância dos profissionais da área pesquisados com a proposta desta dissertação.

Para a elaboração do questionário foi utilizada a metodologia proposta por Aaker *et al.* (2004), que consiste em:

1. Planejar o que vai ser mensurado;
2. Formular perguntas para obter as informações necessárias;
3. Definir o texto e a ordem das perguntas e o layout do questionário;
4. Testar o questionário, com relação a omissões e a ambiguidade;
5. Corrigir problemas decorrentes do teste, caso necessário.

#### **4.3.3 Preparação do Resumo da Proposta**

A fim de introduzir o tema aos convidados, visando aumentar o interesse deles em participar, e ao mesmo tempo visando proporcionar um nivelamento prévio de conhecimentos de forma a otimizar o tempo utilizado no workshop, foi produzido um texto de duas páginas resumindo a pergunta-problema sendo respondida neste trabalho, bem como seu contexto.

#### **4.3.4 Definição de Data e Local do Workshop**

As sessões de workshop foram agendadas pensando em maximizar a adesão dos convidados, em horários e locais com a máxima conveniência, de acordo com as disponibilidades de tempo dos convidados.

#### **4.3.5 Convocação do Público-alvo**

A convocação do público-alvo foi feita através de e-mail enviado a um total de 19 pessoas, contendo o resumo da proposta da dissertação, mencionado na Seção 4.3.5.

#### **4.3.6 Preparação do Material para o Workshop**

O material utilizado nas sessões do workshop foram cópias impressas do resumo e do

questionário, os slides eletrônicos para apresentação e o site publicado do RUP-VC.

#### **4.4 Execução**

Após a preparação, procedeu-se à execução do estudo, com a realização das sessões de workshop. Foram realizadas três dessas sessões:

- Em 23/03/2009 e em 14/04/2009 no Centro de Informática da UFPE
- Em 24/03/2009 no Tribunal de Contas do Estado de Pernambuco.

No total dos três workshops, das 19 pessoas convidadas, houve um total de nove participantes presentes. O índice de abstenção foi de 52,63%.

#### **4.5 Análise e Interpretação dos Resultados**

Em primeiro lugar, o perfil dos respondentes foi analisado. Como visto na Tabela 4.2, todos os respondentes têm formação em Tecnologia da Informação: cinco têm título de Mestre em Ciência da Computação, três têm graduação na mesma área, e um tem especialização em TI.

Outro dado a ressaltar é a experiência dos respondentes em gestão de projetos. Dos nove respondentes, oito têm pelo menos dois anos de experiência profissional em atividades de gestão de projetos, e pelo menos um ano participando do gerenciamento de projetos com RUP. Sete respondentes relataram experiência no exercício da função de gerente de projeto, variando entre quatro e dez anos.

Em resumo, a qualificação dos respondentes enquadra-os no perfil desejado para avaliar a extensão RUP-VC, qual seja, de *practitioners* da área de gerenciamento de projetos de software.

A Tabela 4.3 representa a frequência das respostas para cada uma das proposições constantes no questionário apresentado no Apêndice C. Em seguida, são apresentados os comentários a respeito das respostas apresentadas pelos respondentes a cada proposição, e também contra-argumentações aos motivos apresentados para as respostas que não tiveram concordância plena, isto é, respostas diferentes de C+.

Tabela 4.2 - Perfil dos respondentes

<b>Respon- dente</b>	<b>Escolaridade</b>	<b>Experiência Profissional</b>	<b>Funções exercidas em TI</b>
1	Especialização em Tecnologia da Informação (CIn-UFPE)	Desenvolvimento de Sistemas (15 anos); Gerenciamento de Projetos (6 anos); Gerenciamento de Projetos com o RUP (6 anos);	Gerente de Projeto (3 anos); Desenvolvedor (8 anos); Analista de Processos e Qualidade (8 anos);
2	Graduação em Ciências da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (12 anos); Gerenciamento de Projetos (6 anos); Gerenciamento de Projetos com o RUP (6 anos);	Gerente de Projeto (6 anos); Desenvolvedor (6 anos);
3	Mestrado em Ciência da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (7 anos); Gerenciamento de Projetos (4 anos); Gerenciamento de Projetos com o RUP (4 anos);	Gerente do Setor de Desenvolvimento de Software (3 meses); Gerente de Projeto (4 anos); Desenvolvedor (3 anos);
4	Mestrado em Ciência da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (15 anos); Gerenciamento de Projetos (10 anos); Gerenciamento de Projetos com o RUP (6 anos);	Gerente de Projeto (10 anos); Desenvolvedor (7 anos);
5	Graduação em Ciências da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (2 anos); Gerenciamento de Projetos (4 anos); Gerenciamento de Projetos com o RUP (2 anos);	Gerente de Projeto (4 anos); Desenvolvedor (2 anos); Engenheira de Qualidade (2 meses);
6	Mestrado em Ciência da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (8 anos); Gerenciamento de Projetos (2 anos); Gerenciamento de Projetos com o RUP (2 anos);	Gerente de setor na área de informática (2 anos) Desenvolvedor (8 anos);
7	Mestrado em Ciência da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (10 anos);	Desenvolvedor (6 anos); Testadora de Software (1 ano e meio);
8	Mestrado em Ciência da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (6 anos); Gerenciamento de Projetos (13 anos); Gerenciamento de Projetos com o RUP (5 anos);	Gerente de Projeto (4 anos); Supervisor (9 anos);
9	Graduação em Ciências da Computação (CIn-UFPE)	Desenvolvimento de Sistemas (6 anos); Gerenciamento de Projetos (4 anos); Gerenciamento de Projetos com o RUP (1 ano);	Gerente do Setor de Desenvolvimento de Software (2 anos); Gerente de Projeto (4 anos); Desenvolvedor (6 anos);

**Tabela 4.3** - Tabulação das Respostas do Questionário de Avaliação do RUP-VC

<i>Proposição</i>	<i>Frequência de respostas</i>						
	C+	C <sub>0</sub>	C-	D+	D <sub>0</sub>	D-	SO
<b>2.1</b> O tema escolhido é importante e oportuno.	7	2					
<b>2.2(A)</b> A mudança de foco do gerenciamento de projetos, saindo da criação de um produto ou serviço, para a criação de valor para a organização patrocinadora, demanda que a gerência de projetos suba a um patamar estratégico nas organizações.	4	3	2				
<b>2.2(B)</b> O conceito de valor criado por um projeto considerado neste trabalho é adequado à função estratégica dos projetos.	6	2	1				
<b>2.3</b> Foi acertada a escolha do RUP como base para análise e extensão propostas neste trabalho, de modo a integrar considerações de valor a projetos de desenvolvimento de software e dar ênfase à criação de valor nesses projetos.	4	3	1				1
<b>2.4</b> A Engenharia de Software Baseada em Valor é um <i>framework</i> adequado para integrar considerações de valor ao processo RUP.	3	4	1				1
<b>2.5</b> A forma como foi implementada a Análise de Realização de Benefícios no RUP-VC atende aos objetivos deste elemento-chave.	2	5	1				1

**Tabela 4.3** - Tabulação das Respostas do Questionário de Avaliação do RUP-VC (continuação)

<i>Proposição</i>	<i>Frequência de respostas</i>						
	C+	C <sub>0</sub>	C-	D+	D <sub>0</sub>	D-	SO
<b>2.6</b> A forma como foram implementadas a Identificação e a Conciliação de Proposições de Valor das Partes Interessadas no RUP-VC atende aos objetivos deste elemento-chave.	4	3	2				
<b>2.7</b> A forma como foi implementada a Análise de Caso de Negócio no RUP-VC atende aos objetivos deste elemento-chave.	2	6	1				
<b>2.8</b> A forma como foi implementado o Gerenciamento Contínuo de Riscos e Oportunidades no RUP-VC atende aos objetivos deste elemento-chave.	4	4					1
<b>2.9</b> A forma como foi implementada a Engenharia Simultânea de Sistema e de Software no RUP-VC atende aos objetivos deste elemento-chave.	2	5	1				1
<b>2.10</b> A forma como foram implementados o Monitoramento e o Controle Baseados em Valor no RUP-VC atende aos objetivos deste elemento-chave.	3	5					1
<b>2.11</b> A forma como foi implementada a Mudança como Oportunidade no RUP-VC atende aos objetivos deste elemento-chave.	2	5	1			1	
<b>2.12</b> A aplicação dos elementos-chave da VBSE ao RUP aperfeiçoou o processo de software no sentido de obter ênfase na criação de valor em projetos que utilizem o RUP-VC.	2	4	1				2

---

**Proposição 2.1:** O tema escolhido é importante e oportuno.

---

Sete pessoas concordaram inteiramente com esta proposição.

Dois dos respondentes responderam  $C_0$ , equivalendo a “concordo”. Um deles argumentou que todos os projetos, mesmo os não interessantes, criam valor ao facilitar o dia-a-dia do usuário. O outro afirmou que a ênfase na criação de valor não passa somente pela revisão das práticas e princípios da engenharia de software, tendo consequências na organização bem além desses pontos.

Quanto ao primeiro argumento, não há discordância em que o objetivo de todo projeto é criar algum tipo de valor para os seus clientes. Foi exatamente isso o que se afirmou na discussão apresentada na Seção 2.5.1 deste trabalho. Entretanto, o que se definiu na explicação da proposição como “projetos interessantes para uma organização” foi qualquer projeto que a organização espere que vá lhe trazer algum benefício, ou seja, criar valor. Nesse sentido, caso o projeto não fosse interessante não seria selecionado pela organização. Portanto, o que torna o tema importante e oportuno é a possibilidade de aumentar a *ênfase* na criação desse valor no gerenciamento dos projetos, visando garantir que os mesmos tragam os benefícios esperados pelas organizações.

Com relação ao segundo argumento, também não há discordância entre a proposição e o que foi argumentado pelo respondente, pois, quando a explicação da proposição afirma que “em projetos de software, a ênfase na criação de valor passa pela revisão das práticas e princípios da engenharia de software”, não está sendo dito que a ênfase passa *apenas* pela revisão de práticas e princípios; afirma-se que tal revisão é necessária, e não que ela é suficiente.

---

**Proposição 2.2(A):** A mudança de foco do gerenciamento de projetos, saindo da criação de um produto ou serviço, para a criação de valor para a organização patrocinadora, demanda que a gerência de projetos suba a um patamar estratégico nas organizações.

---

Quatro respondentes concordaram totalmente com a proposição.

Houve três respostas “concordo” (opção  $C_0$ ). Dois desses respondentes argumentaram que a seleção de projetos e programas com ênfase na criação de valor levará em conta questões estratégicas, e que os objetivos dos projetos estarão alinhados com a



estratégia organizacional definida pela gerência sênior. Outro respondente afirmou que atualmente, mesmo sem a ênfase em criação de valor, a gerência de projetos necessita de apoio da alta gerência da organização. Quanto a este último argumento, deve-se ter em mente a diferença entre “gerência de projetos com apoio da alta gerência” e uma efetiva participação estratégica dos projetos. É ponto pacífico que todos os projetos conduzidos em uma organização precisam contar com o apoio da alta gerência, pois a princípio qualquer projeto visa contribuir com os objetivos de negócios. O que se propõe com a ênfase em criação de valor é que haja um verdadeiro alinhamento dos projetos, não somente na seleção destes com base nos objetivos estratégicos, mas também no acompanhamento e monitoramento coordenado entre os projetos de uma organização, através de programas e portfólios de projetos.

Duas respostas concordaram parcialmente (C-) com a proposição. Um desses respondentes entendeu que a proposição significava que os gerentes de projeto deveriam subir de nível no organograma da organização, por isso concordou apenas parcialmente com a proposição. O que se queria dizer realmente com a afirmação é que a atividade de gerência de projetos deve se aproximar e se alinhar com as decisões estratégicas tomadas pela cúpula da organização.

O outro respondente que concordou parcialmente (C-) expôs: “Acredito que a real mudança deva ser a aceitação da ideia de criação de valor no contexto do desenvolvimento de um produto/serviço”, referindo-se à mudança de foco mencionada na proposição. Neste ponto cabe reafirmar que a mudança de foco com vistas à ênfase na criação de valor vai além do contexto do desenvolvimento de um produto ou serviço, ou um conjunto deles. Pode ser que, visando maximizar o valor total gerado, a organização decida que os produtos/serviços concebidos e produzidos pelo projeto devam ser consideravelmente diferentes dos originalmente propostos. Muitos projetos de software enquadram-se neste cenário, dada a dificuldade em definir *a priori* as características dos produtos a serem entregues.

---

**Proposição 2.2(B):** O conceito de valor criado por um projeto considerado neste trabalho é adequado à função estratégica dos projetos.

---

Seis respondentes concordaram totalmente com esta proposição.

Houve duas respostas C<sub>0</sub> (“concordo”) e uma resposta C- (“concordo parcialmente”).

As duas respostas C<sub>0</sub> argumentaram no sentido de que, embora possa ser analisado, valor é algo relativo. Embora se tenha em mente que o conceito de valor definido neste trabalho é apenas um dos inúmeros possíveis conceitos válidos, a proposta em questão teria que partir de algum conceito bem definido para estruturar uma análise do alinhamento do *framework* de processo de software com a ênfase em criação de valor. Procurou-se trazer para a definição do conceito de valor utilizados elementos considerados essenciais ao pensamento estratégico empresarial: retorno financeiro; benefícios de negócios (tangíveis ou intangíveis); e oportunidades, isto é, riscos positivos de negócios, a que os projetos estejam submetidos.

O respondente que concordou parcialmente (C-) argumentou que o conceito de valor criado por um projeto, tal como definido, não é adequado somente à função estratégica, devido à presença do aspecto ROI (retorno sobre investimento). O argumento não apresenta necessariamente discordância com a proposição, pois não se quis dizer que somente a função estratégica se alinha com o conceito de valor definido neste trabalho.

---

**Proposição 2.3:** Foi acertada a escolha do RUP como base para análise e extensão propostas neste trabalho, de modo a integrar considerações de valor a projetos de desenvolvimento de software e dar ênfase à criação de valor nesses projetos.

---

Esta proposição recebeu total concordância de quatro respondentes.

Um dos respondentes não manifestou opinião (opção SO) a respeito da proposição. Outros três concordaram (opção C<sub>0</sub>) e outro concordou parcialmente (C-).

Dentre os que se manifestaram pela concordância, um deles não apresentou seus argumentos. Os outros dois apresentaram os seguintes argumentos:

- “Vê-se que o RUP vem se tornando um padrão de mercado, o que incentiva a sua melhoria contínua”;
- “O RUP é bastante utilizado, assim, analisá-lo com ênfase em valor agregado é muito interessante”.

Aparentemente tais argumentações corroboram a proposição, ao invés de apresentar restrições à mesma. Daí pode-se concluir que há incoerência entre a opção escolhida para resposta e a justificativa dada para a atribuição.

O respondente que concordou parcialmente (opção C-) apontou que a escolha é acertada pelo fato de o RUP ser um *framework* consagrado pelo mercado, mas fez a seguinte ressalva: “Deve-se, porém, avaliar metodologias ágeis, que estão começando a ser mais utilizadas na busca por produtividade”. Esta afirmação reflete o crescente interesse pelas metodologias ágeis percebido nos tempos atuais. Em uma recente pesquisa realizada on-line (AMBLER, 2008), com uma amostra formada por 642 participantes, em sua maioria de organizações de desenvolvimento sediadas nos Estados Unidos, 69% dos respondentes afirmaram que suas organizações estão adotando práticas ágeis de desenvolvimento de software. Devido a isto, um dos trabalhos futuros pretendidos, evoluindo o que foi feito nesta pesquisa, é analisar a ênfase em criação de valor em um *framework* ágil de desenvolvimento de software (vide Capítulo 5).

---

**Proposição 2.4:** A Engenharia de Software Baseada em Valor é um *framework* adequado para integrar considerações de valor ao processo RUP.

---

Esta proposição recebeu total concordância (C+) de três respondentes.

Um respondente não emitiu opinião para esta proposição.

Quatro respostas concordaram com a proposição (opção C<sub>0</sub>). Dentre elas, duas não estavam acompanhadas de argumentos. A argumentação para a terceira resposta foi que “o alinhamento com o estratégico da empresa deve ser feito em todas as fases do desenvolvimento do software e não apenas na fase de concepção do sistema”. A resposta para este argumento é: a extensão RUP-VC já prevê que o alinhamento estratégico de um projeto de software que a utilize seja acompanhado durante todo o ciclo de vida do desenvolvimento. Intervenções como a adição da disciplina de Gerenciamento de Portfólio de Projetos e o monitoramento de valor no nível organizacional buscam atender a esta necessidade.

Um respondente concordou parcialmente (C-). Seu argumento foi: “Tenho dúvidas sobre como se aplicaria para as disciplinas Análise e Design, Implementação, Testes e Gerenciamento de Configuração e Mudanças. Será que essas disciplinas são impactadas?”. Perante este argumento faz-se necessária uma discussão detalhada sobre o escopo escolhido para a extensão, o que se refletiu nas características da mesma.

Inicialmente, foi escolhido o RUP como objeto do estudo em criação de valor. Sendo o RUP um processo prescritivo, de natureza disciplinada, grande parte das decisões

que orientam economicamente os projetos que seguem o processo está concentrada no planejamento (e conseqüentemente no monitoramento e controle) do esforço de desenvolvimento. Portanto, seria natural que a maior parte das oportunidades identificadas para intervenção no RUP estivesse concentrada nas disciplinas com maior vínculo com o *front-end* dos projetos – isto é, com aspectos ligados à organização que executa os projetos e ao seu negócio, como a governança dos projetos, liderança, estratégia, gestão de valor e de benefícios. De fato, os pontos de extensão do RUP-VC estão distribuídos entre as disciplinas:

- Gerenciamento de Projeto: na qual o aspecto técnico da Engenharia de Software tem interface com os aspectos organizacionais operacionais. Em relação à criação de valor, tem o objetivo de garantir o cumprimento das metas tradicionais de sucesso do projeto (cronograma, custo, funcionalidades).
- Requisitos: cuja importância reside no fato de que suas atividades devem trazer à tona os múltiplos critérios de sucesso do sistema, dos diversos interessados, a fim de obter condições ganha-ganha que maximizem o valor criado pelo produto do projeto;
- Ambiente: que inclui as tarefas que adaptam o processo às exigências e restrições do projeto específico, por conseguinte determinando as características que farão o processo mais ou menos adaptável a mudanças, como também mais ou menos capaz de lidar com ameaças e oportunidades.
- Gerenciamento de Portfólio: criada para a extensão RUP-VC, a disciplina propõe lidar com as questões organizacionais de mais alto nível, como estratégia, priorização de recursos, interdependências entre projetos (de software ou não) para a consecução de objetivos de negócios.

Quanto à abrangência e natureza das intervenções que formam o RUP-VC, partiu-se do princípio que a criação de valor deveria ser uma preocupação constante nas diversas áreas das disciplinas afetadas. Considerou-se que essa visão está de acordo com a motivação principal da VBSE (tomada como diretriz da análise), que é incorporar considerações de valor às técnicas e práticas de Engenharia de Software já existentes. Sendo assim, na extensão RUP-VC procurou-se intervir de forma que fosse preservada a estrutura do *framework*. As tarefas, artefatos e atribuições sofreram intervenções pontuais, tendo sido acrescentado ao seu

conteúdo o que se considerou serem as lacunas para o atendimento dos elementos-chave. A exceção foi a criação da disciplina de Gerenciamento de Portfólio, com todo o conjunto de tarefas, artefatos, atribuições e diretrizes: uma intervenção de maior impacto, considerada necessária para satisfazer a vinculação do projeto de desenvolvimento com outros projetos ou com a estratégia da organização, dado que no RUP original não há tal preocupação.

---

**Proposição 2.5:** A forma como foi implementada a Análise de Realização de Benefícios no RUP-VC atende aos objetivos deste elemento-chave.

---

Esta proposição recebeu duas respostas concordando totalmente.

Cinco respondentes concordaram (C<sub>0</sub>). Um deles não emitiu argumentos; os quatro restantes emitiram argumentos que reforçam a adequação das intervenções ao elemento-chave Análise de Realização de Benefícios:

- “O uso da Cadeia de Resultados, em especial, traz mais objetividade à análise”;
- “A ideia de um acompanhamento conjunto dos projetos e seus inter-relacionamentos facilitam a visão estratégica da alta gerência”;
- “As atividades previstas na disciplina de Gerenciamento de Portfólio permitem os benefícios de negócios sejam identificados não apenas na fase de Seleção como também durante toda a execução do projeto”;
- “A inclusão da disciplina Gerenciamento de Portfólio pareceu adequada juntamente com seu uso com a técnica Cadeia de Resultados. Parece ser uma boa estratégia para analisar benefícios”.

Um respondente concordou parcialmente (C-). Este respondente questionou sobre como a organização que adotasse o RUP-VC poderia proceder para atender ao elemento-chave caso não tenha condições de implementar a disciplina de Gerenciamento de Portfólio. Neste caso, sem uma gestão adequada do portfólio de projetos, a organização fica deficiente na definição e na execução da sua estratégia de negócios. A Cadeia de Resultados, cuja inclusão na seleção de projetos foi outra intervenção feita a fim de atender ao elemento-chave, depende de uma gestão estruturada do portfólio de projetos, portanto sua adoção sem a implantação da disciplina de Gerenciamento de Portfólio não surtiria os efeitos desejados.

Por fim, um respondente não emitiu opinião (SO).

---

**Proposição 2.6:** A forma como foram implementadas a Identificação e a Conciliação de Proposições de Valor das Partes Interessadas no RUP-VC atende aos objetivos deste elemento-chave.

---

Esta proposição teve quatro respostas concordando totalmente.

Três respondentes concordaram (C<sub>0</sub>). Um deles não apresentou argumentos para sua resposta. Os outros argumentos foram:

- “Ao fazer tais atividades, ataca-se de forma mais célere o que realmente importa para o comprador do sistema em desenvolvimento”;
- “As tarefas propostas abrangem tanto a parte de planejamento quanto na fase de elaboração dos casos de uso, além de solicitar a avaliação da iteração com relação aos benefícios atingidos”.
- “A proposta é ótima, só não sei a efetividade da tarefa ‘Desenvolver a Visão’. É sempre difícil chegar a um nível de abstração adequado. No fim das contas, contamos com a maturidade do cliente...”.

Sobre este último argumento, cabe ressaltar que a tarefa *Desenvolver a Visão*, responsável por obter o Documento de Visão, é a maneira que o RUP oferece para que as partes interessadas de um projeto de desenvolvimento cheguem a uma visão unificada do produto (software) do projeto. É essencial, então, trabalhar no fluxo desta tarefa a fim de obter uma visão do sistema pactuada entre os principais interessados. Isto influi diretamente no valor criado pelo produto final do projeto. Aumentando a capacidade de levantamento dos requisitos, em especial das expectativas e proposições de valor, e sistematicamente detectando e atacando conflitos entre as expectativas dos diversos interessados, a conciliação torna-se possível.

Por fim, houve uma resposta concordando parcialmente (C-) com esta proposição, na qual o respondente argumentou: “Faltou feedback constante do cliente”, referindo-se a uma possível lacuna na extensão RUP-VC, relativa a obter feedback do cliente quanto às proposições de valor levantadas. O argumento levantado é pertinente, pois a conciliação das

diversas proposições de valor, levantadas entre as diversas partes interessadas – usuários, patrocinadores, equipe de desenvolvimento etc. – só será efetiva para a criação de valor se houver feedback positivo quanto aos resultados do levantamento. Em resposta a este argumento, deve-se observar que a extensão RUP-VC oferece as seguintes propostas de melhoria ao RUP:

- O uso de software colaborativo (*groupware*) em workshops de levantamento de requisitos fornece meios de ampliar o feedback quanto às funcionalidades esperadas ou desejadas, através da ampliação do número de envolvidos participantes dos workshops;
- A criação de um passo adicional, no fluxo da tarefa *Desenvolver a Visão*, para conciliar as expectativas das partes interessadas oferece uma maneira de provocar a discussão e a resolução de conflitos porventura existentes entre requisitos, expectativas dos envolvidos. O feedback acerca de conflitos resolvidos é fundamental para a maximização do valor a ser criado pelo sistema;
- O RUP prevê que, ao longo do projeto, no planejamento do escopo de cada iteração os casos de uso definidos sejam analisados e priorizados de acordo os riscos envolvidos, com a cobertura (abrangência já atendida) e com a criticidade (embutido nesse critério o conjunto conciliado de proposições de valor de cada caso de uso para os interessados). Esta priorização deve idealmente envolver os mesmos interessados, pois alguns casos de uso podem necessitar de esclarecimentos quanto aos critérios; agindo assim, tem-se mais uma instância de feedback do cliente.

---

**Proposição 2.7:** A forma como foi implementada a Análise de Caso de Negócio no RUP-VC atende aos objetivos deste elemento-chave.

---

Dois respondentes concordaram totalmente com esta proposição.

Seis respondentes deram resposta “concordo”. Um deles não apresentou argumentos para a resposta. Outros argumentos apresentados foram:

- “Achei bastante adequada a abordagem da justificativa econômica do produto”;

- “As tarefas de ‘Descrever Opções’, ‘Descrever as Limitações do Projeto’ e na ‘Desenvolver a Previsão Financeira’ permitem a análise dos riscos, benefícios e ROI”;
- “Isto irá facilitar o direcionamento do sistema, já que nas opções haverá um maior aprofundamento na análise para a escolha da que seja mais prolífera em termos de custo/benefício para a empresa”;
- “Os inputs para a construção do Plano de Negócio é nenhum? Achei que poderia ser, por exemplo, a Cadeia de Resultados”.

Este último argumento traduz-se em uma sugestão de melhoria: a adição da Cadeia de Resultados como artefato de entrada para a atividade *Desenvolver Caso de Negócio*. Essa sugestão foi considerada pertinente, e suscitou uma revisão das intervenções feitas no RUP, de modo a verificar se a adição proposta pelo respondente seria pertinente. De fato, na verificação concluiu-se que, havendo um ambiente organizacional que propicie a seleção e o acompanhamento dos projetos em nível estratégico, em que a análise para delimitar o escopo dos vários programas e projetos do portfólio da organização e suas interdependências utilize a Cadeia de Resultados como ferramenta, a análise de negócios dos projetos propostos será uma decorrência natural dessas atividades. Portanto, como melhoria da extensão RUP-VC proposta, foi incluída a Cadeia de Resultados como subsídio na definição do contexto do projeto e dos seus objetivos – passos 2 e 3 da tarefa *Desenvolver Caso de Negócio*, respectivamente. À descrição da tarefa revista foi acrescentado o sufixo “(revisada após avaliação)”.

Um respondente concordou parcialmente (C-). Seu argumento: “Concordo com as etapas sugeridas, mas falta definir sucesso e critérios de sucesso utilizados. A criação de valor é baseada nos critérios de sucesso?”. Os termos “sucesso” e “critérios de sucesso” referem-se ao resumo explicativo da proposição, que afirma “A análise de Caso de Negócio consiste em basear os critérios de sucesso do projeto em um caso de negócio que leve em consideração a criação de valor”. Esclarecendo esta afirmação, deve-se dizer que o caso de negócio de um projeto deve conter a fundamentação (ou justificativa) econômica que leva a considerar o projeto interessante de ser executado pela organização. A Análise de Caso de Negócio, portanto, resulta em explicitar os critérios de sucesso no documento de caso de negócio. Se o documento resultante inclui critérios de sucesso que envolvam não somente retorno



financeiro, mas também benefícios intangíveis e riscos/oportunidades, está sendo dada ênfase à criação de valor.

---

**Proposição 2.8:** A forma como foi implementado o Gerenciamento Contínuo de Riscos e Oportunidades no RUP-VC atende aos objetivos deste elemento-chave.

---

Quatro respondentes concordaram totalmente com a proposição.

Outros quatro responderam “concordo”, representando opção C<sub>0</sub>. Destes, apenas um apresentou argumento para sua resposta: “Acho que um dos pontos-chave é a tarefa de adaptar o processo ao projeto, refletindo a expectativa do cliente”. O respondente refere-se ao ponto de extensão nº 9, “Adaptar o processo de software considerando a exposição ao risco” do RUP-VC, no qual as decisões a respeito do caráter mais ou menos ágil do processo expõem a organização a riscos que podem causar impactos na qualidade do produto de software ou no seu tempo de entrega.

Um respondente não proferiu opinião sobre a proposição.

---

**Proposição 2.9:** A forma como foi implementada a Engenharia Simultânea de Sistema e de Software no RUP-VC atende aos objetivos deste elemento-chave.

---

Dois respondentes concordaram plenamente (C+) com a proposição.

Cinco respondentes concordaram (C<sub>0</sub>). Dois deles não apresentaram os argumentos para a resposta. Os outros argumentaram assim:

- “A correção de rumo trazida por uma revisão formal, adicionada com valores, aumenta de forma considerável a aderência do futuro sistema às necessidades da empresa”;
- “Também considero ‘Revisão dos Marcos de Ciclo de Vida’ como sendo um ótimo momento para realizar a sincronização das atividades”;
- “É fundamental ficar atento para que esta atividade importante para o planejamento estratégico dos sistemas no âmbito da organização não deixe o RUP muito pesado e complexo”.

Os dois primeiros argumentos expressam concordância com a proposição. Já o último expressa preocupação a respeito de um possível aumento de complexidade que possam trazer as intervenções referentes ao elemento-chave ora discutido.

Um respondente concordou parcialmente (C-) com a proposição. O argumento apresentado para a resposta foi: “A avaliação de valor em marcos mais estruturais talvez não seja aplicável, por exemplo, na construção do ambiente de execução (ainda não há ‘produto’ funcional)”. Em resposta a este argumento, cabe lembrar que o RUP prevê que, desde as primeiras iterações da fase de iniciação, os objetivos das iterações sejam mensurados, mesmo antes de ser lançado o primeiro release em produção.

Um respondente não proferiu opinião sobre a proposição.

---

**Proposição 2.10:** A forma como foram implementados o Monitoramento e o Controle Baseados em Valor no RUP-VC atende aos objetivos deste elemento-chave.

---

Esta proposição recebeu total concordância de três respondentes.

Cinco respondentes concordaram. Três deles não justificaram suas respostas. Os outros dois argumentaram:

- “A Fábrica de Experiências traz conceitos de gestão de conhecimento bastante relevantes”;
- “Com a criação do guideline ‘Fábrica de experiência baseada em valor’ é possível realizar um monitoramento baseado em valor”.

Os argumentos em seu teor validam a decisão de escolha da Fábrica de Experiência como uma contribuição importante para o *framework* de processo, no que diz respeito ao monitoramento baseado em valor.

Um respondente não manifestou opinião. Acrescentou o seguinte argumento: “Concordo que outras considerações de valor além do EVM devam ser utilizadas, mas não consegui visualizar onde a fábrica de experiência poderia auxiliar nesta tarefa. A fábrica de experiência melhora a visualização, mas onde agrega valor?”. Cabe comentar esta argumentação. Conforme apresentado na Seção 3.3.6, a Fábrica de Experiência baseada em valor é uma abordagem que auxilia no monitoramento e aperfeiçoamento de projetos de

software inseridos no contexto de um portfólio de projetos organizacional. O *website* do RUP-VC inclui uma diretriz específica para a Fábrica de Experiência baseada em valor, contendo um roteiro genérico de uso da abordagem, onde fica evidenciada a intenção da abordagem.

---

**Proposição 2.11:** A forma como foi implementada a Mudança como Oportunidade no RUP-VC atende aos objetivos deste elemento-chave.

---

Dois respondentes concordaram totalmente com a proposição.

Cinco respondentes concordaram com a proposição, mas apenas dois manifestaram justificativas para suas respostas. São elas:

- “Adaptar o processo é essencial”;
- “Concordo que seja durante a tarefa ‘Adaptar o processo de desenvolvimento para o projeto’. O melhor momento para personalizar o processo de forma a tratar mudança como oportunidade”.

Um respondente concordou parcialmente (C-) com a proposição. Seus argumentos: “Precisa considerar também os custos da mudança. Se já considera isso, pode colocar como concordo totalmente”. Esta argumentação externa preocupação com os custos das mudanças a considerar como oportunidades. Aqui cabe lembrar que, a partir do momento em que uma mudança no mercado, no negócio ou na organização é identificada pela organização como uma oportunidade, portanto um tipo de risco positivo, ela deve ser gerenciada como tal. Assim, a mudança deve ser submetida à tarefa *Identificar e Avaliar Riscos* do RUP, onde seu impacto é estimado. Portanto, quaisquer desvios de esforço, prazo ou custo em relação ao planejado, provocados pela mudança, devem ser determinados.

Um respondente discordou parcialmente (D-) da proposição, com a argumentação: “Não consegui visualizar agilidade no processo”. Deduz-se que o respondente assumiu que a extensão RUP-VC tornaria o RUP mais ágil. É importante esclarecer que o que realmente foi proposto na extensão é a estratégia de adaptar o processo a ser seguido, sempre que o projeto, ou o ambiente no qual está inserido, tiver a característica de mudar frequentemente. As decisões sobre quais adaptações específicas serão feitas, e sobre qual nível de agilidade terá o processo resultante, dependem de vários fatores. Esse assunto é abordado de forma detalhada na diretriz do RUP: “Discriminadores do Processo” (IBM, 2006).

**Proposição 2.12:** A aplicação dos elementos-chave da VBSE ao RUP aperfeiçoou o processo de software no sentido de obter ênfase na criação de valor em projetos que utilizem o RUP-VC.

---

Dois respondentes concordaram totalmente com a proposição.

Quatro respondentes concordaram com a proposição, assinalando a opção C<sub>0</sub>. Dois deles não apresentaram argumentos. Os outros argumentos foram:

- “Em geral, vejo a proposta como uma forma de viabilizar a criação de valor em projetos”;
- “Todas as práticas e técnicas identificadas como importantes pela VBSE foram atendidas de forma satisfatória na extensão do RUP”.

Um respondente concordou parcialmente (C-) com a proposição, e justificou dizendo “É uma afirmativa difícil de se fazer sem o teste prático, mas considero uma proposta bem embasada, levando em consideração sumidades na área como Boehm”. A proposição também teve duas respostas “Sem opinião”, porém os respondentes emitiram argumentos. São eles:

- “Só na prática para avaliar”;
- “Não é possível para a respondente assumir que o processo de software foi aperfeiçoado com a extensão. Seriam necessários estudos de caso com empresas utilizando RUP-VC e reportando a obtenção de valor através dele”.

Estas argumentações trazem à baila as limitações do processo de avaliação escolhido. O ideal seria um experimento contendo estudos de caso com projetos e portfólios organizacionais reais, onde as tarefas estendidas seriam exercitadas e resultados de seu desempenho coletados. Entretanto, devido aos aspectos que serão discutidos na Seção 4.6 a seguir, o caráter essencialmente experimental da avaliação foi limitado à submissão da extensão às opiniões de profissionais experientes.

#### **4.5.1 Conclusões da Análise**

Do exposto nas seções anteriores, os resultados levam a concluir que:

Como visto na seção anterior, a maioria das respostas da amostra demonstrou concordância com a extensão proposta. Houve apenas uma resposta discordante. Pelos resultados pode-se concluir que os respondentes:

- Consideram a engenharia de software baseada em valor um tema importante e oportuno
- Consideram a VBSE um framework adequado para integrar considerações de valor à engenharia de software;
- Consideram acertada a escolha do RUP como processo-base para a extensão;
- Consideram que a forma como foram implementados os elementos-chave no RUP atende aos objetivos dos respectivos elementos.

No entanto, alguns problemas de validade foram verificados na avaliação:

- Os instrumentos utilizados, o workshop de apresentação da VBSE mais o questionário, não conseguem refletir todas as características da extensão, que é extensa. Isso impõe um risco de construção à avaliação, pela sub-representação da proposta a ser avaliada.
- As limitações impostas ao desenho da avaliação podem trazer riscos de validade interna, no sentido de que o tamanho da amostra resultante não foi grande o suficiente para que determinadas características da experiência dos participantes – diferenças entre tipos de sistema, tamanho do projeto e organização na qual trabalhou, por exemplo – não fossem possíveis de ser avaliadas.

#### **4.6 Implantação das Melhorias**

Como visto na seção anterior, as respostas aos questionários foram em sua maioria de concordância com a extensão proposta, tendo sido obtidas poucas sugestões objetivas para melhoria da mesma.

A única intervenção resultante da análise das respostas foi referente à proposição ao

elemento-chave Análise de Caso de Negócio, onde foi inserida a vinculação com a Cadeia de Resultados, como entrada opcional da tarefa. A Cadeia de Resultados é utilizada nos passos 2 e 3 da tarefa, nos quais se definem o contexto de negócios onde o projeto será executado, e os objetivos do produto final.

#### **4.7 Considerações Finais**

Percebeu-se ser difícil testar um processo de software como a extensão RUP-VC de forma experimental, através de um estudo de caso, dentro do prazo disponível para a elaboração de uma dissertação de Mestrado. Outro obstáculo encontrado impedindo tal intuito foi a dificuldade de se encontrar uma organização “do mundo real” interessada em testar um processo não experimentado em seus projetos. Sabe-se que um dos critérios de sucesso da adoção de um processo de software é o interesse da organização e da sua equipe de gerenciamento em fazê-lo (BËRGSTORM; RÄBERG, 2004).

Estes entraves levaram à decisão de adotar outro método para a avaliação do trabalho. O método, baseado em pesquisa de opinião após workshops de apresentação da proposta, por um lado cumpre de certa forma o objetivo de expor o RUP-VC à prática, devido aos requisitos exigidos para o público-alvo: experiência em gestão de projetos de software com RUP, ou experiência em desenvolvimento de software com RUP.

Por outro lado, uma sondagem preliminar com alguns representantes do público-alvo revelou ser inviável obter participação dos convidados caso fosse ministrado um curso completo sobre os temas abordados – VBSE, alinhamento estratégico, e o conteúdo do RUP-VC – para os participantes, devido à pouca disponibilidade de tempo para a participação, típica do perfil profissional do público-alvo. Para contrabalançar esta deficiência, a solução encontrada foi publicar o conteúdo do RUP-VC em um website para possibilitar consulta ao conteúdo durante a resposta ao questionário. Considera-se que essas medidas auxiliaram a conseguir obter a amostra, dado o prazo disponível para a realização da avaliação.

## 5 Conclusões e Trabalhos Futuros

Este Capítulo tem como objetivo relatar as principais contribuições desta dissertação, trabalhos relacionados, trabalhos futuros que poderão ser realizados a partir desta proposta, e considerações finais sobre o desenvolvimento deste trabalho.

### 5.1 Principais Contribuições

Esta dissertação abordou a questão de como o Rational Unified Process lida com a criação de valor no desenvolvimento de software. Seu principal objetivo é identificar e analisar pontos que favorecem a criação de valor, chamados pontos de convergência, e mapear lacunas que podem ser preenchidas com melhorias, ou pontos de extensão. As principais contribuições deste trabalho foram:

- Avançar na discussão da criação de valor no âmbito da Engenharia de Software. Com a análise e a extensão de um *framework* de processo de mercado realizadas neste trabalho, pretende-se suscitar a discussão sobre o grau de importância que a criação de valor assume nas metodologias atualmente utilizadas nos projetos de desenvolvimento de software. Em especial, procurou-se introduzir a preocupação com considerações de valor, e não apenas técnicas, nas tomadas de decisões necessárias tanto à gestão dos projetos de desenvolvimento de produtos que contenham software como parte essencial, quanto ao design dos produtos resultantes.
- Verificar a aplicabilidade dos elementos-chave da VBSE em um processo de software de mercado. A abordagem VBSE, tomada como diretriz para a análise do RUP quanto à ênfase que ele apresenta em relação à criação de valor, sugere através de seus elementos-chave diversas propriedades e características que uma metodologia ou abordagem de desenvolvimento deve possuir para considerar a criação de valor como um de seus objetivos. Buscou-se nesta pesquisa de mestrado obter uma amostra do tipo de esforço que é necessário para evidenciar tornar os sete elementos-chave da VBSE aplicáveis em uma metodologia consagrada, através de uma extensão. Além disso, através da avaliação da extensão criada, procurou-se ter um feedback de profissionais usuários da metodologia a respeito

da aplicabilidade da VBSE.

## 5.2 *Trabalhos relacionados*

A integração de uma perspectiva baseada em valor à Engenharia de Software vem recebendo crescente atenção no meio acadêmico. A literatura aponta para diversos trabalhos que se relacionam com a pesquisa aqui apresentada pelo fato de também proporem intervenções nas práticas e técnicas da Engenharia de Software, entre eles:

- O processo de software MBASE (Model-Based Architecting and Software Engineering) (BOEHM; PORT, 2001) dá suporte a alguns elementos-chave da VBSE, como a Identificação e Conciliação de Proposições de Valor dos Interessados usando a gestão de requisitos da Teoria W; Engenharia Simultânea de Sistemas e Software, apoiada na revisão do projeto nos marcos de ciclo de vida; e Análise de Caso de Negócio, através do estudo de viabilidade econômica.
- O processo VBSDA (Value-Based Software Dependability Achievement) (HUANG, 2005), construído a partir do modelo WinWin em espiral (BOEHM, 1988), e utilizado para sistemas com altas exigências de confiabilidade. Alinha-se aos elementos-chave de Análise de Realização de Benefícios através da modelagem da Cadeia de Resultados; Identificação e Conciliação de Proposições de Valor dos Interessados usando a gestão de requisitos da Teoria W; Engenharia Simultânea de Sistemas e Software, apoiada na revisão do projeto nos marcos de ciclo de vida; Gerenciamento Contínuo de Riscos e Oportunidades e Mudança como Oportunidade, através do monitoramento de mudanças e análise de riscos e oportunidades resultantes das mudanças.
- A abordagem de *avaliação de valor* para processos e produtos de software, definida por Ojala (OJALA, 2008), baseada na Engenharia de Valor, técnica utilizada na indústria que é um esforço organizado para atingir o valor ótimo de um produto, sistema ou serviço, promovendo as funções necessárias ao menor custo (BROWN, 1992). A abordagem implica avaliar custos e benefícios de cada processo de software em uso (que siga ou não algum modelo ou framework), e também dos produtos de software (ou seus requisitos, quando ainda não



existentes), visando a melhoria do seu valor (definido na abordagem em termos dos benefícios e custos).

Diversos trabalhos recentes propõem aplicações de técnicas e práticas baseadas em valor para a Engenharia de Software. Dentre eles estão:

- Huang et al. (2006) apresentam uma aplicação de um processo de software, voltado à produção de software de qualidade a partir dos requisitos das partes interessadas, em um projeto de desenvolvimento de um sistema ERP<sup>19</sup> na China;
- Egyed (2006) apresenta um estudo comparativo entre rastreabilidade manual e automatizada de artefatos de software, e explora formas de ajustar o rastreamento às necessidades baseadas em valor, adequando fatores como corretude, completude e tempestividade;
- Dingsoyr (2006) discute dois métodos de gestão do conhecimento voltados para empresas de software: revisões *postmortem* e *workshops* de processo. Tais métodos são orientados às pessoas – onde reside o conhecimento mais valioso do negócio de desenvolvimento de software – e baseados em valor;
- Reifer (2006) discute as abordagens atuais usadas para valorar propriedade intelectual de software, delinea um framework de valoração que permite que especialistas avaliem todas as formas de ativos intangíveis de software, e discute possíveis obstáculos na aprovação dos resultados da valoração por decisores não especialistas em software (executivos, investidores, juristas etc.);
- Chen (2007) define um framework para priorização da análise de vulnerabilidades e ameaças aos modelos de segurança de componentes de software COTS<sup>20</sup>. O framework se propõe a ser sensível às proposições de valor das partes interessadas;

---

<sup>19</sup> ERP é um acrônimo para *Enterprise Resource Planning*, ou planejamento de recursos empresariais.

<sup>20</sup> COTS é um acrônimo para *Commercial Off-The-Shelf*, ou (componentes de software) comerciais de prateleira.

- Cantone e Kruchten (2008) discutem a aplicação de princípios de engenharia de software baseada em valor na DDRD<sup>21</sup>, e apresentam um experimento analisando a viabilidade do método proposto.

### 5.3 Trabalhos Futuros

O escopo do trabalho ora apresentado foi analisar o RUP com relação à criação de valor, tomando a VBSE como diretriz, e estendê-lo nos pontos onde não fosse alinhado aos sete elementos-chave propostos pela abordagem. Algumas oportunidades de pesquisa podem derivar deste trabalho. Dentre elas estão os seguintes:

Pesquisar pontos de extensão em outras disciplinas. A abrangência desta pesquisa foi delimitada pelos sete elementos-chave da VBSE, que têm aplicação mais direta em algumas das disciplinas do RUP, como discutido na Seção 4.5. Uma possibilidade de trabalho futuro é analisar as outras disciplinas do RUP – Análise e Design, Implementação, Testes e Gerenciamento de Configuração e Mudanças – quanto ao seu alinhamento aos elementos-chave da VBSE. A literatura aponta para alguns trabalhos nesse sentido:

- Testes: Ramler *et al.* (2006) investigam a importância de testes baseados em valor, descrevendo práticas e esboçando um *framework* de processo para o gerenciamento de testes baseados em valor. Srikanth e Williams (2005) propõem a técnica PORT<sup>22</sup>, para priorização de casos de testes em nível de sistema, visando aumentar a qualidade percebida do software sem adicionar *overhead* significativo para a equipe;
- Gerenciamento da Qualidade: Huang e Boehm (2006) apresentam uma abordagem para auxiliar na decisão sobre quanto investir em qualidade de software nos projetos, baseando-se nos modelos COCOMO II, para estimativas de custo, e COQUALMO (STEECE *et al.*, 2002), para estimativas de qualidade.

---

<sup>21</sup> DDRD é um acrônimo para *Design Decisions Rationale Documentation*, ou documentação das razões para as decisões de design (de software).

<sup>22</sup> PORT é um acrônimo para *Priorization of Requirements for Testing*, ou priorização de requisitos para testes.

- Arquitetura e Design: Sullivan *et al.* (1999) propõem uma visão integrada na qual o design de software é em grande parte um processo de tomada de decisões sob conhecimento incompleto e incertezas, dentre elas as ameaças de entradas de novos competidores. KIM et al. (2007) propõem a técnica LiVASAE<sup>23</sup> de avaliação de arquiteturas de software baseada em valor, cujas maiores vantagens são: ter um processo leve, e permitir apoiar um processo de decisão com critérios objetivos e subjetivos

Fazer análise semelhante em uma metodologia ágil. Considerou-se como uma evolução natural desta pesquisa a análise de métodos ágeis de desenvolvimento de software com relação à ênfase na criação de valor dada por tais métodos. Pode ser uma investigação interessante confrontar as práticas ágeis com os elementos-chave da VBSE, pois o primeiro princípio do “Manifesto Ágil” (*Agile Manifesto*) é justamente entregar continuamente ao cliente “software de valor” (BECK et al., 2001).

Realizar um estudo experimental. Por fim, sob alguns aspectos a evolução mais importante deste trabalho seria um experimento, tendo como objeto a extensão RUP-VC, realizado sob o método da experimentação em Engenharia de Software (TRAVASSOS *et al.*, 2002). O ideal seria um estudo controlado, em um contexto in-vivo, com público formado por profissionais de gerenciamento e desenvolvimento de software com RUP, e cujos resultados seriam válidos para o contexto particular considerado nesta pesquisa. Assim, os fatores que modelam o valor, como aversão ao risco, proposições de valor, oportunidades de negócio, quantidade de “burocracia” do processo adotado etc., poderiam ser observados e de alguma forma mensurados de modo a analisar sua influência nos projetos executados. Mais especificamente, o experimento seria um estudo de caso, onde o RUP-VC fosse aplicado em projetos de software de uma organização, e os resultados desses projetos – com relação ao valor criado por eles – fossem comparados com outros projetos simultâneos que seguissem a metodologia usual da organização, ou com informações históricas de projetos passados.

---

<sup>23</sup> LiVASAE é um acrônimo para *Light-weight Value-based Software Architecture Evaluation*, ou avaliação de arquitetura de software baseada em valor.

## 5.4 Considerações Finais

O estudo da criação de valor é um campo emergente de pesquisa, tanto na área de conhecimento de Gerenciamento de Projetos, como na Engenharia de Software. As organizações estão cada vez mais exigentes quanto aos múltiplos aspectos dos benefícios estratégicos que esperam das suas iniciativas de negócios. Com relação aos projetos que envolvem Tecnologia da Informação, as exigências muitas vezes se aliam a uma maior ênfase no controle, devido ao histórico frequentemente não favorável do desempenho desse tipo de projeto nas empresas.

Percebe-se então uma necessidade de evolução das ferramentas para melhorar a questão do desempenho e da realização de benefícios, como o alinhamento estratégico de projetos e a governança de projetos e de TI. Na Engenharia de Software, a criação de valor tem merecido atenção dos pesquisadores do campo da Economia de Software, a fim de incorporar às ferramentas e técnicas atuais e emergentes considerações de valor. A Engenharia de Software Baseada em Valor (VBSE) é a abordagem utilizada neste trabalho como diretriz da pesquisa.

Neste trabalho, foi criada a extensão RUP-VC, que visa acrescentar às disciplinas, tarefas e artefatos do RUP original considerações de valor. Como diretriz para a pesquisa dos pontos que seriam estendidos, foram utilizadas as sugestões e recomendações da VBSE, reunidas em sete elementos-chave para criação de valor em desenvolvimento de software. Como suporte teórico adicional, necessário às intervenções feitas ao RUP, foram trazidas contribuições de outras áreas da Engenharia de Software, e também da gestão de projetos, como o PMBOK e os padrões de gerenciamento de programas e portfólio do PMI.

A extensão RUP-VC foi desenvolvida com o cuidado de se manter a consistência do processo após as intervenções. Isso quer dizer que os elementos essenciais do RUP e seus princípios-chave, que são os fundamentos que caracterizam o processo, mantiveram-se válidos também no RUP-VC. Fundamentos como a criação de uma visão compartilhada e do caso de negócio do sistema; a estruturação de uma arquitetura; o caráter iterativo do processo; e a preocupação em dimensionar o processo ao sistema (*tailoring*), foram mantidas e em alguns casos potencializadas pelas intervenções da extensão RUP-VC.

Como forma de submeter o RUP-VC a uma avaliação prática, optou-se por realizar

uma pesquisa de opinião a respeito da extensão, sendo o público-alvo o conjunto de profissionais com experiência em gestão de projetos de desenvolvimento de software, ou com experiência em desenvolvimento usando o RUP, em atuação no Recife. Infelizmente, no período de tempo disponível para a avaliação, muitos dos convidados não puderam comparecer às sessões de workshop nas quais responderiam o questionário. Mesmo assim, as contribuições da pequena amostra foram bastante significativas, tendo sido obtido uma avaliação considerada positiva.

Em suma, espera-se da pesquisa apresentada nesse trabalho, contribuir com uma nova visão dos objetivos e resultados que se deve almejar com o desenvolvimento de software com o RUP. Também se espera que a ênfase na criação de valor torne-se cada vez mais presente nos métodos e práticas da Engenharia de Software.

## Referências

- AAKER, D.; KUMER, V.; DAY, G. S. (2004). *Pesquisa de Marketing*, 2ª Ed. Atlas, São Paulo. 752pp.
- ADM (Advanced Development Methods, Inc.) (2009). 'Scrum (Control Chaos)'. Disponível em <http://www.controlchaos.com>, acessado em janeiro/2009.
- ALBERTIN, A. L. (2001). 'Valor estratégico dos projetos de Tecnologia da Informação', *Revista de Administração de Empresas* v.41(n.3), pp. 42-50.
- AMBLER, S. (2008). 'Agile Adoption Rate Survey: February 2008', Ambysoft. Disponível em <http://www.ambysoft.com/surveys/agileFebruary2008.html>, acessado em abril de 2009.
- AMBLER, S. W.; NALBONE, J.; VIZDOS, M. J. (2005). *The Enterprise Unified Process: Extending the Rational Unified Process*, Prentice-Hall. 384pp.
- BASIL, V.; CALDEIRA, G.; ROMBACH, H. (1994). The Experience Factory. In J. Marciniak (ed.): *Encyclopedia of Software Engineering*, Wiley.
- BASIL, V.; CALDEIRA, G.; ROMBACH, H. (1994). The Goal Question Metric Approach. In: J. Marciniak (ed.): *The Encyclopedia of Software Engineering*, Wiley.
- BECK, K. (2004). *Programação Extrema explicada*, trad. Adriana Machado e Natália Lopes. Bookman, Porto Alegre. 182pp.
- BECK, K.; BEEDLE, M.; BENNEKUM, A. V.; COCKBURN, A.; CUNNINGHAM, W.; FOWLER, M.; GRENNING, J.; HIGHSMITH, J.; HUNT, A.; JEFFRIES, R.; KERN, J.; MARICK, B.; MARTIN, R. C.; MELLOR, S.; SCHWABER, K.; SUTHERLAND, J.; THOMAS, D. (2001). 'Manifesto for Agile Software Development (The Agile Manifesto)'. Disponível em <http://www.agilemanifesto.org/>, acessado em janeiro/2009.
- BËRGSTROM, S.; RÄBERG, L. (2004). *Adopting the Rational Unified Process: success with the RUP*, Addison Wesley. 245pp.
- BIFFL, S.; AURUM, A.; BOEHM, B.; ERDOGMUS, H.; GRUNBACHER, P., eds. (2006). *Value-Based Software Engineering*, Springer-Verlag, Berlim. 388pp.
- BLACKWELL, D.; GIRSHICK, M. (1954). *Theory of Games and Statistical Decisions*. New York, Wiley. 355pp.
- BOEHM, B. (2003). 'Value-Based Software Engineering', *ACM SIGSOFT Software Engineering Notes* 28(2).
- BOEHM, B. (1988). 'A Spiral Model of Software Development and Enhancement', *IEEE Computer* 21(5), pp. 61-72.
- BOEHM, B. (1981). *Software Engineering Economics*, Prentice-Hall. 767pp.

- BOEHM, B.; HUANG, L. (2003). 'Value-Based Software Engineering: A Case Study', *IEEE Computer* **36** (3), pp. 33-41.
- BOEHM, B.; JAIN, A. (2006). An Initial Theory of Value-Based Software Engineering. In Biffl, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 15-37.
- BOEHM, B.; PORT, D. (2001). 'Balancing discipline and flexibility with the spiral model and MBASE', STSC. Disponível em <http://www.stsc.hill.af.mil/crosstalk/2001/12/boehm.html>, acessado em janeiro/2009.
- BOEHM, B.; PORT, D.; AL-SAID, M. (2000). 'Avoiding the software model-clash spiderweb', *IEEE Computer* **33**, pp. 120-122.
- BOEHM, B.; PORT, D.; JAIN, A.; BASILI, V. (2002). 'Achieving CMMI Level 5 improvements with MBASE and the CeBASE method', STSC Crosstalk. Disponível em <http://www.stsc.hill.af.mil/crosstalk/2002/05/boehm.html>, acessado em janeiro/2009.
- BOEHM, B.; ROSS, R. (1989). 'Theory-W Software Project Management: Principles and Examples', *IEEE Transactions on Software Engineering* **15**(7), pp. 902-916.
- BOEHM, B.; SULLIVAN, K. (2000). 'Software Economics: a roadmap', *Proceedings of the Conference on The Future of Software Engineering*, pp. 319-343.
- BOEHM, B. W.; ABTS, C.; BROWN, A. W.; CHULANI, S.; CLARK, B. K.; HOROWITZ, E.; MADACHI, R.; REIFER, D.; STEECE, B. (2000). *Software Cost Estimation with COCOMO II*, Prentice-Hall. 502pp.
- BOEHM, B. W.; TURNER, R. (2004). *Balancing Agility and Discipline: a Guide for the Perplexed*, Addison Wesley. 266pp.
- BOWMAN, C.; AMBROSINI, V. (2000). 'Value creation versus value capture: towards a coherent definition of value in strategy', *British Journal of Management* **11**, pp. 1-15.
- BROOKS, F. (1987). 'No silver bullet: essence and accidents of software engineering', *IEEE Computer* **20**(4), pp. 10-19.
- BROWN, J. (1992). *Value Engineering: a Blueprint*, Book News, Portland, EUA. 251pp.
- CANTONE, G.; KRUCHTEN, P. (2008). Value-Based Design Decision Rationale Documentation: Principles and Empirical Feasibility Study, in 'Seventh Working IEEE/IFIP Conference on Software Architecture (WICSA 2008)', IEEE Computer Society, pp. 189-198.
- CARMEL, E.; WHITAKER, R. D.; GEORGE, J. F. (1993). 'PD and joint application design: a transatlantic comparison', *Communications of the ACM* **36**(6), pp. 40-48.
- CHEN, Y. (2007). Stakeholder Value Driven Threat Modeling for Off The Shelf Based Systems, in 'Proceedings of the 29th International Conference on Software Engineering (ICSE07 companion)', IEEE Computer Society, pp. 91-92.

COCKBURN, A.; HIGHSMITH, J. COCKBURN, A.; HIGHSMITH, J., ed. (2001). *Agile Software Development*, Addison Wesley. 278pp.

DAVIDSON, M. (2008). 'Survey: Agile interest high, but waterfall still used by many', SearchSoftwareQuality website. Disponível em [http://searchsoftwarequality.techtarget.com/news/article/0,289142,sid92\\_gci1318992,00.html](http://searchsoftwarequality.techtarget.com/news/article/0,289142,sid92_gci1318992,00.html), acessado em março de 2009.

DEBREU, G. (1959). *Theory of Value*. New York, Wiley. 114pp.

DIAS, M. d. L. P. (1994). 'Formação de preços em Jevons, Menger e Walras', *Textos em Economia* 5(1).

DINGSOYR, T. (2006). Value-Based Knowledge Management: the Contribution of Group Processes. In Biffli, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 309-325.

DYE, L. D.; PENNYPACKER, J. S. (2000). Project portfolio management and managing multiple projects: two sides of the same coin?. In 'Proceedings of the Project Management Institute Annual Seminars & Symposium'. Project Management Institute, Houston, Texas.

ECLIPSE FOUNDATION, (s.d.). 'Eclipse Process Framework Project (EPF)', The Eclipse Foundation. Disponível em <http://www.eclipse.org/epf/>.

EGYED, A. (2006). Tailoring Software Traceability to Value-Based Needs. In Biffli, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 287-308.

ERDOGMUS, H.; FAVARO, J.; HALLING, M. (2006). Valuation of software initiatives under uncertainty: concepts, issues and techniques. In Biffli, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 39-66.

FISCHER, R.; URY, W.; PATTON, B. (2005). *Como Chegar ao Sim: negociação de acordos sem concessões*. Trad. Vera Ribeiro e Ana Luiza Borges. Rio de Janeiro, Imago. 211pp.

HUANG, L. (2005). A value-based process for achieving software dependability, in M. LI; B. BOEHM; L. J. OSTERWEIL, ed., 'International Software Process Workshop', Springer Berlin / Heidelberg, pp. 108-121.

HUANG, L.; BOEHM, B.; HU, H.; GE, J.; LU, J.; QIAN, C. (2006). Applying the Value/Petri process to ERP software development in China. in 'Proceedings of the 28th international conference on Software engineering'. ACM, Xangai, China, pp. 502-511.

HUMPHREY, W. (2006), *TSP - Coaching Development Teams*, Addison Wesley. 416pp.

HUMPHREY, W. (1994). *A Discipline for Software Engineering*, Addison Wesley. 789pp.

IBM (International Business Machines) (2006). 'Rational Unified Process Base Plugin



Method Library', version 7.2.

JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. (1998). *The Unified Software Development Process*, Addison Wesley. 463pp.

JACOBSON, I.; GRISS, M.; JONSSON, P. (1997). *Software Reuse: Architecture, Process and Organization for Business Success*, Addison Wesley. 497pp.

KAPLAN, R. S.; NORTON, D. P. (1997). *A estratégia em ação: balanced scorecard*, 17ª Ed. Campus. 360pp.

KERZNER, H. (2006). *Gestão de Projetos: as melhores práticas*, Bookman, Porto Alegre. 821pp.

KIM, C.-K.; LEE, D.-H.; KO, I.-Y.; BAIK, J. (2007). 'A Lightweight Value-based Software Architecture Evaluation'. *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)* **2**, pp. 646-649.

KOTLER, P. (2003). *Marketing Insights from A to Z: 80 concepts every manager needs to know*. Wiley, New Jersey, EUA. 206pp.

KRUCHTEN, P. (2003). *The Rational Unified Process: An Introduction*, Addison-Wesley.

LAMRI (2004). 'RUP Leads as the Basis of Iterative Development', Lamri website. Disponível em: [http://www.lamri.com/resources/Lamri\\_Survey\\_results2.pdf](http://www.lamri.com/resources/Lamri_Survey_results2.pdf), acessado em janeiro/2009.

LAUDON, K. C.; LAUDON, J. P. (2007). *Sistemas de informações gerenciais*, 7ª ed. São Paulo, Pearson - Prentice Hall. 451pp.

LEPAK, D. P.; SMITH, K. G.; TAYLOR, M. S. (2007). 'Value creation and value capture: a multilevel perspective', *Academy of management Review* 32(1), pp. 180-194.

LIMA, M.; SANTOS, F.; BIONE, J.; LINS, T.; BARROS, E. (2005). ipPROCESS: A Development Process for Soft IP-core with Prototyping in FPGA, in 'Proceedings of Forum on Specification & Design Languages (FDL 2005)', pp. 27-28.

LINES, M.; BARNES, J.; HOLMES, J.; AMBLER, S. (2008). 'Agile RUP: Experiences from the trenches', IBM developerWorks. Disponível em [http://www.ibm.com/developerworks/rational/library/edge/08/feb08/lines\\_barnes\\_holmes\\_ambler/index.html](http://www.ibm.com/developerworks/rational/library/edge/08/feb08/lines_barnes_holmes_ambler/index.html), acessado em março de 2009.

MATOS, J. M. (2004). 'A Proposição de Valor e os Obstáculos na sua Execução', JM2 Consultoria e Treinamento. Disponível em [http://www.jmdois.com.br/home/an\\_txt16.htm](http://www.jmdois.com.br/home/an_txt16.htm), acessado em janeiro/2009.

MAXIMIANO, A. C. (1992). *Introdução à Administração*, 3ªed. Atlas, São Paulo. 434pp.

MCKENNA, P. (2005). 'Assessing the economic value of software projects', IBM

developerWorks. Disponível em <http://www.ibm.com/developerworks/rational/library/nov05/mckenna/>, acessado em janeiro/2009.

MORRIS, P.; JAMIESON, A. (2004). *Translating corporate strategy into project strategy: realizing corporate strategy through project management*. Project Management Institute, Newton Square, Pa., EUA. 116pp.

OGC (Office of Government Commerce) (2003). *Managing Successful Programmes*. The Stationery Office, UK. 158pp.

OJALA, P. (2008). 'Experiencing on implementing a value-based approach'. *WSEAS Transactions on Information Science & Applications* 5(4), pp. 385-395.

OMG (Object Management Group) (2008). 'Software & Systems Process Engineering Meta-Model Specification Version 2.0'. Disponível em <http://www.omg.org/cgi-bin/apps/doc?formal/08-04-01.pdf>, acessado em janeiro/2009.

PMI (Project Management Institute) (2008). *The Standard for Portfolio Management*, The Project Management Institute, Newton Square, Pa., EUA.

PMI (Project Management Institute) (2006). *The Standard for Program Management*, The Project Management Institute, Newton Square, Pa., EUA.

PMI (Project Management Institute) (2004). *Guia PMBOK - Um Guia do Conjunto de Conhecimentos em Gerenciamento de Projetos*, The Project Management Institute, Newton Square, Pa., EUA.

PORTER, M. E. (1996). 'What is strategy?'. *Harvard Business Review* 74(6), p. 68.

PRESSMAN, R. S. (2006). *Engenharia de Software, 6ª ed. Trad. Rosângela Dellosso Penteado*. McGraw-Hill, São Paulo. 720pp.

RAMLER, R.; BIFFL, S.; GRÜNBACHER, P. (2006). Value-Based Management of Software Testing. In Biffel, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 226-244.

REIFER, D. J. (2002). *Making the software business case: improvement by the numbers*. Addison Wesley. 300pp.

REIFER, D. J. (2006). Valuing Software Intellectual Property. In Biffel, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 345-366.

ROYCE, W. (1998). *Software project management: a unified framework*, Addison Wesley. 406pp.

SCHWABER, K. (2004). *Agile Project Management with Scrum*, Microsoft Press. 163pp.

SEI (Software Engineering Institute) (2006). 'CMMI for Development, Version 1.2',

Technical report, Software Engineering Institute, Pittsburgh, PA.

SHENHAR, A. J.; MILOSEVIC, D.; DVIR, D.; THAMHAIN, H. (2007). *Linking Project Management to Business Strategy*. Project Management Institute, Newton Square, Pa., EUA. 246pp.

SOFTEX (2007). 'MPS.BR - Melhoria de Processo do Software Brasileiro - Guia Geral versão 1.2', SOFTEX. Disponível em [http://www.softex.br/mpsbr/\\_guias/MPS.BR\\_Guia\\_Geral\\_V1.2.pdf](http://www.softex.br/mpsbr/_guias/MPS.BR_Guia_Geral_V1.2.pdf), acessado em janeiro/2009.

SOLOMON, P. (2002). 'Using CMMI to Improve Earned Value Management', Software Engineering Institute, Pittsburgh, PA., EUA. Disponível em <http://www.sei.cmu.edu/publications/documents/02.reports/02tn016.html>, acessado em janeiro/2009.

SOMMERVILLE, I. (2001). *Software Engineering*, 6<sup>th</sup> ed. Addison Wesley. 711pp.

SRIKANTH, H.; WILLIAMS, L. (2005). On the economics of requirements-based test case prioritization. In 'Proceedings of the seventh international workshop on Economics-driven software engineering research', Saint loius, EUA. pp. 1-3.

STANDISH GROUP (2008). 'Trends Report: Trends in IT value', The Standish Group. Disponível em [www.standishgroup.com/sample\\_research/index.php](http://www.standishgroup.com/sample_research/index.php). Acessado em janeiro/2008.

STANDISH GROUP (1995). 'The Chaos Report', The Standish Group. Disponível em [www.standishgroup.com/sample\\_research/index.php](http://www.standishgroup.com/sample_research/index.php). Acessado em janeiro/2008.

STEECE, B.; CHULANI, S.; BOEHM, B. (2002). Determining software quality using COQUALMO. In Wallace Blischke and D. N. Murthy (ed.): *Case Studies in Reliability and Maintenance*, Wiley, pp. 293-311.

STEPANEK, G. (2005). *Software Project Secrets: why software projects fail*, Apress, New York, EUA. 166pp.

SULLIVAN, K.; CHALASANI, P.; JHA, S.; SAZAWAL, V. (1999). Software design as an investment activity: a real options perspective. In L. Trigeorgis (ed.): *Real Options and Business Strategy: Applications to Decision Making*, Risk Books, pp. 215-262.

TEIXEIRA, G. (s.d.). 'Conheça o que são escalas de medida', site Ser Professor Universitário. Disponível em <http://www.serprofessoruniversitario.pro.br/ler.php?modulo=21texto=1304>, acessado em janeiro de 2009.

THORP, J. (2007). *The Information Paradox*, revised edition. Fujitsu Consulting, Canadá. 175pp.

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E. A. G. (2002). 'Introdução à Engenharia de Software Experimental'(RT-ES-590/02), Technical report, COPPE/UFRJ.

WEAVER, P. (2007). 'Effective project governance - linking PMI's standards to project governance', Mosaic Projects Services Pty Ltd. Disponível em [http://www.mosaicprojects.com.au/PDF\\_Papers/P051\\_GoPM\\_3.pdf](http://www.mosaicprojects.com.au/PDF_Papers/P051_GoPM_3.pdf), acessado em janeiro/2009.

WEILL, P.; ROSS, J. W. (2004). *IT governance: how top performers manage IT decision rights for superior results*. Harvard Business School Press, Boston, Ma., EUA. 269pp.

WEILL, P.; WOODHAM, R. (2002). 'Don't just lead, govern: implementing effective IT governance', *MIT Sloan School of Management, Research Paper Series*.

WINTER, M.; SMITH, C.; MORRIS, P.; CICMIL, S. (2006). 'Directions for future research in project management: The main findings of a UK government-funded research network', *International Journal of Project Management*, pp. 638-649.

WOHLIN, C.; AURUM, A. (2006). Criteria for Selecting Software Requirements to Create Product Value: An Industrial Empirical Study. In Biffl, S., Aurum A. Boehm B. Erdogmus H. Grunbacher P. (ed.): *Value-Based Software Engineering*, Springer-Verlag, pp. 196-217.

WRIGHT, P.; KROLL, M.; PRINGLE, C. (1992). *Strategic Management: Text and Cases*, Allyn and Bacon, Boston, Ma., EUA. 1022pp.

## Apêndice A

### Pontos de Extensão do RUP-VC

Elemento-chave da VBSE	Ponto de extensão	Tipo da intervenção de extensão	Elementos do RUP afetados
1. Análise de Realização de Benefícios	1. Adição de uma Disciplina de Gerência de Portfólio	Adição de disciplina	Disciplina: Gerenciamento de Portfólio
	2. Uso da Cadeia de Resultados na Definição de Programas e Portfólio	Adição de diretriz	Diretriz: Cadeia de Resultados
	3. Uso de Groupware	Extensão de tarefa Adição de diretriz	Tarefa: Identificar Pedidos dos Envolvidos Diretriz: Software Colaborativo
2. Identificação e Conciliação de Proposições de Valor dos Interessados	4. Gerenciamento das Expectativas das Partes Interessadas	Extensão de tarefa Extensão de artefato	Tarefa: Desenvolver a Visão Artefato: Visão
	5. Proposições de Valor como Critérios de Priorização de Casos de Uso	Extensão de tarefas Adição de diretriz	Tarefa: Desenvolver Plano de Iteração Tarefa: Priorizar Casos de Uso Diretriz: Teia de Conflitos de Modelos
3. Análise de Caso de Negócio	6. Análise de Opções considerando Riscos e Benefícios Intangíveis	Extensão de tarefa	Tarefa: Desenvolver Caso de Negócio

Elemento-chave da VBSE	Ponto de extensão	Tipo da intervenção de extensão	Elementos do RUP afetados
4. Gerenciamento Contínuo de Riscos e Oportunidades	7. Gerenciamento de Oportunidades	Extensão de conceito Extensão de artefato Extensão de diretriz	Conceito: Risco Artefato: Lista de Riscos Diretriz: Lista de Riscos
	8. Análise de Riscos a partir das Funções de Utilidade dos Interessados	Extensão de tarefas	Tarefa: Identificar e Avaliar Riscos Tarefa: Desenvolver Plano de Gerenciamento de Riscos.
	9. Adaptar o Processo de Software Considerando a Exposição ao Risco	Extensão de tarefa	Tarefa: Adaptar o Processo de Desenvolvimento para o Projeto
5. Engenharia Simultânea de Sistemas e de Software	10. Revisão de Marco de Ciclo de Vida Baseada em Valor	Extensão de tarefa	Tarefa: Revisão dos Marcos de Ciclo de Vida
6. Monitoramento e Controle baseados em Valor	11. Monitoramento de Valor no Nível do Projeto	Extensão de tarefa Extensão de diretriz	Tarefa: Desenvolver Plano de Medidas Diretriz: Métricas
	12. Monitoramento de Valor no Nível da Organização	Adição de tarefa Adição de diretriz	Tarefa: Monitorar Portfólio Diretriz: Fábrica de Experiência Baseada em Valor
7. Mudança como Oportunidade	13. Permitir Adaptações Visando Agilidade do Processo	Extensão de tarefa Extensão de diretriz	Tarefa: Adaptar o Processo de Desenvolvimento para o Projeto Diretriz: Discriminadores do Processo

## Apêndice B

### Exemplo de Elemento Estendido pelo RUP-VC

#### ***Tarefa: Desenvolver Caso de Negócio (revisada após avaliação)***

##### **Disciplina: Gerenciamento de Projeto**

Esta tarefa define como desenvolver um Caso de Negócios identificando o Produto essencial, Negócios e aspectos Financeiros do desenvolvimento de um novo produto, assim como as limitações principais e opções a considerar.

##### **Objetivo**

- Desenvolver a justificativa econômica do produto.

##### **Relacionamentos**

<b>Funções</b>	Executor Primário: •Coordenador de Projeto	Executores Adicionais:
<b>Entradas</b>	Obrigatório: •Nenhum	Opcional: •Cadeia de Resultados •Visão
<b>Saídas</b>	•Caso de Negócio	

##### **Descrição Principal**

O Caso de Negócio documenta o valor econômico do produto. É o instrumento pelo qual o financiamento do projeto é obtido. Um caso de negócio mal documentado poderá destruir até mesmo as melhores idéias sobre o produto, ao passo que um caso de negócio bem documentado poderá garantir o financiamento adequado aos produtos de valor.

##### **Etapas**

###### **1. *Descrever o Produto***

**Finalidade** Criar uma definição concisa do produto a ser desenvolvido.

Uma descrição resumida do produto com a qual todos os investidores concordem é fundamental para o sucesso do projeto. A descrição do produto deve definir, em alguns parágrafos, o que será o produto, que problemas ele solucionará e por que ele é necessário. A descrição não deve se aprofundar nos itens específicos do problema, mas deve criar um argumento convincente a favor da necessidade do produto. Ela deve ser breve, no entanto, para que seja facilmente compreendida e lembrada por todos os membros da equipe.

###### **2. *Definir o Contexto de Negócios (VBSE EC3)***

**Finalidade** Definir o ambiente no qual o produto será implementado. Definir o mercado do produto.

O contexto do negócio ajudará os investidores no projeto a entender e concordar com o mercado pretendido para o produto. O mesmo conjunto de requisitos, interpretado por diversos clientes, pode estar relacionado a sistemas distintos.

O contexto do negócio define o mercado pretendido para o produto, inclusive o domínio no qual o sistema será operado (por exemplo, telecomunicações, transações bancárias, comércio na Web, etc.) . Se o domínio for bem compreendido, uma descrição resumida será suficiente. Contudo, para alguns mercados, pode ser necessária uma descrição mais completa do espaço de problemas. A definição do mercado deve incluir produtos semelhantes e identificar empresas ou soluções concorrentes.

Se o produto está sendo desenvolvido para atender a um contrato, os termos do contrato devem ser observados. Se importantes marcos precisam ser passados, deve ser observado o cumprimento dos termos do contrato.

Se o produto for o aprimoramento de um produto já existente, este deverá ser descrito.

*A Cadeia de Resultados pode fornecer informação valiosa para a compreensão do contexto, tais como as premissas do negócio.*

### **3. Definir os Objetivos do Produto (VBSE EC3)**

**Finalidade** Informar claramente os objetivos do produto.

Informe os objetivos para o desenvolvimento do produto - os motivos de sua importância. Fazem parte desse processo um esboço e uma avaliação dos riscos da programação. Objetivos claramente definidos e expressos garantem uma base sólida para a formulação de marcos e o gerenciamento de riscos, isto é, a manutenção do projeto sob controle e a garantia de seu sucesso.

*Consulte a Cadeia de Resultados da qual o projeto faz parte para extrair dela informações sobre a contribuição do produto para a organização.*

### **4. Desenvolver a Previsão Financeira**

**Finalidade** Desenvolver projeções do custo e dos lucros do projeto.

Para um produto de software comercial, o Caso de Negócio deve incluir um conjunto de suposições sobre o projeto e a ordem de importância do ROI (retorno do investimento) se essas suposições forem verdadeiras. Por exemplo, o retorno do investimento terá importância cinco, se concluído em um ano, dois, se concluído em dois anos, e um número negativo, após esse tempo. Essas suposições são verificadas no fim da fase de elaboração, quando o escopo e o plano já estão melhor conhecidos. O retorno baseia-se na estimativa de custo e nas estimativas de possíveis lucros.

No caso de projetos de software internos, o retorno será calculado em termos do 'Valor Líquido Presente' do projeto ou em termos de uma taxa interna de retorno. Com o valor líquido presente, o futuro *stream* de fluxos de caixa acumulado no projeto será calculado (inclusive os fluxos de caixa negativos relacionados ao desenvolvimento e suporte do projeto)



e, em seguida, descontados com base em uma taxa de retorno determinada pela organização de acordo com o risco do projeto. Um valor líquido presente superior a zero indica que o projeto representa um benefício economicamente positivo para a empresa.

No caso da taxa interna de cálculo para o retorno, pressupõe-se um valor líquido presente igual a zero e a taxa interna de retorno para a produzi-lo é calculada. Essa taxa interna de retorno (IRR) do projeto é, então, comparada com uma taxa mínima de retorno para os projetos com nível de risco semelhante. Se a IRR do projeto for superior à taxa mínima de retorno, o projeto representará uma vantagem econômica para a empresa.

O valor líquido presente e as taxas internas de retorno também podem ser calculados para os produtos de software comerciais.

A estimativa de recursos engloba o projeto inteiro até sua liberação. Essa estimativa é atualizada em cada fase e cada iteração e se torna mais exata a cada iteração.

Deve ser incluída uma explicação da base usada para as estimativas.

#### **5. Descrever as Limitações do Projeto**

**Finalidade** Definir as restrições do projeto.

Expressar as restrições para o desenvolvimento do projeto. Essas restrições envolvem riscos e custos. Elas podem ser algo como interfaces externas com as quais o sistema deve ser compatível, padrões, certificações ou abordagens técnicas empregadas por questões estratégicas, como a utilização de uma certa tecnologia de banco de dados ou de mecanismos de distribuição.

#### **6. Descrever Opções (VBSE EC3)**

**Finalidade** Apresentar algumas opções para o produto e para o projeto, além de descrever o efeito que causam sobre a previsão financeira e as restrições do projeto.

Descrever as opções para o produto - recursos e características opcionais e os custos e benefícios a eles associados - e as opções para a abordagem do projeto. As opções do projeto devem incluir diversas bases contratuais, diversos ciclos de vida do projeto, diversas combinações de 'fazer' e 'adquirir', e assim por diante. Em cada caso, o efeito da opção sobre a previsão financeira e as restrições (o risco de impacto) deve ser descrito. O objetivo é oferecer aos gerenciadores com autoridade para financiar o projeto alguma amplitude para a tomada de decisões em termos de capacidade, custo, ROI, benefícios intangíveis e aproveitamento de oportunidades de negócios, programação, base para contrato, ciclo de vida de desenvolvimento, restrições técnicas e outros itens.

*Em um contexto de análise de caso de negócio baseado em valor, cada fase do ciclo de vida apresenta diversos trade-offs de negócios, determinando opções de abordagem do projeto.*

*Durante a fase de Iniciação:*

*•Analisar, a partir dos objetivos iniciais do projeto, trade-offs de custo, cronograma e*

funcionalidade, usando técnicas de análise de custo como variável independente.

•Determinar o preço do produto de software (ou outra atividade do projeto que gere receita) usando análise de tendências para prever o volume de vendas necessário.

•Requisitar orçamentos usando análise de custo/benefício para justificar as despesas necessárias.

Durante a fase de Elaboração:

•Conduzir análise entre "fazer" e "adquirir" (make-or-buy), para determinar como preencher melhor a arquitetura usando técnicas de análise de retorno e de oportunidade de investimento.

•Desenvolver orçamentos para o custo do software e cronograma usando modelos como o COCOMO II (BOE2000) para estimar o esforço e a duração necessária.

•Priorizar riscos (positivos e negativos) através de uma avaliação quantitativa dos impactos dos itens de risco.

Durante a fase de Construção:

•Decidir quanto ao lançamento ou não de uma versão ou incremento do produto baseando-se em análise de tendências (por exemplo, olhando para defeitos e níveis de qualidade associados).

•Desenvolver estimativas de custo e cronograma de conclusão, usando modelos para determinar se é possível ou não entregar como prometido, dependendo da taxa de progresso atual.

Durante a fase de Transição:

•Conduzir trade-offs entre "consertar" ou "substituir" (replace/repair) usando técnicas como análise de Pareto, para determinar quais módulos do software estão mais sujeitos a defeitos.

•Desenvolver estimativas quanto à manutenção, usando modelos de custo para prever esforço e duração baseado em projeções estatísticas de mudanças.

## Apêndice C

### Questionário de Avaliação do RUP-VC

#### 1. Perfil do Participante

Nome: \_\_\_\_\_ Data: \_\_\_\_ / \_\_\_\_ / \_\_\_\_

Escolaridade (informe somente o maior grau):

- Nível Básico
- Nível Médio
- Superior Incompleto
- Superior Completo

Curso/Instituição: \_\_\_\_\_

Ano de Conclusão: \_\_\_\_\_

Pós-Graduação (Especialização, Mestrado, Doutorado e Pós-doutorado):

- Especialização
- Mestrado
- Doutorado
- Pós-Doutorado

Curso/Instituição: \_\_\_\_\_

Ano de Conclusão: \_\_\_\_\_

Tempo de experiência em Desenvolvimento de Sistemas: \_\_\_\_\_

Tempo de experiência em Gerenciamento de Projeto: \_\_\_\_\_

Tempo de experiência em Gerenciamento de Projeto com RUP: \_\_\_\_\_

Número de projetos gerenciados (inclusive em andamento): \_\_\_\_\_

Assinale a(s) função(ões) exercida(s):

- Gerente Geral da Organização, Tempo \_\_\_\_\_
- Gerente de Setor, na área de informática, Tempo \_\_\_\_\_
- Gerente do Setor de Desenvolvimento de Software, Tempo \_\_\_\_\_
- Gerente de Projeto, Tempo \_\_\_\_\_
- Desenvolvedor, Tempo \_\_\_\_\_
- Outros. Especificar: \_\_\_\_\_, Tempo \_\_\_\_\_

Existe algum processo formal de desenvolvimento de software utilizado pela organização onde trabalhou/trabalha?

(métodos, ferramentas, técnicas, ciclo de vida, atividades)     Sim     Não

Se sim, especificar qual: \_\_\_\_\_

Existe alguma prática formalizada para criação de valor no processo de desenvolvimento de software utilizado pela organização onde trabalhou/trabalha?

(métodos, ferramentas, técnicas, ciclo de vida, atividades) ( ) Sim ( ) Não

Se sim, especificar qual: \_\_\_\_\_

Se sim: Você considera esse fato importante para o sucesso do projeto?

( ) Sim ( ) Não

Se não: Você considera esse fato como um dos fatores explicativos para o insucesso de projetos? ( ) Sim ( ) Não

## 2. Avaliação do RUP-VC

### 2.1-Sobre a importância e a oportunidade do tema escolhido

#### Resumo:

Uma das tendências para as pesquisas na área de gerenciamento de projetos é o estudo da ênfase na criação de valor em projetos, em vez da teoria atual que dá ênfase na criação de produtos ou serviços. Projetos interessantes para as organizações são os que criam valor, pois trazem benefícios ao negócio. Em projetos de software, a ênfase na criação de valor passa pela revisão das práticas e princípios da engenharia de software.

#### Proposição: O tema escolhido é importante e oportuno.

C+ Concordo totalmente

D- Discordo parcialmente

C<sub>0</sub> Concordo

D<sub>0</sub> Discordo

Sem opinião

C- Concordo parcialmente

D+ Discordo totalmente

Seus argumentos: \_\_\_\_\_

\_\_\_\_\_

### 2.2-Conceito de Criação de Valor em Projetos

#### Resumo:

Do ponto de vista organizacional, a criação de valor relaciona-se com a estratégia de negócios da organização. Pesquisas identificam que, para muitas organizações, a principal preocupação não está mais nos ativos de capital, sistemas e instalações, mas cada vez mais está no desafio de vincular a estratégia de negócios aos projetos, maximizando a geração de receitas, e gerindo a entrega de benefícios em relação aos diferentes grupos de interessados (Winter, 2006).

Nesse sentido, um maior esforço de pesquisa é necessário com relação aos processos de seleção estratégica de projetos e programas nas organizações, de forma a reconhecer diferentes formas de valor, além das representações convencionais da produção e da manufatura. O conceito, considerado neste trabalho de Mestrado, de valor criado por um projeto, consiste nos seguintes aspectos:

- Valor em termos financeiros, na forma do retorno sobre o investimento no projeto;

- Benefícios não quantificáveis, ou intangíveis, trazidos pelo projeto. Por exemplo, melhoria da imagem da organização junto aos clientes;
- Aproveitamento de oportunidades de negócio propiciadas pelo projeto.

No contexto do trabalho de Mestrado destaca-se também um outro conceito importante associado a valor: é a proposição de valor (ou proposta de valor), que vem a ser o conjunto de atributos que um interessado (stakeholder) percebe como essenciais para o(s) produto(s) de um projeto.

**Proposição A: A mudança de foco do gerenciamento de projetos, saindo da criação de um produto ou serviço, para a criação de valor para a organização patrocinadora, demanda que a gerência de projetos suba a um patamar estratégico nas organizações.**

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

**Proposição B: O conceito de *valor criado por um projeto* considerado neste trabalho é adequado à função estratégica dos projetos.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### 2.3-Escolha do processo RUP

**Resumo:**

O RUP é um framework de processo de software largamente utilizado. É bastante abrangente do ponto de vista da engenharia de software, entretanto, uma breve análise indica que seu foco é, nas palavras de Barry Boehm, “*value-neutral*”, isto é, o processo não enfatiza através das suas práticas a criação de valor.

**Proposição: Foi acertada a escolha do RUP como base para análise e extensão propostas neste trabalho, de modo a integrar considerações de valor a projetos de desenvolvimento de software e dar ênfase à criação de valor nesses projetos.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.4-Escolha da Engenharia de Software Baseada em Valor (VBSE) como diretriz da análise

### Resumo:

Com relação a projetos de TI, e mais especificamente a projetos de desenvolvimento de software, acrescenta-se uma preocupação adicional ao problema: o alinhamento das soluções técnicas trazidas por eles aos resultados de negócios da organização.

A Engenharia de Software Baseada em Valor (*Value-Based Software Engineering* ou VBSE) é uma abordagem, criada por Barry Boehm (2003), que aceita o desafio de integrar considerações de valor a todos os princípios e práticas de Engenharia de Software existentes e emergentes. A VBSE propõe desenvolver um framework abrangente para a Engenharia de Software.

### Proposição: A Engenharia de Software Baseada em Valor é um framework adequado para integrar considerações de valor ao processo RUP.

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Para as próximas seções do questionário, recomenda-se que o teor dos resumos seja

complementado pela consulta ao conteúdo da extensão RUP-VC, apresentado no workshop e disponível em <http://www.cin.ufpe.br/~gta/rup-vc>.

## 2.5-Análise de Realização de Benefícios

**Resumo:** A Análise de Realização de Benefícios consiste de práticas e técnicas para identificar e analisar os benefícios de negócio associados ao projeto de software em questão. Este elemento-chave da VBSE foi implantado no RUP-VC através da adição de uma disciplina de Gerenciamento de Portfólio e do uso da Cadeia de Resultados na definição de Programas e Portfólio.

**Proposição:** A forma como foi implementada a Análise de Realização de Benefícios no RUP-VC atende aos objetivos deste elemento-chave.

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.6-Identificação e Conciliação das Proposições de Valor dos Interessados

**Resumo:** A Identificação e a Conciliação de Proposições de Valor das Partes Interessadas consistem em incluir, no gerenciamento de requisitos e do projeto, as proposições de valor que cada um dos interessados associa ao projeto. O RUP-VC implementa este elemento-chave sugerindo o uso de *groupware* e acrescentando o gerenciamento das expectativas das partes interessadas no levantamento de requisitos, e introduzindo as proposições de valor como fatores de priorização de casos de uso.

**Proposição:** A forma como foram implementadas a Identificação e a Conciliação de Proposições de Valor das Partes Interessadas no RUP-VC atende aos objetivos deste elemento-chave.

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.7-Análise de Caso de Negócio

**Resumo:** a Análise de Caso de Negócio consiste em basear os critérios de sucesso do projeto em um caso de negócio que leve em consideração a criação de valor, não apenas o retorno sobre o investimento (ROI). No RUP-VC, é implementado pela análise de opções considerando riscos e benefícios não quantificáveis.

**Proposição:** A forma como foi implementada a Análise de Caso de Negócio no RUP-VC atende aos objetivos deste elemento-chave.

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.8-Gerenciamento Contínuo de Riscos e Oportunidades

**Resumo:** O Gerenciamento Contínuo de Riscos e Oportunidades trata de gerenciar ameaças e oportunidades relativas ao projeto, levando em consideração fatores como a utilidade que cada decisão de projeto tem para cada grupo de interessados. O RUP-VC implementa este elemento-chave através da introdução do gerenciamento de oportunidades (riscos positivos), da análise de riscos utilizando funções de utilidade dos interessados, e da consideração da exposição ao risco na adaptação do processo de software a cada projeto.



**Proposição: A forma como foi implementado o Gerenciamento Contínuo de Riscos e Oportunidades no RUP-VC atende aos objetivos deste elemento-chave.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.9-Engenharia Simultânea de Sistema e de Software

**Resumo:** O elemento-chave Engenharia Simultânea de Sistema e de Software consiste em sincronizar as atividades e decisões de projeto da engenharia de sistemas (ou da modelagem de negócios, no caso de sistemas de informação) e da engenharia de software. No RUP-VC este elemento-chave é implementado tornando a revisão de marco de ciclo de vida uma atividade baseada em valor.

**Proposição: A forma como foi implementada a Engenharia Simultânea de Sistema e de Software no RUP-VC atende aos objetivos deste elemento-chave.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## 2.10-Monitoramento e Controle Baseados em Valor

**Resumo:** O Monitoramento e o Controle Baseados em Valor consistem em técnicas para estender o monitoramento e o controle do progresso do projeto para além das práticas convencionais, como o EVM (*earned value management*), e abranger outras considerações de valor. Este elemento-chave é implementado no RUP-VC através do monitoramento de valor

através do caso de negócio, e do monitoramento de valor com base na Cadeia de Resultados.

**Proposição: A forma como foram implementados o Monitoramento e o Controle Baseados em Valor no RUP-VC atende aos objetivos deste elemento-chave.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### 2.11-Mudança como Oportunidade

**Resumo:** Este elemento-chave consiste em adaptar o RUP, um processo guiado por planejamento (ou “disciplinado” ou “burocrático”), de forma a considerar as mudanças (no projeto ou em seu contexto) como oportunidades de maximizar o valor a ser entregue pelo projeto. É implementado pelo RUP-VC ao permitir adaptações visando a agilidade do processo.

**Proposição: A forma como foi implementada a Mudança como Oportunidade no RUP-VC atende aos objetivos deste elemento-chave.**

<input type="checkbox"/> C+ Concordo totalmente	<input type="checkbox"/> D- Discordo parcialmente	
<input type="checkbox"/> C <sub>0</sub> Concordo	<input type="checkbox"/> D <sub>0</sub> Discordo	<input type="checkbox"/> Sem opinião
<input type="checkbox"/> C- Concordo parcialmente	<input type="checkbox"/> D+ Discordo totalmente	

Seus argumentos: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### 2.12-Adequação dos elementos VBSE ao RUP: a extensão RUP-VC

**Resumo:** A proposta de extensão do RUP para ênfase em criação de valor, denominada RUP-VC, foi elaborada a partir da integração ao RUP dos elementos-chave da VBSE.

**Proposição: A aplicação dos elementos-chave da VBSE ao RUP aperfeiçoou o processo de software no sentido de obter ênfase na criação de valor em projetos que utilizem o RUP-VC.**

C+ Concordo totalmente

D- Discordo parcialmente

C<sub>0</sub> Concordo

D<sub>0</sub> Discordo

Sem opinião

C- Concordo parcialmente

D+ Discordo totalmente

Seus argumentos: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

### ***3. Sugestões e Opiniões***

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_