



## MELHORIA CONTÍNUA DE PROCESSO DE SOFTWARE UTILIZANDO A TEORIA DAS RESTRIÇÕES

Thiago Moreira da Costa

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientadores: Ana Regina Cavalcanti da Rocha  
Gleison dos Santos Souza

Rio de Janeiro  
Maio de 2012

MELHORIA CONTÍNUA DE PROCESSO DE SOFTWARE UTILIZANDO A  
TEORIA DAS RESTRIÇÕES

Thiago Moreira da Costa

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO  
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA  
(COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE  
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE  
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

---

Prof.<sup>a</sup> Ana Regina Cavalcanti da Rocha, D.Sc.

---

Prof. Gleison dos Santos Souza, D.Sc.

---

Prof. Geraldo Bonorino Xexéo, D.Sc.

---

Prof. Adriano Bessa Albuquerque, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MAIO DE 2012

Costa, Thiago Moreira da

Melhoria Contínua de Processos de Software  
Utilizando a Teoria das Restrições / Thiago Moreira da  
Costa. – Rio de Janeiro: UFRJ/COPPE, 2012.

XIII, 233 p.: il.; 29,7 cm.

Orientadores: Ana Regina Cavalcanti da Rocha  
Gleison dos Santos Souza

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de  
Engenharia de Sistemas e Computação, 2012.

Referências Bibliográficas: p. 101-112.

1. Melhoria de Processos de Software. 2. Teoria das  
Restrições. 3. Processos de Software. 4. Processos de  
Raciocínio. I. Rocha, Ana Regina Cavalcanti da *et al.* II.  
Universidade Federal do Rio de Janeiro, COPPE,  
Programa de Engenharia de Sistemas e Computação. III.  
Título.

*A Deus, pela força e graça que abundam em minha vida.*  
*Aos meus pais, raízes que me apoiam e me sustentam com amor e incentivo*

## AGRADECIMENTOS

As primícias a meu Deus, que nunca esteve distante e sempre derramou em minha vida graça, força e desafios, permitindo-me desfrutar do prazer de conquistas. Se há Glória, Ele é o digno.

Aos meus pais, por toda a força e a coragem materializadas em valores, ensinamentos e educação, bens estes dos quais tenho profundo orgulho e que são a base de quem eu sou. Pai, obrigado por todo o esforço, suor e lágrimas, este momento é seu também! Obrigado. Mãe, obrigado pelo cuidado e atenção que acompanharam os primeiros passos até aqui. Isto não seria possível sem sua perseverança!

A Ana Cláudia e Valdeci por me apoiarem em momentos importantes desse resultado. Obrigado!

Aos meus irmãos, Ginna e Victor pelo carinho, incentivo e torcida constantes. Obrigado!

À minha orientadora, Ana Regina, pelo crédito em mim depositado e cuja orientação foi além da acadêmica, contribuindo para meu amadurecimento pessoal e profissional. Obrigado!

Ao meu orientador, Gleison, pela paciência e atenção traduzidas em revisões rigorosas, orientação e praticidade que me ajudaram a focar. Obrigado!

Aos professores Adriano e Xexéo, por aceitarem participar da banca e pela contribuição à pesquisa.

À Larissa por contribuir com a pesquisa através do estudo de viabilidade, pela paciência e atenção. Obrigado!

À família que a vida me trouxe, Massimiliana, Robson, Marcus, Kamilla, Italo, Karol, Lidiane, Anne pelo companherismo e parceria nos momentos difíceis da jornada. Obrigado!

Aos amigos da COPPE, Mylene, Fabrício, Anne, Mariano, David, Taísa, Elaine, Natália, Reinaldo, Andréia, Ahilton, Simões, Monalessa, por contribuírem nos seminários, com sugestões, além do companherismo e dos bons momentos no grupo de pesquisa. Meu obrigado especial a Natalia e Mariano pelas ideias iniciais, Mylene e

Fabício por diversas conversas e andamento em conjunto das nossas pesquisas. Obrigado!

Aos amigos da F. Atech que torceram por mim e me permitiram, neste momento final, ter calma e tranquilidade para finalizar esta etapa. Obrigado!

Às funcionárias do PESC, Taísa, Solange, Mercedes, Sônia e Cláudia, por sua colaboração nos procedimentos administrativos.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## MELHORIA CONTÍNUA DE PROCESSO DE SOFTWARE UTILIZANDO A TEORIA DAS RESTRIÇÕES

Thiago Moreira da Costa

Maio/2012

Orientadores: Ana Regina Cavalcanti da Rocha  
Gleison dos Santos Souza

Programa: Engenharia de Sistemas e Computação

O processo de desenvolvimento de software é composto por elementos que estão envolvidos durante a sua execução. Insumos, procedimentos, ferramentas e pessoas são exemplos de elementos do processo que, ao interagirem para produzir os produtos intermediários e finais, ocasionam efeitos que podem influenciar de maneira desejável ou indesejável o seu desempenho. Tais efeitos são fatores que estão presentes independentemente se os processos são estáveis ou se apresentam comportamento não previsível. A melhoria de processo é um fator importante para organizações de software que almejam ser competitivas. O objetivo desta dissertação é apresentar uma abordagem para melhoria contínua de processo de software por meio da sistematização da investigação de fatores de influência indesejáveis, a fim de identificar o fator que mais influencia o desempenho do processo de maneira indesejada. Além disso, a sistematização apoia a elaboração da solução para a remoção do fator de influência indesejável, com o intuito de maximizar as possibilidades de sucesso. Para isto, são aplicados os conceitos dos Processos de Raciocínio da Teoria das Restrições.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

## CONTINUOUS SOFTWARE PROCESS IMPROVEMENT USING THEORY OF CONSTRAINTS

Thiago Moreira da Costa

May/2012

Advisors: Ana Regina Cavalcanti da Rocha

Gleison dos Santos Souza

Department: Systems and Computing Engineering

Software development process is composed by elements that are involved during its execution. Inputs, procedures, tools, human resources and its characteristics are examples of these elements that might cause desirable or undesirable effects while interacting to produce intermediate or final products. These effects are factors that are present in the development process regardless of whether they are stable or they present an unpredictable behavior. Process improvement is an important factor for organizations that aim to be competitive. The purpose of this dissertation is to present an approach to continuous software process improvement by a systematic investigation of the influence of undesirable factors in order to identify the factor that most influences undesirably the process performance. In addition, the approach supports the development of solution proposals to remove the undesirable factor and the elaboration of improvement candidates in order to maximize the chances of successful implementation and deployment of these candidates. In this context, the concepts of the Thinking Processes of Theory of Constraints were applied.



## ÍNDICE

Capítulo 1 – Introdução .....	1
1.1 Introdução.....	1
1.2 Objetivo da Dissertação .....	2
1.3 Metodologia.....	2
1.4 Estrutura da dissertação.....	5
Capítulo 2 –Melhoria Contínua de Processo de Software .....	7
2.1 Introdução.....	7
2.2 Processo de Software .....	9
2.3 Melhoria Contínua nas Normas e nos Modelos de Processo .....	9
2.3.1 ISO/IEC 15504 .....	10
2.3.2 CMMI.....	12
2.3.3 MR-MPS .....	14
2.4 Controle Estatístico de Processo de Software .....	15
2.5 Seis Sigma Aplicado à Melhoria de Processo de Software.....	19
2.6 Abordagens para Apoiar a Melhoria Contínua de Processos de Software.....	22
2.6.1 Breve Histórico do Estudo Baseado em Revisão Sistemática.....	22
2.6.2 Abordagem para Prevenção de Defeitos Provenientes de Inspeções para Apoiar a Melhoria dos Processos de Engenharia do Software .....	23
2.6.3 Melhoria Contínua de Processo de Software através do Controle Estatístico de Processo .....	25
2.6.4 Melhoria de Processos de Software através do Seis Sigma .....	28
2.6.5 Abordagem para Avaliação de Propostas de Melhoria Em Processos de Software .....	31
2.7 Considerações Finais.....	33
Capítulo 3 –Teoria das Restrições.....	35
3.1 Introdução.....	35
3.2 Teoria das Restrições .....	36
3.2.1 Classificação das Restrições.....	38
3.3 Ciclo de Melhoria Cinco Passos .....	39
3.4 Processos de Raciocínio .....	40
3.4.1 Fundamentos Lógicos .....	42

3.4.2	Processos de Raciocínio .....	46
3.5	Melhoria Contínua com os Processos de Raciocínio .....	50
3.6	Trabalhos Relacionados à Melhoria Contínua .....	50
3.7	Considerações Finais.....	55
Capítulo 4 –Abordagem Para Melhoria Contínua em Processos de Software		
	Utilizando a Teoria das Restrições .....	57
4.1	Introdução.....	57
4.2	Definir objetivo de melhoria .....	60
4.2.1	Descrever objetivo de melhoria .....	61
4.2.2	Avaliar rastreabilidade do objetivo de melhoria aos objetivos de negócio.....	63
4.3	Preparar para Análise .....	63
4.3.1	Planejar melhoria .....	64
4.3.2	Selecionar diagrama causal candidato .....	64
4.3.3	Coletar informações organizacionais de contexto .....	65
4.3.4	Coletar informações de contexto de projetos .....	66
4.3.5	Identificar fatores pertinentes .....	66
4.4	Identificar restrição .....	67
4.4.1	Desenvolver diagrama causal .....	67
4.4.2	Avaliar diagrama causal .....	69
4.4.3	Identificar restrição principal .....	70
4.5	Elaborar proposta de melhoria.....	70
4.5.1	Elaborar proposta de solução.....	71
4.5.2	Avaliar viabilidade da proposta de solução.....	71
4.5.3	Elaborar proposta de melhoria.....	72
4.5.4	Elaborar planejamento de implementação da proposta de melhoria .....	74
4.6	Implementar a melhoria candidata.....	75
4.6.1	Executar piloto.....	76
4.6.2	Armazenar melhoria candidata.....	77
4.7	Registrar resultados.....	77
4.7.1	Registrar artefatos do processo.....	78
4.7.2	Registrar lições aprendidas.....	78
4.8	Considerações Finais.....	78
Capítulo 5 –Avaliação da Abordagem .....		
5.1	Introdução.....	80

5.2	Estudo de Viabilidade .....	81
5.3	Contexto de Aplicação do Estudo de Viabilidade .....	81
5.4	Definição do Estudo de Viabilidade .....	82
5.4.1	Validade Interna do Estudo .....	86
5.4.2	Validade Externa do Estudo .....	86
5.4.3	Validação de Construção do Estudo .....	87
5.4.4	Validação da Conclusão do Estudo .....	87
5.5	Execução do Estudo .....	87
5.6	Análise e Interpretação dos Resultados.....	92
5.7	Considerações Finais.....	95
Capítulo 6	– Conclusão.....	96
6.1	Considerações finais .....	96
6.2	Limitações .....	97
6.3	Contribuições.....	98
6.4	Oportunidades Futuras de Pesquisa .....	99
Referências Bibliográficas	.....	101
Anexo I	– Estudo Baseado em Revisão Sistemática Sobre Melhoria Contínua em Processos de Software.....	113
Anexo II	– Descrição do Processo da Abordagem .....	179
Anexo III	– Descrição dos Procedimentos dos Processos de Raciocínio .....	190
Anexo IV	– Modelos de Formulários do Processo.....	205
Anexo V	– Informações e Artefatos Produzidos e Utilizados Durante o Estudo de Viabilidade .....	215

## ÍNDICE DE FIGURAS

Figura 1.1 - Visão geral da metodologia (CONTE, 2009).....	3
Figura 2.1 - Exemplo de gráfico de controle XR.....	16
Figura 2.2 - Visão Geral da Abordagem DPPI (KALINOWSKI, 2011).....	24
Figura 2.3 - Processo de Investigação (CAIVANO, 2005a).....	26
Figura 2.4 - Abordagem BSR (WANG <i>et al.</i> , 2006).....	27
Figura 2.5 - Modelo do Processo para Realização de Pilotos (SILVA FILHO, 2006) .....	32
Figura 3.1 - Ciclo de Melhoria Cinco Passos.....	40
Figura 3.2 - Modelo de um Diagrama de Resolução de Conflitos .....	47
Figura 3.3 - Ferramenta apropriada dos Processos de Raciocínio de acordo com a dificuldade de implementação (SCHEINKOPF, 1999).....	49
Figura 3.4 - Processos de Raciocínio Integrados.....	51
Figura 3.4 - Um <i>framework</i> de melhoria contínua integrado, combinando Seis Sigma e Teoria das Restrições (EHIE e SHEU, 2005).....	54
Figura 4.1 - Processo da abordagem proposta.....	62
Figura 5.1 - Gráfico de controle do número de dias para abertura e resolução dos chamados do processo.....	88

## ÍNDICE DE TABELAS

Tabela 2.1 - Níveis de Capacidade dos Processos da Norma ISO/IEC 15504 (ISO/IEC, 2003) .....	11
Tabela 2.2 - Lista de origens, motivos e características mensuráveis (TONINI, 2006) .....	21
Tabela 2.3 - Elementos PSP/TSP e ferramentas de Seis Sigma correspondentes (PARK et al., 2007).....	30
Tabela 3.1 - Aplicação dos Processos de Raciocínio em relação às questões de mudança (SCHEINKOPF, 1999) .....	42
Tabela 3.2 – Relacionamentos dos Processos de Raciocínio .....	43
Tabela 3.3 – Categorias de Ressalvas Legítimas (SCHEINKOPF, 1999).....	45
Tabela 3.4 - Comparação das estratégias de melhoria contínua (adaptado) (BROWN et al., 2008).....	52
Tabela 5.1 - Etapas da execução do estudo de viabilidade .....	88
Tabela 5.2 - Tempo de execução das tarefas da primeira etapa do estudo de viabilidade..	89
Tabela 5.3 - Tempo de execução das tarefas da segunda etapa do estudo de viabilidade ..	91

# Capítulo 1 – Introdução

*Este capítulo apresenta as principais questões que motivaram a realização deste trabalho, o objetivo principal a ser atendido com a sua realização, a metodologia de pesquisa, e a estrutura como esta dissertação está organizada.*

## 1.1 Introdução

O processo de desenvolvimento de software é composto por elementos que estão envolvidos durante a sua execução. Insumos, procedimentos, ferramentas e pessoas são exemplos de elementos do processo de desenvolvimento que, ao interagirem para produzir os produtos intermediários e produtos finais, ocasionam efeitos que podem influenciar de maneira desejável ou indesejável o desempenho do processo. Estes efeitos são fatores de influência que estão presentes nos processo de software, independentemente se estes são maduros e estáveis, ou se apresentam comportamento não previsível ao longo do tempo.

A melhoria de processo é um fator importante para organizações de software que almejam ser competitivas. Organizações que mantêm um ciclo contínuo de melhoria podem, de maneira estruturada, melhorar a capacidade de seus processos, a fim de atingir melhores níveis de qualidade, tempo e custo de produção.

Existem diferentes abordagens e técnicas para a melhoria de processo de software. Dentre as abordagens existentes, destacam-se os modelos de maturidade, tais como CMMI-DEV (SEI, 2010) e MR-MPS (SOFTEX, 2011), que estruturam por meio de níveis a capacidade e a maturidade de uma organização de software em relação aos seus processos. Além disso, inúmeras abordagens para a melhoria de processo de software são encontradas na literatura, por exemplo, FLORAC e CARLETON (1999), CAIVANO (2005b), WANG *et al.* (2006), ALBUQUERQUE (2008), KALINOWSKI (2011).

A melhoria de processos que são alvo de melhoria contínua tende a se tornar mais complexa e, conseqüentemente, mais difícil (DETTMER, 1998b). Embora diversas técnicas e abordagens apoiem a melhoria de processo, há oportunidade para

sistematizar a investigação dos fatores de influência do processo de software e para apoiar a elaboração de melhorias que aumentem as possibilidades de sucesso da melhoria.

## **1.2 Objetivo da Dissertação**

O objetivo desta dissertação é propor uma abordagem que apoie a melhoria contínua em processos de software. Esta abordagem é baseada em uma metodologia de melhoria contínua que auxilia a organização a investigar sistematicamente os fatores de influência que causem efeitos indesejáveis ao desempenho do processo de software, e a elaborar melhorias candidatas que considerem os fatores de influência envolvidos e os efeitos causados pelas ações de melhoria aplicadas.

Para atingir este objetivo, são utilizados os princípios da Teoria das Restrições – TDR (*Theory of Constraints*) (GOLDRATT e COX, 1984) e os seus Processos de Raciocínio – PP (*Thinking Process*). Os Processos de Raciocínio são baseados em lógica e utiliza a base do pensamento científico, sendo utilizados nas etapas de identificação da restrição principal ao desempenho do processo, elaboração de alternativas de solução para a restrição principal, análise de viabilidade da proposta de solução e desenvolvimento de plano de implantação da melhoria proposta.

Para facilitar a aplicação dos PP para o contexto de processos de software, a abordagem proposta fornece um conjunto de diretrizes e passos que encapsulam os princípios da Teoria das Restrições, diminuindo a necessidade de um conhecimento profundo sobre a TDR por parte dos responsáveis pela ação de melhoria.

Juntamente com o conjunto de diretrizes, um processo é definido, descrevendo as atividades, tarefas e procedimentos, para apoiar a melhoria contínua de processo de software.

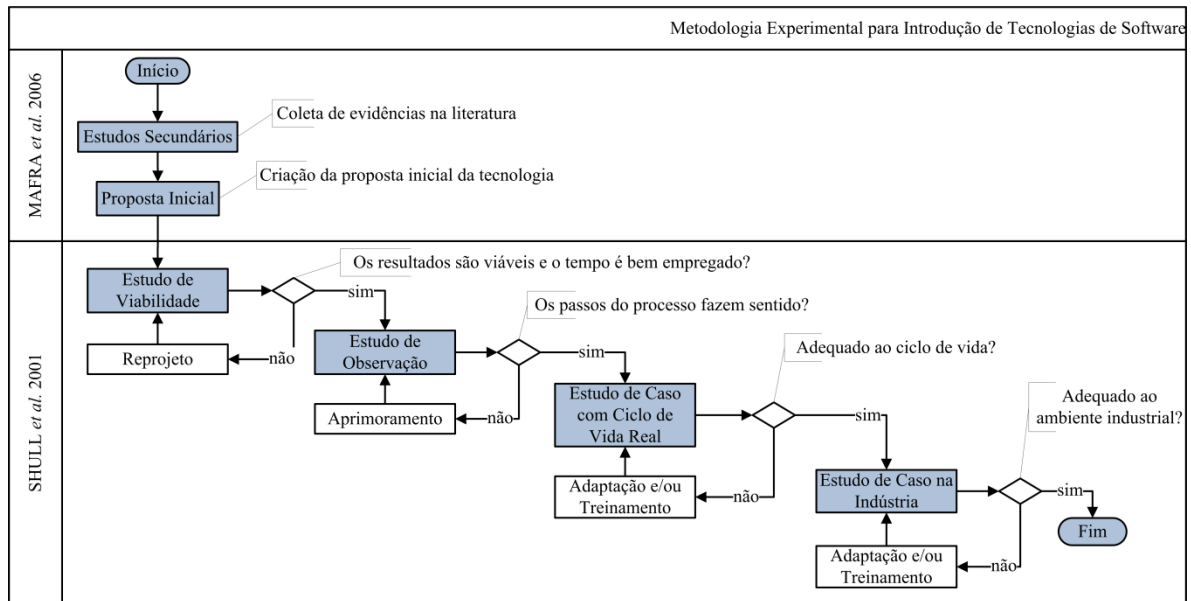
Espera-se que a abordagem proposta forneça meios de melhorar processos de software dentro de um ciclo contínua de melhoria, beneficiando-se da visão sistêmica e estruturada que a Teoria das Restrições oferece.

## **1.3 Metodologia**

Para apoiar a definição e estudo de viabilidade da abordagem proposta nesta dissertação, seguiu-se as três primeiras etapas (estudos secundários, proposta inicial,

estudo de viabilidade) da metodologia proposta originalmente por SHULL *et al.* (2001) e estendida por MAFRA *et al.* (2006). Esta abordagem para apoiar a introdução de tecnologias de software na indústria se fundamenta em estudos experimentais, permitindo observar o que funciona ou não na aplicação da tecnologia que está sendo introduzida.

A Figura 1.1 apresenta a metodologia composta por seis etapas, destacando as etapas da extensão proposta por MAFRA *et al.* (2006).



**Figura 1.1** - Visão geral da metodologia (CONTE, 2009)

Na primeira parte da metodologia, a definição inicial da tecnologia é realizada em duas atividades (CONTE, 2009):

1. Execução de estudos secundários para identificação, avaliação e interpretação de todos os resultados relevantes de acordo com a questão de pesquisa desejada;
2. Criação da versão inicial da tecnologia proposta com base nos resultados coletados dos estudos secundários.

A etapa de Estudos Secundários baseados em revisão sistemática atua como um meio para identificar, avaliar e interpretar o que há disponível e relevante na literatura sobre determinado assunto de interesse (MAFRA e TRAVASSOS, 2006). MAFRA *et al.* (2006) argumentam também que, além da identificação e da caracterização das evidências existentes sobre o assunto pesquisado, indícios de oportunidades de



melhoria e lições aprendidas a partir dos estudos identificados poderiam diminuir drasticamente os riscos associados à definição de uma nova tecnologia.

Com base nos resultados identificados e do conhecimento adquirido por meio dos Estudos Secundários, uma Proposta Inicial da tecnologia é definida.

Na segunda parte da metodologia, descrita em (SHULL *et al.*, 2001), a tecnologia proposta é refinada sucessivamente em etapas fundamentadas em estudos experimentais com o intuito de amadurecer e evoluir a tecnologia em observação antes de considerá-la apta para ser implantada na indústria. As etapas que compõem esta parte da metodologia são (MAFRA *et al.*, 2006):

- Estudo de Viabilidade: nesta etapa é verificada se a tecnologia é viável, atendendo de forma razoável aos objetivos inicialmente definidos. Além disso, a tecnologia pode ser refinada com base nas informações até então coletadas e nos estudos das etapas anteriores;
- Estudo de Observação: por meio de uma aplicação prática da tecnologia observada por pesquisadores, nesta etapa, é adquirida uma compreensão refinada sobre a tecnologia e as eventuais dificuldades enfrentadas pelos participantes;
- Estudo de Caso com Ciclo de Vida: nesta fase, já há indícios de que a tecnologia avaliada é efetiva. Contudo, é necessário que se verifique o efeito de sua aplicação ao ser colocada em interação com outras tecnologias em um ciclo de vida de desenvolvimento. O resultado deste estudo de caso pode servir para refinamento da tecnologia em estudo;
- Estudo de Caso na Indústria: analogamente à fase anterior, a tecnologia é aplicada em condições reais. No entanto, nesta etapa, o ambiente industrial é o cenário de utilização da tecnologia. Resultados advindos de observações da aplicação na indústria podem motivar aprimoramento da tecnologia.

As três primeiras etapas da metodologia apresentada foram utilizadas para investigar a oportunidade de pesquisa, definir e avaliar a viabilidade da abordagem proposta nesta dissertação.

A etapa de Estudos Secundários foi realizada por meio de um estudo baseado em revisão sistemática, cujo protocolo e resultados são apresentados no Anexo I. Na

etapa seguinte, a Proposta Inicial da abordagem proposta, descrita no Capítulo 4, foi desenvolvida e definida com base nos resultados encontrados no estudo executado na etapa anterior. Por fim, na etapa de Estudo de Viabilidade, um estudo para verificar a viabilidade da abordagem proposta foi executado. A descrição do planejamento e execução é apresentada no Capítulo 5.

## 1.4 Estrutura da dissertação

Esta dissertação está organizada em seis capítulos. Este capítulo apresentou a motivação para o desenvolvimento deste trabalho, o objetivo da pesquisa, a metodologia seguida na pesquisa e a organização da dissertação. A organização do texto desta dissertação segue a seguinte estrutura:

- **Capítulo 2 – Melhoria Contínua de Processo de Software:** Descreve os conceitos relacionados à melhoria de processos de software, focando a melhoria contínua para processos de software. Os resultados obtidos, a partir da execução do estudo baseado em revisão sistemática, descritos no Anexo I, são utilizados para caracterizar as abordagens existentes na literatura para melhoria contínua de processos de software;
- **Capítulo 3 – Teoria das Restrições:** Este capítulo apresenta a metodologia de resolução de problemas e melhoria contínua da Teoria das Restrições, descrevendo seus aspectos principais, sua evolução e sua aplicabilidade que justificam a utilização na abordagem proposta nesta dissertação;
- **Capítulo 4 – Abordagem Para Melhoria Contínua em Processos de Software Utilizando a Teoria das Restrições:** Apresenta o processo definido nesta dissertação de mestrado para apoiar a melhoria contínua de processo de software.
- **Capítulo 5 – Estudo de Viabilidade da Abordagem:** Apresenta a contextualização, definição, planejamento, execução e análise do estudo de viabilidade realizado para verificar a aplicabilidade da abordagem proposta em uma organização de software.
- **Capítulo 6 – Conclusão:** Apresenta as considerações finais sobre este trabalho, as limitações e contribuições da dissertação e perspectivas futuras.

- **Anexo I – Estudo Baseado em Revisão Sistemática:** Apresenta o estudo baseado em revisão sistemática realizado para identificar abordagens para melhoria de processo de software sob controle estatístico.
- **Anexo II – Descrição do Processo da Abordagem:** Apresenta a descrição detalhada e estruturada do processo proposta na dissertação.
- **Anexo III – Descrição dos Procedimentos dos Processos de Raciocínio:** Apresenta os procedimentos baseado nos Processos de Raciocínio da Teoria das Restrições utilizados durante a execução do processo da abordagem.
- **Anexo IV – Modelos de Formulário do Processo:** Apresenta os modelos de formulário utilizados durante a execução do processo da abordagem.
- **Anexo V – Informações e Artefatos Produzidos durante o Estudo de Viabilidade:** Apresenta as informações e formulários utilizados e gerados durante a execução do processo da abordagem no estudo de viabilidade.

## Capítulo 2 – Melhoria Contínua de Processo de Software

*Este capítulo apresenta conceitos relacionados à melhoria contínua de processos software, incluindo normas, modelos de maturidade de processos de software que advogam a melhoria contínua como prática importante. Além disso, são descritas as abordagens e técnicas utilizadas para apoiar a melhoria contínua encontradas no estudo baseado em revisão sistemática executado no contexto desta dissertação.*

### 2.1 Introdução

Organizações de desenvolvimento de software estão percebendo que seu principal desafio para produzir produtos de qualidade é gerenciar efetivamente o processo de desenvolvimento de software (NIAZI *et al.*, 2006). De acordo com KITCHENHAM *et al.* (1995), a qualidade de produtos de software deve ser considerada em três dimensões: (i) qualidade do produto em relação aos seus requisitos de qualidade; (ii) qualidade do processo que produz o produto; (iii) qualidade do produto no contexto do ambiente em que será utilizado (homologação).

O termo melhoria de processo de software foi inicialmente introduzido por HUMPHREY (1989) que estabeleceu que processos de desenvolvimento de software capazes e adaptáveis são prerequisites críticos para atingir níveis altos de qualidade em produtos de software.

HUMPHREY (1989) propôs um *framework* de níveis de maturidade de processos de software cujo objetivo era atingir a melhoria contínua com base na medição e no controle dos processos. Este *framework* marcou o início de iniciativas de melhoria de processos de software, tais como CMM/CMMI (SEI, 2010) e o *Software Process Improvement and Capability Determination* (SPICE) que deu origem à ISO/IEC 15504 (ISO/IEC, 2003), provendo às organizações meios de alcançar melhorias na qualidade dos seus produtos e processos de forma incremental.

Vários estudos foram direcionados à melhoria de processos de software, a fim de entender como a adoção de tais práticas poderia acontecer de maneira bem sucedida e quais os fatores críticos de sucesso para a sua implantação (GALINAC, 2009;

MONTONI, 2010). De maneira crescente, organizações desenvolvedoras de software têm adotado os modelos de maturidade e sistemas de qualidade como base para melhorar seus produtos e processos (SANTOS *et al.*, 2007).

A melhoria de processo de software incorporou conceitos da Gestão da Qualidade Total (TQM – *Total Quality Management*), proposta por DEMING (1986), através do seu ciclo de melhoria contínua PDCA (*plan, do, check, action*). No contexto de processo de software, BASILI (1993) propôs uma abordagem que considera um ciclo de melhoria – QIP (*Quality Improvement Paradigm*) – para melhorar o processo de desenvolvimento de software por meio do aprendizado organizacional contínuo.

O estudo baseado em revisão sistemática, executado como parte da metodologia adotada nesta dissertação, identificou diversos trabalhos referentes à melhoria contínua de processos de software. Este estudo teve como propósito inicial responder à questão de pesquisa: “quais métodos, técnicas, estratégias, abordagens e *frameworks* têm sido propostos para auxiliar a identificação de oportunidades de melhoria contínua de processo sob controle estatístico?”.

O resultado do estudo auxiliou em um melhor entendimento das diferentes técnicas e estratégias utilizadas para a melhoria contínua de processo de software, sendo fundamental para a confirmação da oportunidade de pesquisa e para a revisão sistemática de métodos, *frameworks* e abordagens que compõem este capítulo e que foram insumos para identificar a Teoria das Restrições como abordagem para a melhoria contínua de processos. Além do estudo, uma revisão informal da literatura identificou modelos de maturidade para processos de software, normas e trabalhos correlatos importantes quando se trata de melhoria de processo de software.

Este capítulo possui esta nota introdutória e está organizado da seguinte forma: na Seção 2.2 são apresentados os conceitos necessários para se compreender as dificuldades envolvidas na melhoria contínua de processos. Em seguida, na Seção 2.3, é explanado como as normas e modelos de maturidade tratam a melhoria contínua. Na Seção 2.4 e 2.5, são descritas as abordagens de Controle Estatístico de Processo e Seis Sigma, respectivamente. Na Seção 2.6, abordagens para apoiar a melhoria contínua são brevemente descritas, seguida de uma seção final, contendo as considerações finais do capítulo.

## **2.2 Processo de Software**

O desenvolvimento de software é uma tarefa complexa e influenciada por fatores técnicos, culturais e ambientais. As atividades de desenvolvimento de software se diferenciam por serem baseadas intensivamente no esforço humano e colaborativo; e por produzirem, na maioria das vezes, produtos diferentes a cada execução (HONG e GOH, 2003; BALDASSARRE *et al.*, 2004).

Uma definição de processo é descrita em (ISO/IEC, 2008) como sendo um conjunto de atividades interrelacionadas que transforma entradas em saídas. As entradas são requisitos do cliente ou produtos intermediários produzidos pelos processos ao longo do desenvolvimento de software; e as saídas também podem ser produtos intermediários que serão utilizados como insumo por outros processos ou o produto de software final.

No contexto de desenvolvimento de software, processo está mais diretamente relacionado ao ciclo de vida de software que provê uma estrutura para enquadrar e organizar os diferentes fatores e questões relacionados às atividades de desenvolvimento. Desta forma, processo de software pode ser definido como um conjunto coerente de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos que são necessários para conceber, desenvolver, entregar e manter um produto de software (FUGGETTA, 2000); e produtos associados (código fonte, manual, plano do projeto, casos de uso, casos de testes, etc.) (PAULK *et al.*, 1993).

SOLINGEN e BERGHOUT (1999) afirmam que a melhoria do processo de software é crucial para se melhorar a qualidade do produto de software, pois melhora as atividades por meio das quais o produto é construído. DYBA (2002) afirma que uma estratégia de melhoria de processo deve considerar a complexa teia de aspectos humanos e organizacionais, a fim de atingir os objetivos relacionados à qualidade do produto.

## **2.3 Melhoria Contínua nas Normas e nos Modelos de Processo**

A melhoria contínua de processo é tratada em normas e modelos de maturidade, tais como a ISO-9000:2000 (ISO/IEC, 2000), a ISO/IEC 15504 (ISO/IEC, 2003), a ISO/IEC 12207 (ISO/IEC, 2008), o MR-MPS (SOFTEX, 2011) e o CMMI (SEI, 2010). De acordo com essas abordagens, o ciclo contínuo de melhorias dos

processos de software deve instituir na organização uma estratégia que considere diferentes fontes de oportunidades de melhoria para tornar os processos mais eficientes e capazes.

Nos níveis mais altos dos modelos de maturidade, níveis A do MR-MPS e 5 do CMMI-DEV, uma estratégia deve garantir que os processos organizacionais sejam alvo contínuo de melhoria. Os modelos sugerem que os processos mais críticos para atingir os objetivos organizacionais sejam melhorados de forma inovadora ou a partir de refinamentos dos processos existentes. No entanto, por não se tratar do seu objetivo, nenhum procedimento ou especificação de como implementar o processo de melhoria contínua são apresentados.

As normas também requerem que os processos sejam observados de forma a melhorar sua eficiência e capacidade de maneira contínua. Contudo, analogamente aos modelos, as normas não provêm abordagens para implantar o processo de melhoria contínua.

### 2.3.1 ISO/IEC 15504

Evoluída a partir do projeto SPICE<sup>1</sup>, a normal internacional ISO/IEC 15504 – Avaliação de Processos (ISO/IEC, 2003) provê um *framework* para a avaliação de processos. A norma se propõe a servir como modelo de avaliação da capacidade dos processos e como sistematização de iniciativas de melhoria de processos.

A norma é baseada em um modelo bidimensional, contendo uma dimensão de processo e uma dimensão de capacidade. A dimensão de processo é definida por um Modelo de Referência de Processo externo que estabelece um conjunto de processos caracterizados por seus propósitos e resultados esperados. A dimensão de capacidade é determinada a partir de um arcabouço composto de seis níveis de capacidade de processos e seus atributos associados, como observado na

Tabela 2.1.

---

<sup>1</sup> Acrônimo para *Software Process Improvement and Capability Determination*.

**Tabela 2.1** - Níveis de Capacidade dos Processos da Norma ISO/IEC 15504 (ISO/IEC, 2003)

Nível	Descrição
0 – Incompleto	O processo não é implementado ou falha ao atingir seus objetivos
1 – Executado	O processo atinge os objetivos, porém sem padrão de qualidade e sem controle de prazos e custos
2 – Gerenciado	O processo é gerenciado (planejado e monitorado) e ajustado para satisfazer requisitos de qualidade, prazo e custos estabelecidos. Seus produtos de trabalho são gerenciados
3 – Estabelecido	O processo é definido a partir da adaptação de um processo padrão. Processo definido é executado e gerenciado de forma a garantir sua eficácia e eficiência em relação aos seus objetivos. Processo padrão é refinado a partir de informações de execução do processo
4 – Previsível	O processo é controlado e executado dentro de limites de variação definidos e com medições detalhadas e analisadas. O processo é alvo de melhoria, considerando a identificação e remoção das causas atribuíveis <sup>2</sup> de variação
5 – Em otimização	O processo é melhorado continuamente de maneira estratégica, considerando os objetivos de negócio relevantes

A partir do terceiro nível de capacidade (estabelecido), no atributo de processo PA 3.2 (Processo Implementado), o processo é alvo de melhoria. A norma ISO/IEC 15504 estabelece que dados apropriados devam ser coletados e analisados, de forma que se possa demonstrar a adequação e a eficácia do processo, e avaliar onde a melhoria contínua pode ser feita.

---

<sup>2</sup> Causas atribuíveis, ou causas especiais, estão relacionadas a desvios da execução normal do processo ocorrendo de maneira aleatória e acidental, como por exemplo: desvio de procedimento (falta de aderência ao processo), mudança nos insumos do processo, incompetência dos responsáveis (WHEELER e CHAMBERS, 1992).



A partir do quarto nível de capacidade (previsível), o processo é tratado com controle estatístico, produzindo um processo que é estável, capaz e previsível dentro de limites definidos. A norma estabelece que técnicas de análise e controle devam ser aplicadas quando necessárias, a fim de: (i) estabelecer limites de controle de variação do desempenho do processo; (ii) analisar dados relacionados às causas especiais de variação; (iii) aplicar ações corretivas para remover causas especiais de variação; e (iv) atualizar os limites de controle com base nas melhorias que são feitas no processo.

O ciclo de melhoria, no quinto nível de capacidade (em otimização), requer uma estratégia de melhoria, cujos objetivos de melhoria estejam alinhados com os objetivos de negócio relevantes, que garanta que o processo seja alvo de melhoria contínua, investigando oportunidades para implementar melhores práticas e/ou tecnologias inovadoras.

A norma cita requisitos para a melhoria de processo, ressaltando o ciclo de melhoria e as diferenças entre os níveis de capacidade, sem, no entanto, definir uma abordagem para implementar a melhoria de processo.

### **2.3.2 CMMI**

O CMMI (*Capability Maturity Model Integration*) (SEI, 2010) é um modelo de maturidade de melhoria de processo para desenvolvimento de produtos e serviços. Desenvolvido pelo SEI (*Software Engineering Institute*), este modelo consiste em melhores práticas relacionadas às atividades de desenvolvimento e manutenção, englobando o ciclo de vida do produto, desde a sua concepção até a sua entrega e manutenção.

O modelo é composto de 22 áreas de processo. Cada área possui uma declaração de propósito, objetivos específicos (relacionados a práticas específicas daquela área de processo) e objetivos genéricos (aplicados a múltiplas áreas de processo). Para cada objetivo, são definidos práticas, subpráticas e produtos típicos de trabalho.

O CMMI permite duas representações para a implementação de melhoria de processo: estagiada (por níveis de maturidade) e contínua (por níveis de capacidade). A representação estagiada agrupa conjuntos de áreas de processos em níveis de maturidade. Os níveis, em ordem crescente de maturidade, variam de 1 (inicial) a 5 (em otimização). Por sua vez, a representação contínua possibilita à organização

selecionar áreas de processos específicos que deseja melhorar. A avaliação com base na representação contínua é definida em termos de capacidade.

O SEI (SEI, 2010) classifica as áreas de processo em *Gerenciamento de Processo*, *Gerenciamento de Projeto*, *Engenharia*, *Recursividade e Iteração de Processos de Engenharia* e *Suporte*. A área de processo de *Gerenciamento de Processo* (APGP) contém atividades relacionadas à definição, planejamento, implementação, implantação, monitoramento, controle, avaliação, medição e melhoria de processos. A APGP está organizada em áreas de processo de: *Gerenciamento de Processo Básico* e *Gerenciamento de Processo Avançado*.

As APGP Básico agrupam áreas de processo que capacitam a organização a documentar e compartilhar melhores práticas, ativos de processos organizacionais e lições aprendidas na organização. As áreas de processo relacionadas são: *Foco no Processo Organizacional* (OPF); *Definição do Processo Organizacional* (OPD); e *Treinamento Organizacional* (OT).

As áreas de processo de *Gerenciamento de Processo Avançado* capacita a organização a atingir seus objetivos quantitativos de qualidade e desempenho de processo. As áreas de processo relacionadas são: *Desempenho do Processo Organizacional* (OPP); e *Gerenciamento do Desempenho Organizacional* (OPM).

Na área de processo *Desempenho do Processo Organizacional*, as linhas de base do desempenho de processo e modelos de desempenho são analisados para conhecer a habilidade da organização em atingir seus objetivos de negócio e a derivar em objetivos de qualidade e desempenho de processo. Com base neste conhecimento, a área de processo *Gerenciamento do Desempenho Organizacional* possibilita que a organização proativamente selecione e implemente melhorias incrementais ou inovações que possam, de maneira mensurável, melhorar o desempenho da organização (SEI, 2010).

Os objetivos específicos de OPM descrevem os resultados esperados para atingir a melhoria contínua e proativa nos níveis mais alto de maturidade. O primeiro objetivo específico de OPM especifica como o gerenciamento do desempenho de negócio é mantido em contínuo aprimoramento, ao passo que o segundo objetivo específico descreve de que maneira, a partir das oportunidades de aprimoramento, o desempenho organizacional será melhorado.

Embora o CMMI descreva os níveis de maturidade de maneira que oriente a organização de software a adotar a melhoria de processo e o ciclo de melhoria por meio de suas áreas de processo, a abordagem não especifica de que maneira a melhoria contínua de processo será implementada.

### 2.3.3 MR-MPS

O MPS.BR (SOFTEX, 2011) é um programa para Melhoria de Processo do Software Brasileiro coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) que se propõe a definir e aprimorar um modelo de melhoria e avaliação de processos de software com foco nas micro, pequenas e médias empresas.

No intuito de ser reconhecido tanto nacional como internacionalmente, o modelo é aderente a normas internacionais. O modelo de referência MR-MPS contém as definições dos processos com base na norma ISO/IEC 12207:2008 (ISO/IEC, 2008) e estabelece níveis de maturidade de processos com base nos perfis de capacidade estabelecidos na norma ISO/IEC 15504 (ISO/IEC, 2003). Além disso, o modelo é compatível com o modelo CMMI-DEV.

O MR-MPS define sete níveis de maturidade, descritos, a seguir, a partir de o mais alto nível para o mais baixo: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Gerenciado Parcialmente). Para cada nível de maturidade foi atribuído um perfil de processos e de capacidade de processos, de acordo com os níveis de capacidade da norma ISO/IEC 15504, possibilitando que a organização implemente os processos de forma mais gradual e os amadureça concomitantemente.

A melhoria de processos é iniciada a partir do nível E do MR-MPS, por meio do processo *Avaliação e Melhoria do Processo Organizacional* (AMP). Neste estágio, a melhoria é feita com base nas avaliações dos processos padrão da organização, a fim de identificar pontos fracos e oportunidades de melhorias.

A partir do nível B, os resultados de atributos de processos (RAP) especificam que um conjunto de processos críticos seja selecionado com base nos objetivos de negócio relevantes da organização. Tais processos devem ser controlados para se tornarem processos estáveis e previsíveis dentro de limites naturais do processo. Desta

forma, informações referentes ao desempenho dos processos são coletadas e analisadas com técnicas estatísticas, a fim de identificar variações anormais, cujas causas são atribuíveis a problemas de execução do processo.

A principal diferença entre o nível B e o nível A é que, no nível B, os esforços de melhoria estão voltados para a estabilização do desempenho do processo por meio da identificação e remoção das causas atribuíveis; ao passo que, no nível A, estrategicamente o desempenho do processo é melhorado por meio de melhorias incrementais ou inovadoras com base na análise das causas comuns.

Neste estágio, os objetivos de qualidade e desempenho são definidos com base nos objetivos de negócio relevantes. Deste modo, a estratégia de melhoria dos processos colabora para que a organização atinja seus objetivos de negócio atuais e futuros (SOFTEX, 2011). Neste contexto, os resultados de atributo de processo especificam que uma estratégia, que auxilie na realização de melhorias no desempenho do processo de maneira sistemática com foco nas causas comuns de variação do processo, aconteça de forma contínua.

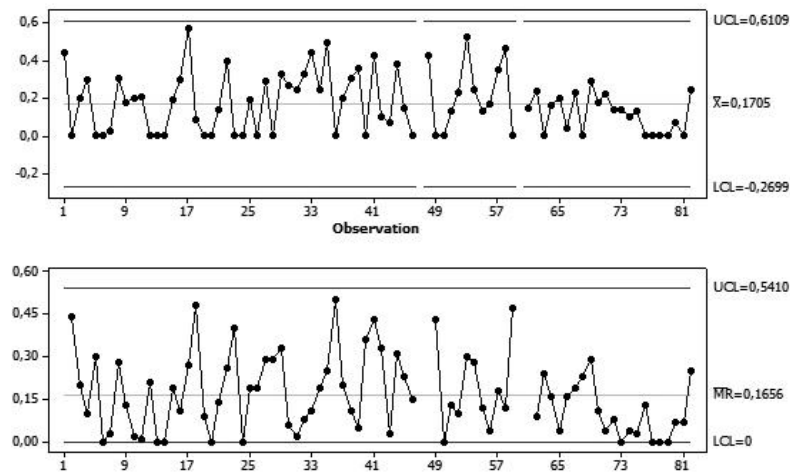
## **2.4 Controle Estatístico de Processo de Software**

O Controle Estatístico de Processo (CEP) tem sido amplamente utilizado na indústria manufatureira para controlar e melhorar processos. A técnica ganhou visibilidade a partir do caso bem sucedido da indústria manufatureira japonesa, desenvolvido por W. Edwards Deming (WHEELER e CHAMBERS, 1992) que forneceu ferramentas e diretrizes para transformar processos *ad-hoc* e caóticos em processos estáveis e previsíveis, sendo fortemente fundamentado na teoria de variação de desempenho de processo de Walter A. Shewhart (SHEWHART, 1939).

De acordo com Shewhart, um processo pode ser caracterizado por medidas de atributos que variam ao longo do tempo devido a variações de causas comuns ou causas atribuíveis. Caso a variação do desempenho do processo esteja relacionada apenas a causas comuns, o processo é dito estável e seu comportamento é previsível dentro de um intervalo de variação (limites de controle). Do contrário, o processo é dito instável e possui causas atribuíveis (ou causas especiais) relacionadas a problemas na sua execução.

Processos mais automatizados e altamente dependentes de máquinas tendem a variar de forma mais previsível devido a causas comuns, tal como acontece com os processos de manufatura. Em processos com uma participação humana mais intensa, tal como acontece nos processos de software, há mais probabilidade de ocorrência de causas especiais que dificilmente podem ser controladas, como por exemplo: motivação (ou ausência dela), ânimo diferenciado por causas pessoais (problemas familiares, perda do time favorito etc.).

Os gráficos de controle apresentam o desempenho do processo na linha do tempo, juntamente com os limites de controle e a linha central de tendência, como pode ser visto no exemplo da Figura 2.1, possibilitando distinguir entre causas comuns e causas atribuíveis. Os valores fora dos limites de controle alertam sobre a existência de possíveis causas atribuíveis e devem ser analisados para identificar as causas-raiz e prevenir suas recorrências. Além disso, outros padrões de comportamentos anômalos observados por meio dos gráficos de controle podem antecipar a identificação de causas atribuíveis (WHEELER e CHAMBERS, 1992).



**Figura 2.1** - Exemplo de gráfico de controle  $\bar{X}\bar{R}$

O CEP aplicado para melhoria de processo de software foi descrito por FLORAC e CARLETON (1999), estabelecendo bases para a estabilização de processos de software e seu controle estatístico.

A abordagem descrita por FLORAC e CARLETON estabelece um ciclo de melhoria que considera as seguintes ações: (i) caso o processo não seja estável, identificar causas atribuíveis de instabilidade e removê-las, a fim de prevenir suas recorrências; (ii) caso o processo não seja capaz (não atinge objetivos organizacionais

ou necessidades do cliente), identificar e propor melhorias, e implementar as mudanças necessárias para alcançar a capacidade desejada; e (iii) caso o processo seja estável e capaz, continuamente melhorar o processo de forma a reduzir a variabilidade e melhorar a qualidade, o custo e o tempo de entrega (FLORAC e CARLETON, 1999).

Entretanto, a aplicação do controle estatístico de processos para a indústria de software ainda não está madura e tem sido área de estudos (GARDINER e MONTGOMERY, 1987; LANTZY, 1992; CARD, 1994; KAN, 1994; RADICE, 1998; FLORAC e CARLETON, 1999; LEWIS, 1999; WELLER, 2002; CARD, 2004; KOMURO, 2006; SARGUT e DEMIRÖRS, 2006). Diferentemente da manufatura, algumas características inerentes do desenvolvimento de software dificultam a aplicação do CEP para processos de software, como por exemplo: dificuldade de definir medidas que garantam o monitoramento da qualidade do produto final (KAN, 1994); produção de itens únicos, em detrimento de produções de partes idênticas, como acontece na manufatura (CARD, 1994); diversidade de fatores que influenciam o desempenho do processo de desenvolvimento (LINDVALL e RUS, 2000); complexidade no processo de criação (CARD, 2004); atividades preponderantemente baseadas em esforço criativo e humano (BALDASSARRE *et al.*, 2007); dentre outros.

BALDASSARRE *et al.* (2007), com base em uma revisão sistemática sobre CEP, apontaram diversos benefícios observados nos trabalhos e relatos de experiência da aplicação do controle estatístico de processos para empresas de software, dentre os quais: a possibilidade de estudar o desempenho do processo e estimar comportamentos futuros; identificar comportamentos anômalos no desempenho, a fim de antecipar ações de melhoria para evitar problemas; embasar decisões estratégicas; e introduzir uma abordagem sistemática para produção e melhoria.

Outros estudos foram publicados apontando diferenças na aplicação do controle estatístico de processos para o desenvolvimento de software. Segundo SARGUT e DEMIRÖRS (2006), o controle estatístico pode não ser aplicável a todos os processos de software, assim como nem todas as técnicas utilizadas do CEP são viáveis. JALOTE e SAXENA (2002) sugerem que os limites de controle devem ser mais estritos do que os propostos para manufatura e que processos como inspeção (revisão por pares), teste e manutenção mostram-se adequados para a aplicação do controle estatístico.

Algumas adaptações foram propostas para adequar o CEP à indústria de software. LANTZY (1992) argumenta que, para uma aplicação bem sucedida de controle estatístico para processos de software, as medidas devem ser correlacionadas às características de qualidade dos produtos definidas pelo cliente; e relacionadas às atividades que produzem itens tangíveis.

FLORAC e CARLETON (1999), a respeito das medidas coletadas, enfatizam a importância de definições operacionais bem definidas, observando questões relacionadas à homogeneidade e subagrupamento dos dados, e à utilização do gráfico de controle correto para cada tipo de dado. Os autores também ressaltam o entendimento sobre múltiplas causas e sistema de causas diversas no momento de analisar os dados, potencializando os benefícios dos gráficos de controle. Neste contexto, TARHAN e DEMIRÖRS (2006) propõem uma abordagem para guiar a identificação de subagrupamentos adequados para um processo, assim como as medidas mais apropriadas para aplicação do controle estatístico para processos de software.

A estabilização do desempenho do processo de software é discutida em (JALOTE e SAXENA, 2002). A escolha dos limites de controle determina o quão frequente um “alarme falso”<sup>3</sup> ocorre ou causas atribuíveis não são alertadas por conta de limites de controle inadequados. JALOTE e SAXENA (2002) propõem uma abordagem para determinar limites de controle ótimos, considerando um modelo de custo para minimizar o desperdício resultante destes erros. De acordo com os autores, limites de controle ótimos para processos como os de controle da qualidade e os de inspeção mostram-se ser mais estritos dos que os utilizados na manufatura.

A abordagem de Controle Estatístico de Processo aplicado à melhoria de processos de software possui diversas vantagens. Embora não haja estudo conclusivo sobre a melhor forma de aplicar as ferramentas estatísticas utilizadas pelo CEP para processos de software, trabalhos encontrados no estudo baseado em revisão sistemática sugerem melhorias nas técnicas. No entanto, tais abordagens não constituem uma sistemática para a melhoria contínua de processo de software, sendo utilizadas apenas na identificação de anomalias no comportamento dos processos.

---

<sup>3</sup> Valores que estão fora dos limites de controle e que, no entanto, são resultados de causas comuns de variações.

## 2.5 Seis Sigma Aplicado à Melhoria de Processo de Software

Seis Sigma é uma metodologia para melhoria da qualidade focada na melhoria do desempenho organizacional de forma disciplinada, utilizando técnicas quantitativas e estatísticas, desenvolvida pela Motorola no final da década de 80 (HAHN *et al.*, 1999). O principal objetivo da metodologia é implementar uma estratégia baseada em medição com foco na melhoria de processo e na redução da variação do desempenho (ANTONY, 2004).

Assim como o Controle Estatístico de Processo, o Seis Sigma também foi desenvolvido como uma iniciativa da Gestão da Qualidade Total – TQM (BLACK e REVERE, 2006). Para alguns autores (ANTONY, 2004; SCHROEDER *et al.*, 2008), no entanto, o Seis Sigma possui aspectos vantajosos quando comparado com iniciativas de TQM anteriores, como por exemplo: (i) *framework* para implementação da abordagem, constituindo um processo disciplinado para atingir os objetivos de melhoria; (ii) infraestrutura de especialistas (*Master Black Belts, Black Belts, Green Belts*) para liderar e implantar a abordagem; (iii) conceito estatístico de variação de processo aplicado, encorajando a redução de defeitos por meio da redução de variabilidade do processo; (iv) forte foco em resultados financeiros e de negócio, em detrimento de foco no processo apenas.

Tecnicamente, o sigma ( $\sigma$ ) é um indicador de variação do processo. Ele representa uma unidade de desvio padrão que representa o grau de dispersão de uma distribuição normal de uma amostra (KHAN e ZHANG, 2008). Seis sigma de variação abrangem 3,4 defeitos por milhão de oportunidades. Um defeito é qualquer evento que não atenda aos requisitos do cliente. Ao conceber, por exemplo, a qualidade do produto em termos de sigma, indica-se que a qualidade final do produto não depende apenas de um bom *design* de produto, mas também da capacidade do processo de produção (HONG e GOH, 2003).

Diversos *frameworks* de implementação do Seis Sigma foram propostos, dos quais TONINI (2006) cita: MAIC (Medir, Analisar, Melhorar, Controlar), DMAIC (Definir, Medir, Analisar, Melhorar, Controlar), *Breakthrough Strategy* (Identificar, Definir, Otimizar, Validar), IDOV (Identificar, Projetar, Otimizar, Validar), DMEDI (Definir, Medir, Explorar, Desenvolver, Implantar).



Dentre as fases do Seis Sigma que compõem os diferentes *frameworks*, a fase Otimizar se destaca por constituir a etapa na qual a definição do produto ou do processo alvo da melhoria é analisado, a fim de minimizar a influência de suas fontes de variabilidade (ANTONY, 2002). Técnicas estatísticas são utilizadas para analisar a variabilidade do desempenho e suas fontes, a fim de apoiar os executores na definição de melhorias, sem, no entanto, propor uma técnica, além da técnica existente na abordagem de CEP, para identificar fontes de variabilidade e removê-las.

No contexto acadêmico, o Seis Sigma tem sido alvo de pesquisa voltada para a melhoria de processo de software (GALINAC e CAR, 2007; PARK *et al.*, 2007; BOFFOLI *et al.*, 2008; GONÇALVES *et al.*, 2008; KHAN e ZHANG, 2008; XIAOSONG *et al.*, 2008). O mesmo debate sobre a aplicabilidade do CEP para processos de software reincide no Seis Sigma, pois ambos fundamentam-se fortemente em evidências quantitativas e estatísticas.

Para MARUGAPPAN e KEENI (2003), a melhoria de processo de software através do Seis Sigma requer uma conhecimento prévio das suas variáveis mensuráveis, de forma que se conheça o comportamento do processo antes e depois da melhoria, a fim de apurar a efetividade da melhoria. ECKES (2001) cita seis principais elementos, nomeados *componentes contribuidores*, que estão envolvidos e, portanto, geram variação em processos, a saber: máquinas, materiais, métodos ou procedimentos, sistemas de mensuração, ambiente e pessoas.

Levando em consideração as características distintas do processo de software, HONG e GOH (2003) propõem uma aplicação do Seis Sigma através da estratégia DMAIC para melhoria contínua de processos de software. A abordagem agrega as fases de desenvolvimento de software (planejamento, elicitação de requisitos, análise de requisitos, *design*, codificação, integração, testes e homologação) nos passos da estratégia DMAIC (definir, medir, analisar, melhorar e controlar). Durante a fase Melhorar, idéias e soluções são desenvolvidas, considerando a eliminação de defeitos, o uso de experimentos e a identificação de possíveis soluções.

TONINI (2006) propõe uma estratégia específica para implantação do Seis Sigma nas organizações desenvolvedoras de software denominado SW-DMAIC, baseada em alguns métodos já consagrados, dentre os quais *Breakthrough Strategy*, IDOV e DMEDI. O autor relaciona características mensuráveis, considerando os

diversos motivos e origens dos processos de desenvolvimento de software, para monitoramento, controle e investigação de problemas, como mostra a Tabela 2.2.

Embora, a estrutura provida pelo Seis Sigma seja robusta e tenha sido aplicada de forma bem sucedida na indústria e mais recentemente na engenharia de software, alguns autores citam barreiras na abordagem a serem superadas, principalmente no âmbito da melhoria contínua e inovação, como por exemplo: a absorção de novas tecnologias e idéias para melhorar a capacidade dos processos (ANTONY, 2004; SCHROEDER *et al.*, 2008).

**Tabela 2.2** - Lista de origens, motivos e características mensuráveis (TONINI, 2006)

Origem	Motivo	Características mensuráveis
Organização		Receita Lucratividade Novos projetos
Destinatários		Satisfação Atendimento Requisitos
Produto	Humanos Técnicos Padrões Sistemas Ambiente	Tamanho Mudanças Quantidade de erros/falhas Sintomas de falhas Duração das falhas Resultado das falhas Severidade das falhas Mecanismo das falhas Custos das falhas Duração das falhas
Gerenciamento	Outros	Prazo Custo Riscos
Desenvolvedores		Capacidade de atendimento Esforço Disponibilidade Assertividade Proficiência Produtividade
Terceiros		Gerência Produtividade Treinamento
Tecnologia		Complexidade Taxa de mudança
Processo		Cumprimento de padrões Quantidade de defeitos Remoção de defeitos
Recursos físicos		Quantidade envolvida Produtividade

## **2.6 Abordagens para Apoiar a Melhoria Contínua de Processos de Software**

No estudo baseado em revisão sistemática desenvolvido no escopo desta dissertação, identificou-se abordagens que tratam de melhoria contínua em processos de software sob controle estatístico. A descrição da pesquisa, protocolo e resultados do estudo podem ser encontrados no Anexo I. Além deste estudo baseado em revisão sistemática, uma revisão da literatura com base em pesquisa e indicação de especialistas da área de melhoria de processo de software foi realizada, identificando trabalhos relacionados à melhoria de processo de software e importantes ao contexto desta dissertação.

Esta seção apresenta um breve histórico do estudo e algumas abordagens identificadas no estudo baseado em revisão sistemática e na revisão da literatura, descrevendo sucintamente suas características, pontos fortes e limitações.

### **2.6.1 Breve Histórico do Estudo Baseado em Revisão Sistemática**

Uma questão principal de pesquisa foi inicialmente definida para identificar quais abordagens tinham sido propostas para auxiliar a identificação das oportunidades de melhoria contínua em processos de software sob controle estatístico. O interesse inicial da pesquisa era propor uma abordagem que apoiasse organizações maduras a otimizarem seus processos sob controle estatístico. A partir desta questão de pesquisa, questões secundárias foram definidas, objetivando identificar também as abordagens que apoiam o processo de investigação da oportunidade de melhoria, assim como as informações que deveriam ser consideradas para auxiliar o processo investigativo para a melhoria de processo.

Com base nestas questões de pesquisa, um estudo preliminar foi executado. Alguns trabalhos encontrados deste estudo focavam em melhorias pontuais, baseando-se em avaliações dos processos, tais como avaliações de aderência dos processos e produtos em relação aos padrões, avaliações de modelos de maturidade; e investigação de oportunidades de melhorias, a partir das informações advindas dos projetos (análise *postmortem*, sugestões de melhorias dos executores, etc.). Dentre estes trabalhos, destacam-se as abordagens para melhoria de processo de software propostos por

BIANCHI (2008a) e ALBUQUERQUE (2008). A abordagem de ALBUQUERQUE (2008) serviu como modelo para o processo definido nesta abordagem.

A partir do estudo preliminar, cinco artigos considerados de controle foram identificados. No total, foram executadas quatro iterações de teste do protocolo. Durante as iterações, a expressão de busca evoluiu de forma a focar na melhoria contínua de processos de software sob controle estatístico.

Na última execução, o estudo retornou 475 trabalhos por meio dos mecanismos de busca, sendo que apenas 10 foram selecionados com base nos critérios definidos no protocolo. A maioria dos trabalhos se concentrava na utilização do controle estatístico de processos para estabilizar os processos ou refinar as ferramentas e conceitos do controle estatístico para o processo de desenvolvimento de software.

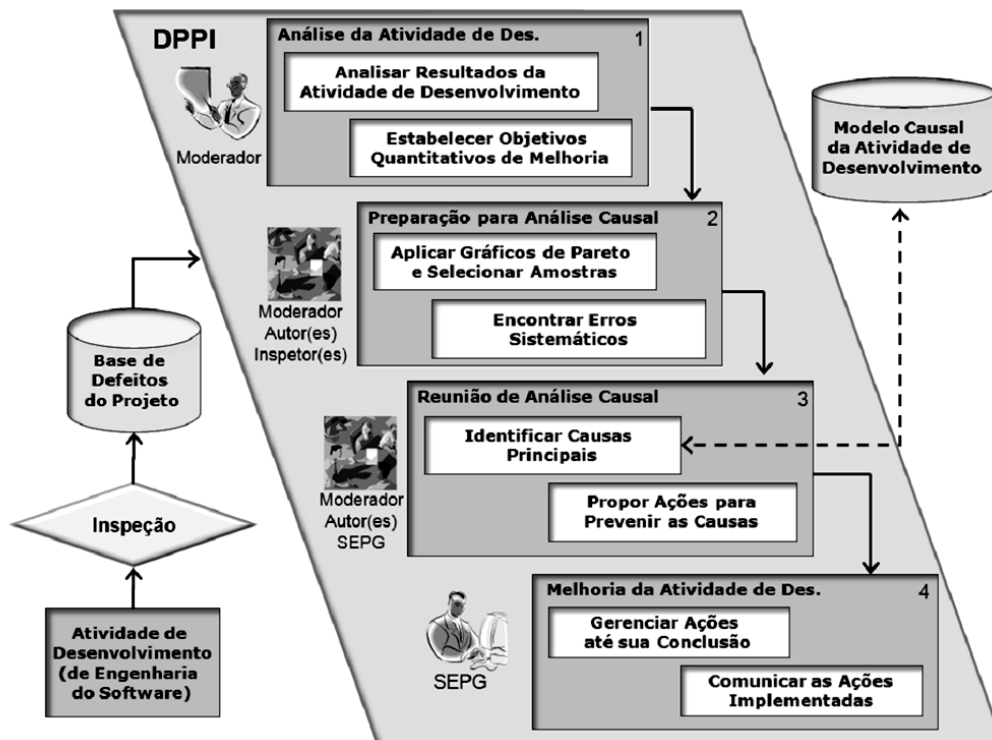
### **2.6.2 Abordagem para Prevenção de Defeitos Provenientes de Inspeções para Apoiar a Melhoria dos Processos de Engenharia do Software**

A abordagem para prevenção de defeitos provenientes de inspeções para apoiar a melhoria dos processos de engenharia do software (DPPI – *Defect Prevention-based Process Improvement*), proposta em (KALINOWSKI, 2011), representa uma abordagem baseada em análise causal de defeitos para melhoria de processos. De acordo com KALINOWSKI, a principal inovação da abordagem é a integração de conhecimento obtido em sucessivas reuniões de análise causal para prover um maior entendimento a respeito das relações de causa-e-efeito dos defeitos da organização através de modelos causais.

A abordagem agrega um mecanismo de aprendizado organizacional sobre as relações de causa-e-efeito na condução da análise causal, contribuindo para um entendimento mais aprofundado do contexto organizacional e, conseqüentemente, para a melhoria de processo por meio de uma análise causal de defeitos mais eficiente.

A Figura 2.2 fornece uma visão geral das atividades da abordagem DPPI. Na atividade *Análise de Atividade de Desenvolvimento*, com base na medição e controle das atividades de desenvolvimento, os defeitos detectados são analisados e comparados com dados históricos. Neste momento, objetivos quantitativos são estabelecidos, considerando o controle e a estabilidade (ou instabilidade) do processo em análise. Caso instável, o objetivo deve visar à remoção de causas atribuíveis. Caso

contrário, um objetivo de desempenho, com base na linha de base do desempenho do processo, é definido.



**Figura 2.2** - Visão Geral da Abordagem DPPI (KALINOWSKI, 2011)

Na atividade *Preparação para Análise Causal*, gráficos de controle dos processos sob controle estatístico e os diagramas de Pareto são analisados e podem ser utilizados para refinar, de forma mais realística, os objetivos e os parâmetros estabelecidos. Nesta etapa, os erros sistemáticos são identificados e sua causa-raiz tratada. O conhecimento registrado no repositório do sistema causal está disponível e pode ser utilizado para identificar os erros sistemáticos. O resultado da análise é registrado no repositório.

Após selecionar a amostra de defeitos, na atividade *Reunião de Análise Causal*, o diagrama de causa-e-efeito é inicialmente preenchido com as causas relacionadas ao erro sistemático em análise. A partir do modelo causal, as probabilidades bayesianas para cada causa são calculadas. Considerando as probabilidades bayesianas, o conhecimento sobre ações que poderiam solucionar os problemas e lições aprendidas são levadas em consideração no momento de propor ações preventivas.

Por final, na atividade *Melhoria da Atividade de Desenvolvimento*, as ações propostas são implementadas por uma equipe competente e gerenciadas até a sua

conclusão. O repositório é alimentado sobre o resultado das ações e esforços de implementação como lições aprendidas.

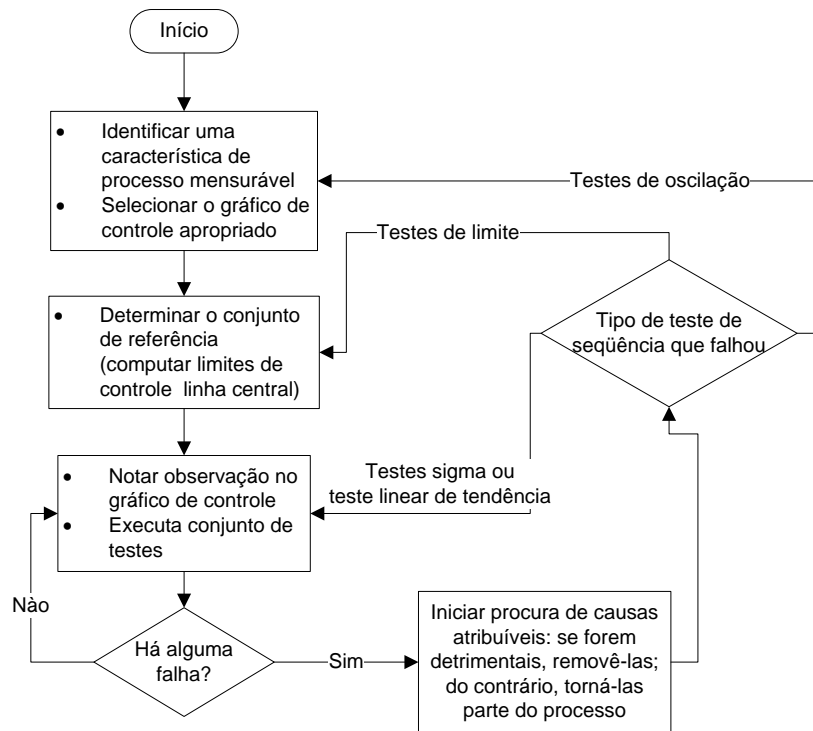
A abordagem proposta por KALINOWSKI (2011) considera processos de software ainda não estabilizados e processos que já estão sob controle estatístico. A remoção de causas comuns de variação por meio de um modelo causal relacionado a defeitos é oportuna para organizações que já possuem uma cultura de análise causal (organizações de alta maturidade). No entanto, o foco de análise causal desconsidera outras fontes de oportunidades de melhoria ou inovações. Além disso, a elaboração de melhoria no processo de software fundamenta-se na existência de um defeito. Não se observa na abordagem DDPI uma sistemática para desenvolver proposta de melhoria para melhorar o desempenho organizacional de maneira mais abrangente.

### **2.6.3 Melhoria Contínua de Processo de Software através do Controle Estatístico de Processo**

A abordagem de melhoria contínua de processo de software através de CEP propõe uma aplicação dos conceitos e ferramentas do controle estatístico de processo, reinterpretando-o no contexto de produção de software (CAIVANO, 2005b). CAIVANO (2005b) argumenta que as características dos processos de software devem ser consideradas no momento de aplicar o CEP e de interpretar seus resultados.

Um conjunto de testes de controle estatístico é compilado para ser utilizado nos gráficos de controle, a fim de gerenciar o processo de software. Informações de como apresentar, interpretar e organizar os testes são providas, considerando o ponto de vista do desenvolvimento de software.

O processo de investigação, como pode ser visto na Figura 2.3, sugere que o desempenho do processo seja monitorado sistematicamente. Após determinar o conjunto de referência de desempenho do processo (limites de controle e linha central), a cada valor no gráfico de controle, os testes devem ser executados.



**Figura 2.3** - Processo de Investigação (CAIVANO, 2005b)

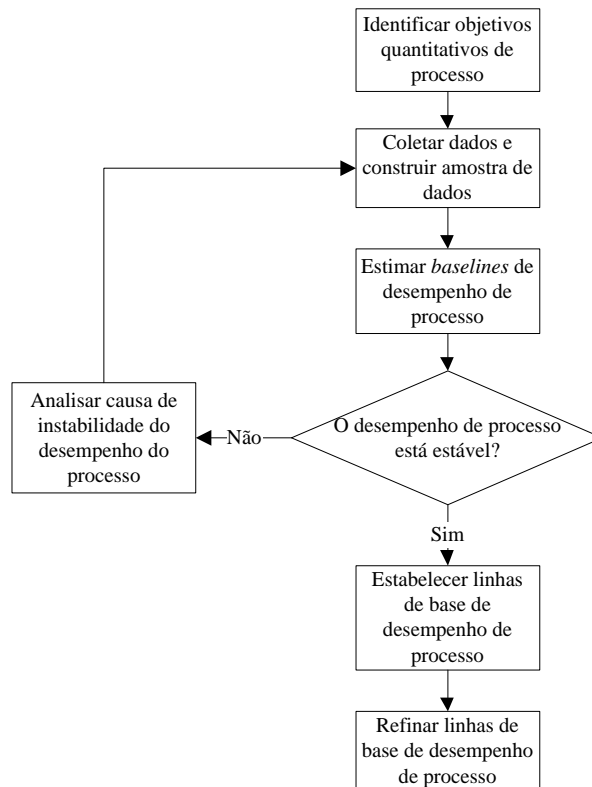
Uma base de experiência é responsável por manter o conhecimento adquirido com a monitoração e resolução de problemas. Por meio de uma tabela de decisão, informações são coletadas, referentes ao gráfico de controle utilizado, condições de contexto, conjunto de testes de estabilidade aplicados, *insights* de solução de gerentes e entendimento da variação do processo durante as atividades de monitoramento.

BOFFOLI (2006) desenvolve o trabalho de CAIVANO (2005b), evoluindo a tabela de decisão e adicionando mais detalhes técnicos sobre os gráficos de controle utilizados. Além disso, BOFFOLI (2006) considera também os conceitos de Calibragem Dinâmica – CD (*Dynamic Calibration*) que tem como objetivo adotar e utilizar um modelo de estimativas até que este esteja devidamente ajustado.

Os principais benefícios da abordagem indicados experimentalmente são: (i) o uso de dados históricos de processos e características de contexto para estabelecer linhas de base (*baselines*) mais realistas; (ii) ser um método não intrusivo. No entanto, embora a abordagem proponha uma sistemática para identificar variações ou tendências anormais por meio do controle estatístico de processo, não há uma estratégia para identificar oportunidades de melhoria contínua nos processos estabilizados.

Dentre os trabalhos identificados no estudo baseado em revisão sistemática, duas abordagens apoiam a melhoria contínua de processo de software por meio de refinamentos das técnicas de controle estatístico, objetivando melhorar o desempenho dos processos da organização. O método BSR, proposto por WANG *et al.* (2006), descreve passos para coletar e medir os processos da organização, com o intuito de estabelecer suas linhas de base (*baselines*) de desempenho e refiná-las continuamente.

O método, apresentado na Figura 2.4, é composto pelos seguintes passos: (i) identificar objetivos quantitativos do processo; (ii) coletar dados e construir amostra de dados; (iii) estimar linhas de base de desempenho do processo (alvo); (iv) analisar as causas de instabilidade no desempenho do processo; (v) estabelecer linhas de base de desempenho do processo; e (vi) refinar linhas de base de desempenho do processo.



**Figura 2.4** - Abordagem BSR (WANG *et al.*, 2006)

De acordo com os autores, após a estabilização dos processos, com base em um entendimento quantitativo das causas comuns de variação inerentes ao processo, o desempenho é refinado continuamente. No entanto, a abordagem proposta não



específica como a identificação da oportunidade de melhoria a partir do entendimento das causas comuns irá acontecer.

CHANG e CHU (2008) propõem uma abordagem para melhoria da análise causal utilizando um controle multivariado estatístico de processo. A análise causal é importante para a melhoria contínua, pois apoia a identificação de problemas e oportunidades de melhoria. Os autores argumentam que um dos problemas de utilizar o controle estatístico (univariado) de processo, como o proposto por SHEWHART (1939), é que eventos de processo de software são observados como variáveis individuais e que apenas um número limitado de variáveis de desempenho é monitorado por vez.

A abordagem utiliza uma sistemática chamada *Multilevel Software Cause Identification* (MSCI) que aplica gráficos  $T^2$  de Hotelling – uma técnica de controle multivariado estatístico de processo. A estrutura de dados do MSCI é organizada em vários níveis de hierarquia, incluindo as áreas de problemas comuns, medidas e atributos.

Um conjunto de medidas (variáveis correlacionadas) é inicialmente analisado por meio de um gráfico  $T^2$ . Se houver variações fora do controle, os resultados obtidos são decompostos, por meio de técnicas de análise de regressão, para evidenciar valores e relações que influenciaram de maneira significativa os sinais fora de controle. Desta maneira, CHANG e CHU (2008) argumentam que o desempenho do processo é otimizado, pois diminui o número de *falsos positivos*, servindo como melhor parâmetro para diferenciar falsos alarmes de causas atribuíveis para limites de controle mais estritos. A melhoria dos gráficos de controle, no entanto, não compõe uma abordagem para a melhoria contínua.

#### **2.6.4 Melhoria de Processos de Software através do Seis Sigma**

Algumas organizações de software têm aplicado o Seis Sigma como meio de melhoria contínua. No contexto de organizações de alta maturidade, GONÇALVES *et al.* (2008) descrevem uma estratégia que propõe identificar, classificar e priorizar ações de melhoria e inovação para organizações de alta maturidade ou que apliquem o Seis Sigma. A estratégia é composta pelas seguintes fases:

1. Identificação – utilização da estratégia DMADV (Definir, Medir, Analisar, Formular, Verificar) para definir os objetivos de desempenho do processo, considerando os objetivos de negócio e a capacidade dos processos.
2. Classificação – ações de melhoria e inovação são classificadas de acordo com a sua complexidade e o conhecimento sobre as causas dos problemas a serem tratados e sobre os benefícios atingidos com as ações. As ações são categorizadas em: ações simples; PDCA (ações de nível médio de complexidade e que implementam um ciclo de melhoria); DMADV (ações de inovações de impacto e benefícios desconhecidos); DMAIC (ações de melhoria de capacidade de escopo organizacional).
3. Priorizar – as relações de custo/benefício das melhorias são analisadas com base em critérios pré-definidos, como, por exemplo: recursos e dados disponíveis, complexidade, tendência, urgência, valor agregado, dentre outros.

A estratégia de GONÇALVES *et al.* (2008) pode apoiar a institucionalização de práticas relacionadas aos processo de *Análise Causal e Resolução* (CAR) e *Inovação e Desenvolvimento Organizacional* (OID). A estratégia considera a melhoria de capacidade dos processos estáveis, contudo, não detalha passos para identificação de oportunidades de melhoria contínua.

Outra abordagem para melhoria de processo de software é proposto por PARK *et al.* (2007). Os autores definem um *framework* para utilizar ferramentas Seis Sigma em PSP/TSP. O PSP (*Personal Software Process*) e o TSP (*Team Software Process*) são abordagens desenvolvidas pelo SEI para auxiliar a implantação de programas de melhoria de processo de software, considerando, respectivamente, desenvolvimento de software em nível individual e em nível de equipe.

Contudo, o PSP/TSP não contempla ferramentas estatísticas para apoiar o controle de processos (PARK *et al.*, 2007). Desta forma, os autores combinaram vários elementos do PSP/TSP, tais como métricas, modelos de estimativa e objetivos do PSP/TSP com ferramentas de Seis Sigma. Os elementos do PSP/TSP que precisam ser analisados ou melhorados foram alvo do mapeamento, como pode ser visto na Tabela 2.3.

As fases do PSP (planejamento, desenvolvimento, *post-mortem*) e do TSP (iniciação da equipe, estratégia de desenvolvimento, plano de desenvolvimento, desenvolvimento de requisitos, *design*, implementação, integração e teste, *post-mortem*) são mapeadas com as ferramentas do Seis Sigma, indicando-se o ciclo (quando necessário), passo e propósito.

**Tabela 2.3** - Elementos PSP/TSP e ferramentas de Seis Sigma correspondentes (PARK *et al.*, 2007)

Elementos PSP/TSP		Funções	Ferramenta Seis Sigma correspondente
Medidas PSP/TSP	Medida de defeito	Priorizar os tipos de defeito para encontrar defeitos mais frequentes	Análise Pareto
		Identificar a causa dos defeitos mais frequentes	Diagrama de causa-e-efeito (Ishikawa)
		Avaliação da estabilidade do processo	Gráficos de controle
	Outras medidas	Análise de relacionamento entre as medidas	Gráfico de dispersão, análise de correlação, análise de regressão
Proxy Based Estimating (PROBE) (HUMPHREY, 1997)	Determinar o nível de adequação do PROBE	ANOVA (STAPLETON, 2009)	
Processo	Documentar o fluxo do processo para facilitar o entendimento de processos complexos	Mapeamento de Processo	
Necessidades do cliente	Elicitação, priorização e análise de requisitos	Análise Kano (SHEN <i>et al.</i> , 2000), SQFD (LIU, 2000)	
Dados de estimativa	Determinar a acurácia de estimativa	T-teste de duas amostras	
Riscos de projeto	Avaliação de riscos	SWFMEA (PENTTI e ATTE, 2002)	

As fases do TSP prevêm um ciclo de melhoria através do passo de revisão do próprio processo à procura de causa-raiz de problemas. O mapeamento feito por PARK *et al.* relaciona os diagramas de causa-e-efeito do Seis Sigma à fase de melhoria do TSP. O controle estatístico por meio dos gráficos de controle é aplicado

nas fases de *design*, desenvolvimento, inspeção, integração e teste. No entanto, não é mencionada nenhuma estratégia para melhoria contínua desses processos.

### **2.6.5 Abordagem para Avaliação de Propostas de Melhoria Em Processos de Software**

Normas e modelos que tratam de melhoria de processos de software, tais como os citados neste capítulo, sugerem que mudanças aplicadas nos processos de software seja avaliada, a fim de prever resultados e impactos relacionados às mudanças. No contexto da melhoria contínua de processo de software, deve ser avaliado se há necessidade de prever resultados preliminares sobre proposta de melhoria dos processos organizacionais. Esta avaliação é fundamental para que a melhoria de processo ocorra de maneira a maximizar suas chances de sucesso e minimizar impactos indesejáveis no desempenho de processo.

O trabalho proposto por SILVA FILHO (2006) identifica quatro instrumentos utilizados para avaliar a efetividade de propostas de melhoria, a saber: estudos experimentais, simulação e pilotos. Por meio de um estudo baseado em revisão sistemática, SILVA FILHO (2006) identifica várias abordagens que utilizam tais instrumentos para avaliar propostas de mudanças para a melhoria de processos de software.

O autor propõe uma abordagem para avaliação da efetividade de propostas de melhoria de processo de software baseado em piloto, apresentada na Figura 2.5. Além do apoio ferramental através do *Pilot*<sup>4</sup>, fornecendo indicadores relacionados à avaliação de propostas de melhoria e realização e de pilotos (SILVA FILHO, 2006).

---

<sup>4</sup> Ferramenta proposta por SILVA FILHO (2006) para apoiar a realização de pilotos.

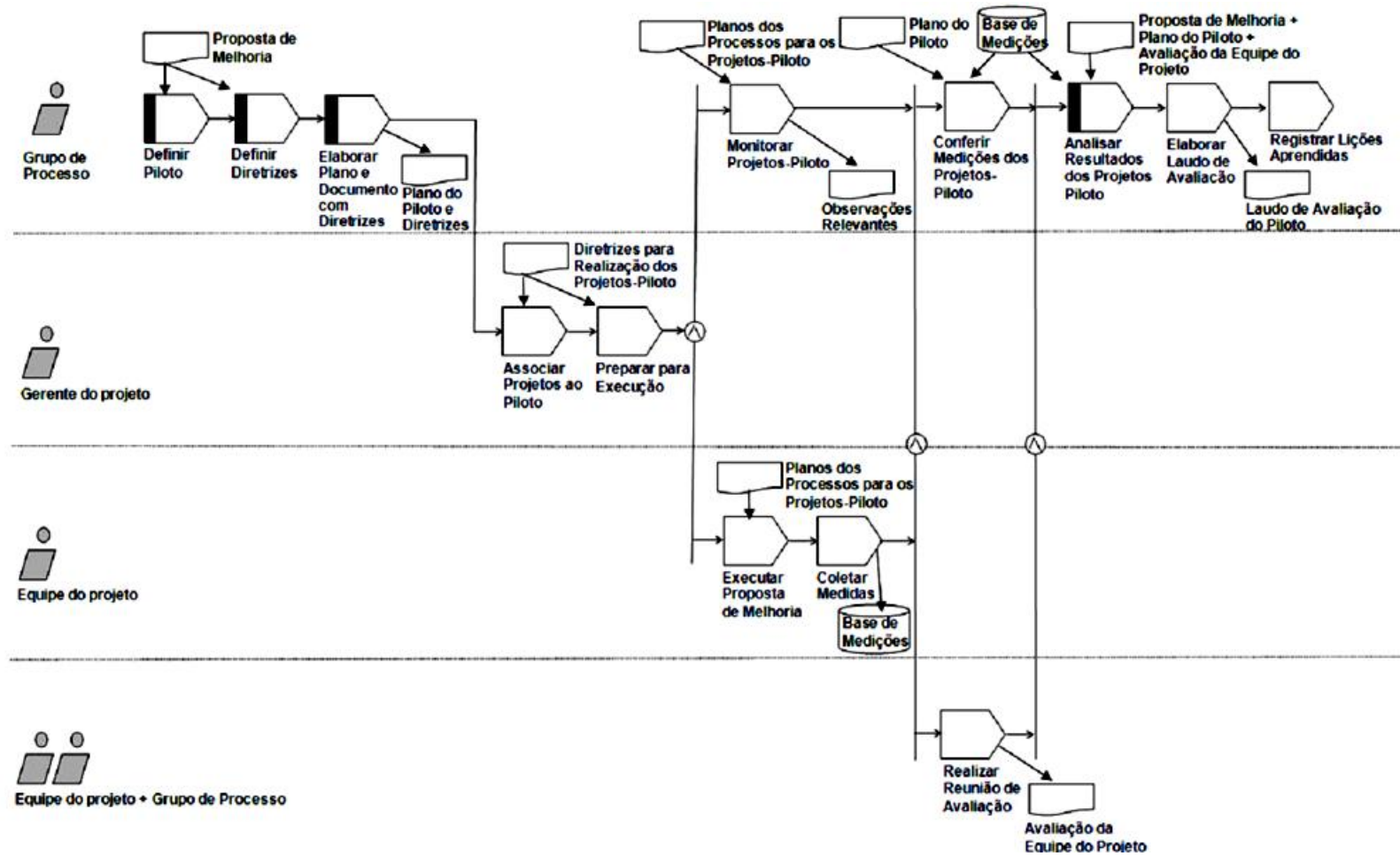


Figura 2.5 - Modelo do Processo para Realização de Pilotos (SILVA FILHO, 2006)

## 2.7 Considerações Finais

Este capítulo apresentou uma revisão da literatura sobre melhoria contínua em processos de software, identificando abordagens que, de alguma forma, apoiassem o ciclo de melhoria contínua. Os conceitos sobre processos de software foram apresentados, a fim de contextualizar as abordagens, problemas e lacunas de pesquisa existentes na melhoria contínua de processos para o desenvolvimento de software.

Focado inicialmente na melhoria contínua de processos de software sob controle estatístico, a pesquisa envolveu um estudo baseado em revisão sistemática que possibilitou identificar abordagens, ferramentas e técnicas utilizadas em organizações de alta maturidade, tais com o CEP, Seis Sigma, Análise de Causa, dentre outros.

Dentre as iniciativas de melhoria de processo de software mais relevantes identificadas, as normas internacionais e modelos de maturidade de processos se destacam por serem de ampla adoção por partes das organizações e empresas, e por serem insumos para práticas características de organizações que buscam a melhoria contínua de seus processos.

O Controle Estatístico de Processos é citado pelos modelos de maturidade como método para apoiar a melhoria contínua através do controle quantitativo e estatístico dos processos. Foram apresentados um breve histórico, conceitos e aplicação do CEP na indústria de software e suas brechas de melhoria. Neste mesmo contexto, a metodologia Seis Sigma, recentemente aplicada à melhoria de processo de software, foi descrita como técnica para alcançar a melhoria contínua de processo em diversos níveis de maturidade por meio de frameworks e ferramentas de análise.

Um aparato sucinto das abordagens identificadas no estudo baseado em revisão sistemática foi descrito, discutindo sua contribuição para a melhoria contínua de processo de software e seus pontos fracos.

Percebe-se que a maioria dos trabalhos implementavam ciclos de melhoria contínua, considerando, na sua maioria, causas especiais de variação de processo como sinal a ser analisado, a fim de melhorar a capacidade do processo. KALINOWSKI *et al.* (2008), no entanto, considera padrões sistemáticos de erros e correlações entre as causas envolvidas em defeitos, colaborando para que causas comuns em processos de software sob controle estatístico que ocasionem defeitos no software sejam

identificadas. A abordagem DBPI (KALINOWSKI *et al.*, 2008) foca na melhoria do processo de software através da prevenção de defeitos e se destaca por considerar sistemicamente as correlações bayesianas entre as causas envolvidas.

Processos maduros – alvo de melhoria contínua – são mais complexos e, portanto, mais difíceis de serem melhorados (DETTMER, 1998b). A complexidade e diversidade de fatores de influência dos processos de software fomentou a necessidade de uma revisão informal da literatura para identificar outras abordagens que considerassem tais características do processo de software. A Teoria das Restrições, metodologia de melhoria desenvolvida inicialmente na manufatura, foi identificada por fazer parte de abordagens que compõem a Gestão da Qualidade Total, tal como, a abordagem de melhoria contínua de Deming (PDCA) e o Seis Sigma. A Teoria das Restrições objetiva melhorar continuamente processos de maneira estruturada e sistemática, tendo aplicações bem sucedidas em diversas áreas de produção e serviço, inclusive em áreas de Tecnologia da Informação.

A despeito da maturidade da organização ou do processo em análise, a Teoria das Restrições considera os fatores de influência que causam variação no comportamento dos processos, a fim de propor soluções, desenvolver propostas de melhoria e implementá-las.

O capítulo seguinte apresenta os conceitos, ferramentas e aplicabilidade da Teoria das Restrições e dos seus Processos de Pensamento que possibilitam justificar a sua escolha, descrevendo os conceitos utilizados na abordagem proposta nesta dissertação.

## Capítulo 3 – Teoria das Restrições

*Este capítulo apresenta a Teoria das Restrições (Theory of Constraints), seus conceitos básicos, abordagem e métodos. Apresenta também a evolução da teoria e sua adoção nas diversas áreas, justificando a sua escolha como metodologia com indícios de adequabilidade ao contexto desta dissertação.*

### 3.1 Introdução

Com o aumento da competitividade e consequente amadurecimento das organizações e seus processos, torna-se cada vez mais difícil implementar inovações ou otimizações de processo. Tais otimizações (ou melhorias incrementais) estão se tornando mais árduas de realizar em razão da crescente complexidade da forma como as redes de estruturas e processos estão estabelecidas e interconectadas nas organizações (PFEIFER *et al.*, 2003).

Alguns fatores existentes em sistemas complexos influenciam a adoção de mudanças bem sucedidas, por exemplo: natureza dinâmica das mudanças que ocorrem no ambiente; variação inerente resultante da interação dos elementos dos processos; tendência de tornar atividades e soluções mais departamentais nos contextos organizacionais, ao invés de integrá-los funcionalmente; e, conseqüentemente, melhoria localmente em detrimento do melhor funcionamento do sistema como um todo (DETTMER, 1998b).

Desta forma, PFEIFER *et al.* (2003) argumentam que a otimização de sistemas maduros somente pode ser alcançada por meio de análises sistêmicas integradas e esforços de melhorias holísticas. W. Edwards Deming corrobora argumentando que melhoria de qualidade não é possível sem um conhecimento profundo do sistema e aponta quatro pilares fundamentais (DEMING, 1992): teoria do conhecimento do sistema; compreensão da variação dos processos (teoria de Shewhart); entendimento dos fatores psicológicos envolvidos e conhecimento apreciativo do sistema.

Neste contexto, é importante que o processo otimizador observe todos os componentes relevantes envolvidos no sistema para que juntos possam contribuir para



a melhoria do sistema por completo, e não apenas para melhorias individualizadas dos componentes (DETTMER, 1998a).

A Teoria das Restrições (TDR), proposta por Eliyahu M. Goldratt na década de 80 (GOLDRATT e COX, 1984), é constituída de uma abordagem de melhoria contínua que preconiza que todo sistema ou processo possui restrições (normalmente um número reduzido de restrições) responsáveis pela limitação do seu desempenho e que, se removidas de maneira sistemática, podem alcançar resultados mais relevantes e positivos em relação ao seu desempenho (SCHEINKOPF, 1999).

A Teoria de GOLDRATT é composta por uma coleção de princípios de sistema e ferramentas para solução de problema de melhoria de desempenho de sistema (DETTMER, 1997). Tais ferramentas compõem os Processos de Raciocínio (PR) e propõem um conjunto de fundamentos lógicos e intuitivos para sistematicamente identificar problemas e propor soluções. Os conceitos da Teoria das Restrições são descritos neste capítulo, assim como os Processos de Raciocínio, com o intuito de contextualizar a escolha da TDR como metodologia aplicável para apoiar a melhoria contínua de processos de software.

Este capítulo contém, além desta seção introdutória, a Seção 3.2, descrevendo a Teoria das Restrições, sua abordagem, fundamentos e elementos principais. Na Seção 3.3, o processo Cinco Passos é explicado. Os Processos de Raciocínio são descritos na Seção 3.4. Na Seção 3.5 são descritos alguns trabalhos envolvendo a Teoria das Restrições e outras abordagens de melhoria contínua. Na Seção 3.7, as considerações finais deste capítulo.

## **3.2 Teoria das Restrições**

A Teoria das Restrições é uma metodologia para apoiar agentes de mudanças dentro das organizações na identificação e compreensão de seus problemas e restrições, provendo ferramentas para a elaboração de soluções inovadoras e de implementação das soluções de maneira sistemática e bem sucedida.

Uma característica que diferencia a perspectiva de melhoria de GOLDRATT é o reconhecimento de que sempre existirá limitação para o desempenho do sistema de interesse e que, a despeito da complexidade da teia de relacionamentos e dos fatores de influência do sistema ou processo, esta limitação é causada por um número muito

pequeno – normalmente um – de fatores de influência do sistema ou do processo (MABIN e BALDERSTONE, 2003). GOLDRATT (1990) ilustra essa limitação por meio de uma corrente. Ele afirma que o desempenho de um sistema ou processo na organização é limitado pelo seu elo mais fraco. Desta forma, se uma organização almeja melhorar seu desempenho, o primeiro passo a ser realizado é identificar o elo mais fraco (ou restrição principal) do sistema ou de seus processos.

Um crescimento considerável de publicações, envolvendo a Teoria das Restrições, tem sido observado, mostrando um aumento de interesse, tanto acadêmico como prático, pelo assunto. MABIN e BALDERSTONE (2003) reúnem resultados obtidos com a aplicação da TDR com base em um mapeamento sistemático de publicações acadêmicas que relatam experiências de implantação da Teoria das Restrições na indústria, destacando os seguintes resultados: (i) redução na entrega e processamento de pedidos (*lead time*); (ii) redução de tempo do ciclo de produção; (iii) redução de inventário e (iv) melhoria de cumprimento de prazos.

CHOE *et al.* (2004) e MABIN (2001) relatam aplicações da Teoria da Restrições para gerenciar mudança organizacional, destacando o apoio prático e sistemático que os Processos de Raciocínio provêm para situações de avaliação, descoberta de pressupostos, resolução de conflitos, planejamento e implementação bem sucedida de mudanças.

Dentre as áreas de conhecimento, RAHMAN (1998) mapeou inicialmente aplicações em engenharia de computadores, engenharia industrial, gestão industrial, engenharia de custos e economia de produção. Posteriormente, MABIN e BALDERSTONE (2003) relataram um aumento da aplicação em diversos setores, inclusive nas indústrias automobilística, de aeronaves, eletrônica, de semi-condutores, serviços, software, saúde, militar e educação; resolvendo problemas dos mais variados tipos, tais como contabilidade, agendamento, desempenho de medidas, qualidade, gerenciamento de mudança organizacional e gerência de projeto.

A utilização dos Processos de Raciocínio, juntamente com a estratégia de melhoria contínua proposta por GOLDRATT – Ciclo de Melhoria Cinco Passos (SCHEINKOPF, 1999) –, colaborou para uma aplicação mais ampla da Teoria das Restrições, pois focam nos conceitos e comportamentos que deveriam direcionar as ações de melhorias para sistemas de qualquer natureza de maneira sistemática e contínua.

Muitos pesquisadores têm considerado similaridades e diferenças entre a Gestão da Qualidade Total (TQM) e a Teoria das Restrições (DETTMER, 1995b; GUPTA e BOYD, 2008). Alguns desses estudos procuram demonstrar que TQM e TDR não são mutuamente exclusivos, sendo possível conceber a Teoria das Restrições como uma abordagem construída com base nas fundações estabelecidas pela Gestão da Qualidade Total (DETTMER, 1997; LEPORE e COHEN, 1999; BOYD e GUPTA, 2004). Ademais, GUPTA *et al.* (2008) argumentam que a Teoria das Restrições possibilita, por meio dos seus Processos de Raciocínio e conceitos, concentrar os esforços de melhoria da qualidade onde se maximizem os seus impactos.

A Teoria das Restrições é organizada da seguinte forma: (i) melhoria de desempenho utilizando gerenciamento de restrições através do Ciclo de Melhoria Cinco Passos; (ii) análise e resolução de restrições utilizando os Processos de Raciocínio.

O conceito de restrição e suas classificações são importantes para o entendimento da aplicação da Teoria das Restrições, assim como Ciclo de Melhoria Cinco Passos e as construções lógicas que fundamentam os Processos de Raciocínio. Estes conceitos e construções são descritos na subseção seguinte.

### **3.2.1 Classificação das Restrições**

Para SCHEINKOPF (1999), uma restrição é definida como qualquer fator ou efeito que limite o alcance do desempenho de um sistema ou processo em relação ao seu propósito.

As restrições são classificadas em três categorias principais: físicas, políticas e comportamentais. Segundo SCHEINKOPF (1999), existem restrições de todas as três categorias em qualquer sistema ou processo, a qualquer momento, e estas estão relacionadas. Restrições comportamentais são resultados de paradigmas e comportamentos que resultam em restrições políticas que, por sua vez, causam restrições físicas (ingerência ou má alocação).

As restrições físicas são os recursos que limitam fisicamente o desempenho do sistema ou processo. Tal limitação pode ser gerada por restrições físicas relacionadas aos insumos (insuficientes ou inadequados), clientes (clientes interno/processo subsequente ou mercado) ou capacidade interna (máquinas, pessoas ou procedimentos não capazes ou não apropriados).

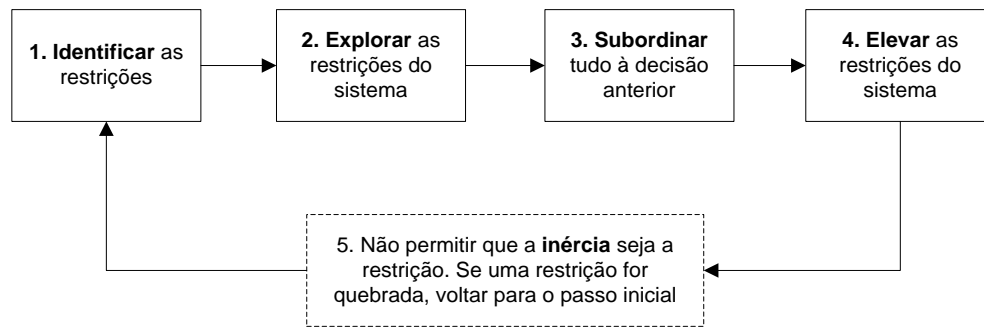
Políticas são regras ou normas que governam a forma como as organizações fazem seus negócios. A partir das políticas, são definidos quando, onde e de que forma são gerenciadas as restrições físicas. Restrições políticas são todas as políticas que impedem a habilidade do sistema de continuar melhorando.

Restrições comportamentais são todas as crenças, paradigmas e pressupostos que determinam como organizações desenvolvem, adotam e seguem restrições políticas. Normalmente, restrições comportamentais existem porque fazem parte da cultura da organização ou dos indivíduos que definiram as políticas da organização.

### **3.3 Ciclo de Melhoria Cinco Passos**

O ciclo de melhoria contínua proposto por GOLDRATT tem o propósito de sistematicamente identificar e explorar sucessivamente as restrições. GUPTA e BOYD (2008) citam que o ciclo de melhoria contínua de GOLDRATT e o ciclo PDCA de DEMING são similares, com uma diferença principal: o Ciclo de Melhoria Cinco Passos reconhece explicitamente que existe um número pequeno de restrições que limita o desempenho do processo. Uma representação do ciclo de melhoria é mostrada na Figura 3.1.

No primeiro passo, deve-se identificar qual restrição física ou política está limitando a produtividade do sistema. No segundo passo, a restrição é analisada a fim de se propor uma melhoria para otimizar o desempenho do sistema ou do processo, a partir da disponibilidade atual dos recursos (obter o máximo do sistema atual sem precisar aumentar o custo de produção). No terceiro passo, deve-se alinhar os esforços dos elementos envolvidos (por exemplo: processo, pessoas, tempo operacional de espera da restrição) com as soluções proposta no passo anterior. Este é o passo mais difícil, pois envolve investigar de que forma a organização, o sistema ou o processo pode contribuir para superar a restrição identificada sem modificar a capacidade produtiva.



**Figura 3.1** – Ciclo de Melhoria Cinco Passos

No quarto passo, a capacidade produtiva do sistema ou do processo é elevada através de, por exemplo, alocação de mais funcionários, recursos para melhorar qualidade ou novos equipamentos. Neste momento, restrições políticas e comportamentais são candidatas a serem removidas, implicando em mudanças profundas, tais como aumento de orçamento ou mudanças estruturais na organização.

SCHEINKOPF (1999) argumenta que a exploração de restrições políticas e comportamentais é importante, pois sem considerá-los a melhoria na capacidade do sistema ou do processo pode ser limitada.

O último passo possibilita o ciclo de melhoria, não permitindo que a inércia seja a restrição e garantindo que sucessivamente as restrições serão identificadas, exploradas e removidas.

### 3.4 Processos de Raciocínio

O Ciclo de Melhoria Cinco Passos provê uma abordagem simples e efetiva para melhoria contínua. No entanto, restrições políticas e comportamentais são comumente mais difíceis de serem tratadas. Nestes casos, os Processos de Raciocínio (PR) propostos por GOLDRATT são úteis para decidir o que mudar, para o que mudar e como promover a mudança.

Os Processos de Raciocínio focam nos fatores de influência que estão atualmente impedindo o sistema ou o processo de atingir seus objetivos. A partir da identificação de sintomas no sistema ou no processo que evidenciem um desempenho abaixo do esperado, os Processos de Raciocínio podem ser aplicados para identificar o que está causando os tais sintomas, o que precisa ser feito para corrigir ou remover suas causas e como as ações de correção devem ser implementadas (MABIN, 1999).

De acordo com MABIN e SCHEINKOPF (MABIN, 1999; SCHEINKOPF, 1999), um ponto forte da Teoria das Restrições e seus Processos de Raciocínio é o mapeamento do sistema sob o ponto de vista do problema a ser resolvido. Desta forma, apenas aspectos considerados pertinentes ao desempenho indesejado do processo ou do sistema são mapeados, permitindo que problemas complexos sejam resolvidos sem recorrer ao modelo completo do sistema ou do processo.

Um conjunto composto por cinco diagramas (árvores) e regras lógicas foi proposto inicialmente por GOLDRATT (1992), com o intuito de apoiar os agentes de mudança no processo de melhoria.

Os diagramas são resultado da aplicação dos Processos de Raciocínio e são compostos por quatro árvores e um diagrama em formato de nuvem, a saber: (i) Árvore da Realidade Atual (ARA); (ii) Diagrama de Resolução de Conflito (DRC); (iii) Árvore da Realidade Futura (ARF); (iv) Árvore de Pré-Requisitos (APR) e (v) Árvore de Transição (ATR). A correspondência da aplicação dos Processos de Raciocínio para responder às questões de mudança é ilustrada na Tabela 3.1.

As regras lógicas são subdivididas em dois subconjuntos:

- Relacionamentos: *Causa Suficiente* (CS) e *Condição Necessária* (CN);
- Categorias de Ressalvas Legítimas (CRL): oito regras de verificação que averiguam se os relacionamentos de *Causa Suficiente* dos elementos estão consistentes e se respeitam princípios básicos do método científico.

De acordo com BALDERSTONE e MARBIN (1998), a principal dificuldade a ser superada no processo de mudança é a tendência comportamental de resistência à mudança. As ferramentas Árvore de Realidade Futura e Árvore de Pré-Requisitos se configuram como ideais para identificar obstáculos e derivar objetivos intermediários que permitam que as ações de melhorias foquem nas soluções operacionais e práticas, sem perder o foco do objetivo principal da melhoria.

**Tabela 3.1** - Aplicação dos Processos de Raciocínio em relação às questões de mudança (SCHEINKOPF, 1999)

O que mudar?	Árvore da Realidade Atual
	Diagrama de Resolução de Conflitos
Para o que mudar?	Árvore da Realidade Futura
	Árvore de Pré-requisitos
Como causar a mudança?	Árvore de Transição

Vários trabalhos procuraram melhorar a metodologia dos Processos de Raciocínio através do desenvolvimento de novas ferramentas de PR ou através da integração das ferramentas com outras metodologias (KIM *et al.*, 2008). A evolução das ferramentas dos Processos de Raciocínio não apenas torna a aplicação da Teoria das Restrições mais abrangente, como também tem permitido torná-la mais adequada e flexível a aplicações nas mais diversas áreas (MABIN e BALDERSTONE, 2003).

### 3.4.1 Fundamentos Lógicos

Os Processos de Raciocínio são fundamentados em dois tipos de relacionamento:

- Causa Suficiente: é o padrão lógico de causa-efeito-causa necessário para representar a existência causal das entidades. Quando se identifica que uma entidade (efeito) existe porque uma ou mais entidades existem (causas), está se usando o raciocínio de *Causa Suficiente*.
- Condição Necessária: é o padrão lógico que verifica quais entidades se fazem necessárias (condições necessárias) para que outra entidade exista (objetivo ou estado futuro). Um dos objetivos da regra lógica da *Condição Necessária* é revelar os requisitos necessários ou pressupostos para algo acontecer. Condições necessárias são regras, políticas, comportamentos, leis que proveem limitações ou fronteiras para o universo de possibilidades que se tem para atingir objetivos e metas.

Exemplos e representações de *Causa Suficiente* e *Condição Necessária* podem ser visto na Tabela 3.2.

**Tabela 3.2 – Relacionamentos dos Processos de Raciocínio**

Regra lógica	Exemplo	Representação
<i>Causa Suficiente</i> (efeito-causa-efeito)	<ul style="list-style-type: none"> <li>• A é suficiente para causar B</li> <li>• C e D são suficientes para causar E</li> </ul> <p style="text-align: center;">E</p>	
	X e Y ocorrendo simultaneamente são suficientes para causar P	
<i>Condição Necessária</i> (pré-condição)	Para atingir S é preciso que T exista/aconteça	
	Para atingir O, são necessários que M e N simultaneamente existam/aconteçam	

Os elementos utilizados para construções das árvores e diagrama baseados em *Causa Suficiente* e *Condição Necessária* possuem características e terminologias específicas. A *Árvore da Realidade Atual*, *Árvore da Realidade Futura* e *Árvore de Transição* são construídas com base no relacionamento de *Causa Suficiente*. As partes que compõem estas árvores são:

- **Entidade:** representa um elemento do sistema ou do processo expresso como uma assertiva e pode ser uma *causa* e/ou um *efeito*. É representado por meio de um retângulo com bordas arredondadas. Exemplo: “A taxa de rotatividade é alta (maior que 20%)”.
- **Seta:** indica um relacionamento entre duas entidades. A entidade na origem da seta é a *causa* e o destino é o *efeito*. Os *pressupostos* são os responsáveis pelo significado e existência da relação de causalidade representada pela seta.
- **Causa:** uma entidade (ou grupo de entidades interligadas por um *conector-lógico-e*) que, dada a sua existência, causará outra entidade existir (*efeito*).



- **Conector-lógico-e:** agrupa entidades para representar o “e” lógico por meio de uma elipse tocando as *setas* que participam do “e” lógico.
- **Efeito:** representa uma entidade que existe como resultado de uma *causa*. Também referenciado como *consequência*.
- **Pressuposto:** razão e significado da existência de uma relação de causa-e-efeito (*seta*).
- **Ponto de entrada:** entidade que não possui uma seta (não é efeito de nenhuma entidade) como destino existente na realidade atual. Representado por um retângulo.
- **Injeção:** entidade que não possui uma seta como destino (não é efeito de nenhuma entidade) e não existe na realidade atual. São utilizadas para simular hipoteticamente pontos de entrada. Ocorrem apenas na Árvore de Realidade Futura e na Árvore de Transição. Representado por um retângulo.

Os diagramas restantes, Diagrama de Resolução de Conflitos e Árvore de Pré-Requisitos, utilizam o relacionamento de *Condição Necessária*. Os elementos que os compõem são listados a seguir:

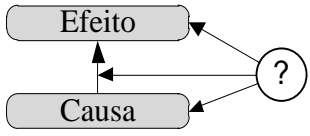
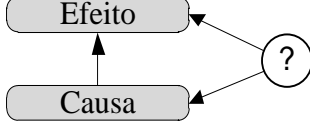
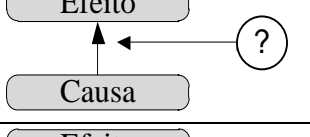
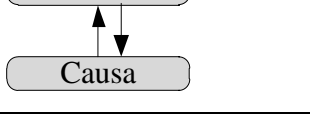
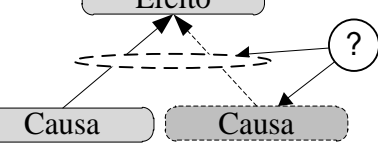
- **Entidade:** representa um único elemento do sistema ou do processo. Diferentemente da *Causa Suficiente*, entidades de *Condição Necessária* não são fundamentalmente assertivas. Termos como *deve*, *pode*, *precisa* e suas variantes negativas podem ser utilizados. É representado por um retângulo com bordas arredondadas.
- **Seta:** indica um relacionamento entre duas entidades. A entidade na origem da seta é a *Condição Necessária* e o destino é o *objetivo*. Toda seta existe com base em um ou mais *pressupostos*.
- **Condição Necessária:** representa uma entidade considerada necessária para que outra entidade (*objetivo*) exista ou que a permita existir.
- **Objetivo:** uma entidade que só pode existir se, pelo menos, outra entidade (*Condição Necessária*) existir.
- **Pressuposto:** representa a razão para a existência do relacionamento de *Condição Necessária* (razão pela qual uma entidade é requerida para que outra exista);

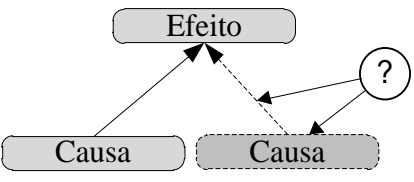
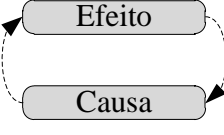
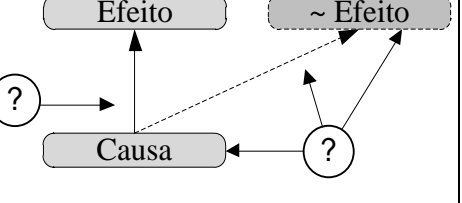
- **Ramo:** representa um agrupamento de entidades relacionadas entre si através de relacionamentos de *Condição Necessária* (uma parte – galho – da árvore).

Os Processos de Raciocínio provêm uma metodologia para testar as concepções da realidade e dos pressupostos por meio de um conjunto de regras de verificação chamado Categorias de Ressalvas Legítimas (CRL). Estas regras agregam conceitos intuitivos do método científico, a fim de prover uma abordagem sistemática para descobrir, verbalizar, questionar e substituir pressupostos existentes (SCHEINKOPF, 1999).

As Categorias de Ressalvas Legítimas são utilizadas para verificar a consistência e completude das inferências e relacionamentos das árvores baseadas no relacionamento de *Causa Suficiente* (ARA, ARF, AT). Um sumário das CRL é apresentado na Tabela 3.3.

**Tabela 3.3** – Categorias de Ressalvas Legítimas (SCHEINKOPF, 1999)

Consistência	Descrição	Representação
1. Esclarecimento ou Claridade	Verificar se a causa ou o efeito descrito na entidade está formulada(o) de forma clara e inequívoca	
2. Existência da entidade	Validar a real existência da entidade (efeito ou causa), verificando se a entidade é verossímil	
3. Existência de casualidade	Consistir a presença do elo causal entre o efeito e a causa. Padrão: <i>Se [causa] Então [efeito]</i>	
4. Tautologia	Evitar redundância na relação causa-e-efeito	
5. Suficiência (insuficiência) de causa	Verificar a existência de uma ou mais entidades que precisam coexistir com a causa original, a fim de tornar a causalidade válida. Padrão: <i>Se [causa] E [causa] Então [efeito]</i>	

Consistência	Descrição	Representação
6. Causa adicional	Verificar a existência de outra causa que também resulte no efeito. Padrão: <i>Se [causa] OU [causa] Então [efeito]</i>	
7. Inversão de causa e efeito	Verificar se a direção da seta (relacionamento de causa-e-efeito) não está invertida.	
8. Existência de efeito previsto (redução ao absurdo)	Verificar se a partir da causa original, um novo efeito (que invalide a relação de causa-e-efeito original) é relacionado	

### 3.4.2 Processos de Raciocínio

Os Processos de Raciocínio são baseados nos fundamentos lógicos descritos na seção anterior. Com base nisto, SCHEINKOPF (1999) argumenta que os Processos de Raciocínio não se limitam apenas aos descritos inicialmente por GODRATT, podendo sofrer adaptação e evolução, como descritos nos trabalhos de (DETTMER, 1997; LEPORE e COHEN, 1999; DAVIES *et al.*, 2004).

Nas subseções seguintes, propósitos, conceitos e peculiaridades de cada Processo de Raciocínio são apresentados.

#### 3.4.2.1 Árvore da Realidade Atual

Para GOLDRATT (1990), realidade está relacionada a uma condição existente, sendo representada por *entidades* relacionadas através de relacionamentos de causa-e-efeito (*setas*). A Árvore da Realidade Atual foi definida com o intuito de representar e analisar uma situação existente indesejada de um sistema, processo ou realidade (MABIN, 1999), a fim de identificar a causa-raiz (restrição principal) da situação indesejada. KIM *et al.* (2008) mencionam que esta ferramenta é particularmente efetiva se a restrição identificada for política ou comportamental.

DETTMER (1997) cita alguns resultados alcançados por meio da ARA, a saber:

- Entendimento de sistemas ou processos complexos;

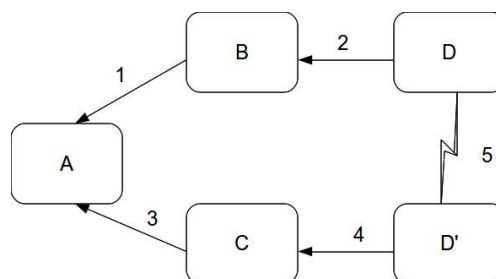
- Identificação de *Efeitos Indesejáveis* (EI) no sistema;
- Mapeamento dos EI por meio de uma cadeia lógica de causa-e-efeito a suas causas-raiz;
- Identificação de um problema principal responsável por 70% ou mais das EI (regra de Pareto);
- Isolamento de um conjunto reduzido de fatores (restrições principais) responsáveis por uma limitação expressiva do sistema ou do processo

### 3.4.2.2 Diagrama de Resolução de Conflitos

O método de Diagrama de Resolução de Conflitos, também conhecido como Dispersão de Nuvem (*Evaporating Cloud*), auxilia na resolução de situações de dilemas e conflitos (GOLDRATT, 1990). É o Processo de Raciocínio mais utilizado, por ser simples e de rápida execução (SCHEINKOPF, 1999). Segundo KIM *et al.* (2008), há diferentes formas de resolver o conflito ou dilema através de um DRC. Basicamente, propostas de solução (*injeções*) podem ser geradas através do exame direto do conflito ou metodologicamente, evidenciando pressupostos limitantes e possíveis meios de invalidá-los ou removê-los.

Segundo MABIN (1999), o DRC objetiva confirmar que o conflito existe, resolver o conflito, evitar o comprometimento da solução, criar solução ganha-ganha e explicar de forma profunda a razão da existência do problema.

O diagrama é composto por cinco entidades, como pode ser visto na Figura 3.2. As *setas* de número 1 e 3 identificam respectivamente a *entidade* B e C como uma *Condição Necessária* para a entidade A. Similarmente, a *seta* de número 2 identifica D como uma *Condição Necessária* para B e a *seta* de número 4 identifica D' como uma *Condição Necessária* para C. A *seta* de número 5 identifica o conflito que há entre as entidades D e D'.



**Figura 3.2** - Modelo de um Diagrama de Resolução de Conflitos

### 3.4.2.3 *Árvore da Realidade Futura*

A Árvore de Realidade Futura, como sugerido pelo seu nome, é uma representação de um cenário futuro sobre o qual se deseja visualizar uma solução proposta. Cautelosamente a árvore é construída com o intuito de considerar as mudanças necessárias que envolvem a solução, os obstáculos a serem superados e o impacto relacionado a estas ações.

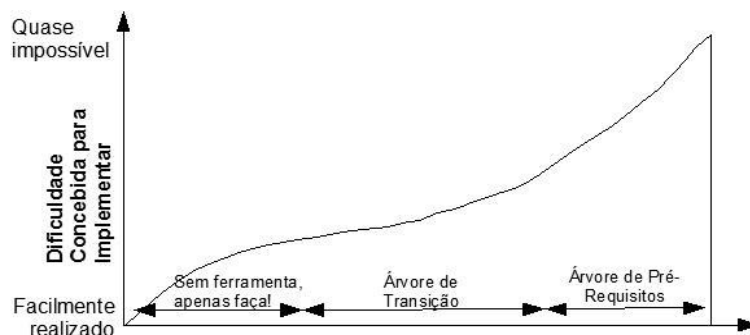
Dentre os propósitos da ARF, pode-se citar (MABIN, 1999):

- Avaliar de maneira efetiva as novas ideias antes de alocar recursos para sua implementação;
- Determinar se a proposta de solução produzirá os efeitos desejados sem gerar efeitos colaterais desconhecidos;
- Revelar onde e quando a solução proposta criará novos problemas ou efeitos colaterais e quais ações adicionais são necessárias para prevenir a ocorrência de efeitos colaterais indesejáveis;
- Prover meios de tornar os efeitos desejáveis auto-sustentáveis, prevendo ciclos de manutenção na proposta de solução que garantam a sua efetividade no sistema ou processo;
- Garantir aplicabilidade e viabilidade da proposta de solução;
- Prover uma ferramenta efetiva para persuadir a alta gerência a tomar determinada decisão;
- Servir como um esboço inicial de planejamento.

### 3.4.2.4 *Árvore de Pré-Requisitos*

De acordo com GOLDRATT (1990), ideias não são soluções. Para ser considerada como tal, a ideia deveria ser implementada por completo e o sistema ou processo deveria estar trabalhando como o esperado. Diferentemente das outras árvores, a Árvore de Pré-Requisitos utiliza o relacionamento de *Condição Necessária*.

SCHEINKOPFT (1999) argumenta que a dificuldade associada a uma solução proposta pode variar entre ideias que já são praticamente ações (que não necessitam de nenhuma ferramenta para implementá-la) e ideias cujos objetivos não serão fáceis de serem implementados. A escolha da ferramenta dos Processos de Raciocínio apropriada para implementação, dependendo da dificuldade envolvida, é ilustrada na Figura 3.3.



**Figura 3.3** - Ferramenta apropriada dos Processos de Raciocínio de acordo com a dificuldade de implementação (SCHEINKOPF, 1999)

DETTMER (1997) corrobora, recomendando o uso da APR quando o objetivo é identificar: i) obstáculos que impeçam a realização de alguma ação desejada; ii) paliativos ou condições necessárias para superar ou neutralizar obstáculos e (iii) sequência requerida de ações para realizar determinado plano ou atingir algum objetivo.

#### 3.4.2.5 *Árvore de Transição*

A *Árvore de Transição* é utilizada para determinar as ações necessárias para implementar a proposta de solução. É referenciada como ferramenta operacional ou tática por investigar os detalhes da implementação e construir o plano de ação.

DETTMER (1997) cita os seguintes propósitos da *Árvore de Transição*:

- Prover um método “passo-a-passo” para implementar soluções propostas;
- Permitir uma visão encadeada do processo de mudança;
- Detectar desvio de progresso em relação a um objetivo definido;
- Adaptar ou redirecionar esforços;
- Comunicar as razões das ações para terceiros;
- Realizar a execução das ações de mudança necessárias para modificar a realidade atual (*injeções*) concebidas na *Árvore de Realidade Futura* ou no Diagrama de Resolução de Conflitos;
- Garantir que os objetivos intermediários identificados na *Árvore de Pré-Requisitos* sejam alcançados (marcos no plano);
- Prevenir que não surjam efeitos indesejáveis a partir da implementação.

### **3.5 Melhoria Contínua com os Processos de Raciocínio**

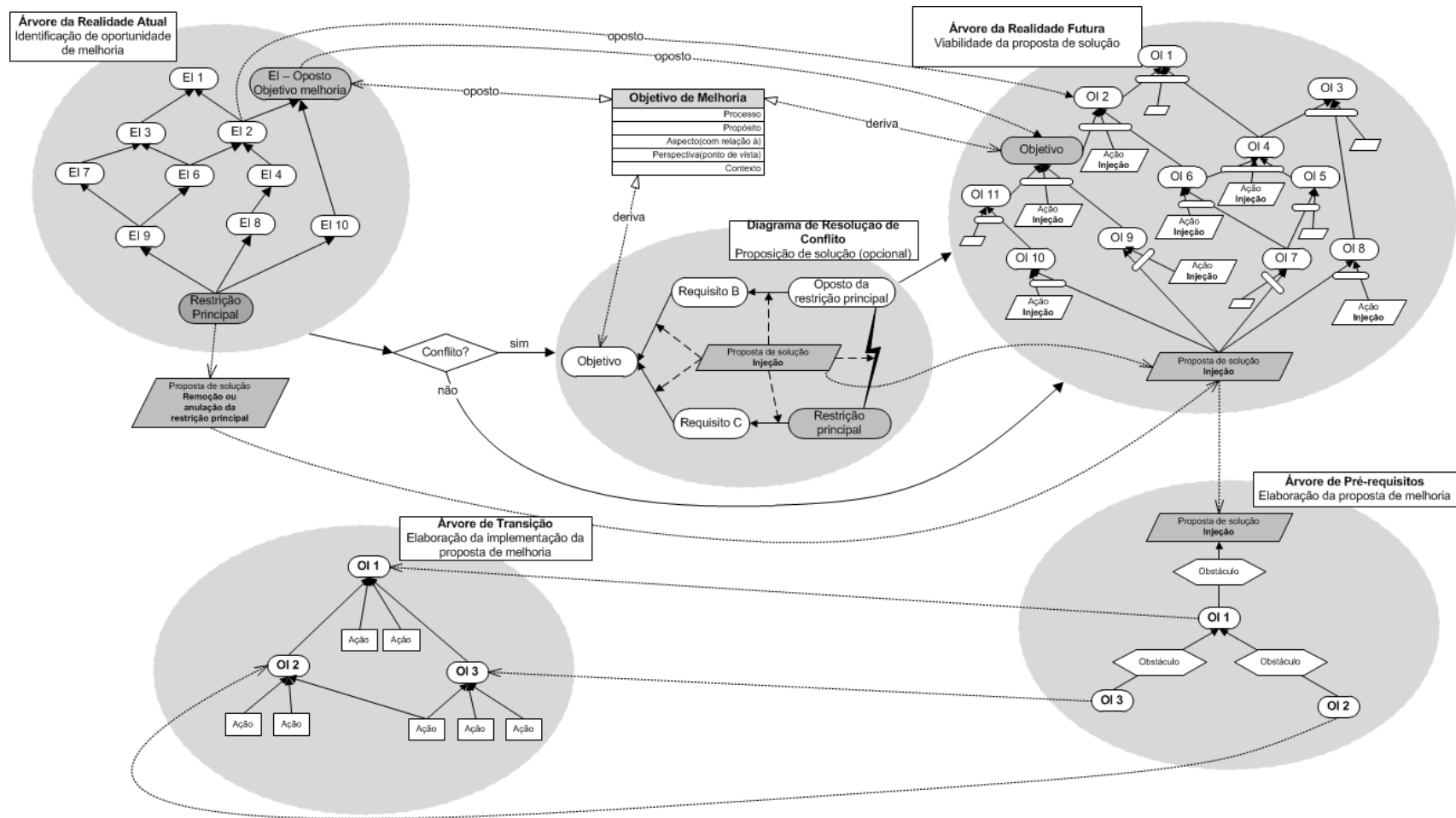
Os Processos de Raciocínio têm aplicações distintas e podem ser utilizados separadamente. No entanto, SCHEINKOPFT (1999) sugere que a utilização em conjunto dos Processos de Raciocínio, a fim de responder as três perguntas apresentadas na Tabela 3.1, potencializa as ferramentas, pois elas se complementam.

A Figura 3.4 apresenta um esquema dos Processos de Raciocínio integrados. No primeiro momento, a ARA é utilizada para representar o cenário atual de determinado ponto de vista, identificando a restrição que mais limita o desempenho do processo (fator de influência que mais causa efeitos indesejáveis). Se a remoção da restrição causar um conflito – causar detrimento para outro passo do processo ou para processos relacionados – o DRC é utilizado para propor soluções que dissipem o conflito. A partir da proposta de solução, que pode ter sido originada da proposta de remoção da restrição principal ou da proposta de resolução de conflito, a ARF verifica a viabilidade da proposta de solução, permitindo construir o cenário futuro onde a melhoria já estaria implantada. Se verificada a viabilidade da solução, a APR é utilizada para dividir o objetivo da melhoria em objetivos intermediários a fim de viabilizar a implementação. Por último, a ATR pode ser aplicada para detalhar os passos a serem seguidos para implementação e implantação da melhoria.

### **3.6 Trabalhos Relacionados à Melhoria Contínua**

De acordo com BROWN *et al.* (2008), duas correntes de melhoria contínua são comuns: melhoria contínua de produtividade, tais como Pensamento Enxuto (*Lean Thinking*) e Teoria das Restrições; e melhoria contínua da qualidade, incluindo a Gestão da Qualidade Total (TQM) e Seis Sigma (SS), por exemplo.

Pesquisas vêm sendo desenvolvidas, propondo combinações entre diferentes abordagens para melhoria contínua (LEPORE e COHEN, 1999; NAVE, 2002; PFEIFER *et al.*, 2003; STRATTON e MANN, 2003; DAVIES *et al.*, 2004; DAVIES *et al.*, 2005; EHIE e SHEU, 2005; PIRASTEH e FARAH, 2006; WATSON *et al.*, 2007)



**Figura 3.4 - Processos de Raciocínio Integrados**



DAVIES *et al.* (2004) apresentam os diferentes aspectos de estratégias de melhoria contínua por meio de um *framework* de comparação entre a Teoria das Restrições, o Pensamento Enxuto – (LT - *Lean Thinking*) e o Seis Sigma. Neste mesmo contexto, PIRASTEH (2006; 2010) propõe uma metodologia para melhoria de inovação de processo, chamada TLS, agregando os pontos fortes das três filosofias de TQM citadas (TDR, LT, SS). Na área de Tecnologia da Informação (TI), SHELTON (2007) realizou uma análise das metodologias de melhoria de processo, apontando diferenças entre o Seis Sigma, o Pensamento Enxuto e a Teoria das Restrições aplicadas à área de TI e destacando situações de aplicação das filosofias de TQM para o gerenciamento de sistemas de informação.

BROWN *et al.* (2008) também comparam algumas estratégias de melhoria contínua, considerando a Gestão Total da Qualidade, a Teoria das Restrições, o Pensamento Enxuto e o Seis Sigma. Uma comparação entre os objetivos, focos primários, efeitos, aspectos culturais, aspectos de processo, benefícios e pontos fracos podem ser visto na Tabela 3.4, sugerindo critérios para seleção de estratégias mais adequadas em um contexto organizacional específico.

No âmbito nacional, LACERDA *et al.* (2010) apresentam uma aplicação da Teoria das Restrições para a Engenharia de Processo, apontando aspectos complementares entre as duas teorias e os benefícios alcançados, dentre os quais: uma melhoria na visão do fluxo dos processos; uma visão sistemática de identificação de problemas nos processos, e, conseqüentemente, uma redefinição melhor dos processos.

No contexto de melhoria de processo de software, COCKBURN (2007) ilustra o problema comumente enfrentado em um ambiente de desenvolvimento de software, no qual a cadeia de produção (elucidação de requisitos do cliente, especificação de funcionalidades, especificação de interface, desenvolvimento e teste) nem sempre está estruturada da melhor maneira, ocasionando super-alocação ou ociosidade. O autor propõe uma melhoria da cadeia crítica de produção de software, típica da teoria de GOLDRATT, através da Teoria das Restrições.

**Tabela 3.4** - Comparação das estratégias de melhoria contínua (adaptado)  
(BROWN *et al.*, 2008)

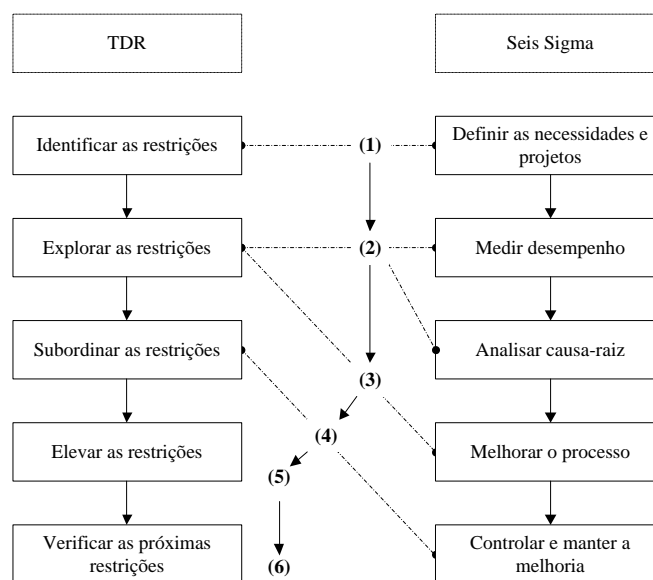
	<b>TQM</b>	<b>Seis Sigma</b>	<b>Pensamento Enxuto</b>	<b>Teorias das Restrições</b>
<b>Objetivo</b>	Melhorar desempenho e qualidade organizacional	Maximizar o sucesso do negócio	Maximizar a produtividade	Melhoria contínua do desempenho

	<b>TQM</b>	<b>Seis Sigma</b>	<b>Pensamento Enxuto</b>	<b>Teorias das Restrições</b>
<b>Foco primário</b>	Cultura de qualidade	Redução sistemática da variação	Minimizar o desperdício	Minimizar as restrições
<b>Efeitos secundários</b>	Satisfação do cliente	Redução do custo	Redução da variação	Menor inventário / gasto
	Redução do custo	Melhoria da produtividade	Melhoria da qualidade	Qualidade melhorada
	Capacitação do empregado	Redução do tempo de ciclo	Redução do tempo de ciclo	Melhoria contínua
	Redução dos gargalos	Mudança cultural	Aumento de flexibilidade	
			Melhoria de produtividade	
<b>Distribuição de conhecimento</b>	Todos os empregados contribuem, visando o conhecimento <i>distribuído</i>	Conhecimento é <i>centralizado</i> com especialistas	Conhecimento é <i>distribuído</i> assim como redução em gasto é responsabilidade dos empregados	Conhecimento é <i>centralizado</i> e focado nas restrições
	Treinamento focado abrangente	Treinamento é muito focado		
<b>Aspectos do processo</b>	Muitas ferramentas disponíveis	<i>Aspectos dominantes</i>	Uso de gerenciamento de fluxogramas do <i>Just-In-Time</i>	<i>Aspectos dominantes</i>
	Ferramentas estatísticas similares ao Seis Sigma	Ferramentas estatísticas específicas	Otimização de processo	Métricas de contabilidades específicas
		Terminologia técnica		Foco sistemático nas restrições
		Estruturas específicas de especialistas		
		Filosofia similar a TQM		
<b>Aspectos da cultura</b>	<i>Aspectos dominantes</i>	Aspectos culturais originados dos aspectos de processos	<i>Aspectos dominantes</i>	Requer uma mudança na abordagem
	Capacitação dos funcionários	Filosofia similar a TQM	Cultura de desperdício mínimo	Estende-se por todas as partes do negócio
	Mudança de filosofia		Ênfase na melhoria contínua	
	Foco nos clientes			
<b>Críticas</b>	Difícil de implementar em pequenas e médias empresas	Cria uma elite de empregados	Não valoriza a análise estatística	Mínimo de intervenção dos empregados
	Processo “fuzzy”		Foco limitado no desperdício	Pode ignorar partes da organização
	Falta de objetivos claros			
<b>Estilo de liderança</b>	Capacitador	Direcionado	Capacitador	Direcionado
<b>Alvos Pressupostos</b>	Toda a organização	Projetos específicos	Todo o desperdício por todo o sistema	Restrições (uma de cada vez)

	TQM	Seis Sigma	Pensamento Enxuto	Teorias das Restrições
		Figuras e números são valorizados	Remoção de desperdício melhora o desempenho do negócio	Velocidade e volume são importantes
		Saída melhora se toda variação for removida		Sistemas existentes são medianos
				Processos são interdependentes

Para STRATTON e MANN (2003) os Processos de Raciocínio da TDR combinados com o processo do TRIZ<sup>5</sup> (A Teoria da Resolução de Problemas Inventivos) podem ajudar sistematicamente na proposição de inovações para problemas. De acordo com os autores, as ferramentas da Teoria das Restrições provêm uma abordagem simples e efetiva para metodologicamente desafiar percepções e pressupostos por trás de conflitos e concepções.

Uma abordagem para melhoria contínua, integrando o Seis Sigma e a Teoria das Restrições, é proposta por EHIE e SHEU (2005) para melhorar o desempenho de sistemas de produção. De acordo com os autores, a integração provê aos gerentes uma plataforma de duas técnicas que trabalham juntas de maneira sinérgica. O *framework* proposto mapeou os passos do ciclo Cinco Passos da TDR com as fases estabelecidas pelo Seis Sigma, como mostrado na Figura 3.5.



**Figura 3.5** - Um *framework* de melhoria contínua integrado, combinando Seis Sigma e Teoria das Restrições (EHIE e SHEU, 2005)

<sup>5</sup> Acrônimo russo para *The Theory of Inventive Problem Solving*, abordagem que codifica, classifica e soluciona metodologicamente problemas de inovação.

### 3.7 Considerações Finais

A Teoria da Restrição ultrapassou os limites de sua aplicação original na manufatura, demonstrando aplicabilidade em diversas áreas. GUPTA e BOYD (2008) analiticamente avaliam a Teoria, justificando a atual credibilidade e esforço de pesquisa que tem sido atribuído à TDR. Os autores analisam a relevância da Teoria identificando critérios que permitam classificar a Teoria das Restrições como promissora, a saber: i) capacidade da Teoria de ser compreendida e aplicada a despeito da área; ii) potencial de avançar em pesquisa de forma significativa por ser fundamentada em conceitos multidisciplinares (produção, *marketing*, economia e contabilidade) e demonstrar aumento de pesquisa em diversas áreas; iii) ter caráter científico, sendo composta por construções válidas e práticas (pressupostos, regras lógicas, medidas e metodologia) que estão relacionadas de maneira logicamente consistente.

Segundo DETTMER (1995b; 1995a), a Gestão TQM concebe o sistema em termos de processos discretos, otimizando a qualidade de cada um separadamente. A Teoria das Restrições, no entanto, propõe melhorar o desempenho do sistema ou do processo como um todo, concentrando-se no seu elo mais fraco.

Como visto neste capítulo, vários pesquisadores apontam similaridades e diferenças entre a TQM e a TDR, explorando tais semelhanças e distinções para propor a integração com a TDR. LEPORE e COHEN (1999) mais especificamente argumentam que a Teoria das Restrições pode ser vista como uma teoria construída nos fundamentos estabelecidos pela Gestão Total da Qualidade. Os autores propõem um processo de dez passos que integra a TDR e a metodologia de melhoria de sistema de Deming (DEMING, 1992).

MABIN *et al.* (2003) argumentam que os conceitos, métodos e ferramentas da Teoria das Restrições têm sido uma fonte de vantagem competitiva para organizações que relataram suas experiências e resultados. Os autores também citam a ocorrência de fatores intangíveis como resultado dos benefícios da TDR na organização, tais como: melhoria na moral da equipe; redução do caos e do estresse; melhor funcionamento da equipe e envolvimento do empregado; melhor “espírito de equipe”; melhoria de flexibilidade e tempo de resposta.

O principal componente da Teoria das Restrições são os Processos de Raciocínio (MABIN, 1999). Eles podem ser definidos como um conjunto de regras lógicas e diagramas (árvores) que proveem um roteiro para melhorar processos e sistemas, guiando o agente de mudança através do processo de decisão através da: (i) estruturação do problema; (ii) identificação do problema; (iii) construção da solução; (iv) identificação de barreiras a serem superadas e (v) implementação da solução;

O objetivo da abordagem apresentada nesta dissertação é a melhoria contínua em processos de software. Vários aspectos da Teoria das Restrições e seus Processos de Raciocínio foram discutidos neste capítulo, apresentando suas principais características e indícios de aplicabilidade à melhoria de processos de software, tais como:

- i) Similaridades e integração com outras metodologias de Gestão Total da Qualidade (TQM) que já foram aplicadas à melhoria de processo de software;
- ii) Regras lógicas para construção dos diagramas que podem mapear os diversos aspectos e fatores de influência inerentes aos processos de software;
- iii) Foco na eliminação de restrições comportamentais e políticas como principais limitantes do desempenho de sistemas e processos. Tais restrições se configuram em causas atribuíveis e comuns do processo e, portanto, são fontes de oportunidades de melhorias no contexto de organizações de software;
- iv) Apoio à institucionalização de mudanças e melhorias no contexto organizacional de maneira sistemática e cíclica (MABIN *et al.*, 2001)

Desta forma, acredita-se que a aplicação dos Processos de Raciocínio tem indícios de adequabilidade à melhoria contínua de software.

No próximo capítulo, a abordagem proposta nesta dissertação é apresentada. Os Processos de Raciocínio são aplicados para a melhoria de processos de software, adaptando-os com base nos conceitos fundamentais da Teoria das Restrições e considerando as características da melhoria contínua de processos de software e demais conceitos e abordagens descritos no Capítulo 2.

# Capítulo 4 – Abordagem Para Melhoria Contínua em Processos de Software Utilizando a Teoria das Restrições

*Este capítulo apresenta a abordagem proposta nesta dissertação para apoiar a melhoria contínua em processos de software utilizando a Teoria das Restrições.*

## 4.1 Introdução

A melhoria contínua em processos é um importante fator de sucesso para organizações de software que almejam produzir software de maneira competitiva. Os principais modelos de maturidade voltados para organizações de software, tais como o CMMI-DEV (SEI, 2010) e o MR-MPS (SOFTEX, 2011), preconizam um ciclo de melhoria dos processos. Neste ciclo, são consideradas oportunidades de melhoria advindas de diversas fontes. Nos níveis iniciais de maturidade destes modelos, as oportunidades de melhoria têm um enfoque de resolução de problemas. O processo ainda não tem seu desempenho previsível e, portanto, possui causas atribuíveis que podem ser removidas, a fim de melhorar sua capacidade. Para os níveis mais altos de maturidade, o ciclo de melhoria é contínuo, considerando métodos quantitativos e o controle estatístico de processo. A melhoria de processo nestes níveis objetiva apoiar a organização em atingir aos seus objetivos estratégicos e de negócio.

Com base no estudo baseado em revisão sistemática da literatura descrito no Capítulo 2 desta dissertação, observaram-se diversos trabalhos que propõem melhorar as abordagens para melhoria de processo de software utilizando técnicas estatísticas. Alguns trabalhos objetivam uma identificação mais acurada das causas atribuíveis de variação do processo, para apoiar a melhoria contínua de tais processos através de remoção de causas-raiz de problemas identificados. Ademais, outros trabalhos argumentam que abordagens e ferramentas já existentes, como, por exemplo, o Seis Sigma, Pensamento Enxuto, *framework* PST/TSP e ferramentas estatísticas, podem ser aplicadas para apoiar a melhoria contínua de processos de software.

As abordagens identificadas propõem um arcabouço para melhoria de processo, sem, no entanto, sistematizar os passos necessários para identificar causas-raiz de problemas ou oportunidades de melhoria que restringem o processo, elaborar propostas de solução para remoção das restrições e implementação da melhoria. Para os processos estáveis, a remoção dos fatores de influência que resultam das causas comuns não é trivial, pois a simples remoção ou alteração de tais fatores de influência pode desencadear efeitos indesejáveis no desempenho do processo ou ocasionar efeitos mínimos que não justifiquem o investimento e esforço empregados na melhoria incremental proposta. Esta oportunidade de pesquisa se configura, provavelmente, pela aplicação relativamente recente de técnicas estatísticas para controle e melhoria de processos de software, assim como pelas características diferenciadas atribuídas aos processos de software.

A investigação de técnicas que utilizassem os conceitos do Controle Estatístico de Processo foi oportuna, pois identificou outras abordagens e técnicas que também estão orientadas à Gestão da Qualidade Total, tais como Seis Sigma, Teoria das Restrições, Pensamento Enxuto e TRIZ.

A partir de uma revisão informal da literatura, descrita no Capítulo 3, foram identificados os Processos de Raciocínio da Teoria das Restrições como abordagem capaz de responder às três questões a seguir: (i) O que mudar? (Principal fator de influência que restringe o desempenho do processo) (ii) Para o que mudar? (Remover o fator de influência que mais restringe o desempenho do processo) (iii) Como promover a mudança? (Quais passos devem ser tomados para aplicar a melhoria proposta e quais ações estão associadas para promover a melhoria com menor possibilidade de causar efeitos colaterais indesejáveis)

Os Processos de Raciocínio proveem ferramentas que permitem identificar de maneira sistemática e com base em regras lógicas as relações de causa-e-efeito que os fatores de influência do processo têm. Desta maneira, características do processo de software, tais como diversidade de fatores que influenciam o seu desempenho (LINDVALL e RUS, 2000) e complexidade no esforço, preponderantemente humano, de criação (CARD, 2004; BALDASSARRE *et al.*, 2007), são consideradas. Além disso, observou-se que a Teoria das Restrições apresenta crescente aumento de relatos de aplicação em diversas áreas, impulsionado principalmente pela aplicabilidade dos

Processos de Raciocínio que orientam os responsáveis pela melhoria a pensar intuitiva e logicamente.

Neste contexto, a aplicação dos Processos de Raciocínio pode apoiar na identificação das causas atribuíveis e das causas comuns de variação do desempenho do processo, visando à remoção do fator de influência que mais restrinja o desempenho do processo. Além disso, os Processos de Raciocínio possibilitam uma elaboração de proposta de solução para as restrições identificadas de forma a maximizar as possibilidades de melhoria e implementação da solução, minimizando os riscos e efeitos colaterais associados à melhoria.

A abordagem proposta apoia a melhoria contínua, mantendo uma Base de Melhoria de Processo (BMP) que pode ser utilizada em ciclos seguintes de melhoria. A reutilização do mapeamento dos relacionamentos de causa-e-efeito possibilita o refinamento do diagrama causal<sup>6</sup> e, conseqüentemente, da identificação de restrições não mapeadas anteriormente.

O processo da abordagem proposta se baseia em abordagens de melhoria de processo de software identificadas na revisão da literatura, mais especificamente em trabalhos desenvolvidos no mesmo grupo de pesquisa (ALBUQUERQUE, 2008; CAMPOS, 2008a). Além da abordagem, formulários foram desenvolvidos para a execução do processo, com o intuito de gerar ativos de processo e raciocínios de decisões que possam ser reutilizados posteriormente para manter o ciclo de melhoria contínua dos processos da organização.

Um processo, apresentado na Figura 4.1, foi definido para a abordagem proposta, sendo composto por seis etapas: (i) Definir objetivo de melhoria; (ii) Preparar para Análise; (iii) Identificar restrição; (iv) Elaborar proposta de melhoria; (v) Implementar a melhoria candidata; (v) Registrar resultados.

Para apoiar a execução do processo proposto nesta abordagem, a descrição do processo de maneira detalhada é apresentada no Anexo II. Além disso, os conceitos

---

<sup>6</sup> O resultado da execução do Processo de Raciocínio da *Árvore de Realidade Atual* é um diagrama em forma de árvore que representa os fatores de influência e os relacionamentos de causa-e-efeito entre eles. Para facilitar o entendimento da finalidade deste Processo de Raciocínio e por se tratar de uma ferramenta mais tipicamente utilizada em organizações de software, ao invés de árvore, o resultado deste Processo de Raciocínio será chamado de diagrama causal dentro do contexto desta abordagem.



relacionados à Teoria das Restrições estão descritos no Anexo III para apoiar a execução dos procedimentos relacionados aos Processos de Raciocínio.

Com o intuito de tornar o processo desta abordagem adequado à melhoria de processo de software, o processo foi submetido a um estudo de viabilidade como descrito no Capítulo 5. A partir da avaliação, as sugestões e adequações foram observadas, resultando no processo descrito abaixo.

As próximas seções (Seção 4.2 à Seção 4.7) descrevem cada etapa do processo. Na Seção 4.8, considerações finais referentes à abordagem proposta são apresentadas.

## **4.2 Definir objetivo de melhoria**

Para organizações de software que buscam melhoria contínua dos seus processos, o alinhamento entre objetivos de negócio e objetivos de melhoria é um fator crítico (MONTONI, 2010). Os modelos de processo de software, CMMI-DEV (SEI, 2010) e MR-MPS (SOFTEX, 2011), para os níveis mais altos de maturidade, requerem que os objetivos de qualidade e desempenho de processos sejam derivados dos objetivos de negócio relevantes (SEI, 2010; SOFTEX, 2011).

Neste contexto, a melhoria da capacidade dos processos em organizações de alta maturidade deve ser priorizada a partir dos objetivos de qualidade e desempenho de processo, com o intuito de perseguir os objetivos de negócio da organização. Além deste cenário, políticas organizacionais podem definir ações de melhoria prioritárias por razões estratégicas, como, por exemplo, melhoria da capacidade do processo de teste para garantir visibilidade em um projeto-chave com intenções de *marketing*.

A abordagem proposta considera que uma decisão anterior dispare o início da execução do processo de melhoria. Como demonstrado na Figura 4.1, a decisão por melhorar um processo foi tomada e inicia-se, assim, o processo descrito nesta abordagem.

O objetivo de melhoria precisa estar definido e descrito de forma que todos os envolvidos na investigação da oportunidade de melhoria e proposição de solução tenham uma noção clara do objetivo a ser perseguido. As tarefas relacionadas à atividade “Definir o objetivo de melhoria” são descritas nas subseções seguintes.

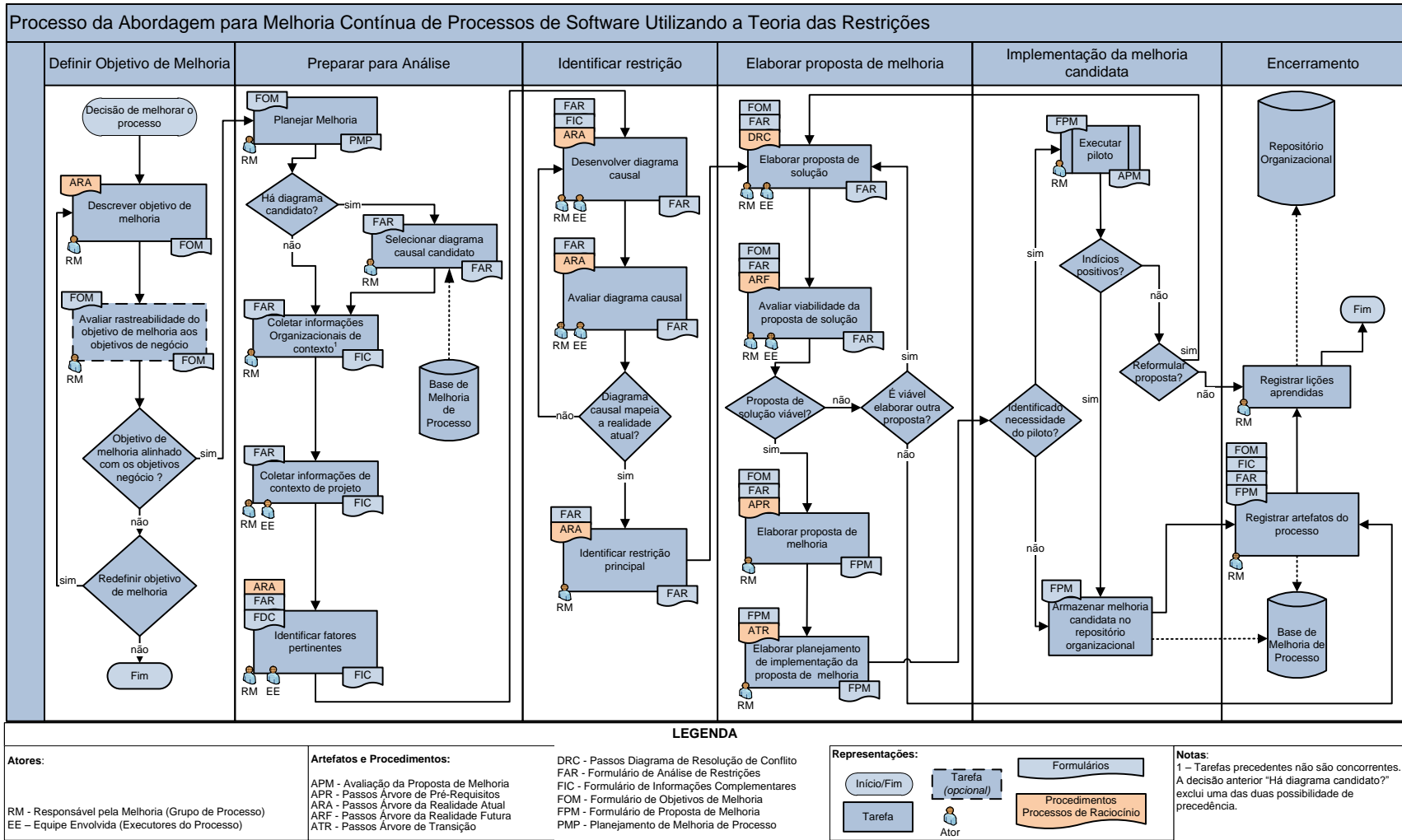
#### 4.2.1 Descrever objetivo de melhoria

Para executar o processo de melhoria, informações mínimas e de contexto são necessárias para se iniciar a investigação por fatores de influência que restrinjam o desempenho do processo. Processos estabilizados e que já possuem um histórico considerável de execuções podem vincular uma gama de dados, informações de contexto, lições aprendidas e demais ativos relacionados.

O objetivo desta tarefa é definir o objetivo de melhoria de forma a delimitar o escopo de investigação e definir a perspectiva de análise. De acordo com o Processo de Raciocínio da *Árvore da Realidade Atual*, as seguintes questões necessitam ser respondidas como ponto inicial da investigação da oportunidade de melhoria: (i) Qual o processo que está se analisando? (ii) Qual o propósito? (iii) Qual o ponto de vista? (iv) Quais são os principais indicadores de sucesso?

Para apoiar esta tarefa, esta abordagem propõe um formulário para preenchimento destas informações – o Formulário de Objetivo de Melhoria (FOM), apresentado na seção IV.A do Anexo IV – no qual o objetivo de melhoria é descrito de maneira estruturada, seguindo a notação baseada em GQM (VAN SOLINGEN e BERGHOUT, 1999). Informações referentes aos objetivos de qualidade e desempenho de processos relacionados à melhoria devem ser especificados no formulário para avaliação posterior dos resultados. O formulário ficará disponível na Base de Melhoria de Processo (BMP) ao final da execução do processo, anexado aos demais formulários gerados ao longo deste processo.

O objetivo descrito no FOM permite que, no ciclo de melhoria contínua, ações de melhoria executadas anteriormente que possuam o mesmo objetivo sejam resgatadas, a fim de reutilizar diagramas causais, raciocínio de decisões e lições aprendidas relacionadas. A Base de Melhoria de Processo deve servir como repositório a ser consultado para resgatar os pacotes de ações de melhoria, servindo como memória dos ciclos de melhoria contínua da organização.



**Figura 4.1 - Processo da abordagem proposta**

#### **4.2.2 Avaliar rastreabilidade do objetivo de melhoria aos objetivos de negócio**

Os objetivos de qualidade e desempenho de processos organizacionais proveem foco e direcionamento à análise do desempenho do processo com o intuito de identificar oportunidades de melhoria (SEI, 2010). Desta forma, o objetivo desta tarefa é verificar se a definição do objetivo de melhoria está alinhado com os objetivos de negócio relevantes por meio de um modelo de rastreabilidade<sup>7</sup>.

Esta tarefa é opcional e existe para manter a cobertura das práticas específicas e resultados esperados em organizações aderentes a níveis mais alto de maturidade do CMMI-DEV e MR-MPS.

O responsável pela ação de melhoria verifica a rastreabilidade do objetivo definido no passo anterior com os objetivos de qualidade e de desempenho da organização, relacionando-os no FOM. Neste momento, o mapeamento é utilizado para verificar se o objetivo definido está relacionado a um ou mais objetivos de qualidade e desempenho e se estes estão dentro das prioridades organizacionais.

Caso o objetivo da melhoria não esteja relacionado a nenhum dos objetivos de qualidade e desempenho de processo atualmente priorizados na organização, duas opções são possíveis: i) reformular o objetivo de melhoria, de forma que contribua com algum dos objetivos de qualidade e desempenho de processo priorizados; ii) revisar as prioridades dos objetivos organizacionais. A revisão destas prioridades é uma prática específica de organizações de alta maturidade e, portanto, precisa ser tratada de maneira sistemática. Este tratamento está fora do escopo desta abordagem, finalizando a execução do processo.

### **4.3 Preparar para Análise**

Esta atividade tem como objetivo planejar a melhoria e recuperar dados de contexto, diagrama causal e informações complementares que serão insumos para a investigação das restrições do processo e proposição da solução. A aplicação da ferramenta do Processo de Raciocínio da *Árvore de Realidade Atual* nesta atividade

---

<sup>7</sup> Existem várias abordagens para estabelecer o mapeamento entre os objetivos estratégicos (objetivos de negócio) e os objetivos táticos (objetivos de qualidade e desempenho de processo), tais como *Quality Function Deployment* (QFD), *Balanced Scoreboard* e *Goal Question Metric* (GQM).

está relacionada à investigação de informações com base no escopo definido no objetivo de melhoria e ao levantamento dos fatores de influência pertinentes que serão insumos para a construção/refinamento do diagrama causal.

As tarefas que compõem esta atividade são apresentadas nas subseções seguintes.

#### **4.3.1 Planejar melhoria**

O objetivo desta tarefa é o planejamento das atividades envolvidas na execução desta abordagem proposta que considera custos, esforço, prazos e riscos envolvidos na execução do processo aqui descrito.

O planejamento da melhoria deve, ainda, conter informações sobre os responsáveis da ação, os papéis envolvidos na análise, os membros da equipe que participarão da ação e demais recursos (ferramentas, salas de reunião, licenças de software, computadores, etc.).

Um modelo de plano – Planejamento de Melhoria de Processo (PMP) – pode ser encontrado na seção IV.B do Anexo IV.

#### **4.3.2 Selecionar diagrama causal candidato**

Esta tarefa objetiva selecionar um diagrama causal, armazenado na Base de Melhoria de Processo (BMP), que atenda aos critérios para ser reutilizado. O diagrama causal é o resultado da aplicação dos passos da construção da *Árvore da Realidade Atual* executados na atividade seguinte do processo desta abordagem – Identificar restrição.

Neste momento, o diagrama causal pode ser identificado como candidato, servindo como ponto de partida para identificação das restrições do processo. Com base no Formulário de Objetivo de Melhoria, o responsável pela ação verifica se há algum registro de melhoria na BMP para o processo em análise cujo propósito, perspectiva da melhoria (com relação à), ponto de vista e contexto sejam os mesmos. Além disso, as lições aprendidas que estão relacionadas à melhoria recuperada da BMP podem apoiar a decisão de reutilização ou não do diagrama causal. O responsável, com base nas lições aprendidas, pode concluir que a melhoria recuperada, vinculada a este diagrama causal, não obteve sucesso e, portanto, não convém ser reutilizado.

Se o diagrama causal for selecionado, o Formulário de Análise de Restrições (FAR) – encontrado na seção IV.D do Anexo IV – deve ser preenchido, contendo as informações referentes ao diagrama causal selecionado.

### 4.3.3 Coletar informações organizacionais de contexto

O objetivo desta tarefa é coletar informações organizacionais de contexto pertinentes para apoiar a análise e investigação da oportunidade de melhoria por meio de informações sobre fatores de influência do processo que permitam identificar restrições de caráter político organizacional ou comportamental.

FLORAC e CARLETON (1999) apresentam uma lista de entidades mensuráveis em processos de software que categorizam entidades que participam ao longo do processo de software e que contribuem para a sua variação. JURISTO e MORENO (2001) classificam e citam origens de variações em processos de software. ECKES (2001) também cita seis elementos principais, nomeados *componentes contribuidores*, que estão envolvidos nos processos de software e, portanto, são fontes de variação em processos. Todas estas fontes, identificadas na revisão da literatura descrita no Capítulo 2, podem ser utilizadas como ponto de partida para identificar entidades, fatores de influência e informações organizacionais que precisam ser observados neste momento. No entanto, estas informações e fatores de influência de processos de software não têm o objetivo de fornecer uma lista exaustiva e completa, mas apenas servir como ponto de partida para a análise. Além disso, outra estrutura existente para guiar o levantamento das informações organizacionais de contexto já utilizada na organização pode ser aproveitada.

As informações devem ser descritas seguindo as regras de objetividade e clareza das *Categorias de Ressalvas Legítimas* dos Processos de Raciocínio. Estas regras, descritas na Seção 3.4.1, auxiliam na construção da Árvore da Realidade Atual e na sua posterior interpretação e leitura. Desta forma, sentenças com interpretação vaga como “A rotatividade é alta” devem ser mais específicas como “A rotatividade mensal é de 10% (alta)”.

Outra forma de levantamento de informações complementares de contexto é a utilização de medidas correlacionadas. Há diversas técnicas estatísticas que poderiam auxiliar na análise dos fatores de influência com base em medidas coletadas. FLORAC e CARLETON (1999) listam alguns atributos mensuráveis de entidades de processos

de software que podem servir para a análise de correlação por meio de métodos estatísticos como, por exemplo, ANOVA. Pode-se, ainda, quando não for possível uma análise de correlação, utilizar uma lista de informações da organização e de suas áreas de apoio (RH, suporte de TI, garantia da qualidade, escritório de projeto) para contextualização do processo de melhoria.

O Formulário de Informações Complementares (FIC), encontrado na seção IV.C do Anexo IV, deve ser preenchido com informações resultantes da execução desta tarefa.

#### **4.3.4 Coletar informações de contexto de projetos**

O objetivo desta tarefa é acrescentar informações pertinentes de contexto dos projetos que possam apoiar a análise e investigação da oportunidade de melhoria por meio da técnica de *Brainstorming*, envolvendo os membros da equipe de projeto (cujo ponto de vista foi definido no FOM) e o responsável pela melhoria. O critério de escolha pelo *Brainstorming* foi calcado na sua simplicidade de aplicação e por ser uma técnica já utilizada nas organizações de software em geral. As oportunidades de melhoria advindas de diversas fontes podem ser consideradas no momento de formular as informações de contexto. A percepção, por exemplo, de uma alta taxa de não conformidade ou falta de aderência à determinada tarefa do processo pode ser indício de um fator de influência negativa.

As informações descritas devem observar as regras de objetividade e clareza das *Categorias de Ressalvas Legítimas* dos Processos de Raciocínio. Fatores de influência indesejáveis relacionados aos projetos devem ser listados sempre com a perspectiva de melhoria do desempenho do processo em análise. Neste momento, o responsável pela melhoria auxilia a equipe envolvida a pensar em alguns aspectos críticos que podem impactar negativamente no desempenho do processo, considerando as entidades e os fatores de influência indesejáveis descritos na seção anterior e preenchendo a FIC com as informações levantadas.

#### **4.3.5 Identificar fatores pertinentes**

Esta tarefa objetiva identificar os fatores que inicialmente aparentam ter maior influência indesejável sobre o processo cujo desempenho se deseja melhorar, estando fundamentada no segundo passo do Processo de Raciocínio da *Árvore da Realidade*

*Atual*. Estes fatores pertinentes serão considerados na tarefa de construção/refinamento do diagrama causal descrita na próxima atividade – Identificar restrição. SCHEINKOPF (1999) aconselha que esta lista se restrinja a não mais de dez e nem menos de cinco fatores.

Para identificação dos fatores pertinentes, SCHEINKOPF (1999) sugere que sejam observados: (i) fatores que indiquem desalinhamento com o propósito de melhoria; e (ii) causas aparentes pelas quais o processo não atinge o objetivo de desempenho desejado.

As informações organizacionais de contexto e de projetos coletadas e registradas no FIC são insumo para apoiar esta identificação. Os fatores de influência pertinentes são descritos em uma lista, considerando as percepções que a equipe envolvida teve da análise dos fatores. A partir da análise das informações de contexto e fatores de influência indesejáveis, por meio de um *Brainstorming* estruturado com a equipe envolvida, os fatores mais pertinentes são listados em ordem de prioridade de maneira consensual entre os membros da equipe envolvida. Ao final desta tarefa, o Formulário de Informações Complementares (FIC) devem ser preenchido com a lista de fatores pertinentes.

#### **4.4 Identificar restrição**

Esta atividade tem como objetivo identificar a principal restrição que limita o desempenho do processo alvo da melhoria. Para atingir este objetivo, os passos 3, 4 e 5 do Processo de Raciocínio da *Árvore da Realidade Atual* são aplicados. Cada passo da ARA é executado por meio das tarefas desta atividade, onde os procedimentos da construção da ARA são aplicados às peculiaridades dos processos de software, resultando no diagrama causal e na restrição principal do desempenho do processo (oportunidade de melhoria).

As subseções seguintes detalham as tarefas para identificar a restrição principal do desempenho do processo.

##### **4.4.1 Desenvolver diagrama causal**

Esta tarefa tem como objetivo construir o diagrama causal aplicando o procedimento do 3º e 4º passo do Processo de Raciocínio da *Árvore da Realidade Atual* descrito em (SCHEINKOPF, 1999). O responsável pela melhoria lidera, com o



apoio da equipe envolvida, a construção do diagrama causal com base nas informações do Formulário de Análise das Restrições (FAR) e do Formulário de Informações Complementares (FIC).

O procedimento do 3º passo da ARA descreve as ações para construir um diagrama causal com base nas relações de causa-e-efeito dos fatores de influência do processo. Com base no relacionamento de *Causa Suficiente* e nas *Categorias de Ressalvas Legítimas*, o diagrama causal é construído com o intuito de refletir os efeitos e fatores de influência indesejáveis da realidade atual do processo, de acordo com a perspectiva e ponto de vista definidos no objetivo de melhoria. No 4º passo da ARA, uma revisão é feita, garantindo que o diagrama esteja claro e que cada relacionamento de causa-e-efeito tenha sido averiguado e desenvolvido, quando pertinente.

Se um diagrama causal tiver sido selecionado para reutilização na tarefa anterior, a primeira ação do procedimento descrito por SCHEINKOPF (1999) no 3º passo da ARA é adaptada para considerar o diagrama causal selecionado como ponto de partida da construção.

De acordo com o nível de maturidade da organização e capacidade do processo, a investigação da oportunidade de melhoria terá um nível de detalhamento diferenciado. Para processos sob controle estatístico, os fatores de influência das causas comuns são relacionados e evidenciam fatores limitadores que com dificuldade seriam percebidos sem o diagrama causal. Para organizações que estejam em níveis iniciais de maturidade e cujo processo alvo da melhoria ainda não seja estável, o desenvolvimento do diagrama causal considerará os fatores de influência das causas atribuíveis (fatores indesejáveis) e direcionará a identificação da oportunidade de melhoria (restrição principal) que mais tenha impacto em termos de efeitos indesejáveis para o processo.

Uma consideração pertinente na construção deste diagrama e dos demais diagramas dos Processos de Raciocínio é que estes possuem representações gráficas específicas. Para construí-lo, ferramentas gráficas voltadas para a construção dos diagramas, tais como *Transformation Logic Tree*<sup>8</sup> e *Flying Logic*<sup>9</sup>, auxiliam substancialmente o trabalho de construção. Contudo, ferramentas gráficas mais

---

<sup>8</sup> <http://www.transformation-logictree.com/>

<sup>9</sup> <http://flyinglogic.com/>

genéricas como, por exemplo, Microsoft Visio<sup>10</sup> e LibreOffice Draw<sup>11</sup> podem apoiar a elaboração dos diagramas.

O procedimento que reúne as ações para construção do diagrama causal de acordo com (SCHEINKOPF, 1999) está apresentado na seção III.B do Anexo III.

No momento da construção, SCHEINKOPF (1999) ressalta que a equipe deve se concentrar em um passo e uma entidade de cada vez, evitando adicionar mais fatores à lista de fatores pertinentes. Os fatores de influência do processo e seus efeitos devem surgir ao longo da aplicação desse procedimento, principalmente pela aplicação das CRL.

Ao final da construção, o diagrama causal deve conter um ou mais efeitos de fatores de influência indesejáveis que representem o oposto do que o objetivo de melhoria almeja. O diagrama causal resultado desta tarefa deve constar no FAR. Na próxima tarefa de avaliação do diagrama causal, descrita na subseção seguinte, a equipe envolvida revisa e, se necessário, refina o diagrama causal.

#### **4.4.2 Avaliar diagrama causal**

Esta tarefa objetiva avaliar o diagrama causal construído na tarefa descrita na subseção anterior. O 4º passo do Processo de Raciocínio da *Árvore da Realidade Atual* descreve a avaliação do diagrama causal por parte dos envolvidos. A leitura do diagrama deve ser feita a partir do elemento no nível mais abaixo em direção aos elementos do topo, utilizando a terminologia “*Se fator de influência A ocorre então fator de influência B ocorre*”.

O consenso da equipe envolvida sobre o mapeamento dos fatores de influência indesejáveis é importante, sendo o principal balizador para concluir a construção do diagrama causal (SCHEINKOPF, 1999). Caso não haja consenso sobre a completude e clareza do diagrama causal, deve-se repetir a execução dos passos 5 e 6 do procedimento descrito na tarefa anterior – Desenvolver diagrama causal.

---

<sup>10</sup> <http://office.microsoft.com/pt-br/visio/>

<sup>11</sup> <http://www.libreoffice.org/>

#### **4.4.3 Identificar restrição principal**

O objetivo desta tarefa é identificar a restrição principal – fator de influência indesejável que mais limita o desempenho do processo. Os dois últimos passos do Processo de Raciocínio da *Árvore da Realidade Atual* descrevem o procedimento para identificar a restrição principal. Esta tarefa responde à pergunta: “O que mudar?”.

Neste momento, os relacionamentos de causa-e-efeito são ponderados, a fim de identificar a restrição principal (*ponto de entrada*) através da qual a maior parte dos fatores de influência indesejáveis presentes no diagrama causal é alcançada, isto é, está relacionada direta ou indiretamente (através de uma cadeia de causa-e-efeito) como efeito.

O procedimento que descreve os passos do Processo de Raciocínio da ARA para identificar a restrição principal está descrito na seção III.C no Anexo III. Nesta tarefa, o FAR deve ser preenchido com a matriz de rastreabilidade das restrições candidatas na qual os fatores de influência e os seus efeitos diretos ou indiretos são mapeados.

#### **4.5 Elaborar proposta de melhoria**

O propósito desta atividade é identificar a melhoria candidata e elaborar um planejamento para sua implementação, descrevendo cronograma, esforço, riscos, custos relacionados. Para atingir este objetivo, uma proposta de solução é gerada. Se for avaliada como financeira e operacionalmente viável, uma proposta de melhoria é definida, juntamente com seu planejamento de implementação. Quatro Processos de Raciocínio da TDR serão aplicados para atingir este objetivo: o *Diagrama de Resolução de Conflito*, a *Árvore da Realidade Futura*, a *Árvore de Pré-Requisitos* e a *Árvore de Transição*.

A partir da restrição principal identificada, a proposta de solução mais óbvia é a sua remoção imediata. Se não for possível ou se ocasionar detrimento de outro passo do processo por conta da remoção da restrição principal, há um conflito que precisa ser resolvido para atingir o objetivo de melhoria. Para isto, o Processo de Raciocínio DRC é aplicado para desenvolver uma proposta de solução que anular a restrição principal através da análise dos pressupostos que sustentam o conflito.

Com base na proposta de solução, os passos da ARF são executados para averiguar se a solução é viável, isto é, se é possível desenvolver uma proposta de melhoria a partir dela. Se for viável, um planejamento para a implementação da proposta de melhoria é elaborado, considerando os procedimentos baseados nos Processos de Raciocínio APR e ATR. As subseções seguintes detalham estas tarefas.

#### **4.5.1 Elaborar proposta de solução**

Esta tarefa tem o objetivo de elaborar uma proposta de solução para remover ou anular a restrição principal ao desempenho do processo, a fim de atingir o objetivo de melhoria. Esta tarefa responde à pergunta: “Para o que mudar?”.

O objetivo de melhoria pode ser alcançado pela proposta de solução direta de remoção da restrição principal do processo. Contudo, para processos sob controle estatístico, os efeitos e fatores de influência indesejáveis são resultados de causas comuns do processo. Desta forma, a simples remoção da restrição principal, por ser inerente, pode não ser possível. Desta forma, se houver um conflito no momento de formular a proposta de solução, o Processo de Raciocínio *Diagrama de Resolução de Conflito* pode ser aplicado para representar o conflito relacionado, possibilitando identificar pressupostos e proposições de solução conciliável para anular o conflito e anular a restrição principal (SCHEINKOPF, 1999). No tocante aos tipos de pressupostos identificados no DRC, um ganho de eficiência e eficácia maior é estimado, quando pressupostos de política e comportamentais são identificados e suas restrições removidas (DETTMER, 1997).

Os passos do Processo de Raciocínio do *Diagrama de Resolução de Conflito* são apresentados na Seção III.D do Anexo III. O Formulário de Análise de Restrições (FAR) deve ser preenchido com as informações produzidas nesta tarefa (diagrama de conflito, pressupostos, proposta de solução e raciocínio).

#### **4.5.2 Avaliar viabilidade da proposta de solução**

Esta tarefa objetiva avaliar a viabilidade da proposta de solução formulada na subseção 4.5.1. Para isto, a *Árvore da Realidade Futura* é aplicada para identificar ações necessárias para implementar a proposta de solução, os riscos envolvidos e as ações de mitigação, a fim de embasar a avaliação de viabilidade da proposta de solução.

Ao final da execução desta atividade, o responsável pela melhoria deve ser capaz de avaliar se a proposta de solução é tecnicamente viável, tendo em vista que as ações que compõem a ARF serem viáveis. A avaliação de viabilidade financeira é verificada por meio da avaliação do orçamento estimado para executar as ações previstas na ARF. As informações referentes ao orçamento e ao prazo destinados à elaboração de melhoria e sua implementação devem servir como critérios para esta análise. As informações produzidas referentes à análise de viabilidade devem constar no FAR.

Se a proposta de solução for considerada inviável, o responsável pela melhoria, com base no planejamento da melhoria, descrito no Formulário de Objetivo de Melhoria (FOM), deve decidir se há tempo hábil e recursos disponíveis para a atividade de elaboração de proposta de solução.

O procedimento baseado no Processo de Raciocínio ARF proposto por (SCHEINKOPF, 1999) é apresentado na Seção III.E do Anexo III.

#### **4.5.3 Elaborar proposta de melhoria**

O objetivo desta tarefa é elaborar uma proposta de melhoria em termos de sua definição e validação para implantação na organização. A definição da proposta de melhoria deve descrever o objetivo da proposta de melhoria (que precisa estar alinhada ao objetivo de melhoria descrita no Formulário de Objetivo de Melhoria), os critérios para avaliar a efetividade da melhoria implementada e a relação de custo-benefício. A decisão por executar um método de validação – piloto – deve ser descrito com base na análise da lista de riscos envolvidos da proposta de solução.

Nesta tarefa, o responsável pela melhoria deve executar o procedimento, solicitando o apoio pontual da equipe envolvida apenas para considerar aspectos técnicos na elaboração da proposta de melhoria. Isto possibilita que o responsável pela ação de melhoria seja mais independente, agilizando a geração da execução da tarefa.

Como descrito no Capítulo 3, a decisão pela utilização da APR deve ser feita de acordo com a complexidade e dificuldade de atingir o objetivo da proposta de solução. Isto significa que, se a dificuldade e impacto gerado pela proposta de solução forem considerados altos, objetivos intermediários devem ser identificados para maximizar o sucesso da proposta de melhoria. Esta decisão impacta tanto na execução do procedimento da APR nesta atividade como na execução da validação, através de

um piloto, na atividade seguinte. O processo de decisão formal, presente em organizações de alta maturidade, deve ser executado para determinar se a proposta de solução deve ser validada por um piloto. Caso a decisão pela execução do piloto for confirmada, o raciocínio do processo de decisão deve ser registrado, assim como a execução do procedimento da APR executado. O Formulário de Proposta de Melhoria (FPM), encontrado na Seção IV.E do Anexo IV, deve ser preenchido com os resultados produzidos nesta tarefa.

O primeiro passo desta tarefa é definir uma declaração do objetivo da proposta de melhoria e os critérios quantitativos de efetividade, tomando como base o objetivo de melhoria descrito no FOM e a proposta de solução desenvolvida na tarefa 4.5.1 – Elaborar proposta de solução. Os critérios quantitativos da efetividade da proposta de melhoria devem estar relacionados aos objetivos de qualidade e desempenho organizacionais descritos no FOM para a melhoria.

O segundo passo é executar o processo de decisão formal da execução da validação através de piloto, com base nos riscos envolvidos, cronograma macro e orçamento estimado referente à proposta de solução, como descrito no FAR. O resultado do processo de decisão deve ser registrado no FPM.

Se o resultado da decisão formal não for pela execução do piloto, a execução desta se encerra após o mapeamento de riscos da proposta de melhoria. Com base nos riscos (efeitos colaterais e obstáculos) e ações de mitigações já identificadas para a proposta de solução, a matriz de riscos deve ser preenchida. Informações adicionais de probabilidade, impacto, disparo de contingência e ações de contingência podem ser preenchidas com as diretrizes existentes na organização para o planejamento de riscos.

Se a execução da validação por piloto for decidida, o responsável pela melhoria deve desenvolver uma estrutura analítica – como, por exemplo, uma WBS (*Work Breakdown Structure*) – para delimitar a melhoria e os objetivos intermediários da proposta de melhoria. Neste passo, a estrutura analítica deve ser construída a partir da árvore criada pelo Processo de Raciocínio APR. O topo da estrutura é o objetivo da proposta de solução, como descrita no FAR. Os níveis intermediários são compostos por objetivos intermediários, respeitando a mesma ordem de precedência definida na APR. As folhas da estrutura analítica são constituídas pelas (*injeções*) para transpor os obstáculos relacionados aos objetivos intermediários. Se necessário, uma estrutura analítica para declarar explicitamente o que está fora do escopo da proposta de

melhoria deve ser elaborada. Contudo, este item é opcional. O FPM apresenta modelos de estrutura analítica de acordo com o descrito acima.

No quarto passo, o mapeamento de riscos é elaborado, listando os riscos de não conseguir superar os obstáculos identificados. Devem ser listados todos os riscos referentes a possíveis efeitos colaterais identificados para a proposta de solução e descritos no FAR. A lista de riscos relacionados aos obstáculos, identificados no FAR, da proposta de solução servirá como referência no procedimento da APR. Informações adicionais da matriz de risco devem ser complementadas seguindo os métodos atualmente utilizados pela organização para o planejamento de riscos.

O procedimento baseado no Processo de Raciocínio da APR descrito em (SCHEINKOPF, 1999) é apresentado na Seção III.F do Anexo III.

A estrutura analítica pode ser gerada a partir da APR, removendo os rótulos referentes aos obstáculos. Ao transformar a APR em estrutura analítica, o mapeamento entre risco e obstáculo deve ser observado, preenchendo-se a coluna referente na matriz.

#### **4.5.4 Elaborar planejamento de implementação da proposta de melhoria**

O objetivo desta tarefa é elaborar um planejamento detalhado da implementação da proposta de melhoria. O planejamento deve ser especificado em termos de cronograma (prazo e esforço), orçamento e análise de custo-benefício. O Processo de Raciocínio *Árvore de Transição* viabiliza a implementação da proposta de melhoria de maneira operacional (LACERDA *et al.*, 2004), gerando o plano de ações detalhado que serve como insumo para a elaboração do cronograma e do orçamento da proposta de melhoria.

Com base nos objetivos intermediários encontrados na estrutura analítica no Formulário de Proposta de Melhoria, o procedimento de construção da ATR permite definir quais atividades são necessárias para operacionalizar cada ação de anulação de obstáculos e ações de mitigação. Se a estrutura analítica não existir no FPM, significa que a proposta de melhoria não prevê execução de piloto e sua implementação não é de alto risco. Neste caso, o objetivo da proposta de solução deverá ser considerado como objetivo final a ser atingido pelo plano de implementação.

O preenchimento da planilha de orçamento e o cronograma deverão ser iniciados com base nas informações contidas no FAR referentes à proposta de solução. Informações suplementares devem ser preenchidas (data estimada, valor de contingência, esforço estimado e precedentes) com base no encadeamento das ações identificadas na tarefa 4.5.2 – Avaliar viabilidade da proposta de solução. A execução do procedimento de ATR irá refinar as informações das ações do cronograma e conseqüentemente o orçamento.

O primeiro passo desta tarefa é executar o procedimento baseado no Processo de Raciocínio da ATR, como apresentado por (SCHEINKOPF, 1999), é descrito na Seção III.G do Anexo III.

Para melhoria em processo sob controle estatístico, durante a execução do procedimento da ATR, um cuidado com a estabilidade do processo deve acontecer. Ao longo do planejamento das ações para implementação da proposta de melhoria, cada decisão por inserir uma ação no planejamento pode impactar o comportamento do processo. O responsável pela melhoria deve considerar revisar o plano de ações com os executores do processo, a fim de identificar possíveis impactos negativos não visualizados no momento da definição do planejamento. Ao final da execução deste passo, o plano de ação para a implantação da proposta de melhoria é a composição das ações da ATR.

O segundo passo desta tarefa é consolidar informações do orçamento com base no cronograma definido. O esforço estimado de cada atividade do cronograma pode derivar em custos. A elaboração do orçamento com base no cronograma deve ser as diretrizes e melhores práticas utilizadas na organização.

O último passo desta tarefa está relacionado à análise de custo-benefício da proposta de melhoria. O objetivo, riscos identificados, custos, esforço e tempo atrelados à melhoria podem subsidiar o cálculo do fator de retorno de investimento da proposta de melhoria. Este raciocínio deve ser registrado no FPM para ser utilizado posteriormente na priorização e na seleção da melhoria candidata.

#### **4.6 Implementar a melhoria candidata**

O objetivo desta atividade é implementar a melhoria candidata de acordo com seu o planejamento, a fim de validar a melhoria candidata como apta à implantação na



organização. A execução de um piloto pode ocorrer como planejado, a fim de avaliar os resultados da melhoria candidata em condições conhecidas e controladas. Neste momento, o responsável pela melhoria deve analisar, com base nos indícios resultantes do piloto, se a melhoria candidata está apta a ser implantada na organização ou se uma nova proposta de solução deve ser desenvolvida. Se válida, a melhoria candidata é armazenada no repositório organizacional, pronta para ser selecionada para implantação na organização, de acordo com o critério de decisão.

Somente após a implantação será possível averiguar se os objetivos de melhoria foram efetivamente alcançados. O processo de priorização, seleção, planejamento, implantação e gerenciamento da implantação da melhoria não faz parte do escopo da abordagem desta dissertação. Contudo, dois resultados advindos da implantação são importantes para este processo: i) os resultados obtidos a partir da implantação da melhoria devem ser armazenados junto aos ativos de processo que originaram a melhoria (formulários), a fim de prover lições aprendidas e instruções para ciclos posteriores de melhoria; ii) a partir do resultado obtido com a implantação, o processo de reformulação da proposta de solução pode ser disparado. Neste momento, deve ser averiguado se houve interrupção considerável que possa impactar na continuação da formulação da proposta de solução, como por exemplo, mudança de equipe.

As subseções seguintes detalham as tarefas para implementar a melhoria candidata.

#### **4.6.1 Executar piloto**

O propósito desta tarefa é conduzir um piloto para avaliar o alcance do objetivo de melhoria proposto. Pilotos são métodos de avaliação que devem ser utilizados para mudanças significativas que envolvam melhorias inovadoras, de alto risco ou cujos resultados podem ser inesperados, pois são métodos de custo elevado. A decisão por execução de um projeto piloto ocorre na atividade anterior – Elaborar proposta de melhoria – fundamentada em critérios organizacionais definidos, tais como o risco envolvido ou o número de áreas funcionais afetadas.

A abordagem descrita por SILVA FILHO (2006) e apresentada no Capítulo 2 propõe um processo para execução de projeto piloto voltado para organizações de software. A entrada do processo da abordagem é a proposta de melhoria. O Formulário

de Proposta de Melhoria deve ser submetido como insumo para o processo de execução do piloto. Ao longo da execução das atividades do *Pilot* (SILVA FILHO, 2006), o cronograma e orçamento da proposta de melhoria, contida no FPM, podem ser atualizados. O resultado da execução do piloto é um laudo, cujas especificações possibilitam identificar se houve indícios de alcance do objetivo de melhoria da proposta.

Neste momento, o responsável pela melhoria verifica se os indícios do laudo são positivos para concluir que a proposta de melhoria é candidata para implantação na organização, ou se os indícios não foram suficientes. Neste caso, com base no orçamento e cronograma definidos no Planejamento de Melhoria de Processo (PMP), o responsável pela melhoria pode decidir por reformular a proposta de solução e, conseqüentemente, a proposta de melhoria por encerrar o processo. Nos dois casos, o Formulário de Análise de Restrições e o Formulário de Proposta de Melhoria devem ser atualizados com o raciocínio utilizado nesta decisão.

#### **4.6.2 Armazenar melhoria candidata**

Esta tarefa tem o objetivo de armazenar a melhoria candidata, descrita no Formulário de Proposta de Melhoria, na Base de Melhoria de Processo (BMP). A melhoria candidata deve ter sido avaliada para averiguar se há indícios de que a melhoria proposta alcançará os objetivos.

Em momento posterior, de acordo com critérios definidos pela organização, a proposta de melhoria deve ser resgatada da BMP para implantação na organização. O processo de implantação não faz parte do escopo deste trabalho. No entanto, é importante ressaltar que neste processo pode haver registro de lições aprendidas ou de modificação na proposta de melhoria importantes e, portanto, devem constar no repositório organizacional.

#### **4.7 Registrar resultados**

O propósito desta atividade é registrar os resultados obtidos, coprodutos de processo gerados e lições aprendidas para permitir que ciclos futuros de melhoria possam reutilizar ativos gerados pela execução do processo (diagrama causal, geração de proposta de solução, análise de viabilidade, lições aprendidas, planos de implementação e resultados de validação de melhoria), de acordo com o preconizado

pelos modelos de melhoria de processos de software. A reutilização da experiência adquirida durante a elaboração de melhorias candidatas é característico de organizações de alta maturidade.

As duas tarefas que compõem esta atividade são descritas nas subseções seguintes.

#### **4.7.1 Registrar artefatos do processo**

O objetivo desta tarefa é registrar na Base de Melhoria de Processo (BMP) os produtos das atividades do processo desta abordagem. Os formulários sugeridos para apoiar a execução do processo reúnem todas as informações relevantes que devem ser armazenadas. Os formulários preenchidos, juntamente com quaisquer anexos ou arquivos gerados referentes à execução do processo, devem ser registrados no repositório da organização.

#### **4.7.2 Registrar lições aprendidas**

Esta tarefa tem o propósito de registrar as lições aprendidas durante a execução do processo desta abordagem no repositório organizacional, servindo como aprendizado da execução do processo e sugestão de melhoria do processo aqui descrito.

As lições registradas devem estar disponíveis para possíveis consultas em futuras execuções do processo.

### **4.8 Considerações Finais**

A abordagem proposta nesta dissertação e apresentada neste capítulo tem o objetivo de apoiar a melhoria contínua de processos de software no contexto de organizações de software. Esta abordagem foi apresentada por meio de um processo de seis atividades, a saber: Definir Objetivo de Melhoria, Preparar para Análise, Identificar Restrição, Elaborar Proposta de Melhoria, Implementar da Melhoria Candidata e Registrar Resultados.

Os Processos de Raciocínio da Teoria das Restrições foram aplicados em algumas tarefas do processo, apoiando a investigação de oportunidade de melhoria, identificação de propostas de solução e elaboração de proposta de melhoria.

Para avaliar a abordagem proposta, um estudo de viabilidade foi realizado e está descrito no próximo capítulo.

## Capítulo 5 – Avaliação da Abordagem

*Este capítulo apresenta a avaliação realizada como parte da metodologia de pesquisa. A avaliação, por meio de um estudo de viabilidade, verificou a aplicabilidade da abordagem e a sua viabilidade, em termos de tempo de execução, em uma organização de software.*

### 5.1 Introdução

De acordo com os princípios da Engenharia de Software Experimental, para o refinamento de novas tecnologias na área de Engenharia de Software, estudos experimentais executados de maneira sistemática e sucessiva são fundamentais para amadurecer e evoluir a tecnologia proposta, e dirimir os riscos inerentes à sua aplicação na indústria (SHULL *et al.*, 2001).

No momento da definição da abordagem descrita nesta dissertação, indícios de sua aplicabilidade foram avaliados por especialista, resultando em pequenos ajustes e refinamentos. Em um momento posterior, o estudo de viabilidade foi executado em uma organização de software, considerando parte do processo, a fim de averiguar a aplicabilidade da abordagem e a viabilidade, em termos do tempo de execução, do processo.

Este capítulo apresenta, além desta introdução, a Seção 5.2, descrevendo a natureza do estudo experimental. Na Seção 5.3, é apresentado o contexto da organização e do processo selecionado para melhoria. Na Seção 5.4, a definição do estudo é descrita, assim como suas questões de pesquisa e seu objetivo. Na Seção 5.5, é apresentada a execução do estudo. Na Seção 5.6, é descrita a análise e as interpretações dos resultados obtidos. Na Seção 5.7, considerações finais a este capítulo são pontuadas.

## 5.2 Estudo de Viabilidade

Segundo SHULL *et. al.* (2001), novas tecnologias propostas podem ser avaliadas no tocante à eficácia dos seus métodos, técnicas e ferramentas, considerando diferentes ambientes de aplicação. Este estudo, de acordo com SHULL *et. al.* (2001), é importante para testar se a tecnologia proposta é viável, justificando a continuação do desenvolvimento da pesquisa.

A metodologia de pesquisa adotada nesta dissertação contempla um estudo de viabilidade para averiguar a aplicabilidade da abordagem proposta para melhoria de processo de software. Desta forma, o estudo foi conduzido com o intuito de verificar a aplicabilidade do processo da abordagem, dos procedimentos dos Processos de Raciocínio, e aferir o tempo de execução e os resultados alcançados com a sua aplicação.

O processo de experimentação seguido neste estudo de viabilidade foi o definido por MAFRA e TRAVASSOS (2006). A subseção a seguir contextualiza a empresa onde foi realizado o estudo de viabilidade. Além da seção de contextualização, as atividades de definição, planejamento, execução, análise e interpretação, que foram executadas neste estudo, estão descritas nas subseções seguintes. Este capítulo contém o resultado da atividade de empacotamento, como previsto por MAFRA e TRAVASSOS (2006).

## 5.3 Contexto de Aplicação do Estudo de Viabilidade

Este estudo foi realizado em uma empresa de desenvolvimento de software que, por questões de confidencialidade, não será identificada. A empresa, com sede no estado do Rio de Janeiro, é certificada com ISO 9001:2008, avaliada nível 2 do CMMI-DEV e nível F do MR-MPS. Atualmente a empresa prepara-se para ser avaliada no nível 3 do CMMI-DEV e nível C do MR-MPS.

A empresa comercializa, além dos seus serviços, produtos considerados COTS<sup>12</sup>. O processo de desenvolvimento, suporte e atendimento de chamados, por

---

<sup>12</sup> *Commercial Off-The-Shelf* – Componentes de software que oferecem um nível de serviço ou funcionalidade.

exemplo, são institucionalizados, a fim de padronizar o tratamento com o cliente e o suporte dos seus produtos.

Alguns produtos, por serem estratégicos para a organização e possuírem uma carteira de clientes maior, são alvos de melhorias mais frequentes. Para o estudo de viabilidade, a empresa selecionou o processo “Atendimento de Chamados” com dados do projeto cuja carteira de clientes era a maior. Este processo é alvo de melhoria contínua pela organização e sua base de medição é confiável, pois é integrada com a ferramenta de abertura e acompanhamento de atendimentos de chamados.

## **5.4 Definição do Estudo de Viabilidade**

Este estudo experimental tem como objetivo analisar qualitativamente a aplicabilidade da abordagem proposta para melhoria de processo de software utilizando a Teoria das Restrições e aferir o seu tempo de execução. Desta forma, o foco da qualidade é a exequibilidade e o tempo decorrido para execução de cada tarefa do processo.

A hipótese nula que se espera refutar ao final deste estudo experimental é:

*H<sub>0</sub> - A abordagem proposta para melhoria de processo de software utilizando a Teoria das Restrições não é aplicável à melhoria de processos de software.*

Deste modo, a hipótese alternativa desejável é que a abordagem proposta para melhoria de processos de software utilizando a Teoria das Restrições seja aplicável à melhoria de processo de software.

A avaliação da viabilidade acontece por meio da perspectiva do pesquisador que observa se os passos do processo conseguem ser executados pelo responsável da ação de melhoria, acompanhando o tempo de execução e pontuando oportunidades de melhoria identificadas.

Enquanto estudo experimental, este estudo de viabilidade não tem intenção de comparar esta abordagem proposta com outras técnicas (manuais ou automatizadas). Desta forma, a fim de identificar oportunidades de melhoria para a abordagem proposta, as notas tomadas pelo pesquisador no momento da execução do processo e as dificuldades percebidas ao longo da execução por parte do Responsável do Grupo de Processo (RGP) e os demais membros da equipe foram analisadas e estão descritas no resultado deste estudo.

O contexto de execução deste estudo de viabilidade é o da indústria em uma organização de software. O subprocesso analisado foi selecionado com base em uma análise quantitativa que precedeu este estudo e que auxiliou na identificação da melhor oportunidade de aplicação da abordagem proposta. Esta análise identificou, através de gráficos de controle e análise de especialista, passos do processo sobre o qual o esforço de melhoria traria resultados mais significativos para o processo. Isto possibilitou identificar os participantes do estudo que poderiam contribuir de maneira mais substantiva.

As questões que se espera que sejam respondidas com este estudo de viabilidade são:

- **Q1:** Qual o grau de satisfação do responsável pela melhoria quanto à facilidade de utilização da abordagem?
- **Q2:** Qual o grau de satisfação do responsável pela melhoria quanto ao tempo de utilização da abordagem?
- **Q3:** Qual o grau de satisfação do responsável pela melhoria quanto ao resultado obtido com a aplicação da abordagem?

Desta forma, utilizando a notação baseada em GQM (BASILI e ROMBACH, 1988), temos:

**Analisar** a execução da abordagem proposta para melhoria de processos de software utilizando a Teoria das Restrições

**Com o propósito de** avaliar a aplicabilidade da abordagem proposta

**Em relação à** exequibilidade e ao tempo de execução do processo da abordagem proposta

**Do ponto de vista** do pesquisador

**No contexto de** melhoria de processo de software em uma equipe que trabalha em “Atendimento de chamados” em uma organização de desenvolvimento de software

No tocante à exequibilidade, as questões Q1 e Q2 referem-se à percepção do responsável pela melhoria – RGP – relacionada à facilidade de execução da abordagem para melhoria de processo de software. A facilidade de execução do processo e clareza nos passos a serem tomados, relacionados à questão Q1, tem o



objetivo de verificar se o processo da abordagem está inteligível e se os passos podem ser seguidos por um especialista em melhoria de processo. O tempo de execução factível, relacionado à questão Q2, busca capturar a percepção dos participantes – responsável do grupo de processo e membro da equipe – sobre a razoabilidade do tempo gasto com a aplicação da abordagem. A questão Q3 ressalta a coerência entre o resultado obtido e a realidade, capturando a percepção de utilidade da abordagem para a melhoria do processo de software.

Os instrumentos utilizados durante o estudo foram o cronograma de acompanhamento do processo que deve registrar o tempo de execução de cada tarefa do processo preenchido pelo pesquisador e pelo responsável pela melhoria; os formulários de consentimento de pesquisa e de avaliação da abordagem que devem ser preenchidos pelo responsável pela melhoria e pelo membro da equipe.

Em uma primeira reunião, o responsável pela melhoria recebeu um treinamento no qual o pesquisador, por meio de uma apresentação, evidenciou o propósito do estudo, a abordagem proposta, contextualização e as atividades do processo. Alguns insumos referentes à abordagem foram providos ao RGP para leitura e contextualização como o processo da abordagem e os procedimentos dos Processos de Raciocínio. Os insumos providos foram:

- **Processo:** descrição das atividades e tarefas do processo conforme descrito no Anexo II;
- **Procedimentos dos Processos de Raciocínio:** descrição dos procedimentos dos Processos de Raciocínio da Teoria das Restrições conforme descrito no Anexo III;
- **Formulários de coleta de informações e de análise:** formulários para apoiar a execução das atividades do processo, conforme apresentados no Anexo IV;

A seleção dos participantes do estudo foi feita por conveniência. O único critério utilizado foi a participação de executores do processo no momento da utilização dos Processos de Raciocínio. Os papéis necessários para a execução do projeto eram: i) responsável do grupo de processo; ii) membro da equipe executora do processo. Desta forma, duas pessoas participaram do estudo, sendo que uma delas

executou o papel de RGP e de membro da equipe, pois fazia parte da área de qualidade e conhecia bem a execução do processo.

Após a assinatura dos formulários de consentimento do estudo de viabilidade, uma apresentação foi feita aos participantes para contextualizá-los sobre os objetivos da aplicação da abordagem proposta na melhoria do processo Atendimento de Chamados. Ao longo da execução da abordagem, o pesquisador, e o responsável pela melhoria, registraram o tempo tomado por cada atividade do processo. Ao final da execução, cada participante preencheu o formulário de avaliação do estudo de viabilidade, pontuando e descrevendo suas percepções em relação à abordagem. Os formulários de consentimento e de avaliação do estudo de viabilidade (modelos e formulários preenchidos) encontram-se no Anexo V.

As variáveis independentes deste estudo de viabilidade estão relacionadas aos participantes do estudo, ao processo selecionado para melhoria e à organização. O responsável do Grupo de Processo possui mais de 10 anos na área de qualidade da empresa, tinha certificações referentes à gerência de projetos e é implementadora MPS credenciada pela SOFTEX, além de participar ativamente da definição, melhoria e acompanhamento dos processos na organização (inclusive do processo Atendimento de Chamados). O membro da equipe selecionado para o estudo tem mais de 2 anos de experiência no produto e no atendimento de chamados, sendo um Analista de Negócios e Especialista de Domínio sênior.

As variáveis dependentes que se almeja observar são: i) facilidade da execução do processo da abordagem; e ii) tempo de execução de cada atividade do processo da abordagem proposta. Este tempo foi contabilizado em minutos e monitorado pelo RGP em paralelo com o pesquisador. A ausência de execução de um passo ou interrupção por conta de falta de entendimento (falta de clareza) nos passos do processo foi anotada pelo pesquisador e relatada nos resultados do estudo de viabilidade.

As percepções dos envolvidos na execução do estudo (formulário de avaliação), as observações feitas pelo pesquisador no momento da execução do estudo de viabilidade e o apoio à execução do processo da abordagem foram analisados qualitativamente. O resultado da análise está descrito na seção seguinte.

De acordo com WÖHLIN *et al.* (2000), é necessário identificar ameaças à validade do estudo experimental, pois tais ameaças podem ter impactado ou limitado os resultados do estudo de viabilidade. Os autores citam as seguintes ameaças:

#### **5.4.1 Validade Interna do Estudo**

Consideram os eventos que não podem ser controlados pelo pesquisador, mas que podem ter seus efeitos minimizados. A validade interna do estudo pode ter sido impactada por: i) RGP não registrar corretamente o tempo decorrido por conta da tutoria que ocorreu em paralelo à execução do estudo; ii) o desempenho dos participantes ser melhor do que o percebido no cotidiano pelo motivo de estarem sendo observados; iii) representatividade (número de participantes) baixa; iv) o processo não foi executado em sua totalidade.

É importante considerar, sobre esta última ameaça à validade interna do estudo, que o processo proposto considera um ciclo de melhoria para processos de software, como citado na revisão da literatura no Capítulo 2. No entanto, seus resultados de melhoria não puderam ser averiguados no escopo desta dissertação por questões de tempo hábil para observar se a melhoria foi efetiva ao longo do tempo. Contudo, ao que tange a aplicação dos Processos de Raciocínio, o estudo compreendeu a execução das tarefas da abordagem cuja composição incluía os Processos de Raciocínio responsáveis pela investigação da oportunidade de melhoria e pela elaboração da estratégia de solução.

#### **5.4.2 Validade Externa do Estudo**

A validade externa do estudo ressalta os eventos externos que podem prejudicar a análise e generalizações dos resultados do estudo.

Por questão de tempo restrito para a pesquisa e disponibilidade de organizações candidatas, a execução do estudo aconteceu apenas em um cenário de uma organização de software. O cenário é composto, dentre outros aspectos, pelo porte da empresa, o perfil dos funcionários, a capacidade dos processos e sua maturidade em termos de melhoria contínua dos seus processos organizacionais. A amostragem com representatividade baixa, diante das várias possibilidades de aplicação da abordagem nos diferentes cenários de melhoria de processo de software, configura uma restrição à generalização dos resultados obtidos.

### **5.4.3 Validação de Construção do Estudo**

A validação de construção do estudo pondera os aspectos que podem prejudicar a interpretação dos relacionamentos estatísticos ou a correlação dos aspectos analisados. Neste estudo de viabilidade, a abordagem foi analisada qualitativamente, não utilizando testes estatísticos. Um aspecto importante a ser considerado é de que os participantes envolvidos na execução não eram familiarizados com a Teoria das Restrições, eximindo assim o vício que beneficiasse a execução com resultados positivos por conta da experiência dos executores.

### **5.4.4 Validação da Conclusão do Estudo**

A conclusão do estudo deve considerar quais aspectos podem prejudicar a análise e interpretação dos insumos coletados.

Os instrumentos utilizados para coleta dos dados foram os modelos dos formulários do processo, apresentados no Anexo IV, além dos questionários de avaliação aplicados, apresentados no Anexo V. Embora não houvesse uma medição de dados de maneira quantitativa, o estudo considerou o tempo de execução e a percepção dos executores de maneira estruturada.

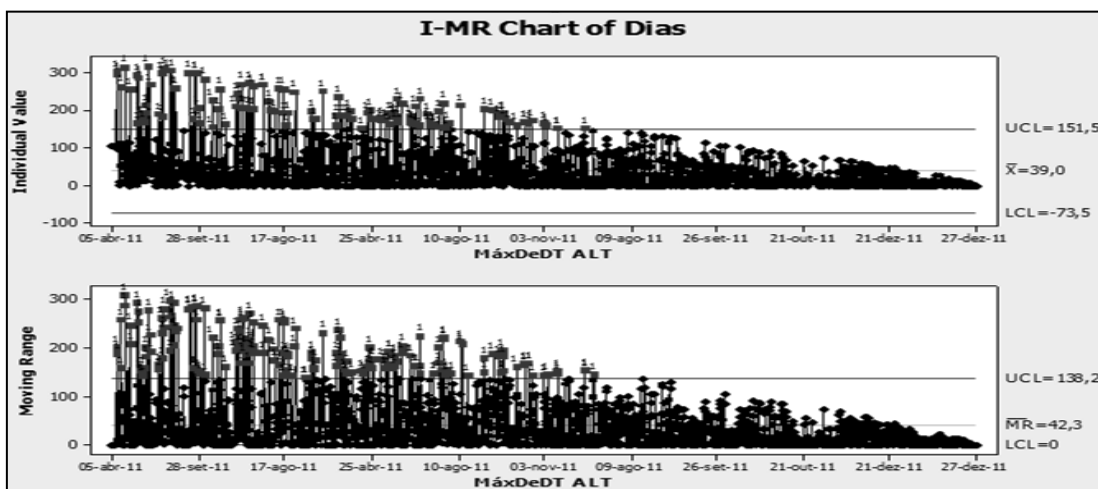
## **5.5 Execução do Estudo**

A execução do estudo de viabilidade aconteceu em etapas, como representado na Tabela 5.1. No primeiro momento, a coleta de dados permitiu uma análise prévia da situação do processo alvo da melhoria e um treinamento para o RGP que ficaria responsável pela melhoria no processo da organização. Após uma análise preliminar ao estudo de viabilidade, considerando os dados advindos do repositório de medição da organização, foi observado, através de gráficos de controle (SHEWHART, 1939), que o processo apresentava melhoria no seu desempenho a partir de setembro de 2011. Isto ficou evidenciado pela diminuição da média dos dias necessários para abertura e fechamento para atendimentos dos chamados. O gráfico de controle que demonstra a melhoria do processo ao longo do tempo em termos de tempo médio de abertura e resolução dos chamados é apresentado na Figura 5.1. O gráfico evidencia a diminuição na variação do desempenho do processo ao longo do tempo. A partir desta análise, foram identificadas pessoas da equipe que poderiam participar do estudo de viabilidade.

**Tabela 5.1** - Etapas da execução do estudo de viabilidade

Atividade	Participantes	Data
<b>Coleta de Dados e Treinamento</b>		
Seleção do processo para melhoria e coleta de dados para análise	Responsável do Grupo de Processo Pesquisador	27/12/2011
Análise dos dados do processo	Pesquisador	09/02/2012
Apresentação (introdução ao estudo de viabilidade e à abordagem proposta)	Responsável do Grupo de Processo Pesquisador	09/02/2012
<b>Execução do Processo da Abordagem Proposta</b>		
Atividades: “Definição do objetivo de melhoria” e “Preparação para Análise”	Responsável do Grupo de Processo	06/03/2012
Atividades: “Identificação da restrição” e “Elaboração da proposta de melhoria”	Responsável do Grupo de Processo Membro da Equipe Pesquisador	15/03/2012

Foram selecionados dois participantes para o estudo de viabilidade. Um participante realizou o papel de Representante do Grupo de Processo e de Membro de Equipe. Esta sobreposição de papéis aconteceu por conta da baixa representatividade dos membros da equipe e, também, porque este representante fazia parte do Grupo de Processo e conhecia profundamente o processo e os problemas relacionados à sua execução. O segundo participante era parte da equipe executora do processo e era especialista de negócio sênior.



**Figura 5.1** - Gráfico de controle do número de dias para abertura e resolução dos chamados do processo

De acordo com o planejamento, o treinamento inicial dado ao RGP, em momento diferente da execução do processo da abordagem, permitiu que o representante maturasse o conhecimento sobre melhoria de processo e sobre os conceitos da Teoria das Restrições utilizados. O objetivo de melhoria definido pela alta direção da empresa foi: “Aumentar a eficiência do processo de Atendimento de Chamado”. A partir desta definição, na segunda etapa da execução do estudo de viabilidade, o RGP executou as atividades de “Definição dos objetivos de melhoria” e “Preparação para análise” sem o apoio do pesquisador, cronometrando o tempo gasto na realização das tarefas. Os tempos de execução destas tarefas realizadas pelo RGP são apresentados na Tabela 5.2. Não houve dúvidas na execução dessas tarefas por parte do RGP. O produto resultante das atividades “Definição do objetivo de melhoria” e “Preparação para Análise” são, respectivamente, o Formulário de Objetivo de Melhoria e o Formulário de Análise de Restrições. Ambos os formulários podem ser encontrados na seção IV.1 e na seção IV.2 do Anexo IV.

**Tabela 5.2** - Tempo de execução das tarefas da primeira etapa do estudo de viabilidade

<b>Atividade</b>	<b>Tarefa</b>	<b>Tempo de execução</b>
Definição do objetivo de melhoria	Descrever objetivo de melhoria	30 minutos
	Avaliar rastreabilidade do objetivo de melhoria aos objetivos de negócio	Não se aplica, pois a organização não possui mapeamento entre os objetivos de melhoria de processo e objetivos de negócio.
Preparação para análise	Planejar melhoria	30 minutos
	Selecionar diagrama causal candidato	Não se aplica, pois não há repositório com diagramas causais existentes na organização.
	Coletar informações organizacionais de contexto	120 minutos
	Coletar informações de contexto de projetos	180 minutos
	Identificar fatores pertinentes	76 minutos

Das tarefas listadas na Tabela 5.2, a tarefa “Avaliar rastreabilidade do objetivo de melhoria aos objetivos de negócio” não foi executada porque a organização não possui um mapeamento entre os objetivos de negócio e os objetivos de melhoria de

processo<sup>13</sup>. A tarefa “Selecionar diagrama causal candidato” também não foi executada pelo motivo de a organização não possuir um repositório de diagramas causais para reutilização.

O tempo de execução da atividade “Identificação da restrição” é apresentada na Tabela 5.3 e ocorreu da seguinte forma:

1. Com base na descrição do processo da abordagem proposta e do procedimento “Construir diagrama causal” – descrito na seção III.B do Anexo III –, o Membro da Equipe I construiu o diagrama causal. O pesquisador em diversos momentos auxiliou no entendimento do procedimento.
2. Após a construção do diagrama causal, o Membro da Equipe II foi chamado para avaliar o diagrama causal. Neste momento, uma breve apresentação sobre o objetivo do estudo de viabilidade foi feita, seguida de uma explicação breve sobre o diagrama causal e o objetivo da tarefa “Avaliar diagrama causal”. O Membro da Equipe II, como previsto pelo processo, avaliou o diagrama e sugeriu alterações. Estas alterações compreenderam novos aspectos do processo que não foram atentados pelo Membro da Equipe I. O diagrama causal resultante foi inserido no Formulário de Análise de Restrições, como apresentado na seção V.G do Anexo V.
3. Após a avaliação do diagrama causal, com base na descrição do processo da abordagem, o RGP executou a tarefa “Identificar restrição principal”. O resultado desta tarefa, a Matriz de Rastreabilidade das Restrições Candidatas, foi registrado no Formulário de Análise de Restrições, como apresentado na seção V.G do Anexo V, assim como a restrição principal e o relato do raciocínio utilizado para decisão.

---

<sup>13</sup> Esta necessidade de manter o mapeamento entre os objetivos de melhoria dos processos e os objetivos estratégicos é típica de organizações de alta maturidade que precisam demonstrar que as melhorias nos processos apoiam a organização a atingir seus objetivos de negócio e objetivos estratégicos.

**Tabela 5.3** - Tempo de execução das tarefas da segunda etapa do estudo de viabilidade

<b>Atividade</b>	<b>Tarefa</b>	<b>Tempo de execução</b>
Identificação da restrição	Desenvolver diagrama causal	70 minutos
	Avaliar diagrama causal	40 minutos
	Identificar restrição principal	20 minutos
Elaboração da proposta de melhoria	Elaborar proposta de solução	60 minutos
	Avaliar viabilidade da proposta de solução	30 minutos
	Elaborar proposta de melhoria	60 minutos
	Elaborar planejamento de implementação da proposta de melhoria	Não se aplica, pois a implementação da melhoria não seria realizada naquele momento.

A aplicação dos procedimentos da Teoria das Restrições – Árvore da Realidade Atual – permitiu identificar a restrição principal de maneira sistemática. Neste estudo, a restrição principal identificada foi a “Falta de planejamento da alta direção”.

A organização sofreu em 2011 um crescimento muito acelerado e que impactou no nível de qualidade e tempo de entrega dos seus serviços e produtos. No entanto, embora identificada como restrição principal, a causa-raiz da restrição, que é a falta de planejamento estratégico organizacional, não poderia ser tratada a fim de que os seus efeitos atuais fossem dirimidos em tempo hábil. A solução de longo prazo para elaborar o planejamento estratégico já estava em andamento na empresa. Desta forma, o RGP descartou esta restrição principal, justificando o raciocínio utilizado no formulário, como pode ser visto na seção V.G do Anexo V.

A segunda restrição de maior relevância identificada pelo processo foi a “Falta de conhecimento em ferramenta de modelagem”. A solução proposta pelo RGP para esta restrição foi adotar e institucionalizar uma ferramenta para modelagem. Não havia nenhum obstáculo ou conflito que se opusesse ou impedisse a implementação da solução proposta. Por esta razão, a tarefa “Elaborar proposta de solução”, que incorpora o procedimento da Teoria das Restrições relacionado à resolução de conflito através do Diagrama de Resolução de Conflito, foi executada sem a necessidade de utilizar o DRC, como prevista no processo.



Após a definição da proposta de solução, o Membro da Equipe I executou a tarefa “Avaliar viabilidade da proposta de solução”. Nesta tarefa, o procedimento referente à Teoria das Restrições que verifica a viabilidade de implementação da solução, resultou no diagrama da Árvore da Realidade Futura, nos efeitos de influência desejáveis, efeitos desejáveis e riscos. A partir disto, o RGP elaborou o cronograma de marcos, preenchendo o Formulário de Análise de Restrições com as informações resultantes desta tarefa, conforme apresentado na seção V.G do Anexo V. Os tempos de realização das tarefas da atividade “Elaboração da proposta de melhoria” estão apresentados na Tabela 5.3.

Para finalizar o estudo de viabilidade, foram aplicados ao Membro da Equipe e ao Responsável do Grupo de Processo os formulários de avaliação que tinham como propósito coletar as percepções sobre o estudo de viabilidade e a abordagem proposta. O preenchimento durou 10 minutos para cada participante. Todos os formulários preenchidos estão apresentados no Anexo V.

## **5.6 Análise e Interpretação dos Resultados**

A partir da medição de tempo de execução do processo, dos formulários de avaliação do estudo de viabilidade e das anotações feitas pelo pesquisador ao longo da execução do estudo, foi possível avaliar a abordagem proposta do ponto de vista de profissionais da área de melhoria de processo de software e de executores de processo de software na indústria. Com isto, as questões de pesquisa relacionadas na Seção 5.4 puderam ser respondidas.

No tocante à questão “Q1: Qual o grau de satisfação do responsável pela melhoria quanto à facilidade de utilização da abordagem?”, a percepção dos participantes do estudo de viabilidade aponta uma dificuldade no entendimento dos procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições. O grau de facilidade de aprender o processo sinalizou uma dificuldade de entender alguns passos do processo. Segundo o RGP, embora os conceitos da Teoria das Restrições sejam simples, a execução dos procedimentos dos Processos de Raciocínio não é trivial. Isto justifica a necessidade de uma tutoria durante a execução do estudo. Os modelos de formulário também contribuíram para a execução do processo da abordagem.

Em relação à questão “Q2: Qual o grau de satisfação do responsável pela melhoria quanto ao tempo de utilização da abordagem?”, o tempo total de execução do processo da abordagem foi de 10 horas e 36 minutos. De acordo com o RGP, para um processo que considera desde a definição de objetivos de melhoria até a definição da estratégia de melhoria por meio da proposta de solução, o tempo consumido na execução da abordagem é razoável para a realidade da empresa. No entanto, foi observado pelo RGP que a sistemática da abordagem é muito exaustiva para identificar oportunidades de melhoria (restrições do processo), tendo em vista que há outras fontes de identificação de oportunidade de melhoria na organização menos custosas.

Sobre a questão de pesquisa “Q3: Qual o grau de satisfação do responsável pela melhoria quanto ao resultado obtido com a aplicação da abordagem?”, o RGP argumentou que, embora houvesse suspeitas sobre a restrição principal do processo, não havia nenhum consenso ou constatação a respeito. A abordagem permitiu identificar a restrição principal de forma sistemática, corroborando com a suspeita dos executores do processo que apontaram para a falta de planejamento da organização e para a falta da institucionalização de uma ferramenta para modelagem de casos de uso e de requisitos.

Ao analisar as questões supracitadas, as seguintes considerações são pertinentes:

- i) Embora o processo alvo da melhoria esteja continuamente sendo melhorado, ele ainda não é um processo maduro. Desta forma, há problemas na execução do processo mais evidentes e que devem ser solucionados. Para processos maduros, em organizações de alta maturidade, o cenário de melhoria de processos de software que estão sob controle estatístico tem um caráter mais minucioso e complexo, justificando a utilização de uma abordagem mais robusta e sistemática para identificar a oportunidade de melhoria e para propor uma solução para melhorar o desempenho do processo;
- ii) A elucidação dos fatores de influência do processo, através da aplicação dos procedimentos baseados nos Processos de Raciocínio, de acordo com o RGP, “permitiu avaliar o problema de forma mais clara”. Segundo o participante, apreciar o diagrama causal da realidade atual do processo ajudou a perceber como os fatores de

influência afetavam o processo. Além disso, foi possível perceber o alcance dos efeitos da proposta de solução ao longo da cadeia de relacionamentos de causa-e-feito que foi impactada pela remoção da restrição principal. Desta forma, há indícios de que a aplicação da abordagem em processos de software sob controle estatístico consideraria os fatores de influência do processo de forma a minimizar possíveis efeitos adversos à estabilidade do desempenho do processo. Ademais, permitiria uma investigação sistemática para identificar a restrição principal do processo, considerando os fatores de influência originadas das causas comuns de variação.

- iii) A percepção sobre a facilidade de aplicação da abordagem, avaliada pelo RGP e membro da equipe, pontuou dificuldades em aprender o processo e em repeti-lo. De acordo com o RGP e o membro da equipe, o acompanhamento do pesquisador, apoiando com uma tutoria, foi fundamental para o sucesso da aplicação. Um ponto a ser melhorado, de acordo com a avaliação do RGP, é a realização de um treinamento sobre a abordagem e, principalmente, sobre os Processos de Raciocínio da Teoria das Restrições para o responsável pela melhoria.
- iv) Os modelos de formulário que apoiam a execução do processo da abordagem auxiliaram o RGP na execução do processo. As informações de instruções contidas nas seções auxiliaram na execução correta de algumas tarefas. Além disso, os formulários preenchidos serviram de artefatos e evidências de aderência aos processos organizacionais e padrões de qualidade da organização.

Com base nas respostas às questões de pesquisa definidas para este estudo de viabilidade e na análise resultante de observações e informações levantadas, é possível refutar a hipótese nula, definida na Seção 5.4, que estabelecia a não aplicabilidade da abordagem proposta à melhoria de processos de software. Conseqüentemente, é possível afirmar que há indícios de aplicabilidade da abordagem proposta para melhoria de processo de software utilizando a Teoria das Restrições. Além disso, de acordo com a resposta à questão Q2, há indicativos de que a aplicação da abordagem também seja viável em termos de tempo de execução.

Ao considerar as ameaças à validade do estudo, descritas na Seção 5.4, não é possível concluir de forma definitiva que a abordagem é viável ou que ela é aplicável a qualquer cenário de melhoria de processo de software. No entanto, este estudo contribuiu para averiguar a aplicabilidade da abordagem em uma situação específica na indústria e para identificar melhorias para a própria abordagem. Outros estudos experimentais se fazem necessários para avaliar a abordagem em cenários mais diversos, como por exemplo, em organizações de alta maturidade com foco em processos sob controle estatístico e críticos para a organização, pois permitiria um melhor aproveitamento da sistemática proposta na abordagem.

## **5.7 Considerações Finais**

O estudo experimental executado com o intuito de avaliar a abordagem proposta nesta dissertação foi descrito neste capítulo, de acordo com a metodologia apresentada na Seção 1.3 do Capítulo 1. O estudo de viabilidade tinha o objetivo de verificar se a abordagem era aplicável à melhoria de processos de software e se era viável em termos de tempo de execução.

As etapas de definição e planejamento do estudo foram descritas na Seção 5.4, seguidas da Seção 5.5 de execução do estudo e da Seção 5.6 de análise e interpretação dos resultados. O estudo foi conduzido em uma empresa de software considerando um processo crítico para a organização.

Este estudo permitiu verificar indícios de que a Teoria das Restrições é aplicável e viável para melhoria de processos de software na indústria. Os Processos de Raciocínio aplicados à investigação de oportunidades de melhoria no processo e à proposição de soluções para melhorar o desempenho do processo apresentaram indícios de que são adequados para a melhoria de processo de software, pois mapeiam a situação atual e os fatores de influência do processo a ser melhorado, auxiliam na definição de uma proposta de solução para melhorar a situação atual do processo e apoiam na definição dos passos necessários para implementar a proposta de solução e avaliar a sua viabilidade.

No próximo capítulo, as conclusões do trabalho, contribuições da dissertação, suas limitações e perspectivas futuras são apresentadas.

## Capítulo 6 – Conclusão

*Este capítulo apresenta as considerações finais e conclusões desta dissertação, descrevendo suas limitações, contribuições e oportunidades futuras de pesquisa e desenvolvimento.*

### 6.1 Considerações finais

Esta dissertação apresentou uma pesquisa sobre melhoria em processos de software. Esta pesquisa iniciou-se com um estudo baseado em revisão sistemática que permitiu identificar a oportunidade de pesquisa relacionada à melhoria contínua de processos de software sob controle estatístico. A partir dos resultados deste estudo, foram identificadas abordagens para melhoria contínua de processos de software. No entanto, nenhuma abordagem encontrada apoiava a investigação de oportunidades de melhoria em processos de software sob controle estatístico através da análise das causas de variação do processo de maneira sistemática.

Após o estudo, uma revisão da literatura focou na identificação de abordagens de melhoria de processos semelhantes às abordagens já aplicadas na engenharia de software, tais como Controle Estatístico de Processos e Seis Sigma. Diversas abordagens de TQM foram identificadas, dentre as quais se destacou a Teoria das Restrições. A TDR possui ferramentas de investigação das causas que restringem o desempenho do processo e de apoio à resolução de conflito e elaboração de solução para as restrições identificadas. Além disso, embora não haja ainda aplicação da TDR para melhoria de processos de software, indícios baseados em pesquisas mostram que há aplicabilidade em diversas áreas do conhecimento. Desta forma, a abordagem descrita neste trabalho utilizou os conceitos da Teoria das Restrições para identificar, de maneira sistemática, as causas de variação do processo de software, a fim de identificar a restrição principal ao desempenho do processo e a elaborar uma proposta de solução e melhoria que maximizasse suas possibilidades de sucesso.

Com o intuito de fundamentar a abordagem descrita, a revisão da literatura compreendeu, no Capítulo 2, questões relacionadas à melhoria de processo de software, definindo conceitos relacionados às peculiaridades do processo de software e as abordagens largamente utilizadas na indústria para a melhoria de processo, tais

como os modelos CMMI-DEV (SEI, 2010) e MR-MPS (SOFTEX, 2011) e demais abordagens identificadas no estudo baseado em revisão sistemática. Ademais, a revisão da literatura também abrangeu um levantamento de abordagens voltadas para a melhoria contínua de processo e os conceitos e fundamentos que justificam a aplicação da Teoria das Restrições na melhoria de processos de software.

Desta forma, o processo da abordagem foi definido com base nos conceitos de melhoria de processos de software, no processo de melhoria definido por ALBUQUERQUE (2008), e nos conceitos da Teoria das Restrições através dos seus Processos de Raciocínio. Juntamente com o processo, os modelos dos formulários que apoiam a execução do processo foram elaborados, tendo em vista a realidade de processos comumente encontrada em organizações que possuem uma cultura de melhoria de processos ou que implementem os modelos de melhoria de processo de software.

Um estudo de viabilidade foi conduzido com o intuito de avaliar a aplicabilidade e o tempo de execução da abordagem. O estudo foi executado em uma empresa de software, considerando uma situação real de melhoria de processo em um processo crítico para a organização. O resultado do estudo mostrou indícios de aplicabilidade e viabilidade da abordagem, e apontou oportunidades de melhoria no processo da abordagem. Além do resultado, a análise sob as informações coletadas durante a execução do processo sugere que a abordagem, embora possa ser aplicada em melhoria de processos de software em organizações de qualquer nível de maturidade, em processos estabilizados ou não, seria melhor aproveitada e mais viável para organizações de alta maturidade ou em processos sob controle estatísticos, devido ao tempo necessário para a execução da abordagem e devido à dificuldade inerente nestas situações que exigem maior rigor.

## **6.2 Limitações**

Embora a abordagem tenha demonstrado indícios de aplicabilidade e viabilidade, há indicativos (com base nas sugestões dos participantes do estudo de viabilidade e da análise do pesquisador) que sugerem que sua aplicação teria um custo-benefício maior em organizações de alta maturidade e em processos sob controle estatístico. Isto acontece, pois, organizações em níveis iniciais de maturidade estão melhorando seus processos em um cenário onde há muitos problemas e oportunidades

de melhoria mais simples e de resolução mais urgentes e de implementação mais direta. No entanto, dependendo do julgamento da organização, a aplicação da abordagem em um processo que não esteja sob controle estatístico, mesmo sendo custosa, é justificável por conta da importância da melhoria no desempenho do processo.

Outra limitação deste trabalho está relacionada ao estudo experimental. Por questões de disponibilidade de organizações de software de alta maturidade e por tempo hábil para execução do estudo, as seguintes considerações são importantes:

- A fim de verificar a aplicabilidade e viabilidade da abordagem nos diferentes cenários, é necessária a execução do processo em outros contextos, tais como, em organizações de alta maturidade, em processos sob controle estatístico e com maior número de participantes envolvidos no estudo;
- Com o objetivo de minimizar o tempo de aplicação da abordagem para não impactar nas atividades rotineiras dos participantes do estudo, somente parte do processo da abordagem foi executada. Atividades relacionadas à tarefa “Elaborar planejamento de implementação da proposta de melhoria”, “Registrar coprodutos do processo” e “Registrar lições aprendidas” não foram executadas. A elaboração do planejamento de implementação da proposta de melhoria não seria possível ser realizada, pois a organização não implantaria a melhoria naquele momento. As outras duas tarefas não foram executadas no momento do estudo. Contudo, os formulários utilizados na execução do estudo foram posteriormente registrados no repositório de ativos de processo da organização.

### **6.3 Contribuições**

As principais contribuições desta dissertação são:

- Apoio à identificação da restrição principal do desempenho do processo (causa-raiz);
- Apoio à resolução de conflitos relacionados à solução para melhoria de processos de software;

- Apoio à elaboração de proposta de solução para remover restrições no desempenho do processo;
- Apoio na identificação de riscos relacionados à melhoria do processo;
- Definição de um processo de melhoria de processo de software;
- Apoio à memória do progresso de melhoria dos processos organizacionais, por meio da Base de Melhoria e dos artefatos gerados ao longo do processo;
- Aplicação dos procedimentos dos Processos de Raciocínio à melhoria de processos de software com base nos conceitos da Teoria das Restrições;
- Apoio à melhoria de processos por meio de formulários que estruturam as informações e o raciocínio utilizado ao longo do processo de investigação e elaboração da solução e melhoria.

#### **6.4 Oportunidades Futuras de Pesquisa**

Com base nos resultados do estudo de viabilidade, pode-se citar como oportunidade de melhoria para a abordagem:

- Juntamente com os procedimentos de construção dos diagramas, adicionar um tutorial sobre a ferramenta adotada;
- Acrescentar como requisito à abordagem, um nível básico de conhecimento sobre a Teoria das Restrições. Além disso, prover um treinamento inicial por meio de tutorial explicativo, livro ou vídeo-aula;
- Elaborar e fornecer um exemplo completo da aplicação da abordagem e apresentação contendo a construção passo a passo dos diagramas dos Processos de Raciocínio;
- Desenvolver uma ferramenta para apoiar o processo, tanto no que diz respeito aos formulários preenchidos, aos diagramas construídos e à recuperação de informações;
- Refinar o processo da abordagem por meio de estudos experimentais.



A evolução da abordagem descrita neste trabalho não se limita apenas a melhorias no processo com sugestões e lições aprendidas. O interesse crescente sobre a Teoria das Restrições e as evoluções dos Processos de Raciocínio possibilitam que a abordagem evolua da mesma forma.

Embora o foco deste trabalho seja melhoria contínua de processos de software por meio de melhorias incrementais, há combinações envolvendo os Processos de Raciocínio com outras abordagens que podem expandir tais fronteiras. Um exemplo desta combinação é a fusão dos conceitos do Diagrama de Resolução de Conflitos com TRIZ para resolução de problemas complexos (DOMB e DETTMER, 1999; PFEIFER *et al.*, 2003; STRATTON e MANN, 2003; YONG, 2010). Ideias de soluções inovadoras são apontadas como o principal resultado desta combinação. De acordo com DOM e DETTMER (1999), a sinergia entre as duas abordagens ocorre, pois o DRC estrutura graficamente os elementos críticos de qualquer conflito, ao passo que TRIZ provê uma abordagem sistemática e estruturada para estimular a geração de ideias inovadoras.

## REFERÊNCIAS BIBLIOGRÁFICAS

ALBUQUERQUE, A.B., 2008, *Avaliação e Melhoria de Ativos de Processos Organizacionais em Ambientes de Desenvolvimento de Software*, Universidade Federal do Rio de Janeiro, Rio de Janeiro.

ANDRADE, J.M.S., 2005, *Avaliação de Processos de Software em Ambientes de Desenvolvimento de Software Orientados à Organização*, Universidade Federal do Rio de Janeiro, Rio de Janeiro.

ANTONY, J., 2002, "Design for Six Sigma: a breakthrough business improvement strategy for achieving competitive advantage", *Work Study*, v. 51, n. 1, pp. 6-8.

ANTONY, J., 2004, "Some pros and cons of six sigma: an academic perspective", *The TQM Magazine*, v. 16, n. 4, pp. 303-306.

BALDASSARRE, M., CAIVANO, D., KITCHENHAM, B., *et al.*, 2007, "Systematic review of statistical process control: An experience report".

BALDASSARRE, T., BOFFOLI, N., CAIVANO, D., *et al.*, 2004, "Managing Software Process Improvement (SPI) through Statistical Process Control (SPC)", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.

BALDERSTONE, S., MABIN, V., 1998, "A Review of Goldratt's Theory of Constraints (TOC)—lessons from the international literature".

BASILI, V., 1993, "The Experience Factory and its relationship to other improvement paradigms", *Software Engineering—ESEC'93*, pp. 68-83.

BASILI, V.R., ROMBACH, H.D., 1988, "The TAME project: Towards improvement-oriented software environments", *Software Engineering, IEEE Transactions on*, v. 14, n. 6, pp. 758-773.

BIOLCHINI, J., MIAN, P.G., NATALI, A.C.C., *et al.*, 2005, *Systematic review in software engineering*, Programa de Engenharia de Sistema e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.

- BLACK, K., REVERE, L., 2006, "Six Sigma arises from the ashes of TQM with a twist", *International Journal of Health Care Quality Assurance*, v. 19, n. 3, pp. 259-266.
- BOFFOLI, N., 2006, "Non-intrusive monitoring of software quality". In: *10th European Conference on Software Maintenance and Reengineering, CSMR 2006, March 22, 2006 - March 24, 2006*, pp. 319-322, Bari, Italy.
- BOFFOLI, N., BRUNO, G., CAIVANO, D., *et al.*, 2008, "Statistical process control for software: A systematic approach". In: *2nd International Symposium on Empirical Software Engineering and Measurement, ESEM 2008, October 9, 2008 - October 10, 2008*, pp. 327-329, Kaiserslautern, Germany.
- BOYD, L.H., GUPTA, M.C., 2004, "Constraints management: what is the theory?", *International Journal of Operations & Production Management*, v. 24, n. 4, pp. 350-371.
- BROWN, A., EATOCK, J., DIXON, D., *et al.*, 2008, "Quality and continuous improvement in medical device manufacturing", *The TQM Journal*, v. 20, n. 6, pp. 541-555.
- CAIVANO, D., 2005a, "Continuous software process improvement through statistical process control". In: *Ninth European Conference on Software Maintenance and Reengineering, CSMR 2005, March 21, 2005 - March 23, 2005*, pp. 288-293, Manchester, United kingdom.
- CAIVANO, D., 2005b, "Continuous software process improvement through statistical process control", pp. 288-293.
- CAMPOS, F.B., 2008a, *Avaliação e Melhoria de Ativos de Processos de Software de Instituições Implementadoras*, COPPE, UFRJ, Rio de Janeiro.
- CAMPOS, F.B., 2008b, *Avaliação e Melhoria de Ativos de Processos de Software de Instituições Implementadoras*, Universidade Federal do Rio de Janeiro.
- CARD, D., 1994, "Statistical process control for software?", *Software, IEEE*, v. 11, n. 3, pp. 95-97.

- CARD, D., 2004, "Statistical techniques for software engineering practice", pp. 722-723.
- CHANG, C.P., CHU, C.P., 2008, "Improvement of causal analysis using multivariate statistical process control", *Software Quality Journal*, v. 16, n. 3, pp. 377-409.
- CHOE, K., HERMAN, S., 2004, "Using Theory of Constraints Tools to Manage Organizational Change: A Case Study of Euripa Labs", *International Journal of Management and Organizational Behavior*, v. 8, n. 6, pp. 540-558.
- COCKBURN, A., 2007, "What engineering has in common with manufacturing and why it matters", *Crosstalk*, pp. 4-7.
- CONTE, T.U., 2009, *Técnica de Inspeção de Usabilidade baseada em Perspectivas de Projeto Web*, COPPE, UFRJ, Rio de Janeiro.
- DAVIES, J., MABIN, V., BALDERSTONE, S., 2005, "The theory of constraints: a methodology apart?--a comparison with selected OR/MS methodologies", *Omega*, v. 33, n. 6, pp. 506-524.
- DAVIES, J., MABIN, V., COX, J., 2004, "The theory of constraints and systems dynamics: A suitable case for multimethodology", pp. 25–29.
- DEMING, E., 1992, *The New Economics, The Theory of Profound Knowledge*.
- DEMING, W.E., 1986, "Out of the crisis", *MIT Center for Advanced Engineering Study*. Cambridge, Mass.
- DETTMER, H.W., 1995a, "Conflict resolution: the theory of constraints approach". In: *Constraints Management Symposium Proceedings, APICS*, v. April, Phoenix, AZ
- DETTMER, H.W., 1995b, "Quality and the Theory of Constraints", *Quality Progress*, v. 28, n. 4, pp. 77-81.
- DETTMER, H.W., 1997, *Goldratt's theory of constraints: a systems approach to continuous improvement*, First Edition ed., Wisconsin, American Society for Quality Press.

- DETTMER, H.W., 1998a, *Breaking the constraints to world-class performance*, American Society for Quality Press.
- DETTMER, H.W., 1998b, "Constraint Theory: A Logic-Based Approach to System Improvement".
- DOMB, E., DETTMER, H.W., 1999, "Breakthrough innovation in conflict resolution: Marrying TRIZ and the Thinking Process", *Proceedings of APICS Constraint Management Special Interest Group*.
- DYBÅ, T., 2002, "Enabling Software Process Improvement: An Investigation of the Importance of Organizational Issues", *Empirical Software Engineering*, v. 7, n. 4, pp. 387-390.
- ECKES, G., 2001, *A Revolução Seis Sigma: o método que levou a GE e outras empresas a transformar processos em lucros*, Sexta edição ed., Rio de Janeiro, Elsevier.
- EHIE, I., SHEU, C., 2005, "Integrating six sigma and theory of constraints for continuous improvement: a case study", *Journal of Manufacturing Technology Management*, v. 16, n. 5, pp. 542-553.
- FLORAC, W., CARLETON, A., 1999, *Measuring the software process: statistical process control for software process improvement*, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.
- FUGGETTA, A., 2000, "Software process: a roadmap", pp. 25-34, Limerick, Ireland.
- GALINAC, T., 2009, "Empirical evaluation of selected best practices in implementation of software process improvement", *Information and Software Technology*, v. 51, pp. 1351-1364.
- GALINAC, T., CAR, Z., 2007, "Software verification process improvement proposal using six sigma". In: *8th International Conference on Product-Focused Software Process Improvement, PROFES 2007, July 2, 2007 - July 4, 2007*, v. 4589 LNCS, pp. 51-64, Riga, Latvia.

- GARDINER, J., MONTGOMERY, D., 1987, "Using statistical control charts for software quality control".
- GOLDRATT, E.M., 1990, *What Is This Thing Called the Theory of Constraints and How Should It Be Implemented?*, Croton-on-Hudson, NY, North River Press.
- GOLDRATT, E.M., COX, J., 1984, *The goal: an ongoing improvement process*.
- GOLDRATT, E.M., COX, J., 1992, *The Goal - A Process of Ongoing Improvement*, NY, North River Press.
- GONÇALVES, F.M.G.S., BEZERRA, C.I.M., BELCHIOR, A.D., *et al.*, 2008, "A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach", pp. 104-111.
- GUPTA, M.C., BOYD, L.H., 2008, "Theory of constraints: a theory for operations management", *International Journal of Operations & Production Management*, v. 28, n. 10, pp. 991-1012.
- HAHN, G.J., HILL, W.J., HOERL, R.W., *et al.*, 1999, "The Impact of Six Sigma Improvement-A Glimpse into the Future of Statistics", *The American Statistician*, v. 53, n. 3, pp. 208-215.
- HONG, G.Y., GOH, T.N., 2003, "Six Sigma in software quality", *TQM Magazine*, v. 15, n. 6, pp. 364-373.
- HUMPHREY, W.S., 1989, *managing the Software Process*, Boston, Addison-Wesley Professional.
- HUMPHREY, W.S., 1997, *Introduction to the personal software process*, Addison-Wesley.
- ISO/IEC, 2000, "ISO 9000:2000", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- ISO/IEC, 2003, "ISO/IEC 15504: Information Technology - Software Process Assessment", *The International Organization for the Standardization and the International Electrotechnical Commission*, v. 15504.

- ISO/IEC, 2008, "ISO/IEC 12207: System and Software Engineering - Software Life Cycle Processes", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- JALOTE, P., SAXENA, A., 2002, "Optimum control limits for employing statistical process control in software process", *IEEE Transactions on Software Engineering*, pp. 1126-1134.
- JURISTO, N., MORENO, A.M., 2001, *Basics of software engineering experimentation*, Springer Netherlands.
- KALINOWSKI, M., 2011, *Uma Abordagem para Prevenção de Defeitos Provenientes de Inspeções para Apoiar a Melhoria dos Processos de Engenharia do Software*, D.Sc., UFRJ/COPPE/PESC - Programa de Engenharia de Sistema e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- KALINOWSKI, M., TRAVASSOS, G.H., CARD, D.N., 2008, "Towards a defect prevention based process improvement approach". In: *EUROMICRO 2008 - Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2008, September 3, 2008 - September 5, 2008*, pp. 199-206, Parma, Italy.
- KAN, S.H., 1994, *Metrics and Models in Software Quality Engineering*, Addison-Wesley Longman Publishing Co., Inc.
- KHAN, A.R., ZHANG, L., 2008, *Applying Six Sigma in Software Companies for Process Improvement*, School of Engineering, Blekinge Institute of Technology, Ronneby, Sweden.
- KIM, S., MABIN, V., DAVIES, J., 2008, "The theory of constraints thinking processes: retrospect and prospect", *INTERNATIONAL JOURNAL OF OPERATIONS AND PRODUCTION MANAGEMENT*, v. 28, n. 2, pp. 155.
- KITCHENHAM, B., 2004, *Procedures for performing systematic reviews*, Keele University, Keele, UK.

- KITCHENHAM, B., PICKARD, L., PFLEEGER, S.L., 1995, "Case Studies for Method and Tool Evaluation", *IEEE software*, v. 12, n. 4 (July), pp. 52-62.
- KOMURO, M., 2006, "Experiences of applying SPC techniques to software development processes", pp. 577-584.
- LACERDA, D.P., CASSEL, R.A., RODRIGUES, L.H., 2010, "Service process analysis using process engineering and the theory of constraints thinking process", *Business Process Management Journal*, v. 16, n. 2, pp. 264-281.
- LACERDA, D.P., KOETZ, A.L., KLIPPEL, M., *et al.*, 2004, "Critical Issues about the Theory Of Constraints Thinking Process".
- LANTZY, M., 1992, "Application of Statistical Process Control to Software Processes, WADAS'92", pp. 113-123.
- LEPORE, D., COHEN, O., 1999, *Deming and Goldratt: The Theory of Constraints and the System of Profound Knowledge*, Great Barrington, MA, North River Press.
- LEWIS, N.D.C., 1999, "Continuous process improvement using Bayesian Belief Networks", *Computers and Industrial Engineering*, v. 37, n. 1, pp. 449-452.
- LINDVALL, M., RUS, I., 2000, "Process diversity in software development", *IEEE software*, v. 17, n. 4, pp. 14-18.
- LIU, X.F., 2000, "Software quality function deployment", *Potentials, IEEE*, v. 19, n. 5, pp. 14-16.
- MABIN, V., 1999, "Goldratt's "Theory of Constraints" Thinking Processes: A Systems Methodology linking Soft with Hard".
- MABIN, V., BALDERSTONE, S., 2003, "The performance of the theory of constraints methodology", *INTERNATIONAL JOURNAL OF OPERATIONS AND PRODUCTION MANAGEMENT*, v. 23, n. 6, pp. 568-595.
- MABIN, V.J., FORGESON, S., GREEN, L., 2001, "Harnessing resistance: using the theory of constraints to assist change management", *Journal of European Industrial Training*, v. 25, n. 2/3/4, pp. 168-191.



- MAFRA, S., BARCELOS, R., TRAVASSOS, G., 2006, "Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software", v. 1, pp. 239–254.
- MAFRA, S., TRAVASSOS, G., 2006, "Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software", *Relatório Técnico ES-687/06, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, Brasil.*
- MONTONI, M., 2010, *Uma Investigação sobre os Fatores Críticos de Sucesso em Iniciativas de Melhoria de Processos de Software*, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- MURUGAPPAN, M., KEENI, G., 2003, "Blending CMM and Six Sigma to meet business goals", *IEEE software*, pp. 42-48.
- NAVE, D., 2002, "How to Compare Six Sigma, Lean and the Theory of Constraints", *Quality Progress*, v. 35, n. 3, pp. 73.
- NIAZI, M., WILSON, D., ZOWGHI, D., 2006, "Implementing Software Process Improvement Initiatives: An Empirical Study", *Springer*, v. 2006, pp. 222-233, LNCS.
- PARK, Y., CHOI, H., BAIK, J., 2007, "A framework for the use of Six Sigma tools in PSP/TSP". In: *SERA 2007: Fifth ACIS International Conference on Software Engineering Research, Management, and Applications, August 20, 2007 - August 22, 2007*, pp. 807-814, Busan, Korea, Republic of.
- PAULK, M., CURTIS, B., CHRISSIS, M., *et al.*, 1993, "Capability maturity model, version 1.1", *IEEE software*, v. 10, n. 4, pp. 18-27.
- PENTTI, H., ATTE, H., 2002, "Failure mode and effects analysis of software-based automation systems".
- PFEIFER, T., TILLMANN, M., AACHEN, G., 2003, "Innovative process chain optimization-Utilizing the tools of TRIZ and TOC for manufacturing". In: *ETRIA World Conference - TRIZ Future.*

- PIRASTEH, R., FARAH, K., 2006, "Continuous Improvement Trio: Combining the theory of constraints, lean, and six sigma to form" TLS" is an innovative process improvement methodology with real-world results", *Apics - The Performance Advantage*, v. 16, n. 5, pp. 31.
- PIRASTEH, R.M., FOX, R.E., 2010, *Profitability with No Boundaries: Optimizing Toc and Lean-Six Sigma*, ASQ Quality Press.
- RADICE, R., 1998, "Statistical Process Control for Software Projects".
- RAHMAN, S., 1998, "Theory of constraints: a review of the philosophy and its applications", *International Journal of Operations & Production Management*, v. 18, n. 4, pp. 336-355.
- SANTOS, G., 2008, *Ambientes de Engenharia de Software Orientados a Corporação*, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- SANTOS, G., MONTONI, M., VASCONCELLOS, J., *et al.*, 2007, "Implementing software process improvement initiatives in small and medium-size enterprises in Brazil".
- SARGUT, K., DEMIRÖRS, O., 2006, "Utilization of statistical process control (SPC) in emergent software organizations: Pitfalls and suggestions", *Software Quality Journal*, v. 14, n. 2, pp. 135-157.
- SCHEINKOPF, L.J., 1999, *Thinking for a change: Putting the TOC thinking processes to use*, Boca Raton, FL, CRC Press.
- SCHOTS, N.C.L., 2010, *Uma Abordagem para a Identificação de Causas de Problemas Utilizando Grounded Theory*, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- SCHROEDER, R.G., LINDERMAN, K., LIEDTKE, C., *et al.*, 2008, "Six Sigma: Definition and underlying theory", *Journal of operations management*, v. 26, n. 4, pp. 536-554.
- SEI, 2010, "CMMI® for Development (CMMI-DEV), V1.3, CMU/SEI-2010", Software Engineering Institute. Disponível em: <http://www.sei.cmu.edu/>.

- SHELTON, M., 2007, *IT Improvement: A Survey and Analysis of Process Improvement Methodologies and Their Application in IT*, Management Information Systems, Bowie State University, Maryland.
- SHEN, X., TAN, K., XIE, M., 2000, "An integrated approach to innovative product development using Kano's model and QFD", *European Journal of Innovation Management*, v. 3, n. 2, pp. 91-99.
- SHEWHART, W.A., 1939, *Statistical Method: From the Viewpoint of Quality Control*, Lancaster Press Inc.
- SHULL, F., CARVER, J., TRAVASSOS, G., 2001, "An empirical methodology for introducing software processes", pp. 296.
- SILVA FILHO, R.C., 2006, *Uma Abordagem para Avaliação de Propostas de Melhoria em Processos de Software*, Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- SOFTEX, 2011, "MPS.BR - Melhoria de Processo do Software Brasileiro - Guia Geral", Disponível em: <http://www.softex.br/mpsbr>.
- SOLINGEN, R.V., BERGHOUT, E., 1999, *The Goal/Question/Metric Method: a practical guide for quality improvement of software development*, London, McGraw-Hill Publishing Company.
- STAPLETON, J.H., 2009, *Linear statistical models*, Hoboken, N.J., Wiley.
- STRATTON, R., MANN, D., 2003, "Systematic Innovation and the Underlying Principles Behind TRIZ and TOC", *Journal of Materials Processing Technology*, v. 139, n. 2003, pp. 120-126.
- TARHAN, A., DEMIRORS, O., 2006, "Investigating suitability of software process and metrics for statistical process control". In: *13th European Conference on Software Process Improvement, EuroSPI 2006, October 11, 2006 - October 13, 2006*, v. 4257 LNCS, pp. 88-99, Joensuu, Finland.
- TARHAN, A., DEMIRORS, O., 2008, "Assessment of software process and metrics to support quantitative understanding". In: *International Conference on Software*

*Process and Product Measurement, IWSM-Mensura 2007, November 5, 2007 - November 8, 2007*, v. 4895 LNCS, pp. 102-113, Palma de Mallorca, Spain.

TONINI, A.C., 2006, *A contribuição do Seis Sigma para a melhoria dos processos de software*, Ms.C., Engenharia de Produção, Universidade de São Paulo, São Paulo.

VAN SOLINGEN, R., BERGHOUT, E., 1999, *The Goal/Question/Metric Method: a practical guide for quality improvement of software development*, McGraw-Hill.

WANG, Q., JIANG, N., GOU, L., *et al.*, 2006, "BSR: A statistic-based approach for establishing and refining software process performance baseline". In: *28th International Conference on Software Engineering 2006, ICSE '06, May 20, 2006 - May 28, 2006*, v. 2006, pp. 585-594, Shanghai, China.

WATSON, K., BLACKSTONE, J., GARDINER, S., 2007, "The evolution of a management philosophy: the theory of constraints", *Journal of operations management*, v. 25, n. 2, pp. 387-402.

WELLER, E., 2002, "Practical applications of statistical process control [in software development projects]", *Software, IEEE*, v. 17, n. 3, pp. 48-55.

WHEELER, D.J., CHAMBERS, D.S., 1992, *Understanding statistical process control*, Second ed., Knoxville, TN, SPC press.

WÖHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M., REGNELL, B., WESSLÉN, A., 2000, *Experimentation in Software Engineering: An Introduction*, Norwell, USA, Kluwer Academic Publishers.

XIAOSONG, Z., ZHEN, H., FANGFANG, G., *et al.*, 2008, "Research on the Application of Six Sigma in Software Process Improvement". In: *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on*, pp. 937-940, 15-17 Aug. 2008.

YONG, C., 2010, *Integrating TOC and TRIZ for Service Process Improvement*,  
Department of Industrial and Systems Engineering, National University of  
Singapore, Singapore.

# **ANEXO I – ESTUDO BASEADO EM REVISÃO SISTEMÁTICA SOBRE MELHORIA CONTÍNUA EM PROCESSOS DE SOFTWARE**

*Neste anexo, é apresentado um estudo baseado em revisão sistemática, conduzido nesta dissertação, com o objetivo de identificar abordagens sobre melhoria contínua em processos de software sob controle estatístico. Os resultados do estudo também se encontram neste anexo.*

## **I.A. Introdução**

A revisão sistemática é um tipo de estudo secundário (KITCHENHAM, 2004), uma metodologia específica de pesquisa, desenvolvida com a finalidade de coletar e avaliar evidências disponíveis pertinentes a um determinado tópico (BIOLCHINI *et al.*, 2005). BIOLCHINI *et al.* (2005) citam também a importância da revisão sistemática para a condução de uma revisão da literatura confiável, abrangente e com valor científico. Ademais, KITCHENHAM (2004) pontua a importância da revisão sistemática para identificar ou confirmar oportunidades de pesquisas nas áreas investigadas.

O processo de condução da revisão sistemática segue uma seqüência de passos metodológicos muito bem definidos de acordo com um protocolo desenvolvido previamente, objetivando reduzir o viés da investigação da revisão da literatura informal. Este protocolo é construído com base em um tópico específico que representa o elemento central da investigação. Questões de pesquisa são pré-definidas, a partir deste elemento central, de maneira específica e estruturada, de tal forma que outros pesquisadores possam reproduzir ou mesmo avaliar o protocolo em relação à adequação dos padrões adotados no estudo (BIOLCHINI *et al.*, 2005).

Esta dissertação tem como objetivo definir uma abordagem para apoiar a melhoria contínua de processos estáveis. Apresentada no Capítulo 4, a abordagem baseia-se em um método estruturado de investigação para identificar oportunidades de melhoria em processos de software estáveis; e propor, implantar e monitorar soluções de melhoria para tais oportunidades. Os conceitos da Teoria das Restrições (*Teory of*

*Constraints*) implementados através dos métodos do Processo de Raciocínio (*Thinking Process*) foram aplicados para o desenvolvimento da abordagem e se propõem a identificar limitantes (oportunidades) que restrinjam o desempenho do sistema e desenvolver mudanças que precisam ser efetuadas no sistema para se atingir o desempenho desejado (melhoria).

Neste contexto, foi realizada esta revisão sistemática para identificar estudos com abordagens para melhoria contínua de processos de software sob controle estatístico. Com a realização desta revisão sistemática, procurou-se caracterizar o estado da arte relacionado à melhoria contínua de processos de software controlados estatisticamente. Além disso, as lacunas de informação ou abordagens que tratassem do assunto serviram como confirmação da oportunidade de pesquisa.

### **I.B. Processo de apoio à condução de estudos baseados em revisão sistemática**

O processo utilizado neste estudo baseado em revisão sistemática foi definido seguindo as orientações descritas em (KITCHENHAM, 2004) juntamente com o processo utilizado em (SANTOS, 2008; MONTONI, 2010; SCHOTS, 2010). MONTONI (2010) agrupa as atividades do estudo de revisão sistemática em três atividades:

#### *1. Desenvolver o protocolo*

Nesta atividade, uma prospecção sobre o tema de pesquisa, por meio de estudos preliminares e revisão informal da literatura, é realizada com a finalidade de definir o escopo do estudo. O protocolo de pesquisa é definido com base nos estudos preliminares e no escopo definido. Neste momento, o protocolo é testado, através de uma simulação, para se certificar de que a definição do problema e as questões de pesquisa estão bem-definidas e são adequadas, e que as expressões lógicas de buscas retornam publicações relevantes e relacionadas ao tópico de pesquisa. Um modelo de protocolo foi adaptado do modelo descrito em (KITCHENHAM, 2004), composto por: contexto do estudo; objetivos; foco de pesquisa; formulação das questões de pesquisa; idiomas; métodos de busca; máquinas de buscas; procedimentos de seleção e critérios para inclusão e exclusão de publicações; procedimentos para extração de dados e definição do formulário de coleta.

O ciclo de melhoria do protocolo (definição, teste, análise dos resultados preliminares) pára quando os pesquisadores concordam com o nível e a taxa de relevância retornada a partir da simulação do protocolo.

### 2. *Conduzir a pesquisa*

Nesta atividade, o estudo é conduzido de acordo com o protocolo definido na atividade anterior. As citações são armazenadas em um gerenciador de referências. Os resultados são analisados como descritos nos procedimentos de análises e os critérios definidos no protocolo.

### 3. *Relatar resultados*

A atividade de relatar resultados é composta por empacotar e publicar resultados. Esta atividade é, portanto, executada por meio da publicação desta dissertação, que contém este estudo de revisão sistemática.

## **I.C. Desenvolver protocolo**

A primeira exploração da área de pesquisa foi realizada de maneira informal. A partir da sugestão dos orientadores e pesquisa sobre modelos de maturidade, como o CMMI-DEV (2010) e o MR-MPS (2011), que advogam a melhoria contínua para organizações de alta maturidade (organizações que têm seus processos críticos sob controle estatístico); publicações de trabalhos desenvolvidos no PESC/COPPE que propunham abordagens para melhoria de processo de software (ANDRADE, 2005; ALBUQUERQUE, 2008; CAMPOS, 2008b); e publicações, oriundas de pesquisa *ad hoc* em máquinas de busca científica, que tratassem de melhoria contínua para processos de software ou melhoria de processo de software para organizações de alta maturidade.

Dentre as publicações encontradas na pesquisa *ad hoc*, destacam-se os trabalhos de KALINOWSKI *et al.* (2008), no qual focava na melhoria contínua com base na prevenção de defeitos; CAIVANO (2005a), que propõe um arcabouço com diferentes tipos de testes estatísticos para identificar oportunidades de melhoria de processo; WANG *at al.* (2006), o qual apoia o estabelecimento e refinamento de *baseline* de desempenho de processo, que é a base do controle e melhoramento de processo; CHANG *et al.* (2008), que propõem um método de monitoração de múltiplas medidas de processo de software simultaneamente para auxiliar no processo de análise causal; e



GONÇALVES *et al.* (2008) que descrevem passos para identificar, classificar e priorizar ações de melhoria e inovações.

Os primeiros indícios evidenciavam que as estratégias para melhoria contínua de processo concentravam-se no esforço de detectar causas atribuíveis de variação (problemas) em processos de software. Observou-se a escassez de abordagens que focassem na melhoria de processos estáveis, isto é, na melhoria contínua de processos sob controle estatístico.

O escopo inicial da revisão sistemática foi definido com a finalidade de levantar trabalhos que propusessem alguma estratégia para melhoria contínua dos processos estáveis – no contexto da alta maturidade ou apenas sob controle estatístico para averiguar a oportunidade de pesquisa. A partir da definição do escopo inicial, um protocolo e planejamento do estudo foram elaborados.

### **i) Objetivo**

**Analisar** publicações científicas sobre melhoria contínua de processo de software

**Com o propósito de** identificar abordagens, métodos, técnicas, estratégias e *frameworks*

**Com relação a** abordagens para identificação de oportunidades de melhoria contínua de processo sob controle estatístico

**Do ponto de vista de** pesquisadores

**No contexto** acadêmico e industrial

### **ii) Questão de pesquisa**

A descrição da questão de pesquisa seguiu o modelo sugerido em (KITCHENHAM, 2004).

- *Problema:* A melhoria contínua de processos de software tem como objetivo propor mudanças que melhorem o desempenho de processos. Em processos sob controle estatístico, todos os problemas (causas atribuíveis) foram devidamente tratados, restando apenas variação inerente à execução do processo. Uma melhoria contínua no contexto da CEP pode ser concebida como uma diminuição da amplitude de variabilidade do processo e/ou uma mudança da tendência central do desempenho.
- *Questão principal (GQ):* Quais métodos, técnicas, estratégias, abordagens e *frameworks* têm sido propostos para auxiliar a identificação de oportunidades de melhoria contínua de processo sob controle estatístico?

- *Questões secundárias (QS):*
  - *QS1:* Quais abordagens, métodos, técnicas, estratégias e *frameworks* apoiam a sistematização do processo de melhoria contínua de processo de software sob controle estatístico?
  - *QS2:* Quais abordagens, métodos, técnicas, estratégias e *frameworks* são utilizados para fornecer informações sobre o processo, a fim de apoiar o processo de investigação da oportunidade de melhoria contínua?
  - *QS3:* Quais informações são coletadas sobre o processo para apoiar o processo de investigação da oportunidade de melhoria contínua?
- *Aplicação:* Organizações consideradas de alta maturidade ou organizações que possuam processos de software sob controle estatístico.
- *População:* Publicações sobre melhoria contínua de processo de software sob controle estatístico.
- *Intervenção:* Abordagem, técnica, estratégia, método ou *framework* para identificar oportunidades de melhoria contínua em processo de software sob controle estatístico.
- *Comparação:* Não há
- *Resultado:* Identificação de abordagens, métodos, técnicas, estratégias e *frameworks* para identificar oportunidades de melhoria contínua de processo de software sob controle estatístico.
- *Artigo(s) de Controle:*
  - Gonçalves, F.M.G.S., *et al.* A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach. 2008.
  - Caivano, D. Continuous software process improvement through statistical process control. in Ninth European Conference on Software Maintenance and Reengineering, CSMR 2005. Manchester, United Kingdom: Institute of Electrical and Electronics Engineers Computer Society.
  - Chang, C.P. and C.P. Chu, Improvement of causal analysis using multivariate statistical process control. Software Quality Journal, 2008. **16**(3): p. 377-409.
  - Kalinowski, M., G.H. Travassos, and D.N. Card. Towards a defect prevention based process improvement approach. in EUROMICRO 2008 - Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2008. Parma, Italy: Inst. of Elec. and Elec. Eng. Computer Society.
  - Wang, Q., *et al.* BSR: A statistic-based approach for establishing and refining software process performance baseline. in 28th International Conference on Software Engineering 2006, ICSE '06, 2006. Shanghai, China: Inst. of Elec. and Elec. Eng. Computer Society.

- *Palavras-Chave e Sinônimos:*
  - **Melhoria contínua de processo:** melhoria de processo + alta maturidade, oportunidade de melhoria.
  - **Alta maturidade:** alto nível de maturidade, CMMI 5, gerência estatística, análise estatística, gerência quantitativa, análise de desempenho de processo, sob controle estatístico, análise de causa-raiz.
  - **CMMI 5:** *CMMI level 5*, CMMI nível 5, MPS nível A
  - **Análise de causa-raiz:** análise causal, análise de causa
  - **Controle estatístico:** controle de processo estatístico
  - **Processo de software**
  - **Abordagem**
  - **Método**
  - **Técnica**
  - **Estratégia**
  - *Framework*

### iii) Fontes de pesquisa

*Critério de seleção de fontes de pesquisa:*

- i. As fontes são bases de dados eletrônicas (máquinas de buscas) listadas no portal CAPES<sup>14</sup>.
- ii. As fontes devem permitir o uso de expressões lógicas de busca ou mecanismo similar para localizar as publicações através do título e resumo (*abstract*) das publicações.
- iii. Estar relacionada à área de exatas ou áreas correlacionadas.
- iv. Disponibilizar a recuperação do texto completo das publicações.
- v. Permitir o uso de expressões lógicas de busca ou mecanismos equivalentes para consultar publicações.

*Idioma:* O inglês foi escolhido por compor a grande maioria dos trabalhos publicados nas conferências e periódicos internacionais relacionado ao tema de pesquisa. Além disso, a maioria das editoras disponíveis no Portal de Periódicos da CAPES utilizam inglês como idioma padrão relevância.

---

<sup>14</sup><http://www.periodicos.capes.gov.br>

*Lista de fontes de pesquisa:* As fontes *Scopus*, *Compendex*, *Web Of Science* e *IEEE* foram selecionadas, pois atendem aos critérios de seleção de fontes supracitados e por serem consideradas máquinas de busca estáveis, consistentes e abrangentes, como evidenciado em (SANTOS, 2008; MONTONI, 2010; SCHOTS, 2010).

*Tipos de documentos:* Artigos, publicações, teses de doutorado e dissertações de mestrados.

*Ano publicação:* Trabalhos publicados entre os anos de 1985 (ano da publicação da primeira publicação de melhoria contínua de software utilizando controle estatístico de processo identificado na revisão da literatura preliminar) até o ano da publicação dessa dissertação.

*Métodos de busca de fontes:* Busca direta nas máquinas de buscas através das expressões de busca definidas neste protocolo.

*Expressão lógica de busca:*

((("process improvement" OR "melhoria de processo" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND (("high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade" OR "highest CMMI maturity level" OR "nível mais alto de maturidade" OR "CMMI level 5" OR "CMMI nível 5" OR "MPS nível A") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo"))))

#### **iv) Critérios e procedimentos de seleção**

*Critérios de inclusão e exclusão:* As publicações devem apresentar propostas de abordagens, métodos, técnicas, estratégias e abordagens para identificar oportunidades de melhoria de processo sob controle estatístico ou para apoiar o

processo de investigação de oportunidade de melhoria contínua de processo controlado estatisticamente. O texto completo da publicação deve estar disponível *on-line*.

*Procedimentos de seleção:* A seleção dos estudos foi feita nas seguintes etapas:

1. *Seleção preliminar e catalogação:* A seleção preliminar foi feita a partir da aplicação da expressão de busca nas máquinas de busca das fontes selecionadas (editoras). Foram coletadas informações sobre título, autor(es), data de publicação, referência completa, resumo (*abstract*) e fonte de origem. Cada publicação foi catalogada e armazenada na ferramenta de gerenciamento de referências EndNotes<sup>15</sup>. As entradas em duplicidade foram mescladas, permanecendo apenas uma referência ao documento retornado em duplicidade pelas fontes.
2. *Seleção dos dados relevantes (1º filtro):* A seleção preliminar não garante que todas as publicações encontradas sejam úteis no contexto do tema de pesquisa, pois a aplicação das expressões de busca é restrita ao aspecto sintático. Desta forma, após a seleção preliminar e catalogação das publicações, os resumos (*abstract*) dos documentos resultantes da seleção preliminar foram lidos, a fim de descartar as publicações que não atendessem a um dos seguintes critérios:
  - CS1 – a publicação deve propor ou descrever abordagens, métodos, técnicas, estratégias e *frameworks* para identificação de oportunidade de melhoria contínua em processo de software sob controle estatístico.
  - CS2 – a publicação deve propor ou descrever abordagens, métodos, técnicas, estratégias e *frameworks* para identificar informações ou elementos de processo que auxiliem na identificação de oportunidade de melhoria contínua de processo de software sob controle estatístico.
  - CS3 – a publicação deve descrever ou indicar fatores importantes para apoiar a melhoria contínua de processo de software sob controle estatístico.

Nesta etapa, os pesquisadores envolvidos – o autor desta dissertação e orientadores – acordaram consensualmente a respeito das publicações descartadas.

3. *Seleção dos dados relevantes (2º filtro):* Nesta última seleção dos artigos, a leitura do texto completo das publicações que restaram da etapa anterior foi conduzida, para se garantir que o material selecionado atendesse aos critérios de seleção CS1, CS2 e CS3. Igualmente ao passo anterior, os pesquisadores

---

<sup>15</sup> <http://www.endnote.com/>

envolvidos concordam em relação ao descarte das publicações que não atenderam a nenhum dos critérios.

## **v) Procedimentos para extração de dados**

Para cada publicação aprovada pelo processo de seleção, um formulário de coleta com as seguintes informações foi preenchido:

- Dados de publicação (título, autor(es), ano da publicação, referência completa);
- Resumo da publicação;
- Contexto de execução;
- Descrição da abordagem, método, técnica, estratégia e/ou *framework* propostos ou descritos para identificar oportunidades de melhoria contínua em processo de software sob controle estatístico;
- Descrição das abordagens, métodos, técnicas, estratégias e/ou *framework* que são utilizados para fornecer informações sobre o processo a fim de apoiar o processo de investigação da oportunidade de melhoria contínua;
- Descrição das informações coletadas sobre o processo para apoiar o processo de investigação da oportunidade de melhoria contínua.

## **vi) Procedimentos para análise**

A análise dos dados foi feita tanto quantitativa como qualitativamente.

A análise quantitativa consiste em fornecer o número de publicações selecionadas para fazerem parte deste estudo baseado em revisão sistemática. A análise qualitativa utilizou como base os dados quantitativos, realizando considerações com o intuito de discutir os resultados da busca e seleção com relação às questões de pesquisadas supracitadas.

### **I.D. Teste do protocolo**

Foram conduzidos vários testes para o protocolo deste estudo. À medida que as buscas com as primeiras versões do protocolo eram realizadas, algumas publicações evidenciaram palavras-chave ou sinônimos que não haviam inicialmente sido considerados. A cada iteração, foi realizada uma verificação das publicações retornadas com base inicialmente no título do texto; seguido pelo resumo da publicação, caso houvesse dúvida quanto ao conteúdo do artigo; e finalmente, caso houvesse indício de que o artigo seria selecionado, leitura do texto completo. Com

base nos resultados desta verificação, a expressão lógica de busca era refinada e novamente aplicada nas máquinas de busca. Durante o aprimoramento da expressão de busca, a questão de pesquisa também foi refinada, refletindo a definição do foco de pesquisa.

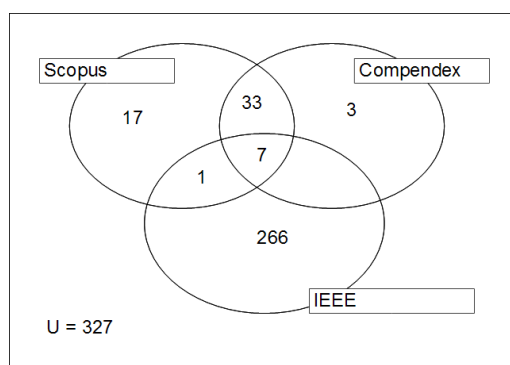
Foram identificadas publicações que serviriam como artigos de controle, isto é, trabalhos que deveriam ser retornados a partir da expressão lógica de busca. Os artigos definidos de controle estão descritos na seção 0.

### i) Primeira iteração de teste

Na primeira execução de testes foi utilizada a seguinte expressão lógica de busca:

*((("continuous improvement" OR "melhoria contínua" OR "incremental improvement" OR "melhoria incremental") OR ("statistical process control" OR "controle estatístico de processo" OR "process performance analysis" OR "análise de desempenho de processo" OR "root-cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia) AND ("software process" OR "processo de software"))*

As editoras utilizadas nesta primeira iteração de teste foram as fontes selecionadas na seção I.C.iii). Foram obtidos 14.186 artigos pela *Web Of Science*. Resolveu-se, neste primeiro momento, não analisar os resultados desta editora, pois, claramente, seria necessário um refinamento na expressão lógica de busca para esta máquina de busca. Além disso, a *Scopus*, *Compendex* e *IEEE* retornaram, respectivamente, 58, 43 e 274 publicações, como mostrado na Figura I.1.



**Figura I.1** - Publicações retornadas pela expressão de busca na primeira iteração das máquinas *Scopus*, *Compendex* e *IEEE*

Com base na leitura dos resumos dos trabalhos, observando os critérios de seleção, foram selecionadas 15 publicações, dentre as quais 13 foram lidas e duas não estavam disponíveis. A distribuição dos trabalhos selecionados dentro das áreas de pesquisa foi:

- *Scopus*: Ciências da Computação (7%)
- *Compendex*: Computação (7%), Sistemas de controle (10%), Gestão (6%), Estatística e Matemática (8%), Engenharia de produção (10%), Manufatura e Computação (100%).

Não houve publicação relevante no conjunto de trabalhos retornado pela *IEEE*. As publicações<sup>16</sup> 283, 339, 379 e 395 foram incluídas como artigos de controle.

## ii) Segunda iteração de teste

Os níveis de relevância encontrados na primeira iteração de teste do protocolo revelaram que os artigos de controle selecionados concentraram-se nas áreas de ciência da computação (ou apenas computação) e engenharia de software. As demais áreas de pesquisas apareceram sempre relacionadas às áreas citadas. Foi adicionada à expressão lógica de todas as máquinas de busca restrições em relação à área de pesquisa.

Observou-se, com base na leitura dos resumos e textos completos das publicações na iteração anterior, que algumas palavras-chave deveriam ser adicionadas, enquanto outras não eram referenciadas como sinônimo de melhoria contínua.

Foi adicionado o termo “melhoria de processo de software” e removidos os termos “*melhoria contínua*”, “*continuous improvement*”, “*melhoria incremental*” e “*incremental improvement*”. As expressões lógicas de busca, para cada editora, foram:

- *Scopus*

*TITLE-ABS-KEY(("software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia) AND ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR*

---

<sup>16</sup> As publicações são referenciadas a partir da numeração descrita na Tabela I.2



*"análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "process performance analysis" OR "análise de desempenho de processo")) AND (LIMIT-TO(SUBJAREA, "COMP") OR LIMIT-TO(SUBJAREA, "ENGI") OR LIMIT-TO(SUBJAREA, "MULT"))*

- *Compendex*

*("software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia) AND ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "process performance analysis" OR "análise de desempenho de processo")) ({723.1} OR {731.1} OR {723.5} OR {913.3} OR {723})  
WN CL*

- *IEEE*

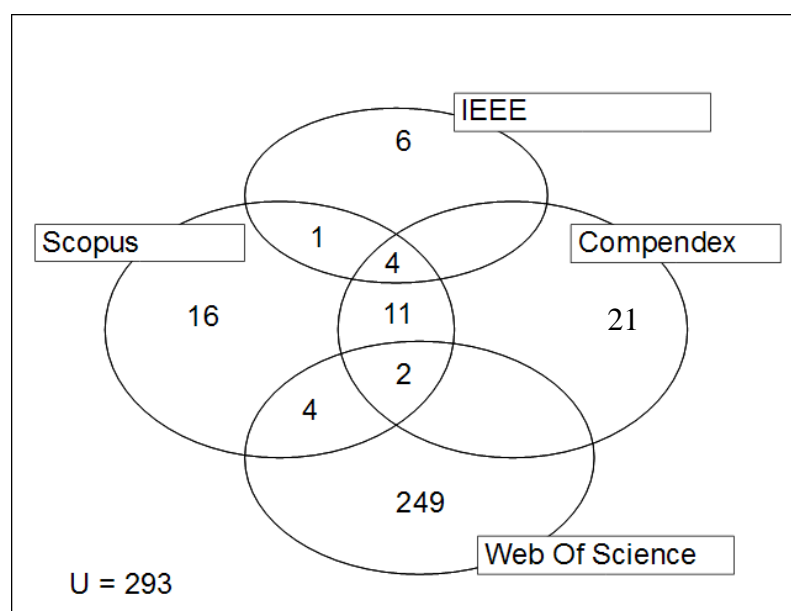
*((("software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia) AND ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "process performance analysis" OR "análise de desempenho de processo")) Refined by: Subject: Computing & Processing (Hardware/Software)*

- *Web Of Science*

*TS=(((software process improvement) OR (melhoria de processo de software) OR (software process) OR (processo de software)) AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia) AND ((root-cause analysis) OR (análise de causa raiz) OR (root cause analysis) OR (análise de causa-raiz) OR (causal analysis) OR (análise de causa) OR (análise causal) OR (statistical process control) OR (controle de processo estatístico) OR (controle estatístico de processo) OR (process performance analysis) OR (análise de*

*desempenho de processo))) Refined by: Subject Areas=( COMPUTER SCIENCE, SOFTWARE ENGINEERING OR COMPUTER SCIENCE, THEORY & METHODS OR COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS OR COMPUTER SCIENCE, INFORMATION SYSTEMS ) Timespan=All Years. Databases=SCI-EXPANDED, SSCI, A&HCI.*

A máquina de busca *Scopus* retornou 38 publicações, incluindo os 17 artigos encontrados na *Compendex*. A máquina de busca *Web Of Science* retornou um total de 606 artigos, contudo, ao aplicar o filtro de áreas de pesquisa relacionadas, obteve-se 256 publicações. A editora *Compendex* retornou 17 artigos e a *IEEE* retornou 11 trabalhos. O resultado é mostrado na Figura I.2.



**Figura I.2** - Publicações retornadas pelas expressões de busca na segunda iteração

Todos os artigos de controle foram retornados pelas expressões de busca. Dentre os 223 artigos encontrados, foram selecionados, com base nos critérios de seleção, doze artigos a partir da leitura dos resumos. Não foi possível obter os textos completos de três trabalhos. Essa nova rodada de teste da expressão de busca não revelou nenhum artigo relevante. Todos os artigos de controle foram retornados na máquina de busca da *Scopus*. Um artigo selecionado não tinha texto completo disponível.

### iii) Terceira iteração de teste

Foi adicionado o termo “*framework*” como sinônimo para sistemática e abordagem, pois se observou a sua ocorrência em algumas publicações. Com o intuito de abranger trabalhos que referenciassem melhoria contínua nos altos níveis de maturidade de modelos de processo de software (CMMI Dev e MR-MPS), foram adicionados os termos “*high maturity level*”, “*nível de maturidade alto*”, “*nível alto de maturidade*” e “*nível de alta maturidade*”.

A fim de adicionar mais publicações que apoiassem a questão de pesquisa secundária CS2, os termos “*gerência quantitativa*”, “*análise estatística*” e “*análise de desempenho de processo*” foram incluídos.

As expressões lógicas de buscas para a terceira iteração de teste do protocolo foram:

- *Scopus*

*TITLE-ABS-KEY(("software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND (("high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo")) AND (LIMIT-TO(SUBJAREA, "COMP") OR LIMIT-TO(SUBJAREA, "ENGI") OR LIMIT-TO(SUBJAREA, "MULT"))*

- *IEEE*

*(("software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND (("high maturity level" OR "nível de maturidade alto" OR "nível*

alto de maturidade" OR "nível de alta maturidade") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo")) *Refined by: Subject: Computing & Processing (Hardware/Software)*

- *Compendex*

((*"software process improvement" OR "melhoria de processo de software" OR "software process" OR "processo de software"*) AND (*\$approach OR \$abordagem OR \$method OR \$método OR \$technique OR \$técnica OR \$strategy OR \$estratégia OR \$framework*) AND ((*"high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade"*) OR (*"root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo"*))) (*{723.1} OR {731.1} OR {723} OR {723.5} OR {913.3} OR {722}*) WN CL

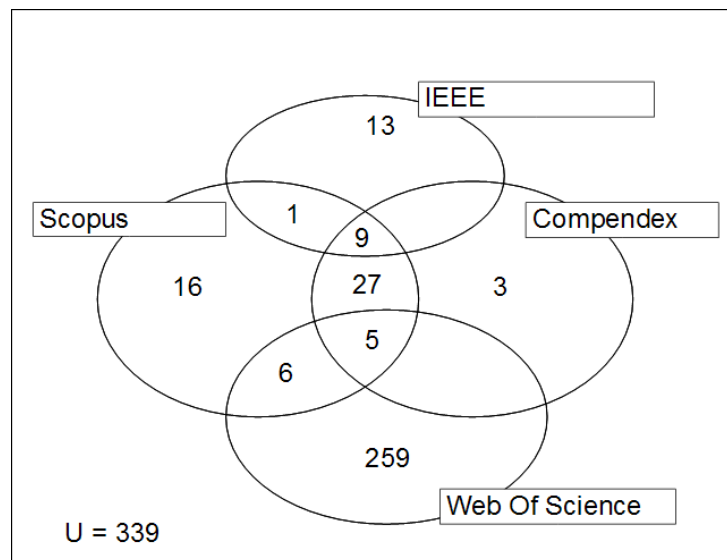
- *Web Of Science*

*TS=(((software process improvement) OR (melhoria de processo de software) OR (software process) OR (processo de software)) AND (approach OR abordagem OR method OR metodo OR technique OR tecnica OR strategy OR estrategia OR framework) AND (((high maturity level) OR (nivel de maturidade alto) OR (nivel alto de maturidade) OR (nivel de alta maturidade)) OR ((root-cause analysis) OR (analise de causa raiz) OR (root cause analysis) OR (analise de causa-raiz) OR (causal analysis) OR (analise de causa) OR (analise causal) OR (statistical process control) OR (controle de processo estatistico) OR (controle estatistico de processo) OR (statistical management) OR (gerencia estatistica) OR (quantitative management) OR*

(*gerencia quantitativa*) OR (*quantitative process management*) OR (*gerencia quantitativa de processo*) OR (*statistical analysis*) OR (*analise estatistica*) OR (*process performance analysis*) OR (*analise de desempenho de processo*)) )) Refined by: Subject Areas=( *COMPUTER SCIENCE, SOFTWARE ENGINEERING OR ENGINEERING, ELECTRICAL & ELECTRONIC OR COMPUTER SCIENCE, THEORY & METHODS* )

O resultado desta iteração de teste é mostrado na Figura I.3. A busca na máquina *Scopus* retornou 64 trabalhos, enquanto a *Compendex*, *IEEE* e a *Web Of Science* retornaram respectivamente 49, 23 e 270 publicações. Nesta rodada, quatro novas publicações foram consideradas relevantes (232, 310, 355, 403).

Novamente, todos os artigos de controle foram encontrados na *Scopus*.



**Figura I.3** - Publicações retornadas pelas expressões de busca na terceira iteração

#### iv) Quarta iteração de teste

Um novo artigo de controle foi adicionado ao protocolo (387). Desta forma, a expressão lógica de busca deveria ser modificada para incluí-lo. Os termos “*CMMI level 5*”, “*CMMI nível 5*”, “*MPS nível A*”, “*highest CMMI maturity level*”, “*nível mais alto de maturidade*”, relacionados à alta maturidade, foram incluídos. Além disso, o termo “*melhoria de processo de software*” (“*software process improvement*”) foi modificado para abranger todas as possibilidades de melhoria de processo. A abrangência das áreas de pesquisa da máquina de busca *Web Of Science* também foi

reduzida, pois nenhum artigo fora da área de computação foi considerado relevante até a terceira iteração de teste.

As expressões de buscas para esta iteração foram:

- *Scopus*

*TITLE-ABS-KEY(("process improvement" OR "melhoria de processo" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND (("high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade" OR "highest CMMI maturity level" OR "nível mais alto de maturidade" OR "CMMI level 5" OR "CMMI nível 5" OR "MPS nível A") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo")) AND (LIMIT-TO(SUBJAREA, "COMP") OR LIMIT-TO(SUBJAREA, "ENGI") OR LIMIT-TO(SUBJAREA, "MULT"))*

- *Compendex*

*(("process improvement" OR "melhoria de processo" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND (("high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade" OR "highest CMMI maturity level" OR "nível mais alto de maturidade" OR "CMMI level 5" OR "CMMI nível 5" OR "MPS nível A") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR*

*"análise estatística" OR "process performance analysis" OR "análise de desempenho de processo")) ({731.1} OR {913.3} OR {723.5} OR {723.1} OR {723}) WN CL*

- *IEEE*

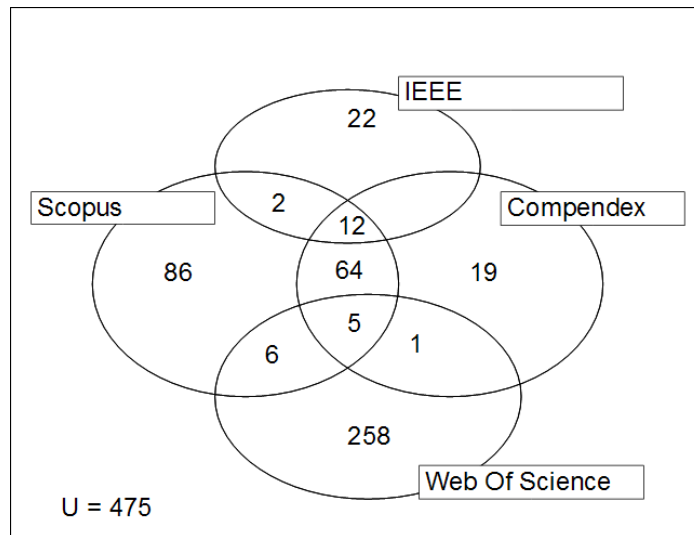
*((("process improvement" OR "melhoria de processo" OR "software process" OR "processo de software") AND (approach OR abordagem OR method OR método OR technique OR técnica OR strategy OR estratégia OR framework) AND ((("high maturity level" OR "nível de maturidade alto" OR "nível alto de maturidade" OR "nível de alta maturidade" OR "highest CMMI maturity level" OR "nível mais alto de maturidade" OR "CMMI level 5" OR "CMMI nível 5" OR "MPS nível A") OR ("root-cause analysis" OR "análise de causa raiz" OR "root cause analysis" OR "análise de causa-raiz" OR "causal analysis" OR "análise de causa" OR "análise causal" OR "statistical process control" OR "controle de processo estatístico" OR "controle estatístico de processo" OR "statistical management" OR "gerência estatística" OR "quantitative management" OR "gerência quantitativa" OR "quantitative process management" OR "gerência quantitativa de processo" OR "statistical analysis" OR "análise estatística" OR "process performance analysis" OR "análise de desempenho de processo")))) Refined by Subject: Computing & Processing (Hardware/Software) , General Topics for Engineers (Math, Science & Engineering)*

- *Web Of Science*

*TS=(((software process improvement) OR (melhoria de processo de software) OR (software process) OR (processo de software)) AND (approach OR abordagem OR method OR metodo OR technique OR tecnica OR strategy OR estrategia OR framework) AND (((high maturity level) OR (nivel de maturidade alto) OR (nivel alto de maturidade) OR (nivel de alta maturidade) OR (highest CMMI maturity level) OR (nivel mais alto de maturidade) OR (CMMI level 5) OR (CMMI nivel 5) OR (MPS nivel A)) OR ((root-cause analysis) OR (analise de causa raiz) OR (root cause analysis) OR (analise de causa-raiz) OR (causal analysis) OR (analise de causa) OR (analise causal) OR (statistical process control) OR (controle de processo estatistico) OR (controle estatistico de processo) OR (statistical management) OR (gerencia estatistica) OR (quantitative management) OR (gerencia quantitativa) OR (quantitative process management) OR (gerencia quantitativa de processo) OR (statistical analysis) OR (analise estatistica) OR (process performance analysis) OR (analise de desempenho de processo)) ) ) Refined by: Subject Areas=( COMPUTER*

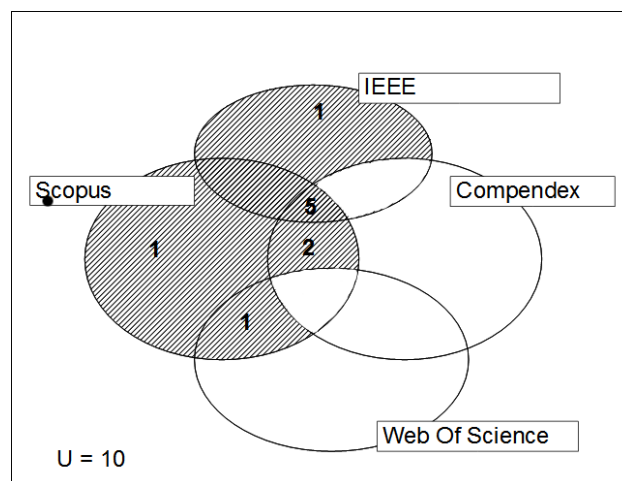
*SCIENCE, SOFTWARE ENGINEERING OR COMPUTER SCIENCE, THEORY & METHODS* )

Os resultados obtidos a partir de as expressões de busca aumentaram o número de publicações retornadas pela *Scopus*, *IEEE* e *Compendex*, respectivamente 175, 36 e 101 artigos, como mostra a Figura I.4. Os artigos retornados pela *Web Of Science* foram os mesmo 270 retornados na iteração anterior.



**Figura I.4** - Publicações retornadas pelas expressões de busca na quarta iteração

Das 11 publicações selecionadas, 1 foi encontrada na *Scopus*; 1 publicação encontrada na *IEEE*; 2 publicação encontrada na *Scopus* e na *Compendex*; 1 publicação encontrada na *Scopus* e na *Web Of Science*; 5 publicações encontradas na *IEEE*, *Compendex* e *Scopus*; e 1 publicação encontrada na *Scopus*, *Compendex* e *Web Of Science*. O resultado, destacando as interseções, é mostrado na Figura I.5.



**Figura I.5** - Publicações selecionadas na quarta iteração



## **I.E. Avaliação do protocolo**

O protocolo foi avaliado em apresentações de seminários realizados na COPPE/UFRJ envolvendo especialistas na área. Os seminários são apresentações realizadas periodicamente pelos alunos de pós-graduação da área de Qualidade de Software da linha de Engenharia de Software. Nos seminários, os alunos da área de Qualidade de Software apresentam o andamento e os resultados de suas pesquisas, a fim de obter retorno e sugestões do(s) orientador(es) e dos demais alunos. Ademais, o resultado da avaliação foi positivo, pois os artigos de controle foram retornados.

## **I.F. Execução da Pesquisa**

Após a aprovação do protocolo, o estudo foi executado nos seguintes momentos: durante a prospecção do tema de pesquisa (março de 2010), antes do exame de qualificação desta dissertação (agosto de 2010) e após a execução da avaliação da abordagem proposta nesta dissertação (maio de 2012). As execuções são descritas nas subseções seguintes.

### **i) 1ª execução – março de 2010**

A primeira etapa descrita nos procedimentos de seleção na seção I.C.vi) apresentou, como resultado da aplicação da expressão lógica de busca nas fontes de pesquisas selecionadas descritas na seção I.C.iii), um total de 474 publicações, dos quais, as máquinas de busca *Scopus*, *Compendex*, *IEEE* e *Web Of Science* retornaram, respectivamente, 175, 36, 101 e 270 publicações, como mostrado na Figura I.4. Não foi possível acessar o texto completo de sete publicações, indicadas na Tabela I.2.

Todos os resumos (*abstract*) dos artigos retornados foram lidos e apenas 28 foram selecionados para a terceira etapa de análise (leitura completa do texto). Desses 28 artigos, apenas 10 atendiam a um dos critérios definidos na seção I.C.iv). Os critérios estão relacionados com as publicações na Tabela I.2.

Para cada publicação selecionada para a terceira etapa, inclusive os artigos de controle, foram extraídos os dados referentes ao formulário descrito na seção I.C.v), a partir da leitura do texto completo. Os formulários preenchidos são apresentados na seção I.H.ii).

## ii) 2ª execução – agosto de 2010

Uma segunda execução do protocolo foi executada no período que antecedeu o exame de qualificação desta dissertação e nenhuma nova publicação foi localizada.

## iii) 3ª execução – maio de 2012

Uma terceira execução do protocolo foi executada no período que antecedeu a defesa desta dissertação e quize novas publicações foram localizadas e são apresentadas na Tabela I.1.

**Tabela I.1** - Lista de publicações retornadas na 3a execução

#	Ano	Autor(res)	Título	Fonte	Critério
475	2011	Shao, Y.E., Hou, C.-D., Chao, C.-H., Chen, Y.-J.	A decomposition approach for identifying the sources of variance shifts in a multivariate process	Scopus	
476	2011	Psaroudakis, J.E.; Eberhardt, A.	A Discrete Event Simulation Model to Evaluate Changes to a Software Project Delivery Process	IEEE	
477	2011	Wells, L.J., Megahed, F.M., Camelio, J.A., Woodall, W.H.	A framework for variation visualization and understanding in complex manufacturing systems	Scopus	
478	2010	Sandhu, P.S.; Goel, R.; Brar, A.S.; Kaur, J.; Anand, S.	A model for early prediction of faults in software systems	IEEE	
479	2010	Tianying Chen; Bosheng Zhou; Wenjie Luo	A Process Optimization Method for High Maturity Process Improvements	IEEE	
480	2010	Bering, T.P.K.; Veldhuis, S.C.	A Quality Framework to check the applicability of engineering and statistical assumptions for automated gauges	IEEE	
481	2010	Tahir, T.; Gencel, C.	A structured goal based measurement framework enabling traceability and prioritization	IEEE , Scopus	
482	2010	Barcellos, M.P.; de Almeida Falbo, R.; Rocha, A.R.	A Well-Founded Software Process Behavior Ontology to Support Business Goals Monitoring in High Maturity Software Organizations	IEEE, Scopus	
483	2011	Atashgar, K., Noorossana, R.	An integrating approach to root cause analysis of a bivariate mean vector with a linear trend disturbance	Scopus	
484	2012	Tarhan, Ayca; Demirors, Onur	Apply Quantitative Management Now	IEEE	
485	2010	Kalinowski, M., Mendes, E., Card, D.N., Travassos, G.H.	Applying DPPI: A defect causal analysis approach using Bayesian networks	Scopus	<b>CS1</b>

486	2011	Shen, W.-H., Hsueh, N.-L., Lee, W.-M.	Assessing PSP effect in training disciplined software development: A Plan-Track-Review model	Scopus	
487	2011	Kalinowski, M., Mendes, E., Travassos, G.H.	Automating and evaluating probabilistic cause-effect diagrams to improve defect causal analysis	Scopus	CS1
488	2011	Zandi, F., Niaki, S.T.A., Nayeri, M.A., Fathi, M.	Change-point estimation of the process fraction non-conforming with a linear trend in statistical process control	Scopus	
489	2011	Monteiro, L.F.S., De Oliveira, K.M.	Defining a catalog of indicators to support process performance analysis	Scopus	
490	2010	Barreto, A.O.S., Rocha, A.R.	Defining and monitoring strategically aligned software improvement goals	Scopus	
491	2011	Lehtinen, T.O.A., Mäntylä, M.V., Vanhanen, J.	Development and evaluation of a lightweight root cause analysis method (ARCA method) - Field studies at four software companies	Scopus	
492	2011	Angeli, I.	How quality management can help in process control: Case studies from Cyprus	Scopus	
493	2011	Tarhan, A., Demirors, O.	Investigating the effect of variations in the test development process: A case from a safety-critical system	Scopus	
494	2011	Cai-fen, G., Ran-zhe, J.	Modeling and controlling work-in-progress in discrete manufacturing systems	Scopus	
495	2010	Rios, B.L.F., Ramírez, S.L.G., Rodríguez-Elias, O.M.	Modeling knowledge flows in software project management processes	Scopus	
496	2011	Liou, J.-C.	On improving CMMI in an immature world of software development	Scopus	
497	2010	Bin Basri, S.; O'Connor, R.V.	Organizational commitment towards software process improvement an irish software vses case study	IEEE, Scopus	
498	2010	Chatzipetrou, P., Angelis, L., Rovegård, P., Wohlin, C.	Prioritization of issues and requirements by cumulative voting: A compositional data analysis framework	Scopus	
499	2011	Sridhar, B., Rajesh, G.	Process innovation reaping customer satisfaction	Scopus	
500	2010	Llamosa-Villalba, R.; Aceros, S.	Process Management Model for Higher Education: Improvement of educational programs in software quality	IEEE	
501	2011	Mingzhi Gao; Zuochang Ye; Dajie Zeng; Yan Wang; Zhiping Yu	Robust spatial correlation extraction with limited sample via L1-norm penalty	IEEE	
502	2010	Tong Shi; Jianbin Chen; JiShun Zhu	Study on assessment framework of software process in agile	IEEE, Scopus	
503	2011	Suradom, N., Homrossukon, S.,	Surface mounting process improvement via constrained	Scopus	

		Luangpaiboon, P.	response surface optimization and conventional statistical analysis		
504	2010	Hicks, B.J., Matthews, J.	The barriers to realising sustainable process improvement: A root cause analysis of paradigms for manufacturing systems improvement	Scopus	
505	2010	Zhang Haohua; Wu Lijuan; Qi Weiyi; Wu Jianjun; Ma Shijun	The New Measure Method for Large-Scale Object-Oriented Software System	IEEE	
506	2011	Zawedde, A.S.A., Klabbers, M.D.M., Williams, D.D., Van Den Brand, M.G.J.M.	Understanding the dynamics of requirements process improvement: A new approach	Scopus	
507	2011	Alba-Castro, M., Gil, S.H.	Validation and calibration of quantitative models for software development effort and size estimation	Scopus	
508	2011	Lehtinen, T.O.A., Mäntylä, M.V.	What are problem causes of software projects? Data of root cause analysis at four software companies	Scopus	
509	2011	Lehtinen, T.O.A.; Mantyla, M.V.	What are Problem Causes of Software Projects? Data of Root Cause Analysis at Four Software Companies	IEEE, Scopus	
510	2011	Schneidewind, N.	What can software engineers learn from manufacturing to improve software process and product?	Scopus	

As fontes utilizadas nesta execução foram restritas a IEEE e Scopus por serem as fontes que apresentaram nas execuções anteriores publicações selecionadas. Com base nos critérios e na metodologia do estudo baseado em revisão sistemática, não foi identificada nenhuma publicação que atendesse aos critérios. Alguns trabalhos que aparecem nesta execução relatavam abordagem para melhoria de processo de software na alta maturidade, como as publicações de número 479 e 484. No entanto, o foco destes trabalhos eram métodos para orientar a aplicação do CEP em organizações de alta maturidade. As publicações de número 485 e 487 foram observadas e selecionadas, muito embora as publicações façam parte do mesmo trabalho citam em (KALINOWSKI *et al.*, 2008) e defendido em sua versão final em (KALINOWSKI, 2011).

### **I.G. Avaliação dos Resultados da Pesquisa**

Os trabalhos encontrados relacionados à primeira questão secundária de pesquisa QS1 – *Quais abordagens, métodos, técnicas, estratégias e frameworks*

*apoiam a sistematização do processo de melhoria contínua de processo de software sob controle estatístico?* – propõem melhoria de processo de software, possuindo em comum a utilização de conceitos e ferramentas estatísticas.

No trabalho de HONG e GOH (2003), por exemplo, é proposta uma aplicação do *framework* DMAIC (*define, measure, analyze, improve, control*) do Seis Sigma, uma abordagem de melhoria de processo bem sucedido na manufatura, aplicando ferramentas do *framework* para melhoria de processos de software. Nesta mesma motivação, PARK *et al.* (2007) propõem um *framework* para utilizar ferramentas do Seis Sigma no PSP/TSP (*Personal Software Process/Team Software Process*), no intuito de incluir conceitos de controle estatístico de processo e análise quantitativa nestes *frameworks* de processos de software propostos pelo SEI (*Software Engineering Institute*). Entretanto, nestes trabalhos, não são propostas sistemáticas para melhoria contínua.

CAIVANO (2005a) descreve uma abordagem para aplicação de controle estatístico de processo (SPC) para melhoria de processo de software. Tais trabalhos apoiam a melhoria contínua de processo de software com base no controle estatístico. BOFFOLI (2006) evolui a abordagem de CAIVANO, considerando a Calibragem Dinâmica (*Dynamic Calibration*) como técnica para alcançar resultados de estimativas mais realísticos com base nos dados gerados pelos projetos.

Alguns trabalhos focam na etapa de investigação da oportunidade de melhoria. A abordagem descrita em (KALINOWSKI *et al.*, 2008), por exemplo, utiliza os conceitos do controle estatístico de forma a encontrar oportunidades de melhoria de processo de software, a partir de um processo de prevenção de defeitos mais elaborado. O trabalho emprega conceitos probabilísticos bayesianos para identificar padrões e os utilizar no processo de análise causal de defeitos, contribuindo para a estabilização e melhoria contínua dos processos de software. Ademais, a estratégia, proposta por GONÇALVES *et al.* (2008), descreve passos para identificar, classificar e priorizar ações de melhoria. No entanto, não descreve passos sistemáticos para a fase de identificação da oportunidade de melhoria.

A questão de pesquisa secundária QS2 – *Quais abordagens, métodos, técnicas, estratégias e frameworks que são utilizados para fornecer informações sobre o processo, a fim de apoiar o processo de investigação da oportunidade de melhoria contínua?* – foi parcialmente respondida. Os trabalhos se concentraram em

*frameworks* ou estratégias para incrementar a eficiência do controle estatístico por meio de análises de causas utilizando controle estatístico de processo multivariado (CHANG e CHU, 2008); ou a partir do refinamento de *baselines* de desempenho (WANG *et al.*, 2006).

Apenas uma publicação foi relacionada à terceira questão secundária de pesquisa QS3 – *Quais informações são coletadas sobre o processo para apoiar o processo de investigação da oportunidade de melhoria contínua?* –, contendo informações, instruções e diretrizes para a seleção de processos e métricas que auxiliem o entendimento quantitativo (TARHAN e DEMIRORS, 2008).

Conclui-se, portanto, com base neste estudo baseado em revisão sistemática, que há escassez de trabalhos que apóiem, de maneira sistemática, a melhoria contínua de processos de software sob controle estatístico. Em geral, observam-se muitos trabalhos que objetivam a melhoria de processos de software aplicando técnicas estatísticas, analisando a viabilidade de aplicação de técnicas bem sucedidas em outras áreas ou definindo estrutura e *frameworks* para auxiliar na tarefa de aplicação dos conceitos estatísticos. Há, portanto, indícios sobre a oportunidade de contribuição da proposta da abordagem desta dissertação, utilizando uma implementação da Teoria das Restrições (*Theory of Constraints*) para investigar oportunidades de melhoria contínua tanto em processo que estão sob controle estatístico como em processos de software não estáveis, pois a despeito da estabilidade, processos de software sofrem influências indesejáveis de fatores que podem ser investigados e tratados de forma sistemática.

## **I.H. Resultados da 1ª execução de março de 2010**

Nesta seção, é apresentado o resultado das etapas de análise descrita na seção I.C.vi) para a execução do estudo baseado em revisão sistemática em março de 2010.

A primeira subseção apresenta a listagem das publicações encontradas a partir da execução das expressões lógicas nas máquinas de busca da *Scopus*, *Compendex*, *IEEE* e *Web Of Science*. A segunda subseção apresenta os formulários extraídos dos artigos selecionados no processo de análise.

### **i) Listagem das publicações retornadas**

Na Tabela I.2, são listadas as 474 publicações retornadas pelas máquinas de busca. Os artigos que foram selecionados para a 2ª etapa de análise (leitura completa

do texto) estão destacados, assim como os artigos selecionados que atendiam a um dos critérios definidos na seção I.C.vi). Os artigos de controle, além de destacados, estão em negrito. Ademais, as publicações selecionadas para 2ª etapa de análise que não estavam disponíveis são indicadas na coluna critério.

**Tabela I.2** - Publicações retornadas na execução de março de 2010

#	Ano	Autor(res)	Título	Fonte	Critério
1	1985	Cibulas, S. M.	Statistical process control applications in improving gasket sealing capabilities	Scopus	
2	1986	Boley, D. C.	Application Of Basic Statistical Process Control Techniques To Rolling Mill Lubricants	Scopus	
3	1986	Bossert, J. L. ; LaBarr, J. A.	Kodak Quality Package	Scopus	
4	1986	Gast, G. E.	Tire Manufacturing Improvement Through Spc	Scopus	
5	1986	McCulloch, L. P.	Implementing Statistical Process Control	Scopus	
6	1986	Schultz, L. E.	What Spc Can Do For You	Scopus	
7	1987	Heeran, J. P.	Utilization Of Experimental Designs In Spc: A Functional Approach To Process Improvement	Scopus	
8	1987	Lackey, J. D.; Gerth, D. M., <i>et al.</i>	Machine Vision For Process Improvement: A Design Experience	Scopus	
9	1987	Smith-Vargo, L.	Engineers Run Experimental Designs To Improve Pcb Quality	Scopus	
10	1989	Anon	SPC for small to medium foundry's sand system. Part 3	Scopus	
11	1989	Fu, H. J.; Haggerty, C. M., <i>et al.</i>	Integrated trend analysis and reporting system	Scopus	
12	1990	Boyer, S. M. ; Philpott, N.	Control of hybrid processes using SPC	Scopus	
13	1990	Christenson, L.	Statistical analysis for manufacturing. A review of three software packages	Scopus	
14	1990	Forsha, A. L.	Fine copper wire production process improvements	Scopus	
15	1990	Gupta, R.; Holdeman, L., <i>et al.</i>	Manufacturing technology and yield studies for MMIC 5-bit digital attenuators and phase-shifters	Scopus	

16	1990	Jeffery, D. R. ; Low, G.	Calibrating estimation tools for software development	Scopus, Compendex, IEEE	
17	1990	Palm, A. C.	SPC versus automatic process control	Scopus	
18	1991	Ammar, R. A.; Wang, J., <i>et al.</i>	Graphic Modeling Technique for Software Execution Time-Estimation	Web of Science	
19	1991	Humphrey, W. S. ; Singpurwalla, N. D.	Predicting (Individual) Software Productivity	Web of Science	
20	1991	Klein, R.	Total Quality management in Bauxite processing the role of MVC	Scopus	
21	1991	Kraus, W. E.	Use of teams for organizational change	Scopus	
22	1991	Mahmud, S. M.	A Time-Domain Analysis of a Multiplex-Bus System	Web of Science	
23	1991	Miller, T. ; Balch, B.	Statistical process control in food processing	Scopus	
24	1991	Mozumder, P. K. ; Strojwas, A. J.	Statistical Control of Vlsi Fabrication Processes	Web of Science	
25	1991	Stoyenko, A. D.; Hamacher, V. C., <i>et al.</i>	Analyzing Hard-Real-Time Programs for Guaranteed Schedulability	Web of Science	
26	1992	Axland, S.	Small wonders	Scopus, Compendex	
27	1992	Edelson, N. M.; Ellis, M. L., <i>et al.</i>	Sequential 3-phase process improvement strategy	Scopus, Compendex	
28	1992	Lucks, M. ; Gladwell, I.	Automated Selection of Mathematical Software	Web of Science	
29	1992	Markhauser, C. P.	Computer-Aided Techniques for the Television System Engineer	Web of Science	
30	1992	Matter, D. J.	Customer evaluations: customer and supplier perspectives [telecommunication switching systems]	IEEE	
31	1992	Owen, M.	SPC in non-manufacturing - applications and opportunities	Scopus, Compendex	
32	1992	Pyzdek, T.	Continuous improvement strategy for software	Scopus, Compendex	2a etapa (indisponível)
33	1992	Slattery, R. H.	Leveraging statistical process control to stimulate continuous improvement	Scopus	2a etapa
34	1992	Sredni, J.	Design of experiments: a tool for continuous process improvement	IEEE	



35	1993	Anon	Refining the chip	Scopus	
36	1993	Benneyan, J. C. ; Chute, A. D.	SPC process improvement and the deming PDCA circle in freight administration	Scopus, Compendex	
37	1993	Bhandari, I.; Halliday, M., <i>et al.</i>	Case study of software process improvement during development	Scopus, Compendex	
38	1993	Boros, L. G.; Damico, J., <i>et al.</i>	An Automatized Computer-Method Utilizing Procomm Plus and Dataease(4.2) Pc and Sas(6.06) Mainframe Software for Isolated, Perfused Guinea-Pig Heart Studies	Web of Science	
39	1993	Dalal, S. R.; Horgan, J. R., <i>et al.</i>	Reliable software and communication: software quality, reliability, and safety	IEEE	
40	1993	Fabien, M. J.	Practical approaches to ISO 9000 certification for rubber & plastics processors	Scopus	
41	1993	Fabien, M. J.	Practical approaches to ISO 9000 certification for rubber plastics processors	Compendex	
42	1993	Glajchen, D.	RF instrument accuracy - some effects of measurement uncertainty	Scopus	
43	1993	Iazeolla, G. ; Marinuzzi, F.	Lispack - a Methodology and Tool for the Performance Analysis of Parallel Systems and Algorithms	Web of Science	
44	1993	Jeffery, D. R.; Low, G. C., <i>et al.</i>	A Comparison of Function Point Counting Techniques	Web of Science	
45	1993	Lindemann, C.	An Improved Numerical Algorithm for Calculating Steady-State Solutions of Deterministic and Stochastic Petri-Net Models	Web of Science	
46	1993	Lukacs, T. J. ; Pilkington, R.	Equipment selection and acceptance	Scopus	
47	1993	Metersky, M. L.	A Decision-Oriented Approach to System-Design and Development	Web of Science	
48	1993	Miller, W. H. ; Poole, M. W.	Advanced Cim Environment for Manufacturing Data-Analysis	Web of Science	
49	1993	Peng, W. W. ; Wallace, D. R.	Software error analysis	Scopus, Compendex	
50	1993	Pengelly, A.; Norris, M., <i>et al.</i>	Software Process Modeling and Measurement - a Qms Case-Study	Web of Science	

51	1993	Rowe, A. ; Whitty, R.	Ami - Promoting a Quantitative Approach to Software Management	Web of Science	
52	1993	Slattery, R. H.	Continuous improvement in P/M processing via statistical process control	Compendex	
53	1993	Sower, V. E.	SPC implementation in the plastic molding industry	Scopus, Compendex	
54	1993	Stanley, B. J. ; Wilson, A. E.	Continuous process improvements and the use of quality control methodologies in the data item description process	Scopus	
55	1993	Tucker, W. T.; Faltin, F. W., <i>et al.</i>	Algorithmic statistical process control. An elaboration	Compendex	
56	1993	Weide, B. W. ; Defazio, S.	A Framework for Modeling Software Engineering Processes	Web of Science	
57	1994	Bhandari, I.; Ray, B., <i>et al.</i>	An Inference Structure for Process Feedback - Technique and Implementation	Web of Science	
58	1994	Dalal, S. R.; Horgan, J. R., <i>et al.</i>	Reliable Software and Communication .2. Controlling the Software-Development Process	Web of Science	
59	1994	Debou, C.; Liptak, J., <i>et al.</i>	Decision-Making for Software Process Improvement - a Quantitative Approach	Web of Science	
60	1994	Frleigh, S. ; Wilson, J.	Visual programming of on-line systems for continuous process improvement	Scopus	
61	1994	Hausler, P. A.; Linger, R. C., <i>et al.</i>	Adopting Cleanroom Software Engineering with a Phased Approach	Web of Science	2a etapa (indisponível)
62	1994	Heiner, M.; Ventre, G., <i>et al.</i>	A Petri-Net Based Methodology to Integrate Qualitative and Quantitative-Analysis	Web of Science	
63	1994	Lanning, D. L. ; Khoshgoftaar, T. M.	Modeling the Relationship between Source Code Complexity and Maintenance Difficulty	Web of Science	
64	1994	Lee, Y. F. ; Ryder, B. G.	Effectively Exploiting Parallelism in Data-Flow Analysis	Web of Science	
65	1994	Thakore, A. K. ; Su, S. Y. W.	Performance Analysis of Parallel Object-Oriented Query-Processing Algorithms	Web of Science	
66	1994	Whittaker, J. A. ; Thomason, M. G.	A Markov-Chain Model for Statistical Software Testing	Web of Science	

67	1995	Debou, C.; Haux, M., <i>et al.</i>	A Measurement Framework for Improving Verification Processes	Web of Science	2a etapa
68	1995	French, V. A.	Applying software engineering and process improvement to legacy defence system maintenance: An experience report	Scopus, IEEE	
69	1995	Hariri, S. ; Mutlu, H.	Hierarchical Modeling of Availability in Distributed Systems	Web of Science	
70	1995	Henry, J.; Rossman, A., <i>et al.</i>	Quantitative evaluation of software process improvement	Scopus, Compendex	2a etapa
71	1995	Hill, J. R. ; Melamed, B.	Testool - a Visual Interactive Environment for Modeling Autocorrelated Time-Series	Web of Science	
72	1995	Lanning, D. L. ; Khoshgoftaar, T. M.	Fault Severity in Models of Fault-Correction Activity	Web of Science	
73	1995	Opdahl, A. L.	Sensitivity Analysis of Combined Software and Hardware Performance Models - Open Queuing-Networks	Web of Science	
74	1995	Parikh, P.	Disciplined approach to process improvement	Scopus, Compendex	
75	1995	Sanders, F. F.	Handling process disturbances and measurement errors	Scopus	
76	1995	Takada, Y. ; Torii, K.	A Method for Measuring Programmer Debugging Performance from Key Strokes	Web of Science	
77	1995	Xie, W.; Xie, M., <i>et al.</i>	Control charts for processes subject to random shocks	Scopus, Compendex	2a etapa (indisponível)
78	1996	Boardman, J. T. ; Cole, A. J.	Integrated process improvement in design and manufacture using a systems approach	Scopus, Compendex	
79	1996	Caulcutt, R.	Responding to process changes	Scopus	
80	1996	Chen, W. H.	The effects of SPC on the target of process quality improvement	Scopus, Compendex	
81	1996	Dabade, B. M. ; Ray, P. K.	Quality engineering for continuous performance improvement in products and processes: A review and reflections	Scopus, Compendex	
82	1996	Darr, M.	Leveraging your IE organization for maximum effectiveness	Scopus	

83	1996	Ermer, D. S. ; Hurtis, G. M.	Advanced SPC for higher-quality electronic card manufacturing	Scopus	
84	1996	Haase, V. H.	Software process assessment concepts	Scopus	
85	1996	Haverkort, B. R. ; Niemegeers, I. G.	Performability modelling tools and techniques	Web of Science	
86	1996	He, Z.; Staples, G., <i>et al.</i>	Fourteen Japanese quality tools in software process improvement	Scopus, Compendex	
87	1996	Hirsh, B. ; Theisinger, J.	Testing by the numbers	Web of Science	
88	1996	Hossain, A.; Choudhury, Z. A., <i>et al.</i>	Statistical process control of an industrial process in real time	Web of Science	
89	1996	Kohonen, T.; Oja, E., <i>et al.</i>	Engineering applications of the self-organizing map	Web of Science	
90	1996	Maier, M. W.	Integrated modeling: A unified approach to system engineering	Web of Science	
91	1996	Sheasby, M. A. ; Wilson, A. W.	Use of SPC in the papermaking industry	Compendex	
92	1996	Smart, J. C. ; Vemuri, V.	Interactive simulated annealing for complex dependency visualization	Web of Science	
93	1996	Thakur, A. ; Iyer, R. K.	Analyze-NOW - An environment for collection & analysis of failures in a network of workstations	Web of Science	
94	1996	Troczynski, T.	Teaching DOE for quality improvement	Compendex	
95	1996	Wang, C. L.; Bhat, P. B., <i>et al.</i>	High-performance computing for vision	Web of Science	
96	1996	Yung, W. K. C.	Integrated model for manufacturing process improvement	Compendex	
97	1997		Proceedings of the 1997 IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop	Scopus	
98	1997	Abumeri, G. H. ; Chamis, C. C.	T/BEST: A computer code for assessing the benefits of advanced aerospace technologies	Web of Science	
99	1997	Bassin, K. A. ; Santhanam, P.	Use of software triggers to evaluate software process effectiveness and capture customer usage profiles	IEEE	
100	1997	Bertolino, A.; Mirandola, R., <i>et al.</i>	A case study in branch testing automation	Web of Science	

101	1997	De Panfilis, S.; Kitchenham, B., <i>et al.</i>	Experiences introducing a measurement program	Compendex , Web of Science	
102	1997	Dean, B. R.; Hand, S., <i>et al.</i>	A cost based strategy for assessing improvements in manufacturing processes	Scopus, Compendex	
103	1997	Dumke, R. R. ; Grigoleit, H.	Efficiency of CAME tools in software quality assurance	Web of Science	
104	1997	Ferguson, R. A.; Martino, R. M., <i>et al.</i>	Data analysis methods for evaluating lithographic performance	Web of Science	
105	1997	Friedman, D. J.; Hansen, M. H., <i>et al.</i>	Model-free estimation of defect clustering in integrated circuit fabrication	Scopus, Compendex	
106	1997	Hardt, W. ; Rosenstiel, W.	Prototyping of tightly coupled hardware/software-systems	Web of Science	
107	1997	Hummel, J.; Azevedo, A., <i>et al.</i>	Annotating the Java bytecodes in support of optimization	Web of Science	
108	1997	Khoshgoftaar, T. M.; Allen, E. B., <i>et al.</i>	Process measures for predicting software quality	IEEE	
109	1997	Madachy, R. ; Khoshnevis, B.	Dynamic simulation modeling of an inspection-based software lifecycle process	Web of Science	
110	1997	Miller, J.; Daly, J., <i>et al.</i>	Statistical power and its subcomponents - Missing and misunderstood concepts in empirical software engineering research	Web of Science	
111	1997	Pillai, K. ; Nair, V. S. S.	Statistical analysis of nonstationary software metrics	Scopus, Compendex, Web of Science	
112	1997	Soares, J. ; Anderson, S.	Modeling process management in construction	Scopus	
113	1997	Vosniakos, G. ; Wang, J.	Software system framework for planning and operation of quality control in discrete part manufacturing	Scopus, Compendex	
114	1998	Averbuch, A.; Ioffe, L., <i>et al.</i>	Two-dimensional parallel solver for the solution of Navier-Stokes equations with constant and variable coefficients using ADI on cells	Web of Science	
115	1998	Cabodi, G.; Quer, S., <i>et al.</i>	Memory optimization in function and set manipulation with BDDs	Web of Science	
116	1998	Card, D. N.	Learning from our mistakes with defect causal analysis	Web of Science	
117	1998	Cwik, T.; Lou, J., <i>et al.</i>	Scalable, finite element analysis of electromagnetic scattering and radiation. Error	Web of Science	

			estimation and H-adaptivity		
118	1998	Eckhart, J. C.	Process improvement through statistical control	Compendex	
119	1998	Han, K. ; Ghosh, S.	A comparative analysis of virtual versus physical process migration strategies for distributed modeling and simulation of mobile computing networks	Web of Science	
120	1998	Hayes, W.	Using a Personal Software ProcessSM to improve performance	Scopus, Compendex, IEEE	
121	1998	Kerkhoff, J.; Eagar, T. W., <i>et al.</i>	A systematic strategy for optimizing manufacturing operations	Scopus	
122	1998	Lalli, V. R.	Space-system reliability: A historical perspective	Web of Science	
123	1998	Lederer, A. L. ; Prasad, J.	A causal model for software cost estimating error	Web of Science	
124	1998	Lee, C. S.	A software tool to enhance analytical performance evaluation technology	Web of Science	
125	1998	Li, S. Q.; Park, S. Y., <i>et al.</i>	SMAQ: A measurement-based tool for traffic modeling and queuing analysis part I: Design methodologies and software architecture	Web of Science	
126	1998	Mills, E. E.	Metrics in the software engineering curriculum	Web of Science	
127	1998	Mohan, R. V.; Tamma, K. K., <i>et al.</i>	Advanced manufacturing of large-scale composite structures: process modeling, manufacturing simulations and massively parallel computing platforms	Web of Science	
128	1998	Ollis, J.	Scientific approach to acceptance sampling of concrete pavement surfaces	Compendex	
129	1998	Ping, W. V.; Yang, Z., <i>et al.</i>	Development of procedure for automated segmentation of pavement rut data	Scopus	
130	1998	Rine, D. C. ; Sonnemann, R. M.	Investments in reusable software. A study of software reuse investment success factors	Web of Science	
131	1998	Thakur, R. P. S.; DeBoer, S. J., <i>et al.</i>	Process simplification in DRAM manufacturing	Scopus, Compendex	
132	1998	Udupa, G.; Singaperumal, M., <i>et al.</i>	Assessment of surface geometry using confocal scanning optical microscope	Web of Science	

133	1998	Watson, R. M.	Implementing self-managed process improvement teams in a continuous improvement environment	Scopus	
134	1999	Buchholz, P. ; Kemper, P.	Modular state level analysis of distributed systems techniques and tool support	Web of Science	
135	1999	Buglione, L. ; Abran, A.	Geometrical and statistical foundations of a three-dimensional model of software performance	Web of Science	
136	1999	Carleton, A. D.	Improving the software process with SPC	Scopus, Compendex	
137	1999	Chulani, S.; Boehm, B., <i>et al.</i>	Bayesian analysis of empirical software engineering cost models	Web of Science	
138	1999	Dalal, S. R.; Hamada, M. S., <i>et al.</i>	How to improve performance of software systems: A methodology and a case study for tuning performance	Web of Science	
139	1999	Fenton, N. E. ; Neil, M.	A critique of software defect prediction models	Web of Science	
140	1999	Hild, C.; Sanders, D., <i>et al.</i>	The thought map	Scopus	
141	1999	Lettieri, P.; Schurgers, C., <i>et al.</i>	Adaptive link layer strategies for energy efficient wireless networking	Web of Science	
142	1999	Lewis, N. D. C.	Continuous process improvement using Bayesian Belief Networks	Scopus, Compendex	2a etapa
143	1999	Lewis, N. D. C.	Assessing the evidence from the use of SPC in monitoring, predicting improving software quality	Scopus, Compendex	
144	1999	Lombardi, G.; Peciola, E., <i>et al.</i>	Towards statistical control of an industrial test process	Web of Science	
145	1999	Maciel, P.; Barros, E., <i>et al.</i>	A Petri net model for hardware/software codesign	Web of Science	
146	1999	Montano, C. B. ; Utter, G. H.	Total quality management in higher education	Compendex	
147	1999	Raffo, D. M.; Vandeville, J. V., <i>et al.</i>	Software process simulation to achieve higher CMM levels	Web of Science	
148	1999	Ryu, M.; Park, J., <i>et al.</i>	Performance re-engineering of embedded real-time systems	Web of Science	
149	1999	Sanders, J. E.; Goldstein, B. S., <i>et al.</i>	Image processing techniques for quantitative analysis of skin structures	Web of Science	
150	1999	Tai, A. T.; Alkalai, L., <i>et al.</i>	On-board preventive maintenance: a design-	Web of Science	

			oriented analytic study for long-life applications		
151	1999	Urban, S. D.; Fu, L., <i>et al.</i>	The implementation and evaluation of the use of CORBA in an engineering design application	Web of Science	
152	1999	Xie, M. ; Goh, T. N.	Statistical techniques for quality	Scopus	2a etapa (indisponível)
153	2000	Anderson, R. J.; Beame, P., <i>et al.</i>	Experiences with the application of symbolic model checking to the analysis of software specifications	Web of Science	
154	2000	Aversa, R.; Di Martino, B., <i>et al.</i>	Reducing parallel program simulation complexity by static analysis	Web of Science	
155	2000	Bauer, C. ; Scharl, A.	Quantitative evaluation of Web site content and structure	Web of Science	
156	2000	Chan, N. H. L. ; Mathiopoulos, P. T.	Efficient video transmission over correlated Nakagami fading channels for IS-95CDMA systems	Web of Science	
157	2000	Chavez, T.	A decision-analytic stopping rule for validation of commercial software systems	Web of Science	
158	2000	Choudhary, A.; Liao, W. K., <i>et al.</i>	Design, implementation and evaluation of parallel pipelined STAP on parallel computers	Web of Science	
159	2000	De Mast, J.; Schippers, W. A. J., <i>et al.</i>	Steps and strategies in process improvement	Scopus, Compendex	
160	2000	Dyba, T.	Improvisation in small software organizations	Scopus, Compendex	
161	2000	Egan, E. W.	Integration of reactive process modeling into semiconductor technology development	Web of Science	
162	2000	Fleury, M.; Downton, A. C., <i>et al.</i>	Pipelined parallelisation of automatic face inspection	Web of Science	
163	2000	Florac, W. A.; Carleton, A. D., <i>et al.</i>	Statistical process control: analyzing space shuttle onboard software process	IEEE	
164	2000	Garcia, J.; Entrialgo, J., <i>et al.</i>	Model-driven monitoring support for the multi-view performance analysis of parallel embedded applications	Web of Science	
165	2000	Goseva-Popstojanova, K. ; Trivedi, K. S.	Failure correlation in software reliability models	Web of Science	



166	2000	Hansen, C. K. ; Thyregod, P.	Analysis of integrated circuit fault data using generalized linear models	Scopus, Compendex	
167	2000	King, G. A.	Quality technique transfer: Manufacturing and software	Web of Science	
168	2000	Lee, J. B. ; Jhon, C. S.	Improving the execution efficiency of barrier synchronization in software DSM through static analysis	Web of Science	
169	2000	Litoiu, M.; Rolia, J., <i>et al.</i>	Designing process replication and activation: A quantitative approach	Web of Science	
170	2000	Lyons, A. ; Barlow, N.	Sustaining performance	Scopus, Compendex	
171	2000	Mizuno, O.; Kikuno, T., <i>et al.</i>	Statistical analysis of deviation of actual cost from estimated cost using actual project data	Scopus	
172	2000	Niles, K. S.	Design of experiments and other analytical process improvement implementation techniques	Scopus	
173	2000	Sanders, D. ; Hild, C. R.	Common myths about Six Sigma	Scopus	
174	2000	Van Voorthuysen, E. J. ; Platfoot, R. A.	A flexible data acquisition system to support process identification and characterization	Scopus, Compendex	
175	2000	Veeravalli, B. ; Barlas, G.	Access time minimization for distributed multimedia applications	Web of Science	
176	2001	Abernathy, D. P.	Avoiding unintended consequences: Managing "Anti-Ys" in a six sigma project	Scopus	
177	2001	Aquilani, F.; Balsamo, S., <i>et al.</i>	Performance analysis at the software architectural design level	Web of Science	
178	2001	Borror, C. M. ; Champ, C. W.	Statistical monitoring performance for startup operations in a feedback control system	Scopus, Compendex	
179	2001	Chowdhury, S.; Chowdhury, S. P., <i>et al.</i>	Transfer function estimation of turboalternator from practical operating data by spectral analysis	Web of Science	
180	2001	Hoopes, B. J. ; Triantis, K. P.	Efficiency performance, control charts, and process improvement: Complementary measurement and evaluation	Scopus, Compendex	

181	2001	Houston, D. X.; Ferreira, S., <i>et al.</i>	Behavioral characterization: finding and using the influential factors in software process simulation models	Web of Science	
182	2001	McCarthy, B. M. ; Stauffer, R.	Enhancing Six Sigma through simulation with iGrafx process for Six Sigma	Scopus, Compendex	
183	2001	McNeese, W. ; Marks, C.	The power of process management	Compendex	
184	2001	Mizuno, O.; Shimoda, D., <i>et al.</i>	Enhancing software project simulator toward risk prediction with cost estimation capability	Web of Science	
185	2001	Mockus, A. ; Weiss, D. M.	Globalization by chunking: A quantitative approach	Web of Science	
186	2001	Moritz, C. A.; Yeung, D., <i>et al.</i>	SimpleFit: A framework for analyzing design trade-offs in raw architectures	Web of Science	
187	2001	Pasquini, A.; Pistolesi, G., <i>et al.</i>	Reliability analysis of systems based on software and human resources	Web of Science	
188	2001	Pignatiello Jr, J. J. ; Samuel, T. R.	Estimation of the change point of a normal process mean in SPC applications	Scopus	
189	2001	Shepperd, M. ; Cartwright, M.	Predicting with sparse data	Web of Science	
190	2001	Sun, H. R.; Zang, X. Y., <i>et al.</i>	Performance of Broadcast and Unknown Server (BUS) in ATM LAN emulation	Web of Science	
191	2001	Waszkowycz, B.; Perkins, T. D. J., <i>et al.</i>	Large-scale virtual screening for discovering leads in the postgenomic era	Web of Science	
192	2001	Wolf, F.; Ernst, R., <i>et al.</i>	Path clustering in software timing analysis	Web of Science	
193	2002	Baghdadi, A.; Zergainoh, N. E., <i>et al.</i>	Combining a performance estimation methodology with a hardware/software codesign flow supporting multiprocessor systems	Web of Science	
194	2002	Bergman, B. L. S.; Fundin, A. P., <i>et al.</i>	Beyond root-cause analysis [product development process improvement]	IEEE	
195	2002	Bernardo, M.; Busi, N., <i>et al.</i>	Integrating TwoTowers and GreatSPN through a compact net semantics	Web of Science	
196	2002	Bisgaard, S.; Hoerl, R. W., <i>et al.</i>	Improving business processes with six sigma	Scopus, Compendex	
197	2002	Cortellessa, V. ; Grassi, V.	A performance-based methodology to early evaluate the effectiveness of mobile	Web of Science	

			software architectures		
198	2002	Cukic, B.; Taylor, B. J., <i>et al.</i>	Automated generation of test trajectories for embedded flight control systems	Web of Science	
199	2002	d'Astous, P. ; Robillard, P. N.	Empirical study of exchange patterns during software peer review meetings	Web of Science	
200	2002	Jardine, A. K. S.	Optimizing condition based maintenance decisions	Web of Science	
201	2002	Kumarakulasingam, P. ; Saiedian, H.	A framework for evaluating the effectiveness of real-time object-oriented models	Web of Science	
202	2002	Leszak, M.; Perry, D. E., <i>et al.</i>	Classification and evaluation of defects in a project retrospective	Web of Science	
203	2002	Nave, D.	How to compare six sigma, lean and the theory of constraints	Scopus	
204	2002	Nishi, Y. ; Iizuka, Y.	Design of stress testing focused on resource	Web of Science	
205	2002	Pechiar, J.; Perera, G., <i>et al.</i>	Effective bandwidth estimation and testing for Markov sources	Web of Science	
206	2002	Petriu, D. C. ; Shen, H.	Applying the UML performance profile: Graph grammar-based derivation of LQN models from UML specifications	Web of Science	
207	2002	Phanindra, V. S.; Murugappan, M., <i>et al.</i>	Process diagnostics	Scopus	2a etapa
208	2002	Ramalingam, G.; Warshavsky, A., <i>et al.</i>	Deriving specialized program analyses for certifying component-client conformance	Web of Science	
209	2002	Ruiz, M.; Ramos, I., <i>et al.</i>	Integrating dynamic models for CMM-based software process improvement	Web of Science	2a etapa (indisponível)
210	2002	Ruiz, M.; Ramos, I., <i>et al.</i>	A dynamic integrated framework for software process improvement	Web of Science	2a etapa
211	2002	Selic, B.	The emerging real-time UML standard	Web of Science	
212	2002	Smidts, C.; Xin, H., <i>et al.</i>	Producing reliable software: an experiment	Web of Science	
213	2002	So, S. S.; Cha, S. D., <i>et al.</i>	Empirical evaluation of a fuzzy logic-based software quality prediction model	Scopus, Web of Science	

214	2002	Ward, A.	Get a manufacturing six pack	Scopus, Compendex	
215	2002	Wooff, D. A.; Goldstein, M., <i>et al.</i>	Bayesian graphical models for software testing	Web of Science	
216	2002	Yang, Z.; Xie, M., <i>et al.</i>	On the performance of geometric charts with estimated control limits	Scopus	
217	2003	Attanasio, C.; Corradini, F., <i>et al.</i>	Static analysis of real-time component-based systems configurations	Web of Science	
218	2003	Baker, G. H.	Reliability analysis of multi element integrated testing in space systems	Web of Science	
219	2003	Bernardo, M. ; Bravetti, M.	Performance measure sensitive congruences for Markovian process algebras	Web of Science	
220	2003	Biffi, S. ; Halling, M.	Investigating the defect detection effectiveness, and cost benefit of nominal inspection teams	Web of Science	
221	2003	Blake, A. B.	A student-enacted simulation approach to software engineering education	Web of Science	
222	2003	Bossert, J.	Lean and six sigma - Synergy made in heaven	Scopus	
223	2003	Bouamama, B. O.	Bond graph approach as analysis tool in thermofluid model library conception	Web of Science	
224	2003	Carlson, J. S. ; Söderberg, R.	Assembly root cause analysis: A way to reduce dimensional variation in assembled products	Scopus	
225	2003	Chan, K. K. ; Spedding, T. A.	An integrated multidimensional process improvement methodology for manufacturing systems	Scopus, Compendex	
226	2003	Conteh, N. Y. ; Forgionne, G.	Delivering enhanced decision making support through intelligent just-in-time decision support systems	Scopus	
227	2003	Dawson, R. ; O'Neill, B.	Simple metrics for improving software process performance and capability: A case study	Web of Science	2a etapa (indisponível)
228	2003	El-Gayar, O. F.	Decision support for software projects: The role of SPC and simulation metamodeling	Scopus, Compendex	
229	2003	Hanebutte, N.; Taylor, C. S., <i>et al.</i>	Techniques of successful application of factor analysis in software measurement	Web of Science	

230	2003	Helminen, A. ; Pulkkinen, U.	Quantitative reliability estimation of a computer-based motor protection relay using Bayesian networks	Web of Science	
231	2003	Henderson-Sellers, B.; Bohling, J., <i>et al.</i>	Creating the OOSPICE model architecture - A case of reuse	Compendex	
232	2003	Hong, G. Y. ; Goh, T. N.	Six Sigma in software quality	Scopus	CS1
233	2003	Jorgensen, M.; Indahl, U., <i>et al.</i>	Software effort estimation by analogy and "regression toward the mean"	Web of Science	
234	2003	Kaufman, L. M.; Salinas, M. H., <i>et al.</i>	Integrate hardware/software device testing for use in a safety-critical application	Web of Science	
235	2003	Kou, G.; Liu, X. T., <i>et al.</i>	Multiple criteria linear programming approach to data mining: Models, algorithm designs and software development	Web of Science	
236	2003	Liukkonen, T. ; Tuominen, A.	Decreasing variation in paste printing using statistical process control	Web of Science	
237	2003	Mason, R. L.; Chou, Y. M., <i>et al.</i>	Systematic patterns in T <sup>2</sup> charts	Scopus	
238	2003	Nembhard, H. B.	Individual observation process monitoring charts for systems with response lags	Scopus, Compendex	
239	2003	Qin, X. Z. ; Lee, W.	Statistical causality analysis of INFOSEC alert data	Web of Science	
240	2003	Raffo, D. M. ; Setamanit, S.-O.	Supporting software process decisions using bi-directional simulation	Scopus, Compendex	
241	2003	Sahinoglu, M.	An empirical Bayesian stopping rule in testing and verification of behavioral models	Web of Science	
242	2003	Snee, R. D.	Eight essential tools	Scopus	
243	2003	Stensrud, E. ; Myrtveit, I.	Identifying high performance ERP projects	Web of Science	
244	2003	Succi, G.; Pedrycz, W., <i>et al.</i>	Practical assessment of the models for identification of defect-prone classes in object-oriented commercial systems using design metrics	Web of Science	
245	2003	Teruel, E.; Franceschinis, G., <i>et al.</i>	Well-defined generalized stochastic petri nets: A net-level method to specify priorities	Web of Science	
246	2003	Theelen, B. D.; Voeten, J. P. M., <i>et</i>	Performance modelling of a network processor using	Web of Science	

		<i>al.</i>	POOSL		
247	2003	Wong, W. E.; Sugeta, T., <i>et al.</i>	Coverage testing software architectural design in SDL	Web of Science	
248	2003	Yu, Y. Y. ; Johnson, B. W.	A BBN approach to certifying the reliability of COTS software systems	Web of Science	
249	2003	Zhang, C. ; Jacobsen, H. A.	Refactoring middleware with aspects	Web of Science	
250	2004	Alex, A.; Rose, J., <i>et al.</i>	Hardware accelerated novel protein identification	Web of Science	
251	2004	Anil, R.; Seshadri, V., <i>et al.</i>	A methodology for managing multi-disciplinary programs with six sigma approach	Scopus	
252	2004	Arul, K. ; Kohli, H.	Six sigma for software application of hypothesis tests to software data	Scopus, Web of Science	
253	2004	Ashayeri, A. ; Degrève, J.	Observational study in a semicontinuous polymer process	Scopus	
254	2004	Ashayeri, A. ; Degrève, J.	Quality control in a semi- continuous polymer production process	Scopus	
255	2004	Baldassarre, T.; Boffoli, N., <i>et al.</i>	Managing Software Process Improvement (SPI) through Statistical Process Control (SPC)	Scopus, Web of Science	2a etapa
256	2004	Balis, B.; Bubak, M., <i>et al.</i>	Adaptation of legacy software to grid services	Web of Science	
257	2004	Balsamo, S.; Di Marco, A., <i>et al.</i>	Model-based performance prediction in software development: A survey	Web of Science	
258	2004	Balsamo, S. ; Simeoni, M.	Integrating performance modeling in the software development process	Web of Science	
259	2004	Chen, H. F.; Zhou, Q., <i>et al.</i>	A fMRI data processing method using a new composite ICA algorithm	Web of Science	
260	2004	Cyr, G.; Bois, G., <i>et al.</i>	Generation of processor interface for SoC using standard communication protocol	Web of Science	
261	2004	Dassau, E. ; Lewin, D. R.	Integrating design and control with six-sigma for bioprocessing applications	Scopus	
262	2004	Gokhale, S. S.; Wong, W. E., <i>et al.</i>	An analytical approach to architecture-based software performance and reliability prediction	Web of Science	

263	2004	Kim, M.; Lee, Y. H., <i>et al.</i>	A systematic decision procedure to identify vital few causes of variation in large-scale petrochemical process	Scopus	
264	2004	Lee, Y. H.; Min, K. G., <i>et al.</i>	Process improvement methodology based on multivariate statistical analysis methods	Scopus, Compendex	
265	2004	Madan, B. B.; Goseva-Popstojanova, K., <i>et al.</i>	A method for modeling and quantifying the security attributes of intrusion tolerant systems	Web of Science	
266	2004	Mosterman, P. J.; Sztipanovits, J., <i>et al.</i>	Computer-automated multiparadigm modeling in control systems technology	Web of Science	
267	2004	Ozarin, N.	Failure modes and effects analysis during design of computer software	Web of Science	
268	2004	Pedersen, K.	Software thinking improvement learning performance improving lessons	Web of Science	
269	2004	Petrie, M. M. L.	Towards an engineering education capability Maturity Model	Scopus	
270	2004	Posadas, H.; Herrera, F., <i>et al.</i>	Single source design environment for embedded systems based on SystemC	Web of Science	
271	2004	Risse, T.; Aberer, K., <i>et al.</i>	Configuration of distributed message converter systems	Web of Science	
272	2004	Schwanke, R. W. ; Lutz, R. R.	Experience with the architectural design of a modest product family	Web of Science	
273	2004	Serrano, M. A.	State of the art and future of research in software process improvement	Scopus, Compendex	
274	2004	Singh, B.; Dhanavanthan, P., <i>et al.</i>	SESYS: Statistical expert systems for information management	Web of Science	
275	2004	Smith, C. L. ; Womack, J. B.	Raytheon assessment of PRISM (R) as a field failure prediction tool	Web of Science	
276	2004	Wang, F.; Schmidt, K., <i>et al.</i>	BDD-based safety-analysis of concurrent software with pointer data structures using graph automorphism symmetry reduction	Web of Science	
277	2004	Wellington, C.	Preparing Computer Science students to question and improve a software development process	Scopus, Compendex, IEEE	

278	2004	Xue, W.; Shu, J. W., <i>et al.</i>	Parallel transient stability simulation for national power grid of China	Web of Science	
279	2004	Yoon, K. A.; Min, S. Y., <i>et al.</i>	Model-based project process analysis using project tracking data	Web of Science	
280	2005	Baldassarre, M. T.; Boffoli, N., <i>et al.</i>	Improving dynamic calibration through statistical process control	IEEE	
281	2005	Brown, A. M.	A new software for carrying out one-way ANOVA post hoc tests	Web of Science	
282	2005	Buchholtz, M.; Gilmore, S., <i>et al.</i>	End-to-end integrated security and performance analysis on the DEGAS choreographer platform	Web of Science	
<b>283</b>	<b>2005</b>	<b>Caivano, D.</b>	<b>Continuous software process improvement through statistical process control</b>	<b>Scopus, Compendex, IEEE</b>	<b>CS1 (Controle)</b>
284	2005	Daniels, L.; Edgar, B., <i>et al.</i>	Using confidence intervals to compare process capability indices	Scopus, Compendex	
285	2005	Doblender, A.; Gossringer, D., <i>et al.</i>	An evaluation of model-based software synthesis from simulink models for embedded video applications	Web of Science	
286	2005	Han, J. S.; Rudnyi, E. B., <i>et al.</i>	Efficient optimisation of transient dynamic problems in MEMS devices using model order reduction	Web of Science	
287	2005	Hassan, A.; Goseva-Popstojanova, K., <i>et al.</i>	UML based severity analysis methodology	Web of Science	
288	2005	Houston, B. F.	Quantile-based metrics applied to process performance	Scopus	
289	2005	Huang, C. Y.	Performance analysis of software reliability growth models with testing-effort and change-point	Web of Science	
290	2005	Huang, Y. W.; Hsieh, B. Y., <i>et al.</i>	Analysis, fast algorithm, and VLSI architecture design for H.264/AVC intra frame coder	Web of Science	
291	2005	Li, P.; Veeravalli, B., <i>et al.</i>	Design and implementation of parallel video encoding strategies using divisible load analysis	Web of Science	
292	2005	Molokken-Ostvoid, K. ; Jorgensen, M.	Expert estimation of web-development projects: Are software professionals in technical roles more optimistic than those in non-technical roles?	Web of Science	



293	2005	Moussiades, L. ; Vakali, A.	PDetect: A clustering approach for detecting plagiarism in source code datasets	Web of Science	
294	2005	Nembhard, H. B. ; Kao, M. S.	Transition monitoring and adjustment for dynamic systems in a process improvement environment	Scopus	
295	2005	Okamura, H.; Miyahara, S., <i>et al.</i>	Rejuvenating communication network system under burst arrival circumstances	Web of Science	
296	2005	Perry, M. B. ; Pignatiello Jr, J. J.	Estimation of the change point of the process fraction nonconforming in SPC applications	Scopus	
297	2005	Pokam, G. ; Bodin, F.	Offline approach for whole-program paths analysis using suffix arrays	Web of Science	
298	2005	Qing, W. ; Mingshu, L.	Measuring and improving software process in China	IEEE	
299	2005	Stolberg, H. J.; Berekovic, M., <i>et al.</i>	A platform-independent methodology for performance estimation of multimedia signal processing applications	Web of Science	
300	2005	Tsantalis, N.; Chatzigeorgiou, A., <i>et al.</i>	Predicting the probability of change in object-oriented systems	Web of Science	
301	2005	Wang, Q. ; Li, M.	Measuring and improving software process in China	Scopus, Compendex	
302	2005	Weheba, G. S. ; Nickerson, D. M.	The economic design of $\bar{x}$ charts: A proactive approach	Scopus	
303	2005	Weheba, G. S. ; Nickerson, D. M.	The economic design of $\bar{x}$ charts: A proactive approach	Compendex	
304	2005	Wu, G. W. ; Yao, L.	A new WCET estimation algorithm based on instruction cache and prefetching combined model	Web of Science	
305	2006		Software Process Improvement - 13th European Conference, EuroSPI 2006, Proceedings	Scopus, Compendex	
306	2006	Andrews, J. H.; Briand, L. C., <i>et al.</i>	Using mutation analysis for assessing and comparing testing coverage criteria	Web of Science	
307	2006	Aversano, L.; Bodhuin, T., <i>et al.</i>	Technology-driven business evolution	Web of Science	
308	2006	Baschiroto, A.; Dallago, E., <i>et al.</i>	Development and comparative analysis of fluxgate magnetic sensor structures in PCB technology	Web of Science	

309	2006	Biro, M.; Deak, C., <i>et al.</i>	From compliance to business success: Improving outsourcing service controls by adopting external regulatory requirements	Scopus, Compendex	
310	2006	Boffoli, N.	Non-intrusive monitoring of software quality	Scopus, Compendex, IEEE	CS1
311	2006	Campos, J. ; Merseguer, J.	On the integration of UML and Petri nets in software development	Web of Science	
312	2006	Cha, J.; Kim, J., <i>et al.</i>	A highly efficient 3D micromixer using soft PDMS bonding	Web of Science	
313	2006	Choi, K. ; Bae, D. H.	Analysis of software-intensive system acquisition using hybrid software process simulation	Web of Science	
314	2006	Cornford, S. L.; Feather, M. S., <i>et al.</i>	DDP: A tool for life-cycle risk management	Web of Science	
315	2006	Cottrell, D. S.	Contractor process improvement for enhancing construction productivity	Scopus, Compendex	
316	2006	Degerli, Y.; Besancon, M., <i>et al.</i>	Performance of a fast binary readout CMOS active pixel sensor chip designed for charged particle detection	Web Of Science	
317	2006	Drigas, A.; Kouremenos, S., <i>et al.</i>	A study of H.263 traffic modeling in multipoint videoconference sessions over IP networks	Web of Science	
318	2006	Galanis, M. D.; Dimitroulakos, G., <i>et al.</i>	Performance improvements from partitioning applications to FPGA hardware in embedded SoCs	Web of Science	
319	2006	Golubic, S.; Galinac, T., <i>et al.</i>	Synergy between business and process improvement models	Scopus, Compendex	
320	2006	Gong, B. ; He, X.	A framework of measuring, analyzing, diagnosing and improving a single software process	Scopus, Compendex, Web of Science	
321	2006	Gong, B.; Li, Z., <i>et al.</i>	Software process quality and analysis method	Scopus, Compendex, Web of Science	
322	2006	Griffiths, D. R. ; Sykulski, J. K.	Automation of finite element aided design of induction motors using multi-slice 2D models	Web of Science	
323	2006	Günther, C. W.; Rinderle, S., <i>et al.</i>	Change mining in adaptive process management systems	Scopus	

324	2006	Hamann, A.; Jersak, M., <i>et al.</i>	A framework for modular analysis and exploration of heterogeneous embedded systems	Web of Science	
325	2006	Karlsson, L.; Regnell, B., <i>et al.</i>	Case studies in process improvement through retrospective analysis of release planning decisions	Web of Science	
326	2006	Kolb, R.; Muthig, D., <i>et al.</i>	Refactoring a legacy component for reuse in a software product line: a case study	Web of Science	
327	2006	Komuro, M.	Experiences of applying SPC techniques to software development processes	Scopus	
328	2006	Kwong-Fu, H. ; Wilson, R. K.	Quality interaction between mission assurance and project team members (Practice for Mission Assurance and other Operations Projects)	Scopus	
329	2006	Lera, I.; Juiz, C., <i>et al.</i>	Performance-related ontologies and semantic web applications for on-line performance assessment of intelligent systems	Web of Science	
330	2006	Miles, E. N.	Improvement in the incident reporting and investigation procedures using process excellence (DMAI <sup>2</sup> C) methodology	Scopus	
331	2006	Miller, S. D.; DeCarlo, R. A., <i>et al.</i>	A control-theoretic approach to the management of the software system test phase	Web of Science	
332	2006	Parthasarathy, S. ; Anbazhagan, N.	Significance of software metrics in ERP projects	Scopus, Compendex, IEEE	
333	2006	Perez-Minana, E. ; Gras, J. J.	Improving fault prediction using Bayesian networks for the development of embedded software applications	Web of Science	
334	2006	Pitard, F. F.	Chronostatistics - A powerful, pragmatic, new science for metallurgists	Scopus	
335	2006	Seah, Y. X.; Palaniappan, M., <i>et al.</i>	Applications of soft defect localization (SDL) on AMD advanced SOI microprocessors	Scopus	
336	2006	Simion, C. ; Borza, S.	Inspection capability's analysis	Scopus	
337	2006	Tarhan, A. ; Demirors, O.	Investigating suitability of software process and metrics for statistical process control	Scopus, Compendex	2a etapa (indisponível)

338	2006	Verho, P.; Jarventausta, P., <i>et al.</i>	Applying reliability analysis in evaluation of life-cycle costs of alternative network solutions	Web of Science	
<b>339</b>	<b>2006</b>	<b>Wang, Q.; Jiang, N., <i>et al.</i></b>	<b>BSR: A statistic-based approach for establishing and refining software process performance baseline</b>	<b>Scopus, Compendex</b>	<b>CS2 (Controle)</b>
340	2006	Wangdee, W. ; Billinton, R.	Impact of load shedding philosophies on bulk electric system reliability analysis using sequential Monte Carlo simulation	Web of Science	
341	2006	Zhang, N. ; Wunsch, D. C.	Speeding up VLSI layout verification using fuzzy attributed graphs approach	Web of Science	
342	2006	Zhang, Y. ; Sheth, D.	Mining software repositories for model-driven development	IEEE	
343	2006	Zhang, Y. F. ; Sheth, D.	Mining software repositories for model-driven development	Web of Science	
344	2007	Alagarsamy, K.; Justus, S., <i>et al.</i>	The knowledge based software process improvement program: A rational analysis	Scopus	
345	2007	Chen, X.; Sorenson, P., <i>et al.</i>	Continuous SPA: Continuous assessing and monitoring software process	Scopus, Compendex	
346	2007	Cortellessa, V.; Pierini, P., <i>et al.</i>	Integrating software models and platform models for performance analysis	Web of Science	
347	2007	Grigg, N. ; Walls, L.	The role of control charts in promoting organisational learning: New perspectives from a food industry study	Scopus	
348	2007	Gupta, M.; Fu, J. C., <i>et al.</i>	Rapid goal-oriented automated software testing using MEA-graph planning	Web of Science	
349	2007	Hall, T.; Jagielska, D., <i>et al.</i>	Motivating developer performance to improve project outcomes in a high maturity organization	Web of Science	
350	2007	Jalote, P.; Mittal, A. K., <i>et al.</i>	On optimum module size for software inspections	Scopus	
351	2007	Jiang, L. T. ; Xu, G. Z.	Modeling and analysis of software aging and software failure	Web of Science	
352	2007	Karyotis, V. ; Papavassiliou, S.	Risk-based attack strategies for mobile ad hoc networks under probabilistic attack modeling framework	Web of Science	

353	2007	Lofgreen, D. D.; Vilela, M. F., <i>et al.</i>	Improved defect and Fourier transform infrared Spectroscopy analysis for prediction of yield for HgCdTe multilayer heterostructures	Web of Science	
354	2007	Pai, G. J. ; Dugan, J. B.	Empirical analysis of software fault content and fault proneness using Bayesian methods	Web of Science	
355	2007	Park, Y.; Choi, H., <i>et al.</i>	A framework for the use of Six Sigma tools in PSP/TSP	Scopus, Compendex, IEEE	CS1
356	2007	Perry, M. B.; Pignatiello Jr, J. J., <i>et al.</i>	Change point estimation for monotonically changing Poisson rates in SPC	Scopus	
357	2007	Perry, M. B.; Pignatiello Jr, J. J., <i>et al.</i>	A magnitude-robust control chart for monitoring and estimating step changes in a poisson rate parameter	Scopus	
358	2007	Rodriguez, A.; Fernandez-Medina, E., <i>et al.</i>	An MDA approach to develop secure business processes through a UML 2.0 extension	Web of Science	
359	2007	Sandberg, E.; Lennox, B., <i>et al.</i>	Scrap management by statistical evaluation of EAF process data	Web of Science	
360	2007	Seliya, N. ; Khoshgoftaar, T. M.	Software quality estimation with limited fault data: a semi-supervised learning perspective	Web of Science	
361	2007	Seliya, N. ; Khoshgoftaar, T. M.	Software quality analysis of unlabeled program modules with semisupervised clustering	Web of Science	
362	2007	Staudhammer, C.; Maness, T. C., <i>et al.</i>	Profile charts for monitoring lumber manufacturing using laser range sensor data	Scopus	
363	2007	Sugihara, M.; Nakamura, K., <i>et al.</i>	Technology mapping technique for increasing throughput of character projection lithography	Web of Science	
364	2007	Sutherland, J.; Jakobsen, C. R., <i>et al.</i>	Scrum and CMMI level 5: The magic potion for code warriors	Scopus, IEEE	
365	2007	Tao, Y. Y.; Dong, Z., <i>et al.</i>	Application of the integrated SPC/PI strategy in discrete manufacturing systems	Scopus, Compendex	
366	2007	Tao, Y.-y.; Dong, Z., <i>et al.</i>	Application of the Integrated SPC/PI Strategy in Discrete Manufacturing Systems	IEEE	
367	2007	Thomashefsky, P.	Getting to the root of the matter	Scopus	

368	2007	Tseng, W. H. ; Fan, C. F.	BBN construction for software process tailoring	Web of Science	
369	2007	Van De Weerd, I.; Brinkkemper, S., <i>et al.</i>	Concepts for incremental method evolution: Empirical exploration and validation in requirements management	Scopus, Compendex	
370	2007	Wang, Q.; Gou, L., <i>et al.</i>	An empirical study on establishing quantitative management model for testing process	Scopus, Compendex	
371	2007	Wilson, S. P. ; Samaniego, F. J.	Nonparametric analysis of the order-statistic model in software reliability	Web of Science	
372	2007	Zhu, Y. D.; Wong, Y. S., <i>et al.</i>	Framework of a computer-aided short-run SPC planning system	Scopus, Compendex	
373	2008	Baldassarre, M. T.; Boffoli, N., <i>et al.</i>	A hands-on approach for teaching systematic review	Scopus, Compendex	
374	2008	Boffoli, N.; Bruno, G., <i>et al.</i>	Statistical process control for software: A systematic approach	Scopus, Compendex	2a etapa
375	2008	Brinkkemper, S.; Van De Weerd, I., <i>et al.</i>	Process improvement in requirements management: A method engineering approach	Scopus	
376	2008	Buyukozkan, G. ; Ruan, D.	Evaluation of software development projects using a fuzzy multi-criteria decision approach	Web of Science	
377	2008	Caballero, I.; Caro, A., <i>et al.</i>	IQM3: Information quality management maturity model	Scopus	
378	2008	Calimeri, F.; Perri, S., <i>et al.</i>	Experimenting with parallelism for the instantiation of ASP programs	Web of Science	
<b>379</b>	<b>2008</b>	<b>Chang, C. P. ; Chu, C. P.</b>	<b>Improvement of causal analysis using multivariate statistical process control</b>	<b>Scopus, Web of Science</b>	<b>CS2 (Controle)</b>
380	2008	Chen, T.; Ma, J., <i>et al.</i>	A virus detection framework based on SPMOS	Scopus, Compendex	
381	2008	Chow, T. ; Cao, D. B.	A survey study of critical success factors in agile software projects	Web of Science	
382	2008	De Espindola, R. S.; Luciano, E. M., <i>et al.</i>	An overview of software process quality instruments adoption at Brazil preliminary results of a survey	Scopus, Compendex	
383	2008	Del Rosso, C.	Software performance tuning of software product family architectures: Two case studies in the real-time embedded systems domain	Web of Science	

384	2008	Dudek, D.	The APS Framework For Incremental Learning of Software Agents	Web of Science	
385	2008	Eaddy, M.; Zimmermann, T., <i>et al.</i>	Do crosscutting concerns cause defects?	Web of Science	
386	2008	Fenton, N.; Neil, M., <i>et al.</i>	On the effectiveness of early life cycle defect prediction with Bayesian Nets	Web of Science	
<b>387</b>	<b>2008</b>	<b>Gonçalves, F. M. G. S.; Bezerra, C. I. M., <i>et al.</i></b>	<b>A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach</b>	<b>Scopus, Compendex , IEEE</b>	<b>CS1 (Controle)</b>
388	2008	Gou, L.; Wang, Q., <i>et al.</i>	Quantitatively managing defects for iterative projects: An industrial experience report in China	Scopus	
389	2008	Goudarzi, M.; Ishihara, T., <i>et al.</i>	A software technique to improve lifetime of caches containing ultra-leaky SRAM cells caused by within-die V <sub>t</sub> variation	Web of Science	
390	2008	Guo, X.; Hii, R., <i>et al.</i>	Influence of substrates and pretreatment on adhesion contact angle as a method for litho process improvement	Compendex	
391	2008	Haider, S. W.; Cangussu, J. W., <i>et al.</i>	Estimation of Defects based on Defect Decay Model: (EDM)-M-3	Web of Science	
392	2008	Hao, W.; Haiqing, W., <i>et al.</i>	Software productivity analysis with CSBSG data set	Scopus, Compendex, IEEE	
393	2008	Jang, G. W.; Kim, K. J., <i>et al.</i>	Integrated topology and shape optimization software for compliant MEMS mechanism design	Web of Science	
394	2008	Jost, A. C.	Small project survival among the CMMI level 5 big processes	Scopus, Compendex	
<b>395</b>	<b>2008</b>	<b>Kalinowski, M.; Travassos, G. H., <i>et al.</i></b>	<b>Towards a defect prevention based process improvement approach</b>	<b>Scopus, Compendex, IEEE</b>	<b>CS1 (Controle)</b>
396	2008	Kojima, T.; Hasegawa, T., <i>et al.</i>	Risk analysis of software process measurements	Scopus, Web of Science	2a etapa
397	2008	Komuro, M. ; Komoda, N.	An Explanation Model for Quality Improvement Effect of Peer Reviews	IEEE	
398	2008	Marculescu, D. ; Garg, S.	Process-driven variability analysis of single and multiple voltage-frequency island	Web of Science	

			latency-constrained systems		
399	2008	Mohammadzadeh, N.; Hessabi, S., <i>et al.</i>	A Framework for Object-Oriented Embedded System Development Based on Oo-Asips	Web of Science	
400	2008	Naing, T.; Zainuddin, Y., <i>et al.</i>	Determinants of information system adoptions in private hospitals in Malaysia	Scopus, Compendex	
401	2008	Olague, H. M.; Etzkorn, L. H., <i>et al.</i>	An empirical validation of object-oriented class complexity metrics and their ability to predict error-prone classes in highly iterative, or agile, software: a case study	Web of Science	
402	2008	Ravichandar, R.; Arthur, J. D., <i>et al.</i>	Improving change tolerance through Capabilities-based design: an empirical analysis	Web of Science	
403	2008	Tarhan, A. ; Demirors, O.	Assessment of software process and metrics to support quantitative understanding	Scopus, Compendex	CS3
404	2008	Wang, Q.; Gou, L., <i>et al.</i>	Estimating fixing effort and schedule based on defect injection distribution	Compendex	
405	2008	Xiaoguang, Y.; Xiaogang, W., <i>et al.</i>	Research on organizational-level software process improvement model and its implementation	Scopus, Compendex	
406	2008	Yan, X.; Wang, X., <i>et al.</i>	Research on Organizational-Level Software Process Improvement Model and Its Implementation	IEEE	
407	2009		Product-Focused Software Process Improvement: 10th International Conference, PROFES 2009, Proceedings	Compendex	
408	2009		Standard Classification for Software Anomalies	IEEE	
409	2009		Draft Standard for IEEE Standard Classification for Software Anomalies	IEEE	
410	2009	Alshathry, O.; Janicke, H., <i>et al.</i>	Quantitative Quality Assurance Approach	Scopus, Compendex, IEEE	
411	2009	Alyahya, M. A.; Ahmad, R., <i>et al.</i>	Effect of Cmmi-Based Software Process Maturity on Software Schedule Estimation	Web of Science	
412	2009	Baldassarre, M. T.; Boffoli, N., <i>et al.</i>	Statistically based process monitoring: Lessons from the trench	Scopus, Compendex	



413	2009	Belotti, P.; Lee, J., <i>et al.</i>	Branching and bounds tightening techniques for non-convex MINLP	Web of Science	
414	2009	Bjrnsen, F. O.; Wang, A. I., <i>et al.</i>	Improving the effectiveness of root cause analysis in post mortem analysis: A controlled experiment	Scopus, Compendex, Web of Science	CS3
415	2009	Chakraborty, S.; Mitra, T., <i>et al.</i>	Cache-aware timing analysis of streaming applications	Web of Science	
416	2009	Chang, C.-P. ; Chu, C.-P.	Software defect prediction using intertransaction association rule mining	Scopus, Compendex, Web of Science	
417	2009	Chang, C.-P.; Chu, C.-P., <i>et al.</i>	Integrating in-process software defect prediction with association mining to discover defect pattern	Scopus, Web of Science	
418	2009	Eichhorn, D.; Koschmider, A., <i>et al.</i>	3D support for business process simulation	Scopus	
419	2009	Fengyu, Z.; Xin, P., <i>et al.</i>	Software Development Process Monitoring Based on Nominal Transformation	IEEE	
420	2009	Fluri, B.; Wursch, M., <i>et al.</i>	Analyzing the co-evolution of comments and source code	Web of Science	
421	2009	Gherbi, A. ; Khendek, F.	From UML/SPT models to schedulability analysis: approach and a prototype implementation using ATL	Web of Science	
422	2009	Gou, L.; Wang, Q., <i>et al.</i>	Quantitative defects management in iterative development with BiDefect	Scopus	
423	2009	Gu, R. R.; Janneck, J. W., <i>et al.</i>	Exploring the Concurrency of an MPEG RVC Decoder Based on Dataflow Program Analysis	Web of Science	
424	2009	Gusev, G. ; Peftieva, T.	Causal analysis: Rational simplicity works	IEEE	
425	2009	Haggarty, O. J.; Knottenbelt, W. J., <i>et al.</i>	Distributed Response Time Analysis of GSPN Models with MapReduce	Web of Science	
426	2009	Kaur, A.; Sandhu, P. S., <i>et al.</i>	Early Software Fault Prediction Using Real Time Defect Data	IEEE	
427	2009	Lee, S. H.; Choi, H. J., <i>et al.</i>	Analysis of ontology development methodology based on and CMMI level 4	Scopus, Compendex	
428	2009	Lin, C. T. ; Huang, C. Y.	Staffing Level and Cost Analyses for Software Debugging Activities Through Rate-Based Simulation	Web of Science	

			Approaches		
429	2009	Napier, N. P.; Mathiassen, L., <i>et al.</i>	Combining Perceptions and Prescriptions in Requirements Engineering Process Assessment: An Industrial Case Study	IEEE	
430	2009	Niazi, M. ; Babar, M. A.	Identifying high perceived value practices of CMMI level 2: An empirical study	Web of Science	
431	2009	Ranjan, P. ; Misra, A. K.	A Novel Approach of Requirement Gathering and Analysis for Agent Oriented Software Engineering (Aose)	Web of Science	
432	2009	Sang Hun, L.; Ho-Jin, C., <i>et al.</i>	Analysis of ontology development methodology based on OTK and CMMI level 4	IEEE	
433	2009	Selby, R. W.	Synthesis, analysis, and modeling of large-scale mission-critical embedded software systems	Compendex	
434	2009	Shaikh, A.; Ahmed, A., <i>et al.</i>	Strengths and weaknesses of maturity driven process improvement effort	Scopus, Compendex, IEEE	
435	2009	Shao, Y. E. ; Hsu, B.-S.	Determining the contributors for a multivariate SPC chart signal using artificial neural networks and support vector machine	Compendex	
436	2009	Shao, Y. E. ; Hsu, B. S.	Determining the contributors for a multivariate SPC chart signal using artificial neural networks and support vector machine	Scopus	
437	2009	Silaghi, G. C.; Araujo, F., <i>et al.</i>	Defeating Colluding Nodes in Desktop Grid Computing Platforms	Web of Science	
438	2009	Stoyanov, S.; Bailey, C., <i>et al.</i>	Optimisation modelling for thermal fatigue reliability of lead-free interconnects in fine-pitch flip-chip packaging	Web of Science	
439	2009	Tessier, J.; Zwirz, T. G., <i>et al.</i>	Multivariate statistical process monitoring of reduction cells	Compendex	
440	2009	Tu, K. K.-W.; Lee, J. C.-S., <i>et al.</i>	A novel statistical method for automatically partitioning tools according to engineers' tolerance control in process improvement	Scopus, Compendex	
441	2009	Turhan, B.; Menzies, T., <i>et al.</i>	On the relative value of cross-company and within-company data for defect prediction	Web of Science	

442	2009	Vaupel, T.	Spectral Domain Techniques for the Fast and Accurate Solution of Larger Microwave and Thin Layer Structures	Web of Science	
443	2009	Woodside, M.; Petriu, D. C., <i>et al.</i>	Performance analysis of security aspects by weaving scenarios extracted from UML models	Web of Science	
444	2009	Wu, I. L. ; Hung, C. Y.	A strategy-based process for effectively determining system requirements in eCRM development	Web of Science	
445	2009	Zhang, S.; Wang, Y.-J., <i>et al.</i>	Personal software process capability assessment method	Scopus, Compendex	
446	2009	Zhao, F.; Peng, X., <i>et al.</i>	Software development process monitoring based on nominal transformation	Scopus, Compendex	
447	2010	Agrawal, A.; Finnie, G., <i>et al.</i>	A General Framework to Measure Organizational Risk during Information Systems Evolution and its Customization	Web of Science	
448	2010	Al Yahya, M.; Ahmad, R., <i>et al.</i>	Impact of CMMI Based Software Process Maturity on COCOMO II's Effort Estimation	Web of Science	
449	2010	Atashgar, K. ; Noorossana, R.	An integrating approach to root cause analysis of a bivariate mean vector with a linear trend disturbance	Scopus	
450	2010	Bause, F.; Buchholz, P., <i>et al.</i>	A Simulation Environment for Hierarchical Process Chains Based on OMNeT plus	Web of Science	
451	2010	Becker, C. ; Rauber, A.	Improving component selection and monitoring with controlled experimentation and automated measurements	Web of Science	
452	2010	Casamayor, A.; Godoy, D., <i>et al.</i>	Identification of non-functional requirements in textual specifications: A semi-supervised learning approach	Web of Science	
453	2010	Chen, M. K. ; Wang, S. C.	The critical factors of success for information service industry in developing international market: Using analytic hierarchy process (AHP) approach	Web of Science	
454	2010	Ferrari, P.; Flammini, A., <i>et al.</i>	On the Implementation and Performance Assessment of a WirelessHART Distributed Packet Analyzer	Web of Science	

455	2010	Finkler, U.	An Analytic Framework for Detailed Resource Profiling in Large and Parallel Programs and Its Application for Memory Use	Web of Science	
456	2010	Garousi, V.	Experience and challenges with UML-driven performance engineering of a Distributed Real-Time System	Web of Science	
457	2010	Godet-Bar, G.; Rieu, D., <i>et al.</i>	HCI and business practices in a collaborative method for augmented reality systems	Web of Science	
458	2010	Guzman, J. G.; Mitre, H. A., <i>et al.</i>	Integration of strategic management, process improvement and quantitative measurement for managing the competitiveness of software engineering organizations	Web of Science	2a etapa
459	2010	Han, D.; Kim, B. S., <i>et al.</i>	DSP-Driven Self-Tuning of RF Circuits for Process-Induced Performance Variability	Web of Science	
460	2010	Happe, J.; Becker, S., <i>et al.</i>	Parametric performance completions for model-driven performance prediction	Web of Science	
461	2010	Hicks, B. J. ; Matthews, J.	The barriers to realising sustainable process improvement: A root cause analysis of paradigms for manufacturing systems improvement	Scopus	
462	2010	Kim, T. H.; Bang, H. J., <i>et al.</i>	A systematic representation of path constraints for implicit path enumeration technique	Web of Science	
463	2010	Koziolek, H.	Performance evaluation of component-based software systems: A survey	Web of Science	
464	2010	Lide, C. ; Glista, S.	An avionics integrity program approach to preventing aircraft electronics humidity and moisture problems	Scopus	
465	2010	Llamosa-Villalba, R. ; Aceros, S.	Process Management Model for Higher Education: Improvement of educational programs in software quality	IEEE	
466	2010	Llamosa-Villalba, R. ; Méndez Aceros, S. E.	Process management model for higher education: Improvement of educational programs in software quality	Scopus, Compendex	
467	2010	Mittas, N. ; Angelis, L.	LSEbA: least squares regression and estimation by analogy in a semi-parametric model for software cost	Web of Science	

			estimation		
468	2010	Mittas, N. ; Angelis, L.	Visual comparison of software cost estimation models by regression error characteristic analysis	Web of Science	
469	2010	Moreno, G. A. ; Smith, C. U.	Performance analysis of real-time component architectures: An enhanced model interchange approach	Web of Science	
470	2010	Mucci, P. J. ; Mohan, T.	An Open Source performance tools software suite for scientific computing	Web of Science	
471	2010	Sandhu, P. S.; Goel, R., <i>et al.</i>	A model for early prediction of faults in software systems	IEEE	
472	2010	Secretan, J.; Georgiopoulos, M., <i>et al.</i>	APHID: An architecture for private, high-performance integrated data mining	Web of Science	
473	2010	Seng, J. L. ; Lin, M. I.	Benchmarking intelligent information integration - A generic construct-based model	Web of Science	
474	2010	Zhang, P. C.; Muccini, H., <i>et al.</i>	A classification and comparison of model checking software architecture techniques	Web of Science	

## ii) Informações extraídas das publicações selecionadas

Nesta seção, são apresentadas as informações extraídas das 11 publicações, inclusive os 4 artigos de controle, selecionadas a partir da execução do protocolo deste estudo baseado em revisão sistemática, descritas na seção I.C.v).

<b>Título</b>	Assessment of software process and metrics to support quantitative understanding
<b>Autor(es)</b>	Tarhan, Ayca; Demirors, Onur
<b>Ano da publicação</b>	2007
<b>Referência completa</b>	Tarhan, A. and O. Demirors. <i>Assessment of software process and metrics to support quantitative understanding</i> . in <i>International Conference on Software Process and Product Measurement, IWSM-Mensura 2007, November 5, 2007 - November 8, 2007</i> . 2008. Palma de Mallorca, Spain: Springer Verlag.
<b>Contexto</b>	Gerência quantitativa
<b>Resumo da publicação</b>	

O artigo propõe uma abordagem para avaliar a adequação de processo de software e medidas para a gerência quantitativa, investigando os atributos de processo e fatores externos, assim como características de usabilidade das métricas. Estudos de avaliação da abordagem são apresentados, nos quais foram utilizadas ferramentas de controle estatístico de processo (SPC) para avaliação.
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>
Não há
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>
A abordagem inclui um processo de avaliação e um modelo de avaliação que define os ativos para avaliar o processo e as medidas. Além disso, uma ferramenta apoia o processo de avaliação.
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>
Não há

<b>Título</b>	A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach
<b>Autor(es)</b>	Gonçalves, F. M. G. S.; Bezerra, C. I. M.; Belchior, A. D.; Coelho, C. C.; Pires, C. G. S.
<b>Ano da publicação</b>	2008
<b>Referência completa</b>	Gonçalves, F.M.G.S., et al. <i>A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach</i> . 2008.
<b>Contexto</b>	Organizações de software de alta maturidade
<b>Resumo da publicação</b>	
O artigo propõe uma estratégia para identificar, priorizar e classificar ações de melhorias e inovação utilizando ferramentas do Seis Sigma para organizações na alta maturidade (CMMI nível 5).	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
A estratégia é composta pelos seguintes passos:	
<ol style="list-style-type: none"> <li>1. <u>Identificação</u>: analisa-se se os processos organizacionais estão alinhados com os objetivos de negócio. Identifica-se a voz da organização (do processo, dos funcionários, da inovação e dos clientes);</li> <li>2. <u>Classificação</u>: Classifica-se, de acordo com a complexidade, e investiga-se as causas dos problemas;</li> <li>3. <u>Priorização</u>: Identifica-se as melhorias com a melhor relação custo/benefício;</li> </ol>	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
Não há	

<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>
Não há

<b>Título</b>	A framework for the use of Six Sigma tools in PSP/TSP
<b>Autor(es)</b>	Park, Y., H. Choi
<b>Ano da publicação</b>	2007
<b>Referência completa</b>	Park, Y., H. Choi, and J. Baik. <i>A framework for the use of Six Sigma tools in PSP/TSP</i> . in <i>SERA 2007: Fifth ACIS International Conference on Software Engineering Research, Management, and Applications, August 20, 2007 - August 22, 2007</i> . 2007. Busan, Korea, Republic of: Inst. of Elec. and Elec. Eng. Computer Society
<b>Contexto</b>	PST/TSP
<b>Resumo da publicação</b>	
<p>O PSP (<i>Personal Software Process</i>) e TSP (<i>Team Software Process</i>) foram desenvolvidos pelos SEI (<i>Software Engineering Institute</i>) como abordagens para auxiliar o processo de melhoria de processo em organizações de software.</p> <p>O PSP/TSP, no entanto, não contemplam ferramentas de análise estatística para apoiar o controle de processo. O artigo propõe um <i>framework</i> de ferramentas para Seis Sigma aplicadas a processo de software e ao PSP/TSP.</p>	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
O framework proposto relaciona, em uma tabela, relaciona a fase no PSP/TSP ( <i>Planning, Development, Postmortem, Team Launch, Development Strategy, Development Plan, Requirements Development, Design, Implementation, Integration and System Test</i> ) e passo à ferramenta Seis Sigma.	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
Não há	
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>	
Não há	

<b>Título</b>	BSR: A statistic-based approach for establishing and refining software process performance baseline
<b>Autor(es)</b>	Qing Wang, Nan Jiang, Lang Gou, Xia Liu, Mingshu Li, Yongi Wang
<b>Ano da publicação</b>	2006
<b>Referência completa</b>	Wang, Q., et al. <i>BSR: A statistic-based approach for establishing and refining software process performance baseline</i> . in <i>28th International Conference on Software Engineering 2006, ICSE '06, May 20, 2006 - May 28, 2006</i> . 2006. Shanghai, China: Inst. of Elec. and Elec. Eng. Computer

	Society.
<b>Contexto</b>	Indústria de Software
<b>Resumo da publicação</b>	
BSR é uma abordagem para estabelecer e refinar o <i>baseline</i> de desempenho dos processos. Os passos da abordagem utilizam SPC, e outras ferramentas de análise estatística para identificar os problemas e análise de causa para identificar as causas-raiz de problema, a fim de melhorar o processo. O artigo também descreve experiências com três organizações.	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
<p>A abordagem é composta por:</p> <ol style="list-style-type: none"> <li>1. Identificar objetivos quantitativos dos processos: Determinar quais objetivos quantitativos deverão nortear as ações de melhoria. Deve-se observar os princípios de <i>Goal-Driven</i>, Determinação (<i>Determinateness</i>) e Estabilidade (SPC);</li> <li>2. Coletar dados e construir amostras de dados: Após utilizar alguma estratégia (como o paradigma GQM) para definir as métricas, deve-se levantar uma quantidade considerável de dados para se ter um <i>insight</i> do desempenho do processo;</li> <li>3. Estimar <i>baselines</i> de desempenho dos processos: Análise da amostra de dados coletados para montar os diagramas (<i>charts</i>) para análise;</li> <li>4. Análise causal da instabilidade dos desempenhos dos processos: analisar as causas dos problemas evidenciados através de ferramentas como diagrama de Pareto, diagrama de causa-e-efeito, diagrama <i>Scatter</i> etc;</li> <li>5. Estabelecer <i>baselines</i> dos desempenhos dos processos: As ações de melhorias devem tentar minimizar as variações, fazendo com que os pontos estejam dentro dos limites (<math>u+\sigma</math>). Após uma estabilidade atingida, deve-se estabelecer o <i>baseline</i>;</li> <li>6. Refinar <i>baselines</i> dos desempenhos dos processos: Para refinar o desempenho do processo, deve-se analisar as variações evidenciadas pelo diagrama de control 3-sigmas de Shewhart e classificadas de acordo com critérios de análises definidos pelo autor</li> </ol>	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
O primeiro passo “Definir objetivos quantitativos” utilize princípios GQM e <i>Goal-Driven</i> para determinar quais informações devem ser reunidas para auxiliar o processo de investigação da melhoria do processo.	
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>	
Não há	
<b>Título</b>	Improvement of causal analysis using multivariate statistical process control



<b>Autor(es)</b>	Ching-Pao Chang, Chih-Ping Chu
<b>Ano da publicação</b>	2008
<b>Referência completa</b>	Chang, C.P. and C.P. Chu, <i>Improvement of causal analysis using multivariate statistical process control</i> . Software Quality Journal, 2008. <b>16</b> (3): p. 377-409.
<b>Contexto</b>	Processos de software
<b>Resumo da publicação</b>	
<p>Abordagem para identificação de problema em processos quantitativamente gerenciados, considerando uma visão estatística do processo. O diferencial da abordagem é a utilização de testes estatísticos para verificar correções entre as medições (que sejam correlacionadas) e descobrir problemas relacionados que não seriam identificados com a análise individual das medições.</p> <p>As correlações entre as variáveis não são consideradas na observação do processo de software, podendo causar uma mal interpretação dos sinais fora do controle e resultando numa aplicação de ações corretivas não apropriadas no processo. É proposto uma abordagem <i>Multilevel Software Cause Identification</i> (MSCI) e um estudo de caso é rodado para analisar a aplicabilidade da abordagem. Ameaças à validade do estudo são discutidas.</p>	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
Não existe sistemática para melhoria de processo.	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
<p>O MSCI é composto pelas seguintes fases (componentes):</p> <ol style="list-style-type: none"> <li><b>1. Monitoramento de Processo:</b> Monitoramento das medições <math>L_1</math> através de diagramas de controle <math>T_2</math>, disparando relatórios de problemas quando sinais "fora de controle" ocorrerem</li> <li><b>2. Análise de medições <math>L_1</math>:</b> Os sinais "fora de controle" são analisados utilizando decomposição <math>T_2</math> para identificar as medições em <math>L_1</math> que contribuíram para que o sinal ocorresse. (denotado <math>L_1'</math>)</li> <li><b>3. Análise de medições <math>L_2</math>:</b> As medições <math>L_1'</math> e <math>L_2</math> são analisadas com o método PLS (<i>Partial Least Squares</i>) para descobrir as medições <math>L_2</math> que afetaram as medições <math>L_1'</math> nos pontos "fora de controle" (denotado <math>L_2'</math>)</li> <li><b>4. Definição de problema:</b> Análise das medições <math>L_1'</math> e <math>L_2'</math> para explicar as causas dos sinais "fora de controle"</li> </ol> <p><b>Teste de hipótese:</b> Os problemas definidos no passo anterior são testados através de hipóteses construídas e testadas usando o método de teste de hipótese estatística (ou teste de hipóteses múltiplas). As hipóteses verificadas podem ser utilizadas para auxiliar na explicação de causas dos sinais "fora do controle"</p>	
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>	
Os dados de atributos e desempenho do processo são utilizados na abordagem. Contudo, não há sugestões de quais informações devem ser utilizadas, além das instruções já conhecidas de SPC.	

<b>Título</b>	Improving the effectiveness of root cause analysis in post mortem analysis: A controlled experiment
<b>Autor(es)</b>	Bjrnson, Finn Olav; Wang, Alf Inge; Arisholm, Erik
<b>Ano da publicação</b>	2009
<b>Referência completa</b>	Bjrnson, F.O., A.I. Wang, and E. Arisholm, <i>Improving the effectiveness of root cause analysis in post mortem analysis: A controlled experiment</i> . Information and Software Technology, 2009. <b>51</b> (Compendex): p. 150-161.
<b>Contexto</b>	Investigação de causa-raiz
<b>Resumo da publicação</b>	
Abordagem para identificação de oportunidades de melhorias (pontos fortes e pontos fracos) através de mapa causal ao invés do diagrama de de espinha de peixe. O contexto de post mortem possibilita que experiências sejam entendidas e lições aprendidas de bons e maus resultados evidenciados. O artigo também contém um estudo experimental para responder se o mapa causal é mais efetivo do que o diagrama de espinha de peixe (concluindo que realmente é) e de que forma eles se diferem (resultados mais quantitativos e com mais possibilidades de descobertas).	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
Não há	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
<p>A abordagem descreve reuniões "<i>post mortem</i>" para elicitare experiências positivas e negativas. As reuniões possuem quatro passos:</p> <ol style="list-style-type: none"> <li>1. Introdução do método de análise post mortem(PMA) e explicação da revisão;</li> <li>2. Seção-KJ 1: elicitacão de experiências positivas;</li> <li>3. Seção-KJ 2: elicitacão de experiências negativas;</li> <li>4. Executa análise de causa-raiz usando mapa causal nas experiências positivas e negativas mais importantes.</li> </ol> <p>As sessões KJ's são conduzidas da seguintes forma: cada participante recebe um número de <i>post-its</i> onde devem escrever quais experiências, do projeto, são as mais importantes e merecem mais atencão. Os <i>post-its</i> são colocados em um quadro por cada um, seguido da explicacão do que está escrito. Após todos postarem seus comentários, os <i>post-its</i> são agrupados e, para cada grupo, é definida uma categoria, utilizando setas para indicar relacões entre as categorias.</p> <p>Para a seccão de análise de causa raiz, todos os participantes recebem novamente <i>post-its</i> e escrevem as causas da experiência que está sendo analisada. Esses <i>post-its</i> são apresentados e posicionados no quadro, similarmente a fase da seccão-KJ. O grupo então agrupa no quadro as causas quando aplicável. Os relacionamentos de causa-efeito são indicados por setas. Os participantes podem postar novas causas que estejam faltando ou que sejam descritas de formas mais específicas/profundas e indicar a sua relacão causa-efeito com setas novamente. O processo deve seguir até o grupo se satisfazer com a análise.</p>	

<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>
Não há

<b>Título</b>	Non-intrusive monitoring of software quality
<b>Autor(es)</b>	Boffoli, N.
<b>Ano da publicação</b>	2006
<b>Referência completa</b>	Boffoli, N. <i>Non-intrusive monitoring of software quality</i> . in <i>10th European Conference on Software Maintenance and Reengineering, CSMR 2006, March 22, 2006 - March 24, 2006</i> . 2006. Bari, Italy: Inst. of Elec. and Elec. Eng. Computer Society.
<b>Contexto</b>	Processo de software; Desenvolvimento de software globalizado

<b>Resumo da publicação</b>
<p>Medição é o instrumento básico para monitorar, controlar e melhorar o processo de software e seus produtos relacionados. O plano de medição deve, pelo menos: (i) não ser intrusivo; (ii) ser flexível em relação a interpretação das medidas; (iii) ser economicamente viável para o(s) projeto(s).</p> <p>Baseado no controle estatístico de processo (SPC) por meio da análise de série temporal de atributos de processo por meio dos diagramas de controle de Shewhart, um <i>framework</i> é proposto para apoiar a aplicação dos conceitos de SPC em processos de software, levando em consideração os requisitos para a medição citados acima.</p>

<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>
<p>O SPC <i>Framework</i> é composto Run-test set (RST), <i>Run-Test Interpretation</i> (RTI) e <i>Investigation Process</i> (IP). Sua estrutura possibilita a aplicação de conceitos de SPC em processos de software. O IP sugere um fluxo para definir, dentre outros, objetos de medição, avaliação de estabilidade e monitoração de processo, análise de teste de seqüência.</p> <p>Além do SPC <i>Framework</i>, o autor cita a utilização de Calibração Dinâmica (DC - <i>Dynamic Calibration</i>) para refinar os modelos de estimativas com base nos dados gerados nos projetos. As estimativas, ao longo do tempo, deve prover resultados satisfatórios.</p> <p>Por fim, o uso de SPC para monitorar processo e resolução de problemas através de processos como "<i>Problem Resolution Management</i>" (PRM) ou Análise e Resolução de Problemas (ARP) são indicados para controlar, monitorar e resolver problemas de forma não intrusiva.</p>

<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>
Não há
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>
Não há

<b>Título</b>	Six Sigma in software quality
<b>Autor(es)</b>	Hong, G.Y. and T.N. Goh
<b>Ano da publicação</b>	2003
<b>Referência completa</b>	Hong, G.Y. and T.N. Goh, <i>Six Sigma in software quality</i> . TQM Magazine, 2003. <b>15</b> (6): p. 364-373.
<b>Contexto</b>	Organizações de software
<b>Resumo da publicação</b>	
<p>O artigo discute a controvérsia da aplicação do método Seis Sigma para a indústria de software. Hong e Goh citam características distintas dos processos de software em relação à manufatura, onde o Seis Sigma é amplamente aplicado e com sucesso, como as seguintes: (i) não-repetitivo; (ii) diferentes entradas e saídas de processo; (iii) alto grau de cognição para executar tarefas; (iv) visualização mais abstrata; (v) fatores externos diferentes (mudança de desenvolver, nível de conhecimento, habilidades em programação e linguagem de programação).</p> <p>O <i>framework</i> DMAIC (<i>define, measure, analyze, improve, control</i>), aplicado a processos de software, pode ser incorporado a diferentes modelos de ciclo de vida (espiral, incremental, cascata), a fim de melhorar os processos continuamente.</p>	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
<p>O <i>framework</i> DMAIC é composto por:</p> <ul style="list-style-type: none"> <li>• <i>Define</i>: o planejamento do programa para software de Seis Sigma (6SSP) deve ser incorporado nos planejamentos de qualidade e marcos do projeto</li> <li>• <i>Measure</i>: o processo deve ser medido para determinar seu desempenho e quantificar o problema. Primeiro é definido o que constitui um defeito, uma oportunidade e uma unidade de medida para fins de cálculo</li> <li>• <i>Analyze</i>: Nesta fase, o dado coletado é analisado e causas-raiz de problemas são determinadas. Desta forma, a oportunidade de melhoria é identificada. Ferramentas estatísticas podem ser usadas para validar as causas-raiz de problemas, identificar origem da variação e determinar as alternativas de soluções mais promissoras</li> <li>• <i>Improve</i>: Idéias e soluções são implementadas para eliminar e prevenir os defeitos. Alguns experimentos e tentativas podem ser necessários para verificar a melhor solução</li> <li>• <i>Controlar</i>: Nesta fase, o desempenho futuro do processo é controlado. A melhoria é institucionalizada através da modificação do sistema e estruturas (<i>staffing</i>, treinamento, incentivos).</li> </ul> <p>Questões do tipo “o que constitui uma oportunidade de melhoria” ou “que unidade de medida deve ser usada para o cálculo do sigma” são citadas pelo autor.</p>	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
Não há	
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>	
Exemplos de medidas são apresentados para cálculo do sigma. Contudo, não há uma lista específica. O autor cita isso como oportunidade de pesquisa.	

<b>Título</b>	Towards a defect prevention based process improvement approach
<b>Autor(es)</b>	Kalinowski, M., G.H. Travassos, and D.N. Card
<b>Ano da publicação</b>	2008
<b>Referência completa</b>	Kalinowski, M., G.H. Travassos, and D.N. Card. <i>Towards a defect prevention based process improvement approach</i> . in <i>EUROMICRO 2008 - Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2008, September 3, 2008 - September 5, 2008</i> . 2008. Parma, Italy: Inst. of Elec. and Elec. Eng. Computer Society.
<b>Contexto</b>	Melhoria de processo de software
<b>Resumo da publicação</b>	
Definição de Análise causal e análise causal de defeito (DCA), citando os benefícios de DCA. O artigo cita a falta de abordagem para auxiliar a análise de causa-efeito nas análises sucessivas de causas. Através de um estudo baseado em revisão sistemática, levanta dados sobre abordagens de DCA. Propõe uma abordagem de DCA integrado ao controle estatístico de processo (SPC) no intuito de controlar e estabilizar processos, melhorar a capacidade e desempenho geral dos processos.	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	
A abordagem proposta - <i>Defect Based Process Improvement Approach</i> (DBPI) - é composta pelos seguintes passos: <ol style="list-style-type: none"> <li>1. <u>Planejamento</u>: Onde se (i) analisa o último resultado DBPI e estabelece objetivos quantitativos de melhoria, (ii) analisa gráficos SPC, resultando num conhecimento mais aprofundado do processo e refinando os objetivos quantitativos, (iii) aplica o princípio de Pareto para selecionar amostras;</li> <li>2. <u>Preparação</u>: Analisar a amostra para encontrar erros sistemáticos;</li> <li>3. <u>Reunião de Análise Causal</u>: Onde se (i) analisa e modifica através do diagrama de <i>Ishikawa</i> e identifica causas centrais (raiz), (ii) propõe ações para prevenir causas;</li> <li>4. <u>Melhoria</u>: Ações de melhoria são executadas. O artigo não descreve como a melhoria pode ser aplicada.</li> </ol>	
<b>Descrição da sistemática utilizada para fornecer informações para apoiar a investigação da oportunidade de melhoria</b>	
Não há	
<b>Descrição das informações coletadas para apoiar a investigação da oportunidade de melhoria</b>	
Não há	

<b>Título</b>	Continuous software process improvement through statistical process control
<b>Autor(es)</b>	Caivano, Danilo

<b>Ano da publicação</b>	2005
<b>Referência completa</b>	Caivano, D. <i>Continuous software process improvement through statistical process control</i> . 2005.
<b>Contexto</b>	
<b>Resumo da publicação</b>	
<p>São analisadas quatro questões de pesquisa:</p> <ul style="list-style-type: none"> <li>i. quais são os mais apropriados indicadores, dentre aqueles usados no controle estatístico de processo SPC para controle de processo de software</li> <li>ii. como esses indicadores deveriam ser interpretados a fim de auxiliar engenheiros de software no gerenciamento de instabilidade de processo de software</li> <li>iii. como SPC pode ser usado para auxiliar eficientemente o monitoramento e investigação da instabilidade de processo de software</li> <li>iv. como a experiência no uso de SPC na comunidade de ES pode ser coletada e formalizada</li> </ul> <p>A partir das questões de pesquisa, o autor propõe uma abordagem para melhoria contínua de processo de software utilizando o SPC</p>	
<b>Descrição da sistemática utilizada para identificar oportunidades de melhoria contínua</b>	

## I.I. Referências Bibliográficas

- ALBUQUERQUE, A.B., 2008, Avaliação e Melhoria de Ativos de Processos Organizacionais em Ambientes de Desenvolvimento de Software, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- ANDRADE, J.M.S., 2005, Avaliação de Processos de Software em Ambientes de Desenvolvimento de Software Orientados à Organização, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- BIOLCHINI, J., MIAN, P.G., NATALI, A.C.C., et al., 2005, Systematic review in software engineering, Programa de Engenharia de Sistema e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- BOFFOLI, N., 2006, "Non-intrusive monitoring of software quality". In: 10th European Conference on Software Maintenance and Reengineering, CSMR 2006, March 22, 2006 - March 24, 2006, pp. 319-322, Bari, Italy.
- CAIVANO, D., 2005, "Continuous software process improvement through statistical process control". In: Ninth European Conference on Software Maintenance and Reengineering, CSMR 2005, March 21, 2005 - March 23, 2005, pp. 288-293, Manchester, United Kingdom.
- CAMPOS, F.B., 2008, Avaliação e Melhoria de Ativos de Processos de Software de Instituições Implementadoras, Universidade Federal do Rio de Janeiro.
- CHANG, C.P., CHU, C.P., 2008, "Improvement of causal analysis using multivariate statistical process control", *Software Quality Journal*, v. 16, n. 3, pp. 377-409.

- GONÇALVES, F.M.G.S., BEZERRA, C.I.M., BELCHIOR, A.D., et al., 2008, "A strategy for identifying, classifying and prioritizing improvement and innovation actions: A CMMI level 5 and Six Sigma approach", pp. 104-111.
- HONG, G.Y., GOH, T.N., 2003, "Six Sigma in software quality", TQM Magazine, v. 15, n. 6, pp. 364-373.
- KALINOWSKI, M., TRAVASSOS, G.H., CARD, D.N., 2008, "Towards a defect prevention based process improvement approach". In: EUROMICRO 2008 - Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2008, September 3, 2008 - September 5, 2008, pp. 199-206, Parma, Italy.
- KITCHENHAM, B., 2004, Procedures for performing systematic reviews, Keele University, Keele, UK.
- MONTONI, M., 2010, Uma Investigação sobre os Fatores Críticos de Sucesso em Iniciativas de Melhoria de Processos de Software, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- PARK, Y., CHOI, H., BAIK, J., 2007, "A framework for the use of Six Sigma tools in PSP/TSP". In: SERA 2007: Fifth ACIS International Conference on Software Engineering Research, Management, and Applications, August 20, 2007 - August 22, 2007, pp. 807-814, Busan, Korea, Republic of.
- SANTOS, G., 2008, Ambientes de Engenharia de Software Orientados a Corporação, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- SCHOTS, N.C.L., 2010, Uma Abordagem para a Identificação de Causas de Problemas Utilizando Grounded Theory, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro.
- SEI, 2010, "CMMI® for Development (CMMI-DEV), V1.3, CMU/SEI-2010", Software Engineering Institute. Disponível em: <http://www.sei.cmu.edu/>.
- SOFTTEX, 2011, "MPS.BR - Melhoria de Processo do Software Brasileiro - Guia Geral", <http://www.softex.br/mpsbr>.
- TARHAN, A., DEMIRORS, O., 2008, "Assessment of software process and metrics to support quantitative understanding". In: International Conference on Software Process and Product Measurement, IWSM-Mensura 2007, November 5, 2007 - November 8, 2007, v. 4895 LNCS, pp. 102-113, Palma de Mallorca, Spain.
- WANG, Q., JIANG, N., GOU, L., et al., 2006, "BSR: A statistic-based approach for establishing and refining software process performance baseline". In: 28th International Conference on Software Engineering 2006, ICSE '06, May 20, 2006 - May 28, 2006, v. 2006, pp. 585-594, Shanghai, China.

## ANEXO II – DESCRIÇÃO DO PROCESSO DA ABORDAGEM

Atividade	1. Definição do objetivo de melhoria
Descrição	O propósito desta atividade é descrever o objetivo de melhoria para o processo alvo da melhoria. Para organizações de alta maturidade ou que possuam objetivos de desempenho de processo alinhados com os objetivos de negócio, esta atividade também tem o propósito de verificar o alinhamento dos objetivos de negócio com o objetivo de melhoria do processo.

Tarefa	1.1. Descrever objetivo de melhoria
Descrição	Preencher o <u>Formulário de Objetivo de Melhoria</u> com o propósito da melhoria, o processo/subprocesso alvo da melhoria, a perspectiva, os indicadores e o contexto.
Pré-tarefa	Variada
Critério de entrada	Tomada de decisão por melhorar o processo de acordo com um objetivo de melhoria. A tomada de decisão pode ter sido originada de uma seleção e priorização do ciclo de melhoria contínua do Grupo de Processo ou por decisão deliberada por conta de estratégia organizacional.
Critério de saída	Objetivo da melhoria e suas delimitações elaboradas
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	-
Artefatos produzidos	<u>Formulário de Objetivo de Melhoria (FOM)</u>
Pós-tarefa	Se houver necessidade de verificar alinhamento dos objetivos de melhoria com os objetivos de negócio da organização (no caso de organizações de alta maturidade ou que exijam o rastreamento dos objetivos de negócio com os objetivos de desempenho de processo), a tarefa <u>Avaliar Rastreabilidade do Objetivo de Melhoria aos Objetivos de Negócio</u> deve ser executada. Do contrário, executar a tarefa <u>Planejar Melhoria</u> da atividade <u>Preparação para Análise</u> .
Ferramentas requeridas	Processador de texto

Tarefa	1.2. Avaliar Rastreabilidade do Objetivo de Melhoria aos Objetivos de Negócio
Descrição	<ol style="list-style-type: none"> <li>1. Se a organização já tiver objetivos de qualidade e desempenho de processo derivados a partir dos objetivos de negócio, verificar o mapeamento entre os objetivos de desempenho com o objetivo de melhoria.</li> <li>2. Se o objetivo de melhoria é derivado a partir de uma decisão da organização (política ou deliberação), mapear decisão com o objetivo de melhoria. Se os objetivos não estiverem alinhados, retornar para a tarefa <u>Descrever o Objetivo de Melhoria</u> para reformular o objetivo de melhoria. Esta tarefa é opcional.</li> </ol>



<b>Tarefa</b>	<b>1.2. Avaliar Rastreabilidade do Objetivo de Melhoria aos Objetivos de Negócio</b>
Pré-tarefa	Descrever o objetivo de melhoria
Critério de entrada	Objetivo de Melhoria ter sido elaborado (FOM)
Critério de saída	Mapeamento entre objetivos de negócio e objetivo de melhoria
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM)
Artefatos produzidos	Formulário de Objetivo de Melhoria (incrementar)
Pós-tarefa	Planejar Melhoria
Ferramentas requeridas	Processador de Texto

<b>Atividade</b>	<b>2. Preparação para Análise</b>
Descrição	Esta atividade tem como objetivo planejar a melhoria e recuperar dados de contexto, diagrama causal e informações complementares que serão insumos para a investigação das restrições do processo e proposição da solução.

<b>Tarefa</b>	<b>2.1. Planejar Melhoria</b>
Descrição	Com base no <u>Formulário de Objetivo de Melhoria</u> e em informações de orçamento disponibilizado para a melhoria, recursos e responsáveis, planejar o cronograma e a distribuição dos recursos e responsáveis nas atividades da melhoria.
Pré-tarefa	<u>Descrever Objetivo de Melhoria</u> ou <u>Avaliar Rastreabilidade do Objetivo de Melhoria os Objetivos de Negócio</u>
Critério de entrada	Objetivo de Melhoria definido (FOM)
Critério de saída	Planejamento da melhoria realizado
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	<u>Formulário de Objetivo de Melhoria (FOM)</u>
Artefatos produzidos	<u>Planejamento da Melhoria de Projeto (PMP)</u>
Pós-tarefa	Selecionar Diagrama Causal Candidato
Ferramentas requeridas	Processador de Texto MSProject ©

<b>Tarefa</b>	<b>2.2. Selecionar Diagrama Causal Candidato</b>
Descrição	Verificar a existência de um diagrama causal que tenha sido construído anteriormente e que seja caracterizado pelo mesmo objetivo de melhoria (de acordo com o FOM). Se existir, o diagrama causal deve ser inserido no <u>Formulário de Análise de Restrições</u> .
Pré-tarefa	Planejar Melhoria
Critério de entrada	Planejamento da melhoria ter sido realizado
Critério de saída	Se existir diagrama causal candidato, captura do diagrama causal e

	atualização do <u>Formulário de Análise de Restrições</u> .
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM)
Artefatos produzidos	Formulário de Análise de Restrições (FAR)
Pós-tarefa	Coletar Informações de Contexto Organizacionais
Ferramentas requeridas	Processador de Texto

<b>Tarefa</b>	<b>2.3. Coletar Informações Organizacionais de Contexto</b>
Descrição	<p>Levantar informações de contexto organizacionais pertinentes que apoiem a investigação da restrição principal do desempenho do processo. De acordo com a perspectiva e do contexto da melhoria (como descrito no FOM), o responsável deve:</p> <ol style="list-style-type: none"> <li>1. Identificar quais aspectos organizacionais aparentemente não contribuem para atingir ao objetivo de melhoria (estes aspectos podem ser de caráter político, organizacional/departamental ou comportamental/cultural);</li> <li>2. Descrever estas informações em forma de declarações de acordo com as Categorias de Ressalvas Legítimas (vide Anexo III na seção III.A, regras 1 e 2), refletindo a realidade a serem redigidas de maneira assertiva, clara e objetiva.</li> </ol>
Pré-tarefa	<u>Planejar Melhoria</u> ou <u>Selecionar Diagrama Causal Candidato</u>
Critério de entrada	Ter definido o objetivo de melhoria (FOM) e ter planejado a ação de melhoria (PMP)
Critério de saída	Lista de cinco a dez declarações que descrevam os aspectos organizacionais que aparentam não contribuir para atingir ao objetivo da melhoria.
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida (opcional)
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM)
Artefatos produzidos	Formulário de Informações Complementares (FIC)
Pós-tarefa	Coletar Informações de Contexto de Projeto
Ferramentas requeridas	Processador de Texto

<b>Tarefa</b>	<b>2.4. Coletar Informações de Contexto de Projetos</b>
Descrição	<p>Levantar informações de contexto do(s) projeto(s) cuja equipe envolvida participa/participou que apoiem a investigação da restrição principal do desempenho do processo. De acordo com a perspectiva e do contexto da melhoria (como descrito no FOM), o responsável deve:</p> <ol style="list-style-type: none"> <li>1. Identificar, juntamente com a equipe, quais aspectos de execução do processo aparentam não contribuir para atingir ao objetivo de melhoria (estes aspectos podem ser de caráter político, departamental, comportamental ou cultural);</li> <li>2. Descrever estas informações em forma de declarações de acordo com as Categorias de Ressalvas Legítimas (vide Anexo III na seção III.A, regras 1 e 2), refletindo a realidade a serem redigidas de</li> </ol>

	maneira assertiva, clara e objetiva.
Pré-tarefa	Coletar Informações de Contexto Organizacionais
Critério de entrada	Ter coletado informações de contexto organizacionais que aparentemente não contribuem para atingir ao objetivo de melhoria
Critério de saída	Lista de cinco a dez declarações que descrevam os aspectos de projetos que aparentam não contribuir para atingir ao objetivo da melhoria.
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Informações Complementares (FIC)
Artefatos produzidos	Formulário de Informações Complementares (incrementar)
Pós-tarefa	Identificar Fatores Pertinentes
Ferramentas requeridas	Processador de Texto

<b>Tarefa</b>	<b>2.5. Identificar Fatores Pertinentes</b>
Descrição	Identificar os fatores pertinentes a partir da lista de fatores de contexto organizacionais e da lista de fatores de contexto de projetos (FIC). Por meio de um <i>brainstorming</i> , selecionar de cinco a dez declarações advindas das listas que aparentem ter maior impacto restritivo ao desempenho do processo <sup>17</sup> .
Pré-tarefa	Coletar Informações de Contexto de Projeto
Critério de entrada	Listas de declarações de aspectos organizacionais e de projetos que não contribuem para atingir ao objetivo de melhoria
Critério de saída	Lista de cinco a dez fatores pertinentes
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Informações Complementares (FIC)
Artefatos produzidos	Formulário de Análise de Restrições (FAR)
Pós-tarefa	Construir Diagrama Causal
Ferramentas requeridas	Processador de Texto

<b>Atividade</b>	<b>3. Identificar Restrição</b>
Descrição	Esta atividade tem como objetivo identificar a principal restrição que limita o desempenho do processo alvo da melhoria. Para atingir este objetivo, os passos 3, 4 e 5 do Processo de Raciocínio da <i>Árvore da Realidade Atual</i> (ARA) são aplicados. Cada passo da ARA é executado por meio de uma tarefa desta atividade, onde os procedimentos da construção da ARA são aplicados, resultando no diagrama causal e na identificação da restrição principal do desempenho do processo (oportunidade de melhoria).

<sup>17</sup> As informações de contexto são analisadas pela equipe que pode, por meio de um *brainstorming*, priorizar as declarações mais pertinentes. Neste momento, o foco é ter um ponto inicial para começar a construir o diagrama causal (na próxima tarefa – Construir Diagrama Causal) e não identificar a restrição principal.

<b>Tarefa</b>	<b>3.1. Desenvolver Diagrama Causal</b>
Descrição	1. Se houver diagrama causal candidato, reaproveitá-lo, iniciando a construção a partir do diagrama. 2. Desenvolver o diagrama causal com base no procedimento de construção da Árvore da Realidade Atual (ARA) (vide Anexo III na seção III.B). O procedimento da ARA permitirá construir o diagrama, a partir dos fatores pertinentes identificados na tarefa <u>Identificar Fatores Pertinentes</u> . A equipe deve ser contextualizada sobre o objetivo da melhoria, sobre os passos e conceitos da ARA que serão seguidos. A aplicação do procedimento pode ser apoiada por ferramentas multimídia ( <i>datashow</i> ).
Pré-tarefa	Identificar Fatores Pertinentes
Critério de entrada	Ter identificado fatores pertinentes de contextos organizacionais e de projetos e ter definido o objetivo de melhoria
Critério de saída	Diagrama Causal construído
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Informações Complementares (FIC) Formulário de Análise de Restrições (FAR) Procedimento da ARA: Construir Diagrama Causal
Artefatos produzidos	Formulário de Análise de Restrições (incrementar)
Pós-tarefa	Avaliar Diagrama Causal
Ferramentas requeridas	Processador de Texto <i>Transformation LogicTree</i> © <sup>18</sup> ou <i>Flying Logic</i> © <sup>19</sup>

<b>Tarefa</b>	<b>3.2. Avaliar Diagrama Causal</b>
Descrição	Avaliar o diagrama causal construído na tarefa anterior <sup>20</sup> . As regras de Categoria das Ressalvas Legítimas devem ser observadas e as relações de causa-e-efeito devem espelhar o cenário do processo que se deseja melhorar. Se não houver um consenso em relação ao diagrama, retornar para a tarefa <u>Construir Diagrama Causal</u> no passo 5 descrito no procedimento da ARA.
Pré-tarefa	Construir Diagrama Causal
Critério de entrada	Diagrama causal construído
Critério de saída	Validação do diagrama causal por parte da equipe envolvida
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Análise de Restrições (FAR)

<sup>18</sup> <http://www.transformation-logictree.com/>

<sup>19</sup> <http://flyinglogic.com/>

<sup>20</sup> A leitura do diagrama, preferencialmente, pode ser feita dos elementos inferiores em direção aos elementos do topo, utilizando a terminologia: “*Se fator de influência A ocorre então fator de influência B ocorre*”.

Artefatos produzidos	-
Pós-tarefa	Identificar Restrição Principal
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> ©

<b>Tarefa</b>	<b>3.3. Identificar Restrição Principal</b>
Descrição	1. Identificar a restrição principal por meio da aplicação do procedimento da Árvore da Realidade Atual (ARA) descrito na seção III.C do Anexo III.
Pré-tarefa	Avaliar Diagrama Causal
Critério de entrada	Diagrama causal que reflita a realidade do processo/subprocesso alvo da melhoria de acordo com a equipe envolvida
Critério de saída	Restrição Principal que limita o desempenho do processo
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Análise de Restrições (FAR) Procedimento da ARA: Identificar Restrição Principal
Artefatos produzidos	Formulário de Análise de Restrições (incrementar)
Pós-tarefa	Elaborar Proposta de Solução
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> © Processador de Texto

<b>Atividade</b>	<b>4. Elaboração da Proposta de Melhoria</b>
Descrição	O propósito desta atividade é elaborar uma melhoria candidata através da formulação de uma proposta de solução, da análise de viabilidade da proposta de solução e da formulação de uma melhoria candidata com base na solução proposta.

<b>Tarefa</b>	<b>5.1. Elaborar Proposta de Solução</b>
Descrição	<p>1. Elaborar uma proposta de solução que remova ou anule a restrição principal ao desempenho do processo, a fim de atingir o objetivo de melhoria. A partir da restrição principal identificada, a proposta de solução pode ser simplesmente a remoção da restrição<sup>21</sup> ou, se não for possível a sua remoção, resolver o conflito. Para resolver o conflito em atingir o objetivo de melhoria, executar o procedimento de construção do Diagrama de Resolução de Conflito (DRC) descrito na seção III.D do Anexo III.</p> <p>2. Se o DRC for desenvolvido, atualizar o FAR com o diagrama; os pressupostos identificados para cada elemento do diagrama; o pressuposto selecionado como proposta de solução; o raciocínio utilizado para a seleção da proposta de solução; e o objetivo da proposta de solução.</p> <p>3. Se recorrência da tarefa ocorrer por conta da inviabilidade da</p>

<sup>21</sup> Normalmente a simples remoção da restrição é possível quando se trata de uma restrição física. Por exemplo: Restrição principal: “Falta de uma ferramenta de automação de teste”, solução: “Aquisição de uma ferramenta para automação de teste”.

<b>Tarefa</b>	<b>5.1. Elaborar Proposta de Solução</b>
	proposta de solução anterior, selecionar outra solução identificada para os pressupostos do DRC já desenvolvido e adicionar a justificativa de inviabilidade à solução proposta inviabilizada.
Pré-tarefa	<u>Identificar Restrição Principal</u> ou <u>Avaliar Viabilidade da Proposta de Solução</u>
Critério de entrada	Restrição principal ter sido identificada
Critério de saída	Proposta de solução ter sido elaborada
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Análise de Restrições (FAR) Procedimento do DRC: Elaborar Proposta de Solução
Artefatos produzidos	Formulário de Análise de Restrições (incrementar)
Pós-tarefa	<u>Avaliar Viabilidade da Proposta de Solução</u>
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> © Processador de Texto

<b>Tarefa</b>	<b>5.2. Avaliar Viabilidade da Proposta de Solução</b>
Descrição	<ol style="list-style-type: none"> <li>1. Avaliar a viabilidade da proposta de solução através da construção da Árvore da Realidade Futura (ARF). Por meio da aplicação do procedimento da ARF, descrito na seção III.E do Anexo III, construir a árvore com o cenário futuro no qual a proposta de solução é implementada e avaliada;</li> <li>2. Através da ARF, identificar e preencher no FAR as ações necessárias pra implementar a melhoria, os riscos à implementação, os efeitos colaterais e as ações de mitigação;</li> <li>3. A partir das ações necessárias para implementar, atualizar o FAR com as ações de contingência e de mitigação relacionadas aos riscos e aos efeitos colaterais, estimando custos e esforço;</li> <li>4. Com base nas estimativas de custos e esforço, descrever o raciocínio que justifica a viabilização ou inviabilização da proposta. O responsável deve averiguar, caso a proposta não seja viável, se é viável elaborar nova proposta de solução. Neste caso, a execução do processo retorna para a tarefa Elaborar Proposta de Solução.</li> </ol>
Pré-tarefa	Elaborar Proposta de Solução
Critério de entrada	Ter elaborado uma proposta de solução (definir objetivo da proposta de solução)
Critério de saída	Cronograma macro, recursos, orçamento estimado, riscos e ações de mitigações identificados
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Análise de Restrições (FAR) Procedimento da ARF: Avaliar Viabilidade da Proposta de Solução
Artefatos produzidos	Formulário de Análise de Restrições (incrementar)
Pós-tarefa	<u>Elaborar Proposta de Melhoria</u> ou <u>Elaborar Proposta de Solução</u>
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> © Processador de Texto

<b>Tarefa</b>	<b>5.3. Elaborar Proposta de Melhoria</b>
Descrição	<ol style="list-style-type: none"> <li>1. Descrever o objetivo da melhoria proposta, considerando o objetivo da proposta de solução (FAR) e o objetivo de melhoria (FOM).</li> <li>2. Elaborar um estratégia para a proposta de melhoria por meio do procedimento da Árvore de Pré-Requisitos (APR) descrito na seção III.F do Anexo III;</li> <li>3. Elaborar a matriz de riscos envolvidos identificados na execução do passo anterior e preencher no FAR;</li> <li>4. Elaborar a cronograma macro dos itens de ação identificados na execução do passo 2 e preencher no FAR;</li> <li>5. Elaborar a matriz de orçamento estimado dos itens de ação identificados na execução do passo 2 e preencher no FAR;</li> <li>6. Listar recursos estimados para execução dos itens de ação identificados na execução do passo 2 e preencher no FAR;</li> </ol>
Pré-tarefa	Avaliar Viabilidade da Proposta de Solução
Critério de entrada	Ter averiguado a viabilidade da proposta de solução
Critério de saída	Estratégia para a Proposta de Melhoria elaborada
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida (opcional)
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Análise de Restrições (FAR) Procedimento da APR: Elaborar Proposta de Melhoria
Artefatos produzidos	Formulário de Análise de Restrições (incrementar)
Pós-tarefa	<u>Elaborar Planejamento de Implementação da Proposta de Melhoria</u>
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> © Processador de Texto

<b>Tarefa</b>	<b>5.4. Elaborar Planejamento de Implementação da Proposta de Melhoria</b>
Descrição	<ol style="list-style-type: none"> <li>1. Descrever o objetivo da melhoria proposta, considerando o objetivo da proposta de solução (FAR) e o objetivo de melhoria (FOM);</li> <li>2. Elaborar uma proposta de melhoria por meio do procedimento da Árvore de Pré-Requisitos (APR) descrito na seção III.G do Anexo III;</li> <li>3. Elaborar a matriz de riscos com base nos riscos e ações de mitigação identificadas para a proposta de solução e preencher no Formulário de Proposta de Melhoria (FPM);</li> <li>4. Elaborar a Estrutura Analítica de Projeto (EAP), identificando os objetivos de melhoria, objetivos intermediários e ações necessárias para implementar a melhoria;</li> <li>5. Elaborar a Estrutura Análise de Projeto Inversa (EAP Inversa) para descrever claramente o que não é objetivo desta melhoria (opcional);</li> <li>7. Elaborar o cronograma detalhado a partir das ações identificadas na execução do passo 2 e preencher no FPM;</li> <li>8. Elaborar a matriz de orçamento a partir das ações listadas no cronograma detalhado e preencher no FPM;</li> <li>9. Listar os recursos necessários para execução do cronograma detalhado e preencher no FPM;</li> </ol>
Pré-tarefa	<u>Elaborar Proposta de Melhoria</u>
Critério de entrada	Ter elaborado uma estratégia para a proposta de melhoria
Critério de saída	Ter elaborado um planejamento de implementação da proposta de

<b>Tarefa</b>	<b>5.4. Elaborar Planejamento de Implementação da Proposta de Melhoria</b>
	melhoria
Responsável	Responsável do Grupo de Processo
Participantes	Equipe envolvida (opcional)
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Análise de Restrições (FAR) Procedimento da APR: Elaborar Proposta de Melhoria
Artefatos produzidos	Formulário de Proposta de Melhoria (FPM)
Pós-tarefa	-
Ferramentas requeridas	<i>Transformation LogicTree</i> © ou <i>Flying Logic</i> © Processador de Texto

<b>Atividade</b>	<b>6. Implementação da Melhoria Candidata</b>
Descrição	O objetivo desta atividade é implementar a melhoria candidata de acordo com seu o planejamento, a fim de validar a melhoria candidata como apta à implantação na organização.

<b>Tarefa</b>	<b>6.1. Executar piloto</b>
Descrição	1. Com base no Formulário de Proposta de Melhoria, executar o projeto piloto de acordo com a abordagem <i>Pilot</i> descrita por (SILVA FILHO, 2006) ; 2. Atualizar cronograma e orçamento da proposta de melhoria, contida no FPM, de acordo com os resultados obtidos no projeto piloto; 3. Com base no Formulário de Objetivo de Melhoria (FOM), verificar se o objetivo de melhoria foi atingido;
Pré-tarefa	<u>Elaborar planejamento de implementação da proposta de melhoria</u>
Critério de entrada	Planejamento de Proposta de Melhoria ter sido elaborado
Critério de saída	Avaliação da implementação da proposta de melhoria
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Formulário de Proposta de Melhoria (FPM) Procedimento da abordagem <i>Pilot</i>
Artefatos produzidos	Formulário de Proposta de Melhoria (FPM)
Pós-tarefa	1. Se o objetivo de melhoria não for atingido, com base no Planejamento de Melhoria do Processo (PMP), verificar se é viável a reformulação da proposta de solução e da melhoria. a. Se possível, retornar para a Tarefa 5.1 – Elaborar Proposta de Melhoria. b. Se não for viável a reformulação da proposta de solução, executar a Tarefa 6.2 – Registrar coprodutos do processo; 2. Se o objetivo de melhoria for atingido, executar a Tarefa 6.2 – Armazenar melhoria candidata.
Ferramentas requeridas	Processador de Texto



<b>Tarefa</b>	<b>6.2. Armazenar Melhoria Candidata no Repositório Organizacional</b>
Descrição	Armazenar o FPM como melhoria candidata no repositório organizacional.
Pré-tarefa	<u>Executar piloto</u> ou <u>Elaborar planejamento de implementação da proposta de melhoria</u>
Critério de entrada	Resultados do piloto ou raciocínio dos indícios de viabilidade de implementação da proposta de melhoria
Critério de saída	Atualização do repositório organizacional com a melhoria candidata
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	Formulário de Proposta de Melhoria (FPM)
Artefatos produzidos	Repositório Organizacional (atualizado com Proposta de melhoria candidata)
Pós-tarefa	Registrar coprodutos do processo
Ferramentas requeridas	Processador de Texto

<b>Atividade</b>	<b>7. Encerramento</b>
Descrição	O objetivo desta atividade é registrar os resultados e lições aprendidas produzidos ao longo da execução do processo. A Base de Melhoria de Processo deve receber os formulários produzidos, incluindo as melhorias candidatas que serão resgatadas posteriormente para serem implantadas na organização. As lições aprendidas devem ser registrar no repositório organizacional, a fim de guardar a memória de melhoria do processo.

<b>Tarefa</b>	<b>7.1. Registrar artefatos do processo</b>
Descrição	Registrar os produtos (artefatos) referentes à melhoria candidata na Base de Melhoria de Processo (BMP), caso a proposta de melhoria tenha sido considerada viável para implantação na organização.
Pré-tarefa	<u>Elaborar planejamento de implementação da proposta de melhoria</u>
Critério de entrada	Planejamento de Proposta de Melhoria ter sido elaborado e avaliada positivamente
Critério de saída	Avaliação da implementação da proposta de melhoria
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	Formulário de Objetivo de Melhoria (FOM) Planejamento de Melhoria de Processo (PMP) Formulário de Análise de Restrições (FAR) Formulário de Proposta de Melhoria (FPM)
Artefatos produzidos	Formulário de Proposta de Melhoria (FPM)
Pós-tarefa	Registrar lições aprendidas
Ferramentas requeridas	Processador de Texto

<b>Tarefa</b>	<b>7.2. Registrar lições aprendidas</b>
Descrição	Registrar as lições aprendidas ao longo da execução do processo de melhoria na Base Organizacional.
Pré-tarefa	<u>Registrar artefatos do processo</u>
Critério de entrada	Execução das atividades anteriores
Critério de saída	Lições aprendidas registradas na Base Organizacional
Responsável	Responsável do Grupo de Processo
Participantes	-
Artefatos requeridos	-
Artefatos produzidos	-
Pós-tarefa	
Ferramentas requeridas	Processador de Texto

# ANEXO III – DESCRIÇÃO DOS PROCEDIMENTOS DOS PROCESSOS DE RACIOCÍNIO

Neste anexo são apresentados os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições e seus conceitos básicos que apoiam a execução do processo da abordagem proposta nesta dissertação. Os Processos de Raciocínio são compostos por quatro árvores e um diagrama em formato de nuvem, a saber: (i) Árvore da Realidade Atual; (ii) Diagrama de Resolução de Conflito; (iii) Árvore da Realidade Futura; (iv) Árvore de Pré-Requisitos e (v) Árvore de Transição. A correspondência da aplicação dos Processos de Raciocínio para responder às questões de mudança é ilustrada na Tabela III.1.

**Tabela III.1** - Aplicação dos Processos de Raciocínio em relação às questões de mudança (SCHEINKOPF, 1999)

O que mudar?	Árvore da Realidade Atual
	Diagrama de Resolução de Conflitos
Para o que mudar?	Árvore da Realidade Futura
	Árvore de Pré-requisitos
Como causar a mudança?	Árvore de Transição

## III.A. Fundamentos Lógicos

Os Processos de Raciocínio são fundamentados em dois tipos de relacionamento:

- *Causa Suficiente*: é o padrão lógico de causa-efeito-causa necessário para representar a existência causal das entidades. Quando se identifica que uma entidade (efeito) existe porque uma ou mais entidades existem (causas), está se usando o raciocínio de *Causa Suficiente*.
- *Condição Necessária*: é o padrão lógico que verifica quais entidades se fazem necessárias (condições necessárias) para que outra entidade

exista (objetivo ou estado futuro). Um dos objetivos da regra lógica da *Condição Necessária* é revelar os requisitos necessários ou pressupostos para algo acontecer. Condições necessárias são regras, políticas, comportamentos, leis que provêm limitações ou fronteiras para o universo de possibilidades que se tem para atingir objetivos e metas.

Exemplos e representações de Causa Suficiente e Condição Necessária podem ser visto na Tabela III.2.

**Tabela III.2** – Relacionamentos dos Processos de Raciocínio

Regra lógica	Exemplo	Representação
<i>Causa Suficiente</i> (efeito-causa-efeito)	<ul style="list-style-type: none"> <li>A é suficiente para causar B</li> <li>C e D são suficientes para causar E</li> </ul>	<pre> graph BT   A((A)) --&gt; B((B))   C((C)) --&gt; E((E))   D((D)) --&gt; E           </pre>
	X e Y ocorrendo simultaneamente são suficientes para causar P	<pre> graph BT   X((X)) --&gt; O1(( ))   Y((Y)) --&gt; O1   O1 --&gt; P((P))           </pre>
<i>Condição Necessária</i> (pré-condição)	Para atingir S é preciso que T exista/aconteça	<pre> graph BT   T((T)) --&gt; S((S))           </pre>
	Para atingir O, são necessários que M e N simultaneamente existam/aconteçam	<pre> graph BT   M((M)) --&gt; O1(( ))   N((N)) --&gt; O1   O1 --&gt; O((O))           </pre>

Os elementos utilizados para construções das árvores e diagrama baseados em Causa Suficiente e Condição Necessária possuem características e terminologias específicas. A *Árvore da Realidade Atual*, *Árvore da Realidade Futura* e *Árvore de Transição* são construídas com base no relacionamento de Causa Suficiente. As partes que compõem estas árvores são:

- **Entidade:** representa um elemento do sistema ou do processo expresso como uma assertiva e pode ser uma *causa* e/ou um *efeito*. É representado por meio de um retângulo com bordas arredondadas. Exemplo: “A taxa de rotatividade é alta (maior que 20%)”.
- **Seta:** indica um relacionamento entre duas entidades. A entidade na origem da seta é a *causa* e o destino é o *efeito*. Os *pressupostos* são os

responsáveis pelo significado e existência da relação de causalidade representada pela seta.

- **Causa:** uma entidade (ou grupo de entidades interligadas por um *conector-lógico-e*) que, dada a sua existência, causará outra entidade existir (*efeito*).
- **Conector-lógico-e:** agrupa entidades para representar o “e” lógico por meio de uma elipse tocando as *setas* que participam do “e” lógico.
- **Efeito:** representa uma entidade que existe como resultado de uma *causa*. Também referenciado como *consequência*.
- **Pressuposto:** razão e significado da existência de uma relação de causa-e-efeito (*seta*).
- **Ponto de entrada:** entidade que não possui uma seta (não é efeito de nenhuma entidade) como destino existente na realidade atual. Representado por um retângulo.
- **Injeção:** entidade que não possui uma seta como destino (não é efeito de nenhuma entidade) e não existe na realidade atual. São utilizadas para simular hipoteticamente pontos de entrada. Ocorrem apenas na Árvore de Realidade Futura e na Árvore de Transição. Representado por um retângulo.

Os diagramas restantes, Diagrama de Resolução de Conflitos e Árvore de Pré-Requisitos, utilizam o relacionamento de Condição Necessária. Os elementos que os compõem são listados a seguir:

- **Entidade:** representa um único elemento do sistema ou do processo. Diferentemente da *Causa Suficiente*, entidades de *Condição Necessária* não são fundamentalmente assertivas. Termos como *deve*, *pode*, *precisa* e suas variantes negativas podem ser utilizados. É representado por um retângulo com bordas arredondadas.
- **Seta:** indica um relacionamento entre duas entidades. A entidade na origem da seta é a *Condição Necessária* e o destino é o *objetivo*. Toda seta existe com base em um ou mais *pressupostos*.
- **Condição Necessária:** representa uma entidade considerada necessária para que outra entidade (*objetivo*) exista ou que a permita existir.

- **Objetivo:** uma entidade que só pode existir se, pelo menos, outra entidade (*Condição Necessária*) existir.
- **Pressuposto:** representa a razão para a existência do relacionamento de *Condição Necessária* (razão pela qual uma entidade é requerida para que outra exista);
- **Ramo:** representa um agrupamento de entidades relacionadas entre si através de relacionamentos de *Condição Necessária* (uma parte – galho – da árvore).

Os Processos de Raciocínio proveem uma metodologia para testar as concepções da realidade e dos pressupostos por meio de um conjunto de regras de verificação chamado Categorias de Ressalvas Legítimas (CRL). Estas regras agregam conceitos intuitivos do método científico, a fim de prover uma abordagem sistemática para descobrir, verbalizar, questionar e substituir pressupostos existentes (SCHEINKOPF, 1999).

As Categorias de Ressalvas Legítimas são utilizadas para verificar a consistência e completude das inferências e relacionamentos das árvores baseadas no relacionamento de Causa Suficiente (ARA, ARF, AT). Um sumário das CRL é apresentado na Tabela III.3.

**Tabela III.3** – Categorias de Ressalvas Legítimas (SCHEINKOPF, 1999)

Consistência	Descrição	Representação
1. Esclarecimento ou Claridade	Verificar se a causa ou o efeito descrito na entidade está formulada(o) de forma clara e inequívoca	
2. Existência da entidade	Validar a real existência da entidade (efeito ou causa), verificando se a entidade é verossímil	
3. Existência de casualidade	Consistir a presença do elo causal entre o efeito e a causa. Padrão: <i>Se [causa] Então [efeito]</i>	
4. Tautologia	Evitar redundância na relação causa-e-efeito	

Consistência	Descrição	Representação
5. Suficiência (insuficiência) de causa	Verificar a existência de uma ou mais entidades que precisam coexistir com a causa original, a fim de tornar a causalidade válida. Padrão: <i>Se [causa] E [causa] Então [efeito]</i>	
6. Causa adicional	Verificar a existência de outra causa que também resulte no efeito. Padrão: <i>Se [causa] OU [causa] Então [efeito]</i>	
7. Inversão de causa e efeito	Verificar se a direção da seta (relacionamento de causa-e-efeito) não está invertida.	
8. Existência de efeito previsto (redução ao absurdo)	Verificar se a partir da causa original, um novo efeito (que invalide a relação de causa-e-efeito original) é relacionado	

### III.B. Procedimento da Árvore da Realidade Atual: Construir Diagrama Causal

O procedimento a seguir reúne as ações para construção do diagrama causal de acordo com o processo da Árvore da Realidade Atual descrito em (SCHEINKOPF, 1999):

1. Se houver diagrama causal a ser reutilizado, ir para o passo 3. Senão, identificar dois fatores, a partir da lista de fatores pertinentes do Formulário de Análise de Restrições (FAR), que aparentam estar envolvidos em um relacionamento de *Causa Suficiente* (CS) e relacioná-los através de uma das conexões descritas a seguir de acordo com os critérios:
  - (i) Conexão Linear: Se um dos fatores pertinentes é causa do outro, adicione o relacionamento de causalidade. Para verificar se a relação

de causa-e-efeito existe, teste a sentença: *Se fator de influência A ocorre então fator de influência B ocorre.*

- (ii) Conexão V: Se houver uma causa em comum (outro elemento da lista de fatores pertinentes) através da qual os dois fatores pertinentes estão relacionados, adicione a causa ao diagrama e relacione cada um dos fatores pertinentes como efeito da causa adicionada;
2. Escrutinar este relacionamento criado, utilizando as *Categorias de Ressalvas Legítimas* (CRL). Caso a aplicação das CRL tenha inserido um novo elemento (efeito ou fator de influência) no diagrama, verificar se o novo elemento se relaciona com algum outro elemento do diagrama. Repetir este passo até que todas as ressalvas tenham sido observadas para todos os elementos do diagrama causal;
3. A partir da lista remanescente dos fatores pertinentes, selecionar um fator pertinente que aparenta estar relacionado em um relacionamento de *Causa Suficiente* com algum dos elementos existentes no diagrama;
4. Repetir a ação 2 e 3, conectando um fator pertinente por vez apenas, até que todos os fatores pertinentes da lista estejam conectados no diagrama;
5. Procurar por efeitos de influência indesejáveis percebidos pela equipe envolvida que não esteja representada no diagrama causal. Se houver, descrever o elemento (efeito ou fator indesejável), de acordo com as regras das CRL, relacioná-lo com causa(s) e/ou efeito(s) existente(s) através de relacionamentos de *Causa Suficiente*. Escrutinar estes relacionamentos, considerando e reorganizando o restante do diagrama, se necessário;
6. Procurar por relacionamentos de causa-e-efeito existentes na realidade que não foram representados no diagrama causal, submetendo cada novo relacionamento criado às CRL. Escrutinar os relacionamentos de causa-e-efeito e os fatores de influência, reorganizando o restante do diagrama, se necessário.

No momento da construção, SCHEINKOPF (1999) ressalta que a equipe deve se concentrar em um passo e uma entidade de cada vez, evitando adicionar mais fatores à lista de fatores pertinentes. Os fatores de influência do processo e seus efeitos



devem surgir ao longo da aplicação desse procedimento, principalmente pela aplicação das CRL.

Ao final da construção, o diagrama causal deve conter um ou mais *fatores de influência indesejáveis* que representem o oposto do que o objetivo de melhoria almeja.

### III.C. Procedimento da *Árvore da Realidade Atual*: Identificar Restrição Principal

O procedimento a seguir reúne as ações para identificar a restrição principal com base no processo da *Árvore da Realidade Atual* descrito em (SCHEINKOPF, 1999).

Para cada *ponto de entrada*<sup>22</sup>, determinar o grau de impacto (alcance) dos efeitos de influência encadeados. Para isto:

1. Para cada *ponto de entrada*, identificar quais os fatores de influência são alcançados a partir do *ponto de entrada*, como mostrada na Tabela III.4. Preencher o FAR contendo a Matriz de Rastreabilidade.
2. Selecionar como restrição principal o *ponto de entrada* com maior alcance (maior número de fatores de influência alcançados). Se houver empate, selecionar como restrição principal o *ponto de entrada* cujos fatores de influência alcançados são considerados mais críticos ou menos desejados. Preencher o FAR com a restrição principal e seu raciocínio de seleção (se houver empate ou a restrição principal não poder ser removida).

**Tabela III.4** - Exemplo da Matriz de Rastreabilidade das Restrições Candidatas

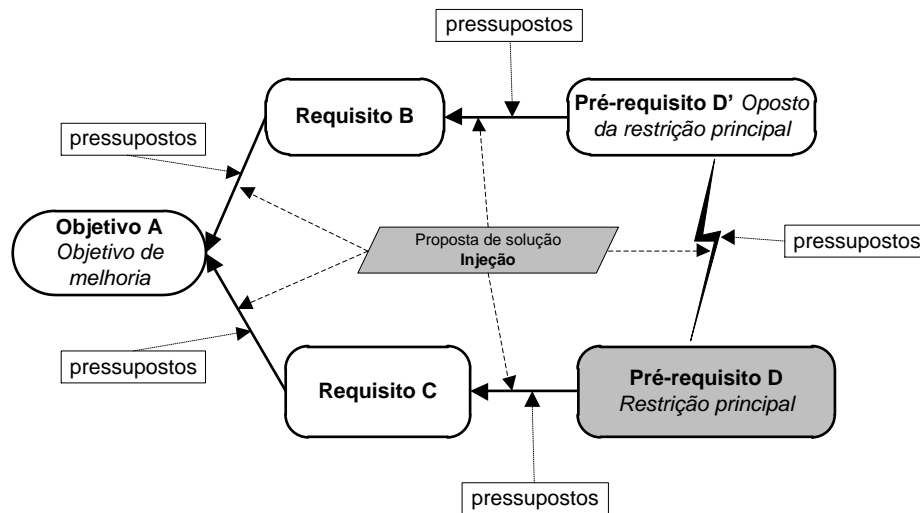
	Efeito A	Efeito B	Efeito C	Efeito D	Grau de impacto
Ponto de entrada 1		X			25%
Ponto de entrada 2	X		X	X	75%
Ponto de entrada 3	X	X			50%

---

<sup>22</sup> Pontos de entrada são causas no nível mais abaixo do diagrama que não são efeito de nenhuma causa.

### III.D. Procedimento do Diagrama de Resolução de Conflito: Elaborar Proposta de Solução

O Diagrama de Resolução de Conflito possui uma estrutura fixa, como apresentado no modelo da Figura III.6.



**Figura III.6** - Diagrama de Resolução de Conflito

Articular o problema de acordo com o modelo descrito na figura acima da seguinte forma:

1. Definir o objetivo da resolução do conflito na lacuna Objetivo A como sendo o objetivo da melhoria, de acordo como descrito no Formulário de Objetivo de Melhoria;
2. Identificar a restrição principal na lacuna de pré-requisitos D e o seu oposto na lacuna de pré-requisito D’;
3. Identificar através do relacionamento de *Condição Necessária* (CN) quais os requisitos cujas *condições necessárias* das lacunas D e D’ estão relacionadas, preenchendo as lacunas C e B respectivamente;
4. Revisar o diagrama com a equipe envolvida para se certificar que todos conseguem visualizar o conflito descrito no diagrama
5. Para cada relacionamento de *Condição Necessária* (seta), evidenciar *pressupostos* e identificar potenciais soluções (*injeções*) para resolver o

conflito, preenchendo o FAR com o DRC, as informações dos pressupostos e as propostas de solução. As soluções podem ser, por exemplo:

- a. Remoção ou modificação do *pressuposto* (restrição física, política ou comportamental) para anular ou remover o conflito;
  - b. Inserção de um novo elemento (*injeção*) que extinga o conflito;
6. Selecionar uma proposta de solução (injeção) das possíveis soluções identificadas, preenchendo o objetivo da proposta de solução e o raciocínio de definição da proposta de solução.

### **III.E. Procedimento da Árvore da Realidade Futura: Avaliar Viabilidade da Proposta de Solução**

#### **Contextualização**

A partir da Matriz de Rastreabilidade das Restrições Candidatas (presente no FAR), os efeitos indesejáveis (EI) e fatores de influência indesejáveis (FII) que são efeitos (diretos ou indiretos) da restrição principal são utilizados nesta avaliação. A Árvore da Realidade Futura gera um cenário futuro cuja remoção de cada EI e FII é alcançado pela inserção de ações (*injeções*). Para cada efeito indesejável ou fator de influência indesejável (realidade atual), uma ação (*injeção*) é criada para anular o EI e o FII (realidade futura), gerando um efeito desejável (ED) ou fator de influência desejável (FID).

Ações (*injeções*) inseridas para construir a realidade futura podem ocasionar efeitos colaterais. A ARF verifica a implementação da realidade futura através de um mecanismo que considera os efeitos colaterais como galhos negativos da árvore (ramos) que são dirimidos sistematicamente e obstáculos<sup>23</sup> que precisam ser transpostos. Estes efeitos colaterais e obstáculos se configuram como riscos relacionados à proposta de solução e as ações (*injeções*) definidas para anulá-los (ações de mitigação). A seção referente à proposta de solução em avaliação, no Formulário de Análise de Restrições, deve ser preenchida com os riscos (efeitos

---

<sup>23</sup> Obstáculos e efeitos colaterais são efeitos indesejáveis. Os obstáculos estão relacionados a objeções que tornam difícil implementar uma ação (*injeção*), ao passo que efeitos colaterais estão relacionados a fatores de influência indesejáveis que podem ocorrer após a implementação de uma ação (*injeção*).

colaterais e obstáculos) identificados, assim como as suas respectivas ações de mitigação (*injeções*).

Com base nas ações (*injeções*) inseridas na ARF, um cronograma macro, recursos e um orçamento estimado devem ser elaborados e preenchidos na seção referente à proposta de solução em análise, no FAR. O cronograma macro deve conter as ações, assim como suas estimativa de esforço e precedência. O orçamento estimado deve conter a ação ou recurso e o valor estimado.

Ao final da execução desta atividade, o responsável pela melhoria deve ser capaz de avaliar se a proposta de solução é técnica e financeiramente viável. As informações referentes ao orçamento e ao prazo destinados à elaboração de melhoria e sua implementação devem servir como critérios para esta análise. O raciocínio referente à análise de viabilidade deve constar no FAR, na seção da proposta de solução em análise. Os efeitos desejáveis e os fatores de influência desejáveis alcançáveis também devem constar na mesma seção do FAR.

Se a proposta de solução for considerada inviável, o responsável pela melhoria, com base no planejamento da melhoria, descrito no Formulário de Objetivo de Melhoria (FOM), deve decidir se há tempo hábil e recursos disponíveis para a atividade de elaboração de uma nova proposta de solução.

### **Procedimento**

O procedimento para identificar as ações de implementação necessárias, riscos e ações de mitigação e contingência é baseado no Processo de Raciocínio ARF proposto por SCHEINKOPF (1999), como descrito a seguir:

1. A partir da Matriz de Rastreabilidade das Restrições Candidatas (encontradas no FAR), listar todos os EI e FII que são efeito direto ou indireto da restrição principal;
2. Para cada EI ou FII listado no passo anterior, identificar um ED ou FID relacionado (o oposto do efeito indesejável ou do fator de influência indesejável), criando uma lista de ED/FID.

Por exemplo: Se “a taxa de rotatividade de funcionários é 20% ao ano (alta)” é um fator de influência indesejável, então um objetivo intermediário relacionado poderia ser: “a taxa de rotatividade de

funcionários é 5% ao ano (baixa)”. A lista de objetivos intermediários deve constar no FAR;

3. Selecionar um ED a partir da lista criada no passo 2 que aparente ser um efeito evidente do objetivo da proposta de solução (*injeção*) e os relacionar por meio de um relacionamento de *Causa Suficiente*;
4. Escrutinar este relacionamento criado utilizando as *Categorias de Ressalvas Legítimas* (CRL). Caso um novo elemento (efeito ou fator de influência) seja inserido na árvore, verificar se o novo elemento se relaciona com algum outro elemento da árvore. Repetir este passo até que todas as ressalvas tenham sido observadas para todos os elementos da árvore;
5. A partir da lista remanescente dos efeitos desejáveis e fatores de influência desejáveis, selecionar um que aparente ser efeito evidente de algum fator de influência presente na árvore e relacioná-lo com base no relacionamento de *Causa Suficiente*;
6. Escrutinar o relacionamento criado, utilizando as CRL, como descrito no passo 4;
7. Repetir a ação 5 e 6, conectando um ED ou FID por vez apenas, até que todos da lista estejam conectados na árvore;
8. A partir dos elementos da árvore, listar os efeitos colaterais potenciais ou obstáculos para implementação da proposta de solução<sup>24</sup>.
9. A partir da lista de efeitos colaterais e obstáculos, selecionar um elemento que aparente ter um relacionamento de causa-e-efeito com algum elemento presente na árvore e relacioná-lo. Adicionar o efeito colateral ou obstáculo à lista de riscos no FAR. Se não houver efeitos colaterais, ir ao passo 15;
10. Escrutinar o relacionamento criado utilizando as CRL, como descrito no passo 3. Marcar o galho da árvore desenvolvido, pois este deverá ser

---

<sup>24</sup> A equipe envolvida, por meio de uma rápida reunião de *Brainstorming*, deve identificar e listar os possíveis obstáculos e efeitos colaterais que possam surgir no cenário futuro representado pela árvore.

removido logo após a inserção de uma ação (*injeção*) para evitar este efeito colateral;

11. Adicionar uma ação (*injeção*)<sup>25</sup> que possa impedir que o efeito colateral ocorra ou que anule o obstáculo, adicionando-a como ação de mitigação ao risco relacionado na lista de riscos no FAR;
12. Remover o galho da árvore referente ao efeito colateral, deixando apenas a *injeção* inserida para mitigar a ocorrência do efeito colateral ou anular o obstáculo;
13. Verificar se a *injeção* inserida se relaciona com algum outro elemento da árvore (se é *injeção* ou causa para algum elemento), utilizando as CRL.
14. Se houver efeito colateral ou obstáculo remanescente, adicioná-lo na lista de efeitos colaterais e obstáculos e retornar ao passo 9;
15. Verificar se é possível responder: Como esta proposta de solução conseguirá se manter após aplicada? Procurar por possíveis ciclos na árvore<sup>26</sup>. Adicionar ações (*injeções*), se necessário, para fechar pelo menos um ciclo de auto-manutenção na árvore, escrutinando o relacionamento como descrito no passo 4.

### **III.F. Procedimento da Árvore de Pré-Requisito: Elaborar Proposta de Melhoria**

O procedimento baseado no Processo de Raciocínio da Árvore de Pré-Requisito descrito em (SCHEINKOPF, 1999) é descrito a seguir:

1. Recuperar os efeitos desejáveis (ED) e fatores de influência desejáveis (FID) no FAR referente à proposta de solução, e listá-los como objetivos intermediários (OI);

---

<sup>25</sup> A equipe em uma rápida sessão de *Brainstorming* deve sugerir uma *injeção* que remova a possibilidade de ocorrência do efeito colateral.

<sup>26</sup> Estes ciclos permitem que a proposta de solução tenha um mecanismo de auto-manutenção que determina sua sustentação. Um exemplo de um ciclo de auto-manutenção é a criação de um relatório periódico ou apresentação que posicione por meio de resultados o alcance da melhoria implementada e justifique o investimento contínuo na melhoria implantada.

2. Recuperar os obstáculos e efeitos colaterais (riscos) referentes à proposta de solução no FAR, e listá-los como obstáculos;
3. Para cada obstáculo, verificar se há um OI relacionado na lista de efeitos desejados (ED) e fatores de influência desejáveis (FID). Se não existir, adicioná-lo à lista;
4. A partir da lista dos objetivos intermediários, identificar dois objetivos intermediários, OI1 e OI2, que aparentam ter uma precedência entre eles (OI2 é temporalmente anterior ao OI1, por exemplo);
5. Iniciar um *ramo* da árvore, relacionando os objetivos intermediários identificados (OI1 e OI2) no passo anterior e rotulando a seta com o obstáculo superado pelo objetivo intermediário precedente (OI2);
6. Escrutinar este relacionamento criado, utilizando a regra lógica do relacionamento de *Condição Necessária* - CN.

Neste passo, pode-se chegar à conclusão que os dois objetivos intermediários relacionados não têm nenhuma precedência ou que há algum obstáculo ainda não identificado que precisa ser considerado.

6.1. No primeiro caso, remover o relacionamento entre os objetivos intermediários (OI1 e OI2), devolver OI1 e OI2 à lista de objetivos intermediários e retornar ao passo 4;

6.2. No segundo caso, identificar um novo objetivo intermediário (OI3) para bloquear o novo obstáculo identificado e inseri-lo como objetivo intermediário entre o OI2 e OI1, analogamente ao descrito no passo 5. Adicionar OI3 à lista dos objetivos intermediários e o novo obstáculo à lista dos obstáculos;

7. Selecionar outro objetivo intermediário da lista que aparente ter precedência com algum objetivo intermediário da árvore, adicioná-lo na árvore analogamente como descrito nos passos 4, 5 e 6;

8. Repetir o passo 7 até que todos os objetivos intermediários da lista, que possam ser conectados a este *ramo* da árvore, sejam utilizados<sup>27</sup>.
9. Os objetivos intermediários remanescentes na lista podem possuir relação temporal ou de *Condição Necessária* entre si. Se este for o caso, um novo *ramo* da árvore deve ser construído e a execução do procedimento retornando para o passo 5;
10. Relacionar todos os objetivos intermediários remanescente na lista, e todos os demais *ramos* da árvore como condições necessárias para alcançar o objetivo principal da melhoria proposta, analogamente como descrito nos passos 6 e 7;
11. Preencher o cronograma de marcos no FAR com os objetivos intermediários, completando as estimativas de recursos e orçamento com base nos marcos identificados.

### **III.G. Procedimento da Árvore de Pré-Requisito: Elaborar Proposta de Melhoria**

O procedimento baseado no Processo de Raciocínio da Árvore de Transição, como descrito em (SCHEINKOPF, 1999), é descrito a seguir:

1. Se o Processo de Raciocínio de APR foi executado<sup>28</sup> e os objetivos intermediários identificados, selecionar um objetivo que aparente ser o mais próximo a ser atingindo a partir da realidade atual. Se não há objetivos intermediários, o objetivo da proposta de solução deve ser utilizado;
2. Determinar uma ação (*injeção*) como situação inicial. Esta *injeção* deve ser a primeira ação a ser tomada a partir da condição atual (primeira ação do plano de implementação) em direção ao objetivo selecionado no passo anterior.

---

<sup>27</sup> Os objetivos intermediários que não se relacionarem com nenhum ramo da árvore formada deve se relacionar de maneira direta com o objetivo principal da proposta de melhoria.

<sup>28</sup> A decisão formal pela execução da validação por piloto foi positivada na tarefa anterior e os objetivos intermediários foram identificados através do procedimento de construção da APR.



3. Relacionar a ação com o objetivo e escrutinar o relacionamento, utilizando as *Categorias de Ressalvas Legítimas* (CRL) (para informações sobre as CRL, vide Seção 3.4.1). Caso um novo elemento (efeito ou fator de influência) seja inserido na árvore, verificar se o novo elemento se relaciona com algum outro elemento da árvore. Repetir este passo até que todas as ressalvas tenham sido observadas para todos os elementos da árvore<sup>29</sup>;
4. Identificar a próxima ação para modificar a situação corrente e alcançar o objetivo intermediário, relacionando-a com um elemento da árvore (efeito ou fator de influência que, após execução da ação, deve causar uma nova situação mais próxima do objetivo a ser atingido). Este passo deve ser executado até se observe que o objetivo intermediário foi atingido<sup>30</sup>;
5. Com o intuito de considerar todas as situações (esperadas e indesejáveis) e identificar todas as ações necessárias para sistematicamente alcançar o objetivo intermediário, utilizar as CRL para desenvolver a árvore.  
  
Situações indesejáveis devem ser modificadas através de ações de mitigação para dirimirem o risco de acontecerem.
6. Registrar as ações identificadas para evitar que efeitos colaterais ocorram no mapeamento de riscos, referenciando o efeito colateral. Registrar as ações identificadas para alcançar o objetivo intermediário na matriz de risco, referenciando o obstáculo ao qual o objetivo intermediário se propõe transpor como ação de mitigação.

---

<sup>29</sup> A partir do nível mais inferior da árvore até o seu topo, as ações devem seguir uma sequência lógica de execução

<sup>30</sup> A sentença “**Se** ação existe e situação atual ocorre **então** nova situação passa a existir” pode facilitar a identificação das ações.

# ANEXO IV – MODELOS DE FORMULÁRIOS DO PROCESSO

Neste anexo são apresentados os modelos dos formulários utilizados para auxiliar a execução do processo desta abordagem. Os modelos possuem instruções de preenchimento em itálico e cinza que devem ser removidos no momento do preenchimento.

## IV.A. Formulário de Objetivo de Melhoria (FOM)

### Introdução

Este formulário apresenta o objetivo de melhoria, definindo o processo investigado, propósito de melhoria, perspectiva e contexto.

### Objetivo

**Analisar** *Processo / Sub Processo*

**Com o propósito de** *objetivo quantitativo / qualitativo*

**Com relação a** *aspecto a ser melhorado*

**Do ponto de vista** *quem perceberá a melhoria*

**No contexto** *em qual contexto*

### Objetivos de qualidade e desempenho organizacionais relacionados

*Para organizações de alta maturidade ou que mantenham um mapeamento entre os objetivos estratégicos com os objetivos de melhoria de processo, os objetivos de qualidade e de desempenho organizacionais devem ser listados nesta seção.*

## IV.B. Planejamento da Melhoria de Processo (PMP)

### Introdução

Este planejamento tem como objetivo definir o escopo da melhoria de processo, identificando papéis e responsáveis. Juntamente com este planejamento, um cronograma é definindo, contendo prazo e esforço para cada atividade.

### Escopo

O escopo deste planejamento da melhoria tem como foco analisar o <processo> com o propósito de <objetivo quantitativo/qualitativo> com relação a <aspecto> do ponto de vista dos <partes interessadas> no contexto dos <projetos ou programas de melhoria>.

### Papéis e Responsáveis

As tarefas realizadas na execução deste planejamento de melhoria de processo são relacionadas aos seus respectivos papéis na Tabela IV.1.

**Tabela IV.1** - Atividades x Papéis

Atividade	Papéis		
	RM	EE	
Descrever objetivo de melhoria	X		
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio	X		
Planejar Melhoria	X		
Selecionar diagrama causal	X		
Coletar informações organizacionais de contexto	X		
Coletar informações de contexto de projetos	X	X	
Identificar fatores pertinentes	X	X	
Desenvolver diagrama causal	X	X	
Avaliar diagrama causal	X	X	
Identificar restrição principal	X		
Elaborar proposta de solução	X	X	
Avaliar viabilidade da proposta de solução	X	X	
Elaborar proposta de melhoria	X		
Elaborar planejamento de implementação da proposta de melhoria	X		
Armazenar melhoria candidate	X		
Registrar coprodutos do processo	X		
Registrar lições aprendidas	X		

*Legenda:* **RM** – Responsável do Grupo de Melhoria, **EE** – Equipe Envolvida

A relação dos responsáveis por papéis e objetivos intermediários é descrita na Tabela IV.2.

**Tabela IV.2** - Responsáveis x Papéis

Responsável	Papéis		
	RM	EE	
<i>Responsável do Grupo de Melhoria</i>	X		
<i>Membro 1 da equipe</i>		X	
<i>Membro 2 da equipe</i>		X	

Membro 3 da equipe		X	
--------------------	--	---	--

Legenda: **RM** – Responsável do Grupo de Melhoria, **EE** – Equipe Envolvida

### Orçamento

**Tabela IV.3** - Orçamento Estimado x Orçamento Realizado

Atividade	Valor estimado	Valor realizado
Descrever objetivo de melhoria		
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio		
Planejar Melhoria		
Selecionar diagrama causal		
Coletar informações organizacionais de contexto		
Coletar informações de contexto de projetos		
Identificar fatores pertinentes		
Desenvolver diagrama causal		
Avaliar diagrama causal		
Identificar restrição principal		
Elaborar proposta de solução		
Avaliar viabilidade da proposta de solução		
Elaborar proposta de melhoria		
Elaborar planejamento de implementação da proposta de melhoria		
Armazenar melhoria candidate		
Registrar coprodutos do processo		
Registrar lições aprendidas		

### Cronograma

**Tabela IV.4** - Cronograma de realização da melhoria

Atividade	Início	Término	Esforço
Descrever objetivo de melhoria			
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio			
Planejar Melhoria			
Selecionar diagrama causal			
Coletar informações organizacionais de contexto			
Coletar informações de contexto de projetos			
Identificar fatores pertinentes			
Construir diagrama causal			
Avaliar diagrama causal			
Identificar restrição principal			
Elaborar proposta de solução			
Avaliar viabilidade da proposta de solução			
Elaborar proposta de melhoria			
Elaborar planejamento de implementação da proposta de melhoria			

Armazenar melhoria candidate			
Registrar coprodutos do processo			
Registrar lições aprendidas			

### **Outros Recursos**

*Listar os recursos que devem ser alocados e reservados, tais como sala de reunião, instrumentos de apresentação, computadores, impressora, dentre outros.*

## **IV.C. Formulário de Informações Complementares (FIC)**

### **Introdução**

Este formulário contém informações das informações de contexto que apoiaram a identificação de fatores de influência do processo em melhoria.

### **Informações organizacionais de contexto**

*Descrever informações referentes à organização que pode influenciar de maneira indesejável o processo alvo da melhoria, tais como problemas departamentais, restrições de política organizacional, percepção de cultura organizacional.*

### **Informações de contexto de projetos**

*Descrever informações referentes a projetos que estão executando o processo alvo da melhoria que pode influenciar de maneira indesejável, tais como, cronograma muito restrito, falta de recursos alocados.*

### **Fatores pertinentes**

*Listar cinco a dez fatores pertinentes identificados no contexto organizacional e de projeto.*

## **IV.D. Formulário de Análise de Restrições (FAR)**

### **Introdução**

Este formulário apresenta o resultado da análise das restrições do processo e as proposições de soluções relacionadas às restrições. Todo o raciocínio utilizado para investigar o processo e identificar a principal restrição, solucionar conflitos para implementar a melhoria candidata e a avaliação das melhorias candidatas são encontradas nas seções seguintes.

*Este formulário contém blocos de instruções para guiar o seu preenchimento com a formatação em itálico e texto em azul. Como exemplo, este bloco é de instrução e devem ser removidos após o preenchimento do formulário.*

### **Diagrama Causal**

*Inserir o diagrama causal desenvolvido na tarefa “Desenvolver diagrama causal” ou a referência do caminho e arquivo do diagrama.*

### **Modelo Causal Reutilizado**

*Se houver reutilização do diagrama causal a partir do repositório de modelos causais, como previsto na tarefa “Selecionar diagrama causal candidato”, inserir o modelo causal utilizado ou a referência do caminho e arquivo do diagrama.*

### **Matriz de Rastreabilidade das Restrições Candidatas**

*Montar a matriz de rastreabilidade de acordo com os passos da tarefa “Identificar restrição principal”.*

	<i>Efeito A</i>	<i>Efeito B</i>	<i>Repetir para todos os efeitos identificados no diagrama</i>	<b>Grau de impacto</b>
<i>Ponto de entrada 1</i>				
<i>Ponto de entrada 2</i>				
<i>Repetir para todos os pontos de entrada identificados no diagrama</i>				

### **Restrição Principal**

#### **Raciocínio de seleção da restrição principal**

*Aplicável no caso de desempate de grau de impacto ou impossibilidade de tratar restrição principal.*

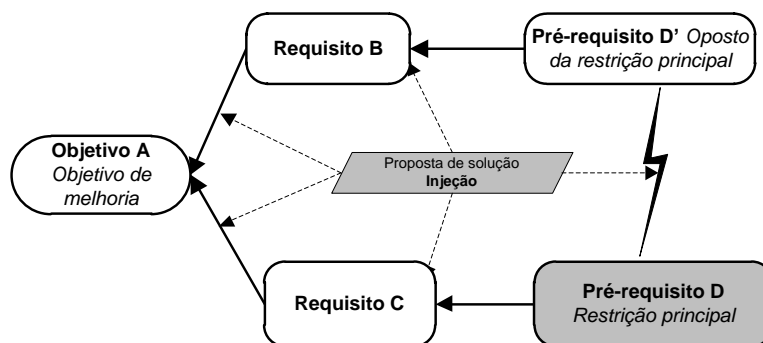
### **Propostas de Solução**

#### **Proposta de Solução I**

*Repetir esta seção para todas as execuções da tarefa “Elaborar proposta de solução”. As recorrências desta tarefa podem ter surgido por decisão em passo posterior, quando se averigua que a proposta de solução que está sendo desenvolvida não é viável.*

#### **Diagrama de Resolução de Conflito**

*Se aplicável, inserir o diagrama de resolução de conflito desenvolvido na tarefa “Elaborar proposta de solução” ou a referência do caminho e arquivo do diagrama.*



## Pressupostos identificados

Com base no procedimento da tarefa “Elaborar Proposta de Solução”, identificar, para cada condição necessária representada no diagrama de resolução de conflito e para o conflito existente, os pressupostos e as possíveis soluções para anular ou remover o pressuposto.

### (i) Pressupostos da Condição Necessária 1

Pressuposto: *pressuposto elucidado*

Proposta de solução: *Proposta de solução para anular ou remover o pressuposto*

### (ii) Pressupostos da Condição Necessária 2

Pressuposto: *pressuposto elucidado*

Proposta de solução: *Proposta de solução para anular ou remover o pressuposto*

### (iii) Pressupostos da Condição Necessária 3

Pressuposto: *pressuposto elucidado*

Proposta de solução: *Proposta de solução para anular ou remover o pressuposto*

### (iv) Pressupostos da Condição Necessária 4

Pressuposto: *pressuposto elucidado*

Proposta de solução: *Proposta de solução para anular ou remover o pressuposto*

### (v) Pressupostos do Conflito 5

Pressuposto: *pressuposto elucidado*

Proposta de solução: *Proposta de solução para anular ou remover o pressuposto*

## Pressuposto e proposição de solução selecionada

Identificar o pressuposto e a proposta de solução selecionada para solucionar o conflito.

## Objetivo da proposta de solução

Descrever o objetivo da proposta de solução, como prevista na tarefa “Elaborar Proposta de Solução”. A descrição do objetivo da solução proposta deve conter o objetivo da melhoria do processo, como descrita no Formulário de Objetivo de Melhoria, e a proposição de solução.

## Raciocínio de definição da proposta de solução

Se aplicável, descrever o raciocínio utilizado para formular o objetivo da proposta de solução. Utilize esta seção para explicar com mais detalhes de que forma a proposição de solução permitirá atingir o objetivo de melhoria do processo.

## Efeitos desejáveis e Fatores de influência desejáveis

Listar os efeitos desejáveis e fatores de influência desejáveis relacionados à proposta de solução. Utilizar os efeitos e fatores indesejáveis identificados no diagrama causal. Os efeitos desejáveis normalmente são o oposto do efeito desejável identificado no diagrama.

## Riscos envolvidos (obstáculos/efeitos colaterais)

Listar os efeitos indesejáveis (efeitos colaterais) que venham a surgir na Árvore da Realidade Futura e as ações de mitigação que devem compor a estratégia de forma que estes efeitos não aconteçam.

Risco	Tipo <sup>31</sup>	Ações de mitigação
<i>Efeito indesejável identificado na Árvore da Realidade Futura que foi bloqueado com uma ação de mitigação</i>		

## Cronograma de marcos

Descrever os marcos (milestones) da proposta de solução. As ações (injeções) adicionadas à Árvore da Realidade Futura para alcançar o objetivo da proposta de solução devem ser listadas abaixo.

Marco	Esforço	Precedência

## Recursos estimados

Descrever aqui os recursos (humanos e físicos) que se estima ser utilizado na implementação desta proposta de solução.

---

<sup>31</sup> Tipo: O – Obstáculo, E – Efeito colateral.



## **Orçamento estimado**

*Descrever aqui o orçamento que se estima ser utilizado na implementação desta proposta de solução*

<b>Marco/Recurso</b>	<b>Valor estimado</b>
<b>Total</b>	

## **Laudo de viabilidade da proposta de solução**

*Após avaliação da proposta de solução através de um projeto piloto ou avaliação de especialista, descrever o raciocínio ou laudo resultado que justifique a viabilidade ou inviabilidade da proposta de solução.*

## **IV.E. Formulário de Proposta de Melhoria (FPM)**

### **Introdução**

Este formulário apresenta uma proposta de melhoria para o processo <processo>. Esta melhoria candidata foi submetida a uma avaliação e teve indícios de viabilidade para implantação na organização. Esta proposta de melhoria contém informações sobre o objetivo da melhoria, escopo, mapeamento de risco, alocação de recursos, orçamento, cronograma dentre outras informações que devem ser preenchidas após o processo de implantação na organização. Este formulário deve ser retornado à Base de Melhoria de Processo para constar como ativo do processo alvo da melhoria e para registrar os resultados do processo de implantação.

### **Objetivo da Proposta de Melhoria**

*Descrever o objetivo da melhoria candidata. Esta descrição está relacionada ao objetivo da proposta de solução que originou esta proposta de melhoria.*

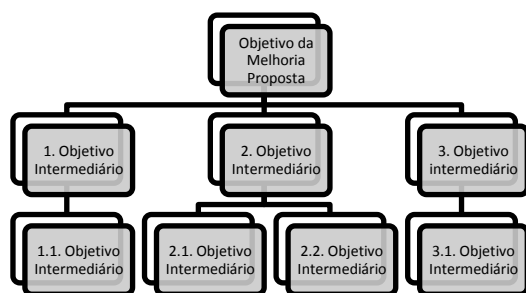
### **Objetivos Quantitativos e Qualitativos da Proposta de Melhoria**

*Descrever os objetivos quantitativos e qualitativos que se deseja atingir ao final da implantação da proposta de melhoria.*

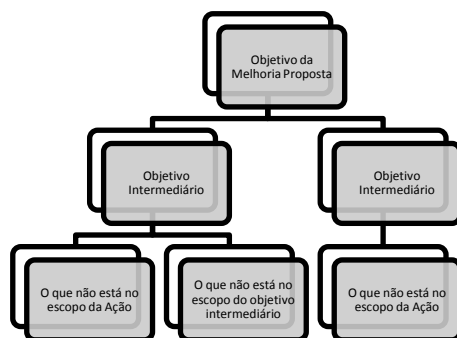
### **Escopo da Melhoria**

*Delimitar, por meio de uma estrutura analítica, o escopo da melhoria. A estrutura analítica pode ser preenchida com base nos marcos (milestones) identificados na proposta de solução. Se necessário, identificar o que está fora do escopo da melhoria através de um EAP de não –escopo.*

### **Estrutura Analítica**



### Não-escopo



### Mapeamento de Riscos

Mapear os riscos, analisando qualitativa e quantitativamente os riscos e identificando ações de mitigação e de contingência. As ações de mitigação podem ser derivadas a partir dos riscos identificados na proposta de solução.

Risco	O <sup>32</sup>	Probabilidade	Impacto	Disparo de contingência	Ações de mitigação	Ações de contingência	Situação
<i>Risco baseado em efeito colateral ou em obstáculo identificado</i>		%	%	<i>Condição de disparo das ações de contingência</i>	<i>Ação de mitigação 1</i>	<i>Ação de contingência 1</i>	<i>Valores possíveis: Identificado Em mitigação Transferido Em contingência Mitigado</i>
					<i>Ação de mitigação 2</i>	<i>Ação de contingência 2</i>	
					<i>Ação de mitigação 3</i>	<i>Ação de contingência 3</i>	

### Recursos

Listar os recursos (humanos e físicos) estimados para implantar a melhoria.

Nome	Papel / Cargo

### Orçamento

Listar os itens ou ações, data estimada de início, custo estimado e valor referente à contingência de riscos para a ação planejada.

Item/Ação	Data estimada	Custo estimado	Contingência

<sup>32</sup> Identificado a partir de um obstáculo

### **Análise de Custo-Benefício**

*Descrever a análise de custo-benefício obtido a partir da implantação ou da avaliação da proposta de melhoria por meio de um projeto piloto.*

### **Validação e Resultado da proposta de melhoria**

*Descrever o raciocínio utilizado na decisão de validar a melhoria candidata como melhoria implantada e*

# ANEXO V – INFORMAÇÕES E ARTEFATOS PRODUZIDOS E UTILIZADOS DURANTE O ESTUDO DE VIABILIDADE

Neste anexo são apresentados os formulários preenchidos no estudo de viabilidade e as informações levantadas ao longo da sua execução. Na Seção V.A, o modelo utilizado para obter o consentimento dos envolvidos na pesquisa é apresentado. Nas Seções V.B e V.C, são apresentados os questionários da avaliação da abordagem preenchidos pelos participantes do estudo de viabilidade. Nas Seções V.D a V.G, são apresentados, respectivamente, o Formulário de Objetivo de Melhoria, o Formulário de Planejamento de Melhoria, o Formulário de Informações Complementares e o Formulário de Análise de Restrições.

## V.A. Modelo do termo de consentimento de pesquisa

### TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

#### **Pesquisa: “Avaliação de aplicabilidade da abordagem de melhoria contínua de processo de software utilizando a Teoria das Restrições”**

Prezado Senhor (a),

Como parte de uma pesquisa de mestrado, uma abordagem para melhoria contínua de processo de software utilizando a Teoria das Restrições foi desenvolvida e tem sido avaliada experimentalmente. Você está sendo convidado a participar de uma pesquisa que estudará os resultados de utilização dessa abordagem em uma avaliação de aplicabilidade para melhoria em um processo da organização. O objetivo da pesquisa é avaliar se a abordagem é adequada (aplicável) à melhoria de processo de software. Sua participação na pesquisa **não** é obrigatória.

#### **1) Procedimento**

A abordagem será aplicada para a melhoria do processo de **Atendimento de Chamados**. Você receberá treinamento sobre a abordagem e a utilizará para avaliar a aplicabilidade das atividades do processo da abordagem selecionadas para a aplicação. Para participar deste estudo solicito a sua especial colaboração em: (1) executar as atividades selecionadas do processo da abordagem, (2) permitir que os dados resultantes da sua avaliação sejam estudados, (3) informar o tempo gasto nas atividades do processo da abordagem, (4) participar de um treinamento para aplicação da abordagem e (5) participar de uma entrevista ou responder um questionário sobre a utilização da abordagem. Quando os dados forem coletados, seu nome será removido dos mesmos e não será utilizado em nenhum momento durante a análise ou apresentação dos resultados.

#### **2) Tratamento de possíveis riscos e desconfortos**

Serão tomadas todas as providências durante a coleta de dados de forma a garantir a sua privacidade e seu anonimato. Os dados coletados durante o estudo destinam-se estritamente a atividades de pesquisa relacionadas à abordagem, não sendo utilizados em qualquer forma de avaliação profissional ou pessoal.

### 3) Benefícios e Custos

Espera-se que, como resultado deste estudo, você possa aumentar seu conhecimento sobre melhoria de processo, de maneira a contribuir para o aumento da qualidade de sistemas de software com os quais você trabalhe. Este estudo também contribuirá com resultados importantes para a pesquisa de um modo geral nas áreas de Qualidade de Software e Melhoria de Processo de Software.

Você não terá nenhum gasto ou ônus com a sua participação no estudo e também não receberá qualquer espécie de reembolso ou gratificação devido à participação **na pesquisa**.

### 4) Confidencialidade da Pesquisa

Toda informação coletada neste estudo é confidencial e seu nome e o da sua organização não serão identificados de modo algum, a não ser em caso de autorização explícita para esse fim.

### 5) Participação

Sua participação neste estudo é muito importante e voluntária. Você tem o direito de não querer participar ou de sair deste estudo a qualquer momento, sem penalidades. Em caso de você decidir se retirar do estudo, favor notificar um pesquisador responsável.

### 6) Declaração de Consentimento

Li ou alguém leu para mim as informações contidas neste documento antes de assinar este termo de consentimento. Declaro que toda a linguagem técnica utilizada na descrição deste estudo de pesquisa foi explicada satisfatoriamente e que recebi respostas para todas as minhas dúvidas. Confirmando também que recebi uma cópia deste Termo de Consentimento Livre e Esclarecido. Compreendo que sou livre para me retirar do estudo em qualquer momento, sem qualquer penalidade. Declaro ter mais de 18 anos e dou meu consentimento de livre e espontânea vontade para participar deste estudo.

Local e Data:

Organização: _____	
Participante	Pesquisador
Nome: _____	Nome: Thiago Moreira da Costa
Assinatura: _____	Assinatura: _____

## V.B. Formulário de Avaliação do Membro da Equipe #1

Por gentileza, responda as questões a seguir considerando sua experiência durante a execução da abordagem de melhoria de processo de software utilizando a Teoria das Restrições:

1. Em relação à sua percepção sobre a facilidade de aplicação do processo da abordagem, qual o seu grau de concordância em relação às seguintes afirmações:

	Concordo Totalmente (100%)	Concordo Amplamente (99% -	Concordo Parcialmente (69% -	Discordo Parcialmente (50% -	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)

		70%)	51%)	31%)		
Foi fácil aprender o processo da abordagem				X		
Consegui ter uma visão consistente de como o processo identificaria a oportunidade de melhoria e propor a solução para melhorar o processo	X					
É fácil lembrar como as atividades (diferente de procedimentos) são executadas no processo da abordagem			X			
É fácil lembrar como os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições (construção dos diagramas) são executados			X			
Considero a abordagem fácil de usar			X			

Comentários (opcional):

A abordagem não é trivial, porém, através de um *mentoring* de 2 horas, a aplicação ficou mais fácil.

2. Em relação à sua percepção sobre a utilidade da abordagem, qual o seu grau de concordância em relação às afirmações abaixo:

	Concordo Totalmente (100%)	Concordo Amplamente (99% - 70%)	Concordo Parcialmente (69% - 51%)	Discordo Parcialmente (50% - 31%)	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)
A abordagem permitiu identificar ou confirmar oportunidade de melhoria (ou problema) no processo <b>Atendimento de Chamado</b>	X					
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de identificar problemas ou oportunidades de melhoria em processos de software	X					
A abordagem permitiu propor soluções para a oportunidade de melhoria (ou problema) identificada no processo <b>Atendimento de Chamado</b>	X					
A abordagem permitiu avaliar as proposições de solução e identificar riscos e obstáculos	X					
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de formular soluções para problemas ou	X					

oportunidades de melhoria em processos de software						
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de avaliar proposições de solução e de identificar riscos e obstáculos futuros	X					
A abordagem facilitou a melhoria do processo <b>Atendimento de Chamado</b>	X					
Eu considero a abordagem útil para melhoria de processo de software	X					

Comentários (opcional):

A abordagem direciona a melhoria, excluindo os riscos e ampliando as chances de atuarmos na causa-raiz.

3. Em relação a algumas questões específicas da Teoria das Restrições e seus Processos de Raciocínio, qual o seu grau de concordância em relação às afirmações abaixo:

	Concordo Totalmente (100%)	Concordo Amplamente (99% - 70%)	Concordo Parcialmente (69% - 51%)	Discordo Parcialmente (50% - 31%)	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)
Foi fácil seguir a execução dos procedimentos guiados pelo responsável do grupo de processo	X					
A execução dos procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições (construção dos diagramas) tomou um tempo plausível para análise	X					
As informações de contextualização e conceitos da Teoria das Restrições estavam claras e apoiaram a execução do processo	X					
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições aumentaram a minha percepção de informações e características dos processos de software que apoiaram na identificação da oportunidade de melhoria e na elaboração da proposta de solução	X					

Comentários (opcional):

Sem comentários.

4. Use o espaço a seguir para comentários gerais que julgar necessários sobre a abordagem, sua aplicação, oportunidades de melhoria para a abordagem, dificuldades encontradas, etc.

Tive dificuldade para desenvolver a árvore da realidade atual (diagrama causal). Barreiras como o entendimento da ferramenta (*jthinker*) e critérios de elaboração das causas geram uma dificuldade inicial, resolvida totalmente com o *mentoring*. Após isto, o problema foi sanado e ficou mais simples identificar a restrição e o tratamento.

A abordagem nos permitiu avaliar o problema através de uma ótica mais limpa. Quando estamos diante de um problema realmente grave dentro de um processo, vários elementos contribuem para piorar aquele fator. Atacar as causas de forma distribuída pode até resolver, porém pode gastar mais esforço. A abordagem aplica um foco de luz na solução. O caminho fica óbvio.

## V.C. Avaliação do Membro da Equipe #2

Por gentileza, responda as questões a seguir considerando sua experiência durante a execução da abordagem de melhoria de processo de software utilizando a Teoria das Restrições:

1. Em relação à sua percepção sobre a facilidade de aplicação do processo da abordagem, qual o seu grau de concordância em relação às seguintes afirmações:

	Concordo Totalmente (100%)	Concordo Amplamente (99% - 70%)	Concordo Parcialmente (69% - 51%)	Discordo Parcialmente (50% - 31%)	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)
Foi fácil aprender o processo da abordagem		x				
Consegui ter uma visão consistente de como o processo identificaria a oportunidade de melhoria e propor a solução para melhorar o processo		x				
É fácil lembrar como as atividades (diferente de procedimentos) são executadas no processo da abordagem			x			
É fácil lembrar como os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições (construção dos diagramas) são executados			x			
Considero a abordagem fácil de usar		x				

Comentários (opcional):



2. Em relação à sua percepção sobre a utilidade da abordagem, qual o seu grau de concordância em relação às afirmações abaixo:

	Concordo Totalmente (100%)	Concordo Amplamente (99% - 70%)	Concordo Parcialmente (69% - 51%)	Discordo Parcialmente (50% - 31%)	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)
A abordagem permitiu identificar ou confirmar oportunidade de melhoria (ou problema) no processo <b>Atendimento de Chamado</b>		x				
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de identificar problemas ou oportunidades de melhoria em processos de software		x				
A abordagem permitiu propor soluções para a oportunidade de melhoria (ou problema) identificada no processo <b>Atendimento de Chamado</b>		x				
A abordagem permitiu avaliar as proposições de solução e identificar riscos e obstáculos			x			
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de formular soluções para problemas ou oportunidades de melhoria em processos de software			x			
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições melhoraram a minha capacidade de avaliar proposições de solução e de identificar riscos e obstáculos futuros			x			
A abordagem facilitou a melhoria do processo <b>Atendimento de Chamado</b>		x				
Eu considero a abordagem útil para melhoria de processo de software		x				

Comentários (opcional):

3. Em relação a algumas questões específicas da Teoria das Restrições e seus Processos de Raciocínio, qual o seu grau de concordância em relação às afirmações abaixo:

	Concordo Totalmente (100%)	Concordo Amplamente (99% - 70%)	Concordo Parcialmente (69% - 51%)	Discordo Parcialmente (50% - 31%)	Discordo Amplamente (30% - 1%)	Discordo Totalmente (0%)
Foi fácil seguir a execução dos procedimentos guiados pelo responsável do grupo de processo			x			
A execução dos procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições (construção dos diagramas) tomou um tempo plausível para análise			x			
As informações de contextualização e conceitos da Teoria das Restrições estavam claras e apoiaram a execução do processo			x			
Os procedimentos baseados nos Processos de Raciocínio da Teoria das Restrições aumentaram a minha percepção de informações e características dos processos de software que apoiaram na identificação da oportunidade de melhoria e na elaboração da proposta de solução		x				

Comentários (opcional):

4. Use o espaço a seguir para comentários gerais que julgar necessários sobre a abordagem, sua aplicação, oportunidades de melhoria para a abordagem, dificuldades encontradas, etc.

Sem comentários.

## V.D. Formulário de Objetivo de Melhoria (FOM)

### Introdução

Este formulário apresenta o objetivo de melhoria, definindo o processo investigado, propósito de melhoria, perspectiva e contexto.

### **Objetivo**

**Analisar** atendimento do Chamado

**Com o propósito de** diminuir o tempo de atendimento para 15 dias

**Com relação ao** tempo de atendimento do analista de negócios

**Do ponto de vista** do cliente

**No contexto** dos chamados ao Sistema X

### **Objetivos de qualidade e desempenho organizacionais relacionados**

O processo Atendimento do chamado tem por objetivo receber chamados de atendimento de manutenção dos softwares em produção. O Objetivos e Metas Organizacionais, define que o objetivo quantitativo deste procedimento da qualidade é ser realizado em até 15 dias.

## **V.E. Planejamento da Melhoria de Processo (PMP)**

### **Introdução**

Este planejamento tem como objetivo definir o escopo da melhoria de processo, identificando papéis e responsáveis. Juntamente com este planejamento, um cronograma é definindo, contendo prazo e esforço para cada atividade.

### **Escopo**

Analisar o processo Atendimento do Chamado, com o propósito de diminuir o tempo de atendimento para 15 dias, com relação ao tempo de atendimento do analista de negócios, do ponto de vista do cliente, no contexto dos chamados ao Sistema X.

### **Papéis e Responsáveis**

As atividades realizadas na realização deste planejamento de melhoria de processo são relacionadas aos seus respectivos papéis na Tabela V.1.

**Tabela V.1 - Atividades x Papéis**

Atividade	Papéis		
	RM	EE	
Definir objetivo de melhoria	X		
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio	X		
Planejar Melhoria	X		
Selecionar diagrama causal	X		
Coletar informações organizacionais de contexto	X		
Coletar informações de contexto de projetos	X	X	
Identificar fatores pertinentes	X	X	
Construir diagrama causal	X	X	
Avaliar diagrama causal	X	X	
Identificar restrição principal	X		
Elaborar proposta de solução	X	X	
Avaliar viabilidade da proposta de solução	X	X	
Elaborar proposta de melhoria	X		

*Legenda:* **RM** – Responsável do Grupo de Melhoria, **EE** – Equipe Envolvida

A relação dos responsáveis por papéis e objetivos intermediários é descrita na Tabela V.2.

**Tabela V.2 - Responsáveis x Papéis**

Responsável	Papéis		
	RM	EE	
Responsável do Grupo de Melhoria	X		
Membro 1 da equipe		X	
Membro 2 da equipe		X	

*Legenda:* **RM** – Responsável do Grupo de Melhoria, **EE** – Equipe Envolvida

## Orçamento

Atividade	Valor estimado	Valor realizado
Descrever objetivo de melhoria	1h	0,5h
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio	0,5h	0,5h
Planejar Melhoria	3h	2h
Coletar informações organizacionais de contexto	4h	2h
Coletar informações de contexto de projetos	4h	3h
Identificar fatores pertinentes	2h	1h
Construir diagrama causal	4h	
Avaliar diagrama causal	2h	
Identificar restrição principal	1h	
Elaborar proposta de solução	4h	
Avaliar viabilidade da proposta de solução	2h	
Elaborar proposta de melhoria	4h	

## Cronograma

Atividade	Início	Término	Esforço
Definir objetivo de melhoria	17/02	17/02	30m
Avaliar rastreabilidade do objetivo de melhoria dos objetivos de negócio	-	-	-
Planejar Melhoria	01/03	01/03	30m
Selecionar diagrama causal	-	-	-
Coletar informações organizacionais de contexto	05/03	05/03	2h
Coletar informações de contexto de projetos	14/03	14/03	3h
Identificar fatores pertinentes	15/03	15/03	1h16m
Construir diagrama causal	20/03	20/03	1h10m
Avaliar diagrama causal	20/03	20/03	40m
Identificar restrição principal	20/03	20/03	20m
Elaborar proposta de solução	20/03	20/03	1h
Avaliar viabilidade da proposta de solução	20/03	20/03	30m
Elaborar proposta de melhoria	20/03	20/03	1h

## Outros Recursos

- Utilizar pacote Office para elaborar os documentos e ferramenta de construção de árvores
- Ferramenta jThinker para construir os diagramas (árvores)

## V.F. Formulário de Informações Complementares (FIC)

### Introdução

Este formulário contém informações das informações de contexto que apoiaram a identificação de fatores de influência do processo em melhoria.

### Informações de contexto organizacionais

- A ferramenta TFS – *Team Foundations Server*, foi incluída na Organização em 2009, na época da implantação do MPS.BR, nível F. Sua utilização incluía gestão de configuração, IDE de desenvolvimento .net, build, *branches*. A parte de abertura de tarefas foi iniciada em janeiro de 2011, com a substituição de ferramenta de gestão de chamados anterior. Então, os colaboradores começaram a utilizar o Chamado no TFS desde janeiro, como a substituição do processo anterior;

- A nova ferramenta de gestão de chamados, apesar de mais robusta, requeria certo treino para gerar as *queries* de consulta dos chamados abertos assim como seu preenchimento;
- Em Fevereiro e Março, respectivamente, a Organização perdeu dois colaboradores líderes deste produto. A saída dos recursos gerou uma grande baixa de produtividade, já que o produto não era totalmente documentado e possui requisitos (setor de saúde) complexos;
- Os novos recursos contratados não eram seniores na tecnologia, já que o mercado estava aquecido e difícil de contratar, nem no conhecimento específico do negócio. A sua entrada refletiu em queda de produtividade da equipe;
- O processo de implantação do novo nível de maturidade do MPS.BR, nível C, iniciado em junho de 2011;
- Foram contratados dois novos colaboradores para atuar em duas áreas nascentes: Gestão de Configuração do projeto e Teste, entrando em maio e junho de 2011, respectivamente;
- A equipe de desenvolvimento teve um aumento de 30% em maio de 2011;

### **Informações de contexto de projetos**

- O produto X vem sendo estabilizada desde o seu início, em relação aos seus requisitos;
- O produto X também está sendo modularizado, com a ideia dos módulos atuarem de forma independente;
- Os usuários deste software já têm mais de 2 anos de uso, e sua implantação tem seguido a um estágio de estabilidade;
- A ferramenta TFS já se encontra institucionalizada. Ninguém atua sem a abertura de um chamado. Todos conhecem o workflow pré-formatado do chamado;
- A equipe de *helpdesk* tem pleno domínio da ferramenta, para a sua gestão e apoio de chamados de atendimento.

### **Fatores pertinentes**

- Equipe pequena de analista de requisitos;
- Falta de documentação estruturada do produto X legado (caso de uso modelado) e requisitos;

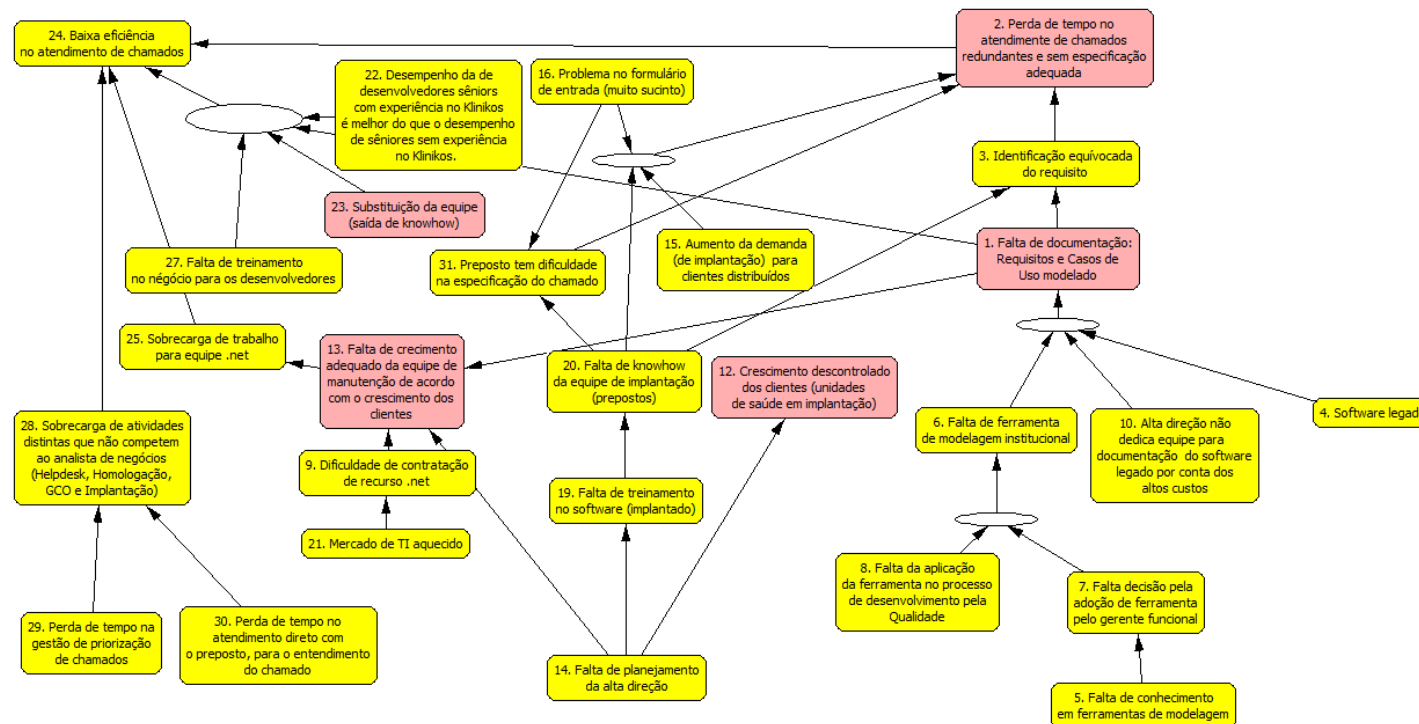
- Repetição de chamados com mesma solicitação;
- Software especialista necessita de muito conhecimento de negócio para ser manipulado sem ocorrer erros;
- A falta de cenários de teste podem gerar problemas na avaliação “exploratória”;
- Crescimento descontrolado de unidades de saúde que utilizam o software, sem o aumento adequado da equipe de manutenção;
- Substituição de tecnologia da ferramenta de atendimento ao chamado;
- Saída de *knowhow* da equipe
- Entrada de novos desenvolvedores na equipe com pouco conhecimento do domínio do produto

# V.G. Formulário de Análise de Restrições (FAR)

## Introdução

Este formulário apresenta o diagrama causal construído para analisar as relações de causa-e-efeito dos fatores envolvidos no processo. O modelo causal utilizado para o diagrama, se existir, é informado, assim como os seus critérios de seleção. Juntamente com o diagrama causal, os fatores críticos e principais restrições acompanham as soluções propostas de melhoria.

## Diagrama Causal



<http://code.google.com/p/jthinker>



## Matriz de Rastreabilidade das Restrições Candidatas

	1	2	3	6	7	9	12	13	19	20	24	25	28	31	Grau de impacto
21.Mercado de TI aquecido						x		x			x	x			28,57%
<b>14.Falta de planejamento da alta direção</b>		x	x				x	x	x	x	x	x			<b>57,14%</b>
04.Software legado	x	x	x								x				28,57%
08.Falta de aplicação da ferramenta no processo de desenvolvimento pela qualidade	x	x	x	x							x				35,71%
<b>05.Falta de conhecimento em ferramenta de modelagem</b>	x	x	x	x	x						x		x		<b>50,00%</b>
27.Falta de treinamento no negócio para os desenvolvedores											x				7,14%
10.A alta direção não dedica equipe para a documentação do software legado por conta dos altos custos	x	x	x								x				28,57%
23.Substituição da equipe (saída de knowhow)											x				7,14%
16.Problema no formulário de entrada (muito sucinto)		x									x			x	21,43%
15. Aumento de demanda (de implantação) para clientes distribuídos		x									x				14,29%
29. Perda de tempo na gestão de priorização de chamados											x		x		14,29%
30. Perda de tempo no atendimento direto com o preposto, para o atendimento do chamado											x		x		14,29%
22. Desempenho de desenvolvedores sêniores com experiência no produto X é melhor do que o desempenho de sêniores sem experiência no mesmo produto.											x				7,14%

### Restrição Principal

- Falta de conhecimento em ferramenta de modelagem

**Raciocínio de seleção da restrição principal (se aplicável – desempate de grau de impacto)**

A falta de planejamento que considerasse o crescimento agressivo de vendas e serviços vinculados ao produto X foi a causa raiz identificado. No entanto, esta causa influenciou o desempenho atual do processo e não pode ser corrigida de forma que o desempenho seja melhorado. Sua causa-raiz pode ser tratada que é o planejamento estratégico da organização. Esta medida já está sendo tomada e tem um resultado a longo prazo.

A falta de conhecimento de uma ferramenta de modelagem que apoie a modelagem de casos de uso e a definição dos requisitos foi secundariamente a restrição mais forte ao desempenho e pode ter sua causa-raiz trabalha de forma a melhorar o desempenho atual do processo.

**Soluções propostas**

**Solução 1**

**Diagrama de Resolução de Conflito**

Não se aplica, pois não há conflito a ser resolvido.

**Pressupostos identificados**

Não se aplica

**Pressuposto selecionado da proposição de solução**

Não se aplica

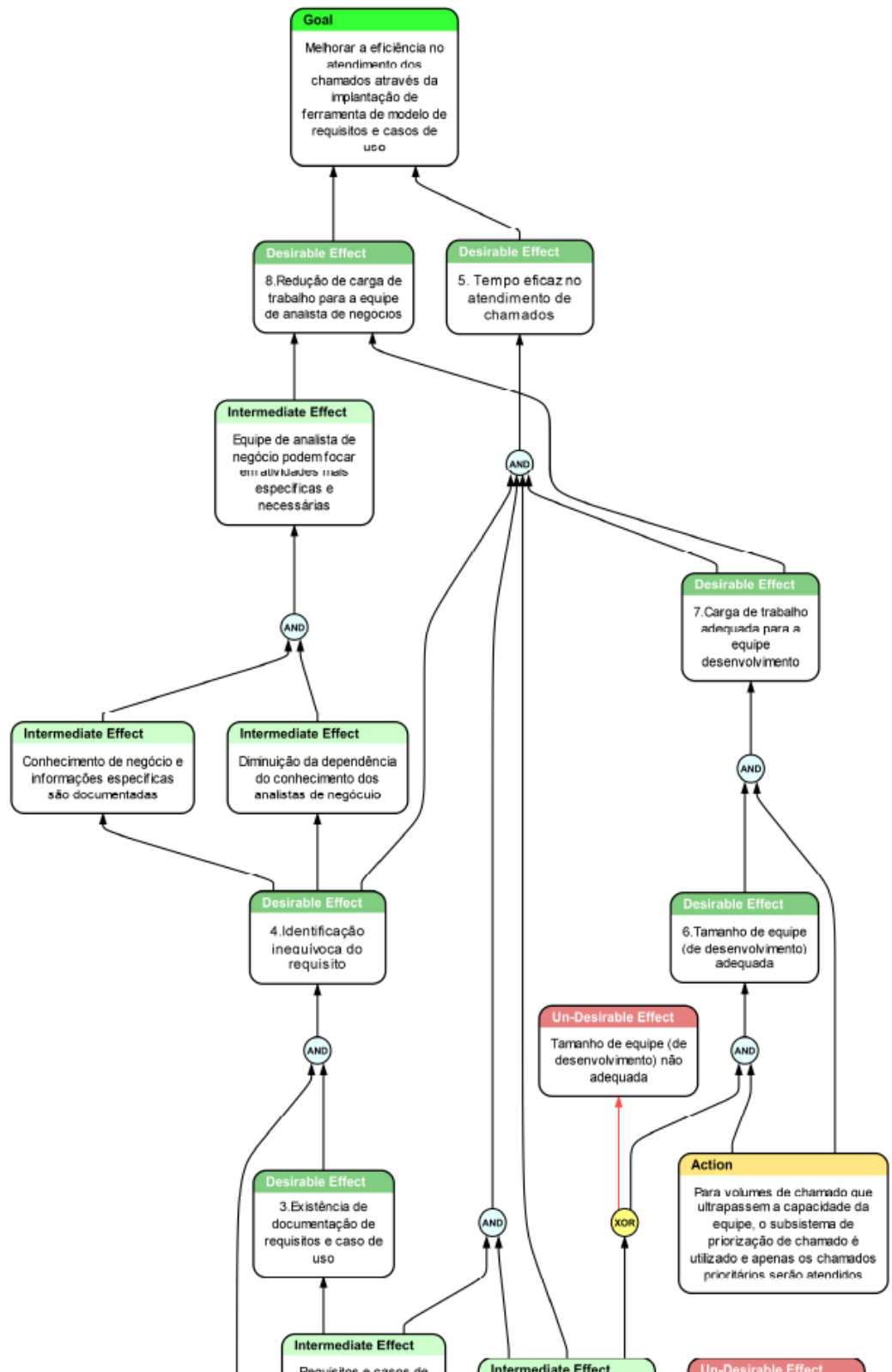
**Objetivo da solução proposta**

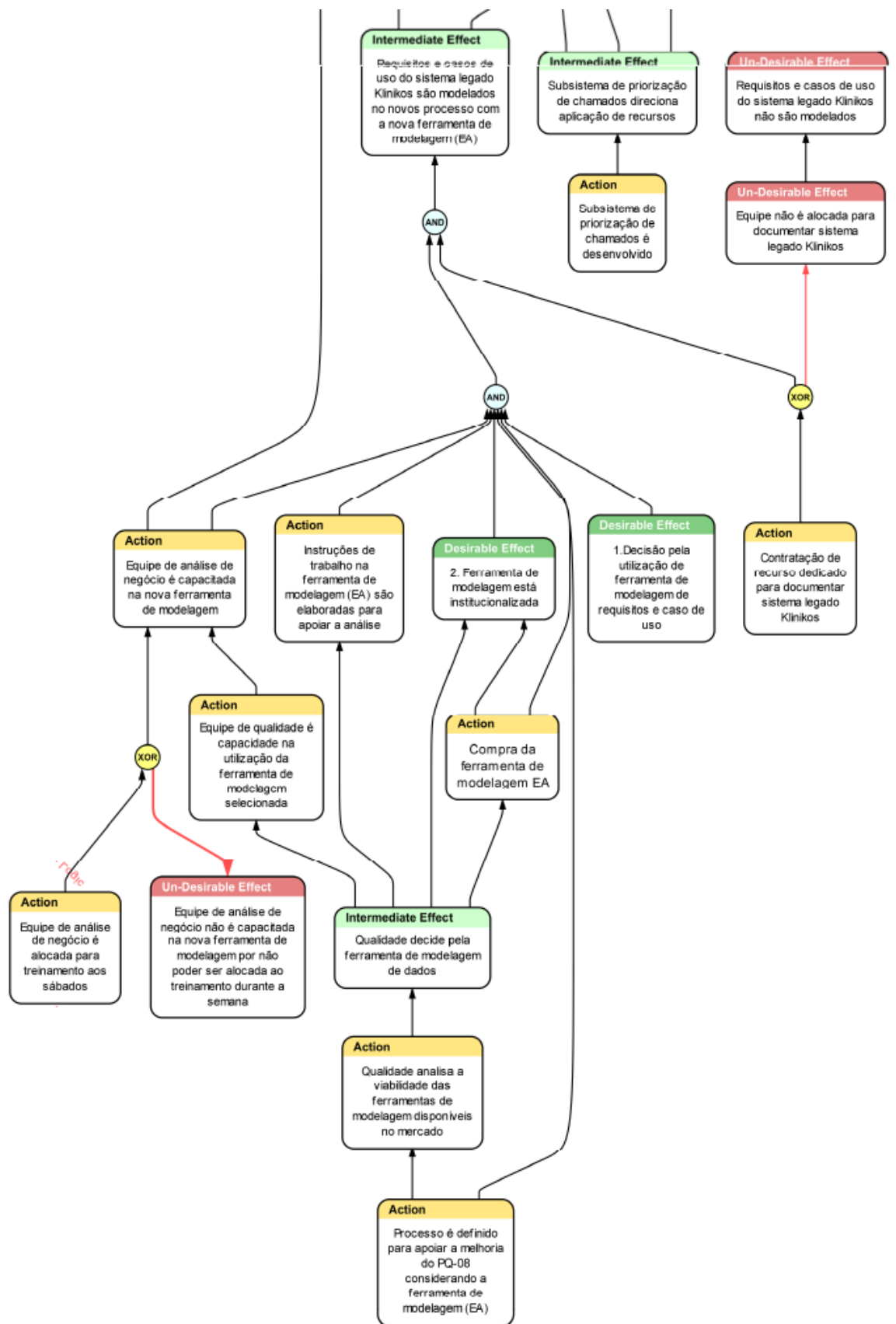
Melhorar a eficiência no atendimento do chamado através da implantação de uma ferramenta de modelagem de requisitos e caso de uso para documentar o software legado e o software mais recente.

**Raciocínio de definição da proposta de solução (opcional)**

Minimizar o impacto do desenvolvimento deste software especialista, que fica dependente dos analistas de negócio, que conhecem o aplicativo.

## Análise de Viabilidade da Proposta de Solução (Árvore da Realidade Futura)





### Efeitos desejáveis e Fatores de influência desejáveis

1. Decisão pela utilização de ferramenta de modelagem de requisitos e caso de

uso;

2. Ferramenta de modelagem está institucionalizada;
3. Existência de documentação de requisitos e caso de uso;
4. Identificação inequívoca do requisito;
5. Tempo eficaz no atendimento de chamados;
6. Tamanho de equipe (de desenvolvimento) adequada;
7. Carga de trabalho adequada para a equipe desenvolvimento;
8. Redução de carga de trabalho para a equipe de analista de negócios.

#### **Riscos envolvidos (obstáculos/efeitos colaterais)**

<b>Risco</b>	<b>Tipo<sup>33</sup></b>	<b>Ações de mitigação</b>
Equipe de análise de negócio não é capacitada na nova ferramenta de modelagem por não poder ser alocada ao treinamento durante a semana	E	Equipe de análise de negócio é alocada para treinamento aos sábados
Requisitos e casos de uso do sistema legado X não são modelados	E	Contratação de recurso dedicado para documentar sistema legado X
Tamanho de equipe (de desenvolvimento) não adequada	O	Para volumes de chamado que ultrapassem a capacidade da equipe, o subsistema de priorização de chamado é utilizado e apenas os chamados prioritários serão atendidos
	O	Subsistema de priorização de chamados direciona aplicação de recursos

#### **Cronograma de marcos**

---

<sup>33</sup> Tipo: O – Obstáculo, E – Efeito colateral.

<b>Marco</b>	<b>Esforço</b>	<b>Precedência</b>
1. Definir processo considerando a ferramenta de modelagem		
2. Analisar a viabilidade das ferramentas de modelagem disponíveis no mercado		1
3. Reservar alocação de recursos para treinamento no sábado		
4. Compra de ferramenta de modelagem		2
5. Capacitar equipe de qualidade a utilizar a ferramenta de modelagem		2
6. Alocar equipe de análise de negócio para treinamento da ferramenta de modelagem aos sábados		3,5
7. Elaborar instruções de trabalho para ferramenta de modelagem		4
8. Contratação de recurso dedicado para documentação do sistema X legado		
9. Desenvolvimento de módulo de priorização de chamados		
10. Implantar o módulo de priorização para apoiar a seleção de chamados de backlog		

### **Recursos estimados**

N/A

### **Orçamento estimado**

<b>Marco/Recurso</b>	<b>Valor estimado</b>
1. Definir processo considerando a ferramenta de modelagem	Não informado
2. Analisar a viabilidade das ferramentas de modelagem disponíveis no mercado	Não informado
3. Reservar alocação de recursos para treinamento no sábado	Não informado
4. Compra de ferramenta de modelagem	Não informado
5. Capacitar equipe de qualidade a utilizar a ferramenta de modelagem	Não informado
6. Alocar equipe de análise de negócio para treinamento da ferramenta de modelagem aos sábados	Não informado
7. Elaborar instruções de trabalho para ferramenta de modelagem	Não informado
8. Contratação de recurso dedicado para documentação do sistema X legado	Não informado
9. Desenvolvimento de módulo de priorização de chamados	Não informado
10. Implantar o módulo de priorização para apoiar a seleção de chamados de backlog	Não informado

### **Laudo de viabilidade da proposta de solução**

N/A