

**UNIVERSIDADE PAULISTA – UNIP**

**TOMADA DE DECISÃO EM FÁBRICA DE SOFTWARE  
BASEADA EM CONTROLE ESTATÍSTICO DE PROCESSO**

**WALTER HENRIQUE DE FARIAS SILVA**

Dissertação apresentada ao programa  
de Pós-graduação em Engenharia de  
Produção da Universidade Paulista –  
UNIP para a obtenção do título de  
Mestre em Engenharia de Produção.

Orientador: Prof. Dr. Mauro de  
Mesquita Spinola

**SÃO PAULO  
2007**

**UNIVERSIDADE PAULISTA – UNIP**

**TOMADA DE DECISÃO EM FÁBRICA DE SOFTWARE  
BASEADA EM CONTROLE ESTATÍSTICO DE PROCESSO**

**WALTER HENRIQUE DE FARIAS SILVA**

Orientador: Prof. Dr. Mauro de Mesquita Spinola.

Área de Concentração: Gestão da Informação.

Dissertação apresentada ao Programa de Pós-graduação em Engenharia de Produção da Universidade Paulista – UNIP para a obtenção do título de Mestre em Engenharia de Produção.

SÃO PAULO  
2007

SILVA, Walter Henrique de Farias

Tomada de decisão em fábrica de *software* baseada em  
controle estatístico de processo. /

Walter Henrique de Farias Silva. São Paulo, 2007.

109 p.

Dissertação (Mestrado) – Universidade Paulista, 2007.

Área de Concentração: Gestão da Informação

Linha de Pesquisa: Produção de Software

Orientador: Prof. Dr. Mauro de Mesquita Spinola

1. Controle Estatístico de Processo.
2. Fábrica de Software.
3. Melhoria de Processo.

**Tomada de decisão em fábrica de software baseada em CEP**

---

Dedicatória

À minha esposa, Ana Carolina, pelo amor,  
pela dedicação, pelo estímulo e por sua visão  
crítica na revisão deste trabalho.

Aos meus pais que são a base de toda a  
minha formação.

**Tomada de decisão em fábrica de software baseada em CEP**

---

Agradecimentos

Aos professores Mauro de Mesquita Spinola e Pedro Luiz de Oliveira Costa Neto, pela valiosa orientação, durante a realização deste trabalho, pelo estímulo constante, amizade e apoio em todos os momentos.

Ao David Card, criador do PSM pela sua contribuição na validação dos conceitos e relacionamento do CEP e o PSM.

A todos os familiares e amigos que sempre estiveram ao meu lado, nesse período de pós-graduação, sempre com palavras de apoio e incentivo para que eu prosseguisse.

A todos os funcionários da empresa, pela sua amizade e valiosa colaboração durante a realização deste trabalho.

A todos os professores da Pós-Graduação em Engenharia de Produção que diretamente contribuíram para a minha formação.

Aos meus amigos e colegas do Curso de Pós-Graduação em Engenharia de Produção, especialmente Regina Asato, Simone Canuto e Rosana Kronig pelos alegres e bons momentos de convivência.

**Tomada de decisão em fábrica de software baseada em CEP**

Sumário

**ÍNDICE**

<b>RESUMO .....</b>	<b>VIII</b>
<b>ABSTRACT .....</b>	<b>IX</b>
<b>LISTA DE ABREVIACÕES, SIGLAS E SÍMBOLOS.....</b>	<b>X</b>
<b>LISTA DE FIGURAS .....</b>	<b>XI</b>
<b>LISTA DE QUADROS .....</b>	<b>XIII</b>
<b>1. INTRODUÇÃO .....</b>	<b>14</b>
1.1. Objetivos da pesquisa.....	16
1.2. Justificativa da pesquisa .....	16
1.3. Estrutura do trabalho.....	18
<b>2. Metodologia de pesquisa.....</b>	<b>20</b>
2.1. Questões de pesquisa .....	25
2.2. Processo de pesquisa.....	26
<b>3. Fundamentos .....</b>	<b>30</b>
3.1. Conceito de qualidade .....	31
3.2. Medição Prática de Software e Sistemas ( <i>Practical Software and Systems Measurement</i> ) .....	36
3.2.1. Modelo de medida da informação.....	38
3.2.2. Modelo de medição de processo .....	41
3.3. Controle estatístico de processo .....	43
3.3.1. Gráficos de Controle .....	47
3.3.1.1. Gráficos de controle para variáveis .....	50
3.3.1.1.1. Gráficos de controle de $X$ e $R$ .....	50
3.3.1.1.2. Gráficos da média e desvio-padrão ( $x$ -barra e $S$ ).....	51
3.3.1.1.3. Gráficos do valor individual e amplitude móvel ( $X$ e $Rm$ ) .....	53
3.3.1.2. Gráficos de controle para atributos .....	54
3.3.1.2.1. Gráficos da fração defeituosa ( $p$ ) .....	54
3.3.1.2.2. Gráfico do número de defeituosos na amostra ( $np$ ) .....	55
3.3.1.2.3. Gráfico do número de defeitos na amostra ( $c$ ) .....	55
3.3.1.2.4. Gráfico do número de defeitos por unidade de inspeção ( $u$ )....	56
3.3.2. Estabilidade do processo .....	58

**Tomada de decisão em fábrica de software baseada em CEP**

Sumário

3.3.3. Seleção do gráfico de controle para atributos .....	60
3.3.4. Aplicação do CEP em processos de software .....	61
3.4. Fábrica de Software .....	64
3.4.1. Indicadores de desempenho.....	69
4. Modelo de Medição de Referência .....	73
5. Primeiro ciclo da pesquisa-ação: análise dos indicadores .....	79
5.1. Desvio de escopo.....	79
5.2. Defeitos por caso de uso .....	83
5.3. Produtividade .....	86
6. Segundo ciclo da pesquisa-ação: melhoria do modelo de referência ...	89
6.1. Desvio de escopo.....	89
6.2. Defeitos por caso de uso .....	96
7. Conclusões .....	102
7.1. Principais resultados e contribuições .....	103
7.2. Limitações da pesquisa.....	104
7.3. Trabalhos futuros .....	105
Referencias Bibliográficas .....	106

**Tomada de decisão em fábrica de software baseada em CEP**

---

Resumo

**RESUMO**

SILVA, Walter Henrique de Farias. **Tomada de Decisão em Fábrica de Software baseada em Controle Estatístico de Processo.** Dissertação (Mestrado em Engenharia de produção). Universidade Paulista, 2007.

Palavras-chave: **fábrica de software, controle estatístico de processo, medição e análise, qualidade de processo.**

As fábricas de software têm como objetivo principal a melhoria dos seus processos para racionalizar a utilização dos recursos e diminuir os seus custos operacionais. Essas empresas possuem processos de desenvolvimento que de alguma forma proporcionam previsibilidade e diminuição de retrabalho, procurando sempre diminuir os custos. Avaliar se esse processo de desenvolvimento de software está sendo seguido é um desafio para quem os produz. Esta pesquisa tem o objetivo de propor um mecanismo para que o tomador de decisão da fábrica de software disponha da informação necessária para melhoria do seu processo produtivo. Esse modelo foi criado com base nos conceitos de PSM, CEP e fábrica de software. Com base em um modelo de referência, baseado nas necessidades de uma fábrica, foram selecionados três projetos-projeto-pilotos para aplicação do modelo. Após a coleta dos dados das aplicações, criaram-se gráficos de controle e as decisões de melhoria foram tomadas. A principal conclusão é que o controle estatístico de processo teve um impacto positivo na tomada de decisão devido à facilidade de interpretação.

**Tomada de decisão em fábrica de software baseada em CEP**

---

Abstract

**ABSTRACT**

SILVA, Walter Henrique de Farias. **Make of Decision in Software Factory based in Statistical Process Control.** Dissertation (Master of Science in Production Engineering). Universidade Paulista, 2007.

Key words: **software factory, statistical process control, measurement and analysis, process quality.**

The software factories have as goal main the improvement of its processes to rationalize the use of the resources and to diminish its operational costs. These companies possess development process that of some form provides to previsibility and reduction of re-work, looking for always to diminish the costs. To evaluate if this process of software development this being been followed is a challenge for these companies. This research has the objective to consider a mechanism so that the borrower of decision of the software factories can have the necessary information for improvement of its productive process. This model was created on the basis of the concepts of PSM, SPC and Software Factory. After the creation of the model of reference, based on the necessities of a software factory, had been selected 3 projects pilots for application of the model. After the collection of the data of the model, the control graphs had been created and the improvement decisions had been taken. The statistical process control had a positive impact in the taking of decision due the interpretation easiness.

**Tomada de decisão em fábrica de software baseada em CEP**

---

Lista de Abreviaturas, Siglas e Símbolo

**LISTA DE ABREVIACÕES.**

PSM = Pratical Software and Systems Measurement

CEP = Controle estatístico de processo

CMMI = Capability Maturity Model Integration

ISO = International Organization for Standardization

SPICE = Software Process Improvement Capability dEtermination

IEEE = Institute of Electrical and Electronic Engineering

GQM = Goal Question Metrics

**Tomada de decisão em fábrica de software baseada em CEP**

Lista de Figuras

**LISTA DE FIGURAS**

Figura 1 – Processo para construção da pesquisa.....	26
Figura 2 – Relacionamento do modelo de medida da informação.....	38
Figura 3 – Evolução da necessidade de informação dentro do plano de medição .....	41
Figura 4 – Modelo de Processo de Medição.....	42
Figura 5 – Um processo estatisticamente controlado .....	45
Figura 6 – Um processo fora de controle estatístico.....	46
Figura 7 – Gráfico de controle.....	48
Figura 8 – Passos para avaliar a estabilidade do processo.....	59
Figura 9 – Fluxograma para seleção de gráficos para atributos (Fonte: RAMOS, 2002, p.325) .....	60
Figura 10 – Gráfico de desvio de escopo do Projeto-piloto 1 .....	80
Figura 11 – Gráfico de desvio de escopo do Projeto-piloto 1, ajustado .....	82
Figura 12 – Gráfico de desvio de escopo do Projeto-piloto 2 .....	83
Figura 13 – Gráfico de defeitos por caso de uso do projeto-piloto 3.....	84
Figura 14 – Gráfico de defeitos por caso de uso do projeto-piloto 3, sem os pontos 5 e 6 .....	85
Figura 15 – Gráfico de defeitos por caso de uso do projeto-piloto 3, sem os pontos 21 e 22 .....	86
Figura 16 – Gráfico X para o indicador de produtividade na codificação Java .....	87
Figura 17 – Gráfico Rm para o indicador de produtividade na codificação Java .....	88
Figura 18 – Gráfico Rm para o indicador de desvio de escopo do projeto-piloto 1 .....	90
Figura 19 – Gráfico X para o indicador de desvio de escopo do projeto-piloto 1 .....	91
Figura 20 – Gráfico Rm para o indicador de desvio de escopo do projeto-piloto .....	91

**Tomada de decisão em fábrica de software baseada em CEP**

**Lista de Figuras**

Figura 21 – Gráfico <i>X</i> para o indicador de desvio de escopo do projeto-piloto 2.....	92
Figura 22 – Gráfico <i>Rm</i> para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 7 da figura 20.....	93
Figura 23 – Gráfico <i>X</i> para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 7 da figura 20 .....	94
Figura 24 – Gráfico <i>Rm</i> para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 21 e 25 da figura 22.....	95
Figura 25 – Gráfico <i>X</i> para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 21 e 25 da figura 22 .....	96
Figura 26 – Gráfico <i>Rm</i> para o indicador de defeitos por caso de uso .....	97
Figura 27 – Gráfico <i>X</i> para o indicador de defeitos por caso de uso .....	97
Figura 28 – Gráfico <i>Rm</i> para o indicador de defeitos por caso de uso, sem as causas especiais da figura 27 .....	99
Figura 29 – Gráfico <i>X</i> para o indicador de defeitos por caso de uso, sem as causas especiais da figura 27 .....	99
Figura 30 – Gráfico <i>Rm</i> para o indicador de defeitos por caso de uso, sem as causas especiais da figura 29.....	100
Figura 31 – Gráfico <i>X</i> para o indicador de defeitos por caso de uso, sem as causas especiais da figura 29.....	101

**Tomada de decisão em fábrica de software baseada em CEP**

---

Lista de Quadros

**LISTA DE QUADROS**

Quadro 1 – Medições operacionais, baseado em Fernandes (2004) .....	70
Quadro 2 – Construção mensurável para necessidade de informação quantidade de defeitos produzidos por caso de uso.....	74
Quadro 3 – Construção mensurável para necessidade de informação produtividade das equipes de desenvolvimento .....	75
Quadro 4 – Construção mensurável para necessidade de informação desvio de escopo .....	76

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 1. Introdução

#### **1. INTRODUÇÃO**

Mesmo com o avanço crescente da tecnologia e dos processos que apoiam o desenvolvimento de software, seu gerenciamento e sua melhoria são desafios para as empresas prestadoras de serviços nessa área. Essas empresas buscam o aprimoramento dos seus processos, no entanto, saber quais deles precisam ser aprimorados e se as correções alcançaram os resultados esperados é o objetivo destas desenvolvedoras de software.

A obtenção de certificados internacionais de qualidade, por exemplo, CMMI (*Capability Maturity Model Integration*), ISO-9000 (*International Organization for Standardization*), ISO/IEC 15504 (SPICE - *Software Process Improvement Capability dEtermination*) ajuda a demonstrar qualidade ao cliente. O certificado de qualidade é uma maneira de selecionar fornecedores e, muitas vezes, é um pré-requisito para o começo de um novo negócio. Entretanto, ter o certificado não é garantia de boa qualidade mas, não tê-lo, é uma desvantagem competitiva. Assim, muitos engenheiros de software afirmam que uma forma efetiva de assegurar a qualidade do produto de software é assegurar a qualidade do processo de desenvolvimento usado para gerar esse produto (KITCHENHAM, 1996) (PFLEEEGER, 1998) (HUMPREY, 1989) (HARVEY, 1994).

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 1. Introdução

Segundo o padrão IEEE – *Standard Glossary of Software Engineering Terminology* (IEEE, 1990), um processo é definido como uma seqüência de passos realizados para um determinado propósito. Já o processo de software é um conjunto de atividades, métodos, práticas e tecnologias que as pessoas utilizam para desenvolvê-lo e mantê-lo abrangendo também os produtos a ele relacionados (PAULK, 1993). Se alguma parte do processo for negligenciada, o produto final é prejudicado. Portanto, o foco na qualidade do processo de desenvolvimento de software e em sua manutenção é tão importante quanto o foco na qualidade do produto final.

Para assegurar a qualidade do processo é preciso controlá-lo. Um corolário de Tom DeMarco nos diz que “você não pode controlar aquilo que você não mede” (DEMARCO, 1982). Medição é o instrumento chave para caracterizar, avaliar, prever e melhorar o processo de desenvolvimento de software. Um programa de medição permite às pessoas detectar tendências, antecipar problemas e melhorar a qualidade do produto, melhorando a qualidade do processo. O uso de medidas pelas organizações propicia a informação correta para a tomada de decisão que contribui para atingir as metas do negócio. Já o uso de medidas nos projetos permite avaliar e controlar o desempenho do processo de software definido para o projeto. Além disso, o mero fato de medir possibilita identificar os pontos de desvio que servirão de base para aprimorar o processo de desenvolvimento de software.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 1. Introdução

#### **1.1. Objetivos da pesquisa**

O objetivo principal desta pesquisa é apresentar um modelo de medição e análise dos indicadores baseados no controle estatístico para aprimoramento do processo de desenvolvimento de software utilizado em uma fábrica de software. Esse modelo de medição consiste em definir os indicadores críticos, coletar e validar os dados dos indicadores escolhidos pela organização empregando como base o PSM (*Practical Software and System Measurement*); analisar estatisticamente os dados com técnicas apropriadas – controle estatístico de processo; identificar as causas especiais de processos e encaminhar ações corretivas aos responsáveis, para que os desvios sejam corrigidos e removidos.

#### **1.2. Justificativa da pesquisa**

Um processo de software de uma fábrica de software deve focar a remoção de defeitos o mais breve possível registrando os dados sobre ao encontrá-los; por isso as inspeções de software são relevantes. Além disso, a organização deve usar esses dados de inspeção não somente para avaliar a qualidade do produto como também para melhorar o processo de desenvolvimento (BURTON, 2000) (HERBSLEB, 1997).

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 1. Introdução

Segundo HUMPREY (1988), é importante ressaltar que um processo de software ideal deve ter as seguintes características: 1) ser previsível, ou seja, ter custo estimado de forma precisa, ter compromisso com o cronograma estabelecido e garantir boa qualidade nos produtos resultantes; 2) propiciar controle estatístico de processo – CEP, princípio básico da gerência de processo, que pode antecipar custo, tempo e qualidade; e 3) fornecer medidas, uma vez que não é possível ter CEP sem medidas, e executá-las de maneira alinhada às metas e às necessidades da organização.

Portanto, organizações que usam medidas são capazes de entender melhor seus problemas e suas capacidades reais, manter compromissos firmados, prever tendências, antecipar imprevistos e melhorar o processo de desenvolvimento de software. Ou seja, as medidas permitem entender o processo de desenvolvimento e os produtos que ele gera (PRESSMAN, 1997), (PFLEEGER, 1998) e tomar decisões mais seguras e eficientes (CARD, 2003a), (JONES, 2003).

A justificativa para realização desta pesquisa está associada à melhoria de processo em uma fábrica de software, ou seja, identificar pontos aperfeiçoamento no processo. Para isso, os seguintes objetivos devem ser alcançados:

1. Selecionar os indicadores para aprimoramento no processo.
2. Identificar os pontos de aperfeiçoamento nos processos selecionados.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **1. Introdução**

3. Determinar o plano de correção para cada ponto de melhoria do processo.

Nesse contexto, o modelo de referência tem como objetivo criar um mecanismo padrão para coleta de dados e interpretação dos gráficos de controle para aperfeiçoamento do processo de desenvolvimento. A fábrica de software estudada nesta pesquisa possui um processo de desenvolvimento há cerca de um ano e está procurando aprimorá-lo para atender às novas demanda de seus clientes.

### **1.3. Estrutura do trabalho**

Para atender o objetivo desta pesquisa o trabalho foi dividido em seis capítulos que estão estruturados da seguinte forma:

- Capítulo 1 Introdução

Esse capítulo apresenta o assunto abordado na pesquisa, a justificativa, e a estrutura do trabalho.

- Capítulo 2 Metodologia

Descreve em detalhes a metodologia de pesquisa adotada.

- Capítulo 3 Fundamentos

Com base no problema descrito no capítulo 1 foram relacionados conceitos que serviram de base para a construção das hipóteses.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **1. Introdução**

Nesse capítulo os conceitos fundamentais para a formulação do modelo de medição.

- Capítulo 4 Modelo de medição de Referência

Esse capítulo apresenta o modelo de medição que foi utilizado como referência nos projetos-projeto-pilotos em uma fábrica de software brasileira.

- Capítulo 5 Primeiro ciclo da pesquisa-ação (análise dos indicadores)

Descreve os resultados da aplicação do modelo de referência e as análises dos gráficos de controle para os indicadores do modelo de referência.

- Capítulo 6 Segundo ciclo da pesquisa-ação: melhoria do modelo de referência

Aborda o segundo ciclo de pesquisa-ação para realizar a melhoria no modelo de referência.

- Capítulo 7 Conclusões

Contém a contribuição dos conceitos utilizados no capítulo 3 na aplicação do modelo de medição de referência, descrito no capítulo 4, e sugestões de trabalhos futuros.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 2. Metodologia de pesquisa

#### **2. Metodologia de pesquisa**

Segundo THIOLLENT (2005), a pesquisa-ação é um tipo de pesquisa social com base empírica que é concebida e realizada em estreita associação a uma ação ou resolução de um problema coletivo, no qual os pesquisadores e os participantes representativos da situação ou do problema estão envolvidos de modo cooperativo ou participativo. Resume ainda os seguintes aspectos na aplicação da estratégia metodológica:

- Ampla e explícita interação entre pesquisadores e pessoas implicadas na situação investigada.
- A interação resulta no consenso referente à priorização de soluções e ações concretas.
- O objeto de investigação é resultado da situação social e problemas encontrados.
- O objetivo da pesquisa-ação consiste em resolver ou pelo menos esclarecer os problemas da situação observada.
- Há um acompanhamento das decisões e ações e de toda a atividade intencional dos atores da situação.
- A pesquisa não se limita a uma forma de ação: pretende-se aumentar o conhecimento dos pesquisadores e o conhecimento ou o nível de consciência das pessoas e grupos considerados.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **2. Metodologia de pesquisa**

A pesquisa-ação, ainda segundo THIOLLENT (2005), é configurada dependendo dos objetivos e do contexto. O autor identifica os seguintes casos:

- A pesquisa-ação é organizada para realizar os objetivos práticos de um ator social homogêneo que disponha de suficiente autonomia para encomendar e controlar a pesquisa.
- A pesquisa-ação é realizada dentro de uma organização na qual existe hierarquia ou grupos cujos relacionamentos apresentam divergências.
- A pesquisa-ação é organizada em meio aberto em função de instituições exteriores à comunidade. Os pesquisadores elucidam os diversos interesses implicados.

Em resumo, na pesquisa-ação existem objetivos práticos de natureza bastante imediata: propor soluções quando possível e acompanhar ações correspondentes, ou pelo menos fazer progredir a consciência dos participantes no que diz respeito à existência de soluções e obstáculos (THIOLLENT, 2005).

Segundo THIOLLENT (2005), a pesquisa-ação é um método ou uma estratégia de pesquisa que estabelece uma estrutura coletiva, participativa e ativa visando a captação de informações. Para que não haja ambigüidade, uma pesquisa pode ser qualificada de pesquisa-ação quando houver realmente uma ação por parte das pessoas ou grupos implicados no problema em observação. Além disso, é preciso que a ação seja não-trivial,

**Tomada de decisão em fábrica de software baseada em CEP**

---

2. Metodologia de pesquisa

ou seja, constitua uma ação problemática que mereça investigação a ser elaborada e conduzida (THIOLLENT, 2005).

Segundo TRIPP (2005), é importante que se reconheça a pesquisa-ação como um dos inúmeros tipos de investigação-ação, que é um termo genérico para qualquer processo que siga um ciclo no qual se aprimora a prática pela oscilação sistemática entre agir no campo da prática e investigar a respeito dela. Planeja-se uma mudança para a melhora de sua prática, aprendendo mais, no correr do processo, tanto a respeito da prática quanto da própria investigação. TRIPP (2005) define a pesquisa-ação como uma forma de investigação-ação que emprega técnicas de pesquisa consagradas para informar a ação que se decide tomar para melhorar a prática.

A questão é que a pesquisa-ação requer ação tanto nas áreas da prática quanto da pesquisa, de modo que, em maior ou menor medida, terá características tanto da prática rotineira quanto da pesquisa científica (TRIPP, 2005).

O ciclo da pesquisa-ação inclui todas as atividades do ciclo básico de investigação-ação e, freqüentemente, é representado do mesmo modo, e embora pareça de início suficientemente claro, não é inteiramente preciso em sua distinção e em seu seqüenciamento da ação e do monitoramento das fases. Na maioria dos tipos de investigação-ação, habitualmente se monitoram os efeitos de sua própria ação durante a fase de ação, enquanto na pesquisa-ação se produzirão dados sobre os efeitos de uma mudança da prática durante a implementação, ocorrendo ambos antes e depois da implementação (TRIPP, 2005).

### **Tomada de decisão em fábrica de software baseada em CEP**

---

#### **2. Metodologia de pesquisa**

A nomenclatura também constitui problema porque planejamento, monitoramento e avaliação são formas diferentes de ação, de modo que a implementação é mais adequada para o que é chamado, muitas vezes, de fase de ação (TRIPP, 2005).

Segundo esse autor, fica mais claro representar o ciclo da pesquisa-ação como uma seqüência de três fases de ação nos dois diferentes campos da prática e da investigação sobre a prática. Também se torna explícito que se deve planejar tanto para a mudança na prática, quanto para a avaliação dos efeitos dessa mudança. Ainda, segundo TRIPP (2005), isso é importante na pesquisa-ação, porque o planejamento de como avaliar os efeitos da mudança na prática é, em geral, muito mais rigoroso do que em muitos outros tipos de investigação-ação.

O reconhecimento é uma análise situacional que produz ampla visão do contexto da pesquisa-ação, práticas atuais, dos participantes e envolvidos. Paralelamente a projetar e implementar a mudança para melhoria da prática, o reconhecimento segue exatamente o mesmo ciclo da pesquisa-ação, planejando como monitorar e avaliar a situação atual, e, a seguir, interpretando e avaliando os resultados a fim de elaborar uma mudança adequada da prática no primeiro ciclo de pesquisa-ação de melhoria (TRIPP, 2005).

A natureza iterativa do processo de investigação-ação talvez seja a sua característica isolada mais distintiva. Muito embora todos os processos de melhoria e desenvolvimento tendam a incluir todas as fases do ciclo básico de investigação-ação, nem todos o fazem na mesma seqüência nem

### **Tomada de decisão em fábrica de software baseada em CEP**

---

2. Metodologia de pesquisa

repetem o ciclo de uma maneira corrente, para realizar o aperfeiçoamento de modo incremental. A maioria das soluções de problemas, por exemplo, o desenvolvimento organizacional ou a pesquisa experimental, não é investigação-ação segundo esse critério. A pesquisa-ação, como uma forma de investigação-ação, é um processo corrente, repetitivo, no qual o que se alcança em cada ciclo fornece o ponto de partida para mais melhoria no seguinte (TRIPP, 2005).

Segundo FRANCO (2005), quando se pretende investigar a dimensão da ação na pesquisa-ação, tem-se também por finalidade refletir seu sentido, suas configurações, bem como seu “entranhamento” no processo investigativo. Nessa direção, tem-se a preocupação de identificar as ações necessárias à construção/compreensão do objeto de estudo em questão, bem como as ações fundamentais para transformar tais compreensões em produção de conhecimento. Portanto, o grande interesse é permitir conhecer as ações necessárias à compreensão dos processos que estruturam a pedagogia da mudança da práxis na situação em investigação.

Considerando-se a pesquisa-ação um processo eminentemente interativo, a análise da qualidade da ação entre os sujeitos que dela participam é fundamental para definir sua pertinência epistemológica e seu potencial praxiológico (FRANCO, 2005).

Como o objetivo desta pesquisa é a tomada de decisão para melhoria de processos em uma fábrica de software utilizaremos a pesquisa-ação como estrutura metodológica. A pesquisa-ação foi escolhida por sua característica de interatividade entre o objeto pesquisado e o pesquisador.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 2. Metodologia de pesquisa

#### **2.1. Questões de pesquisa**

A questão principal desta pesquisa é como um gerente de fábrica de software pode selecionar os processos que serão medidos e como tomar decisões de melhoria de processos baseado nas técnicas de CEP.

Para guiar a pesquisa-ação foram levantadas as seguintes questões de pesquisa:

- Como identificar pontos de melhoria no processo?
- Como saber se o processo que foi implementado está sendo seguido por todos os membros do time?
- Como identificar os indicadores para um gerente de fábrica de software monitorar a execução dos projetos?
- Após a identificação dos pontos de melhoria e aplicar as ações para correção do processo, como saber se essas ações realmente tiveram os efeitos esperados?

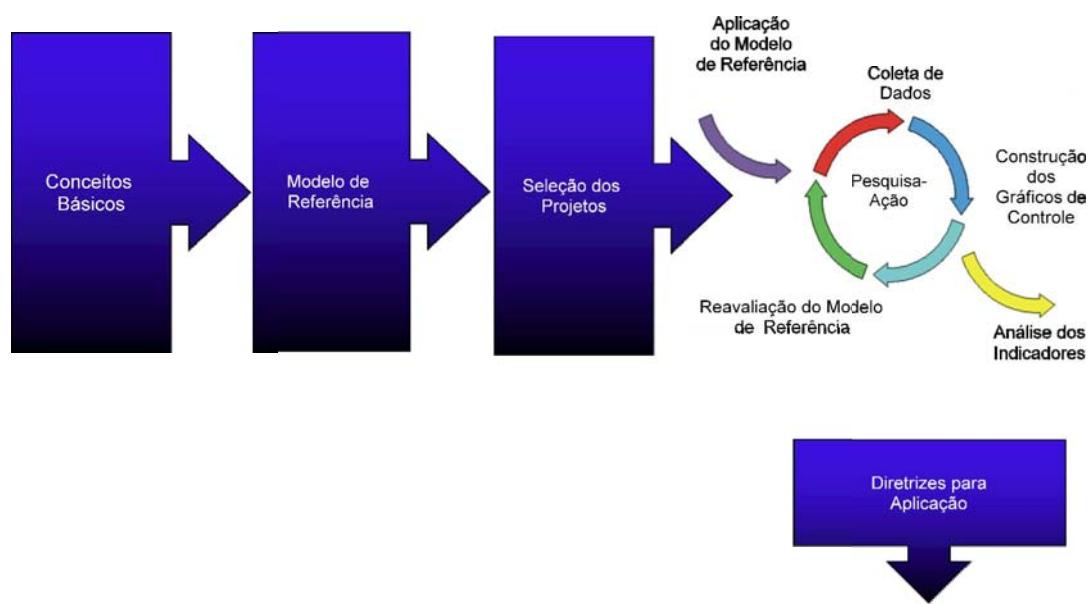
## **Tomada de decisão em fábrica de software baseada em CEP**

### **2. Metodologia de pesquisa**

#### **2.2. Processo de pesquisa**

Os passos para a construção deste trabalho estão ilustrados na figura

1. Os próximos itens detalham cada uma das etapas da pesquisa.



**Figura 1 – Processo para construção da pesquisa.**

## **Tomada de decisão em fábrica de software baseada em CEP**

### **2. Metodologia de pesquisa**

**Conceitos básicos:** na pesquisa bibliográfica foram pesquisados artigos de revistas e de congressos nacionais e internacionais sobre a aplicação do controle estatístico de processo em processos de desenvolvimento de software, utilização de modelos de medição de software baseados no *Practical Software and Systems Measurement* (PSM), conceitos gerais sobre qualidade e fábrica de software. Além dos artigos de revistas e de congressos nacionais e internacionais, foram utilizados livros para a construção da fundamentação teórica necessária para construção a elaboração desta pesquisa.

**Modelo de medição de referência:** tratando-se de uma pesquisa-ação, um modelo de medição de referência, criado e executado pelo autor, é aplicado em projetos-projeto-pilotos para avaliar seu resultado. Esse modelo de medição foi construído com base no processo de desenvolvimento de software utilizado em uma fábrica de software. O PSM é o modelo de medição empregado para selecionar quais indicadores serão usados para avaliar o processo. O Controle estatístico de processo foi aplicado como ferramenta para analisar o desempenho da execução do processo, baseado nas medidas definidas no modelo de medição.

**Seleção dos projetos:** para avaliar o modelo de medição de referência é necessária sua aplicação em um ou mais projetos-projeto-pilotos. Neste trabalho o projeto-piloto tem uma importância fundamental, pois todo o processo de medição foi validado conforme a necessidade de informações dos tomadores de decisão em uma fábrica de software. De acordo com essas necessidades, que foram relatadas pelo gerente da

### **Tomada de decisão em fábrica de software baseada em CEP**

---

#### 2. Metodologia de pesquisa

fábrica desoftware como fundamentais para a gestão efetiva dos projetos, o processo de medição foi construído, aplicado e avaliado.

**Aplicação do modelo de referência:** o autor, como parte integrante do time de desenvolvimento da fábrica de software, executa a coleta dos dados e cria os gráficos de controle necessários para análise da execução do processo de desenvolvimento. Treinamento da equipe de desenvolvimento no modelo de referência e acompanhamento na coleta de dados são algumas atividades executadas nessa etapa.

**Coleta de dados:** o autor utiliza um conjunto de formulários, criados para facilitar a coleta dos dados, para recolher o material necessário para criação dos gráficos de controle.

**Construção dos gráficos de controle:** após a coleta dos dados, os gráficos de controle foram construídos para interpretação dos indicadores do modelo de referência. Para a elaboração dos gráficos foram avaliadas as características dos dados coletados e sua periodicidade.

**Análise dos indicadores:** após toda a coleta e construção dos gráficos de controle os indicadores foram analisados com os gerentes de projetos e a gerência da fábrica de software. As ações para a melhoria do processo foram relacionadas e planos de ações foram criados.

**Reavaliação do modelo de referência:** no segundo ciclo da pesquisa-ação o modelo de medição de referência será alterado, caso seja necessário.

### **Tomada de decisão em fábrica de software baseada em CEP**

---

#### 2. Metodologia de pesquisa

**Diretrizes para aplicação:** após a análise dos indicadores serão apresentadas algumas diretrizes para utilização do controle estatístico de processo para aplicação em desenvolvimento de software.

A pesquisa-ação consistirá dois ciclos onde o primeiro ciclo terá o objetivo de validar o modelo de medição de referência e o segundo o objetivo de efetuar as melhorias necessárias para o modelo de referência.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

#### **3. Fundamentos**

Este capítulo apresenta os conceitos e a fundamentação teórica que foram usados como base para criar o modelo de medição proposto neste trabalho. Conceitos de qualidade, controle estatístico de processo, PSM, fábrica de software, estão apresentados de forma resumida.

Para entender a importância da qualidade para o desenvolvimento de software, foi realizada uma pesquisa na origem da qualidade. Essa fundamentação procura ressaltar a importância da qualidade com ênfase no processo e não na qualidade direcionada ao produto.

Após o entendimento da importância da qualidade direcionada ao processo é necessário selecionar os indicadores que serão utilizados para acompanhar a execução do processo de desenvolvimento de software. Para a construção desses indicadores será utilizado o modelo PSM (*Practical Software and Systems Measurement*).

Os indicadores serão analisados utilizando o CEP (Controle Estatístico de Processo) como ferramenta gráfica para análise do desempenho do processo.

O conceito de fábrica de software é importante para entender o funcionamento da empresa que serviu como projeto-piloto para aplicação do modelo de referência.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

#### **3.1. Conceito de qualidade**

O movimento da qualidade teve início na década de 1920, quando Walter Shewhart desenvolveu o primeiro gráfico de controle de processo, enquanto trabalhava como estatístico na Bell Labs. Durante a Segunda Guerra Mundial, aumentou a necessidade de utilização da estatística para monitorar a qualidade da produção devido, em parte, às grandes quantidades de material bélico produzido e à falta ou deficiência de mão-de-obra dentro do país. Após a guerra deu-se mais ênfase ao volume de produção, ficando a qualidade em segundo plano. Apenas no Japão, com a forte contribuição de Deming e Juran, o foco continuou na qualidade e seu resultado pode ser observado na Prática.

Na década de 1970, as mercadorias japonesas foram consideradas entre as melhores do mundo (DAVIS, 2001). Dobrando-se a evidência da qualidade dos produtos japoneses, os Estados Unidos iniciaram, na década de 1980, uma nova fase produtiva com uma mudança de foco, por parte das empresas, para a qualidade.

Deming e Juran foram, mais uma vez, peças-chave nessa mudança e seus métodos hoje são difundidos em todo o mundo por uma legião de seguidores que os divulgam, aplicam e adaptam à realidade de cada empresa.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

O termo Qualidade é extremamente vasto e abrangente, haja vista a definição de Juran (1993) “Qualidade é adequação ao uso”. Segundo PALADINI (1997), essa definição, tendo como alvo o consumidor, pode ser considerada como um conceito técnico da qualidade, segundo o qual todos os elementos que tiverem alguma participação, direta ou indireta, na produção do bem ou do serviço, serão responsáveis por um produto para atender às necessidades do cliente e até, se possível, superar.

CAMPOS (1992) e WERKEMAN (1995) tornaram mais prático o conceito de Juran, ao definir qualidade como todas aquelas dimensões que afetam a satisfação das necessidades das pessoas e, por conseguinte a sobrevivência da empresa. Essas dimensões foram definidas traduzidas da seguinte forma:

- Qualidade intrínseca: é a qualidade de um produto, expressa por suas características específicas, que promovem a satisfação do cliente interno (etapas intermediárias) ou externo (etapa final), englobando entre outros aspectos a qualidade do produto em si, a qualidade das pessoas, a qualidade do sistema e a qualidade da informação. Produtos defeituosos não seriam apenas inconvenientes para os consumidores, mas também diminuiriam as vendas.
- Custo: é o resultado apurado de todas as atividades ligadas à fabricação do produto, desde o custo final até os custos intermediários tais como compras, vendas, recrutamento, treinamento etc. Deming (1990) defende muito bem esse

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

binômio qualidade x preço, ao afirmar que “Qualidade é atender às necessidades dos clientes a um preço que eles estejam dispostos a pagar”.

- Entrega: no prazo certo, no local certo e na quantidade certa. Uma empresa precisa fabricar produtos na quantidade exigida pelos consumidores e fornecê-los, se possível, antes da data especificada para a entrega.
- Moral: é importante que todos os funcionários se sintam satisfeitos no seu trabalho. O bom ambiente de trabalho e a satisfação na atividade é um componente importante na qualidade final do produto.
- Segurança: essa dimensão se refere não só à segurança dos trabalhadores envolvidos na produção, como também do próprio consumidor.

Ainda dentro de um melhor entendimento do conceito de qualidade, levando em conta o ambiente onde é produzida, vale observar a estruturação da qualidade por PALADINI (1997), que a conceitua segundo três estruturas: qualidade *in-line*, qualidade *on-line* e qualidade *off-line*.

- Qualidade *in-line*: é a qualidade obtida em nível de processo produtivo, podendo ser caracterizada como um conjunto bem definido de elementos básicos voltados para o processo de fabricação. Esse modelo prioriza, fundamentalmente, os esforços para a correção e a prevenção de defeitos, mas

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

também engloba os de redução de custos, de desperdício, de retrabalho, ou seja, busca também a otimização de processo. É um modelo que abrange o conceito de CROSBY (1999) que afirma que a qualidade é a conformidade do produto às suas especificações.

- Qualidade *off-line*: é a qualidade gerada pelas áreas não diretamente ligadas ao processo de fabricação, mas relevantes para adequar o produto ao uso que dele se espera desenvolver. É o caso dos departamentos de compras, vendas, marketing.
- Qualidade *on-line*: é definida como o esforço feito pela empresa para ter sempre um produto adequado ao consumidor. A empresa precisa captar o mais rapidamente possível alterações em preferências, hábitos ou comportamentos de consumo, e repassá-las ao processo produtivo, de forma a adaptar o processo, no menor espaço de tempo, à nova realidade de mercado. Esse modelo acompanha o produto desde o seu nascimento, na forma da qualidade do projeto, e durante todo o seu ciclo de vida, ajustando permanentemente suas características às exigências do mercado.

Segundo GARVIN (2002), as empresas devem se preocupar com oito dimensões para definir suas metas e seus programas de qualidade. São elas:

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

- Desempenho: relativa à aplicação principal do produto.
- Complementos (características secundárias): relativos a aspectos que complementam a aplicação do produto.
- Confiabilidade: referente à não ocorrência de falhas em determinado intervalo de tempo.
- Conformação: o produto deve atender às especificações ou padrões preestabelecidos.
- Durabilidade: referente à vida útil do produto.
- Assistência Técnica: envolve conceitos como tempo, precisão, atendimento etc., relacionados à manutenção do produto após uma falha.
- Estética: relativa às sensações agradáveis, aos sentimentos que o produto deve proporcionar.
- Qualidade Percebida: o consumidor não consegue avaliar todos os aspectos da qualidade do produto, ou serviço e infere outros aspectos.

Segundo GARVIN (2002), as oito dimensões juntas abrangem uma vasta gama de conceitos, devendo-se considerar que algumas dimensões envolvem atributos mensuráveis dos produtos; outras refletem preferências pessoais. Algumas dimensões são objetivas e intemporais; outras variam de acordo com a moda. As dimensões podem ser características intrínsecas dos produtos, ao passo que outras são características atribuídas.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

GARVIN (2002) ressalta que cada uma das principais abordagens concentra-se “implicitamente numa dimensão diferente da qualidade”.

### **3.2. Medição Prática de Software e Sistemas (*Practical Software and Systems Measurement*)**

O PSM é uma iniciativa que foi criada no departamento de defesa dos Estados Unidos para melhorar o gerenciamento de aquisições de grandes projetos. Quase que imediatamente, a adoção do PSM fez com que os fornecedores, melhorassem os seus processos. O PSM, inicialmente, focou o seu desenvolvimento em um guia de medições para o gerente de projetos (CARD, 2003b).

Segundo MURDOCH *et al.* (2003), o PSM foi criado para aumentar a visibilidade dos gerentes de projetos e dos gerentes de subcontratações sobre as atividades de desenvolvimento de software. O PSM tem sua origem baseada no GQM (*Goal Question Metrics*), que é uma abordagem de Basili e Weiss, de 1984. Ele é baseado em um processo que incentiva:

- A identificação das necessidades de informação.
- A interpretação das necessidades de informação como se estivesse dentro de uma categoria de informação.
- A Identificação de conceitos mensuráveis dentro de cada categoria de informação.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

- A identificação de uma possível medição associada a cada conceito mensurável.

A possível medição é utilizada como guia para implementar a medição atual em termos de um ou mais atributos do produto de trabalho atual ou outras entidades existentes, ou introduzidas dentro do projeto. Por exemplo, a possível medição de linha de código pode ser implementada com uma saída particular de um analisador de código, ou um campo específico de um projeto de banco de dados (MURDOCH *et al.*, 2003).

Segundo MURDOCH *et al.* (2003), o PSM recomenda que um processo de medição em geral contenha duas atividades chaves: planejamento e execução da medição, e duas atividades de suporte: avaliar a medição, e estabilizar e manter o comprometimento.

A abordagem PSM consiste em três processos básicos: customizar a medição, aplicar a medição e implementar o processo. Os primeiros dois processos, customizar a medição e aplicá-la, são fundamentais para a medição dos projetos (STATZ, 1999).

Segundo MCGARRY *et al.* (2001), as medidas de software ajudam os gerentes de projetos a entender melhor o trabalho. O PSM direciona o desenvolvimento da medição da informação em um projeto de forma estruturada empregando o modelo de medição da informação. Esse modelo, procura descrever as atividades e as tarefas de medição.

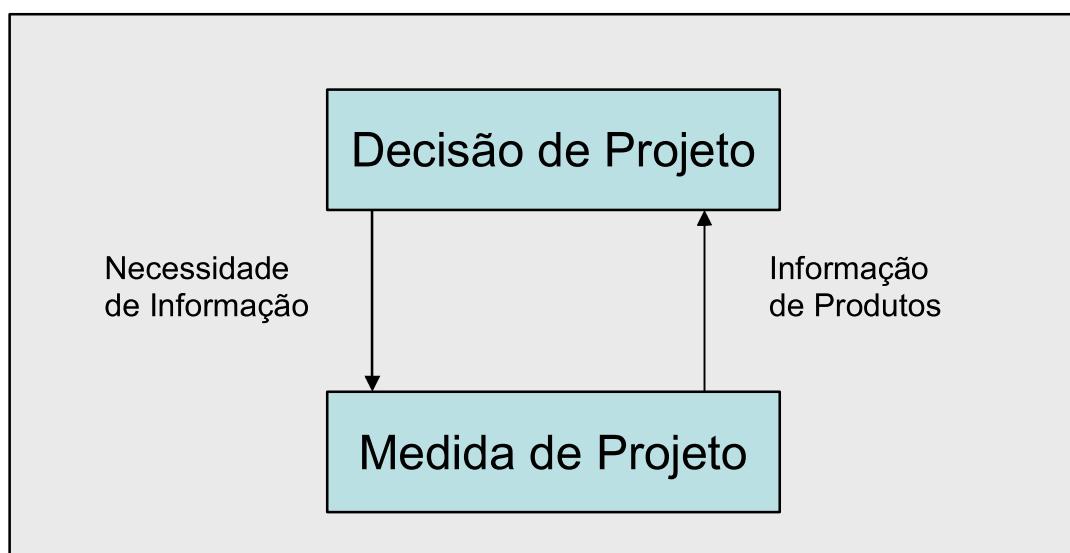
## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

#### **3.2.1. Modelo de medida da informação**

Um dos objetivos do esforço na padronização é a definição de uma terminologia básica dos conceitos de medição. O resultado dessa estrutura define o relacionamento entre as necessidades de informação, os diversos tipos de medição (medida básica, indicadores, medida derivada) e as entidades mensuráveis. Esse modelo de medida da informação estabelece um produto genérico de medição. Classifica ainda sete categorias de necessidades de informação (CARD, 2003b).

O modelo de medida da informação é o mecanismo que vincula as necessidades de informação definidas para um projeto ou produto de software, às entidades que podem ser medidas. O modelo de medida da informação suporta diretamente o planejamento da medida e as atividades de análise (MCGARRY *et al.*, 2001).



**Figura 2 – Relacionamento do modelo de medida da informação** (Fonte: MCGARRY *et al.*, 2001).

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

A figura 2 mostra o relacionamento entre as decisões de um projeto e suas medidas. O modelo de medida da informação ajuda a definir a necessidade de informação dos tomadores de decisão focando nas atividades de planejamento para estas medidas, na seleção e especificação das medidas mais apropriadas para as necessidades (MCGARRY *et al.*, 2001). Segundo MCGARRY *et al.* (2001), as informações dos produtos integram com os resultados das medidas, com os critérios de decisão estabelecidos e com as recomendações dos tomadores de decisão ou ações alternativas.

Segundo MCGARRY *et al.* (2001), um projeto de software típico, apresenta muitas necessidades de informação ao longo do tempo. O modelo de medida da informação provê um mecanismo formal para relacionar as definições das necessidades de informação aos processos de engenharia de software e aos produtos que podem ser medidos. O desenvolvimento do plano de medição para um projeto particular requer uma instância desse modelo de medição da informação que especifique as necessidades de informação. O modelo de medida da informação estabelece e define a estrutura para relacionar diferentes conceitos e termos da medição. Definir e implementar o programa de medida envolvem a construção de muitas escolhas em relação à coleta, organização e análise dos dados, requisições de sucesso e abordagem sistemática considerando e descrevendo essas escolhas. Sem uma comunicação concisa e uma terminologia consistente, a comunicação e a análise dos dados ficam comprometidas.

## **Tomada de decisão em fábrica de software baseada em CEP**

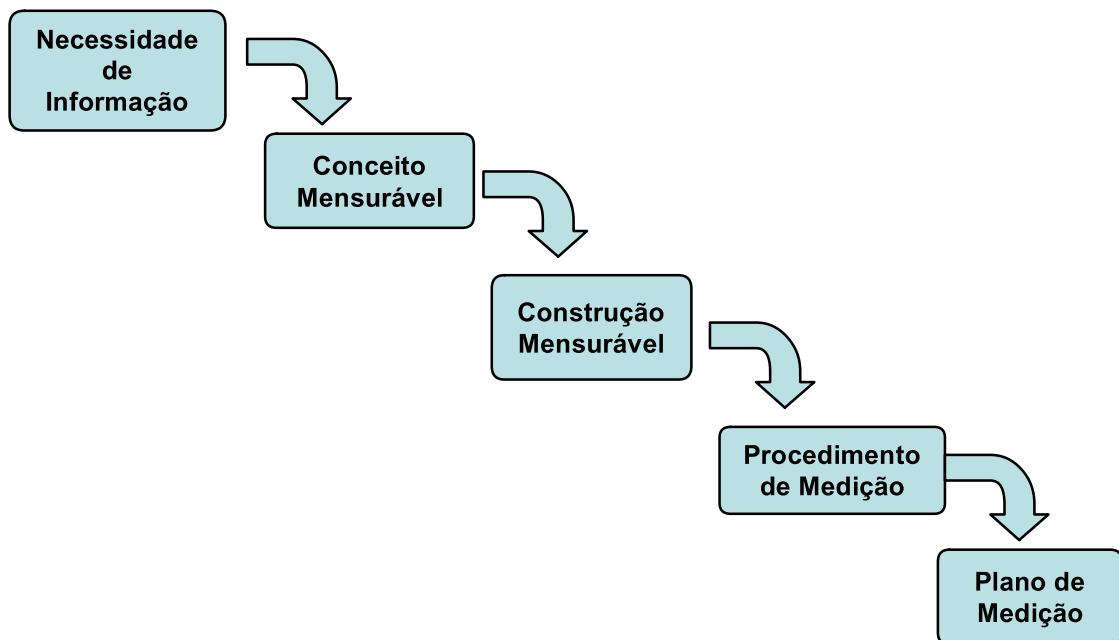
### **3. Fundamentos**

O planejamento da medição é iniciado com o reconhecimento dos gerentes ou engenheiros que especificam as necessidades de informação requeridas para dar suporte aos tomadores de decisão. Os dados ajudam a satisfazer a definição das necessidades de informação, que podem ser obtidas de diferentes elementos do processo de software diferentes e das características de produtos, chamados de entidades de software. Os conceitos mensuráveis formam uma idéia sobre as entidades e como elas deverão ser medidas para satisfazer as necessidades de informação. Eventualmente, os conceitos mensuráveis podem ser formalizados por meio de construções mensuráveis que especificam exatamente o que é aquela medida e como os dados foram combinados para produzir aquele resultado afim de satisfazer a necessidade de informação. O procedimento de medição define o mecanismo de coleta e a organização dos dados necessários para instanciar a construção mensurável (MCGARRY *et al.*, 2001).

Segundo MCGARRY *et al.* (2001), os planos de medição podem ser construídos para atender a uma única necessidade de informação. Na maioria dos casos um plano de medição endereça várias necessidades de informação. A figura 3 ilustra a evolução desse processo dentro de um plano de medição.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**



**Figura 3 – Evolução da necessidade de informação dentro do plano de medição** (Fonte: MCGARRY *et al.*, 2001).

### **3.2.2. Modelo de medição de processo**

Segundo CARD (2003a), diversas atividades têm de ser realizadas para se obter os resultados significativos. O modelo de medição define quatro processos básicos:

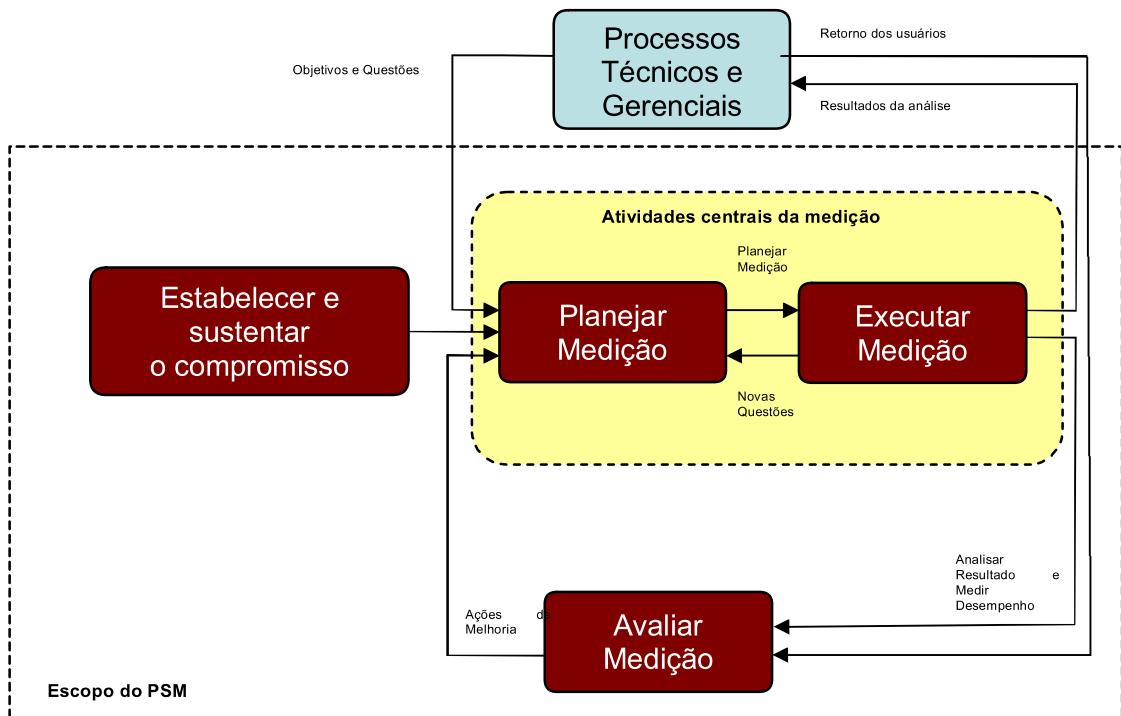
- Planejar a medição: é a atividade do modelo de medição de processo que entende as necessidades de informação do projeto e define as medidas apropriadas para atendê-las. Projeta o produto da medição (resultado).
- Executar a medição: é a atividade que concentra a captura dos dados que irão produzir os resultados exigidos para tomada a decisão.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

- Avaliar a medição: é a atividade do modelo de medição de processo que se concentra na avaliação da medição, e na avaliação do processo de medição.
- Estabelecer e sustentar o compromisso: é a atividade que propõe estabelecer e sustentar o compromisso com o programa de medição e com a empresa.

A figura 4 ilustra o relacionamento entre as atividades do modelo de medição de processo.



**Figura 4 – Modelo de Processo de Medição** (Fonte: MCGARRY *et al.*, 2001).

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

A quinta atividade, processo técnico e gerencial, também pertence ao modelo de processo de medição. Embora, tecnicamente, não tenha atividades específicas de medição, a parte técnica e gerencial tem uma interface direta com o modelo de medição de processo (MCGARRY *et al.*, 2001).

### **3.3. Controle estatístico de processo**

O início formal do controle estatístico de processo deu-se por volta de 1924, quando Walter A. Shewhart desenvolveu e aplicou os gráficos de controle nos Bell Telephone Laboratories. Como era um dispositivo para uso em “chão de fábrica”, por operários com pouca instrução, os gráficos de controle tinham de ser extremamente simples de construir e utilizar. Essa simplicidade motivou seu uso indiscriminado em uma infinidade de aplicações, infelizmente muitas vezes de forma ingênuas; ou seja, eles serviram apenas para decorar paredes. Sem bom conhecimento estatístico para basear os gráficos de controle, sem um estudo detalhado do processo e, por vezes, sem uma intervenção profunda nele, não é possível transformar o gráfico de controle em uma ferramenta efetiva para o monitoramento de processos (COSTA *et al.*, 2004).

Segundo COSTA *et al.* (2004), a eficácia de um gráfico de controle é medida pela rapidez com que esse dispositivo detecta alterações no processo. A análise da relação entre o custo de operação e a eficácia do gráfico deve nortear a escolha de seus parâmetros de implementação: o tamanho das amostras, o intervalo de tempo entre amostragens e o fator

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

que estabelece o posicionamento dos limites de controle no gráfico. Desse modo, se a intenção é monitorar um processo por gráficos de controle, devem-se analisar criteriosamente quais valores atribuir a esses três parâmetros.

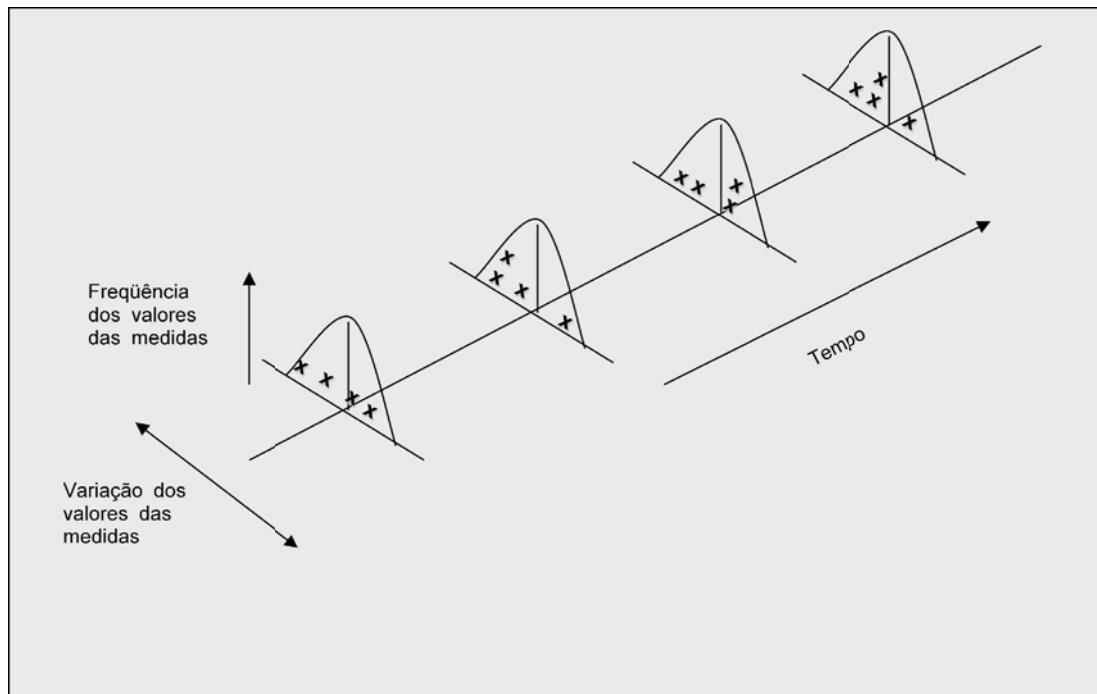
Quando estamos em uma linha de produção, temos a sensação de que todas as unidades produzidas são exatamente iguais. Contudo, olhando com um pouco mais de cuidado, descobrimos que não são tão iguais assim. A expressão variabilidade do processo tem a ver com as diferenças existentes entre as unidades produzidas. Se a variabilidade do processo for grande, as diferenças entre as unidades produzidas serão fáceis de observar; ao contrário, se a variabilidade do processo for pequena, tais diferenças serão difíceis de observar (COSTA *et al.*, 2004).

Desde o início da Revolução Industrial, Shewhart preocupou-se em estudar a variabilidade dos processos. Suas explicações sobre a impossibilidade de se produzir itens exatamente iguais são aceitas até hoje. Segundo ele, todo e qualquer processo, por mais bem projetado e controlado que seja, possui em sua variabilidade um componente impossível de ser eliminado. Trata-se da variabilidade natural do processo, que é fruto de uma série de pequenas perturbações, ou causas aleatórias, contra as quais pouco ou nada se pode fazer. A figura 5 ilustra um processo sujeito apenas a causas aleatórias. A característica X apresenta uma variabilidade que pode ser representada por uma distribuição de probabilidades (usualmente normal ou bem aproximada para uma distribuição normal) que se mantém estável ao longo do tempo. Quando o processo apresenta

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

apenas a variabilidade natural, em razão de causas aleatórias, diz-se que ele está no estado de controle estatístico, ou simplesmente em controle (COSTA *et al.*, 2004).



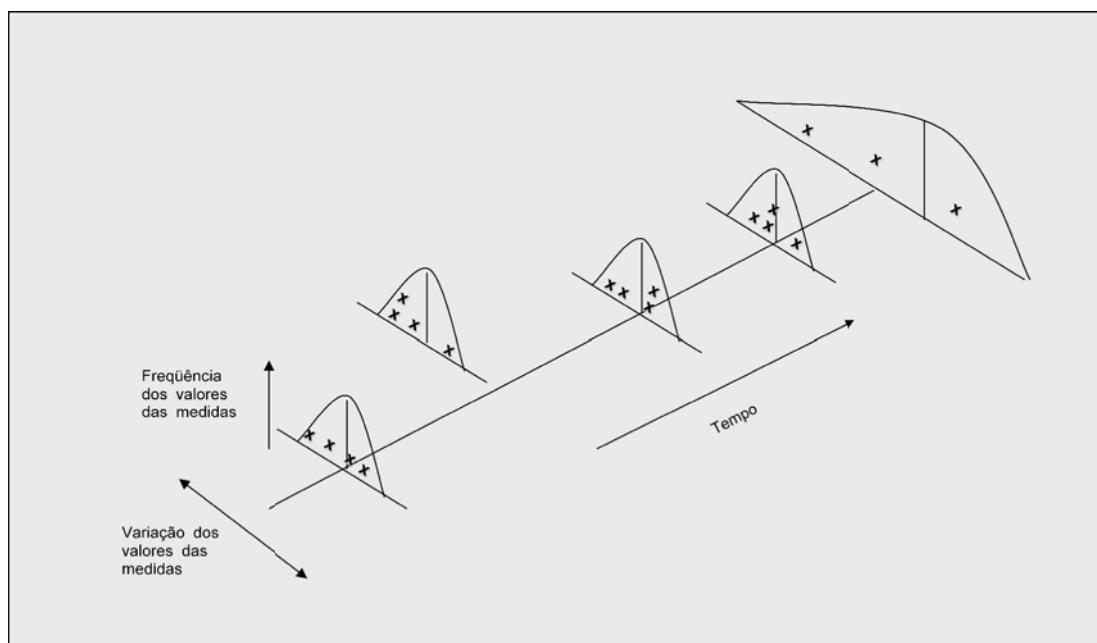
**Figura 5 – Um processo estatisticamente controlado** (Fonte: FLORAC *et al.*, 1999).

Nenhum processo, porém, deixa de estar sujeito também à ocorrência ocasional de perturbações maiores, chamadas de causas especiais, que têm o efeito de deslocar a distribuição da variável aleatória X e/ou aumentar sua dispersão. Uma causa especial é um problema ou modo de operação anormal do processo, que pode, portanto, ser corrigido ou eliminado: um ajuste incorreto ou um desajuste de uma máquina, o rompimento de um tubo e um lote de matéria-prima com defeito são exemplos de causas especiais.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

A figura 6 ilustra o caso em que a causa especial desloca a distribuição de X. As causas especiais são sempre possíveis de eliminar; algumas, contudo, demandam correções significativas no processo. Quando além das causas aleatórias de variabilidade, causas especiais estiverem presentes, diz-se que o processo está fora de controle (COSTA *et al.*, 2004).



**Figura 6 – Um processo fora de controle estatístico** (Fonte: FLORAC *et al.*, 2001).

Segundo COSTA *et al.* (2004), os processos devem ser permanentemente monitorados, para detectar a presença de causas especiais (que aumentam sua dispersão e/ou tiram sua média do valor-alvo). Detectada essa presença, deve-se proceder a uma investigação para identificar a(s) causa(s) especial(is) e investir em sua eliminação. A principal ferramenta utilizada para monitorar os processos e sinalizar a presença de causas especiais são os gráficos de controle.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

Segundo RAMOS (2002, p.295), o CEP (Controle estatístico de processo) prega o controle conduzido simultaneamente à produção (controle do processo) ou execução do serviço, em vez da pura inspeção posterior, quando o produto ou serviço já se encontram terminados (controle do produto). Quando do surgimento de problemas, a ação deve ser no processo (causa) que gerou o defeito, e não no produto (efeito) em si. Não se melhora a qualidade por meio da inspeção. Ela já vem com o produto quando este deixa o processo e, portanto, antes de inspecioná-lo.

A eficiência de desempenho de um processo em termos de qualidade e produtividade depende de dois fatores: a forma pela qual ele foi projetado e como é operado (RAMOS, 2002, p. 295).

### **3.3.1. Gráficos de Controle**

Segundo RAMOS (2002, p.298), existem duas categorias de gráficos de controle:

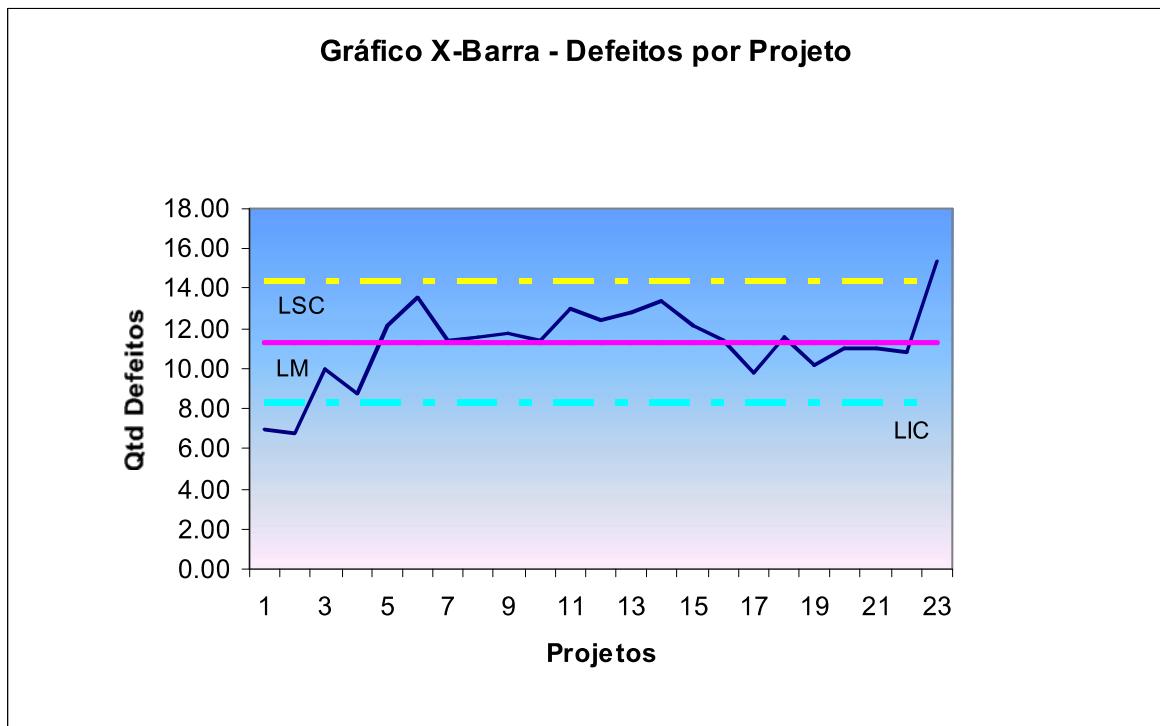
- Variáveis: consistem nas características cujo valor é o resultado de algum tipo de medição (peso, tempo, valor, comprimento etc.).
- Atributos: são as características cujo resultado é decorrente de uma classificação ou contagem (números de defeituosos, número de defeitos, número de erros etc.).

Um gráfico de controle é um conjunto de pontos (amostras), ordenados no tempo, que são interpretados em função de linhas horizontais, chamadas de limite superior de controle (LSC), linha média (LM), e limite

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **3. Fundamentos**

inferior de controle (LIC) (RAMOS, 2002, p.299). A figura 7 apresenta um gráfico de controle típico.



**Figura 7 – Gráfico de controle** (Fonte: SILVA et al., 2006).

Segundo RAMOS (2002, p.299), a teoria estatística desenvolvida por Walter A. Shewhart para o cálculo dos limites de controle baseia-se na idéia de que, sendo o processo estudado estável, então uma estatística qualquer, calculada com base nos dados fornecidos pelas amostras, terá uma probabilidade próxima a 1 de estar no intervalo de mais ou menos três desvios padrões a partir da média da população. Quando um valor cair fora desse intervalo, assume-se que a hipótese de estabilidade do processo já não é valida, indicando a presença de uma causa especial de variação.

Segundo RAMOS (2002, p.299), no cálculo dos limites de controle e obtenção de amostras, as seguintes regras devem ser obedecidas:

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

- O desvio padrão utilizado deve ser sempre estimado com base na variação dentro da amostra, ou seja, na média das dispersões das amostras, e não na disposição total dos dados.
- Os gráficos sempre utilizam limites de controle localizados à distância de três desvios padrões da linha média.
- Os dados devem ser obtidos e organizados em amostras (ou subgrupos) segundo algum critério racional, visando permitir a obtenção das respostas necessárias.
- O conhecimento obtido por meio dos gráficos de controle deve ser empregado para modificar as ações, conforme adequado.

Shewhart usa a expressão  $\bar{X} \pm t\sigma_x$  para caracterizar a tolerância dos limites de variabilidade. Ele argumenta que, na prática, o método para estabilizar os limites de tolerância para os dados observados depende da teoria estatística que permita estimar a média  $\bar{X}$  e o desvio padrão  $\sigma_x$  da distribuição, porém isso requer uma experiência empírica para justificar a escolha de  $t$ . Os resultados das experiências realizadas por Shewarts indicam que  $t = 3$  provê valores econômicos aceitáveis. Nessa experiência, 3-sigmas para os limites de tolerância, quando usado para controle, provê um critério efetivo de diferenciação entre as causas comuns e as causas especiais, sem levar em consideração a base da distribuição (FLORAC *et al*, 1999).

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

#### **3.3.1.1. Gráficos de controle para variáveis**

##### **3.3.1.1.1. Gráficos de controle de $\bar{X}$ e R**

Se uma variável a ser controlada é uma variável contínua, o usual é monitorar o processo por um par de gráficos de controle: um para monitorar a centralidade e outro para monitorar a dispersão da variável. Na maioria das vezes, os gráficos empregados são o da média amostral  $\bar{X}$ , para monitorar a centralidade, e o da amplitude amostral R, para monitorar a dispersão (COSTA *et al*, 2004).

As fórmulas para cálculo dos limites de controle estão descritas abaixo:

$$LSC_{\bar{x}} = \bar{\bar{x}} + A_2 \cdot \bar{R}$$

$$LM_{\bar{x}} = \bar{\bar{x}}$$

$$LIC_{\bar{x}} = \bar{\bar{x}} - A_2 \cdot \bar{R}$$

onde:

$$A_2 = \frac{3}{d_2 \cdot \sqrt{n}}$$

Dois termos aparecem nessas expressões que merecem maiores esclarecimentos:  $n$  e  $d_2$ . O primeiro é o tamanho da amostra e decorre do fato de a dispersão (desvio padrão) das médias das amostras ser menor que a dos valores individuais, e o segundo é necessário para corrigir um vício introduzido quando se substitui  $\sigma(x)$  por  $\bar{R}$ . O valor de  $A_2$  e  $d_2$  são função da amostra ( $n$ ) (RAMOS, 2002, p.301).

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

Para o caso da amplitude, seus limites de controle ficam estabelecidos de acordo com:

$$LSC_R = D_4 \cdot \bar{R}$$

$$LM_R = \bar{R}$$

$$LIC_R = D_3 \cdot \bar{R}$$

onde:

$$D_4 = 1 + 3 \cdot \frac{d_3}{d_2}$$

e

$$D_3 = 1 - 3 \cdot \frac{d_3}{d_2}$$

Segundo RAMOS (2002, p. 301), para os valores menores que 7, não existe o fator  $D_3$ .

### **3.3.1.1.2. Gráficos da média e desvio padrão (X-Barra e S)**

Esses gráficos são similares aos gráficos  $\bar{X}$  e  $R$ . Embora o cálculo do desvio padrão da amostra ( $s$ ) seja mais difícil do que da amplitude ( $R$ ), o fato é que, quando são adotadas amostras de tamanhos maiores ( $n > 10$ ), não mais se deve empregar  $R$  para avaliar a variabilidade do processo, pois ele se torna muito ineficiente quando comparado com  $s$  (RAMOS, 2002, p. 306).

Segundo RAMOS (2002, p. 306), as fórmulas para cálculo para os limites de controle estão descritas abaixo:

$$LSC_{\bar{x}} = \bar{x} + 3 \cdot \frac{\bar{s}}{c_4 \cdot \sqrt{n}}$$

$$LM_x = \bar{x}$$

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

$$LSC_{\bar{x}} = \bar{x} - 3 \cdot \frac{\bar{s}}{c_4 \cdot \sqrt{n}}$$

O termo  $c_4$  é um fator necessário para corrigir um vício introduzido quando se substitui  $\sigma(x)$  por  $\bar{s}$ , similar ao fator  $d_2$ , que é empregado para  $\bar{R}$ . Assim as fórmulas podem ser reescritas de acordo com (RAMOS, 2002, p.306):

$$LSC_{\bar{x}} = \bar{x} + A_3 \cdot \bar{s}$$

$$LM_{\bar{x}} = \bar{x}$$

$$LSC_{\bar{x}} = \bar{x} - A_3 \cdot \bar{s}$$

onde:

$$A_3 = \frac{3}{c_4 \cdot \sqrt{n}}$$

Os valores de  $A_3$  e  $c_4$  são função do tamanho da amostra (n).

Conforme RAMOS (2002, p. 307), para o desvio padrão, seus limites de controle ficam de acordo com:

$$LSC_s = B_4 \cdot \bar{s}$$

$$LM_s = \bar{s}$$

$$LSC_s = B_3 \cdot \bar{s}$$

onde:

$$B_4 = 1 + 3 \cdot \frac{c_5}{c_4}$$

e

$$B_3 = 1 - 3 \cdot \frac{c_5}{c_4}$$

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

Para  $n < 6$ , não existe o fator  $B_3$  (RAMOS, 2002, p. 307).

#### **3.3.1.1.3. Gráficos do valor individual e amplitude móvel ( $X$ e $Rm$ )**

Quando somente valores individuais estiverem disponíveis, torna-se necessário o emprego desses tipos de gráficos. A amplitude móvel ( $Rm$ ) é definida como a diferença (em módulo) entre  $m$  valores individuais consecutivos. Os limites desses gráficos são calculados de acordo com (RAMOS, 2002, p.309):

$$LSC_{\bar{x}} = \bar{x} + E_2 \cdot \bar{R}m$$

$$LM_{\bar{x}} = \bar{x}$$

$$LSC_{\bar{x}} = \bar{x} - E_2 \cdot \bar{R}m$$

onde:

$$E_2 = \frac{3}{d_2}$$

e

$$LSC_{Rm} = D_4 \cdot \bar{R}m$$

$$LM_{Rm} = \bar{R}m$$

$$LSC_{Rm} = D_3 \cdot \bar{R}m$$

Segundo esse autor (2002, p. 314), quando se escolhe um gráfico de controle para variáveis, é importante sempre ter em mente que sua escolha depende do tamanho da amostra ( $n$ ) em que se deseja empregá-lo. Quando o tamanho da amostra é maior do que 1, existem duas opções básicas:

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

média e amplitude ( $\bar{X}$  e  $R$ ) ou média e desvio padrão ( $\bar{X}$  e  $S$ ). Contudo, à medida que  $n$  aumenta, a amplitude vai se tornando cada vez mais ineficiente para estimar a variabilidade do processo e, por isso, quando  $n > 10$ , somente se devem empregar os gráficos da média com o desvio padrão.

Por outro lado, quando o tamanho da amostra é unitário, apenas o gráfico do valor individual com amplitude móvel ( $X$  e  $Rm$ ) pode ser empregado, normalmente, tomando-se as amplitudes móveis dois a dois ou três a três (RAMOS, 2002, p.314).

### **3.3.1.2. Gráficos de controle para atributos**

#### **3.3.1.2.1. Gráficos da fração defeituosa (p)**

A fração defeituosa da amostra é definida como a razão entre o número de defeituosos encontrados na amostra ( $d$ ) e o tamanho da amostra ( $n$ ). A distribuição de probabilidade da fração defeituosa é a binomial. Entretanto, quando os tamanhos de amostras forem suficientemente grandes para atenderem às restrições (RAMOS, 2002, p.316):

$$n \cdot \bar{p} > 5 \text{ e } n \cdot (1 - \bar{p}) > 5$$

com

$$\bar{p} = \frac{\sum d_i}{\sum n_i}$$

então no lugar da distribuição binomial pode-se utilizar a distribuição normal (aproximação da binomial pela normal). Nesse caso, os limites de controle estabelecidos em  $\pm 3 \cdot \sigma$  continuam válidos e, para a fração defeituosa, ficam:

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

$$\mu(p) \pm 3 \cdot \sigma(p)$$

Como  $\mu(p)$  e  $\sigma(p)$  não são conhecidos, serão estimados com base nos dados das amostras, passando a ser (RAMOS, 2002, p. 317):

$$LSC_p = \bar{p} + 3 \cdot \sqrt{\frac{\bar{p} \cdot (1 - \bar{p})}{n}}$$

$$LM_p = \bar{p}$$

$$LIC_p = \bar{p} - 3 \cdot \sqrt{\frac{\bar{p} \cdot (1 - \bar{p})}{n}}$$

#### **3.3.1.2.2. Gráfico do número de defeituosos na amostra ( $np$ )**

Segundo RAMOS (2002, p. 320), a diferença desse gráfico em relação ao anterior, é a diferença que se deseja marcar o número de defeituosos encontrados na amostra. Seus limites de controle são:

$$LSC_{np} = n \cdot \bar{p} + 3 \cdot \sqrt{n \cdot \bar{p} \cdot (1 - \bar{p})}$$

$$LM_{np} = n \cdot \bar{p}$$

$$LSC_{np} = n \cdot \bar{p} - 3 \cdot \sqrt{n \cdot \bar{p} \cdot (1 - \bar{p})}$$

#### **3.3.1.2.3. Gráfico do número de defeitos na amostra (c)**

Segundo RAMOS (2002, p. 320), a distribuição de probabilidade do número de defeitos na amostra é a de Poisson. Entretanto, quando:

$$\bar{c} > 5$$

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

com

$$\bar{c} = \frac{\sum c_i}{k}$$

onde  $k$  = quantidade total de amostras então, no lugar da distribuição de Poisson podemos utilizar a distribuição normal. Nesse caso, os limites de controle ficam:

$$\mu(c) \pm 3 \cdot \sigma(c)$$

como  $\mu(c)$  e  $\sigma(c)$  são desconhecidos, resulta

$$LSC_c = \bar{c} + 3 \cdot \sqrt{\bar{c}}$$

$$LM_c = \bar{c}$$

$$LIC_c = \bar{c} - 3 \cdot \sqrt{\bar{c}}$$

#### **3.3.1.2.4. Gráfico do número de defeitos por unidade de inspeção ( $u$ )**

Conforme RAMOS (2002, p. 323), o número de defeitos por unidade de inspeção ( $u$ ) é definido como a razão entre o número de defeitos na amostra ( $c$ ) e o tamanho da unidade de inspeção ( $n$ ), ou seja:

$$u = \frac{c}{n}$$

Por unidade de inspeção entende-se certa quantidade de itens, comprimento, volume, tempo etc. tomada como adequada para a finalidade de inspeção. É fácil perceber que:

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

$$\bar{u} = \frac{\bar{c}}{n}$$

Os limites de controle, nesse caso, ficam:

$$LSC_u = \bar{u} + 3 \cdot \sqrt{\frac{\bar{u}}{n}}$$

$$LM_u = \bar{u}$$

$$LIC_u = \bar{u} - 3 \cdot \sqrt{\frac{\bar{u}}{n}}$$

Quando se trabalha com atributos, é necessário garantir que as amostras tenham tamanhos mínimos para que haja a possibilidade de os problemas aparecerem. Em outras palavras, amostras muito pequenas fazem com que os gráficos de controle se tornem totalmente ineficazes (RAMOS, 2002, p.324).

Para que os gráficos  $p$  ou  $np$  sejam eficazes, deve-se ter:  $n \cdot \bar{p} > 5$  e  $n \cdot (1 - \bar{p}) > 5$ . Isso garantirá que as amostras possibilitem que os itens defeituosos apareçam, evitando-se o problema anteriormente citado. Da mesma forma, para os gráficos  $c$  ou  $u$  serem eficazes, deve-se ter  $\bar{c} > 5$ . Esse critério possibilita o surgimento de defeitos suficientes na amostra, de modo que o gráfico possa apontá-los (RAMOS, 2002, p. 325).

Segundo esse autor (2002, p. 325), assim como para as variáveis, também é muito importante selecionar o tipo de gráfico de controle adequado para os atributos. Essa escolha é feita em função de dois itens: a

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

categoria de gráfico (classificação ou contagem) e o tamanho da amostra (fixo ou variável).

#### **3.3.2. Estabilidade do processo**

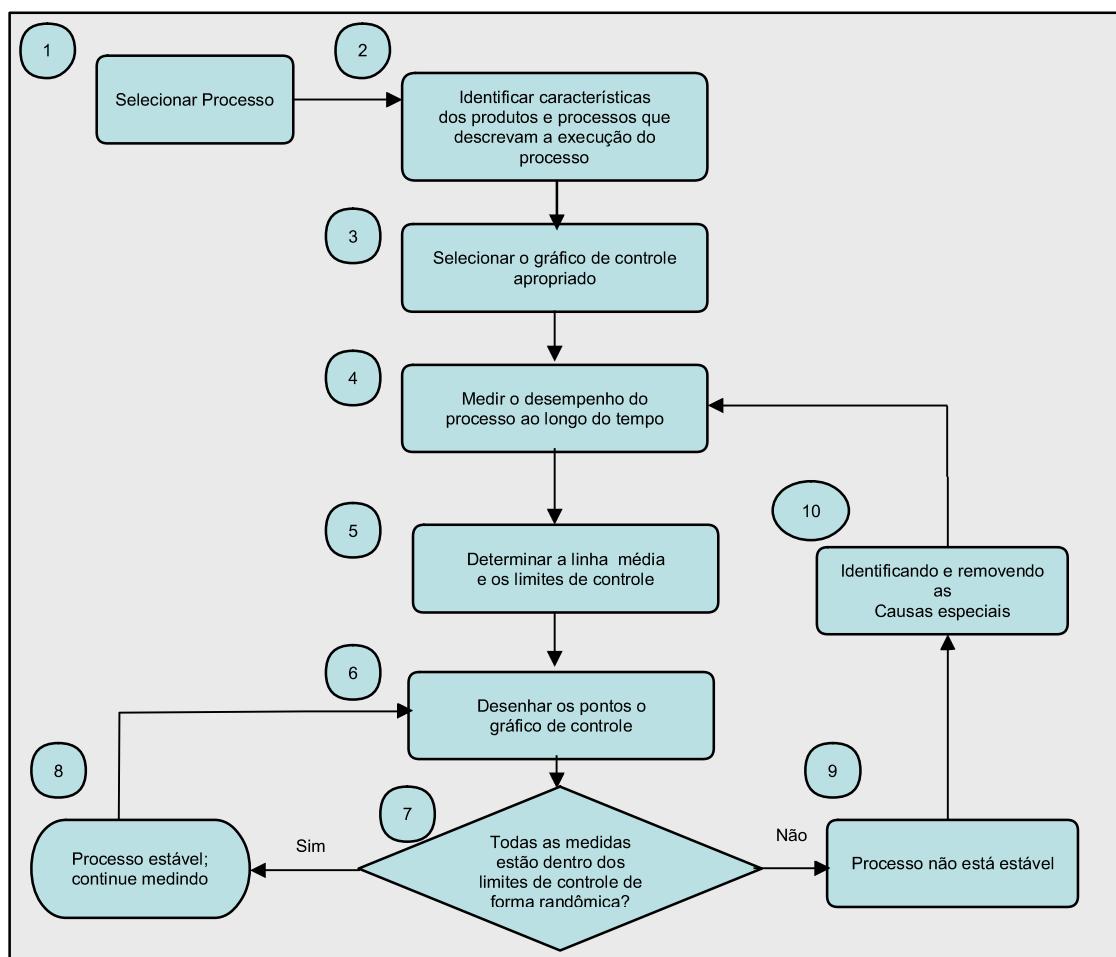
A figura 8 mostra os passos necessários para a investigação da estabilidade do processo. Quando os gráficos de controle não apresentam causas especiais, o processo está estatisticamente controlado, ou estável. A média e os limites de variações são previsíveis com um alto grau de certeza para um futuro imediato. Um processo estável é o centro da habilidade de uma organização produzir os seus produtos de acordo com o seu planejamento. Se o centro do processo for melhorado sua produção poderá produzir melhor os produtos que serão mais competitivos (FLORAC, 1999).

Em um processo estável, as variações encontradas são apenas causas comuns. Todas as variações do processo estável são causadas pela herança de fatores que são partes do próprio processo. Variações em razão de causas especiais, como erros causados pelo operador, mudanças ambientais, desvios do processo e mudanças das características em matéria-prima e recursos, têm sido removidas do processo e prevenidas ou incorporadas como parte permanente do processo (FLORAC, 1999).

Entretanto, buscar reduzir as variações em razão de causas comuns mediante análises mais profundas do processo, é de grande importância para a melhoria contínua, significando uma contribuição econômica e qualitativamente mais importante que a simples eliminação das causas especiais de variação, conforme ressalta COSTA NETO (2007, p. 92).

## Tomada de decisão em fábrica de software baseada em CEP

### 3. Fundamentos



**Figura 8 – Passos para avaliar a estabilidade do processo** (Fonte: FLORAC, 1999).

## Tomada de decisão em fábrica de software baseada em CEP

### 3. Fundamentos

#### 3.3.3. Seleção do gráfico de controle para atributos

Segundo RAMOS (2002, p.325), a escolha do gráfico de controle para os atributos é feita em função de dois itens: a categoria de gráfico (classificação ou contagem) e o tamanho da amostra (fixo ou variável). Seguindo o roteiro definido na figura 9.

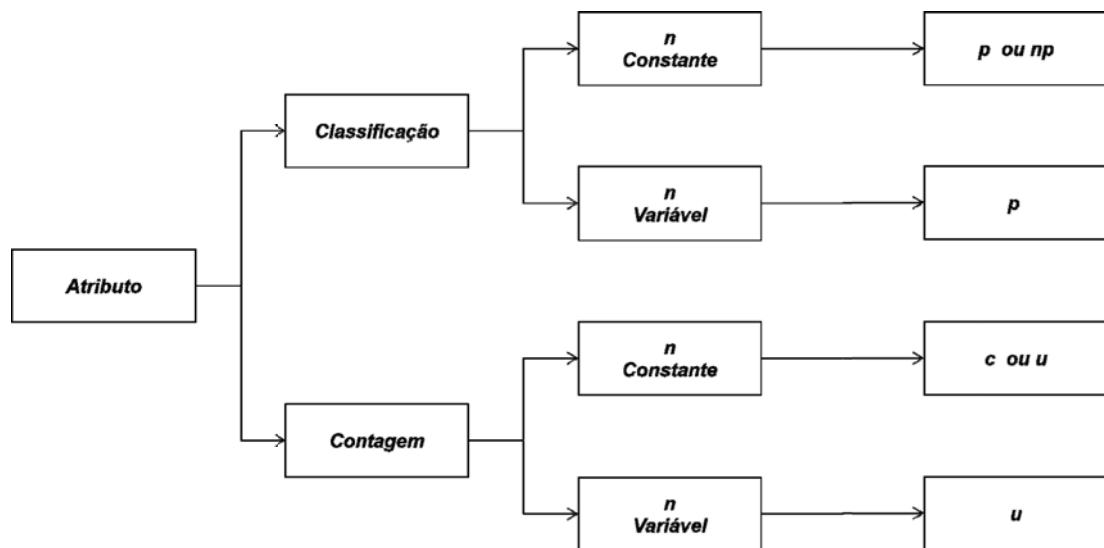


Figura 9 – Fluxograma para seleção de gráficos para atributos (Fonte: RAMOS, 2002, p.325)

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

#### **3.3.4. Aplicação do CEP em processos de software**

Segundo WELLER (2000), métodos quantitativos como o controle estatístico de processo para processos de desenvolvimento de software podem prover um positivo custo benefício. O autor usa o controle estatístico de processo em inspeção de código e emprega os dados de teste para avaliar a qualidade do produto durante a execução dos processos de testes a fim de acompanhar a qualidade do produto após a sua entrega.

Segundo LANTZY (1992), no sentido estatístico, o controle de processo está relacionado à habilidade de trazer características mensuráveis das saídas do processo entre os limites de controle. Ao contrário do processo de manufatura, as entradas e as saídas do processo de software são diferentes para cada instância do processo.

Muitas das aplicações do CEP para software sofrem as influências voltadas à aplicação nos processos de manufatura. Teóricos não consideram diferenças fundamentais entre os processos de software e de manufatura. A analogia mais apropriada é que os processos de software, o processo de projeto e implementação, da operação de manufatura especialmente a execução deles. Apesar da dificuldade, o processo de software é um processo que pode trazer o significado do controle estatístico (LANTZY, 1992).

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

Os fundamentos estatísticos do CEP são relativamente simples e podem ser utilizados por diversos níveis, não somente pelos especialistas em medição, uma vez que o processo de medição esteja estabilizado. E mesmo quando não se está empregando um processo formal de CEP, a base do controle estatístico de processo pode ajudar na análise da efetividade do processo (CARD, 1994).

Para tornar real o sentido da utilização do CEP para processos de software, devem-se selecionar os processos em que há problemas no desenvolvimento de software. É importante identificar quais os mais críticos que ajudariam nos objetivos do projeto ou organização, e que seja atributos do processo ou produto que requer melhoria (GUPTA, 2002).

Segundo KOMURU (2006), o CEP tem sido aplicado no desenvolvimento de software em alguns processos como: revisão em pares, processos de teste e monitoramento e controle. Esse autor (2006) aponta que as atividades do processo de manufatura da indústria estão focadas na produção do produto, enquanto as atividades do desenvolvimento de software estão centradas no processo. Para auxiliar na aplicação do CEP para o desenvolvimento de software, algumas características, precisam ser relacionadas.

Segundo SILVA (2007), o CEP pode ser utilizado como uma ferramenta para identificar necessidades de treinamento para melhoria de escrita dos requisitos de software. O autor apresenta o CEP como uma ferramenta para avaliar e monitorar a maturidade no processo de gerência de requisitos.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

Segundo KOMURU (2006), em geral, o processo de desenvolvimento de software possui as seguintes características:

- Uso intensivo da criatividade humana, o processo de desenvolvimento de software é mais dependente da atividade e da criatividade humana do que a indústria da manufatura.
- Envolve várias causas comuns, relacionadas ao primeiro ponto, existem vários fatores que afetam o desempenho do processo, por exemplo: ferramentas, métodos, tipos de software e ambiente de desenvolvimento, destacados por FLORAC *et al* (1999).
- Dificuldade de obter uma grande quantidade de dados homogêneos, especialmente os que têm valores para o negócio.

Segundo GUPTA (2002), dependendo das características dos dados coletados empregam-se os gráficos por variáveis ou por atributos. Muitas das medições que podem ser realizadas no desenvolvimento de software utilizam os atributos de dados. Para esse grupo de dados é necessário fazer uma análise sobre o tipo de distribuição que o conjunto de dados apresenta.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

#### **3.4. Fábrica de Software**

Em 1991, CUSUMANO (1991) afirmou que o termo Fábrica de Software foi criado nos anos 1960 e 1970, nos Estados Unidos e no Japão, e tinha como foco principal a proposta de desenvolver softwares utilizando preceitos de engenharia associados à manufatura. A partir daí, o mercado passou a usar os termos Indústria de Software ou Manufatura de Software para designar os mesmos preceitos.

Conforme COOKE (2000), não existem empresas de serviço. Existem apenas empresas cujos componentes de serviço estão mais ou menos presentes do que em outras empresas.

Como em uma Fábrica de Software os serviços correspondem a uma parcela preponderante na composição bens/serviços de seu produto final ela será chamada simplesmente de empresa de serviços. Ainda de acordo com COOKE (2000), o sistema em que as empresas que constroem softwares se inserem é chamado de sistema de operação.

Segundo WILD (1977):

1. Um sistema de operação é uma configuração de recursos combinada para a provisão de bens ou serviços.
2. Os objetivos de um sistema de operação são a satisfação do consumidor e a produtividade dos recursos empregados.
3. As diferentes estruturas dos sistemas de manufatura, transporte,

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

suprimento e serviços indicam semelhanças entre manufatura e suprimento e entre transporte e serviço.

COOKE (2000) afirma que, uma empresa de desenvolvimento de software possui seu principal valor agregado híbrido (ênfase tanto na presença do consumidor, *front-office*, como nas operações de suporte, *back-office*, cuja ênfase no contato ao consumidor está em bens facilitadores (equipamentos e instalações)).

Em 1991, SWANSON *et al.* já afirmavam:

“A companhia multinacional e de origem americana Celite teve sucesso com a abordagem denominada Fábrica de Software, que não se fixa simplesmente em exercícios de geração de código, mas em um conjunto de técnicas integradas. Essas técnicas geram um aumento significativo tanto na produtividade como na qualidade dos códigos liberados. As estimativas da Celite mostram que a maioria das funções que levavam de 20 a 30 dias para ficarem prontas, agora demoram de 1 a 2 dias”

Segundo FERNANDES (2004), o objetivo de uma fábrica de software deve ser a geração de produtos requeridos pelos usuários ou clientes, com o mínimo de defeitos possível e a um preço (custo) competitivo e compatível, que forneça a margem necessária para os investimentos em manutenção e melhoria da fábrica.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

Ainda segundo FERNANDES (2004), uma fábrica de software deve possuir os seguintes atributos básicos (independente de seu escopo de fornecimento):

- Estabelecer processo definido e padrão para o desenvolvimento de produtos de software.
- Manter forte gerenciamento da interface com o usuário e / ou clientes, tanto no sentido de recebimento de solicitações, como de entrega dos produtos solicitados.
- Padronizar a entrada para a fábrica (a ordem de serviço ou solicitação de serviço).
- Basear as estimativas de prazo e custo na capacidade real de atendimento da fábrica para determinada demanda.
- Determinar métodos padrões de estimativas baseados em históricos.
- Providenciar, de preferência, tempos padrões de atendimento já estabelecidos de acordo com o domínio da aplicação, da plataforma tecnológica e do tamanho da demanda (programa e/ou projeto).
- Controlar os perfis de recursos humanos e alinhá-los ao tipo de demanda (natureza e complexidade) da fábrica.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

- Manter rigoroso controle dos recursos em termos de sua alocação, disponibilidade, necessidade futura e produtividade (que deve ser medida).
- Determinar um processo para o planejamento e controle da produção.
- Controlar o *status* das múltiplas demandas em seu processo e permitir seu rastreamento.
- Controlar todos os itens de software (documentos, métodos, procedimentos, ferramentas e código), criando uma biblioteca de itens.
- Monitorar eficientemente o andamento da execução de cada demanda.
- Construir os produtos de software de acordo com métodos, técnicas e ferramentas padronizados.
- Proporcionar processos distintos para atendimento a demandas de naturezas diferentes.
- Treinar e manter aptos todos os recursos humanos para cumprir as tarefas de desenvolvimento de software e para operar processos operacionais e de gestão.
- Propiciar processos de atendimento (resolução de problema) para os usuários e/ou clientes.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### **3. Fundamentos**

- Estabelecer mecanismos de apuração, apropriação e controle de custos.
- Proporcionar mecanismos de medições de atributos de sua operação, tais como: tempos médios de atendimento, densidade de defeitos dos produtos, eficiência de remoção de defeitos, exatidão das estimativas, e assim sucessivamente.
- Manter absoluto controle sobre os níveis de serviços acordados com seus usuários e/ou clientes.
- Melhorar seus processos de forma contínua, visando o aumento de sua produtividade e a redução de seus custos de operação.
- Possibilitar que o ambiente de hardware e software da fábrica seja estável e esteja alinhado às necessidades de seus usuários e/ou clientes.

FERNANDES (2004) define uma fábrica de software como um processo estruturado, que seja controlado e melhorado de forma contínua, considerando abordagens de engenharia industrial, orientado para o atendimento a múltiplas demandas de natureza e escopo distintas, visando a geração de produtos de software, conforme os requerimentos documentados dos usuários e/ou clientes, da forma mais produtiva e econômica possível.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 3. Fundamentos

#### **3.4.1. Indicadores de desempenho**

Segundo FERNANDES (2004), a gestão para a melhoria somente se concretiza por meio de medições, que permitam um entendimento do comportamento do processo. Esse entendimento é a base para que se mantenha o processo sob controle e para que sejam estabelecidas metas de melhoria. Na engenharia de software, essas medições são conhecidas como métricas.

Conforme definição dada por DASKALANTONAKIS (1992), métricas são métodos de determinar quantitativamente a extensão em que o projeto, o processo e o produto de software são influenciados por atributos, o que inclui a fórmula para determinar não só o valor da métrica, como também sua forma de apresentação e as diretrizes de utilização e interpretação dos resultados obtidos, no contexto do ambiente de desenvolvimento de software. As medições táticas têm como objetivo possibilitar a gestão do ambiente de software em sua totalidade, em termos de projetos, processos e produtos. Verificam tendências do ambiente relacionadas a qualidade, custo, produtividade, ciclos de tempo; verificam também o impacto de mudanças de processo e de tecnologia e analisam comparativamente ambiente de hardware e software, equipes de desenvolvimento em termos de qualidade, custos e produtividade (FERNANDES, 2004).

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

As medições operacionais, por sua vez, ocorrem no nível de cada produto ou produto de software. FERNANDES (2004) sugere algumas medições operacionais para controle da operação de uma fábrica de software. Essas medições estão relacionadas nos quadros 1.

**Quadro 1 – Medições operacionais, baseado em FERNANDES (2004)**

<b>Tipos de Medições</b>	<b>Medições</b>
	Estimativa de tamanho de software
	Estimativa de prazo
	Estimativa de esforço
	Estimativa de recursos
Medições realizadas para o desenvolvimento	Estimativa de custo
	Estimativa de qualidade
	Estimativa de confiabilidade
	Estimativa de produtividade
	Estimativa de defeitos
	Estimativa de retrabalho
Medições realizadas para a gestão do produto	Número de defeitos pós-release
	Desvio das estimativas
Medições no desenvolvimento	Produtividade das estimativas
	Produtividade das revisões
	Progresso na remoção de defeitos

## **Tomada de decisão em fábrica de software baseada em CEP**

### **3. Fundamentos**

<b>Tipos de Medição</b>	<b>Medição</b>
	Progresso físico
	Custo realizado
	Composição de tipos de defeitos
	Complexidade do software
	Origens de defeitos
Medições realizadas para a gestão do produto	Falhas esperadas para o software
	Intensidade atual de falhas
	Estimativa atualizada do número de defeitos pós-release
	Estimativa de atendimento
	Custo e produtividade do atendimento
Medições realizadas para a melhoria de processos	Exatidão das estimativas
	Tamanho do software entregue
	Produtividade do desenvolvimento
	Custo por unidade de tamanho do software
	Crescimento funcional do software
	Reutilização do software
	Complexidade do software
	Esforço por fase do ciclo de vida

**Tomada de decisão em fábrica de software baseada em CEP**

**3. Fundamentos**

<b>Tipos de Medição</b>	<b>Medição</b>
	Eficiência da remoção de defeitos
	Densidade de defeitos

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 4. Modelo de Medição de Referência

#### **4. Modelo de Medição de Referência**

Partindo das medições operacionais sugeridas por FERNANDES (2004), necessárias para a gestão operacional em uma fábrica de software, foram identificados na área gestora, os indicadores que iriam trazer maiores benefícios para a melhoria dos processos operacionais.

Selecionaram-se três indicadores para avaliação do processo e tomadas de decisão para melhoria. Esses indicadores são: defeitos por caso de uso, produtividade (linhas de código), desvio de escopo funcional. Com base neles, seguindo o PSM, foram criadas as construções mensuráveis para documentar os relacionamentos entre a necessidade de informação e as medições que seriam necessárias para construção dos indicadores. Os quadros 2, 3 e 4 descrevem as construções mensuráveis.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **4. Modelo de Medição de Referência**

**Quadro 2 – Construção mensurável para necessidade de informação quantidade de defeitos produzidos por caso de uso**

<b>Construção Mensurável</b>	
<b>Necessidade de Informação</b>	Quantidade de defeitos produzidos por caso de uso
<b>Categoria de Informação</b>	Qualidade do Produto
<b>Conceito Mensurável</b>	Correção Funcional
<b>Indicador</b>	Defeitos por caso de uso
<b>Modelo de Análise</b>	Usar o gráfico de controle para analisar os casos de uso que sofreram mais defeitos
<b>Critérios de decisão</b>	Investigar cada ponto que estiver fora dos limites de controle
<b>Medida Derivada</b>	Quantidade de defeitos por caso de uso
<b>Função de Medição</b>	Somar os defeitos relacionados para cada caso de uso
<b>Medidas Básicas</b>	1 - Quantidade de Casos de Testes 2 - Quantidade de falhas em cada caso de Teste
<b>Método de Medição</b>	1 - Contar quantos casos de teste cada caso de uso produziu 2 - Contar quantas falhas cada caso de teste produziu
<b>Tipo de Método</b>	1 - Objetivo 2 – Objetivo
<b>Escala</b>	1 - Inteiro 2 – Inteiro
<b>Tipo de Escala</b>	1 - Intervalo 2 – Intervalo
<b>Unidade de Medida</b>	1 - Caso de Uso 2 - Caso de Teste
<b>Entidades Relevantes</b>	1 - Diagrama de Caso de Casos de Uso 2 - Log de execução dos Testes
<b>Atributos</b>	1 - Quantidade de Casos de Teste 2 - Quantidade de Defeitos

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **4. Modelo de Medição de Referência**

**Quadro 3 – Construção mensurável para necessidade de informação produtividade das equipes de desenvolvimento**

<b>Construção Mensurável</b>	
<b>Necessidade de Informação</b>	Produtividade das equipes de desenvolvimento
<b>Categoria de Informação</b>	Performance do Processo
<b>Conceito Mensurável</b>	Eficiência do Processo
<b>Indicador</b>	Produtividade
<b>Modelo de Análise</b>	Usar o gráfico de controle para analisar os recursos mais produtivos
<b>Critérios de decisão</b>	Investigar cada ponto que estiver fora dos limites de controle
<b>Medida Derivada</b>	Produtividade do Projeto
<b>Função de Medição</b>	Dividir o tamanho do projeto pelo esforço
<b>Medidas Básicas</b>	1 - Esforço do Projeto 2 - Quantidade de linhas de código
<b>Método de Medição</b>	1 - Contar quantas horas foram necessárias para codificar cada caso de uso 2 - Contar as linhas de código produzidas por caso de uso
<b>Tipo de Método</b>	1 - Objetivo 2 - Objetivo
<b>Escala</b>	1 - Inteiro 2 - Inteiro
<b>Tipo de Escala</b>	1 - Intervalo 2 - Intervalo
<b>Unidade de Medida</b>	1 - Horas 2 - Linha de Código
<b>Entidades Relevantes</b>	1 - Folha de Ponto 2 - Realização de Caso de Uso
<b>Atributos</b>	1 - Horas Apontadas 2 - Ponto e vírgula

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **4. Modelo de Medição de Referência**

**Quadro 4 – Construção mensurável para necessidade de informação desvio de escopo**

<b>Construção Mensurável</b>	
<b>Necessidade de Informação</b>	Desvio de escopo que houve na construção dos projetos
<b>Categoria de Informação</b>	Tamanho e Estabilidade do produto
<b>Conceito Mensurável</b>	Tamanho funcional e estabilidade
<b>Indicador</b>	Desvio de Escopo
<b>Modelo de Análise</b>	Usar o gráfico de controle para analisar os casos de uso que se desviaram do escopo inicial
<b>Critérios de decisão</b>	Investigar cada ponto que estiver fora dos limites de controle
<b>Medida Derivada</b>	Produtividade do Projeto
<b>Função de Medição</b>	Identificar os casos de uso que sofreram aumento de escopo
<b>Medidas Básicas</b>	1 - Quantidade de <i>Use Case Point</i> - Fase de Iniciação 2 - Quantidade de <i>Use Case Point</i> - Fase de Elaboração
<b>Método de Medição</b>	1 - Contar os <i>Use Case Points</i> para cada caso de uso 2 - Contar os <i>Use Case Points</i> para cada caso de uso
<b>Tipo de Método</b>	1 - Objetivo 2 - Objetivo
<b>Escala</b>	1 - Inteiro 2 - Inteiro
<b>Tipo de Escala</b>	1 - Intervalo 2 - Intervalo
<b>Unidade de Medida</b>	1 - <i>Use Case Points</i> 2 - <i>Use Case Points</i>
<b>Entidades Relevantes</b>	1 - Caso de Uso 2 - Caso de Uso
<b>Atributos</b>	1 - Quantidade de <i>Use Case Point</i> 2 - Quantidade de <i>Use Case Point</i>

## **Tomada de decisão em fábrica de software baseada em CEP**

### **4. Modelo de Medição de Referência**

Após a definição dos indicadores e de suas construções mensuráveis, foi necessário selecionar os gráficos de controle que seriam construídos para cada indicador relacionado na construção mensurável. Para a seleção dos gráficos, foi necessário entender o comportamento de um processo de desenvolvimento de software é como um processo de manufatura, em que o CEP é utilizado com maior freqüência.

Os indicadores descritos na construção mensurável foram analisados da seguinte forma:

- O desvio de escopo é um indicador que apresenta a quantidade de desvio que uma funcionalidade sofreu durante o ciclo de vida do desenvolvimento de software, sendo assim uma contagem. A amostra é uma amostra individual e constante, com isso, poderíamos usar, segundo o fluxograma proposto por RAMOS (2002, p. 325), os gráficos *c* ou *u*. No entanto, estamos utilizando o desvio de escopo por caso de uso. Segundo STEVENSON (1981), a distribuição de Poisson é útil para descrever as probabilidades do número de ocorrências num campo ou intervalo contínuo (em geral tempo ou espaço). Como teremos o desvio por caso de uso como o dado tratado por esse indicador, tomaremos essa distribuição como uma distribuição de Poisson. Com isso, o gráfico que se enquadra nesse cenário é o gráfico *c*.

### **Tomada de decisão em fábrica de software baseada em CEP**

---

#### 4. Modelo de Medição de Referência

- A quantidade de defeitos por caso de uso apresenta cenário semelhante ao do indicador de desvio de escopo por caso de uso. Sendo assim, utilizamos o gráfico *c*.
- A produtividade será avaliada como a quantidade de horas que um desenvolvedor leva para escrever 100 linhas de código de uma mesma linguagem de programação. Nesse caso, será considerado um software desenvolvido em Java. Consideramos o gráfico *c* para analisar este indicador.

Para todos os indicadores foram construídos formulários para orientar a coleta de dados e facilitar a construção dos gráficos de controle.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)

### **5. Primeiro ciclo da pesquisa-ação: análise dos indicadores**

Neste capítulo será descrito o primeiro ciclo da pesquisa-ação, para isso serão avaliados os indicadores e os gráficos propostos no modelo de referência, descritos no capítulo 4. Cada indicador será avaliado e recomendações de melhoria de processo serão descritas.

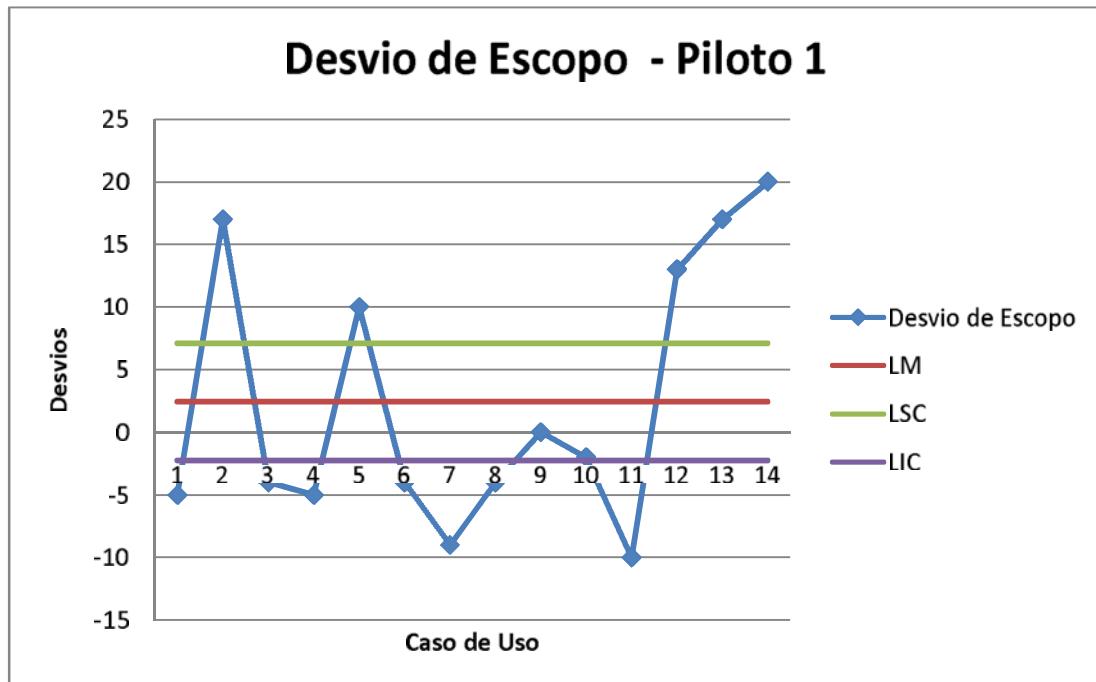
#### **5.1. Desvio de escopo**

O primeiro indicador escolhido para ser coletado e avaliado foi o desvio de escopo funcional. Os dados para esse indicador foram colhidos em dois momentos: na definição do escopo do projeto (fase de proposta) e na entrega do projeto. No desenvolvimento de software as mudanças no escopo do projeto são tratadas como fatores comuns no desenvolvimento. No entanto, conhecer quais são as mudanças e os impactos que ela traz em custo e prazo no resultado do projeto é essencial para a gestão efetiva de uma fábrica de software.

Para construção deste gráfico foram coletados os tamanhos funcionais dos casos de uso identificados no momento da definição do escopo. Esse tamanho funcional foi contado utilizando uma técnica de estimativa conhecida como UCP (*Use Case Point*). Dessa forma, o desvio de escopo foi considerado como sendo a diferença entre o que foi entregue e o que foi estimado na definição do escopo. Os valores negativos apontados no gráfico mostram que a funcionalidade que foi entregue é menor do que foi

### **Tomada de decisão em fábrica de software baseada em CEP**

5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores) estimado. No entanto, foram entregues funcionalidades com tamanho superior ao que foi estimado. A figura 10 mostra o indicador de desvio de escopo para o projeto-piloto 1.



**Figura 10 – Gráfico de desvio de escopo do projeto-piloto 1**

O gráfico de figura 10 evidencia que o processo apresenta alguns pontos que não estão dentro dos limites de controle aceitáveis da perspectiva do CEP. Esses pontos representam oportunidades de melhoria no processo de gerência de requisito. Cada causa especial foi investigada e seu resultado está relacionado abaixo:

- O tempo para fazer a estimativa inicial do projeto foi bem menor do que o praticado na empresa.

### **Tomada de decisão em fábrica de software baseada em CEP**

---

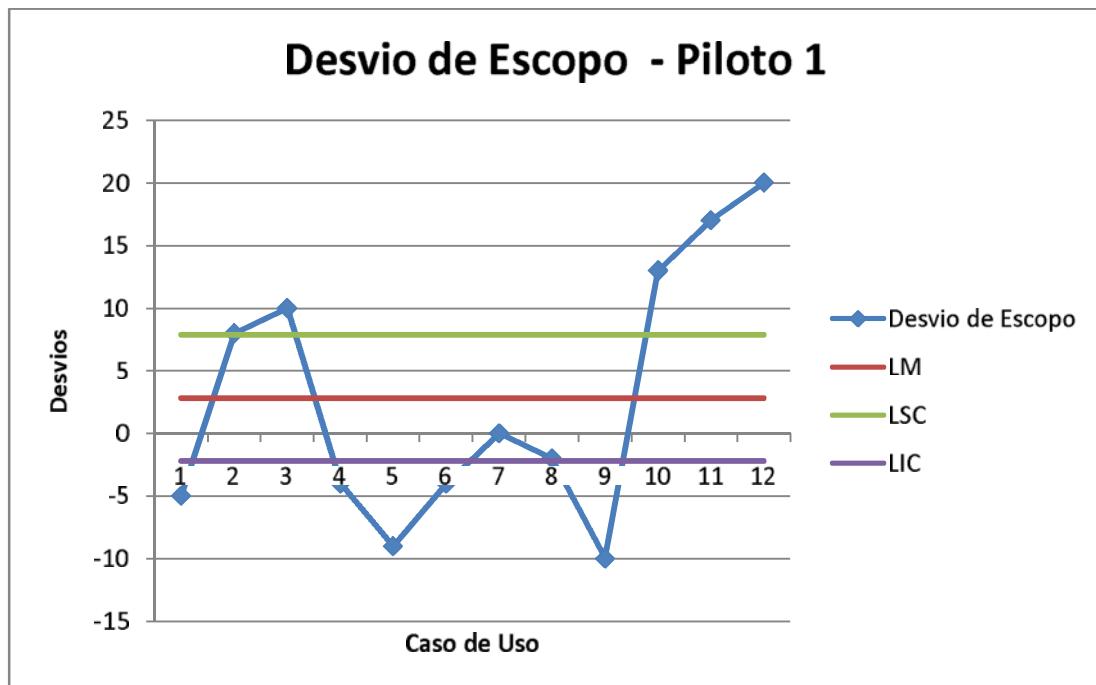
#### 5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)

- O analista que fez a estimativa inicial do projeto não foi o mesmo que fez o levantamento dos dados para implementação do projeto.
- Houve várias mudanças no escopo do projeto que não foram documentadas nem passaram por análises de impacto.
- Não existe um processo formal para solicitação de mudança de escopo e não há prática de realização de análise de impacto para as mudanças.

Durante o processo de investigação para identificar os motivos do desvio de escopo, foi constatado que algumas funcionalidades de um caso de uso foram implementadas em outro. Com isso, algum desvio negativo que tenha sido identificado pelo gráfico não significa que a funcionalidade não tenha sido entregue. Apenas que ela está contida em um caso de uso diferente do previsto. Após essa conclusão, o gráfico foi reorganizado para refletir esse novo comportamento do processo, conforme apresentado na figura 11.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**



**Figura 11 – Gráfico de desvio de escopo do projeto-piloto 1, ajustado**

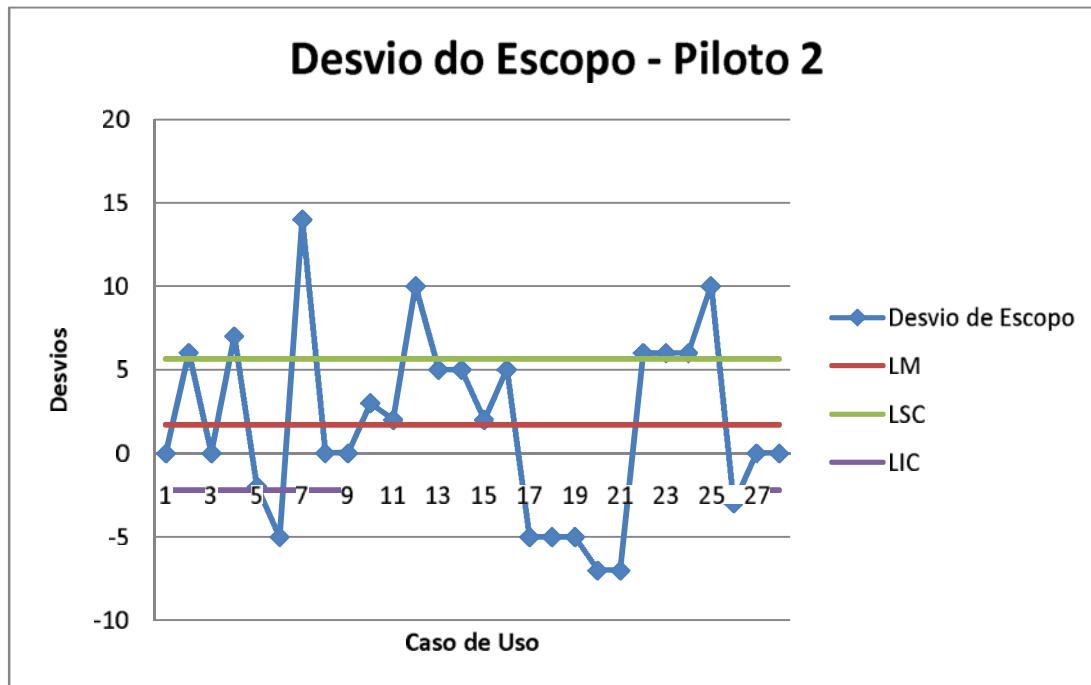
Mesmo com a reorganização das funcionalidades, o processo se apresenta instável de acordo com o CEP. Embora com maior controle os desvios aconteceram e demonstraram-se os mesmos fatores descritos no projeto-piloto 1. Além desses, foi constatado que o processo de desenvolvimento de software não está sendo utilizado da mesma forma pelos participantes do processo.

O segundo projeto-piloto apresentou o gráfico de controle da figura 12, que traz um processo que causou alguns desvios considerados fora dos limites de controle. No entanto, ao contrário do projeto-piloto 1, houve mais pontos dentro dos limites do que fora. Como o projeto-piloto 2 é maior (na perspectiva funcional) do que o projeto-piloto 1 o gerente aplicou maior

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**

controle ao projeto, fazendo com o que os desvios de escopo fossem menores e mais controlados do que ocorreu no projeto-piloto 1.



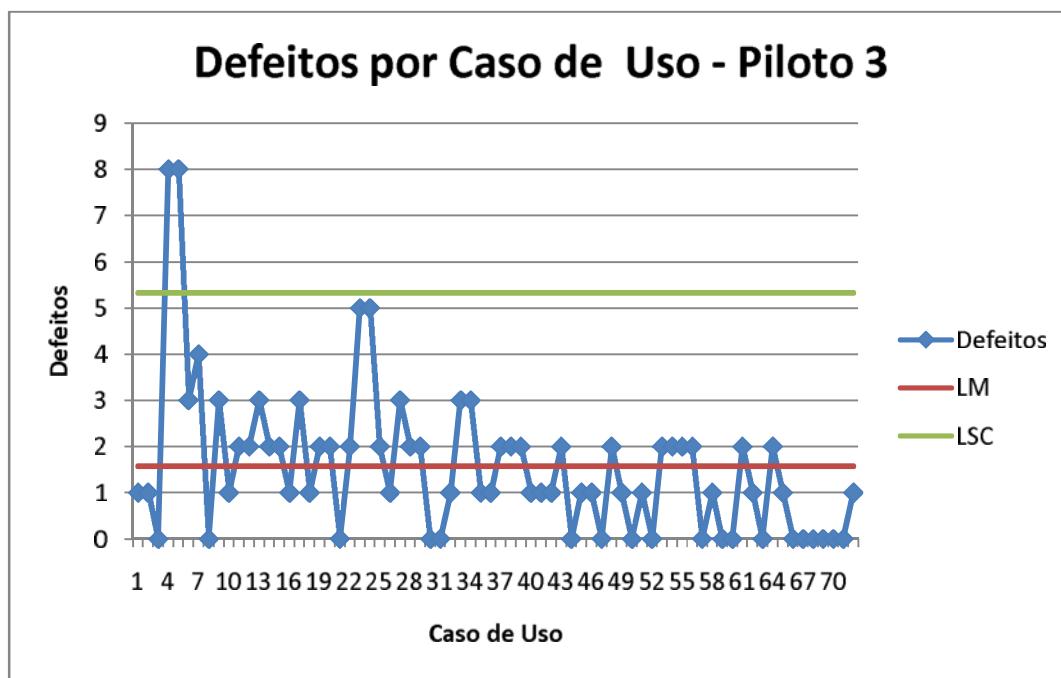
**Figura 12 – Gráfico de desvio de escopo do projeto-piloto 2**

### **5.2. Defeitos por caso de uso**

O segundo indicador a ser coletado e avaliado foi o de defeitos por caso de uso. Nesse indicador foi considerada a execução de um projeto-piloto para avaliar o processo de construção do software. O gráfico resultante da coleta de dados está representado na figura 13.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**



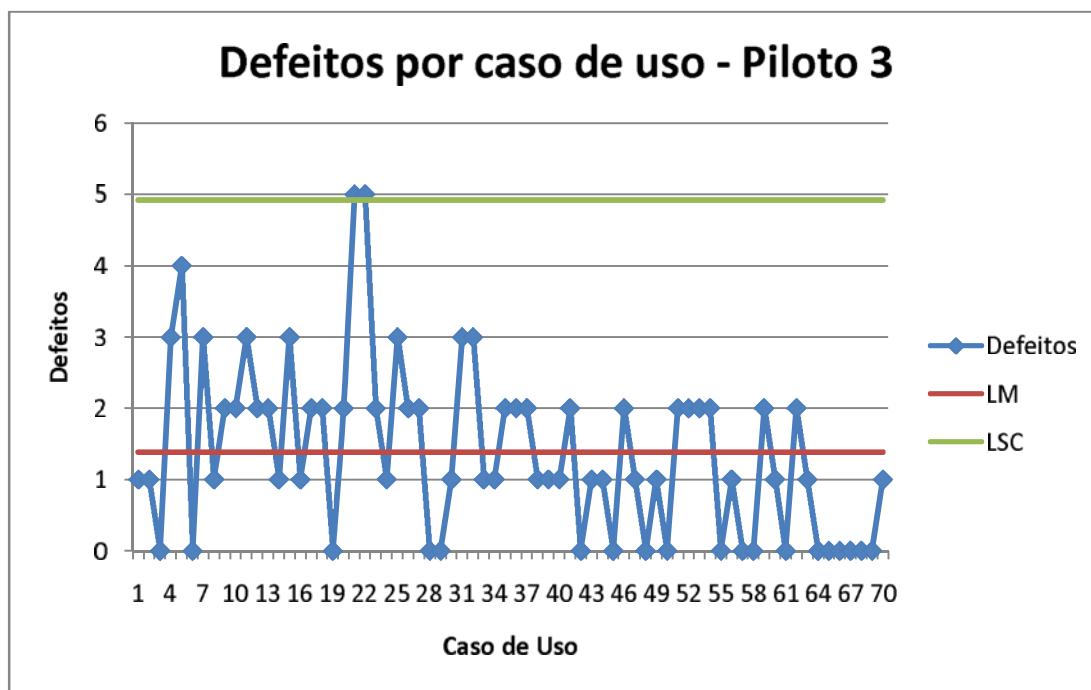
**Figura 13 – Gráfico de defeitos por caso de uso do projeto-piloto 3**

Seguindo o roteiro de melhoria de processos do CEP (ver figura 8), as causas especiais apontadas no gráfico foram investigadas. O resultado dessa investigação indicou que nesses dois pontos (que estão acima do limite superior de controle), as funcionalidades implementadas eram telas de cadastros. Essas funcionalidades possuíam uma grande quantidade de validações e por isso o processo de construção desses softwares apresentou como resultado mais defeitos por caso de uso.

Seguindo o processo de investigação, esses dois pontos foram removidos e o gráfico foi reconstruído conforme apresentado na figura 14.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**



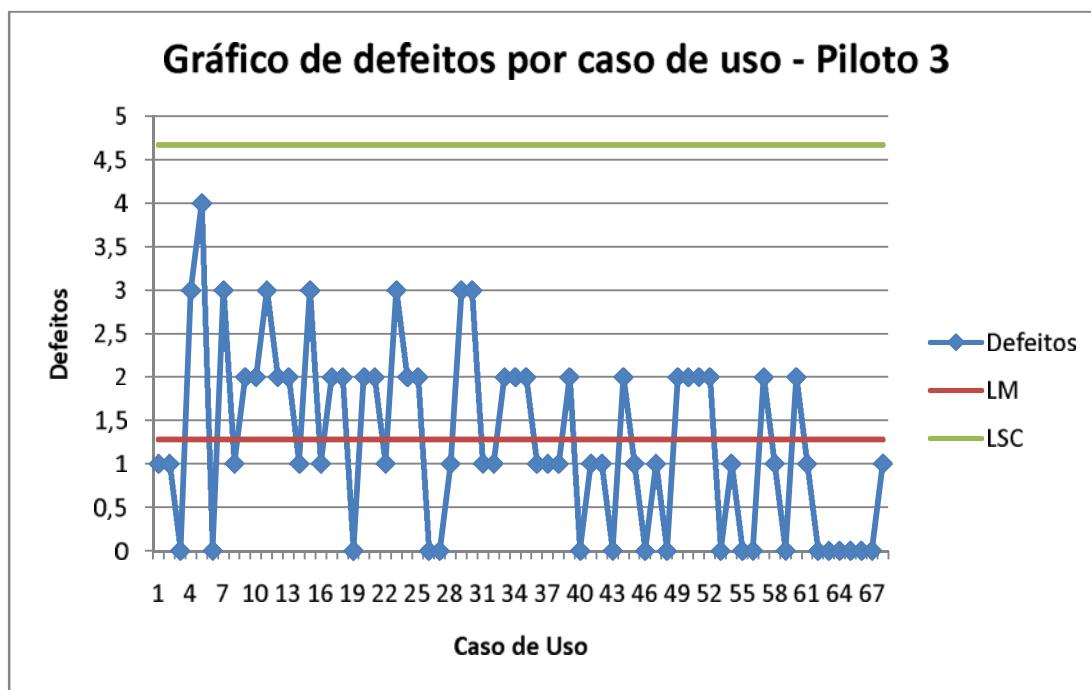
**Figura 14 – Gráfico de defeitos por caso de uso do projeto-piloto 3, sem os pontos 5 e 6**

Após a retirada desses pontos o processo apresentou mais dois pontos fora do limite de controle superior. Ao investigar os casos de uso que apresentaram esses dois pontos fora do limite, houve uma relação com os pontos fora do limite de controle apresentado na figura 13. Nos dois pontos fora dos limites de controle da figura 13, os casos de uso eram responsáveis pela funcionalidade de cadastro de clientes. Os pontos que estão fora dos limites na figura 14 são os casos de uso que fazem a consulta ao cadastro de clientes.

Os dois pontos foram removidos e o gráfico foi reconstruído conforme apresenta a figura 15.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**



**Figura 15 – Gráfico de defeitos por caso de uso do projeto-piloto 3, sem os pontos 21 e 22**

Após a remoção desses dois pontos, o processo se apresentou estável de acordo com as regras definidas no CEP. Após a investigação, detectou-se como ponto de melhoria a comunicação entre o processo de gerência de requisitos e testes. Por isso, foi recomendado *workshop* de melhores práticas de escrita de requisitos e a importância de um requisito para o processo de testes.

### **5.3. Produtividade**

O terceiro indicador de melhoria de processo diz respeito à produtividade no desenvolvimento utilizando a linguagem de programação

### **Tomada de decisão em fábrica de software baseada em CEP**

5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores) Java. Esse processo possui um alto nível de automação para geração de código em razão da sua arquitetura não permitir alterações. O projeto-piloto para exercitar este indicador possuía 17 classes (programas) Java. As figuras 16 e 17 apresentam o gráfico  $\bar{X}$  e  $R, m$  respectivamente do processo de codificação.

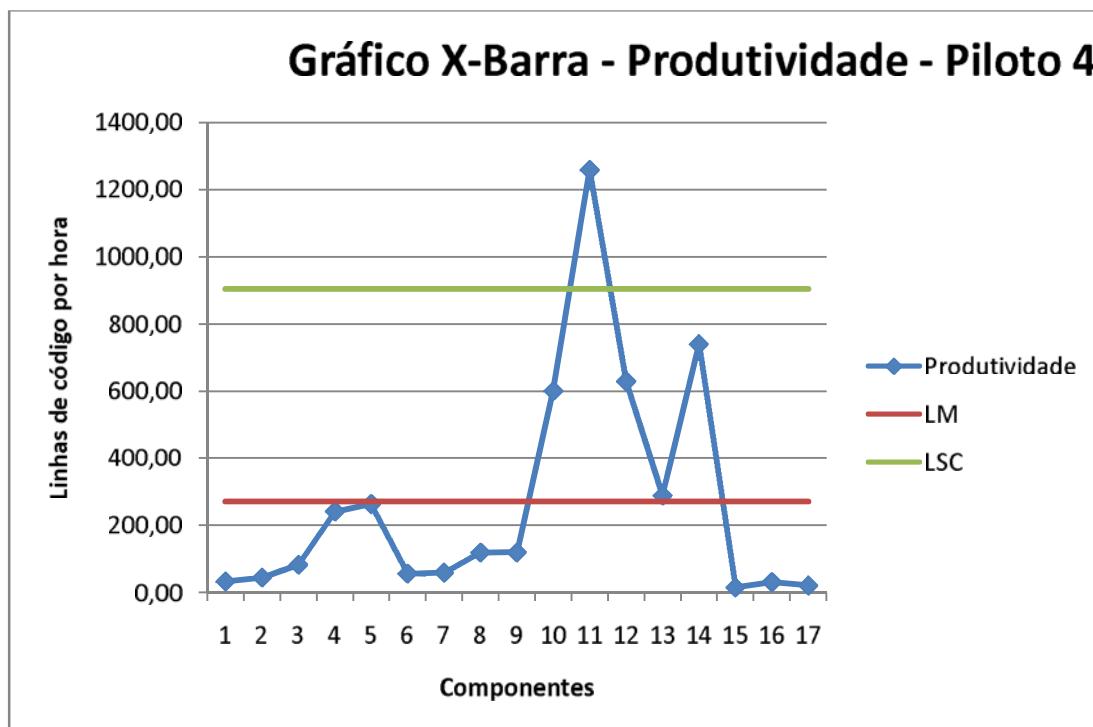
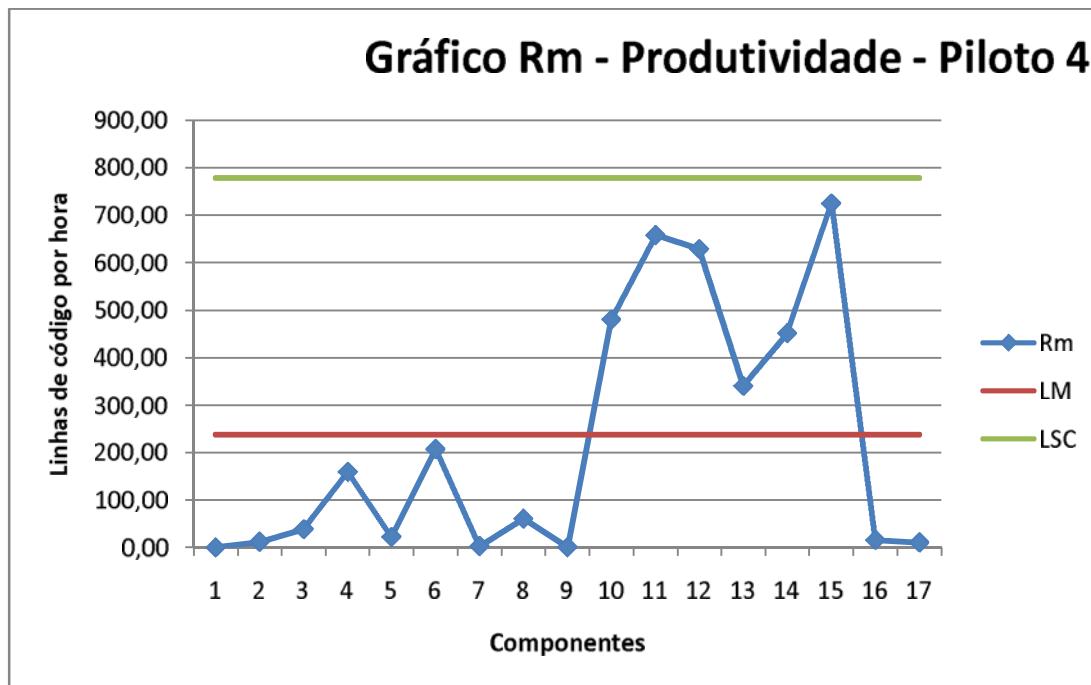


Figura 16 – Gráfico  $\bar{X}$  para o indicador de produtividade na codificação Java

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **5. Primeiro ciclo da pesquisa-ação (Análise dos indicadores)**



**Figura 17 – Gráfico Rm para o indicador de produtividade na codificação Java**

Analisando inicialmente o gráfico  $Rm$ , verificamos que não há causas especiais de variação atuando na dispersão (variabilidade) do processo, já que não há pontos fora dos limites de controle, e estes se distribuem aleatoriamente (ao acaso) em torno da média. Portanto, o valor de  $\bar{R}m$  pode ser considerado satisfatório para representar a variabilidade desse processo.

O gráfico de controle para valores individuais ( $\bar{X}$ ) mostra que existe um ponto fora do limite. Ao investigá-lo descobriu-se que apesar do alto índice de automação desse processo essa classe precisou da intervenção do desenvolvedor. O processo foi considerado fora de controle.

### **Tomada de decisão em fábrica de software baseada em CEP**

---

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência

## **6. Segundo ciclo da pesquisa-ação: melhoria do modelo de referência**

Segundo RAMOS (2002, p. 325), para que os gráficos de controle c sejam eficazes, deve-se ter  $\bar{c} > 5$ . Esse critério possibilita o surgimento de defeitos suficientes na amostra, de modo que o gráfico possa apontá-los. Nos gráficos apresentados no capítulo 5, especificamente os que utilizaram o gráfico c, verificou-se que nos indicadores de desvio de escopo e densidade de defeitos, o  $\bar{c}$  foi menor que 5.

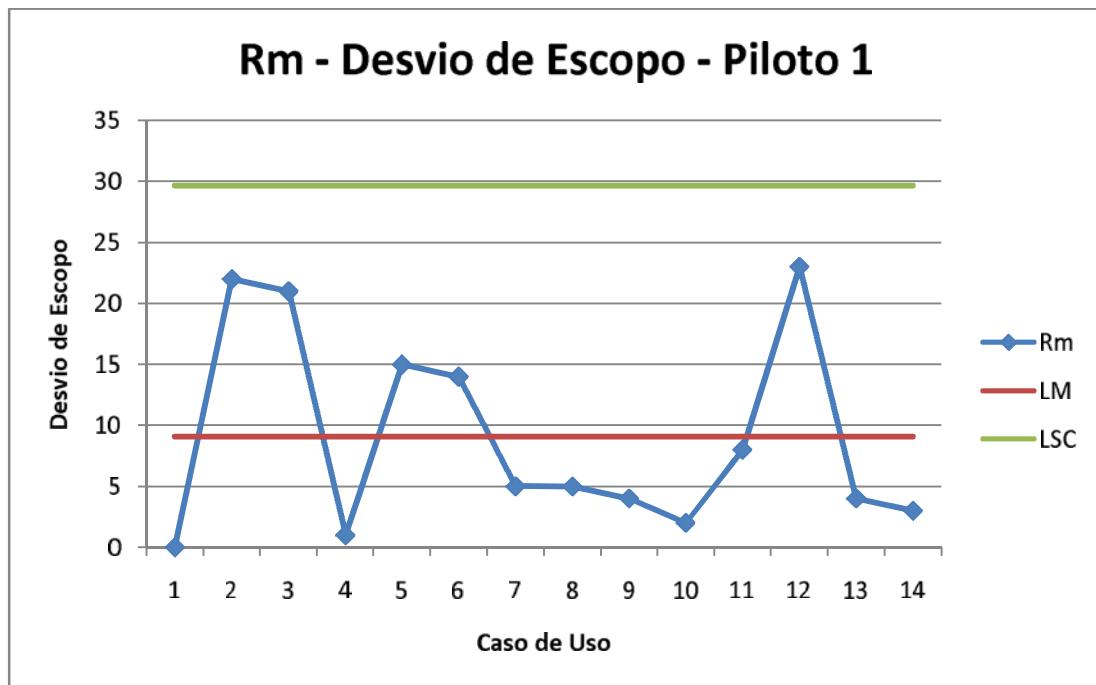
Seguindo o roteiro da pesquisa-ação, o modelo de referência foi alterado para refletir a necessidade de alterar o gráfico de controle c para  $\bar{X}$  e  $Rm$ . Neste capítulo serão descritos as análises dessa troca de gráfico, e apresentados os novos gráficos de controle.

### **6.1. Desvio de escopo**

O gráfico c exposto no capítulo 5, figura 10, apresentou a linha média igual a 2,43, ou seja  $\bar{c} < 5$ . Com isso, o gráfico c foi trocado pelo par de gráficos  $\bar{X}$  e  $Rm$ . A figura 18 apresenta o gráfico  $Rm$  para o projeto-piloto 1.

### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência

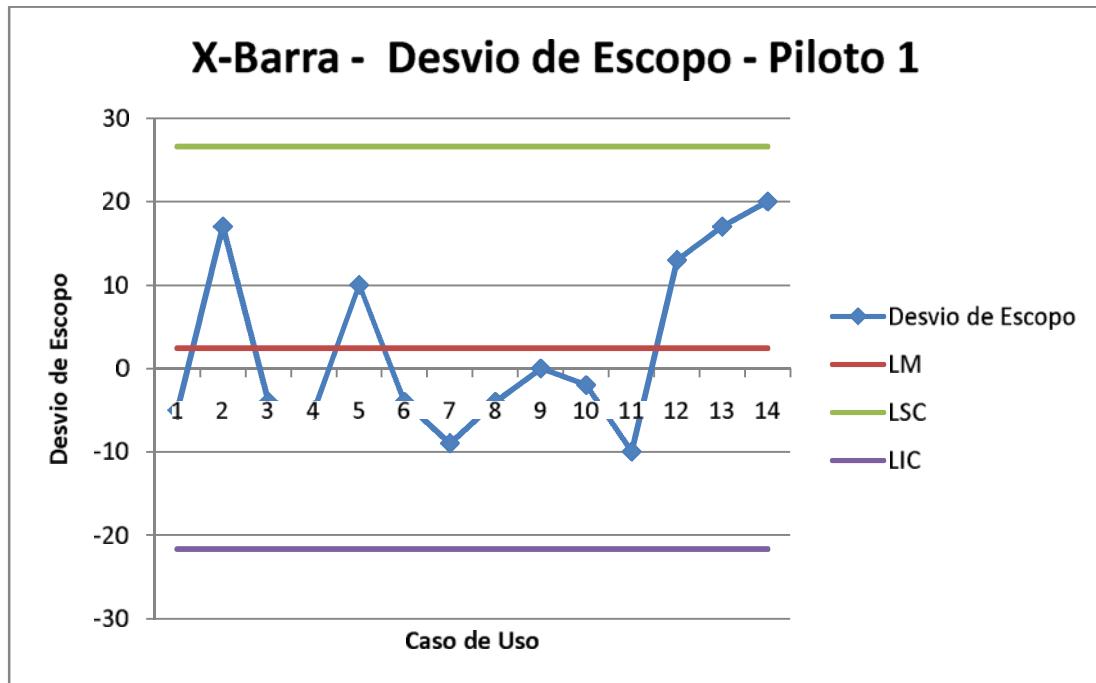


**Figura 18 – Gráfico Rm para o indicador de desvio de escopo do projeto-piloto 1**

Esse gráfico apresenta um processo que não possui causas especiais e por isso, podemos dizer que ele está estatisticamente controlado. A figura 19 expõe o gráfico  $\bar{X}$ , que assim como o  $Rm$ , também indica um processo estatisticamente controlado.

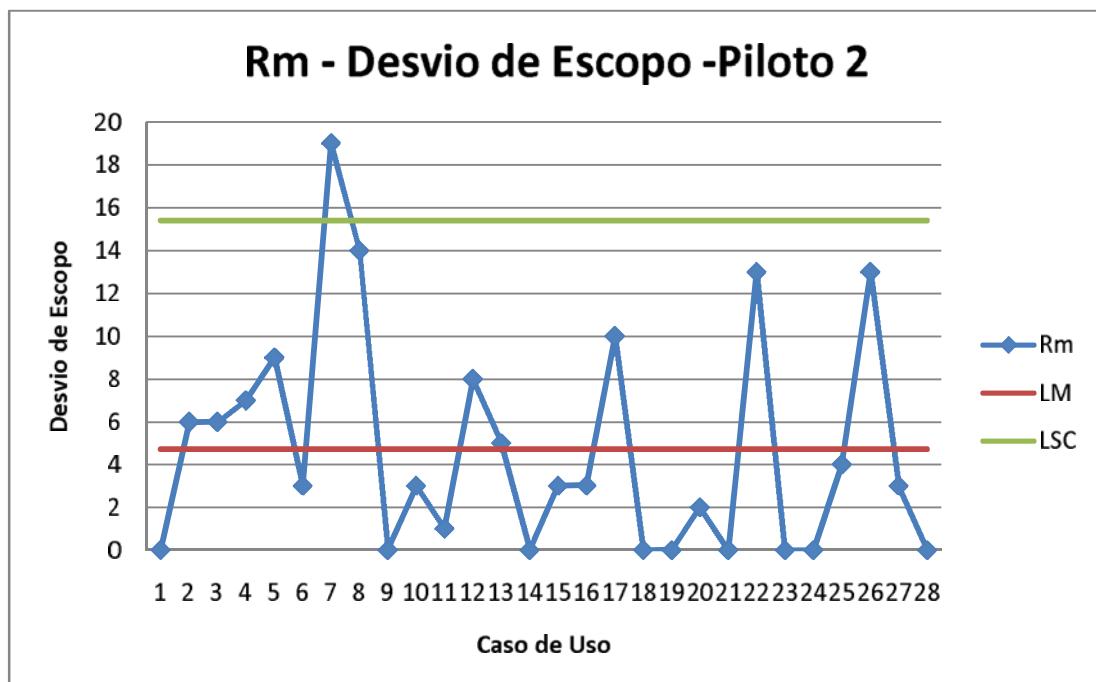
### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência



**Figura 19 – Gráfico  $\bar{X}$  para o indicador de desvio de escopo do projeto-piloto 1**

O projeto-piloto 2 apresentou os gráficos ilustrados nas figuras 20 e 21.

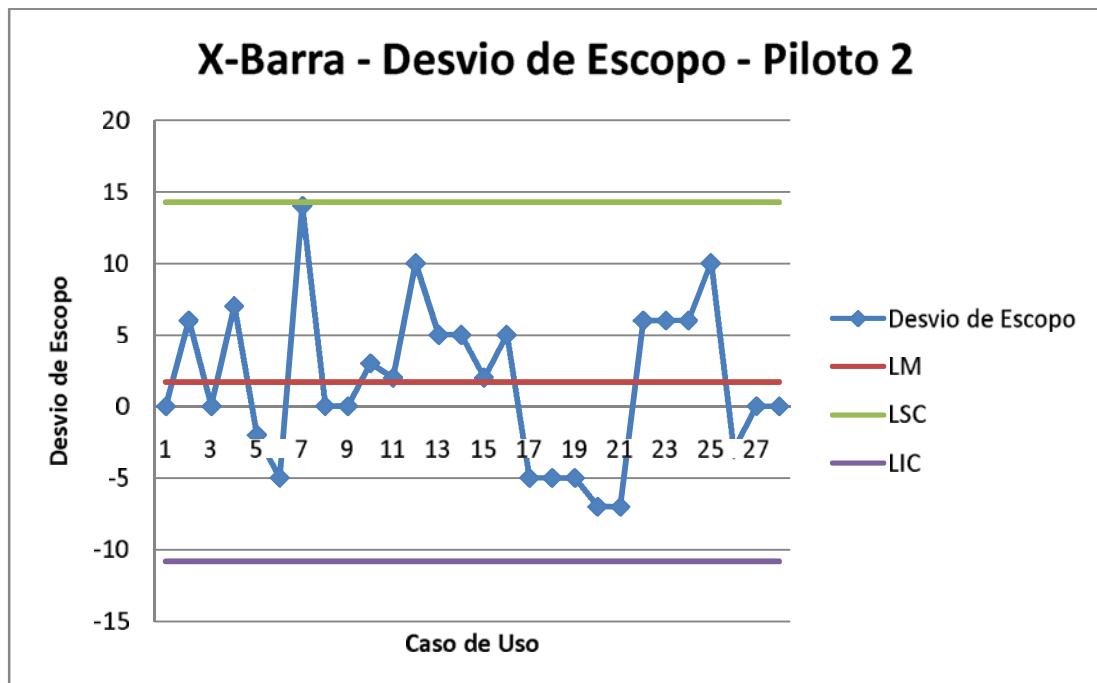


**Figura 20 – Gráfico  $R_m$  para o indicador de desvio de escopo do projeto-piloto 2**

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência**

A figura 20 apresenta o gráfico  $Rm$ , que demonstra que o ponto 7 está fora dos limites de controle. Ao analisar esse ponto notou-se que houve uma falha na estimativa para definição do escopo inicial do projeto. A figura 21 traz o gráfico  $\bar{X}$ , que, mesmo estando estável, evidencia o ponto 7 muito próximo do limite.

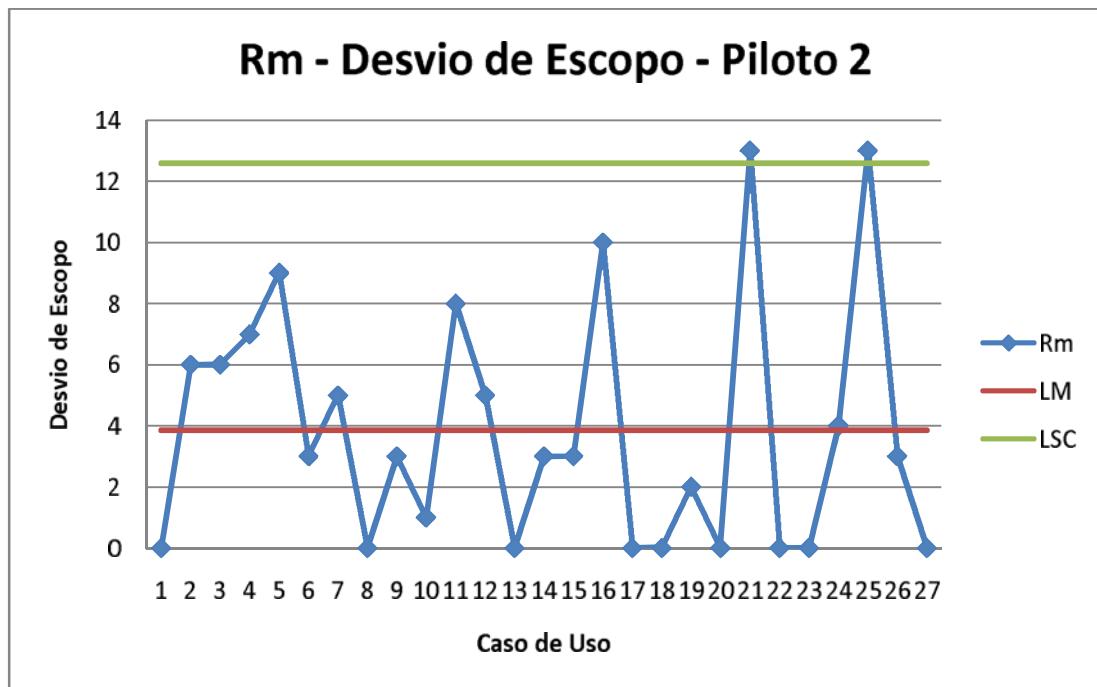


**Figura 21 – Gráfico  $\bar{X}$  para o indicador de desvio de escopo do projeto-piloto 2**

Seguindo o roteiro de investigação, o ponto 7 foi retirado e os gráficos foram reconstruídos e apresentados nas figuras 22 e 23.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência**

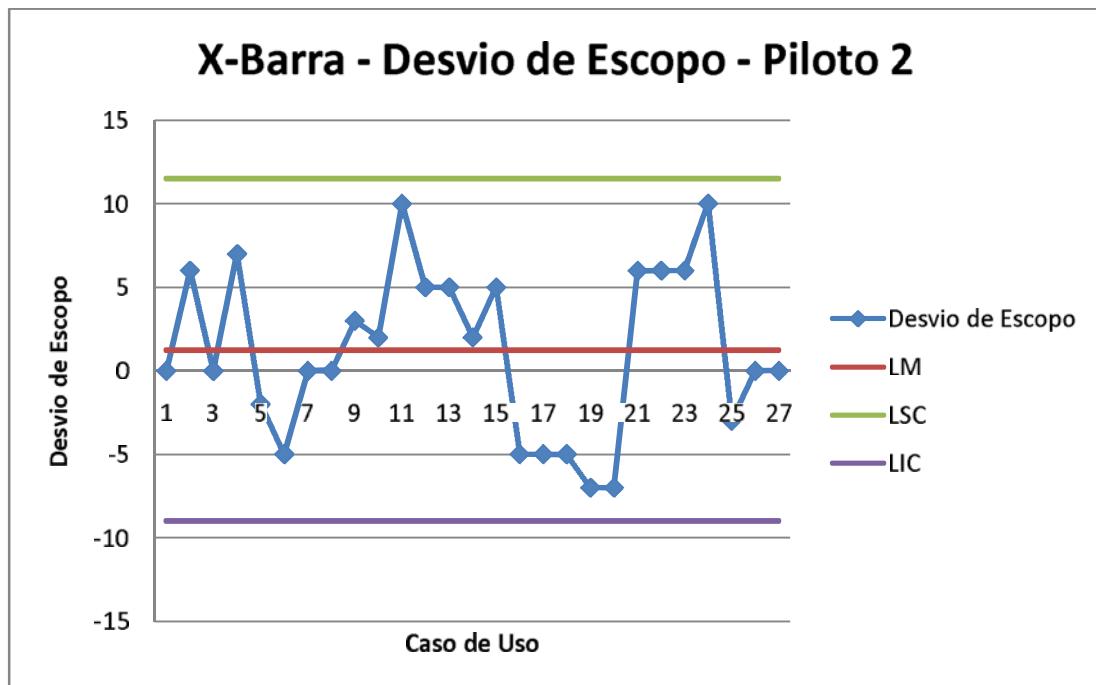


**Figura 22 – Gráfico *Rm* para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 7 da figura 20**

Após retirar o ponto 7, constante na figura 20, o gráfico da execução do processo demonstrou que existem dois pontos fora dos limites. Seguindo o processo de investigação, os pontos foram investigados, retirados e o gráfico refeito. Assim como na figura 20, eles apresentaram problemas no processo de estimativa no início do projeto.

**Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência

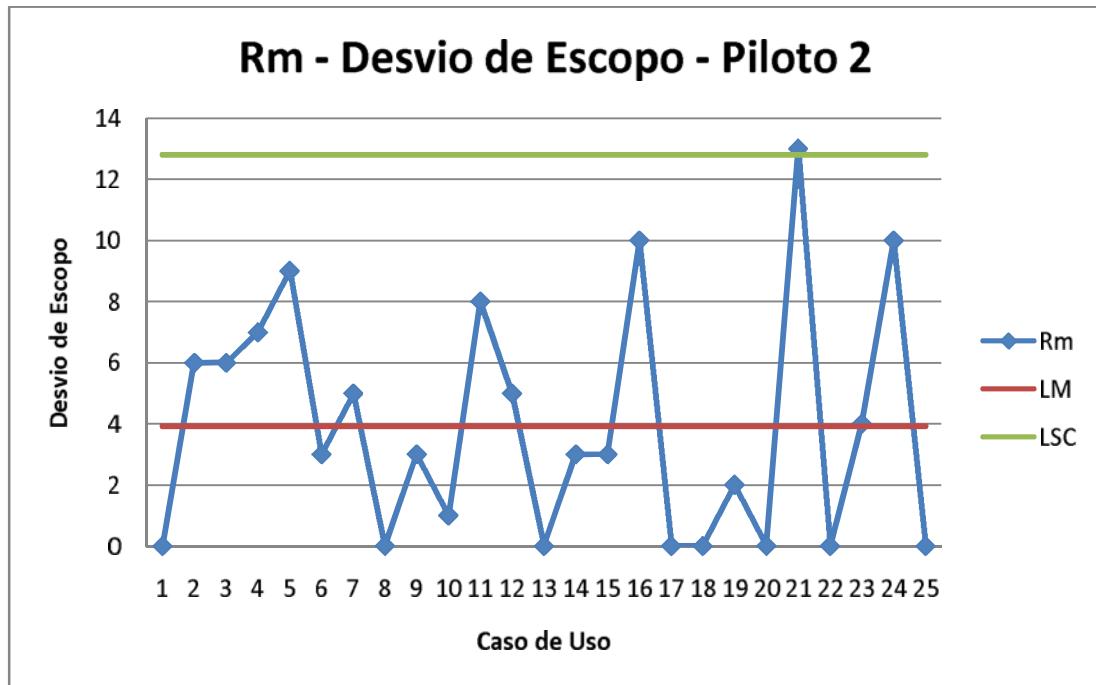


**Figura 23 – Gráfico  $\bar{X}$  para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 7 da figura 20**

A figura 23 expõe o processo estável, mesmo com o gráfico  $Rm$  indicando causas especiais. As figuras 24 e 25 trazem os gráficos após a remoção dos pontos 21 e 25 da figura 22.

### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência

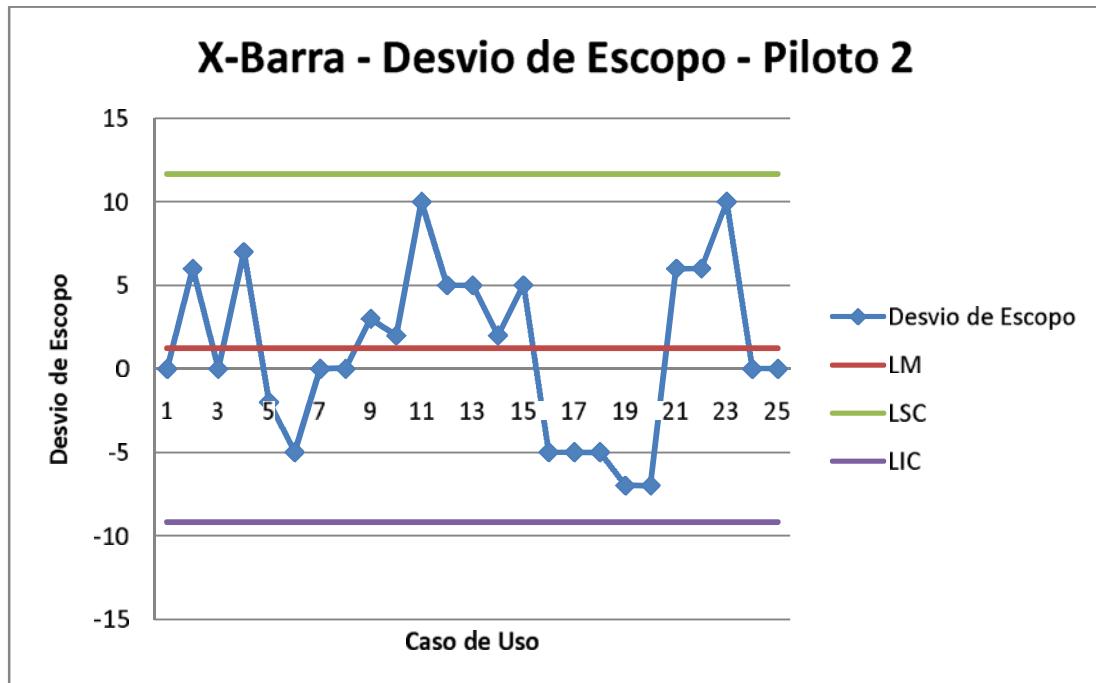


**Figura 24 – Gráfico *Rm* para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 21 e 25 da figura 22**

A figura 24 apresenta o ponto 21 fora dos limites de controle. Mesmo tendo esse ponto fora do limite, o ciclo de investigação foi finalizado. A figura 25 exibe o gráfico  $\bar{X}$ .

### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência



**Figura 25 – Gráfico  $\bar{X}$  para o indicador de desvio de escopo do projeto-piloto 2, sem o ponto 21 e 25 da figura 22**

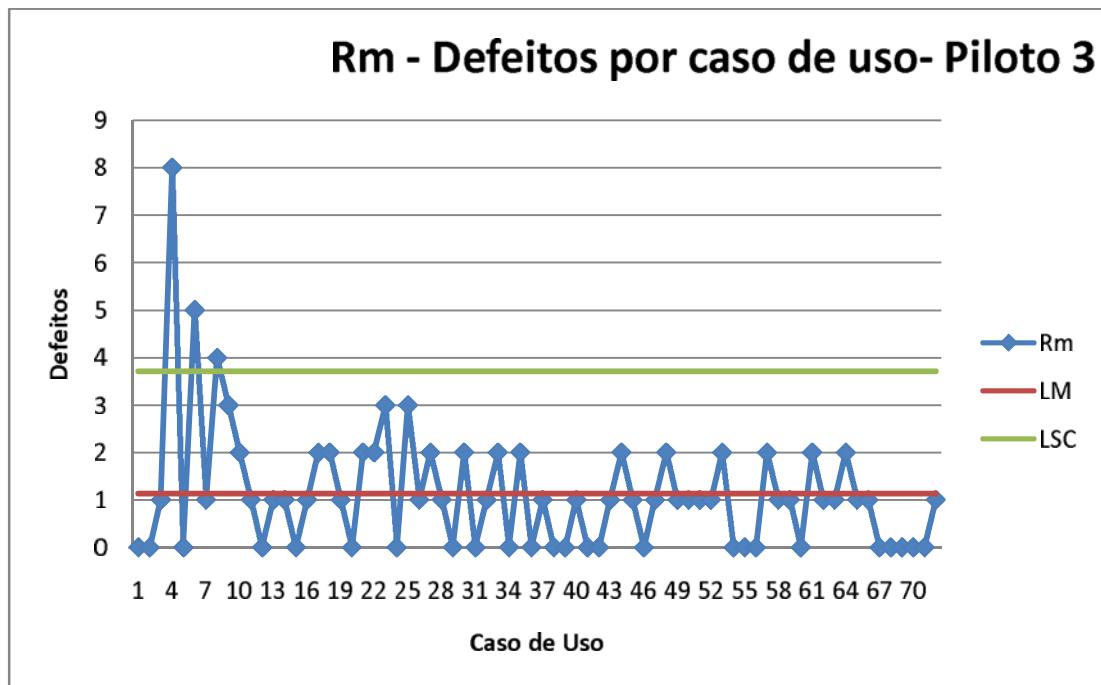
Assim como nos pontos retirados dos gráficos expostos nas figuras 20, 22 e 24 as causas especiais encontradas estão relacionadas à falta de identificação das funcionalidades no momento da estimativa inicial do projeto. Foram recomendados treinamentos e acompanhamento no processo de identificação de requisitos utilizando a técnica de caso de uso.

### **6.2. Defeitos por caso de uso**

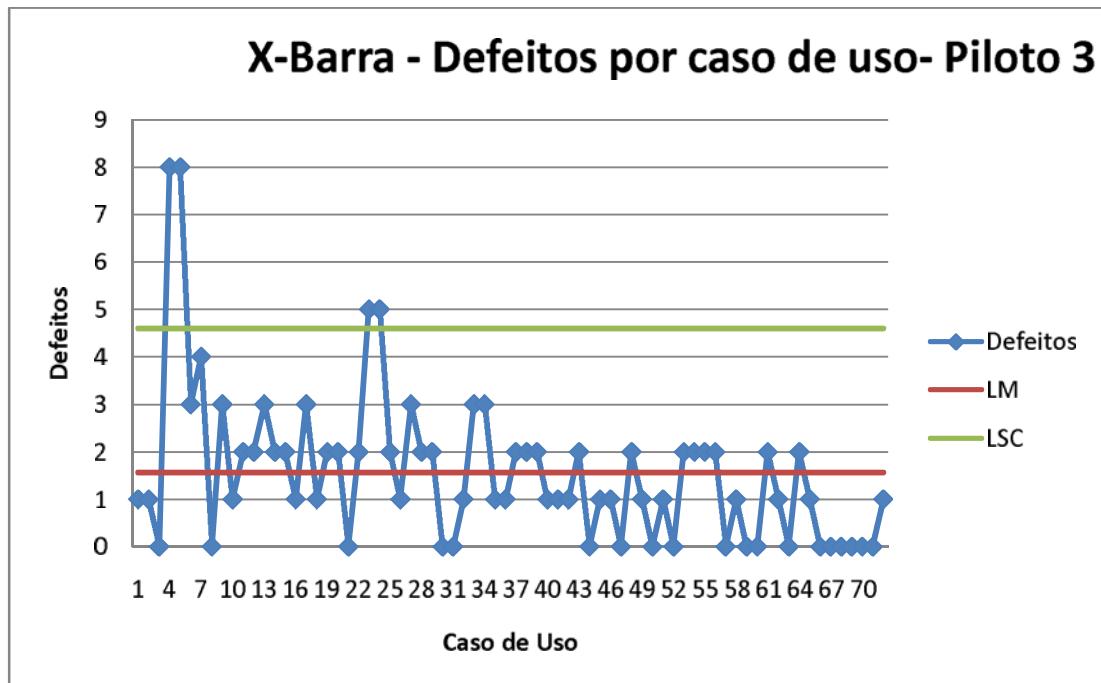
Assim como o indicador de desvio de escopo, o indicador de defeitos por caso de uso apresentou  $\bar{c} < 5$ . Com isso, o gráfico  $c$  foi trocado pelo par de gráficos  $\bar{X}$  e  $Rm$ . A figura 26 apresenta o gráfico  $Rm$  e a figura 27 apresenta o gráfico  $\bar{X}$ .

### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência



**Figura 26 – Gráfico Rm para o indicador de defeitos por caso de uso**



**Figura 27 – Gráfico X para o indicador de defeitos por caso de uso**

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência**

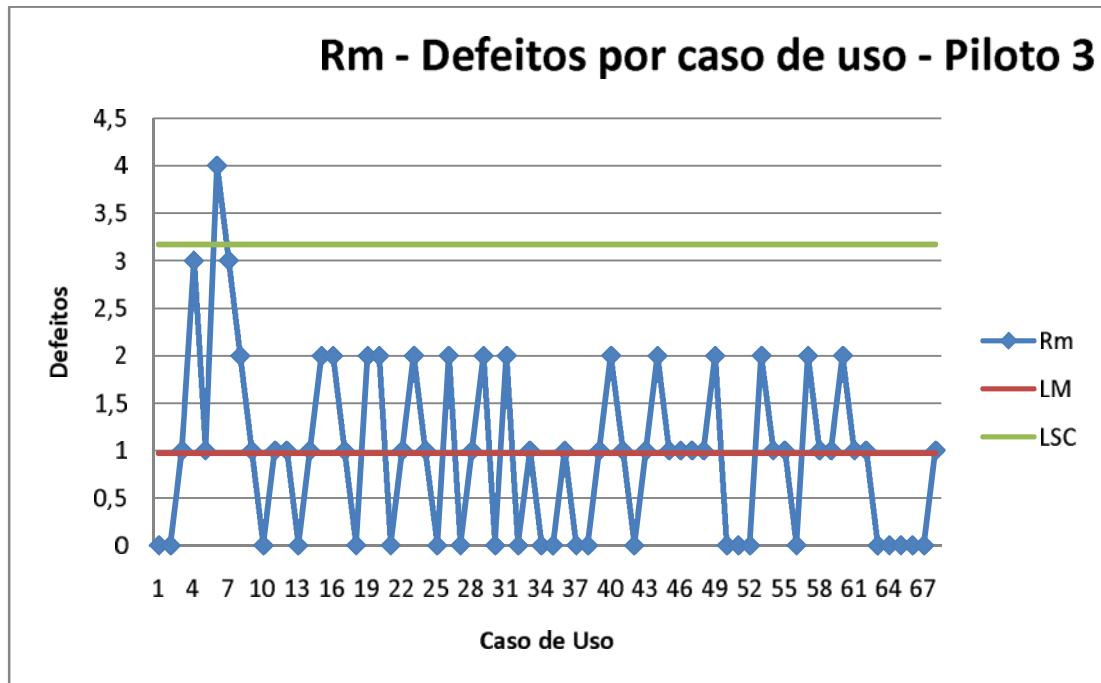
As figuras 26 e 27 apresentam causas especiais. Diferentemente do indicador de desvio de escopo do projeto-piloto 2, as investigações para as figuras 26 e 27 serão baseadas no gráfico  $\bar{X}$ . Por meio da investigação, os seguintes resultados foram encontrados:

- Os pontos 4 e 5 são referentes a uma funcionalidade de cadastro de cliente. Essa funcionalidade possui uma quantidade elevada de validações, que por sua vez, não estavam claramente definidas nas especificações de requisitos.
- Os pontos 23 e 25 são consultas de clientes, e estão relacionadas às funcionalidades de cadastro. Essas consultas possuíam filtros e regras que não estavam claras nas especificações de requisitos.

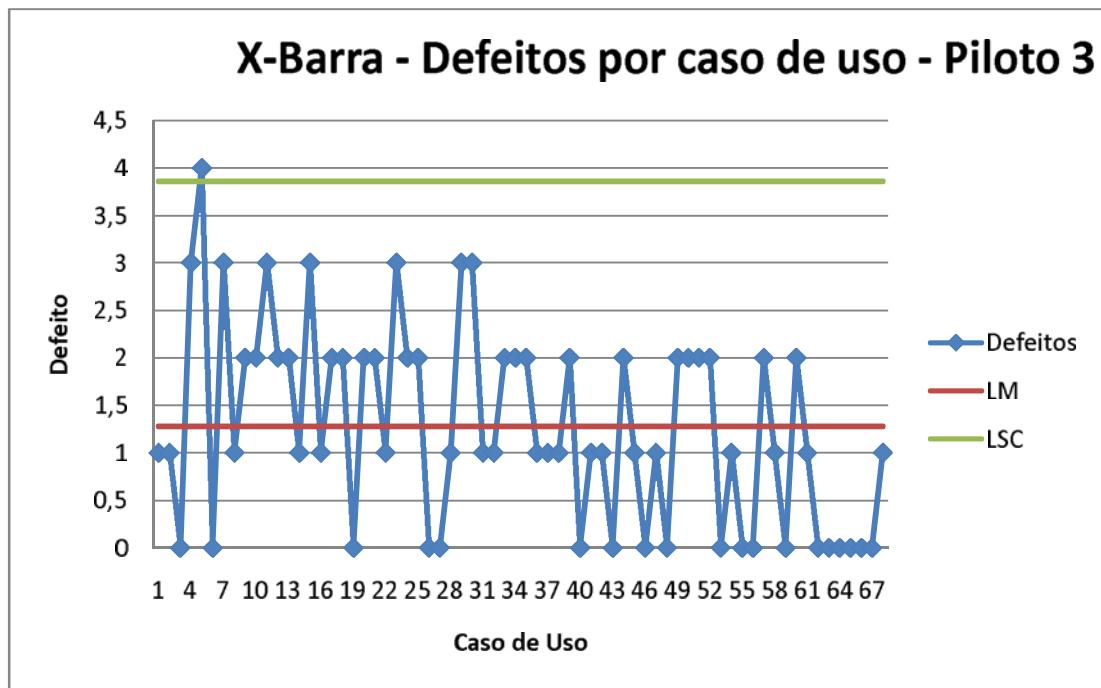
Seguindo o roteiro de investigação, esses pontos foram removidos e os gráficos reconstruídos. Eles estão apresentados nas figuras 28 e 29.

### **Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência



**Figura 28 – Gráfico  $R_m$  para o indicador de defeitos por caso de uso, sem as causas especiais da figura 27**



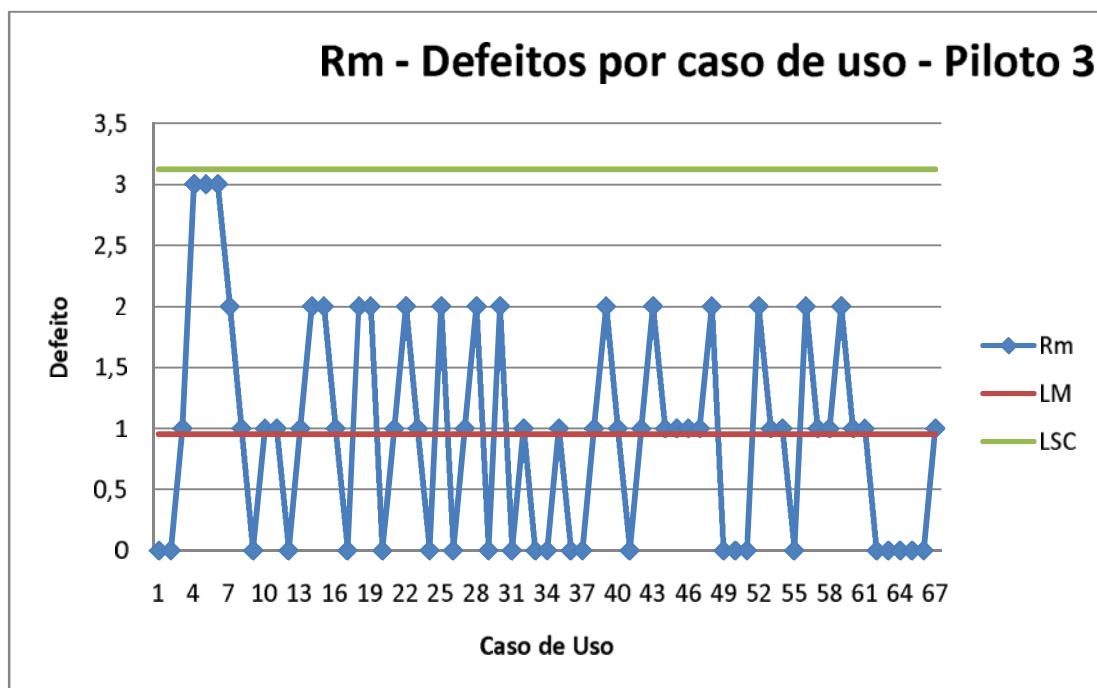
**Figura 29 – Gráfico  $\bar{X}$  para o indicador de defeitos por caso de uso, sem as causas especiais da figura 27**

A figura 29 apresenta uma causa especial, o ponto 5. Ao investigar o porquê dos defeitos encontrados, foram identificados os mesmos problemas

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência**

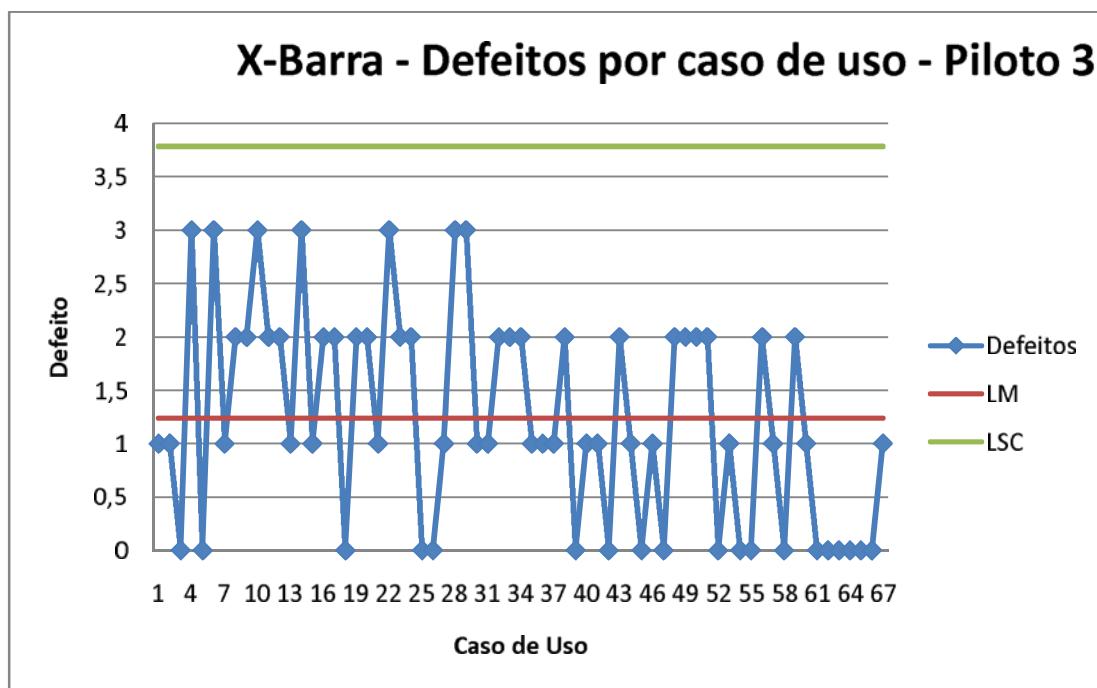
assinalados nas causas especiais da figura 27, ou seja, os requisitos nas funcionalidades de cadastro não estavam bem descritos. Essa causa especial foi removida e o gráfico foi reconstruído, estando ilustrado nas figuras 30 e 31, que exibem o processo estável.



**Figura 30 – Gráfico *Rm* para o indicador de defeitos por caso de uso, sem as causas especiais da figura 29**

**Tomada de decisão em fábrica de software baseada em CEP**

6. Segundo ciclo da pesquisa-ação: Melhoria do modelo de referência



**Figura 31 – Gráfico  $\bar{X}$  para o indicador de defeitos por caso de uso, sem as causas especiais da figura 29**

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 7. Conclusões

#### **7. Conclusões**

Três grandes atividades foram realizadas nesta pesquisa. A primeira delas foi a fundamentação (Capítulo 3) dos conceitos de qualidade, PSM, CEP e fábrica de software. Nessa fundamentação pesquisaram-se artigos que fundamentassem a aplicação dos conceitos de CEP no desenvolvimento de software.

A segunda atividade foi a construção do modelo de referência para aplicação do controle estatístico. Nesse modelo foi utilizado o PSM como guia para selecionarmos os indicadores adequados para cada necessidade de informação que um gerente de fábrica deve dispor para tomar uma decisão de melhoria de processo. Os indicadores foram baseados nos indicadores operacionais para gestão de uma fábrica de software onde foram escolhidos, em reunião com o gerente de fábrica, três indicadores para os projetos projeto-pilotos. Após a escolha dos indicadores, foram selecionados três projetos projeto-pilotos para aplicarmos o modelo de referência.

A terceira atividade, execução da pesquisa-ação, consistiu na aplicação do modelo de referência e na coleta dos dados. Na aplicação do modelo foram realizadas as análises dos gráficos de controle de processo e relacionadas as decisões tomadas para cada indicador de gestão operacional.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 7. Conclusões

#### **7.1. Principais resultados e contribuições**

Os principais resultados e contribuições desta dissertação estão resumidos abaixo:

- ✓ **Seleção dos indicadores:** o PSM se mostrou um modelo didático e simples para fazer o mapeamento da unidade de medida que será utilizada na coleta dos dados e na composição do indicador.
- ✓ **Avaliação do desempenho do processo utilizando gráficos:** após a disseminação de um processo de desenvolvimento, é necessário avaliar se esse processo está sendo bem executado e se ele está atendendo às necessidades. A interpretação dos indicadores de forma gráfica se mostrou didática para os tomadores de decisão. Os gerentes de projetos e da fábrica de software entenderam o comportamento do gráfico de forma simples.
- ✓ **Aplicação nos projetos pilotos:** os projetos-pilotos permitiram avaliar o processo de desenvolvimento atual e os gráficos de controle ajudaram a descobrir os pontos de melhoria de que o processo necessita.
- ✓ **Melhoria de processo:** a forma do controle estatístico de processo apresenta facilita para o engenheiro de processo, a busca por pontos de aperfeiçoamento no processo de desenvolvimento de software. Após encontrar os pontos de melhoria e corrigir o processo, o engenheiro pode avaliar se o efeito desejado foi atendido.

## **Tomada de decisão em fábrica de software baseada em CEP**

### **7. Conclusões**

- ✓ **Recomendações para uso:** nesta pesquisa foram definidos os gráficos de controle antes de fazer a coleta dos dados. Com isso, verificou-se que o gráfico escolhido previamente no modelo de referência não atendia ao que se necessitava. Assim, algumas recomendações ficaram inválidas.

Mesmo com essas contribuições é preciso aplicar o CEP em uma quantidade maior de projetos para se ter uma posição mais assertiva sobre a sua aplicabilidade no desenvolvimento de software.

### **7.2. Limitações da pesquisa**

As principais limitações desta dissertação estão resumidas abaixo:

- ✓ **Indicador de produtividade:** a seleção do indicador de produtividade não permitiu tomar uma decisão de forma conclusiva, apenas apontou que um componente sofreu uma intervenção manual acima do esperado.
- ✓ **Processo de coleta manual:** toda a coleta de dados foi realizada de forma manual, isso acabou acarretando uma demora para a construção do gráfico. O ideal é que a coleta seja realizada de forma automática, com isso o gráfico será construído rapidamente.
- ✓ **Regras de CEP:** nesta pesquisa foi utilizada apenas a regra 1, ou seja, se o ponto está dentro ou fora dos limites especificados. Seria interessante analisar a tendência dos pontos representados no gráfico.

## **Tomada de decisão em fábrica de software baseada em CEP**

---

### 7. Conclusões

#### **7.3. Trabalhos futuros**

Os possíveis trabalhos provenientes desta dissertação estão resumidos abaixo.

- ✓ **Indicadores:** selecionar outros indicadores para gerência de requisitos, baseados não somente no desvio de escopo, mas em técnicas de escrita, para avaliar a qualidade do requisito escrito pelos analistas.
- ✓ **Aplicação do modelo no processo iterativo:** nesta pesquisa, o processo avaliado é um processo com características de cascata tradicional (*Waterfall*). No processo iterativo, os indicadores de desvio de escopo e quantidade de erros poderiam dar possibilidade do gerente de projeto de assumir ações de forma proativa.
- ✓ **Regras de CEP:** nesta pesquisa foi utilizada apenas a regra 1, ou seja, se o ponto está dentro ou fora dos limites especificados. É preciso avaliar se as outras regras de CEP poderiam ser aplicadas ao processo de desenvolvimento de software. Verificar se as tendências podem realmente melhorar a detecção de causas especiais e desvios no processo.

**Tomada de decisão em fábrica de software baseada em CEP**

Referências Bibliográficas

**Referências Bibliográficas**

BURTON, D.; OVER, J. A quantitative approach to software quality management. In: SEI-SEPG CONFERENCE, 2000, Seattle. **Anais**. Seattle 2000. 23 p.

CAMPOS, V. Falconi. **TQC: Controle da Qualidade Total**. 5. ed. Belo Horizonte: Fundação Christiano Ottoni 1992.

CARD, D.; MACIVER, R. **Applying PSM to Enterprise Measurement**. U.S. Army TACOM, 2003. 12 p. Technical Report.

CARD, David N; JONES, Cheryl. Status Report: Practical Software Measurement. **Third Internacional Conference On Quality Software**, p.315, 2003.

CARD, David. Statistical Process Control for Software?. **IEEE Software**, p.95-97, 1994.

COOKE, C.S. **Gestão de Serviços**: Proposição de um método para obtenção de vantagem competitiva através da fidelização do consumidor. 2000. Dissertação (mestrado em Engenharia de Produção) – Escola Politécnica da USP, São Paulo, 2000.

COSTA NETO, Pedro Luiz de Oliveira (Coord.). **Qualidade e competência nas decisões**. São Paulo: Bluscher 2007.

COSTA, Antonio Fernando Branco. EPPRECHT, Eugenio Kahn. CARPINETTI, Luiz Cesar Ribeiro. **Controle Estatístico de Qualidade**. São Paulo: Atlas 2004.

CROSBY, Philip B. **Qualidade é Investimento**. 7. ed. Rio de Janeiro: Livraria José Olympio Editora 1999.

CUSUMANO, Michael. **Japan's Software Factories: A Challenge to US Management**. Oxford University Press: New York, 1991.

CUSUMANO, Michael. The Software Factory: A Historical Interpretation. **IEEE Software**, vol. 6, n. 2, p.23 - 30, mar/abr. 1989.

DASKALANTONAKIS, Michael K. A practical view of software measurement and implementation experiences within Motorola. **IEEE Transactions**, vol. 18, n. 18, nov. 1992.

DAVIS, M.M.; AQUILANO, N.J.; CHASE, R.B. **Fundamentos da Administração da Produção**. 3. ed. Porto Alegre: Bookman 2001.

DEMARCO, T. **Controlling software projects: management measurement and estimation**. Upper Saddle River: Yourdon Press, 1982.

### **Tomada de decisão em fábrica de software baseada em CEP**

#### **Referências Bibliográficas**

DEMING, W. Edwards. On probability as a basis for action. **The American Statistician** 29. Alexandria, nº. 4 p. 146 – 152, 1975

DEMING, W. Edwards. **Qualidade: A Revolução da Administração**. Rio de Janeiro: Marques-Saraiva 1990.

FERNANDES, Agnaldo Aragon. **Fábrica de Software**: implantação e gestão de operações. São Paulo: Atlas 2004.

FLORAC, William A.; CARLETON, Anita. **Measuring the software process** : statistical process control for software process improvement. Reading: Addison Wesley 1999.

FLORAC, William A.; CARLETON, Anita. **Practical Software Measurement** : measuring for process management and improvement. GUIDEBOOK CMU/SEI-97-HB-003, Software Engineering Institute, 1997.

FRANCO, Maria Amélia Santoro. Pedagogia da pesquisa-ação. **Educação e Pesquisa**, vol.31, n.3, p.483-502, dez 2005.

GARVIN, David. **Gerenciamento da Qualidade**: a visão estratégica e competitiva. Rio de Janeiro: Qualitymark, 2002.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas 2002.

GUPTA, Rajesh. Use SPC Techniques in Software Process: Learning of a SW-CMM Level 5 Organization. **SEPG**, p.1-18, 2002.

HARVEY, W.; ROSENBAUN, S. Schlumberger's software improvement program. **IEEE Transactions on Software Engineering**, vol. 20, n. 11, Nov 1994.

HERBSLEB, J. et al. Software quality and the capability maturity model, **Communication of The ACM**, vol. 6, Jun 1997.

HUMPREY, W. S. Characterizing the software process: a maturity framework. **IEEE Software**, vol. 5, n. 2, p. 73-79, Mar 1988.

HUMPREY, W. S. **Managing the software process**. Reading: Addison Wesley, 1989.

IEEE Software Engineering Standards, **Std. 610.12-1990**, p. 47-48. 1990.

JONES, C. Implementing a Successful Measurement Program: Tried and True Practices and Tools. **Cutter IT Journal**, vol. 16, n. 11, p. 12-18, Nov 2003.

JURAN, J.M. **Juran na Liderança pela Qualidade**. 3. ed. São Paulo: Pioneira 1993.

**Tomada de decisão em fábrica de software baseada em CEP**

Referências Bibliográficas

KITCHENHAM, B.; PFLEEGER, S. L. Software quality: the elusive target. **IEEE Software**, vol. 13, n. 1, p. 12-21, Jan 1996.

KOMURU, Mutsumi. Experiences of applying SPC techniques to software development process. In: Proceeding of the 28th international conference on Software engineering, **ICSE**, p. 577-584, 2006.

LANTZY, Mark A. Application of Statistical Process Control to the Software In: Proceedings of the ninth Washington Ada Symposium on Ada, Virginia, **Proceedings**, p. 113-123, 1992.

MCGARRY, John. CARD, David. JONES, Cheryl. LAYMAN, Beth. CLARK, Elizabeth. DEAN, Joseph. HALL, Fred. **Practical Software Measurement**: Objective information for decision makers. Reading: Addison Wesley 2001.

MURDOCH, John. CLARK, Graham, POWELL, Antony, CASELEY, Paul. Measuring Safety: applying PSM to the system safety domain . In: Proceeding of the 8th Australian workshop on Safety critical systems and software, **SCS**, p. 47-55, 2003.

PALANDINI, E. Pacheco. **Qualidade Total na Prática**: implantação e avaliação de sistemas de qualidade. 2. ed. São Paulo: Atlas 1997.

PAULK, M. et al. Capability maturity model: version 1.1. **IEEE Software**, vol.10, n. 4, p. 18-27, Jul 1993.

PFLEEGER, S.L. **Software engineering**: theory and practice. Prentice Hall, 1998.

PRESSMAN, R. S. **Software Engineering** a practitioner's Approach. 5. ed. Londres: McGrawHill 1997.

RAGLAND, B. Measure, Metric or Indicator: What's the Differences? **Crosstalks**, Vol. 8, n. 3, p. 29-30, mar. 1995.

RAMOS, Alberto W. Mantendo o Processo sob Controle. In: ROTONDARO, Roberto G. **Seis Sigma**: Estratégia gerencial para a melhoria de processos, produtos e serviços. São Paulo: Atlas 2002. Cap. 10.

SILVA, Walter Henrique de Farias. SPINOLA, Mauro de Mesquita. ASATO, Regina Yoneko. Utilizando o controle estatístico de processo para avaliar a maturidade do processo de gerência de requisitos. In: Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería Del Conocimiento, 5, Puebla, **JIISIC**, p. 55-58, 2006.

SILVA, Walter Henrique de Farias. SPINOLA, Mauro de Mesquita. ASATO, Regina Yoneko. Utilização do controle estatístico de processo para avaliar a maturidade do processo de gerência de requisitos. **IEEE América Latina**, vol 5, n.5, p. 381-384, 2007.

**Tomada de decisão em fábrica de software baseada em CEP**

---

Referências Bibliográficas

SLACK, Nigel. CHAMBERS, Stuart. JOHNSTON, Robert. **Administração da Produção**. 2. ed. São Paulo: Atlas 2002.

STATZ, Joyce. Practical Software Measurement. In: Proceeding of the 21st international conference on Software engineering, **ICSE**, p. 667-668, 1999.

STEVENSON, William Jr. **Estatística aplicada à administração**. São Paulo: Harbra 1981.

SWANSON, Kent.; MCCOMB, Dave; SMITH, Jill; MCCUBBREY, Don. The Application Software Factory: Applying Total Quality Techniques to Systems Developement. **MIS Quarterly**, vol. 15, n. 4, p. 567-579, Dec 1991.

THIOLLENT, Michel. **Metodologia da pesquisa-ação**. 14. ed. São Paulo: Cortês 2005.

TRIPP, David. Pesquisa-ação: uma introdução metodológica. **Educação e Pesquisa**, vol.31, n.3, p.443-466, dez 2005.

WELLER, Edward F. Pratical Applications of Statistical Process Control. **IEEE Software**, vol. 17, n. 03,p.48-55, maio/junho 2000.

WERKEMA, Maria Cristina. **Ferramentas Estatística Básicas para o Gerenciamento de Processo**. Volume 2. Belo Horizonte: Fundação Christiano Ottoni 1995.

WILD, R. **Concepts for operations management**. Interscience: Chicheter, 1977.