

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO**  
(Bacharelado)

**SOFTWARE DE APOIO AO PROCESSO DE GERÊNCIA DA  
CONFIGURAÇÃO SEGUNDO NORMAS E MODELOS DA  
QUALIDADE**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE  
REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA  
DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA  
COMPUTAÇÃO — BACHARELADO

**EDUARDO ALEXANDRE BARBARESCO**

BLUMENAU, DEZEMBRO/2000

2000/2-21

# **SOFTWARE DE APOIO AO PROCESSO DE GERÊNCIA DA CONFIGURAÇÃO SEGUNDO NORMAS E MODELOS DA QUALIDADE**

**EDUARDO ALEXANDRE BARBARESCO**

ESTE TRABALHO DE CONCLUSÃO DE CURSO, FOI JULGADO ADEQUADO PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

**BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO**

---

Prof. Everaldo Artur Grahl — Orientador na FURB

---

Prof. José Roque Voltolini da Silva — Coordenador do TCC

## **BANCA EXAMINADORA**

---

Prof. Everaldo Artur Grahl

---

Prof. Carlos Eduardo Negrão Bizzotto

---

Prof. Oscar Dalfovo

## **DEDICATÓRIA**

Dedico este trabalho a meu pai, minha mãe pelo amor, incentivo e compreensão durante mais esta fase da minha vida.

## **AGRADECIMENTOS**

Agradeço a Deus, por ter me dado forças em mais um momento decisivo da minha vida.

Ao professor Everaldo Artur Grahl, pela atenção dispensada a minha pessoa durante todo o desenvolvimento deste trabalho.

E a todos que direta ou indiretamente me ajudaram ao longo desta caminhada.

# SUMÁRIO

LISTA DE FIGURAS E TABELAS.....	VII
RESUMO .....	VIII
ABSTRACT .....	VIII
1 INTRODUÇÃO .....	1
1.1 Origem.....	1
1.2 Objetivo .....	2
1.3 Estrutura do texto .....	2
2 GERENCIAMENTO DA CONFIGURAÇÃO.....	4
2.1 Conceitos .....	4
2.2 Linhas básicas(baselines) .....	4
2.3 Itens de configuração de software .....	6
2.4 Atividades.....	6
2.4.1 Identificação dos itens da configuração .....	7
2.4.2 Controle da configuração .....	8
2.4.3 Relato do “status” da configuração .....	9
2.4.4 Auditoria da configuração.....	9
2.5 Ferramentas .....	10
2.6 Importância da atividade em pequenas empresas.....	10
2.7 Pesquisa sobre a utilização do processo de gerência de configuração .....	11
3 NORMAS E MODELOS DA QUALIDADE .....	12
3.1 Norma ISO/IEC 12207 – Processos de ciclo de vida de software .....	12
3.2 Processo de gerência da configuração segundo a norma ISO/IEC 12207 .....	12
3.3 Modelo CMM/SEI.....	14
3.4 Processo de gerência da configuração segundo o modelo CMM/SEI.....	15
3.5 Norma ISO 9000-3 .....	17
3.6 Processo de gerência da configuração segundo a norma ISO 9000-3.....	17
3.7 Norma ISO/IEC 15504/SPICE .....	18
3.8 Processo de gerência da configuração segundo a norma ISO/IEC 15504 .....	18
4 COMPARATIVO ENTRE AS NORMAS E O MODELO.....	21
4.1 Introdução.....	21
4.2 Identificação dos itens da configuração do software.....	21
4.3 Controle da configuração do software.....	22
4.4 Relato de “status” da configuração do software.....	22
4.5 Auditoria da configuração .....	23
4.6 QUADRO COMPARATIVO DO PROCESSO.....	23
4.7 Justificativa para a especificação.....	25
4.8 Justificativa para os relatórios .....	26
5 DESCRIÇÃO DO SOFTWARE.....	27
5.1 Análise Essencial.....	27
5.2 Lista de eventos .....	28
5.3 Diagrama de contexto.....	31
5.4 Diagrama de fluxo de dados .....	32
5.5 Diagrama entidade-relacionamento.....	38
5.6 Principais telas.....	43
5.7 Relatórios.....	47

5.8 Relatório de Situação dos Itens .....	47
5.9 Relatório de Alterações por Versão de Item.....	48
5.10Relatório de Pedidos de Modificação por Versão .....	49
6 CONCLUSÕES .....	50
6.1 Considerações finais.....	50
6.2 Sugestões .....	50
REFERÊNCIAS BIBLIOGRÁFICAS .....	52

## LISTA DE FIGURAS E TABELAS

FIGURA 1 - PROCEDIMENTO DE MODIFICAÇÃO.....	5
FIGURA 2 - PROCESSO DE ALTERAÇÃO SEGUNDO O MODELO CMM/SEI.....	15
FIGURA 3 - DIAGRAMA DE CONTEXTO.....	31
FIGURA 4 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 1 A 5.....	33
FIGURA 5 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 6 A 9.....	34
FIGURA 6 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 10 A 12.....	35
FIGURA 7 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 13 A 15.....	36
FIGURA 8 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 16 A 17.....	37
FIGURA 9 - DIAGRAMA ENTIDADE-RELACIONAMENTO - NÍVEL LÓGICO.....	40
FIGURA 10 - DIAGRAMA ENTIDADE-RELACIONAMENTO - NÍVEL FÍSICO.....	42
FIGURA 11 - CADASTRO DE PROJETOS.....	43
FIGURA 12 - CADASTRO DE FERRAMENTAS.....	43
FIGURA 13 - CADASTRO DE TIPOS DE ITENS.....	44
FIGURA 14 - CADASTRO DE FASES DO CICLO DE VIDA.....	44
FIGURA 15 - CADASTRO DE ANALISADORES.....	45
FIGURA 16 - CADASTRO DE ITENS DE CONFIGURAÇÃO.....	45
FIGURA 17 - CADASTRO DE VERSÕES DOS ITENS.....	46
FIGURA 18 - CADASTRO DOS PEDIDOS DE MODIFICAÇÃO DOS ITENS.....	46
FIGURA 19 - CADASTRO DE ALTERAÇÕES DE VERSÃO.....	47
FIGURA 21 - RELATÓRIO DAS ALTERAÇÕES POR VERSÃO.....	48
FIGURA 22 - RELATÓRIO DOS PEDIDOS DE MODIFICAÇÃO POR VERSÃO.....	49
TABELA 1 - COMPARATIVO DO PROCESSO DE GERÊNCIA DE CONFIGURAÇÃO.....	24
TABELA 2 - JUSTIFICATIVA PARA AS ENTIDADES DO MER.....	25
TABELA 3 - JUSTIFICATIVA PARA AS RELATÓRIOS.....	26

## **RESUMO**

Este trabalho trata do processo de gerência da configuração de software que é um conjunto de atividades definidas para administrar mudanças no ciclo de vida de um software. Um software de apoio a este processo foi desenvolvido a partir do estudo comparativo entre o processo de gerência da configuração existente nas normas de qualidade ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504(SPICE) e no modelo CMM/SEI. O software construído permite o cadastro, controle e consultas sobre os itens de configuração e suas versões produzidas ao longo de um projeto de software, bem como permite a geração de relatórios gerenciais.

## **ABSTRACT**

This work care for the software configuration management process that is an activities set defined to manage changes in life cycle of a software. A support software to this process was going developed from the comparative study between management existing configuration process in norm of quality ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504(SPICE) and in the model CMM/SEI. The built software allows the register, control and consultations on the configuration items and her versions produced along a software project, as well as it allows the managerial reports generation.



# 1 INTRODUÇÃO

Este capítulo apresenta a origem, objetivo e organização do trabalho.

## 1.1 ORIGEM

A competitividade no mercado de informática, requer que cada vez mais o processo de desenvolvimento de software possa ser feito de forma rápida e controlada. Estas tentativas têm sido aprimoradas e adaptadas, com o objetivo de otimizar todos os processos de tal forma que se possa garantir a qualidade de um produto final.

Contudo, existem várias organizações atualmente, que apresentam problemas de concepção de seus produtos. Sejam problemas na ordem de prazos a serem cumpridos, estimativas irreais, ou erros no seu próprio cronograma de trabalho.

Normalmente neste tipo de organização além da falta de qualquer tipo de mecanismo que permita aprender com a experiência de projetos anteriores, faltam ainda habilidades para efetuar um controle efetivo do processo em questão. Por isso as modificações apresentam elevado grau de riscos e inevitavelmente estes erros são repassados a versões posteriores, o que acarreta em mais problemas futuros. Estes problemas em sua grande maioria fazem com que processos vitais em seu desenvolvimento não sejam realizados como testes e revisões, e sendo assim o produto demonstrado, é frequentemente repleto de erros, documentação praticamente inexistente e total falta de um histórico de desenvolvimento([BER1999]).

Observa-se portanto, que as modificações são praticamente inevitáveis em uma etapa de desenvolvimento e que diversas são as origens que atualmente fazem com que elas aconteçam: os avanços tecnológicos, novas maneiras de implementar facilidades, evolução de um ambiente operacional e em grande parte devido a correção de defeitos.

Um dos processos importantes é o de gerência da configuração de software, que tem como finalidade garantir a integridade dos produtos de software, através do controle efetivo do processo de desenvolvimento.

A configuração do software compreende todos os itens de informação que são obtidos durante o seu ciclo de vida. O estabelecimento e a manutenção da integridade desses itens de informação constituem o processo de gerenciamento da configuração de software ([PRE1995]). O termo “item de configuração” é usado para definir cada uma das partes em que se pode decompor a entidade de software sob controle. Um item de configuração pode também ser identificado como todo e qualquer ente passível de manipulação, ou seja, que possa ser unicamente identificado e gerenciado.

De uma forma geral, [MAR1998] relata que existem quatro atividades básicas a serem cumpridas pelo processo de gerência da configuração: identificação da configuração, controle da configuração, administração de estados e auditoria da configuração. Cabe ainda ressaltar que essas atividades são muito trabalhosas e difíceis de serem aplicadas, e não raramente necessitam de um conhecimento prévio sobre o assunto ou um grau de experiência adquirida anteriormente.

Com isto pode-se aplicar o uso de ferramentas de gerenciamento, o que em alguns casos, pode comprometer o atual estágio de desenvolvimento visto a necessidade de se adquirir alguma familiaridade com a ferramenta, bem como um treinamento à toda equipe envolvida no processo.

Atualmente existem no mercado algumas opções de software que podem atender a esta demanda. Um exemplo é o Starteam [CHO2000], distribuído no Brasil pela Choose Technologies que permite criar toda a colaboração técnica, dá suporte a Gerência de Configuração, e também a atividades como a Gerência de Projetos.

Existem várias normas e modelos que possuem atividades previstas para o gerenciamento da configuração. Para este trabalho pretende-se utilizar: o CMM/SEI (Modelo de Capacidade e Maturidade), a NBR ISO/IEC 12207, a NBR ISO 9000-3 e a ISO/IEC 15504/SPICE.

Conforme [SAN1994], o princípio do modelo CMM/SEI, criado em 1986, é a de que aumentando a qualidade dos processos que compõem a criação do software, aumenta a qualidade do software como um todo. Cabe então, ao modelo poder prover todas as soluções que satisfaçam as necessidades que existem em cada etapa.

E seguindo [ABN1997], a ISO/IEC 12207 é uma norma que procura prover uma estrutura que cobre todo o ciclo de vida do software, desde a concepção de idéias até a descontinuação do mesmo, e consiste dos processos de aquisição e fornecimento de produtos e serviços de software.

Já a norma ISO 9000-3 trás diretrizes para a aplicação da NBR 19001 ao desenvolvimento, fornecimento e manutenção de software. Esta NBR 9000-3 aborda basicamente situações em que um "software" específico é desenvolvido como parte de um contrato, de acordo com as especificações do comprador.

E por fim a ISO/IEC 15504/SPICE que é uma norma em elaboração conjunta pela ISO(International Organization for Standardization) e pelo IEC(International Electrotechnical Commission), constitui-se de um padrão para a avaliação do processo de software, visando determinar a capacitação de uma organização.

Através de um mapeamento realizado destas normas e o modelo de qualidade, pretende-se chegar à um padrão comum no que se refere as atividades que devem ser adotadas neste tipo de processo, tendo como meta o desenvolvimento de uma ferramenta de auxílio á pratica da gerência da configuração.

## **1.2 OBJETIVO**

O objetivo deste trabalho é especificar e implementar um software, que atenda a um conjunto de atividades que auxilie o processo de gerência da configuração, segundo as normas de qualidade ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504/SPICE e o modelo CMM.

## **1.3 ESTRUTURA DO TEXTO**

A seguir é apresentada uma síntese dos próximos capítulos constantes desse trabalho.

O segundo capítulo apresenta conceitos, atividades, ferramentas sobre gerência da configuração de software, importância da atividade na pequena empresa, e também uma pesquisa sobre a utilização da atividade de gerenciamento da configuração.

O terceiro capítulo apresenta as quatro normas e o modelo de qualidade estudados e também demonstra detalhadamente o processo de gerência da configuração de cada alternativa.

O quarto capítulo mostra um comparativo entre as normas e o modelo de qualidade no que diz respeito ao processo de gerência da configuração.

O quinto capítulo apresenta a especificação e implementação do software de apoio a gerência da configuração de sistemas.

O sexto capítulo apresenta as conclusões deste trabalho e as sugestões para que o mesmo possa ter continuidade e seja melhorado.

## 2 GERENCIAMENTO DA CONFIGURAÇÃO

Este capítulo oferece uma visão geral sobre o processo de gerência da configuração, onde serão enfocados assuntos como conceitos, atividades, ferramentas, importância da atividade em pequenas empresas, e também uma pesquisa realizada sobre a utilização da atividade.

### 2.1 CONCEITOS

De acordo com [MAR1998], "a gerência da configuração de software é o processo cujo objetivo é a identificação de configurações do software em pontos discretos no tempo e o sistemático controle das mudanças ocorridas em uma determinada configuração, com o propósito de manter a integridade do software através do seu ciclo de vida".

Já segundo [BER1999], "o gerenciamento da configuração é um importante elemento da garantia da qualidade de software e sua principal função é administrar as mudanças que ocorrem na configuração do software durante o ciclo de vida com o propósito de manter a integridade e rastreabilidade do software".

No decorrer do processo de Engenharia de Software, o número de itens de configuração cresce muito rapidamente, pois sabe-se que cada Item de Configuração gera outros e assim por diante. O que acontece é que surgem então inúmeras variáveis durante estas etapas, mudanças que podem ocorrer durante todo o ciclo de produção do software.

O processo de gerência da configuração visa assim, através de um conjunto de atividades bem definidos estabelecer uma maneira para administrar as mudanças em todo o ciclo de vida do projeto de software, além de ser considerado uma atividade de garantia de qualidade do software, sua implantação requer alguns cuidados antes de ser devidamente aplicado, pois deve haver um profundo conhecimento do contexto na qual a organização está inserida([SCO2000]).

### 2.2 LINHAS BÁSICAS(BASELINES)

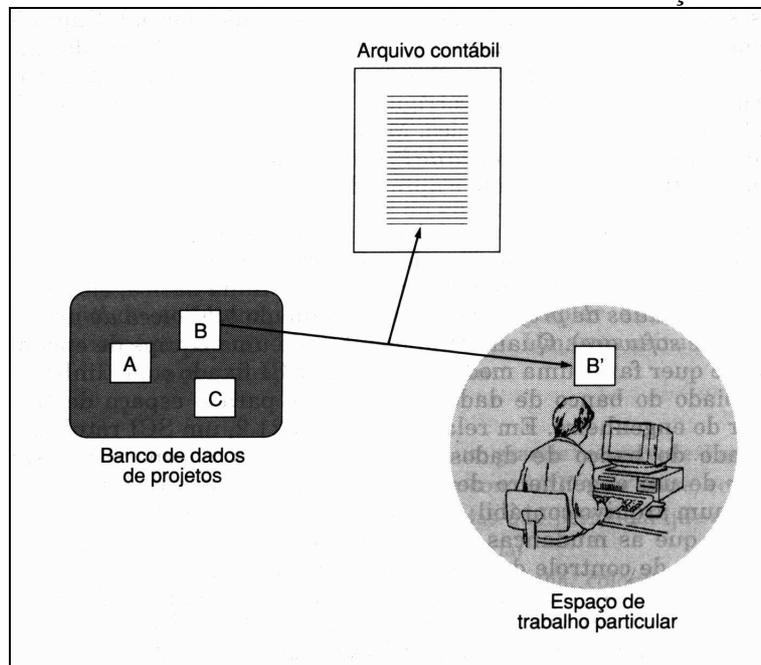
No processo de Engenharia de Software, segundo [PRE1995], considera-se uma linha básica um marco de referência no desenvolvimento de software. A entrega de um item de configuração ou vários, após realizada uma revisão formal por uma equipe de projeto, geram uma linha básica, ou seja a partir do momento que todas as partes da especificação forem revisadas, corrigidas e depois aprovadas, qualquer modificação que se faça necessária deverá de ser avaliada e aprovada.

Conforme ainda relata [MAR1998], essas configurações-base identificadas aos terminos das fases do desenvolvimento, servem como ponto de partida ou então, como referência para a próxima etapa de desenvolvimento, ou seja, o término de uma etapa de desenvolvimento juntamente com as alterações efetuadas nos itens de configuração da configuração-base atual constituem a próxima configuração-base do software.

Segundo [PRE1995], ocorrendo esta etapa todo e qualquer item de configuração transformado em linha básica é depositado num banco de dados do projeto (também chamado de biblioteca de projetos ou repositório de software). Conforme demonstrado na figura 1, o membro de uma equipe de engenharia de software que quiser realizar alguma modificação em um item de configuração fixado como linha básica, deverá copiar o mesmo para seu ambiente particular de desenvolvimento afim de realizar suas mudanças que se fizerem necessárias, registrar suas atividades em um arquivo de contabilização e só após a sua mudança devidamente aprovada e implementada, será usada para atualizar o item de configuração no repositório de software de forma definitiva.

Convém salientar que na maioria dos casos o item de configuração fixado como linha básica utilizado será bloqueado, a fim de que nenhum membro de uma equipe possa utilizá-lo até que a mudança a ser realizada seja devidamente revista e aprovada.

FIGURA 1 - PROCEDIMENTO DE MODIFICAÇÃO.



Fonte: [PRE1995].

## 2.3 ITENS DE CONFIGURAÇÃO DE SOFTWARE

Conforme [SAN1994], os itens de configuração podem ser descritos como uma coleção de elementos de software tratados como uma unidade, com o propósito de gerenciamento da configuração do software.

Uma outra definição de itens de configuração é a apresentada por [PRE1995], que considera como uma informação que é criada ao longo do processo de engenharia de software, e pode ser considerado uma única sessão de uma especificação, um conjunto de testes, um procedimento em pascal ou o próprio manual do usuário.

Além de muitos dos itens de configuração, muitas vezes algumas organizações de software também procuram catalogar suas ferramentas utilizadas para composição de seu projeto como ferramentas cases, compiladores e editores, como partes da configuração do software. Considerando que todas essas ferramentas que foram utilizadas no projeto foram usadas para criar partes do projeto como código fonte, dados e documentação, elas devem estar à disposição quando precisarem ser feitas mudanças na configuração.

Ainda que o surgimento de problemas possa ocorrer no desenvolvimento do software, poderá ocorrer que uma determinada versão de uma ferramenta poderá gerar um resultado atípico diferente da versão original. Por isso em um processo de gerência da configuração abrangente, as ferramentas utilizadas também podem e devem ser arquivadas e fixadas como linha básica.

Segundo uma abordagem ainda demonstrada por [PRE1995], os itens de configuração de software podem ser organizados em conjunto a fim de formar o que pode chamar-se de objetos de configuração, que são arquivados também no banco de dados geral do projeto. Cabe salientar que um objeto de configuração tem um nome e atributos e sendo assim pode estar relacionado a outros objetos que se fizer necessário, com isto o projetista pode facilmente determinar quais os objetos (e conseqüentemente os itens) que poderiam ser afetados em uma mudança posterior.

## 2.4 ATIVIDADES

Da perspectiva da administração do projeto, o processo de gerenciamento da configuração de software deve evoluir juntamente com o produto identificando, controlando, gravando e reportando as mudanças na configuração do software. Pelo lado do desenvolvedor cabe facilitar o desenvolvimento e a implementação efetiva destas modificações. Em função disso, algumas entre várias questões complexas devem ser levantadas:

- como uma organização identifica e administra as muitas versões existentes de um programa?
- como uma organização controla as mudanças realizadas no software após a entrega ao cliente?
- como podemos garantir que as mudanças foram efetivamente realizadas?

- qual o artifício usado por uma equipe técnica para manter todos os membros avisados das mudanças realizadas?

Devido a estas questões [BER1999], [PRE1995] e [MAR1998] estabelecem que o processo de gerência da configuração é composto por um conjunto de quatro atividades básicas que devem ser cumpridas pelo processo, que são:

1. Identificação dos itens da configuração;
2. Controle da configuração;
3. Relato do “status” da configuração;
4. Auditoria da configuração.

Cabe salientar que dentre estas atividades, existe ainda a presença do Plano de Gerência da Configuração aplicada ao processo. Apesar de ser considerada uma estratégia de abordagem inicial, e não uma atividade propriamente, sua importância é vital, pois através dele é que será possível definir prazos, atividades e responsabilidades, servindo assim como um apoio documental ao processo.

## **2.4.1 IDENTIFICAÇÃO DOS ITENS DA CONFIGURAÇÃO**

Segundo [MAR1998], a atividade de identificação da configuração é o processo que permite identificar cada configuração-base durante o ciclo de vida do software, assim como todos os itens que compõem uma configuração.

As tarefas presentes nesta etapa visam primeiramente levantar de uma forma abrangente todos os itens da configuração, organizando-os de uma forma hierárquica obtendo assim uma visão estrutural do sistema.

Em segundo procura-se selecionar todos os itens identificados anteriormente, adotando-se alguns critérios previamente discutidos, como múltiplo uso, segurança, reuso, interface e manutenção.

Na terceira etapa objetiva-se especificar como cada item da configuração se relaciona com outros itens, para isso [BER1999] define cinco relacionamentos que devem ser observados: Equivalência, Dependência, Derivação, Sucessão e Variante. Através deste tipo de controle, pode-se rastrear todas as relações entre os itens da configuração e assim mantendo a integridade das informações.

Na quarta etapa é necessário que se nomeie todas as coleções de objetos recentemente criados, atribuindo nomes únicos aos componentes, a fim de poderem ser esquematizados de uma forma estrutural. Sabe-se que cada objeto possui um conjunto de características, como uma descrição, um nome e um tipo que possa definir se o mesmo é um documento, um dado ou um programa.

E por fim na última etapa, é realizado o planejamento de linhas de referência, que consiste em estabelecer marcos que caracterizam a entrega de um ou mais itens da

configuração. Comumente estas linhas são estabelecidas ao término de cada fase do ciclo de vida do software e tem por finalidade proporcionar um melhor controle sobre as mudanças.

No que se refere ainda a identificação do objeto de configuração também deve ser levadas em consideração relacionamentos que por ventura possam existir entre os objetos, como por exemplo, um diagrama de Entidade Relacionamento que faz parte de um modelo de dados. Com isto cria-se uma hierarquia de itens, e seus inter-relacionamentos podem ser descritos através de uma linguagem específica, que demonstra as interdependências entre os objetos de configuração e possibilita a construção de uma versão automaticamente .

Através deste método, [PRE1995] conclui que é possível se criar o que é chamado de Gráfico de Evolução que descreve minuciosamente um histórico de todas as alterações que os objetos sofrerão ao longo do ciclo de desenvolvimento. Particularidades existentes como pequenas mudanças de versão (releases), uma grande atualização do software ou até mesmo a descontinuação do projeto são facilmente descritos.

## **2.4.2 CONTROLE DA CONFIGURAÇÃO**

Genericamente o controle da configuração consiste na avaliação, coordenação e na decisão sobre a implantação ou não de uma mudança proposta num item da configuração.

Conforme [BER1999], o controle da configuração é dividido em dois controles básicos que são: o controle de versões e o controle de mudanças.

Genericamente o controle de versões propõe uma combinação de práticas e ferramentas que possibilitam administrar diversas versões dos mais variados objetos de configuração que são concebidos durante o processo de engenharia de software. A importância deste controle é imprescindível pois, é através dele que pode-se obter configurações alternadas do software, em diferentes etapas distintas do projeto e durante todo o seu ciclo de vida.

Para [BER1999], no controle de versão hoje existem inúmeras metodologias que podem ser postas em prática, o que varia essencialmente é a aplicação de atributos diferentes em cada iniciativa, sejam eles inicialmente usados de forma numérica ou na sua forma mais moderna com uso de gerenciamento de linhas básicas, “congelando” a versão corrente do sistema, evitando transtornos e mudanças descontroladamente.

Quanto ao o controle de mudanças, sabe-se que ele por si só evidencia o fato de que deve haver comumente em um projeto, um grande esforço para que as mudanças não sejam realizadas de forma desenfreada. Deve haver portanto, segundo [MAR1998], um conjunto bem definido e documentado de procedimentos e atividades para o encaminhamento, registro e implementação de modificações do produto.

Com isto torna-se essencial a criação de um pedido formal de mudança na qual será submetido e avaliado por um grupo ou perito técnico que revelará possíveis problemas sejam efeitos colaterais, impacto sobre outros itens do projeto como um todo ou também o custo para se implantar a devida mudança.

Todos os resultados que se fizerem necessários serão expostos através de um relatório de mudança aplicado por um grupo ou uma autoridade competente que tomará todas as decisões finais.

Após todos os procedimentos realizados, é gerada uma ordem de mudança de engenharia, que descreverá todas as modificações a serem postas em prática, as restrições que devem ser respeitadas e os critérios de revisão e auditoria.

### **2.4.3 RELATO DO “STATUS” DA CONFIGURAÇÃO**

A produção do relatório de "status" pode ser resumida como um relatório de contabilização documentando exatamente de todo o ocorrido até o momento, e pode ser usados também para análise e melhoria do processo de desenvolvimento de software.

Através do relatório de “status” da configuração, [BER1999] demonstra que é possível identificar o autor da modificação proposta, o tipo da modificação e o momento da modificação efetuada, além de recuperar um histórico das modificações nos itens de configuração.

Todo o volume de documentação pertencente ao relato de status normalmente é disponibilizado em um banco de dados on-line, para que sirva como um importante instrumento dos desenvolvedores, além é claro servir como um meio de informação e comunicação para todos os envolvidos no processo.

### **2.4.4 AUDITORIA DA CONFIGURAÇÃO**

Segundo [SCO2000], a atividade de auditoria da configuração de software vem garantir se cada item da configuração satisfaz as características físicas e funcionais requeridas após todas as mudanças realizadas no software.

Desta forma, apesar de todos os controles exercidos até o momento da geração da ordem de mudança, é perfeitamente admissível que alguma mudança possa não ter sido devidamente realizada, por isso é indispensável, que dois procedimentos sejam executados([PRE1995]):

- 1) Revisões técnicas formais;
- 2) A auditoria da configuração.

Na revisão técnica formal, procura-se focalizar a exatidão técnica do objeto de configuração modificado, e os responsáveis pelo procedimento procuram ainda avaliar potenciais riscos que possam existir, omissões e consistências.

No processo de auditoria, é realizada uma complementação à etapa anterior de revisão, tendo como foco avaliar características que não são consideradas durante o processo anterior, como questões que envolvem exatidão técnica na revisão e se os padrões de engenharia de software foram adequadamente seguidos.

De forma resumida nesta atividade, a ação específica a ser tomada é uma conferência completa de todos os itens que foram alterados ao longo do desenvolvimento, perante os relatórios de modificações, o que garantirá que as passagens de uma configuração à outra não descaracterizará o produto.

## 2.5 FERRAMENTAS

Sabe-se que no processo de gerência da configuração, estão envolvidas uma série de atividades trabalhosas que são realizadas, por isso para que possam de ser realizados mais facilmente e eficientemente possível é altamente recomendado o apoio automatizado de ferramentas para os procedimentos executados.

Segundo [BER1999], existem diversas ferramentas de apoio ao gerenciamento da configuração de software, como o RCS(Revision Control System), o CCC(Change and Configuration Control), o Adele, o DSEE(Domain Software Engineering Environment) e o Sherpa DMS(Design Management System), porém estas ferramentas não oferecem um suporte integral a todas as atividades de gerenciamento da configuração de software.

Contudo [BER1999], ressalta que as ferramentas apresentadas em sua grande maioria não são de fácil acesso para a maioria das empresas que delas necessitam, pois também deve ser considerado que suas potencialidades só podem ser bem aproveitadas por usuários que estão bem treinados na ferramenta.

## 2.6 IMPORTÂNCIA DA ATIVIDADE EM PEQUENAS EMPRESAS

O contexto atual apresentado por [MAR1998], mostra que a realidade dos desenvolvedores nacionais de software é bem caótico, visto que boa parte do que é produzido em nosso país é realizado por empresas de pequeno porte com no máximo 10 funcionários.

Estas empresas lutam comumente com problemas como escassez de recursos e as deficiências tecnológicas para manter seus produtos no mercado. O processo de desenvolvimento de software é considerado um processo crítico e sem fases bem definidas.

Todas as especificações, projeto e codificação são realizados de maneira bem confusa, os testes realizados não são executados de uma maneira racional e sistemática. A documentação é bem deficiente e sem qualquer processo de atualização, com isto o produto posto em circulação em sua grande maioria é composta por inúmeros defeitos e sempre em constante processo de modificação.

A experiência relatada em [MAR1998], sobre a utilização de gerência da configuração na empresa DIGIARTE, aplicada ao produto gerenteMaster, evidencia o fato de que é perfeitamente possível estabelecer práticas de gerência da configuração em pequenos desenvolvedores de software, conseguindo assim não só melhorar todo o processo de desenvolvimento como também introduzir outras práticas de engenharia de software importantes para os produtos e processos de software de melhor qualidade.

É importante ressaltar que neste caso, foi realizado primeiramente através de várias reuniões na organização, uma coleta de várias informações e dados que podiam ser úteis em uma etapa posterior que foi a de elaborar um plano de gerência da configuração. A partir daí se originou um processo interativo, onde sucessivamente foi refinado o plano até se atingir uma versão final, sendo que esta abordagem propiciou dois benefícios: por um lado todos os integrantes da empresa puderam absorver melhor os conceitos sobre o processo e por outro houve um entendimento das reais necessidades da empresa.

Como resultados obtidos evidenciou-se que após 6 meses de aplicação disciplinada do processo, conseguiu-se as seguintes melhorias: evolução do produto mais racional, maior rapidez em incorporações de novas funcionalidades, capacidade de condução de linhas paralelas de evolução para testar novas idéias, capacidade de atender a pedidos de modificação de clientes com versões mais antigas, redução de custos de modificações, assim como também uma substancial melhora em toda parte de documentação do produto.

Para se obter resultados á contento ainda segundo [MAR1998], é indiscutível que se procure seguir uma estratégia de abordagem para a implantação do processo de gerência da configuração, havendo antes de tudo um comprometimento da hierarquia gerencial da empresa, e também uma aceitação pelo pessoal técnico.

## **2.7 PESQUISA SOBRE A UTILIZAÇÃO DO PROCESSO DE GERÊNCIA DE CONFIGURAÇÃO**

Recentemente, a Fundação Centro Tecnológico para Informática – CTI, com apoio da Secretaria de Política de Informática e Automação, realizou uma pesquisa inédita de abrangência nacional sobre a utilização das práticas de gerência da configuração de software, que teve como seu principal objetivo levantar dados das empresas brasileiras de software quanto à utilização de técnicas sobre gerência da configuração de software em seus processos de desenvolvimento ([CTI1999]).

Nesta pesquisa foram levantadas através de um questionário contemplando 25 questões específicas, itens como importância dos desenvolvimento de software, caracterização da gerência da configuração de software - práticas utilizadas nas empresas, problemas relativos à gerência dos produtos de software; grau de conhecimento técnico, caracterização da qualidade dos produtos desenvolvidos e levantamento das necessidades das empresas para a melhoria nos processos de desenvolvimento de software.

Revelou-se assim através da pesquisa inúmeros pontos relevantes, como por exemplo a inexistência da adoção de qualquer tipo ferramenta de gerência da configuração por 225 empresas de um universo de 351 empresas pesquisadas, o que representa 64,1% de todas as organizações questionadas.

Por isso um dos objetivos imediatos a serem conquistados através do gerenciamento da configuração é poder propiciar portanto uma melhoria no que diz respeito a qualidade dos produtos de software desenvolvidos, fazendo com que as mudanças possam ser monitoradas, ajustadas e relatadas durante a etapa de desenvolvimento.

## **3 NORMAS E MODELOS DA QUALIDADE**

Este capítulo trata da norma ISO/IEC 12207, do modelo CMM/SEI, da norma ISO 9000-3 e da futura norma ISO/IEC 15504(SPICE). Será apresentada uma breve descrição de cada alternativa, assim como uma explicação do processo de gerência da configuração previsto em cada alternativa.

### **3.1 NORMA ISO/IEC 12207 – PROCESSOS DE CICLO DE VIDA DE SOFTWARE**

Conforme [ABN1997] a norma NBR ISO/IEC 12207 – Processos de Ciclo de Vida do Software tem como princípio fundamental o estabelecimento de uma estrutura comum para os processos de ciclo de vida do software, para ser utilizada como referência. Além disso a norma contempla que o desenvolvimento e a manutenção devem ser conduzidos da mesma forma que a disciplina de engenharia de software.

A estrutura da norma cobre o ciclo de vida do software desde a concepção de idéias até a descontinuação do software, e consiste dos processos de aquisição e fornecimento de produtos e serviços de software. Adicionalmente, a estrutura provê o controle e a melhoria destes processos.

De uma forma geral, esta norma forma um conjunto bem abrangente, com isto uma organização pode dependendo de seu objetivo adaptar a norma, selecionando um subconjunto apropriado para satisfazê-lo, na forma de um produto ou projeto específico. Cabe salientar que ela apresenta suas características voltadas para a utilização em um software específico, embutido ou que apenas faça parte de um sistema de forma integrada.

Maiores informações sobre esta norma podem ser vistas em [ABN1997].

### **3.2 PROCESSO DE GERÊNCIA DA CONFIGURAÇÃO SEGUNDO A NORMA ISO/IEC 12207**

A norma ISO/IEC 12207, agrupa as atividades e tarefas que podem ser executadas durante o ciclo de vida do software em processos fundamentais, processos de apoio e processos organizacionais.

Dentre os processos de apoio existentes, a [ABN1997] descreve o processo de gerência da configuração como um processo de aplicação de procedimentos administrativos e técnicos, por todo o ciclo de vida de software, destinado a:

- identificar e definir os itens de software em um sistema, e estabelecer suas linhas básicas(baseline);
- controlar as modificações e liberações dos itens;

- registrar e apresentar a situação dos itens e dos pedidos de modificação;
- garantir a completeza, a consistência e a correção dos itens;
- controlar o armazenamento, a manipulação e a distribuição dos itens.

Sendo assim neste processo haverá então a execução das seguintes atividades:

- a) **Implementação do Processo** – Nesta atividade um plano de gerência da configuração deve ser desenvolvido. O plano deverá descrever: as atividades da gerência da configuração; procedimentos e cronograma para executar estas atividades; as organizações responsáveis pela execução destas atividades; e seu relacionamento com outras organizações, como por exemplo a de desenvolvimento ou manutenção de software. O plano deve ser documentado e implementado. Neste aspecto o plano poderá fazer parte do plano de gerência da configuração do sistema.
- b) **Identificação da Configuração** – Neste item, uma sistemática para o projeto deve ser estabelecida para a identificação dos itens de software e suas versões a serem controladas. Para cada item de software e suas versões deve ser identificado o seguinte: a documentação que estabelece a linha básica (baseline); as referências de versão; e outros detalhes de identificação.
- c) **Controle da Configuração** – Aqui deve ser executado as seguintes diretrizes: identificação e registro dos pedidos de alteração; análise e avaliação das alterações; aprovação ou rejeição do pedido; e implementação, verificação e liberação do item de software modificado. Devem existir registros de auditoria, de tal forma que, para cada modificação, a sua razão e a sua autorização possam ser rastreadas. Deve ser realizado controle e auditoria de todos os acessos aos itens de software controlados que tratam de funções críticas de proteção ou segurança.
- d) **Relato da situação da configuração** – Nesta etapa devem ser preparados registros de gerenciamento e relatórios de situação que mostrem a situação e o histórico dos itens de software controlados, incluindo a linha básica (baseline). Os relatórios de situação deveriam incluir o número de alterações em um projeto, as últimas versões do item de software, identificadores de liberação, a quantidade de liberações e as comparações entre elas.
- e) **Avaliação da configuração** – Deve ser determinado e garantido o seguinte: a completeza funcional dos itens de software em relação aos seus requisitos e a completeza física dos itens de software (ou seja, se seu projeto e código refletem uma descrição técnica atualizada).
- f) **Gerência de liberação e distribuição** – Nesta última etapa a liberação e distribuição de produtos de software e documentação devem ser formalmente controladas. Cópias matrizes do código e da documentação devem ser mantidas durante a vida do produto de software. O código e a documentação que contenham funções críticas de proteção ou segurança devem ser manipulados, armazenados,

empacotados e distribuídos de acordo com as políticas das organizações envolvidas.

### 3.3 MODELO CMM/SEI

O modelo CMM/SEI teve sua origem em meados de 1986, sendo desenvolvido pelo SEI(Software Engineering Institute) da Carnegie Mellon University, como uma tentativa de classificar os processos de desenvolvimento em níveis.

Segundo [SAN94], este modelo é basicamente uma estrutura que descreve os elementos de um processo eficiente de software e um caminho evolucionário que aumenta a maturidade dos processos nas organizações de software.

Neste modelo, o processo de desenvolvimento do software é classificada em cinco níveis crescentes de maturidade que são: inicial, repetitivo, definido, gerenciado e otimizado. Cada nível caracteriza uma camada para a melhoria contínua do processo.

**Inicial** - O processo de desenvolvimento de software é caracterizado como uma atividade, existe a ausência de instrumentos gerenciais mínimos, o que provoca um planejamento não efetivo (muitas vezes caótico), poucos processos são definidos e o sucesso depende de esforços individuais.

**Repetitivo** - Neste nível são estabelecidos processos básicos de administração de projetos como critérios de custo, cronogramas e funcionalidades. Assim mesmo esta caracterizado porque o planejamento e o acompanhamento do projeto são estáveis, e sucessos anteriores podem ser repetidos.

**Definido** - No nível definido as atividades de gerência e engenharia estão documentadas, padronizadas e integradas dentro de um processo padrão de desenvolvimento de software para a organização, de maneira tal que formam um processo único, coerente e completo. Todos os projetos utilizam a versão aprovada do padrão para executar processos de desenvolvimento e a manutenção de software.

**Gerenciado** - No nível gerenciado, a organização estabelece metas quantitativas de qualidade para produtos e processos de software. São medidas a qualidade e produtividade para atividades importantes do processo, abrangendo todos os projetos, e tais medidas são parte de um programa organizacional de medições.

**Otimizado** - No nível otimizado, a organização inteira está voltada para o processo de melhoria contínua. Existem meios para identificar fraquezas e fortalecer o processo de forma pró-ativa, de acordo com a meta de prevenir a ocorrência de defeitos (existe uma retroalimentação quantitativa - feedback dos processos), assim como um manejo eficiente de tecnologias e idéias novas.

Maiores informações sobre este modelo podem ser vistas em [SAN1994] e [FIO1998].

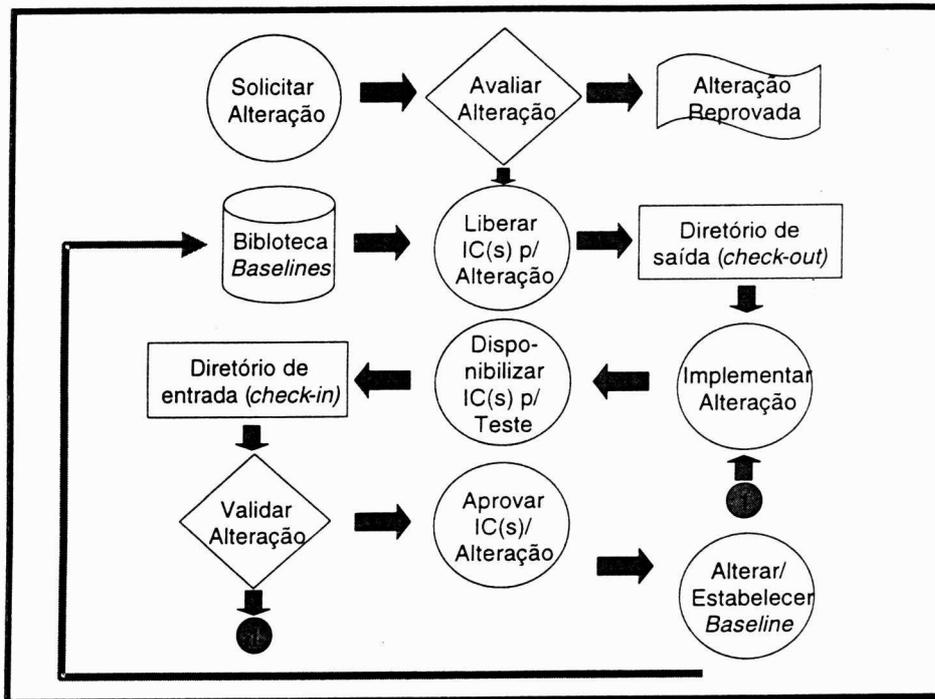
### 3.4 PROCESSO DE GERÊNCIA DA CONFIGURAÇÃO SEGUNDO O MODELO CMM/SEI

Conforme [FIO1998], o modelo CMM suporta e define os conceitos de gerência da configuração da mesma forma como apresentado anteriormente na seção 2 – Processo de Gerência da Configuração, e está situado no nível 2: repetitivo, que tem como focos aspectos relacionados ao estabelecimento de controles básicos na administração de projetos.

Seguindo o modelo, todo o planejamento das atividades de gerência da configuração de software, itens da configuração e *baselines* estão definidos em um plano de gerência da configuração de software. Requisições de alterações e relatórios de problemas são veículos indicados para iniciar um processo de alteração e uma Comissão de Controle de Gerência da Configuração de Software é formada para garantir a gerência das *baselines*.

A seguir pode ser visto na figura 2, um diagrama que demonstra um processo básico de alteração na gerência da configuração segundo o CMM:

FIGURA 2 - PROCESSO DE ALTERAÇÃO SEGUNDO O MODELO CMM/SEI.



Fonte: [FIO1998].

No modelo CMM segundo [FIO1998], as práticas de gerenciamento da configuração estabelecem metas que são essenciais para o alcance do sucesso no processo que são:

- a) Meta 1 - Planejar as atividades de gerência da configuração de software.
- b) Meta 2 - Identificar, controlar e tornar disponível os itens de software selecionados
- c) Meta 3 - Controlar as alterações nos itens de software identificados.
- d) Meta 4 - Informar pessoas e grupos envolvidos acerca do estado e do conteúdo das baselines de software.

Neste sentido [LUC1997] evidencia que o cumprimento destas metas está intimamente relacionada às características comuns, que são:

- a) **Compromissos:** Descreve as ações que a organização deve tornar para assegurar que o processo está estabelecido e vai durar. Inclui práticas sobre políticas organizacionais e responsabilidades.
- b) **Habilitações:** Descreve as pré-condições que devem existir no projeto ou na organização para implementar o processo de software de maneira competente. Isto tipicamente envolve recursos, estrutura da organização e treinamento.
- c) **Atividades:** Descreve as regras e procedimentos necessários para implementar uma área chave de processo. Tipicamente envolve o estabelecimento de planos e procedimentos, execução do trabalho, acompanhamento e tomada de ações corretivas sempre que seja necessário.

- d) **Medições e Análises:** Descrevem a necessidade de medir o processo e analisar tais medições. Normalmente incluem exemplos de medições que podem ser adotadas para determinar o status e a efetividade das atividades a serem empreendidas.
- e) **Verificação da implementação:** Descreve os passos para assegurar que as atividades são executadas de acordo com o processo que tem sido estabelecido. Normalmente vai acompanhada de revisões e auditorias da gerência e do grupo de garantia de qualidade do software.

### 3.5 NORMA ISO 9000-3

Apesar de um processo de desenvolvimento de software ser considerado distinto dos demais processos de fabricação de outros produtos conforme estabelece [ASS1993], é extremamente necessário que esse campo de tecnologia seja provido de orientações adicionais para o estabelecimento de sistemas da qualidade onde estejam envolvidos os produtos de “software”.

A norma ISO 9000-3 define algumas diretrizes, a fim de tornar possível a aplicação da NBR 19001 as organizações que desenvolvem, fornecem e mantêm software. Estas diretrizes destinam-se a descrever os controles e métodos sugeridos para a produção de software que atendam aos requisitos do comprador, evitando-se não conformidade em todos os estágios.

Maiores informações sobre esta norma podem ser vistas em [ASS1993].

### 3.6 PROCESSO DE GERÊNCIA DA CONFIGURAÇÃO SEGUNDO A NORMA ISO 9000-3

O processo de gerência da configuração segundo [ASS1993], é um processo não dependente de fase, sendo que é considerada uma atividade de suporte que visa fornecer um mecanismo para a identificação, controle e acompanhamento das versões de cada item de software.

Segundo o modelo, o fornecedor do software deve desenvolver e implementar um plano de gestão que inclua: as organizações envolvidas na gestão da configuração, as atividades de gestão de configuração, ferramentas, técnicas, metodologias usadas e também o estágio ao qual os itens de configuração devem ser submetidos ao controle propriamente dito.

Quanto as atividades definidas pela norma, elas se subdividem em:

- a) **Identificação e rastreabilidade de configuração:** O fornecedor deve manter procedimentos para identificar os itens de “software”, durante todas as fases do ciclo de vida, desde a especificação até o desenvolvimento. Nesta atividade ainda devem ser aplicados procedimentos para assegurar que os seguintes pontos possam ser identificados em cada versão do item de software: especificações funcionais e técnicas, ferramentas de desenvolvimento usadas no projeto, todas as interfaces com outros itens de software e hardware e todos os documentos produzidos pelo item de configuração.

- b) **Controle de alterações:** O fornecedor deve estabelecer e manter procedimentos para identificar, documentar, analisar criticamente e autorizar quaisquer alterações em itens de “software” sob a gestão de configuração. Por isso antes da aceitação de uma alteração, sua validade deve ser confirmada, e os efeitos causados sobre outros itens devem ser identificados e examinados. Cabe ainda a atividade fornecer métodos para informar aos envolvidos quanto às alterações e para demonstrar a rastreabilidade entre as alterações e as partes modificadas dos itens de “software”.
- c) **Relatório de situação da configuração:** O fornecedor deve estabelecer e manter procedimentos para registrar, administrar e relatar a situação de itens de “software”, de requisições de alteração e da implementação das alterações aprovadas.

### 3.7 NORMA ISO/IEC 15504/SPICE

Segundo [EMA1998], a ISO/IEC 15504, conhecida atualmente como projeto SPICE (*Software Process Improvement and Capability dEtermination*), é uma norma em elaboração formada pela ISO (*International Organization for Standardization*) e pelo IEC (*International Electrotechnical Comission*), e constitui-se de um padrão para a avaliação do processo de software, visando determinar a capacitação de uma organização.

Além de cobrir todos os aspectos da Qualidade do Processo de Software, ela visa ainda orientar a organização para uma melhoria contínua do processo e sua constituição baseia-se em várias séries de modelos de avaliação de processos existentes.

Segundo [BAR1997], a norma ISO/IEC 15504 inclui um modelo de referência, que serve de base para o processo de avaliação. Este modelo é um conjunto padronizado de processos fundamentais, que orientam para uma boa engenharia de software. Este modelo é dividido em cinco grandes categorias de processo: Cliente-Fornecedor, Engenharia, Suporte, Gerência e Organização. Cada uma destas categorias é detalhada em processos mais específicos pela norma.

Além dos processos, o SPICE define também os 6 níveis de capacitação de cada processo, que pode ser incompleto, executado, gerenciado, estabelecido, previsível e otimizado.

Maiores informações sobre esta norma podem ser vistas em [EMA1998] e [BAR1997].

### 3.8 PROCESSO DE GERÊNCIA DA CONFIGURAÇÃO SEGUNDO A NORMA ISO/IEC 15504

Segundo [EMA1998], o processo de gerência da configuração se encontra no nível denominado de SUP.2 – Desempenhar o processo de Gerência da Configuração, e é considerado uma atividade de suporte - processos que podem ser empregados por qualquer um dos outros processos existentes.

O propósito de executar o processo de gerência da configuração segundo o autor, é de se estabelecer e manter a integridade de todos os produtos do projeto de software ao longo do ciclo de vida e sendo assim propõe algumas atividades que devem ser desempenhadas:

- a) **SUP. 2.1** Estabelecer um sistema de biblioteca de gerência da configuração: Criar um repositório de software com acesso através de controles que possam prover:
  - armazenamento e recuperação de itens da configuração (e suas versões);
  - compartilhar e transferir itens da configuração entre grupos envolvidos;
  - recuperação de arquivos de versões de itens da configuração;
  - correta criação de produtos da biblioteca.
- b) **SUP. 2.2** Identificar itens da configuração: Identificar cada produto obtido do trabalho desenvolvido, para ser posto sob gerência da configuração. Podem ser considerados como produto :
  - requisitos, projetos, código, testes;
  - outras baselines do produto(por exemplo, documentação do usuário);
  - planos de projeto de software;
  - padrões e procedimentos.
- c) **SUP. 2.3** Manter descrições dos itens da configuração: Fornecer e manter descrições de cada item da configuração, identificando:
  - sua decomposição em componentes de configuração de nível mais baixos;
  - a pessoa responsável por cada item;
  - quando colocada sob gerência da configuração.
- d) **SUP. 2.4** Administrar pedidos de mudança: Gravar, revisar, aprovar e rastrear todas as requisições de mudanças e relatórios de problemas para todos os itens da configuração e suas versões.
- e) **SUP. 2.5** Controlar as mudanças: Prover um controle de acesso para ajudar a manter a clareza e integridade dos itens da configuração gerenciados pela biblioteca do sistema.
- f) **SUP. 2.6** Construir versões do produto: Construir versões do produto somente através dos itens da configuração armazenados na biblioteca do sistema e somente quando autorizado.
- g) **SUP. 2.7** Manter históricos dos itens da configuração: Manter um histórico de cada item da configuração, e as ações realizadas através do gerenciamento da

configuração em um nível de detalhe suficiente para permitir que possam ser feitas recuperações de versões prévias.

- h) **SUP. 2.8** Relatar o estado da configuração: Regularmente reportar os resultados das atividades anteriormente realizadas de cada item da configuração.

## 4 COMPARATIVO ENTRE AS NORMAS E O MODELO

Este capítulo apresenta o comparativo realizado entre as normas de qualidade e o modelo estudado, referente o processo de gerência da configuração, que resultou numa proposta de atividades básicas a serem realizadas na gerência da configuração.

### 4.1 INTRODUÇÃO

Para o comparativo do processo de gerência da configuração, perante as normas e modelos de qualidade foram utilizados dois critérios: primeiramente foi estabelecido através do capítulo 2, as atividades básicas e essenciais para o processo de gerência da configuração. Em segundo, procurou-se analisar cada modelo e norma de qualidade para verificar se as mesmas atendiam às atividades propostas, originando assim uma tabela comparativa.

Concluiu-se então que para atender ao processo de gerência da configuração, as seguintes atividades devem ser executadas:

- a) identificação dos itens da configuração do software;
- b) controle da configuração do software;
- c) relato de “status” da configuração do software;
- d) auditoria da configuração.

Uma descrição de cada atividade definida, assim como comentários sobre as comparações realizadas é apresentado a seguir.

### 4.2 IDENTIFICAÇÃO DOS ITENS DA CONFIGURAÇÃO DO SOFTWARE

A norma ISO/IEC 12207 – processos de ciclo de software, define em seu item 6.2.2 Processo de Gerência da Configuração, que deve ser estabelecida uma sistemática para a identificação dos itens da configuração e suas versões a serem controladas. Para cada item da configuração e suas versões deve ser identificado o seguinte: a documentação que estabelece a linha básica(baseline); as referências de versão; e outros detalhes de identificação.

No modelo CMM, é estabelecido na meta 2 – Identificar, controlar e tornar disponível os itens de software selecionados, além de compromissos e habilitações, atividades que tem como objetivo: estabelecer um repositório para as baselines, identificar os itens da configuração e controlar a liberação dos produtos de software.

Já na ISO 9000-3, o fornecedor do software deve identificar os itens da configuração durante todas as fases do software, além disso cada item de software individual deve possuir uma identificação única. A norma ainda enfatiza a obrigatoriedade da aplicação de alguns procedimentos para que se possa identificar alguns pontos em cada versão dos itens, como: especificações funcionais e técnicas, ferramentas de desenvolvimento que foram utilizadas,

interfaces com outros itens de software e hardware, bem como todos os documentos e arquivos relacionados com o item de “software”.

Na norma ISO/IEC 15504, o item SUP. 2.2 – Identificar itens de configuração, especifica que cada produto obtido do trabalho desenvolvido, deve ser posto sob gerência da configuração, como: requisitos, projetos, códigos, padrões, procedimentos e outros.

Percebeu-se através do estudo comparativo uma grande afinidade em relação a esta atividade.

### **4.3 CONTROLE DA CONFIGURAÇÃO DO SOFTWARE**

A norma ISO/IEC 12207, relaciona no seu item 6.2.3 – Controle de Configuração, que as atividades de controle da configuração devem: identificar e registrar os pedidos de alteração, analisar e avaliar todas as alterações, aprovar ou rejeitar os pedidos de modificação e finalmente verificar e liberar o item de software modificado. É ressaltado ainda pela norma, que deve haver registros de auditoria, para cada modificação, a sua razão e sua autorização para que possam ser devidamente rastreadas futuramente.

Quanto ao modelo CMM, a atividade esta mencionada na meta 3 – controlar as alterações nos itens de software identificados - deve haver além de todos os controles nas alterações das baselines, o gerenciamento das requisições de mudanças e relatórios de problemas para todos os itens.

A ISO 9000-3 contempla esta atividade basicamente da mesma forma como o modelo CMM, e as outras normas, incluindo ainda que deverá ser realizada antes da aceitação de um alteração, a confirmação de sua validade, e os efeitos que podem causar sobre outros itens do projeto.

A norma ISO/IEC 15504, define no item SUP. 2.4 e SUP. 2.5 que todas as mudanças devem ser registradas na forma de pedidos de mudanças, onde estas requisições devem ser gravadas, revisadas e aprovadas. Além disso a atividade deve prover mecanismos para controle sobre os itens, mantendo a integridade geral do sistema.

De uma forma geral todas as alternativas estudadas chegaram a uma mesma definição quanto a esta atividade, tornando muito clara a tarefa de como deve ser posta em prática sua execução.

### **4.4 RELATO DE “STATUS” DA CONFIGURAÇÃO DO SOFTWARE**

Quanto a atividade de relato de “status”, a norma ISO/IEC 12207 em seu item 6.2.4 – Relato da situação da configuração - relata que devem ser preparados registros de gerenciamento e relatórios de situação que mostrem o estado atual do software, bem como um histórico de todos os itens de software controlados, incluindo a linha básica (baseline). Os relatórios de situação devem incluir o número de alterações em um projeto, as últimas versões do item de software, identificadores de liberação, a quantidade de liberações e as comparações entre elas.

No modelo CMM, na meta 4 – informar pessoas e grupos envolvidos acerca do estado e conteúdo das baselines de software – é descrito que deve haver as seguintes tarefas para o cumprimento da atividade: registrar o estado dos itens de configuração e divulgar as atividades de gerência da configuração.

Já a ISO 9000-3, como também a ISO/IEC 15504, assinalam de maneira idêntica que além de manter um histórico sobre cada item, deve haver regularmente a geração de relatórios a quem interessar, notificando as modificações sobre o projeto.

Notou-se de uma forma geral que todas as normas e o modelo estudado suportam esta atividade. Sua execução é essencial, tendo em vista a importância da sua contribuição em poder relatar todas as informações pertinentes ao processo aos interessados.

## 4.5 AUDITORIA DA CONFIGURAÇÃO

De uma forma geral a única norma estudada que trata a atividade de maneira bem explícita é a ISO/IEC 12207, que em seu item 6.2.5 - Avaliação da configuração, descreve que deve ser determinado e garantido a completeza funcional dos itens de software em relação aos seus requisitos e a completeza física dos itens de software, ou seja, se seu projeto e código refletem uma descrição técnica atualizada.

Já quanto ao modelo CMM, existem as etapas de verificação presentes em cada uma das metas de trabalho, ou seja, elas descrevem os passos necessários para assegurar que as atividades são executadas de acordo com o processo que foi estabelecido.

Quanto a norma ISO 9000-3 e a ISO/IEC 15504, observou-se que o tratamento dado a atividade, ocorre de maneira bastante implícita, ou seja, não havendo portanto um item específico e sim sendo desenvolvida juntamente com outras atividades como controle de configuração e relato de "status", servindo como uma etapa de complemento à elas.

Apesar de somente a norma ISO/IEC 12207 explicitamente citar esta atividade, optou-se por mantê-la no roteiro proposto devido a importância no contexto da gerência da configuração.

## 4.6 QUADRO COMPARATIVO DO PROCESSO

Após realizado o estudo do processo de gerência de configuração das normas ISO/IEC 12207, ISO 9000-3, CMM e a ISO/IEC 15504/SPICE, chegou-se a um comparativo, conforme a tabela 1. Essa tabela apresenta o que cada norma diz a respeito do processo de gerência de configuração, mais especificamente às atividades padrão que devem ser realizadas.

Na tabela a primeira coluna indica as atividades que devem ser atendidas no processo de gerência de configuração, e as demais colunas indicam os nomes das normas de qualidade.

O critério adotado para especificar essa tabela foi separar cada atividade do processo de gerência de configuração das normas de qualidade e colocá-los lado a lado para fazer um comparativo para verificar se há pontos em comum nessas normas.

TABELA 1 - COMPARATIVO DO PROCESSO DE GERÊNCIA DE CONFIGURAÇÃO.

Atividade \ Norma	ISO/IEC 12207	ISO 9000-3	SPICE	CMM
a) <b>Identificação dos itens de configuração.</b>	Item 6.2.2 - Para cada item de software e suas versões deve ser identificado: a documentação que estabelece a linha básica( <i>baseline</i> ); as referências de versão; e outros detalhes de identificação	O fornecedor deve estabelecer e manter procedimentos para identificar itens de "software" durante todas as fases, desde a especificação até o desenvolvimento, cópia e entrega.	SUP 2.2 - Identificar cada produto do trabalho para ser colocado sob gerência de configuração.	Atividade 4 - Identificar os artefatos de software que serão colocados sob gerência de configuração.
b) <b>Controle de configuração do software.</b>	Item 6.2.3 - Deve ser executado o seguinte: identificação e registro dos pedidos de alteração; análise e avaliação das alterações; aprovação ou rejeição dos pedidos.	O fornecedor deve estabelecer e manter procedimentos para identificar, documentar analisar criticamente e autorizar quaisquer alterações em itens de "software".	SUP 2.5 - Fornecer controle de acesso para ajudar a manter a clareza e integridade dos artigos de software no sistema de gerência de configuração.	Atividade 6 - Controlar alterações nas <i>baselines</i> de acordo com um procedimento documentado.
c) <b>Relato de "Status" da configuração do software.</b>	Item 6.2.4 - Devem ser preparados registros de gerenciamento e relatórios de situação que mostrem a situação e o histórico dos itens de software controlados, incluindo as <i>baselines</i> .	O fornecedor deve estabelecer e manter procedimentos para registrar, administrar e relatar a situação de itens de software, de requisições de alteração e da implementação das alterações aprovadas.	Regularmente reportar os resultados de cada atividade realizada e o status de cada item de configuração.	Atividade 8 - Registrar o estado dos itens de configuração.
d) <b>Auditoria de configuração</b>	Item 6.2.5 - Deve ser determinado e garantido a completeza funcional dos itens de software em relação aos seus requisitos e a completeza física dos itens de software.			Atividade 10 - Conduzir auditorias nas <i>baselines</i> de software.

## 4.7 JUSTIFICATIVA PARA A ESPECIFICAÇÃO

Após o comparativo ser realizado procurou-se com base na teoria proposta, obter em maiores detalhes os dados para formalização do Diagrama de Entidade-Relacionamento. Todas as entidades explícitas no modelo, foram criadas através de algum requisito ou de algum detalhe das normas ou o modelo estudado ou ainda através do capítulo 2 - Gerenciamento de Configuração do Software, presente neste trabalho. A tabela 2 apresenta de onde foram obtidas as entidades para o MER:

TABELA 2 - JUSTIFICATIVA PARA AS ENTIDADES DO MER.

<b>Entidade criada</b>	<b>Origem</b>
Alteracao	a) ISO/IEC 12207: processo 6.2.3; b) CMM: Meta 3 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.3.2; d) ISO/IEC 15504/SPICE: processo SUP 2.4 e 2.5; e) Gerenciamento de Configuração: capítulo 2.4.2.
Autor	a) ISO/IEC 12207: processo 6.2.1 e 6.2.2; b) ISO/IEC 15504/SPICE: processo SUP 2.3.
Analizador	a) ISO/IEC 12207: processo 6.2.3; b) Gerenciamento de Configuração: capítulo 2.4.2.
FaseCiclo	a) ISO/IEC 12207: processo 6.2; b) CMM: Meta 1, atividade 4 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.2, item d); d) ISO/IEC 15504/SPICE: processo 2.3; e) Gerenciamento de Configuração: item 2.4.1 e 2.4.2.
Ferramenta	a) CMM: item Recomendações – Ferramentas presente em [FIO1998]; b) ISO 9000-3: cláusula 6.1.2, item c e 6.1.3.1, item b).
Item	a) ISO/IEC 12207: processo 6.2.2.1; b) CMM: Meta 2, atividade 4 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.3.1; d) ISO/IEC 15504/SPICE: processo SUP 2.2; e) Gerenciamento de Configuração: capítulo 2.4.1.
PedidoModificacao	a) ISO/IEC 12207: processo 6.2.3.1; b) CMM: Meta 3, atividade 5 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.3.2; d) ISO/IEC 15504/SPICE: processos SUP 2.4 ; e) Gerenciamento de Configuração: capítulo 2.4.2.
Projeto	a) ISO/IEC 15504/SPICE: processo SUP.2.
VersãoItem	a) ISO/IEC 12207: processo 6.2.2.1; b) CMM: Meta 2 e Meta3, atividades 4, 5 e 6 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.1, item a);

	d) ISO/IEC 15504/SPICE: processos SUP 2.3 e 2.7; e) Gerenciamento de Configuração: capítulo 2.4.2.
Responsavel	a) ISO/IEC 12207: processo 6.2.2.1 e 6.2.3.1; b) CMM: Meta 2, atividade 4 presente em [FIO1998]; c) ISO 9000-3: cláusula 6.1.2, item a); d) ISO/IEC 15504/SPICE: processo SUP 2.3.
Requerente	a) ISO/IEC 12207: processo 6.2.3.1; b) ISO/IEC 15504/SPICE: processo SUP 2.4.
Tipo	a) ISO/IEC 12207: processo 6.2.2.1; b) CMM: Meta 2, atividade 4 presente em [FIO1998]; c) ISO/IEC 15504/SPICE: processo SUP 2.2; d) Gerenciamento de Configuração: capítulo 2.4.1.

## 4.8 JUSTIFICATIVA PARA OS RELATÓRIOS

Com base no modelo e nas normas estudadas ainda, procurou-se também criar relatórios que pudessem apoiar da melhor forma possível o processo de gerenciamento.

A tabela 3 apresenta a justificativa para cada relatório oferecido pelo software:

TABELA 3 - JUSTIFICATIVA PARA AS RELATÓRIOS.

<b>Relatório</b>	<b>Origem</b>
Relação de situação dos itens	a) ISO/IEC 12207: processo 6.2.4; b) CMM: Meta 4: Informar pessoas e grupos envolvidos acerca do estado e conteúdo dos itens ( <i>baselines</i> ) de software; c) ISO 9000-3: cláusulas 6.1.3.2 e 6.1.3.3; d) ISO/IEC 15504/SPICE: processo SUP 2.8; e) Gerenciamento de Configuração: item 2.4.3 - Relato do “status” da configuração.
Relação dos pedidos de alteração por versão	a) ISO/IEC 12207: processo 6.2.3; b) ISO 9000-3: cláusula 6.1.3.2.
Relação das alterações por versão do item	a) ISO/IEC 12207: processo 6.2.2; b) ISO 9000-3: cláusula 6.1.3.3; c) ISO/IEC 15504/SPICE: processo SUP 2.4.

## 5 DESCRIÇÃO DO SOFTWARE

Este capítulo trata sobre a especificação e implementação do software. Serão apresentadas a lista de eventos, com uma breve descrição sobre cada um, o diagrama de entidade relacionamento, tanto em nível lógico como físico, o dicionário de dados, o diagrama de contexto e os diagramas de fluxo de dados. Além disso pode ser visto a apresentação da maioria das telas do sistema, e uma breve explicação sobre o funcionamento das mesmas.

Cabe lembrar que o objetivo do software proposto é apoiar o processo de gerência de configuração de uma empresa baseando-se nas normas ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504/SPICE e também no modelo CMM/SEI.

### 5.1 ANÁLISE ESSENCIAL

Segundo [POM1994], durante muito tempo houve debates entre os profissionais de desenvolvimento de sistemas sobre por qual perspectiva se deveria começar a especificação de um sistema: pelos dados ou pelas funções? Hoje, pode-se dizer com uma visão mais abrangente do problema, que: devemos começar a especificação de um sistema pela identificação dos eventos que o afetam.

A análise essencial considera três perspectivas: funções, dados e controles. Em relação ao grau de abstração, a análise essencial considera dois níveis: o nível essencial e o nível de implementação. O nível essencial é representado pelo chamado modelo essencial e o nível de implementação é representado pelo chamado modelo de implementação.

Modelo Essencial - apresenta o sistema num grau de abstração completamente independente de restrições tecnológicas. Antes que um sistema seja implementado, é necessário conhecer-se a sua verdadeira essência, não importando saber se sua implementação vai ser manual ou automatizada, e nem mesmo que tipo de hardware ou software vai ser usado. É o que a análise estruturada chama de modelo lógico; no caso, o modelo essencial corresponde ao modelo lógico proposto, que é a denominação que a Análise Estrutura dá ao modelo lógico desejado para o sistema que ainda vai ser implementado. A Análise Estruturada recomenda também a elaboração do modelo lógico atual, que é a denominação por ela dada ao modelo lógico do sistema existente, o qual vai ser substituído pelo sistema em desenvolvimento. A Análise Essencial não recomenda a elaboração do modelo lógico atual, a não ser em casos muito particulares e, assim mesmo, parcialmente, apenas quando for necessário para auxiliar o entendimento do sistema a ser desenvolvido.

Modelo de Implementação - apresenta o sistema num grau de abstração completamente dependente de restrições tecnológicas. É derivado do modelo essencial. Diz respeito à implementação do sistema. Neste modelo, são colocadas todas as características tecnológicas, importando saber se parte de sua implementação vai ser manual ou automatizada, que tipo de "hardware" ou "software" vai ser usado etc. É o que a Análise Estruturada chama de modelo físico; no caso, modelo físico proposto, que é a denominação que a Análise Estruturada dá ao modelo físico desejado para o sistema que ainda vai ser implementado. A Análise Estruturada recomenda também a elaboração do modelo físico atual, que é a denominação por ela dada ao modelo físico atual, que é a denominação por ela dada ao modelo físico do sistema existente, o qual vai ser substituído pelo sistema em

desenvolvimento. A Análise Essencial não recomenda a elaboração do modelo físico atual, preceituado pela Análise Estruturada, a não ser em casos muito particulares, e também só com objetivos de auxiliar o entendimento do sistema a ser desenvolvido.

É comum realizar comparações da Análise Essencial, com a Análise Estruturada. No entanto, podemos acrescentar que a proposta da Análise Essencial acelera sobremaneira a velocidade da especificação do sistema, pois uma desvantagem sobre a Análise Estruturada é quanto ao tempo consumido na fase de especificação.

Algumas vantagens que podem ser ressaltadas da Análise Essencial em relação à Análise Estruturada são:

- 1) A Análise Estruturada aborda duas perspectivas do sistema - funções e dados, ao passo que a Análise Essencial aborda três perspectivas - funções, dados e controles, sendo, portanto mais completa.
- 2) Um dos problemas mais relevantes na especificação de um sistema é como efetuar o seu particionamento, ou seja, como, por exemplo, a partir do Diagrama de Contexto, decompor um sistema nos DFDs de nível de abstração mais baixo. A Análise Estruturada propõe um particionamento através de uma abordagem do tipo "Top-down". Embora esta seja uma boa maneira de se apresentar um problema complexo - começando da visão geral e ir descendo, passo a passo, numa visão hierárquica, a níveis de detalhes cada vez menores -, na prática, ela não se mostra eficiente como estratégia de projeto para a decomposição de sistemas. A Análise Essencial propõe uma outra forma de particionamento, a qual é baseada nos eventos, e que tem demonstrado ser muito mais efetiva do que a abordagem "Top-down", pois torna mais fácil a identificação das funções e entidades que compõe o sistema.
- 3) A Análise Essencial permite a construção dos modelos de dados e de funções concomitantemente, o que garante a correspondência entre os dois modelos. Vale lembrar que estas duas perspectivas apresentam-se sempre juntas, de modo interdependente.

Por isso, visando um melhor entendimento na prática deste estudo e por consequência da concepção do software, foi utilizado como método de especificação a Análise Essencial de Sistemas, com o suporte da ferramenta CASE *Power Designer* 6.1 da *Sybase Inc.*, [SYB2000], versão demonstração. E sua implementação realizada no ambiente de programação Visual Delphi 3.0 da *Imprise Inc.* ([CAN1997]).

## 5.2 LISTA DE EVENTOS

Segundo [POM1994], "um evento pode ser definido informalmente como um acontecimento do mundo exterior que requer do sistema uma resposta".

A rigor, o valor de um sistema está na sua capacidade de responder com eficácia a todos os estímulos a que for submetido. Assim, um sistema é construído para responder a estímulos. A cada estímulo, o sistema deve reagir produzindo uma resposta predeterminada.

Com isso, o software proposto possui a seguinte lista de eventos:

1. Analista cadastra autor.

Neste evento o analista responsável ou quem estiver operando o software deverá realizar o cadastramento dos autores dos itens de configuração, ou seja, a pessoa que responderá pela criação de determinado item mantido pelo sistema.

2. Analista cadastra requerente.

O cadastro de requerente tem por objetivo manter o nome da pessoa responsável pelo pedido de modificação em determinada versão do item de configuração.

3. Analista cadastra analisador.

O cadastro de analisador tem por objetivo manter o nome da pessoa responsável pela análise do pedido de modificação e das versões de um determinado item da configuração.

4. Analista cadastra responsável.

O cadastro de responsável visa identificar as pessoas responsáveis por determinada versão de um item da configuração.

5. Analista cadastra fase do ciclo.

Neste cadastro é possível a inclusão de fases do ciclo de vida do software. Este assinalamento tem sua importância pois, em determinados itens convém se estabelecer a fase em que determinado item se encontra no desenvolvimento.

6. Analista cadastra projeto.

No cadastro de projetos, é incluído todos projetos de software mantidos pelo sistema.

7. Analista cadastra tipo.

Neste evento o analista mantém os diversos tipos de itens de software que existem. Como exemplo, podemos ter um tipo procedimento, documento, ferramenta e outros.

8. Analista cadastra ferramenta.

Analista cadastra as ferramentas que serão usadas para compor determinado item da configuração. Como por exemplo, Borland Delphi, Power Designer e outros.

9. Analista cadastra item.

Neste evento o analista cadastra o item da configuração que será utilizado pelo sistema, como um manual do usuário, uma função de cálculo ou uma unidade de código.

10. Analista registra versão do item.

Neste evento o analista registrará uma versão para um determinado item da configuração, ou quantas versões como assim desejar.

11. Analista registra pedido de modificação de versão.

Neste evento o analista registra um pedido formal de modificação para determinada versão de um item da configuração, originado por um requerente previamente cadastrado.

12. Analista registra alteração de versão do item.

Neste evento o analista registra uma alteração para uma determinada versão de um item da configuração, originado por pedido de modificação de versão previamente aprovado e analisado.

13. Analisador recebe pedido de modificações por versão.

Neste evento o analisador recebe um relatório contendo os pedidos de modificação por versão.

14. Analisador entra com a análise do pedido de modificação.

O analisador entrará com a análise do pedido de modificação registrado, aprovando-o ou não.

15. Analisador entra com a análise da versão do item.

O analisador entra com a análise da versão do item, aprovando-a ou não.

16. Cliente recebe relação de situação dos itens;

Neste evento o cliente recebe um relatório de situação do itens de configuração.

17. Cliente recebe relação das alterações por versão.

Neste evento o cliente recebe um relatório contendo as alterações por versão de item.

### 5.3 DIAGRAMA DE CONTEXTO

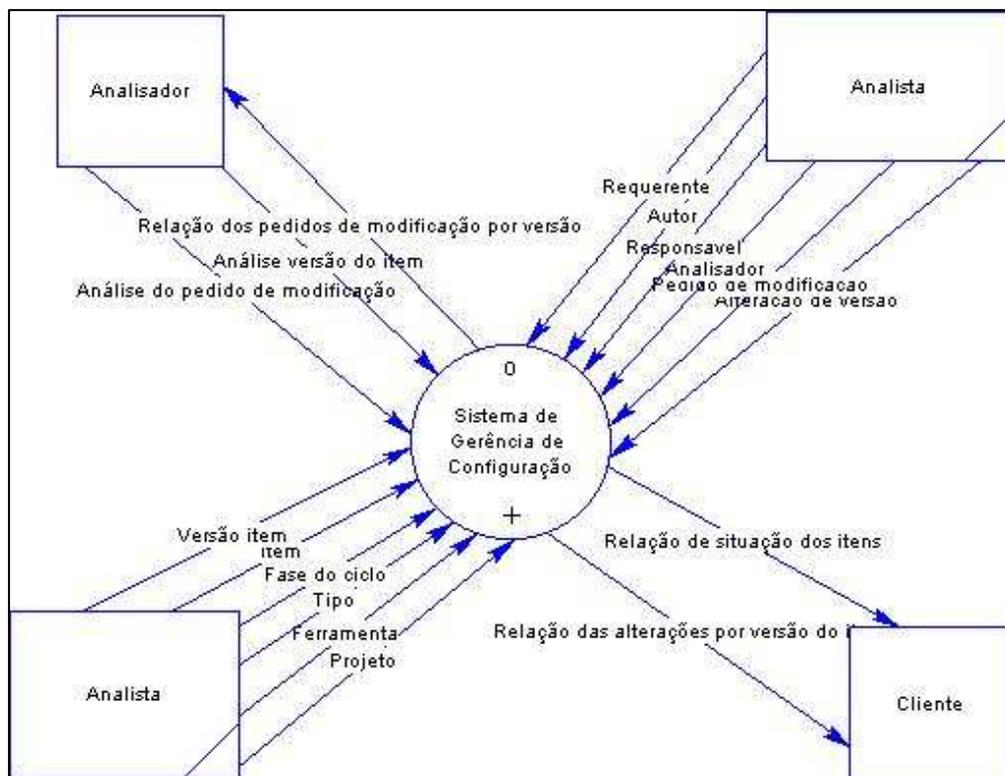
O Diagrama de Contexto estabelece os limites entre o sistema e o seu ambiente. É utilizado para se obter uma visão macro do software, comunicações entre o sistema, o ambiente e as entidades com as quais se comunica.

Na figura 3 é apresentado o Diagrama de Contexto do software, onde a pessoa responsável pelo sistema, representado pela figura do analista(Entidade Externa) realiza todos os cadastramentos básicos inicialmente no software como projetos, ferramentas, tipos de item, autores, requerentes, responsáveis, analisadores e fases do ciclo de vida.

Após estes cadastramentos o analista poderá incluir os itens de configuração que serão mantidos pelo software, bem como suas versões que assim forem surgindo. Para cada modificação exigida pelo requerente, deverá se realizado um pedido de modificação, que após previamente analisado e aprovado, deverá resultar em uma alteração para uma versão do item. Além disso cabe ainda também ao analisador verificar a integridade de determinada versão, aprovando-a ou não.

O software ainda permite a geração de relatórios de acompanhamento para o cliente, que são: relação das alterações por versão do item e relação de situação dos itens de configuração.

FIGURA 3 - DIAGRAMA DE CONTEXTO.



## **5.4 DIAGRAMA DE FLUXO DE DADOS**

O objetivo do Diagrama de Fluxo de Dados é mostrar um sistema completo ou parte dele, de onde os dados surgem, para onde vão, quando são armazenados, que processos os transformam e as interações entre armazenamento de dados e processos.

Nas figuras 4, 5, 6, 7 e 8, são apresentados o diagrama de fluxo de dados para cada evento conforme descrito no item 5.1 - Lista de Eventos:

FIGURA 4 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 1 A 5

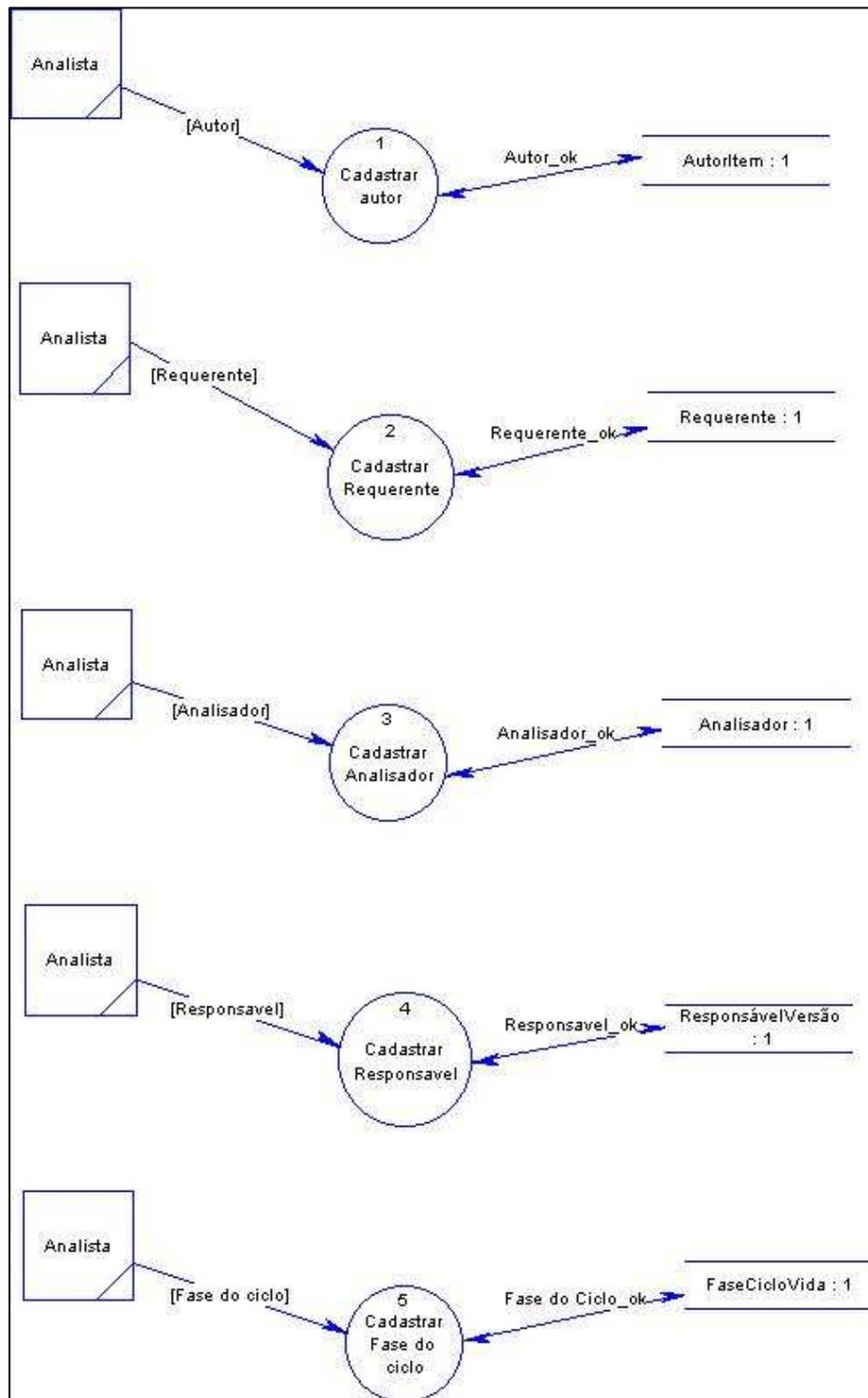


FIGURA 5 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 6 A 9

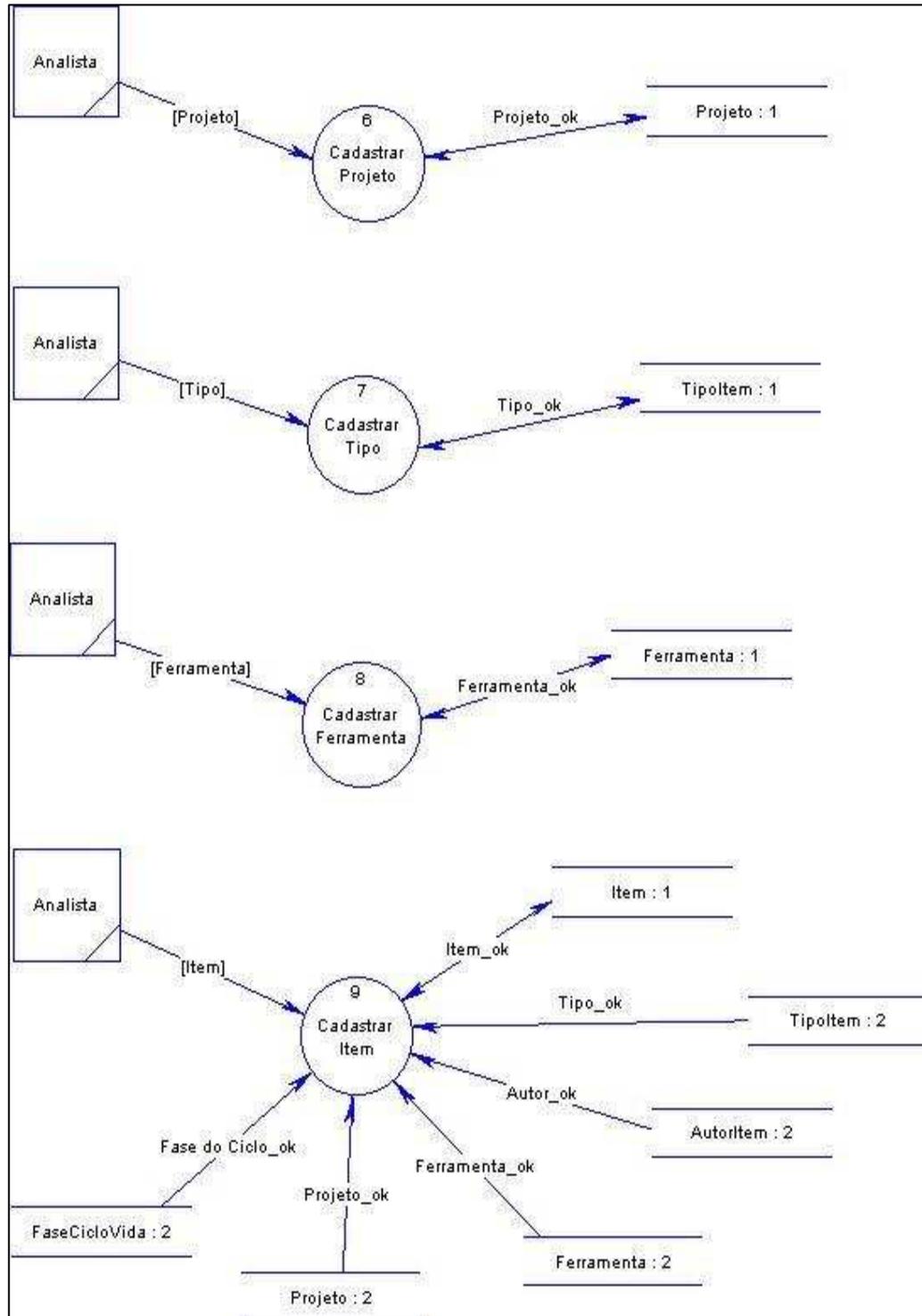


FIGURA 6 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 10 A 12

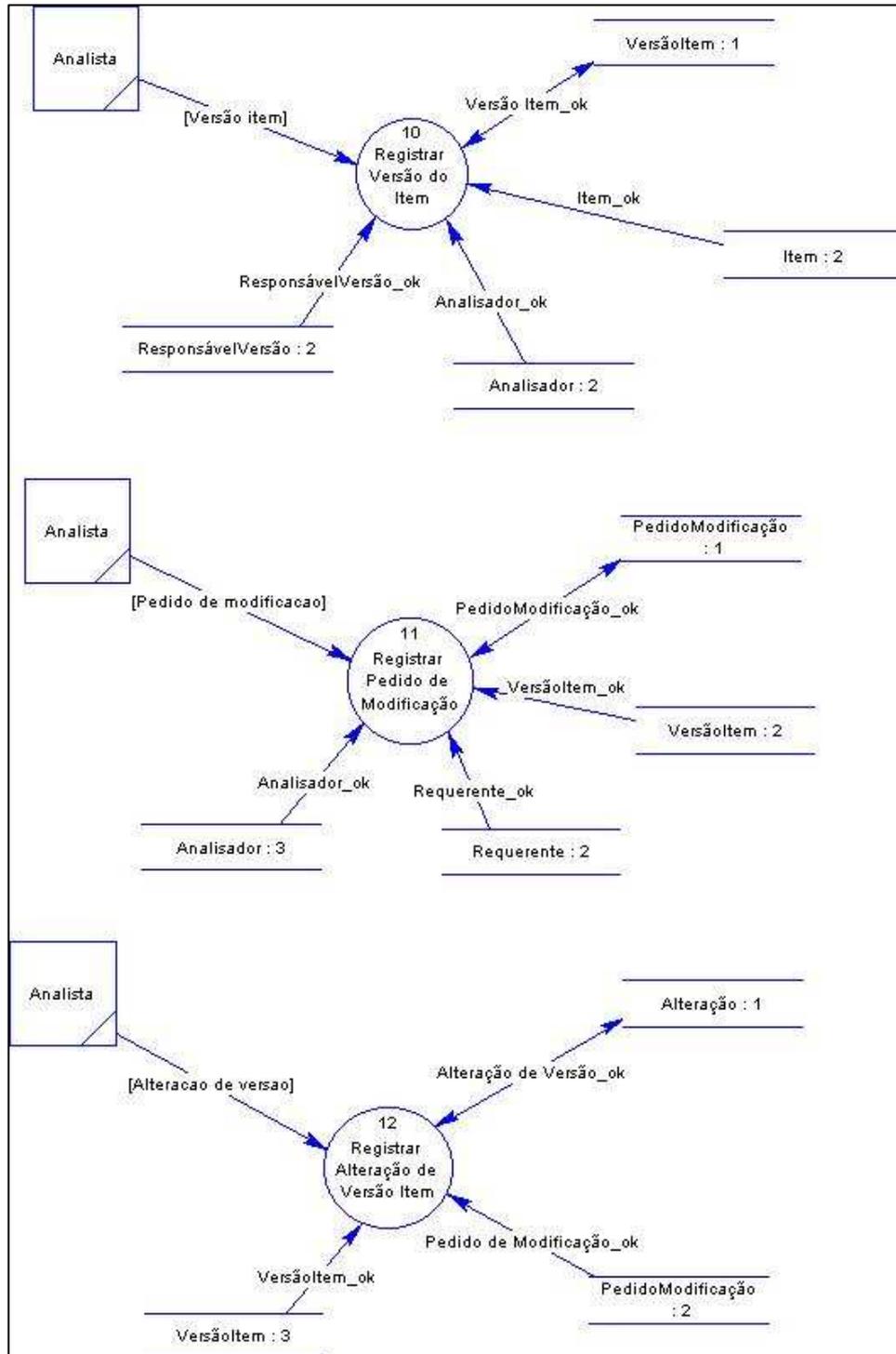


FIGURA 7 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 13 A 15

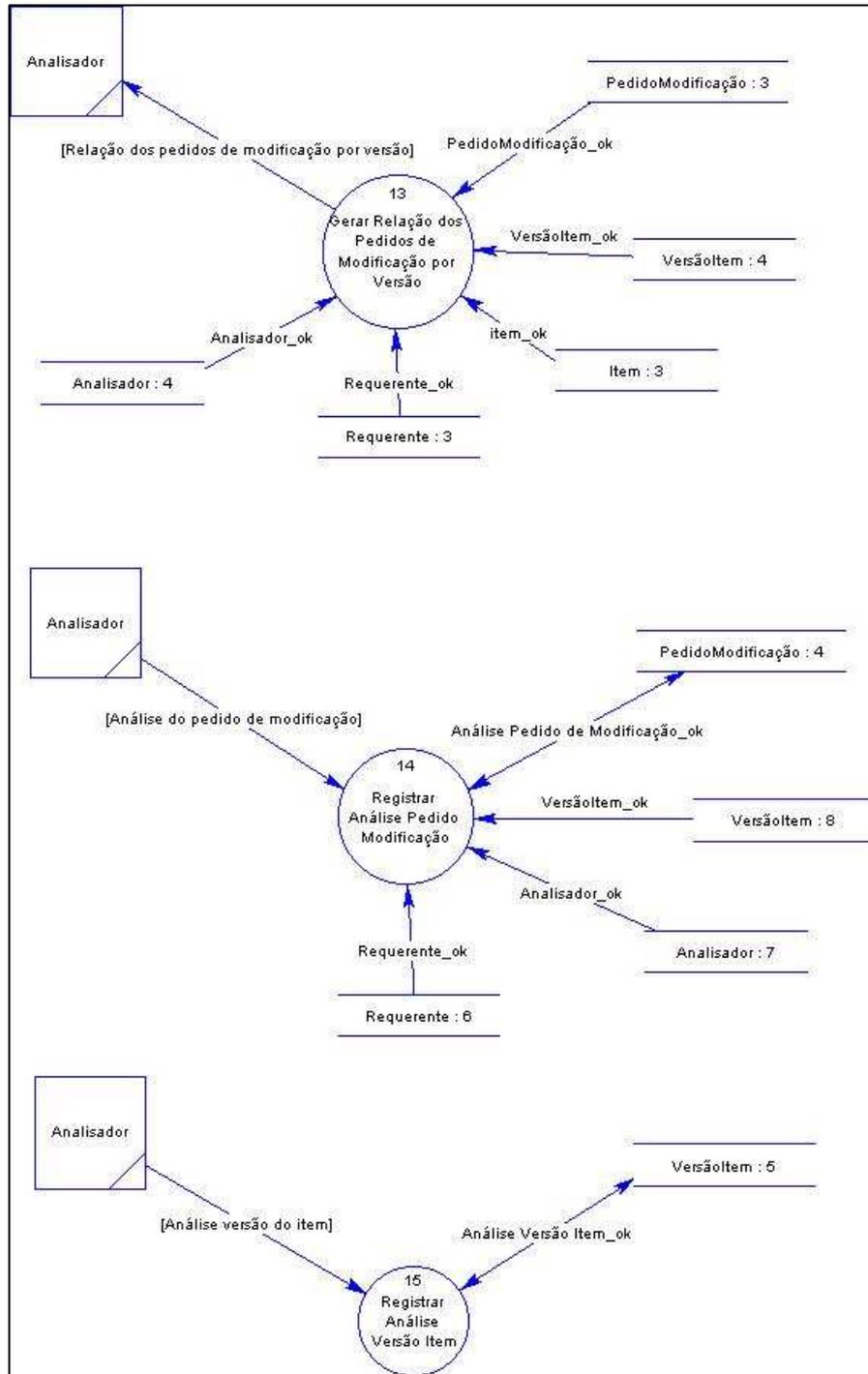
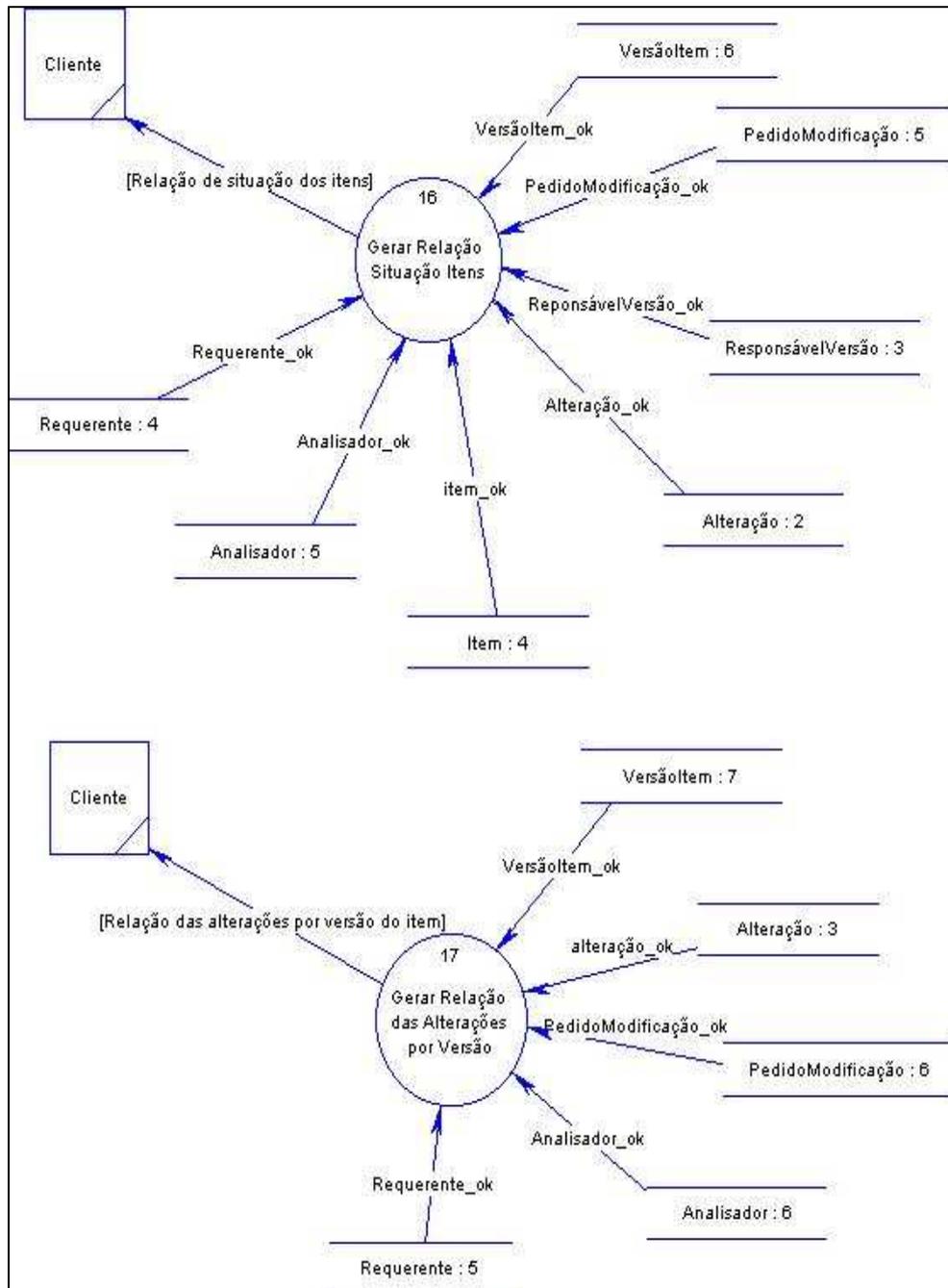


FIGURA 8 - DIAGRAMA DE FLUXO DE DADOS - EVENTOS 16 A 17



## 5.5 DIAGRAMA ENTIDADE-RELACIONAMENTO

O Diagrama Entidade-Relacionamento é um diagrama utilizado para detalhar as associações existentes entre as entidades de dados do sistema. Todas as entidades do software, como já mencionado se originaram de um estudo particular realizado com base nas normas utilizadas, conforme já detalhado no capítulo 4, item 4.6 - Justificativa para a especificação. A figura 9 apresenta este diagrama no nível lógico e em seguida na figura 10 a nível físico.

O Dicionário de Dados do software, foi gerado através da Ferramenta CASE Power Designer, que tem por objetivo fornecer suporte textual para complementar a informação mostrada no Diagrama Entidade-Relacionamento, sendo considerado um grupo organizado de definições, de todos os elementos de dados do sistema.

Para a documentação do Dicionário de Dados é utilizado o seguinte formato:

- a) a coluna *Name* apresenta uma breve descrição do atributo;
- b) a coluna *Code* apresenta o nome que identifica o atributo na tabela;
- c) a coluna *type* apresenta o tipo do atributo;
- d) a coluna *I* identifica se o atributo é chave primária da tabela;
- e) a coluna *M* identifica se é obrigatório o preenchimento do atributo.

### Entidade: ALTERAÇÃO

NAME	CODE	TYPE	I	M
Código alteração	COD_ALTER	I	Sim	Sim
Descrição alteração	DSC_ALTER	TXT50	Não	Sim
Data alteração	DT_ALTER	D	Não	Sim
Tipo alteração	TP_ALTER	A1	Não	Sim

### Entidade: ANALISADOR

NAME	CODE	TYPE	I	M
Código analisador	COD_ANAL	I	Sim	Sim
Nome analisador	NOM_ANAL	TXT50	Não	Sim

### Entidade: AUTORITEM

NAME	CODE	TYPE	I	M
------	------	------	---	---

Código autor item Nome autor item	COD_AUTORITEM NOM_AUTORITEM	I TXT50	Sim Não	Sim Sim
--------------------------------------	--------------------------------	------------	------------	------------

**Entidade: FASECICLOVIDA**

NAME	CODE	TYPE	I	M
Código fase ciclo Descrição fase ciclo	COD_FASECICLO DSC_FASECICLO	I TXT50	Sim Não	Sim Sim

**Entidade: FERRAMENTA**

NAME	CODE	TYPE	I	M
Código ferramenta Descrição ferramenta Nome fabricante	COD_FER DSC_FER NOM_FAB	I TXT50 TXT50	Sim Não Não	Sim Sim Sim

**Entidade: ITEM**

NAME	CODE	TYPE	I	M
Código item Descrição item Data criação item Nome arquivo eletrônico Item ativo Observação item	COD_ITEM DSC_ITEM DT_ITEM NOM_ARQELETR SIT_ATIVO DSC_OBSERV	I TXT50 D TXT50 BL LVA70	Sim Não Não Não Não Não	Sim Sim Sim Não Não Não

**Entidade: PEDIDO MODIFICAÇÃO**

NAME	CODE	TYPE	I	M
Código pedido modificação Descrição modificação Data pedido modificação Data análise modificação Indicação aprovação Indicação analisado Tipo modificação	COD_PEDMODIF DSC_MODIF DT_PEDMODIF DT_ANALMODIF IND_APROV IND_ANALIS TP_MODIF	I TXT70 D D BL BL A1	Sim Não Não Não Não Não Não	Sim Sim Sim Não Não Não Não

**Entidade: PROJETO**

NAME	CODE	TYPE	I	M
------	------	------	---	---

Código projeto	COD_PROJ	I	Sim	Sim
Descrição projeto	DSC_PROJ	TXT50	Não	Sim
Data início projeto	DT_INICPROJ	D	Não	Sim
Data término projeto	DT_TERMPROJ	D	Não	Sim

**Entidade: REQUERENTE**

NAME	CODE	TYPE	I	M
Código requerente	COD_REQUER	I	Sim	Sim
Nome requerente	NOM_REQUER	TXT50	Não	Sim

**Entidade: RESPONSAVELVERSAO**

NAME	CODE	TYPE	I	M
Código responsável versão	COD_RESP	I	Sim	Sim
Nome responsável versão	NOM_RESP	TXT50	Não	Sim

**Entidade: TIPOITEM**

NAME	CODE	TYPE	I	M
Código tipo item	COD_TIPOITEM	I	Sim	Sim
Descrição tipo	DSC_TIPOITEM	TXT50	Não	Sim
Diretório arquivos	DSC_DIRETITEM	TXT50	Não	Sim

**Entidade: VERSAOITEM**

NAME	CODE	TYPE	I	M
Código versão	COD_VERSAO	I	Sim	Sim
Descrição versão	DSC_VERSAO	TXT50	Não	Sim
Data criação versão	DT_CRIACAO	D	Não	Sim
Data análise versão	DT_ANALISE	D	Não	Sim
Observação versão	DSC_OBSERV	LVA70	Não	Não
Indicação analisada	IND_ANAL	BL	Não	Não

FIGURA 9 - DIAGRAMA ENTIDADE-RELACIONAMENTO - NÍVEL LÓGICO

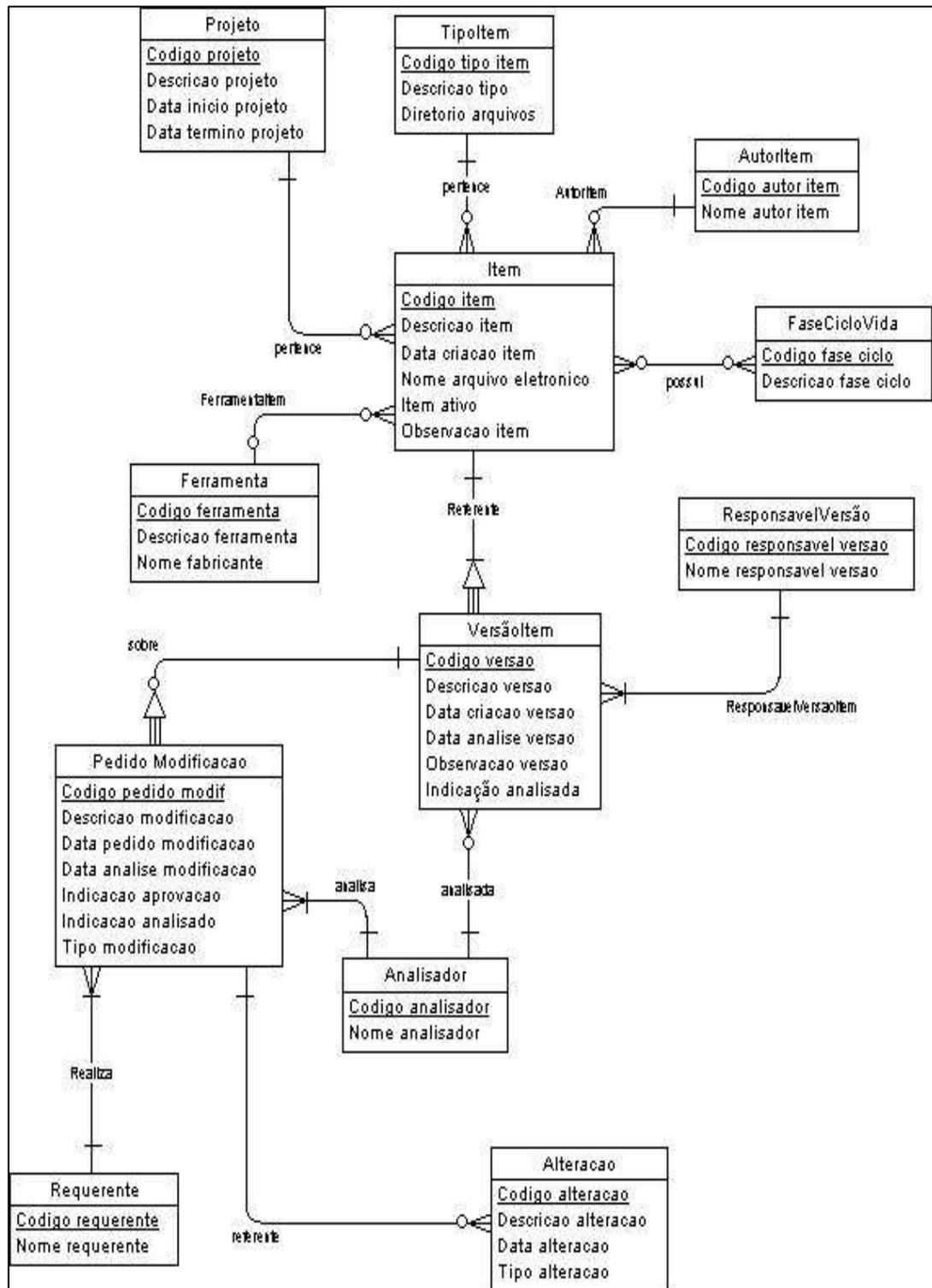
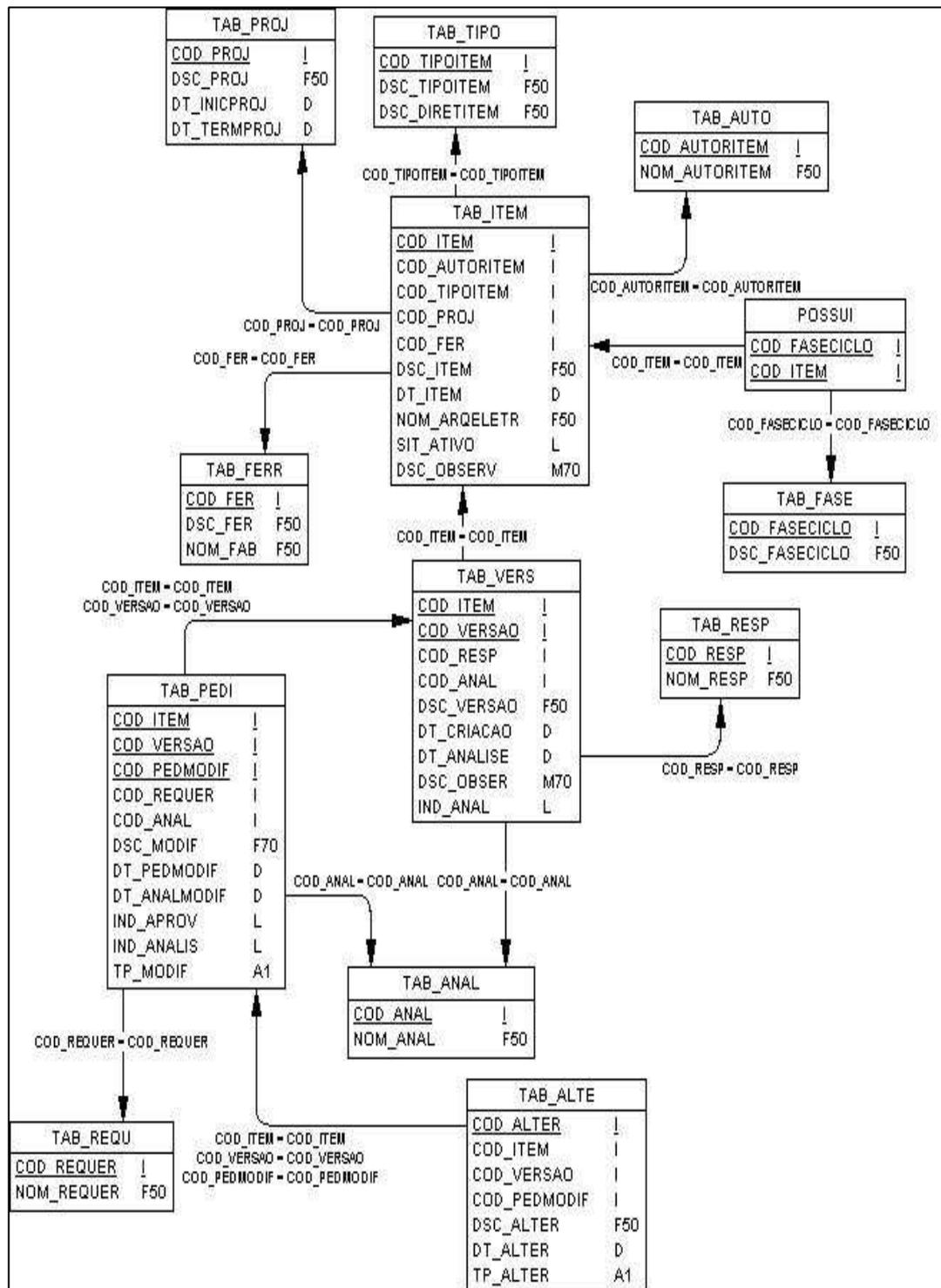


FIGURA 10 - DIAGRAMA ENTIDADE-RELAÇONAMENTO - NÍVEL FÍSICO



## 5.6 PRINCIPAIS TELAS

A seguir são apresentadas as principais telas do software, uma breve descrição sobre a sua funcionalidade, bem como a aplicação de um exemplo real para um entendimento geral.

A figura 11, apresenta a tela de cadastro de Projetos, o primeiro cadastro que deve ser realizado pelo analista. Nesta tela são cadastrados alguns dados referente aos projetos de software que serão mantidos pelo sistema, como uma breve descrição para o projeto iniciado, sua data de início e a data de término estimada.

FIGURA 11 - CADASTRO DE PROJETOS



A captura de tela mostra uma janela de diálogo intitulada "Cadastro de Projetos". O formulário contém os seguintes campos:

- Código: 1
- Descrição: Software de Análise de Custo
- Data de Início: 13/01/00
- Data de Término: 20/05/00

Na barra de controle inferior, há os seguintes botões: "Incluir", "Alterar", "Excluir", duas setas (uma para a esquerda e uma para a direita), "Confirmar", "Cancelar" e "Fechar".

A figura 12, apresenta a tela de cadastro de Ferramentas, caso o item de configuração que será posteriormente mantido pelo sistema, necessitar de uma ferramenta para sua composição. Deve se levar em consideração que em alguns casos, o cadastramento não será obrigatório, tendo em vista que as próprias ferramentas utilizadas em um projeto podem ser mantidas como itens de configuração. Os dados necessários são a descrição da ferramenta utilizada e o nome do fabricante da ferramenta.

FIGURA 12 - CADASTRO DE FERRAMENTAS

Cadastro de Ferramentas

Código 1

Descrição Borland Delphi 3.0

Nome do Fabricante Inprise

Incluir Alterar Excluir ← → Confirmar Cancelar Fechar

A figura 13, apresenta a tela de cadastro de tipos de Itens, nela é possível a manutenção dos diferentes tipos de itens adotados pelo sistema tais como procedimentos, documentos, testes ou outros quaisquer. Os campos solicitados são a descrição do tipo e um diretório onde os mesmos estão sendo armazenados.

FIGURA 13 - CADASTRO DE TIPOS DE ITENS

Cadastro de tipos de item

Código 1

Descrição Procedimento

Diretório c:\projetos\procedimentos

Incluir Alterar Excluir ← → Confirmar Cancelar Fechar

Na figura 14, é mostrada a tela de cadastro de fases do ciclo de vida do software, onde poderá a qualquer momento o usuário realizar a inclusão de etapas importantes na concepção do software, como uma fase de estudo, modelagem, design ou outras.

FIGURA 14 - CADASTRO DE FASES DO CICLO DE VIDA

Cadastro de Fases do Ciclo de Vida

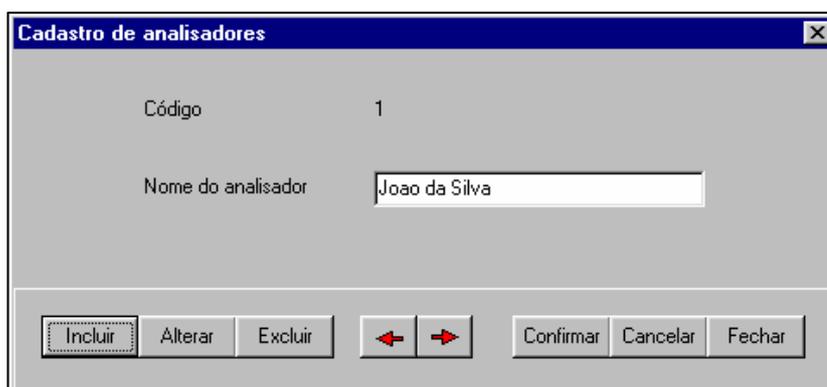
Código 1

Descrição Estudo

Incluir Alterar Excluir ← → Confirmar Cancelar Fechar

Na figura 15, é visualizado o cadastro de analisadores, onde é permitido a manutenção das pessoas responsáveis pela análise de determinada versão de item ou pedidos de modificação que existirem.

FIGURA 15 - CADASTRO DE ANALISADORES



Código	1
Nome do analisador	Joao da Silva

Incluir Alterar Excluir ← → Confirmar Cancelar Fechar

Convém lembrar que o cadastro de Autores, Requerentes e Responsáveis não será demonstrado, visto que sua apresentação segue o mesmo padrão do cadastro de analisadores.

A figura 16, apresenta a tela de Cadastro de Itens de Configuração, onde é possível cadastrar o projeto a que pertence o item, o tipo de item que se refere, sua descrição, a fase do ciclo de sua criação, data de criação, se está ativo no projeto corrente ou não, ferramenta usada, autor, nome do arquivo eletrônico do item e também um campo para observações.

FIGURA 16 - CADASTRO DE ITENS DE CONFIGURAÇÃO



Código	6
Projeto de Software	Software de Análise de Custo
Tipo do item	Documento
Descrição do item	Manual do usuário
Fase do Ciclo	Implementação
Data de criação	13/12/00
Ferramenta	SQS DOC Construtor de Help
Autor do item	Eduardo Alexandre Barbaresco
Nome do arquivo eletrônico	c:\manual.doc
Observações	

Item Ativo

Incluir Alterar Excluir ← → Confirmar Cancelar Fechar

A figura 17, apresenta a tela de cadastro de versões dos itens de configuração, permitindo informar o item a que se refere a versão, uma descrição, o seu responsável, a data de criação da versão, analisador, data da análise, observações se forem necessárias e também um indicativo se a mesma já foi analisada ou não.

FIGURA 17 - CADASTRO DE VERSÕES DOS ITENS

A imagem mostra uma janela de software intitulada "Cadastro de Versões de Itens". O formulário contém os seguintes campos:

- Código: 7
- Item: Manual do usuário
- Descrição da Versão: Manual do usuário versão 1.0a
- Responsável: Monica de Alcantra
- Data da Criação: 13/12/00
- Analisador: Joao da Silva
- Data da Análise: 14/01/01
- Observações: teste

Na parte inferior da janela, há uma barra de botões com as opções: Incluir, Alterar, Excluir, setas esquerda e direita, Confirmar, Cancelar e Fechar. O campo "Data da Análise" possui uma caixa de seleção marcada com o texto "Analisada".

Outra tela muito importante do software é mostrada na figura 18, no qual trata dos pedidos de modificação que serão requeridos para determinada versão do item de configuração. Neste momento são informados os seguintes dados: item de configuração, a versão do item que se deseja modificar, o requerente do pedido, o analisador para o pedido, uma descrição da modificação a ser realizada, data que foi realizado o pedido, data da análise, indicador se o pedido foi analisado e aprovado e também o tipo de modificação que será realizada, ou seja, se irá ser realizada uma mudança a nível de código, documentação, modelagem, procedimento ou outros.

FIGURA 18 - CADASTRO DOS PEDIDOS DE MODIFICAÇÃO DOS ITENS

Na figura 19, é mostrada a tela de alterações de versão, onde é realizado o cadastro das alterações originadas através dos pedidos de modificação aprovados. Os campos solicitados são: O item de configuração modificado, a sua versão, o pedido de modificação, a descrição da alteração, a data da alteração e o tipo de alteração (adaptativa, corretiva ou evolutiva).

FIGURA 19 - CADASTRO DE ALTERAÇÕES DE VERSÃO

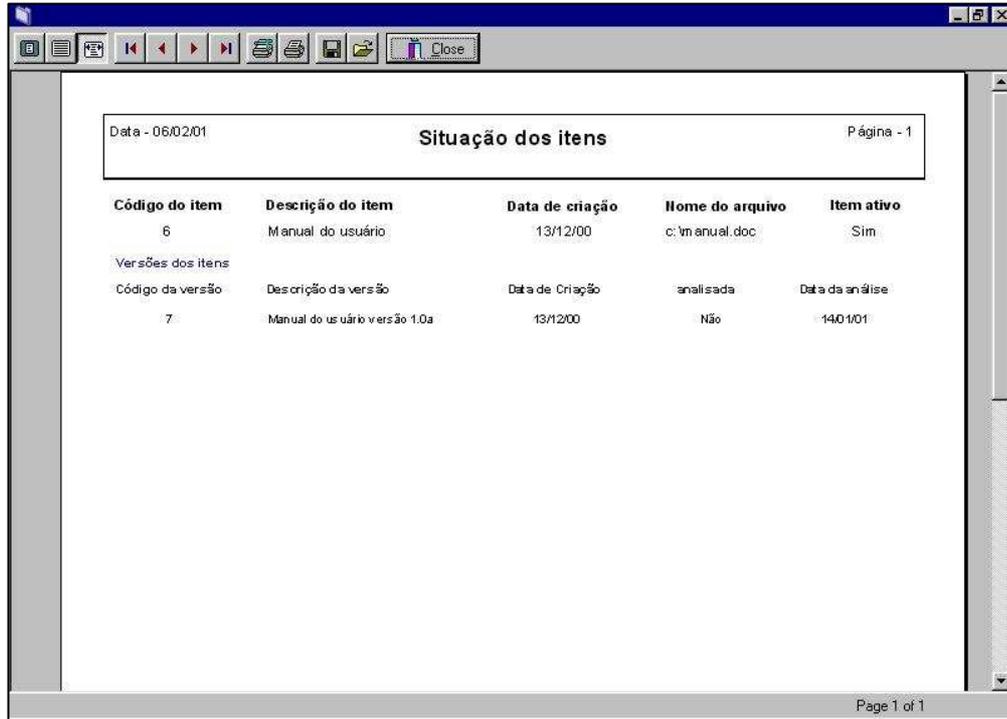
## 5.7 RELATÓRIOS

São três os relatórios oferecidos pelo sistema, que são destacados a seguir: o de situação dos itens, alterações por versão de item, e o de pedidos de modificação por versão. Cabe ressaltar que todos os exemplos abaixo mostrados, são provenientes dos dados incluídos ao longo da apresentação do sistema, no item 5.6 - Principais telas.

## 5.8 RELATÓRIO DE SITUAÇÃO DOS ITENS

Conforme a figura 20, neste relatório é mostrado a situação em que se encontra determinado item, sendo que através dele poderá se descobrir quantas versões determinado item possui.

FIGURA 20 - RELATÓRIO DE SITUAÇÃO DOS ITENS.



Código do item	Descrição do item	Data de criação	Nome do arquivo	Item ativo
6	Manual do usuário	13/12/00	c:\manual.doc	Sim

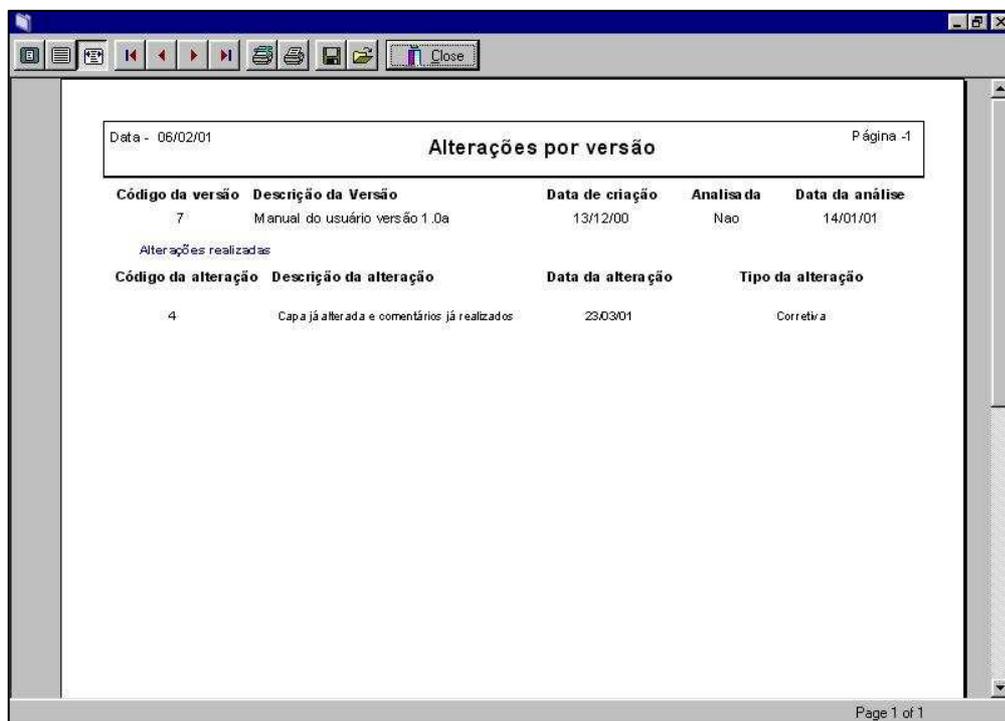
Versões dos itens

Código da versão	Descrição da versão	Data de Criação	analisada	Data da análise
7	Manual do usuário versão 1.0a	13/12/00	Não	14/01/01

## 5.9 RELATÓRIO DE ALTERAÇÕES POR VERSÃO DE ITEM

Neste relatório o cliente poderá visualizar a qualquer momento todas as alterações realizadas em determinada versão de um item de configuração, conforme pode ser visto na figura 21.

FIGURA 20 - RELATÓRIO DAS ALTERAÇÕES POR VERSÃO



Código da versão	Descrição da Versão	Data de criação	Analisa da	Data da análise
7	Manual do usuário versão 1.0a	13/12/00	Não	14/01/01

Alterações realizadas

Código da alteração	Descrição da alteração	Data da alteração	Tipo da alteração
4	Capa já alterada e comentários já realizados	23/03/01	Corretiva

## 5.10 RELATÓRIO DE PEDIDOS DE MODIFICAÇÃO POR VERSÃO

Neste último relatório é apresentado os pedidos de modificação por versão de item, que serve como importante instrumento de auxiliar o analista responsável nas tomadas de decisão no projeto. Maiores detalhes podem ser vistos na figura 22.

FIGURA 21 - RELATÓRIO DOS PEDIDOS DE MODIFICAÇÃO POR VERSÃO

Data - 06/02/01 Página - 1

Código da versão	Descrição da Versão	Data de criação	Analisa da	Data da análise
7	Manual do usuário versão 1.0a	13/12/00	True	14/01/01

Pedidos de modificação

Código do pedido	Descrição do pedido	Data do pedido	Aprovado	Analisado	Data da análise
4	Mudar capa e incluir comentários	20/02/00	Sim	Sim	23/02/00

Page 1 of 1

## 6 CONCLUSÕES

A seguir são descritas as considerações finais e sugestões deste trabalho.

### 6.1 CONSIDERAÇÕES FINAIS

De forma geral, as normas que foram adotadas para estudo no desenvolvimento deste trabalho, mostraram-se igualmente satisfatórias. Contudo, de todas as normas utilizadas, a norma ISO/IEC 12207 e o modelo CMM/SEI são os que melhor detalham o processo de gerência de configuração, sendo que este último inclusive propõe metas subdividindo o processo, apontando recomendações, verificações e também apresentando alguns conceitos importantes. As outras duas normas a ISO 9000-3 e ISO/IEC 15504/SPICE não apresentam muitos detalhes, sobre o processo estudado.

As principais dificuldades verificadas no desenvolvimento deste trabalho foram as definições da estrutura inicial do software, visto que as normas da qualidade estudadas não oferecem muitos detalhes que possam esclarecer como devem ser implementadas as entidades, sendo que boa parte foram baseadas também na teoria utilizada.

O objetivo inicialmente proposto foi realizado, visto que o software implementado conseguiu atender ao conjunto de atividades básicas presentes no processo de gerência de configuração que são: identificação, controle, relato de "status" e auditoria dos itens de configuração.

O plano de gerência de configuração, previsto neste tipo de processo não foi tratado neste trabalho, por ser considerada uma tarefa administrativa inicial aplicada ao processo. Sua elaboração requer que sejam devidamente explicitadas atividades a serem realizadas, procedimentos, cronogramas e seu relacionamento com outras organizações, de forma que é basicamente constituído através de entrevistas e reuniões entre os envolvidos.

Quanto ao software desenvolvido, é certo que o mesmo pode sofrer melhorias. Apesar de ser usado até o momento exclusivamente pelo autor deste trabalho, os resultados iniciais demonstram a facilidade de uso e o suporte básico ao processo de gerência de configuração, mantendo controle sobre os itens de configuração, suas versões, pedidos de modificação, alterações e também permitindo relatar seus estados através dos relatórios propostos.

Deve se ressaltar que sua melhor qualidade foi a de estar implementado conforme as especificações das normas de qualidade ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504/SPICE e o modelo CMM.

### 6.2 SUGESTÕES

O software poderia sofrer algumas melhorias como por exemplo a capacidade de conter cadastros mais avançados de usuários do sistema, procurando impor algum nível de segurança e fornecendo relatórios adicionais.

A adoção dos itens de configuração pelo sistema poderia ser alterada, objetivando-se automatizar melhor o processo, estabelecendo uma interface mais amigável ao analista,

propiciando mais abstrações, como procedimentos do tipo arrastar e soltar itens em sua interface, poupando assim o usuário de alguns cadastros eventuais.

Sugere-se também a criação de arquivos de ajuda contextual no software, para que o mesmo possa auxiliar todas as pessoas que estão envolvidas no processo e que possam manusear o software corretamente.

Para ampliação deste trabalho recomenda-se também o estudo de outras normas de qualidade e a busca de normas específicas sobre o processo de gerência de configuração.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [ABN1997] ABNT – Associação Brasileira de Normas Técnicas. **NBR ISO/IEC 12207 – Processos de Ciclo de vida de software**. NBR 12207. Rio de Janeiro, 1997.
- [ASS1993] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Normas de gestão da qualidade e garantia da qualidade – parte 3, diretrizes para aplicação da NBR 9001 ao desenvolvimento, fornecimento e manutenção de software**, NBR ISO 9000-3. Rio de Janeiro, 1993.
- [BAR1997] BARRETO, José Júnior. **Qualidade de software**. 16/12/1997. Endereço eletrônico: <http://www.barreto.com.br/qualidade/>.
- [BER1999] BERALDI, Lairce C.; FERNANDES, Fernando C.B; SCUDELLER, André. **Melhoria da Qualidade de Software através de Gerenciamento de Configuração**. Revista de Informática – Instituto de Ensino Avançado em Computação, v.1, n.2, p.62-71, junho. 1999.
- [CAN1997] CANTÚ, Marco. **Dominando o Delphi 3: A Bíblia**. São Paulo : Makron Books, 1997.
- [CHO2000] CHOOSE TECHNOLOGIES INC. **Starteam** 1999. Endereço Eletrônico: <http://www.choose.com.br/produtos.htm>. Data da consulta: 31/08/2000.
- [CTI1999] CTI – Fundação Centro Tecnológico para Informática. **Pesquisa sobre Gerência de Configuração de Software**. 1999. Endereço Eletrônico: <http://jacques.ic.cti.br/ic/pqps/gcs/Index.html>. Data da consulta: 31/07/2000.
- [EMA1998] EMAM, Khaled El et al; DROUIN, Jean-Normand; MELO, Walcélio. **SPICE – The Theory and Practice of Software Process Improvement and Capability Determination**. IEEE Computer Society, 1998.
- [FIO1998] FIORINI, Sueli T. & STAA, Arndt von & BAPTISTA, Renan Martins. **Engenharia de software com CMM**. Rio de Janeiro : Brasport, 1998.
- [LUC1997] LUCENA, Carlos J. P. de. **ISO 9000 versus MMC**. 11/08/1997. Endereço eletrônico: <http://www.les.inf.puc-rio.br/~wcourse/socinfo/nestor/Cuerpo>.
- [MAR1998] MARTINO, Wagner Roberto De; CRESPO, Adalberto Nobiato; PRIMO, Francisco Formoso; JR, Miguel de Teime e Argollo. **Implantação de Gerência de Configuração de Software em ambiente de Pequena Empresa: Abordagem e Aplicação**. In: WQS'98 – Workshop Qualidade de Software (1998 : Campinas). **Anais...**Campinas: CTI – Fundação Centro Tecnológico para Informática, 1998. p.67-75.
- [POM1994] POMPILHO, S.. **Análise essencial**. Rio de Janeiro : Infobook, 1994.
- [PRE1995] PRESMANN, Roger. **Engenharia de software**. Trad. de José Carlos Barbosa dos Santos. São Paulo : Makron Books, 1995.
- [SAN1994] SANDERS, J. & Curran, E. **Software quality**. Dublin: ACM Press Books, 1994.

- [SCO2000] SCOTT, John A. & Nisse, D. **Software Configuration Management**. California, cap.7, pág.7-2 – 7-12, abril. 2000.
- [SYB2000] SYBASE INC. **Power designer** 2000. Endereço Eletrônico:  
<http://www.sybase.com/products/>. Data da consulta: 09/08/2000.