

Este exemplar corresponde a redação final da tese
defendida por Adriana Delfino dos Santos
em 16 de 12 de 1999
Orientador



Universidade Estadual de Campinas - UNICAMP

Faculdade de Engenharia Elétrica e de Computação - FEEC

Departamento de Engenharia de Computação e Automação - DCA

Tese de Mestrado

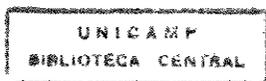
*Estabelecimento de
processos da
ISO/IEC TR 15504 em
organização de software
que possui
Gerenciamento da
Qualidade Total*

Adriana Delfino dos Santos

Orientador: Prof. Dra. Beatriz Máscia Daltrini

Banca examinadora: Prof. Dr. Mario Jino

Prof. Dr. Oswaldo Luiz Agostinho



UNIDADE	BC
N.º CHAMADA:	T/UNICAMP
	Sa59e
V.º	1
Es.	
T.º	41127
PREÇO	278/00
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	R\$ 11,00
DATA	16-06-00
N.º CPD	

CM-00142468-6

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Sa59e Santos, Adriana Delfino dos
Estabelecimento de processos da ISO/IEC TR 15504 em organização de software que possui gerenciamento da qualidade total / Adriana Delfino dos Santos --Campinas, SP: [s.n.], 1999.

Orientador: Beatriz Máscia Daltrini.
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Engenharia de software. 2. Gestão da qualidade total. 3. Controle de processo. I. Daltrini, Beatriz Máscia. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Resumo

A melhoria de processo de software é um dos fatores que contribuem para uma organização de software melhorar seu nível de competitividade em um mercado altamente competitivo e de economia globalizada. Este trabalho propõe uma maneira de estabelecer processos de software, baseados no modelo de referência de processo da ISO/IEC TR 15504 *Information Technology - Software Process Assessment*, em uma organização que adote ou esteja implementando uma abordagem de gerenciamento da qualidade total (GQT). Considerando-se que esta organização adota um método de gestão de processo - para gerenciar a rotina do trabalho do dia-a-dia, em uma abordagem GQT - utilizam-se as ferramentas de definição de escopo, macrodiagrama, fluxograma e descrição das atividades do fluxograma para descrever os processos. A descrição dos processos permite uma visão sistêmica das atividades e assegura que boas práticas de engenharia de software serão usadas. Outro elo com a abordagem de GQT é o uso de um modelo de gerência de projeto de software que estabelece uma seqüência de utilização dos processos durante a vida do projeto de software. Desta maneira, a organização prepara-se para futuras avaliações do processo de software, objetivando a melhoria gradativa e contínua de seus processos.

Abstract

Software Process Improvement is one of the factors that contribute to improve the level of competitiveness of the software organization in a highly competitive market of a global economy. This work suggests a way to establish software process, based on the reference model of the ISO/IEC TR 15504 Information Technology - Software Process Assessment, in an organization that adopts or is implementing a total quality management (TQM) approach. Considering that this organization adopts a process management method - to implement the daily work routine management in a TQM approach - tools for scope definition, macrodiagram, flowchart and flowchart activities description are used to describe the processes. The description of the processes permits an overview of the activities and assures that good software engineering practices will be used. Another link with the TQM has to do with the software project management activity establishing an utilization sequence of the processes during the software project life. This way, the organization becomes ready for future assessments of the software process aiming the gradual and continuous improvement of its processes.

Agradecimentos

Expresso aqui meu profundo reconhecimento à Empresa Brasileira de Pesquisa Agropecuária - Embrapa pela oportunidade de realizar este treinamento, na pessoa do Dr. Moacir Pedroso Jr., Chefe Geral da unidade Embrapa Informática Agropecuária.

Estendo os agradecimentos a todos os colegas de trabalho que me prestaram inestimável incentivo e apoio. Em especial, a Roberto H. Higa e Kléber X. S. Souza que com suas experiências contribuíram para definir o escopo deste trabalho, e à Fernanda, Carla, Laurimar, Cássia, Alessandra, Márcia, Evandro e José Ruy que sempre tinham um tempinho para me ouvir.

Hermes do Amaral Pacheco, agradeço-lhe por ter despertado em mim o interesse por qualidade em processo de software e incentivado-me a estudá-lo.

Agradeço aos meus familiares pelo apoio e incentivo durante todo este período, em especial à minha mãe Vanda e minha avó Sebastiana. Também agradeço a meu esposo Wilson que compartilhou comigo os momentos de alegria e também os de dificuldade, sempre tendo uma palavra ou um gesto de carinho e incentivo, mesmo estando entre nós os livros e o computador ...

Enfim, agradeço a *Deus* que presenteou-me com a vida e está sempre presente nela.

Índice

- 1 Introdução, 1
- 2 Qualidade, 5
 - 2.1 Conceito, 5
 - 2.2 Evolução dos conceitos de qualidade, 5
 - 2.3 Qualidade em processos de software, 9
 - 2.4 O modelo de avaliação de processo de software - ISO/IEC TR 15504, 10
 - 2.4.1 Modelo de Referência - Dimensão Processo, 14
 - 2.4.2 Modelo de Referência - Dimensão Capacidade, 15
- 3 Organização por projeto, 18
 - 3.1 Gerência de projeto, 18
 - 3.2 Estrutura de decomposição de trabalho - EDT, 19
 - 3.2.1 Estrutura de decomposição de produto - EDP, 21
 - 3.2.2 Blocos administrativos e gerenciais, 22
 - 3.2.3 Declaração de trabalho de bloco, 23
 - 3.2.4 Consolidação da EDT, 24
- 4 Metodologia, 29
- 5 Descrição dos processos, 34
 - 5.1 CUS.3 Processo de elicitação de requerimentos, 38
 - 5.2 MAN.2 Processo de gerência de projeto, 52
 - 5.3 SUP.2 Processo de gerência de configuração, 71
 - 5.4 SUP.6 Processo de revisão, 92
- 6 Modelo de gerência de projeto de software, 105
- 7 Aplicação do modelo de gerência de projeto de software na Embrapa, 110
 - 7.1 Apresentação da empresa, 110
 - 7.2 Aplicação do modelo de gerência de projeto de software, 113
- 8 Conclusões, 119
- 9 Referências bibliográficas, 122
- Anexo I Características de produtos de trabalho, 125
- Anexo II Descrição de escopo de um subconjunto de processos, 154
- Anexo III Formato do formulário *Esboço de Requerimento de Produto*, 172
- Anexo IV Formato do formulário *Lista de Perspectivas sobre o Sistema*, 173
- Anexo V Formato do documento *Especificação de Requerimentos de Cliente*, 179

Índice de Figuras

- Figura 1.1 Implementação de programa de Gerenciamento da Qualidade Total (GQT) ou similar, 1
- Figura 1.2 Distribuição das empresas, segundo ano de implantação de programa de GQT ou similar, 2
- Figura 1.3 Conhecimento de modelos de classificação de maturidade de processo, 2
- Figura 1.4 Tecnologia integrada para melhoria de processo de software, 4
- Figura 2.1 Avaliação de processo de software, 11
- Figura 2.2 Contexto de avaliação de processo, 12
- Figura 2.3 Dimensões do Modelo de Referência, 12
- Figura 2.4 Modelo de avaliação da Norma ISO/IEC TR 15504, 13
- Figura 2.5 Os processos na Dimensão Processo, 17
- Figura 3.1 Fases do Ciclo de Vida Genérico de Projeto, 18
- Figura 3.2 Árvore de decomposição do projeto "ALfa", 20
- Figura 3.3 Estrutura de decomposição do produto "P", 23
- Figura 3.4 Esquema da declaração de trabalho de um bloco, 24
- Figura 3.5 Matriz de responsáveis / tarefas, 25
- Figura 3.6 A matriz de controle de contrato, 26
- Figura 3.7 Uma árvore de especificações, 27
- Figura 3.8 Um trecho do gráfico de Gantt, 28
- Figura 4.1 Tabela de descrição de escopo de processo, 30
- Figura 4.2 Formato de macrodiagrama de processo, 31
- Figura 4.3 Exemplo de um fluxograma, 32
- Figura 4.4 Formato de descrição de processo, 33
- Figura 5.1 Dimensões de uma organização de software: ações e seus relacionamentos, 35
- Figura 5.2 Modelo de processo de software adaptado da ISO/IEC TR 15504, 37
- Figura 5.3 Correspondência entre atividades de desenvolvimento da EDT e práticas básicas ISO/IEC TR 15504, 53
- Figura 5.4 Exemplo de organograma de Grência de Configuração, 77
- Figura 5.5 Exemplo de processo para entrada de novo item na SCML, 81
- Figura 5.6 Exemplo de conjunto de documentação de configuração e software, 84
- Figura 6.1 Modelo de gerência de projeto de software: fases e macro-atividades, 105
- Figura 6.2 Seqüência da utilização dos processos de software pelas macro-atividades de Gerência de Projeto de Software, 109

- Figura 7.1 Níveis gerenciais da Embrapa e as figuras programáticas, 112
- Figura 7.2 Trecho da tabela de definição de escopo para o processo *CUS.3 Elicitação de requerimentos* adaptada para a Embrapa Informática Agropecuária, 115
- Figura 7.3 Trecho da tabela de definição de escopo para o processo *MAN.2 Gerência de projeto* adaptada para a Embrapa Informática Agropecuária, 115
- Figura 7.4 Macrodiagrama do processo *CUS.3 Elicitação de requerimentos* adaptado para a Embrapa Informática Agropecuária, 116
- Figura 7.5 Macrodiagrama do processo *MAN.2 Gerência de projeto* adaptado para a Embrapa Informática Agropecuária, 117
- Figura 7.6 Seqüência da utilização dos processos adaptada para a Embrapa Informática Agropecuária, 118
- Figura 8.1 Primeiro passo para melhoria de capacidade de processo de software, 121

Índice de Tabelas

Tabela 3.1	Informações de declaração de bloco, 23
Tabela 5.1	Sugestão de perguntas para o encontro preliminar entre cliente e desenvolvedor, 49
Tabela 5.2	Exemplo de relatórios de status de configuração, 76
Tabela 5.3	Estabelecimento de linhas básicas de projeto, 82
Tabela 5.4	Exemplo de categorias de software, 83
Tabela 5.5	Exemplo de níveis de liberação e autoridades associadas para categoria Software Produto, 85
Tabela 5.6	Exemplo de um pacote de mudança, 87
Tabela 5.7	Exemplo de critérios para avaliar Pacote de Mudança proposto, 87
Tabela 6.1	Descrição da seqüência de instâncias de processos por macro-atividades, 108

1 Introdução

O estudo da competitividade da indústria brasileira de software realizado em 1993 pelo Ministério de Ciência e Tecnologia identificou que a indústria brasileira estava começando a participar do mercado de economia globalizada, apresentando um crescimento substancial [Brasil, 1993]. Esta participação demandava o crescimento e a modernização da indústria e da prestação de serviços, baseados não só em inovação e incorporação de novas tecnologias, mas também no capital e na capacidade gerencial das empresas. O atendimento destas demandas promoveriam a competição de forma agressiva e em crescentes níveis de qualidade e produtividade.

Em 1998, segundo a última pesquisa de Qualidade no Setor de Software Brasileiro, publicada também pelo Ministério da Ciência e Tecnologia, a indústria de software está melhorando a sua capacidade gerencial com o objetivo de garantir a sua sobrevivência no mercado de economia globalizada [Qualidade..., 1998]. Em 1996 e 1997, das 589 empresas pesquisadas, 83,5% elaboravam planos estratégicos, planos de negócios ou planos de metas. Destas, 46% com atualização sistemática dos planos e 69% incluíam nestes planos metas ou diretrizes para a qualidade.

A maioria das empresas adota ou está implantando *programa de Gerenciamento da Qualidade Total (GQT) ou similar* (veja Figura 1.1), podendo-se observar um número crescente no processo de implantação a cada ano, de tal modo que mais da metade desses programas foram implantados nos anos de 1995 e 1996 (veja Figura 1.2). Na apuração do número de empresas que implantaram programa de GQT em 1997, foram considerados apenas os meses de janeiro a julho, quando se encerrou o trabalho de campo da pesquisa; 3 empresas não responderam à pergunta.

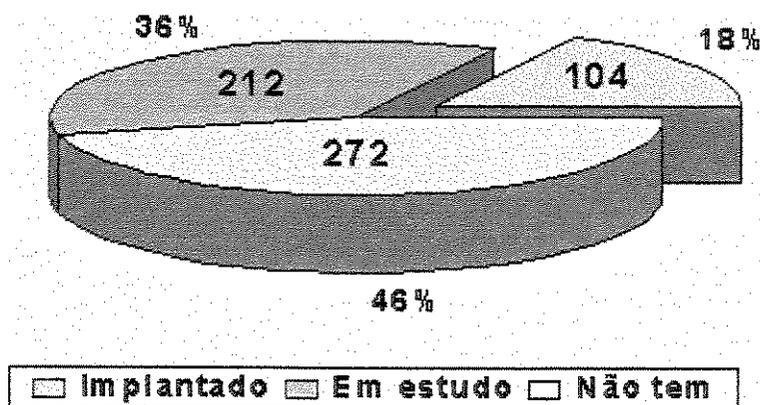


Figura 1.1 Implantação de programa de Gerenciamento da Qualidade Total (GQT) ou similar [Qualidade ..., 1998]

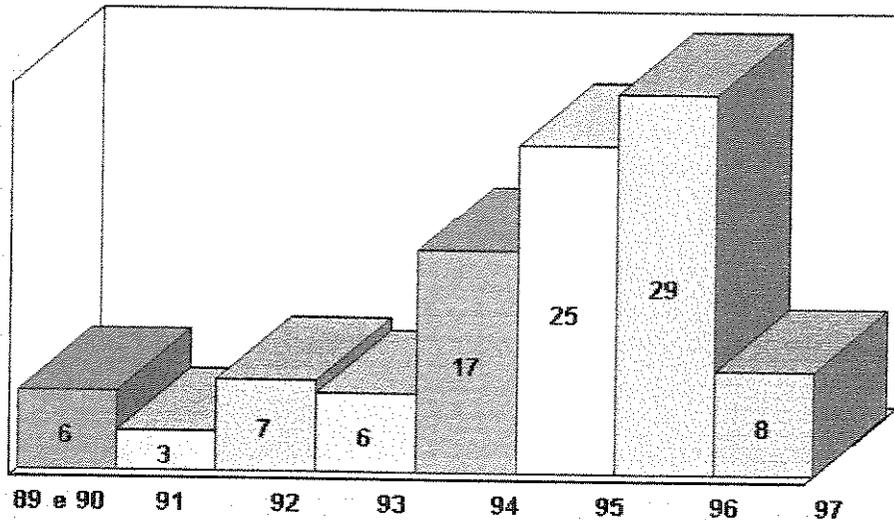


Figura 1.2 Distribuição das empresas, segundo ano de implantação de programa de GQT ou similar [Qualidade ..., 1998]

No que diz respeito à qualidade no processo de desenvolvimento de software, a indústria brasileira está começando a buscar a maturidade no seu processo de produção. Entende-se por organizações maduras aquelas em que papéis e responsabilidades são bem definidos, existe uma base histórica, a qualidade de produto e de processo é monitorada e é possível julgá-la, o processo pode ser atualizado e existe boa comunicação entre o gerente e seu grupo.

Existem vários modelos que classificam a maturidade do processo e a pesquisa focou nos modelos: Capability Maturity Model (CMM) e Software Process Improvement and Capability dEetermination (SPICE) (veja Figura 1.3). A busca da maturidade é verificada pela utilização do CMM por 5% das empresas pesquisadas, além do modelo ser conhecido por 29% delas. Com relação ao SPICE, 18% das empresas conhecem-no e apenas 1% utilizam seu modelo.

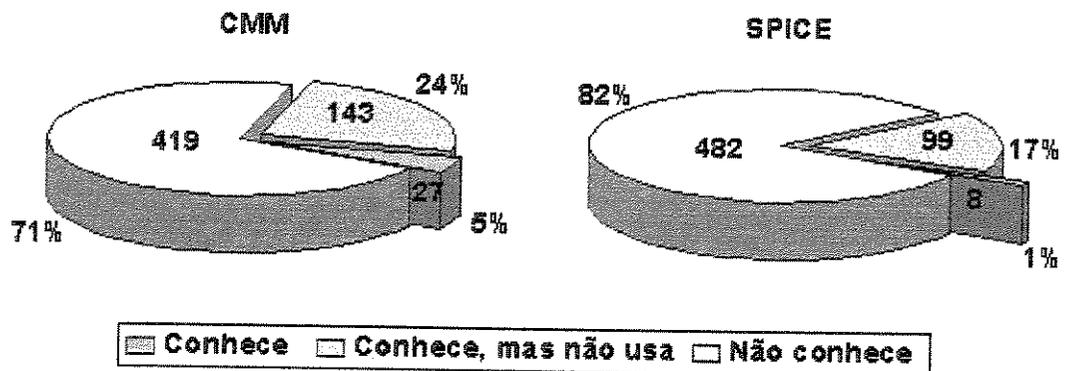


Figura 1.3 Conhecimento de modelos de classificação de maturidade de processo [Qualidade ..., 1998]

O CMM é um modelo proposto por Watts S. Humphrey, a partir de propostas de Philip B. Crosby, que vem sendo aperfeiçoado pelo Software Engineering Institute - SEI da Carnegie Mellon University. Este modelo classifica o sistema da qualidade de uma equipe ou empresa produtora de software, baseado na capacidade de seu processo de desenvolvimento [Humphrey, 1997]. *Capacidade de processo* é a habilidade de um processo obter uma meta requerida.

O SPICE é um projeto que envolve a comunidade internacional de especialistas em desenvolvimento de software e tem como objetivo unificar e harmonizar as diversas abordagens de avaliação de processo de software. O modelo do SPICE classifica os processos de software de uma organização, através da caracterização da prática corrente, identificando os pontos fortes e fracos e da caracterização da habilidade do processo para controlar ou evitar causas significantes de execução pobre de qualidade, custo e cronograma. Os resultados do projeto são a base para o futuro padrão internacional para avaliação de processo de software. Recentemente, foi publicado o relatório técnico¹ da *ISO/IEC² TR 15504 Information Technology - Software Process Assessment*.

Apesar da indústria de software brasileira estar melhorando sua capacidade gerencial, no que se refere à gestão empresarial, na área de desenvolvimento de software, a sistematização da produção é pouco adotada e problemas como garantia, mensuração e teste da qualidade de software ainda persistem no desenvolvimento de novos produtos. Como consequência destes problemas, (1) o desenvolvedor não entende bem o que o cliente precisa; (2) existem grandes distorções de custo e prazo em relação ao planejado; (3) muito esforço é direcionado ao retrabalho e (4) a documentação de desenvolvimento/usuário é pouca ou desatualizada.

Atualmente, organizações de software que possuem programa de gerenciamento da qualidade total precisam estabelecer um programa de qualidade de processo de software que incorpore a sistematização da produção de software. Normalmente, estas organizações têm uma estrutura organizada por projeto de software. Com o objetivo de contribuir para o estabelecimento de um programa de qualidade de processo de software, este trabalho propõe uma tecnologia que integra as dimensões *Gerenciamento da Qualidade Total - Qualidade em Processo de Software - Organização por Projeto*, como pode ser visto na Figura 1.4.

A tecnologia é composta pelos itens *descrição de processos de software e modelo de gerência de projeto de software*. O primeiro item integra as dimensões *Gerenciamento da Qualidade Total* e *Qualidade em Processo de Software*, considerando que a descrição dos processos faz parte da implantação do programa de GQT. Esta descrição permite uma visão sistêmica das atividades e assegura que boas práticas de engenharia de software serão usadas. Além disso, os processos de software estão baseados no modelo de referência para processo e capacidade de processo da ISO/IEC TR 15504. Este modelo promove a melhoria gradativa dos processos e permite que a organização alcance melhores níveis de maturidade, de acordo com padrões internacionais.

¹ Relatório técnico (Technical Report - TR) da ISO/IEC é a publicação prévia de um provável Padrão Internacional, para que os países membros da ISO constatem a sua real necessidade num período de três anos.

² ISO - International Organization for Standardization e IEC - International Electrotechnical Commission.

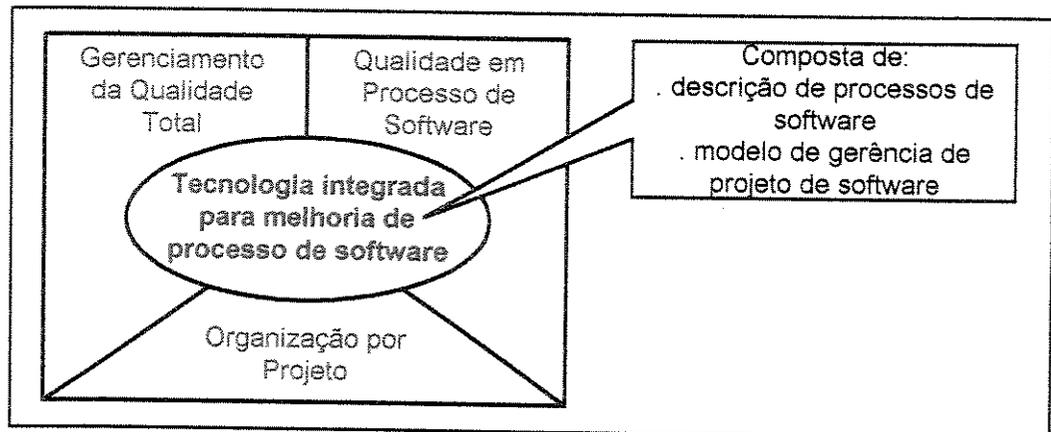


Figura 1.4 Tecnologia integrada para melhoria de processo de software

O segundo item integra as dimensões *Qualidade em Processo de Software* e *Organização por Projeto* através do estabelecimento de uma seqüência de utilização dos processos descritos ao longo do desenvolvimento de um projeto de software.

Os capítulos subsequentes deste trabalho descrevem a tecnologia e as ferramentas utilizadas em seu desenvolvimento.

O *Capítulo 2* apresenta a evolução dos conceitos de qualidade tanto em gestão empresarial como em processos de software, incluindo o modelo de avaliação de processo da ISO/IEC TR 15504, publicado em dezembro de 1998.

O *Capítulo 3* apresenta os conceitos gerais de estrutura organizada por projeto, incluindo ferramenta de apoio à gerência de projeto.

No *Capítulo 4*, são apresentadas as ferramentas de um Método de Gestão de Processo utilizadas para descrição dos processos de software. O Método de Gestão de Processo faz parte de uma abordagem de Gerenciamento pela Qualidade Total.

O *Capítulo 5* descreve os processos, utilizando-se as ferramentas apresentadas no capítulo anterior. A descrição do processo segue o Modelo de Referência para Processos e Capacidade de Processo da ISO/IEC TR, organiza as informações de escopo de processo contidas em diferentes partes da ISO/IEC TR 15504, identifica as interfaces do processo e o divide em práticas básicas¹. Também identifica atividades e define a seqüência de execução destas atividades, através de uma compilação das diferentes tecnologias (métodos, técnicas e ferramentas) de engenharia de software constantes da literatura. Esta descrição de processos contribuirá para uma indústria de software implementar seu método de Gestão de Processo, preparando-se para uma melhoria gradual e contínua da capacidade de seus processos.

¹ *Prática básica* é uma atividade de engenharia de software que contribui para a geração de saídas (resultados) de processo ou para aumentar a capacidade de um processo de software [ISO/IEC 15504, 1998], também conhecida como "boa prática de engenharia de software".

O *Capítulo 6* apresenta um Modelo de Gerência de Projeto de Software que estabelece uma seqüência para utilização dos processos de software descritos no Capítulo 5.

O *Capítulo 7* mostra a aplicação do modelo de gerência de projeto de software em uma empresa que adota um sistema abrangente de gerenciamento de qualidade total.

O *Capítulo 8* apresenta as conclusões do trabalho.

2 Qualidade

2.1 Conceito

O termo qualidade está bastante presente no cotidiano das pessoas como por exemplo nos comerciais de TV, nas escolas, no trabalho, nos bancos, nos supermercados, nos hospitais, em laboratórios de patologia. Seja por selo de certificação como ISO-9000, ISO-9002, INMETRO, ou por selo indicativo de participação em programas de excelência, como por exemplo *Programa de Excelência para Laboratórios Médicos*.

Mas qual é a definição de *qualidade*?

Na definição do dicionário da língua portuguesa [Ferreira, 1988], "*Qualidade é propriedade, atributo ou condição das coisas e das pessoas capaz de distingui-las das outras e de lhes determinar a natureza*".

Na literatura, muitas definições de qualidade têm sido propostas como por exemplo, *Qualidade é uma percepção de classe, excelência, um tipo de padrão referencial ou um reflexo das necessidades e expectativas do cliente*. Para Oakland, a palavra qualidade é muitas vezes empregada com o significado de *excelência* de um produto (bem ou serviço) [Oakland, 1994]. Para algumas empresas de engenharia, a palavra pode ser usada para indicar que a peça de metal está com certas características físicas e dimensionais estabelecidas muitas vezes na forma de uma especificação particularmente "*apertada*". Em um hospital, ela pode ser usada para indicar alguma espécie de "*profissionalismo*". Em administração, *qualidade é simplesmente o atendimento das exigências do cliente* e isso não é restrito às características funcionais do produto. Porém, todas as propostas possuem um senso comum que podem ser resumidas em *Qualidade é a satisfação do cliente obtida através da utilização otimizada de recursos do fornecedor*.

2.2 Evolução dos conceitos de qualidade

Durante o processo de produção primitiva, a qualidade do produto era baseada na perícia do artesão que aplicava arte e talento na manufatura. Cada produto era produzido em pequenas quantidades, as peças eram ajustadas manualmente umas às outras e a inspeção era feita de maneira informal para garantir a qualidade do produto final [Ferreira, 1995].

No início do século XX, a qualidade era garantida pela inspeção pós-produção, que enfatizava a conformidade com as especificações estabelecidas e seu objetivo era detectar problemas de fabricação no setor de produção das empresas [Ferreira, 1995].

A partir dos anos 30, nos Estados Unidos Shewhart e Deming iniciaram o enfoque de controle estatístico do processo [Cheng et al., 1995]. O controle era centrado no "como" da formação do produto final e visava buscar a aproximação entre a "qualidade de fabricação" e a "qualidade de especificação de projeto" e estreitar a faixa de variação da "qualidade de fabricação". Sua contribuição era para prevenção de problemas na fabricação de componentes e produtos, pela possibilidade de permitir detectar a tendência de um processo de fabricação [Ferreira, 1995].

Por volta de 1959, surge outra etapa marcante no movimento da qualidade, o enfoque da garantia da qualidade pelo desenvolvimento do produto [Cheng et al., 1995]. Este enfoque, além de necessitar dos dois enfoques anteriores (inspeção pós-produção e controle estatístico do processo, respectivamente), buscava uma aproximação entre a "qualidade exigida" pelos clientes e a "qualidade do produto recebido", passando pela "qualidade de especificação" e "qualidade de fabricação" do produto. Se no primeiro enfoque, visava-se detectar algo errado no produto final já pronto e no segundo, formar bem o que foi especificado, este terceiro visava conceber bem o que se propunha produzir e entregar de acordo com as necessidades e os desejos captados dos clientes. A qualidade passou de uma matéria restrita e baseada na produção de fábrica para uma matéria com maiores implicações para a função gerencial [Ferreira, 1995]. Foram incorporados à qualidade, além da estatística, os elementos:

- *os custos da qualidade*, que consideram a análise dos componentes: custos de falhas internas (sucata, retrabalho, análise de falhas, etc.), custo de falhas externas, custos de avaliação e custos de prevenção [Ferreira, 1995];
- *o controle total da qualidade*, que integra todos os departamentos da organização, tais como *marketing*, engenharia, produção e serviço com o objetivo de atuar num nível mais econômico permitindo a total satisfação do cliente; este método considera que a qualidade é feita pelas pessoas através de seu envolvimento nas várias etapas do ciclo industrial ou no trabalho de lançamento de um novo produto [Campos, 1992];
- *a engenharia de confiabilidade*, que tem por objetivo garantir um desempenho aceitável do produto ao longo do tempo, enfatiza a atenção dada à qualidade durante o processo do projeto (*design*) e considera os seguintes elementos: probabilidade, desempenho, tempo, condições de operação e aplicação do produto e condições de armazenamento e transporte do produto [Ferreira, 1995];
- *o defeito zero*, que está associado à prevenção de defeitos, *fazendo-se o trabalho certo na primeira vez* e considera que todos os defeitos são gerados pelos operadores, porém, na realidade, a maioria dos defeitos são relativos à gerência do processo e apenas uma pequena parte à operação [Ferreira, 1995].

Nas últimas décadas, a qualidade tem tido um grande destaque no sentido de prevenir problemas e obter vantagem competitiva. Muitos gurus aparentemente apresentam teorias diferentes sobre gerenciamento de qualidade. Na realidade, todos falam o mesmo idioma porém usam diferentes dialetos; os princípios básicos são comuns, tanto ao definir a qualidade como ao considerá-la através das atividades da empresa. A qualidade precisa ser administrada [Oakland, 1994].

Para administrar a qualidade por toda a empresa, existem vários sistemas integrados de gestão que buscam garantir a sobrevivência da empresa e promover o crescimento do ser humano. Por definição, um sistema de gerenciamento integrado de gestão de qualidade (GQT) "*significa que a cultura da organização é definida pela busca constante da satisfação do cliente através de um sistema integrado de ferramentas, técnicas e treinamento. Isso envolve a melhoria contínua dos processos organizacionais, resultando em produtos (bens ou serviços) de qualidade*" [Sashkin & Kiser, 1994].

Algumas idéias do GQT foram incorporadas durante o próprio processo evolutivo da qualidade e outras, mais recentes, surgiram da necessidade das empresas manterem uma vantagem competitiva sustentável, frente à globalização da economia [Ferreira, 1995]. Estas mais recentes estão voltadas a questões relacionadas com: maneiras de promover um alinhamento mais adequado entre os objetivos da empresa e todos os seus funcionários através de um planejamento estratégico [Certo & Peter, 1993]; maneiras de promover o trabalho de equipes autogerenciáveis (empowerment); maneiras de incentivar uma forma diferente de fazer as coisas, promovendo uma mudança cultural; e maneiras de medir a satisfação do cliente relacionando sua experiência, expectativas e desejos.

Os sistemas mais discutidos na literatura são Controle da Qualidade Total (TQC - Total Quality Control) no estilo japonês [Campos, 1992], também chamado de Company-Wide Quality Control, e Gerenciamento da Qualidade Total (TQM -Total Quality Management) na visão americana [Oakland, 1994].

O GQT adota uma abordagem abrangente que visa melhorar a competitividade, a eficácia e a flexibilidade por meio de planejamento, organização e compreensão de cada atividade, envolvendo cada indivíduo em cada nível [Oakland, 1994]. Ou seja, é a integração de todas as funções e processos dentro de uma organização com o objetivo de realizar a melhoria contínua da qualidade de produto (bem ou serviço). A meta é a satisfação do cliente.

O Gerenciamento pelas Diretrizes, apresentado por Campos [Campos, 1992], é uma das ferramentas para implementar o GQT. Este gerenciamento é um sistema administrativo, praticado por todas as pessoas da organização e que visa garantir a sobrevivência da organização a longo prazo através de uma *visão estratégica* e através de *direcionamento* da prática do controle da qualidade por todas as pessoas da organização (também chamado de gerenciamento da rotina do trabalho do dia-a-dia) segundo a visão estratégica estabelecida.

A *visão estratégica* e as diretrizes são estabelecidas através da adoção da abordagem de administração estratégica¹, ou seja, o rumo para estabelecimento das diretrizes é obtido com base em análise do sistema organização-ambiente e em crenças e valores da organização.

O *gerenciamento da rotina do trabalho do dia-a-dia* é implantado através de um método de Gestão de Processo. Este método introduz uma visão sistêmica do trabalho e mostra a interdependência existente entre fornecedores, executores e clientes do processo, como parte de uma cadeia destinada a gerar resultados organizacionais.

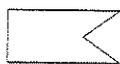
¹ *Administração estratégica* é definida como um processo contínuo e interativo que visa manter uma organização como um conjunto apropriadamente integrado a seu ambiente [Certo & Peter, 1993].

A Gestão de Processo é composta de três etapas: planejamento e organização, análise e aperfeiçoamento, acompanhamento e controle do processo [Embrapa, 1998b] e [Campos, 1992].

Planejamento e Organização

A etapa de Planejamento e Organização do processo consiste de atividades de identificação, priorização e descrição de processos. A descrição do processo tem por objetivo o entendimento do processo o qual é expresso pelo escopo do processo, macrodiagrama, desenho do fluxograma e detalhamento do fluxograma através de descrição de atividades.

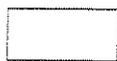
- O *escopo do processo* compreende a descrição de seu conteúdo básico, onde são identificados: nome do processo; objetivo, que indica de forma resumida, o que é feito pelo processo, como é feito e para quem é feito; entradas do processo - insumos; saídas do processo - resultados gerados pelo processo; início do processo, ou seja, a atividade que inicia o processo; clientes do processo - pessoas, setores da organização ou outras organizações, entre outros, que recebem os produtos ou saídas do processo; e indicadores de desempenho, que são parâmetros de avaliação de eficiência e eficácia do processo.
- O *macrodiagrama* é uma ferramenta de planejamento que mostra as entradas e saídas, fornecedores e clientes de um processo, bem como as atividades que possuem interface ou inter-relação com outro processo. Ele é um detalhamento da descrição do escopo do processo, de forma que suas informações devem ser consistentes com o escopo.
- O *fluxograma* é a representação gráfica das atividades ou fases de um processo, na seqüência como elas ocorrem, permitindo entender, a partir da representação visual, como o processo é executado. O fluxograma permite que os membros da equipe visualizem pontos de referência comuns e adotem uma linguagem padrão ao executar o processo. São utilizados diversos símbolos, dentre os quais cabe destacar os indicados a seguir.



Sinal de Entrada: sinaliza entrada do processo, normalmente o nome de uma prática básica..



Sinal de Saída: sinaliza interação com outra instância de processo, que pode ser dele mesmo.



Operação: representa os passos que podem existir em um processo.



Decisão: representa uma operação de decisão que determina o caminho a seguir dentre os vários possíveis.



Conexão: representa uma saída ou entrada para outra parte do fluxograma.



Início/Finalização: indica o início ou fim do processo.

Análise e Aperfeiçoamento

A etapa de *Análise e Aperfeiçoamento* do processo consiste da identificação e priorização de problemas, identificação e priorização das causas, proposição e priorização de soluções, planejamento das soluções e implementação dos aperfeiçoamentos.

Uma fonte para localização de problemas é a avaliação de processo. Esta avaliação é conduzida para verificar o cumprimento dos padrões e se cada processo está conseguindo atender às especificações de características de qualidade do produto [Campos, 1992].

Acompanhamento e Controle

A etapa de *Acompanhamento e Controle* do processo consiste do acompanhamento da implantação dos aperfeiçoamentos introduzidos no processo e do controle do processo. O acompanhamento utiliza-se do comportamento observado nos indicadores ou na forma como as atividades são executadas. O controle é uma fase de gestão do processo que visa manter o que foi planejado e organizado, de acordo com o padrão estabelecido.

2.3 Qualidade em processos de software

Nos anos 70, o desenvolvimento de software não conseguia acompanhar o crescimento da demanda e da capacidade de processamento das plataformas de hardware e esta situação ficou conhecida como "a crise do software". O cerne desta crise estava vinculado a dificuldades enfrentadas em submeter o processo de trabalho associado a uma forma de organização que permitisse a progressiva racionalização das atividades envolvidas. O progresso na engenharia de software promete proporcionar incrementos significativos na produtividade e qualidade da produção de software, sendo uma importante vantagem competitiva.

O avanço da fronteira tecnológica envolve três frentes. A primeira refere-se à evolução das linguagens de programação, em especial aquelas baseadas em técnicas orientadas a objeto e aquelas destinadas às aplicações para a Internet. A segunda refere-se à difusão de ferramentas CASE (*Computer-Aided Software Engineering*), que permitem a automação parcial de diferentes tarefas relacionadas ao desenvolvimento de novos "softwares" com impactos positivos sobre produtividade e qualidade. A terceira decorre de melhoramentos na forma de organizar e gerir as atividades de desenvolvimento de software, sistematizando-se o processo de produção de software. A sistematização do processo de produção de software era incipiente até o início da década de 90 [Brasil, 1993], principalmente se comparado ao estágio alcançado pela maioria dos setores industriais, o que ficava evidente pelas grandes dificuldades que eram enfrentadas para garantir, mensurar e testar a qualidade de novos produtos de software.

Nos anos 80, iniciativas da área militar dos Estados Unidos e do Reino Unido tinham como objetivo melhorar o mecanismo de seleção de fornecedor de software, buscando diminuição do custo associado com software, redução dos riscos associados com projetos de software e melhoria da qualidade do software que estava sendo desenvolvido [Emam et al., 1998]. O mecanismo de seleção de fornecedor de software normalmente analisava propostas, enquanto que o de fornecedor de hardware analisava o seu processo de fabricação [Humphey, 1997].

A indústria de software começou a usar vários métodos que focavam na melhoria de processo de software e/ou na determinação de sua capacidade. Dentre eles, destacam-se o Capability Maturity Model - CMM (Estados Unidos), BootStrap Methodology (Comunidade Européia), HealthCheck e SDT (Reino Unido) e Trillium (Canadá). Alguns deles baseiam-se em padrões internacionais como série ISO 9000, IEEE¹, IEC e são de domínio público [Emam et al., 1998].

Para cada um destes modelos de avaliação, a indústria de software tinha que preparar o material requerido, além de capacitar seus profissionais no modelo. Como consequência, o tempo consumido na preparação de material requerido para avaliação estava se aproximando do tempo utilizado para desenvolvimento do software. Esta situação mundial gerou a demanda por um padrão internacional de avaliação de processo de software que fosse consenso da comunidade mundial produtora de software. Além disso, as empresas que queriam alcançar níveis internacionais de competitividade precisavam de um modelo de avaliação que lhes proporcionasse uma auto-avaliação.

Criou-se então, o projeto SPICE - Software Process Improvement and Capability dEtermination para agilizar o desenvolvimento de um padrão internacional de avaliação de processo de software e para unificar e harmonizar as diversas abordagens existentes, além de atender às demandas identificadas [Emam et al., 1998]. Os resultados do projeto SPICE foram a base para a publicação, em 1998, da norma *ISO/IEC TR 15504 - Information Technology - Software Process Assessment* contendo um modelo de avaliação de processo de software, após duas fases de experiências na utilização do modelo proposto pela comunidade mundial.

O modelo de avaliação da ISO/IEC TR 15504 está fundamentado nos conceitos de qualidade, e em ferramentas de gerenciamento da qualidade tais como administração estratégica e gestão de processo.

2.4 O modelo de avaliação de processo de software da ISO/IEC TR 15504

Avaliação de processo é a avaliação disciplinada do processo usado por uma organização, juntamente com um conjunto de critérios, para determinar a capacidade desse processo executar dentro de metas de qualidade, custo e cronograma. O objetivo é caracterizar a prática corrente, identificando pontos fracos e fortes, e a habilidade do processo para controlar ou evitar causas significantes de execução pobre de qualidade, custo e cronograma. [Emam et al., 1998].

A ISO/IEC TR 15504 define um modelo de avaliação de processo de software que está fundamentado em processos de software. Um processo é examinado por uma avaliação e os resultados desta avaliação conduzem para melhoria de processo (auto-avaliação) ou para determinação de capacidade de processo. A determinação de capacidade identifica capacidade e riscos de processo e motiva a melhoria de processo. A melhoria de processo identifica mudanças para o processo. A Figura 2.1 mostra os objetos do modelo de avaliação e seu relacionamento.

¹ IEEE - Institute of Electrical and Eletronics Engineering.

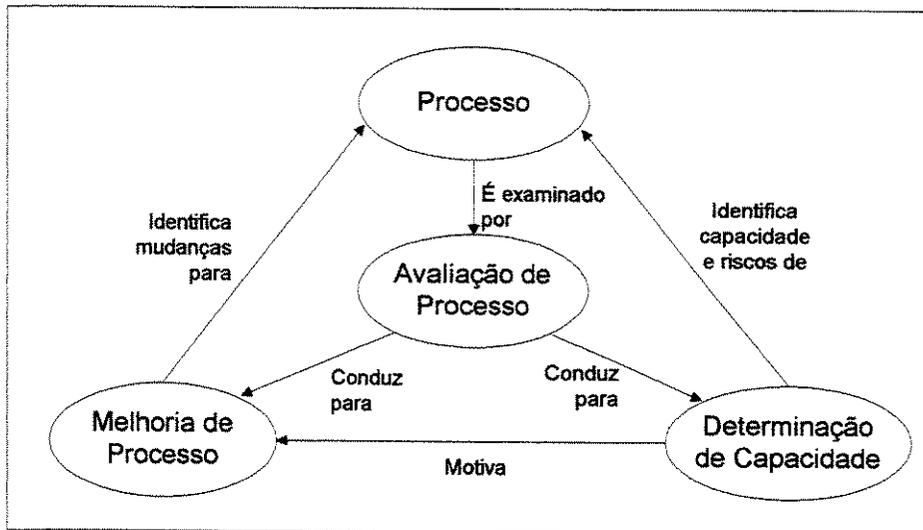


Figura 2.1 Avaliação de processo de software [ISO15504, 1998]

O contexto de uma avaliação de processo é apresentado na Figura 2.2. A identificação de uma necessidade de melhoria de processo ou de determinação de capacidade inicia a preparação para uma avaliação de processo. A preparação consiste da identificação de dados de entrada de preparação - tais como patrocinador, propósito da avaliação, escopo, restrições e responsabilidades de avaliação. Atividades de avaliação - como planejamento, coleta e validação de dados, taxaço de processo¹ e apresentação de resultados - são realizadas durante a avaliação. Um modelo de avaliação compatível com o da norma é requerido, para proceder-se à avaliação. Este modelo compatível deve possuir um modelo de referência, em que processos são definidos através de declaração de propósitos e identificação de atributos, e também possuir um conjunto de indicadores tanto de execução de processo como de capacidade de processo. A saída de avaliação de processo contém o registro completo da avaliação, desde a preparação, as atividades executadas, o modelo utilizado e os resultados.

O modelo de avaliação definido na ISO/IEC TR 15504 está baseado num modelo de referência que provê uma base comum para execução de avaliações de capacidade de processo de software, permitindo o relato de resultados usando uma escala de taxaço (avaliação) comum [ISO15504, 1998].

O modelo de referência define um modelo de capacidade de processo de duas dimensões. Na primeira dimensão, nomeada "dimensão processo", os processos associados com software são definidos e classificados em cinco categorias: Cliente-Fornecedor (CUS²), Engenharia (ENG), Suporte (SUP), Gerenciamento (MAN) e Organização (ORG). Na segunda dimensão, chamada dimensão "capacidade", uma série de atributos de processo agrupados em níveis de capacidade são definidos. Os atributos de processo provêm as características mensuráveis de capacidade de processo. Uma visão geral do modelo de referência pode ser vista na Figura 2.3.

¹ *Taxaço de processo* é o valor que lhe é atribuído durante a realização de uma avaliação.

² As siglas não foram traduzidas para facilitar a ligação com a ISO/IEC TR 15504, que não possui correspondente em português. (CUS - Customer-Supplier, ENG - Engineering, SUP - Support, MAN - Management e ORG - Organization)

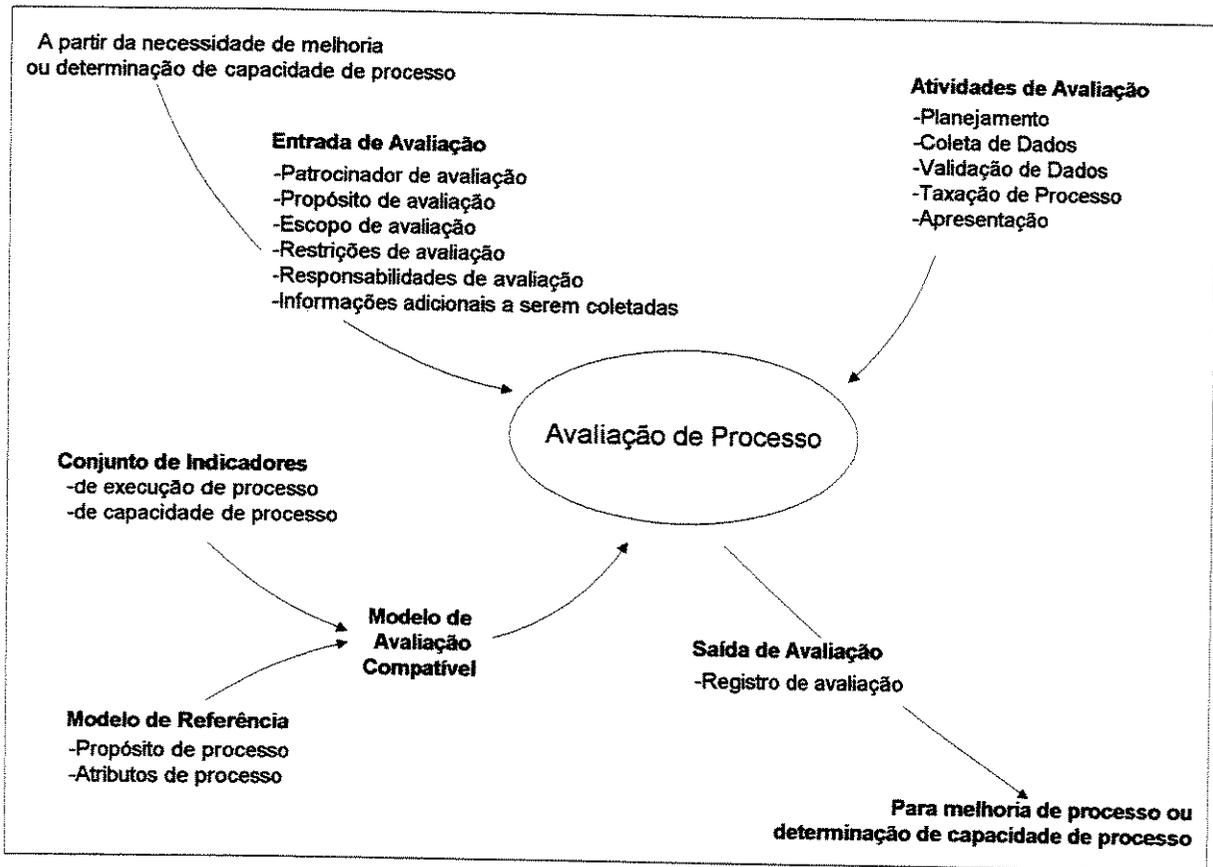


Figura 2.2 Contexto de avaliação de processo [ISO15504, 1998]

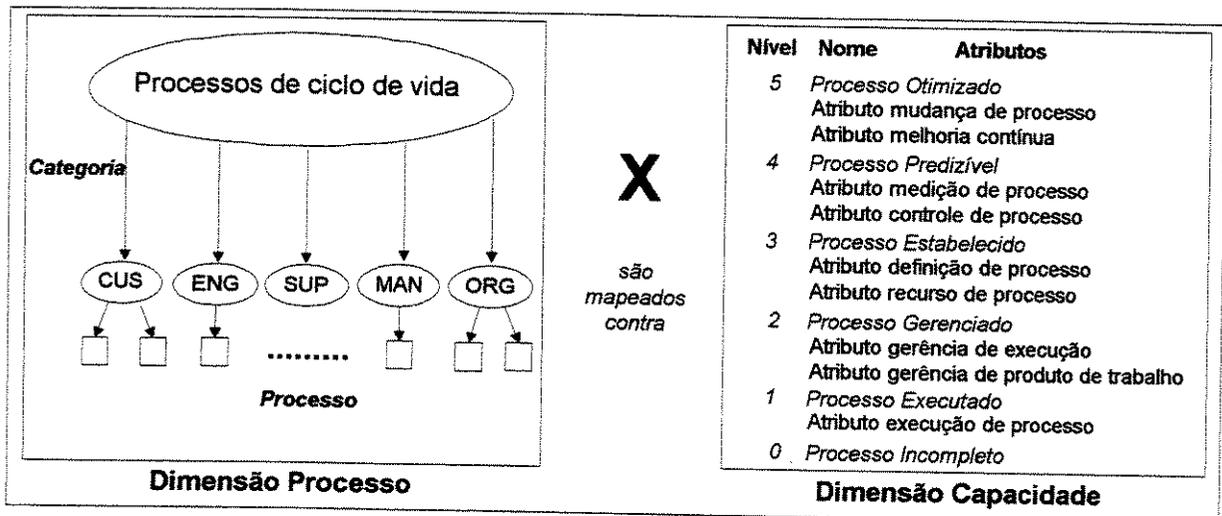


Figura 2.3 Dimensões do Modelo de Referência [ISO15504, 1998]

O modelo de referência não pode ser usado sozinho como base para condução de avaliações confiáveis e consistentes de capacidade de processo, porque o nível de detalhe fornecido não é suficiente. As descrições de propósito e atributos de processo precisam ser suportados com conjuntos abrangentes de indicadores de execução e capacidade de processo.

O modelo de avaliação expande o modelo de referência pela adição da definição e uso de indicadores de avaliação como mostra a Figura 2.4. O modelo de avaliação está baseado sobre o princípio de que capacidade de processo pode ser avaliada pela demonstração da obtenção de atributos de processo. Cada processo na "dimensão processo" tem um conjunto de práticas básicas associadas. *Práticas básicas* são atividades de engenharia de software que contribuem para a geração de saídas (resultados) de processo ou para aumentar a capacidade de um processo de software [ISO/IEC15504, 1998]. A execução destas práticas provê uma indicação da extensão da obtenção do propósito do processo. Similarmente, cada atributo de processo na "dimensão capacidade" tem um conjunto de práticas de gerenciamento associadas. A execução desta práticas provê uma indicação da extensão da obtenção do atributo no processo instanciado.

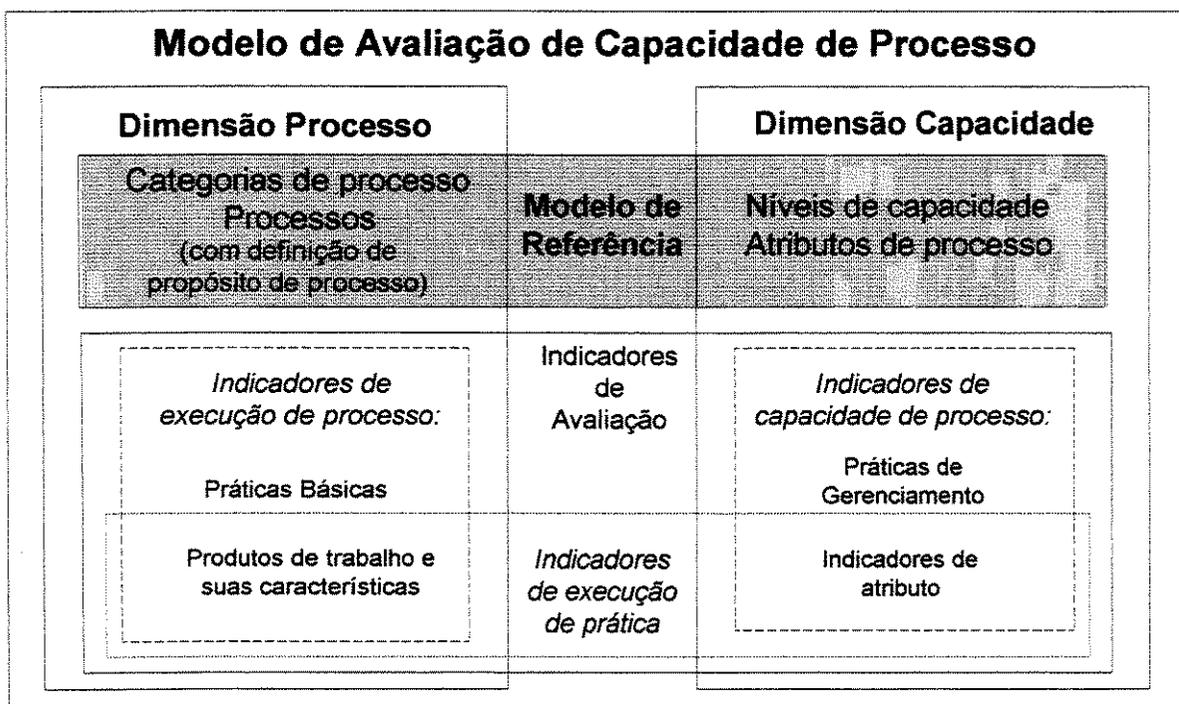


Figura 2.4 Modelo de avaliação da ISO/IEC TR 15504 [Emam et al., 1998]

Os indicadores definidos no modelo de avaliação representam tipos de evidência objetiva¹ que poderiam ser encontrados em uma instanciação de um processo e portanto poderiam ser usados para julgar obtenção de capacidade.

Os indicadores de execução de processo, além das práticas básicas, também compreendem os produtos de trabalho² de entrada e saída e suas características. Estes indicadores endereçam explicitamente a obtenção do propósito do processo.

¹ *Evidência objetiva* é informação qualitativa ou quantitativa, registro ou declarações de fatos relativos a características de um item ou serviço, ou à existência e implementação de um elemento de processo, que está baseado sobre observação, medida ou teste e que pode ser verificado.

² *Produto de trabalho* é um artefato associado com a execução de um processo.

As práticas básicas e os produtos de trabalho são indicadores de uma execução de processo de nível 1. A presença dos produtos de trabalho com a existência de suas características e evidência de execução de práticas básicas provêm evidências objetivas da obtenção do propósito do processo.

Os indicadores de capacidade de processo, além das práticas de gerenciamento, compreendem:

- características de execução de prática que provêm guia sobre a implementação da prática;
- características de recurso e infra-estrutura que provêm mecanismos para auxiliar no processo de gerenciamento; e
- processos associados da "dimensão processo" que suportam prática de gerenciamento.

O conjunto de práticas de gerenciamento pretende ser aplicável para todos os processos da "dimensão processo" do modelo de referência. Evidência da execução das práticas de gerenciamento definidas pode ser derivada das características de execução da prática. As características de execução da prática e características de recurso e infra-estrutura auxiliam a estabelecer evidência objetiva da extensão de obtenção do atributo de processo definido.

2.4.1 Modelo de referência - dimensão *Processo*

O modelo de referência agrupa os processos na "dimensão processo" em três agrupamentos de processos de ciclo de vida os quais contêm cinco categorias de processo, de acordo com o tipo de atividade que eles endereçam. Estes agrupamentos são apresentados na Figura 2.5. Os processos são identificados pela sigla da categoria, por um número sequencial dentro da categoria e pelo nome do processo.

Os *Processos Primários do Ciclo de Vida* consistem das categorias de processos **Cliente-Fornecedor** (CUS) e **Engenharia** (ENG) que são descritas como segue:

- a categoria de processo **Cliente-Fornecedor** consiste de processos que impactam diretamente o cliente, suportam desenvolvimento e transição do software para o cliente e fornecem subsídios para operação e uso corretos do produto e/ou serviço de software;
- a categoria **Engenharia** consiste de processos que especificam, implementam ou mantêm diretamente o produto de software, sua relação para o sistema e a sua documentação relacionada.

Os *Processos de Suporte do Ciclo de Vida* consistem da categoria de processo **Suporte** (SUP) e esta categoria é descrita como segue:

- a categoria **Suporte** consiste de processos que podem ser utilizados por qualquer dos demais processos (incluindo outros processos de suporte) em vários pontos do ciclo de vida de software.

Os *Processos Organizacionais do Ciclo de Vida* consistem das categorias de processo **Gerenciamento** (MAN) e **Organização** (ORG) que são descritas como segue:

- a categoria **Gerenciamento** consiste de processos que contêm práticas de natureza genérica que podem ser usadas por qualquer pessoa que gerencie qualquer tipo de projeto ou processo dentro de um ciclo de vida de software;
- a categoria **Organização** consiste de processos que estabelecem as metas de negócio da organização e que desenvolvem processos, produtos e ativos de recursos que auxiliarão a organização a obter suas metas de negócio, quando usados por seus projetos.

Cada processo no modelo de referência é descrito em termos de uma afirmação de propósito. Estas afirmações compreendem os objetivos funcionais únicos do processo quando instanciado em um ambiente particular. A afirmação de propósito inclui identificação de material adicional, as saídas de implementação bem sucedida do processo. A satisfação do propósito de um processo representa o primeiro passo na construção de capacidade de processo.

O modelo de referência não define como ou em que ordem os elementos das afirmações de propósito de processo são alcançadas. Os propósitos do processo serão alcançados em uma organização através de várias atividades, tarefas e práticas detalhadas sendo executadas para produzir produtos de trabalho. Estas tarefas, atividades e práticas executadas e as características dos produtos de trabalho produzidos são os indicadores que demonstram onde o propósito do processo está sendo alcançado.

2.4.2 Modelo de referência - dimensão *Capacidade*

A dimensão *Capacidade* é expressa, no modelo de avaliação, em termos de atributos de processo que são agrupados em níveis de capacidade de processo, como mostra a Figura 2.3.

Capacidade de processo é definida sobre uma escala ordinal de seis pontos que habilita a capacidade ser avaliada a partir do início da escala, **Incompleto**, até o topo final da escala, **Otimizado**. A escala representa aumento de capacidade de processo executado a partir da execução *que não é capaz* de obter resultados de processo para a execução *que é capaz* de reunir metas de processo e de melhoria relevantes, as quais são explicitamente derivadas das metas de negócio da organização. Portanto, a escala define uma rota bem-definida para melhoria de cada processo individual.

A medida de capacidade está baseada em um conjunto de atributos de processo que são usados para determinar se um processo atingiu uma dada capacidade. Os atributos são medidos sobre uma escala de porcentagem e portanto provêm um discernimento mais detalhado nos aspectos específicos de capacidade de processo requerida para suportar melhoria e determinação de capacidade de processo.

Os níveis de capacidade são descritos como segue:

- *Nível 0: Processo Incompleto*, neste nível o processo não está implementado ou falha para obter seus resultados; existe pouca ou nenhuma evidência de qualquer obtenção sistemática de qualquer dos atributos definidos;

- *Nível 1: Processo Executado*, o propósito do trabalho geralmente é obtido; não existe planejamento ou acompanhamento para obtenção do propósito; existem produtos de trabalho para o processo identificáveis e estes testemunham o alcance do propósito; o atributo de processo definido é execução de processo;
- *Nível 2: Processo Gerenciado*, além de obter o descrito no nível 1, o processo transporta produtos de trabalho de qualidade aceitável dentro de escalas de tempo definidas; a execução de acordo com procedimentos especificados é planejada e acompanhada; produtos de trabalho estão em conformidade com os padrões e requisitos especificados; os atributos definidos são gerenciamento de execução e gerenciamento de produto de trabalho;
- *Nível 3: Processo Estabelecido*, além de obter o descrito no nível 2, o processo é executado e gerenciado usando um processo definido; este processo é seguido por toda a instituição; os atributos definidos para este nível são definição de processo e recurso de processo;
- *Nível 4: Processo Predizível*, além de obter o descrito no nível 3, o processo definido é executado na prática, dentro de limites de controle definidos, para alcançar suas metas; a execução é objetivamente gerenciada e a qualidade dos produtos de trabalho é conhecida quantitativamente; medidas detalhadas de execução são coletadas e analisadas; os atributos definidos para este nível são medição de processo e controle de processo;
- *Nível 5: Processo Otimizado*, além de obter o descrito no nível 4, a otimização do processo consiste na condução de idéias e tecnologias inovativas e em mudanças no processo para alcançar metas ou objetivos definidos; os atributos definidos para este nível são mudança de processo e melhoria contínua.

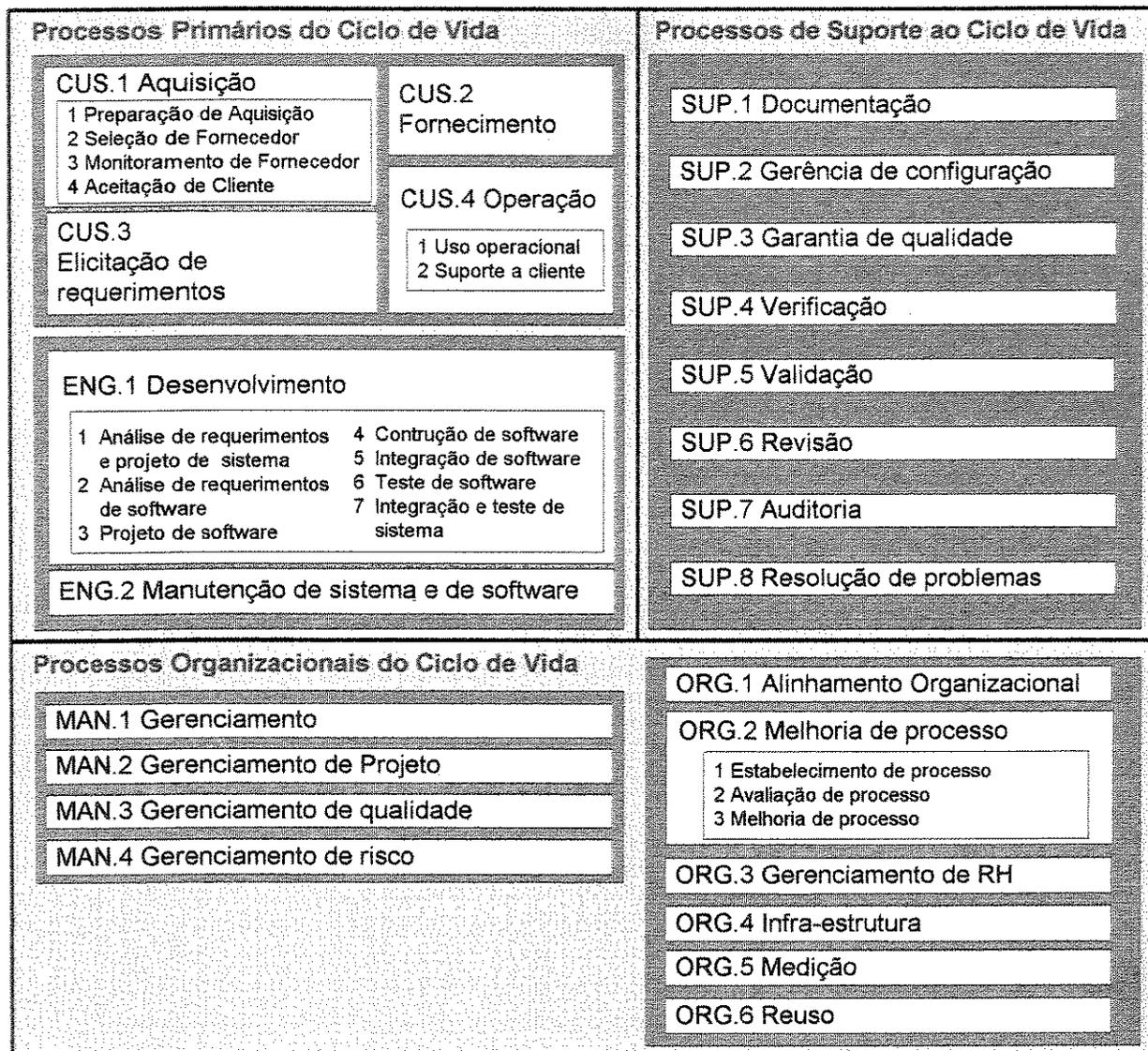


Figura 2.5 Os processos da Dimensão Processo [ISO15504, 1998]

3 Organização por projeto

O termo *projeto* é entendido como um conjunto de ações executadas de forma coordenada por uma organização transitória, ao qual são alocados os insumos necessários para, em um dado prazo, alcançar um objetivo determinado [Valeriano, 1998].

Na indústria de software, normalmente utiliza-se uma organização por projeto na área técnica. Esta organização constitui-se de uma equipe inteiramente dedicada a um projeto, cujos membros são temporariamente desvinculados da estrutura departamental, passando à coordenação de um gerente de projeto. A *gerência de projeto* é uma função administrativa que, de forma sistemática, planeja, decide, põe em prática as ações conseqüentes e utiliza os meios para alcançar seus objetivos.

Os itens a seguir apresentam a visão geral de um modelo de gerência de projeto e de uma ferramenta para planejamento e controle de projeto, a EDT - Estrutura de Decomposição de Trabalho.

3.1 Gerência de projeto

O projeto, por ser um processo com duração finita e ter de atingir um objetivo em um determinado prazo, tem início e fim, passando por algumas fases que constituem o ciclo de vida do projeto. Segundo Valeriano [Valeriano, 1998], convencionou-se chamar de *Ciclo de Vida Genérico de Projeto* o ciclo que consiste de quatro fases (veja Figura 3.1): *Conceptual*¹; de Planejamento e Organização; de Implementação; de Encerramento.

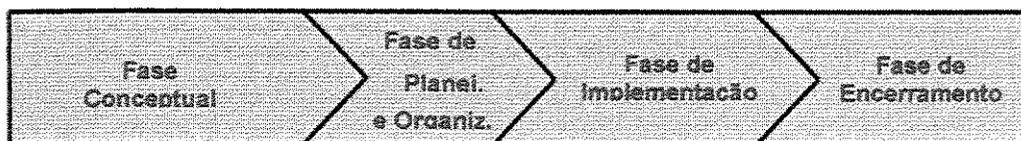


Figura 3.1 Fases do Ciclo de Vida Genérico de Projeto [Valeriano, 1998]

A fase *Conceptual* inclui atividades que vão desde a idéia inicial do produto ou assunto a pesquisar, passando pela elaboração de uma proposta para a execução de um projeto (planejamento preliminar) e chegando até a aprovação desta proposta.

A fase de *Planejamento e Organização* tem por objetivo elaborar o planejamento detalhado da proposta de projeto aprovada e estabelecer a sua organização - condições básicas para a execução que se segue - de forma a permitir a ação descentralizada dos executantes, a articulação entre eles bem como um efetivo controle.

A fase de *Implementação* consiste em executar o que foi planejado e ajustar a execução de forma a manter os esforços dirigidos no sentido da consecução do objetivo, nas condições

¹ *Conceptual*: próprio para a concepção.

planejadas. Esta fase também compreende um complexo conjunto de habilidades do gerente e dos executores, um elevado espírito de cooperação, muita coordenação e competência da equipe.

A fase de *Encerramento* efetiva a transferência dos resultados do projeto, com aceitação do seu cliente, seguida de uma avaliação geral do projeto e, por fim, da desmobilização dos meios e recursos postos à disposição deste projeto.

Para Valeriano [Valeriano, 1998], esta seqüência abriga tanto projetos cujos objetivos são itens materiais e/ou "softwares" (por exemplo, os projetos de pesquisa & desenvolvimento), como também respostas a questões técnicas ou científicas (por exemplo, os projetos de pesquisa). O detalhamento destas fases é que deverá amoldar-se ao tipo de projeto, sua natureza, dimensão, grau de complexidade, etc. As fases não são estanques nem totalmente sucessivas, elas se sobrepõem, como pode ser visto na Figura 3.1.

3.2 Estrutura de Decomposição do Trabalho - EDT

A ferramenta *Estrutura de Decomposição do Trabalho - EDT* apóia o planejamento, execução e controle de um projeto. A EDT é uma forma de apresentação do projeto que o explicita em suas partes físicas, em softwares, serviços e outros tipos de trabalho, a qual organiza, define e mostra graficamente tanto o produto a ser feito como o trabalho a ser realizado para obtê-lo [Valeriano, 1998].

A EDT oferece uma linguagem comum, simples e racional para todos os participantes do desenvolvimento do projeto e constitui-se em um ponto-chave para todo o prosseguimento dos trabalhos. Ela é resultado de um esforço cooperativo executado na fase de planejamento preliminar pelo gerente do projeto, as demais partes envolvidas e os especialistas (futuros membros da equipe de desenvolvimento). Pode ser apresentada de duas maneiras:

- sob a forma de um organograma, também conhecida como "árvore de decomposição do projeto", recomendada para uso em integração e montagens, interfaces, relacionamentos e interdependência de especificações, entre outros; ou
- como uma relação ou tabela, com seu uso mais indicado para cronogramas e orçamentos, distribuição de pessoal e de material, descrições de tarefas, entre outros.

A EDT como árvore de decomposição tem cada uma de suas partes em detalhamentos sucessivos, representada por um retângulo ou "bloco" tendo a aparência de uma árvore invertida, como apresentada na Figura 3.2.

Os blocos da EDT são as peças contrutivas da EDT e sua caracterização é objeto do subitem seguinte. São chamados de "blocos" especialmente quando se trata da forma visual com que se apresentam: retângulos interligados em uma árvore de decomposição. Mas também são chamados de "pacotes de trabalho" ou de "tarefas" quando o foco está mais voltado para as atribuições ou responsabilidades dadas a seus executores. Com estas nuances, as três expressões serão empregadas quase que indistintamente neste texto: blocos, pacotes de trabalho e tarefas.

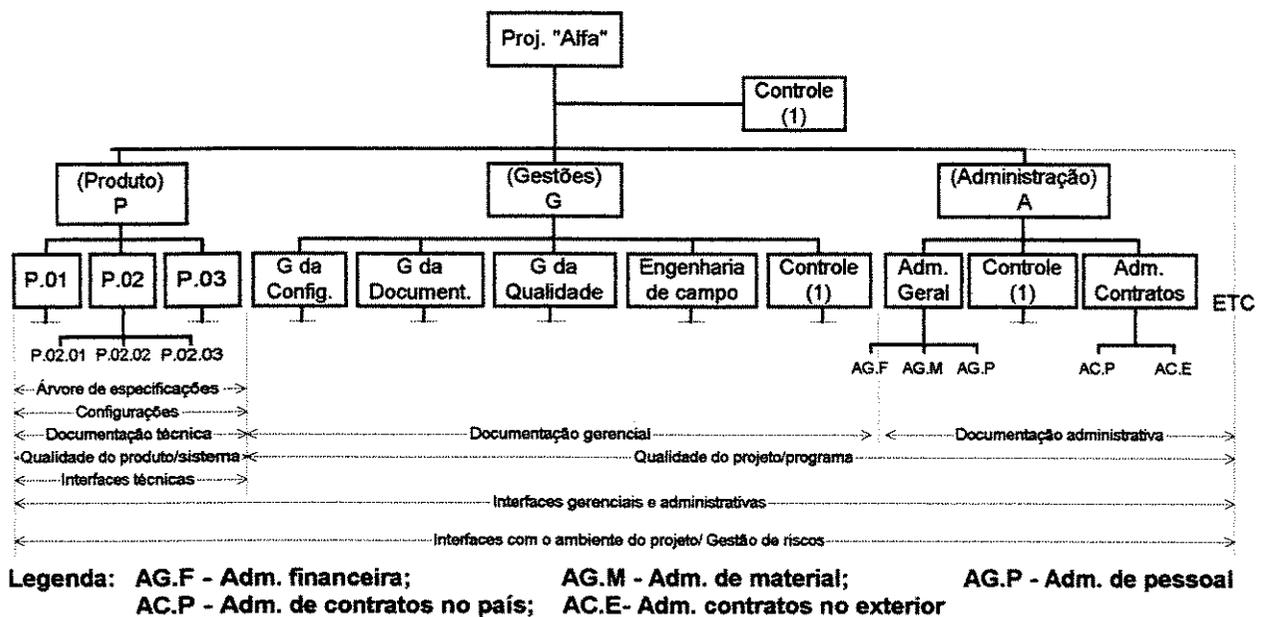


Figura 3.2 - Árvore de decomposição do projeto "Alfa" [Valeriano, 1998]

A EDT pode ser construída com os seguintes módulos:

- *estrutura de decomposição de produto - EDP*, que representa o produto dividido em suas partes lógicas, interligadas hierarquicamente, com os itens ou componentes mais elementares na parte inferior da EDP, sendo sucessivamente integrados até se chegar ao produto final (veja, na Figura 3.2 o módulo da EDP, a partir do bloco "P");
- *módulo de gestões específicas*, que contém o conjunto de gerências parciais ou especializadas, tais como qualidade, configuração, documentação, etc.; o gerente de projeto poderá agrupar estas gestões específicas em um módulo paralelo à EDP, como mostra a Figura 3.2, Módulo "G";
- *módulo administrativo*, que cuidará da parte administrativa, ou seja, para fazer as ligações com diversos órgãos para tratar do fluxo de recursos materiais e financeiros, de pessoal, relatórios e prestações de conta, etc.; o gerente pode dar essas atribuições executivas a um "bloco" logo no segundo nível, subdividindo-o como conveniente, como mostra a Figura 3.2, módulo "A".

A Figura 3.2 mostra a árvore de decomposição expandida em três e quatro níveis. Observa-se que a estrutura do projeto contém a EDP (objetivo do projeto) e também a estrutura da administração do projeto, quando aplicável, ou seja, quando o projeto tiver encargos administrativos referentes a pessoal, material, finanças, contrato, etc., a ponto de ser necessária uma estrutura administrativa própria e uma estrutura das gestões, se o gerente de projeto decidir descentralizar tarefas gerenciais.

Ainda na Figura 3.2, observa-se que o gerente do projeto pode localizar o *Controle* no bloco das gestões (se preferir estruturar um controle especial para o projeto e atribuí-lo a um

líder). Se preferir, pode ter o controle mais diretamente ligado a si próprio e pode ainda, atribuí-lo a um bloco da administração.

Ao estabelecer a EDT, deve-se ter em vista que: ela é resultado de um trabalho de equipe, em que todos os aspectos do projeto devem estar representados; ela deve explicitar a estrutura de decomposição do produto e as tarefas técnicas, gerenciais e administrativas necessárias para obtê-lo; não deve refletir a estrutura da organização nem decompor o produto em disciplinas; ela é determinada com a finalidade básica de descentralizar o gerenciamento do projeto de forma lógica e racional; dela decorrem o gerenciamento e as tarefas pertinentes a todas as demais áreas de tratamento específico do projeto.

A elaboração da EDT decorre da harmonização e da consolidação dos resultados de estudos efetuados em quatro áreas altamente interativas, três das quais são de análise e a última é um conjunto de sínteses:

- decomposição do produto sucessivamente em suas partes constitutivas - EDP;
- determinação dos blocos administrativos e gerenciais;
- elaboração da declaração de trabalho dos blocos;
- consolidações de dados: (a) da EDT; (b) do orçamento-mestre; (c) do cronograma-mestre; e (d) do escopo/declaração de trabalho

3.2.1 Estrutura de decomposição de produto - EDP

A estrutura de decomposição de produto - EDP é composta por:

- um bloco inicial, que representa o sistema/produto integrado, pronto para operar; suas características, em geral, são definidas pelo usuário/cliente em termos de necessidades;
- blocos intermediários, que são decompostos em suas partes constituintes e caracterizam-se, essencialmente, por suas atividades de montagens, integração e ensaios de subconjuntos;
- blocos elementares, que são os blocos dos últimos níveis e devem:
 - ter um objetivo definido, com requerimentos para seu recebimento ou aceitação;
 - ter um prazo a ser cumprido, definido por datas de início e de término;
 - ter apenas um responsável;
 - ser atribuídos a apenas uma unidade organizacional;
 - ter seu orçamento próprio, sendo os menores centros de custo;
 - se aplicáveis, ter eventos-marco, representativos do progresso.

O tamanho e a duração dos blocos elementares deverão ser tais que representem pequenas partes do esforço para que o término destes pequenos blocos conduza a freqüentes obtenções de valores de custos e avaliações dos prazos cumpridos no projeto, bem como razoável número de

verificações físicas, de modo a permitir um efetivo controle. Por outro lado não deverá haver exagerada e injustificada fragmentação para não acarretar excessivo e desnecessário trabalho gerencial e conseqüente elevação de custos e tempo.

A definição dos blocos elementares é importante quanto aos aspectos de execução e de controle. Neste ponto é que será executado o trabalho previsto e é onde também será exercido o controle de desempenho, de custos e de prazos. Os demais blocos dos níveis superiores serão formados pelas integrações sucessivas, a partir destes blocos elementares.

O desenvolvimento da EDP deve: estabelecer os requerimentos correspondentes a seus itens ou componentes para cada bloco resultante da decomposição, à medida que os componentes do produto vão sendo desagregados; definir soluções para os blocos e seus respectivos processos, técnicas e/ou materiais necessários ou disponíveis para que sejam concretizados para atender aos requerimentos estabelecidos; identificar os executantes e um responsável pela tarefa; levantar e organizar dados de custos /prazos; realizar mentalmente a integração.

Segundo Valeriano [Valeriano, 1998], na elaboração da EDP são importantes algumas observações, como:

- 1) cada nível deve originar requerimentos de sua necessidade e não soluções a serem supridas pelos níveis que lhe sucedem;
- 2) nenhum requerimento deve ser atribuído a qualquer item sem que tenha sido exigido por requerimento de nível superior, ou deste decorrente; apesar de evidente, esta recomendação costuma ser negligenciada, surgindo requerimentos autônomos, independentes e, portanto, supérfluos;
- 3) todos os requerimentos, em quaisquer níveis, devem ser rastreáveis até o requerimento de mais alto nível: o da missão do sistema, por exemplo; esta exigência pode ser violada, desde cedo, pela inobservância da recomendação anterior e também pela evolução dos trabalhos, quando ajustes e compromissos promovem, por erro ou falha no processo, a perda de identidade de requerimentos, o que deve ser evitado por meio de uma competente gestão de configuração.

3.2.2 Blocos administrativos e gerenciais

O tarefas administrativas e também as de gestões específicas vão sendo agregadas à EDT - à medida que a determinação da EDP vai sendo desenvolvida - e posteriormente desmembradas em suas componentes menores: serviços técnicos (p.ex.: consultoria, dimensionamentos, ensaios, montagens e integrações), serviços administrativos (p.ex.: contratos, compras), controles (p.ex.: físicos, financeiros, cronológicos).

Os cuidados a tomar durante a elaboração da EDP são válidos para os módulos da administração e para os gerenciais: não se deve criar tarefa que não seja justificada por necessidade de tarefa (ou bloco) de nível superior; todas as tarefas devem ser rastreáveis ao objetivo do projeto e devem concorrer para sua consecução.

Também é recomendado por Valeriano [Valeriano, 1998] que se proceda à integração mentalizada, como na EDP, para os blocos administrativos e de gestões específicas estabelecidos pelo gerente de projeto.

3.2.3 Declaração de trabalho de bloco

A declaração de trabalho de um bloco contempla informações referentes tanto ao aspecto de gerenciamento/administração do projeto como aos aspectos básicos do projeto (resultados, prazos e custo). A Tabela 3.1 apresenta as informações necessárias para a declaração de um bloco agupadas por aspecto.

Tabela 3.1 - Informações de declaração de bloco		
Aspectos	Informações	Observações
gerenciamento	<ul style="list-style-type: none"> identificação responsável 	dados para controle
resultado	<ul style="list-style-type: none"> descrição sintética da meta a atingir resultados especificações ou outros documentos aplicáveis condições do recebimento (medidas de desempenho técnico e operacional) 	descrição e desempenho
prazos	<ul style="list-style-type: none"> duração 	será transformada em datas de início e término
custos	<ul style="list-style-type: none"> insumos a receber de outros blocos e da organização (pessoal, material, informações, serviços, etc.) 	serão transformados em custos

Da árvore de composição do projeto "Alfa", apresentada na Figura 3.2, destaca-se o bloco P.02.01, conforme mostra a Figura 3.3. A indicação das informações da declaração de trabalho deste bloco pode ser vista na Figura 3.4.

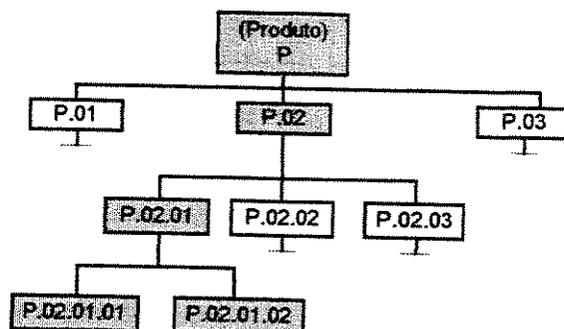


Figura 3.3 - Estrutura de decomposição do Produto P [Valeriano, 1998]

Verifica-se que insumos, resultados e interfaces serão os pontos que precisarão da interveniência do gerente de um dado nível nas tarefas do nível inferior: administrar o fluxo dos insumos, dar como cumprida a tarefa de aceitar (receber) o produto de cada um dos blocos constituintes e administrar ou arbitrar os conflitos que por ventura surjam nas interfaces.

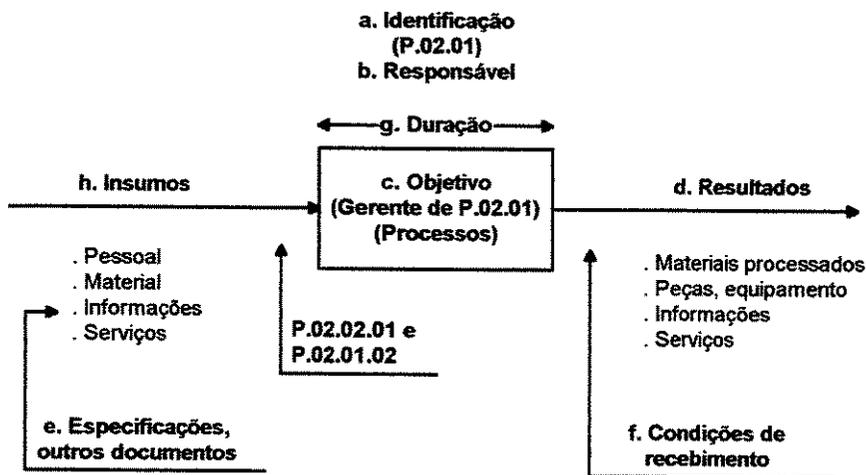


Figura 3.4 - Esquema da declaração de trabalho de um bloco [Valeriano, 1998]

Observa-se na Figura 3.4 que:

- o gerente de P.02.01 é responsável pela execução das tarefas de seu bloco (processos) para atingir o objetivo (c) e apresentar seus resultados (d), nos prazos estabelecidos (g);
- o gestor de insumos providencia o fornecimento de insumos (h);
- o gerente de P.02 controla as interfaces de P.02.01 e faz o controle de recebimento de P.02.01 (f);
- a equipe de planejamento estabelece e ajusta as especificações (e) e as condições de recebimento (f);
- o gerente de P.02.01 controla o recebimento de P.02.01.01 e de P.02.01.02 e também recebe os produtos destes itens.

3.2.4 Consolidação da EDT

Nesta atividade, devem ser consolidados: a EDT (declarações de trabalho de bloco e requerimentos); os prazos (cronograma-mestre); os custos (orçamento-mestre); e o escopo e a declaração de trabalho. A seguir, estas atividades são detalhadas.

Consolidação da EDT: declarações de trabalho de bloco

A declaração de trabalho especifica claramente o responsável pela tarefa atribuída ao bloco, seja ele um órgão da própria organização (projeto, compra, etc.), seja uma entidade contratada/fornecedora, o próprio cliente, uma pessoa da equipe do projeto, etc.

A consolidação das declarações de trabalho dos blocos é feita pela geração da *Matriz de responsáveis/tarefas* e da *Matriz de contratos*.

A *Matriz de responsáveis/tarefas* é uma forma de integração de responsabilidades para execução por equipe, e a Figura 3.5 mostra um exemplo.

Tarefa \ Responsáveis	R1	R2	R3	R4	R5	R6
P	G							
- Especificação	A							
- Integração					E	E		
- Ensaio						N	E	
P.01		G						
- Pedido de serviço				N				E
- Integração					E			
- Tarefa X					N		E	
- Tarefa Y					N			E
P.01.01			G				E	
- Etc.								
- Etc.								E
P.01.02								
- Etc.								
- Etc.								

Legenda: R - responsável G - Gerencia E - Executa A - Aprova N - É notificado / cópia

Figura 3.5 Matriz de responsáveis / tarefas [Valeriano, 1998]

A *Matriz de controle de contratos* relaciona os menores níveis de decomposição do projeto com os níveis executores das organizações por eles responsáveis, quando ocorrer a modalidade de "contratações".

Um exemplo simplificado de matriz de controle de contratos é mostrada na Figura 3.6, o qual contém apenas as partes de P.02.

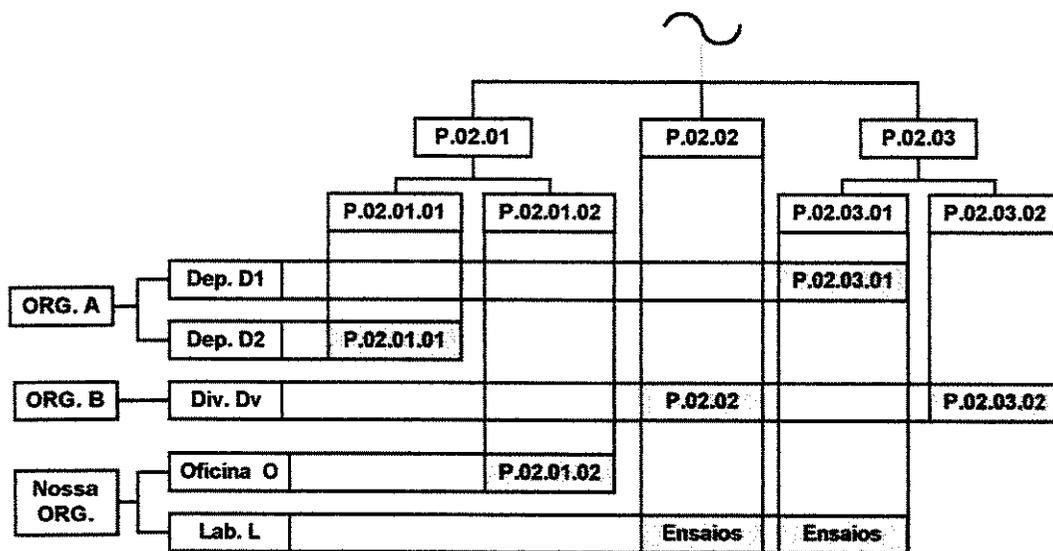


Figura 3.6 A matriz de controle de contrato [Valeriano, 1998]

Consolidação da EDT: requerimentos - árvore de especificações

A consolidação de requerimentos usa a hierarquização do produto, como estabelecida na EDT, para relacionar, da mesma forma, os requerimentos e as especificações do produto e de suas partes constitutivas. O resultado é o que se chama de árvore de especificações, que vem a ser uma estruturação dos documentos normativos do produto, mostrada em uma forma homóloga à da EDP [Valeriano, 1998].

Ao se definir o objetivo de um projeto, ainda que de forma rudimentar ou embrionária, há uma especificação para o produto, representada pelas características, condições ou requerimentos que o mesmo deve observar. Assim, muitos projetos são iniciados, com base nas especificações ditas preliminares e que vão evoluindo, sendo detalhadas e aperfeiçoadas até que se chegue às especificações finais ou definitivas. Ao se declarar o objetivo, muitos dos requerimentos podem estar implícitos, mas no decorrer da execução, eles devem ser tornados explícitos e detalhados.

É comum a introdução de alterações sucessivas nas especificações, ajustando e refinando os requerimentos das partes do produto, cujo controle é objeto e preocupação da gestão de configuração. É óbvio que os documentos da árvore de especificações (Especificações técnicas e Normas de interface, Procedimentos de montagem etc.) têm de ser constantemente compatíveis entre si, ainda que necessitassem ser revistos e modificados no decorrer do projeto. A Figura 3.7 mostra uma árvore de especificações que contém, para simplificar, apenas um documento para cada parte do produto, podendo vir a tê-los desmembrados ou incorporados, de acordo com a conveniência.

Na Figura 3.7 podem ser identificados:

- a especificação do produto/sistema;

- as especificações de desempenho de partes;
- as especificações de detalhes de partes;
- uma especificação de processo.

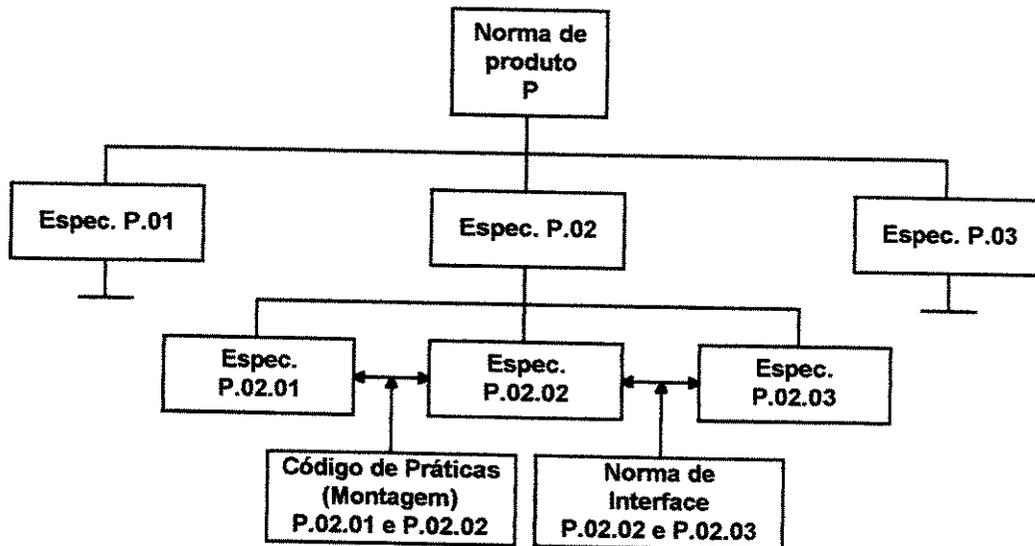


Figura 3.7 Uma árvore de especificações [Valeriano, 1998]

Consolidação de prazos

A consolidação de prazos destinada à obtenção dos cronogramas consiste em:

- levantar ou avaliar as durações das tarefas dos blocos da EDT, formulando três hipóteses para a duração da tarefa: a otimista (quando se espera que tudo vai dar certo); a mais provável (uma hipótese conservadora, cujo valor não é, necessariamente, a média dos dois extremos); e a pessimista (quando deverá prevalecer a Lei de Murphy e seus corolários); chamando-se o , m e p aos valores correspondentes às hipóteses e adotadas suposições simples, calculam-se:

- o valor esperado, d para a duração da tarefa:

$$d = (o + 4m + p)/6$$

- o desvio padrão s :

$$s = (p - o)/6$$

- relacionar as tarefas umas às outras, consideradas as precedências e condicionantes existentes, isto é, obter uma rede de precedência;
- montar um cronograma-mestre, ou seja, "amarrar ao calendário" o diagrama de precedência das tarefas de maior nível do projeto para que se tenha uma visão conjunta do projeto quanto ao que deve suceder no decorrer do tempo; este cronograma-mestre

geralmente é mostrado na forma de um diagrama de barras ou gráfico de Gantt, do qual fluirão todos os outros, referentes às partes do projeto, em seus vários níveis e graus de detalhamento; a Figura 3.8 mostra um trecho de gráfico de Gantt que refere-se à Figura 3.3.

- organizar os outros cronogramas parciais a partir do cronograma-mestre, como por exemplo: cronogramas de tarefas ou blocos, pelos quais indicam-se os inícios e terminos das tarefas, podendo incluir eventos no decorrer das mesmas; cronogramas de pessoas, em que figuram as dedicações ou comprometimentos de cada participante nas diversas tarefas, com as respectivas datas de ocupação; cronogramas de materiais, com datas em que devem ser entregues os materiais de que o projeto necessita (máquinas, equipamentos, softwares, etc.); e cronograma de custos, com a previsão do fluxo de recursos financeiros (valores, datas de desembolso e utilização previstas e tarefas em que serão empregados).

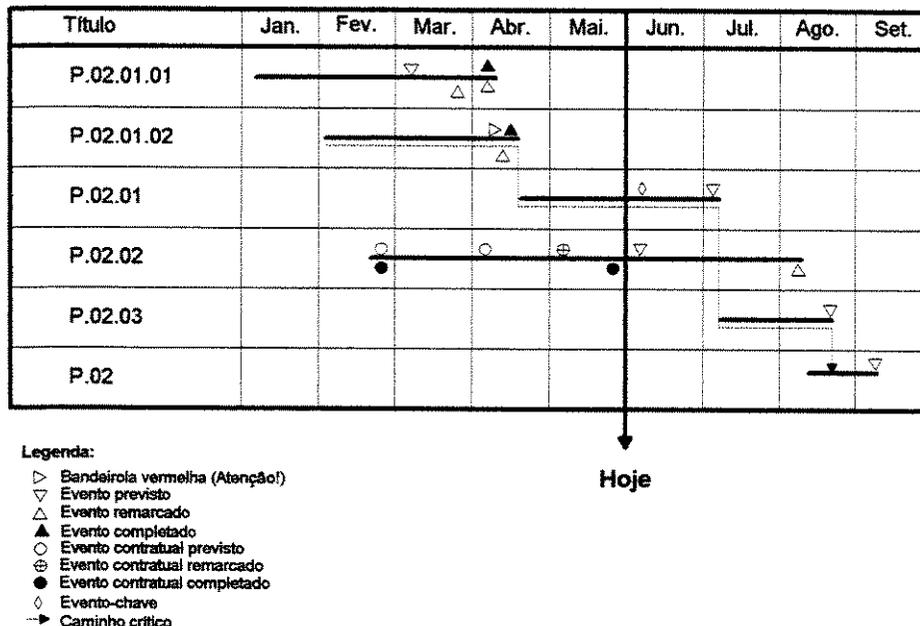


Figura 3.8 Um trecho do gráfico de Gantt [Valeriano, 1998]

Consolidação de custos

A consolidação de custos consiste na geração de um plano de contas que apresente os recursos orçamentários para todo o período de duração do projeto. Detalhar os custos discriminando os valores por natureza da despesa, segundo as normas de orçamentação vigentes.

Consolidação de escopo/declaração de trabalho

A consolidação de escopo consiste da compatibilização do escopo com as declarações de trabalho dos blocos para garantir consistência de informações.

4 Metodologia

Para atingir o objetivo de descrever processos de software, este trabalho adotou o método de Gestão de Processo (descrito no Capítulo 2), focalizando nas atividades e ferramentas da etapa de Planejamento e Organização.

A seqüência de atividades executadas foi a seguinte:

- *identificação de processos*, fundamentada no modelo de referência para processos e capacidade de processo da ISO/IEC TR 15504, como mostrado na Figura 2.5;
- *priorização de processos*, que considerou prioritários os processos que contribuem diretamente para amenizar os seguintes problemas:
 - não entender bem o quê o cliente precisa;
 - existência de grandes distorções de custo e prazo, em relação ao planejado;
 - dispêndio de muito esforço em retrabalho;
 - existência de pouca documentação de desenvolvimento e/ou de usuário ou documentação desatualizada;
- *descrição de processo prioritário*, utilizando-se as ferramentas: tabela de definição de escopo, macrodiagrama, fluxograma e detalhamento das atividades do fluxograma.

O formato das ferramentas e a origem das informações utilizadas na descrição de processo são apresentadas a seguir.

Definição de escopo

As informações da tabela de definição de escopo foram reunidas da *Parte 2 - Um modelo de referência para processos e capacidade de processo (normativo)* e da *Parte 5 - Um guia de modelo de avaliação e indicador (informativo)* da ISO/IEC TR 15504 [ISO15504, 1998]. A *Parte 2* identifica os processos, declara o seu propósito, indicando em alto nível os objetivos de sua execução, e declara os resultados observáveis de uma implementação com sucesso. A *Parte 5* é uma parte informativa da norma que contém para cada processo um conjunto de boas práticas de engenharia de software. Os anexos desta parte contêm, respectivamente, uma lista de produtos de trabalho de entrada e saída de cada processo e as características de cada produto de trabalho. As práticas básicas, os produtos de trabalho e suas características constituem os indicadores de execução de processo.

A tabela de definição de escopo segue o formato mostrado na Figura 4.1.

O campo *Nome do processo* contém o identificador da categoria do processo (CUS, ENG, SUP, MAN ou ORG), o número do processo dentro da categoria (veja Figura 2.5) e o nome por extenso. As duas primeiras informações seguem a nomenclatura da ISO/IEC TR 15504 em inglês para manter uma linguagem única que facilita a consulta à norma. Por exemplo, "*MAN.2 Processo de Gerenciamento de Projeto*" denota categoria Gerenciamento, processo número 2, nomeado Processo de Gerenciamento de Projeto. Informações deste campo foram obtidas da *Parte 2* da ISO/IEC TR 15504.

Nome do processo:	IDProc <nome processo>		
Objetivo:	< propósito deste processo >		
Entradas:	id) <Nome produto>	Saídas:	id) <Nome produto>
Início do processo:			
Conteúdo:	<conteúdo do processo>		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> • <resultado 1>; • <resultado 2>; • <iresultado n>. 		
Clientela a ser atendida:	< lista de clientes >		
Indicadores de desempenho:	Existência e adequação das práticas básicas aplicadas neste processo: < lista de práticas básicas > Existência dos produtos de trabalho de entrada e saída e suas características.		

Figura 4.1 Tabela de descrição de escopo de processo adaptada de [Embrapa, 1998b]

As *Entradas* e *Saídas* são representadas por produtos de trabalho que possuem uma identificação (id). Esta identificação permite localizá-los facilmente no Anexo I que relaciona as características-chave de todos os produtos de trabalho utilizados pelos processos descritos. Tanto os produtos de trabalho como as suas características foram obtidos da *Parte 5* da ISO/IEC TR 15504. Quando um produto de trabalho de entrada também está alocado como produto de saída indica que o processo atualiza este produto.

Os *Indicadores de desempenho* de execução de processo são as práticas básicas, os produtos de trabalho e as suas características descritos para o processo. Estes indicadores são utilizados para o *nível de capacidade 1 Processo executado* e a avaliação deve responder às questões: (1) a prática básica contribui para a meta do processo? (2) produtos de trabalho e suas características existem? (3) produtos de trabalho e suas características são adequados? Para outros níveis de capacidade têm-se outros indicadores definidos na ISO/IEC TR 15504. O conjunto de indicadores se torna mais elaborado de acordo com o objetivo do nível de capacidade desejado, buscando-se a melhoria contínua e gradual do processo.

Macrodiagrama

O macrodiagrama detalha a descrição do processo, dividindo-o em subprocessos e alocando a cada subprocesso, produtos de trabalho de entrada e saída, relacionando fornecedor e cliente dos produtos, respectivamente. O formato utilizado para o macrodiagrama está apresentado na Figura 4.2.

Este trabalho considera o termo subprocesso como sinônimo de prática básica, por isso a identificação dos subprocessos é obtida da ISO/IEC TR 15504, que indica para cada processo um conjunto de práticas básicas.

A alocação dos produtos de trabalho e a identificação de fornecedor e de cliente de processo é resultado da análise de material disponível na literatura de engenharia de software, portanto, uma das contribuições deste trabalho. Quando um produto de trabalho de entrada também está alocado como produto de saída indica que a prática básica atualiza este produto.

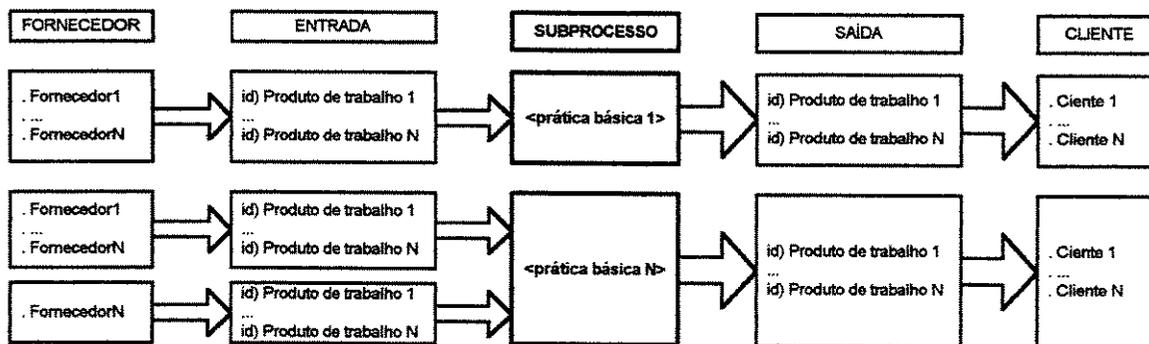


Figura 4.2 Formato de macrodiagrama de processo

A coluna *Fornecedor* identifica os fornecedores de insumos para o processo, como por exemplo, instituições, departamentos internos da empresa, outro processo, entre outros.

A coluna *Entrada* lista os produtos de trabalho de entrada de uma atividade (ou subprocesso) relacionados com cada fornecedor e seguem a nomenclatura utilizada na descrição do escopo do processo.

A coluna *Subprocesso* identifica atividade que possui interface ou interrelação com outro processo. Para manter uma linguagem única com a ISO/IEC TR 15504, as atividades descritas nesta coluna seguem a forma: CP.PR.BP.NP. Onde CP é a categoria do processo, PR é o número do processo, dentro da categoria, BP é o texto utilizado para significar prática básica (*Base Practice*) e NP é o número da prática básica dentro do processo. Por exemplo, "MAN.2.BP1" denota a prática básica número 1 (Definir o escopo de trabalho) na categoria de processo MAN (Gerenciamento) para o processo número 2 (Gerenciamento de projeto).

A coluna *Saídas* lista os produtos de trabalho de saída de um subprocesso utilizando a mesma nomenclatura da descrição do escopo do processo.

A coluna *Clientes* identifica os clientes do processo atendidos pela execução de um subprocesso e os relaciona com seus respectivos produtos de trabalho. Cliente é alguém ou algo que utiliza os resultados da atividade, pode ser função de pessoas (por exemplo, gerente de projeto), departamentos internos da empresa, outras instituições, outros processos, etc.

Cada subprocesso é representado por um retângulo de linha de contorno mais grossa que os demais. Para cada subprocesso são definidas as suas entradas e as suas saídas. Os fornecedores devem ser agrupados por produtos de entrada comuns.

Fluxograma

A ferramenta fluxograma representa graficamente a seqüência das atividades ou fases de um processo. Como a ISO/IEC TR 15504 não estabelece uma seqüência de execução das práticas básicas e identifica de forma bem genérica as atividades da prática, a elaboração do fluxograma, detalhando a seqüência de atividades de cada prática é uma contribuição deste trabalho.

A Figura 4.3 mostra um exemplo de fluxograma. Ele utiliza os símbolos descritos no subitem 2.2.1 *Gestão de processo*. O fluxograma é identificado pelo nome do processo que segue a mesma nomenclatura do campo *nome do processo* da tabela de definição de escopo. No exemplo da figura, tem-se *MAN.2 Gerência de projeto*. À esquerda do fluxograma estão definidos os responsáveis pelas atividades da prática básica (na figura, representado por *<responsável pelas atividades>*) e, em alguns casos, os responsáveis por um conjunto de atividades (na figura representado por *<responsável 1>* e *<responsável 2>*).

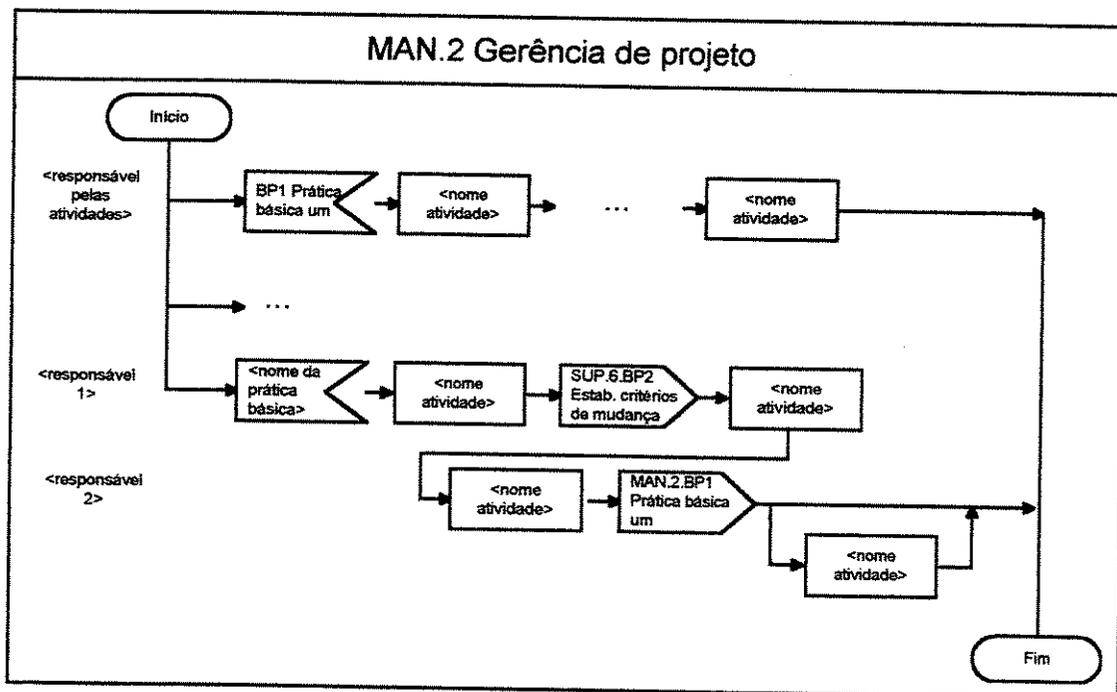


Figura 4.3 Exemplo de um fluxograma

Como um processo pode ser instanciado em diferentes momentos, para executar diferentes práticas básicas, utiliza-se o símbolo de *Sinal de Entrada* para sinalizar qual prática básica deve ser executada nesta instanciação. No exemplo da figura, está representado por *BP1 Prática básica um* e *<nome da prática básica>*.

Para representar a comunicação com outras instâncias de processo, é utilizado o símbolo *Sinal de Saída*. No exemplo da figura, *SUP.6.BP2 Estab. critérios de mudança* indica que ativa uma instância do processo *SUP.6 Revisão* para aplicar a prática básica *BP2 Estabelecer critérios de mudança*. No caso de *MAN.2.BP1 Prática básica um*, indica que ativa uma outra instância do processo *MAN.2 Gerência de projeto* para aplicar a prática básica *BP1 Prática básica um*.

Descrição de atividades de processo

A ferramenta de descrição de atividades de processos detalha as atividades definidas no fluxograma, agrupadas por práticas básicas, mantendo a consistência com o fluxograma. A Figura 4.4 apresenta o formato para descrição de atividade. As práticas estão em ordem crescente de identificação de prática. A descrição das atividades apresenta uma breve definição da prática básica (extraída da ISO/IEC TR 15504), a relação dos produtos de entrada e saída de cada atividade e, em seguida, um roteiro das tarefas que devem ser executadas na atividade.

<IDprática Nome da prática> <Descrição da prática.>	
<IDprática> Atividade <nome da atividade> A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.	
<i>Entrada:</i>	<i>Saída:</i>
id) Produto de entrada 1	id) Produto de saída 1
id) ...	id) ...
id) Produto de entrada <i>m</i>	id) Produto de saída <i>n</i>
a < tarefa 1 >	
b ...	
c < tarefa <i>p</i> >	

Figura 4.4 Formato de descrição de processo

Uma atividade pode não requerer produto de trabalho de entrada ou de saída. Isto é indicado pelo uso da frase *não há*. Para produto de trabalho de entrada, o uso desta frase indica que a atividade não requer entrada relevante para gerar produto de saída. Para produto de trabalho de saída, indica que a atividade não gera produto de saída relevante ou que a atividade ativa outro processo, onde o produto de saída é gerado.

O roteiro de tarefas e a seqüência de atividades do fluxograma são resultantes de uma compilação de métodos, técnicas ou procedimentos existentes na literatura de Engenharia de Software para cada um dos processos selecionados. Portanto, ele é uma contribuição deste trabalho. Quando necessário, o roteiro propõe formato de formulário e/ou documento que deve ser gerado.

As atividades, os produtos de trabalho e suas características devem estar consistentes com os níveis anteriormente descritos: fluxograma, macrodiagrama e escopo.

5 Descrição dos processos

O desenvolvimento deste trabalho considerou uma organização de software que:

- está adotando uma abordagem de Gerenciamento pelas Diretrizes, para implementar seu programa de Gerenciamento da Qualidade Total;
- está implantando o método de Gestão de Processo para gerenciar a rotina do trabalho do dia-a-dia;
- adota estrutura de desenvolvimento de software organizada por projeto;
- deseja melhorar a capacidade de seus processos;
- quer entender melhor o que o seu cliente precisa;
- possui grandes distorções de custo e prazo em relação ao planejado;
- possui pouca documentação de desenvolvimento/usuário ou documentação desatualizada.

A partir destas considerações, identificou-se que esta organização possui três dimensões de interesse deste trabalho: (1) Gerenciamento da Qualidade Total, (2) Qualidade em Processo de Software e (3) Organização por Processo (estrutura). A Figura 5.1 apresenta uma visão geral destas dimensões e suas principais ações. As ações estão representadas por retângulos coloridos. A cor verde indica dimensão *Gerenciamento da Qualidade Total*, a amarela indica a dimensão *Qualidade em Processo de Software* e a azul indica dimensão *Organização por Processo*.

Observou-se também a existência de três relacionamentos entre estas dimensões, representados na Figura 5.1 pelas setas vazadas. O primeiro ocorre com a ação *Planejamento e organização de processo* sendo implementada pela ação *Identificação, priorização e descrição de processo de software*. O segundo relacionamento ocorre com a ação *Gerência de projeto de software* utilizando os processos descritos, de uma forma ordenada, para alcançar os objetivos do projeto de software. O terceiro relacionamento ocorre com a ação *Análise e melhoria* executando uma *Avaliação de processo de software*.

Para viabilizar a implantação do primeiro e do segundo relacionamentos, este trabalho desenvolveu uma tecnologia integrada para melhoria de processo de software. A atividade de viabilizar a implantação do terceiro relacionamento não pertence ao escopo deste trabalho, mas esta tecnologia gera a base que viabilizará a sua implantação.

A tecnologia integrada consiste de dois itens: descrição de processos de software e modelo de gerência de projeto de software. O item *descrição de processos de software* considera a descrição dos processos como parte da implantação do método de Gestão de Processo - dentro do programa de Gerenciamento da Qualidade Total - e baseia-se nos processos de software Modelo de Referência para processo e capacidade de processo da ISO/IEC TR 15504. O item *modelo de gerência de projeto de software* estabelece uma ordem de utilização dos processos descritos para alcançar os objetivos de projeto de software. Este item será discutido no próximo capítulo.

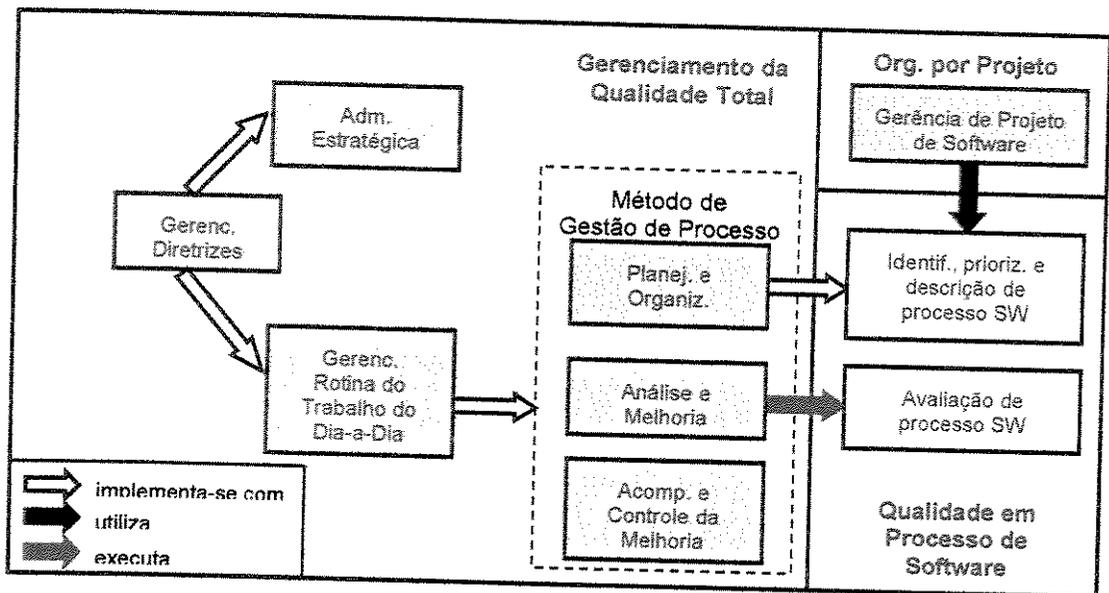


Figura 5.1 Dimensões de uma organização de software: ações e seus relacionamentos

A descrição de processos permite uma visão sistêmica das atividades e, por estar baseada na futura norma internacional, assegura o uso de boas práticas de engenharia de software. Para uma empresa, o estabelecimento destes processos começa a formar a base necessária para melhorar o nível de maturidade de seus processos de forma gradual e contínua, incorporando qualidade a seus produtos. Com isso, na ação *Análise e Melhoria* é possível utilizar o modelo de avaliação de processo de software da ISO/IEC TR 15504 para identificar a capacidade de processos e seus pontos fracos. Ações de melhoria poderão ser definidas a partir dos resultados da avaliação ou das práticas de gerenciamento, representadas no modelo de referência de capacidade de processo como atributos de nível de capacidade (veja Figura 2.3). Por isso, a descrição dos processos é uma importante contribuição deste trabalho.

O desenvolvimento deste item seguiu a metodologia descrita no Capítulo 4 e apresentou os resultados descritos a seguir.

A atividade *identificação de processos* resultou em uma adaptação do modelo de referência para processos e capacidade de processo da ISO/IEC TR 15504, que pode ser vista na Figura 5.2. O modelo classifica os processos em cinco categorias, de acordo com o tipo de atividade que eles endereçam. As cinco categorias - Cliente-Fornecedor (CUS), Engenharia (ENG), Suporte (SUP), Gerenciamento (MAN) e Organização (ORG) - estão agrupadas em três conjuntos de processos de ciclo de vida: Processos Primários, Processos de Suporte e Processos Organizacionais. Do modelo original, foram excluídos os processos *ORG.1 Alinhamento organizacional*, *ORG.2.1 Estabelecimento de processo*, *ORG.2.2 Avaliação de processo*, *ORG.2.3 Melhoria de processo* e *ORG.3 Gerenciamento de recursos humanos* porque assume-se que estes processos são tratados na dimensão *Gerenciamento da Qualidade Total*.

A atividade *priorização de processos* do modelo adaptado de processo de software, analisou os processo do modelo, segundo o critério de priorização definido no Capítulo 4 e selecionou os processos *CUS.3 Elicitação de requerimentos*, *MAN.2 Gerência de projeto*, *SUP.2*

Gerência de configuração e SUP.6 Revisão. Esta priorização fez-se necessária devido à quantidade de processos que compõem o modelo - um total de 31 processos - e ao esforço que a descrição deles demanda - compilação de material disponível na literatura de engenharia de software. Além disso, o tempo disponível para desenvolvimento do trabalho acadêmico também não seria suficiente.

A atividade *descrição de processos* utiliza todas as ferramentas descritas no *Capítulo 4*. O nível de detalhamento das atividades dos processos restringe-se às necessidades das macro-atividades *Planejamento Preliminar e Aprovação* do modelo de Gerência de projeto (veja *Capítulo 6*). Para auxiliar na implementação do processo, alguns formatos de formulários e de documentos foram desenvolvidos e os anexos III, IV e V mostram alguns exemplos. Estes anexos estão referenciados na descrição das atividades dos processos.

O Anexo I relaciona as características dos produtos de trabalho utilizados pelos processos descritos. Esta relação foi traduzida do *Annex C (informative) Work products and their characteristics* (Parte 5) da ISO/IEC TR 15504. Alguns produtos de trabalho não constam desta relação pois não pertence ao contexto dos processos prioritários e outros - de numeração acima de 200 - foram acrescentados devido à necessidade dos processos prioritários.

Ainda na atividade de descrição de processos, alguns processos não prioritários tiveram somente o escopo definido com o objetivo de auxiliar a compreensão dos processos prioritários. Esta definição encontra-se no Anexo II, e engloba os processos a seguir.

- ENG.1.1 Processo de análise de requerimentos e design de sistema.
- ENG.1.2 Processo de análise de requerimentos de software.
- ENG.1.3 Processo de design de software.
- ENG.1.4 Processo de construção de software.
- ENG.1.5 Processo de integração de software.
- ENG.1.6 Processo de teste de software.
- ENG.1.7 Processo de integração de sistema e teste.
- ENG.2 Processo de manutenção de sistema e software.
- SUP.1 Processo de documentação.
- SUP.3 Processo de garantia de qualidade.
- SUP.4 Processo de verificação.
- SUP.5 Processo de validação.
- ORG.4 Processo de infra-estrutura.

Os processos não prioritários *CUS.1 Aquisição*, *CUS.2 Fornecimento*, *CUS.4 Operação*, *SUP.7 Auditoria*, *SUP.8 Resolução de problemas*, *MAN.1 Gerenciamento*, *MAN.3 Gerenciamento de qualidade*, *MAN.4 Gerenciamento de risco*, *ORG.5 Medição* e *ORG.6 Reuso* não estão descritos neste trabalho.

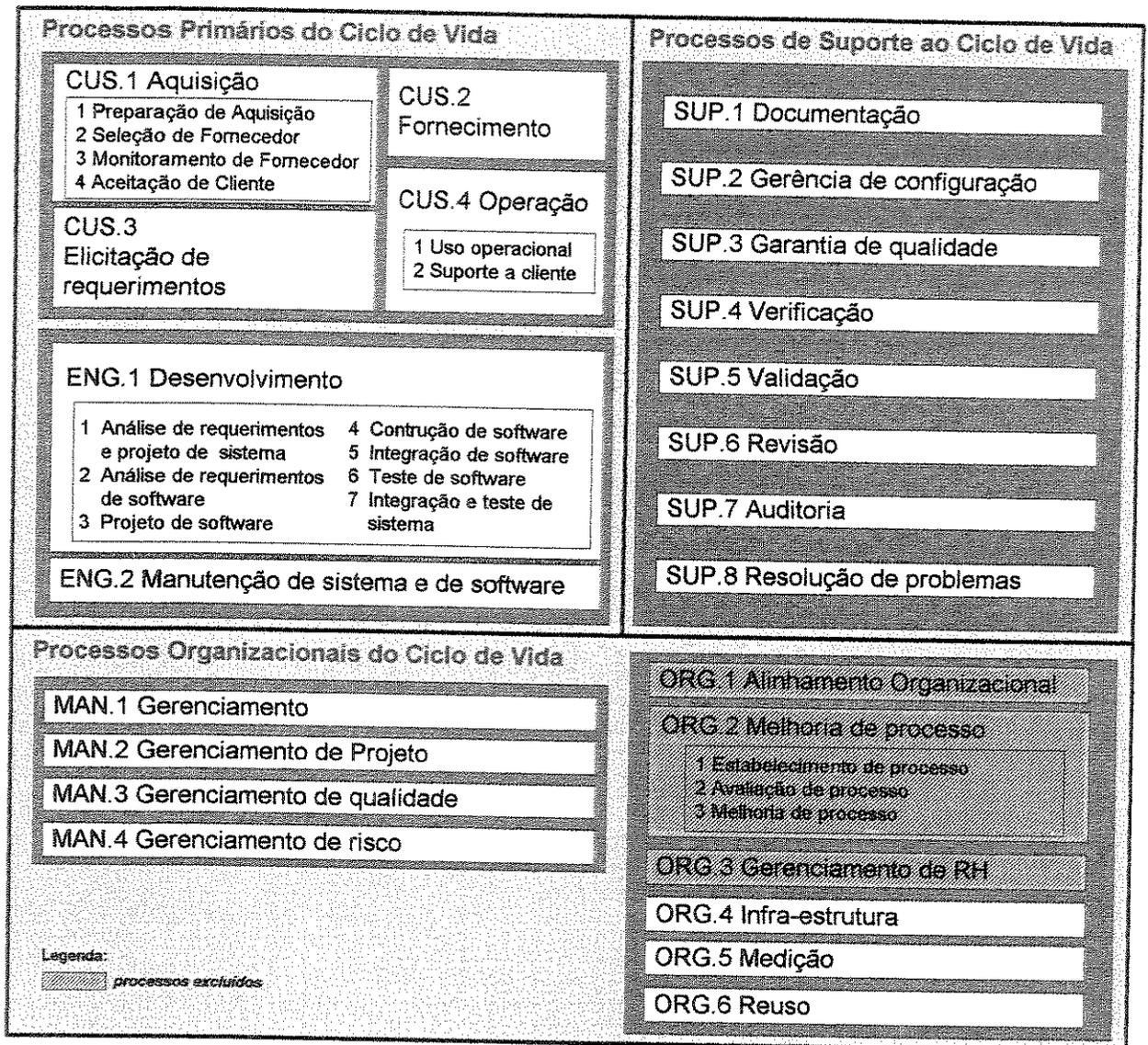


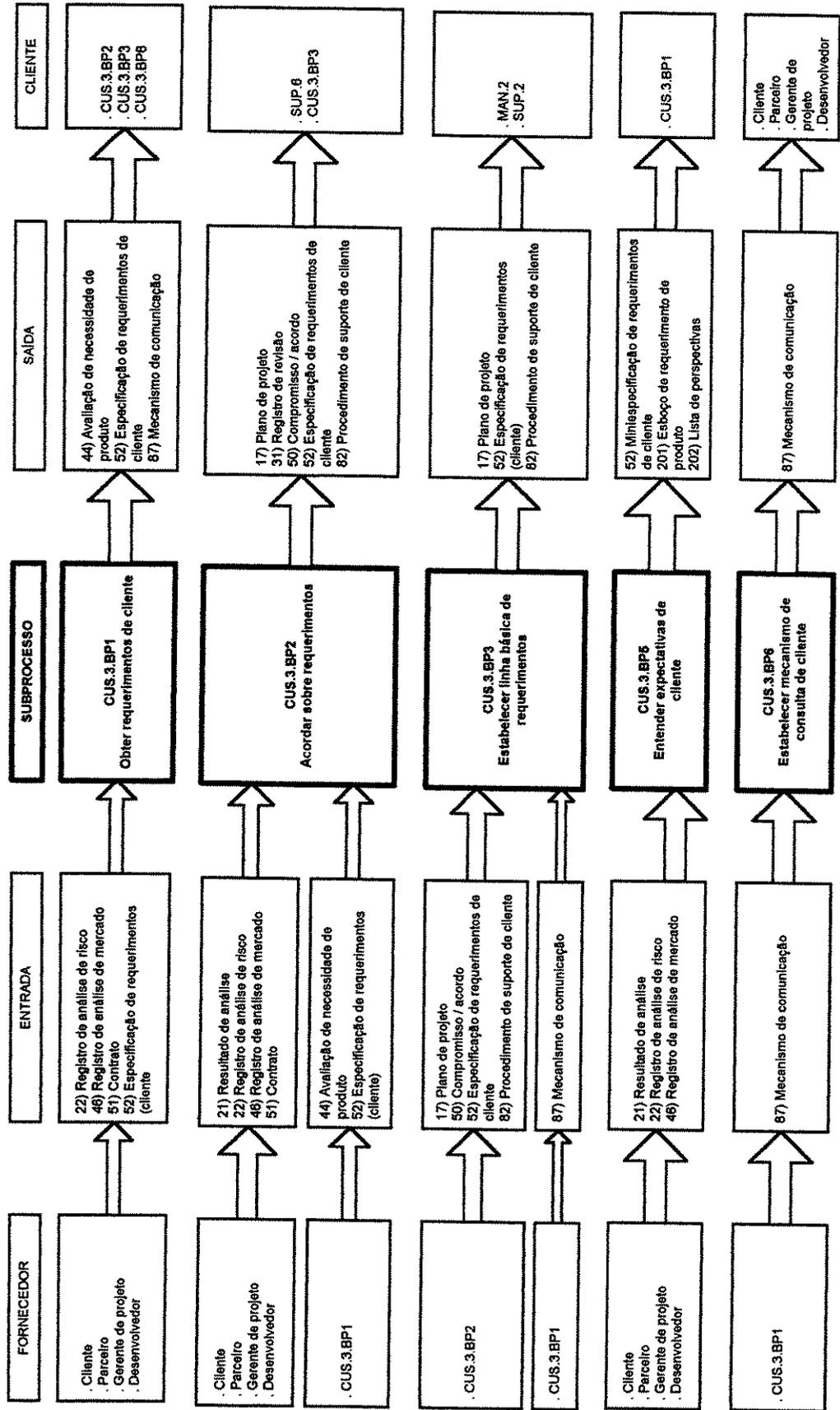
Figura 5.2 Modelo de processo de software, adaptado da ISO/IEC TR 15504

5.1 CUS.3 Processo de elicitação de requerimentos

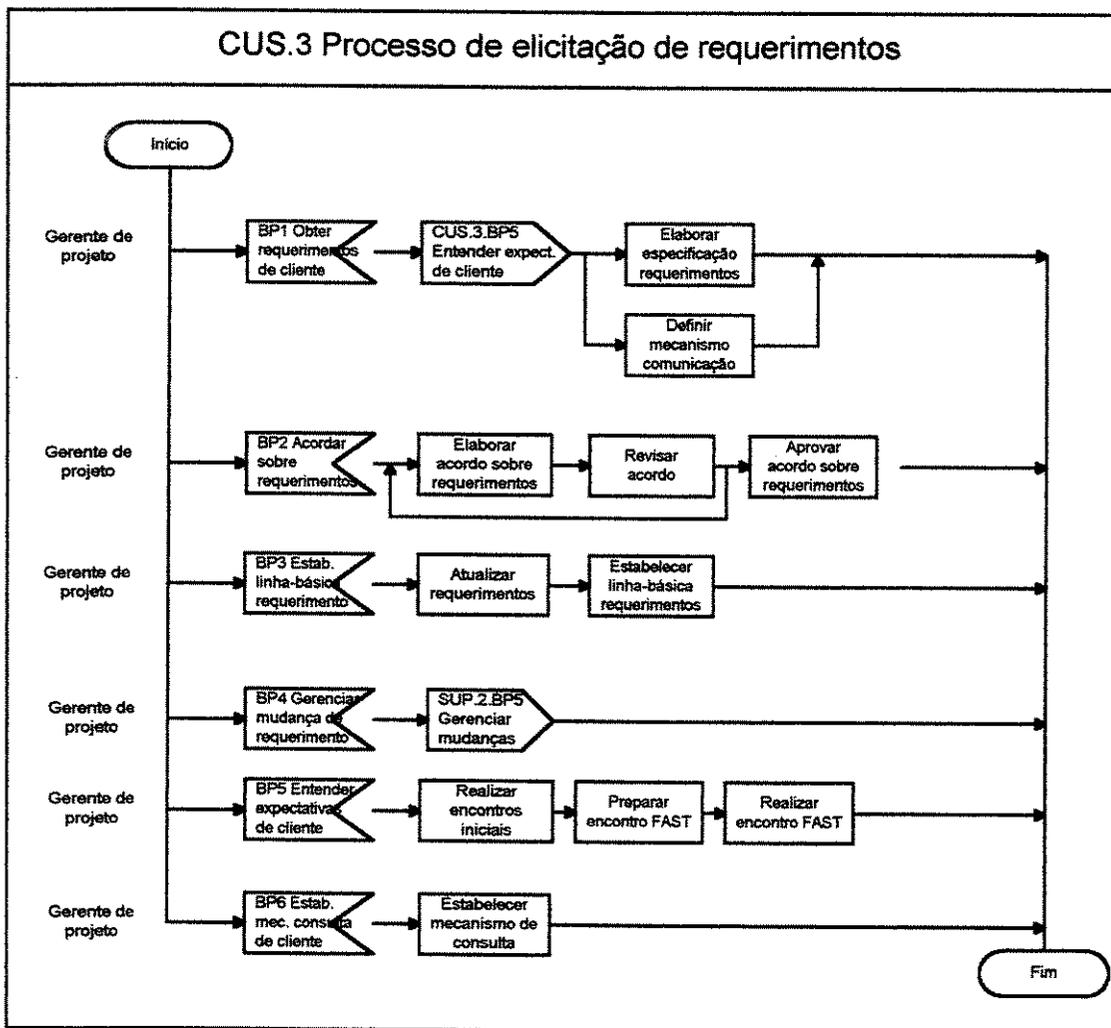
5.1.1 Escopo

Nome do processo:	CUS.3 Processo de elicitação de requerimentos		
Objetivo:	O propósito deste processo é obter, processar e acompanhar o desenvolvimento de necessidades e requerimentos de cliente durante a vida do produto e/ou serviço de software. Além disso, o processo deve estabelecer uma linha-básica de requerimentos, a partir da qual são definidos os produtos de trabalho de software necessários.		
Entradas:	21) Resultado de análise (funcional) 22) Registro de análise de risco 46) Registro/relatório de análise de mercado 51) Contrato 52) Especificação de requerimentos (cliente)	Saídas:	17) Plano de projeto 31) Registro de revisão 44) Avaliação de necessidades de produto 50) Compromisso / acordo 52) Especificação de requerimentos (cliente) 82) Procedimento de suporte de cliente 87) Mecanismo de comunicação 201) Esboço de requerimento de produto 202) Lista de perspectivas
Início do processo:	Obtenção de requerimentos e pedidos de cliente		
Conteúdo:	Obtém requerimentos e pedidos de cliente; entra em acordo sobre os requerimentos; estabelece linha-básica de requerimentos de cliente; gerencia mudanças de requerimentos de cliente; entende as expectativas de cliente; estabelece mecanismo de consulta de cliente.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: uma comunicação contínua com o cliente estará estabelecida; os requerimentos de cliente acordados estarão definidos; um mecanismo estará estabelecido para incorporar novos requerimentos de cliente à linha-básica de requerimentos estabelecida; um mecanismo estará estabelecido para contínuo monitoramento de necessidades de cliente; um mecanismo estará estabelecido para garantir que clientes possam facilmente determinar o status e a disposição de seus pedidos; melhorias resultantes de mudança de tecnologia e de necessidades de cliente estarão identificadas e seu impacto gerenciado.		
Clientela a ser atendida:	gerente de projeto, desenvolvedor, cliente, parceiro		
Indicadores de desempenho:	Existência e adequação das práticas base aplicadas neste processo: CUS.3.BP1, CUS.3.BP2, CUS.3.BP3, CUS.3.BP5, CUS.3.BP6. Existência dos produtos de trabalho de entrada e saída e suas características.		

5.1.2 Macro-diagrama de CUS.3 Processo de elicitação de requerimentos



5.1.3 Fluxograma



5.1.1 Atividades

CUS.3.BP1 Obter requerimentos de cliente

Obter e definir requerimentos de cliente através de solicitação direta ao cliente e entradas de usuário, e através de revisão de propostas de negócio de cliente, de ambiente de operação e de hardware, e de outros documentos relacionados com requerimentos de cliente¹.

CUS3.BP1 Atividade Instanciar CUS.3.BP5

- a. Instanciar o processo *CUS.3 Elicitação de requerimentos de cliente*, aplicando a prática básica *BP5 Entender as expectativas de cliente*.

CUS3.BP1 Atividade Elaborar especificação de requerimentos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
22) Registro de análise de risco	44) Avaliação das necessidades do produto
46) Registro/relatório de análise de mercado	52) Especificação de requerimentos (cliente)
51) Contrato ²	
52) Mini-especificação de requerimentos (cliente)	

- a. Completar o produto de trabalho 52) *Especificação de requerimentos (cliente)* com os requerimentos a partir das informações da mini-especificação, do contrato (quando existir um) e dos registros de análise de riscos.

Os requerimentos devem ser identificados por um código que é único e devem estar organizados nas categorias: Funcional, Técnico, Operacional, Qualidade e Manutenção. Dentro da categoria serão identificados segundo a notação:

¹ Todos os requerimentos devem ser verificáveis ou podem ser avaliados.

² A elaboração do produto de trabalho 51) *Contrato* é realizada por um processo que não faz parte do contexto deste trabalho.

TRnnnn

- onde:
- TR representa o tipo do requerimento
F = funcional T = técnico M = manutenção
O = operacional Q = qualidade
 - nnnn representa um número sequencial de requerimento, iniciando de 0001.

Os **requerimentos funcionais** contêm a descrição de funções que o sistema executa. As entradas, os processos e as saídas são claramente identificados e ordenados para melhor descrever processos. Pode-se usar a técnica Use Case para identificar cenários [Booch, 1999].

Os **requerimentos técnicos** descrevem as características técnicas do produto, divididas nas subcategorias:

- hardware (configuração de hardware, ambiente de trabalho, etc.);
- software (tecnologia a ser usada, linguagem de programação, etc.);
- interfaces de comunicação (colaborações com outros sistemas, interfaces interna e externa, engenharia humana, etc.);
- capacidade (expectativa de volume de dados, número de terminais a ser suportado, número de usuários simultâneos, etc.);
- desempenho (número de transações dentro de um período de tempo em condições normais ou de pico, tempo de resposta, tempo aceitável para processos off-line, etc.);
- segurança de acesso;
- confiabilidade;
- sistema e *design*.

Os **requerimentos operacionais** descrevem os aspectos operacionais, estes incluem:

- instalação;
- suporte;
- documentação;
- treinamento;
- ambiente;
- armazenamento (de produto).

Os **requerimentos de qualidade** descrevem o nível de qualidade do produto de acordo com as características e subcaracterísticas de qualidade da norma ISO/IEC 9126 [ISO9126, 1994]. O ponto de vista do cliente é considerado e requerimentos para todas as sub-características devem ser definidos. Nesta categoria, também incluem-se os requerimentos estatutários e regulatórios.

Os **requerimentos de manutenção** descrevem os requerimentos de mudança para um software ou sistema que se encontra em operação.

O formato proposto para o produto de trabalho 52) *Especificação de requerimentos (cliente)* está no Anexo V.

- b. Elaborar o produto de trabalho 44) *Avaliação das necessidades do produto*, baseado nos produtos de entrada da atividade, contendo:
- definição de necessidade:
 - razão da necessidade do produto;
 - características e funções desejadas ;
 - requerimentos a serem satisfeitos;
 - restrições:
 - limitações de custo;
 - requerimentos de data/cronograma;
 - software de suporte específico requerido;
 - requerimentos de interfaces ;
 - equipamento ou hardware associado requerido;
 - padrões e/ou requerimentos regulatórios;
 - impactos operacionais;
 - características de patente, copyright e licença;
 - caso de negócio:
 - benefícios esperados;
 - custo esperado (incluindo instalação projetada, conversão e/ou manutenção) x expectativas de vantagem;
 - janela de mercado/oportunidade, datas de entrega alvo.

CUS.3.BP1 Atividade Definir mecanismo de comunicação

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
não há	87) Mecanismo de comunicação

- a. Definir e descrever uma maneira para distribuir informação tal que:
- possua descrição clara do que está sendo comunicado;
 - possua habilidade para especificar informação de data enviada;
 - possua habilidade para distribuir para todos os impactados;
 - possua identificação do impacto: hardware, desenvolvimento, cliente, organização, etc.);
 - forneça uma identificação clara de/para que/quem a mensagem se aplica;
 - possua mecanismo de destinatário para responder quando requerido (retorno de informação);
 - o meio de distribuição usado seja acessível a todos os envolvidos / impactados;
 - a lista de distribuição seja corrente e inclua todos os envolvidos/impactados;
 - possua habilidade para especificar informação de data de retorno alvo.
- b. Estabelecer procedimentos e responsabilidades para atualização e divulgação de informação utilizando o mecanismo definido.

Por exemplo, este mecanismo pode ser a combinação de correio eletrônico e uma home page de acesso comum a todos os envolvidos no projeto. O correio eletrônico formalizaria discussões e aprovações e a home page conteria informações que permitissem o acompanhamento do status do projeto por todos os envolvidos.

CUS.3.BP2 Acordar sobre requerimentos

Obter acordo entre os times sobre os requerimentos de cliente, obtendo o aval de representantes de todos os times e de outras partes contratualmente envolvidas para trabalhar estes requerimentos.

CUS.3.BP2 Atividade Elaborar acordo sobre requerimentos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
21) Resultado de análise	17) Plano de projeto (preliminar)
22) Registro de análise de risco	31) Registro de revisão
44) Avaliação de necessidade de produto	50) Compromisso / acordo
46) Registro/relatório de análise de mercado	52) Especificação de requerimentos (cliente)
51) Contrato	82) Procedimento de suporte de cliente
52) Especificação de requerimentos (cliente)	

- a. Instanciar o processo *MAN.2 Gerência de Projeto* para gerar o produto de trabalho 17) *Plano de Projeto (preliminar)*, baseado nos produtos de entrada desta atividade e aplicando a prática básica *BP10 Estabelecer e implementar planos de projeto* (neste momento, somente estabelecer os planos).
- b. Elaborar o produto de trabalho 82) *Procedimento de suporte de cliente*, contendo:
- tarefas para seguir em provimento de suporte definido;
 - definir a disponibilidade e a cobertura do suporte provido:
 - # hot-line (se houver);
 - horas de disponibilidade;
 - pericia (*expertise*) apropriada;
 - custo;
 - definir um esquema para classificação de solicitação de cliente e/ou problemas:
 - definição de tipo de solicitação;
 - definição de prioridade/severidade;
 - definição de expectativas de tempo de resposta, pelo tipo e severidade;
 - padrões para quais informações reter a partir do cliente, tais como:
 - companhia e localização;
 - detalhes de informações de contato;
 - descrição da solicitação;
 - referência para informação de suporte enviada (dumps, files);
 - informação de configuração do *site* de sistema de cliente (produto, liberação, versão, última atualização, etc.);
 - sistema(s) impactado(s);
 - impacto para operações de sistemas existentes;
 - o quanto é crítica a situação;
 - requerimentos de resposta / fechamento de cliente esperados;
 - definição de procedimentos de escalação de cliente (por exemplo, ordem de solicitação, grau de severidade do problema, etc.);
 - identificação de *ferramentas de suporte de cliente* disponíveis e procedimentos para usá-las, tais como:

- mecanismo usado para registrar solicitações de cliente;
- relatórios de status;
- sistemas disponíveis para reproduzir problemas;
- habilidade para reproduzir ambiente de software de clientes;
- habilidade para reproduzir problemas;
- emuladores de apoio;
- roteiros de apoio;
- "dial-in ports";
- ferramentas para análise de *dumps*.

c. Elaborar o produto de trabalho 50) *Compromisso/ acordo*, no formato definido pelas normas vigentes da empresa, e considerando os produtos de trabalho de entrada desta atividade.

CUS.3.BP2 Atividade Revisar acordo

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto (preliminar)	17) Plano de projeto (preliminar)
52) Especificação de requerimentos (cliente)	31) Registro de revisão
82) Procedimento de suporte de cliente	52) Especificação de requerimentos (cliente)
	82) Procedimento de suporte de cliente

- a. Instanciar o processo *SUP.6 Revisão*, aplicando as práticas básicas *BP1 Preparar revisão*, *BP4 Conduzir revisão técnica* e *BP7 Determinar ações para resultado de revisão*, para realizar uma revisão crítica dos produtos de trabalho de entrada. Nesta revisão, segundo Valeriano [Valeriano, 1998], deve-se:
- determinar se os requerimentos estabelecidos para o sistema foram completa e apropriadamente identificados e documentados;
 - confrontar os requerimentos funcionais estabelecidos para o sistema com os requerimentos definidos pelo cliente/usuário;
 - considerar o sistema ao longo do ciclo de vida, envolvendo os aspectos de produção, apoio logístico, software e testes, sempre tendo em vista o atendimento dos requerimentos do sistema;
 - avaliar e formalizar [Valeriano, 1998]:
 - a especificação de requerimentos do sistema;
 - os planos de gerência do projeto;
 - o sistema de documentação;
 - o procedimento de suporte de cliente;
 - outros aspectos que dizem respeito ao sistema.

- b. Solucionar os problemas apontados no produto de trabalho 31) *Registro de revisão*, e atualizar os produtos impactados.

CUS.3.BP2 Atividade Aprovar acordo sobre requerimentos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
50) Compromisso / acordo	não há

- a. Submeter para aprovação o produto de trabalho 50) *Compromisso / acordo*, seguindo os procedimentos internos da empresa.

CUS.3.BP3 Estabelecer linha-base de requerimentos de cliente

Documentar os requerimentos de cliente e estabelecer como uma linha-base para uso de projeto e monitoramento contra as necessidades de cliente.

CUS.3.BP3 Atividade Atualizar requerimentos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	17) Plano de projeto
50) Compromisso / acordo	52) Especificação de requerimentos (cliente)
52) Especificação de requerimentos (cliente)	82) Procedimento de suporte de cliente
82) Procedimento de suporte de cliente	

- a. Atualizar os produtos de entrada conforme as modificações negociadas e aprovadas constantes do produto 50) *Compromisso / acordo* aprovado.

CUS.3.BP3 Atividade Estabelecer linha-básica de requerimentos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto (preliminar) 52) Especificação de requerimentos (cliente) 82) Procedimento de suporte de cliente 87) Mecanismo de comunicação	não há

- a. Instanciar processo *SUP.2 Gerência de Configuração*, aplicando a prática básica *BP9 Gerenciar itens de configuração*, para estabelecer a linha-básica de requerimentos. Colocar sob controle de mudança os produtos *17) Plano de Projeto (preliminar)*, *52) Especificação de requerimentos (cliente)* e *82) Procedimento de suporte de cliente*.
- b. Avisar a todos os envolvidos que existe uma linha-básica de requerimentos, utilizando o mecanismo definido no produto de trabalho *87) Mecanismo de comunicação*.

CUS.3.BP5 Entender as expectativas de cliente

Revisar com clientes e usuários seus requerimentos para melhor entender suas necessidades e expectativas.

CUS3.BP5 Atividade Realizar encontros iniciais

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
não há	201) Esboço de requerimento de produto

- a. Agendar um encontro ou entrevista preliminar entre o cliente e o desenvolvedor.
- b. No encontro, iniciar a comunicação com perguntas de livre contexto com o objetivo de obter uma compreensão básica do problema, saber quais as pessoas que querem uma solução, qual a natureza da solução que é desejada e qual a efetividade do primeiro encontro em si [Pressman, 1995].

A Tabela 5.1 apresenta algumas sugestões de perguntas, agrupadas em três grupos por tipo de pergunta. Estas e outras perguntas ajudarão a "quebrar o gelo" e a iniciar a comunicação essencial para um bem-sucedido levantamento de informações. Esta abordagem de perguntas e respostas em um encontro deve ser usada somente para os encontros iniciais e depois ser

substituída por uma forma que combine elementos de solução de problemas, negociação e especificação, como por exemplo *Facilitated Application Specification Techniques - FAST*¹.

- c. Gerar um esboço de requerimento de produto de uma ou duas páginas, quando o desenvolvedor e o cliente tiverem uma visão global do escopo do problema e de uma solução.

O formato proposto para o produto de trabalho 201) *Esboço de requerimento de produto* está no Anexo III.

Tabela 5.1 Sugestões de perguntas para encontro preliminar entre cliente e desenvolvedor	
Cliente / Metas globais / Benefícios	Melhor compreensão do problema pelo desenvolvedor / Cliente verbaliza suas percepções sobre uma solução
1. Quem está por trás do pedido deste trabalho?	1. Como você caracteriza um <i>bom</i> resultado (saída) que seria gerado por uma solução bem-sucedida?
2. Quem usará a solução?	2. Qual problema(s) essa solução resolverá?
3. Qual o benefício econômico de uma solução bem-sucedida?	3. Você poderia mostrar-me (ou descrever-me) o ambiente em que a solução será usada?
4. Há outra fonte para a solução exigida?	4. Existem questões de desempenho ou restrições especiais que afetarão a maneira pela qual a solução é abordada?
Efetividade do encontro	
1. Você é a pessoa certa para responder a essas perguntas? Suas respostas são <i>oficiais</i> ?	
2. Minhas perguntas são pertinentes ao problema que você tem?	
3. Estou fazendo perguntas demais?	
4. Há mais alguém que possa fornecer informações adicionais?	
5. Existe algo mais que eu deva perguntar-lhe?	

CUS3.BP5 Atividade Preparar encontro FAST

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
21) Resultado de análise 22) Registro de análise de risco 46) Registro/relatório de análise de mercado 201) Esboço de requerimento de produto	202) Lista de perspectivas

¹ Muitas abordagens diferentes de FAST têm sido propostas, dentre elas a *Joint Application Development (JAD)* desenvolvida pela IBM, e *The METHOD*, desenvolvida pela Performance Resources, Inc., Falls Church, VA.

- a. Agendar um encontro com a participação de representantes do cliente e do desenvolvedor e designação de um moderador.
- b. Reunir o seguinte material necessário para o encontro FAST:
 - 201) *Esboço de requerimento de produto*;
 - 202) *Lista de perspectivas*, que deverá ser construída durante a revisão do produto de trabalho 201) *Esboço de requerimento de produto*; o formato proposto para a lista de perspectivas está no Anexo IV;
 - orientações sobre como um encontro FAST é conduzido.
- c. Distribuir o material acima antes da data do encontro.

CUS3.BP5 Atividade Realizar encontro FAST

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
202) Lista de perspectivas	52) Miniespecificação de requerimentos (cliente)

- a. Realizar o encontro FAST executando os seguintes passos [Pressman, 1995]:
 - discutir e estabelecer consenso sobre a necessidade e justificativa do novo produto;
 - cada participante deve apresentar suas listas para crítica e discussão; as listas podem ser afixadas às paredes da sala usando grandes folhas de papel; idealmente, cada tópico da lista pode ser manipulado separadamente, de forma que as listas sejam combinadas, entradas sejam apagadas e adições sejam feitas; nessa fase, a crítica e o debate são estritamente proibidos;
 - criar uma lista combinada para cada área abrangendo todos os temas e que reflita as idéias apresentadas; esta lista elimina as entradas redundantes e acrescenta quaisquer novas idéias que possam aparecer durante a apresentação, mas não apaga nenhuma;
 - iniciar a discussão sobre as listas combinadas - coordenada pelo moderador - com o objetivo de desenvolver uma lista consensual de cada área de assunto (objetos, operações, restrições e desempenho); cada lista combinada é abreviada, ampliada ou novamente redigida para refletir adequadamente o sistema/produto a ser desenvolvido; as listas são então guardadas para ação posterior;
 - dividir a equipe em subequipes menores, assim que as listas de consenso forem concluídas;

- desenvolver em cada subequipe uma *miniespecificação* de uma ou mais entradas de cada uma das listas; a *miniespecificação* é uma elaboração das palavras ou frases contidas numa lista;
 - cada subequipe deve apresentar cada uma das miniespecificações a todos os participantes do encontro FAST para discussão; adições, supressões e elaboração adicional são feitas e em alguns casos, o desenvolvimento de miniespecificações desvendará novos objetos, operações, restrições ou exigências de desempenho que serão acrescentadas às listas originais; durante as discussões, a equipe pode levantar questões que não serão resolvidas durante o encontro e uma lista de questões deve ser guardada para que possa agir sobre essas idéias posteriormente;
 - cada subequipe devem elaborar uma lista de critérios de validação para o produto/sistema - após a conclusão das miniespecificações - e apresentá-la à equipe;
 - criar uma lista de consenso dos critérios de validação;
 - responsabilizar um ou mais participantes (ou pessoas de fora) pela tarefa de escrever o esboço completo de especificação usando todas as entradas do encontro FAST.
- b. Gerar o produto de trabalho 52) *Miniespecificação de requerimentos (cliente)*, escrevendo o esboço completo de especificação usando todas as entradas do encontro FAST. O formato proposto para a miniespecificação é o mesmo do produto de trabalho 52) *Especificação de requerimentos* e encontra-se no Anexo V. Alguns itens do produto podem ficar em aberto para posterior descrição, tendo em vista que é um esboço.

CUS.3.BP6 Estabelecer um mecanismo de consulta de cliente

Prover uma maneira pela qual o cliente pode ter ciência do status e disposição de seus requerimentos de mudança¹.

CUS.3.BP6 Atividade Estabelecer mecanismo de consulta

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
87) Mecanismo de comunicação	87) Mecanismo de comunicação

- a. Definir um mecanismo de consulta sobre o status de cada solicitação. Este mecanismo pode estar incluído no produto de trabalho 87) *Mecanismo de comunicação*.

¹ Isto pode incluir reuniões com o cliente ou comunicação formal para revisar os status de seus requerimentos e requisitos; remeter para o processo SUP.6 Revisão.

5.2 MAN.2 Processo de Gerência de Projeto

A gerência de projeto é apoiada pela ferramenta *Estrutura de Decomposição de Trabalho - EDT*. Esta ferramenta provê uma linguagem comum, simples e racional para todos os participantes do desenvolvimento do projeto e constitui-se em um ponto-chave para todo o prosseguimento dos trabalhos.

A Figura 5.3 apresenta um resumo das atividades a serem realizadas para desenvolver uma EDT (veja item 3.2 *Estrutura de Decomposição de Trabalho - EDT*), relacionando-as com as práticas básicas do processo, estabelecidas pela norma ISO/IEC TR 15504.

A numeração à direita das atividades identifica o nível de detalhamento das mesmas e o código à esquerda identificam as práticas básicas deste processo que têm correspondência com as atividades e são as seguintes:

- BP2 Determinar estratégia de desenvolvimento;
- BP4 Dimensionar e estimar tarefas e recursos;
- BP5 Desenvolver estrutura de decomposição de trabalho;
- BP6 Identificar requerimentos de infra-estrutura;
- BP7 Estabelecer cronograma de projeto;
- BP8 Alocar responsabilidades.

Resumo das atividades de desenvolvimento da estrutura de decomposição de trabalho - EDT

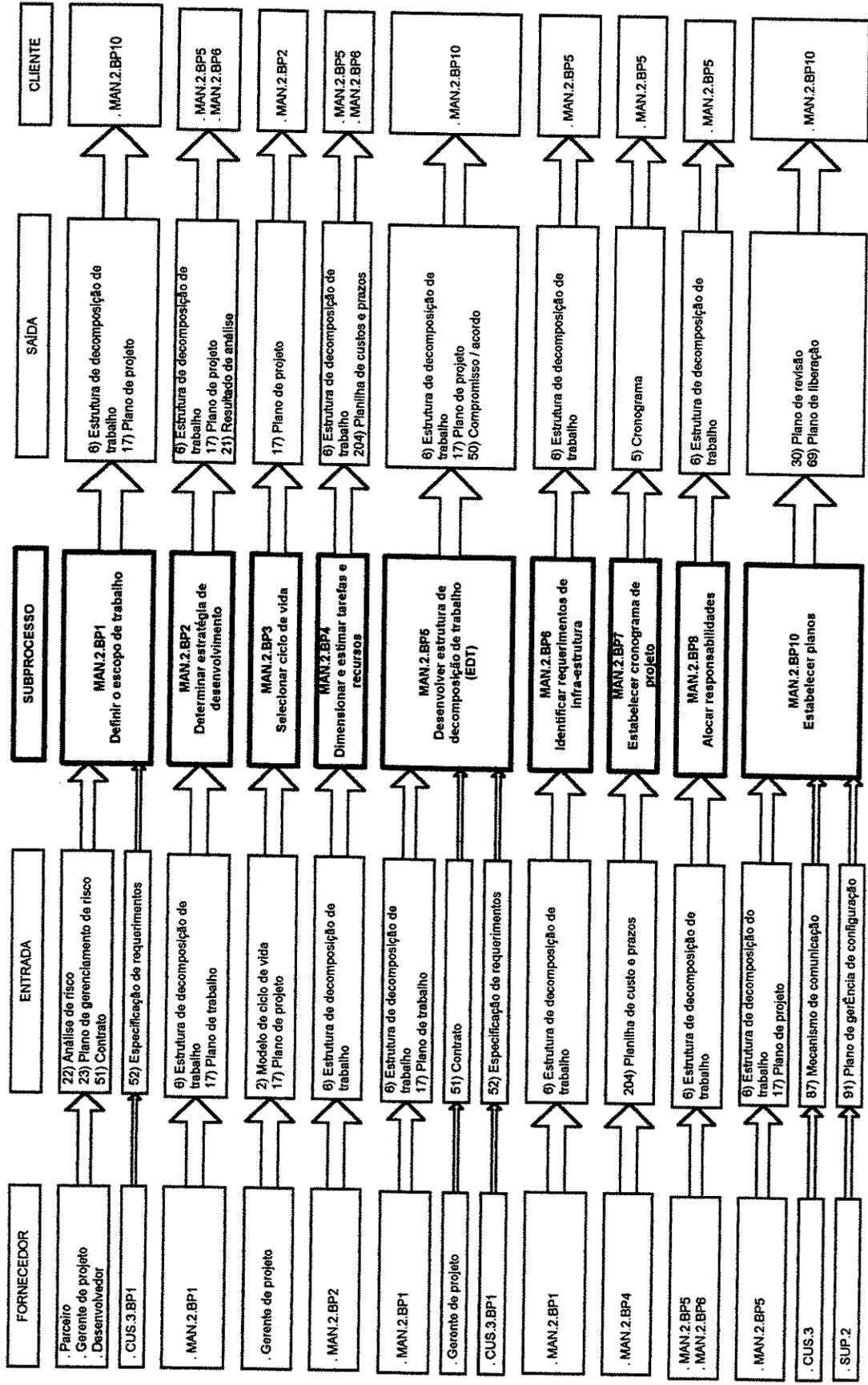
<p>(1) Desenvolver EDP.</p>	<p>(1.1) Definir sistema de identificação para numeração dos blocos da EDT. (BP5)</p> <p>(1.2) Decompor o produto em suas partes constituintes</p> <p>(1.2.1) Definir meta, determinar blocos de nível imediatamente abaixo, estabelecer requerimentos destes blocos e consolidar em especificações. (BP5)</p> <p>(1.2.2) Identificar soluções para os blocos e definir processos, técnicas e materiais necessários ou disponíveis para concretizá-las. (BP2)</p> <p>(1.2.3) Identificar executantes e responsável pela tarefa e demarcar os limites da tarefa e de suas atribuições. (BP4)</p> <p>(1.2.4) Levantar e organizar dados de custos/prazos. (BP4)</p> <p>(1.2.5) Realizar uma integração mentalizada. (BP4)</p>
<p>(2) Desenvolver os blocos gerenciais e administrativos.</p>	<p>(2.1) Determinar os blocos gerenciais e administrativos. (BP6)</p> <p>(2.2) Agregar à EDT as tarefas de gestões específica e administrativas. (BP5)</p> <p>(3.1) Determinar os dados do bloco com a maior exatidão possível. (BP5)</p>
<p>(3) Elaborar as declarações de trabalho dos blocos.</p> <p>(4) Consolidar dados</p>	<p>(4.1) Estrutura de decomposição de trabalho</p> <p>(4.1.1) Consolidar declaração de trabalho (matriz responsável/tarefa e matriz de controle de contratos). (BP8)</p> <p>(4.1.2) Consolidar requerimentos (árvore de especificações). (BP5)</p> <p>(4.2) Consolidar orçamento-mestre no Plano de Contas. (BP5)</p> <p>(4.3) Cronograma-mestre</p> <p>(4.3.1) Levantar ou avaliar as durações dos blocos da EDT. (BP7)</p> <p>(4.3.2) Obter um diagrama ou rede de procedência. (BP7)</p> <p>(4.3.3.) Montar um cronograma-mestre. (BP7)</p> <p>(4.3.4) Organizar outros cronogramas parciais, a partir do cronograma-mestre. (BP7)</p> <p>(4.4) Consolidar escopo/declaração de trabalho. (BP5)</p>

Figura 5.3 Correspondência entre atividades de desenvolvimento da EDT e práticas básicas da ISO/IEC TR 15504

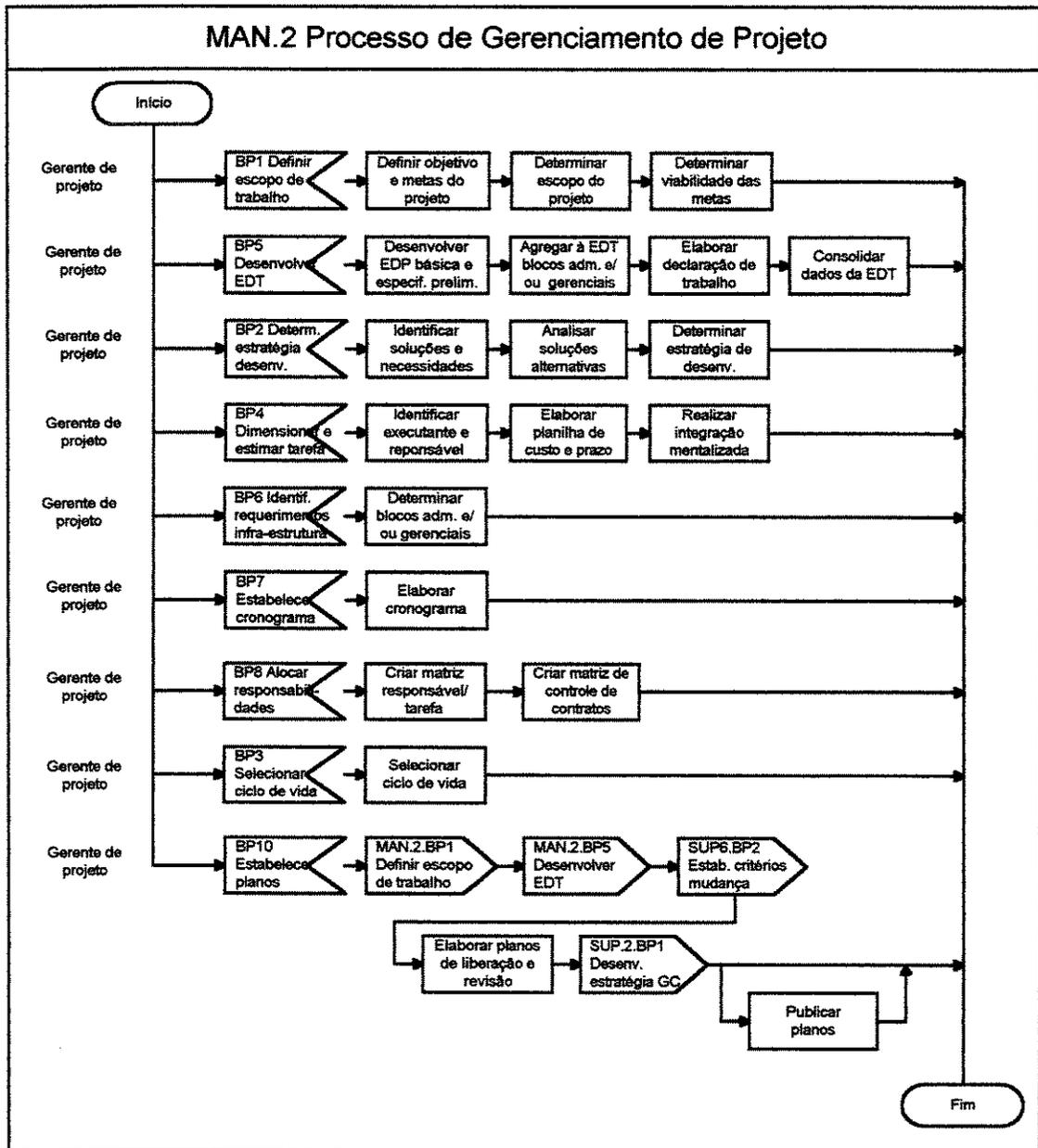
5.2.1 Escopo

Nome do processo:		MAN.2 Processo de gerência de projeto	
Objetivo:	O propósito deste processo é identificar, estabelecer, coordenar e monitorar atividades, tarefas e recursos necessários para um projeto produzir um produto e/ou serviço reunindo os requerimentos.		
Entradas:	2) Modelo de ciclo de vida 22) Análise de risco 23) Estratégia/plano de gerenciamento de riscos 51) Contrato 52) Especificação de requerimentos (cliente, software, sistema) 87) Mecanismo de comunicação 91) Estratégia / plano de gerência de configuração	Saídas:	5) Cronograma 6) Estrutura de decomposição de trabalho 17) Plano de projeto 21) Resultado de análise 30) Estratégia / plano de revisão 50) Compromisso / acordo 69 Plano de liberação 204) Planilha de custo e prazos da EDT
Início do processo:	Definição de escopo e objetivos de projeto.		
Conteúdo:	Define o escopo do trabalho; determina estratégia de desenvolvimento; seleciona modelo de ciclo de vida de software; dimensiona e estima tarefas e recursos; desenvolve estrutura de decomposição de trabalho; identifica requerimentos de infra-estrutura; estabelece cronograma de projeto; aloca responsabilidades; identifica interfaces; estabelece e implementa planos de projeto; acompanha progresso contra planos e age para corrigir desvios.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> ● a viabilidade de obtenção de metas do projeto com recursos disponíveis e restrições estará avaliada; ● as tarefas e recursos necessários para completar o trabalho estarão dimensionados e estimados; ● interfaces entre elementos no projeto e com outros projetos ou unidades organizacionais estarão identificadas e monitoradas; ● planos para execução do projeto estarão desenvolvidos e implementados; ● progresso do projeto estará monitorado e relatado; ● ações para corrigir desvios do plano e prevenir repetição de problemas identificados no projeto estarão tomadas quando o alvo do projeto não foi atingido. 		
Clientela a ser atendida:	Gerente de projeto		
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: MAN.2.BP1, MAN.2.BP2, MAN.2.BP3, MAN.2.BP4, MAN.2.BP5, MAN.2.BP6, MAN.2.BP7, MAN.2.BP8, MAN.2.BP10.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>		

5.2.2 Macro-diagrama de MAN.2 Processo de Gerenciamento de projeto



5.2.3 Fluxograma



5.2.4 Atividades

A seqüência das atividades deste processo, descritas para as etapas de *Planejamento Preliminar e Aprovação*, merecem as considerações a seguir:

- 1) as tarefas não são absolutamente seqüenciais nem inteiramente estanques, mas há uma ordem aproximadamente cronológica de seus términos mas não dos inícios;
- 2) embora deva-se cristalizar os resultados dos estudos e certas decisões, em determinados momentos, poderá haver necessidade de revê-los e modificá-los; mesmo dando por encerrada a elaboração de produtos de trabalho, eles poderão ser atualizados, como consequência de entendimentos na consolidação do projeto (etapa de *Aprovação*);
- 3) entretanto, uma vez atingida a etapa de *Aprovação*, nada mais deverá ser modificado, a não ser para detalhamento do que foi aprovado; exceto, evidentemente, por motivos relevantes, mediante aprovação de todos os envolvidos, tal como se faz com um contrato.

MAN.2.BP1 Definir o escopo de trabalho

Definir o trabalho a ser empreendido pelo projeto e determinar que a obtenção das metas de projeto é viável com os recursos disponíveis e restrições.

MAN.2.BP1 Atividade Definir objetivo e metas de projeto

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
22) Análise de risco	17) Plano de projeto
23) Estratégia/plano de gerência de risco	
51) Contrato	
52) Especificação requerimentos (cliente)	

- a. Declarar o objetivo do projeto através de um enunciado precisa e inequivocadamente definido, e não mais que isso [Valeriano, 1998], ou seja, declarar *o quê fazer* e não *o como fazer*. Caso exista mais que um objetivo, é necessário fixar um deles como principal e os demais como secundários.

Objetivo é aquilo (produto ou serviço) que será aceito e recebido pelo cliente/usuário, dando por encerrado o projeto; *como fazê-lo*, quais suas características, custo e prazos são partes dos planejamentos do projeto.

Baseado nos produtos de entrada, a redação do objetivo deve conter:

- a *ação*, definida por um verbo - preferencialmente no infinitivo - e que deve iniciar a declaração do objetivo (p.ex.: desenvolver, levantar, obter, transportar, etc.);
 - o *objeto*, sobre o qual a ação se exerce e/ou da qual ele resulta (p. ex.: uma simulação, um software, um dispositivo, etc.);
 - *requerimentos, restrições ou condições* complementares: de desempenho, de tempo, de local, de qualidade, de áreas de aplicação, etc.
- b. Declarar as metas do projeto, seguindo as orientações de redação de objetivo, considerando que as *metas* são objetivos parciais, geralmente a cargo das frações executivas do projeto (os blocos de estrutura de decomposição do trabalho, como apresentado mais adiante). As metas de qualidade são estabelecidas para os vários pontos de verificação (checkpoints) dentro do ciclo de vida.
- c. Iniciar a elaboração do produto 17) *Plano de projeto*, utilizando objetivo e metas de projeto definidos.

MAN.2.BP1 Atividade Definir escopo do projeto

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
22) Análise de risco	6) Estrutura de decomposição de trabalho
23) Estratégia/plano de gerência de risco	17) Plano de projeto
51) Contrato	
52) Especificação requerimentos (cliente)	

- a. Delimitar as funções que o software deve realizar de forma quantitativa, considerando requerimentos de desempenho, restrições, interfaces e confiabilidade. Utilizar como base os produtos de entrada desta atividade e não considerar *como* essas funções serão implementadas [Pressman, 1995]. Os produtos de entrada são a base para esta atividade.

A definição do escopo deve:

- declarar as principais funções do software e, se necessário, refinar para obter mais detalhes;
- declarar explicitamente o desempenho através da identificação de requerimentos de processamento e de tempo de resposta (p. ex., número de usuários simultâneos, quantidade de clientes, tempo máximo de resposta permitido);
- declarar as restrições e/ou as limitações impostas ao software pelo hardware externo, memória disponível ou outros sistemas existentes (p. ex., custo do produto que restringe o tamanho da memória);

- declarar as interfaces do software, ou seja, todas as interações do software com o seu ambiente de execução:
 - hardware que executa o software (p. ex., processador, periféricos) e dispositivos que são indiretamente controlados pelo software (máquinas, displays);
 - software que já existe e que deve ser ligado ao novo software (p. ex., rotinas de acesso a banco de dados, pacotes de sub-rotinas, sistema operacional);
 - pessoas que fazem uso do software por meio de terminais ou outros dispositivos de entrada/saída;
 - procedimentos que procedem ou sucedem o software como uma série sequencial de operações;
- declarar a confiabilidade requerida, se for possível;
- descrever os fatores mitigantes (p. ex., algoritmos desejados que são bem compreendidos e estão disponíveis).

O escopo do projeto de software é elaborado durante a etapa de *Planejamento Preliminar* e pode sofrer modificações até o término da etapa de *Aprovação*. Entretanto, uma vez aprovado, tanto o escopo como o objetivo tornam-se peças que só podem ser alteradas com a formal concordância de todas as partes envolvidas.

- b. Atualizar o produto de trabalho 17) *Plano de projeto* com o escopo.
- c. Gerar o produto de trabalho 6) *Estrutura de decomposição do trabalho* com um esboço da estrutura de decomposição do produto - EDP, a partir do objetivo do projeto. O produto deve ser decomposto em seus componentes físicos e/ou em suas partes lógicas: subsistemas, partes destes, itens e assim sucessivamente (veja *MAN.2.BP5 Desenvolver EDT*). Nesta fase, decompor 2 ou três níveis.

MAN.2. BP1 Atividade Determinar viabilidade das metas

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	17) Plano de projeto

- a. Analisar o ambiente do sistema e identificar riscos e oportunidades.
- b. Determinar que a obtenção das metas de projeto é viável com os recursos disponíveis e restrições. Esta atividade deve basear-se nos resultados dos trabalhos da prática *MAN.2.BP5 Desenvolver a EDT*.
- c. Atualizar o produto 17) *Plano de trabalho* com o resultado das tarefas acima.

MAN.2.BP2 Determinar estratégia de desenvolvimento

Avaliar opções disponíveis para obtenção das metas de projeto e determinar, baseado nos riscos e oportunidades, qual estratégia será adotada.

MAN.2.BP2 Atividade Identificar soluções e necessidades

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho 17) Plano de projeto	21) Resultado de análise

- a. Identificar várias soluções para os requerimentos dos blocos e definir processos, técnicas e materiais necessários ou disponíveis para concretizá-las.
- b. Registrar os resultados desta atividade no produto de trabalho 21) *Resultado de análise*, item "o que foi analisado".

MAN.2.BP2 Atividade Analisar soluções alternativas

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto 21) Resultado de análise	21) Resultado de análise

- a. Analisar as soluções alternativas considerando a obtenção das metas de projeto. Ainda que poucos detalhes sejam discutidos, as alternativas possibilitam a seleção de uma abordagem "melhor", dadas as restrições impostas por prazo de entrega, orçamento, disponibilidade de pessoal, interfaces técnicas e outros fatores [Pressman, 1995].
- b. Registrar a análise das soluções alternativas acrescentando ao produto de trabalho 21) *Resultado de análise*, os itens:
 - quem fez a análise;
 - os critérios de análise usados:
 - critério de seleção ou esquema de priorização usado;
 - critérios de decisão;
 - critérios de qualidade;
 - registros de resultados:

- o que foi decidido/selecionado;
 - razão para seleção;
 - suposições feitas;
 - riscos potenciais;
- aspectos de corretitude para analisar incluem:
 - completude;
 - facilidade de entendimento;
 - testabilidade;
 - verificabilidade;
 - viabilidade;
 - validade.

MAN2.BP2 Atividade Determinar estratégia de desenvolvimento

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho
17) Plano de projeto	17) Plano de projeto
21) Resultado de análise	

- a. Determinar a estratégia de desenvolvimento a ser adotada, considerando os produtos de entrada.
- b. Aplicar a prática básica MAN.2.BP3 Selecionar modelo ciclo de vida.
- c. Atualizar o produto de trabalho 17) *Plano de projeto* com a estratégia de desenvolvimento determinada.
- d. Atualizar o produto 6) *Estrutura de decomposição de trabalho* com a solução selecionada.

MAN.2.BP3 Selecionar modelo de ciclo de vida de software

Selecionar um modelo de ciclo de vida para o projeto o qual seja apropriado para o escopo, magnitude e complexidade do projeto¹.

¹ Esta prática base está relacionada com a MAN.2.BP1.

MAN.2.BP3 Atividade Selecionar ciclo de vida

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
2) Modelo de ciclo de vida	17) Plano de projeto
17) Plano de trabalho	

- a. Definir o ciclo de vida mais adequado para o projeto e registrá-lo no produto de trabalho 17) *Plano de projeto*.

Em [Pressman, 1995] são apresentados alguns modelos que podem ser adotados ou adaptados para as necessidades de um projeto em particular. A definição do ciclo de vida deve conter:

- o modelo selecionado ou uma referência a ele;
- a relação de produtos de trabalho desenvolvidos em cada fase;
- para cada produto de trabalho, a relação de seus atributos;
- os pontos de verificação (*checkpoints*).

Uma sugestão de produtos de trabalho e atributos pode ser encontrada na ISO/IEC TR 15504 [ISO15504, 1998].

MAN.2.BP4 Dimensionar e estimar tarefas e recursos

Dimensionar e estimar tarefas e recursos necessários para completar o trabalho pela avaliação das opções disponíveis para obtenção das metas do projeto e pela consideração de riscos e oportunidades existentes¹.

MAN.2.BP4 Atividade Identificar executantes e responsável

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- a. Identificar executantes, ou seja, qual órgão dispõe ou deverá desenvolver os processos, as técnicas e os materiais: entidade externa, órgão da própria organização para execução e gerente funcional, membro de equipe, etc. para gerenciar a tarefa. O ambiente deve ser esquadrihado para se identificar processos, técnicas e materiais disponíveis que satisfaçam às necessidades do bloco.

¹ Para a identificação de riscos existentes veja MAN.4.BP2.

- b. Identificar, se possível o líder do bloco ou o responsável por ele, bem como os participantes na execução da tarefa. É da maior importância que os compromentimentos com os membros da equipe do projeto sejam efetivados no decorrer desta tarefa, de maneira formal com os gerentes funcionais e com os responsáveis por contratações, admissões ou outras formas necessárias de movimentação de pessoal. O mesmo deverá ser feito quanto à cessão de materiais, equipamentos, áreas de trabalho, etc.
- c. Atualizar o produto de trabalho 6) *Estrutura de decomposição de trabalho* com executantes e responsáveis.

MAN.2.BP4 Atividade Elaborar planilha de custo e prazo

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	204) Planilha de custo e prazos da EDT

- a. Levantar, em cada bloco e para cada uma das tarefas, de forma paciente e criteriosa, os prazos e os custos associados a cada um de seru fatores produtivos e insumos (recursos e serviços). Estes dados devem ser metódica e progressivamente agregados, no produto de trabalho 204) *Planilha de custos e prazos da EDT*, de forma a servir de subsídios para a elaboração do cronograma-mestre e do orçamento-mestre do projeto.

MAN.2.BP4 Atividade Realizar integração mentalizada

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- a. Integrar mentalmente, agregando idealmente, todos os itens da EDT desde os blocos elementares, até chegar ao mais alto nível: o produto ou o sistema [Valeriano, 1998]. Durante este trabalho, novas tarefas e novos blocos poderão se mostrar necessários. Por exemplo, tarefas que emergem nos processos de integração, de transportes especiais, de ensaios e avaliações, as quais, por sua vez podem exigir dispositivos, ferramentas, equipamentos, etc. não diretamente integrante do produto e que somente agora podem vir à luz, ao se exercitar os processos conceituais de integração, de ensaios, etc.

Além disso, também poderão ser eliminados itens, partes de itens, requerimentos, tarefas, etc. cujas necessidades ou propriedades não se justifiquem. É importante que cada bloco tenha um objetivo facilmente identificável quando atingido, seja este uma parte do produto, uma tarefa gerencial ou administrativa e que responda e satisfaça uma necessidade também claramente identificada.

- b. Atualizar o produto de trabalho 6) *Estrutura de decomposição de trabalho*.

MAN.2.BP5 Desenvolver estrutura de decomposição de trabalho

Desenvolver uma estrutura de decomposição de trabalho incorporando tarefas de projeto (enunciáveis e seqüência) e relacionando estas com os recursos requeridos para realizá-las e a estratégia a ser seguida.

MAN.2.BP5 Atividade Desenvolver EDP

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho
17) Plano de projeto	17) Plano de projeto
51) Contrato	
52) Especificação de requerimentos (cliente)	

- Definir um sistema de identificação para numeração dos blocos EDT [Valeriano, 1998]. Por exemplo, pode-se utilizar a apresentada na Figura 3.1.
- Desenvolver a EDP básica, até o segundo ou terceiro níveis, conforme descrito no item 3.2 *Estrutura de Decomposição do Trabalho - EDT*, baseando-se nos produtos de entrada. Definir meta, determinar blocos de nível imediatamente abaixo, estabelecer requerimentos destes blocos, consolidar em especificações e atualizar o produto 6) *Estrutura de decomposição de trabalho*.
- Atualizar o produto 17) *Plano de projeto*.
- Aplicar as práticas *MAN.2.BP2 Determinar estratégia de desenvolvimento* e *MAN.2.BP4 Dimensionar e estimar cada tarefa* a cada um dos blocos da EDP básica para complementá-la.

MAN.2.BP5 Atividade Agregar à EDT blocos administrativos e/ou gerenciais

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- Aplicar a prática *MAN.2.BP6 Identificar requerimentos de infra-estrutura* para agregar os blocos administrativos e/ou gerenciais à EDT.

MAN.2.BP5 Atividade Elaborar declaração de trabalho

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- a. Elaborar a declaração de trabalho para cada bloco da EDT, representando-a no esquema mostrado no item 3.2 *Estrutura de Decomposição de Trabalho -EDT*.
- b. Atualizar o produto 6) *Estrutura de decomposição de trabalho*, com a declaração de trabalho de cada bloco.

MAN.2.BP5 Atividade Consolidar dados da EDT

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho
17) Plano de projeto	17) Plano de projeto
	50) Compromisso / acordo

- a. Aplicar a prática *MAN.2.BP8 Alocar responsabilidades* para consolidar as declarações de trabalho da EDT.
- b. Gerar a árvore de especificações de requerimentos para consolidar os requerimentos, usando a hierarquização do produto, como estabelecida na EDT, para relacionar, da mesma forma, os requerimentos e as especificações do produto e de suas partes constitutivas.
- c. Aplicar a prática *MAN.2.BP7 Estabelecer cronograma* para consolidar os prazos.
- d. Consolidar custos, num plano de contas que apresente os recursos orçamentários para todo o período de duração do projeto.
- e. Consolidar escopo, compatibilizando-o com as declarações de trabalho dos blocos para garantir consistência de informações.
- f. Atualizar os produtos 6) *Estrutura de decomposição de trabalho* e 17) *Plano de projeto*.
- g. Gerar o produto 50) *Compromisso / acordo*.

MAN.2.BP6 Identificar requerimentos de infra-estrutura

Identificar e selecionar elementos de ambiente e de recurso humano necessários para suportar a estratégia e execução de projeto.

MAN.2.BP6 Atividade Determinar blocos administrativo e/ou gerenciais

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- a. Determinar os blocos administrativos e/ou gerenciais, se o gerente assim decidir. Demembrar as suas tarefas em suas componentes menores: serviços técnicos (p. ex.: consultorias, dimensionamentos, ensaios, montagens e integrações, por exemplo), serviços administrativos (p. ex.: contratos, compras), controles (p. ex.: físicos, financeiros, cronológicos).

Os cuidados a tomar durante a elaboração da EDP são válidos para os módulos da administração e gerenciais:

- não se deve criar tarefa que não seja justificada por necessidade de tarefa (ou bloco) de nível superior;
 - todas as tarefas devem ser rastreáveis ao objetivo do projeto, e devem concorrer para a sua execução.
- b. Aplicar as práticas *MAN.2.BP2 Determinar estratégia de desenvolvimento* e *MAN.2.BP4 Dimensionar e estimar tarefa* para os blocos administrativos e/ou gerenciais.

MAN.2.BP7 Estabelecer cronograma de projeto

Estabelecer o cronograma de projeto baseado na estrutura de decomposição de trabalho, estimativas e elementos de infra-estrutura.

MAN.2.BP7 Atividade Elaborar cronograma

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
204) Planilha de custos e prazos da EDT	5) Cronograma

- a. Elaborar o produto *5) Cronograma*, baseado nas descrições do item *3.2 Estrutura de Decomposição de Trabalho -EDT* e produto de entrada, realizando a seguinte seqüência de tarefas:

- levantar ou avaliar as durações das tarefas do projeto (ou blocos da EDT);
- relacionar umas às outras, consideradas as precedências e condicionantes existentes, isto é, obter um diagrama ou rede de precedência;
- montar um cronograma-mestre, ou seja, "amarrar ao calendário" o diagrama de precedência das tarefas de maior nível do projeto; e
- organizar os outros cronogramas parciais, a partir do cronograma-mestre.

MAN.2.BP8 Alocar responsabilidades

Identificar os indivíduos e grupos específicos (que contribuem para e são impactados pelo projeto); alocá-los para responsabilidades específicas e garantir que os compromissos são compreendidos e aceitos, fundamentados e alcançáveis.

MAN.2.BP8 Atividade Criar matriz responsável/tarefa

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- a. A partir das declarações de trabalho dos blocos, gerar a *Matriz de responsáveis/tarefas*, e atualizar o produto 6) *Estrutura de decomposição de trabalho*.

MAN2.BP8 Atividade Criar matriz de contratos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	6) Estrutura de decomposição de trabalho

- b. A partir das declarações de trabalho dos blocos, gerar a *Matriz de controle de contratos* e atualizar o produto 6) *Estrutura de decomposição de trabalho*.

MAN.2.BP10 Estabelecer planos

Prover um mecanismo para garantir que planos de projeto estão formalmente desenvolvidos e disponíveis a todos envolvidos com o projeto.

MAN.2.BP10 Atividade Elaborar planos de liberação e revisão

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
6) Estrutura de decomposição de trabalho	30) Plano de revisão
17) Plano de projeto	69) Plano de liberação

- a. Elaborar os produtos de trabalho 30) *Plano de revisão* e 69) *Plano de liberação*.

As informações a seguir são comuns a ambos os planos e estão baseadas na ISO/IEC TR 15504:

- identificação do proprietário do plano;
- objetivo do que será realizado;
- suposições feitas;
- restrições;
- riscos;
- tarefas a serem realizadas;
- cronograma, marcos e datas-alvo;
- dependências críticas;
- disposição de manutenção para o plano;
- método/abordagem para realizar o plano;
- identificação de:
 - posse de tarefa;
 - critério de qualidade;
 - auditoria a ser executada;
 - produtos de trabalho requeridos;
- recursos para realizar os objetivos do plano:
 - staff;
 - tempo;
 - materiais/suprimentos;
 - orçamento;
- plano de contingência para tarefas não-completadas;
- status do plano (aprovado, em revisão, em elaboração, etc.).

Para o Plano de liberação, acrescentar as informações de identificação, baseadas na ISO/IEC TR 15504:

- funcionalidade a ser concluída em cada liberação;
- componentes de software requeridos (i.e.: hardware, software, documentação, etc.);
- mapeamento de requisitos e requerimentos de cliente satisfeitos para liberações específicas de produto .

Para o Plano de revisão, acrescentar as informações a seguir que estão baseadas na ISO/IEC TR 15504:

- definição de:
 - o quê será revisado;
 - papéis e responsabilidades de revisores;
 - critério para revisão¹;
 - tempo de preparação esperado;
 - cronograma para revisões;
- identificação de:
 - procedimentos para condução de revisão;
 - entradas e saídas de revisão;
- perícia esperada em cada revisão;
- registros de revisão para manter;
- medidas de revisão para manter;
- recursos e ferramentas alocados para a revisão.

A seguir é apresentado um exemplo de papéis e responsabilidades, baseados em [Pacheco et al., 1997]:

- *líder/coordenador de revisão*, cuja função é assegurar que as atividades de revisão sejam planejadas e executadas corretamente; durante as reuniões, servir de moderador a fim de que as discussões resultem em definições claras de ações futuras;
- *autor/produtor/apresentador*, cuja tarefa é introduzir o produto para o restante do grupo e também assumir a responsabilidade pelas correções dos erros ou falhas que forem detectadas;

¹ Fornecido pela aplicação da prática básica SUP.6.BP2 Estabelecer critérios de revisão.

- *secretário*, que tem o papel de anotar todas as decisões tomadas na reunião, assegurando que as modificações requeridas sejam efetivamente executadas; também é sua responsabilidade preencher o registro de revisão;
- *representante técnico externo*, é um papel que deve ser desempenhado por alguém experiente, preferencialmente não vinculado ao projeto, e que possa estar atento aos aspectos de viabilidade técnica, adequação e qualidade do produto final;
- *representante do usuário/cliente*, que deve assegurar que o sistema em desenvolvimento atenda a seus interesses e que a solução encontrada indique o rumo correto.

MAN.2.BP10 Atividade Publicar planos

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	não há
30) Plano de revisão	
69) Plano de liberação	
91) Plano de gerência de configuração	
87) Mecanismo de comunicação	

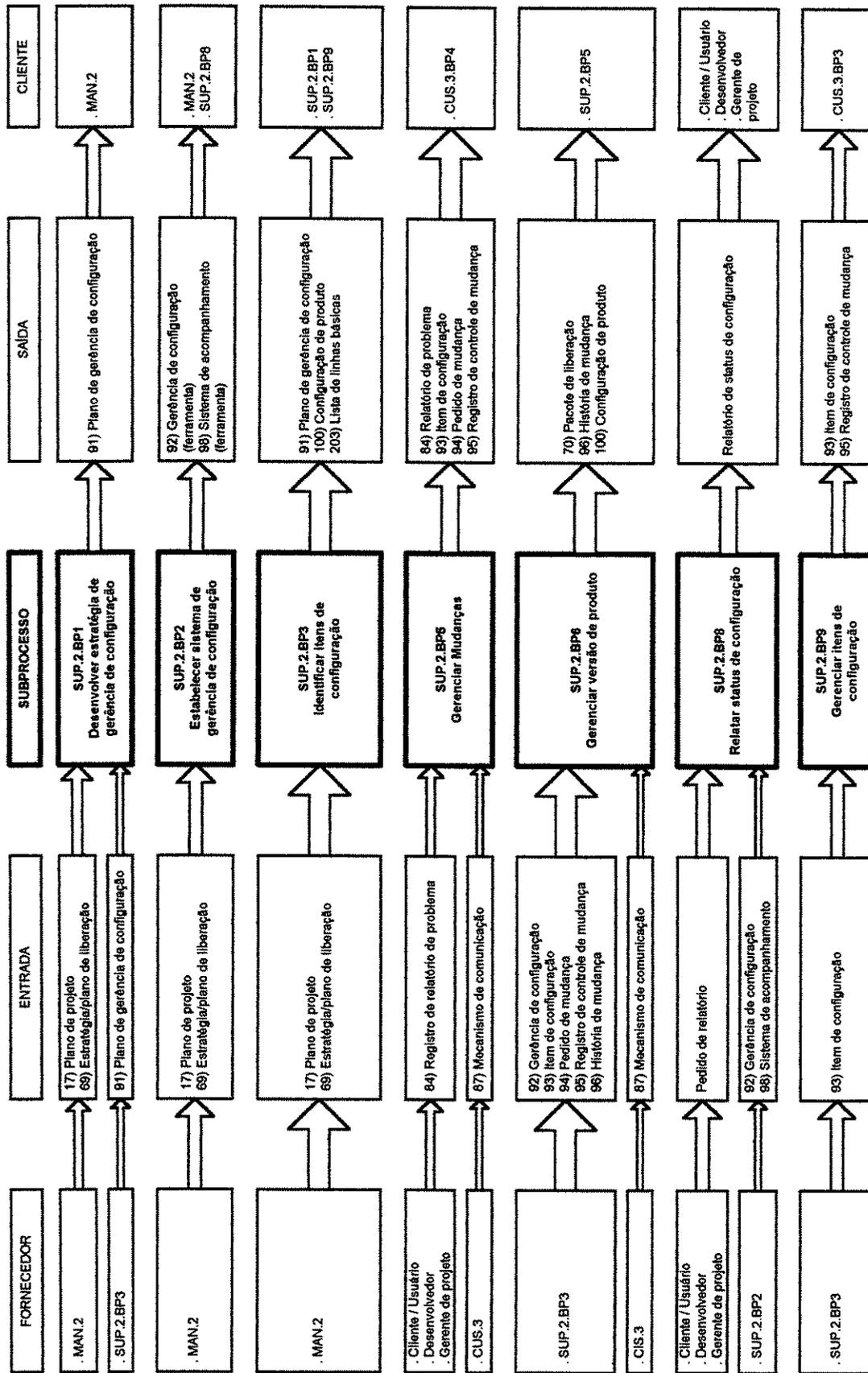
- Divulgar os produtos de entrada que representam planos a todos os interessados, utilizando o mecanismo definido em 87) *Mecanismo de comunicação*.

5.3 SUP.2 Processo de Gerência de Configuração

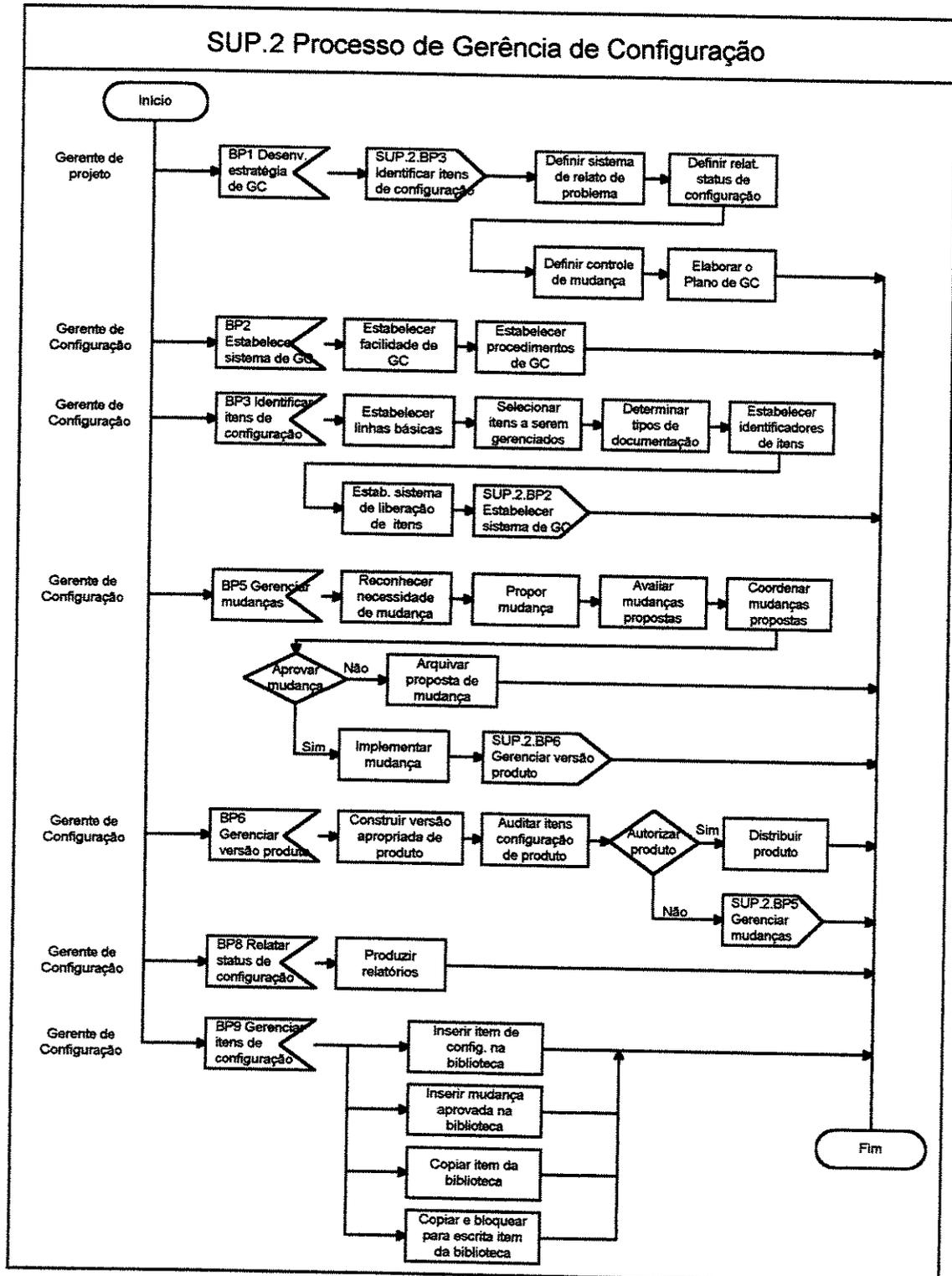
5.3.1 Escopo

Nome do processo:		SUP.2 Processo de gerência de configuração	
Objetivo:	O propósito deste processo é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto.		
Entradas:	17) Plano de projeto 69) Estratégia/plano de liberação 84) Relatório de problema 87) Mecanismo de comunicação 96) História de mudança	Saídas:	70) Pacote de liberação 84) Relatório de problema 91) Plano de gerência de configuração 92) Gerência de configuração - ferramenta 93) Item de configuração 94) Pedido de mudança 95) Registro de controle de mudança 96) História de mudança 98) Sistema de acompanhamento - ferramenta 100) Configuração de produto 203) Lista de linhas-básicas
Início do processo:	Desenvolver estratégia de gerência de configuração.		
Conteúdo:	Desenvolve uma estratégia de gerência de configuração; estabelece sistema de gerência de configuração; identifica itens de configuração; mantém descrição de item de configuração; gerencia mudanças; gerencia versões de produtos; mantém história de item de configuração; relata status de configuração; gerencia liberação e entrega de itens de configuração.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • uma estratégia de gerência de configuração estará desenvolvida; • todos os itens gerados pelo processo ou projeto estarão identificados, definidos e com sua linha básica estabelecida; • modificações e liberações dos itens estarão controladas; • o status dos itens e requisitos de modificação estarão registrados e relatados; • a completude e consistência dos itens estará garantida; • armazenamento, manipulação e entrega dos itens estarão controlada. 		
Cientela a ser atendida:	Cliente / usuário, desenvolvedor, gerente de projeto, MAN.2, CUS.3.		
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: SUP.2.BP1, SUP.2.BP2, SUP.2.BP3, SUP.2.BP5, SUP.2.BP6, SUP.2.BP8, SUP.2.BP9.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>		

5.3.2 Macro-diagrama de SUP.2 Processo de gestão de configuração



5.3.3 Fluxograma



5.3.4 Atividades

SUP.2.BP1 Desenvolver estratégia de gerência de configuração (GC)

Determinar estratégia de gerência de configuração, incluindo atividades de gerência de configuração e cronograma para execução destas atividades.

SUP2.BP1 Atividade Definir sistema de relato de problema

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
91) Plano de gerência de configuração	91) Plano de gerência de configuração

- a. Identificar os requerimentos para relatar problemas, considerando os seguintes aspectos:
 - as diferentes circunstâncias em que um problema ocorre, tais como durante revisão técnica, teste de integração, condução de teste de aceitação, operação do item no campo, entre outras;
 - a fase do ciclo de vida à qual o item pertence;
 - informações necessárias em cada fase do ciclo de vida;
 - formulários requeridos para relatar problema em cada fase do ciclo de vida.
- b. Identificar requerimentos para analisar o problema e tomar decisões, considerando os aspectos:
 - mecanismo de priorização ou classificação de problema;
 - formulários para declarar prioridade de problema, causa de falha e ação corretiva a ser realizada e responsáveis pela análise.
- c. Estabelecer um sistema de relato de problema que implemente os requerimentos identificados.
- d. Atualizar o produto *91) Plano de gerência de configuração* com o sistema de relato de problema estabelecido.

SUP2.BP1 Atividade Definir controle de mudança

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
não há	91) Plano de gerência de configuração

- a. Definir controle de mudança e atualizar o produto 91) *Plano de gerência de configuração* com o controle de mudança definido.

A seguir é apresentado um exemplo, adaptado de [Buckley, 1996] e [Pressman, 1995].

1. Reconhecer a necessidade de mudança, através de:

- identificação do problema e inserção dele no Sistema de Relato de Problema;
 - revisão do relato de problema para garantir que a declaração está clara, que todos os elementos essenciais foram registrados e que o problema - como declarado - é uma falha válida;
 - fixação e registro no Sistema de Relato de Problema de uma prioridade de ação e alocação de recursos para determinar a causa da falha;
 - registro da causa da falha no Sistema de Relato de Problema, quando ela for determinada.
 - determinação de ação corretiva e registro dela no Sistema de Relato de Problema; as informações determinadas nesta tarefa serão usadas para iniciar a próxima atividade.
2. Propor mudança que represente uma solução completa para o problema relatado. Esta proposta precisa relacionar todos os itens de configuração que seriam afetados pela mudança, qual a mudança que seria feita, o custo e o prazo para liberação das mudanças implementadas, considerando-se todos as atividades de garantia de qualidade.
3. Analisar proposta de mudança quanto ao impacto nos custos e prazos e ao tipo de solução do problema (o mais completa e consistente possível).
4. Se proposta de mudança não aprovada, arquivá-la em local apropriado e encerrar a atividade.
5. Implementar a mudança e gerenciar a versão do produto.

SUP2.BP1 Atividade Definir relatório de status de configuração

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
91) Plano de gerência de configuração	91) Plano de gerência de configuração

- a. Identificar as informações a serem registradas e os relatórios a serem feitos.

Estas informações vêm do contrato e de políticas da empresa. O contrato identifica os itens a serem entregues e os relatórios a serem feitos. Também estabelecem as formas requeridas para reportagem de mudanças, o formato e a frequência de relatórios de reportagem de status de configuração requeridos contratualmente. Requerimentos especiais, que não constam do contrato também podem existir. As políticas da empresa declaram como a empresa está organizada e opera.

A Tabela 5.2 mostra alguns exemplos de relatórios típicos de status de configuração que foram adaptados de [Buckley, 1996].

Tabela 5.2 Exemplo de relatórios de status de configuração		
Número	Cobertura do Relatório	Notas
1	Revisão de especificação	Versão corrente em uso e história de mudanças, ambos propostos e aprovados.
2	Revisão de software	
3	Componentes de itens de configuração	Lista de todos os componentes de todos os itens de configuração, por número de versão do arquivo.
4	Listagem de contratos	Acompanha todas as mudanças para contratos existentes, tanto para o contrato principal como para os subcontratos.
5	Acompanhamento de mudança	Acompanha todas as mudanças, desde quando proposta até seu término de implementação.
6	Relatórios de problema	Acompanha todos os relatórios de problema, desde o relato até o fechamento.
7	Relatório de implementação de mudança	Status de implementação de mudanças aprovadas.
8	História de manutenção	Ações de manutenção executadas sobre cada unidade no campo.
9	Ações de auditoria	Status de cada item de ação desde seu início até o término.

- b. Atualizar o produto de trabalho 91) *Plano de gerência de configuração* com os relatórios definidos.

SUP2.BP1 Atividade Elaborar Plano de Gerência de Configuração (GC)

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	91) Plano de gerência de configuração
69) Estratégia/plano de liberação	
91) Plano de gerência de configuração	

a. Complementar o produto de trabalho 91) *Plano de gerência de configuração* com os seguintes tópicos¹:

- *Introdução* com informações sobre: propósito do documento; escopo do plano (pode ter uma lista dos itens que serão controlados); referências, definições, acrônimos e abreviaturas; e uma introdução para as próximas seções, identificando cada seção e apêndice com pequena descrição sobre seu conteúdo;
- *Organização*, que descreve qual a organização da gerência de configuração; pode-se utilizar um organograma para ilustrá-la; a Figura 5.4 mostra um exemplo (adaptado de [Buckley, 1996]);

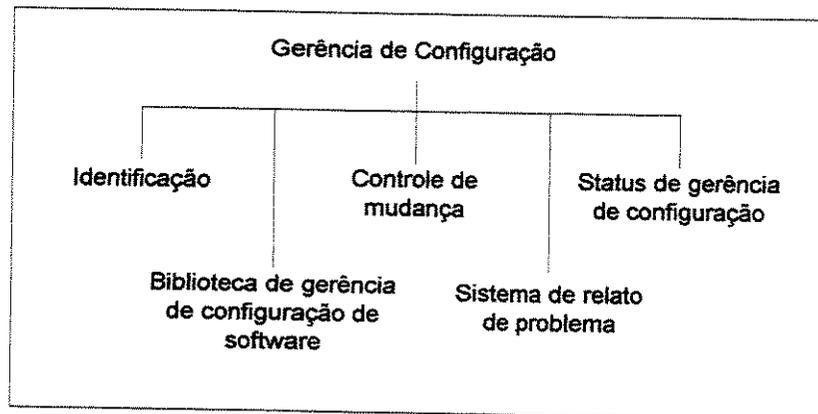


Figura 5.4 Exemplo de organograma de Gerência de Configuração

- *Tarefas*, contendo descrição das tarefas que serão executadas pela Gerência de Configuração²; o grau de detalhamento das tarefas depende da cultura de gerência de configuração na empresa: quanto menor o detalhamento no plano, maior o detalhamento na descrição dos procedimentos de suporte à gerência de configuração;
- *Cronograma*, que deve: identificar quando as tarefas serão realizadas (isto está fortemente amarrado ao cronograma do projeto); definir marcos que mostram

¹ Baseado em [Buckley, 1996] e na ISO/IEC TR 15504 [ISO15504, 1998].

² A tarefa de auditoria pertence ao processo SUP.7 Auditoria da categoria Processos de Suporte ao Ciclo de Vida, portanto está fora do escopo deste trabalho.

quando uma capacidade funcional estará sendo obtida (p. ex.: o primeiro relatório de status de configuração, a conclusão do teste de integração, etc.);

- *Recursos*, que define os recursos humanos (quantos, quando e requerimentos de habilidade associados) e materiais (equipamento para executar tarefas de gerência de configuração, tais como computador, impressora, mesa, telefone, máquina de fotocópia, fita de back up; software para gerência de configuração, sistema de relato de problemas, ferramenta para reportagem de status de configuração, etc.);
- *Treinamento*, que define o treinamento necessário para os envolvidos nas atividades de gerência de configuração, que pode ser em etapas: visão geral do processo de gerência de configuração; treinamento organizacional (que define os procedimentos); e treinamento na ferramenta de gerência de configuração
- *Procedimentos*, contendo definição dos procedimentos para controle de mudança.

SUP.2.BP2 Estabelecer sistema de gerência de configuração (GC)

Estabelecer um sistema de gerência de configuração incluindo bibliotecas, procedimentos e ferramentas.

SUP.2.BP2 Atividade Estabelecer facilidade de GC

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	92) Gerência de configuração - ferramenta
69) Estratégia/plano de liberação	98) Sistema de acompanhamento

- a. Estabelecer um sistema de gerência de configuração que atenda aos requerimentos¹ descritos a seguir. Este estabelecimento é representado pelos produtos de trabalho 92) *Gerência de configuração - ferramenta* e 98) *Sistema de acompanhamento*.

O sistema deve possuir mecanismos para:

- armazenamento dos itens em um repositório único para o projeto;
- controle e acompanhamento de versão de item;
- controle e acompanhamento de mudança;
- controle e acompanhamento de produto;
- controle e acompanhamento de versão de produto.

¹ Os requerimentos citados estão baseados em [Mikkelsen et al., 1997] e [Buckley, 1996].

Para controlar a configuração, o sistema requer recursos para:

- controle de versão, armazenamento e recuperação de itens de configuração (e suas versões);
- compartilhamento e transferência de itens de configuração entre grupos afetados;
- controles efetivos sobre acesso aos itens;
- descrição de itens de configuração;
- recuperação de versões de arquivo de itens de configuração;
- criação correta de produtos a partir da biblioteca de gerência de configuração de software;
- poder re-criar quaisquer configurações de liberação ou teste;
- habilidade para relatar status de configuração;
- mudar pedidos de mudança para itens de configuração a serem acompanhados.

Para acompanhar a gerência de configuração, o sistema requer:

- habilidade para registrar informação de dono de processo e cliente;
- habilidade para registrar informação de configuração de sistema relacionada;
- habilidade para registrar informação necessária sobre problema ou ação:
 - data de abertura e data prevista de fechamento;
 - severidade/criticalidade de item;
 - status necessário de quaisquer problemas ou ações;
 - informação sobre o dono do problema ou da ação;
 - prioridade de resolução de problema;
- habilidade para registrar proposta de solução ou de plano de ação;
- habilidade para prover informação de status de configuração;
- recursos que permitam que a informação esteja disponível para todos com a necessidade de conhecê-la;
- sistema(s)/registros de controle de mudança integrados;

SUP2.BP2 Atividade Estabelecer procedimentos de GC

- a. Estabelecer procedimentos para controle de mudança.

A seguir é apresentado um exemplo baseado em [Pressman, 1995].

1. Registrar, durante a reunião de revisão, todas as questões que foram levantadas.
2. Resumir, ao final da reunião, as questões levantadas e gerar uma lista de questões de revisão.

Esta lista serve a dois propósitos: identificar as áreas problemáticas do produto e servir como lista de conferência de itens de ação que oriente o autor quando correções forem feitas.

3. Elaborar um relatório de revisão resumido, que deve responder às três perguntas: *o quê foi revisado? quem fez a revisão? quais foram as descobertas e conclusões?*

Em anexo, pode suplementar com a lista de revisão.

4. Identificar as ações corretivas requeridas, tais como:
 - identificação de risco;
 - lista priorizada de desvios e problemas descobertos;
 - ações e tarefas a serem executadas para fixar o problema;
 - alocação de ação corretiva para indivíduos;
 - status e datas de fechamento para problemas identificados.
5. Gerar o produto de trabalho *31) Registro de revisão* com as informações dos itens acima.

- b. Definir a Biblioteca de Gerência de Configuração de Software (Software Configuration Management Library - SCML), que deve possuir as seguintes habilidades [Buckley, 1996]:
 - entrada de novos arquivos;
 - restrição de acesso e uso de arquivos específicos;
 - designação de arquivos como parte de uma grande coleção;
 - uso de software de suporte e arquivos de script sob condições controladas para transformar arquivos fonte e/ou intermediários em arquivos finais (p. ex.: comando MAKE do UNIX).
 - manutenção de descrição de item de configuração; a descrição deve conter, pelo menos os seguintes dados: autor do item, data de entrada ou de liberação na SCML; identificação de versão de liberação; o caminho (pathname) para o arquivo e Log que possui um comentário do autor que descreve porque esta versão particular de liberação de arquivo foi criada;
 - suporte à relato de problemas.
- c. Estabelecer procedimentos para utilizar a biblioteca SCML [ISO15504, 1998].

A Figura 5.5 mostra um exemplo do processo de entrada de novo item na SCML, adaptado de Buckley [Buckley, 1996]. Os processos de inserção de item e de inserção de mudanças

aprovadas na SCML são similares ao processo de entrada de novo item, conforme mostrado na Figura 5.5. O processo de controle de acesso deve ser definido de acordo com as facilidades que a ferramenta de Gerência de Configuração possua.

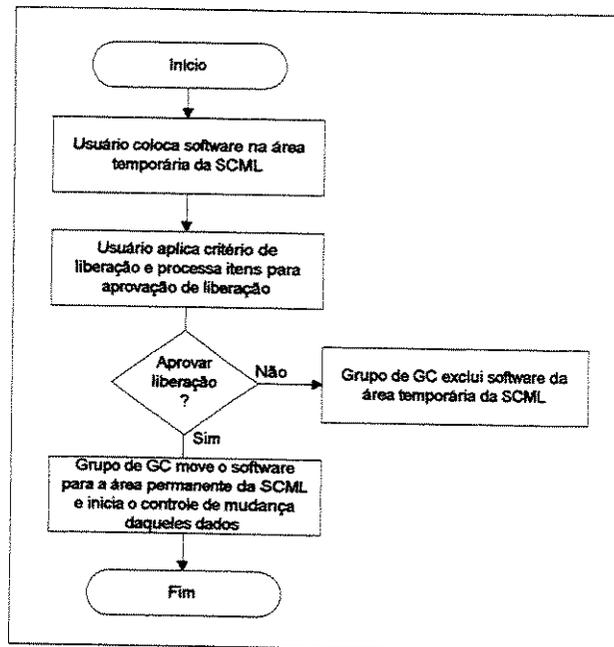


Figura 5.5 Exemplo de processo para entrada de novo item na SCML

SUP.2.BP3 Identificar itens de configuração

Identificar itens de configuração, tais como sistema de software, módulos, componentes e documentos relacionados pela identificação da documentação que estabelece a linha-base; as referências de versão e outros detalhes de identificação relevantes.

SUP2.BP3 Atividade Estabelecer linhas básicas

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto 69) Estratégia de liberação	203) Lista de linhas básicas

- a. Estabelecer as quatro linhas básicas de projeto: funcional, alocada, configuração de desenvolvimento e produto [Buckley, 1996] e registrar no produto de trabalho 203) *Lista de linhas básicas*.

A estas linhas básicas estão associadas as seguintes perguntas: quais são os componentes desta linha básica (documentos ou outros produtos) e quando ela é estabelecida? A Tabela 5.3

relaciona, para cada linha básica, o conteúdo e quando ela é estabelecida. Os atributos da tabela podem variar de acordo com o projeto que será executado e foram adaptados de [Buckley, 1996].

Tabela 5.3 Estabelecimento de linhas básicas de projeto		
Linha básica	Conteúdo	Quando estabelecida
Funcional	Especificação de sistema	Aprovação do projeto
Alocada	Especificação de requerimentos de software	Revisão de requerimentos de software
Configuração de desenvolvimento	Documentos de design de alto-nível	Revisão de design preliminar
	Documentos de design detalhado	Revisão de design detalhado
	Código fonte e executável	Teste de unidade
Produto	Documentos de design de software	Auditoria de configuração física do produto de software
	Código fonte e executável	

SUP.2.BP3 Atividade Selecionar itens a serem gerenciados

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	100) Configuração de produto

- a. Identificar e definir categorias de software a ser gerenciado.

A Tabela 5.4 apresenta um exemplo de categorias de software típicas (adaptado de [Buckley, 1996]).

- b. Identificar e definir o processo de produção de software em cada categoria, se diferente daquele constante do documento Plano de Projeto.
- c. Identificar os tipo de software que serão gerenciados em cada categoria.

A seguir são apresentados alguns exemplos [Mikkelsen, 1997]:

- arquivos fonte;
- arquivos do tipo "header";
- arquivos de dados;
- scripts de construção (p. ex.: make);
- arquivos de documentação (p. ex.: MS-Word);

- arquivos de ajuda;
- ícones;
- "bitmaps";
- outros tipos de arquivos de gráficos;
- arquivos de configuração;
- arquivos de defeitos;
- arquivos de sistema (p. ex.: CONFIG.SYS, AUTOEXEC.BAT);
- dados de cliente;
- planilhas.

d. Registrar as informações levantadas no produto de trabalho 100) *Configuração de produto*.

Categoria	Título	Definição
I	Software Produto	Software desenvolvido como um produto final ou uma parte de um produto final.
II	Software fornecido por vendedor	Software que já existe como um produto e é fornecido e mantido por um vendedor.
III	Software de Teste	Software que é desenvolvido para suportar o teste de aceitação formal das categorias de software I e II.
IV	Software de suporte a produto	Software que é desenvolvido para suportar o processo formal de produção de software mas que não está formalmente identificado como um parte do produto final.
V	Documentação	Arquivos fontes dos documentos . Não inclui softwares que dão apoio à geração de documentos nem processadores de texto; estes estão cobertos pela categoria II.
VI	Outros	Todos os outros softwares não definidos que incluam software para suportar testes informais.

SUP2.BP3 Atividade Determinar tipos de documentação

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	91) Plano de gerência de configuração

- a. Determinar tipos de documentação de configuração de software para item de configuração, baseado na EDT (veja subitem *MAN.2.BP5 Atividade Consolidar EDT*) e registrá-los no produto de trabalho 91) *Plano de gerência de configuração*.

A Figura 5.6 mostra um exemplo de documentação de configuração e software (adaptado de [Buckley, 1996]). A linha básica funcional consiste de um documento, a *Especificação de sistema*. A linha básica alocada consiste de dois tipos de documentos (*Especificação de requerimentos de software* e *Especificação de requerimentos de interface*) que declaram requerimentos para os itens de configuração de software. A linha básica de configuração de desenvolvimento consiste dos documentos de design associados e do software. A linha básica de produto consiste do documento *Especificações de Produto* (normalmente a documentação de design) e do software. O software, normalmente, é parte tanto da configuração de desenvolvimento como da linha básica de produto.

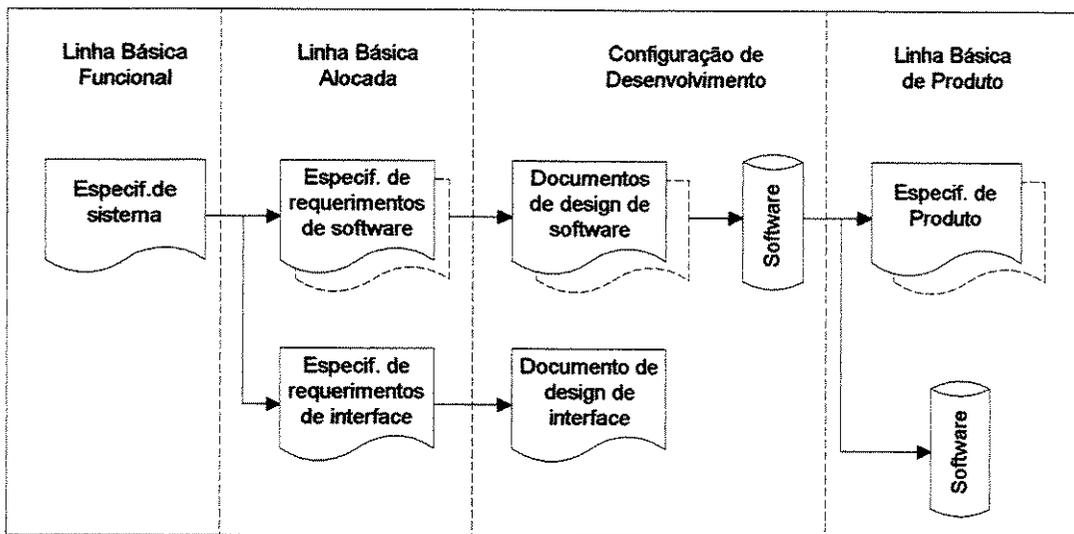


Figura 5.6 Exemplo de conjunto de documentação de configuração e software

SUP2.BP3 Atividade Estabelecer identificadores de itens

- Utilizar o mesmo esquema de identificação estabelecido na EDT (veja subitem *MAN.2.BP5 Atividade Desenvolver EDP*).

SUP2.BP3 Atividade Estabelecer sistema de liberação de itens

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
91) Plano de gerência de configuração	91) Plano de gerência de configuração

- Definir os níveis nos quais cada categoria de software será liberada, baseando-se nas categorias definidas no subitem *SUP.2.BP3 Atividade Selecionar itens a serem gerenciados*.

- b. Declarar os critérios a serem aplicados a cada item a ser aceito dentro de um nível de liberação específico.
- c. Designar as pessoas autorizadas a liberar o item dentro dos níveis especificados.
- d. Designar pessoas autorizadas a liberar mudanças para item que é liberado para um nível específico.

A Tabela 5.5 mostra um exemplo de um conjunto de níveis de liberação e autoridades associadas para a categoria Software Produto [Beckley, 1996].

Tabela 5.5 Exemplo de níveis de liberação e autoridades associadas para categoria Software Produto				
Nível de liberação prévio	Autoridade de liberação	Critério de liberação	Novo nível de liberação	Autoridade de Controle de Mudança de Liberação
Nenhum	Indivíduo responsável	Nenhum	Teste de unidade	Gerente de unidade
Teste de unidade	Gerente de unidade	Término com sucesso de teste de unidade	Teste de integração	Gerente de integração
Teste de integração	Gerente de integração	Término com sucesso de teste de integração	Teste de aceitação de software	Gerente do projeto
Teste de aceitação de software	Gerente de projeto	Término com sucesso de teste de aceitação de software	Teste de aceitação de sistema	Cliente

- e. Estabelecer as condições sob as quais os procedimentos estabelecidos possam ser evitados, se existirem.
- f. Registrar no produto de trabalho 91) *Plano de gerência de configuração* o sistema de liberação de itens.

SUP.2.BP5 Gerenciar mudanças

Registrar e relatar status de itens de configuração e pedidos de modificação. Mudanças para quaisquer itens de configuração seriam revisadas e autorizadas.

SUP2.BP5 Atividade Reconhecer necessidade de mudança

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
84) Relatório de problema	84) Relatório de problema 95) Registro de controle de mudança

- a. Registrar o problema em formulário apropriado do Sistema de Relato de Problema estabelecido, que represente o produto de trabalho *84) Relatório de problema*.
- b. Revisar as informações fornecidas sobre a ocorrência do problema. Um ou mais gerentes devem participar da revisão. Uma revisão inicial é feita para garantir que a declaração do problema está clara, que todos os elementos essenciais de informação que estavam disponíveis foram registrados e que o problema, como declarado é uma falha válida.
- c. Fixar uma prioridade de ação e alocar recursos para determinar a causa da falha. Registrar as decisões tomadas em formulário apropriado do Sistema de Relato de Problema estabelecido.
- d. Registrar a causa da falha. A atividade de determinar a causa da falha está fora do escopo do processo de Gerência de Configuração, porém ela deve fornecer as informações obtidas, através de formulário apropriado do Sistema de Relato de Problema estabelecido.
- e. Determinar ação corretiva e registrá-la em formulário apropriado do Sistema de Relato de Problema estabelecido. As informações determinadas nesta tarefa serão usadas para iniciar a próxima atividade.

SUP2.BP5 Atividade Propor mudança

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
84) Relatório de problema	94) Pedido de mudança (inclui o pacote de mudança)

- a. Ao receber o produto de trabalho *84) Relatório de problema*, iniciar um ciclo de revisão informal e aprovação. Se não aprovado, o originador (ou solicitante) da mudança é notificado e o processo termina.
- b. Propor uma solução para o problema.

- c. Preparar o *Pacote de Mudança*¹. A Tabela 5.6 mostra um exemplo da composição de um pacote de mudança para item de configuração que já era uma linha básica e foi adaptada de Buckley [Buckley, 1996].
- d. Elaborar o produto de trabalho 94) *Pedido de mudança* e anexar nele o *Pacote de mudança*.

Componentes do Pacote de mudança	Item a ser mudado	
	Documentação de configuração	Software
Proposta de mudança de engenharia	Sim	Sim
Aviso de mudança de especificação	Sim	Sim
Páginas de mudança	Sim	---
Arquivos fonte antigos	---	Sim
Arquivos fonte revisados	---	Sim

SUP2.BP5 Atividade Avaliar mudanças propostas

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
94) Pedido de mudança	não há

- a. Avaliar o produto de trabalho 94) *Pedido de mudança* proposto, através da avaliação do pedido contra critérios estabelecidos. A Tabela 5.7 apresenta alguns exemplos de critérios expressos sob a forma de questões, extraídos de Buckley [Buckley, 1996].

Número	Questão
1	O Pedido de mudança é claro? Por exemplo, a solução pode ser implementada de maneira objetiva por uma pessoa que não tenha um conhecimento detalhado do sistema?
2	A solução é consistente? Por exemplo, esta solução conflita com outros itens já aprovados ou em processo de aprovação?
3	O pacote provê um grau razoável de garantia de que a solução, se implementada, resolverá o assunto?
4	A solução é completa? Por exemplo, a solução inclui todas as mudanças que serão requeridas para resolver o assunto?
5	O pedido de mudança identifica os custos e prazos para implementação da solução?

¹ *Pacote de mudança* é um conjunto de documentos que descrevem e justificam mudanças para a documentação de configuração e para o software.

SUP2.BP5 Atividade Coordenar mudanças propostas

- a. Instanciar o processo *SUP.6 Revisão*, para distribuir o produto de trabalho *94) Pedido de Mudança* para revisão e promover reunião de discussão sobre as mudanças com os envolvidos.

SUP2.BP5 Atividade Arquivar proposta de mudança

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
87) Mecanismo de comunicação	não há
94) Pedido de mudança	

- a. Arquivar o *94) Pedido de mudança* em área apropriada da SCML.
- b. Notificar o solicitante da mudança que ela não foi aprovada através do *87) Mecanismo de comunicação* estabelecido.

SUP2.BP5 Atividade Implementar mudança

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
94) Pedido de mudança	93) Item de configuração
	95) Registro de controle de mudança

- a. Colocar a proposta de mudança aprovada numa fila de mudanças, considerando a sua prioridade.
- b. Designar pessoal para fazer a mudança.
- c. Alterar os itens de configuração, de acordo com o *94) Pedido de mudança*, gerando nova versão de *93) Itens de configuração* e registrar as mudanças em *95) Registro de controle de mudança*.
- d. Estabelecer uma linha básica de teste.
- e. Realizar as atividades de teste.
- f. Liberar a mudança para a próxima versão de produto.

SUP.2.BP6 Gerenciar versão de produto

Liberação e entrega de produto de quaisquer itens de configuração seriam revisados e autorizados.

SUP2.BP6 Atividade Construir versão apropriada de produto

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
92) Gerência de configuração	70) Pacote de liberação
93) Item de configuração	96) História de mudança
96) História de mudança	100) Configuração do produto

- a. Construir a versão apropriada do produto, a partir de roteiros de construção oferecidos pela ferramenta de Gerência de Configuração, gerando o produto de trabalho 70) *Pacote de liberação*.
- b. Gerar o produto de trabalho 100) *Configuração do produto*, contendo descrição de versão do produto construído, informações sobre cada item de configuração do produto e sobre as mudanças efetuadas, em relação à versão anterior.

SUP2.BP6 Atividade Auditar itens de configuração de produto

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
93) Item de configuração	não há
94) Pedido de mudança	
95) Registro de controle de mudança	
96) História de mudança	
100) Configuração do produto	

- a. Revisar os itens de configuração de produto, considerando as mudanças que foram feitas ou a solicitação de recuperação de versão específica.

SUP2.BP6 Atividade Distribuir produto

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
70) Pacote de liberação	não há
87) Mecanismo de comunicação	

- a. Colocar o produto de trabalho 70) *Pacote de liberação* a ser distribuído no meio apropriado (magnético ou óptico).
- b. Notificar e/ou entregar a todos os interessados o produto de trabalho 70) *Pacote de liberação*.

SUP.2.BP8 Relatar status de configuração

Regularmente relatar status de cada item de configuração e sua relação na integração de sistema corrente.

SUP2.BP8 Atividade Produzir relatórios

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
Solicitação de relatórios de status de configuração	Relatórios de status de configuração
92) Gerência de configuração (ferramenta)	
98) Sistema de acompanhamento	

- a. Recuperar do sistema de gerência de configuração (92) *Gerência de configuração* e 98) *Sistema de acompanhamento*) as informações necessárias para gerar os relatórios de status de configuração, através das facilidades do sistema de Gerência de Configuração.

SUP.2.BP9 Atividade Gerenciar itens de configuração

O armazenamento, a manutenção, a liberação e entrega de itens de configuração seriam controlados.

SUP.2.BP9 Atividade Inserir item de configuração na biblioteca

- a. Executar o procedimento definido no subitem SUP.2.BP3 *Estabelecer biblioteca de gerência de configuração de software*.

SUP2.BP9 Atividade Inserir mudança aprovada na biblioteca

- a. Executar o procedimento definido no subitem *SUP.2.BP3 Estabelecer biblioteca de gerência de configuração de software.*

SUP2.BP9 Atividade Copiar item da biblioteca

- a. Executar o procedimento definido no subitem *SUP.2.BP3 Estabelecer biblioteca de gerência de configuração de software.*

SUP2.BP9 Atividade Copiar e bloquear para escrita item da biblioteca

- a. Executar o procedimento definido no subitem *SUP.2.BP3 Estabelecer biblioteca de gerência de configuração de software.*

5.4 SUP.6 Revisão

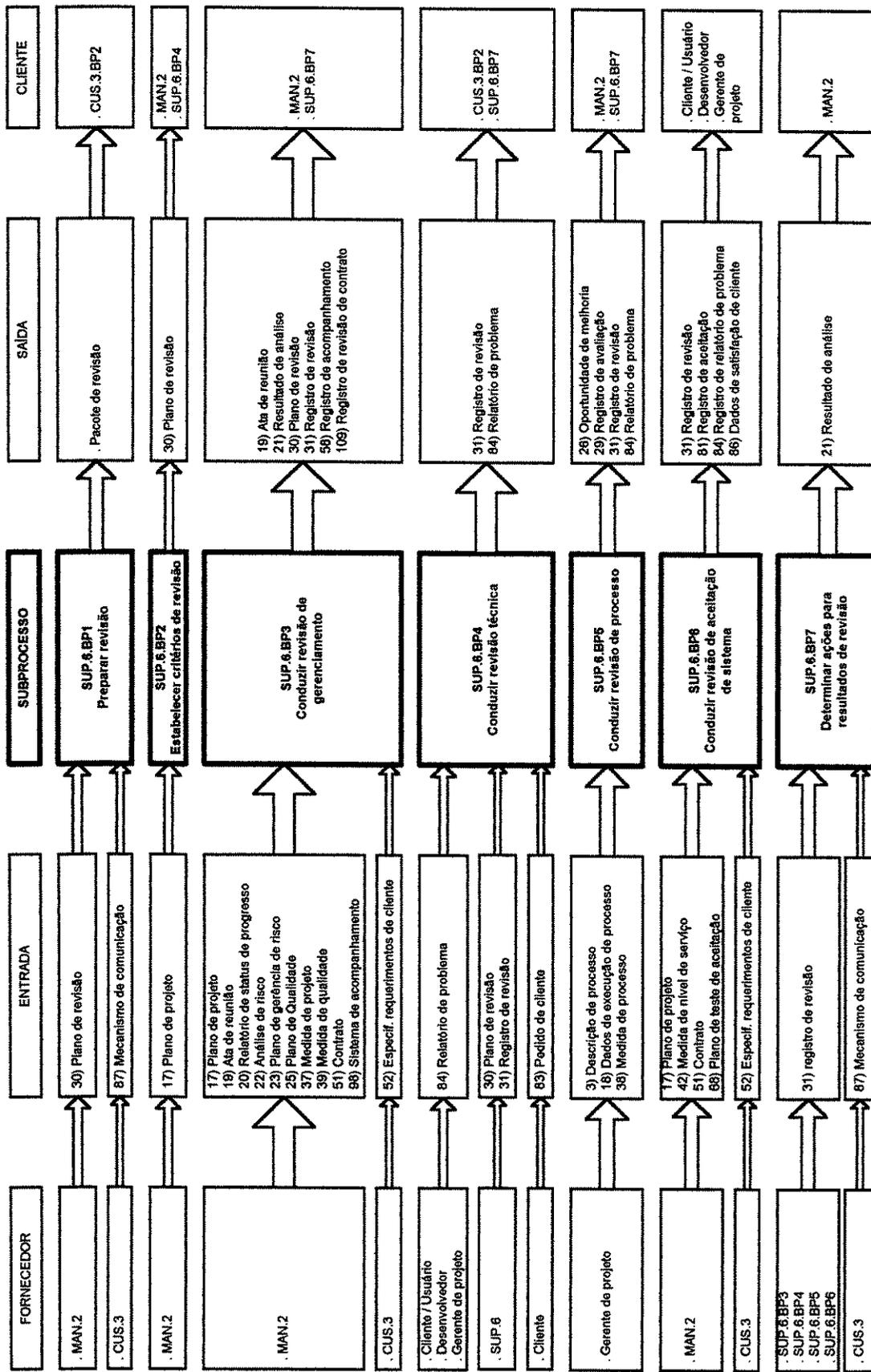
5.4.1 Escopo

Nome do processo:		SUP.6 Processo de revisão	
Objetivo:	O propósito deste processo é manter um entendimento comum do progresso com o cliente, contra os objetivos do contrato e o que seria feito para auxiliar garantir o desenvolvimento de um produto que satisfaz o cliente. Revisões estão no gerenciamento de projeto e nos níveis técnicos e são mantidas durante a vida do projeto ¹ .		
Entradas:	3) Descrição de processo 17) Plano de projeto 18) Dados de execução de processo 19) Atas de reunião 20) Relatório de status de progresso 22) Análise de risco 23) Plano de gerenciamento de riscos 25) Estratégia/plano de qualidade 30) Estratégia/plano de revisão 31) Registro de revisão 37) Medida de projeto 38) Medida de processo 39) Medida de qualidade 42) Medida de nível de serviço 51) Contrato 52) Especificação de requerimento (cliente) 68) Plano de teste de aceitação 83) Pedido de cliente 84) Relatório de problema 87) Mecanismo de comunicação 98) Sistema de acompanhamento	Saídas:	19) Ata de reunião 21) Resultado de análise 26) Oportunidade de melhoria 29) Registro de avaliação/auditoria 30) Plano/estratégia de revisão 31) Registro de revisão 58) Registro / mapeamento de acompanhamento 81) Registro de aceitação 84) Registro de relatório de problema 86) Dados de satisfação de cliente 109) Registro de revisão de contrato
Início do processo:	Preparação de revisão.		
Conteúdo:	Prepara revisão; estabelece critério de revisão; conduz revisão de gerenciamento; conduz revisão técnica; conduz revisão de processo; conduz revisão de aceitação de sistema; determina ações para resultados de revisão e acompanha ações para resultados de revisão.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> • revisões periódicas estarão mantidas em marcos pré-determinados; • o status e os produtos de uma atividade de um processo serão avaliados através de atividades de revisão entre os clientes, fornecedores e outras partes interessadas; • resultados de revisão serão conhecidos de todas as partes afetadas; • itens de ação resultantes de revisões estarão acompanhados para fechamento. 		

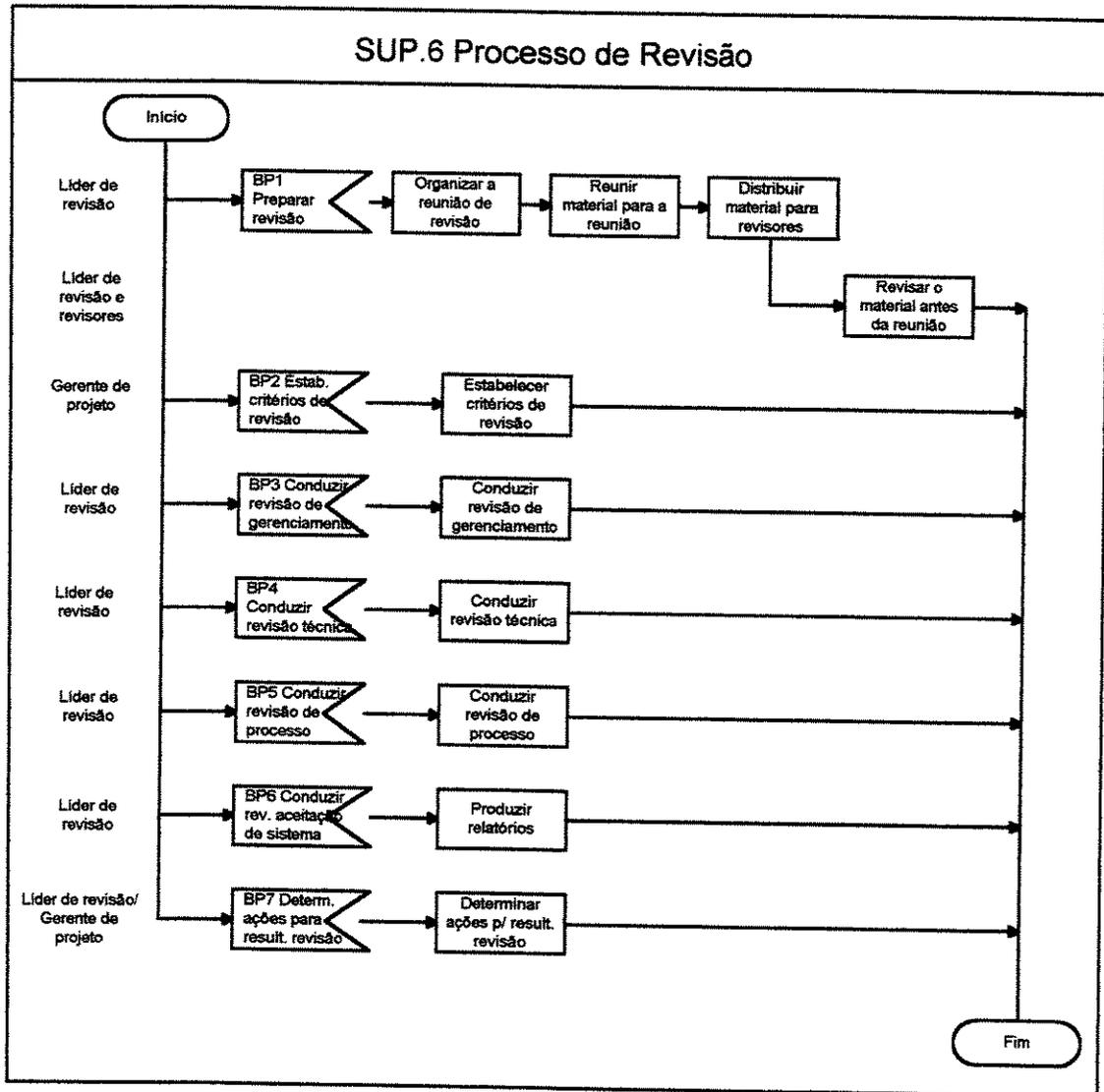
¹ A norma ISO/IEC 12207 contém requerimentos específicos para revisões de gerenciamento de projeto e revisões técnicas.

<i>Clientela a ser atendida:</i>	Cliente/usuário, gerente de projeto, desenvolvedor.
<i>Indicadores de desempenho:</i>	Existência e adequação das práticas base aplicadas neste processo: SUP.6.BP1, SUP.6.BP2, SUP.6.BP3, SUP.6.BP4, SUP.6.BP5, SUP.6.BP6, SUP.6.BP7. Existência dos produtos de trabalho de entrada e saída e suas características.

5.4.2 Macro-diagrama do SUP.6 Processo de Revisão



5.4.3 Fluxograma



5.4.4 Atividades

SUP.6.BP1 Preparar revisão

Para preparar uma revisão interna (inter-processos) ou externa (desenvolvedor / cliente), os seguintes itens seriam preparados:

- escopo da revisão;
- tópicos da revisão;
- participantes;
- lista de distribuição de partes afetadas;
- responsabilidades dos participantes;
- saídas desejadas;
- um cronograma;
- requerimentos de recurso e facilidade.

SUP.6.BP1 Atividade Organizar a reunião de revisão

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
30) Plano de revisão	não há

- Considerar as seguintes restrições para reunião de revisão [Pressman, 1995]:
 - tipicamente, envolver entre três e cinco pessoas na revisão, sendo uma delas necessariamente o autor dos produtos submetidos à revisão;
 - uma preparação antecipada para a reunião deve acontecer, porém ela não deve exigir mais que duas horas de trabalho para cada pessoa;
 - a duração da reunião de revisão deve ser inferior a duas horas.
- Avaliar o produto quanto aos seus atributos. Basear-se na relação de produtos e seus atributos estabelecidos no Plano de projeto.
- Se produto não aprovado, avisar o líder de projeto e terminar o processo.
- Definir o escopo da revisão, considerando as restrições acima. O foco deve estar em um produto ou um componente de software - como por exemplo uma parte de uma especificação de requisitos, um design detalhado de um módulo, uma relação de códigos fonte para um módulo.

- e. Definir os tópicos da revisão.
- f. Identificar os participantes de acordo com a perícia estabelecida no 30) *Plano de revisão*.
- g. Elaborar uma lista de distribuição de partes afetadas.
- h. Definir as responsabilidades dos participantes, segundo cada papel estabelecido no produto de trabalho 30) *Plano de revisão*.
- i. Especificar as saídas desejadas.
- j. Estabelecer um cronograma de atividades.
- k. Estabelecer requerimentos de recursos e facilidades, por exemplo, sala para quatro pessoas com um microcomputador ligado à rede local, data-show, quadro branco ou flip-chart, pincéis adequados, café, água, etc.

SUP.6.BP1 Atividade Preparar material para a reunião

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
não há	Pacote de revisão

- a. Preparar o pacote de revisão com:
 - a agenda da reunião (por exemplo: data, horário, local, participantes e suas responsabilidades e pauta);
 - critérios de revisão (por exemplo: checklists para identificação de problema, resolução de problema, acordo de problema, etc.);
 - material a ser revisado (pode ser uma cópia impressa de um documento, ou a indicação de onde ele está disponível em meio magnético - biblioteca de gerência de configuração, área temporária para revisão, etc.).

SUP.6.BP1 Atividade Distribuir material para revisores

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
Pacote de revisão 87) Mecanismo de comunicação	não há

- a. Distribuir o pacote de revisão utilizando o 87) *Mecanismo de comunicação* estabelecido.

SUP.6.BP1 Atividade Revisar o material antes da reunião

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
Pacote de revisão	não há

- a. Revisar o produto, considerando o escopo, os tópicos e os critérios de revisão estabelecidos.

Espera-se que o revisor gaste entre uma ou duas horas para esta tarefa. O revisor deve estar atento para os seguintes pontos: o que está sendo revisado é o produto e não o seu autor; e, se a revisão for de identificação de problemas, não tente propor soluções neste momento [Pacheco et al., 1997].

SUP.6.BP2 Estabelecer critérios de revisão

Estabelecer critérios para uma revisão, tais como para identificação, resolução e acordo de problema.

SUP.6.BP2 Atividade Estabelecer critérios de revisão

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	30) Plano de revisão

- a. Estabelecer e registrar no 30) *Plano de Revisão* os critérios de revisão.

Os critérios de revisão podem ser *checklists*, requerimentos ou padrões (de documentação, de identificação de componentes, de formato de documento, etc.) adotados no projeto. A seguir, para revisões técnicas de identificação de problema, são sugeridos alguns critérios adaptados de [Pressman, 1995] e de [Pacheco et al., 1997].

Planejamento Preliminar - Revisão interna: Produtos 17) *Plano de projeto (preliminar)*, 52) *Especificação de requerimentos (cliente)* e 82) *Procedimento de suporte a cliente*

1. O propósito do sistema contribui para a missão da empresa?
2. Os requerimentos estabelecidos para o sistema foram apropriadamente identificados e documentados?
3. Os requerimentos estão definidos de forma precisa e não ambígua?
4. Os requerimentos podem ser verificados?
5. O escopo do projeto foi definido apropriadamente?

6. Existe um conhecimento adequado das necessidades dos usuários, por parte dos desenvolvedores?
7. O sistema é técnica e economicamente viável?
8. As funções, o desempenho e as interfaces foram definidas adequadamente?
9. Os requerimentos funcionais estabelecidos para o sistema estão consistentes com os requerimentos definidos pelo usuário/cliente?
10. Os benefícios esperados justificam os custos de desenvolvimento do sistema?
11. A análise do ambiente real do usuário e dos riscos inerentes ao desenvolvimento justificam o esforço para o desenvolvimento do sistema?
12. Os requerimentos são consistentes com o cronograma e recursos disponíveis?
Considere o sistema ao longo do ciclo de vida, envolvendo os aspectos de produção, logística, software e testes, sempre tendo em vista o atendimento aos requerimentos do sistema.
13. Foi estabelecido um mecanismo de verificação e validação do sistema?
14. O sistema de documentação atende aos requerimentos do cliente? Por exemplo: manual de instalação, manual do usuário, etc.
15. O procedimento de suporte de cliente é viável técnica e economicamente para a empresa?

Planejamento Preliminar - Revisão externa: Produtos 17) Plano de projeto (preliminar), 52) Especificação de requerimentos (cliente) e 82) Procedimento de suporte a cliente

1. Os requerimentos estabelecidos para o sistema foram apropriadamente identificados e documentados?
2. Os requerimentos podem ser verificados?
3. O escopo do projeto foi definido apropriadamente?
4. Existe um conhecimento adequado das necessidades dos usuários, por parte dos desenvolvedores?
5. O sistema é técnica e economicamente viável?
6. As funções, o desempenho e as interfaces foram definidas adequadamente?
7. Os requerimentos funcionais estabelecidos para o sistema estão consistentes com os requerimentos definidos pelo usuário/cliente?
8. Os benefícios esperados justificam os custos de desenvolvimento do sistema?
9. A análise do ambiente real do usuário e dos riscos inerentes ao desenvolvimento justificam o esforço para o desenvolvimento do sistema?
10. Existe um plano de gerenciamento dos riscos?
11. Os requerimentos são consistentes com o cronograma e recursos disponíveis?
Considere o sistema ao longo do ciclo de vida, envolvendo os aspectos de produção,

logística, software e testes, sempre tendo em vista o atendimento dos requerimentos do sistema.

12. Foi estabelecido um mecanismo de verificação e validação do sistema?
13. O sistema de documentação atende aos requerimentos do cliente? Por exemplo: manual de instalação, manual do usuário, etc.
14. Atividades de garantia de qualidade estão previstas e orçadas?
15. O procedimento de suporte de cliente atende às necessidades dos usuários?

Planejamento Detalhado

1. Os objetivos do software estão definidos de forma clara?
2. A terminologia é clara?
3. Os recursos são adequados aos objetivos?
4. Existem, entre os recursos necessários, alguns indisponíveis?
5. Os riscos mais importantes foram identificados?
6. Existe um plano alternativo para as situações de risco?
7. O cronograma das atividades é consistente e realista?
8. A equipe de desenvolvimento está definida de forma adequada?
9. Todas as normas e padrões para desenvolvimento foram definidos?

Plano de Teste

1. O plano de teste é consistente com o resto do projeto?
2. O cronograma de teste está claramente definido?
3. As principais fases de teste estão claramente definidas?
4. A documentação está de acordo com os padrões e normas de desenvolvimento definidas?

Plano de Gerência de configuração

1. Os itens de configuração foram definidos de forma adequada?
2. O plano define claramente o procedimento para acompanhamento das alterações durante todo o ciclo de vida?
3. Foram definidos controles para garantir a atualização da documentação?
4. Foi definida uma ferramenta de controle?
5. Foram definidas as responsabilidades para aprovação de mudança?

Plano de Revisão

1. Todos os pontos de revisão, ao longo do progresso, foram claramente definidos?
2. O grupo revisor foi qualificado e quantificado de forma a atender às exigências do projeto?
3. As regras e critérios para orientar as reuniões de revisão são explícitas e claras?
4. Os procedimentos pós-revisão estão claramente definidos?
5. Existe um formulário padrão para confecção dos relatórios de revisão?

SUP.6.BP3 Conduzir revisão de gerenciamento

Conduzir periodicamente revisões de gerenciamento para interpretar e avaliar:

- proposta contra requerimentos;
- obtenção contra plano de projeto e cronograma;
- riscos;
- preparação para transferir para o próximo processo.

SUP.6.BP3 Atividade Conduzir revisão de gerenciamento

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	19) Ata de reunião
19) Atas de reunião	21) Resultado de análise
20) Relatório de status de progresso	30) Plano de revisão
22) Análise de risco	31) Registro de revisão
23) Plano de gerência de risco	58) Registro de acompanhamento
25) Plano de Qualidade	109) Registro de revisão de contrato
37) Medidas de projeto	
39) Medidas de qualidade	
51) Contrato	
52) Especificação de requerimentos (cliente)	
98) Sistema de acompanhamento	

- a. Executar os procedimentos de condução de revisão estabelecidos no produto de trabalho 30) *Plano de revisão* para os produtos de entrada.

SUP.6.BP4 Conduzir revisão técnica

Conduzir periodicamente revisões técnicas para interpretar e avaliar características técnicas e status contra requerimentos de cliente e critérios de aceitação documentados em contrato.

SUP.6.BP4 Atividade Conduzir revisão técnica

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
30) Plano de revisão	31) Registro de revisão
31) Registro de revisão	84) Registro de relatório de problema
83) Pedido de cliente	
84) Relatório de problema	

- a. Executar os procedimentos de condução de revisão estabelecidos no produto de trabalho 30) *Plano de revisão* para os produtos de entrada.

SUP.6.BP5 Conduzir revisão de processo

Conduzir periodicamente revisões de processo para interpretar e avaliar viabilidade e capacidade dos processos correntes para um projeto.

SUP.6.BP5 Atividade Conduzir revisão de processo

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
3) Descrição de processo	26) Oportunidade de melhoria
18) Dados de processo	29) Registro de avaliação / auditoria
38) Medidas de processo	31) Registro de revisão
	84) Registro de relatório de problema

- a. Executar os procedimentos de condução de revisão estabelecidos no produto de trabalho 30) *Plano de revisão* para os produtos de entrada.

SUP.6.BP6 Conduzir revisão de aceitação de sistema

Conduzir periodicamente revisões de aceitação de sistema demonstrar para o cliente que a completude e a corretitude do sistema final em configuração e funcionalidade cumpre com

padrões e especificações apropriados, e satisfazos critérios de aceitação documentados no contrato.

SUP.6.BP6 Atividade Conduzir revisão de aceitação de sistema

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
17) Plano de projeto	31) Registro de revisão
42) Medidas de nível de serviço	81) Registro de aceitação
51) Contrato	84) Registro de relatório de problema
52) Especificação de requerimentos (cliente)	86) Dados de satisfação de cliente
68) Plano de teste de aceitação	

- a. Executar os procedimentos de condução de revisão estabelecidos no produto de trabalho 30) *Plano de revisão* para os produtos de entrada.

SUP.6.BP7 Determinar ações para resultados de revisão

Analisar relatório de revisão; distribuir relatório de revisão; propor solução(ões) para resultados de revisão; determinar prioridade para ações.

SUP.6.BP7 Atividade Determinar ações para resultados de revisão

A tabela a seguir relaciona os produtos de trabalho de entrada e saída da atividade e, em seguida, estão relacionadas e descritas as tarefas contidas nesta atividade.

Entrada:	Saída:
31) Registro de revisão	21) Resultado de análise
87) Mecanismo de comunicação	

- a. Analisar os resultados da revisão registrados no produto de trabalho 31) *Registro de revisão*, gerando o produto de trabalho 21) *Resultado de análise*, considerando os seguintes itens:
- o que foi analisado;
 - quem fez a análise;
 - os critérios de análise usados (critérios de seleção ou esquema de priorização; critérios de decisão e critérios de qualidade);
 - registro de resultados (o quê foi decidido/selecionado; razão para seleção; suposições feitas; riscos potenciais).

- b. Distribuir o produto de trabalho 31) *Registro de revisão* utilizando o 87) *Mecanismo de comunicação* estabelecido.
- c. Propor soluções para os resultados de revisão baseando-se nos produtos de entrada.
- d. Determinar prioridade para execução das ações que solucionam os problemas relatados no registro de revisão.

6 Modelo de Gerência de Projeto de Software

O modelo de gerência de projeto de software é o resultado do detalhamento das fases do Ciclo de Vida Genérico de Projeto para projeto de software (veja item 3.1 *Gerência de projeto*) e a Figura 6.1 mostra este modelo. Os retângulos representam macro-atividades das fases do ciclo de vida.

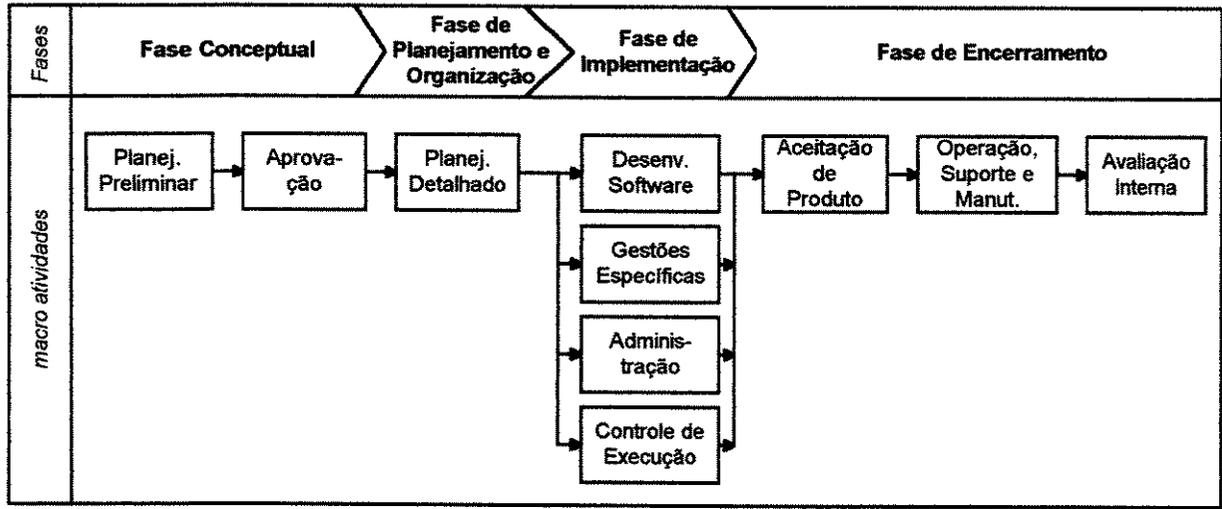


Figura 6.1 Modelo de gerência de projeto de software: fases e macro-atividades

A macro-atividade *Planejamento Preliminar* tem por objetivo elaborar e submeter para aprovação uma proposta de execução de projeto de através: de identificação de necessidades e objetivo do projeto; estabelecimento dos limites do projeto (abrangência) e sua localização no ambiente; identificação de oportunidades e riscos; elaboração da estrutura de decomposição do trabalho - EDT, cronograma e orçamento; e delineamento do controle de execução.

A macro-atividade *Aprovação* consiste de um ato formal de aprovação do planejamento preliminar, equivalente a um contrato entre as partes que deve ser cumprido. Este "contrato" pode ser modificado por motivos relevantes, mediante acordo entre as mesmas entidades compromissadas. Se aprovado, o planejamento preliminar incorpora as mudanças negociadas e passa a ser a base para as próximas macro-atividades.

A macro-atividade *Planejamento Detalhado* tem por objetivo elaborar o planejamento detalhado e estabelecer a organização do projeto de software, a partir do planejamento preliminar aprovado. Portanto, trata-se de expandir o planejamento e organizar o projeto, dando-lhe uma estrutura e fixando o funcionamento, pela determinação de atribuições, de responsabilidades, das descentralizações, das condições de controle, quanto a prazos, custos e desempenho.

A macro-atividade *Desenvolvimento de Software* recebe as especificações gerais do projeto de software da etapa anterior - Planejamento Detalhado - e traduz estas informações para uma especificação de requerimentos de software. Baseado nesta especificação, desenvolve uma

arquitetura de software que atende aos requerimentos de software e constrói os programas fontes que a implementam. Nesta etapa também são testados os programas individualmente e a sua integração até a obtenção do produto de software completo.

A macro-atividade *Gestões Específicas* compreende as gestões técnicas de suporte ao desenvolvimento do produto como gestão da configuração, da qualidade, de interfaces, de riscos, de requerimentos, de documentação, etc., conforme estabelecido pelo gerente na EDT.

A macro-atividade *Administração* compreende atividades administrativas que fazem a ligação com diversos órgãos para tratar do fluxo de recursos materiais e financeiros, de pessoal, de contratos, relatórios e prestações de conta etc., conforme estabelecido pelo gerente na EDT.

A macro-atividade *Controle de Execução* é executada em paralelo com a execução do projeto de software e abrange de forma matricial, todas as tarefas do projeto: as técnicas, as gerenciais e as administrativas. Nesta macro-atividade, o objetivo é cumprir o que foi detalhado nos planos detalhados de controle. Compreende o ajuste e difusão dos planos da qualidade, a execução dos controles de custos, prazos e execução física e a execução de revisões (técnicas, gerenciais e específicas).

A macro-atividade *Aceitação de Produto* compreende a verificação da quitação dos compromissos assumidos no início do projeto, ao ser aprovado o planejamento preliminar e as condições nele embutidas. A aceitação do produto pelo cliente deve pautar-se pelas previsões de testes e revisões constantes das especificações do produto (plano de qualidade, planos detalhados de testes e revisões, etc.), consistindo, em geral, de testes técnicos e operacionais.

A macro-atividade *Operação, Suporte e Manutenção* consiste da assistência ao usuário através de serviços de apoio do produto, como treinamento de usuário, suporte técnico, manutenção preventiva, corretiva e adaptativa.

A etapa de *Avaliação Interna* consiste de uma auto-análise da execução do projeto como um todo. Procede-se a uma última revisão para colher o máximo de ensinamentos que advêm dos acertos e dos erros porventura cometidos. Com isso, a equipe procede em benefício de sua evolução e no sentido de prevenir insucessos futuros e, ao mesmo tempo, recomendar procedimentos que se mostraram valiosos. Pode-se contar com pessoas alheias ao projeto para auxiliar na avaliação e exploração de erros e acertos em benefício de projetos futuros e dos procedimentos gerenciais, técnicos e administrativos.

Seqüência de utilização dos processos de software pelas macro-atividades

O modelo de referência para processos e capacidade de processo da ISO/IEC TR 15504 especifica os processos de cada categoria de forma estática, deixando para a organização que for utilizá-los estabelecer a seqüência de utilização apropriada à sua estrutura.

Supondo-se que uma organização adote o Modelo de Gerência de Projeto de Software, como definido no início deste capítulo, este trabalho contribui com a especificação de uma seqüência de utilização dos quatro processos de software descritos no capítulo 5 *Descrição dos processos*. Esta seqüência é um detalhamento das macro-atividades do Modelo de Gerência de Projeto de Software e mostra de uma forma explícita a ligação do modelo com o sistema de Gerenciamento da Qualidade Total. Esta ligação é representada pela utilização dos processos de software descritos por um método de Gestão de Processo. A implementação deste modelo incorpora qualidade ao software.

A Figura 6.2 mostra a seqüência de utilização das instâncias dos processos para as macro-atividades *Planejamento Preliminar*, *Aprovação* e *Planejamento Detalhado*. As demais macro-atividades não estão representadas, pois não pertencem ao escopo deste trabalho.

A parte *Macro-atividades* da Figura 6.2 relaciona as macro-atividades do escopo deste trabalho. A parte *Instâncias* estabelece a seqüência de ativação e uso das instâncias dos processos de desenvolvimento de software. Os retângulos representam a instanciação de um processo para aplicar uma prática básica ou um conjunto de práticas básicas. A identificação do processo e das práticas seguem o formato:

XXX.n (YYY) onde:

- **XXX** é a categoria do processo (CUS - cliente/fornecedor; MAN - gerenciamento; SUP - suporte, ENG - engenharia);
- **n** é a identificação do processo dentro da categoria; neste caso, CUS.3 Elicitação de requerimentos, MAN.2 Gerência de projeto, SUP.6 Revisão e SUP.2 Gerência de configuração;
- **YYY** é a identificação de uma prática básica ou de um conjunto de práticas válidas para o processo XXX.n; o nome das práticas pode ser visto nos macrodiagramas dos respectivos processos.

Uma instância pode ser expandida para apresentar as principais atividades da prática básica que está sendo aplicada. Na Figura 6.2, a instância *CUS.3 (BP2)* é expandida e mostra as atividades *elaborar acordo*, *revisar acordo* e *aprovar acordo*.

O símbolo "... " indica que outras instâncias poderão ser ativadas, quando outros processos forem descritos.

Interpretando-se a Figura 6.2, tem-se a seqüência de atividades de gerência de projeto de software relacionada na Tabela 6.1.

Tabela 6.1 - Descrição da seqüência de instâncias de processos, por macro-atividade

<i>Planejamento Preliminar</i>	<ol style="list-style-type: none"> 1. CUS.3.BP1: início da gerência de projeto pela obtenção de requerimentos de cliente. 2. CUS.3.BP2 - elaborar acordo: obtidos os requerimentos, elaborar acordo sobre requerimentos, que basicamente é o plano preliminar de projeto. <ol style="list-style-type: none"> 2.1 MAN.2.BP10: elaborar plano de projeto, plano de revisão, e plano de gerência de configuração. <ol style="list-style-type: none"> 2.1.1 SUP6.BP2: definir de critérios de revisão. 2.1.2 SUP.2.BP1 e BP2: desenvolver estratégia de gerência de configuração e estabelecer um sistema de gerência de configuração, respectivamente. 3. CUS.3.BP2 - revisar acordo: revisar o acordo internamente (considerando os recursos e aspectos técnicos e econômicos). <ol style="list-style-type: none"> 3.1 SUP.6.BP1, BP4 e BP7: preparar revisão, conduzir revisão e determinar ações para resultados de revisão, respectivamente.
<i>Aprovação</i>	<ol style="list-style-type: none"> 4. CUS.3.BP2 - aprovar acordo: apresentar o acordo ao cliente para que alterações sejam negociadas e aprovadas. <ol style="list-style-type: none"> 4.1 SUP.6.BP1, BP4 e BP7: preparar revisão com participação do cliente, conduzir revisão e determinar ações para resultados de revisão, respectivamente. 5. CUS.3.BP3: aprovado o acordo, estabelecer linha básica de requerimentos. <ol style="list-style-type: none"> 5.1 SUP.2.BP9: gerenciar de itens de configuração
<i>Planejamento Detalhado</i>	<ol style="list-style-type: none"> 6. CUS.3.BP4: gerenciar as mudanças de requerimentos. <ol style="list-style-type: none"> 6.1 SUP.2.BP6: gerenciar mudanças, que precisam ser aprovadas através de uma revisão técnica formal. <ol style="list-style-type: none"> 6.1.1 SUP.6. BP1, BP4 e BP7: preparar revisão técnica, conduzir revisão e determinar ações para resultados de revisão, respectivamente.

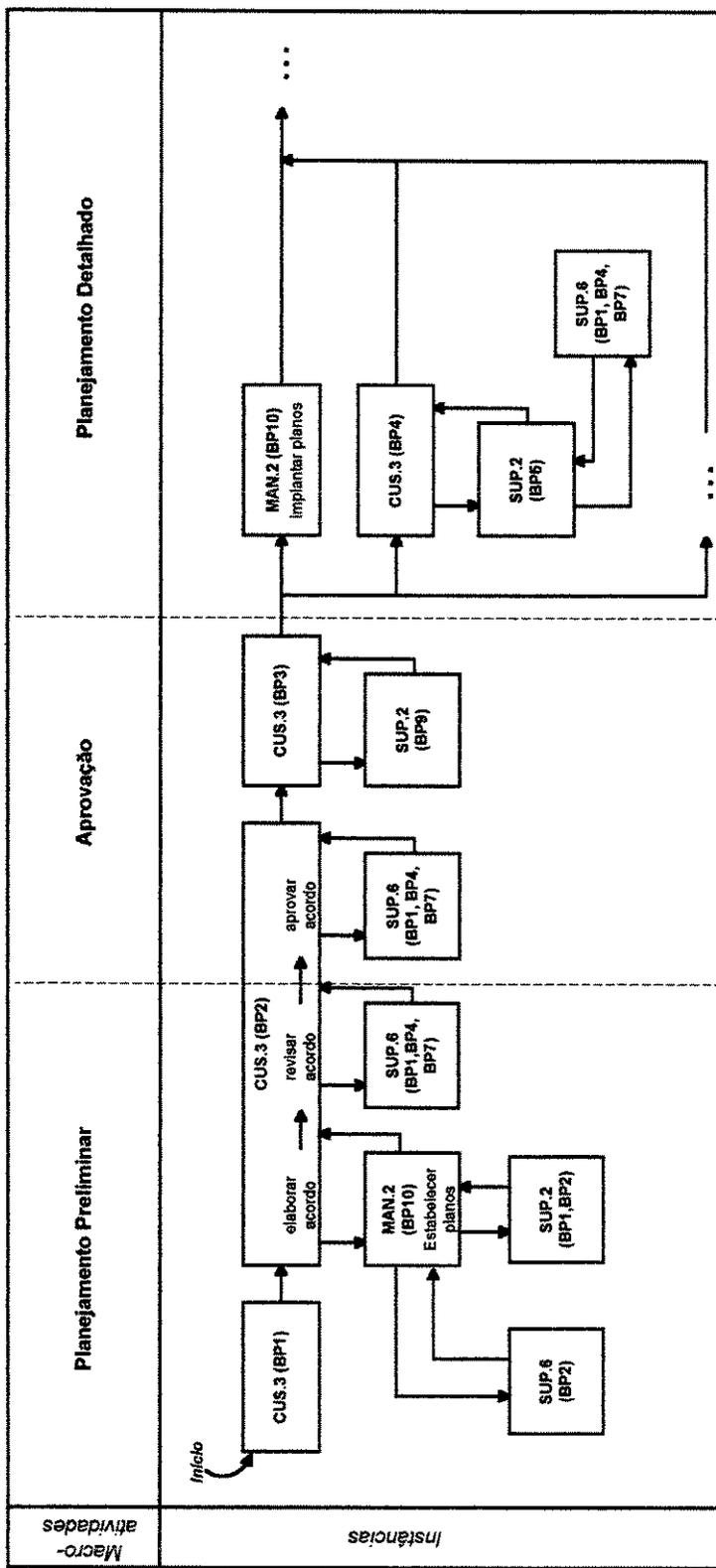


Figura 6.2 Sequência de utilização dos processos de software pelas macro-atividades de Gerência de Projeto de Software

7 Aplicação do modelo de gerência de projeto de software na Embrapa

7.1 Apresentação da empresa

O agronegócio brasileiro é o conjunto dos setores envolvidos com a atividade agropecuária e que inclui desde o desenvolvimento e a produção de insumos - sementes, adubos, máquinas e implementos - até o processamento industrial, armazenamento, transporte e colocação do produto nas prateleiras do mercado.

O agronegócio brasileiro encontra-se em rápido processo de transformação e é crescente a integração da produção agrícola com o setor de insumos e com a fase de processamento dos produtos primários. Essa integração no agronegócio exige da produção agrícola ganhos em produtividade e qualidade, devido à concorrência internacional tanto no mercado interno como no externo [Embrapa, 1998a].

Neste contexto, a Empresa Brasileira de Pesquisa Agropecuária - Embrapa vem valorizando soluções que permitam ao agronegócio brasileiro competir em condições de igualdade, e até mesmo de superioridade, sem prejuízos ou agressões ao meio ambiente. Sua missão é viabilizar soluções para o desenvolvimento sustentável do agronegócio brasileiro por meio de geração, adaptação e transferência de conhecimentos e tecnologias em benefício da sociedade [Embrapa, 1998c].

Desde o início da década de 90, a Embrapa vem buscando uma administração moderna, que permita à empresa ser líder e inovadora em seu segmento, ter uma boa capacidade de resposta e assim, ser competitiva. O primeiro passo foi adotar o Gerenciamento pelas Diretrizes e conseqüentemente implantar o processo de administração estratégica [Certo & Peter, 1993] tendo como primeiro resultado o Plano Diretor da Embrapa e o Plano Diretor de cada uma das unidades descentralizadas, distribuídas geograficamente pelas diversas regiões do Brasil.

A introdução dos princípios de gerenciamento da qualidade total, levou a empresa a desenvolver o Sistema Embrapa de Planejamento - SEP, fundamentado no Ciclo PDCA (Plan, Do, Check, Act) [Campos, 1992] como modelo de planejamento, envolvendo quatro fases: planejar, executar, controlar e agir corretivamente [Goedart et al., 1994]. O SEP é composto de figuras programáticas e de mecanismos de articulação.

Figuras programáticas

As figuras programáticas têm o objetivo de ordenar as ações executadas de forma relativamente autônoma por uma unidade ou instituição de pesquisa e de organizar as ações que exigem o esforço conjugado e harmônico de várias unidades ou instituições da Embrapa e do Sistema Nacional de Pesquisa Agropecuária - SNPA [Goedert et al., 1994]. São elas:

- *Planos Diretores*: Plano Diretor da Embrapa (PDE), Plano Diretor de Unidade Descentralizada (PDU) e Plano Diretor da Sede (PDS); eles são instrumentos de

planejamento estratégico que definem o rumo da instituição para o cumprimento de sua missão;

- *Planos Anuais de Trabalho (PAT)*: são documentos operacionais que sintetizam a programação anual das unidades da empresa ou instituições do SNPA, em cumprimento de suas respectivas missões e objetivos. O PAT especifica para cada unidade, as necessidades de recursos humanos, materiais e financeiros para a operacionalização dos projetos aprovados e o alcance de seus objetivos.
- *Programa* que tem caráter de longo prazo e agrega ações e atividades executadas por várias ou todas as entidades de uma instituição ou mesmo por várias instituições;
- *Projeto* que tem caráter de curto e médio prazo, reunindo ações que visam atender uma ou mais demandas definidas no programa; frequentemente, o projeto tem caráter sistêmico e interdisciplinar, envolvendo o trabalho em equipes multidisciplinares de uma ou mais instituições, com competência para atingir os objetivos almejados;
- *Subprojeto*: é figura programática auxiliar, através da qual o pesquisador ou a equipe de pesquisadores ordena as atividades a serem desenvolvidas com o objetivo de solucionar problemas específicos e relevantes dentro de cada projeto; o subprojeto é assim uma figura mais de âmbito uniinstitucional ou local, podendo ser elaborado apenas no âmbito interno das unidades ou instituições participantes que compõem determinado projeto.

As relações de dependência entre as figuras programáticas e os níveis gerenciais em que estão alocadas podem ser vistas na Figura 7.1. Os projetos devem atender às demandas de programas e estas devem estar alinhadas com o planejamento estratégico tanto da unidade (PDU) como da empresa (PDE).

A empresa adotou o método de *gestão de processo* para o gerenciamento da rotina do trabalho do dia-a-dia. Considerando-se a empresa como um todo, um projeto de pesquisa é considerado seu processo básico. Por isso, no nível tático, implementa-se a gestão de processos através da gestão dos projetos de pesquisa do SEP. Porém, do ponto de vista de uma unidade da Embrapa, a gerência do trabalho do dia-a-dia realizado em um projeto/subprojeto do SEP está fundamentada nos processos operacionais desta unidade, que alimentam o SEP (nível tático - programa) com as informações geradas/coletadas.

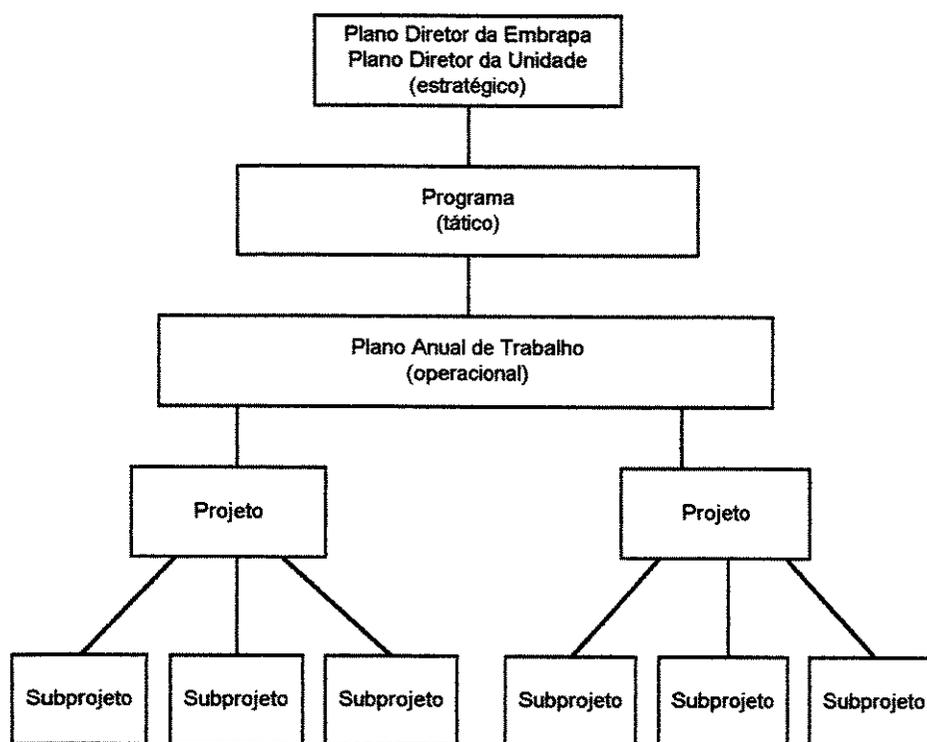


Figura 7.1 Níveis gerenciais da Embrapa e as figuras programáticas do SEP

Mecanismos de comunicação

Os mecanismos de articulação têm por objetivo assegurar o atendimento das demandas da sociedade, a perfeita integração e participação interinstitucional, bem como a qualidade técnica da programação. São eles:

- *Conselhos Assessores Nacionais (CN) e Regionais (CR)*, que são órgãos de caráter consultivo e de assessoramento ao processo de definição das prioridades em atendimento às demandas tecnológicas de mercado, detectadas junto aos usuários, clientes e beneficiários da pesquisa;
- *Comissões Técnicas de Programa (CTP)* que são responsáveis pelo planejamento e montagem inicial dos respectivos programas, com base nas prioridades nacionais e regionais, em articulação com os conselhos; definidos os programas, cada comissão técnica passa a exercer sua função primordial de análise, acompanhamento e avaliação dos projetos integrantes de seu programa; as comissões têm caráter deliberativo e constituem um mecanismo importante na busca da qualidade e eficácia da programação;
- *Comitês Técnicos Internos (CTI)* que provêm o assessoramento técnico à chefia de unidade no que se refere à formulação, ao acompanhamento e à avaliação, no âmbito interno, de seus próprios projetos e subprojetos, antes de encaminhá-los às CTP.

A Embrapa Informática Agropecuária

A Embrapa Informática Agropecuária é uma das unidades descentralizadas (UD) da Embrapa e tem como missão institucional viabilizar soluções tecnológicas e competitivas no âmbito da informática, visando atender às demandas do agronegócio, contribuindo para o desenvolvimento sustentável, em benefício da sociedade [Embrapa..., 1999].

Um dos objetivos estabelecidos no seu Plano Diretor para o período de 1999 a 2003, é consolidar a UD como um centro de pesquisa e de referência nas suas áreas de atuação. Para a área de software, foi criado o *Núcleo de Engenharia de Sistemas* que tem por objetivo o acompanhamento, a adoção e o desenvolvimento de processos de produção de software visando o suporte à pesquisa e à produção agropecuária, o contínuo aperfeiçoamento da unidade nesses processos e a indução de oferta de produtos de software agropecuário de qualidade, baseado nas tecnologias agropecuárias geradas pelo Sistema Nacional de Pesquisa Agropecuária - SNPA.

Atualmente, no nível operacional da unidade, está sendo implementada a gestão de processo para alguns dos processos da área administrativa. Em breve, este método de gestão deverá ser aplicado à área de software.

Um projeto de software, na Embrapa Informática Agropecuária, está vinculado a um projeto/subprojeto de pesquisa do SEP.

Algumas tecnologias são utilizadas para incorporar características de qualidade aos projetos de software, como por exemplo, uso de técnicas de prototipação para especificação de requerimentos, ferramenta para modelagem de dados, controle de versão de software, controle de mudanças em software operacional e avaliação da qualidade de pacote de software, baseado na ISO/IEC 9126 [ISO9126, 1994].

7.2 Aplicação do modelo

Como parte prática do trabalho, aplicou-se o modelo de Gerência de Projeto de Software na Embrapa Informática Agropecuária, considerando-se que esta unidade:

- adota uma abordagem de Gerenciamento pelas Diretrizes, implementado por sistema de administração estratégica (planos diretores, sistema de planejamento - SEP, entre outros) e pelo método de Gestão de Processo para gerenciamento da rotina do trabalho do dia (ainda não implementado na área de software, mas previsto para os próximos anos);
- possui estrutura organizada por projeto de pesquisa;
- deseja melhorar a capacidade de seus processos de software;
- quer entender melhor o quê seu cliente precisa;

- possui problemas como distorções de custo e prazo em relação ao planejado, dispêndio de esforço em retrabalho e documentação de projeto/usuário é pouca ou desatualizada.

Frente à estas considerações, foi necessário ajustar o modelo apenas no que se refere às seguintes particularidades da empresa: nomenclatura de produto de trabalho e atividade de avaliação de projeto de software.

Como os projetos de software estão vinculados a um projeto/subprojeto de pesquisa do Sistema Embrapa de Planejamento - SEP, objetivos e metas de projeto são negociados com as CTP e formalizados no documento *Relatório de elaboração de projeto/subprojeto do SEP*. O produto de trabalho correspondente a este documento no modelo proposto é 50) *Compromisso / acordo*. É necessária, então, a substituição de seu nome para 50) *Relatório de elaboração de projeto/subprojeto do SEP*. Esta substituição deixa a nomenclatura consistente com o sistema de planejamento da empresa e requer alteração nas descrições dos processos *CUS.3 Elicitação de requerimentos* e *MAN.2 Gerência de projeto*.

As alterações abrangeram as ferramentas: tabela de definição de escopo, macrodiagrama e descrição de atividades de fluxograma para os processos citados anteriormente. Os fluxogramas destes processos não sofreram alterações. Alguns dos pontos modificados podem ser vistos nas figuras 7.2 , 7.3, 7.4 e 7.5. Os demais processos não sofreram alterações.

A atividade de avaliação de projeto, que envolve a negociação de objetivos e metas de projeto, é realizada no Sistema Embrapa de Planejamento - SEP. Este sistema possui um mecanismo próprio de funcionamento, no qual a participação do gerente do projeto é através do *Relatório de elaboração de projeto / subprojeto do SEP*. Portanto, esta atividade está fora do contexto de processo de software. Este ajuste está representado na Figura 7.6, através da adaptação da seqüência de utilização dos processos, indicada com um retângulo de cor de fundo cinza.

Nome do processo:		CUS.3 Processo de elicitação de requerimentos	
Objetivo:	O propósito deste processo é obter, processar e acompanhar o desenvolvimento de necessidades e requerimentos de cliente durante a vida do produto e/ou serviço de software. Além disso, o processo deve estabelecer uma linha-básica de requerimentos, a partir da qual são definidos os produtos de trabalho de software necessários.		
Entradas:	21) Resultado de análise (funcional) 22) Registro de análise de risco 46) Registro/relatório de análise de mercado 51) Contrato 52) Especificação de requerimentos (cliente) 87) Mecanismo de comunicação	Saídas:	17) Plano de projeto 31) Registro de revisão 44) Avaliação de necessidades de produto 50) Relatório de elaboração de projeto/subprojeto do SEP 52) Especificação de requerimentos (cliente) 82) Procedimento de suporte de cliente 87) Mecanismo de comunicação 201) Esboço de requerimento de produto 202) Lista de perspectivas

Figura 7.2 Trecho da tabela de definição de escopo do processo CUS.3 Elicitação de requerimentos adaptada para a Embrapa Informática Agropecuária

Nome do processo:		MAN.2 Processo de gerência de projeto	
Objetivo:	O propósito deste processo é identificar, estabelecer, coordenar e monitorar atividades, tarefas e recursos necessários para um projeto produzir um produto e/ou serviço reunindo os requerimentos.		
Entradas:	2) Modelo de ciclo de vida 22) Análise de risco 23) Estratégia/plano de gerenciamento de riscos 51) Contrato 52) Especificação de requerimentos (cliente, software, sistema) 87) Mecanismo de comunicação 91) Estratégia / plano de gerência de configuração	Saídas:	5) Cronograma 6) Estrutura de decomposição de trabalho 17) Plano de projeto 21) Resultado de análise 30) Estratégia / plano de revisão 50) Relatório de elaboração de projeto/subprojeto do SEP 69) Plano de liberação 204) Planilha de custo e prazos da EDT

Figura 7.3 Trecho da tabela de definição de escopo do processo MAN.2 Gerência de projeto adaptada para a Embrapa Informática Agropecuária

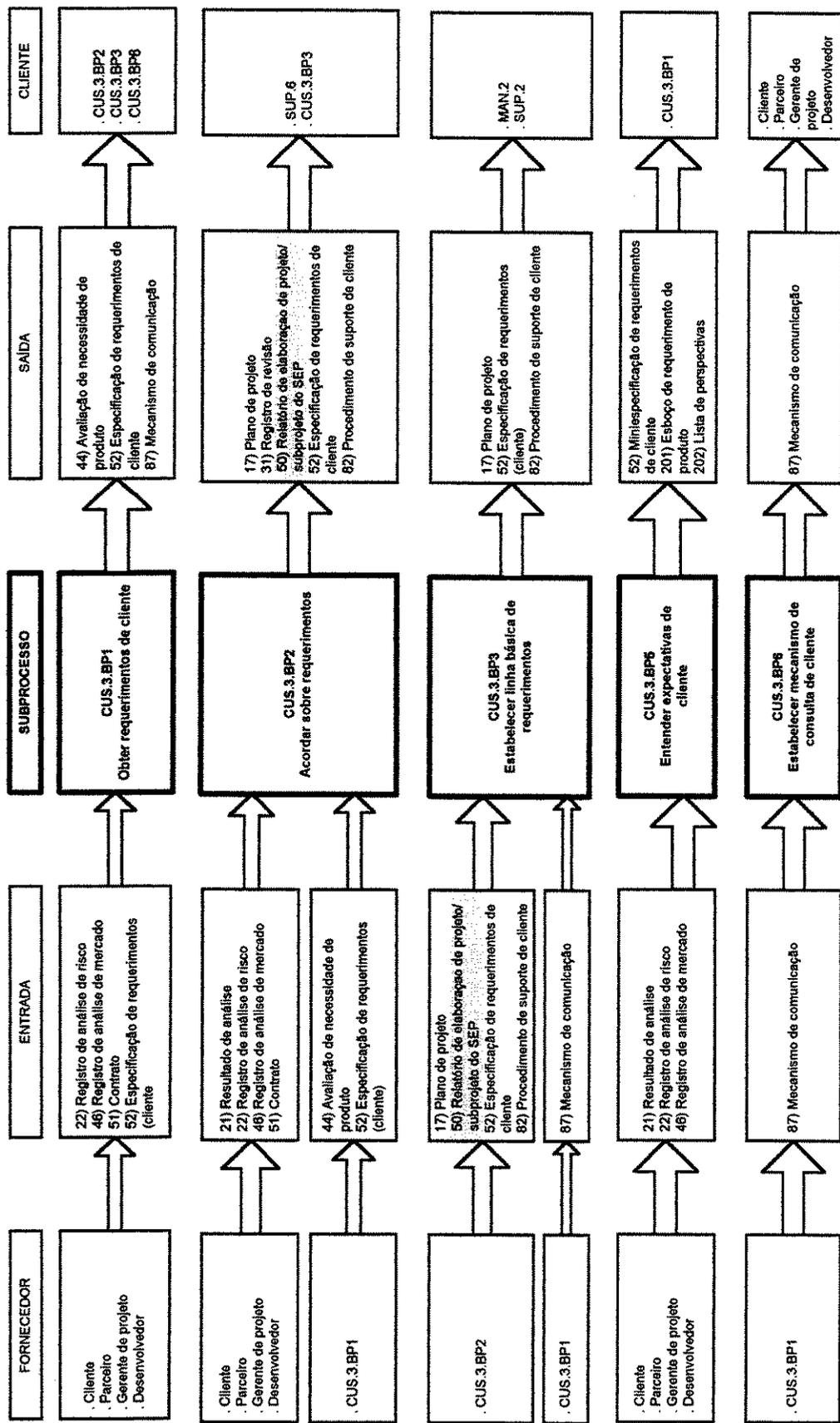


Figura 7.4 Macrodiagrama do processo CUS.3 Elicitação de requerimentos adaptado para a Embrapa Informática Agropecuária

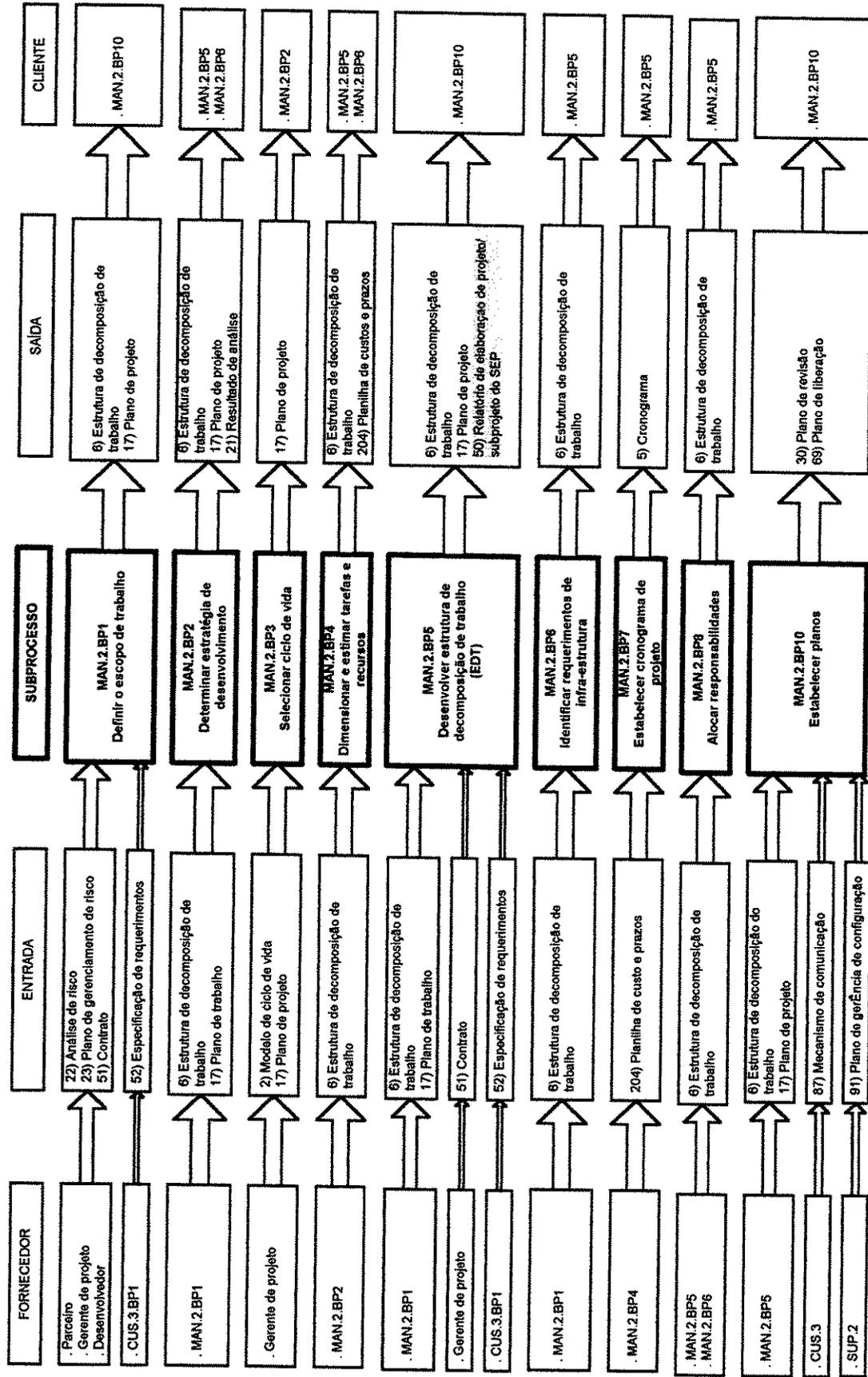


Figura 7.5 Macrodiagrama do processo MAN.2 Gerência de projeto adaptado para a Embrapa Informática Agropecuária

8 Conclusões

A indústria brasileira de software tem um grande potencial para participar do mercado de economia globalizada. Pesquisas mostram que ela está melhorando a sua capacidade gerencial, adotando sistemas modernos de gestão empresarial e que também está começando a buscar a sistematização de seu processo de produção de software. Por isso, ainda persistem problemas como garantir, mensurar e testar a qualidade de novos produtos de software.

Para contribuir com a sistematização do processo de produção de software, este trabalho propôs uma tecnologia que integra as dimensões *Gerenciamento da Qualidade Total*, *Qualidade em Processo de Software* e *Organização por Projeto*. Para isso, considerou que a maioria das empresas de software brasileiras:

- adota ou está implementando uma abordagem de Gerenciamento pelas Diretrizes para implementar seu programa de Gerenciamento da Qualidade Total;
- está implementando um método de Gestão de Processo para gerenciar a rotina do trabalho do dia-a-dia;
- adota uma estrutura de desenvolvimento de software organizada por projeto;
- deseja melhorar a capacidade de seus processos de software;
- quer entender melhor o que o seu cliente precisa;
- possui grandes distorções de prazo e custo em relação ao planejado;
- a documentação de desenvolvimento/usuário que possui é pouca ou desatualizada.

Fundamentando-se no Modelo de Referência da ISO/IEC TR 15504, em conceitos de gerenciamento da qualidade total e na literatura de engenharia de software, o desenvolvimento desta tecnologia resultou nas seguintes contribuições: um modelo de processo de software adaptado da ISO/IEC TR 15504; uma descrição para quatro processos, consistindo de escopo, macrodiagrama, fluxograma e descrição detalhada das atividades do fluxograma; e um modelo de gerência de projeto de software.

A descrição dos processo orientou-se pelo método de Gestão de Processo, executando as atividades de identificação dos processos de software, seleção dos mais prioritários e descrição dos processos selecionados através de definição de escopo, macrodiagrama, fluxograma e descrição das atividades do fluxograma.

A identificação dos processos baseou-se no modelo de referência para processos e capacidade de processos da ISO/IEC TR 15504. Foram excluídos cinco processos do modelo original pois considera-se que os mesmos tratados pela abordagem abrangente de gerenciamento da qualidade total.

Os processos *CUS.3 Elicitação de requerimentos*, *MAN.2 Gerência de Projeto*, *SUP.2 Gerência de Configuração* e *SUP.6 Revisão* foram considerados prioritários, pois o

estabelecimento deles contribui para a concretização das considerações deste trabalho.

A definição do escopo de cada processo selecionado resultou da reunião de informações contidas em dois documentos da ISO/IEC TR 15504 em uma tabela de definição de escopo.

A elaboração do macrodiagrama detalhou os processos, subdividindo-o em práticas básicas (recomendadas pela ISO/IEC TR 15504) e alocou os produtos de trabalho de entrada e saída definidos no escopo a cada uma das práticas básicas, além de relacionar fornecedor e cliente dos produtos, respectivamente. Com isso, pôde-se identificar as interfaces do processo.

A elaboração do fluxograma resultou do levantamento feito na literatura de engenharia de software sobre as atividades de cada prática básica e seu contexto de uso, distribuindo-as no gráfico numa seqüência lógica de uso para facilitar o entendimento. Tomou-se este cuidado porque a norma apenas relaciona as práticas, sem preocupar-se com a sua seqüência de uso.

A descrição de atividades de processo também resultou do levantamento acima, preocupando-se em orientar sobre técnicas, métodos ou ferramentas a serem utilizados para gerar os produtos de trabalho, sem prender-se a uma tecnologia específica. Pois o objetivo é gerar os produtos de trabalho com as suas características definidas para que eles "testemunhem" a execução das práticas básicas. A maneira "como" eles são gerados é uma sugestão inicial para que o desenvolvedor tenha acesso rápido a uma opção. Inclusive, nesta descrição ele também encontra referências para outras opções.

A definição de um Modelo de Gerência de Projeto de Software foi requerida para estabelecer os processos e possibilitar a sua implementação na realidade de uma empresa. Informações foram levantadas sobre modelos de gerência de projeto e fez-se uma especialização para projeto de software. Esta especialização foi detalhada para definir a seqüência de utilização dos processos.

Para verificar a aplicabilidade da tecnologia, ela foi aplicada no contexto da Embrapa Informática Agropecuária. Verificou-se que o modelo está bastante robusto, requerendo apenas ajustes à nomenclatura e procedimentos internos da empresa.

A utilização dos modelos propostos por este trabalho em uma organização que está implantando o método de Gestão de Processo contribui para promover uma melhoria gradativa e contínua dos processos (veja Figura 5.1). Na primeira etapa, o método estabelece os processos de software e assegura a utilização de boas práticas de engenharia de software e a geração de artefatos que "testemunham" a sua execução. Com isso, forma-se a base necessária para avaliar os processos de software. Na segunda etapa, pode-se utilizar o modelo de avaliação de processo de software da ISO/IEC TR 15504 para determinar a sua capacidade e seus pontos fortes e fracos. Os resultados da avaliação serão os subsídios para a elaboração de um plano de melhorias. Na terceira etapa, acompanha-se e controla-se as ações de melhoria implementadas e o ciclo se repete continuamente.

A implantação do modelo de Gerência de Projeto de Software permite que os processos de software sejam utilizados de uma forma organizada para atingir os objetivos do projeto, e consequentemente os objetivos de negócios da organização.

A sistematização do processo de software através de uma descrição que orienta-se pelos indicadores de execução do processo - práticas básicas e produtos de trabalho e as suas características - é o primeiro passo de uma longa caminhada em direção à maturidade do processo de software. Esta caminhada caracteriza-se pela incorporação gradativa de indicadores de capacidade de processo. A Figura 8.1 ilustra os níveis de capacidade de processo, sendo o nível 5 considerado processo maduro. O primeiro passo, representado na figura pela seta vermelha, indica que o processo está estabelecido e habilitado para incorporar novos atributos de qualidade e assim, melhorar a sua capacidade.

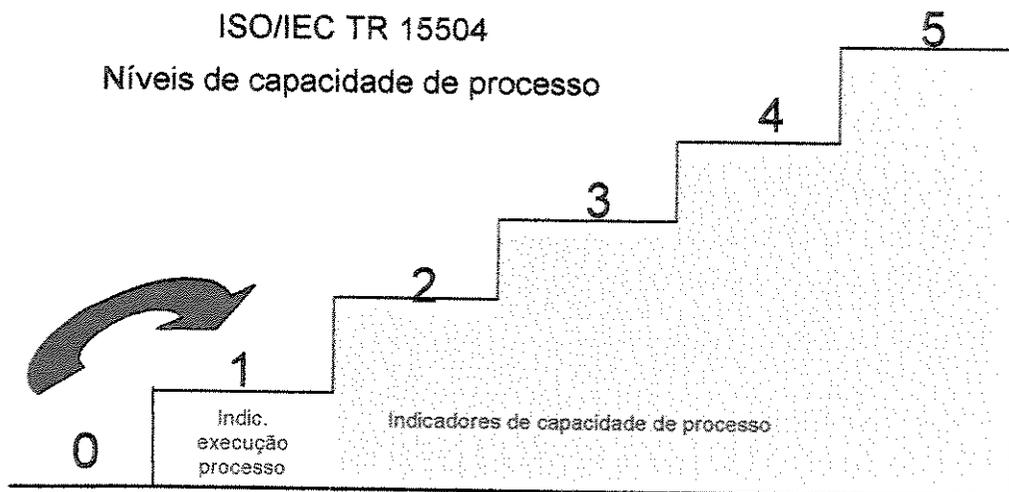


Figura 8.1 Primeiro passo para melhoria de capacidade de processo de software

Os próximos passos nesta caminhada de melhoria contínua de processo envolvem os seguintes trabalhos:

- desenvolvimento de uma ferramenta de suporte à implementação dos processos, como por exemplo, uma aplicação de controle do fluxo de trabalho ou uma *home page* que contenha descrição dos processos (em hipertexto) e *templates* de produtos de trabalho;
- descrição dos vinte e sete processos considerados por este trabalho como *não prioritários*, com as características de processo de nível de capacidade 1;
- incorporar às descrições dos processos as características de qualidade, pelo menos para os níveis de capacidade 2 e 3;
- com relação ao modelo de avaliação, promover o treinamento de avaliadores neste modelo e desenvolver uma ferramenta de apoio à execução de avaliação baseada no modelo.

9 Referências bibliográficas

- [Booch, 1999] BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *The unified modeling language user guide*. Reading: Addison-Wesley, 1999. 482p.
- [Brasil, 1993] BRASIL. Ministério de Ciência e Tecnologia. FINEP. PADCT. *Estudo da competitividade da indústria brasileira: competitividade da indústria de software*. Campinas, 1993. 65p.
- [Buckley, 1996] BUCKLEY, F. J. *Implementing configuration management: hardware, software and firmware*. Los Alamitos: IEEE Computer Society, 1996. 368p.
- [Campos, 1992] CAMPOS, V. F. *TQC: controle da qualidade total (no estilo japonês)*. Belo Horizonte, MG: UFMG, Escola de Engenharia, Fundação Christiano Ottoni, 1992. (Rio de Janeiro: Bloch Ed.)
- [Certo & Peter, 1993] CERTO, S. C., PETER, J. P. *Administração estratégica: planejamento e implementação da estratégia*. São Paulo: MAKRON Books, 1993. p.469.
- [Cheng et al., 1995] CHENG, L. C., SCAPIN, C. A., OLIVEIRA, C. A., KRAFETUSKI, E., DRUMOND, F. B., BOAN, F.S., PRATES, L. R., VILELA, R. M. *QFD: Planejamento da qualidade*. Belo Horizonte, MG: UFMG, Escola de Engenharia, Fundação Christiano Ottoni, 1995.
- [Emam et al., 1998] EMAM, K.E., DROUIN, J.-N., MELO, W. *SPICE: the theory and practice of software process improvement and capability determination*. Los Alamitos: IEEE Computer Society, 1998.
- [Embrapa, 1996a] EMBRAPA (Brasília, DF). Manual do Sistema Embrapa de Planejamento: elaboração e aprovação de projeto. *Boletim de Comunicações Administrativas*, Brasília, v.22, n.28, p.82-96, jul. 1996. (Norma, 037.01.03.01.5.020).
- [Embrapa, 1996b] EMBRAPA (Brasília, DF). Manual do Sistema Embrapa de Planejamento: elaboração e aprovação de suprojeto. *Boletim de Comunicações Administrativas*, Brasília, v.22, n.28, p.97-108, jul. 1996. (Norma 037.01.03.01.5.021).
- [Embrapa, 1998a] EMBRAPA. Assessoria de Comunicação Social (Brasília, DF). *A nossa Embrapa: III Plano Diretor*. Brasília, [1998]. 7p. (Encarte da Folha Embrapa, n. 35).
- [Embrapa, 1998b] EMBRAPA. Departamento de Organização e Desenvolvimento (Brasília, DF). *Gestão de Processo: Tecnologia gerencial voltada para resultados*. Brasília, 1998. 45p. (apostila de curso).

- [Embrapa, 1998c] EMBRAPA. Secretaria de Administração Estratégica (Brasília, DF). *III Plano diretor da Embrapa - realinhamento estratégico 1999-2003*. Brasília: EMBRAPA-SPI, 1998. 36p.
- [Embrapa..., 1999] EMBRAPA INFORMÁTICA AGROPECUÁRIA (Campinas, SP). *Plano Diretor da Embrapa Informática Agropecuária*. Campinas, 1999. No prelo.
- [Ferreira, 1988] FERREIRA, A. B. H. *Dicionário Aurélio básico da língua portuguesa*. Rio de Janeiro, RJ: Nova Fronteira, 1988.
- [Ferreira, 1995] FERREIRA, J. I. A. X. *A evolução da qualidade e sua contribuição para o ganho de vantagem competitiva das empresas*. Campinas, SP: IMECC-UNICAMP, 1995. 145p. (Tese de mestrado)
- [Goedert et al., 1994] GOEDERT, W. J., PAEZ, M. L. D., CASTRO, A.M.G. *Gestão em ciência e tecnologia: pesquisa agropecuária*. Brasília: EMBRAPA-SPI, 1994.
- [Humphrey, 1997] HUMPHREY, W. Por dentro da criação da Capability Maturity Model. *Developers' Magazine*, v.1, n.7., p.36-38, jul. 1997. Entrevista.
- [ISO15504, 1998] ISO/IEC. *ISO/IEC TR 15504:1998(E) - Information technology, software process assessment*. Parts:1-9. Canada, 1998.
- [ISO9126, 1994] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (Rio de Janeiro, RJ). *NBR ISO/IEC 9126 - Tecnologia de informação, Avaliação de produto de software, características de qualidade e diretrizes para o seu uso*. Rio de Janeiro, 1994. 11p.
- [Mikkelsen, 1997] MIKKELSEN, T., PHERIGO, S. *Practical software configuration management: the latenight developer's handbook*. Upper Saddle River: Prentice Hall, 1997. 301p.
- [Oakland, 1994] OAKLAND, J. S. *Gerenciamento da qualidade total: TQM*. São Paulo, SP: Nobel, 1994.
- [Pacheco et al., 1997] PACHECO, H.A., SANTOS, A.D. dos, FIGUEREDO, K.N., CHAIM, M. L., PEDROSO JÚNIOR, M., FILETO, R. *Cartilha Azul: guia do processo de desenvolvimento de software do CNPTIA*. Campinas: Embrapa-CNPTIA, 1997. 53p. (EMBRAPA-CNPTIA. Documentos, 1).
- [Pressman, 1995] PRESSMAN, R. S. *Engenharia de software*. São Paulo: Makron Books, 1995. 1056p.
- [Qualidade..., 1998] QUALIDADE NO SETOR DE SOFTWARE BRASILEIRO. 1997. Brasília: Ministério da Ciência e Tecnologia, n.2, 1998. 120p.

[Sashkin & Kiser, 1994] SASHKIN, M., KISER, K. J. Gestão da qualidade total na prática: o que é TQM, como usá-la e como sustentá-la a longo prazo. Rio de Janeiro, RJ: Campus, 1994.

[Valeriano, 1998] VALERIANO, D. L. *Gerência em projetos: pesquisa, desenvolvimento e engenharia*. São Paulo: Makron Books, 1998.

ANEXO I - Características-chave de produtos de trabalho

As características de produtos de trabalho (PT) listadas neste anexo na Tabela I.2 podem ser usadas durante a revisão de entradas e saídas potenciais de uma implementação de processo de organização. As características são dadas como guia para oferecer evidência objetiva e suporte a avaliação de um determinado processo. Estas características devem ser observadas dentro do contexto do propósito do processo e não são uma lista de verificação que toda empresa deve ter. Os produtos de trabalho do intervalo de 1 a 109 foram extraídos da ISO/IEC TR 15504 e os acima de 200 foram definidos por este trabalho. Os campos da Tabela I.2 contêm as informações a seguir.

<i>Identificador do produto de trabalho</i>	Um número de identificador para produto de trabalho que é usado para referenciar o produto de trabalho.
<i>Classificação de produto de trabalho</i>	Fornecer uma classificação dos produtos de trabalho em três categorias: Organização, Projeto e Registros, como mostrado na Tabela I.1.
<i>Tipo de produto de trabalho</i>	Fornecer um exemplo de um nome típico associado com as características do produto de trabalho. Este nome é fornecido como um identificador de tipo de produto de trabalho que a prática ou processo pode produzir.
<i>Características de produto de trabalho</i>	Fornecer um exemplo das características potenciais associadas com os tipos de produto de trabalho.

Tabela I.1 Classificação de Produtos de Trabalho

No. Categoria de PT	Categoria de PT	No. Classificação de PT	Classificação de PT
1	Organização	1.1 1.2 1.3 1.4	Política Procedimento Padrão Estratégia
2	Projeto	2.1 2.2 2.3 2.4 2.5 2.6	Plano Requerimento Design Implementação Produto "Interim deliverable"
3	Registros	3.1 3.2 3.2 3.4	Relatório Registro Medida Dados

Na Tabela I.2, alguns produtos de trabalho estão seguidos pelo símbolo + (n). Isto significa que o produto de trabalho é um exemplo de um produto de trabalho genérico. A descrição de produto de trabalho genérico correspondente às características comuns a todos os produtos de trabalho deste tipo. Portanto, as características comuns seriam consideradas em adição às características específicas incluídas para o produto de trabalho em questão.

Tabela I.2 Características-chave de Produtos de Trabalho

ID	Classe de PT	Tipo de PT	Características de PT
1	1.2	Metodologia de desenvolvimento de software	<ul style="list-style-type: none"> ▪ identificação da abordagem / método usado para desenvolver software ▪ identificação do modelo de ciclo de vida (cascata, espiral, construção serial, etc) usado para desenvolver software ▪ provê uma descrição de alto-nível do processo, atividades e controles
2	1.3	Modelo de ciclo de vida	<ul style="list-style-type: none"> ▪ descrição de alto-nível de atividades executadas em cada fase do ciclo de vida ▪ seqüenciamento das fases do ciclo de vida ▪ identificação de dependências de fase crítica de ciclo de vida ▪ identificação de entradas e saídas requeridas para cada fase do ciclo de vida ▪ identificação do modelo de pontos-chave de decisão (<i>milestones</i>) ▪ identificação dos pontos de controle de qualidade no modelo
3	1.2	Descrição de processo	<ul style="list-style-type: none"> ▪ uma descrição detalhada do processo inclui: <ul style="list-style-type: none"> ▪ <i>tailoring</i> do processo padrão (se aplicável) ▪ propósito do processo ▪ tarefa e atividades a serem executadas e ordenação das tarefas ▪ dependências críticas entre atividades e tarefas ▪ tempo esperado requerido para executar tarefa ▪ produtos de trabalho de entrada e de saída ▪ ligações entre produtos de trabalho de entrada e de saída ▪ identifica entrada de processo e critério de saída ▪ identifica interfaces internas e externas para o processo ▪ identifica medidas de processo ▪ identifica expectativas de qualidade ▪ identifica papéis funcionais e responsabilidades
4	1.2	Procedimento ou prática de trabalho	<ul style="list-style-type: none"> ▪ cada tarefa a ser executada está unicamente identificada ▪ cada tarefa está sequenciada por ordem de execução ▪ cobertura de informação de suporte (i.e., comandos e colocação de parâmetros, etc.) quando requerido para operações ▪ estabelece regras pelas quais espera-se que o staff opere ▪ aprovado por pessoa autorizada

5	2.1	Cronograma	<ul style="list-style-type: none"> ▪ identifica as tarefas a serem executadas ▪ identifica a data de início e conclusão para as tarefas requeridas ▪ considera a identificação de tarefas críticas e dependências de tarefas ▪ identifica status de conclusão de tarefas vs. data planejada ▪ tem um mapeamento para dados de recursos planejados
6	2.1	Estrutura de desdobramento de trabalho	<ul style="list-style-type: none"> ▪ define tarefas a serem executadas ▪ documenta proprietários para as tarefas ▪ documenta dependências críticas entre tarefas ▪ documenta produtos de trabalho de entrada e saída ▪ documenta as dependências críticas entre os produtos de trabalho definidos
7	2.5	Produto de trabalho	<ul style="list-style-type: none"> ▪ define os atributos associados com um artefato de uma execução de processo: <ul style="list-style-type: none"> ▪ elementos-chave a serem representados no produto de trabalho ▪ forma e estilo esperados ▪ meio esperado (papel, eletrônico) e atributos de armazenamento definidos
8	1.3	Interface	<ul style="list-style-type: none"> ▪ define relações entre dois produtos, processo ou tarefas de processo ▪ define critérios e formato para o quê é comum para ambos ▪ define critérios de dependência de tempo ou seqüência de ordem críticos
9	1.3	Padrão	<ul style="list-style-type: none"> ▪ identificação de quem / o quê eles aplicam ▪ cada requerimento é único ▪ cada requerimento é marcado com um identificador ▪ expectativas para conformidade são identificadas ▪ conformidade para requerimentos podem ser demonstradas ▪ condições para <i>tailoring</i> ou exceções para os requerimentos são incluídas
10	1.3	Padrão de codificação	<ul style="list-style-type: none"> ▪ cobertura para software inclui, mas não é limitado (como apropriado para a aplicação): <ul style="list-style-type: none"> ▪ convenções de nomes de dados ▪ define linguagens, compiladores, sistema gerenciador de banco de dados, etc. requeridos ▪ formato requerido de código, estrutura, comentários ▪ padrão de estrutura de dados, tipos, classes ▪ melhores práticas ▪ uso requerido de ferramentas: dicionário de dados, ferramentas CASE associadas ▪ requerimento de compatibilidade para software existente e/ou hardware

			<ul style="list-style-type: none"> ▪ considerações de segurança ▪ considerações de desempenho ▪ padrão de mensagens e código de erro ▪ padrões de interface: interface homem-máquina, interfaces de sistema externas, equipamentos periféricos, hardware ▪ armazenamento e recuperação de código-fonte e módulos de objetos ▪ padrões de qualidade e confiabilidade
11	3.3	Estimativa	<ul style="list-style-type: none"> ▪ cobertura (como apropriado para a aplicação) para elementos tais como: <ul style="list-style-type: none"> ▪ tamanho ▪ esforço ▪ custo ▪ cronograma ▪ recursos ▪ estimativas são realistas e alcançáveis: <ul style="list-style-type: none"> ▪ em linha com recursos alocados ▪ em linha com registros históricos (quando eles existirem) ▪ fonte de dados necessária para fazer estimativas estava disponível e completa ▪ fonte de dados estava validada
12	1.1	Meta (negócio, qualidade, organizacional, time, treinamento, desempenho, processo)	<ul style="list-style-type: none"> ▪ identifica o objetivo a ser obtido ▪ identifica quem é esperado alcançar a meta ▪ identifica quaisquer metas de suporte incrementais ▪ identifica quaisquer condições / restrições ▪ identifica o período para alcance; ▪ são razoáveis e alcançáveis com os recursos alocados ▪ são correntes estabelecidas para projeto ou organização corrente ▪ usadas para monitorar progresso ▪ são otimizadas para suportar critérios de desempenho e planos conhecidos
13	1.1	Visão	<ul style="list-style-type: none"> ▪ provê informação sobre toda estratégia para a unidade organizacional, organização ou negócio ▪ é autorizada no nível mais alto ▪ define os objetivos principais a serem alcançados
14	1.1	Política	<ul style="list-style-type: none"> ▪ autorizada ▪ disponível para todas as pessoas impactadas pela política ▪ estabelece práticas / regras a serem aderidas
16	2.1	Plano (atributos gerais que se aplicam para todos os planos, i.e.,	<p>Como apropriado para a aplicação e propósito:</p> <ul style="list-style-type: none"> ▪ identificação do proprietário do plano ▪ inclui:

		Negócio, Organização, Projeto, Qualidade, Revisão e Teste)	<ul style="list-style-type: none"> ▪ o objetivo do que será realizado ▪ suposições (hipóteses) feitas ▪ restrições ▪ riscos ▪ tarefas a serem realizadas ▪ cronograma, marcos e datas alvo ▪ dependências críticas ▪ disposição de manutenção para o plano ▪ método/abordagem para realizar o plano ▪ identifica: <ul style="list-style-type: none"> ▪ posse de tarefa ▪ critério de qualidade ▪ auditoria a ser executada ▪ produtos de trabalho requeridos ▪ inclui recursos para realizar objetivos do plano <ul style="list-style-type: none"> ▪ tempo ▪ staff ▪ materiais/equipamentos ▪ orçamento ▪ inclui plano de contingência para tarefas não-completadas ▪ status do plano (aprovado, em revisão, em elaboração, etc)
17	2.1	Plano de projeto + (16)	<ul style="list-style-type: none"> ▪ define: <ul style="list-style-type: none"> ▪ produtos de trabalho a serem desenvolvidos ▪ modelo de ciclo de vida e metodologia a ser usada ▪ requerimentos de cliente ▪ tarefas a serem executadas ▪ posse da tarefa ▪ recursos de projeto ▪ cronograma, marcos e datas-alvo ▪ critério de qualidade ▪ identifica: <ul style="list-style-type: none"> ▪ dependências críticas ▪ produtos de trabalho requeridos ▪ riscos de projeto e plano de abrandamento de risco/ações de contingência para tarefas não-completadas
18	3.4	Dados de execução de processo	<ul style="list-style-type: none"> ▪ dados comparando execução de processo contra níveis esperados: <ul style="list-style-type: none"> ▪ produtos de trabalho de entrada e de saída definidos disponíveis ▪ atas de reunião ▪ registros de mudanças

			<ul style="list-style-type: none"> ▪ critério de conclusão de tarefa encontrado ▪ critério de qualidade encontrado ▪ alocação e acompanhamento de recurso
19	3.2	Atas de reunião	<ul style="list-style-type: none"> ▪ documenta atas mantidas ▪ define: <ul style="list-style-type: none"> ▪ propósito de reunião ▪ participantes ▪ data e local mantidos ▪ o que foi realizado ▪ pontos em aberto ▪ próxima ação
20	3.2	Registro/relatório de status de progresso	<ul style="list-style-type: none"> ▪ registro de status de um ou mais planos (atual contra planejado), tais como: <ul style="list-style-type: none"> ▪ status de tarefas atuais contra tarefas planejadas ▪ status de resultados atuais contra objetivos / metas estabelecidos ▪ status de alocação de recursos atual contra recursos planejados ▪ status de custo atual contra orçamento estimado ▪ status de tempo atual contra cronograma planejado ▪ status de qualidade atual contra qualidade planejada ▪ registro de quaisquer desvios das atividades planejadas e porquê razão
21	3.4	Resultado de análise	<ul style="list-style-type: none"> ▪ o quê foi analisado ▪ quem fez a análise ▪ os critérios de análise usados: <ul style="list-style-type: none"> ▪ critério de seleção ou esquema de priorização usado ▪ critérios de decisão ▪ critérios de qualidade ▪ registros de resultados: <ul style="list-style-type: none"> ▪ o quê foi decidido/selecionado ▪ razão para seleção ▪ suposições feitas ▪ riscos potenciais ▪ aspectos de corretitude para analisar incluem: <ul style="list-style-type: none"> ▪ completude ▪ facilidade de entendimento ▪ testabilidade ▪ verificabilidade ▪ viabilidade

			<ul style="list-style-type: none"> ▪ validade ▪ consistência ▪ adequação de conteúdo
22	3.2	Registro/relatório de análise de risco	<ul style="list-style-type: none"> ▪ identifica os riscos analisados ▪ registra os resultados da análise: <ul style="list-style-type: none"> ▪ maneiras potenciais para amenizar o risco ▪ suposições feitas ▪ restrições
23	1.4 / 2.1	Estratégia / plano de gerenciamento de risco + (59)	<ul style="list-style-type: none"> ▪ riscos de projeto identificados e priorizados ▪ mecanismo para acompanhar o risco ▪ limiar critérios para identificar quando ação corretiva é requerida ▪ maneiras propostas para atenuar riscos: <ul style="list-style-type: none"> ▪ <i>work around</i> ▪ atividades / tarefas de ações corretivas ▪ critérios de monitoramento ▪ mecanismos para medir risco
24	1.1	Statement / política de qualidade + (14)	<ul style="list-style-type: none"> ▪ <i>statement</i> é oficial e aprovado ▪ apresenta compromissos com os princípios de qualidade ▪ identifica quem é esperado seguir a política
25	1.4 / 2.1	Estratégia / plano de qualidade + (16)	<ul style="list-style-type: none"> ▪ objetivos / metas para qualidade ▪ define as atividades e tarefas requeridas para garantir qualidade ▪ referencia produtos de trabalho relacionados ▪ método de avaliação / garantia de qualidade ▪ referencia quaisquer requerimentos "regulatory" , padrões requerimentos de cliente ▪ identifica os critérios de qualidade esperados ▪ especifica o período de monitoramento e pontos de verificação de qualidade para o ciclo de vida definido e atividades associadas planejadas ▪ período-alvo para alcançar a qualidade desejada ▪ método para metas alcançadas: <ul style="list-style-type: none"> ▪ tarefas a serem executadas ▪ dono das tarefas ▪ auditoria a ser executada ▪ recursos e compromissos ▪ identifica os critérios de qualidade para produtos de trabalho e tarefas de processos ▪ specifies the threshold / tolerance level allowed prior to requiring corrective actions ▪ define medidas de qualidade e dados de <i>benchmark</i> ▪ define o mecanismo de coleta e registro e tempo de coleta

			<ul style="list-style-type: none"> ▪ specifies mechanism to feed collected quality record back into process impacted by poor qualidade ▪ aprovado pela função / organização reponsável por qualidade
26	1.4	Oportunidade de melhoria	<ul style="list-style-type: none"> ▪ identifica o quê é o problema ▪ identifica o que causa de um problema ▪ sugere o quê poderia ser feito para consertar o problema ▪ identifica o valor (exceto benefício) em execução de melhoria ▪ identifica a penalidade para <i>não fazer a melhoria</i>
27	3.3	Crítérios de qualidade	<ul style="list-style-type: none"> ▪ define expectativas para qualidade: <ul style="list-style-type: none"> ▪ estabelece o que é um produto de trabalho adequado (elementos requeridos, completude esperada, precisão, etc.) ▪ identifica o quê constitui a completude das tarefas definidas ▪ estabelece crítérios de transição de ciclo de vida e requerimentos de entrada e saída para cada processo e/ou atividade definidos ▪ estabelece atributos de desempenho esperados ▪ estabelece atributos de confiabilidade de produto
28	3.2	Registro de qualidade	<ul style="list-style-type: none"> ▪ define que informação manter ▪ define quais tarefas/atividades/processo produz a informação ▪ define quando os dados serão coletados ▪ define fonte de quaisquer dados associados ▪ identifica os crítérios de qualidade associados ▪ identifica quaisquer medidas usando a informação ▪ identifica quaisquer requerimentos de aderência para criar o registro ou satisfeitos pelo registro
29	3.2	Registro de avaliação / auditoria	<ul style="list-style-type: none"> ▪ apresenta o propósito da avaliação ▪ método usado para avaliação ▪ requerimentos usados para avaliação ▪ suposições e limitações ▪ identifica informação de contexto e de escopo requeridos: <ul style="list-style-type: none"> ▪ data de avaliação ▪ unidade organizacional avaliada ▪ informação do patrocinador ▪ time de avaliação ▪ participantes ▪ escopo / cobertura ▪ informação de avaliadores ▪ instrumento de avaliação usado (check-list, ferramenta, etc.) ▪ registros de resultado <ul style="list-style-type: none"> ▪ identifica as ações corretivas requeridas ▪ oportunidades de melhorias

30	1.4/ 2.1	Estratégia de revisão/plano + (16)	<ul style="list-style-type: none"> ▪ define: <ul style="list-style-type: none"> ▪ o que será revisado ▪ papéis e responsabilidades de revisores ▪ critério para revisão (check-lists, requerimentos, padrões, etc.) ▪ tempo de preparação esperado ▪ cronograma para revisões ▪ identificação de: <ul style="list-style-type: none"> ▪ procedimentos para condução de revisão ▪ entradas e saídas de revisão ▪ perícia esperada em cada revisão ▪ registros de revisão para manter ▪ medidas de revisão para manter ▪ recursos e ferramentas alocados para a revisão
31	3.2	Registro de revisão	<ul style="list-style-type: none"> ▪ provê informação de contexto sobre a revisão <ul style="list-style-type: none"> ▪ o que foi revisado ▪ relaciona revisores que participaram ▪ status da revisão ▪ provê informação sobre a cobertura da revisão <ul style="list-style-type: none"> ▪ check-lists ▪ critérios de revisão ▪ requerimentos ▪ conformidade a padrões ▪ registros de informações sobre a preparação para a revisão ▪ tempo de preparação gasto para a revisão ▪ revisores, papéis e perícia ▪ identifica as ações corretivas requeridas <ul style="list-style-type: none"> ▪ identificação de risco ▪ lista priorizada de desvios e problemas descobertos ▪ as ações e tarefas a serem executadas para fixar o problema ▪ posse para ação corretiva ▪ status e datas de fechamento para problemas identificados
32	2.1	Plano de reuso + (16)	<ul style="list-style-type: none"> ▪ define a política sobre a qual itens serão reusados ▪ define padrões para construção de objetos reusáveis <ul style="list-style-type: none"> ▪ define os atributos de componentes reusáveis ▪ expectativa de qualidade/confiabilidade ▪ padrão para convenções de nomes ▪ define o repositório de reuso (biblioteca, ferramenta CASE, arquivo, base de dados, etc.) ▪ identifica componentes reusáveis

			<ul style="list-style-type: none"> ▪ diretório de componente ▪ descrição de componente ▪ aplicabilidade de seu uso ▪ método para recuperá-lo e usá-lo ▪ restrições para modificações e uso ▪ método para uso de componentes reusáveis ▪ estabelece meta para componentes reusáveis
33	1.4/ 2.2	Estratégia de reuso/ especificação	<ul style="list-style-type: none"> ▪ identifica se as metas para reuso estão declaradas (apresentadas) ▪ identifica o compromisso para criação de componentes reusáveis ▪ identifica quais linhas de produto e tipo de artefatos seriam suportados com reuso ▪ identifica componentes de sistema e de software que possam ser reusados dentro da organização ▪ identifica o repositório e ferramentas de reuso
34	2.5	Objeto reusável	<ul style="list-style-type: none"> ▪ desenvolvido para ser: <ul style="list-style-type: none"> ▪ altamente confiável ▪ definido genericamente (nomes e estruturas genéricos, etc.) ▪ interfaces (entradas e saídas) claras ▪ dados encapsulados ▪ modificação controlada ▪ <i>modifications are downward compatible</i> ▪ especificação para uso definida ▪ especificação para feito definida
35	2.5	Repositório de reuso	<ul style="list-style-type: none"> ▪ repositório para componentes reusáveis (biblioteca, arquivo, base de dados, ferramenta) ▪ capacidade de armazenamento e recuperação ▪ habilidade para "dar uma olhada" (browse) no conteúdo ▪ listagem de conteúdo com descrição de atributos reusáveis ▪ habilidade para identificar informação de sistema associado <ul style="list-style-type: none"> ▪ tipo do objeto mantido ▪ software/aplicações suportados ▪ informação de configuração de hardware associado ▪ informação de parâmetro requerido
36	3.3	Medida (general applies to all specific measures)	<ul style="list-style-type: none"> ▪ disponível para aqueles com uma necessidade de saber ▪ entendida por aqueles que espera-se que a usem ▪ provê valor para a organização / projeto ▪ não-perturba o fluxo de trabalho ▪ destinada a processo, modelo de ciclo de vida e organização: <ul style="list-style-type: none"> ▪ é precisa ▪ fonte de dados está validada

			<ul style="list-style-type: none"> ▪ resultados são validados para garantir precisão ▪ has appropriate analysis and commentary to allow meaningful interpretation by users
37	3.3	Medida de projeto + (36)	<p>Cobertura para elementos-chave no plano de projeto, tais como:</p> <ul style="list-style-type: none"> ▪ monitorar processos chave e tarefas críticas fornecendo informação de status de projeto sobre: <ul style="list-style-type: none"> ▪ execução de projeto contra plano estabelecido ▪ utilização de recurso contra plano estabelecido ▪ tempo de cronograma contra plano estabelecido ▪ qualidade de processo contra expectativas e/ ou critérios de qualidade ▪ qualidade de produto contra expectativas e/ ou critérios de qualidade ▪ problemas e tendências de execução de software <i>highlight</i> ▪ referenciar quaisquer metas estabelecidas
38	3.3	Medida de processo + (36)	<ul style="list-style-type: none"> ▪ medidas sobre a execução de processo: <ul style="list-style-type: none"> ▪ habilidade suficiente para produzir produtos de trabalho ▪ aderência ao processo ▪ tempo que toma para executar o processo ▪ defeitos relacionados ao processo ▪ medir o impacto de mudança de processo ▪ medir a eficiência do processo
39	3.3	Medida de qualidade + (36)	<ul style="list-style-type: none"> ▪ mede atributos de qualidade de produtos de trabalho definidos <ul style="list-style-type: none"> ▪ produto é adequado para fazer o trabalho intencionado ▪ produto é livre de defeito ▪ produto é usável ▪ produto está completo ▪ produto é preciso ▪ produto é confiável ▪ mede os resultados de atividades de projeto <ul style="list-style-type: none"> ▪ tarefas são executadas sobre um cronograma ▪ desenvolvimento de produto está dentro dos recursos e compromissos alocados ▪ mede atributos de qualidade da qualidade do produto final e de confiabilidade
40	3.3	Medida de risco + (36)	<ul style="list-style-type: none"> ▪ identifica a probabilidade de o risco ocorrer ▪ estabelece medidas para cada risco definido ▪ mede a mudança no estado do risco
41	3.3	Medida de campo + (36)	<ul style="list-style-type: none"> ▪ atributos de medida de desempenho de operação de sistema em posição de campos, tais como :

			<ul style="list-style-type: none"> ▪ defeitos de campo ▪ execução (desempenho) contra medida de nível de serviço definida ▪ habilidade de sistema para reunir requerimentos de cliente definidos ▪ tempo de suporte requerido ▪ reclamações de usuário ▪ requisições de cliente para ajuda ▪ tendências de desempenho ▪ relatórios de problemas ▪ melhorias requisitadas
42	3.3	Medida de nível de serviço + (36)	<ul style="list-style-type: none"> ▪ tomando medidas de tempo-real enquanto um sistema está operacional, mede o desempenho do sistema ou nível de serviço esperado ▪ identifica coisas como: <ul style="list-style-type: none"> ▪ capacidade ▪ <i>throughput</i> ▪ desempenho operacional ▪ serviço operacional ▪ <i>service outage time</i> ▪ <i>up time</i> ▪ <i>job run time</i>
44	2.2	Avaliação de necessidade de produto	<p>Cobertura para elementos-chave (como apropriado para a aplicação):</p> <ul style="list-style-type: none"> ▪ definição de necessidade: <ul style="list-style-type: none"> ▪ razão da necessidade do produto ▪ características e funções desejadas ▪ requerimentos a serem satisfeitos ▪ restrições: <ul style="list-style-type: none"> ▪ limitações de custo ▪ requerimentos de data/cronograma ▪ software de suporte específico requerido ▪ requerimentos de interfaces ▪ equipamento ou hardware associado requerido ▪ padrões e/ou requerimentos regulatórios ▪ impactos operacionais ▪ características de patente, copyright e licença ▪ caso de negócio: <ul style="list-style-type: none"> ▪ benefícios esperados ▪ custo esperado (incluindo instalação projetada, conversão e/ou manutenção) x expectativas de vantagem ▪ <i>market window</i>, datas de entrega alvo

45	1.4 / 2.1	Estratégia / plano de aquisição + (16)	<ul style="list-style-type: none"> ▪ identifica o quê necessita ser adquirido ▪ estabelece a abordagem para aquisição do produto ou serviço; opções podem incluir: <ul style="list-style-type: none"> ▪ off-the-shelf ▪ desenvolver internamente ▪ desenvolver através de contrato ▪ melhorar produto de software existente ▪ ou combinação das opções acima ▪ estabelece critérios de avaliação: <ul style="list-style-type: none"> ▪ estratégia de aceitação ▪ identifica quaisquer restrições / riscos
46	3.1/ 3.2	Registro/retatório de análise de mercado	<ul style="list-style-type: none"> ▪ contém informações sobre: <ul style="list-style-type: none"> ▪ o que foi analisado ▪ os critérios de seleção & esquema de priorização usados ▪ os critérios de análise usados ▪ registra os resultados que identificam: <ul style="list-style-type: none"> ▪ oportunidades de mercado e <i>market window</i> ▪ direcionadores de negócio ▪ custo/benefício ▪ clientes potenciais e informações de seu perfil ▪ quaisquer suposições feitas ▪ soluções alternativas consideradas e/ou rejeitadas ▪ riscos e retrições (características regulatórias) ▪ definir a proposta de produto e liberação alvo (<i>target release</i>)
50	3.2	Compromisso / acordo	<ul style="list-style-type: none"> ▪ assinado por todas as partes envolvidas no compromisso ou acordo ▪ estabelece o que trata o compromisso ▪ estabelece os recursos requeridos para atender por completo o acordo, tal como: <ul style="list-style-type: none"> ▪ tempo ▪ pessoal ▪ orçamento ▪ equipamento ▪ facilidades
51	3.2	Contrato (produto ou serviço)	<ul style="list-style-type: none"> ▪ assinado ▪ define o que vai ser comprado/entregue ▪ identifica "time frame" para entrega ou datas de serviço contratado ▪ identifica condições monetárias ▪ identifica qualquer informação de garantia ▪ identifica qualquer informação de copyright e licença

			<ul style="list-style-type: none"> ▪ identifica quaisquer requerimentos de serviço de cliente ▪ referencia para qualquer desempenho e expectativas / restrições / monitoramento de qualidade ▪ padrões e procedimentos a serem usados ▪ como apropriado ao contrato, o seguinte é considerado: <ul style="list-style-type: none"> ▪ referências a quaisquer critérios de aceitação ▪ referências a quaisquer necessidade de cliente especiais (i.e., requerimentos de confidencialidade, segurança, hardware, etc) ▪ referências para quaisquer procedimentos de gerência de mudança e resolução de problema ▪ identifica quaisquer interfaces para agentes independentes e subcontratadores ▪ identifica papel do cliente ou parceiro no processo de desenvolvimento e manutenção
52	2.2	<p>Especificação de requerimento (interno ou externo)</p> <p>(produto, serviço, cliente, sistema, software, documentação, ambiente)</p>	<ul style="list-style-type: none"> ▪ cada requerimento é identificado ▪ cada requerimento é único ▪ cada requerimento é verificável ou pode ser avaliado ▪ inclui requerimentos estatutório e regulatório ▪ inclui características / requerimentos a partir da revisão de contrato ▪ consideração é dada para o seguinte (como apropriado para o produto ou serviço e tipo de requerimento) ▪ requerimentos de produtos/aplicações: <ul style="list-style-type: none"> ▪ identifica quaisquer características funcionais requeridas ▪ identifica quaisquer considerações/restrições de desempenho(execução) necessárias ▪ identifica quaisquer considerações/restrições de interface externa/interna necessárias ▪ identifica quaisquer características/restrições de sistema requeridas ▪ identifica quaisquer considerações/restrições de engenharia humana ▪ identifica quaisquer considerações/restrições de segurança (<i>security</i>) ▪ identifica quaisquer considerações/restrições ambientais ▪ identifica quaisquer considerações/restrições operacionais ▪ identifica quaisquer considerações/restrições de manutenção ▪ identifica quaisquer considerações/restrições de documentação associada ▪ identifica quaisquer considerações/restrições de instalação ▪ identifica quaisquer considerações/restrições de suporte ▪ identifica quaisquer considerações/restrições de <i>design</i> ▪ identifica quaisquer considerações/restrições de <i>safety</i> / confiabilidade ▪ identifica quaisquer requerimentos/expectativas de qualidade

			<ul style="list-style-type: none"> ▪ inclui requerimentos de armazenamento (produtos) ▪ requerimentos de serviços: <ul style="list-style-type: none"> ▪ identifica quaisquer expectativas de desempenho ▪ identifica quaisquer cronograma de tempo ou restrições ▪ identifica quaisquer tarefas a serem executadas ▪ identifica quaisquer responsabilidades ▪ identifica o método de comunicação e relatório de projeto esperado ▪ identifica quaisquer expectativas/controles de qualidade ▪ requerimentos de documento: <ul style="list-style-type: none"> ▪ propósito/objetivos definidos ▪ conteúdos propostos (cobertura) definidos ▪ audiência intencionada definida ▪ identificação de informação de versão de software e de sistema suportada ▪ identificação de requerimentos de software e designs associados satisfeitos pelo documento ▪ identificação de estilo, formato, e padrões de mídia esperados ▪ definição do requerimento de distribuição intencionado ▪ inclui requerimentos de armazenamento
53	2.3	Design/arquitetura de sistema	<ul style="list-style-type: none"> ▪ provê uma visão geral de todo <i>design</i> de sistema ▪ descreve a inter-relação entre os componentes de sistema ▪ descreve a relação entre os componentes de sistema e software ▪ especifica o design para cada componente de sistema requerido; consideração é dada para coisas como: <ul style="list-style-type: none"> ▪ requerimentos de capacidade/memória ▪ requerimentos de interfaces de hardware ▪ requerimentos de interfaces de usuário ▪ requerimentos de interface de sistema externa ▪ requerimentos de execução ▪ estruturas de comandos ▪ características de segurança/proteção de dados ▪ colocação de parâmetros de sistema ▪ operações manuais ▪ componentes reusáveis ▪ mapeamento de requerimentos para componentes de sistema
54	2.3	Design de software de alto-nível	<ul style="list-style-type: none"> ▪ descreve a visão geral da estrutura de software ▪ identifica os componentes de software requeridos ▪ identifica a relação entre componentes de software ▪ consideração é dada para:

			<ul style="list-style-type: none"> ▪ quaisquer características de execução de software requeridas ▪ quaisquer interfaces de software requeridas ▪ quaisquer características de segurança requeridas ▪ quaisquer requerimentos de design de banco de dados ▪ quaisquer atributos de manipulação & descoberta de erro requeridos
55	2.3	Design de software de baixo-nível	<ul style="list-style-type: none"> ▪ provê design detalhado (pode ser representado como um protótipo, fluxograma, diagrama de relação de entidade, pseudo-código, etc.) ▪ provê formato de dados de entrada/saída ▪ provê especificação de necessidades de armazenamento de dados ▪ estabelece convenções de nomes de dados requeridas ▪ define o formato de estruturas de dados requeridas ▪ define os campos de dados e propósito para cada elemento de dado requerido ▪ provê as especificações da estrutura de programa
56	2.5	Unidade de software (código)	<ul style="list-style-type: none"> ▪ segue padrões de codificação estabelecidos (como apropriado para a linguagem e a aplicação), o código deve ser/ter: <ul style="list-style-type: none"> ▪ comentado ▪ estruturado ou otimizado ▪ convenção de nomes significativos ▪ informação de parâmetro identificada ▪ códigos de erros definidos ▪ mensagens de erro descritivas e significativa ▪ formatação - indentada, níveis ▪ segue padrões de definição de dados (como apropriado para a linguagem e a aplicação): <ul style="list-style-type: none"> ▪ variáveis definidas ▪ tipo de dados definidos ▪ classes e estruturas de dados herdadas definidas ▪ objetos definidos ▪ relações de entidades definidas ▪ layout de base de dados são definidos ▪ estruturas de arquivos e <i>blocking</i> são definidas ▪ estruturas de dados são eficientes ▪ algoritmos definidos são eficientes ▪ interfaces de funções são definidas
57	2.4	Lista de construídos	<ul style="list-style-type: none"> ▪ identificação de conjuntos do sistema de aplicação de software ▪ identificação de componentes de sistema requeridos (colocação de parâmetros, macro bibliotecas, bases de dados, linguagens de controle de tarefa, etc.) ▪ ordem de seqüência necessária identificada para compilação de versão de software

			<ul style="list-style-type: none"> ▪ bibliotecas fonte de entrada e saída identificadas
58	3.2	Registro/mapeamento de rastreabilidade	<ul style="list-style-type: none"> ▪ identifica requerimentos a serem acompanhados ▪ identifica um mapeamento de requerimento para produtos de trabalho de ciclo de vida ▪ provê a ligação de requerimentos para decomposição de produto de trabalho (i.e., requerimento->design->código->teste->entregáveis, etc.) ▪ provê mapeamento <i>forward</i> e <i>backward</i> de requerimentos para produtos de trabalho associados durante todas as fases do ciclo de vida ▪ <i>Nota:</i> isto pode ser incluído como uma função de um outro produto de trabalho definido (exemplo: uma ferramenta CASE para decomposição de design pode ter uma habilidade de mapeamento como parte de suas características)
59	1.4/ 2.1	Estratégia/plano de teste (todos planos de teste + (16)	<ul style="list-style-type: none"> ▪ identificação de propósito de teste ▪ identificação de proprietário responsável do plano de teste ▪ identificação da abordagem para execução de teste ▪ identificação de componentes a serem testados ▪ identifica conjuntos e seqüência para teste ▪ identifica versão urgente (insistente) ▪ identificação de configuração de sistema requerida (componentes de software, hardware, interface) ▪ identificação dos donos de desenvolvimento associados para componentes a serem testados ▪ identificação de roteiros de teste/casos de teste associados ▪ ordenação de seqüência de como o teste será executado ▪ identificação de requerimentos os quais serão validados pelos testes (i.e., requerimento de cliente, requerimentos regulatórios e requerimentos de sistema) ▪ identificação de mecanismo de reportagem de problema ▪ identificação de ferramentas de teste e recursos requeridos (canais de teste, analisadores, emuladores de teste, etc) ▪ identificação do cronograma de teste ▪ identificação dos critérios de conclusão do teste ▪ identificação de auditorias a serem executadas ▪ bibliotecas-fonte oficiais e versões de software definidas
60	2.3	Roteiro de teste	<ul style="list-style-type: none"> ▪ define o que está sendo testado ▪ define a configuração de sistema requerida para o teste ▪ identifica todos os componentes de software requeridos ▪ identifica iniciações especiais, colocação de parâmetros, etc. ▪ identifica a data de entrada requerida ▪ seqüência a ordem de casos de teste ▪ define os resultados de teste esperados ▪ identifica quais requerimentos estavam reunidos pela execução do

			teste
61	2.3	Caso de teste	<ul style="list-style-type: none"> ▪ provê conjunto executável de instruções de teste ▪ propósito definido ▪ define o resultado de teste esperado ▪ requerimentos mapeados para roteiros de teste
63	1.4/ 2.1	Estratégia/plano de teste unitário + (59)	<ul style="list-style-type: none"> ▪ identifica estratégia para verificação de funcionalidade de unidade (i.e., um programa, um bloco, um módulo, uma rotina) contra os requerimentos e design ▪ especifica como requerimentos de programa básico serão verificados
64	1.4/ 2.1	Estratégia/plano de teste de software + (59)	<ul style="list-style-type: none"> ▪ identifica estratégia para verificação de características de software e/ou funções de operação como definido nos requerimentos
65	1.4/ 2.1	Estratégia/plano de teste de integração + (59)	<ul style="list-style-type: none"> ▪ propósito de integração definido: <ul style="list-style-type: none"> ▪ validação de um subconjunto do sistema (todos programas requeridos para fazer um trabalho de sub-sistema, um trabalho de característica, etc.) ▪ validação da integração do software para outros componentes de sistema (hardware, equipamento de suporte, sistema interfaceado)
66	1.4/ 2.1	Estratégia/plano de teste de sistema + (59)	<ul style="list-style-type: none"> ▪ identifica estratégia para verificação sa integração de componentes de sistema como definido na especificação de arquitetura de sistema ▪ provê cobertura de teste para todos componentes do sistema: <ul style="list-style-type: none"> ▪ software ▪ hardware ▪ interfaces externas ▪ documentação de cliente ▪ atividades de instalação ▪ iniciação ▪ programas de conversão
67	1.4/ 2.1	Estratégia/plano de teste de regressão + (59)	<ul style="list-style-type: none"> ▪ plano para validação de que sistemas/funções-características existentes não tenham sido impactadas por uma mudança ▪ plano para validação de que mudança não teve impacto em componentes de trabalho do sistema (interfaces, operações, etc.) ▪ plano para validação de que a mudança é compatível com os requerimentos de sistema existentes ▪ identificação dos requerimentos para componente de sistema não-mudado ▪ identificação de quais componentes de sistema passarão pelo teste de regressão ▪ identificação das mudanças feitas ▪ identificação dos casos de teste de regressão a serem executados ▪ condições para execução de teste de regressão
68	1.4/	Estratégia/plano de teste de aceitação +	<ul style="list-style-type: none"> ▪ atividade identificadas a serem executadas para <i>test deliverable</i> de

	2.1	(59)	<p>produto de cliente final</p> <ul style="list-style-type: none"> ▪ identifica quem tem responsabilidade para execução de atividades de teste de aceitação (fornecedor ou cliente) ▪ identifica os requerimentos de configuração de sistema para localizar ▪ identifica os requerimentos de instalação de sistema para localizar ▪ provê um plano para validação do <i>software entregue</i> ▪ identifica como validar se atividades de instalação na locação (casa) do cliente foram executadas corretamente ▪ identifica como validar os requerimentos de cliente satisfeitos e entregáveis ▪ identifica roteiros de teste/casos de teste associados ▪ identifica ações a serem tomadas sobre aceitação do produto ▪ refere-se ao plano de qualidade
69	1.4/ 2.1	Estratégia/plano de liberação + (16)	<ul style="list-style-type: none"> ▪ identifica a funcionalidade a ser incluída em cada liberação ▪ identifica os componentes de software associados requeridos (i.e., hardware, software, documentação, etc.) ▪ mapeamento de requisitos e requerimentos de cliente satisfeitos para liberações particulares do produto
70	2.5	Pacote de liberação	<ul style="list-style-type: none"> ▪ inclui o software ▪ inclui elementos de liberação associados, tais como: <ul style="list-style-type: none"> ▪ componentes de software do sistema ▪ hardware requerido ▪ documentação de cliente associada ▪ definições de parâmetros definidas ▪ linguagem de comandos definida ▪ instruções de instalação ▪ carta de liberação
71	2.5	Informação de liberação (notas)	<ul style="list-style-type: none"> ▪ cobertura para elementos-chave (como apropriado para a aplicação): <ul style="list-style-type: none"> ▪ descrição do que é novo ou mudado (incluindo características removidas) ▪ sistema de informação e requerimentos ▪ identificação de programas de conversão e instruções ▪ identificação da lista de componentes (identificação de versão incluída): módulos de software, bibliotecas; lista de documentação associada; requerimentos de hardware associados; ▪ informação de parâmetros e/ou comandos novos/mudados ▪ informação de backup e recuperação ▪ lista de problemas conhecidos e falhas em aberto, informação de advertência, etc. ▪ identificação de procedimentos de verificação e diagnóstico ▪ informação de suporte técnico

			<ul style="list-style-type: none"> ▪ informação de copyright e licença
72	2.6	Software integrado	<ul style="list-style-type: none"> ▪ todos os componentes especificados sobre uma lista construída de software para o conjunto está incluída ▪ conjunto totalmente configurado dos componentes de software <ul style="list-style-type: none"> ▪ parâmetros definidos ▪ comandos definidos ▪ dados carregados ou convertidos
73	2.5	Sistema	<ul style="list-style-type: none"> ▪ todos os componentes de liberação de produto estão incluídos ▪ qualquer hardware requerido ▪ software integrado ▪ documentação de cliente ▪ conjunto totalmente configurado dos <i>componentes do sistema</i> <ul style="list-style-type: none"> ▪ parâmetros definidos ▪ comandos definidos ▪ dados carregados ou convertidos
77	3.2	Lista de distribuição	<ul style="list-style-type: none"> ▪ lista de lista corrente de recebedores e seus endereços de entrega ▪ identificação do mídia esperada para entrega (manual, CD-ROM, email, etc.)
78	2.5	Instruções de entrega	<ul style="list-style-type: none"> ▪ cobertura para elementos-chave (como apropriado para a aplicação): <ul style="list-style-type: none"> ▪ ordenação sequencial de tarefas a serem executadas ▪ se aplicável, liberações identificadas ▪ identificação de todos os componentes entregues com informação de versão ▪ identificação de quaisquer procedimentos de backup e restauração necessários
79	3.2	Registro de entrega	<ul style="list-style-type: none"> ▪ registro de itens entregues eletronicamente para o cliente ▪ identificação de: <ul style="list-style-type: none"> ▪ quem enviou ▪ endereço onde entregou ▪ a data de entrega ▪ registro de recebimento de produto entregue
80	2.5	Guia de manipulação e armazenamento	<ul style="list-style-type: none"> ▪ define as tarefas para executar em produtos de manipulação e armazenamento incluindo: <ul style="list-style-type: none"> ▪ fornecimento de cópias <i>master</i> do código e documentação ▪ recuperação de desastre ▪ endereçamento apropriado de características críticas de segurança ▪ provê uma descrição de como armazenar o produto incluindo: <ul style="list-style-type: none"> ▪ ambiente de armazenamento requerido ▪ o meio de proteção para usar

			<ul style="list-style-type: none"> ▪ materiais de pacote requeridos ▪ quais itens necessitam ser armazenados ▪ avaliações a serem feitas no produto armazenado ▪ prover instruções de recuperação
81	3.2	Registro de aceitação	<ul style="list-style-type: none"> ▪ registro do recibo de entrega ▪ identificação da data recebida ▪ identificação dos componentes entregues ▪ registra a verificação de quaisquer critérios de aceitação de cliente definidos ▪ assinado pelo recebedor
82	1.2	Procedimento de suporte a cliente	<p>Cobertura para elementos-chave (como apropriado ao produto ou contrato):</p> <ul style="list-style-type: none"> ▪ tarefas para seguir em provimento de suporte definido ▪ definir a disponibilidade e a cobertura do suporte provido: <ul style="list-style-type: none"> ▪ # hot-line ▪ horas de disponibilidade ▪ perícia (expertise) apropriada ▪ custo ▪ definir um esquema para classificação de requisito de cliente e/or problemas: <ul style="list-style-type: none"> ▪ definição de tipo de requisito ▪ definição de prioridade/severidade ▪ definição de expectativas de tempo de resposta, pelo tipo e severidade ▪ padrões para quais informações reter a partir do cliente, tais como: <ul style="list-style-type: none"> ▪ companhia e localização ▪ detalhes de informações de contato ▪ descrição do requisito ▪ referência para informação de suporte enviada (dumps, files) ▪ informação de configuração do <i>site</i> de sistema de cliente (produto, liberação (release), versão, last update) ▪ sistema(s) impactado(s) ▪ impacto para operações de sistemas existentes ▪ o quanto é crítico (criticality) o requisito ▪ requerimentos de resposta / fechamento de cliente esperados ▪ definição de procedimentos de escalação de cliente ▪ identificação de <i>ferramentas de suporte de cliente</i> disponíveis e procedimentos para usá-las, tais como: <ul style="list-style-type: none"> ▪ mecanismo usado para registrar requisitos de cliente ▪ relatórios de status

			<ul style="list-style-type: none"> ▪ sistemas disponíveis para reproduzir problemas ▪ habilidade para reproduzir ambiente de software de clientes ▪ habilidade para reproduzir problemas ▪ emuladores de apoio ▪ roteiros (<i>scripts</i>) de apoio ▪ <i>dial-in ports</i> ▪ ferramentas para análise de <i>dumps</i>
83	3.2	Registro de requisição de cliente (externo ou interno)	<ul style="list-style-type: none"> ▪ identifica propósito de requisito, tais como: <ul style="list-style-type: none"> ▪ novo desenvolvimento ▪ melhoria ▪ cliente interno ▪ operações ▪ documentação ▪ informacional ▪ identifica informação de status de requisito, tais como: <ul style="list-style-type: none"> ▪ data abertura ▪ status corrente ▪ data assinalada e dono responsável ▪ data verificada ▪ data fechada ▪ identifica prioridade/severidade de requisito ▪ identifica informação de cliente, tal como: <ul style="list-style-type: none"> ▪ companhia/pessoa iniciando o requisito ▪ informação de contato e detalhes ▪ informação de configuração de localização de sistema ▪ sistema(s) impactado(s) ▪ impacto para operações de sistemas existentes ▪ grau de quanto o requisito é crítico ▪ requerimentos de resposta/encerramento de cliente esperados ▪ identifica requerimentos ou padrões necessitados ▪ identifica informação enviada com requisito (i.e., RFPs, dumps, etc.)
84	3.2	Registro de relatório de problema	<ul style="list-style-type: none"> ▪ identifica o nome e detalhes para contato de quem identificou o problema ▪ identifica informação de configuração de sistema (tais como: versão de liberação, software de sistema, configuração de hardware, etc.) ▪ identifica o grupo / pessoa(s) responsável (is) para prover(em) um conserto ▪ inclui uma descrição do problema ▪ identifica quaisquer informações de suporte associadas (dumps, arquivos, etc)

			<ul style="list-style-type: none"> ▪ identifica a severidade do problema (crítico, maior, menor, ...) ▪ identifica o status do problema reportado ▪ identifica os componentes do produto afetado ▪ identifica a informação de liberação de produto de software e versão aplicável ▪ identifica a data de <i>abertura</i> ▪ identifica o problema da liberação alvo será consertada em ▪ identifica a data de fechamento esperada ▪ identifica quaisquer relatórios de problema associados, requisitos de cliente, problemas duplicados e consertos associados ▪ identifica quaisquer critérios de fechamento ▪ identifica ações de re-inspeção
86	3.4	Dados de satisfação de cliente	<ul style="list-style-type: none"> ▪ determina níveis de satisfação de cliente com produtos de software e serviços ▪ mecanismo para coletar dados sobre satisfação de cliente: <ul style="list-style-type: none"> ▪ resultados de dados de execução em campo ▪ resultados de pesquisa de satisfação de cliente ▪ notas de entrevista ▪ atas de reunião de reuniões de cliente
87	1.2	Mecanismo de comunicação	<p>Uma maneira para distribuir informação:</p> <ul style="list-style-type: none"> ▪ descrição clara do que está sendo comunicado ▪ habilidade para especificar informação de data enviada ▪ habilidade para distribuir para todos os impactados ▪ identificação do impacto: hardware, desenvolvimento, cliente, organização, etc.) ▪ provê uma identificação clara de/para que/quem a mensagem se aplica ▪ mecanismo de destinatário para responder quando requerido (retorno de informação) ▪ o meio de distribuição usado é acessível para todos com a necessidade para conhecer ▪ a lista de distribuição é corrente e inclui todos com a necessidade para conhecer ▪ habilidade para especificar informação de data de retorno alvo
91	1.4/ 2.1	Estratégia/plano de gerência de configuração + (16)	<ul style="list-style-type: none"> ▪ define ou referencia os procedimentos para controle de mudanças para itens de configuração ▪ define medidas usadas para determinar o status das atividades de gerência de configuração ▪ define critérios de auditoria de gerência de configuração ▪ deve estar aprovado pelo gerente de configuração ▪ identifica ferramentas de biblioteca ou mecanismo de configuração ▪ inclui registros de gerenciamento e relatórios de status que apresentam o status e o histórico dos itens de software controlados

			<ul style="list-style-type: none"> ▪ especificar a localização e mecanismo de acesso para a biblioteca de gerência de configuração ▪ mecanismo arquivável e recuperável especificado
92	2.4	Gerência de configuração (arquivo, biblioteca, sistema)	<ul style="list-style-type: none"> ▪ controle de versão, armazenamento e recuperação de itens de configuração (e suas versões) ▪ compartilhamento e transferência de itens de configuração entre grupos afetados ▪ controle efetivo sobre acesso ▪ mantém descrições de item de configuração ▪ recuperação de versões de arquivo de itens de configuração ▪ criação correta de produtos a partir da biblioteca ▪ pode re-criar quaisquer configurações de liberação ou teste ▪ habilidade para relatar status de configuração ▪ muda para itens de configuração a serem acompanhados para requisitos de mudança/usuário
93	2.5	Item de configuração	<ul style="list-style-type: none"> ▪ item que é mantido sob controle de configuração (software, documentos, produtos de trabalho) ▪ identificação de versão é mantida ▪ descrição do item está disponível incluindo coisas como: <ul style="list-style-type: none"> ▪ tipo do item ▪ biblioteca, arquivo, sistema de gerência de configuração associado ▪ dono responsável ▪ data quando colocado sob controle de configuração ▪ informação de status (i.e., desenvolvimento, "é linha básica", liberado) ▪ relação para itens de configuração de mais baixo nível ▪ identificação de registro de controle de mudança ▪ identificação de história de mudança
94	3.2	Pedido de mudança	<ul style="list-style-type: none"> ▪ identifica propósito de mudança ▪ identifica status de requisito (novo, aceito, rejeitado) ▪ identifica informação de contato de requisitante ▪ sistema(s) impactado(s) ▪ impacto definido para operação de sistema(s) existente(s) ▪ impacto definido para documentação associada ▪ criticidade do requisito, dada necessitada para
95	3.2	Registro de controle de mudança	<ul style="list-style-type: none"> ▪ usado como um mecanismo para controlar mudança para produtos que são linha básica / produtos em bibliotecas de liberação de projeto oficial ▪ registra a mudança pedida e faz uma linha básica de produto (produtos de trabalho, software, documentação de cliente, etc) ▪ identificação de sistema e documentos impactados com a mudança ▪ identificação do requisitante da mudança

			<ul style="list-style-type: none"> ▪ identificação de parte responsável pela mudança ▪ identificação de status da mudança ▪ ligação para requisitos de cliente associados, requisitos de mudança internos, etc. ▪ aprovações apropriadas
96	2.5	História de mudança	<ul style="list-style-type: none"> ▪ registros históricos de todas as mudanças feitas para um objeto (documento, arquivo, módulo de software, etc.) <ul style="list-style-type: none"> ▪ descrição de mudança ▪ informação de versão sobre objeto mudado ▪ data de mudança ▪ informação de requisitante de mudança ▪ informação de registro de controle de mudança
97	3.2	Ação corretiva (logs, planos, atas)	<ul style="list-style-type: none"> ▪ identifica o problema inicial ▪ identifica o dono para realização de ação definida ▪ define uma solução (série de ações para consertar o problema) ▪ identifica a data de abertura e data alvo de conclusão ▪ contém um indicador de status ▪ indica continuação de ações de auditoria
98	2.5	Sistema de acompanhamento	<ul style="list-style-type: none"> ▪ habilidade para registrar informação de dono de processo e cliente ▪ habilidade para registrar informação de configuração de sistema relacionada ▪ habilidade para registrar informação sobre problema ou ação necessários <ul style="list-style-type: none"> ▪ data de abertura e data prevista de fechamento ▪ severidade/criticalidade de item ▪ status de quaisquer problemas ou ações necessárias ▪ informação sobre o dono do problema ou ação ▪ prioridade de resolução de problema ▪ habilidade para registrar resolução ou plano de ação propostos ▪ habilidade para prover informação de status de gerenciamento ▪ informação está disponível para todos com a necessidade de conhecer ▪ sistema(s)/registros de controle de mudança integrados
99	2.5	Work-around (soluções temporárias)	<ul style="list-style-type: none"> ▪ identificação de problema ▪ informação de liberação e de sistema ▪ solução temporária, data prevista para conserto atual identificado ▪ descrição da solução ▪ limitações e restrições sobre uso ▪ requerimentos de operação adicionais ▪ procedimentos especiais ▪ liberações aplicáveis

			<ul style="list-style-type: none"> ▪ backup/recuperação de informação ▪ procedimentos de verificação ▪ instruções de instalação temporária
100	2.5	Configuração de produto	<ul style="list-style-type: none"> ▪ visão geral da configuração do sistema ▪ define cada componente e sua posição na arquitetura do sistema ▪ define as interfaces-chave de sistema ▪ define quaisquer considerações de rede ▪ define a configuração de hardware ▪ define qualquer execução do sistema/colocações de parâmetros
101	2.3	Design de base de dados	<p>Cobertura para elementos-chave (como apropriado para a aplicação):</p> <ul style="list-style-type: none"> ▪ definição de características de design: <ul style="list-style-type: none"> ▪ sistema gerenciador de base de dados usado ▪ tipo de sistema (relacional, hierárquico, orientado a objeto, networked) ▪ formato de registros, tabelas e objetos ▪ modo de acesso a base de dados ▪ software associado (programas, formatos de tela de usuário, relatórios) ▪ linguagem de base de dados suportada ▪ definição de divisões lógica e física, modelos: <ul style="list-style-type: none"> ▪ registros (layout de dados, campos, tabelas, estruturas) ▪ nomes de campos e definições ▪ definições de dados, classes, estrutura, etc. ▪ entidade/relacionamento ▪ classes, esquema de herança ▪ definição de visão de usuário <ul style="list-style-type: none"> ▪ layouts de tela ▪ acesso de campo ▪ acesso de dados ▪ comandos ▪ considerações de interface de entrada/saída ▪ informação de uso de base de dados (conteúdo, sistemas de aplicação, restrições de uso, etc.) ▪ identificação de restrições: <ul style="list-style-type: none"> ▪ considerações de segurança ▪ considerações de acesso a dados ▪ backup e recuperação de considerações ▪ considerações de reinício de sistema ▪ considerações de gerações de sistema ▪ considerações de execução

102	3.2	Registro de backup/ recuperação	<ul style="list-style-type: none"> ▪ data de backup ▪ listagem do que foi backed-up com versões associadas ▪ listagem de onde foi backed-up para ▪ identificação de atributos e configuração de sistema associados no tempo do back-up ▪ identificação de procedimentos de recuperação associados
103	1.4 / 2.1	Estratégia / plano de recuperação	<ul style="list-style-type: none"> ▪ identifica o que deve ser recuperado: ▪ procedimentos / métodos para executar recuperação: <ul style="list-style-type: none"> ▪ cronograma para recuperação ▪ tempo requerido para recuperação ▪ dependências críticas ▪ recursos requeridos para recuperação ▪ lista de backups mantidos ▪ pessoal responsável para recuperação e papéis designados ▪ materiais especiais requeridos ▪ produtos de trabalho requeridos ▪ equipamento requerido ▪ documentação requerida ▪ posições e armazenamento de backups ▪ informação de contato sobre quem notificar sobre a recuperação ▪ procedimentos de verificação ▪ estimativa de custo para recuperação
104	2.5	Ambiente de desenvolvimento	<ul style="list-style-type: none"> ▪ <i>floor plan</i> ▪ considerações de segurança (safety) de ambiente ▪ requerimentos regulatórios ▪ requerimentos contratuais ▪ considerações de segurança (security) ▪ requerimentos de ambiente especiais (e.g. ar condicionado, raised floor, power, etc.) ▪ espaço de trabalho individual necessário definido ▪ requerimento de estações de trabalho ▪ software de suporte ▪ ferramentas ▪ equipamento de comunicação ▪ plano de recuperação de desastre
105	2.5	Documentação de cliente	<ul style="list-style-type: none"> ▪ cobertura para elementos-chave: manual de usuário, manual de operação, manual de manutenção, etc. como apropriado para a aplicação. ▪ documentos de sistema externos: <ul style="list-style-type: none"> ▪ guia de visão geral de sistema, arquitetura e design

			<ul style="list-style-type: none"> ▪ guia de características (descrição funcional de componentes do sistema) ▪ guia de diagnóstico: mensagens de erro e código ▪ guia de referência de comandos de operação ▪ guia de instalação, operações e manutenção ▪ guia de suporte técnico ▪ documentos de cliente interno <ul style="list-style-type: none"> ▪ requerimentos designs ▪ planos de teste ▪ planos ▪ registros ▪ documentação mantida sincronizada com liberação mais recente de software associada: <ul style="list-style-type: none"> ▪ disponível com entrega de uma versão nova ou mudada do software ▪ atualizada com manutenção de liberações (como apropriado para resolução de requisito de mudança) ▪ ordenação de procedimentos ▪ distribuição da localização corrente e lista de manutenção mantida
107	2.5	Componente de sistema	<ul style="list-style-type: none"> ▪ componentes de hardware ▪ componentes de software ▪ componentes manuais ▪ documentação de cliente ▪ materiais de treinamento
108	3.2	Registro de pessoal	<ul style="list-style-type: none"> ▪ informação relevante sobre pessoa incluindo: <ul style="list-style-type: none"> ▪ nome, endereço, data de nascimento, estado civil ▪ <i>grade, pay, appraisal history</i> ▪ histórico disciplinar
109	3.2	Registro de revisão de contrato + (31)	<ul style="list-style-type: none"> ▪ escopo do contrato e requerimentos ▪ contingências ou riscos possíveis ▪ alinhamento do contrato com o plano estratégico de negócios da organização ▪ proteção de informação patenteada ▪ requerimentos que diferem daqueles na documentação original ▪ capacidade para reunir requerimentos contratuais ▪ responsabilidade para trabalho subcontratado ▪ terminologia ▪ habilidade de cliente para reunir obrigações contratuais
201		Esboço de requerimento de produto	<ul style="list-style-type: none"> ▪ escopo do problema ▪ percepção global de uma solução

202		Lista de perspectivas	<ul style="list-style-type: none"> ▪ objetos que fazem parte do ambiente e circundam o sistema ▪ objetos que são produzidos pelo sistema ▪ objetos que são usados pelo sistema para que execute suas funções ▪ operações (processos ou funções) que manipulam ou interagem com o objeto ▪ restrições (custo, tempo, tamanho, etc.) ▪ critérios de desempenho (velocidade, precisão, etc.) ▪ critérios de validação.
203		Lista de linhas básicas	<p>Para cada tipo de linha básica tem-se:</p> <ul style="list-style-type: none"> ▪ tipo linha básica (funcional, alocada, configuração de desenvolvimento ou produto) ▪ conteúdo (produtos de trabalho que compõem a linha básica) ▪ quando a linha básica é estabelecida

Anexo II

Descrição de escopo dos seguintes processos:

- *ENG.1.1 Processo de análise de requerimentos e design de sistema;*
- *ENG.1.2 Processo de análise de requerimentos de software;*
- *ENG.1.3 Processo de design de software;*
- *ENG.1.4 Processo de construção de software;*
- *ENG.1.5 Processo de integração de software;*
- *ENG.1.6 Processo de teste de software;*
- *ENG.1.7 Processo de integração de sistema e teste;*
- *ENG.2 Processo de manutenção de sistema e software;*
- *SUP.1 Processo de documentação;*
- *SUP.3 Processo de garantia de qualidade;*
- *SUP.4 Processo de verificação;*
- *SUP.5 Processo de validação;*
- *ORG.4 Processo de infra-estrutura.*

ENG.1.1 Processo de análise de requerimentos e design de sistema

Nome do processo:		ENG.1.1 Processo de análise de requerimentos e design de sistema	
Objetivo:	O propósito deste processo é estabelecer os requerimentos (funcionais e não-funcionais) e a arquitetura do sistema, identificando quais requerimentos de sistema seriam alocados para quais elementos do sistema e quais liberações.		
Entradas:	<p>44) Avaliação de necessidades de produto</p> <p>46) Análise de mercado</p> <p>52) Especificação de requerimentos (cliente)</p> <p>52) Especificação de requerimentos (manutenção)</p> <p>83) Requisito de cliente</p> <p>94) Requisito de mudança</p> <p>98) Sistema de rastreamento</p>	Saídas:	<p>8) Interface</p> <p>30) Estratégia/plano de revisão</p> <p>52) Especificação de requerimentos (sistema)</p> <p>53) Design/arquitetura de sistema</p> <p>58) Registro/mapeamento de rastreabilidade</p> <p>68) Estratégia/plano de teste de aceitação</p> <p>69) Estratégia ou plano de liberação</p> <p>87) Mecanismo de comunicação</p>
Início do processo:			
Conteúdo:	Identifica requerimentos de sistema; analisa requerimentos de sistema; descreve arquitetura de sistema; aloca requerimentos; desenvolve estratégia de liberação; comunica requerimentos de sistema; estabelece rastreabilidade.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • requerimentos de sistema estarão desenvolvidos conforme as necessidades declaradas do cliente; • uma solução estará proposta a qual identifica os elementos principais do sistema; • os requerimentos estarão alocados para cada um dos principais elementos do sistema; • uma estratégia de liberação estará desenvolvida a qual define a prioridade para implementação de requerimentos de sistema; • os requerimentos de sistema estarão aprovados e atualizados conforme necessidade; • os requerimentos, a solução proposta e suas relações estarão comunicados para todas as partes afetadas. 		
Clientela a ser atendida:			
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: ENG.1.1.BP1, ENG.1.1.BP2, ENG.1.1.BP3, ENG.1.1.BP4, ENG.1.1.BP5, ENG.1.1.BP6, ENG.1.1.BP7.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>		

ENG.1.2 Processo de análise de requerimentos de software

Nome do processo:		ENG.1.2 Processo de análise de requerimentos de software	
Objetivo:	O propósito deste processo é estabelecer os requerimentos dos componentes de software do sistema.		
Entradas:	17) Plano de projeto 44) Avaliação de necessidade de produto 52) Especificação de requerimento (cliente) 52) Especificação de requerimento (manutenção) 53) Design/arquitetura de sistema 58) Registro/mapeamento de rastreabilidade 83) Pedido de cliente 87) Mecanismo de comunicação 84) Relatório de problema 94) Pedido de mudança	Saídas:	8) Interface 21) Resultado de análise 31) Registro de revisão 52) Especificação de requerimento (software) 58) Registro/mapeamento de rastreabilidade 69) Estratégia ou plano de liberação 87) Mecanismo de comunicação 93) Item de configuração 100) Configuração de produto 101) Design de banco de dados 105) Documentação de cliente
Início do processo:	Especificar requerimentos de software.		
Conteúdo:	Especifica requerimentos de software; determina impacto de ambiente de operação; avalia e valida requerimentos com cliente; desenvolve critério de validação de software; desenvolve estratégia de liberação; atualiza requerimentos; comunica requerimentos de software; avalia requerimentos de software.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> os requerimentos alocados para os componentes de software do sistema e suas interfaces estarão definidos conforme as necessidades declaradas do cliente; requerimentos de software analisados, corretos e testáveis estarão desenvolvidos; impacto de componentes de software sobre o ambiente de operação estará entendido; uma estratégia de liberação estará desenvolvida, a qual define a prioridade para implementação de requerimentos de software; os requerimentos de software estarão aprovados e atualizados conforme necessidade; consistência estará estabelecida entre requerimentos de sistema, design de sistema e requerimentos de software; os requerimentos de software estarão comunicados para todas as partes afetadas. 		
Clientela a ser atendida:	Cliente, arquitetos de software, desenvolvedores, garantia de qualidade/testadores.		
Indicadores de desempenho:	Existência e adequação das práticas base aplicadas neste processo: ENG.1.2.BP1, ENG.1.2.BP2, ENG.1.2.BP3, ENG.1.2.BP4, ENG.1.2.BP5, ENG.1.2.BP6, ENG.1.2.BP7, ENG.1.2.BP8.		

Existência dos produtos de trabalho de entrada e saída e suas características.

ENG.1.3 Processo de design de software

Nome do processo:	ENG.1.3 Processo de design de software		
Objetivo:	O propósito deste processo é definir um design para o software que implementa os requerimentos e pode ser testado contra eles.		
Entradas:	2) Modelo de ciclo de vida 33) Estratégia /plano de reuso 52) Especificação de requerimento (software) 53) Design/arquitetura de sistema 59) Estratégia/plano de teste	Saídas:	54) Design de software de alto-nível 55) Design de software de baixo-nível 58) Registro/mapeamento de rastreabilidade 63) Estratégia/plano de teste de unidade 101) Design de banco de dados 105) Documentação de cliente
Início do processo:	Desenvolver design da arquitetura do software		
Conteúdo:	Desenvolve design da arquitetura do software e das interfaces externas e internas; verifica o design do software; desenvolve o design detalhado e estabelece rastreabilidade entre requerimentos de software e o design de software.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • um design arquitetural será desenvolvido o qual descreve os componentes de software maiores que implementarão os requerimentos de software; • interfaces internas e externas de cada componente de software serão definidas; • um design detalhado será desenvolvido o qual descreve unidades de software que podem ser construídas e testadas; • consistência será estabelecida entre requerimentos de software e designs de software. 		
Cientela a ser atendida:			
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: ENG.1.3.BP1, ENG.1.3.BP2, ENG.1.3.BP3, ENG.1.3.BP4.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>		

ENG.1.4 Processo de construção de software

Nome do processo:		ENG.1.4 Processo de construção de software	
Objetivo:	<p>O propósito deste processo é produzir unidades de software executáveis e verificar que elas refletem bem o design de software.</p> <p>Nota: Parte deste processo é similar ao processo <i>Processo de Verificação</i> (SUP.4).</p>		
Entradas:	<p>10) Padrão de codificação</p> <p>35) Repositório de reuso</p> <p>52) Especificação de requerimentos (software)</p> <p>52) Especificação de requerimentos (sistema)</p> <p>53) Design/arquitetura de sistema</p> <p>54) Design de software de alto-nível</p> <p>55) Design de software de baixo-nível</p> <p>63) Estratégia/plano de teste de unidade</p> <p>91) Estratégia/plano de gerência de configuração</p> <p>101) Design de banco de dados</p>	Saídas:	<p>56) Unidades de software (código)</p> <p>59) Estratégia/plano de teste</p> <p>60) Roteiro de teste de unidade</p> <p>61) Caso de teste</p> <p>62) Resultado de teste</p> <p>63) Estratégia/plano de teste de unidade</p> <p>93) Item de configuração</p>
Início do processo:	Desenvolver unidades de software.		
Conteúdo:	Desenvolve unidades de software e procedimentos de verificação de unidade; verifica as unidades de software; estabelece rastreabilidade.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • critérios de verificação serão definidos para todas as unidades de software contra seus requerimentos; • unidades de software definidas pelo design serão produzidas; • unidades de software definidas pelo design serão produzidas; • verificação das unidades de software contra o design será realizada. 		
Clientela a ser atendida:			
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: ENG.1.4.BP1, ENG.1.4.BP2, ENG.1.4.BP3, ENG.1.4.BP4.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>		

ENG.1.5 Processo de integração de software

Nome do processo:		ENG.1.5 Processo de integração de software	
Objetivo:	O propósito deste processo é combinar as unidades de software, produzindo itens de software integrados e verificar que as unidades de software integradas refletem bem o design de software. Nota: Parte deste processo é similar ao processo <i>Processo de Verificação</i> (SUP.4).		
Entradas:	52) Especificação de requerimentos (sistema) 52) Especificação de requerimentos (software) 52) Especificação de requerimentos (manutenção) 53) Design/arquitetura de sistema 54) Design de software de alto-nível 55) Design de software de baixo-nível 56) Unidades de software (código) 57) Lista de "conjuntos"construídos 60) Roteiro de teste de unidade 61) Casos de teste 69) Estratégia/plano de liberação 95) Registro de controle de mudança	Saídas:	65) Estratégia/plano de teste de integração 67) Estratégia de teste de regressão 58) Registro/mapeamento de rastreabilidade 57) Lista de "conjuntos" construídos 64) Plano de teste de item de software 60) Roteiro de teste de item de software 61) Casos de teste 62) Resultado de teste 72) Software integrado
Início do processo:			
Conteúdo:	Desenvolve estratégia de integração de software; desenvolve estratégia de teste de regressão de item de software integrado; desenvolve testes para itens de software integrados; testa itens de software integrados; integra itens de software integrados; realiza teste de regressão de itens de software integrados.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> • uma estratégia de integração será desenvolvida para unidades de software consistente com a estratégia de liberação; • critérios de verificação para itens de software serão desenvolvidos os quais garantem conformidade com os requerimentos de software alocados aos itens; • itens de software definidos pela estratégia de integração serão produzidos; • itens de software serão verificados usando os critérios de aceite definidos; • resultados de teste de integração serão registrados; • consistência será estabelecida entre requerimentos de software e itens de software; • uma estratégia de teste de regressão será desenvolvida para re-verificação de itens de software, caso ocorra uma mudança nas unidades de software; • teste de regressão será executado quando necessário. 		
Clientela a ser atendida:			
Indicadores de desempenho:		Existência e adequação das práticas base aplicadas neste processo: ENG.1.5.BP1, ENG.1.5.BP2, ENG.1.5.BP3, ENG.1.5.BP4, ENG.1.5.BP5,	

	ENG.1.5.BP6. Existência dos produtos de trabalho de entrada e saída e suas características.
--	--

ENG.1.6 Processo de teste de software

Nome do processo: ENG.1.6 Processo de teste de software	
Objetivo:	O propósito deste processo é testar o software integrado, produzindo um produto que satisfará os requerimentos de software.
Entradas:	<p>52) Especificação de requerimentos (sistema)</p> <p>52) Especificação de requerimentos (software)</p> <p>52) Especificação de requerimentos (manutenção)</p> <p>53) Design/arquitetura de sistema</p> <p>54) Design de software de alto-nível</p> <p>55) Design de software de baixo-nível</p> <p>57) Lista de "conjuntos" construídos</p> <p>60) Roteiro de teste</p> <p>61) Casos de teste</p> <p>64) Plano de teste de item de software</p> <p>67) Estratégia de teste de regressão</p> <p>69) Estratégia/plano de liberação</p> <p>72) Software integrado</p> <p>92) Gerência de configuração (arquivo, biblioteca, sistema)</p> <p>95) Registro de controle de mudança</p> <p>105) Documentação de cliente</p>
Saídas:	<p>58) Registro/mapeamento de rastreabilidade</p> <p>60) Roteiro de teste de software integrado</p> <p>61) Casos de teste</p> <p>62) Resultado de teste</p> <p>65) Estratégia/plano de teste de integração</p> <p>67) Estratégia de teste de regressão</p> <p>72) Software integrado</p> <p>100) Configuração de produto</p> <p>105) Documentação de cliente</p>
Início do processo:	
Conteúdo:	Desenvolve estratégia de teste de software integrado, incluindo estratégia de regressão; desenvolve testes para software integrado; testa software integrado; realiza teste de regressão de software integrado.
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • critérios de aceitação para software integrado serão desenvolvidos, os quais verificam a conformidade com os requerimentos de software; • software integrado será verificado usando critérios de aceitação definidos; • resultados de teste estarão registrados; • uma estratégia de teste de regressão será desenvolvida para reteste do software integrado, caso ocorra uma mudança nos itens de software; • teste de regressão será executado quando necessário.
Clientela a ser atendida:	
Indicadores de desempenho:	Existência e adequação das práticas base aplicadas neste processo: ENG.1.6.BP1, ID ENG.1.6Proc.BP2, ENG.1.6.BP3, ENG.1.6.BP4.

Existência dos produtos de trabalho de entrada e saída e suas características.
--

ENG.1.7 Processo de integração de sistema e teste

Nome do processo:		ENG.1.7 Processo de integração de sistema e teste	
Objetivo:	<p>O propósito deste processo é integrar o componente software com outros componentes, tais como operações manuais ou hardware, produzindo um sistema completo que satisfará as expectativas do cliente expressadas nos requerimentos de sistema. Os recursos alocados para integração de sistema incluiriam alguma familiaridade com o componente software.</p> <p>Nota: Parte deste processo é similar aos processos <i>Processo de Verificação (SUP.4)</i> e <i>Processo de Validação (SUP.5)</i>.</p>		
Entradas:	<p>52) Especificação de requerimentos (sistema)</p> <p>52) Especificação de requerimentos (software)</p> <p>52) Especificação de requerimentos (manutenção)</p> <p>53) Design/arquitetura de sistema</p> <p>54) Design de software de alto-nível</p> <p>60) Roteiro de teste</p> <p>61) Casos de teste</p> <p>64) Estratégia/plano de teste de software</p> <p>66) Estratégia/plano de teste de sistema</p> <p>67) Estratégia/plano de teste de regressão</p> <p>69) Estratégia/plano de liberação</p> <p>72) Software integrado</p> <p>92) Gerência de configuração (arquivo, biblioteca, sistema)</p> <p>95) Registro de controle de mudança</p> <p>105) Documentação de cliente</p> <p>107) Componentes de sistema</p>	Saídas:	<p>16) Estratégia/plano de integração de sistema</p> <p>57) Lista de "conjuntos" construídos</p> <p>58) Registro/mapeamento de rastreabilidade</p> <p>66) Plano de teste de integração de sistema</p> <p>60) Roteiro de teste de integração de sistema</p> <p>61) Casos de teste</p> <p>62) Resultados de teste</p> <p>66) Plano de teste de sistema</p> <p>67) Estratégia de teste de regressão</p> <p>60) Roteiro de teste de sistema</p> <p>73) Sistema</p> <p>100) Configuração de produto</p> <p>105) Documentação de cliente</p>
Início do processo:			
Conteúdo:	<p>Desenvolve estratégia de integração e teste de sistema; desenvolve estratégia de teste de regressão de sistema; constrói conjuntos de unidades de sistema; desenvolve testes para conjuntos de sistema; testa conjuntos de sistema; desenvolve testes para sistema; testa sistema integrado; realiza teste de regressão de conjuntos de sistema ou sistema integrado.</p>		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • uma estratégia de integração será desenvolvida para construir conjuntos de unidade de sistema de acordo com a estratégia de liberação; • critérios de aceitação para cada conjunto serão desenvolvidos para verificar conformidade com os requerimentos de sistema alocados para as unidade; • conjuntos de sistema serão verificados usando os critérios de aceitação 		

	<p>definidos; um sistema integrado, demonstrando conformidade com os requerimentos (funcionais, não funcionais, de operação e manutenção) e validação que um completo conjunto de componentes entregáveis e usáveis existe, será construído;</p> <ul style="list-style-type: none"> • resultados de teste serão registrados; • uma estratégia de teste de regressão será desenvolvida para reteste conjuntos ou sistema integrado, caso ocorra uma mudança nos componentes existentes; • teste de regressão será executado quando necessário.
<i>Cientela a ser atendida:</i>	
<i>Indicadores de desempenho:</i>	<p>Existência e adequação das práticas base aplicadas neste processo: ENG1.7.BP1, ENG1.7.BP2, ENG1.7.BP3, ENG1.7.BP4, ENG1.7.BP5, ENG1.7.BP6, ENG1.7.BP7, ENG1.7.BP8.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>

ENG.2 Processo de manutenção de sistema e de software

Nome do processo:		ENG.2 Processo de manutenção de sistema e software	
Objetivo:	<p>O propósito deste processo é gerenciar modificação, migração e retirada de componentes de sistema (tais como hardware, software, operações manuais e rede, se houver) em resposta a requisitos de cliente. A origem dos requisitos pode ser um problema descoberto ou uma necessidade para melhoria ou adaptação. O objetivo é modificar e/ou retirar sistemas e/ou software existentes enquanto preservando a integridade de operações organizacionais.</p> <p>Nota1: Os requerimentos iniciais definidos podem ser providos pelo <i>Processo de elicitação de requerimentos (CUS.3)</i>.</p> <p>Nota2: Este processo interage com outros processos tais como <i>Processo de Operação (CUS.4)</i>, <i>Processo de Suporte a Cliente (CUS.4.2)</i> e <i>Processo de Resolução de problemas (SUP.8)</i>.</p>		
Entradas:	<p>17) Plano de projeto</p> <p>21) Resultado de análise</p> <p>33) Estratégia/plano de reuso</p> <p>34) Estratégia de teste</p> <p>41) Medida de campo</p> <p>44) Avaliação de necessidade de produto</p> <p>51) Contrato</p> <p>52) Especificação de requerimentos (cliente)</p> <p>53) Design/arquitetura de sistema</p> <p>67) Estratégia de teste de regressão</p> <p>69) Estratégia/plano de liberação</p> <p>80) Guia de manipulação e armazenamento</p> <p>83) Requisito de cliente</p> <p>84) Relatório de problema</p> <p>92) Gerência de configuração (arquivo, biblioteca, sistema)</p> <p>94) Requisito de mudança</p> <p>95) Controle de mudança</p> <p>99) "Work-around"(soluções temporárias)</p> <p>102) Registro de Back-up/Recuperação</p> <p>107) Componentes de sistema</p>	Saídas:	<p>16) Estratégia/plano de manutenção</p> <p>52) Especificação de requerimento (manutenção)</p> <p>21) Resultado de análise</p> <p>52) Especificação de requerimentos (sistema)</p> <p>52) Especificação de requerimentos (software)</p> <p>95) Controle de mudança</p> <p>69) Estratégia/plano de liberação</p> <p>70) Pacote de liberação</p> <p>71) Informação de liberação</p>
Início do processo:			
Conteúdo:	<p>Determina os requerimentos de manutenção; desenvolve estratégia de manutenção; analisa problemas de usuários e melhorias; determina modificações para a próxima atualização; implementa e testa modificações; atualiza sistema de usuário; retira sistema de usuário.</p>		
Término do processo:		<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • uma estratégia de manutenção será desenvolvida para gerenciar modificação, migração e retirada de componentes de software de acordo 	

	<p>com a estratégia de liberação;</p> <ul style="list-style-type: none"> • o impacto de organização, operações e interfaces sobre o sistema existente em operação será definido; • documentos de especificações e design e estratégias de teste serão atualizados; • componentes de sistema modificados serão desenvolvidos com testes associados que demonstram que os requerimentos de sistema não estarão comprometidos; • atualizações de sistema e software serão migradas para o ambiente do cliente; • sobre requisitos, sistema e software serão reformados a partir do uso em uma maneira controlada que minimiza transtornos para os clientes.
<i>Clientela a ser atendida:</i>	
<i>Indicadores de desempenho:</i>	<p>Existência e adequação das práticas base aplicadas neste processo: ENG.2.BP1, ENG.2.BP2, ENG.2.BP3, ENG.2.BP4, ENG.2.BP5, ENG.2.BP6, ENG.2.BP7.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>

SUP.1 Processo de Documentação

Nome do processo:		SUP.1 Processo de documentação	
Objetivo:	O propósito deste processo é desenvolver e manter documentos que registram informações produzidas por um processo ou atividade.		
Entradas:	9) Padrão 27) Critério de qualidade 30) Plano de revisão 44) Avaliação de necessidade de produto 52) Especificação de requerimento (cliente) 52) Especificação de requerimento (documentação) 53) Arquitetura/design de sistema 77) Lista de distribuição 78) Instrução de entrega 83) Requisito de cliente 84) Relatório de problema 92) Sistema de gerência de configuração 94) Requisito de mudança	Saídas:	7) Produto de trabalho 14) Política de documentação 17) Plano de projeto (documentação) 31) Registro de revisão 52) Especificação de requerimento (documentação) 79) Registro de entrega 81) Registro de aceitação 93) Item de configuração 95) Controle de mudança 96) História de mudança
Início do processo:	Desenvolver política de documentação.		
Conteúdo:	Desenvolve política de documentação; estabelece padrões de documentos; especifica requerimentos de documentação; desenvolve documento; verifica documento; distribui documento; mantém documento.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> • uma estratégia identificando os documentos a serem produzidos durante o ciclo de vida do produto de software será desenvolvida; • os padrões a serem aplicados para o desenvolvimento de documentos será identificado; • todos os documentos a serem produzidos pelo processo ou projeto estarão identificados • o conteúdo e propósito de todos os documentos estarão especificados, revisados e aprovados; • todos documentos estarão desenvolvidos e publicados de acordo com os padrões identificados; • todos os documentos estarão mantidos de acordo com critérios especificados. 		
Clientela a ser atendida:			
Indicadores de desempenho:	Existência e adequação das práticas base aplicadas neste processo: SUP.1.BP1, SUP.1.BP2, SUP.1.BP3, SUP.1.BP4, SUP.1.BP5, SUP.1.BP6, SUP.1.BP7. Existência dos produtos de trabalho de entrada e saída e suas		

características.

SUP.3 Processo de garantia de qualidade

Nome do processo:	SUP.3 Processo de garantia de qualidade		
Objetivo:	O propósito deste processo é prover garantia que produtos de trabalho e processos de um processo ou projeto cumprem com seus requerimentos especificados e aderem-se a seus planos estabelecidos.		
Entradas:	1) Metodologia de desenvolvimento de software 3) Descrição de processo 4) Procedimentos e práticas de trabalho 7) Descrição de produto de trabalho 9) Padrão 17) Plano de projeto 24) Declaração (política) de qualidade 25) Estratégia/plano de qualidade 27) Critério de qualidade 28) Registro de qualidade 30) Estratégia /plano de revisão 37) Medida de projeto 38) Medida de processo 39) Medida de qualidade 52) Especificação de requerimento	Saídas:	3) Descrição de processo 4) Procedimento e prática de trabalho 9) Padrão 19) Atas de reunião 20) Registro/relatório de status de progresso 21) Resultado de análise 25) Plano/estratégia de qualidade 27) Critério de qualidade 28) Registro de qualidade 29) Registro de avaliação /auditoria 31) Registro de revisão 97) Ação corretiva
Início do processo:	Desenvolver estratégia de garantia de qualidade.		
Conteúdo:	Desenvolve estratégia de garantia de qualidade; estabelece padrões de qualidade; define registros de qualidade; garante qualidade de atividades de processo; garante qualidade de produtos de trabalho; relata resultados de qualidade; manipula desvios.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none">• uma estratégia para condução de atividades e tarefas do processo de garantia de qualidade estarão desenvolvidas, implementadas e mantidas;• evidência de atividades e tarefas de qualidade estarão produzidas e mantidas;• problemas ou não-conformidades com requerimentos de contratos estarão identificadas;• aderência de produtos de software, processos e atividades para padrões e procedimentos aplicáveis e requerimentos estará verificada objetivamente. <p>Nota1: Para ser imparcial, garantia de qualidade deve ter liberdade organizacional e autoridade de pessoas diretamente responsáveis pelo desenvolvimento do produto de software ou execução do processo.</p> <p>Nota2: Garantia de qualidade seria coordenada com , e poderia fazer uso de, os resultados de outros processos de suporte, tais como Verificação, Validação, Revisão, Auditoria e resolução de problema.</p> <p>Nota3: Estabelecimento de um sistema de gerência de garantia de qualidade de acordo com a ISO 9001 estabelecerá um processo de garantia de qualidade</p>		

	competente.
Clientela a ser atendida:	
Indicadores de desempenho:	<p>Existência e adequação das práticas base aplicadas neste processo: SUP.3.BP1, SUP.3.BP2, SUP.3.BP3, SUP.3.BP4, SUP.3.BP5, SUP.3.BP6, SUP.3.BP7.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>

SUP.4 Processo de verificação

Nome do processo:	SUP.4 Processo de verificação		
Objetivo:	<p>O propósito deste processo é confirmar que cada produto de trabalho e/ou serviço de software de um processo ou projeto reflete bem os requerimentos especificados.</p> <p>Nota1: O processo normalmente invoca a execução de teste de produtos de trabalho para garantir que eles satisfaçam seu uso intencionado.</p> <p>Nota2: O processo está diretamente ligado com execução do processo <i>ENG.1.6 Processo de teste de software</i> e do processo <i>ENG.1.7 Processo de integração e teste de sistema</i>.</p> <p>Nota3: A norma ISO/IEC 12207 contém requerimentos específicos de conteúdo de plano de verificação.</p> <p>Nota4: O processo pode envolver execução de técnicas como revisão aos pares, prova formal e análise de acompanhamento, entre outras.</p>		
Entradas:	<p>1) Metodologia de desenvolvimento de software</p> <p>3) Descrição de processo</p> <p>6) Estrutura de decomposição de trabalho</p> <p>7) Produto de trabalho</p> <p>9) Padrão</p> <p>10) Padrão de codificação</p> <p>14) Política de verificação</p> <p>17) Plano de projeto</p> <p>20) Relatório de status de progresso</p> <p>25) Estratégia/plano de qualidade</p> <p>27) Critério de qualidade</p> <p>30) Estratégia /plano de revisão</p> <p>52) Especificação de requerimentos</p> <p>59) Estratégia/plano de teste</p> <p>84) Registro de relato de problema</p> <p>98) Sistema de acompanhamento</p>	Saídas:	<p>16) Plano/estratégia de verificação</p> <p>19) Atas de reunião</p> <p>21) Resultado de análise</p> <p>28) Registro de qualidade</p> <p>29) Registro de avaliação /auditoria</p> <p>31) Registro de revisão</p> <p>39) Medida de qualidade</p> <p>58) Registro/mapeamento de acompanhamento (rastreamento)</p> <p>84) Registro de relato de problema</p> <p>97) Ações corretivas</p> <p>98) Sistema de acompanhamento</p>
Início do processo:	Desenvolver estratégia de verificação.		
Conteúdo:	Desenvolve estratégia de verificação; conduz verificação; determina ações para resultados de verificação; acompanha ações para resultados de verificação.		
Término do processo:	<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • uma estratégia de verificação estará desenvolvida e implementada; 		

	<ul style="list-style-type: none"> • critérios de verificação de todos os produtos de trabalho de de software estarão identificados; • atividades de verificação requeridas estarão executadas; • defeitos identificados estarão encontrados e removidos dos produtos de trabalho de software; • resultados das atividades de verificação estarão disponíveis para o cliente e outras organizações envolvidas.
<i>Clientela a ser atendida:</i>	
<i>Indicadores de desempenho:</i>	<p>Existência e adequação das práticas base aplicadas neste processo: SUP.4.BP1, SUP.4.BP2, SUP.4.BP3, SUP.4.BP4.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>

SUP.5 Processo de validação

<i>Nome do processo:</i>	SUP.5 Processo de validação
<i>Objetivo:</i>	<p>O propósito deste processo é confirmar que os requerimentos para um uso intencionado específico do produto de software estão satisfeitos.</p> <p>Nota1: O processo normalmente invoca a execução de teste de produtos de trabalho para garantir que eles satisfaçam seu uso intencionado.</p> <p>Nota2: O processo está diretamente ligado com execução do processo <i>ENG.1.7 Processo de integração e teste de sistema.</i></p>

Entradas:	1) Metodologia de desenvolvimento de software 3) Descrição de processo 7) Produto de trabalho 9) Padrão 14) Política de verificação 17) Plano de projeto 25) Estratégia/plano de qualidade 27) Critério de qualidade 30) Estratégia /plano de revisão 39) Medidas de qualidade 52) Especificação de requerimentos (teste) 58) Registro/mapeamento de acompanhamento (rastreabilidade) 59) Estratégia/plano de teste 64) Plano de teste de software 65) Estratégia/plano de teste de integração 67) Estratégia/plano de teste de regressão 84) Registro de relato de problema	Saídas:	16) Plano/estratégia de verificação 19) Atas de reunião 21) Resultado de análise 28) Registro de qualidade 31) Registro de revisão 39) Medida de qualidade 58) Registro/mapeamento de acompanhamento (rastreabilidade) 60) Roteiro de teste 61) Caso de teste 62) Resultado de teste 84) Registro de relato de problema 97) Ações corretivas 98) Sistema de acompanhamento
Início do processo:	Desenvolver estratégia de validação.		
Conteúdo:	Desenvolve estratégia de validação; executa validação; determina ações para resultados de validação; acompanha ações para resultados de validação.		
Término do processo:	Como resultado de uma implementação de processo bem sucedida: <ul style="list-style-type: none"> • uma estratégia de validação estará desenvolvida e implementada; • critérios para validação de todos os produtos de trabalho requeridos estarão identificados; • atividades de validação requeridas estarão executadas; • todos os problemas identificados estarão resolvidos; • evidência estará provida que os produtos de trabalho de software como desenvolvidos estarão apropriados para o seu uso intencionado; • resultados das atividades de validação estarão disponíveis para o cliente e outras organizações envolvidas. 		
Clientela a ser atendida:			
Indicadores de desempenho:	Existência e adequação das práticas base aplicadas neste processo: SUP.5.BP1, SUP.5.BP2, SUP.5.BP3, SUP.5.BP4. Existência dos produtos de trabalho de entrada e saída e suas características.		

ORG.4 Processo de infra-estrutura

Nome do processo:		ORG.4 Processo de infra-estrutura	
Objetivo:	<p>O propósito deste processo é manter uma infra-estrutura estável e confiável que é necessária para suportar a execução de quaisquer outros processos. A infra-estrutura pode incluir hardware, software, métodos, ferramentas, técnicas, padrões e facilidades para desenvolvimento, operação ou manutenção.</p> <p>Nota: Este processo suporta execução de atributo de processo 3.2 naquelas instâncias onde é invocado.</p>		
Entradas:	<ul style="list-style-type: none"> 1) Metodologia de desenvolvimento de software 2) Modelo de ciclo de vida 7) Descrição de produto de trabalho 9) Padrão 12) Metas de negócio 13) Visão 14) Política 17) Plano de projeto 21) Resultado de análise 23) Gerência de risco 24) Declaração/política de qualidade 25) Estratégia/plano de qualidade 32) Plano de reuso 33) Estratégia de reuso 44) Avaliação de necessidade de produto 52) Especificação de requerimentos (ambiente) 52) Especificação de requerimentos (produto / serviço / cliente / sistema / software) 84) Relato de problema 94) Requisito de mudança 95) Controle de mudança 	Saídas:	<ul style="list-style-type: none"> 1) Metodologia de desenvolvimento de software 2) Modelo de ciclo de vida 3) Descrição de processo 9) Padrão 14) Política 17) Plano de projeto 27) Critérios de qualidade 31) Registro de revisão 32) Plano de reuso 33) Estratégia de reuso 35) Repositório de reuso 39) Medida de qualidade 52) Especificação de requerimentos (produto / serviço / cliente / sistema / software) 87) Mecanismo de comunicação 92) Gerência de configuração (arquivo, biblioteca, sistema) 95) Registro de controle de mudança 102) Registro de backup/recuperação 103) Estratégia/plano de recuperação 104) Ambiente de desenvolvimento
Início do processo:			
Conteúdo:	<p>Identifica requerimentos de ambiente de engenharia de software; provê um ambiente de engenharia de software;</p> <p>provê suporte para indivíduos usando a infra-estrutura de engenharia de software; mantém ambiente de engenharia de software; provê um espaço de trabalho que conduz para uma execução produtiva; garante integridade e segurança de dados e; provê facilidade de acesso remoto.</p>		
Término do processo:		<p>Como resultado de uma implementação de processo bem sucedida:</p> <ul style="list-style-type: none"> • uma infra-estrutura estará estabelecida que é consistente com e suporta procedimentos de processo, padrões, ferramentas e técnicas aplicáveis; 	

	<ul style="list-style-type: none"> • a infra-estrutura reunirá todos requerimentos para funcionalidade, execução, segurança, disponibilidade, espaço, equipamento, custo, tempo e integridade de dados.
<i>Cientela a ser atendida:</i>	
<i>Indicadores de desempenho:</i>	<p>Existência e adequação das práticas base aplicadas neste processo: ORG.4.BP1, ORG.4.BP2, ORG.4.BP3, ORG.4.BP4, ORG.4.BP5, ORG.4.BP6, ORG.4.BP7.</p> <p>Existência dos produtos de trabalho de entrada e saída e suas características.</p>

Anexo IV - Formato do Formulário *Lista de Perspectivas*

Logomarca

Formulário Lista de Perspectivas sobre o Sistema

Projeto

(código/nome):

Considerações

1. O participante de encontro FAST deve construir esta lista de perspectivas durante a revisão do 201) Esboço de requerimento de produto, antes da realização do encontro. A lista de perspectivas contém: uma lista de *objetos* que fazem parte do ambiente e circundam o sistema; uma lista de *objetos* que são produzidos pelo sistema; uma lista de *objetos* que são usados para que o sistema execute suas funções; uma lista de *operações* (processos ou funções) que manipulam ou interagem com os objetos; uma lista de *restrições* (por exemplo, custo, tamanho); uma lista dos *critérios de desempenho* (por exemplo, velocidade, precisão) e uma lista de *critérios de validação*.
2. A lista de perspectivas não precisam ser exaustivas, mas devem refletir as perspectivas de cada pessoa sobre o sistema. Elas serão manipuladas no encontro FAST.
3. Um encontro FAST consiste dos seguintes passos:
 - a. discutir e estabelecer consenso sobre a necessidade e justificativa do novo produto;
 - b. cada participante deve apresentar suas listas para crítica e discussão; as listas podem ser afixadas às paredes da sala usando grandes folhas de papel; idealmente, cada tópico da lista pode ser manipulado separadamente, de forma que as listas sejam combinadas, entradas sejam apagadas e adições sejam feitas; nessa fase, a crítica e o debate são estritamente proibidos;
 - c. criar uma lista combinada para cada área abrangendo todos os temas e que reflita as idéias apresentadas; esta lista elimina as entradas redundantes e acrescenta quaisquer novas idéias que possam aparecer durante a apresentação, mas não apaga nenhuma;
 - d. iniciar a discussão sobre as listas combinadas - coordenada pelo moderador - com o objetivo de desenvolver uma lista consensual de cada área de assunto (objetos, operações, restrições e desempenho); cada lista combinada é abreviada, ampliada ou novamente redigida para refletir adequadamente o sistema/produto a ser desenvolvido; as listas são então guardadas para ação posterior;
 - e. dividir a equipe em subequipes menores, assim que as listas de consenso forem concluídas;
 - f. desenvolver em cada subequipe uma *miniespecificação* de uma ou mais entradas de cada uma das listas; a *miniespecificação* é uma elaboração das palavras ou frases contidas numa lista;

- g. cada subequipe deve apresentar cada uma das miniespecificações a todos os participantes do encontro FAST para discussão; adições, supressões e elaboração adicional são feitas e em alguns casos, o desenvolvimento de miniespecificações desvendará novos objetos, operações, restrições ou exigências de desempenho que serão acrescentadas às listas originais; durante as discussões, a equipe pode levantar questões que não serão resolvidas durante o encontro e uma lista de questões deve ser guardada para que possa agir sobre essas idéias posteriormente;
- h. cada subequipe devem elaborar uma lista de critérios de validação para o produto/sistema - após a conclusão das miniespecificações - e apresentá-la à equipe;
- i. criar uma lista de consenso dos critérios de validação;
- j. responsabilizar um ou mais participantes (ou pessoas de fora) pela tarefa de escrever o esboço completo de especificação usando todas as entradas do encontro FAST.

Anexo V - Formato do Documento *Especificação de Requerimentos de Cliente*

O texto a seguir é uma ferramenta do tipo "modelo do MS-Word" e contém as características do produto de trabalho 52) *Especificação de requerimentos (cliente)* organizadas no formato de documento recomendado pela ISO/IEC TR 15504.

Logomarca <Identificação da empresa>

Código de Projeto:

Nome:

Documento: *Especificação de Requerimentos de Cliente*

Versão:

Documento: *X.nn* Data: *dd/mm/aaaa*

Software: *X.nn*

Sistema: *X.nn*

Índice

<índice geral>

Lista de Figuras

<índice de figuras>

Lista de Tabelas

<índice de tabelas>

Nomenclatura

Estilo

<identificação de estilo >

Formato

<identificação de formato >

Padrões de mídia esperados

<identificação de padrões de mídia esperados>

Audiência intencionada

- Cliente
- Arquiteto de software
- Desenvolvedores
- Garantia de qualidade/testadores
- <outros>

Distribuição

<definir requerimento de distribuição intencionado>

Armazenamento

<incluir requerimentos de armazenamento>

1 Requerimentos de cliente

Para facilitar a localização, os requerimentos estão organizados nas categorias Funcional, Técnico, Operacional, Qualidade e Manutenção. Dentro da categoria são identificados segundo a notação:

TRnnnn

- onde:
- TR representa o tipo do requerimento
 - F = funcional T = técnicos
 - O = operacionais Q = qualidade
 - M = manutenção
 - nnnn representa um número sequencial de requerimento, iniciando de 0001.

As seções a seguir apresentam a descrição dos requerimentos e seus respectivos identificadores. A *Requerimento* descreve quaisquer requerimentos, expectativas, características, considerações ou restrições para o produto que está sendo especificado.

1.1 Requerimentos Funcionais

ID	Requerimento
Fnnnn	<requerimento>
Fnnnn	<requerimento>

1.2 Requerimentos Técnicos

ID	Requerimento
Hardware	
Tnnnn	<requerimento>
Tnnnn	<requerimento>
Software	
Tnnnn	<requerimento>
Tnnnn	<requerimento>
Interfaces de comunicação (externa, interna e engenharia humana)	
Tnnnn	<requerimento>
Tnnnn	<requerimento>

Capacidade

Tnnnn <requerimento>

Tnnnn <requerimento>

Desempenho

Tnnnn <requerimento>

Tnnnn <requerimento>

Acesso à aplicação ("Security")

Tnnnn <requerimento>

Tnnnn <requerimento>

"Safety/reliability"

Tnnnn <requerimento>

Tnnnn <requerimento>

Sistema e design

Tnnnn <requerimento>

Tnnnn <requerimento>

1.3 Requerimentos Operacionais

ID	Requerimento
----	--------------

Instalação

Onnnn <requerimento>

Onnnn <requerimento>

Suporte

Onnnn <requerimento>

Onnnn <requerimento>

Documentação

Onnnn <requerimento>

Onnnn <requerimento>

Treinamento

Onnnn <requerimento>

Onnnn <requerimento>

Ambiente

Onnnn <requerimento>

Onnnn <requerimento>

Armazenamento (de produto)

Onnnn <requerimento>

Onnnn <requerimento>

Outros

Onnnn <requerimento>

Onnnn <requerimento>

1.4 Requerimentos de qualidade (inclusive os critérios de aceitação)

ID	Requerimento
-----------	---------------------

Estatutório/Regulatório

Onnnn <requerimento>

Onnnn <requerimento>

Outros

Qnnnn <requerimento>

Qnnnn <requerimento>

1.5 Requerimentos de Manutenção

ID	Requerimento
-----------	---------------------

Mnnnn <requerimento>

Mnnnn <requerimento>

2 Referências Bibliográficas

3 Anexo I

4 Apêndice A