



UNIVERSIDADE FEDERAL DE LAVRAS

**Curso: Melhoria de Processo de
Software - MPS**

**IMPLANTANDO GERÊNCIA DE
CONFIGURAÇÃO COM O APOIO DA
FERRAMENTA SUBVERSION (SVN) PARA
ATINGIR OS RESULTADOS ESPERADOS DO
PROCESSO GCO DO NÍVEL F DO MPS.BR**

ANNA CAROLINA PEREIRA

LAVRAS
MINAS GERAIS - BRASIL
2008

ANNA CAROLINA PEREIRA

**IMPLANTANDO GERÊNCIA DE
CONFIGURAÇÃO COM O APOIO DA
FERRAMENTA SUBVERSION (SVN) PARA
ATINGIR OS RESULTADOS ESPERADOS DO
PROCESSO GCO DO NÍVEL F DO MPS.BR**

Monografia (ou Trabalho de Conclusão) apresentada (o) ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação *Lato Sensu* em Melhoria de Processo de Software, para a obtenção do título de especialização.

Orientador
Prof.^a Dr.:Ana Liddy Cenni de Castro Magalhães

LAVRAS
MINAS GERAIS - BRASIL
2008

ANNA CAROLINA PEREIRA

**IMPLANTANDO GERÊNCIA DE
CONFIGURAÇÃO COM O APOIO DA
FERRAMENTA SUBVERSION (SVN) PARA
ATINGIR OS RESULTADOS ESPERADOS DO
PROCESSO GCO DO NÍVEL F DO MPS.BR**

Monografia (ou Trabalho de Conclusão) apresentada (o) ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação *Lato Sensu* em Melhoria de Processo de Software, para a obtenção do título de especialização.

APROVADA em ___ de _____ de _____.

Prof. _____

Prof. _____

Prof. _____

Dr. Ana Liddy Cenni de Castro Magalhães

LAVRAS
MINAS GERAIS - BRASIL

AGRADECIMENTOS

“A Deus, por me possibilitar essa grande vitória. Sem ele nada seria possível! Obrigada, meu Senhor.”

Aos meus pais, José Caixeta Pereira e Maria de Fátima Luiz Pereira, responsáveis pela minha existência, dedico este trabalho como um retorno simbólico por todas as oportunidades oferecidas, pelo amor, pela educação e pelo exemplo de vida. Obrigada, vocês são tudo para mim.

Ao meu namorado, Roger Serrati, pela paciência e por estar comigo a cada passo da minha Pós-Graduação, me dando forças para continuar e me apoiando nos momentos mais difíceis que a vida nos propôs. Obrigada meu amor. Te amo.

Aos meus irmãos, Otávio e Gabriel, pelo carinho e compreensão em relação às minhas ausências no decorrer desta Pós-Graduação.

À professora Ana Liddy pelo carinho, compreensão, atenção e pela a grande dedicação em me orientar neste trabalho.

Às minhas amigas do Curso MPS, turma 207, Alexia Lage, Gabriela Rocha, Josiane Brietzke e Juliana Pohren, pois o nosso trabalho em equipe resultou no sucesso da conclusão desse curso. Obrigada amigas pelo carinho e pela amizade.

À UFLA, e todos os seus funcionários, por todo carinho e dedicação, o que garantiu a qualidade do curso de Melhoria de Processos de Software. Com certeza, essa é uma universidade exemplar no ensino a distância.

SUMÁRIO

LISTA DE FIGURAS.....	6
LISTA DE TABELAS	7
1. Introdução.....	8
2. Principais Conceitos Relacionados	9
2.1. Fundamentos de Gerência de Configuração.....	9
2.2. O Modelo de Referência MPS.BR	10
2.3. A Ferramenta Subversion	13
2.3.1. Conceitos Básicos da Ferramenta SVN.....	14
2.3.2. Funcionamento Básico Subversion	14
2.3.3. Ramificação e Rotulação (Branch/Tag)	16
2.3.4. Sugestões de Utilização do Subversion	18
3. Estudo de Caso	18
3.1. Descrição da Organização	19
3.2. Processo de Gerência de Configuração	19
3.2.1. Planejar a Gerência de Configuração do Projeto.....	21
3.2.2. Preparar Ambiente de Configuração	22
3.2.3. Solicitar Mudanças	23
3.2.4. Gerar Baseline e Release	23
3.2.5. Realizar Auditoria de Configuração.....	24
3.2.6. Gerenciar Liberação e Distribuição de Release.....	25
3.3. Processo de Gerência de Configuração SWB x Resultados Esperados GCO do modelo MPS.BR.....	25
4. Conclusão	26
5. Referências Bibliográficas	28

LISTA DE FIGURAS

Figura 1- Níveis de Maturidade do MR-MPS [Goulart, Consultoria].....	10
Figura 2 - Arquitetura Cliente-Servidor	13
Figura 3 – Funcinamento Básico da Ferramenta [De MARTINHO, 2008a]	15
Figura 4 – Operação Branch/Tag [De MARTINHO, 2008a]	16
Figura 5 – Operação Merge [De MARTINHO, 2008a]	17
Figura 6 – Operação Switch [De MARTINHO, 2008a].....	17
Figura 7 - Layout Repositório [De MARTINHO, 2008a].....	18
Figura 8 – Processo de Gerência de Configuração.....	20

LISTA DE TABELAS

Tabela 1 – Resultados Esperados de Gerência de Configuração MR-MPS [MPS, 2007b]	11
Tabela 2 – Conceitos SVN	14
Tabela 3 – Principais Comandos do Subversion	16
Tabela 4 – Processo de Gerência de Configuração SBW X GCO do modelo MPS.BR	26

Implantando Gerência de Configuração com o apoio da ferramenta Subversion (SVN) para atingir os resultados esperados do processo GCO do nível F do MPS.BR

Anna Carolina Pereira

Universidade Federal de Lavras (UFLA)

Lavras – MG – Brasil

anna.carol.p@gmail.com

***Abstract.** This document presents a case study on the process configuration management with the support of the Subversion tool, aiming at achieving the expected results of the level F MPS.BR model. This case study was implemented in a medium businesses company in an effort to manage the large number of items of information produced and maintained during a software project life cycle. The Software Configuration Management has resulted from the need to monitor these changes to prevent loss of control of information in the software project.*

***Resumo.** Este documento apresenta um estudo de caso sobre o processo de Gerência de Configuração com o apoio da ferramenta Subversion, visando atingir os resultados esperados do modelo MPS.BR nível F. Esse estudo de caso foi aplicado em uma empresa de médio porte com o intuito de gerenciar a grande quantidade de itens de informação produzida e alterada ao longo do ciclo de vida de um projeto de software. A Gerência de Configuração de Software surgiu da necessidade de controlar essas alterações para evitar a perda de controle de informação do projeto de software.*

1. Introdução

Atualmente, as empresas de Tecnologia da Informação (TI) de pequeno e médio porte estão em busca de um programa de melhoria de processos de software com o intuito de aprimorar a qualidade de seus produtos e processos envolvidos na construção de projetos de software.

Ao longo do ciclo de vida de um projeto de software, uma grande quantidade de itens de informação é produzida e alterada. A Gerência de Configuração de Software

surgiu da necessidade de controlar essas alterações para evitar a perda de controle do projeto, beneficiando a sua gestão e o seu desenvolvimento. Nesse sentido, este artigo apresenta uma abordagem para o processo de Gerência de Configuração de Software referenciada na Gerência de Configuração proposta pela norma ISO/IEC 12207 e pelo *Software Engineering Body of Knowledge (SWEBOK)*.

A Gerência de Configuração (GC) é uma disciplina cujo objetivo é identificar a Configuração de um sistema em pontos distintos no tempo, que se aplica a todos os produtos gerados durante o desenvolvimento de um projeto. Além disso, controla sistematicamente as mudanças identificadas e mantém a integridade e a rastreabilidade da Configuração por todo o ciclo de vida do sistema.

O objetivo deste trabalho é propor um processo de Gerência de Configuração que utilize a ferramenta Subversion, para uma empresa de médio porte que está implementando e pretende ser avaliada em breve no nível F do modelo MPS.BR.,

Como motivação para o desenvolvimento do tema deste trabalho, são apresentados alguns pontos que se destacam por justificarem a importância da Gerência de Configuração, entre eles: GC é uma das disciplinas de base para a produção de software de qualidade; é parte integrante das principais iniciativas de melhoria da qualidade (ISO 9000, CMMI, ISO/IEC 15504, ISO/IEC 12207 e MPS.BR); permite melhorar a visibilidade de um esforço de desenvolvimento por meio do registro da evolução do software; facilita a comunicação do estado de documentos e código, bem como das mudanças neles aplicadas; propicia uma abordagem sistemática a uma das maiores causas de atrasos de cronograma e estouros de orçamento, que são as mudanças que ocorrem durante os projetos.

Em face disso, este trabalho consiste, além dessa introdução, da sessão 2, que contém os principais conceitos relacionados à temática desenvolvida; da seção 3, que apresenta o Estudo de Caso realizado na organização SWB Soluções Integradas, um histórico das tentativas anteriores de investimento em programas de qualidade e o processo de Gerência de Configuração com o apoio da ferramenta Subversion. Também apresenta a definição do processo, procedimentos, políticas e artefatos padrões, para garantir a melhoria contínua dos processos de desenvolvimento realizados na organização, que possam garantir o melhor desempenho da empresa no mercado de software. Finalmente, da seção 4, que apresenta a conclusão geral do artigo.

2. Principais Conceitos Relacionados

A seguir, serão apresentados os principais conceitos relacionados à Gerência de Configuração, ou seja, as noções conceituais essenciais para o entendimento desse processo [De MARTINHO, 2008b]

2.1. Fundamentos de Gerência de Configuração

- **Item de Configuração:** É uma agregação de hardware, software, ou ambos, que é designada para a Gerência de Configuração e tratada como uma entidade única para esse processo [CMU/SEI, 2006]. Pode constituir-se de qualquer elemento de informação, associado de um produto de software, identificado de maneira única e selecionado para controle, por exemplo: um documento, um conjunto de produtos de trabalho que constituem uma especificação, um componente de produto de software (código, funções, classes).
- **Baseline:** Consiste em um conjunto de produtos de trabalho, itens de Configuração que foram formalmente submetidos à análise crítica e aprovados, definidos e fixados

em um determinado momento durante o ciclo de vida da Configuração. A partir disso, são utilizados como base para a continuação do desenvolvimento de software, podendo apenas ser modificados por meio de procedimentos formais de controle de mudança [CMU/SEI, 2006].

- **Biblioteca de Software:** Coleção controlada de software, documentação relacionada e ferramentas de apoio ao desenvolvimento, também conhecida como repositório do projeto. Possui como principais funcionalidades o armazenamento, controle de versões, controle de acesso dos Itens de Configuração(IC) e a criação de ambiente cooperativo para elaboração compartilhada dos artefatos de projeto.
- **Build:** Termo utilizado para identificar as atividades associadas aos processamentos de códigos fontes para obter os arquivos finais (executáveis / binários). Essas atividades podem ser organizadas em várias ações: verificação de dependências, compilação de fontes, integração de executáveis e geração de dados.
- **Versão:** Instância de um IC que foi formalmente aprovado por superiores, ou seja, uma melhoria da versão anterior.
- **Release de Sistemas:** Representa a notificação formal e distribuição de uma versão aprovada do software que será entregue ao cliente.

2.2. O Modelo de Referência MPS.BR

O MPS.BR é um programa para Melhoria de Processo do Software Brasileiro, coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), contando com apoio do Ministério da Ciência e Tecnologia (MCT) [MPS.BR 2007a]. Esse programa está estruturado em três componentes principais: Modelo de Referência (MR-MPS), Modelo de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS) [MPS, 2007a].

O MR-MPS contém 23 áreas de processos e seus respectivos níveis de maturidade que foram descritos e estruturados com base nas normas ISO/IEC 12207 e ISO/IEC 15504, além do modelo CMMI-DEV. Ele é definido por meio de sete níveis de maturidade, seqüenciais e acumulativos, conforme Figura 1.

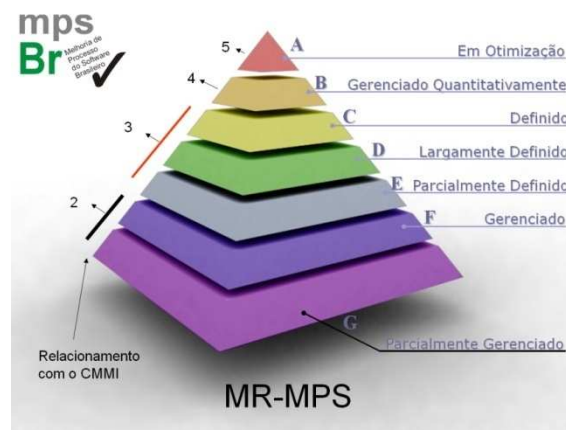


Figura 1- Níveis de Maturidade do MR-MPS [Goulart Consultoria, 2007]

O nível F contém o Processo de Gerência de Configuração, com os seguintes resultados esperados:

Processo de Gerência de Configuração do MR-MPS	
GCO 1:	Um Sistema de Gerência de Configuração é estabelecido e mantido.
GCO 2:	Os itens de Configuração são identificados.
GCO 3:	Os itens de Configuração, sujeitos a um controle formal, são colocados sob baseline.
GCO 4:	A situação dos itens de Configuração e das baselines é registrada ao longo do tempo e disponibilizada.
GCO 5:	As modificações em itens de Configuração são controladas e disponibilizadas.
GCO 6:	As auditorias de Configuração são realizadas objetivamente para assegurar que as baselines e os itens de Configuração estejam íntegros, completos e consistentes.
GCO 7:	O armazenamento, o manuseio e a liberação de itens de Configuração e baselines são controlados.

Tabela 1 – Resultados Esperados de Gerência de Configuração MR-MPS [MPS, 2007b]

A Gerência de Configuração (GC) engloba um conjunto de atividades de apoio ao desenvolvimento de software que permite que as mudanças inerentes sejam absorvidas pelo projeto de maneira controlada, mantendo a estabilidade e integridade dos dados na evolução do software.

O processo de GC contempla a execução das seis atividades descritas a seguir [MURTA, 2006]:

- **Planejar Gerência de Configuração:** Elemento de GC que consiste da definição de como a disciplina será aplicada num esforço de desenvolvimento, visando à elaboração de um plano para consolidar a definição e a monitoração de sua aplicação. O plano de Gerência de Configuração estabelece os objetivos e a abrangência da GC a ser implantada, bem como caracteriza as atividades que serão executadas, descrevendo os processos a serem realizados. Também identifica os papéis responsáveis por realizar as atividades contidas no Plano de GC, define os recursos necessários, incluindo as ferramentas de software e recursos de hardware, e estabelecendo cronogramas e relacionamentos entre as atividades.
- **Identificar Configuração:** Tem por objetivo possibilitar: (i) selecionar os itens de Configuração que serão produzidos ao longo do ciclo de vida do projeto e identificar aqueles que deverão ter suas modificações controladas; (ii) definir o esquema de nomes e números, que possibilitem a identificação inequívoca dos itens de Configuração no grafo de versões e variantes; (iii) registrar, em documentação técnica, as características físicas e funcionais dos itens de Configuração e identificar o responsável por cada item.
- **Controlar Configuração:** Tem por objetivo manter o controle, o registro e o acompanhamento da evolução das mudanças realizadas nos itens de Configuração ao longo do processo de desenvolvimento. Para que esses itens possam evoluir de

forma controlada, essa função estabelece as seguintes atividades: (i) solicitar mudança, que pode ser executada por qualquer membro da equipe que deseja alterar um item de Configuração. A solicitação de mudança com o objetivo de manutenção pode ter várias finalidades, tais como: corretiva, evolutiva, adaptativa ou preventiva. Após a solicitação ela deverá ser registrada; (ii) analisar o pedido de solicitação de mudança de acordo com a sua classificação de modificação, na qual seus responsáveis deverão analisar o pedido de alteração e, em seguida, fornecer um parecer aprovando ou não a solicitação. Caso seja aprovada, deverão estabelecer prioridade em relação às demais solicitações efetuadas anteriormente; (iii) analisar o impacto, após a aprovação da solicitação da mudança, o qual visa a relatar esses impactos em esforço, cronograma e custo, bem como definir uma proposta de implementação da manutenção; (iv) avaliar a modificação pelo Comitê de Controle da Configuração (CCC) – o qual estabelece se a modificação será implementada, rejeitada ou postergada, em função do laudo fornecido pela análise de impacto da modificação; (v) implementar a modificação, caso a solicitação tenha sido aprovada após avaliação; (vi) verificar a modificação com relação à proposta de implementação levantada na análise de impacto; (vii) atualizar a baseline, que pode ou não ser liberada para o cliente em função da sua importância e questões de marketing associadas.

- **Relatar Situação:** armazenar e divulgar as informações necessárias para gerenciar uma Configuração de maneira efetiva. As informações englobam a listagem da identificação de Configuração aprovada, o *status* de mudanças propostas e o *status* de implementação das mudanças aprovadas. Todos os envolvidos no processo de Gerência de Configuração deverão ter acesso a *essas* informações para serem utilizadas nos processos de medição, estimativa de esforço, custos futuros e geração de relatórios gerenciais.
- **Auditoria da Configuração:** Avaliação e revisão da Configuração. Consiste na execução de auditorias físicas e funcionais sobre a linha base antes da sua liberação, com o intuito de assegurar a sua completude e corretude. A auditoria é uma atividade que determina o grau com que um IC satisfaz as características funcionais e físicas, seguindo essa ordem.
- **Auditoria Funcional da *Baseline*:** Tem o objetivo de garantir que o Item de Configuração que está sendo auditado esteja de acordo com suas especificações; comprovando que esse item foi devidamente testado e que os testes evidenciam o atendimento a todos os seus requisitos. O documento referencial para a auditoria são as especificações do produto.
- **Auditoria Física *Baseline*:** Tem o objetivo de garantir que a representação física do produto, e toda a documentação auditada, esteja consistente com o produto elaborado (construído). Essa auditoria busca verificar o formato, a consistência e a completeza de toda a documentação e sua compatibilidade com o produto desenvolvido. O documento referencial para a auditoria é o produto em si. As auditorias de Configuração verificam se a *baseline* como um todo, incluindo as diversas modificações que ocorreram no projeto, está correta e completa para ser liberada. O responsável pela auditoria de Gerência de Configuração deve ter um bom conhecimento técnico em GC e não deve ter nenhum vínculo no projeto que está sendo auditado.

É importante ressaltar que a auditoria de Gerência de Configuração não deve ser confundida nem possuir sobreposição com a auditoria de Garantia da Qualidade, uma das atividades do processo Garantia da Qualidade (GQA), pois são atividades distintas e podem ser executadas em paralelo no projeto.

- **Gerência de Liberação e Entrega:** Consiste no controle formal da liberação de produtos de software e de toda documentação do projeto, englobando a identificação, o empacotamento e a entrega dos elementos integrantes de uma distribuição de software. A distribuição se refere ao ato de liberar produtos de software, e também ao conjunto de itens que são disponibilizados na liberação, identificando as versões de cada Configuração que serão disponibilizadas. A distribuição pode ser tanto interna, para a própria organização (testes, avaliações, inspeções), ou externa, para os clientes, com a entrega do produto e implantação no seu próprio ambiente.

2.3. A Ferramenta Subversion

O Subversion (SVN) é uma ferramenta de controle de versão *free/open-source* que permite armazenar documentos de qualquer natureza, efetuando controle de acesso aos repositórios e mantendo as alterações por meio de um controle de revisões. O SVN foi desenvolvido tendo como base a ferramenta *Concurrent Version System (CVS)*, baseado nos seguintes modelos Cópia-Modificação-Mesclagem e Reserva-Modifica-Libera, com suporte para o modelo “*Checkin-Checkout*”

O SVN possui uma arquitetura cliente-servidor, conforme a figura 2, voltada para ambientes cooperativos, cujo propósito é de não ser exclusiva para ambientes de desenvolvimento. No servidor são encontrados os repositórios dos arquivos armazenados em um banco de dados e diversos “*clients*”, que são utilizados para acessar os documentos.

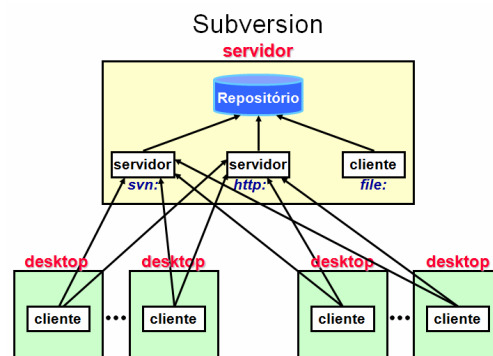


Figura 2 - Arquitetura Cliente-Servidor

O Subversion gera arquivos e diretórios que são colocados num repositório central, que atuará de forma similar a um servidor de arquivos. Essa ferramenta pode ser utilizada tanto na plataforma Windows quanto no Linux.

O Subversion possui suporte para dois modelos de trabalho cooperativo:

- **Cópia-Modificação-Mesclagem** – Nesse modelo não há o conceito de travamento ou reserva para uso exclusivo dos documentos contidos no repositório de dados. Pode haver inúmeras pessoas modificando simultaneamente cópias distintas de um mesmo documento, e não ser possível por meio da ferramenta, saber quantas e quem são essas pessoas. Podem, ainda, ocorrer conflitos e, caso ocorram, o trabalho

deverá ser cooperativo, utilizando a comunicação e o bom senso de todos para tentar resolvê-los coletivamente entre os que realizaram alterações simultâneas no documento.

- **Reserva-Modifica-Libera** – Esse modelo é mais conhecido como “*Checkin-Checkout*”, pois, para modificar um arquivo, primeiro é necessário reservá-lo. Essa reserva garante que ninguém mais consiga modificar o arquivo enquanto não for liberado. As reservas de arquivos são visíveis a toda a equipe de forma que os conflitos possam ser totalmente evitados. Esse é um método seguro, porém restritivo e indicado para arquivos binários em geral, para os quais constitui-se em melhor solução.

Para acessar o Subversion é necessário um aplicativo de interface como, por exemplo, o TortoiseSVN, que é uma ferramenta gráfica que se integra ao Windows e que facilita o acesso e uso do Subversion.

2.3.1. Conceitos Básicos da Ferramenta SVN

A tabela 2 apresenta os principais conceitos relacionados à Gerência de Configuração utilizando o Subversion. Essas definições são primordiais para o entendimento das funcionalidades da ferramenta [De MARTINHO, 2008a].

Repositório:	Local onde são armazenados os arquivos mantidos sob controle de versão, cujo acesso é somente através do servidor via url do repositório.
Área de Trabalho:	Cópia local dos arquivos de um repositório, utilizada para realizar modificações e envio dos arquivos atualizados ao servidor.
Mudança	Cada mudança estabelece uma nova revisão no repositório de dados, ou seja, é um conjunto de modificações que leva o repositório de uma revisão para a revisão imediatamente seguinte (revisão N → revisão N+1)
Revisão:	Numeração incremental que é específica de um documento ou componente de um software, normalmente sem relação direta com a versão global do mesmo. , ou seja, refere-se a um estado específico do repositório.
Trunk:	Trunk é caminho principal de desenvolvimento Significa “Versão Corrente”, ou seja, tronco ou principal. Local onde são armazenados os elementos que constituem o caminho principal de desenvolvimento
Branching:	Significa “Versão especial”, ou seja, ramos do diretório onde são criados os caminhos alternativos de desenvolvimento, ou seja, é um mecanismo para criação de caminhos alternativos ao principal. Cada caminho corresponde a um subdiretório.
TAG's:	As tags são comumente usadas para dar um nome simbólico a uma determinada revisão. As tags poderão conter o nome do programa ou número do release.

Tabela 2 – Conceitos SVN

2.3.2. Funcionamento Básico Subversion

A figura 3 aborda o fluxo básico de funcionamento da ferramenta Subversion e apresentam em quais processos são executados os comandos do SVN.

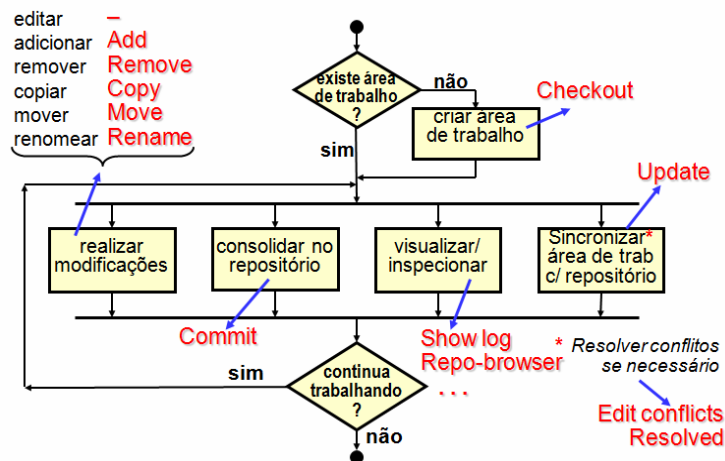


Figura 3 – Funcionamento Básico da Ferramenta [De MARTINHO, 2008a]

A tabela 3 a seguir apresenta os principais comandos importantes para o funcionamento da ferramenta do Subversion como apoio no processo de Gerência de Configuração. Todas as operações abaixo são aplicadas a partir da área de trabalho do Subversion, que pode estar na máquina local ou em servidor.

Add	Esse comando é utilizado para adicionar novo arquivo no repositório de Dados, os arquivos adicionados sofreram alguma alteração, estes arquivos estão dentro do repositório local, para atualização no servidor de produção/ servidor, deverá em conjunto, executar o comando <i>SVN Commit</i> .
Add/Remove	Adiciona e remove os arquivos do repositório.
Blame	Essa opção demonstra para o desenvolvedor a última alteração realizada no arquivo texto, linha a linha, e quem modificou.
Check for modifications	Essa opção no menu apresenta o(s) arquivo(s) que ainda não fazem <i>Commit</i> (atualização no repositório do servidor).
Checkout	Obtem uma cópia de um diretório para a área de trabalho local.
Commit	Envia as alterações da cópia local para o repositório de produção.
Delte	Essa opção no menu deleta o arquivo pelo controle de versão, mas versões anteriores permanecem. Nota: caso <i>delete</i> o arquivo fisicamente, é permitido restaurá-lo.
Get lock	Esse comando é utilizado para reservar o documento em manutenção e sinalizar a um outro desenvolvedor, que tentar alterar o mesmo arquivo, que esta sobre o comando de outro desenvolvedor. Informa, ainda, situações em que se deseja controlar alterações em arquivos específicos. Nota: para saber se outro está utilizando o arquivo, basta selecionar a opção <i>Get lock</i> .
Merge	Tem a função de mesclar trabalhos executados em paralelo em um mesmo arquivo e possui o objetivo de fundir estes trabalhos particulares em uma versão final do arquivo. O sistema de controle de versão frequentemente auxilia na fusão, mas o ser humano é o responsável de fazê-la funcionar corretamente.
Revert	Esta opção no menu permite desfazer as alterações realizadas, antes do <i>Commit</i> , voltando ao estado original, dentro do repositório local (PC do desenvolvedor).

Resolved	Essa opção no menu permite resolver conflitos de alterações que ocorreram em um mesmo documento. Este comando é utilizado quando não foi possível executar o <i>Merge</i> com sucesso.
Show log	Essa opção mostra todas as revisões (versões) existentes no repositório local, podendo realizar comparações entre duas revisões (Botão direito ou marcar as duas revisões) e demonstrar as diferenças. Nota: o ícone verde significa que está sobre o controle de versão da aplicação e o ícone vermelho indica que foi alterado.
Switch	Transforma área de trabalho de uma URL para acessar outra URL. Essa ação é permitida somente dentro do mesmo repositório de dados.
Update	Essa opção do menu permite atualizar / sincronizar a área de trabalho com repositório de dados contido no Servidor.

Tabela 3 – Principais Comandos do Subversion

2.3.3. Ramificação e Rotulação (*Branch/Tag*)

Todo projeto da SWB possui uma estrutura de pasta padrão no repositório de dados Subversion, portanto, no decorrer do projeto, é necessário criar Ramificação e Rotulação para gerar as *baselines* do projeto.

Ramificação e Rotulação no Subversion é a cópia de um elemento ou de diretórios para outro ponto do repositório do projeto, o qual denomina-se de Ramo. Um ramo ou caminho no sistema Subversion não é um conceito especial, é apenas uma cópia de um elemento.

Todas as operações contidas abaixo foram realizadas por meio da ferramenta TortoiseSVN. Para criar um Ramo no Subversion é necessário executar os seguintes comandos:

- Ramificação: **Branch/Tag** (a operação *Branch/Tag*, portanto, é apenas um nome diferente para a operação de cópia). Passos para criar um ramo ou rótulo(tag) para um elemento: 1 - Selecionar o elemento para qual se deseja criar o ramo (em geral própria raiz da área de trabalho), 2 - Executar a operação “*Branch/Tag*”, 3 - Preencher adequadamente a janela de diálogo de “*Branch/Tag*” conforme a figura 4.

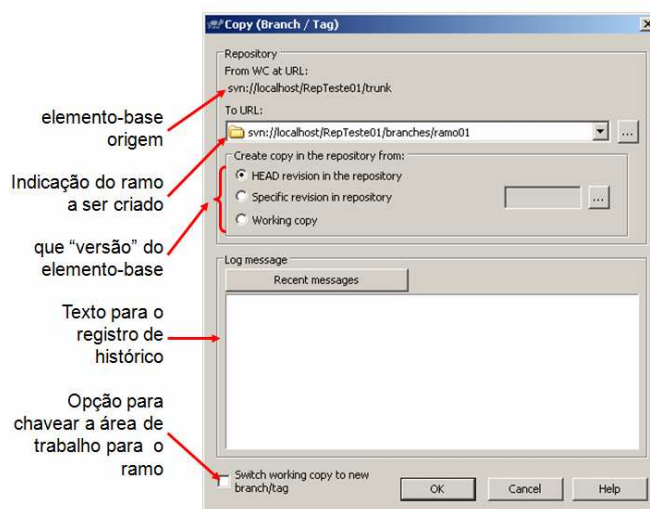


Figura 4 – Operação Branch/Tag [De MARTINHO, 2008a]

- Mescla: *Merge*:

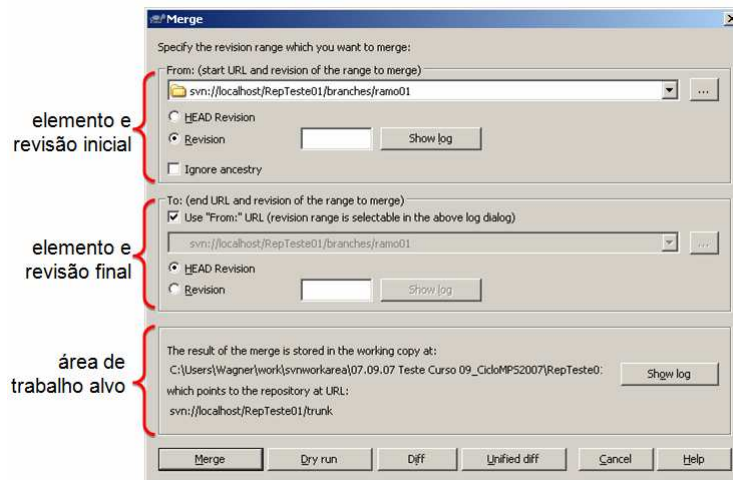


Figura 5 – Operação Merge [De MARTINHO, 2008a]

- Chaveamento de área de trabalho: *Switch* (após selecionar a opção Switch no menu do Subversion é necessário informar a **nova URL que** será acessada).

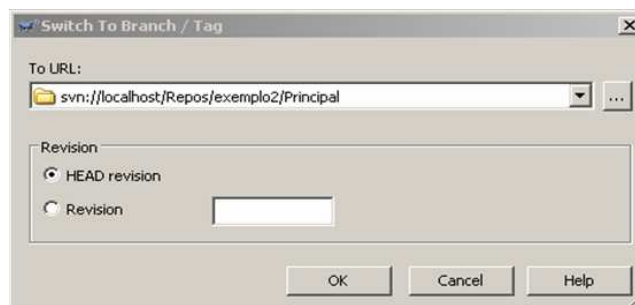


Figura 6 – Operação Switch [De MARTINHO, 2008a]

A ramificação é utilizada para proporcionar um trabalho paralelo com as equipes de desenvolvimento (operacional) e, geralmente, é aplicada em codificação, ou seja, cada desenvolvedor executa seu trabalho para não impactar o do colega. Ao final da iteração, ou do trabalho, executa-se o *Merge* para unir as atividades dos desenvolvedores.

A Rotulação é uma versão estabilizada das ramificações do projeto, é um mecanismo para criação da *baseline*.

Recomenda-se organizar o repositório do projeto em 3 diretórios, facilitando, assim, o trabalho dos envolvidos no projeto:

- **Trunk** (tronco ou principal): repositório principal, original.
- **Branches** (ramos): diretório alternativo de desenvolvimento.
- **Tags** (rótulos): diretório onde se encontra versão estável.

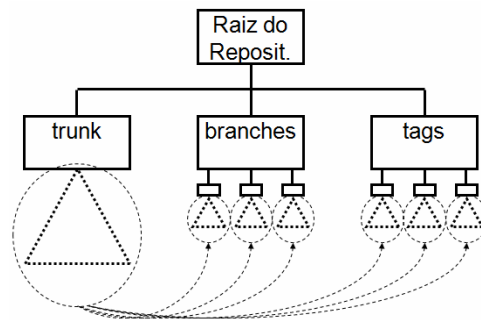


Figura 7 - Layout Repositório [De MARTINHO, 2008a]

2.3.4. Sugestões de Utilização do Subversion

Não é indicado trabalhar diretamente nos arquivos dos repositórios. Recomenda-se sempre criar uma área de trabalho local e posteriormente submeter os arquivos ao servidor, no qual os números de revisões do repositório sofrem alterações. O número de revisão é seqüencial, iniciando em um e incrementando a cada alteração dos documentos. Cada documento tem armazenado no repositório sua última situação e a situação em cada uma das revisões. Desta forma, será garantida a possibilidade de reversão a qualquer momento. Todo o conteúdo do repositório fica registrado, permitindo a identificação da revisão, data, hora e responsável pela revisão.

Para se obter um ótimo funcionamento da ferramenta Subversion, alinhada com o processo de Gerência de Configuração, é essencial a execução das melhores práticas de uso do Subversion, tais como:

- organizar o repositório com um layout adequado;
- conhecer o ciclo básico de trabalho:
 - fazer *Checkout* de uma cópia de trabalho do repositório/projeto (*checkout*),
 - atualizar a cópia de trabalho com as alterações mais recentes no repositório/projeto (*update*),
 - realizar alterações na cópia de trabalho (*add, delete, copy, move*),
 - examinar/verificar alterações entre a cópia de trabalho e o repositório/projeto (*status, diff, revert*),
 - consolidar alterações no repositório/projeto com a cópia de trabalho (*merge, resolved*),
 - atualizar o repositório/projeto com a cópia de trabalho (*commit*).
- fazer *commit* apenas de conjuntos de alterações lógicos;
- relacionar o sistema de controle de versões com o sistema de gestão de incidentes;
- usar mensagens descritivas no *commit* de merges;
- criar *branches* e *tags* para construção das *baseline*

3. Estudo de Caso

O presente estudo de caso possui objetivo de propor um processo de Gerência de Configuração de acordo com as necessidades da organização e que contemple todos os

resultados esperados da área de processos Gerência de Configuração do modelo de referência MPS.BR nível F.

3.1. Descrição da Organização

A empresa SWB Soluções Integradas localiza-se em Uberlândia – MG, Triângulo Mineiro, atua no mercado desde 1995 e possui clientes nos mais diversos segmentos.

Com o objetivo de melhorar seus processos de desenvolvimento de Software, a SWB já realizou outras iniciativas de melhoria, porém, muitas delas acabaram sendo abandonadas devido à falta de planejamento e alinhamento das necessidades da empresa, tornando essas iniciativas frustradas.

Em 2007, por meio de uma iniciativa do SEBRAE e Trisoft, a empresa aderiu ao projeto de Melhoria de Processos Rumo ao MPS.BR, com o objetivo de avaliar a sua unidade organizacional Fábrica de Software SWB no nível F do MPS.BR, tendo como data limite para a sua efetivação janeiro de 2009. Esse projeto, que encontra-se em andamento e conta com o apoio da diretoria da empresa e comprometimento dos colaboradores, utiliza como referência o MR-MPS, o Rational Unified Process (RUP) e o PMBOK na implementação das práticas do modelo.

No programa de melhoria de processos da organização foram estabelecidos vários objetivos, tais como:

- alcançar uma maior maturidade em desenvolvimento de software (após um ano de projeto);
- institucionalizar a cultura de processos (até o final do ano de 2009);
- reduzir em 15% os custos dos projetos (após a institucionalização dos processos);
- reduzir em 20% o re-trabalho nos projetos (após a institucionalização dos processos).

A organização SWB apresenta os seguintes resultados de acordo com sua análise SWOT: **(i) pontos fortes:** pessoas capacitadas e motivadas. A preocupação com a qualidade do produto final, e comprometimento dos colaboradores com o processo e satisfação do cliente, é uma premissa corporativa; **(ii) fraquezas:** falta de padrão institucionalizado, existente apenas para alguns projetos. A grande rotatividade de recursos gera altos custos de capacitação; dessa forma, as pessoas executam mais de uma função, gerando sobrecarga em uns e ociosidade em outros. O constante abandono dos projetos em processo é crítico; **(iii) ameaças:** comoditização da prestação de serviços de desenvolvimento, alta rotatividade de pessoal – sendo esses assimilados por grandes corporações – e concentração de projetos em poucos clientes; **(iv) oportunidades:** diferenciação por meio de avaliações de qualidade e definição de processos e papéis, levando o produto a adquirir maior qualidade [PORTO, 2008].

3.2. Processo de Gerência de Configuração

Atualmente, a organização não possui processo de Gerência de Configuração consolidado, somente utiliza a ferramenta de controle de versão. Este trabalho irá propor um processo de GCO com o apoio da ferramenta Subversion, objetivando atingir os resultados esperados do MPS.BR.

O processo de Gerência de Configuração é composto por várias atividades, no qual cada atividade possui os seus respectivos responsáveis por efetuar os passos para executar o processo. A figura 8 apresenta o processo de Gerência de Configuração

proposto para a empresa SWB. Ele é composto pelas atividades (1,2,4,5,6,7) que contêm os passos a serem executados com o propósito de gerar os produtos de trabalho do processo e pelo processo “Solicitar Mudança”(3) que contêm várias atividades.

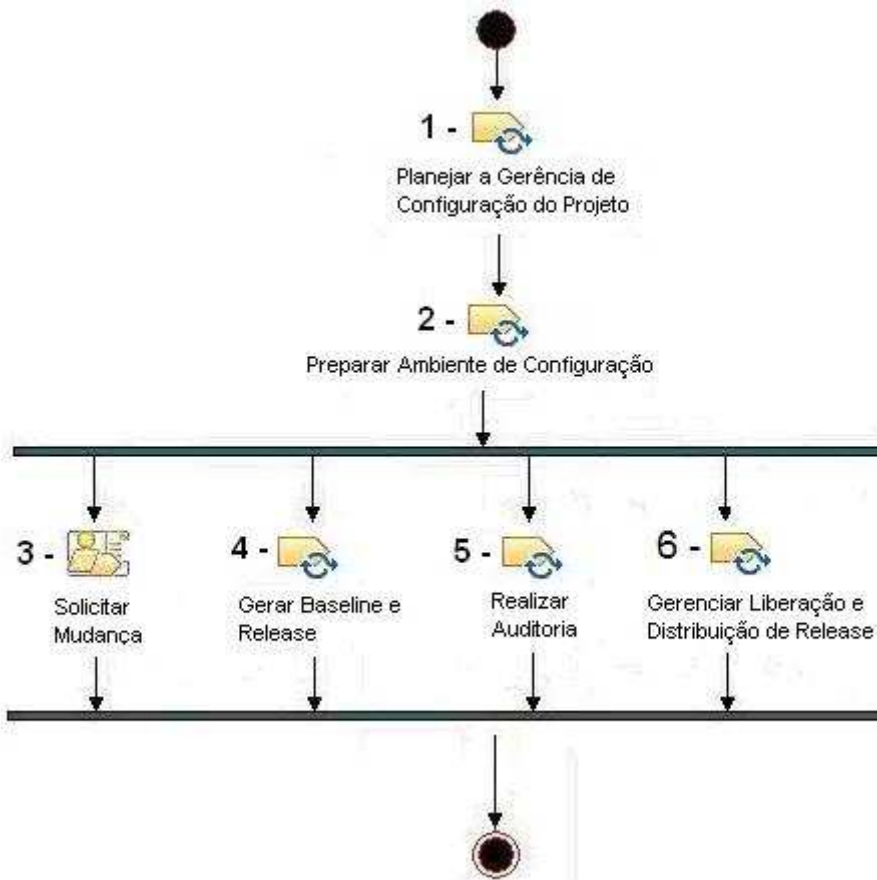


Figura 8 – Processo de Gerência de Configuração

Na SWB, ao ser criado um novo projeto no sistema interno de controle de projetos, automaticamente é enviado um e-mail para o Analista de Suporte solicitando a elaboração do repositório de dados, com sua respectiva estrutura de pastas e permissões aos envolvidos, pois toda a documentação gerada no decorrer do processo deve ser armazenada no repositório de dados da organização.

Papéis:

As funções desempenhadas por cada um dos sujeitos envolvidos nos processos podem ser assim definidas:

- **Gerente de Configuração:** responsável por disponibilizar o ambiente e a infraestrutura geral de Gerenciamento de Configuração para todos os envolvidos no projeto. O gerente deve assegurar a integridade dos dados, realizar auditoria dos produtos de trabalhos gerados no processo de Gerência de Configuração, facilitando a liberação do produto e as atividades de controle de mudanças e defeitos. O gerente de Configuração também é responsável por elaborar o Plano de Gerência de Configuração e acompanhar o andamento das atividades de GC com base nas solicitações de mudança.

- **Gerente de Projetos:** responsável por planejar e acompanhar os projetos frente ao cliente, pela contínua manutenção do “Pool de Recursos” (mobilização e desmobilização). O Gerente de projeto deve garantir, ainda, a comunicação entre os envolvidos no projeto.
- **Analista de Suporte:** responsável por instalar e configurar o software, o Hardware e os servidores da organização, desenhar a rede interna da empresa, garantir a integridade da segurança da informação, definir acesso às informações da organização e propor suporte ao usuário.

3.2.1. Planejar a Gerência de Configuração do Projeto

Objetivo da Atividade: Gerenciar a Configuração do projeto durante todo o seu ciclo de vida.

Papel Responsável: Gerente de Configuração.

Entrada: Políticas da Organização; Características do projeto a ser realizado.

Saída: “Plano de Gerência de Configuração”.

O plano de Gerenciamento de Configuração é um documento que descreve todas as atividades de Gerência de Configuração que serão executadas durante o ciclo de vida do projeto, e detalha os envolvidos e suas responsabilidades no projeto, recursos necessários, incluindo ferramentas de *software* e *Hardware* para garantir a integridade dos dados do projeto de *software*.

O plano de GC é elaborado na fase de planejamento do projeto, no qual deverá conter as informações necessárias para realizar de forma eficiente as tarefas de gerenciamento de Configuração. O projeto sofrerá modificações durante todo o seu ciclo de vida com o intuito de garantir o acompanhamento das atividades de Gerência de Configuração.

Seguem abaixo os passos a serem executados para realizar a atividade de Planejar Configuração do Projeto:

1 - Elaborar do Plano de Gerência de Configuração

O Plano de Gerência de Configuração é apresentado no Anexo I deste documento e é composto pelos seguintes itens:

- **Objetivos:** descrever os objetivos esperados no projeto com a Gerência de Configuração.
- **Papéis e Responsabilidades:** descrever os papéis e as responsabilidades de todos os envolvidos na atuação do processo GC.
- **Infra-Estrutura (Software, Hardware e Permissões Repositório de Dados):** identificar os itens de Configuração que serão utilizados no projeto e quais os envolvidos terão acesso a cada IC do projeto.

Os itens de configuração geralmente incluem produtos de trabalhos técnicos e gerenciais gerados no decorrer do projeto, entre eles planos diversos, especificação de requisitos, modelagem de dados e código.

Cabe ao gerente de projeto selecionar quais serão os itens de configuração que serão gerenciados no decorrer do projeto. Este procedimento deverá ser realizado a cada projeto da organização.

- **Baselines de projeto:** conjunto de itens de Configuração que se encontra em uma determinada versão e deverá estabelecer a periodicidade de construção das *baselines* do projeto.
- **Definições, Siglas e Abreviações:** descrever termos, siglas e abreviações particulares ao processo de Gerência de Configuração.
- **Registro Alterações:** registrar todas as alterações ocorridas durante o projeto.
- **Referências (Documentos Associados e Arquivos Associados):** registrar todas as referências que serão importantes para a execução do processo de Gerência de Configuração.
- **Auditorias de Gerência de Configuração:** registrar os itens de configurações e o respectivo resultado da auditoria do GC.
- **Padrões (Identificação de Artefatos, Identificação de *Baseline* e Versão de Artefatos):** instruções relevantes a serem aplicadas ao processo de Gerência de Configuração.

2 - Aprovar o plano de Configuração: Deverá ser aprovado pelo gerente de projetos, que deverá analisar todas as áreas que serão afetadas com a Gerência de Configuração e Mudanças.

3 - Revisar Plano de Gerência de Configuração: Deverá ser revisado no início de cada Marco do projeto. Após cada revisão do projeto, deverá ser incluído o status da gerência de Configuração no *Status Report* para que todos os envolvidos estejam cientes da situação atual do processo de GC.

Todos os itens de Configuração identificados no Plano de Gerência de Configuração, incluindo o Plano de GC, deverão ser armazenados no repositório de dados utilizando a ferramenta Subversion.

Após finalizar a elaboração do plano de Configuração, o Gerente deverá enviar um email ao analista de suporte, encaminhando o plano e solicitando a preparação do ambiente de Configuração.

3.2.2. Preparar Ambiente de Configuração

Objetivo da Atividade: Preparar o ambiente de Configuração para garantir a integração de todos os itens e ferramentas com o Subversion.

Papel Responsável: Analista de Suporte.

Entrada: “Plano de Gerência de Configuração” .

Saída: “Ambiente de Configuração”.

1 - Preparar Ambiente de Configuração: Cabe ao analista de suporte preparar o ambiente de desenvolvimento conforme o que for detalhado pelo Gerente de Configuração no Plano de Configuração. O Analista de suporte deverá: criar o repositório de dados no Subversion; criar a base para registro das solicitações de mudança; providenciar as ferramentas necessárias, seguindo as versões solicitadas; instalar as ferramentas no servidor apropriado e integrá-las com outras (sincronizar dados entre elas) e principalmente com o Subversion, caso haja possibilidade. O analista de suporte deverá verificar se todos os envolvidos no projeto estão com acessos corretos ao Subversion, conforme consta no Plano de Configuração.

3.2.3. Solicitar Mudanças

Objetivo da Atividade: Gerenciar e garantir que todas as mudanças propostas no projeto sejam avaliadas e aplicadas de maneira controlada e consistente para proporcionar a rastreabilidade entre os objetos afetados e minimizar o impacto, caso ocorra mudança no projeto.

Papel Responsável: Envolvidos no Projeto.

Entrada: “Solicitação de Mudança”.

Saída: “Mudança Aprovada ou Reprovada”.

O processo de Solicitação de mudança é auxiliar. Nele são executadas todas as atividades de solicitação de mudanças, se aplicando tanto à Gerência de Configuração, quanto às demais áreas de processos da organização (Gerência de Projetos, Gerência de Requisitos, Garantia da Qualidade, Mediação, Validação e Verificação). É importante ressaltar que todas as alterações solicitadas no decorrer do projeto deverão ser analisadas pelo Comitê de Controle de Mudanças (CCM), para verificar se a mudança é realmente necessária, qual a sua urgência e o seu impacto no seu ambiente (projetos e interfaces).

Toda mudança solicitada no decorrer do projeto deverá ser analisada com o auxílio da ferramenta Subversion, para verificar se todos os objetos afetados foram realmente atualizados. Para realizar essa verificação, deverão ser utilizados os comandos *Blame* e *Check for modifications*.

3.2.4. Gerar *Baseline* e Release

Objetivo da Atividade: Gerar uma *baseline* que contém um conjunto de produtos de trabalho resultante do projeto, resultando em uma versão estável do projeto.

Papel Responsável: Gerente de Configuração.

Entrada: “Plano de Gerência de Configuração”.

Saída: “*Baseline* do Projeto”.

1 - Planejar *Baseline*: O plano de Configuração deve inserir informações sobre as *baselines* como, por exemplo, o nome da *baseline*. Além disso: Quais serão as *baselines* estabelecidas no projeto? Com qual periodicidade serão geradas? Quais critérios serão estabelecidos para gerar a *baseline* e outras informações? Essas informações poderão ser inseridas no início ou nos marcos do projeto.

De acordo com o ciclo de vida dos projetos da organização, deve-se gerar no mínimo duas *baselines* no decorrer do projeto: término da “**Fase de Planejamento**” e término da “**Fase de Execução**”. Caso o GP ache necessário gerar mais *baselines* no decorrer do projeto, não há problema algum.

2 - Selecionar Itens de Configuração: Utilizar a ferramenta Subversion para identificar os itens de Configuração que irão compor a *Baseline*. Os itens de GC deverão estar nas versões corretas; para esta verificação será necessário utilizar o comando *ShowLog*.

3 - Criar *Baseline*: Para criar a *baseline* será necessário conhecer o conceito de Ramificação e Rotulação, utilizando a ferramenta Subversion.

Para criar a *baseline* será necessário verificar se os *Branches* estarão sendo utilizados corretamente pelos envolvidos no projeto. Antes de criar a *baseline* será preciso efetuar Merge das *Branches*, gerando as *Tags* que irão compô-la.

O próximo passo será unir os itens de Configuração e as *Tags*, gerando a *Baseline*, nomeando a mesma com a seguinte estrutura: < N° PROJETO_ANO>-<Nome_Projeto>-<Nome_Baseline>-<Vx.y>-<Data>, conforme descrito no Plano de Configuração.

Após gerar a *Baseline*, a mesma não poderá sofrer alteração em nenhum dos itens de Configuração, pois as alterações deverão ser efetuadas somente em *Trunk* e *Branches*.

3.2.5. Realizar Auditoria de Configuração

Objetivo da Atividade: Verificar se todos os itens de Configuração contidos na *baseline* estão conforme os requisitos e padrões especificados.

Papel Responsável: Gerente de Configuração.

Entrada: “Plano de Gerência de Configuração”.

Saída: “Plano de Gerência de Configuração”.

1 - Planejar Auditoria: O Gerente de Configuração deverá enviar e-mail ao Gerente de Projeto relatando que a *baseline* irá passar por auditorias e que, ao finalizá-las, o mesmo lhe enviará o relatório com o resultado para que ações corretivas sejam planejadas.

2 - Realizar Auditoria de Configuração: O Gerente de Configuração deverá acessar o repositório de dados para acessar a *baseline* e seus itens de Configuração. Ele deverá executar auditoria Funcional e Física para garantir que o produto que será liberado possua qualidade, conforme as expectativas do cliente. Formalmente estas auditorias são realizadas na seguinte ordem: Funcional e posteriormente a Física.

Auditoria Funcional: Essa auditoria possui a finalidade de verificar se todos os itens de Configuração estão consistentes conforme suas especificações, busca, também, a comprovação de que o IC foi devidamente testado e de que os testes evidenciam o atendimento de todos os seus requisitos. Essa auditoria é aplicada na especificação do produto.

Auditoria Física: Essa auditoria possui a finalidade de verifica se a *Baseline* está composta pela versão mais recente de todos os itens de Configuração especificados e se os padrões e procedimentos foram aplicados. Busca, também, verificar formato, consistência e completeza da documentação e sua compatibilidade com o produto desenvolvido. Essa auditoria é aplicada no produto em si.

Para acessar os arquivos no repositório de dados é necessário utilizar o comando *Checkout* na sua área de trabalho, assim, os documentos serão disponibilizados somente para leitura e para auditoria.

3 - Registrar resultado Auditoria: Cabe ao Gerente de Configuração registrar o resultado da auditoria no plano de Gerência de Configuração. Caso haja alguma não

conformidade/inconsistência/incorrekções, o mesmo deverá reporta-se ao Gerente de Projeto para planejar ações corretivas e acompanhar as ações até que as mesmas sejam finalizadas.

3.2.6. Gerenciar Liberação e Distribuição de Release

Objetivo da Atividade: Integração sistemática de um conjunto de itens de Configuração que serão disponibilizados para liberação e distribuição, tanto para ambientes internos quanto externos.

Papéis: Gerente de Configuração e Gerente de Projeto.

Entrada: “Plano de Gerência de Configuração”.

Saída: “Plano de Gerência de Configuração”.

1 - Gerenciar Configuração: O Gerente de Configuração é responsável por controlar todos os itens de Configuração do projeto, mantendo a integridade dos dados. O mesmo deve gerenciar o repositório de dados evitando retrabalho e que informações sensíveis sejam acessadas por pessoas não autorizadas.

Portanto, é necessário verificar se os envolvidos no projeto estão utilizando a ferramenta correta: *check-in, check-out, tags e branches*

2 - Planejar e Realizar Liberação e Distribuição: O Gerente de Configuração, junto ao Gerente de Projeto, deverá planejar o controle de liberação e distribuição de *build* e documentação do projeto. Para efetuar a liberação, deve gerar o *build* para o empacotamento dos itens de Configuração, garantindo a rastreabilidade das *baselines* contidas no pacote. É necessário estabelecer quem serão as pessoas que terão acesso e receberão esse pacote do *software*.

Para o Gerente de Configuração efetuar a liberação do pacote será necessário a autorização do Gerente de Projetos para que o mesmo acompanhe todo o histórico de distribuição do referido pacote.

Toda mudança solicitada após a liberação do release será entregue na próxima liberação do software.

3.3. Processo de Gerência de Configuração SWB x Resultados Esperados GCO do modelo MPS.BR

A tabela 4 relaciona as atividades do processo de Gerência de Configuração Proposto para SWB com os resultados esperados da área de processo de Gerência de Configuração do nível F de maturidade do modelo MPS.BR.

Resultados Esperado MPS.BR	Atividades do Processo de Gerência de Configuração SWB	Planejar Configuração Do Projeto	Preparar Ambiente de Configuração	Solicitação de Mudança	Gerar <i>Baseline</i> e Release	Realizar Auditoria	Gerenciar Liberação e Distribuição de Release
RAP 3: A execução do processo é planejada.							
GCO 1: Um Sistema de Gerência de Configuração é estabelecido e mantido.							
GCO 2: Os itens de Configuração são identificados.							
GCO 3: Os itens de Configuração, sujeitos a um controle formal, são colocados sob <i>baseline</i> .							
GCO 4: A situação dos itens de Configuração e das <i>baselines</i> é registrada ao longo do tempo e disponibilizada.							
GCO 5: As modificações em itens de Configuração são controladas e disponibilizadas.							
GCO 6: As auditorias de Configuração são realizadas objetivamente para assegurar que as <i>baselines</i> e os itens de Configuração estejam íntegros, completos e consistentes.							
GCO 7: O armazenamento, o manuseio e a liberação de itens de Configuração e <i>baselines</i> são controlados.							

Tabela 4 – Processo de Gerência de Configuração SBW X GCO do modelo MPS.BR

Pode-se observar que o processo proposto para a SWB, com o auxílio da ferramenta Subversion, contempla todos os resultados esperados do modelo MPS.BR, mas não se pode afirmar que o processo já está apto para ser avaliado, pois ainda falta contemplar os atributos de processos e os resultados dos atributos do processo do modelo MPS.BR.

4. Conclusão

A Gerência de Configuração vem se firmando como fundamental na implantação de Melhoria de Processos nas empresas, tornando-se a base para manter a integridade dos dados e evitar retrabalho, além de se relacionar com todos os outros da organização.

Para toda organização que deseje implantar o processo de Gerência de Configuração, sugere-se adotar uma ferramenta de controle de versão para tornar o trabalho mais eficiente. Nesse sentido, este trabalho demonstrou como é possível adaptar-se à ferramenta Subversion e ao processo de Gerência de Configuração, tornando as atividades mais simples e menos burocráticas.

Para o estudo de caso apresentado, embora as modificações na organização ainda estejam em fase inicial, à ferramenta Subversion já foi implantada e ótimos resultados foram obtidos, pois os projetos ficaram mais organizados, utilizando a estrutura de pastas que proporcionou trabalho corporativo e em paralelo sem, contudo,

impactar no trabalho de outros colaboradores. A ferramenta também proporcionou que o trabalho na modalidade *Home Office* se tornasse mais eficiente.

A situação anterior na organização era bastante crítica, pois os desenvolvedores utilizavam a própria máquina para armazenar seus trabalhos, sem fazer uso do conceito de *Tag* e *Branches*. Quando iniciavam a integração dos desenvolvimentos de software, vários erros ocorriam, resultando em um enorme retrabalho. Outro problema é que, quando o desenvolvedor era desligado da organização, o seu trabalho não ficava acessível ao restante da equipe, resultando em um grande desgaste entre todos os envolvidos no projeto.

Portanto, pode-se observar a importância e o sucesso da Gerência de Configuração com o auxílio do Subversion. Por outro lado, implantar simplesmente uma ferramenta não traz melhorias significativas, mas encoraja a adoção da prática correta para melhorar o processo de desenvolvimento.

O ganho de produtividade e qualidade só é obtido com a implantação do processo, com treinamento específico para utilização completa e correta da ferramenta e do processo de Gerência de Configuração: caso contrário, há apenas um aumento na burocracia na organização.

Devido ao fato da implementação do processo de Gerência de Configuração estar ainda no início, não foi possível até o momento realizar uma avaliação formal, mas existe o interesse da empresa em realizá-la tão logo seja possível. .

5. Referências Bibliográficas

CMU/SEI, (2006) - *Software Engineering Institute* - SEI. CMMI® for Development, Version 1.2, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, agosto, 2006. <http://www.sei.cmu.edu>. - Acessado em: 03/08/2008.

Goulart, Consultoria (2007) – *Consultoria em Serviços de TI*. <http://goulartconsult.com/welcome.html> - Acessado em: 19/11/2008.

MPS.BR (2007a) *SOFTEX* – Associação para Promoção da Excelência do Software Brasileiro. MPS.BR – Guia Geral (Versão 1.2), http://www.softex.br/mpsbr/guias/MPS.BR_Guia_Geral_V1.2.pdf. - Acessado em: 03/08/2008.

MPS.BR (2007b) *SOFTEX* – Associação para Promoção da Excelência do Software Brasileiro. MPS.B - Guia de Implementação – Parte 2: Nível F (Versão 1.1), http://www.softex.br/mpsbr/guias/MPS.BR_Guia_de_Implementacao_Parte_2_v1.1.pdf. - Acessado em: 03/08/2008.

MURTA, Leonardo Gresta Paulino. (2006) “Gerência de Configuração no Desenvolvimento Baseado em Componentes”. – Tese de Doutorado Rio de Janeiro, 2006. (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 2006).

PORTO, Josiane Brietzke; Pereira, Anna Carolina; Pohren, Juliana. (2008) *Proposta de Melhoria de Processos da SWB Soluções Integradas usando o MR-MPS e a abordagem PRO2PI*. Trabalho da Disciplina ISO 15504 do curso de Pós-Graduação Lato Sensu em Melhoria de Processo de Software da Universidade Federal de Lavras - UFLA, Lavras, 2008

De MARTINHO, Wagner Roberto (2008a) – *Ferramenta de Gerência de Configuração Subversion*. - Centro de Pesquisas Renato Archer – DMPS (Divisão de Melhoria de Processos de Software).

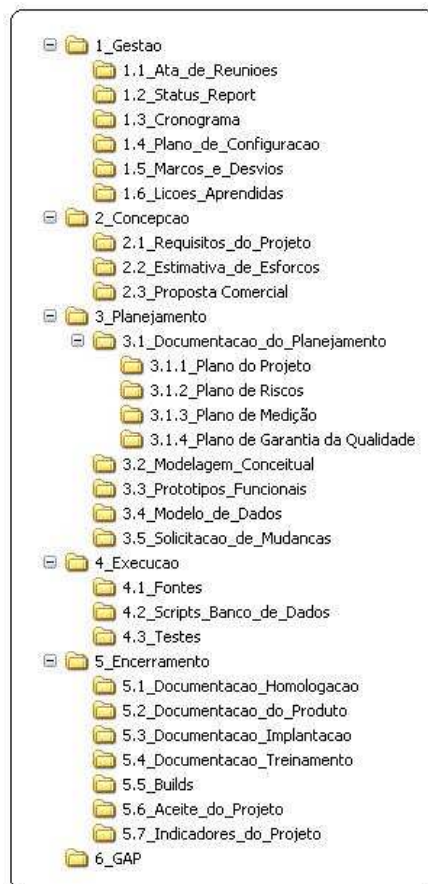
De MARTINHO, Wagner Roberto (2008b) – *Gerência de Configuração - Integridade dos Produtos do Desenvolvimento*. - Centro de Pesquisas Renato Archer – DMPS (Divisão de Melhoria de Processos de Software).

ANEXO A: Políticas de Gerência de Configuração

A SWB, com o intuito de melhorar seus processos e garantir a qualidade de seus produtos, propõe que todos os projetos da organização, com tempo de execução acima de 300hs, devem utilizar a metodologia definido na organização. Caso o projeto seja menor que 300hs, o processo deverá ser customizado. O objetivo de documentar as políticas da organização é apresentar publicamente o que ela espera em relação a todas as atividades descritas no processo padrão da empresa. Essas políticas organizacionais foram construídas de acordo com os aspectos da engenharia de software, seguindo os modelos do MPS.BR e CMMI.

Segue abaixo diretrizes definidas pela organização:

- **GC.Nº01-v1.0:** Para todo e qualquer projeto da organização, o mesmo deverá ser armazenado no repositório de dados SWB, utilizando a ferramenta Tortoise/SVN.
- **GC.Nº02-v1.0:** Para todo e qualquer projeto, todas as documentações deverão ser adicionadas em suas respectivas pastas, contidas na estrutura padrão da organização, conforme a imagem abaixo:



- **GC.Nº03-v1.0:** Toda as *pastas* adicionadas na estrutura deverão seguir o mesmo padrão de nomenclatura e numeração.
- **GC.Nº04-v1.0:** Todos os *documentos* da organização deverão ser armazenados nas respectivas pastas seguindo a nomenclatura padrão.
- **GC.Nº05-v1.0:** Todas as *Baselines* deverão ser criadas pelo Gerente de Projetos e/ou Líder Técnico, utilizando a ferramenta Subversion e seguindo a nomenclatura padrão.
- **GC.Nº06-v1.0:** Todo e qualquer projeto deverá conter o *Plano de Configuração*, seguindo todas as normas e procedimentos contidos no processo de Gerência de Configuração.
- **GC.Nº07-v1.0:** Todo *Backup* da organização deverá ser realizado diariamente, semanalmente e mensalmente.
- **GC.Nº08-v1.0:** Para todo encerramento de projeto da organização o Gerente de Projeto será o responsável por verificar a ocorrência de arquivos travados (*check-outlock* de arquivo) no repositório de dados.
- **GC.Nº09-v1.0:** O Gerente de Projeto será responsável por gerenciar a garantia do projeto (por 3 meses). Após o vencimento da garantia, deverá ser aberto um chamado através do *Help Desk*, integrado ao analista de suporte, solicitando o bloqueio do diretório do projeto, ficando liberado somente acesso para leitura aos arquivos contidos no mesmo.

ANEXO B: Plano de Gerência de Configuração

Plano de Gerência de Configuração



Uberlândia - MG

Revisões do Documento

(Atenção! Só acrescente uma linha no quadro a seguir, indicando uma nova versão).

Versão	Data	Responsável	Observações
1.0			Criação do documento

Aprovações do Documento

Data	Responsável	Observações

Índice

Configurações do Projeto.....	34
Objetivos / Escopo.....	34
Envolvidos no Projeto	34
Infra - Estrutura	34
Software.....	34
Hardware e serviços necessários	35
Permissões Repositório do Projeto	35
Baselines de Projeto.....	36
Itens de Configuração.....	36
Baseline.001	36
Baseline.002	36
Definições, Siglas e Abreviações.....	37
Registro Alterações.....	37
Alterações	37
Referências	37
Documentos Associados.....	38
Arquivos Associados	38
Auditorias da Gerência de Configuração.....	38
Auditoria.001	38
Auditoria.002.....	38
Padrões	39
Identificação dos Artefatos	39
Identificação de Baseline.....	39
Versão de Artefatos	39

Cliente:	<nome_empresa_cliente>	Nº. Projeto:	<provável_n°.projeto>
Nome Projeto:	<provável_nome_projeto>		
Analista responsável:	<analista_SWB_responsável>	Data:	<data_elaboração_doc>

Configurações do Projeto

Objetivos / Escopo
<p>Este documento descreve o Plano de Gerência de Configuração do projeto/produto <nome do projeto>, definindo as ações de Configuração que serão adotadas.</p> <p>Os objetivos são:</p> <ul style="list-style-type: none"> Descrever a organização, nomenclaturas e regras de versionamento para o projeto. Descrever os itens do projeto que ficarão sob Gerência de Configuração. <p><Caso haja mais objetivos, continuar listando-os. Apagar esta instrução após o preenchimento. ></p> <p>< Uma breve descrição do escopo deste Plano de Gerenciamento de Configuração, ou seja, detalhar toda a infra-estrutura que será utilizada durante o projeto.></p>

Envolvidos no Projeto		
Nome	Papel/ Função	Responsabilidade
<nome dos envolvidos no projeto>	<Papel e funções que ocupam no projeto>	<email do envolvido no projeto>

Configuração			
Gerência de Mudança			
Gerência de Projetos			
Gerência de Requisitos			
Métricas			
Sistema Operacional			
Ferramentas de Teste			
<Outros>			

<Caso houver mais hardwares ou serviços, continuar listando-os. Apagar esta instrução após o preenchimento.>

Hardware e serviços necessários		
Tipo de Ferramenta	Situação	Quantidades
Estações para desenvolvimento		
Acesso via VPN no ambiente de homologação do cliente		
Área para armazenamento dos dados do projeto no SVN		
<outros>		

Permissões Repositório do Projeto	
Diretórios	Nome(s) e Tipo de Permissão
1 - Gestão 1.1 - Atas de Reuniões 1.2 - Status Report 1.3 - Cronograma 1.4 - Plano de Configuração 1.5 - Marcos e Desvios 1.6 - Lições Aprendidas	<Nome (Tipo Permissão) > L : Leitura, E : Escrita. <i>Exemplo:</i> <ul style="list-style-type: none"> • <i>Papel - Desenvolvedor (E/L)</i> • <i>Papel - Testador (L)</i>
2 - Concepção 2.1 - Visão do Projeto 2.2 - Estimativa de Esforços 2.3 - Propostas	
3 - Planejamento 3.1 - Documentação do Planejamento 3.1.1 - Plano do Projeto 3.1.2 - Plano de Riscos 3.1.3 - Plano de Medição 3.1.4 - Plano de Garantia de Qualidade 3.2 - Modelagem Conceitual 3.3 - Protótipos Funcionais 3.4 - Modelo de Dados	

3.5 - Solicitação de Mudanças	
4 - Execução 4.1 - Fontes 4.2 - Scripts Banco de Dados 4.3 - Testes	
5 - Encerramento 5.1 - Documentação Homologação 5.2 - Documentação do Produto 5.3 - Documentação Implantação 5.4 - Documentação Treinamento 5.5 - <i>Builds</i> 5.6 - Aceite do Projeto 5.7 - Indicadores do Projeto	

Baselines de Projeto

A seguir, lista de todas as *baselines* que serão gerados no projeto.

Itens de Configuração				
IC	Descrição (itens)	ID <i>Baseline</i>	Responsável	Versão
• Espec. de Requisitos		BI_001		
• Espec. de Requisitos Código-fonte				

Baseline.001					
Nome <i>Baseline</i>					
Instante de Estabelec.:					
Critério de Estabelec.:					
Conteúdo	Situação	Data	Versão	Responsável	ID Auditoria
Iteração_001	em elaboração	09/10/08	1.0	Maria	Aud_001

Baseline.002					
Nome <i>Baseline</i>					
Instante de Estabelec.:					
Critério de Estabelec.:					
Conteúdo	Situação	Data	Versão	Responsável	ID Auditoria

Definições, Siglas e Abreviações.

A tabela a seguir lista todas as definições, siglas e abreviações usados neste documento.

[Descrever as abreviações, definições e siglas relevantes ao documento em ordem alfabética.]

Sigla	Descrição
Baseline	Conjunto de especificações ou produtos de trabalho que foi formalmente revisada. Serve de referência para desenvolvimento futuro e pode ser alterada somente através de procedimentos de controle de alteração.
GCO	Gerência de Configuração.
CCB	<i>Change Control Board</i> - Comitê de Controle de Configuração.
SQA	<i>Software Quality Assurance</i> - Papel que executará a auditoria.

Registro Alterações

A tabela a seguir lista todas as alterações solicitadas via *Help Desk* para a área de suporte da organização.

Alterações						
ID	Nome	Descrição	Data	Responsável	Solicitante	Status
GC.001						
GC.002						
GC.003						
GC.004						

Referências

As tabelas a seguirem listam todos os documentos e arquivos relacionados com Configuração no projeto.

Documentos Associados			
Nome Documento	Versão	Data	Responsável

Arquivos Associados	
Nome Arquivo	Descrição

Auditorias da Gerência de Configuração

As tabelas a seguir listam todas as auditorias realizadas nas *Baselines* do projeto.

Auditoria.001				
Nome <i>Baseline</i>:	P2027_2008-Faturamento Martins- <i>Baseline</i> P2027_001-2.0-2008/10/07			
Tipo de Auditoria:	Funcional: <input checked="" type="checkbox"/>		Física: <input type="checkbox"/>	
Responsável Auditoria:	José da Silva			
Nome IC	Data	Responsável	Não Conforme	Observação
Tela de Cadastro	09/10/08	Maria	x	A tela não esta conforme o requisito xxx
Plano do Projeto	09/10/08	João		Auditado - OK
Resultado da Auditoria				
<Descrever o resultado da auditoria e evidenciar os mais críticos encontrados na <i>Baseline</i> >				

Auditoria.002				
Nome <i>Baseline</i>:	<Nome <i>Baseline</i> >			
Tipo de Auditoria:	Funcional: <input type="checkbox"/>		Física: <input type="checkbox"/>	
Responsável Auditoria:	<Nome do Responsável pela auditoria>			
Nome IC	Data	Responsável	Não Conforme	Observação
<Nome do Item>	<Data>	<Nome responsável>	<Marque com X se for "SIM" >	<Descrever observação importante>
Resultado da Auditoria				
<Descrever o resultado da auditoria e evidenciar os mais críticos encontrados na <i>Baseline</i> >				

Padrões

Identificação dos Artefatos

Todos os itens de Configuração (exceto código fonte) devem ser identificados baseados na nomenclatura descrita a seguir:

<N° PROJETO_ANO>-<Nome_Projeto>-<Nome_Documento>

<N° PROJETO_ANO>:	Número do Projeto conforme a base de administração de projetos.
<Nome_Projeto>:	Nome do projeto conforme a base de administração de projetos.
<Nome_Documento>:	Nome do documento

Identificação de Baseline

Toda *baseline* deve ser identificada baseada na nomenclatura descrita a seguir:

< N° PROJETO_ANO>-<Nome_Projeto>-<Nome_Baseline>-<Vx.y>-<Data>

<N° PROJETO_ANO>:	Número do Projeto conforme a base de administração de projetos.
<Nome_Projeto>:	Nome do projeto conforme a base de administração de projetos.
<Nome_Documento>:	Nome do documento
<Vx.y>:	Versão da <i>baseline</i>
<Data>:	Data em que a <i>baseline</i> foi gerada.

Versão de Artefatos

Essa seção descreve padrões que definem o versionamento dos artefatos. Todos os artefatos devem ter um número de versão segundo o padrão descrito a seguir: X.Y

X:	É um número decimal que representa uma versão final do artefato;
Y:	É um número decimal que representa um draft da versão X do artefato.

O número de versão dos artefatos muda de acordo com as regras descritas a seguir:

- a primeira versão do artefato deve ser 0.1;
- a cada versão do artefato, o valor Y deve ser incrementado;
- após cada aprovação do artefato, a versão X deve ser incrementada de uma unidade e o valor Y retorna para 0, sendo assim, gerada uma nova versão oficial.