

O USO DE SOFTWARE PARA CONTROLE DE VERSÕES COMO FERRAMENTA DE APOIO À PRODUÇÃO DE MATERIAIS INSTRUCIONAIS DA EDUCAÇÃO A DISTÂNCIA - EaD

VITÓRIA – ES – 04 2010

José Mário Costa Junior – Ifes - icjunior@ifes.edu.br

Saymon Castro de Souza, Esp – Ifes – saymon@ifes.edu.br

Vanessa Battestin Nunes, Msc, MPS.BR – Ifes – vanessa@ifes.edu.br

Danielli Veiga Carneiro, Msc – Ifes – danielli@ifes.edu.br

Elton Siqueira Moura, Msc, MBA – Ifes – elton@ifes.edu.br

Categoria (Métodos e Tecnologias)

Setor Educacional (Educação Universitária)

Natureza (Modelos de Planejamento)

Classe (Relatos de Experiência)

RESUMO

A utilização dos sistemas de controle de versões pressupõe entender como funciona sua estrutura, de forma a padronizar os conceitos envolvidos, facilitando até mesmo o treinamento da equipe. A maioria dos aspectos básicos se aplica aos diversos tipos de ferramentas encontrados no mercado. Faz-se então importante entender a lógica de funcionamento desses sistemas, o uso do software de controle de versões Subversion e o aplicativo de interface gráfica auxiliar TortoiseSVN - ambos ferramentas livres - para gerenciar versões de materiais e demais objetos de aprendizagem desenvolvidos no setor de produção de material do CEAD/Ifes. Os principais resultados foram a catalogação dos materiais e objetos de aprendizagem e a diminuição do retrabalho e erros no setor de produção de materiais didáticos.

Palavras Chave: Controle de Versão, Produção de Material, Gerência de Configuração, Auditoria.

1. Introdução

Produzir materiais para a Educação a Distância (EaD) é uma atividade complexa que exige alto grau de organização das pessoas envolvidas no processo. Geralmente acontecem muitas interações e modificações nos materiais até que ele seja considerado apropriado para utilização. Isso é verdade especialmente na produção dos materiais impressos, os quais exigem análises mais aprofundadas e acabam originando várias outras requisições, como produção de imagens, revisão de texto, diagramação, dentre outros.

O Centro de Educação a Distância do Instituto Federal do Espírito Santo (CEAD/Ifes) definiu um processo de produção para os materiais impressos. A Figura 1 apresenta as etapas principais desse processo.

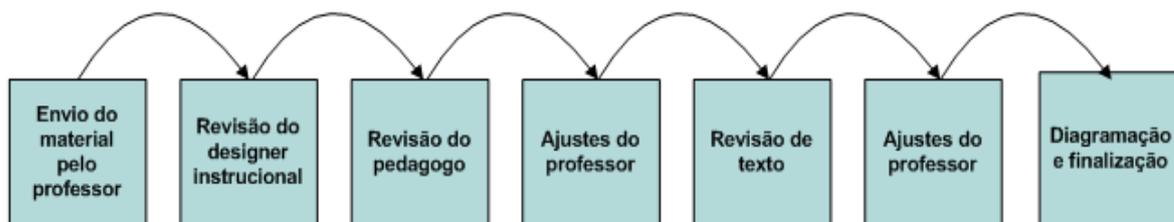


Figura 1: Parte do processo da produção de material impresso

Após elaborar o material didático, baseando-se no mapa de atividades, o professor conteudista envia sua primeira versão para o designer instrucional. O designer acrescenta vários comentários e sugestões. Depois, o pedagogo acrescenta as observações pertinentes a sua área. O professor então altera o material, levando em consideração as ideias do designer instrucional e do pedagogo. Posteriormente o material é enviado ao revisor de texto para correções e depois o professor realiza os últimos ajustes. Logo após, a versão do arquivo é enviada para diagramação por empresa terceirizada e finalizado.

Percebe-se pelo processo que o material é modificado muitas vezes e por pessoas desempenhando diversos papéis. Controlar as versões é algo realmente desafiador. São muitas trocas de e-mails e muitos documentos a armazenar. Caso alguma versão se perca, o resultado final fica seriamente comprometido. Por outro lado, guardar vários documentos de um mesmo material onera os servidores de arquivo e gera desorganização, demandando grande tempo de organização de

diretórios pela equipe. Esses problemas podem gerar vários transtornos como: incompatibilidade entre os grupos de trabalho, inconsistências, retrabalho, atraso na entrega e insatisfação do aluno [1].

Este trabalho propõe o uso do software de controle de versões *Subversion* e o aplicativo de interface gráfica auxiliar *TortoiseSVN* - ambas ferramentas livres - para gerenciar versões de materiais e demais objetos de aprendizagem desenvolvidos no setor de produção de material do CEAD/Ifes.

2. Ferramentas de controle de versões

A utilização dos sistemas de controle de versões pressupõe entender como funciona sua estrutura, de forma a padronizar os conceitos envolvidos, facilitando até mesmo o treinamento da equipe. A maioria dos aspectos básicos se aplica aos diversos tipos de ferramentas encontrados no mercado. Faz-se então importante entender a lógica de funcionamento desses sistemas.

Os programas de versionamento trabalham com o conceito de repositório. Um repositório é um lugar central onde ficam armazenadas as cópias principais de todas as versões dos arquivos de trabalho [2]. Por isso, geralmente há uma máquina servidora para armazená-lo. Ela deve ser segura e cópias de segurança dos arquivos devem ser realizadas periodicamente.

O usuário não trabalha diretamente nos arquivos do repositório. Ao invés disso, usa uma área/cópia de trabalho que contém a cópia dos arquivos do projeto e que é monitorada para identificar as mudanças realizadas (controle de modificações). Essa área é individual e isolada das demais áreas de trabalho.

Quando alterações são feitas localmente na cópia de trabalho, é preciso atualizar as versões no repositório. O contrário também é verdade: para se trabalhar com segurança, é necessário ter na cópia de trabalho local a versão mais atual dos arquivos do repositório.

A sincronização entre a área de trabalho e o repositório é feita através dos comandos *commit* (ou *check in*) e *update* (ou *check out*) [5]. O *commit* envia um pacote contendo uma ou mais modificações feitas na área de trabalho (origem) ao

repositório (destino). O update faz o inverso, isto é, envia as modificações contidas no repositório (origem) para a área de trabalho (destino).

Cada commit gera uma nova revisão no repositório, contendo as modificações feitas, data e autor (controle de versões). Uma revisão funciona como uma "foto" de todos os arquivos e diretórios em um determinado momento da evolução do projeto [6]. As "fotos" antigas são mantidas e podem ser recuperadas e analisadas sempre que desejado (relato da configuração). O conjunto dessas revisões é justamente o histórico do projeto.

A figura 2 resume o esquema de funcionamento dos sistemas de controle de versões.

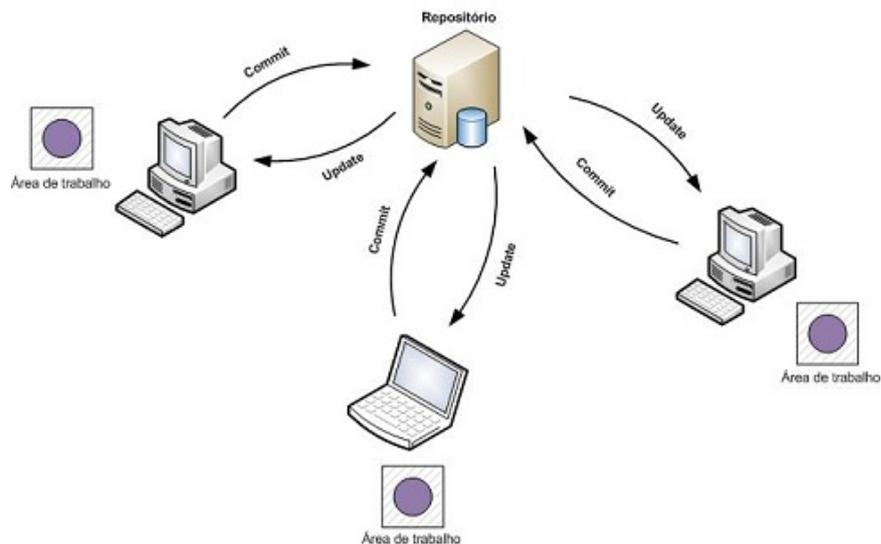


Figura 2: funcionamento de um sistema de controle de versões

A tríade histórico, colaboração e variações no desenvolvimento do material são as principais diretrizes para que haja efetivamente um controle de versão. O histórico registra toda a evolução do trabalho, cada alteração sobre cada arquivo. Com essas informações sabe-se quem fez o que, quando e onde. Além disso, permite reconstruir uma revisão específica do arquivo sempre que desejado. A colaboração possibilita que várias pessoas trabalhem em paralelo sobre os mesmo arquivos. As variações da diretriz do desenvolvimento do material mantém linhas diferentes de evolução do mesmo, permitindo o acesso às várias versões de um determinado material.

3. Utilização do *Subversion* e do *TortoiseSVN* no CEAD/lfes

Com o propósito de organizar sua produção de material didático e objetos de aprendizagem, o CEAD / lfes optou pela utilização do *TortoiseSVN* [4] e do *Subversion* [3]; [7] para controle de versões, por serem ferramentas livres, muito utilizadas e documentadas.

O *TortoiseSVN* (<http://tortoisesvn.net/>) é um aplicativo cliente para o software *Subversion* (<http://subversion.apache.org/>). O *TortoiseSVN* auxilia o usuário no controle de versão dos seus arquivos e diretórios, por meio de uma interface simples e intuitiva. Ele é executado nas máquinas cliente. O *Subversion* é quem realiza o controle de versão e é executado no servidor. A figura 3 mostra a interação entre o *Subversion* e o *TortoiseSVN*.

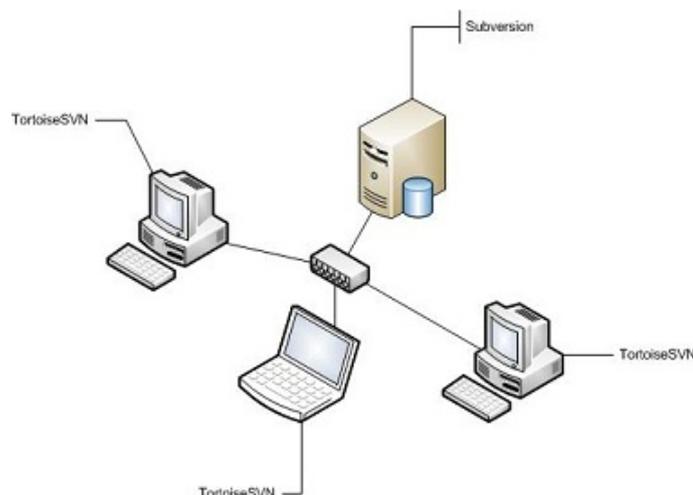


Figura 3: - interação entre o *Subversion* e *TortoiseSVN*

Após instalado, o *TortoiseSVN* se integra ao sistema operacional Microsoft Windows, facilitando a utilização. A figura 4 mostra os ícones de utilização no menu *pop-up* quando se clica com o botão direito em qualquer pasta do computador.

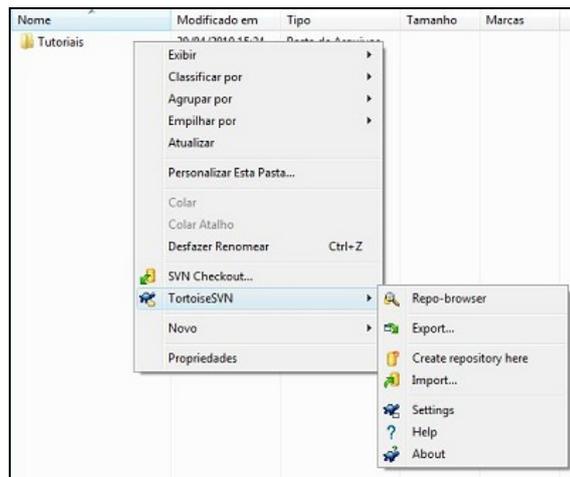


Figura 4: Funcionalidades do TortoiseSVN integradas ao sistema operacional Windows.

Para a utilização do sistema, foi necessária a reorganização de diretórios, para que o repositório estivesse organizado e com entendimento claro. Optou-se por criar um diretório para cada curso oferecido pelo CEAD/Ifes. Dentro do diretório dos cursos organizou-se de acordo com a grade de períodos e disciplinas. A figura 5 apresenta uma visão geral do repositório de arquivos.

Pode-se observar que o curso "Tecnologia em Análise e Desenvolvimento de Sistemas" é distribuído em seis períodos. Cada período possui suas disciplinas. Na figura é possível visualizar os diretórios para o terceiro período do curso. Já a figura 6 apresenta a organização para uma disciplina, para a qual são geradas as subdivisões de acordo com os tipos de objetos produzidos.

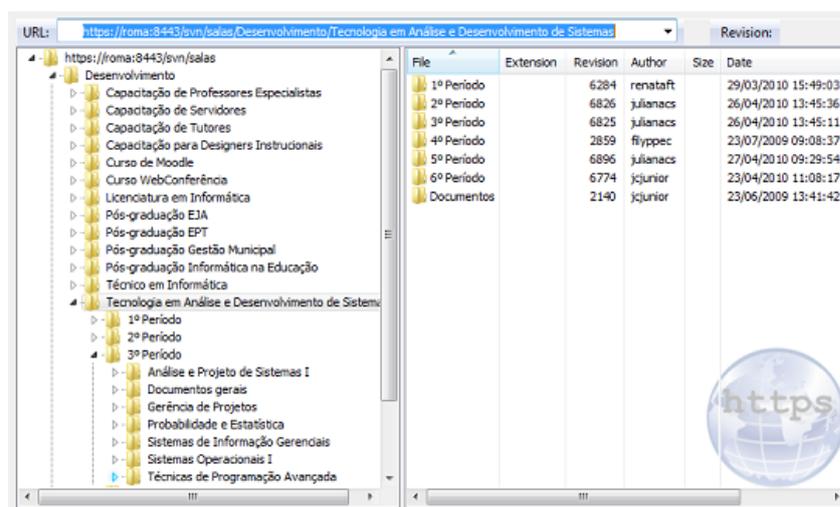


Figura 5: Organização do repositório de arquivos.

File	Extension	Revision	Author	Size	Date
📁 Animações		478	jcjunior		27/05/2009 15:20:44
📁 Documentos		3512	jcjunior		03/11/2009 13:43:59
📁 Imagens		3217	julianacs		05/10/2009 14:21:06
📁 Tutoriais		474	jcjunior		27/05/2009 15:20:03
📁 Vídeos		474	jcjunior		27/05/2009 15:20:03

Figura 6: Organização de diretórios por disciplina.

Para alterar qualquer arquivo do repositório, é necessário criar localmente a área de trabalho. Para fazer isso, o *TortoiseSVN* permite que o usuário escolha o local em sua máquina que abrigará as cópias de arquivos do repositório. A figura 7 exemplifica esse processo:

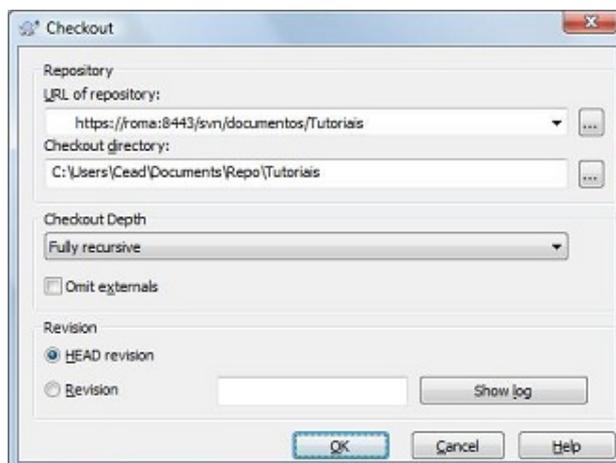


Figura 7: Baixando arquivos do repositório para máquina cliente.

Após este processo, o diretório local de trabalho ficará ligado ao repositório. Os arquivos com o ícone verde sobre o ícone tradicional do sistema operacional estão atualizados com o repositório, ou seja, são versões mais atuais. Já os ícones vermelhos sinalizam que o arquivo está alterado localmente, mas não no repositório. Há então a necessidade de executar o comando *commit*, atualizando o arquivo no repositório e criando uma nova versão do mesmo.

A figura 8 apresenta um exemplo de arquivos com indicações de controle de versão.

Nome	Modificado em
Tecnologias da Informação e Comunicação-1.doc	30/04/2010 12:44
Tecnologia da Informação e Comunicação.indd	30/04/2010 12:38
Tecnologia da Informação e Comunicação_WEB.pdf	30/04/2010 12:38
Link	30/04/2010 12:38

Figura 8: Arquivos locais com indicação de sincronização com repositório.

Um recurso muito útil do *TortoiseSVN* é o relatório de revisões. É possível visualizar todas as alterações realizadas em um determinado arquivo, quem as realizou e quando. Pode-se também obter uma versão mais antiga para nova edição ou simples conferência. Na figura 9 há um *log* (registro) de alterações em um arquivo que está em sua segunda versão.

Revision	Actions	Author	Date	Message
6981		jcjunior	12:49:57, sexta-feira, 30 de abril de 2010	
4957		afreitas	10:37:44, segunda-feira, 22 de fevereiro de 2010	

Action	Path
Modified	/Desenvolvimento/Capacitação de Professores Especialistas/Tecnologias da Informação e Comunicação/Documen

Showing 2 revision(s), from revision 4957 to revision 6981 - 1 revision(s) selected.

Hide unrelated changed paths

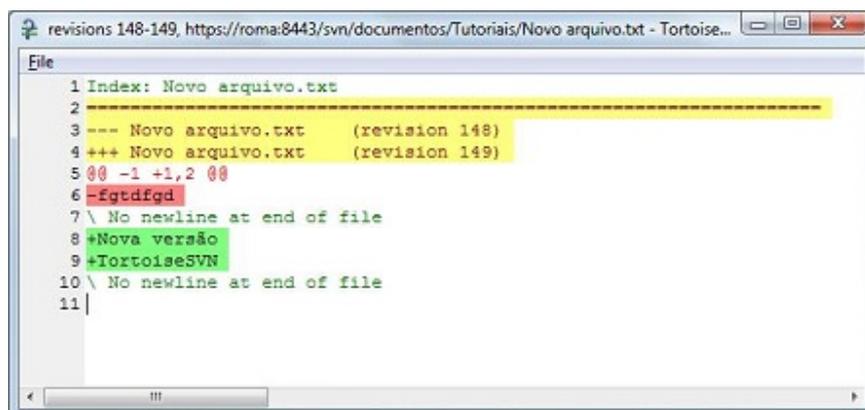
Stop on copy/rename

Include merged revisions

Buttons: Show All, Next 100, Refresh, Statistics, Help, OK

Figura 9: Relatório de revisões de um arquivo sob controle de versão.

Para análises ainda mais aprofundadas de diferenças entre versões, o *TortoiseSVN* permite comparar o conteúdo de arquivos, destacando as diferenças entre um e outro. Assim, é possível saber exatamente quais foram as modificações efetuadas por um usuário em um arquivo. É importante ressaltar que este recurso, apesar de muito útil, não é aplicável a todos os tipos de arquivo. A figura 10 mostra um arquivo de texto sob análise, com cores destacando cada alteração.



```
revisions 148-149, https://roma:8443/svn/documentos/Tutoriais/Novo arquivo.txt - Tortoise...
File
1 Index: Novo arquivo.txt
2 -----
3 --- Novo arquivo.txt      (revision 148)
4 +++ Novo arquivo.txt      (revision 149)
5 @@ -1 +1,2 @@
6 -fgtdfgd
7 \ No newline at end of file
8 +Nova versão
9 +TortoiseSVN
10 \ No newline at end of file
11 |
```

Figura 10: Análise de revisões em um arquivo de texto.

Nesta seção foram apresentadas apenas algumas funcionalidades principais do *TortoiseSVN* como interface gráfica do *Subversion*. Existem muitas outras que permitem um controle apurado de versões. No entanto, as citadas são as mais utilizadas nas tarefas cotidianas da produção do CEAD/Ifes e já proporcionam grandes facilidades na organização do material didático.

4. Resultados e Trabalhos Futuros

A implantação dos softwares *Subversion* e *TortoiseSVN* permitiu a catalogação dos materiais e objetos de aprendizagem produzidos no CEAD/Ifes. Isso provocou a diminuição do retrabalho da equipe de produção de material, já que o repositório é sempre consultado a cada nova requisição de confecção de materiais. Se já houver material que atenda à solicitação, ele pode ser reutilizado ou até mesmo adaptado, o que minimiza o esforço de uma nova construção.

A taxa de erros provocados por trabalhos em versões inadequadas diminuiu drasticamente. Esse foi um fato relevante, já que o envio de uma versão inadequada para impressão causa um problema praticamente incorrigível em termos de custo.

Atualmente cerca de vinte e cinco funcionários do setor de produção de material estão utilizando o controle de versões. Há de se destacar que houveram dificuldades iniciais no uso do *TortoiseSVN* por alguns membros da equipe. No entanto, após desenvolvimento e disponibilização de manuais de instrução e da

realização de treinamentos o uso do sistema se estabilizou sem maiores problemas.

Como trabalhos futuros, destaca-se a necessidade de integração com o servidor de autenticação de usuários. Assim, quando a pessoa se autenticasse na tela inicial do sistema operacional, não seria necessário se autenticar novamente para utilizar o repositório do controle de versões, o que acontece hoje. Ou seja, a base de usuários seria unificada.

Há ainda a previsão de expansão do uso do controle de versões para todo o CEAD/Ifes e não apenas na produção de materiais didáticos, possibilitando também o acompanhamento de alterações nos documentos administrativos e outros artefatos.

Referências

- [1] NUNES, Vanessa Battestin. **Integrando Gerência de Configuração de Software, Documentação e Gerência de Conhecimento em um Ambiente de Desenvolvimento de Software**. Dissertação (Mestrado em Informática), Universidade Federal do Espírito Santo, Vitória, 2005.
- [2] MASON, Mike. ***Pragmatic version control using Subversion***. LLC, 1.ed, 2005.
- [3] Subversion. **Site oficial do Subversion** <<http://subversion.apache.org/>>. Acesso em: abril de 2010.
- [4] TortoiseSVN. **Site oficial do TortoiseSVN** <<http://tortoisesvn.org>>. Acesso em: abril de 2010.
- [5] RAMOS, Erivan de S., FREITAS, Rejane C. **Análise Comparativa de Sistemas de Controle de Versões Baseados em Código Aberto** <<http://www.infobrasil.inf.br/userfiles/28-05-S1-1-68581-Analise%20Comparativa.pdf>>. Acesso em: abril 2010.
- [6] SILVA, Sandra R. Q. M. **Controle de versões - um apoio à edição colaborativa na Web**. Dissertação (Mestrado em Ciência da Computação), Instituto de Ciências Matemáticas e de Computação – IMCP-USP, São Carlos, 2005.
- [7] TROMBETTA, André; KOROGI, Genilson; PERALTA, Karine. **Sistema de Controle de Versões**. Faculdade de Informática – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS).