

Um Suporte ao Controle de Versões na *Web*

M arinalva D ias Soares

Renata Pontin de M attos Fortes

{*msoares,renata*}@ *icmc.sc.usp.br*

Departamento de Ciências de Computação e Estatística - ICMC - USP
Caixa Postal 668 - CEP 13560-970 - São Carlos, SP

Resumo

Em um mundo computacional em constante evolução, a *Web* se apresenta com o um ambiente caracterizado pelo desenvolvimento acelerado de suas informações. Além de requerer muitas modificações e com extrema frequência, os desenvolvedores enfrentam dificuldades em suas atividades, quando envolvem muitas pessoas trabalhando em paralelo no desenvolvimento de uma aplicação (ou de um conjunto de páginas). Portanto, necessitam de suporte para controle de configuração com vistas à garantia de qualidade básica. Este trabalho apresenta a ferramenta *VersionWeb*, que foi desenvolvida para proporcionar aos desenvolvedores um modo fácil de controle de versões de páginas da *Web*, através da própria *Web*.

Palavras-chave: evolução de informações na *Web*, controle de versões, gerenciamento de configuração, Engenharia de Software na *Web*.

1. Introdução

Sistemas e aplicações baseados na *Web* disponibilizam rapidamente uma diversidade de conteúdos de informação e funcionalidades para acesso aos usuários, a uma grande população de usuários finais. As aplicações *Web* evoluem continuamente e, de forma semelhante às aplicações de software convencional, sofrem inevitáveis mudanças, com o agravante de que não é raro algumas delas (especialmente seu conteúdo) serem atualizadas com enorme frequência (em curto espaço de tempo). Torna-se imprescindível, portanto, um suporte para controle de atualizações na *Web* com vistas à garantia de qualidade básica (Pressman 2001).

No ambiente *Web* de desenvolvimento, os desenvolvedores se deparam com dificuldades nas suas atividades quando estas envolvem muitas pessoas trabalhando em paralelo na produção de uma página, de um conjunto de páginas ou mesmo de uma aplicação. Pois, em geral, eles trabalham independentemente em suas próprias cópias, acarretando assim problemas relacionados com a integração de todo o trabalho, ou seja, enfrentam perdas e/ou sobreposição de informações (Somerville et al., 1998). Os usuários internautas¹, por sua vez, também sofrem as consequências da falta de gerenciamento de versões das informações contidas na *Web*, pois frequentemente se surpreendem ao visitar uma página e percebem que esta já não possui o mesmo conteúdo ou até mesmo o que ela não existe mais, devido à rápida, mas natural, evolução das informações na *Web*.

As características de sistemas baseados na *Web* influenciam os processos de Engenharia na *Web*. A evolução contínua requer um modelo de processo incremental e interativo de produção de novas versões de maneira muito rápida. A Engenharia na *Web* visa um processo que seja utilizado para criar aplicações *Web* de alta qualidade, e em bora a Engenharia na *Web*

¹ Internauta é o nome pelo qual, comumente, o usuário que navega pela WWW é chamado.

não seja cópia exata da Engenharia de Software, possui muitos dos conceitos e princípios fundamentais da Engenharia de Software, enfatizando as mesmas técnicas e atividades de gerenciamento (Pressman 2001). No entanto, devido à rápida popularidade da *Web*, observa-se que seus desenvolvedores ainda não se preocupam com as atividades de gerenciamento; simplesmente pensam que seu mundo é realmente diferente e que as abordagens da Engenharia de Software convencional não se aplicam.

A atividade abrangente, aplicada em todo o processo de Engenharia de Software, que é responsável por gerenciar a evolução de sistemas de software grandes e complexos, denomina-se Gerenciamento de Configuração de Software (GCS) (Pressman 2001). Introduzir os conceitos e ferramentas de GCS para desenvolvedores que não possuem familiaridade com os mesmos é imprescindível para que a produtividade incorpore garantia de qualidade básica de forma disciplinada e sistemática, mas isso em geral, infelizmente não é tarefa fácil.

Existem sistemas que auxiliam os desenvolvedores no controle de versões e localização de alterações de documentos HTML, com o V-Web (Somerville et al. 1998) e AIDE (Douglas et al., 1998), por exemplo. Ambos utilizam o RCS (*Revision Control System*) (Ritchy 1985) para o controle de versões e localização das alterações e scripts CGI² para interação entre os usuários e as funcionalidades do sistema. Já a WebRC (*Configuration Management for a Cooperation Tool*) foi a primeira ferramenta baseada em cooperação projetada para a *Web* com suporte ao gerenciamento de configuração. Essa ferramenta auxilia a cooperação de grupos na *Web* com o conceito de *workspace* de cooperação (Fröhlich, Nedjl. 1997).

Embora tais sistemas não estivessem disponíveis para utilização, foi possível obter uma visão geral das características relacionadas às tarefas de controle de versão na *Web* que se apresentam na literatura. Dessa forma, a ferramenta *VersionWeb* foi projetada para oferecer recursos de controle de versões por meio da ativação do CVS (*Concurrent Versions System*) (CVS 2001) a partir da *Web*.

Este artigo apresenta a ferramenta *VersionWeb* (Soares, Fortes 1999; Soares 2000; Soares et al. 2000), desenvolvida com os objetivos específicos de: **(a)** apoiar os desenvolvedores no trabalho colaborativo de páginas ou aplicações *Web*, permitindo fazer controle de versões das informações e evitando perda ou sobreposição; **(b)** possibilitar aos internautas visualizar diferentes versões de uma mesma página e localizar as diferenças entre elas durante a navegação. Na Seção 2 é descrita a *VersionWeb* e na Seção 3 são apresentados resultados de testes iniciais que foram realizados com a mesma e as conclusões.

2. Características de *VersionWeb* para Controle de Versões na *Web*

A ferramenta *VersionWeb* visa auxiliar o trabalho cooperativo realizado por desenvolvedores de páginas *Web* que podem estar situados em locais diferentes e remotos. Ela permite que os usuários internautas localizem alterações em uma página, recuperem versões anteriores e visualizem as diferenças entre elas. Sua característica é a combinação de um bom sistema de controle de versões com uma interface amigavelmente acessível, por meio de *browsers Web*.

Para apoiar sua principal característica funcional, a *VersionWeb* utiliza o CVS (*Concurrent Versions System*) para o controle de versões dos arquivos sob seu gerenciamento. CVS é um sistema de controle de versões de arquivos (CVS 2001), atualmente muito utilizado por organizações e desenvolvedores de software, que permite:

- a armazenagem e recuperação de versões anteriores de arquivos (texto e binários) eficientemente;

² *Common Gateway Interface*

- suporte do histórico de todas as alterações feitas nos arquivos ou árvore de diretório que ele gerencia;
- exibição das diferenças de conteúdo entre duas versões de um arquivo;
- desenvolvimento paralelo, permitindo que mais de uma pessoa trabalhe em um mesmo arquivo ao mesmo tempo;
- acesso confiável às árvores de diretórios de *hosts* remotos usando protocolos Internet.
- adição, remoção e alteração de arquivos e diretórios do **repositório**³;
- associação de nomes simbólicos a um conjunto específico de revisões;
- execução em várias plataformas (Unix, Windows 95, Windows NT, Macintosh e VM S).

A comunicação entre a *VersionWeb* e o CVS foi implementada por meio de *scripts* CGI que ficam residentes na máquina servidora, juntamente com o CVS. A interface com o usuário da *VersionWeb* foi desenvolvida a partir de formulários HTML, oferecendo assim, as operações básicas de controle de versão para execução dos comandos CVS, uma vez que este é totalmente orientado a linha de comando. Na **Figura 1** é apresentada a interface da *VersionWeb* para gerenciamento de arquivos, exibindo todo o conteúdo (arquivos e diretórios) de repositório CVS sob gerenciamento e os botões que disponibilizam os comandos CVS.

Para a realização das tarefas de controle de versão através da *VersionWeb*, o cenário de interação foi definido de maneira que os usuários interajam com o CVS através de um "Módulo de Interface de aplicação" da *VersionWeb* – MIV (baseado em formulários HTML) e a seguir submetam pedidos ao "Módulo de Comunicação com CVS" – MCC (*scripts* CGI residentes fisicamente na mesma máquina que o servidor de *Web* e o CVS). Esses pedidos, geralmente, são comandos CVS a serem executados, com o *checkout*, *commit*, etc. De maneira geral, o processamento dos pedidos ocorre da seguinte forma:

- usuário acessa formulários HTML de MIV para requisição de serviços *VersionWeb*;
- servidor de *Web* recebe os pedidos do usuário e os passa ao MCC;
- MCC recebe e decodifica os pedidos do usuário;
- MCC que recebeu o pedido faz uma chamada ao CVS (também fisicamente localizado na mesma máquina) com os comandos a serem executados;
- CVS executa os comandos e retorna o resultado ao MCC que o requisitou;
- MCC envia os resultados ao servidor de *Web*;
- servidor de *Web* devolve os resultados ao cliente, ou seja, ao usuário de MIV que iniciou a interação com a ferramenta.

A partir do funcionamento básico oferecido pela *VersionWeb*, descrito acima, fica transparente aos usuários o uso do CVS e seus comandos para controle de versões. Adicionalmente, *VersionWeb* possibilita a distinção de três tipos de usuários: **(a) administradores**, responsáveis por gerenciar todos os usuários da ferramenta; **(b) autores**, autorizados a manipular os arquivos que estão sob o controle de versão, arquivos estes que residem no repositório CVS no servidor; e **(c) internautas e/ou grupos específicos de internautas**, que são usuários com permissões para apenas visualizar as versões de uma página, as diferenças entre elas, receber notificação de novas versões da página disponíveis e enviar comentários.

³ Em geral, as ferramentas de controle de versões possuem um *repositório* (uma hierarquia de diretórios com um) onde são armazenadas as cópias mestre dos arquivos. A partir dessas cópias mestre são reconstruídas todas as árvores de versões que são geradas através das modificações feitas.

Para o acesso às suas funcionalidades principais, a *VersionWeb* requer a autenticação do usuário, que é obrigatória para autores e administradores. Neste artigo, para efeito de apresentação da *VersionWeb*, são descritas a seguir somente as funcionalidades de gerenciamento de arquivos relacionadas ao controle de versões acessíveis pelos autores.

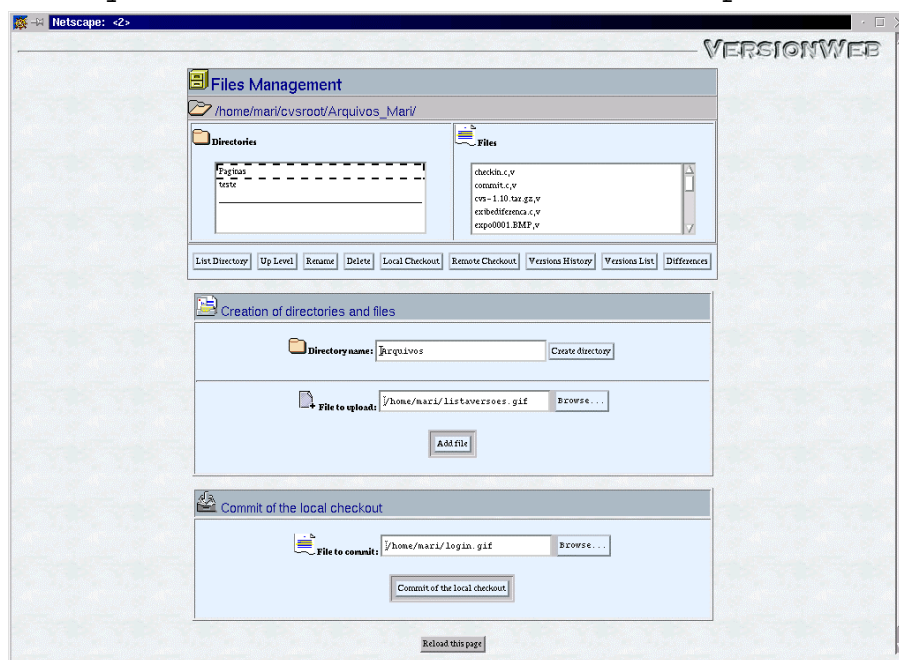


Figura 1 - Interface principal de gerenciamento de arquivos da *VersionWeb*

A partir da tela de interação apresentada na **Figura 1** estão disponíveis as operações básicas de manipulação de arquivos e as operações básicas usuais do CVS para controle de versões, que estão descritas na **Tabela 1**.

Tabela 1 – Lista de operações da interface principal de gerenciamento de arquivos da *VersionWeb*

Operação	Descrição Funcional
List Directory /Up Level	Desce/Sobe um nível na árvore de diretório
Renome	Renomeia diretório/arquivo
Delete	Remove diretório/arquivo
Remote Checkout	Faz <i>checkout</i> (da versão corrente) de um arquivo abrindo uma janela para edição do conteúdo do arquivo (texto) no próprio <i>browser</i> .
LocalCheckout	Faz <i>checkout</i> (da versão corrente) de um arquivo/diretório para a máquina do usuário.
Versions History	Apresenta a lista de versões de um arquivo com seus respectivos autores, data, hora e com entários para visualização.
Versions List	Apresenta a lista de todas as versões e <i>branches</i> de um arquivo e operações para fazer <i>checkout</i> (remoto e local) de versões anteriores, <i>commit</i> , gerar <i>branches</i> e <i>revert</i> alterações entre versões.
Diffs	Exibe um formulário com opções para visualizar as diferenças entre duas versões do arquivo através de cores.
Create directory	Cria um diretório no repositório CVS;
Add file	Faz <i>upload</i> de arquivos para o repositório CVS.
Commit of the local checkout	Faz o <i>commit</i> de um arquivo e gera uma versão subsequente àquela da qual foi feito o <i>checkout</i> local.
Reload this page	Atualiza o conteúdo do repositório CVS na interface.

A seleção de quais operações básicas deveriam estar disponibilizadas, para controle de versões através da *VersionWeb*, foi realizada por meio de um estudo de caso com sessões práticas utilizando-se o CVS. E para a execução com pleta de algum as operações disponíveis, de gerenciamento de arquivos da *VersionWeb* (**Figura 1**), foram desenvolvidas sete (7) outras telas para interação, prosseguindo a aquisição de informações necessárias aos comandos CVS.

Na **Figura 2** encontra-se a interface que permite a recuperação de versões específicas (o autor pode fazer *checkout* remoto e/ou local) e a geração de *branches*. A partir dessa interface, o autor pode reverter alterações feitas entre duas versões de um arquivo.

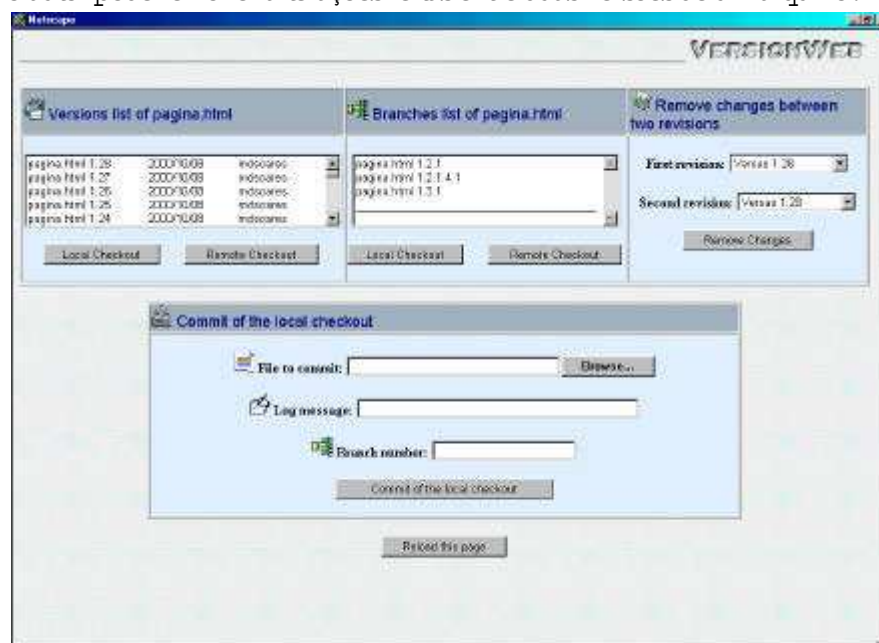


Figura 2 – Interface com a lista de versões e *branches* de um arquivo

O outro objetivo da *VersionWeb*, de permitir que os internautas, durante a navegação, tivessem acesso às informações que alguma vez estiveram disponíveis, foi desenvolvido por meio da recuperação de versões anteriores dessas páginas e localização de suas alterações. Para tanto, o administrador pode escolher entre permitir que todos os internautas, ou um grupo específico destes, tenham acesso às versões da página. Se o administrador optar por restringir o acesso a apenas grupos específicos de internautas, *VersionWeb* irá requerer que o internauta faça parte de grupos já cadastrados e que passe pelo processo de autenticação de usuários (Soares et al. 2000). Para a viabilização desse objetivo, a página que o internauta estiver visitando, que estiver sob controle de versão, conterá um *link* para a ferramenta *VersionWeb*.

3. Conclusões

Os objetivos inicialmente determinados para *VersionWeb* foram alcançados, uma vez que ela: (a) apóia os desenvolvedores no trabalho colaborativo de documentos ou aplicações, permitindo fazer controle de versões das informações sem que haja perda ou sobreposição; e (b) possibilita aos internautas visualizarem diferentes versões de uma mesma página e localizar as diferenças entre elas durante a navegação.

Além disso, *VersionWeb* contribuiu no sentido de: disponibilizar uma ferramenta para fazer controle de versões via *Web*; disponibilizar prontamente o conteúdo atual para os demais desenvolvedores de uma equipe, pois à medida que novas versões são geradas, se tornam imediatamente disponíveis e acessíveis (por meio de *browser* e conexão à Internet); e proporcionar aos desenvolvedores um mecanismo ágil e adequado para o controle de versões, sem requerer conhecimentos em detalhes do processo de GCS.

Visando obter as primeiras impressões sobre a interação dos usuários com a *VersionWeb* e sobre o grau de importância desse tipo de recurso para desenvolvedores, foram feitos alguns testes de usabilidade (dado um conjunto pré-definido de tarefas de controle de versão em um formulário para serem realizadas na *VersionWeb*) com 20 usuários do próprio Departamento de Computação do ICMC-USP. Esses usuários, em geral, possuíam nível de conhecimento computacional alto, mas com diferentes níveis de conhecimento sobre controle de versões. Eram estudantes de Computação, sendo 70% em nível de Mestrado, 15% em nível de Graduação e 15% em nível de Doutorado. Esse teste de usabilidade, embora tenha sido aplicado a um número relativamente pequeno de usuários, foi muito importante, pois mostrou que as funcionalidades da *VersionWeb* atenderam às tarefas de gerenciamento de alterações das informações de forma satisfatória.

Além disso, a *VersionWeb* tem sido adotada, desde janeiro de 2001, por grupos de pesquisadores do ICMC-USP para ensinar e disseminar o controle de versões. A ferramenta encontra-se disponível para download em "<http://versionweb.sourceforge.net/>".

Uma outra linha de pesquisas que está se iniciando, a partir do desenvolvimento de *VersionWeb*, é a de proporcionar que suas funcionalidades sejam disponibilizadas para reuso por aplicações *Web* em geral.

Referências Bibliográficas

- (CVS 2001) CVS, "Concurrent Versions Systems", disponível *on-line* em : <http://home.cvs.org>. Visitado em maio de 2001.
- (Douglas et al. 1998) F. Douglas, T. Ball, Y.-F. Chen, E. Koutsofios, "The AT&T Internet Difference Engine: Tracking and viewing changes on the web", *World Wide Web*, Volume 1, N.º.1, 27-44, 1998.
- (Fröhlich, Nejl 1997) P. Fröhlich, W. Nejl "WebRC - Configuration Management for a Cooperation Tool" *Lecture Notes in Computer Science*, 1235, Springer, 175-185, 1997
- (Pressman 2001) R. S. Pressman, *Software Engineering - A Practitioner's Approach*. McGraw Hill, Fifth edition, 2001.
- (Soares, Fortes 1999) M. D. Soares, R. P. M. Fortes "Gerenciamento de Controle de Versões de Páginas Web" In: IV Workshop de Teses em Engenharia de Software. *Anais Florianópolis - SC, Brasil*, 12 outubro de 1999, p.33-37.
- (Soares 2000) M. D. Soares "Gerenciamento de Versões de Páginas Web" Dissertação de mestrado do ICMC-USP, 2000. 114p.
- (Soares et al. 2000) M. D. Soares, R. P. M. Fortes, D. A. Moreira "VersionWeb: A Tool for Helping Web Pages Version Control" In: International Conference on Internet Multimedia Systems and Applications (IMSAA 2000), *Proceedings*. Las Vegas USA, November 2000. pp.275-280.
- (Somerville et al. 1998) I. Somerville, T. Rodden, P. Rayson, A. Kirby, A. Dix "Supporting information evolution on the WWW", *World Wide Web*, vol.1, n.1, 45-54, 1998.
- (Tichy 1985) W. F. Tichy, "RCS - A system for version control", *Software Practice and Experience*, Vol.15, N.º 7, 637-654, 1985.