



CONTROLE DE VERSÃO

Vocabulário

Profª. Ana Flávia

Vocabulário do Controle de Versão

- São termos úteis e independentemente de qualquer sistema de controle de versão em particular: eles são os substantivos e verbos básicos de uma colaboração em rede.

Vocabulário do Controle de Versão

- ***commit***

- Realizar uma mudança no projeto; mais formalmente, armazenar uma mudança no banco de dados de controle de versão de uma forma que ela possa ser incorporada em versões futuras do projeto.

Vocabulário do Controle de Versão

- ***commit***

O "Commit" pode ser usado com um verbo ou como um substantivo. Como um substantivo, ele é essencialmente um sinônimo de "mudança".

Por exemplo: "Eu acabei de comitar uma correção para corrigir o bug de crash do servidor que o pessoal tem reportado no Mac OS X. Jay, você pode por favor revisar o commit e verificar se eu não estou usando o alocador indevidamente?"

Vocabulário do Controle de Versão

- **log message**

- Um pequeno comentário anexado a cada *commit*, descrevendo a natureza e o propósito do *commit*.
- Estão entre os documentos mais importantes no projeto: elas são a ponte entre a linguagem altamente técnica das mudanças individuais de um código e uma linguagem mais direcionada aos usuários de funcionalidades, correções de bugs, e progresso do projeto.

Vocabulário do Controle de Versão

- **update**

- Solicitar que as mudanças dos demais (commits) sejam incorporadas em sua cópia local do projeto; ou seja, deixar a sua cópia "atualizada".
- Esta é uma operação muito comum; a maioria dos desenvolvedores atualizam seus códigos diversas vezes por dia, de maneira que eles sabem que estão executando algo muito próximo do que os demais estão trabalhando, e caso eles encontrem um bug, eles tem certeza que ele ainda não foi corrigido.

Por exemplo: "Olá, eu notei que o código de indexação está sempre desprezando o último byte. Este é um novo bug?" "Sim, mas ele foi corrigido na semana passada—atualize seu código e ele deve desaparecer."

Vocabulário do Controle de Versão

- **repositório**

- Um banco de dados no qual mudanças são armazenadas.
- Alguns sistemas de controle de versão são centralizados: há um único repositório mestre que armazena todas as mudanças feitas no projeto. Outros são descentralizados: cada desenvolvedor tem seu próprio repositório, e alterações podem ser trocadas entre repositórios arbitrariamente.

Vocabulário do Controle de Versão

- **repositório**

- O sistema de controle de versão mantém controle das dependências entre as alterações, e quando é hora de lançar uma versão, um conjunto particular de alterações é aprovado para aquele lançamento.

Vocabulário do Controle de Versão

- ***checkout***

- O processo de se obter uma cópia do projeto a partir de um repositório. Um checkout geralmente gera uma árvore de diretórios chamada de cópia de trabalho (*working copy*), a partir da qual alterações podem ser comitadas de volta ao repositório original.
- Em alguns sistemas de controle de versão descentralizados, cada cópia é em si mesma um repositório, e mudanças podem ser extraídas de (ou enviadas a) qualquer repositório que esteja disposto a aceitá-las.

Vocabulário do Controle de Versão

- **Cópia de Trabalho**

- Uma árvore de diretórios privada do desenvolvedor contendo o código fonte do projeto, e possivelmente suas páginas da web ou outros documentos.
- Uma cópia de trabalho também contém um pouco de metadata gerenciado pelo sistema de controle de versão, que conta de que repositório vem a cópia de trabalho, que "revisões" (veja abaixo) dos arquivos estão presentes, etc.
- Geralmente, cada desenvolvedor tem sua própria cópia de trabalho, na qual ele faz e testa mudanças, e da qual ele envia.

Vocabulário do Controle de Versão

- ***revision* (revisão), *change* (mudança), *changeset* (grupo de mudança)**
 - Uma "*revision*" é geralmente uma encarnação específica de um arquivo ou diretório em particular.
 - Por exemplo, se um projeto iniciar na revisão 6 do arquivo F, e então alguém efetua uma mudança e o *commit* do arquivo F, isto produz a revisão 7 de F.
 - Alguns sistemas também usam "*revision*", "*change*", or "*changeset*" para se referir a um conjunto de modificações em que se é efetuado o *commit* uma única vez, como uma unidade conceitual.

Vocabulário do Controle de Versão

- ***diff***

- Uma representação textual de uma mudança.
- Uma *diff* mostra quais linhas foram modificadas e como elas foram modificadas, e adiciona algumas linhas próximas para contextualizar em ambos os lados.
- Um desenvolvedor que já está familiarizado com o código pode geralmente confrontar uma *diff* com o código e identificar do que se trata a mudança, e até identificar possíveis bugs.

Vocabulário do Controle de Versão

- ***tag*** (ou etiqueta)

- Uma informação para uma coleção em particular de arquivos para revisões especificadas.
- *Tags* são usadas geralmente para preservar *snapshots* interessantes do projetos.
 - Por exemplo, uma *tag* é gerada para cada release pública, exatamente com os conjuntos de arquivos/revisões que compreendem aquela release.

Vocabulário do Controle de Versão

- ***branch* (ou ramo)**

- Uma cópia do projeto sob o controle de versão, porém isolada de uma maneira que as mudanças realizadas no *branch* não afetem o resto do projeto e vice-versa, exceto quando as mudanças são deliberadamente mescladas ("*merge*") de um lado para o outro.
- *Branches* (ou ramos) também são conhecidas como "linhas de desenvolvimento". Mesmo quando um projeto não possui *branches* explícitas, o desenvolvimento ainda é considerado como sendo executado no "*branch* principal", também conhecido como "linha principal" ou "*trunk*".

Vocabulário do Controle de Versão

- ***branch (ou ramo)***

- Branches oferecem uma maneira isolada de diferentes linhas de desenvolvimento entre eles.
- Por exemplo, um *branch* pode ser usado para um desenvolvimento experimental que poderia desestabilizar muito a linha principal (ou *trunk*).
- Assim como uma *branch* pode ser usada como um local para estabilizar uma nova release. Durante o processo de release, o desenvolvimento regular continua sem interrupções no *branch* principal do repositório; enquanto isso, na *branch* da release, nenhuma mudança é permitida com exceção daquelas que são aprovadas pelos gerentes de releases. Desta forma, disponibilizar uma release não precisa interferir no andamento dos trabalhos de desenvolvimento.

Vocabulário do Controle de Versão

- **merge (a.k.a. port) (ou junção)**
 - Mover uma mudança de um *branch* para o outro. Isso inclui um merge da linha principal com algum *branch*, ou vice versa. Na verdade, estes são os tipos mais comuns de merge; é raro transportar mudanças entre dois *branches* não-principais. Veja “Singularidade de Informação” para mais detalhes sobre este tipo de merge.

Vocabulário do Controle de Versão

- "Merge" tem um segundo significado: é o que o sistema de controle de versão faz quando identifica que duas pessoas mudaram o mesmo arquivo mas sem interferências. Como as duas mudanças não interferem uma com a outra quando uma pessoa atualiza sua cópia do arquivo (já contendo suas próprias mudanças), as mudanças da outra pessoa serão automaticamente agregadas.
- Isto é muito comum, especialmente em projetos que contém muitas pessoas trabalhando no mesmo código. Quando duas modificações diferentes se interferem, o resultado é um "conflito".

Vocabulário do Controle de Versão

- ***conflict* (ou conflito)**
 - É o que acontece quando duas pessoas tentam modificar a mesma parte do código de maneiras diferentes.
 - Todos os sistemas de controle de versão detectam conflitos automaticamente, e notificam pelo menos um humano de que suas mudanças conflitam com as de outra pessoa. O humano notificado deve então resolver o conflito, e comunicar a resolução para o sistema de controle de versão.

Vocabulário do Controle de Versão

- ***lock* (ou bloqueio)**

- Um jeito de declarar a intenção de modificar exclusivamente um arquivo ou diretório.
- Por exemplo, "Não consigo commitar nenhuma mudança para as páginas web no momento. Parece que Alfred as bloqueou enquanto conserta as imagens de fundo." Nem todos os sistemas de controle de versão suportam a habilidade de bloquear, e daqueles que suportam, nem todos requerem que a habilidade seja usada. Isso se dá porque desenvolvimento paralelo é a regra, e bloquear pessoas de acessar um arquivo é contrário à este ideal.