

## 2 Operações em controle de versões

Para melhor compreensão do funcionamento dos sistemas de controle de versão, é importante conhecer as principais operações que os envolvem. A seguir, são apresentados os elementos mais utilizados pela maioria dos controles de versionamento:

- **Commit (Checkin):** criação de uma nova versão do projeto.
- **Checkout:** recuperação de uma versão específica do projeto ou arquivo.
- **Revert:** possibilita ao desenvolvedor descartar as mudanças realizadas em estação local, recuperando a mesma versão do repositório.
- **Diff:** garante a possibilidade de comparação do arquivo na estação local com qualquer outra versão do repositório.
- **Delete:** permite a exclusão de um arquivo do repositório. Quando as demais estações de trabalho realizarem um *update*, o arquivo será efetivamente excluído do repositório.
- **Lock:** possibilita o travamento de determinado arquivo, de forma que nenhum outro usuário o modifique.



### Saiba mais

Uma **tag** é simplesmente uma representação mais simples de se lembrar, para seres humanos, de uma determinada versão do código.

Confira, a seguir, alguns conceitos relevantes com relação aos controles de versões.

- **Repositório de versões:** armazenamento de todas as versões dos arquivos sob o controle versões.
- **Repositório de versões centralizado:** cada estação de trabalho local contém apenas uma versão específica da árvore de versões do repositório central, e todos os usuários realizam as operações de controle de versões nesse repositório.
- **Repositório de versões distribuído:** cada estação local possui um repositório acoplado, de modo que o usuário possui um repositório próprio para realizar o controle de versões. As operações realizadas sobre os arquivos

são feitas no repositório local do usuário, e operações específicas dos repositórios distribuídos são utilizadas para sincronizar repositórios diferentes.

- **Árvore de revisões ou de versões:** estrutura lógica que mapeia todas as versões armazenadas no repositório para determinado arquivo ou conjunto de arquivos.



### Fique atento

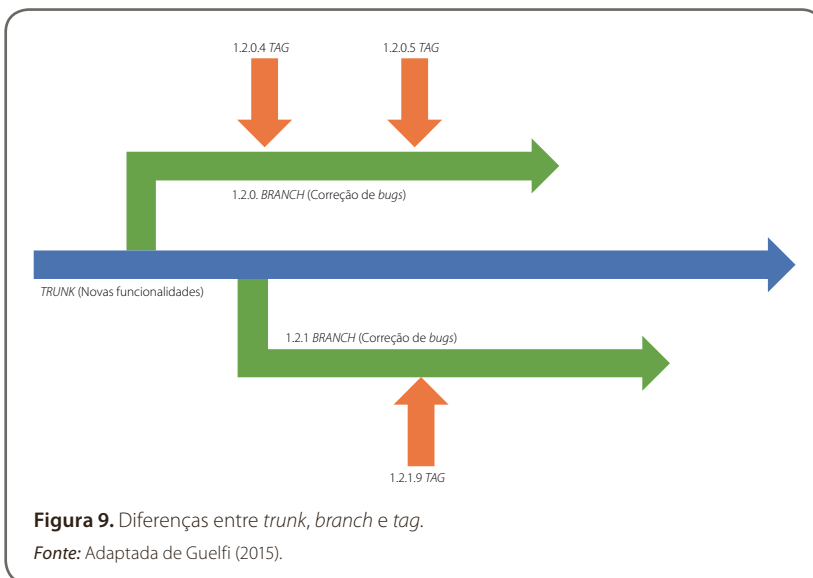
É importante ressaltar que o repositório evita a perda de dados, porém o que está na estação local não pode ser recuperado se houver um problema com o computador. É possível, ainda, ignorar arquivos para que eles não sejam comitados nem apareçam na lista de arquivos passíveis de *commit*.

Para cada um dos tipos de controle de versão, há um conjunto de operações básicas, conforme o Quadro 1, a seguir.

**Quadro 1.** Operações básicas dos controles de versão centralizado e distribuído

Descrição	Controle de versão centralizado	Controle de versão distribuído
Criação de estação de trabalho ou repositório.	<i>CHECKOUT</i>	<i>CLONE</i>
Envio de modificações para o repositório, gerando uma revisão.	<i>COMMIT</i>	<i>COMMIT</i>
Alteração da estação de trabalho em uma revisão.	<i>UPDATE</i>	<i>UPDATE</i>
Importação de revisões feitas em outro repositório.		<i>PULL</i>
Envio de revisões locais para outro repositório.		<i>PUSH</i>

Uma característica importante dos controles de versão é a possibilidade de separar modificações em um caminho diferente para cada desenvolvimento, conforme a Figura 9. Esse caminho, conhecido como **branch** (seta verde), é utilizado especialmente para a implementação de novas funcionalidades, sem comprometer o caminho principal da implementação, denominado **trunk**, (seta azul), com erros de compilação e *bugs*. A **branch** só será integrada ao **trunk** quando ela se tornar estável. Há também a possibilidade de congelamento de revisão, denominada **tag** (seta cor de laranja), isto é, é um estado fixo do produto que possui um conjunto de funcionalidades estáveis que não sofrerão mais nenhuma alteração.



### 3 Versionamentos distribuído e centralizado

Existem dois tipos de sistemas de controles de versão: centralizado e distribuído. Ambos possuem repositórios e estações de trabalho, porém a diferença entre eles está em como cada um está estruturado e organizado. A seguir, são apresentados os detalhes e as diferenças entre esses dois tipos de versionamento.