Produtividade com

por Giordano Lins

Microsoft CERTIFIED Technology Specialist

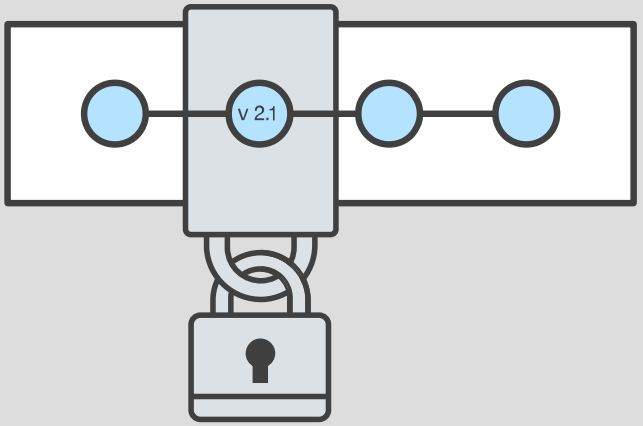
Microsoft CERTIFIED Professional

DETRAN-PE

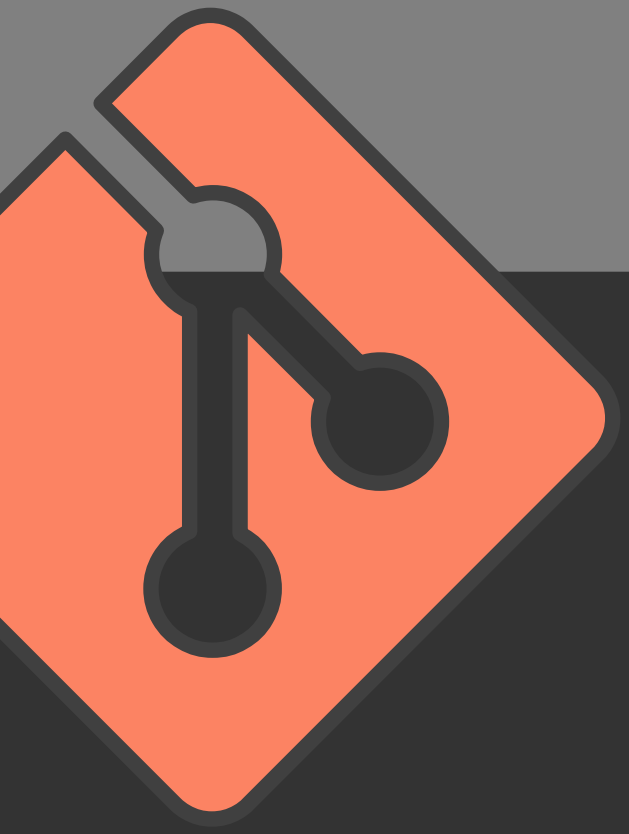
IVIA

O que é o controle de versão

Sistemas de controle de versão são uma categoria de ferramentas de software que ajudam times de desenvolvimento de software a gerenciar mudanças no código fonte desenvolvido ao longo do tempo.

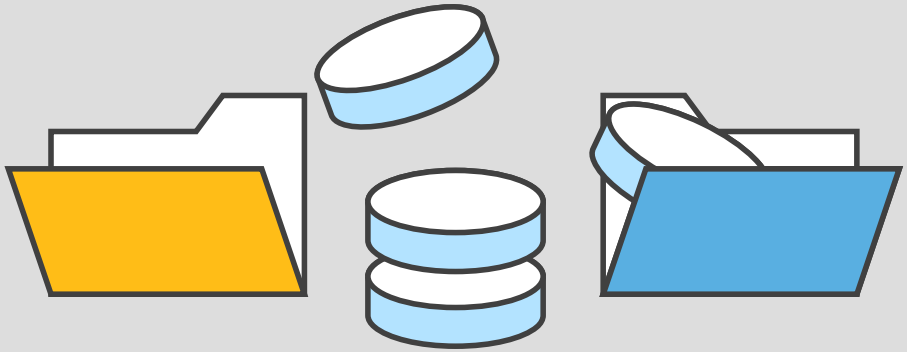


O código fonte **é um repositório de conhecimento e entendimento dos problemas do domínio do projeto**, que os desenvolvedores coletaram e refinaram com cuidado e esforço. Portanto, **o código fonte de um projeto é um artefato de valor inestimável**.



O que é o Git

Ele é de longe o sistema de controle de versão mais utilizado do mundo atualmente. O Git é **maduro**, e mantido ativamente como um projeto **open source** desenvolvido originalmente em 2005 por Linus Torvalds.



Por que usar o Git em sua organização

O Git tem como base a arquitetura distribuída. Além disso, foi desenhado com foco no desempenho, segurança e flexibilidade. O Git altera o fluxo de desenvolvimento impacta diretamente no negócio.

Git para Desenvolvedores

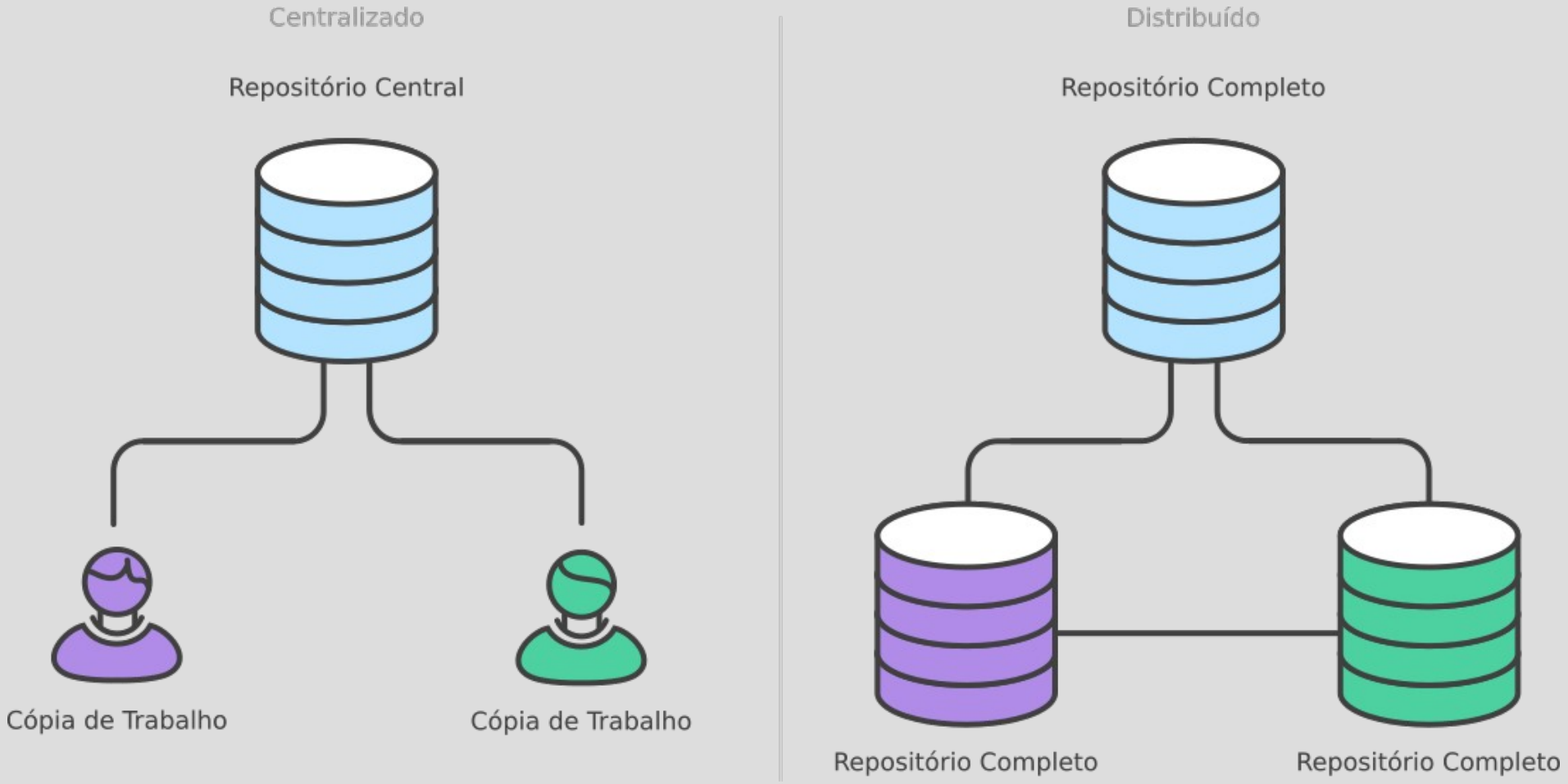
DETRAN-PE

IVIA

Desenvolvimento Distribuído

No **SVN**, cada desenvolvedor obtém uma **cópia de trabalho** que aponta para um repositório central. No Git, entretanto, em vez de uma cópia de trabalho, **cada desenvolvedor recebe sua própria cópia local completa do repositório**, com todo o histórico dos commits.

Desenvolvimento Distribuído



DETRAN-PE

IVIA

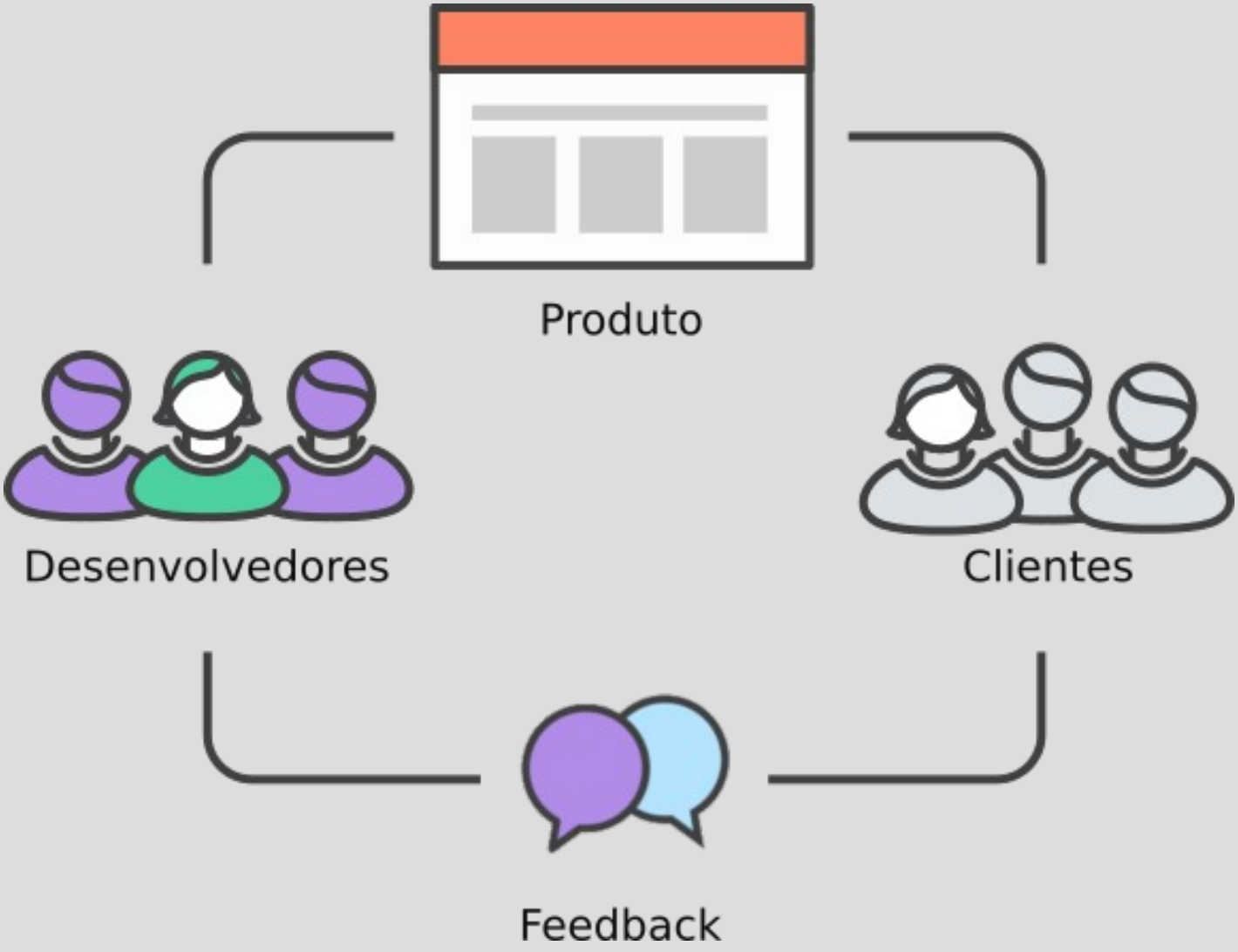
Desenvolvimento Distribuído

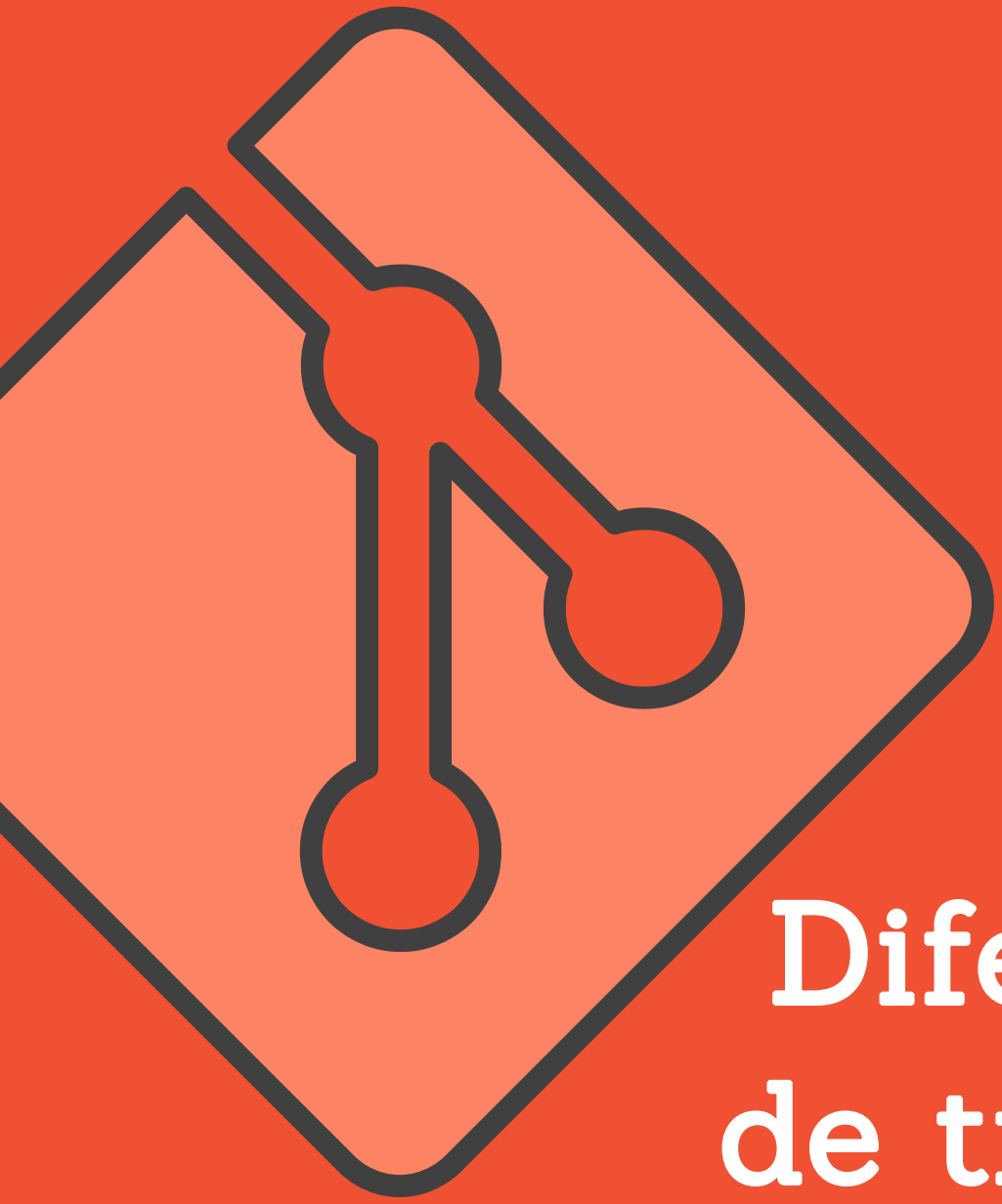
Ter uma cópia completa do repositório implica mais velocidade para o Git, uma vez que não há necessidade de conexão de rede para criar commits, inspecionar versões anteriores de um arquivo, ou realizar comparações entre commits.

Pull Requests, Comunidade e Ciclo de Releases mais velozes

Um **pull request** é uma forma de solicitar a outro desenvolvedor que realize o merge de um de seus branches no repositório.

O Git sugere um ciclo de entrega mais curto. Ele encoraja um fluxo ágil onde os desenvolvedores compartilham mudanças menores e mais frequentes.


Produto
Desenvolvedores
Clientes
Feedback

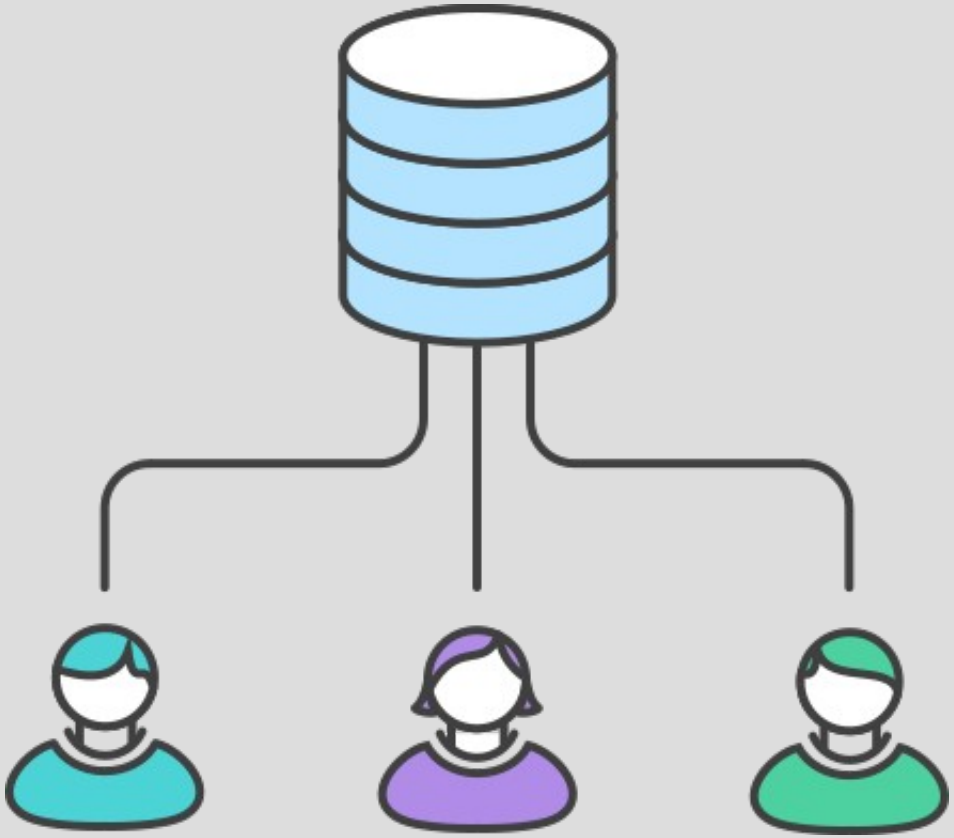


Diferentes fluxos de trabalho no Git

DETRAN-PE

IVIA

Todos trabahando em um repositório central



DETRAN-PE

IVIA

"Fulano" Trabalhando em uma funcionalidade

DETRAN-PE

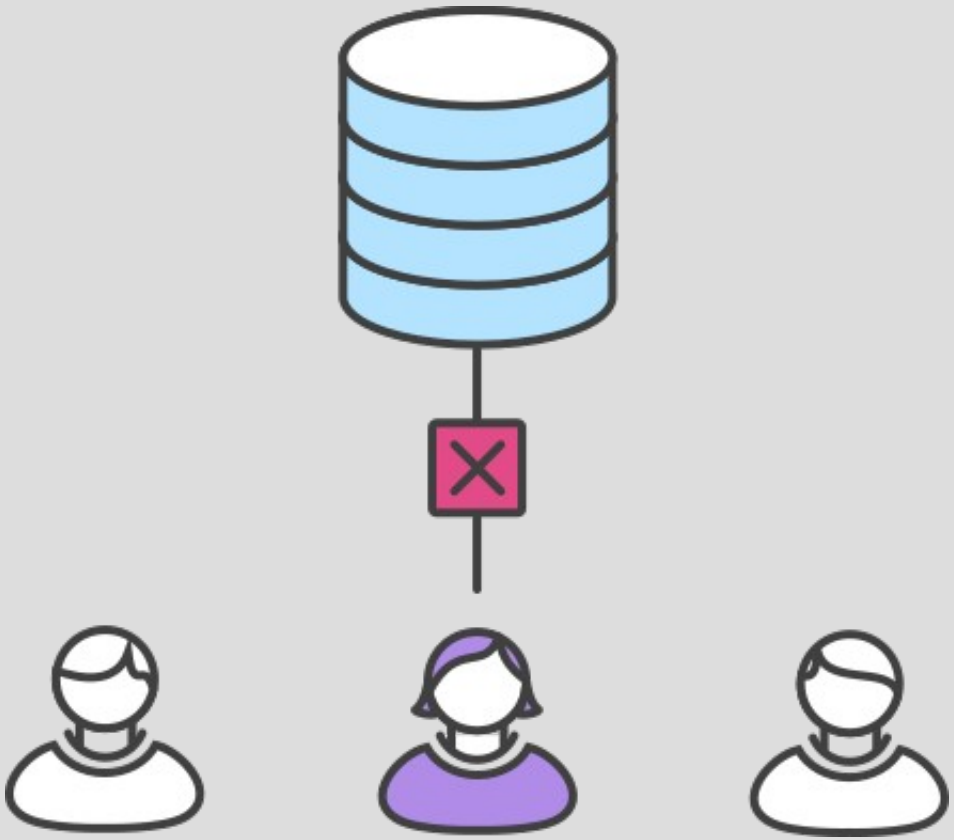
IVIA

"Fulana" Trabalhando em uma funcionalidade

DETRAN-PE

IVIA

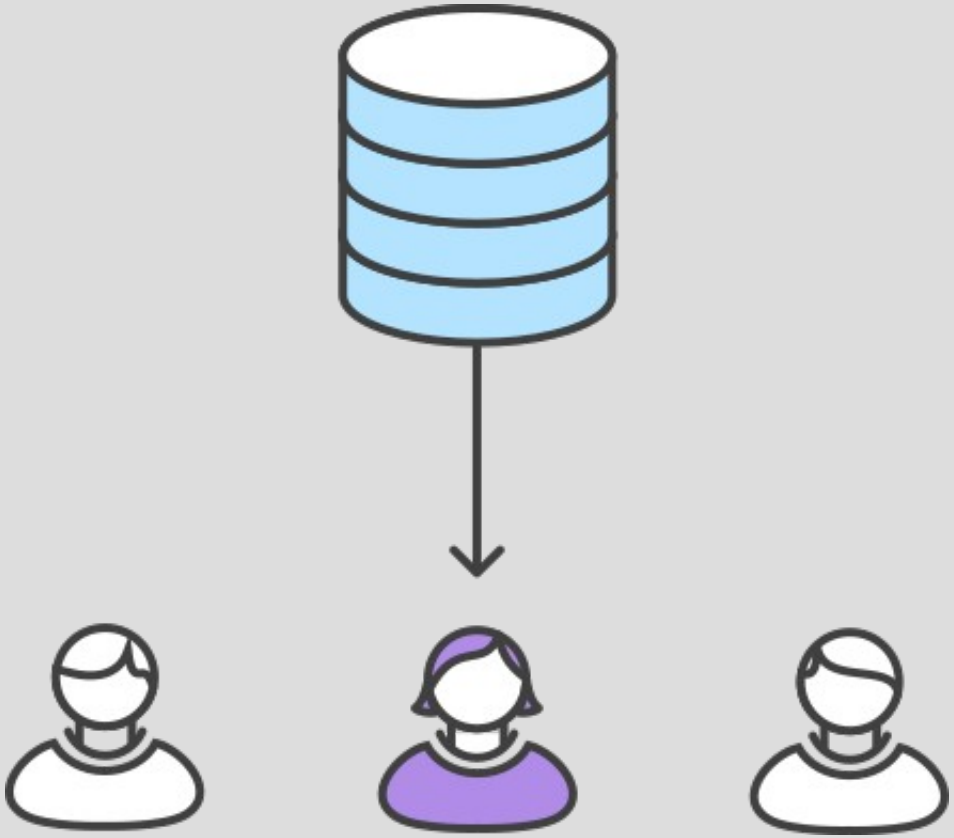
"Fulana" tenta publicar a dela em seguida



DETRAN-PE

IVIA

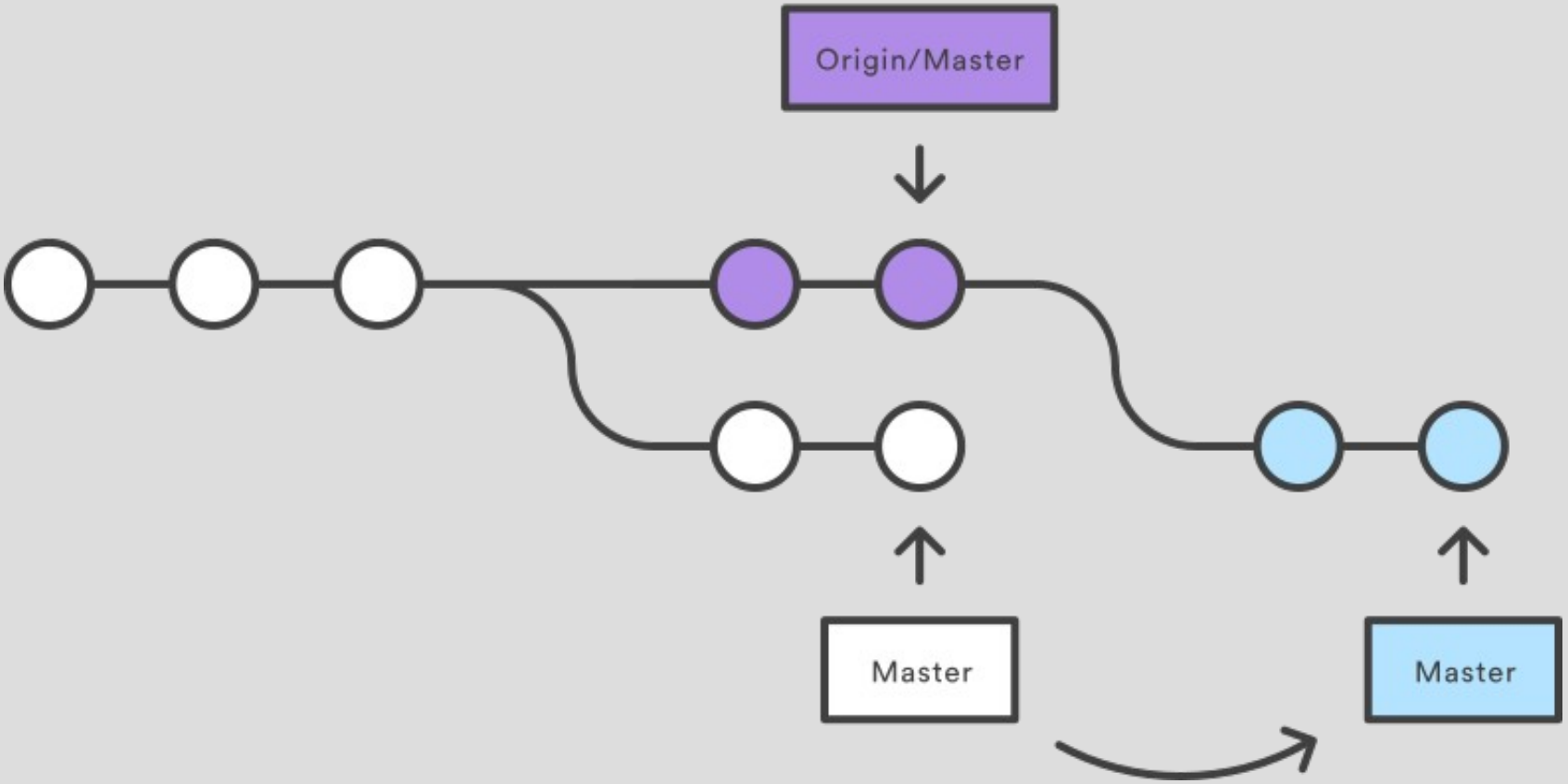
"Fulana" realiza então um "rebase" da funcionalidade de fulano



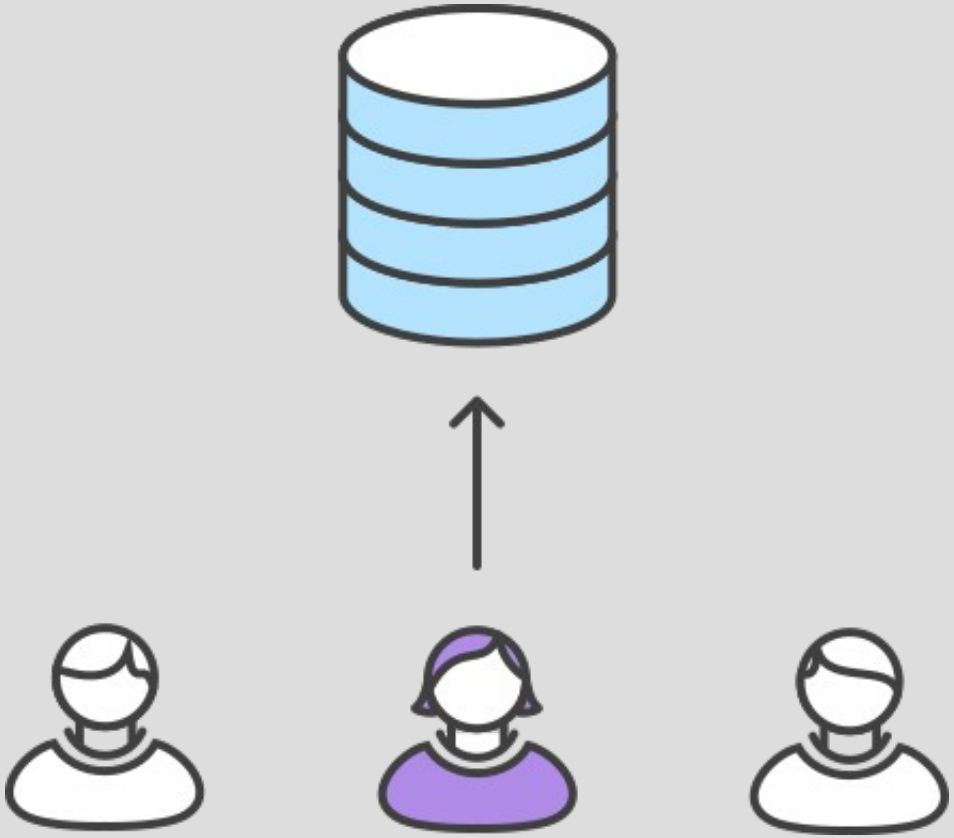
DETRAN-PE

IVIA

Rebase



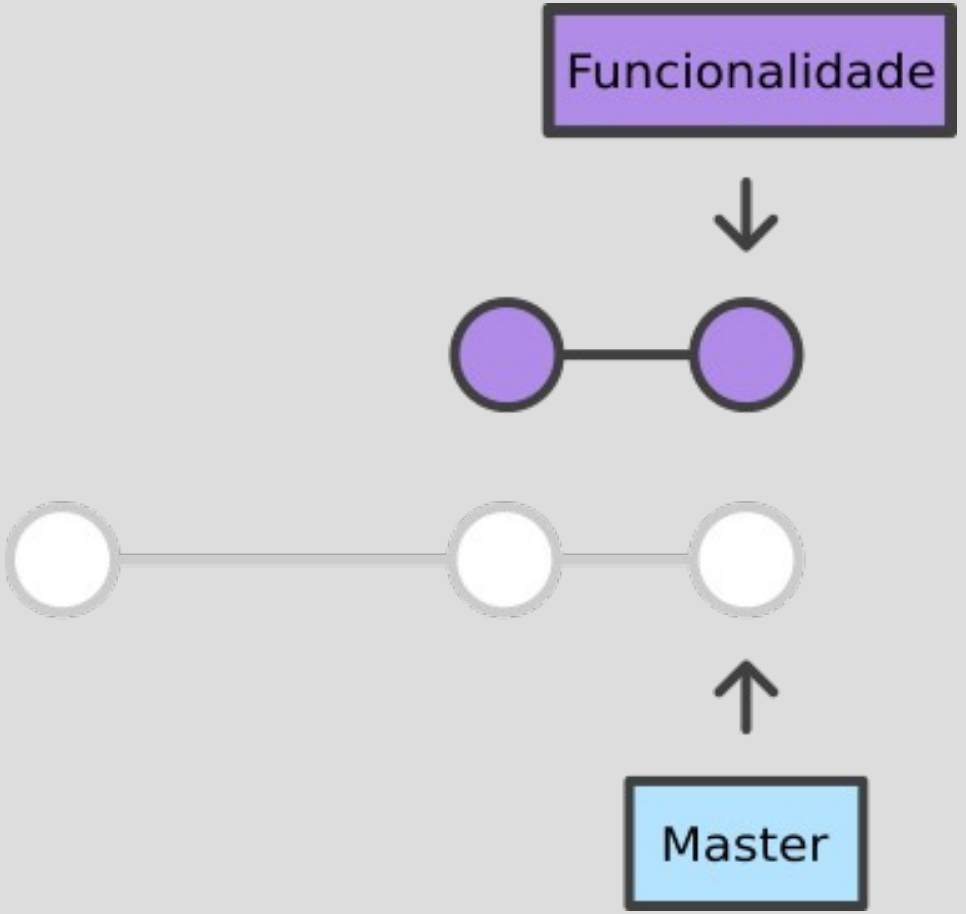
"Fulana" publica sua funcionalidade



Workflow de branches por funcionalidade

Uma das maiores vantagens do Git é sua capacidade de criar e gerenciar ***branches***. Ao contrário dos sistemas centralizados, os branches do Git são "baratos" e fáceis de **mesclar**. Além disso, essa metodologia visa **criar ambientes isolados de desenvolvimento para cada mudança realizada na base de códigos**.

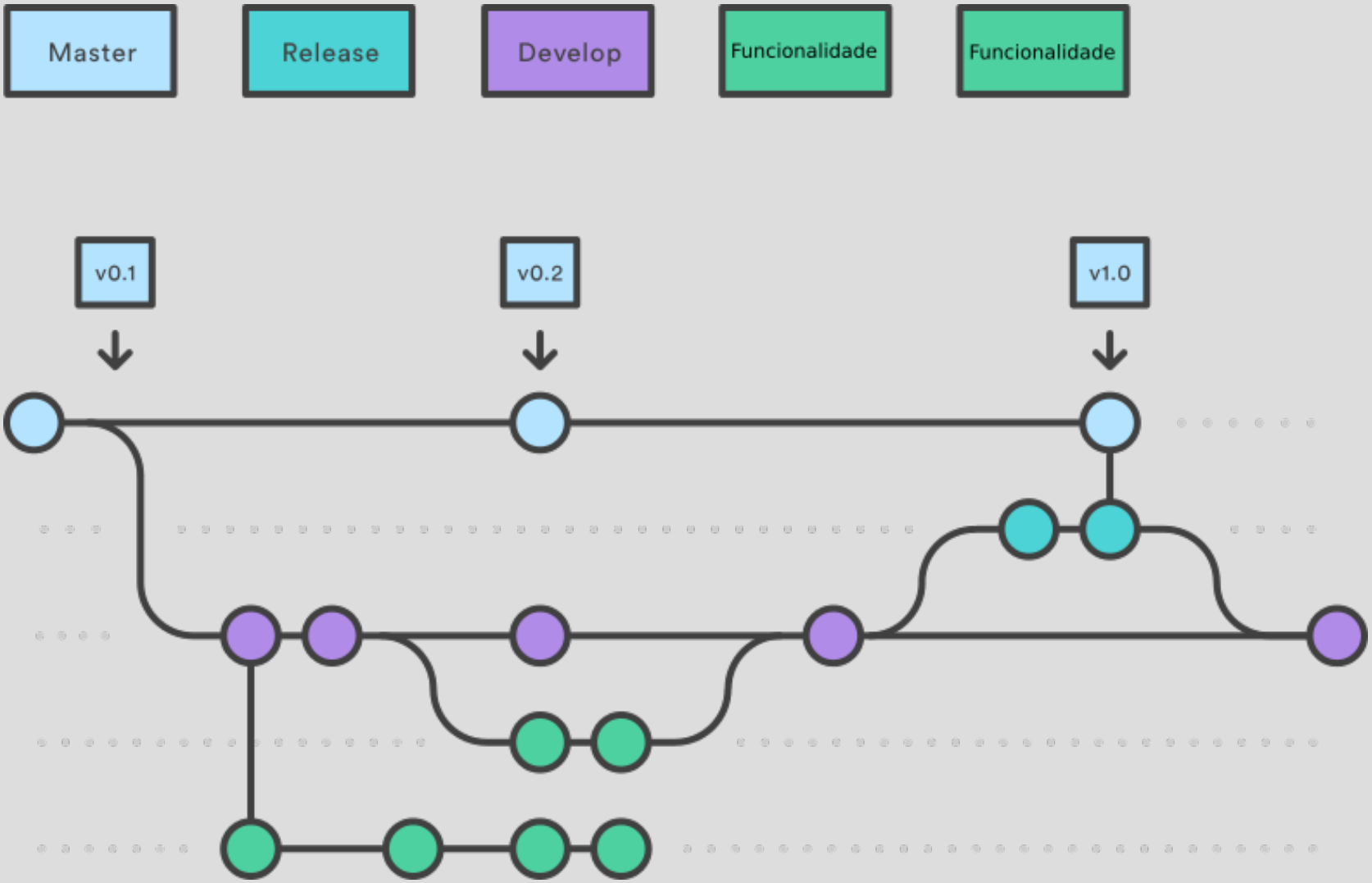
Workflow de branches por funcionalidade



DETRAN-PE

IVIA

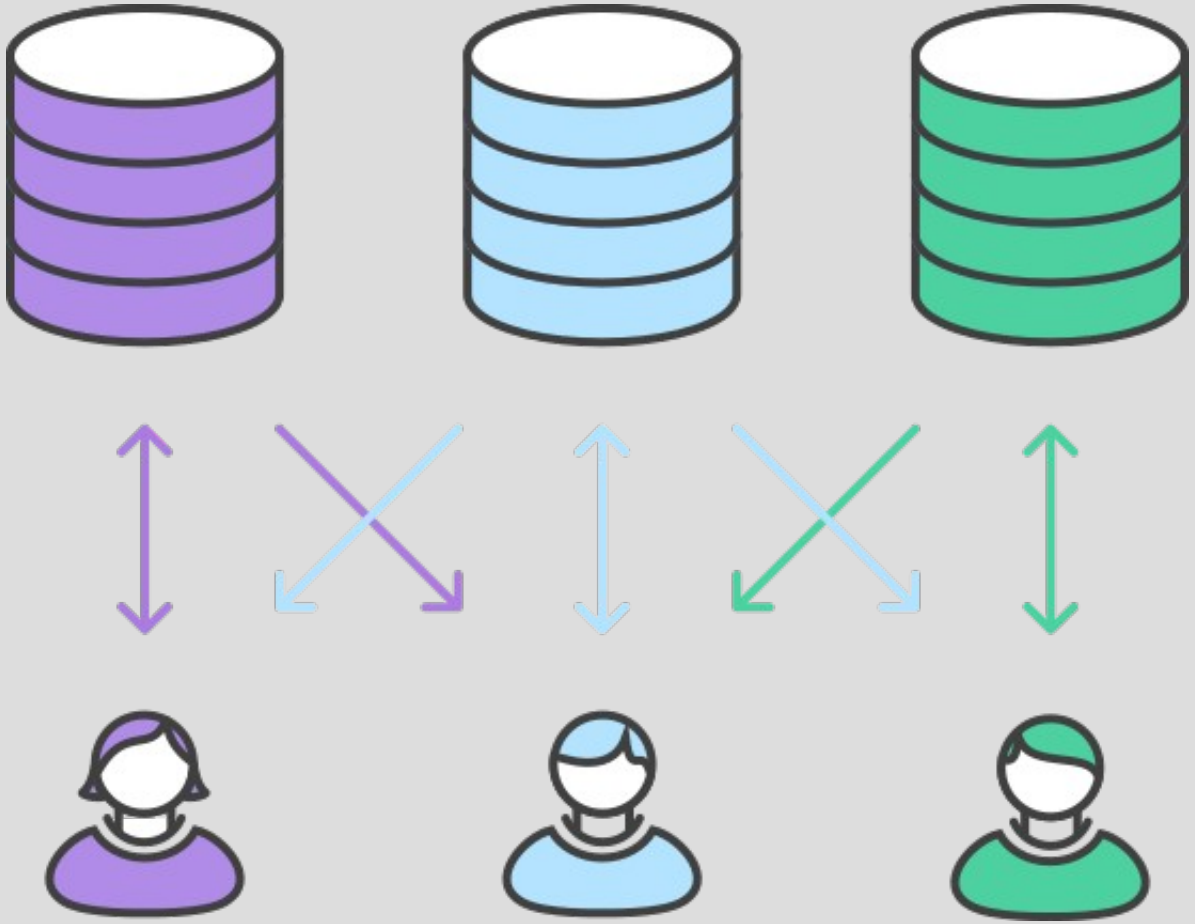
GitFlow



DETRAN-PE

IVIA

Forking



DETRAN-PE

IVIA

Referências

Esta apresentação é uma tradução, compilação e adaptação dos tutoriais contidos em:

https://www.atlassian.com/git/

Outros links:

https://git-scm.com/ (Site oficial do Git)
https://git-scm.com/book/en/v2 (Livro oficial do Git)