



Auditoria e Qualidade de Sistemas

Prof. Edgard Davidson C. Cardoso



Qualidade de Sistemas

INTEGRAÇÃO CONTÍNUA



O que é Integração Contínua

“Interação contínua é uma prática de desenvolvimento de software onde os membros de um time integram seu trabalho frequentemente, geralmente cada pessoa integra pelo menos diariamente - podendo haver múltiplas integrações por dia”



Martin Fowler

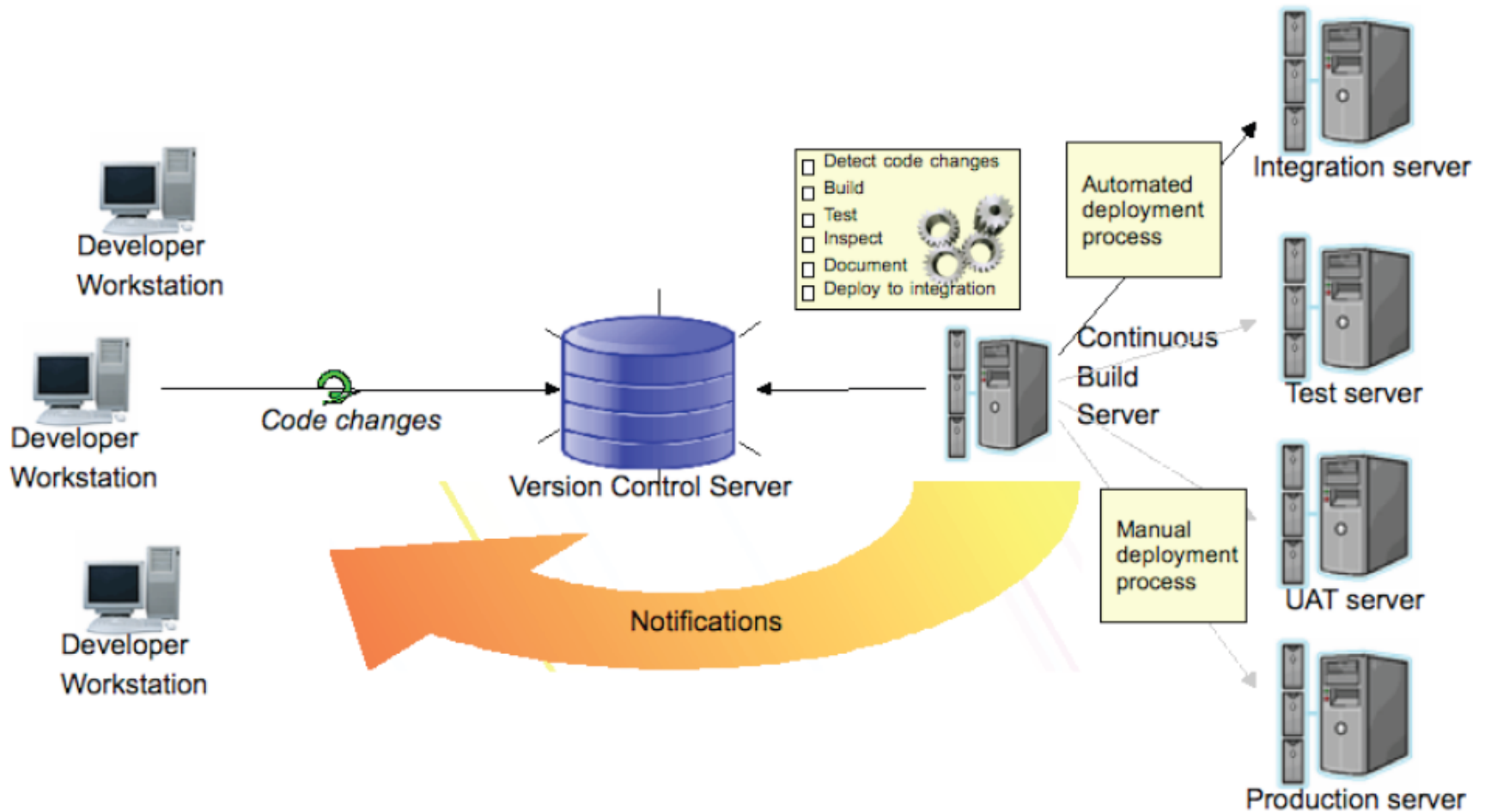


O Que é integração contínua

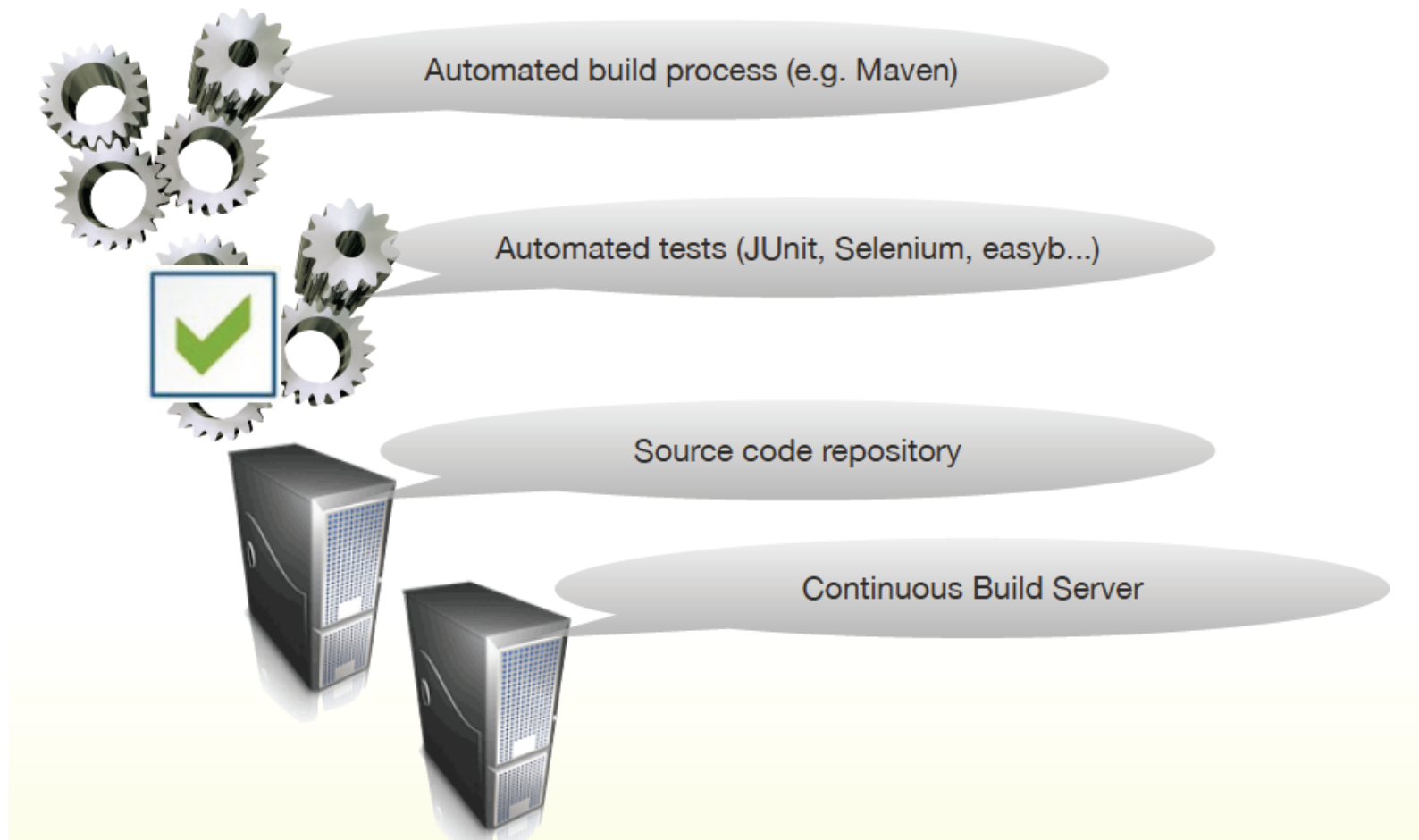
“A **Integração Contínua** pode ser descrita como **integração automática** com processo de **build automático** e que roda **testes de forma automática** e automaticamente detecta falhas em cada pedaço.”

<http://blog.caelum.com.br/integracao-continua/>

Integração Contínua – O que está envolvido?



Integração Contínua – O que você precisa





Entendendo a Integração contínua

- Integração Contínua tornou-se muito importante na comunidade de desenvolvimento de software e isso provavelmente ocorreu devido ao grande impacto causado pelas metodologias ágeis.
- Em equipes que adotaram tais metodologias (eXtreme Programming, Scrum, entre outras), integração contínua é um dos pilares da agilidade,
 - garantindo que todo o sistema funcione a cada build de forma coesa,
 - mesmo que sua equipe seja grande e diversas partes do código estejam sendo alteradas ao mesmo tempo.



Entendendo a Integração contínua

- a grande vantagem da integração contínua está no **feedback instantâneo**. Isso funciona da seguinte forma:
 - a cada **commit** no repositório, **o build é feito automaticamente**, com todos os testes sendo executados de forma automática e falhas sendo detectadas.
 - Se algum commit não compilar ou quebrar qualquer um dos testes, a equipe toma conhecimento **instantaneamente** (através de email, por exemplo, indicando as falhas e o commit causador das mesmas).
 - A equipe pode então corrigir o problema o **mais rápido possível**, o que é fundamental para não introduzir erros ao criar novas funcionalidades, **refatorar**, etc.
 - Integração contínua é mais uma forma de trazer segurança em relação a mudanças: você pode fazer modificações sem medo, pois será avisado caso algo saia do esperado.



Entendendo a Integração Contínua

- **Mas porque eu não rodo pessoalmente os testes na minha máquina e só então faço o commit?**
 - Simples: seu projeto pode ser tão grande, que os testes (em especial os de aceite) demoram um tempo considerável para serem executados e você não vai querer esperar todo esse tempo a cada commit pra poder continuar a trabalhar.
 - Nesse caso, o recomendado é rodar os testes que envolvem as partes que você modificou e só então commitar, deixando para o **servidor de integração contínua** o trabalho de realizar todos os testes do sistema e garantir que tudo esteja funcionando.



O que é Integração Contínua?

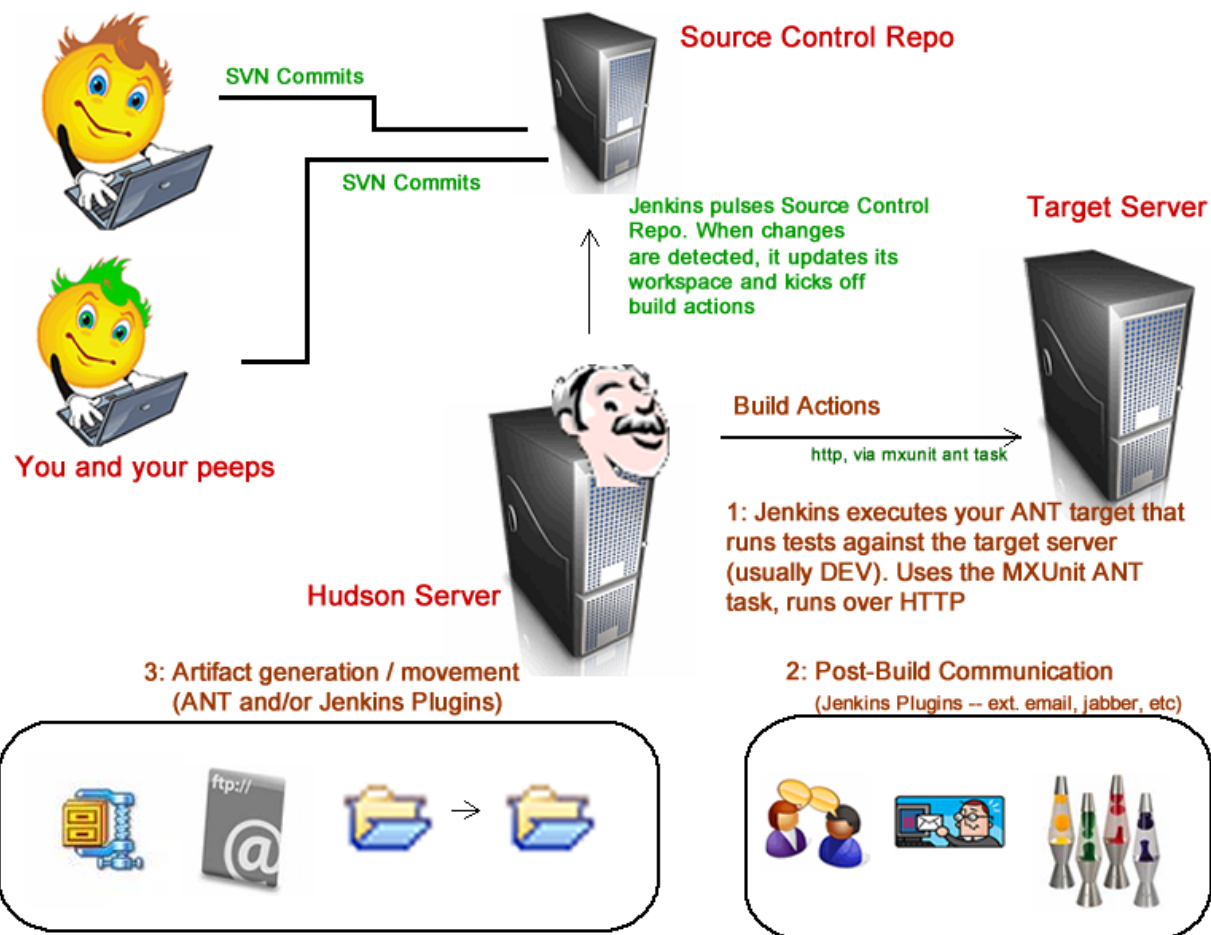
- Prática Ágil
- Processo contínuo
- Controle de qualidade
- Bugs detectados mais cedo
- Aumento da qualidade
- Reduz
 - Custo, tempo, trabalho



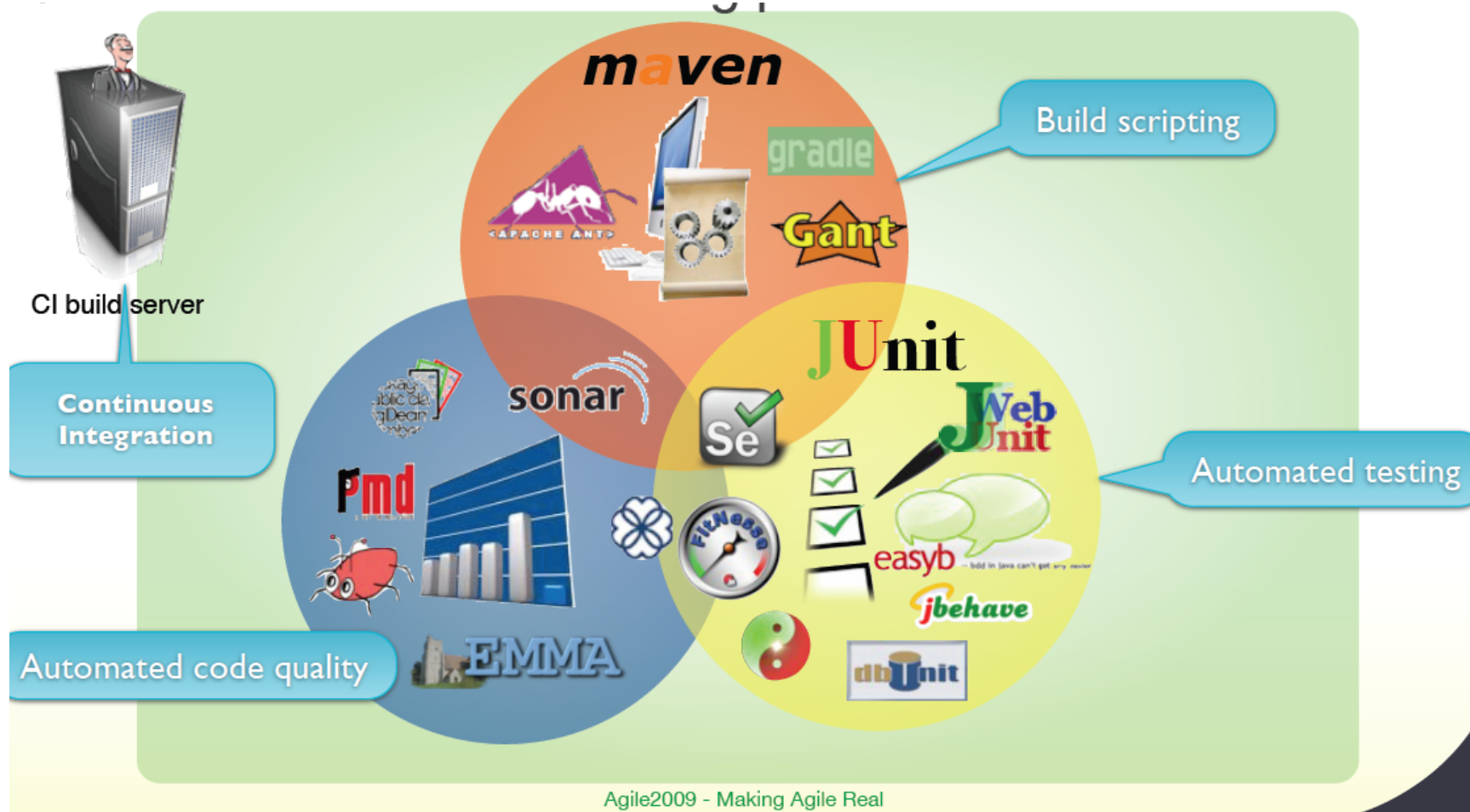
Por que Integração Contínua

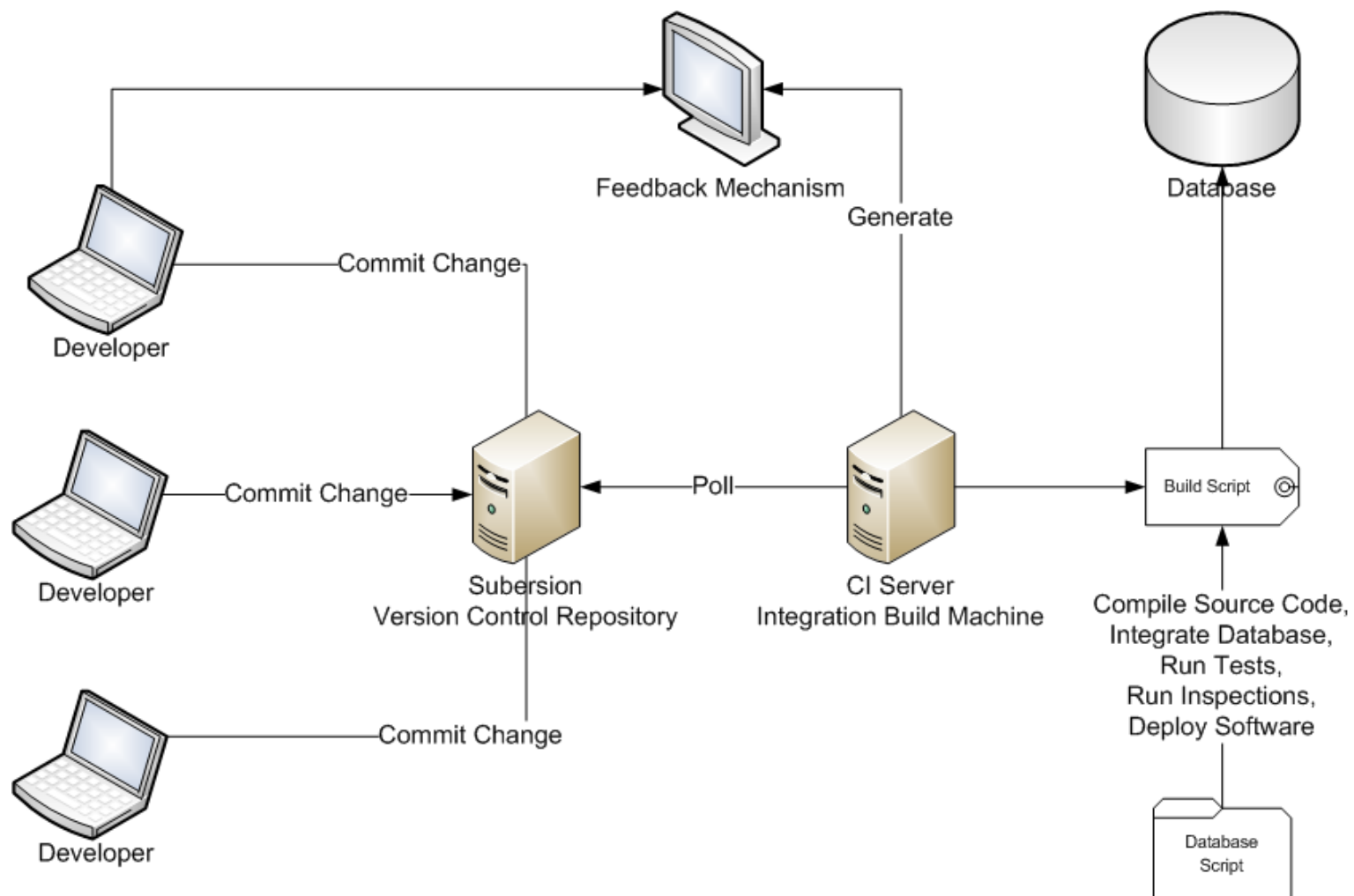
- Reduzir Riscos
- Feedback instantâneo
- Deploy contínuo



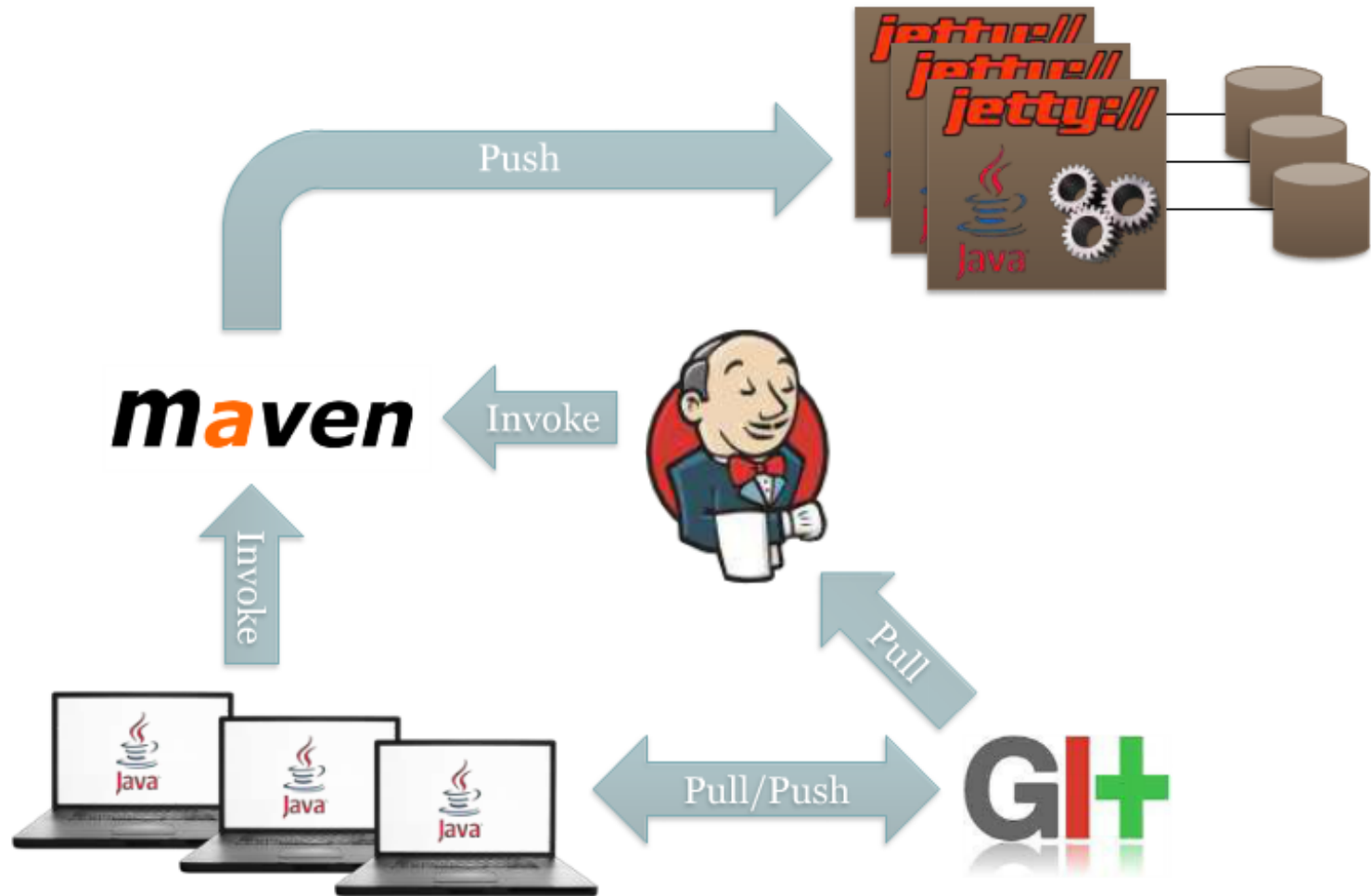


Visão Geral

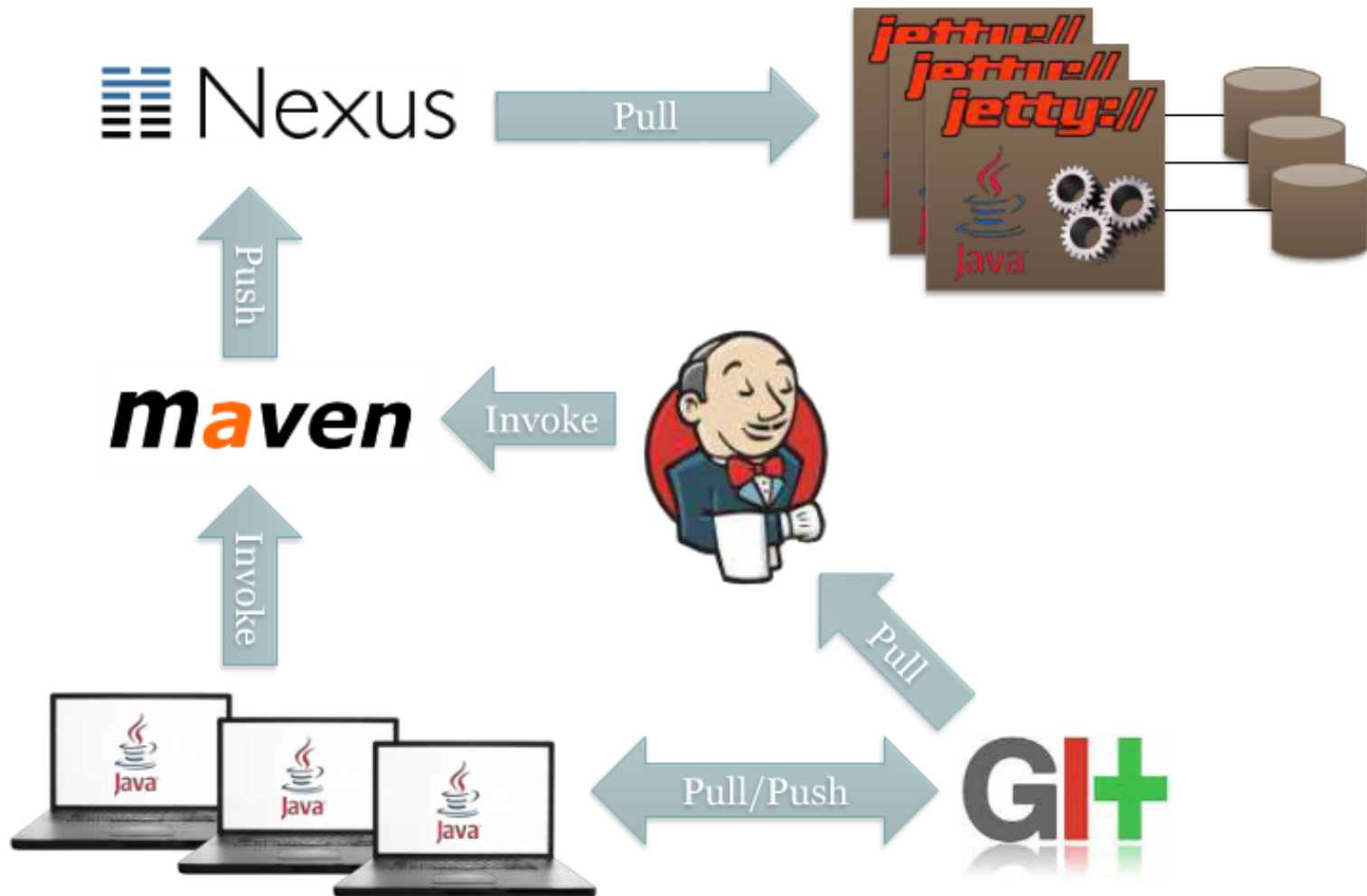




Continuous Deployment (Push)



Continuous Deployment (Pull)



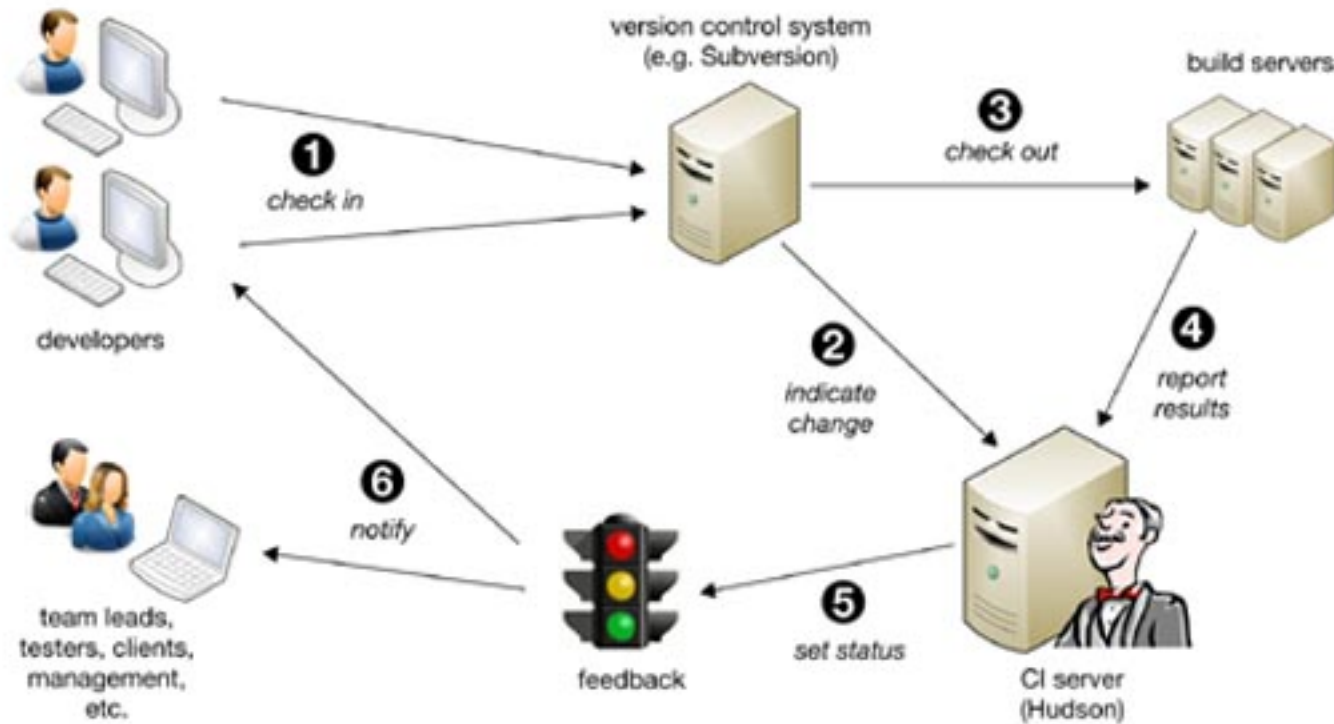


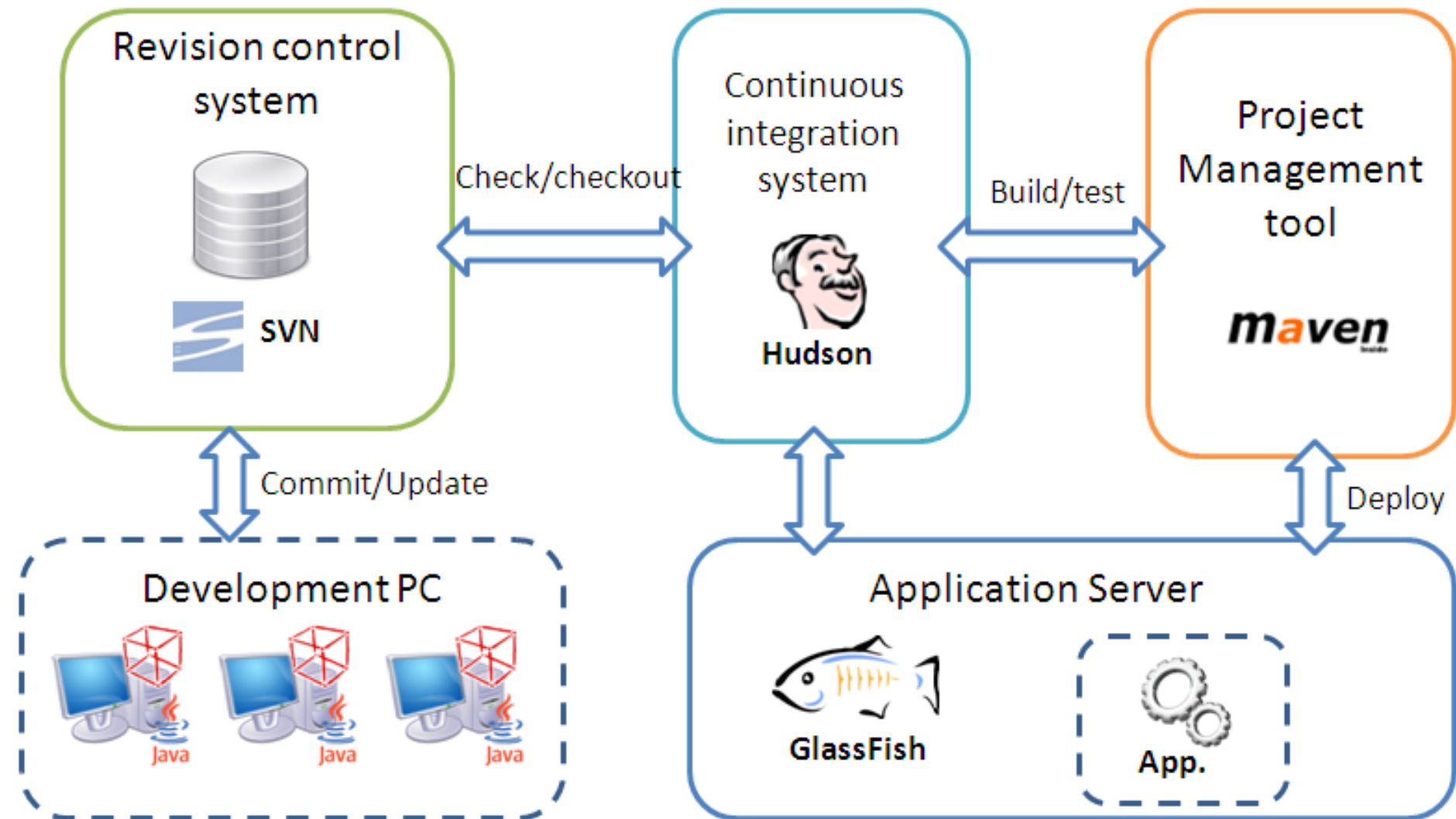
- Para teste unitário
 - Junit, TestNG
- Para Mock Teste
 - Jmock, Mockito, EasyMock
- Para BDD
 - JBehave
- Para controle de versão
 - Subversion, CVS, SVN, git...
- Para automação de build
 - And, Maven
- Para inspeção contínua
 - Sonar, JDepend, PMD, Findbugs, Checkstyle...
- Para integração continua
 - Hudson, Jenkins, CruiseControl

Integração Contínua

HUDSON







Integração contínua



Jenkins



O que é o Jenkins

- Jenkins é um servidor de integração contínua de código aberto
- É uma versão renovada(e “forkada”) do **Hudson**.
- Jenkins (<http://jenkins-ci.org>) é:
 - Fácil de instalar
 - Fácil de usar
 - Multi Tecnologia
 - Multi Plataforma
 - Extensível