КРПС

Непрерывная интеграция

Лекция №7

(версия 1.0)

«Continuous Integration» — это лекарство от страха. Помогает при программировании.
Dr. Zoidberg

Continuous Integration

Согласно Wikipedia термин Continuous Integration введен Мартином Фаулером (Martin Fowler) и Кентом Беком (Kent Beck). Данный термин был придуман ими для обозначения практики частой сборки (интеграции) проекта. Максимально частая сборка является логичным продолжением цепочки

итерационные сборки → ночные сборки → непрерывная сборка

Continuous Integration

Непрерывная интеграция (CI, англ. Continuous Integration) — это практика разработки программного обеспечения, которая заключается в выполнении частых автоматизированных сборок проекта для скорейшего выявления и решения интеграционных проблем. В обычном проекте, где над разными частями системы разработчики трудятся независимо, стадия интеграции является заключительной. Она может непредсказуемо задержать окончание работ. Переход к непрерывной интеграции позволяет снизить трудоёмкость интеграции и сделать её более предсказуемой за счет наиболее раннего обнаружения и устранения ошибок и противоречий.

Принципы

На выделенном сервере организуется служба, в задачи которой входят:

- получение исходного кода из репозитория;
- сборка проекта;
- выполнение тестов;
- развёртывание готового проекта;
- отправка отчетов.

Локальная сборка может осуществляться:

- по внешнему запросу,
- по расписанию,
- по факту обновления репозитория и по другим критериям.

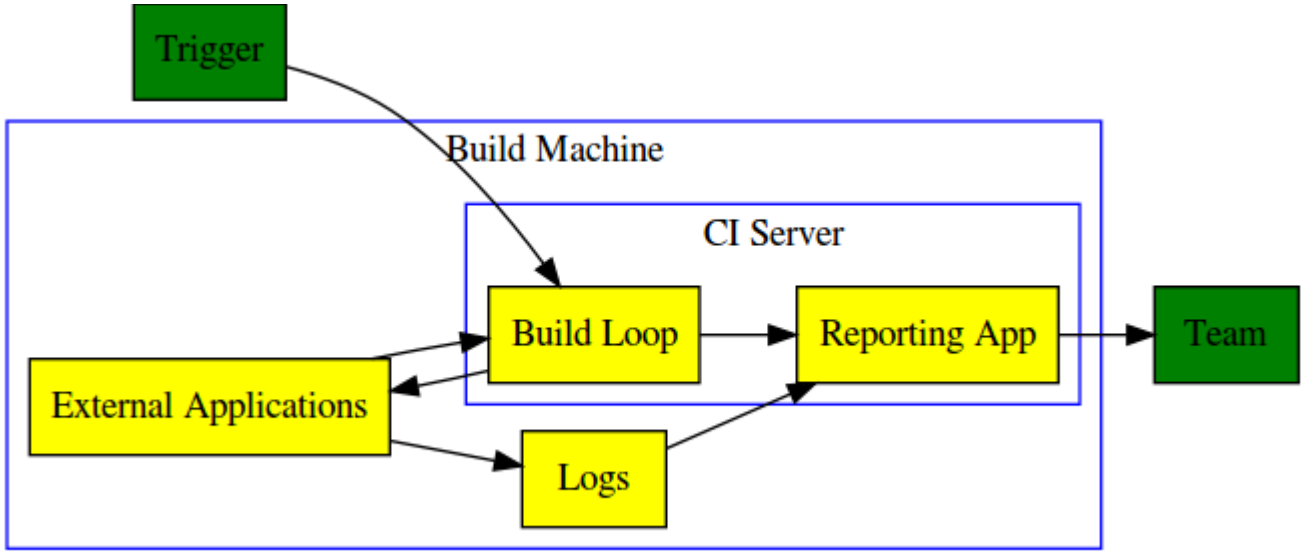
Достоинства

- проблемы интеграции выявляются и исправляются быстро, что оказывается дешевле;
- немедленный прогон модульных тестов для свежих изменений;
- постоянное наличие текущей стабильной версии вместе с продуктами сборок — для тестирования, демонстрации, и т. п.
- немедленный эффект от неполного или неработающего кода приучает разработчиков к работе в итерактивном режиме с более коротким циклом.

Недостатки

- затраты на поддержку работы непрерывной интеграции;
- потенциальная необходимость в выделенном сервере под нужды непрерывной интеграции;
- немедленный эффект от неполного или неработающего кода отучает разработчиков от выполнения периодических резервных включений кода в репозиторий.
- в случае использования системы управления версиями исходного кода с поддержкой ветвления, эта проблема может решаться созданием отдельной «ветки» (англ. branch) проекта для внесения крупных изменений (код, разработка которого до работоспособного варианта займет несколько дней, но желательно более частое резервное копирование в репозиторий). По окончании разработки и индивидуального тестирования такой ветки, она может быть объединена (англ. merge) с основным кодом или «стволом» (англ. trunk) проекта.

Примерная схема сервера CI



Best practices

- Поддержание репозитария с исходным кодом;
- Автоматизация сборки;
- Создание самотестируемых сборок;
- Каждый вносит изменения в базовую линию каждый день;
- Каждое вносимое изменение должно быть рабочим;
- Ускорение сборки;

- Тестирование в рабочем окружении;
- Делать более доступными последние изменения;
- Каждый может видеть результаты последней сборки
- Автоматизация развертывания

Процесс интеграции

Continuous integration процесс состоит из нескольких этапов, некоторые из которых обязательны, другие нет:

- Trigger — обязателен
- Update — обязателен
- Analyse — не обязателен
- Build — обязателен
- UnitTest — крайне желателен
- Deploy — нужен по обстоятельствам
- Test — не обязателен, но крайне желателен
- Archive — желателен
- Report — обязателен

Средства

- Bamboo
- Hudson и Jenkins
- CruiseControl
- TeamCity
- BuildBot
- Travis CI
- Team Foundation Server

CI в ИрГУПС

Hudson

Hudson — инструмент непрерывной интеграции, написанный на Java. Запускается в контейнере сервлетов, таких как Apache Tomcat или GlassFish. Поддерживает инструментарий для работы с разными системами контроля версий, включая CVS, Subversion, Mercurial, Git и Clearcase, может собирать проекты Apache Ant и Apache Maven, а также исполнять shell-скрипты и команды Windows.

Источники

- Википедия (https://ru.wikipedia.org/wiki/%CD%E5%EF%F0%E5%F0%FB%E2%ED%E0%FF_%E8%ED%F2%E5%E3%F0%E0%F6%FF , http://en.wikipedia.org/wiki/Continuous_integration , https://ru.wikipedia.org/wiki/Hudson)
- http://lib.custis.ru/Continuous_Integration
- Martin Fowler. Continuos Integration (http://www.martinfowler.com/articles/continuousIntegration.html)
- James Shore. Continuous Integration on a Dollar a Day (http://www.jamesshore.com/Blog/Continuous-Integration-on-a-Dollar-a-Day.html)