Going Literate in Amadeus

Vincent Latombe
Amadeus IT Group
http://www.amadeus.com

June 25, 2014

#jenkinsconf

About me

- Vincent Latombe a.k.a vlatombe
- Developer advocate at Amadeus

- Used to build for Java/JEE env., now also C++/Python/Ruby
- Heavy Jenkins user since 2010
- Maintainer of the Clearcase plugin
- Contributes to core (windows fixes), git and literate

Agenda

- What is Literate ?
- Literate in Practice
- Literate in Amadeus
- Under the hood

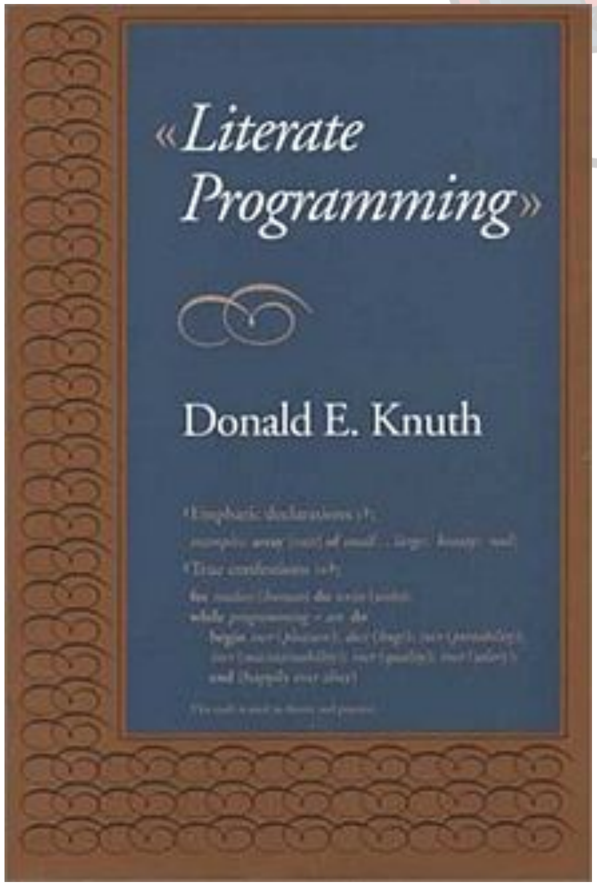
Part I

WHAT IS LITERATE???

WTF IS LITERATE???

Literate Programming

- Introduced by Donald Knuth in 1983
- Explain program logic using natural language with snippets of code and macros

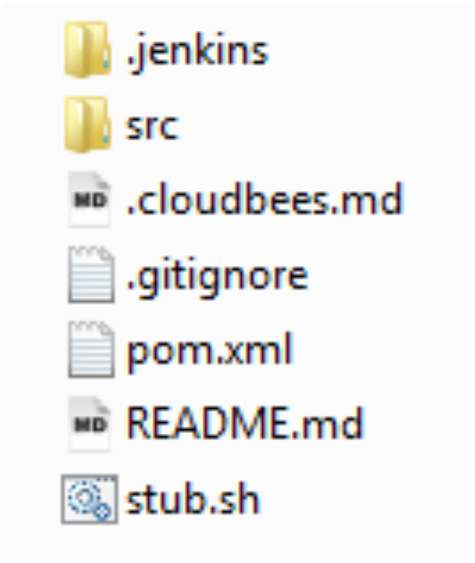


Literate Builds

- Applies Literate paradigm to builds
- Describes the build steps in a marker file or simply in README.md
- Build definition stored in the same SCM as the code it is building

Literate Builds

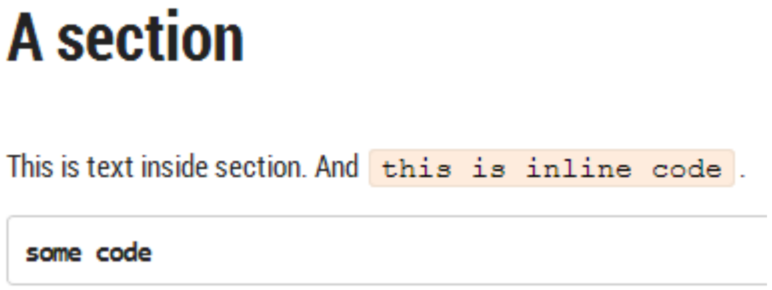
- Marker file within the SCM
- May contain the build definition
- Fallback to README.md if empty



```
.jenkins
src
.cloudbees.md
.gitignore
pom.xml
README.md
stub.sh
```

Markdown Cheatsheet

```
A section
=========
This is text inside section.
And `this is inline code`.

  some code
```

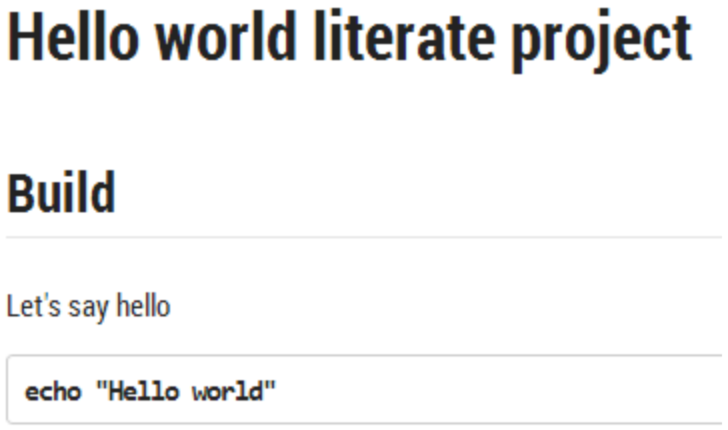

A section

This is text inside section. And `this is inline code`.

```
some code
```

Hello world example

```
Hello world literate project
============================


Build
-----


Let's say hello


    echo "Hello world"
```



Hello world literate project

Build

Let's say hello

```
echo "Hello world"
```

Ant example

```
Environments
============
Notice how we specify the environment to build with by providing
Jenkins node labels or tool installer names in code snippet
sections attached to bullet points in an "environment" section?

* `ant-1.8`, `java-1.7`

Build
=====

    ant clean dist
```

Matrix build? No problem

```
Complex project
===============

Environments
============

We have two different environments that the build must be run
on:

* `linux`, `gcc-4.2`, `ant-1.8`, `maven-3.0.5`, `java-1.7`
* `windows`, `vs-pro-2012`, `ant-1.8`, `maven-3.0.5`, `java-1.7`
```

Build commands per environment

```
Build
=====

We have two different sets of build instructions, one for
building with visual studio and the other for building with GCC

* On `gcc-4.2`, we start by building the native code

    ./configure
    make

* On `vs-pro-2012`, we have a batch file to do the native steps

    call build-native.bat
```

Additional tasks

- Defined in the marker file
- Enabled per branch using the job configuration
- Lightweight promotion
  - Self-promotion
  - Manual promotion (with optional parameters)

Extensions

- Files stored in SCM under .jenkins folder
- Basic: <extensionName>.xml, contains the config snippet
- Implement Agent to expose higher-level configuration files

Part II

LITERATE IN PRACTICE

DEMO TIME !

DEMO TIME

- Basic project
- Multi environment
- Multi branches
- Promotion
- Parameters

Part III

LITERATE IN AMADEUS

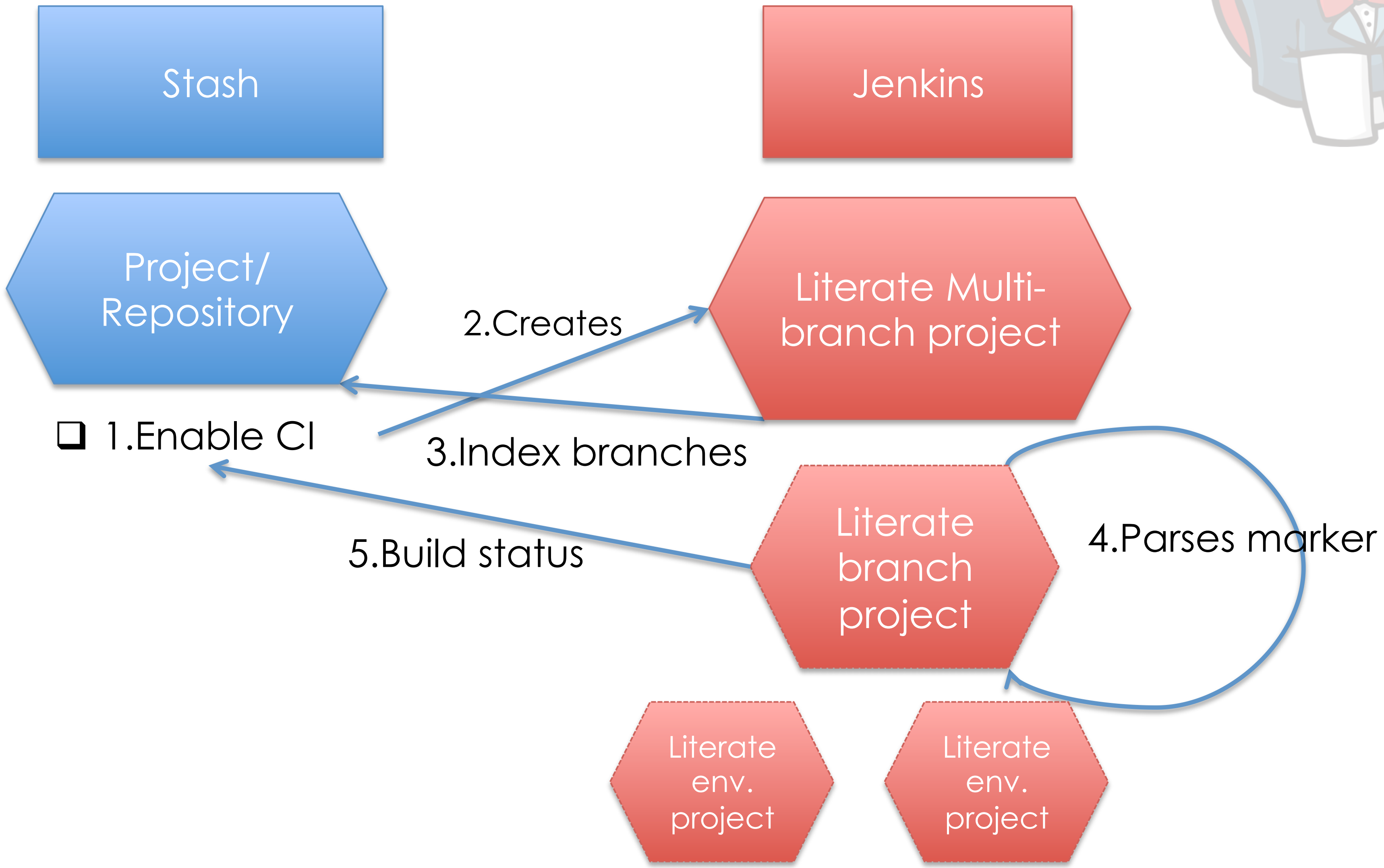
Our context

- About 3000 developers
- Commercial products
  - ~1000 Java/JEE
  - ~2000 C++
- For internal tooling, add
  - Python
  - Ruby
  - Perl

It’s about control!

- Fully open
- Locked down
- Balanced
  – Template-based solutions: only defined attributes can be customized
  – Literate: commands can be customized, post-build actions can be whitelisted

Infrastructure



YAML Parser

- Markdown considered too textual, too much space for syntax errors, even if using README.md was tempting
- We have some YAML lovers internally ;-)
- Easy to implement thanks to existing Literate architecture

Markdown vs YAML

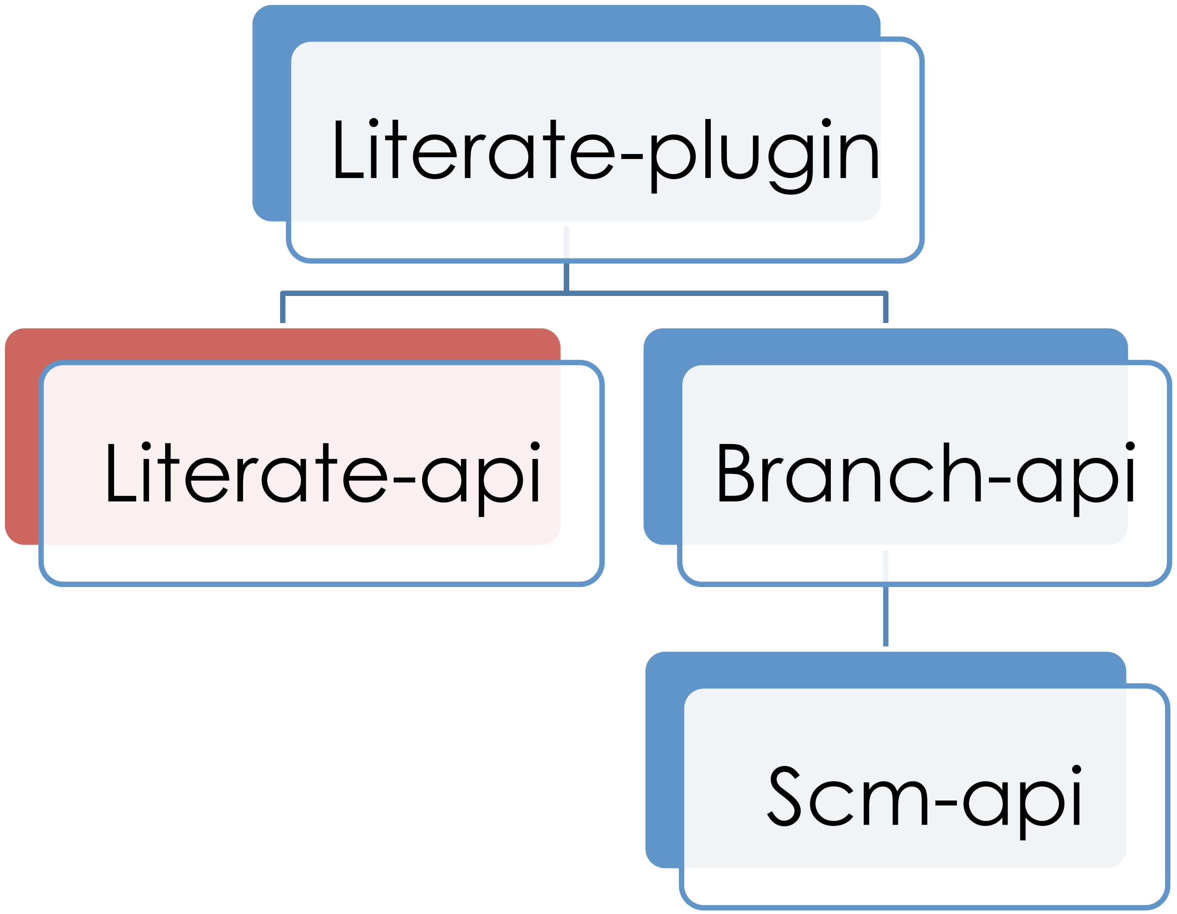
```
Environments
============

* `windows`, `java-1.6`
* `linux`, `java-1.7`

Build
=====

    mvn –B clean verify
```

```
environments:
  - [windows, java-1.6]
  - [linux, java-1.7]
build: mvn –B clean verify
```

Part IV

UNDER THE HOOD

Architecture



Literate-api

- Doesn’t depend on Jenkins
- Pluggable model builders (Markdown, YAML, …)

Scm-API

- jenkins.scm.api.SCMSource : provider for hudson.scm.SCM instances
- Able to comprehend multiple heads on a SCM

Branch-api

- Toolkit to handle multi-branch
  - Dead branch policy
  - Untrusted branches
  - Build retention
  - Throttling
- Foundation for
  - Multi-branch freestyle project
  - Multi-branch template project

Literate-plugin

- New extension points
  - LiterateBranchProperty : decorate branch and environments with anything (buildwrapper, properties…)
  - Agent : Builds Publisher instances from files in the SCM.

Towards a release

- Develop the ecosystem
  - More plugins integration
  - Github/Bitbucket/Stash pull request support
- Integration with isolation features
  - Sandbox commands within a container

Thank You To Our Sponsors

Platinum

CloudBees

Gold

zend The PHP Company

MidVision™ Release the innovation

codecentric

Silver

XebiaLabs the deployment automation company

SOASTA Test Faster. Release Sooner.

Corporate

cloudera®

Community

Java User Group Berlin Brandenburg

Lightweight Java User Group Munich