Writing Custom Plugins

Elton Stoneman
CONSULTANT & TRAINER

@EltonStoneman | blog.sixeyed.com

Development tools

Java + Maven

Plugin templates

Dev Jenkins instance

Jenkins integration

Extension points

Web UI, resources

Job definition, runtime

Plugin publishing

Building the HPI

Installing manually

Testing in Jenkins

Demo

Building a "Hello, World" plugin

- Setting up the environment
- Using Maven templates
- Testing a build-step plugin

```
docker run ...
 maven:3.6.3-jdk-8
```

◄ Install Java and Maven

```
mvn archetype:generate
-Dfilter="io.jenkins.archetypes:"
```

◄ Launch template wizard

```
mvn verify
```

◄ Build & test

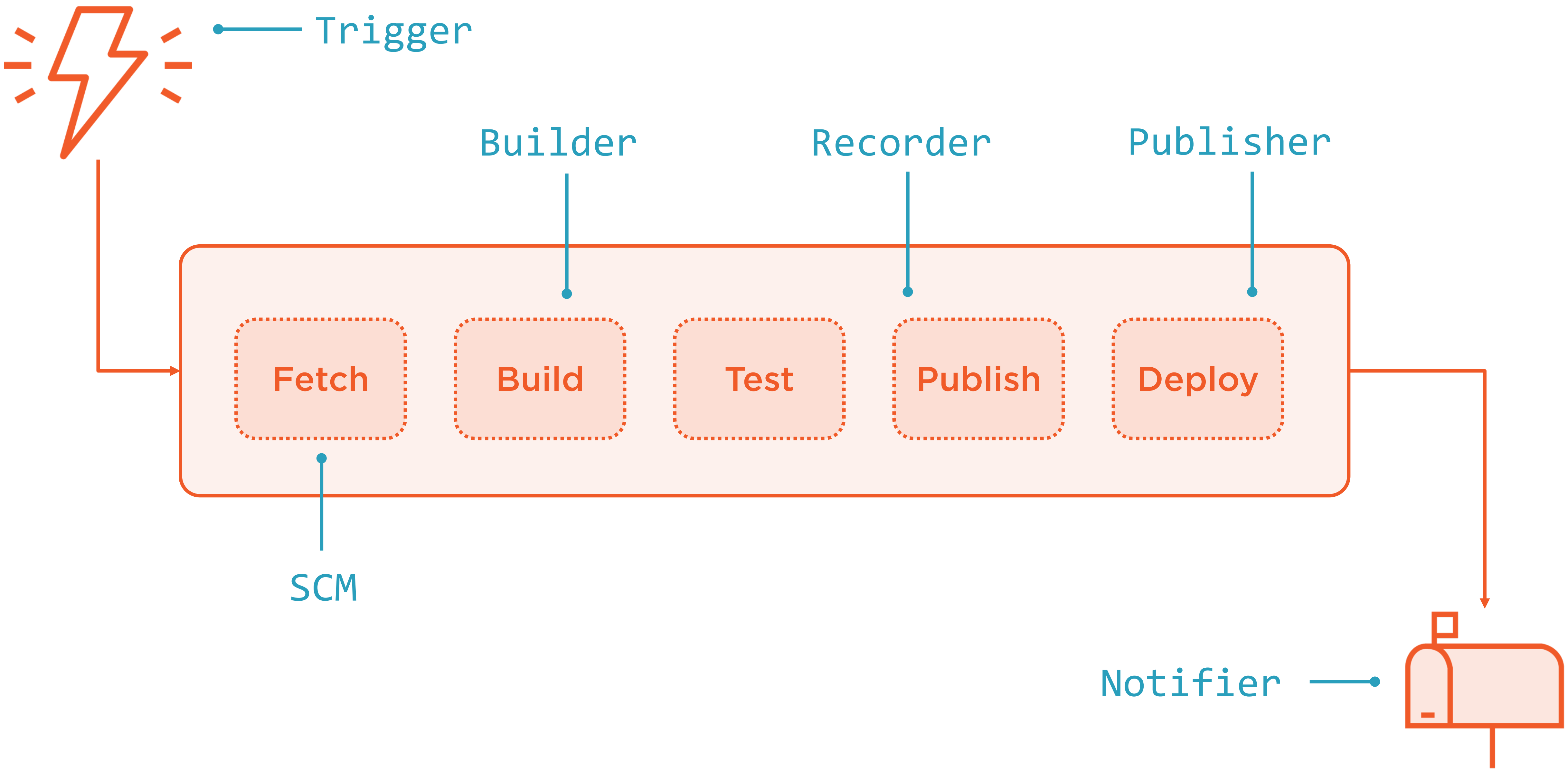
```
mvn hpi:run
```

◄ Run dev Jenkins server

Build Step Plugin

HelloWorldBuilder.java

```java
public class HelloWorldBuilder extends Builder implements SimpleBuildStep {
  ...
  @Symbol("helloWorld")
  @Extension
  public static final class DescriptorImpl extends BuildStepDescriptor<Builder> {
    ...
```


Trigger
Builder
Recorder
Publisher
Fetch
Build
Test
Publish
Deploy
SCM
Notifier

Demo

Building a useful plugin

- Custom plugin code
- Hooking into the Jenkins UI
- Testing a post-build plugin

Docker Hub Link Plugin

DockerHubLinkRecorder.java

```java
public class DockerHubLinkRecorder
  extends Recorder {
  ...
  @Symbol("dockerHub")
  @Extension
  public static class DescriptorImpl
   extends
   BuildStepDescriptor<Publisher> {
```

DockerHubLinkAction.java

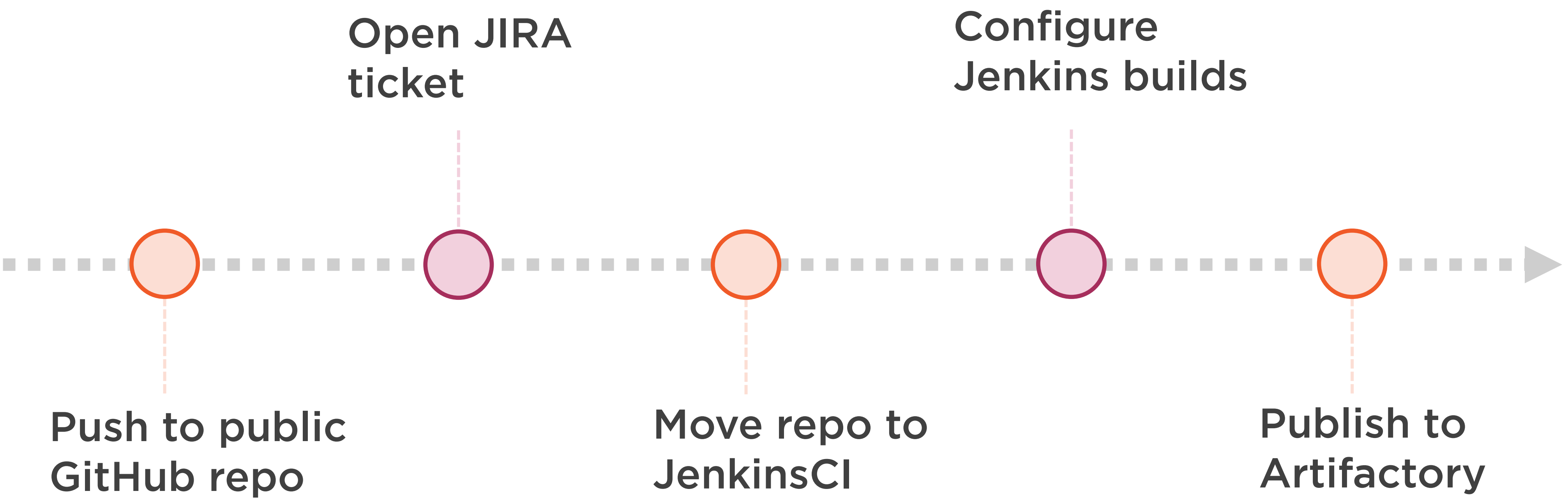
```java
public class DockerHubLinkAction
  implements RunAction2 {
  ...
  private String repositoryOwner;
  ...
  public String getImageTag() {...}

  public String getImageUrl() {...}
```

Docker Hub Link UI

index.jelly

```
<l:layout title="Docker Hub">
  <l:side-panel>
    <st:include page="sidepanel.jelly" it="${it.run}" optional="true" />
  </l:side-panel>
  <l:main-panel>
    <h1>Name: ${it.getImageTag()}</h1>
    <h2><a href="${it.getImageUrl()}">${it.getImageUrl()}</a></h2>
    ...
```



Push to public GitHub repo

Open JIRA ticket

Move repo to JenkinsCI

Configure Jenkins builds

Publish to Artifactory

https://is.gd/gujezi

Demo

Deploying custom plugins

- Building the HPI
- Installing the plugin manually
- Testing in a full Jenkins instance

```
mvn hpi:hpi
```

◀ Build plugin HPI

Summary

Development environment

- Java + Maven
- Project templates & builds
- Docker for non-Java people

Plugin code

- Extension points
- Job definition UI
- Runtime with job stages

Testing plugins

- Using the dev Jenkins instance
- Deploying HPI to a running Jenkins

Up Next:
Managing and Upgrading Plugins