Getting Started with Cloud Foundry

Deploying via Eclipse or Spring Tool Suite

Build, deploy, run, monitor

Overview

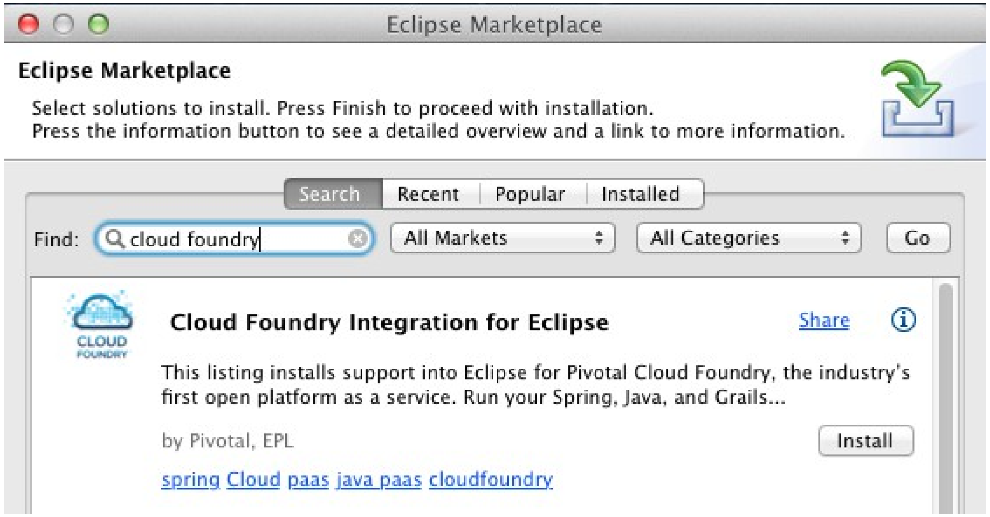
- After completing this lesson, you should be able to:
  - Setup Eclipse or Spring Tool Suite to use Cloud Foundry
  - Deploy an application to CloudFoundry
  - Manage application instances

Roadmap

- **Get Setup**
- Deploying an Application
- Managing Application Instances

Setting up Eclipse / Spring Tool Suite

- Select Help → Eclipse Marketplace
- Search for *Cloud Foundry* plugin and install (if need be)



Eclipse Marketplace

Eclipse Marketplace

Select solutions to install. Press Finish to proceed with installation.
Press the information button to see a detailed overview and a link to more information.

Find: cloud foundry

Cloud Foundry Integration for Eclipse

This listing installs support into Eclipse for Pivotal Cloud Foundry, the industry's first open platform as a service. Run your Spring, Java, and Grails...

by Pivotal, EPL

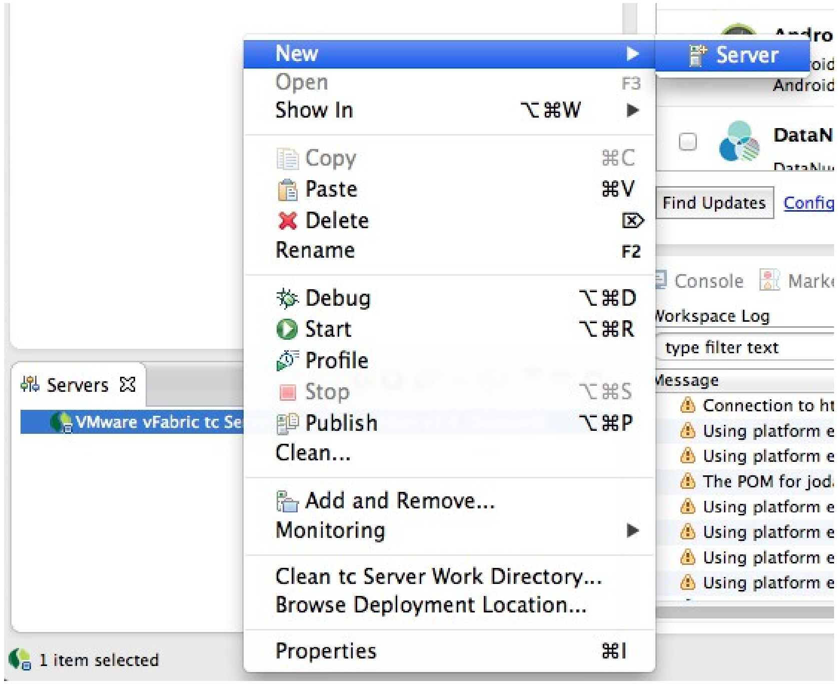
spring Cloud paas java paas cloudfoundry

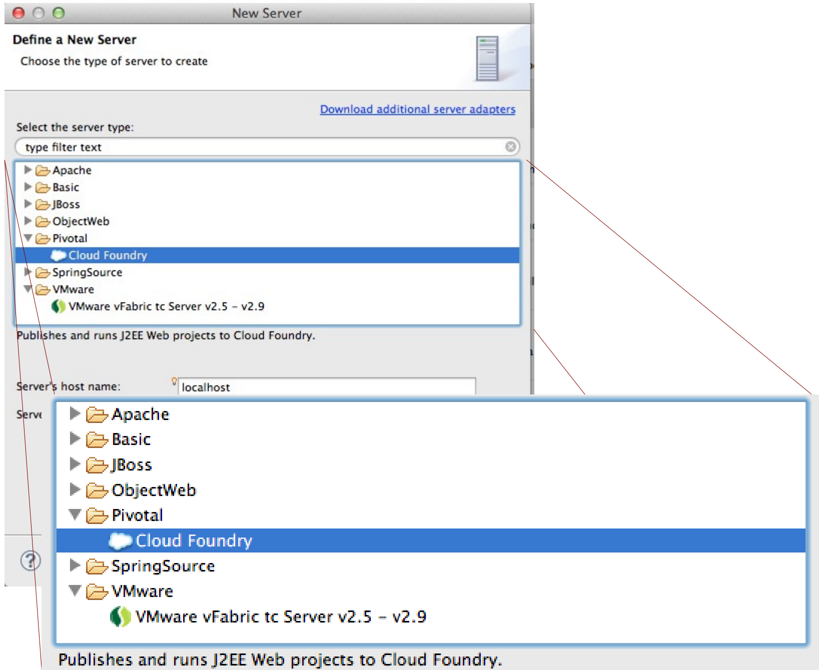
Do this now...

– The install takes time to run.

Setting up a New Server

- In white-area of Servers panel, right click New → Server
- In popup, under Pivotal select Cloud Foundry




New Server

Define a New Server

Choose the type of server to create

Download additional server adapters

Select the server type:

type filter text

Publishes and runs J2EE Web projects to Cloud Foundry.

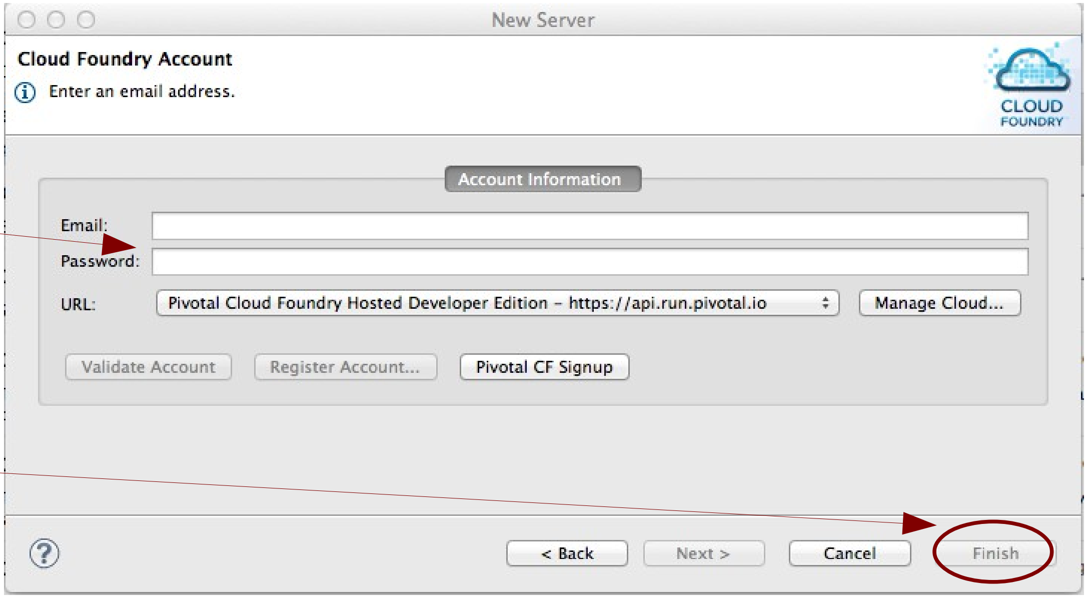
Server's host name: localhost

Fill in Details of you CF Account

- Fill in registration dialog
  - Use Manage Cloud to specify a different URL
    - Another public PaaS or for your private cloud
  - Note there is a Signup button here

Same email and password
That you signed-up with

Click Finish when done



New Server

Cloud Foundry Account

ⓘ Enter an email address.

Account Information

Email:

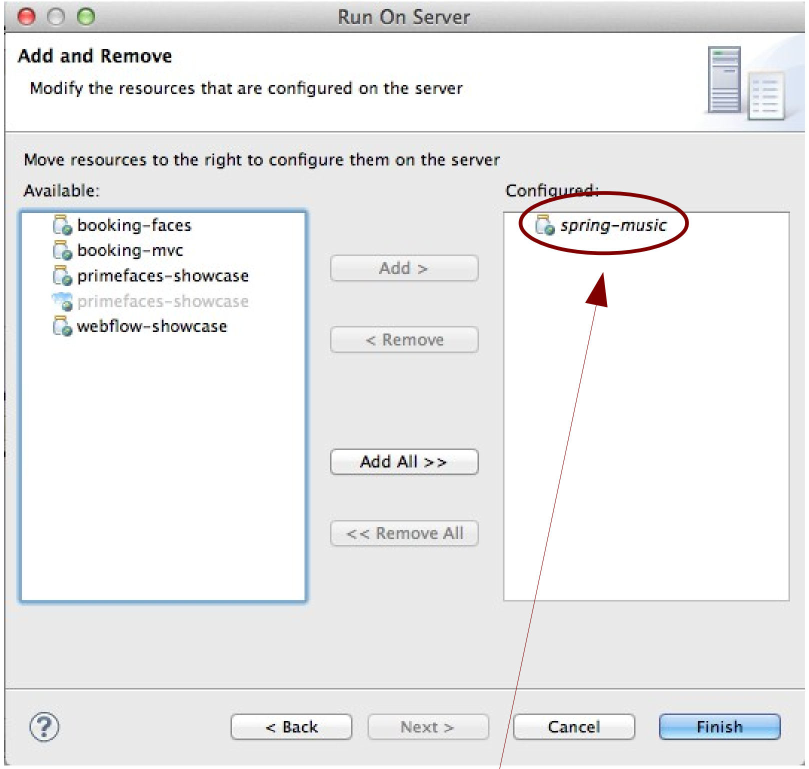
Password:

URL: Pivotal Cloud Foundry Hosted Developer Edition - https://api.run.pivotal.io

Roadmap

- Get Setup
- **Deploying an Application**
- Managing Application Instances

Deployment


Run On Server

Add and Remove

Modify the resources that are configured on the server

Move resources to the right to configure them on the server

Available:

- booking-faces
- booking-mvc
- primefaces-showcase
- primefaces-showcase
- webflow-showcase

Configured:

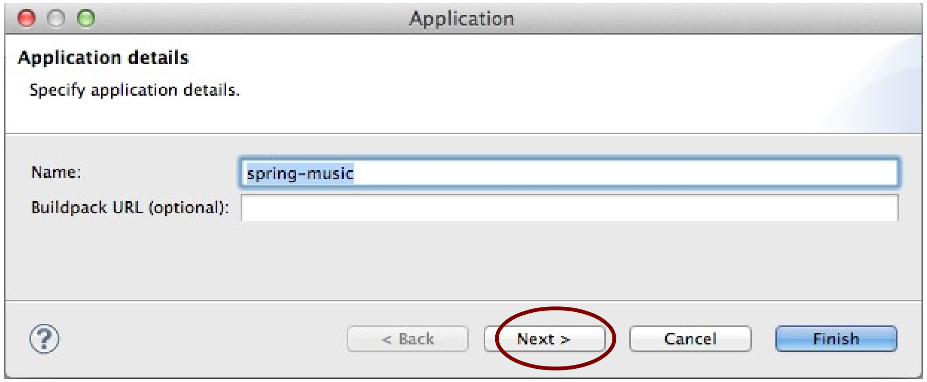
- *spring-music*

- We are now ready to deploy an application
- Select a project in Eclipse
  - Right click and select
    - Run As … → Run on Server
    - *Just like any other server*
  - Select Cloud Foundry server
  - Click Next
  - In next dialog make sure your project is in the RHS list
    - Just as you would normally
  - Click Finish to deploy

Now **things get different ...**

Application details

- Two dialogs appear
  - Application details … just click next for now
  - Launch deployment … Pick a *unique* URL
  - All apps deploy to <sub-domain>.cfapps.io
  - For now, just click *Finish* to deploy

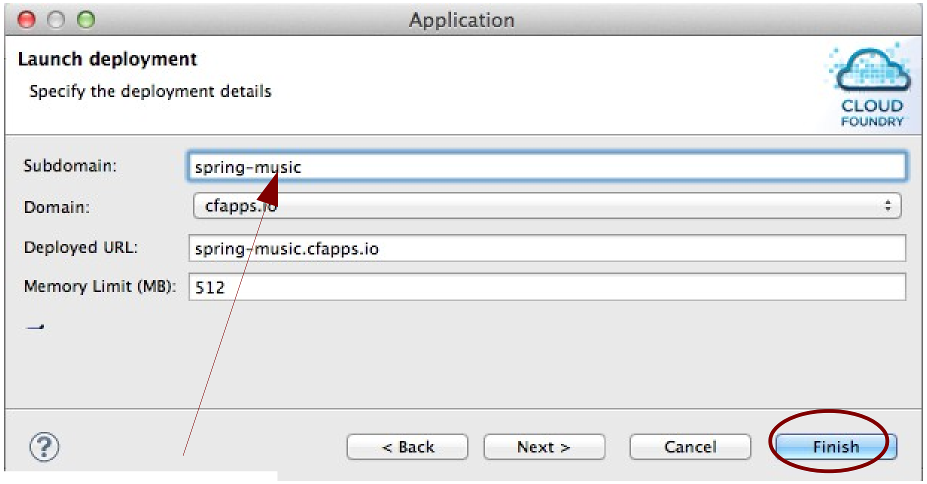

Application

Application details

Specify application details.

Name: spring-music

Buildpack URL (optional):


Application

Launch deployment

Specify the deployment details

Subdomain: spring-music

Domain: cfapps.io

Deployed URL: spring-music.cfapps.io

Memory Limit (MB): 512

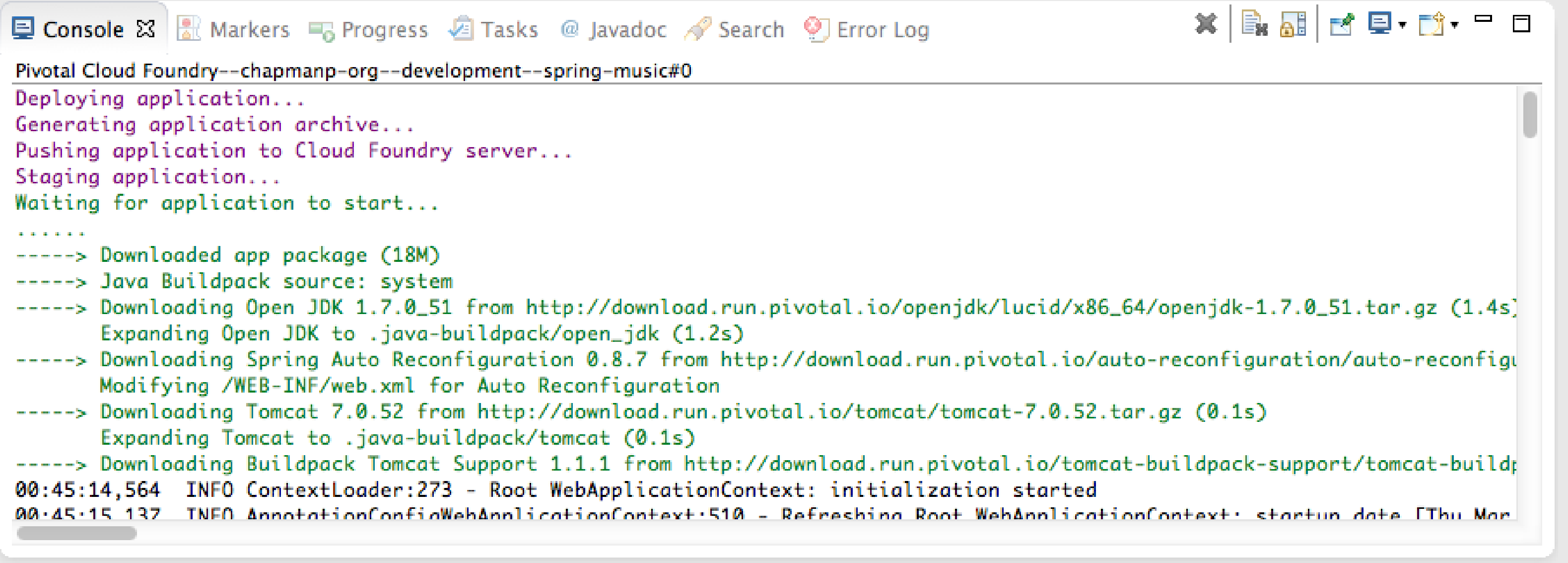
Will this be unique? Change sub-domain to make sure

What just happened?

- Eclipse Connected to Cloud Foundry using your credentials
- It 'pushed' your application to CF and told it to deploy it
  - The whole application is uploaded – takes a while
  - CF “staged” your application
    - Recognized Java / WAR, prepared a “droplet” containing JRE and Tomcat server
  - “Droplet” was deployed to a container and began running
  - All requests to the *Deployed URL* route to your application
- <u>Same process as when using the CLI</u>

Watching it Run

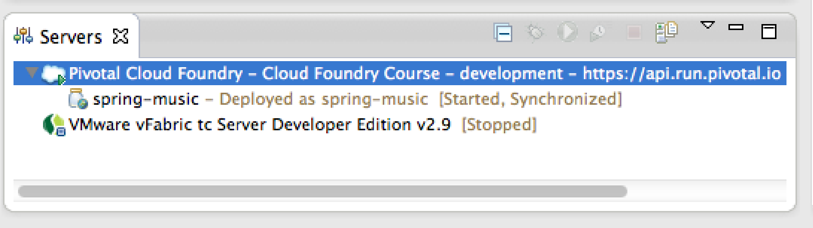
- In the Console View



Pivotal Cloud Foundry--chapmanp-org--development--spring-music#0

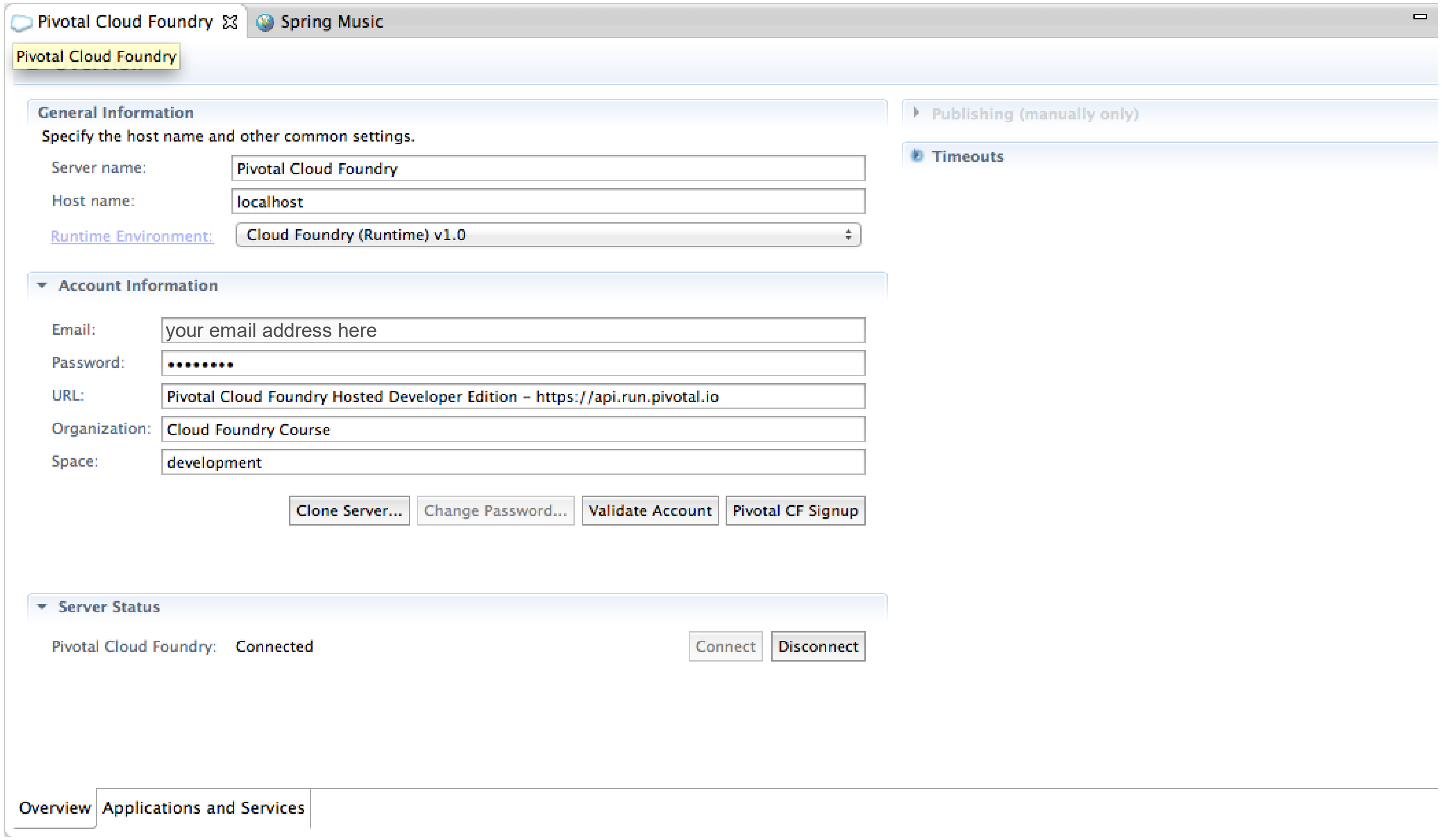
```
Deploying application...
Generating application archive...
Pushing application to Cloud Foundry server...
Staging application...
Waiting for application to start...
......
-----> Downloaded app package (18M)
-----> Java Buildpack source: system
-----> Downloading Open JDK 1.7.0_51 from http://download.run.pivotal.io/openjdk/lucid/x86_64/openjdk-1.7.0_51.tar.gz (1.4s)
       Expanding Open JDK to .java-buildpack/open_jdk (1.2s)
-----> Downloading Spring Auto Reconfiguration 0.8.7 from http://download.run.pivotal.io/auto-reconfiguration/auto-reconfigu
       Modifying /WEB-INF/web.xml for Auto Reconfiguration
-----> Downloading Tomcat 7.0.52 from http://download.run.pivotal.io/tomcat/tomcat-7.0.52.tar.gz (0.1s)
       Expanding Tomcat to .java-buildpack/tomcat (0.1s)
-----> Downloading Buildpack Tomcat Support 1.1.1 from http://download.run.pivotal.io/tomcat-buildpack-support/tomcat-buildp
00:45:14,564  INFO ContextLoader:273 - Root WebApplicationContext: initialization started
00:45:15,137  INFO AnnotationConfigWebApplicationContext:510 - Refreshing Root WebApplicationContext: startup date [Thu Mar
```

- In the Dashboard
  - Double click



Pivotal Cloud Foundry – Cloud Foundry Course – development – https://api.run.pivotal.io
spring-music – Deployed as spring-music [Started, Synchronized]
VMware vFabric tc Server Developer Edition v2.9 [Stopped]

Overview Dashboard Tab



General Information

Specify the host name and other common settings.

Server name: Pivotal Cloud Foundry

Host name: localhost

Runtime Environment: Cloud Foundry (Runtime) v1.0

Account Information

Email: your email address here

Password: ••••••••

URL: Pivotal Cloud Foundry Hosted Developer Edition – https://api.run.pivotal.io

Organization: Cloud Foundry Course

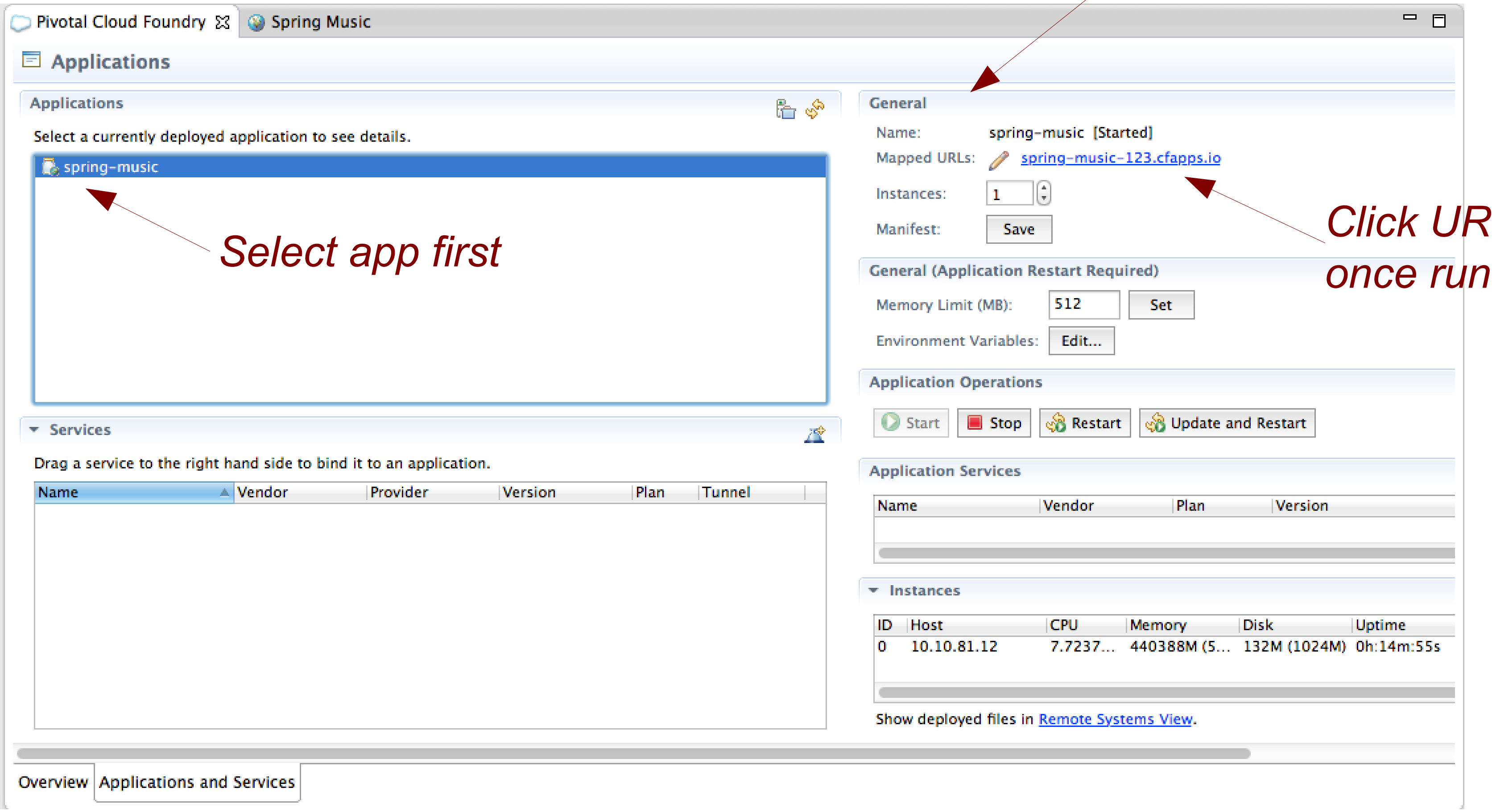
Space: development

Server Status

Pivotal Cloud Foundry: Connected

Applications Tab

Application status here



Pivotal Cloud Foundry | Spring Music

Applications

Applications

Select a currently deployed application to see details.

spring-music

Select app first

Services

Drag a service to the right hand side to bind it to an application.

Name	Vendor	Provider	Version	Plan	Tunnel

General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

Instances: 1

Manifest: Save

Click URL once running

General (Application Restart Required)

Memory Limit (MB): 512 Set

Environment Variables: Edit...

Application Operations

Application Services

Name	Vendor	Plan	Version

Instances

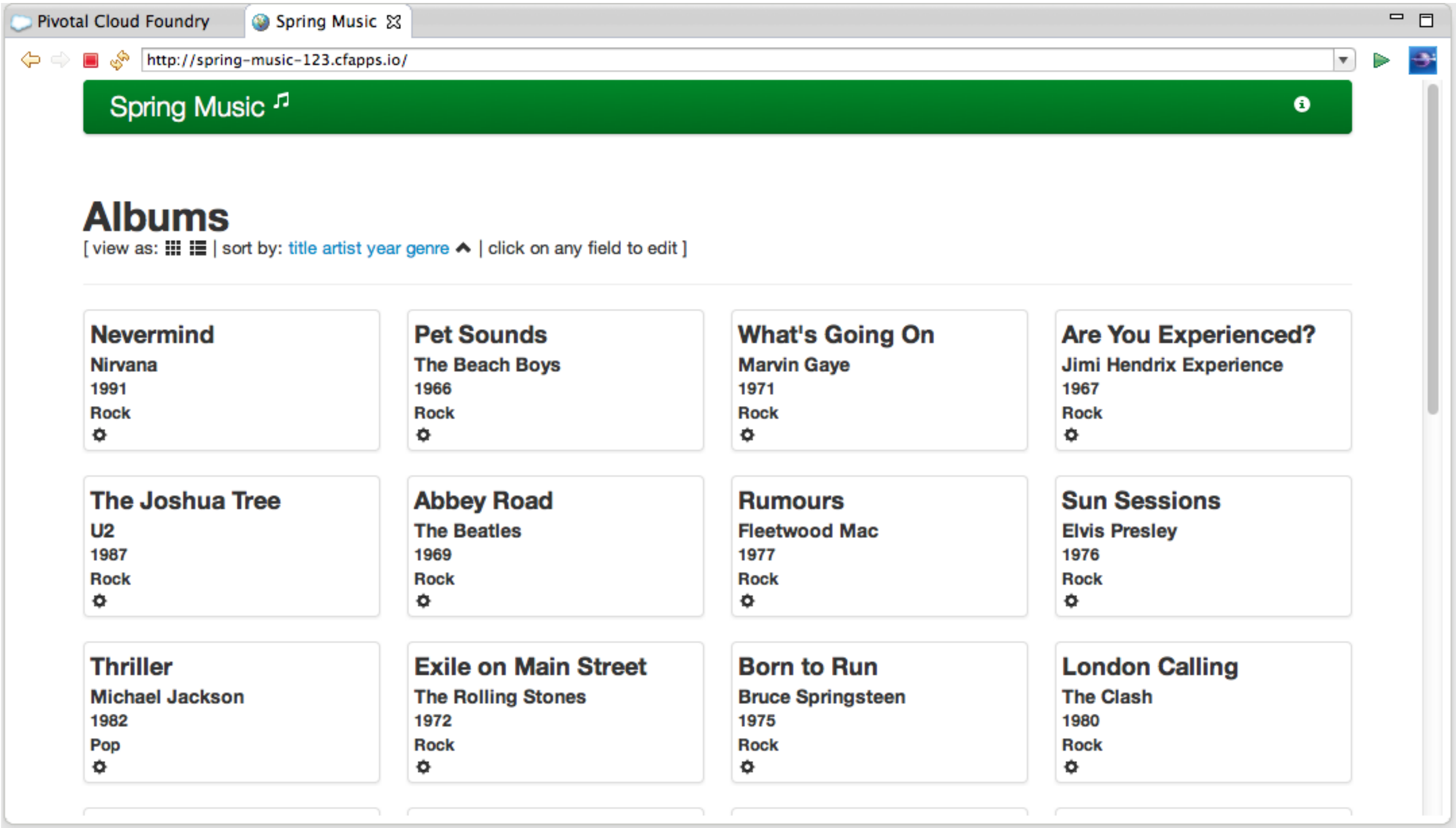
ID	Host	CPU	Memory	Disk	Uptime
0	10.10.81.12	7.7237...	440388M (5...	132M (1024M)	0h:14m:55s

Show deployed files in Remote Systems View.

Note URL: added -123 to make it *unique*

See Your Application Running in Eclipse

- Eclipse pops up a browser window open at your URL
  - Or use the browser of your choice



Albums

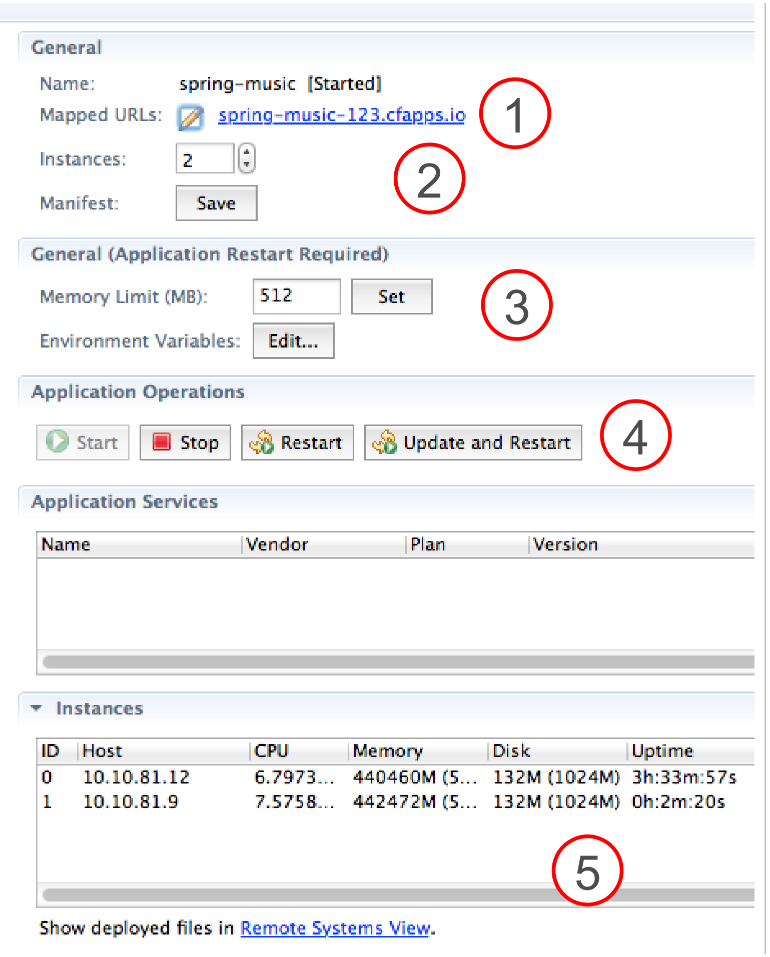
[view as: | sort by: title artist year genre | click on any field to edit]

Nevermind Nirvana 1991 Rock	**Pet Sounds** The Beach Boys 1966 Rock	**What's Going On** Marvin Gaye 1971 Rock	**Are You Experienced?** Jimi Hendrix Experience 1967 Rock
The Joshua Tree U2 1987 Rock	**Abbey Road** The Beatles 1969 Rock	**Rumours** Fleetwood Mac 1977 Rock	**Sun Sessions** Elvis Presley 1976 Rock
Thriller Michael Jackson 1982 Pop	**Exile on Main Street** The Rolling Stones 1972 Rock	**Born to Run** Bruce Springsteen 1975 Rock	**London Calling** The Clash 1980 Rock

Roadmap

- Getting Setup
- Deploying an Application
- **Managing Application Instances**

Cloud Foundry Dashboard



General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

Instances: 2

Manifest: Save

General (Application Restart Required)

Memory Limit (MB): 512 Set

Environment Variables: Edit...

Application Operations

Application Services

Name	Vendor	Plan	Version

Instances

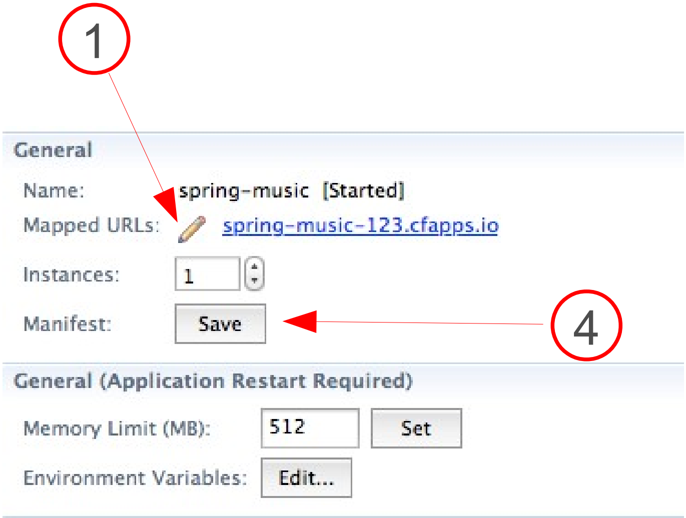
ID	Host	CPU	Memory	Disk	Uptime
0	10.10.81.12	6.7973...	440460M (5...	132M (1024M)	3h:33m:57s
1	10.10.81.9	7.5758...	442472M (5...	132M (1024M)	0h:2m:20s

Show deployed files in Remote Systems View.

- The right-side panel of Applications and Services tab
  - Below *General*
- Control your application
  1. Change mapped URL
  2. Add/remove instances
  3. Change memory
  4. Stop/start
  5. Monitor instances

Change Mapped URL

- Click pencil (edit) icon
- Can add, edit or remove URLs
- Save when done



General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

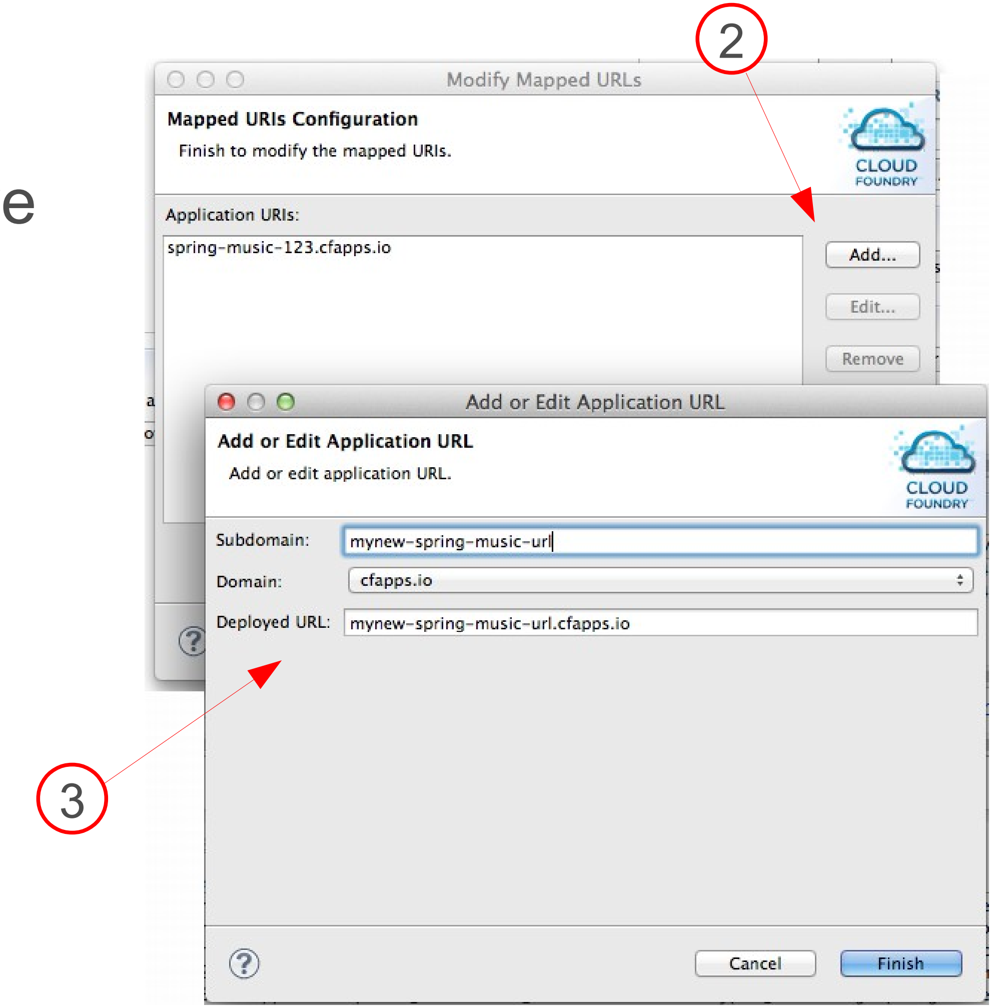
Instances: 1

Manifest: Save

General (Application Restart Required)

Memory Limit (MB): 512 Set

Environment Variables: Edit...



Modify Mapped URLs

Mapped URIs Configuration

Finish to modify the mapped URIs.

Application URIs:

spring-music-123.cfapps.io

Add or Edit Application URL

Add or Edit Application URL

Add or edit application URL.

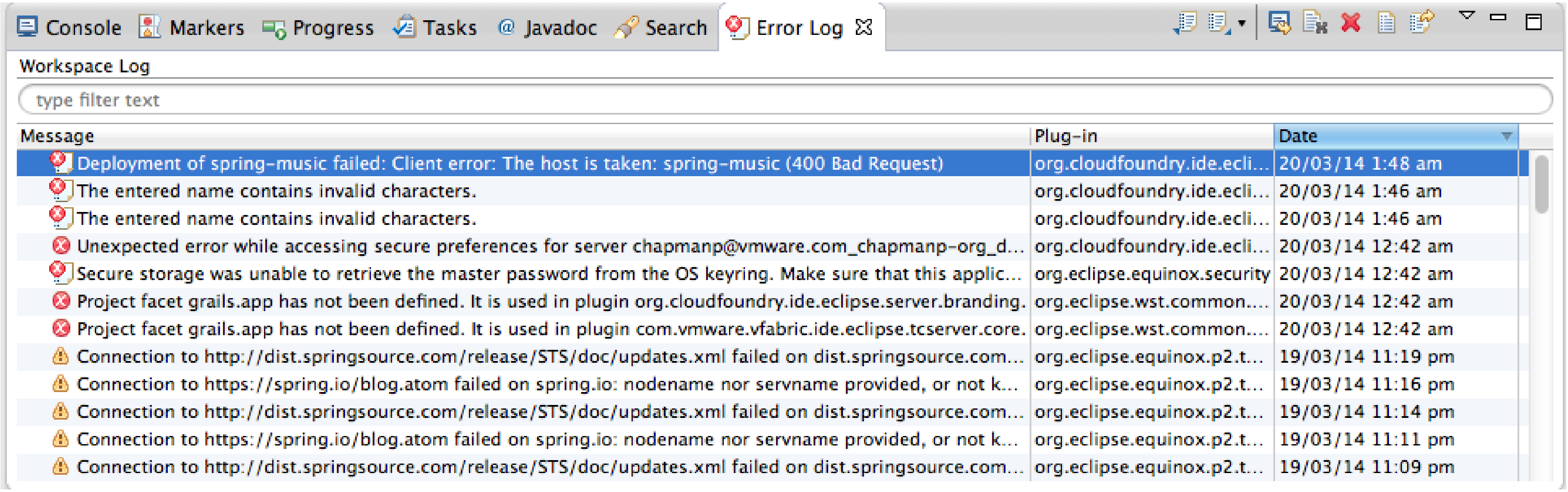
Subdomain: mynew-spring-music-url

Domain: cfapps.io

Deployed URL: mynew-spring-music-url.cfapps.io

Choosing Your URL

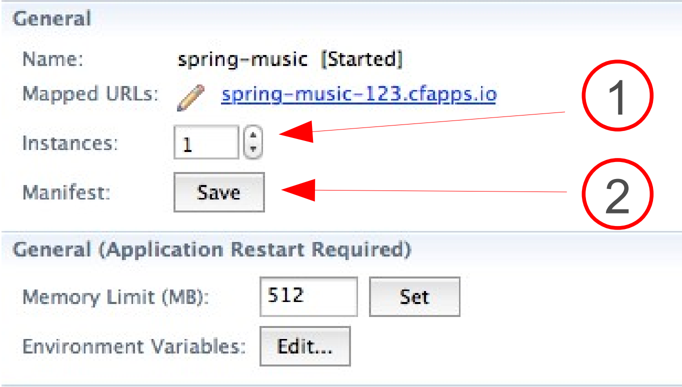
- All applications mapped to cfapps.io domain
- Your URL must be unique
  - Get a Bad Request 400 if you try to use an existing URL



Message	Plug-in	Date
Deployment of spring-music failed: Client error: The host is taken: spring-music (400 Bad Request)	org.cloudfoundry.ide.ecli...	20/03/14 1:48 am
The entered name contains invalid characters.	org.cloudfoundry.ide.ecli...	20/03/14 1:46 am
The entered name contains invalid characters.	org.cloudfoundry.ide.ecli...	20/03/14 1:46 am
Unexpected error while accessing secure preferences for server chapmanp@vmware.com_chapmanp-org_d...	org.cloudfoundry.ide.ecli...	20/03/14 12:42 am
Secure storage was unable to retrieve the master password from the OS keyring. Make sure that this applic...	org.eclipse.equinox.security	20/03/14 12:42 am
Project facet grails.app has not been defined. It is used in plugin org.cloudfoundry.ide.eclipse.server.branding.	org.eclipse.wst.common....	20/03/14 12:42 am
Project facet grails.app has not been defined. It is used in plugin com.vmware.vfabric.ide.eclipse.tcserver.core.	org.eclipse.wst.common....	20/03/14 12:42 am
Connection to http://dist.springsource.com/release/STS/doc/updates.xml failed on dist.springsource.com...	org.eclipse.equinox.p2.t...	19/03/14 11:19 pm
Connection to https://spring.io/blog.atom failed on spring.io: nodename nor servname provided, or not k...	org.eclipse.equinox.p2.t...	19/03/14 11:16 pm
Connection to http://dist.springsource.com/release/STS/doc/updates.xml failed on dist.springsource.com...	org.eclipse.equinox.p2.t...	19/03/14 11:14 pm
Connection to https://spring.io/blog.atom failed on spring.io: nodename nor servname provided, or not k...	org.eclipse.equinox.p2.t...	19/03/14 11:11 pm
Connection to http://dist.springsource.com/release/STS/doc/updates.xml failed on dist.springsource.com...	org.eclipse.equinox.p2.t...	19/03/14 11:09 pm

Instances

- By default one instance of your application runs up
  - Typically a Tomcat server
- To handle large loads you need multiple servers
  - Known as instances
  - Run behind load balancer
- How many instances do I need?
  - Design issue – covered later
- Modify as shown



General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

Instances: 1

Manifest: Save

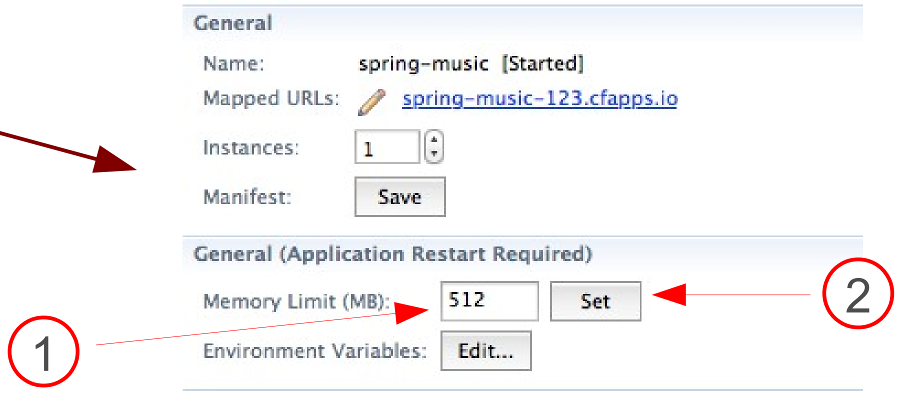
General (Application Restart Required)

Memory Limit (MB): 512 Set

Environment Variables: Edit...

Memory Allocation

- Define how much memory our process gets to run in
  - 512 is the default
    - good for a typical application under test (1 or 2 users)
  - How much memory do I need in production
    - Another good question for later!
- Easy to configure …
  - But the server has to be *restarted*



General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

Instances: 1

Manifest: Save

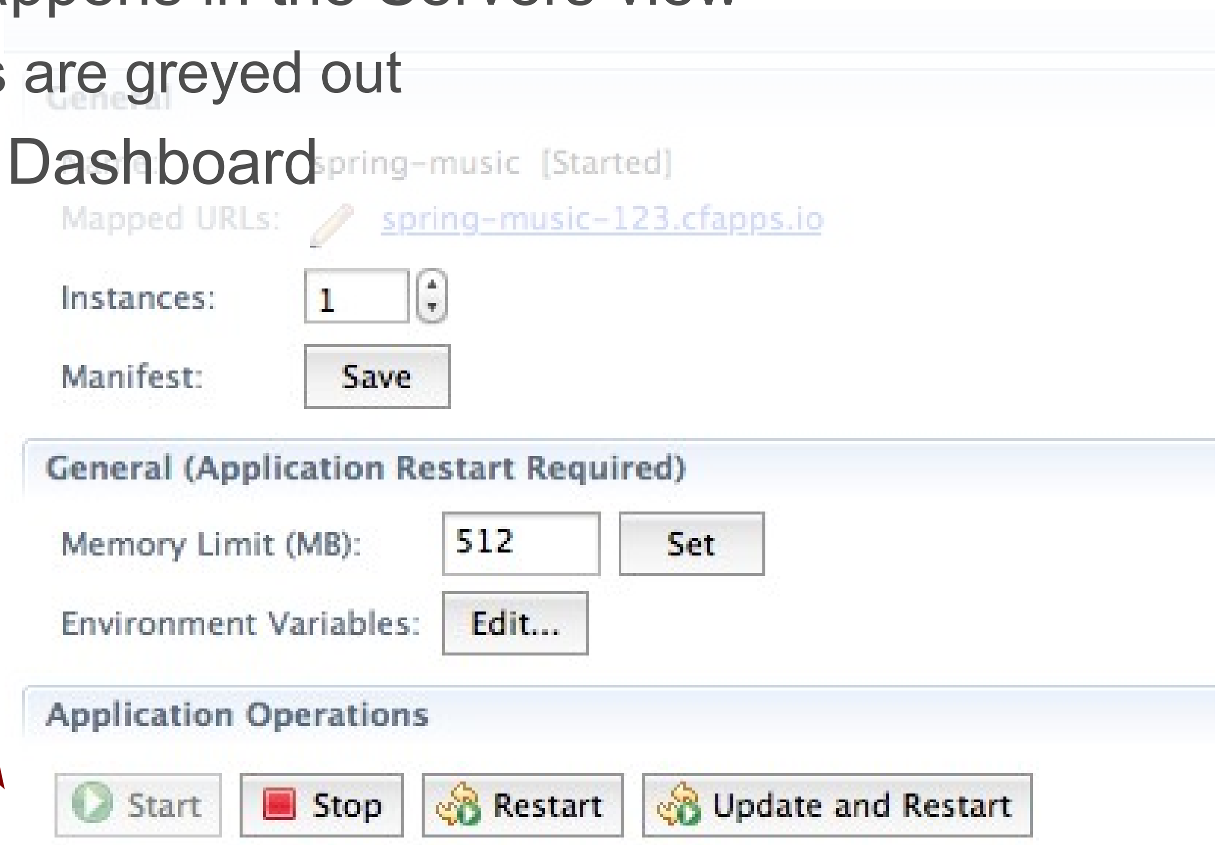
General (Application Restart Required)

Memory Limit (MB): 512 Set

Environment Variables: Edit...

Stopping and Starting

- Normally this happens in the Servers view
  - Those buttons are greyed out
- Instead use the Dashboard



General

Name: spring-music [Started]

Mapped URLs: spring-music-123.cfapps.io

Instances: 1

Manifest:

General (Application Restart Required)

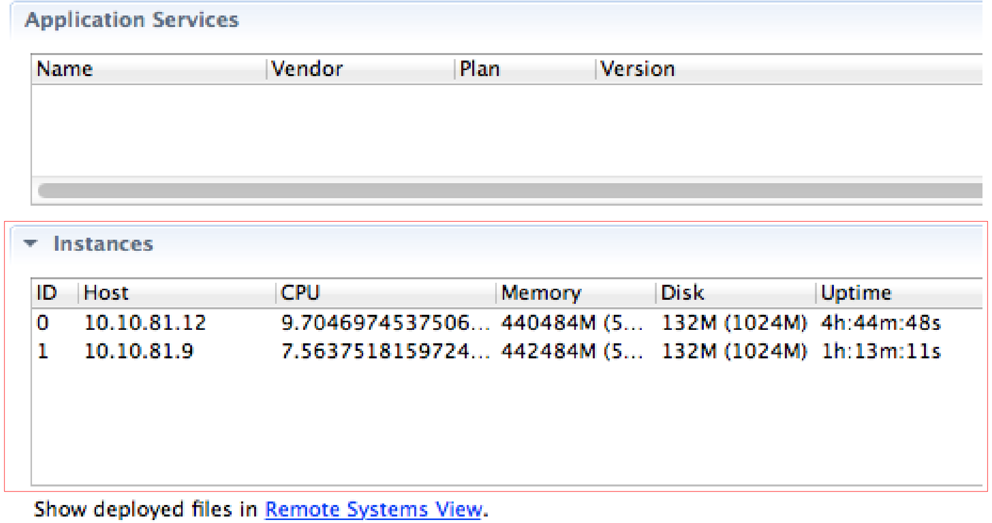
Memory Limit (MB): 512

Environment Variables:

Application Operations

Monitoring Instances

- The very bottom panel shows all your instances
  - Provides statistics
  - *Not* real-time
- To refresh
  - Click refresh icon on the application list



Application Services

Name	Vendor	Plan	Version

Instances

ID	Host	CPU	Memory	Disk	Uptime
0	10.10.81.12	9.7046974537506...	440484M (5...	132M (1024M)	4h:44m:48s
1	10.10.81.9	7.5637518159724...	442484M (5...	132M (1024M)	1h:13m:11s

Show deployed files in Remote Systems View.

Summary: Cloud Foundry Dashboard

- In the Application and Services tab
  - Configure your application (below *General* on right-side)
  - Options
    - Modify mapped URL
    - Change number of instances
    - Change the amount of memory allocated
    - Start and stop the application
    - Monitor instances
  - You may have noticed we missed two options (later)
    - Add or remove services
    - Set environment variables

Summary

- After completing this lesson, you should have learned:
  - Get setup to use Cloud Foundry
    - See Appendices for installing Cloud Foundry plug-in into Eclipse or Spring Tool Suite
  - Deploy an application to CloudFoundry using Eclipse
  - Manage application instances

Lab

Deploy an existing application to Cloud Foundry using Eclipse

Appendices

- Describe installation of Cloud Foundry plug-in into
  - Appendix A: STS
  - Appendix B: Standard Eclipse

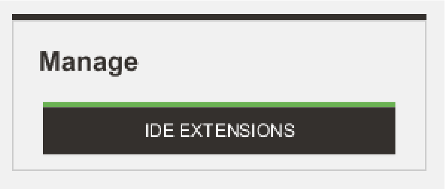
For full details see:

http://docs.cloudfoundry.org/devguide/deploy-apps/sts.html

Appendix A: Installing CF Plugin into STS

- Click Spring leaf icon in STS
  - Displays dashboard

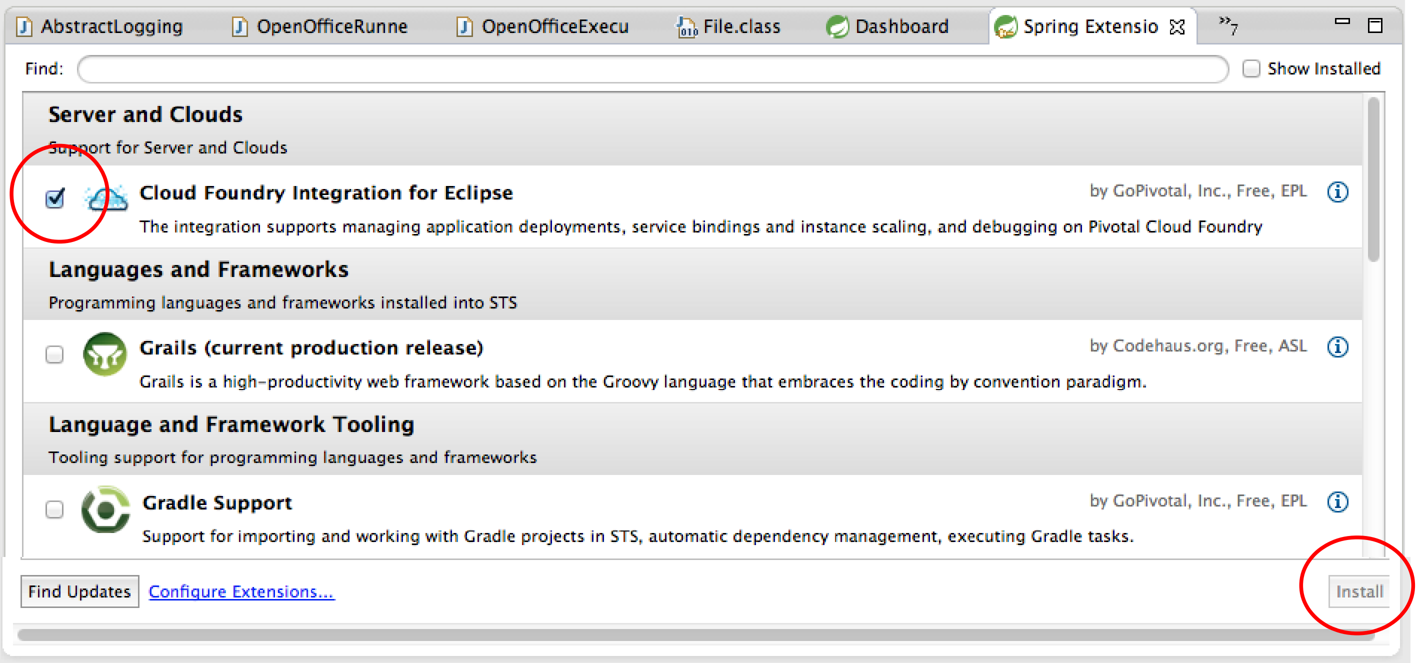
- At bottom right under Manage
  - click “IDE EXTENSIONS”



See: http://docs.cloudfoundry.org/devguide/deploy-apps/sts.html#install-to-sts

Select Cloud Foundry Integration for Eclipse

- Select checkbox and click *Install* button
  - If not listed, enter “cloud foundry” in Find and hit enter



Server and Clouds
Support for Server and Clouds

☑ **Cloud Foundry Integration for Eclipse** by GoPivotal, Inc., Free, EPL
The integration supports managing application deployments, service bindings and instance scaling, and debugging on Pivotal Cloud Foundry

Languages and Frameworks
Programming languages and frameworks installed into STS

□ **Grails (current production release)** by Codehaus.org, Free, ASL
Grails is a high-productivity web framework based on the Groovy language that embraces the coding by convention paradigm.

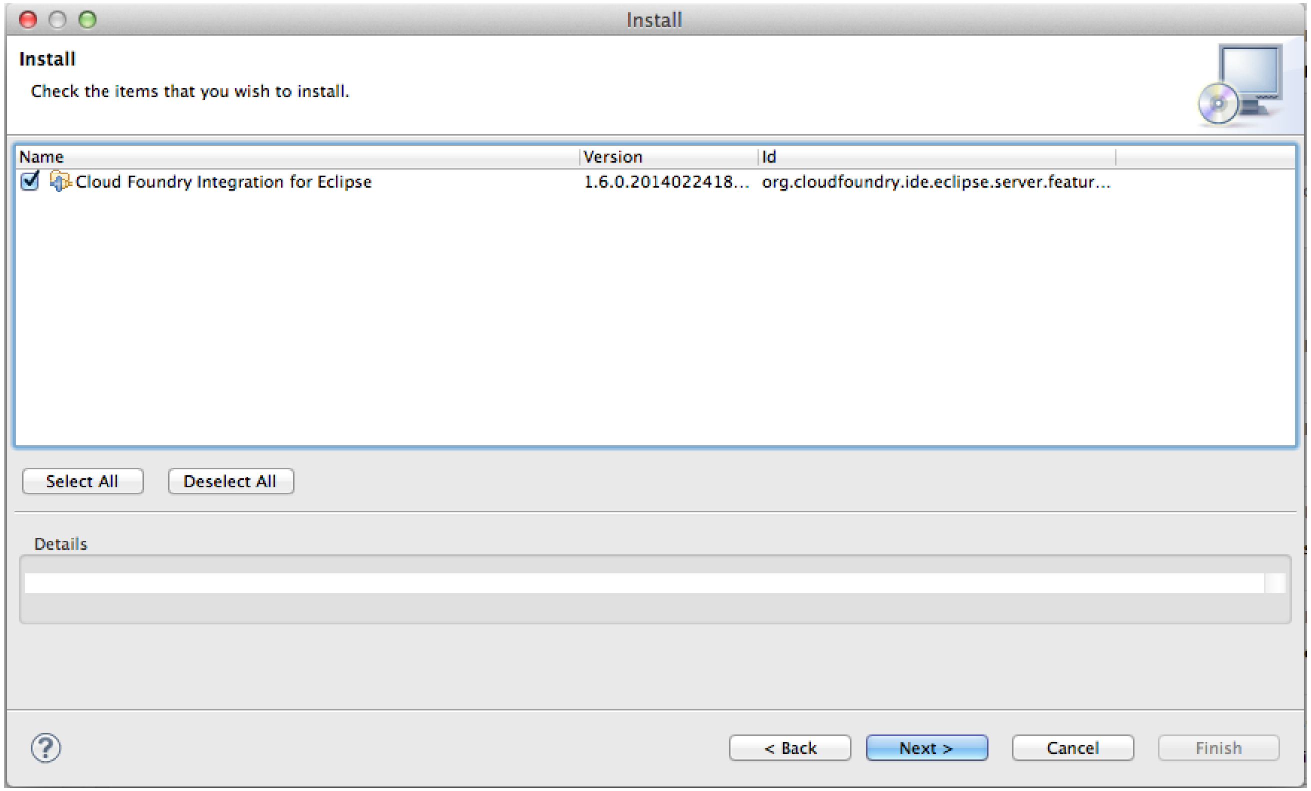
Language and Framework Tooling
Tooling support for programming languages and frameworks

□ **Gradle Support** by GoPivotal, Inc., Free, EPL
Support for importing and working with Gradle projects in STS, automatic dependency management, executing Gradle tasks.

Pivotal

Runs up a Wizard

- Click Next



Install

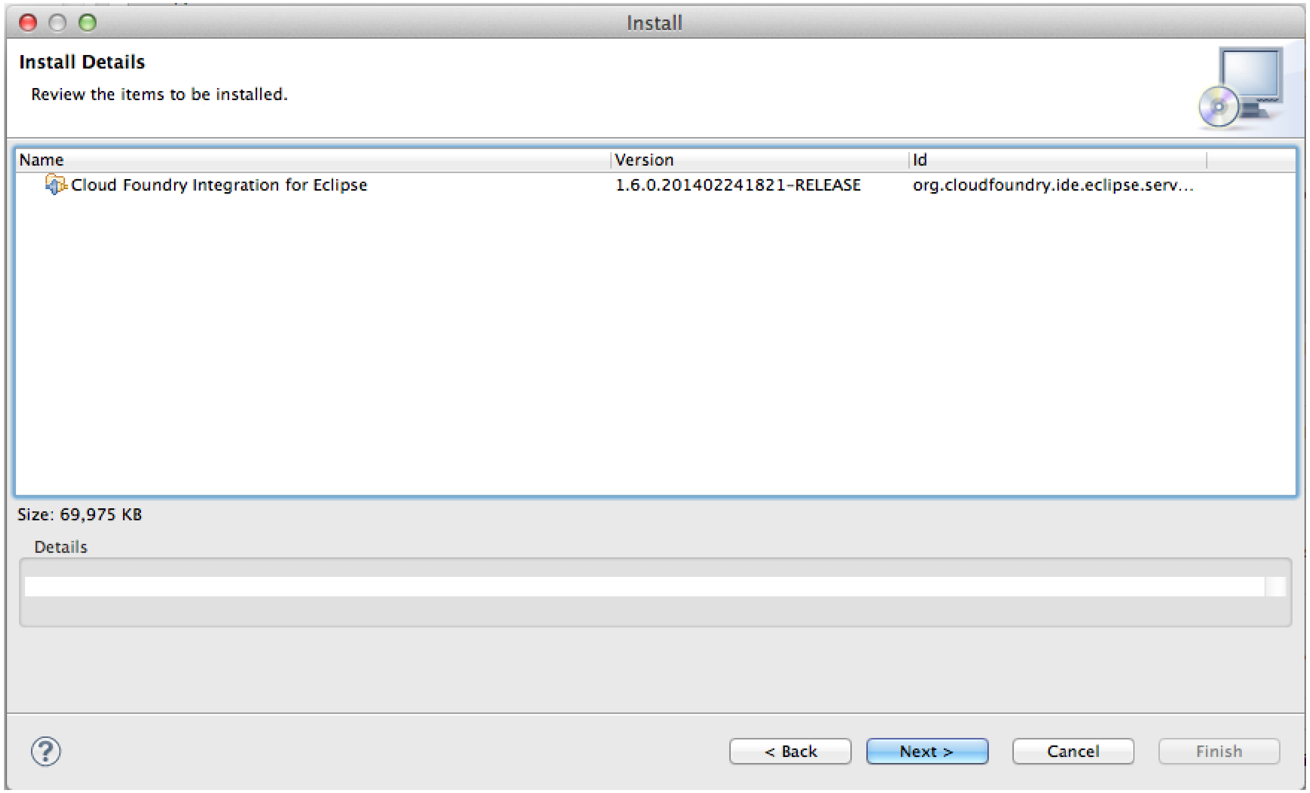
Install

Check the items that you wish to install.

Name	Version	Id
☑ Cloud Foundry Integration for Eclipse	1.6.0.2014022418...	org.cloudfoundry.ide.eclipse.server.featur...

Wizard – Step 2

- Click Next again



Install

Install Details

Review the items to be installed.

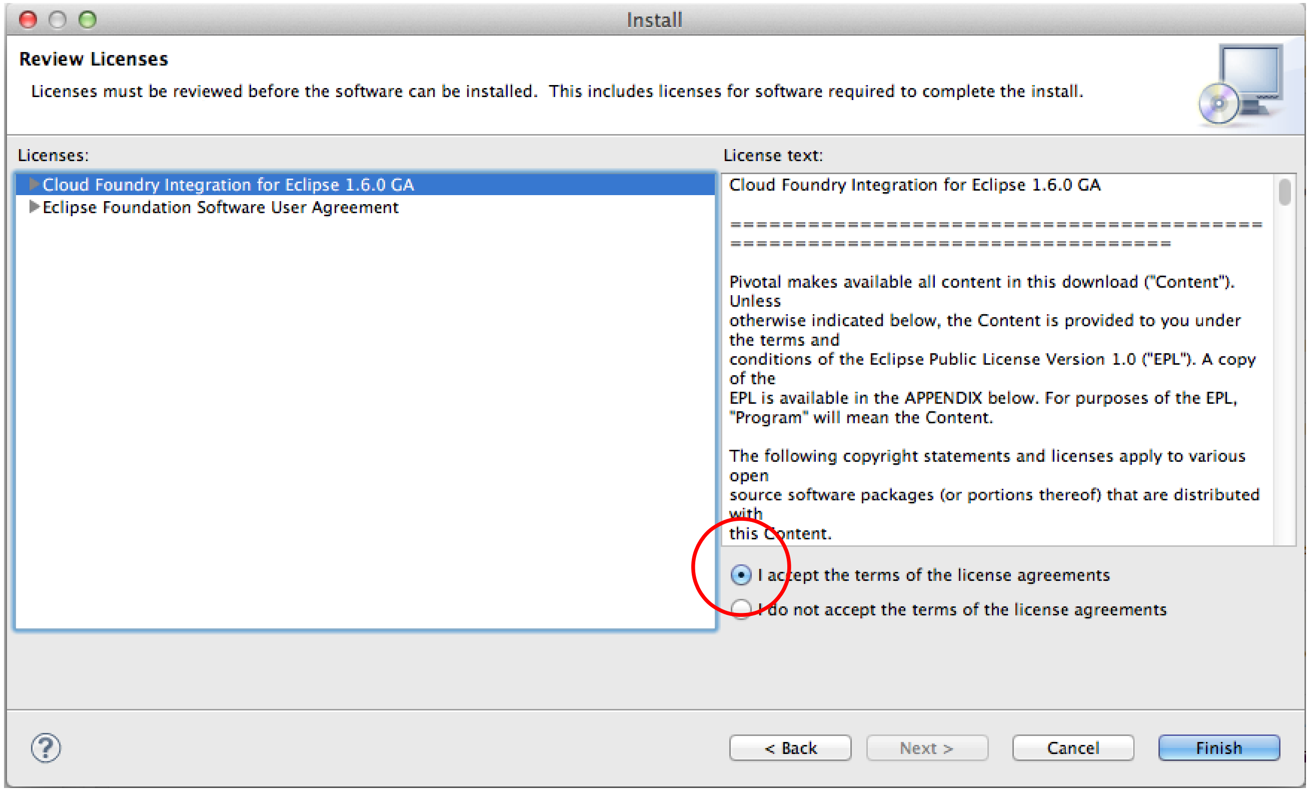
Name	Version	Id
Cloud Foundry Integration for Eclipse	1.6.0.201402241821-RELEASE	org.cloudfoundry.ide.eclipse.serv...

Size: 69,975 KB

Details

Wizard – Step 3

- Accept license agreement, and the installer will run



Review Licenses

Licenses must be reviewed before the software can be installed. This includes licenses for software required to complete the install.

Licenses:

- Cloud Foundry Integration for Eclipse 1.6.0 GA
- Eclipse Foundation Software User Agreement

License text:

Cloud Foundry Integration for Eclipse 1.6.0 GA

```
==========================================
===================================
```

Pivotal makes available all content in this download ("Content"). Unless otherwise indicated below, the Content is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is available in the APPENDIX below. For purposes of the EPL, "Program" will mean the Content.

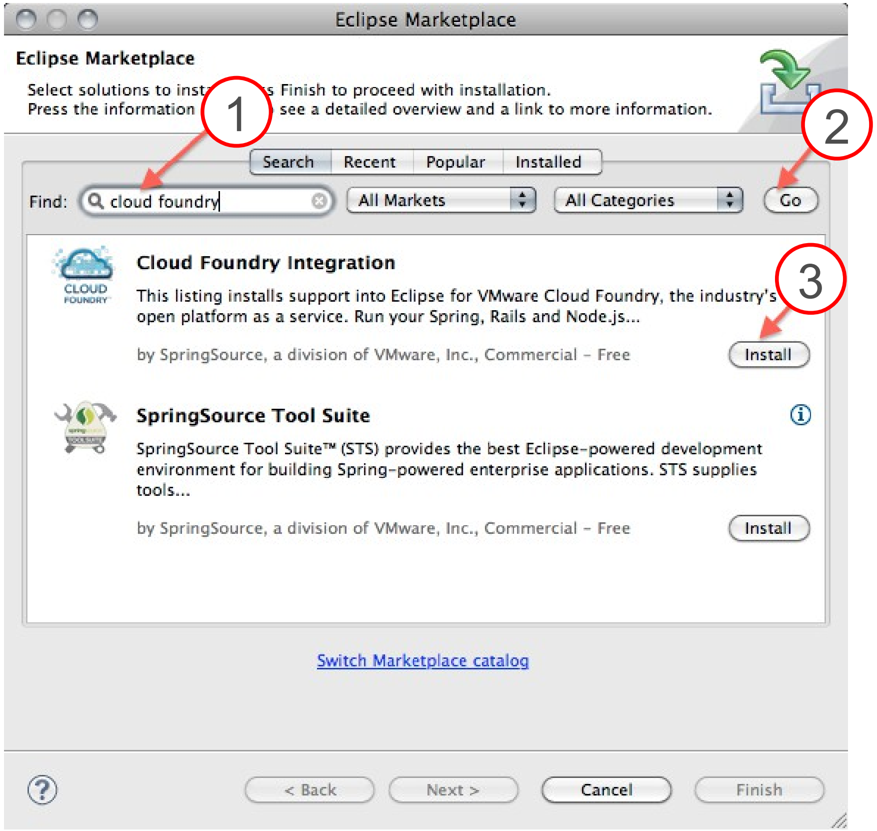
The following copyright statements and licenses apply to various open source software packages (or portions thereof) that are distributed with this Content.

● I accept the terms of the license agreements

○ I do not accept the terms of the license agreements

Appendix B: Installing into Eclipse

- From Help menu
  - Eclipse Marketplace
- Enter “cloud foundry” into Find box
- Click “*Go*”
- Click “Cloud Foundry Integration” *Install* button



Eclipse Marketplace

Select solutions to inst s Finish to proceed with installation.
Press the information see a detailed overview and a link to more information.

Find: cloud foundry

Cloud Foundry Integration

This listing installs support into Eclipse for VMware Cloud Foundry, the industry's open platform as a service. Run your Spring, Rails and Node.js...

by SpringSource, a division of VMware, Inc., Commercial - Free

SpringSource Tool Suite

SpringSource Tool Suite™ (STS) provides the best Eclipse-powered development environment for building Spring-powered enterprise applications. STS supplies tools...

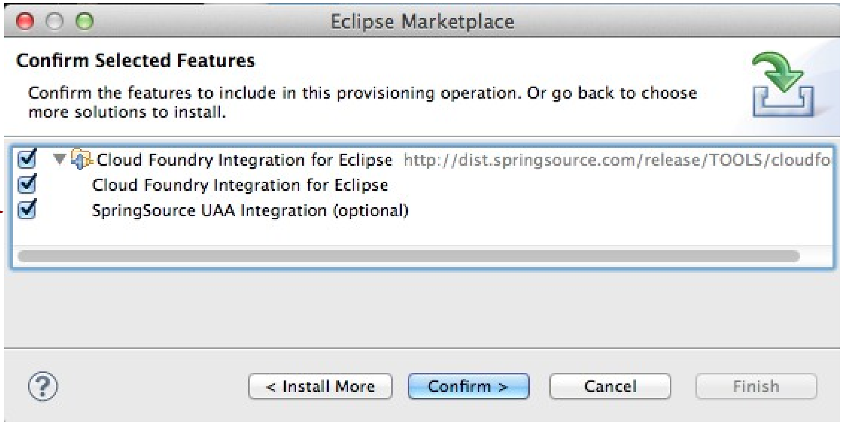
by SpringSource, a division of VMware, Inc., Commercial - Free

See: http://docs.cloudfoundry.org/devguide/deploy-apps/sts.html#install-to-eclipse

Confirm Selected Features

- Popup window lists what will be installed
  - – “Cloud Foundry Integration for Eclipse”
  - – “SpringSource UAA Integration” (optional)
    - Reports tool usage data, *anonymously*
    - Helps us track usage of free software
    - Deselect to stop plugin usage statistics being sent
  - – Click Confirm.

Optional, deselect if you wish →



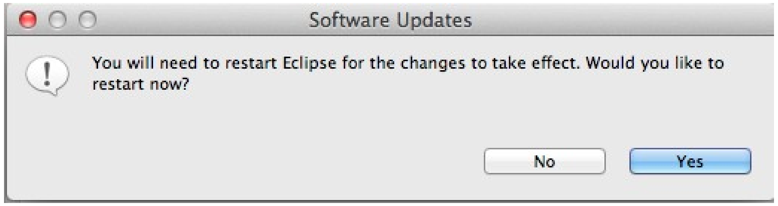
Confirm Selected Features

Confirm the features to include in this provisioning operation. Or go back to choose more solutions to install.

☑ ▼ Cloud Foundry Integration for Eclipse http://dist.springsource.com/release/TOOLS/cloudfo
☑ Cloud Foundry Integration for Eclipse
☑ SpringSource UAA Integration (optional)

Last Few Steps

- Accept the license agreement
- Click Finish
- Installer runs (takes a while)
- Eventually you are asked to restart Eclipse



You will need to restart Eclipse for the changes to take effect. Would you like to restart now?