**PIVOTAL CLOUD FOUNDRY**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Table of Contents** | **Page No** |
| 1 | Cloud Foundry Introduction. | 2 |
| 2 | PCF Architecture | 3 |
| 3 | Advantages of PCF | 4 |
| 4 | Application Deployment process | 4 |
| 5 | Configure Cloud server | 9 |
| 6 | Deploy Sample Applications | 13 |
| 7 | PCF operations | 15 |

**Cloud Foundry Introduction**

Cloud Foundry is an [open source](https://en.wikipedia.org/wiki/Open-source_software), multi cloud application [platform as a service](https://en.wikipedia.org/wiki/Platform_as_a_service) (PaaS) governed by the Cloud Foundry Foundation organization. The software was originally developed by [VMware](https://en.wikipedia.org/wiki/VMware) and then transferred to [Pivotal Software](https://en.wikipedia.org/wiki/Pivotal_Software), a joint venture by [EMC](https://en.wikipedia.org/wiki/EMC_Corporation), [VMware](https://en.wikipedia.org/wiki/VMware) and [General Electric](https://en.wikipedia.org/wiki/General_Electric).

Cloud Foundry is promoted for [continuous delivery](https://en.wikipedia.org/wiki/Continuous_delivery) as it supports the full application development lifecycle, from initial development through all testing stages to deployment. Cloud Foundry’s container-based architecture runs apps in any programming language over a variety of cloud service providers. This multi-cloud environment allows developers to leverage the cloud platform that suits specific app workloads and move those workloads as necessary within minutes with no changes to the app.

The Cloud Foundry platform is available from either the Cloud Foundry Foundation as open source software or from multiple commercial providers as either a product or a service. Cloud Foundry is [open source software](https://en.wikipedia.org/wiki/Open_source_software) and hence available to anyone. Deploying Cloud Foundry involves interfacing with the underlying infrastructure using the Cloud Foundry [BOSH](https://en.wikipedia.org/wiki/BOSH_(software)) deployment system, another open source toolgoverned by the Cloud Foundry Foundation.

**PCF Architecture**

**Pivotal Cloud Foundry (PCF)** is commercial version of Cloud Foundry product .It provides extra tools for installation and administration not included in the open software product. Pivotal Web Services (PWS), however, is an instance of Pivotal Cloud Foundry hosted on [Amazon Web Services](https://en.wikipedia.org/wiki/Amazon_Web_Services) (AWS).

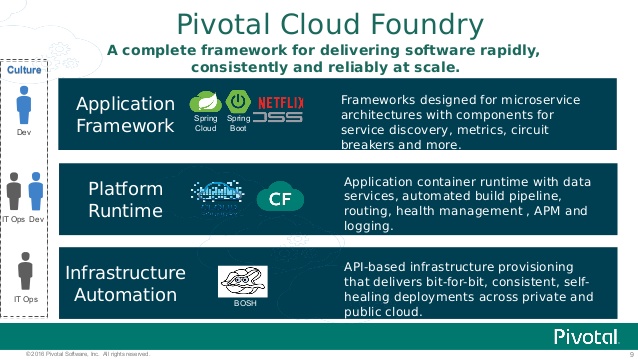
****

Figure: Functional Architecture of PCF.

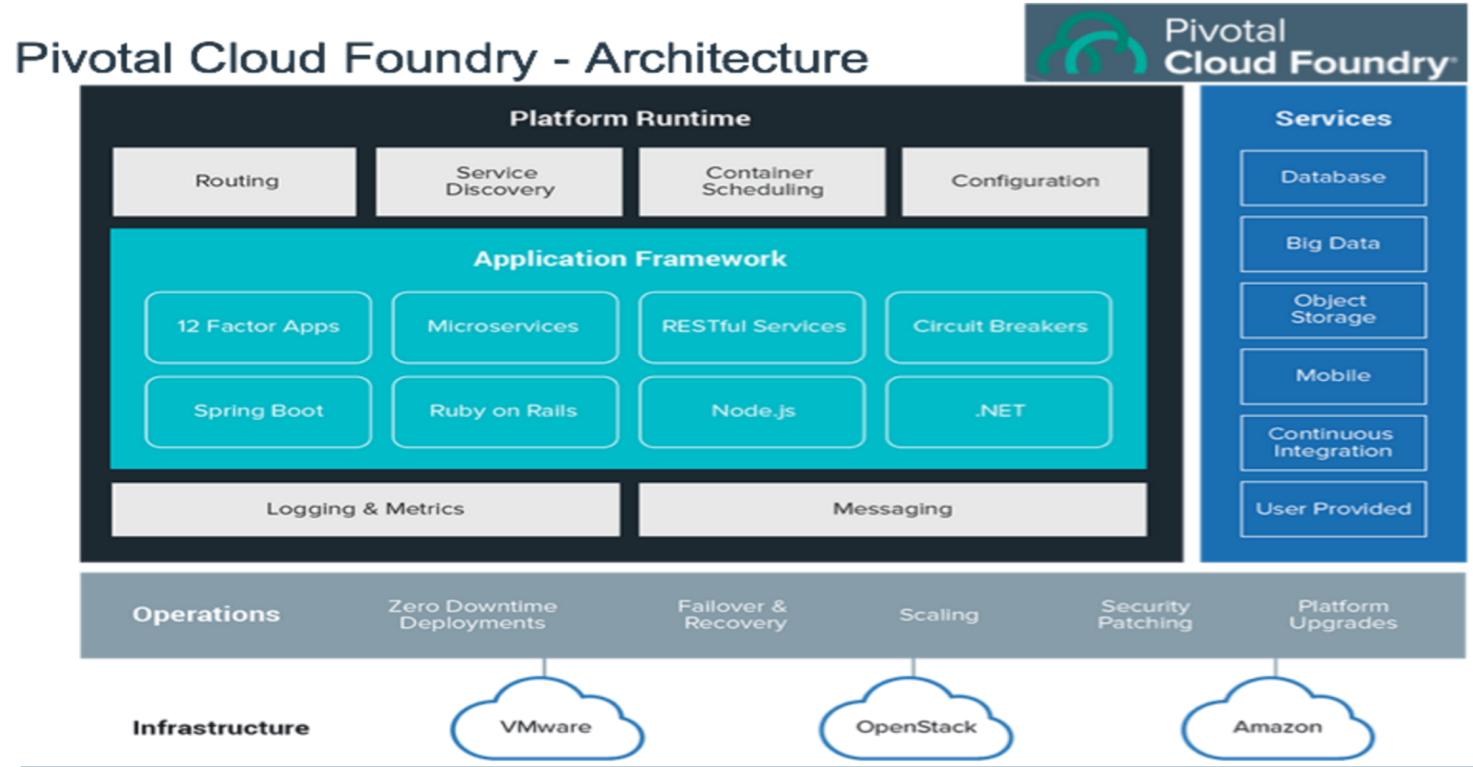
****

Figure : Technical Architecture of PCF.

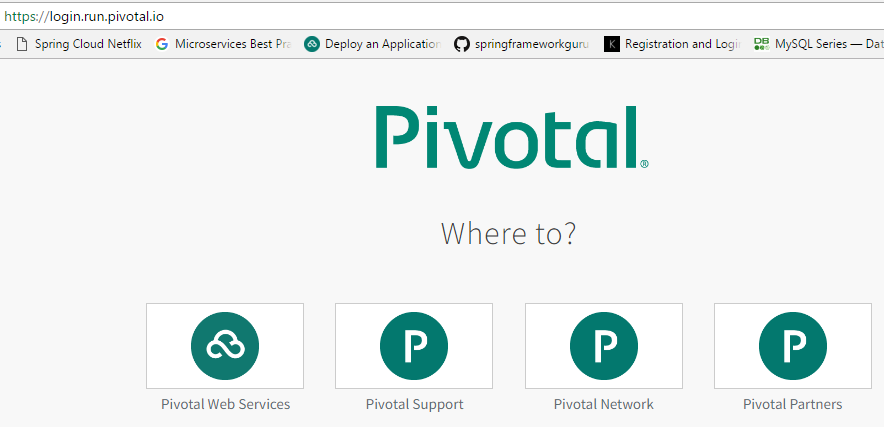
**Advantages of PCF**

* **Pivotal Cloud Foundry is the best platform for Spring and Spring Boot Applications.**
* [Spring’s](https://pivotal.io/spring-app-framework) microservice patterns and[Spring Boot’s](https://pivotal.io/spring-app-framework#buildanything) executable jars are ready-made for Pivotal Cloud Foundry.
* **Spring Cloud Services** deliver turnkey microservice operations and security on Pivotal Cloud Foundry. Includes Config Server, Service Registry, and Circuit Breaker Dashboard.
* **A Native Windows and .NET experience.** Use familiar .NET tools and the .NET Hosted Web Core buildpack. Push applications to containers running on Windows Server 2012 R2.
* **PCF Metrics** is monitoring re-imaged for microservices. Intuitive, time-based displays of events, logs, metrics, and distributed tracing show the full picture of system behavior.
* **Container-ready.** Pivotal Cloud Foundry supports the OCI format for Docker images. Run all types of containerized applications, regardless of protocol: HTTP/s and TCP routes are both supported.
* **Buildpacks** provide framework and runtime dependencies for your apps. Write your code, then cf push. The platform then provides everything it needs to operate.
* **App Autoscaler** automates horizontal scaling. Add more capacity to Pivotal Cloud Foundry applications when traffic spikes. Scale down again as traffic returns to normal. Set thresholds and triggers.
* **Fully integrated with CI/CD.** Automate your software build, test, and delivery pipelines with Pivotal Cloud Foundry and CI/CD tools like [Concourse](http://concourse.ci/). Treat every commit as a delivery!
* Use **Service Brokers** to easily extend your apps with a growing ecosystem of databases, API gateways, and more. Choose from [dozens of add-on services](https://pivotal.io/platform/services).

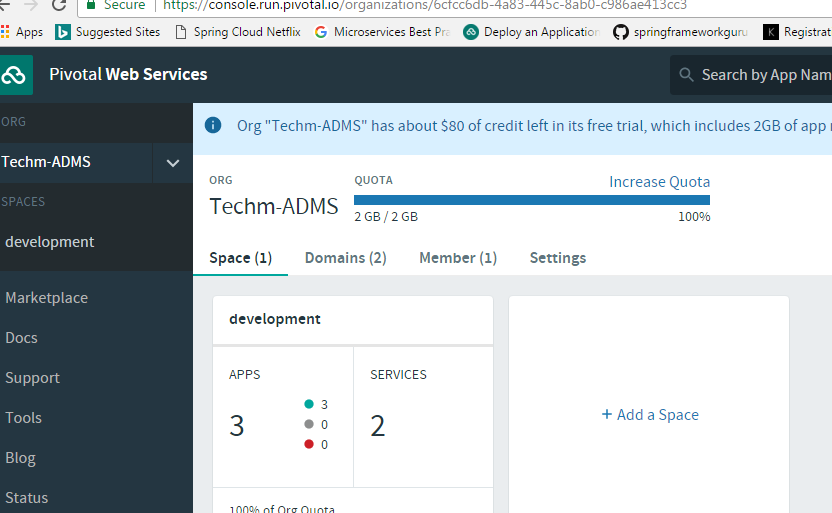
**Application Deployment process**

Before Deploying apps in cloud ensure you must have an account in Pivotal cloud Foundry.

<https://login.run.pivotal.io/login>



Select -> Pivotal Web Service.



We have two types of deployment options available in PCF.

1. Command Line Tool.
2. Eclipse Plugin.
3. **Command Line Tool**

Download and install the Cloud Foundry Command Line Interface (cf CLI).

Try the following command to test that the cf CLI works:

cf help

You can use the cf CLI to perform all commands on apps deployed to PCF.

**Deploy the Sample App**

Now that you have the cf CLI installed and a Pivotal Web Services (PWS) account, you are really close to deploying the sample app.This sample app is built with Spring Boot to get you up and running as quickly as possible.

Download the app with git:

git clone https://github.com/cloudfoundry-samples/cf-sample-app-spring.git

If you don't have Git installed, you can download a zip file of the app at <https://github.com/cloudfoundry-samples/cf-sample-app-spring/archive/master.zip>

Navigate to the app directory:

cd cf-sample-app-spring

Sign in to PWS:

cf login -a https://api.run.pivotal.io

Push the app to PWS:

cf push

Open the sample app in your browser:

2.Eclipse Plugin.

The Cloud Foundry Eclipse Plugin is an extension that enables Cloud Foundry users to deploy and manage Java and Spring applications on a Cloud Foundry instance from Eclipse or Spring Tool Suite (STS).

You can use the plugin to perform the following actions:

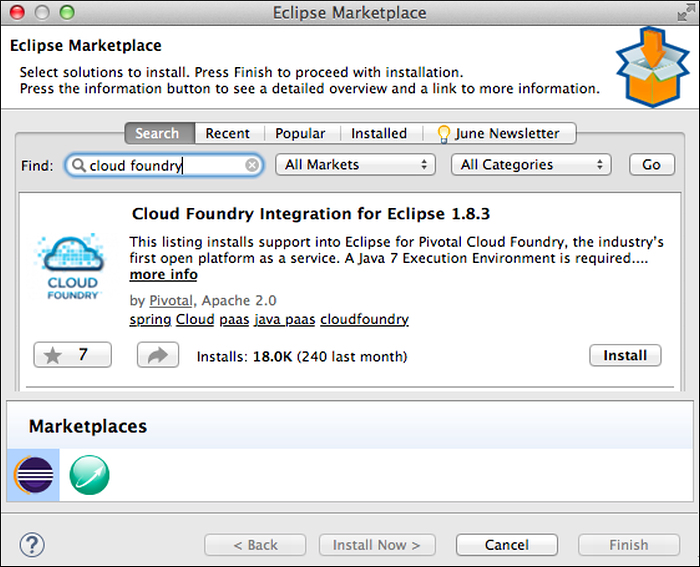
Deploy applications from an Eclipse or STS workspace to a running Cloud Foundry instance. The Cloud Foundry Eclipse plugin supports the following application types:

* Spring Boot
* Spring
* Java Web
* Java standalone
* Grails

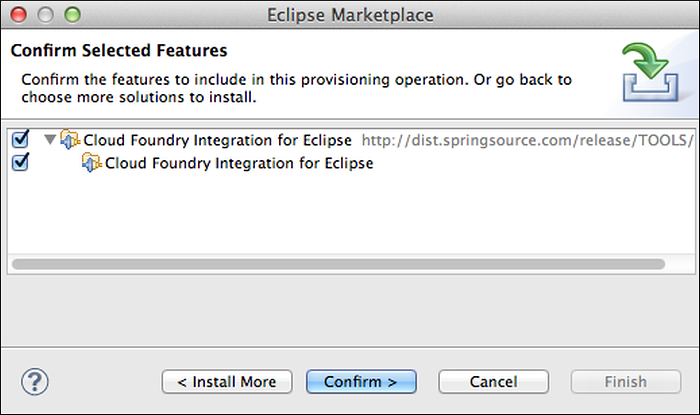
**Install to Eclipse from Marketplace**

Follow the instructions below to install the Cloud Foundry Eclipse Plugin to Eclipse from the Eclipse Marketplace.

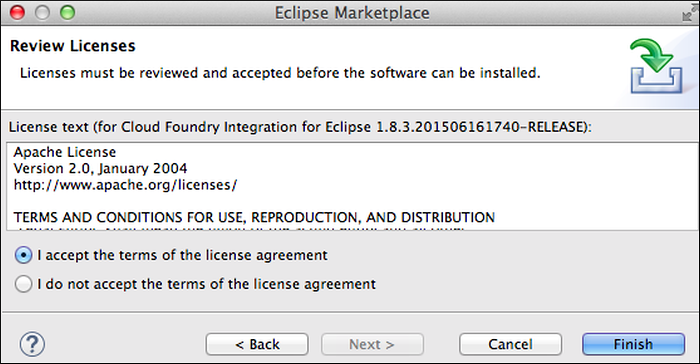
1. Start Eclipse.
2. From the Eclipse **Help** menu, select **Eclipse Marketplace**.
3. In the Eclipse Marketplace window, enter “Cloud Foundry” in the **Find** field. Click **Go**.
4. In the search results, next to the listing for Cloud Foundry Integration, click **Install**.



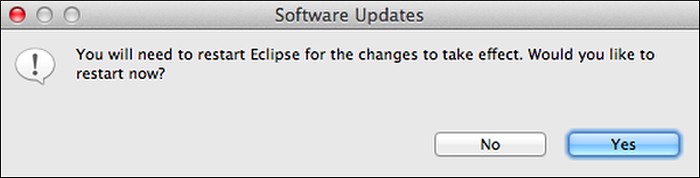
5.In the **Confirm Selected Features** window, click **Confirm**.



6.The **Review Licenses** window appears. Select “I accept the terms of the license agreement” and click **Finish**.



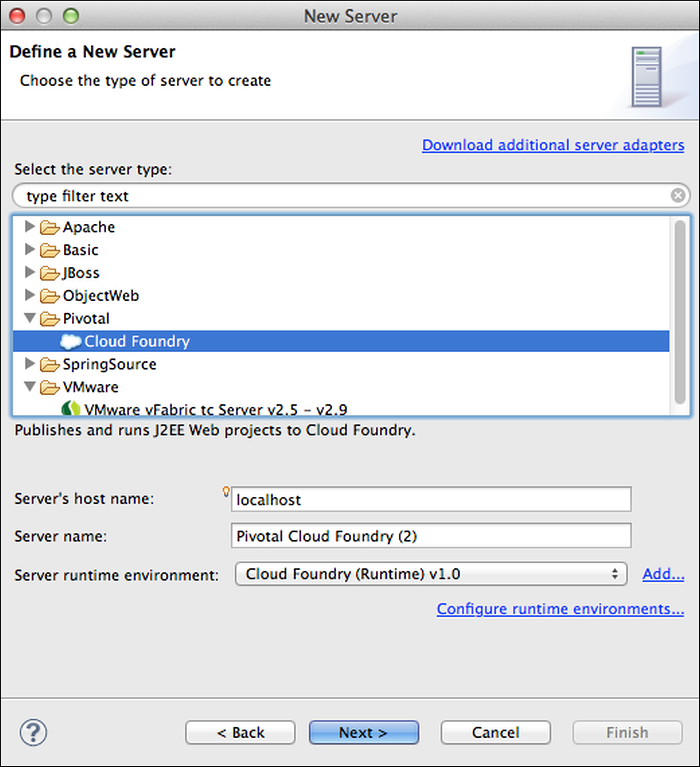
7.The **Software Updates** window appears. Click **Yes** to restart Eclipse.



**Configure Cloud Foundry Server**

You will create a server for each space in Cloud Foundry to which you will deploy applications. Once you create your first Cloud Foundry service instances using the instructions below, you can create additional instances using the [Clone Server](https://docs.run.pivotal.io/buildpacks/java/sts.html#clone) feature.

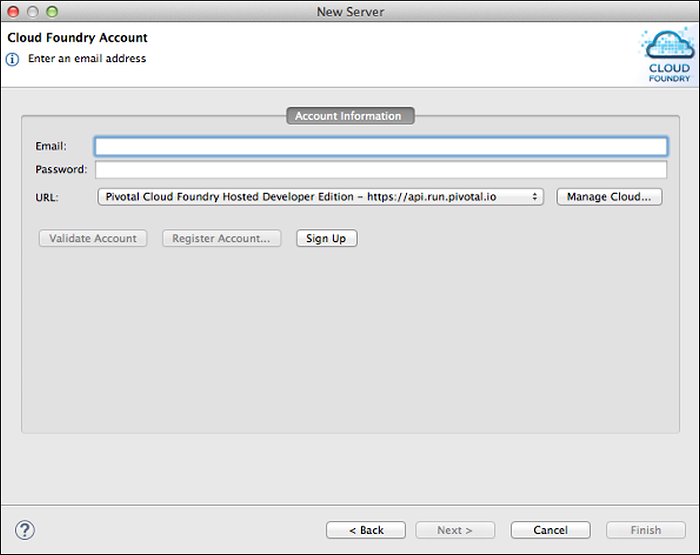
1. Right-click the **Servers** view and select **New > Server**.
2. In the **Define a New Server** window, expand the **Pivotal** folder, select **Cloud Foundry**, and click **Next**.



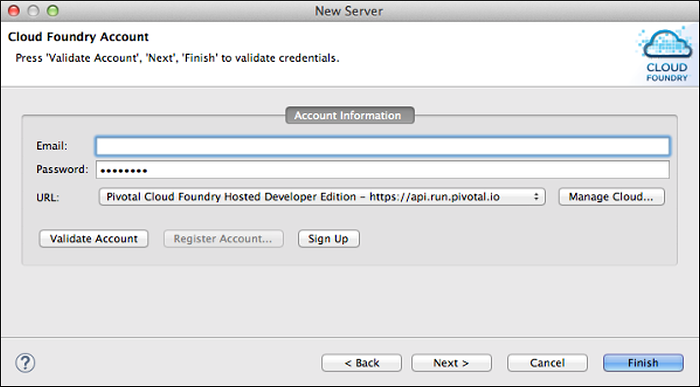
3.In the **Cloud Foundry Account** window, if you already have a Pivotal Cloud Foundry Hosted Developer Edition account, enter your email account and password credentials and click **Validate Account**.

By default, the **URL** field points to the Pivotal Cloud Foundry Hosted Developer Edition URL, <https://api.run.pivotal.io>

If you do not have a Cloud Foundry account and want to register a new Pivotal Cloud Foundry Hosted Developer Edition account, click **Sign Up**. After you create the account, you can complete this procedure.

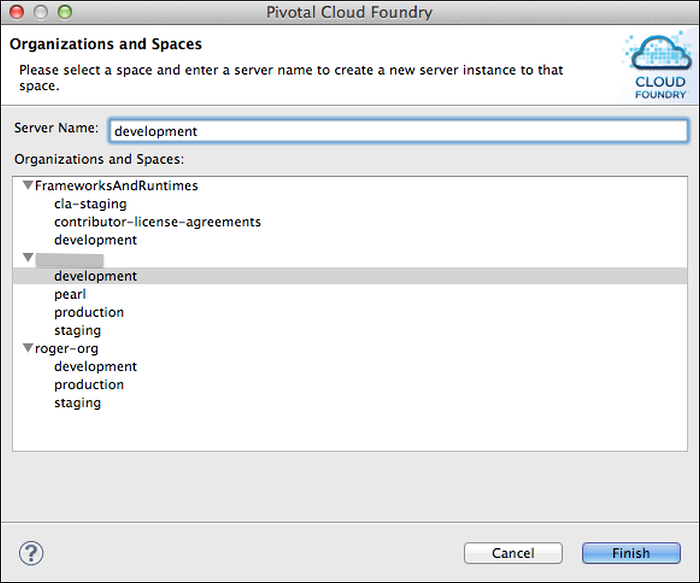


4. The **Cloud Foundry Account** window is refreshed and displays a message indicating whether or not your credentials were valid. Click **Next**.



5.In the **Organizations and Spaces** window, select the space that you want to target, and click **Finish**

**NOTE:** If you do not select a space, the server will be configured to connect to the default space, which is the first encountered in a list of your spaces.



1. Once you have successfully configured the Pivotal Cloud Foundry server, it will appear in the **Servers** view of the Eclipse

**Deploy Sample Application**

To deploy an application to Cloud Foundry using the plugin:

1. To initiate deployment either:

Drag the application from the **Package Explorer** view onto the Pivotal Cloud Foundry server in the **Servers** view, or Right-click the Pivotal Cloud Foundry server in the **Servers** view, select **Add and Remove** from the server context menu, and move the application from the **Available** to the **Configured** column.

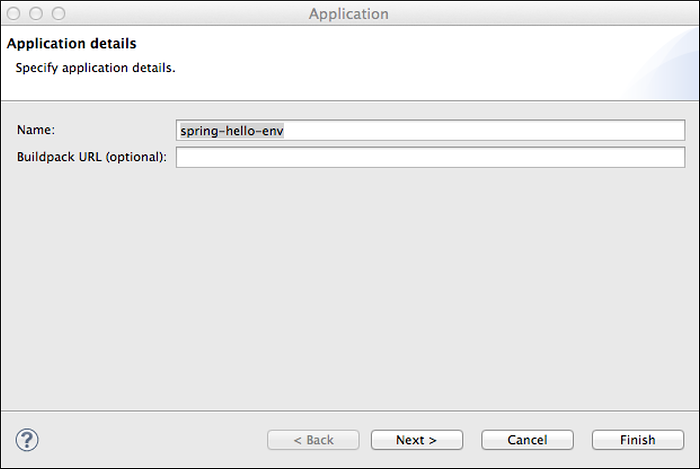
1. In the **Application Details** window:

By default, the **Name** field is populated with the application project name. You can enter a different name. The name is assigned to the deployed application, but does not rename the project.

1. If you want to use an external build pack to stage the application, enter the URL of the buildpack. You can deploy the application without further configuration by clicking **Finish**.

Note that because the application default values may take a second or two to load, the **Finish** button might not be enabled immediately. A progress indicator will indicate when the application default values have been loaded, and the **Finish** button will be enabled.

Click **Next** to continue.



In the **Launch Deployment** window:

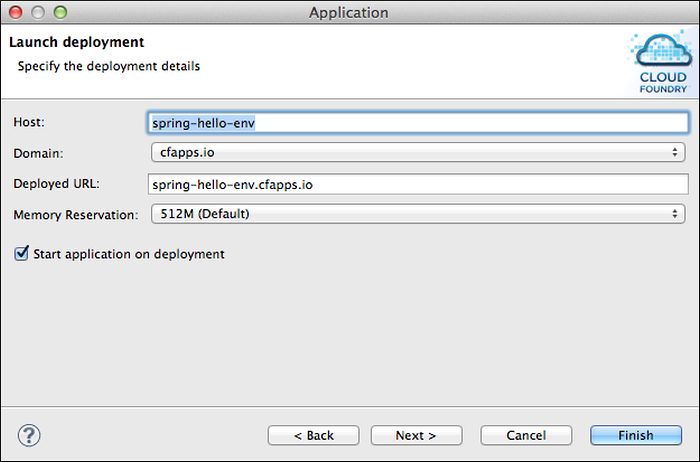
**Host** — By default, contains the name of the application. You can enter a different value if desired. If you push the same application to multiple spaces in the same organization, you must assign a unique **Host** to each.

**Domain** — Contains the default domain. If you have mapped custom domains to the target space, they appear in the pull-down list.

**Deployed URL** — By default, contains the value of the **Host** and **Domain** fields, separated by a period (.) character.

**Memory Reservation** — Select the amount of memory to allocate to the application from the pull-down list.

**Start application on deployment** — If you do not want the application to be started on deployment, uncheck the box.



The **Services Selection** window lists services provisioned in the target space. Checkmark the services, if any, that you want to bind to the application, and click **Finish**.

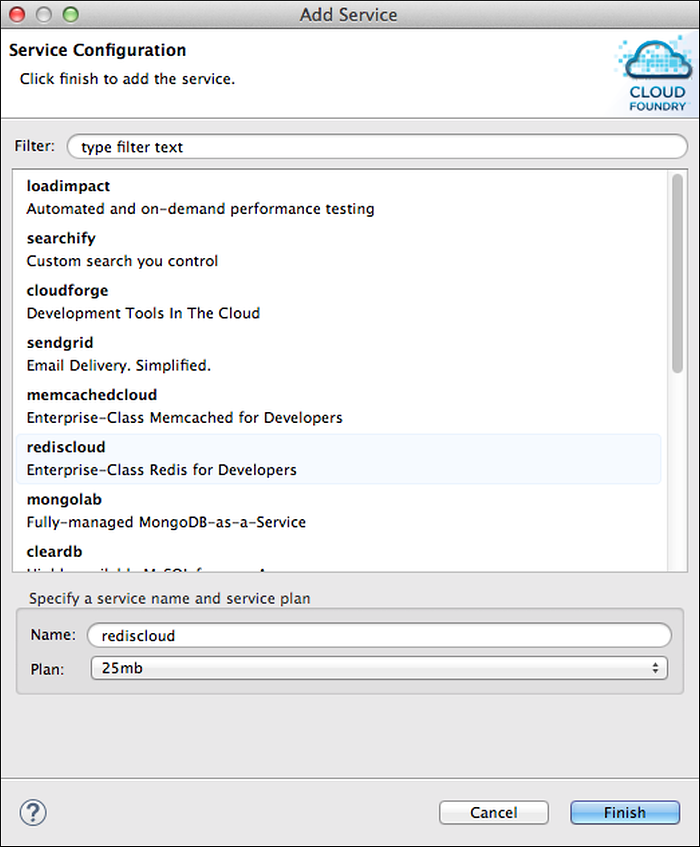
You can bind services to the application after deployment.

**PCF Operations**

1.Create a Service

Before you can bind a service to an application, you must create it. To create a service:

1. Select the **Applications and Services** tab.
2. Click the icon in the upper right corner of the **Services** pane.
3. In the **Service Configuration** window, enter a text pattern to **Filter** for a service. Matches are made against both service name and description.
4. Select a service from the **Service List**. The list automatically updates based on the filter text.
5. Enter a **Name** for the service and select a service **Plan** from the drop-down list.



## 2.Bind and Unbind Services

You can bind a service to an application when you deploy it. To bind a service to an application that is already deployed, drag the service from the **Services** pane to the **Application Services** pane. To unbind a service, right-click the service in the **Application Services** pane, and select **Unbind from Application**.

## 3.Push Application Changes

The Cloud Foundry editor supports these application operations:

* **Start** and **Stop** — When you **Start** an application, the plugin pushes all application files to the Cloud Foundry instance before starting the application, regardless of whether there are changes to the files or not.
* **Restart** — When you **Restart** a deployed application, the plugin does not push any resources to the Cloud Foundry instance.
* **Update and Restart** — When you run this command, the plugin pushes only the changes that were made to the application since last update, not the entire application. This is useful for performing incremental updates to large applications.

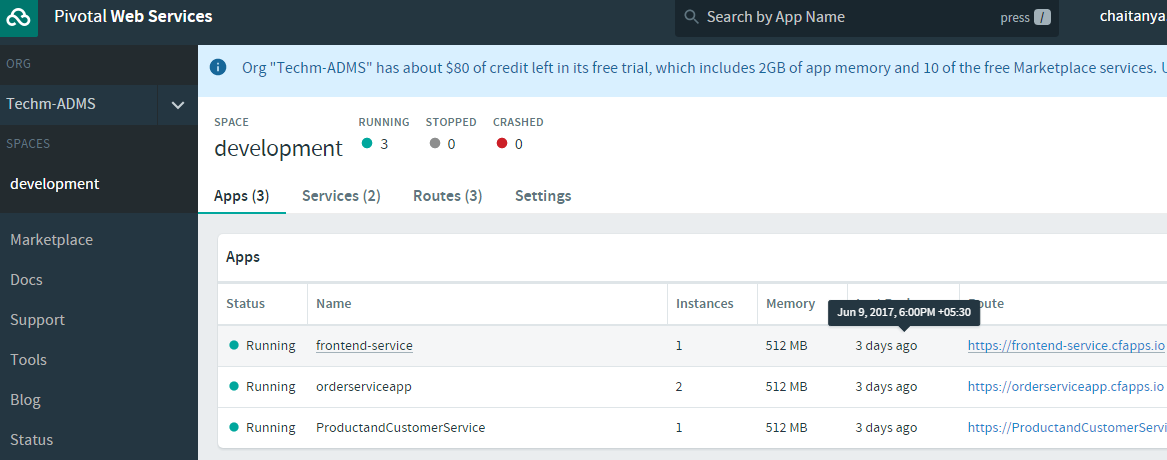
## 4.Undeploy an Application

To un deploy an application, right click the application in either the **Servers** or the **Applications** pane and click **Remove**.

**5.PCF Dash Board**

Once Application moved to cloud access pivotal dash board with below URL.

<https://api.pivotal.run.io>



This will list out all deployed application with current status.

Select services with console as well by clicking services tab

