

Deployit Online Demo Scenario

January, 2011

Contents

Introduction	1
After the Demo	1
Feedback	1
Overall	2
Deploy the PetClinic/1.0 package to the development environment	2
Import the PetClinic/1.1 package	4
Upgrade to PetClinic/1.1 on the development environment	5
Deploy PetClinic/1.1 to the test environment	6
Report on deployments	8
Troubleshoot Deployments	8
Next Steps	9

Introduction

Welcome to the Deployit Online Demo! This is where you get to play around with our product — import packages, deploy and upgrade an application, compare servers and see reports. Use our product in a safe, sandboxed environment to see how it works and how it fits in your environment. These instructions give you a “self-guided tour” through the product’s main functionality.

The demo contains test data that is designed to showcase Deployit’s functionality. Even though there is only a limited set of data in this demo, Deployit is designed to handle enterprise-scale data, comprised of hundreds of servers and applications.

After the Demo

Once you're finished with the demo, you will probably want to:

- request a full copy of Deployit ?
- request a demo for your team ?
- request a Proof of Concept implementation in your environment ?

You'll find these links again at the end of the demo.

Feedback

As always, we appreciate your feedback! Please use the *Feedback* tab on the right-hand side of these instructions.

Overall

We are going to walk through a demo scenario that contains some fairly typical actions for our demo application, *PetClinic*. The scenario consists of the following steps:

1. Deploy the *PetClinic* application version *1.0* (which is already in the system) to the *Development* environment.
2. Import the new *1.1* version of the *PetClinic* application.
3. Upgrade the *PetClinic* application on the *Development* environment to the *1.1* version.
4. Deploy the *1.1* version on the *Test* environment.
5. View a deployment report.
6. Troubleshoot our deployment on the *Test* environment.

Now, let's start!

Deploy the PetClinic/1.0 package to the development environment

This is Deployit's main functionality — performing a deployment. The demo environment comes pre-populated with some applications and environments for you to play around with. Don't worry, you can't mess up anything. If you end up in a bad situation, simply reload the page in your browser to reset the demo.

The first thing we're going to do is to deploy the *PetClinic* application version *1.0* (a **package** in Deployit speak) to the WebSphere Development environment. During the deployment, Deployit takes the members of your package and turns them into *deployed items* on the target environment and infrastructure. When configuring the deployment, you will need to tell Deployit where and how to create the deployed items. Luckily, Deployit can also generate default deployed items based on the type of the package member. This is what we'll do in this deployment.

Follow these steps:

1. The application starts with an open Deployment Tab. If you have closed the Deployment Tab, open a new one by clicking on the New Deployment button:

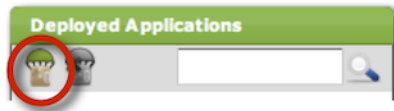


Figure 1: New Deployment Button

2. In the **Package Browser**, locate the package containing the *PetClinic* application, version *1.0*.
3. Drag the package and drop it into the Deployment Tab:

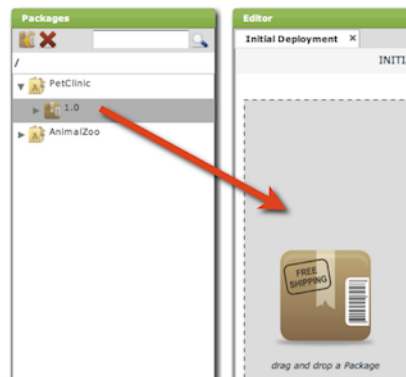


Figure 2: Drag and Drop a Package

The Deployment Tab will show the contents of the package in the Package Box.

4. In the **Deployed Application Browser**, locate the *WebSphere Development* environment.
5. Drag the environment and drop it into the Deployment Tab:

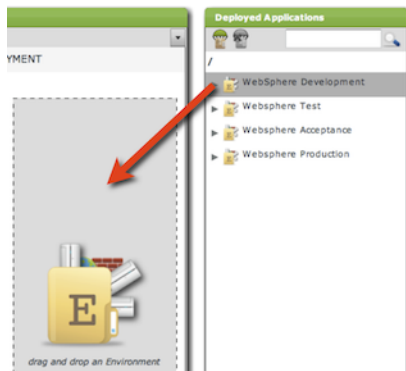


Figure 3: Drag and Drop an Environment

The Deployment Tab will show the contents of the environment in the Environment Box.

6. Let Deployit determine the default deployed items for your deployment by clicking on the Generate Default Deployed Items button:



Figure 4: Generate Deployed Items Button

7. Now that the configuration is finished, generate the deployment steplist by clicking on the **Next** button.
8. The steplist reflects all steps Deployit will take to execute your deployment. Click the **Deploy** button to start the deployment.
9. Once the deployment is done, close the tab using the **Close** button. You will see your new deployed application in the **Deployed Application Browser**:

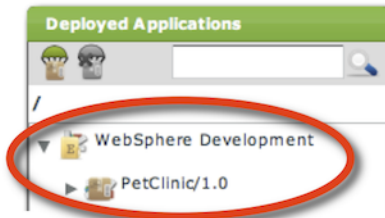


Figure 5: Deployed PetClinic/1.0 Application

Import the PetClinic/1.1 package

Now let's see how we can get a new package into Deployit. Let's say your development team has just finished version 1.1 of the PetClinic application and you want to import it into Deployit. The packages Deployit understands are archives (JAR archives to be precise) containing the application code (WAR or EAR file) and all of the configuration it needs to run (a datasource, topics or queues, etc.). This package can be deployed and promoted throughout the DTAP environments.

Follow these steps:

1. Start the **Import Wizard** by clicking on the **Import Package** button:
2. Select **Import deployment package from server** as the source of your import.
3. From the dropdown, select the *PetClinic/1.1* package to be imported:
4. Click on the **Import** button to start the import process.
5. Click on the **Close** button to finish the import. You will see the *1.1* version of the *PetClinic* application in the **Package Browser**.

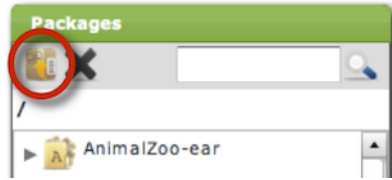


Figure 6: Import Package Button

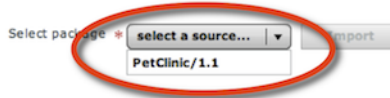


Figure 7: Select Package to Import

Upgrade to PetClinic/1.1 on the development environment

One of Deployit's strengths is dealing with application *upgrades*, that is, replacing an already installed version of an application on an environment with a later version. Deployit makes it as easy as possible to reuse configuration done for the initial deployment so that an upgrade can often be performed with only a few mouse clicks. This makes it possible for developers to do their own deployments to the test environment without requiring assistance from the Operations department. Even if a newer version of your application has some structural changes (includes other EAR, WAR archives or removes a particular datasource or queue), Deployit will still reuse as much configuration as possible. Let's see how this works.

Follow these steps:

1. Start a new deployment by clicking on the **New Deployment** button:

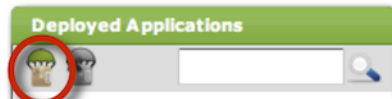


Figure 8: New Deployment Button

2. In the **Deployed Application Browser**, locate the *PetClinic/1.1* deployed application you created in the first step.
3. Drag the deployed application and drop it into the Deployment Tab. This indicates to Deployit that you want to upgrade this deployment:
4. In the **Package Browser**, locate the package containing the *PetClinic* application, version *1.1*.
5. Drag the package and drop it into the Deployment Tab. The Deployment Tab will show the contents of the package in the Package Box:
6. Deployit has already reused all deployed items from the previous deployment as you can see. There is no need to configure anything.

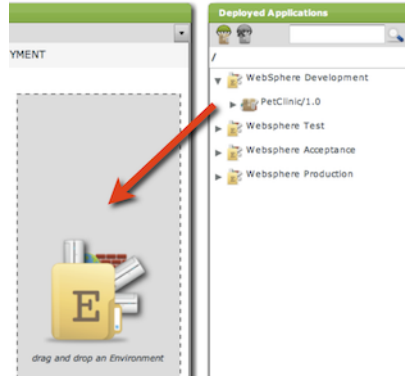


Figure 9: Drag and Drop a Deployment

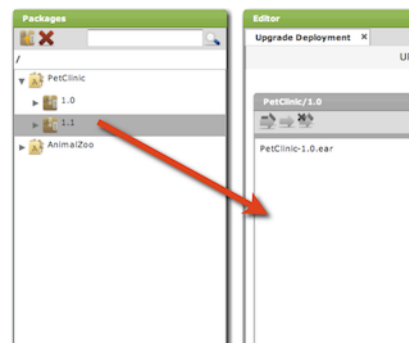


Figure 10: Drag and Drop the New Package

From this point onwards, the deployment follows a familiar course.

7. Generate the deployment steplist by clicking on the **Next** button.
8. The steplist reflects all steps Deployit will take to execute your deployment. Click the **Deploy** button to start the deployment.
9. Once the deployment is done, close the tab using the **Close** button. You will see the upgraded *PetClinic* application in the **Deployed Application Browser** under the *WebSphere Development* environment.

Deploy PetClinic/1.1 to the test environment

We've seen how easy existing deployed applications can be upgraded using a newer package. This package contains an *environment independent* application. This means that the same package can be used unchanged to deploy to another environment, even if it is completely different. Let's see how this works by deploying our *PetClinic/1.1* application to the *WebSphere Test Environment*.

Follow these steps:

1. Open a new deployment tab by clicking on the **New Deployment** button:
2. In the **Package Browser**, locate the package containing the *PetClinic* application, version *1.1*.

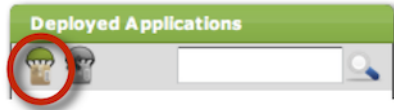


Figure 11: New Deployment Button

3. Drag the package and drop it into the Deployment Tab.
4. In the **Deployed Application Browser**, locate the *WebSphere Test* environment.
5. Drag the environment and drop it into the Deployment Tab.

Notice that this environment contains **two** servers where the previous environment contained only one server. Environments can contain any number of servers (application, database or otherwise), clusters, nodes, cells, all depending on your configuration and the middleware used.

6. Let Deployit determine the defaults mapping for your deployment by clicking on the Generate Default Deployed Items button:

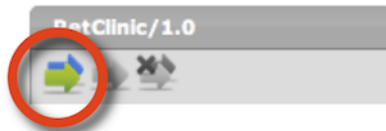


Figure 12: Generate Deployed Items Button

As you can see, Deployit generates deployed items for all of the targets in the environment.

7. Suppose you want to configure the deployed item slightly differently for each server. For instance, let's say you want to give both deployments a different *virtual host* setting. Double-click on the deployed item under the first server:

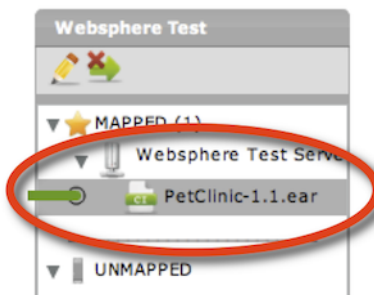


Figure 13: Configure Deployed Item

The Deployed Item Configuration balloon opens up. Find the field marked *virtual host* and enter a value. Click **Save** to save the changes.

8. Now double-click on the deployed item under the second server and enter a different virtual host. Again click **Save** to save the changes and close the balloon.
9. Now that the configuration is finished, generate the deployment steplist by clicking on the **Next** button.
10. The steplist reflects all steps Deployit will take to execute your deployment. As you can see, Deployit added steps to deploy the application to both servers in the environment. Click the **Deploy** button to start the deployment.
11. Once the deployment is done, close the tab using the **Close** button. You will see your new deployed application in the **Deployed Application Browser**.

As you can see, deploying the same package to another environment follows exactly the same procedure.

Report on deployments

Deployit contains a wealth of information about your application, environments and deployments. Some of this information can be used to report on the state of your environments or the number of your deployments. Let's say you want to know which versions of which applications are running on your test environment.

Follow these steps:

1. Switch to the **Reports** screen. The left hand list contains all available report types, the report data will be shown in the right-hand side.
2. Select the report type **Deployed applications per environment**. A tab opens in the editor window, allowing you to enter report parameters.
3. Select the *WebSphere Test* environment:

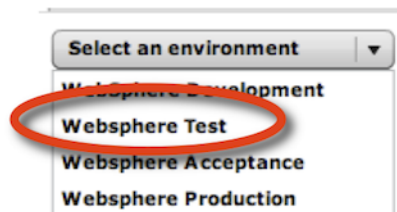


Figure 14: Select a Report Environment

An overview of all applications and versions on the test environment is immediately shown. This report shows you who deployed what, when on the specified environment.

Note: in this demo, the data in the report is fictitious.

Troubleshoot Deployments

Deployit can also help you to troubleshoot problems with your deployments. Suppose the PetClinic application that you've deployed on your test environment displays some flaky behavior. It seems to be working fine on one of the servers, but not on the other. You suspect it may have something to do with the server settings. How can you find out what the problem is? Deployit allows you to compare *servers* (or any other CIs) with each other and easily spot the differences. Let's see how this works.

Follow these steps:

1. If you're not there already, open the **Repository** screen.
2. Locate the servers in the *WebSphere Test* environment under the **Infrastructure** node in the repository (they are called *WebSphere Test Server 1* and *WebSphere Test Server 2*).
3. Select the first of the two servers by clicking on it:

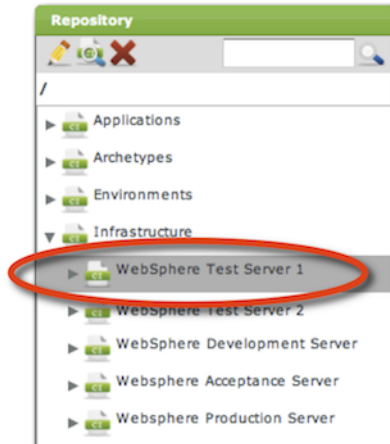


Figure 15: Select Server

4. Open the Comparison Tab by clicking on the **Compare** button:



Figure 16: Compare CI Button

This shows you the server in a tabular format with all of it's properties listed.

5. Now drag the second server into the comparison:

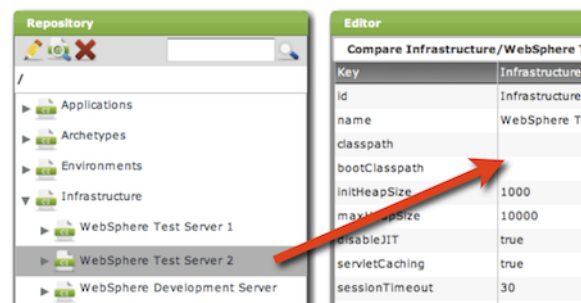


Figure 17: Drag Server to Compare

Deployit shows the differences between the two CIs in red. Looks like the servers have two different memory settings. That may be causing your problem. Now it should be easy to fix.

Next Steps

That's the end of our guided tour. We hope you have an idea of what the product can do and how you can use it. Please feel free to explore the demo further if you wish.

Now that you are finished with the demo, you will probably want to:

- request a full copy of Deployit ?
- request a demo for your team ?
- request a Proof of Concept implementation in your environment ?