Run IBM MobileFirst Platform Foundation on IBM Containers

Overview

This tutorial demonstrates how to take a locally developed IBM MobileFirst Platform Foundation project and run it on Bluemix. To achieve this result, you go through the following steps: set up your host computer with the required tools (Cloud Foundry CLI, Docker, and IBM Containers Extension (cf ic) Plug-in), set up your Bluemix environment, build a MobileFirst Platform Foundation Server image, deploy your project runtime and push it to the Bluemix repository. Finally, you run the image on IBM Containers as a single Container or a Container group and update it with the MobileFirst project application and adapter.

Note: Windows OS is currently not supported.
Note: The MobileFirst Server Configuration Tools cannot be used for deployments to IBM Containers.

Prerequisite: Make sure to read the Introduction to IBM MobileFirst Platform Foundation on IBM Containers (../) tutorial.

Topics

- Register an account at Bluemix
- Set up your host machine
- Run IBM MobileFirst Platform Foundation on IBM Containers

Register an account at Bluemix

If you do not yet have an account, visit the Bluemix website (http://www.bluemix.net) and click **Get Started Free** or **Sign Up**. You'll need to fill up a registration form before you can move on to the next step.

The Bluemix Dashboard

After signing in to Bluemix, you are presented with the Bluemix Dashboard, which provides an overview of the active Bluemix **space**. By default, this work area receives the name "dev". You can create multiple work areas/spaces if needed.

Set up your host machine

To manage containers and images, you need to install the following tools: Docker, Cloud Foundry CLI and IBM Containers (cf ic) plugin.

Docker

Go to the Docker Documentation (https://docs.docker.com/) > on the left menu, select **Install > Docker Engine**, select your OS type and follow the instructions to install the Docker Toolbox.
Note: IBM does not support Docker's Kitematic software.

In OS X there are two options to run Docker commands:

- From the OS X Terminal
- From the Docker Quickstart Terminal

If you choose to work from the Docker Quickstart Terminal no further setup is needed. You must work only from it.
If you choose to work from the OS X Terminal, do the following:

- Run the command: `docker-machine env default`
- Set the result as environment variables, for example:

```
$ docker-machine env default
export DOCKER_TLS_VERIFY="1"
export DOCKER_HOST="tcp://192.168.99.101:2376"
export DOCKER_CERT_PATH="/Users/mary/.docker/machine/machines/default"
export DOCKER_MACHINE_NAME="default"
```

> For further information consult the Docker documentation.

Cloud Foundry Plugin and IBM Containers plugin

1. Install the Cloud Foundry CLI (https://github.com/cloudfoundry/cli/releases?cm_mc_uid=85906649576514533887001&cm_mc_sid_50200000=1454307195).
2. Install the IBM Containers Plugin (cf ic) (https://console.ng.bluemix.net/docs/containers/container_cli_cfic_install.html).

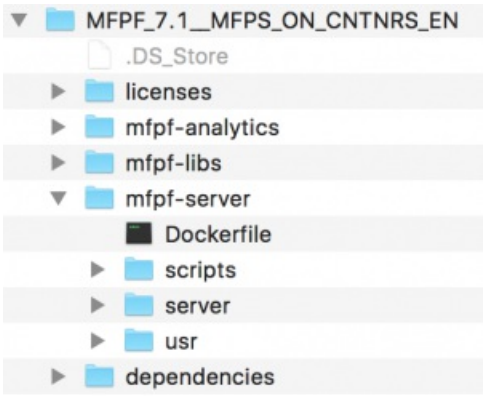
Run IBM MobileFirst Platform Foundation on IBM Containers

To run IBM MobileFirst Platform Foundation on IBM Containers, you must first create an image that will later be pushed to Bluemix.

If you have not downloaded the IBM MobileFirst Platform Foundation V7.1 .zip file yet, click the button below, follow the instructions and download it (search for "CN8BPEN").

Follow the instructions and download the IBM MobileFirst Platform Foundation v7.1 customization .zip (http://www-01.ibm.com/support/docview.wss?uid=swg2C7000005)

Structure of the ibm-mfpf-container-7.1.0.0-eval.zip archive



The extracted ZIP file contains the files for building an image (`dependencies` and `mfpf-libs`), the files for building and deploying an IBM MobileFirst Platform Foundation Operational Analytics Container (`mfpf-analytics`), and files for configuring an IBM MobileFirst Platform Server Container (`mfpf-server`). This tutorial does not cover the analytics part.

The mfpf-server folder

- `Dockerfile`: text document that contains all the commands in order to build an image.
- `usr` folder:
  - `bin` folder: Contains the script file that gets executed on the start of the container. You can add your own custom code to be executed.
  - `config` folder: Contains server key store configuration, User registry configuration, MobileFirst Platform Foundation Server properties (includes runtime configuration – analytics, attribute store etc).
  - `env` folder: Contains server environment configuration (ports, application root names etc).
  - `jre-security` folder: You can update the JRE security related files (truststore, policy jars etc) by placing them in this folder.
  - `projects` folder: The location of your MobileFirst Platform project runtime (`.war` file).
  - `security` folder: The key store, trust store and the LTPA keys files (ltpa.keys) should be placed here.
  - `ssh` folder: Contains the id_rsa.pub file - the ssh public key file to enable ssh on the container.
  - `wxs` folder: Contains the data cache / extreme scale client library when Data Cache is used as attribute store for the server.

  > This tutorial refers only to the `projects` folder.
- `server` folder: Contains elements that are required for the IBM MobileFirst Platform Foundation Operational Server deployment.
- `scripts` folder: This folder contains the `args` folder, which contains a set of configuration files. It also contains scripts to run for logging in to Bluemix, building a Mobilefirst Platform Foundation Server image, deploying your project runtime, and for pushing and running the image on Bluemix. You can choose to run the scripts interactively or by pre-configuring the configuration files as will be further explained.

Step 1: Create an IBM MobileFirst Platform Foundation project

Create a new MobileFirst project or use an existing one. You can find tutorials on how to create a new project, and their associated sample projects, in the Getting Started with Foundation (../../) page.

Step 2: Prerequisites

1. Login to the IBM Bluemix environment.

   This step is mandatory because you will be running IBM Containers commands during the following steps.

   Run:

   ```
   cf login
   ```

When prompted, enter the following information:

- Bluemix API endpoint
- Email
- Password
- Organization, if you have more than one
- Space, if you have more than one

2. To run IBM Containers commands, you must first log in into the IBM Container Cloud Service.
   Run:

   ```
   cf ic login
   ```

3. Make sure that the `namespace` for container registry is set. The `namespace` is a unique name to identify your private repository on the Bluemix registry. The namespace is assigned once for an organization and cannot be changed.
   Choose a namespace according to following rules:

   - It can contain only lowercase letters, numbers, or underscores (_).
   - It can be 4 - 30 characters. If you plan to manage containers from the command line, you might prefer to have a short namespace that can be typed quickly.
   - It must be unique in the Bluemix registry.

   To set a namespace, run the command:

   ```
   cf ic namespace set <new_name>
   ```

   To get the namespace that you have set, run the command:

   ```
   cf ic namespace get
   ```

> To learn more about IC commands, use the `ic help` command.

Step 3: Using the configuration files

Note: If you choose to run the scripts interactively, you can skip the configuration but it is strongly suggested to at least read and understand the arguments you will need to provide.
The `args` folder contains a set of configuration files which contain the arguments that are required to run the scripts. Fill in the arguments' values in the following files:

initenv.properties
prepareserverdbs.properties
prepareserver.properties
startserver.properties

Step 4: Running the scripts

As explained above you can choose to run the scripts interactively or by using the configuration files:

- Using the configuration files - run the scripts and pass the respective configuration file as an argument
- Interactively - run the scripts without any arguments

The following demonstrate the first option.

1. **initenv.sh – Logging in to Bluemix**
   Run the `initenv.sh` script in order to create an environment for building and running IBM MobileFirst Platform Foundation on the IBM Containers:

   ```
   ./initenv.sh args/initenv.properties
   ```

2. **prepareserverdbs.sh - Prepare the MobileFirst Server database**
   The `prepareserverdbs.sh` script is used to configure your MobileFirst project database. You will need to run it separately, once for the admin database and once for every MobileFirst project runtime database.

   - For the admin database make sure to comment out the `RUNTIME_NAME` argument and run:

     ```
     ./prepareserverdbs.sh args/prepareserverdbs.properties
     ```

   - For each MobileFirst project runtime database - first uncomment the project `RUNTIME_NAME` argument, change it value to match the specific project war file and run:

```
./prepareserverdbs.sh args/prepareserverdbs.properties
```

3. **initenv.sh (Optional) – Logging in to Bluemix**
   This step is required only if you need to create your containers in a different Organization and Space than where the dashDB service instance is available. If yes, then update the initenv.properties with the new Organization and Space where the containers have to be created (and started), and rerun the `initenv.sh` script:

   ```
   ./initenv.sh args/initenv.properties
   ```

4. **prepareserver.sh - Prepare a Mobilefirst Platform Foundation Server image**
   Uncomment the `PROJECT_LOC` argument and run the `prepareserver.sh` script in order to build a MobileFirst Platform Foundation Server image, deploy your project runtime and push it to to your Bluemix repository:

   ```
   ./prepareserver.sh args/prepareserver.properties
   ```

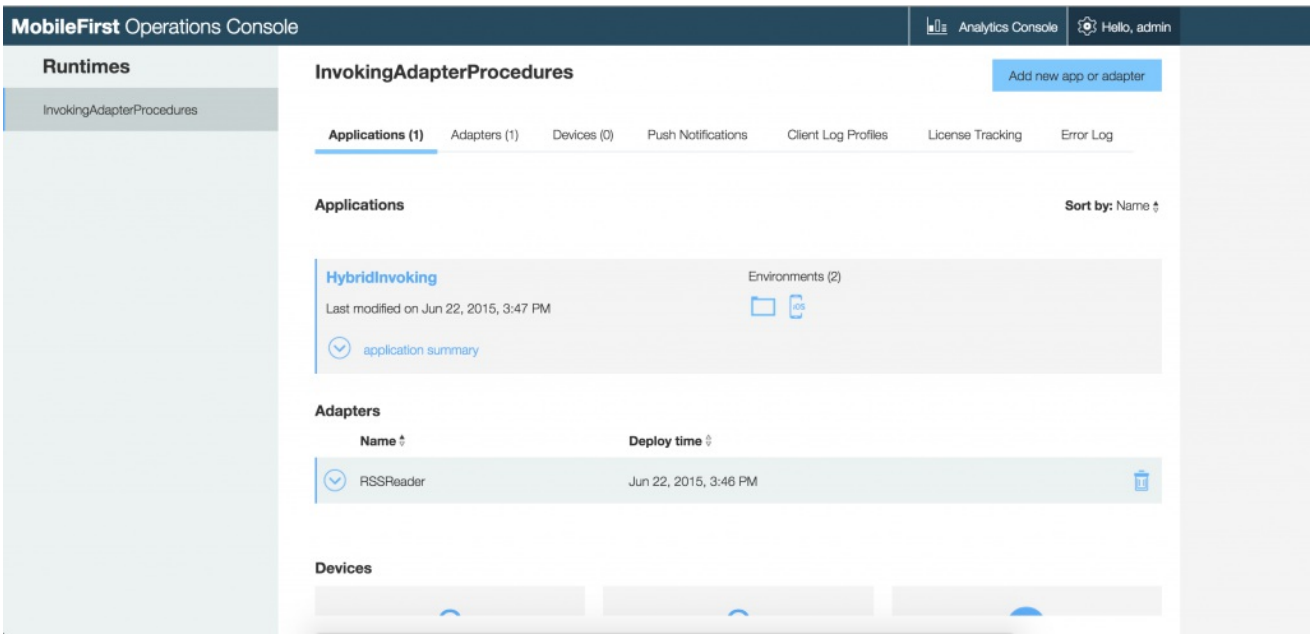
   To view all available images in your Bluemix repository run:

   ```
   cf ic images
   ```

   The list contains the image name, date of creation and ID.
5. **startserver.sh - Running the image on an IBM Container**
   The `startserver.sh` script is used to run the Mobilefirst Server image on an IBM Container. It also Binds your image to the public IP you configured in the `SERVER_IP` property.
   - Run:

     ```
     ./startserver.sh args/startserver.properties
     ```

   - Launch the MobileFirst Console by loading the following URL: http://<server_ip>:9080/worklightconsole (it may take a few moments).
   - Upload the `.wlapp` and `.adapter` files.
   - Update the application's `worklight.plist` (for iOS) and/or `wlclient.properties` (for Android, Windows Universal, Windows Phone) with the protocol, host and port values of the IBM Container.
   - You can now run your application to verify that it successfully connects to the MobileFirst Server, running on IBM Containers.



Last modified on

Privacy
(http://www.ibm.com/privacy/us/en/)
Terms of use
(file:////home/travis/build/MFPSamples/DevCenter/site/terms-of-use/)
Third party notice
(file:////home/travis/build/MFPSamples/DevCenter/site/third-party-notice/)

(https://twitter.com/ibmmobiledev)
YouTube
(https://www.youtube.com/channel/UC1-C7tAjtaKzn-ci2Qusu97Q)
GitHub
(https://github.com/MobileFirst-Platform-Developer-Center)

(https://github.com/MobileFirst-Platform-Developer-Center/DevCenter/issues/new)
Contribute
(https://github.com/MobileFirst-Platform-Developer-Center/DevCenter/blob/master/contributing.m
Report abuse
(https://www.ibm.com/developerworks/commu