

Using the MobileFirst Platform Operations Console

Overview

The MobileFirst Platform Operations Console is a web-based UI which enables simplified work flows for both the developer and the administrator to create, monitor, secure and administer applications & adapters.

Jump to:

- Accessing the console
- Navigating the console

Accessing the console

The MobileFirst Operations Console can be accessed in the following ways:

From a locally installed MobileFirst Server

Desktop Browser

From your browser of choice, load the URL `http://localhost:9080/mfpconsole` (`http://localhost:9080/mfpconsole`). The username/password are *admin/admin*.

Command-line

From a **Command-line** window, with the MobileFirst CLI installed, run the command: `mfpdev server console`

MobileFirst Studio

In Eclipse, click on the **Open MobileFirst Console** button.



From a remotely installed MobileFirst Server

Desktop Browser

From your browser of choice, load the URL `http://the-server-host:server-port-number/mfpconsole`

The host server can be a customer-owner server, or running on a service such as Bluemix. The username/password are *admin/admin*.

Command-line

From a **Command-line** window, with the MobileFirst CLI installed,

1. Add a remote server definition:

Interactive Mode

Run the command: `mfpdev server add` and follow the on-screen instructions.

Direct Mode

Run the command with the following structure: `mfpdev server add [server-name] --URL`

```
[remote-server-URL] --login [admin-username] --password [admin-password] --contextroot [admin-service-name]
```

. For example:

```
mfpdev server add MyRemoteServer http://my-remote-host:9080/ --login TheAdmin --password ThePassword --contextroot mfpadmin
```

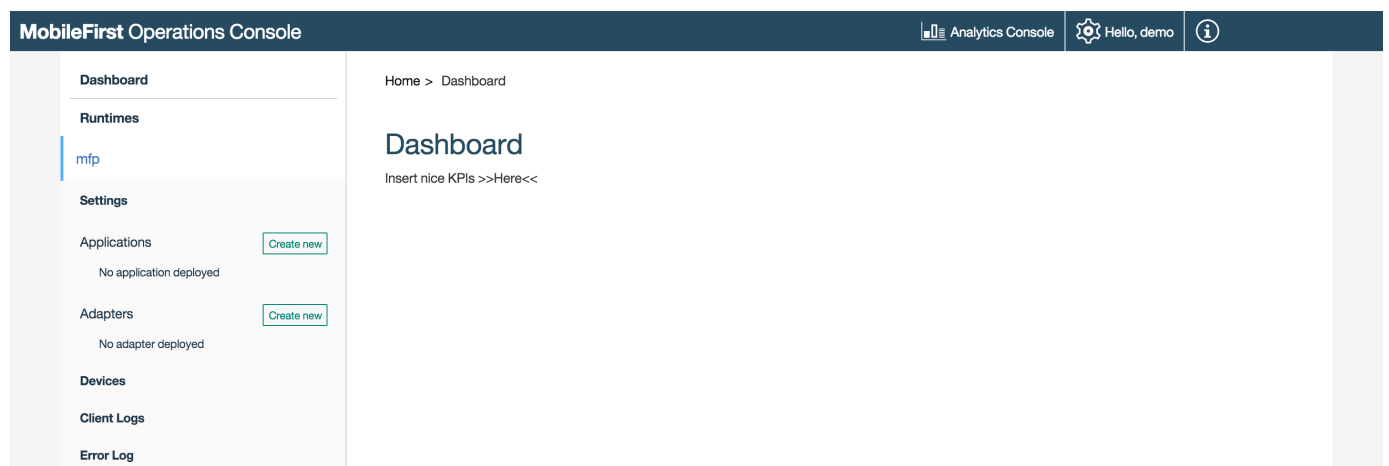
2. Run the command: `mfpdev server console MyRemoteServer`

Learn more about the various CLI commands in the [Using CLI to manage MobileFirst artifacts \(../../client-side-development/using-cli-to-manage-mobilefirst-artifacts/\)](#) tutorial.

Navigating the console

Dashboard

The Dashboard provides a glance view of the deployed projects.



Runtime settings

Edit runtime properties such as Analytics server URL, global security variables, server keystore and confidential clients.

MobileFirst Operations Console

Analytics ConsoleHello, demo

Dashboard

Runtime

mfp

Settings

Applications

Create new

Adapters

Create new

Devices

Client Logs

Error Log

Home > mfp > Settings

Settings

Runtime PropertiesKeystoreConfidential Clients

Runtime Properties

Analytics

enabled *false

username *

consoleURL *

serverURL *

logBufferSize *500

Default Value: 500

Edit

Applications

Creating applications

Provide basic application values and download Starter Code.

MobileFirst Operations Console

Analytics ConsoleHello, demo

Dashboard

Runtime

mfp

Settings

Applications

Create new

Adapters

Create new

Devices

Client Logs

Error Log

Home > mfp > Register an Application

Register an Application

Choose Platform

Android

IOS

Windows

Windows Phone

Bundle ID *

Application Identifier

Version *

Application Version

Register application

Managing applications

Manage and configure registered applications by use of Direct Update (../client-side-development/direct-update/), Remote Disable, Application Authenticity (../authentication-and-security/application-authenticity/), and setting security parameters (../authentication-and-security/authentication-concepts/).

MobileFirst Operations Console

Analytics ConsoleHello, demo

< Back

mfp

Applications

Create new

MyBankApp

Platform

^ iOS1

1.0

Push

Home > mfp > MyBankApp > iOS 1.0

Delete version

MyBankApp iOS v 1.0

ManagementAuthenticitySecurityConfiguration Files

Last modified: Jan 10, 2016, 8:03 AM

Application Access

Status: *

Active

Active and Notifying

Access Disabled

Direct Update

Update a Cordova cross-platform application by uploading a new web resource file.

No Web Resources deployedUpload File

Authentication and Security

Configure application security paramteres.

MobileFirst Operations Console

Analytics ConsoleHello, demo

< Back

mfp

Applications

Create new

MyBankApp

Platform

^ iOS1

1.0

Push

Home > mfp > MyBankApp > iOS 1.0

Delete version

MyBankApp iOS v 1.0

ManagementAuthenticitySecurityConfiguration Files

Configurations

Maximum token expiration (seconds) *

3600

Edit

Ipsum lorem

Map Scope Elements to Security Checks

Configure one or more authentications required for the client to get proper permissions for accessing a protected resource.

Create New

You didn't map scope elements to security checks yet

Get started by clicking "Create New"

Application Mandatory Scope

Configure one or more authentications required in order to get proper permissions for running the application. This can include out-of-the-box security checks or scope elements mapped to security checks.

Create New

Notifications

Set-up push notifications (../notifications/push-notifications-overview/) and related parameters, such as tags, as well as sending notifications.

The screenshot shows the MobileFirst Operations Console interface. The left sidebar contains a navigation menu with 'Back', 'mfp', 'Applications' (with a 'Create new' button), 'MyBankApp', 'Platform' (with a dropdown for 'iOS' and a '1.0' version indicator), and a 'Push' button. The main content area is titled 'Push' and shows the breadcrumb 'Home > mfp > MyBankApp > Push'. Below this, there are tabs for 'Send Push', 'Tags', and 'Push Settings' (which is active). The 'Push Settings' section is titled 'Push Notification Settings' and includes a link to the 'Push Notifications Guide'. It has two sub-tabs: 'Apple Production' (active) and 'Apple Sandbox'. Under 'Apple Production', there is a section for 'Apple Push Certificates for production' with a link to the 'Apple Push Certificates Guide'. Below this is an 'Upload .p12 file' section with a file upload button and a 'Password' field with a 'Save' button. At the bottom, there is a section for 'GCM Push Credentials' with a link to the 'Google Cloud Messaging Documentation'.

Adapters

Creating adapters

Register an adapter and download Starter Code.

The screenshot shows the MobileFirst Operations Console interface. The left sidebar contains a navigation menu with 'Dashboard', 'Runtimes', 'mfp', 'Settings', 'Applications' (with a 'Create new' button), 'Adapters' (with a 'Create new' button), 'Devices', 'Client Logs', and 'Error Log'. The main content area is titled 'Create a new Adapter' and shows the breadcrumb 'Home > mfp > Create a new Adapter'. Below this, there is a message: 'It seems like you don't have any adapters, lets get started' with a 'Deploy Adapter' button. A section titled 'Follow these steps to set up an adapter' (with a 'Hide guide' link) lists five steps: 1. Setting up your environment (including 'Installing the command line interface (CLI)' and 'Installing Maven'), 2. Start with a sample adapter, 3. In your IDE of choice, edit the adapter code - REST end points and adapter descriptor, 4. Build and package, and 5. Upload adapter.

Devices

Administrators can search for devices that access the MobileFirst Server and can manage access rights. Devices can be searched for using either user ID or using a friendly name.

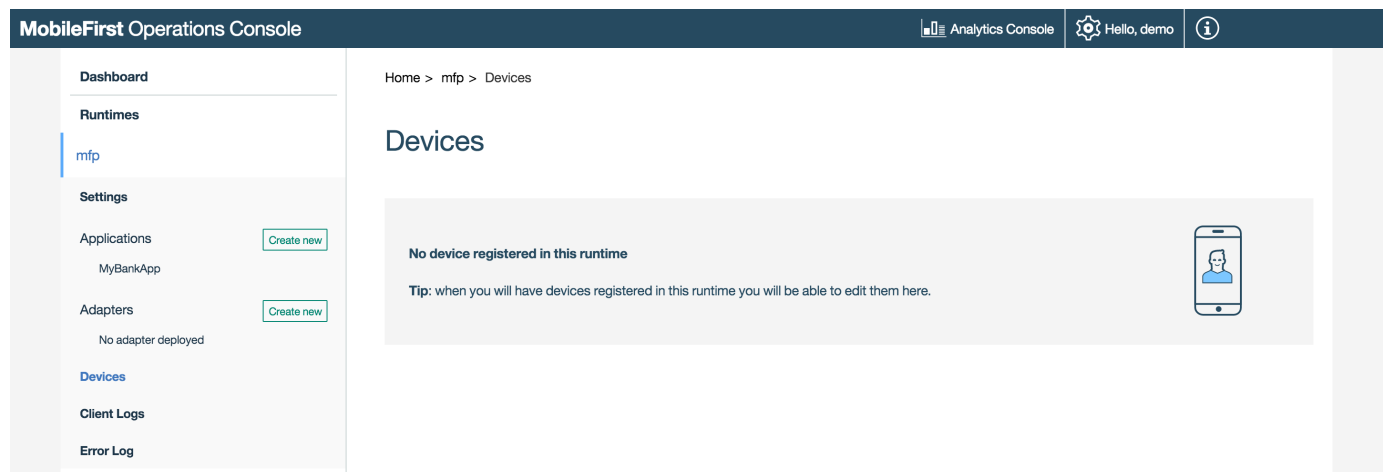
The user ID is the identifier that was used to log-in.

A friendly name is a name that is associated with the device to distinguish it from other devices that share the user ID. You can set the friendly name on the client by using the client-side JavaScript APIs:

`WL.Device.setFriendlyName` and `WL.Device.getFriendlyName`.

For more information, see the topic about device access management in the MobileFirst Operations Console in the user documentation.

replace with image showing logged devices



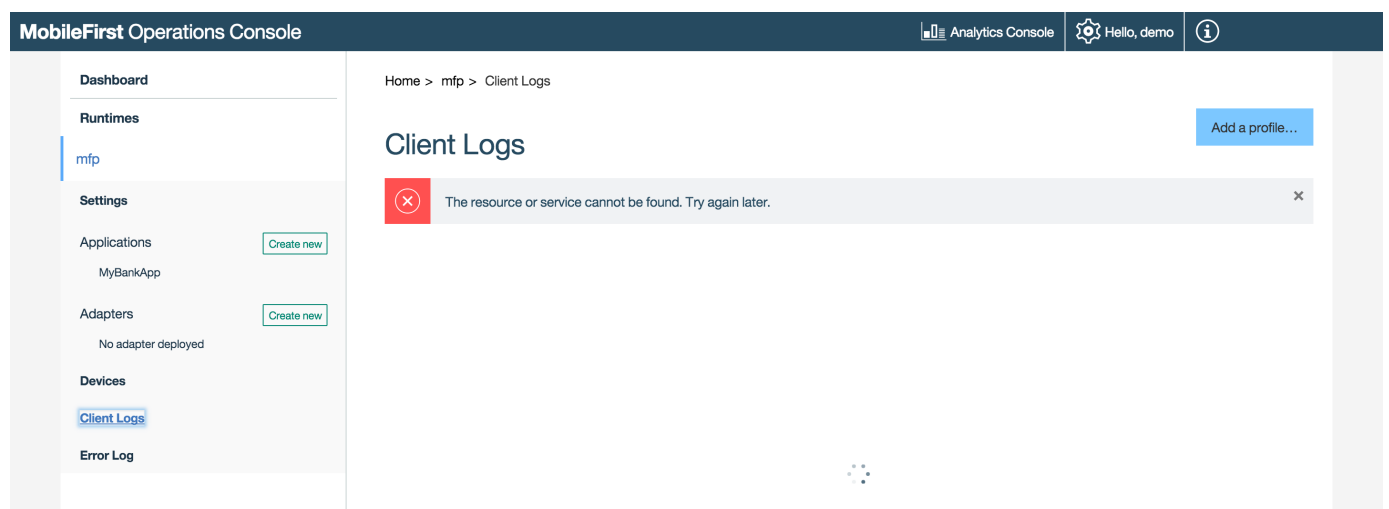
Client logs

Administrators can use log profiles to adjust client logger configurations, such as log level and log package filters, for any combination of operating system, operating system version, application, application version, and device model.

When an administrator creates a configuration profile, the log configuration is concatenated with responses to explicit WLCClient `connect` and `invokeProcedure/WLResourceRequest` API calls, and is applied automatically.

For more information, see the topic about client-side log capture configuration from MobileFirst Operations Console in the user documentation.

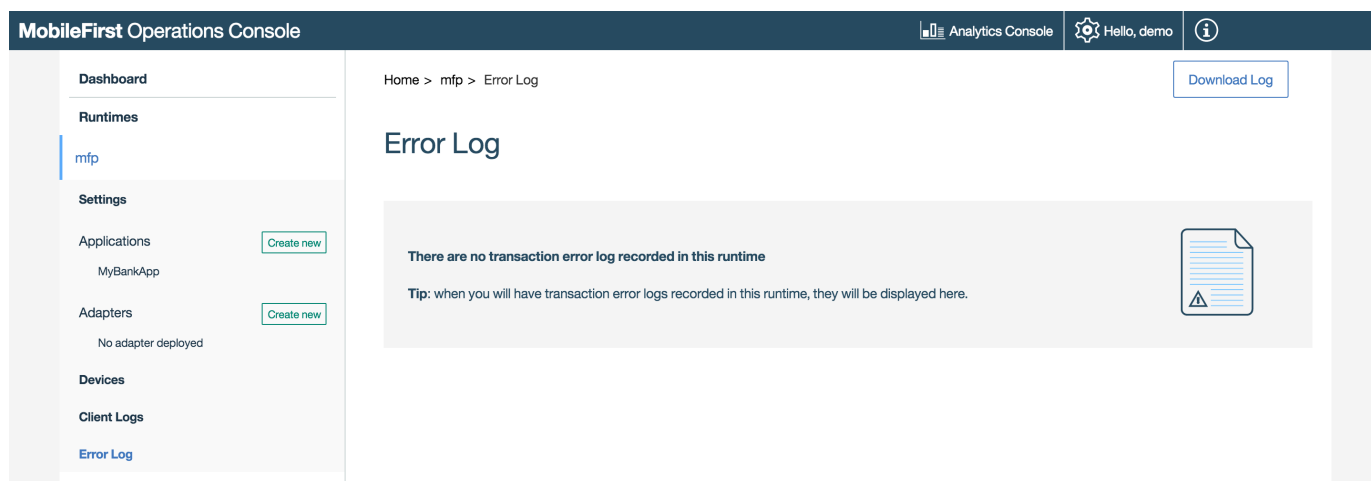
replace with image showing client logs



Error log

The Error log shows a list of the failed management operations that were initiated from the MobileFirst Operations Console, or from the command line, on the current runtime environment. Use the log to see the effect of the failure on the servers.

For more information, see the topic about error log of operations on runtime environments in the user documentation.



License tracking

Accessible from the top Settings buttons.

License terms vary depending on which edition (Enterprise or Consumer) of MobileFirst Platform Foundation is being used. License tracking is enabled by default and tracks metrics relevant to the licensing policy, such as active client devices and installed applications. This information helps determine whether the current usage of MobileFirst Platform is within the license entitlement levels and can prevent potential license violations.

By tracking the usage of client devices and determining whether the devices are active, administrators can decommission devices that should no longer be accessing the service. This situation might arise if an employee has left the company, for example.

For more information, see the topic about license tracking in the user documentation.

replace with image showing license information

