

Using the MobileFirst 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`.

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 either a customer-owned server, or the IBM Bluemix service, IBM Mobile Foundation (`../ibm-containers/`).

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
```

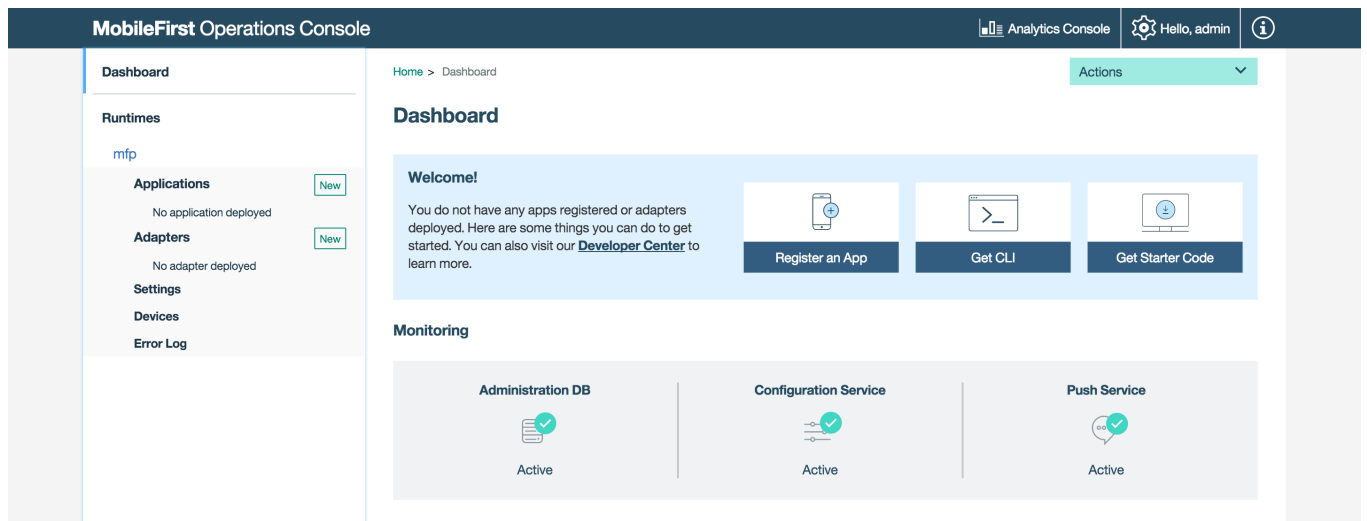
2. Run the command: `mfpdev server console MyRemoteServer`.

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

Navigating the console

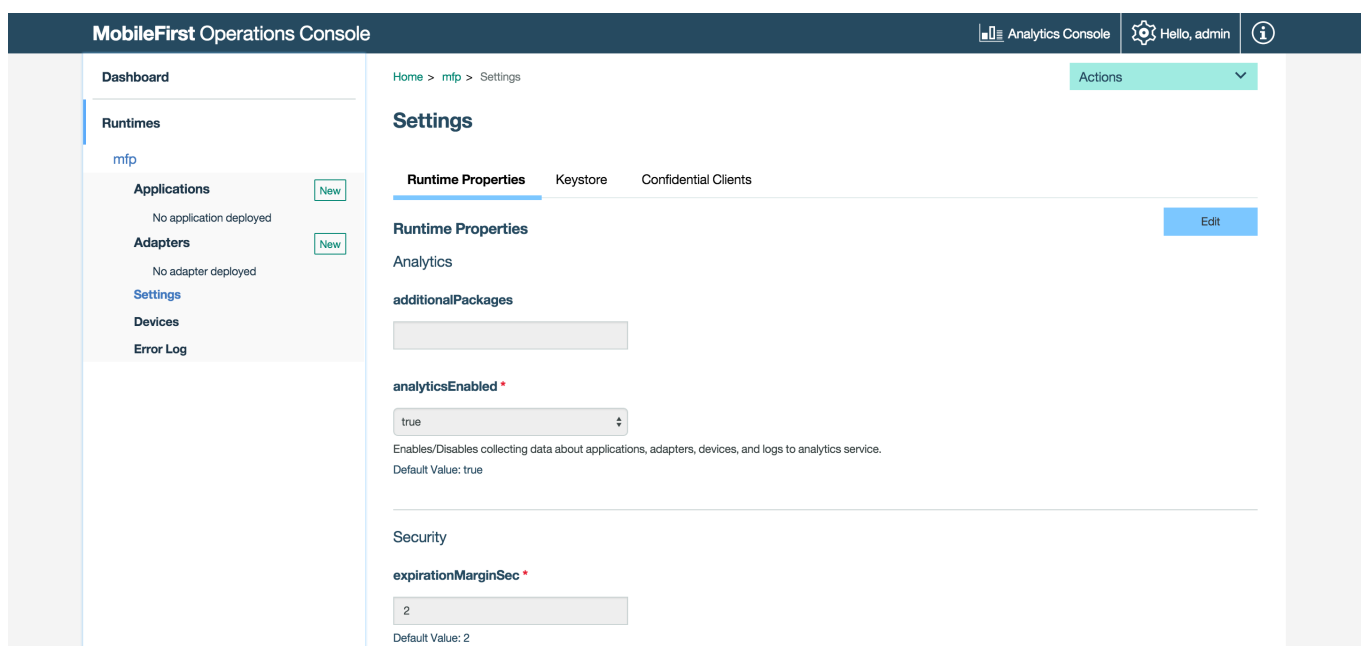
Dashboard

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.



Applications

Creating applications

Provide basic application values and download Starter Code.

Managing applications

Manage and configure registered applications by use of [Direct Update](#) ([../using-the-mfpf-sdk/direct-update/](#)), [Remote Disable](#), [Application Authenticity](#) ([../authentication-and-security/application-authenticity/](#)), and setting security parameters ([../authentication-and-security/authorization-concepts/](#)).

Authentication and Security

Configure application security parameters, such as the default token expiration value, map scope elements to security checks, define mandatory application scopes and configure security check options.

Learn more ([../authentication-and-security/](#)) about the MobileFirst security framework.

The screenshot shows the MobileFirst Operations Console interface. The top navigation bar includes 'MobileFirst Operations Console', 'Analytics Console', and user information 'Hello, admin'. The left sidebar shows a breadcrumb trail: 'Home > mfp > MyApp > iOS 1.0'. The main content area is titled 'MyApp | iOS v 1.0 | com.sample.myapp' and has tabs for 'Management', 'Authenticity', 'Security' (selected), 'Log Filters', and 'Configuration Files'. The 'Security' section is titled 'Security' and contains the text: 'This is where you will set up the advanced security framework configuration offered by MobileFirst Platform to protect your enterprise data and APIs.' Below this, there is a 'Token Configurations' section with the text: 'Configure the access tokens provided by the MobileFirst Server'. A form field for 'Maximum token expiration (seconds) *' is set to '3600' with an 'Edit' button. Below that, there is a 'Map scope elements to security checks' section with the text: 'Configure one or more authentications required for the client to get proper permissions for accessing a protected resource.' and a 'Create New' button. At the bottom, a message states: 'You have not mapped any scope elements to security checks. Get started by clicking "Create New"' with an icon of a smartphone.

Notifications

Set-up push notifications ([../notifications/push-notifications-overview/](#)) and related parameters, such as certificates and GCM details, define tags, as well as send notifications to devices.

The screenshot shows the MobileFirst Operations Console interface for the 'Push' settings. The top navigation bar is the same as the previous screenshot. The left sidebar shows a breadcrumb trail: 'Home > mfp > com.sample.myapp > Push'. The main content area is titled 'Push' and has tabs for 'Send Push', 'Tags', and 'Push Settings' (selected). The 'Push Settings' section is titled 'Push Notification Settings' and contains the text: 'Configure your push notifications here. For detailed instructions, take a look at our [Push Notifications Guide](#).' Below this, there is a form for 'Apple Push Notifications Certificate'. It includes a link to 'Learn more about how to generate and use Apple Push Notification service (APNs) certificates in [Apple Push Notifications certificates guide](#)'. A 'Choose use *' section has two radio buttons: 'Production' (selected) and 'Sandbox'. A 'Select PKCS 12 (.p12) File *' section has a 'No file selected' button and a 'Browse' button. A 'Password *' section has an 'Enter Password' input field and a 'Save' button.

Adapters

Creating adapters

Register a MobileFirst adapter ([../adapters/](#)) and download Starter Code, as well as update an adapter on-the-fly by updating its properties without needing to re-build and re-deploy the adapter artifact.

MobileFirst Operations Console

Analytics ConsoleHello, admin

Dashboard

Runtimes

mfp

Applications

New

No application deployed

Adapters

New

No adapter deployed

Settings

Devices

Error Log

Home > mfp > Create a new Adapter

Actions

Create a new Adapter

Deploy Adapter

Adapters are used to securely connect back-end systems to client applications and cloud services. Adapters are built as Maven projects and can be written in JavaScript or Java.

Follow these steps to create an adapter

Hide guide

1 Set up your development environment

Development of adapters can be done in any environment that supports Maven. In order to build, deploy and update adapters, you can use Maven or the MobileFirst Platform Foundation command line tool.

Installing the command line interface (CLI)

If you haven't, download and install Node.JS. Install the CLI using npm utility of Node.js. The MFP CLI will automatically be downloaded by the npm utility.

```
npm install -g mfpdev-cli
```

Installing Maven

Follow the instructions on the [Apache Maven website](#) to download and install Maven.

Adapter properties

After an adapter is deployed, it can be configured in the console.

MobileFirst Operations Console

Analytics ConsoleHello, demo

Dashboard

Runtimes

mfp

Settings

Applications

Create new

com.ibm.hellocordova

oid

Adapters

Create new

javaAdapter

Devices

Client Logs

Error Log

Home > mfp > javaAdapter

Delete

javaAdapter

ConfigurationsResourcesConfiguration Files

Resources

| URL | Methods | Security |
|--------------------------------|---------|----------|
| /users | GET | |
| /users/helloUserQuery | GET | |
| /users/newUsers | PUT | |
| /users/{first}/{middle}/{last} | POST | |
| /users/{username} | GET | |

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.

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

MobileFirst Operations Console

Analytics ConsoleHello, admin

Dashboard

Runtimes

mfp

Applications

New

PinCodeSwift

Adapters

New

PinCodeAttempts

ResourceAdapter

Settings

Devices

Error Log

Home > mfp > Devices

Actions

Devices

Search device by user identifier or display name (3 chars at least).

| Device ID | Display Name | User ID | Model & OS Version | Device Status | Actions |
|--------------------------------------|--------------|---------|--------------------|---------------|---------|
| 7D518329-2B1A-4E2E-B04F-7BF9EA5B972B | | | iPhone ios 9.2 | Active | |

Installed Applications

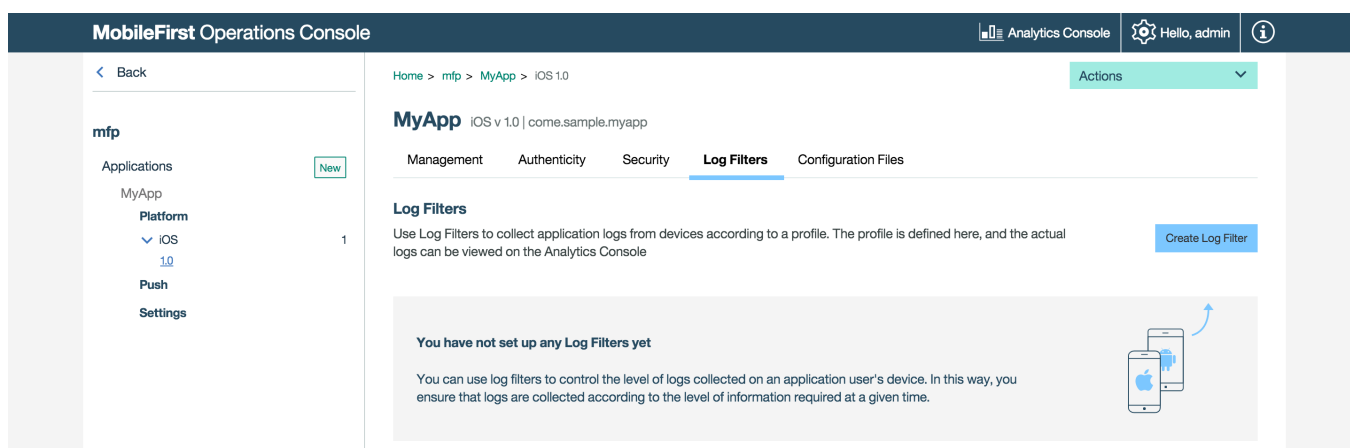
| Application Name | Application Status |
|-------------------------|--------------------|
| com.sample.PinCodeSwift | Enabled |

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 API calls such as `WLResourceRequest`, and is applied automatically.

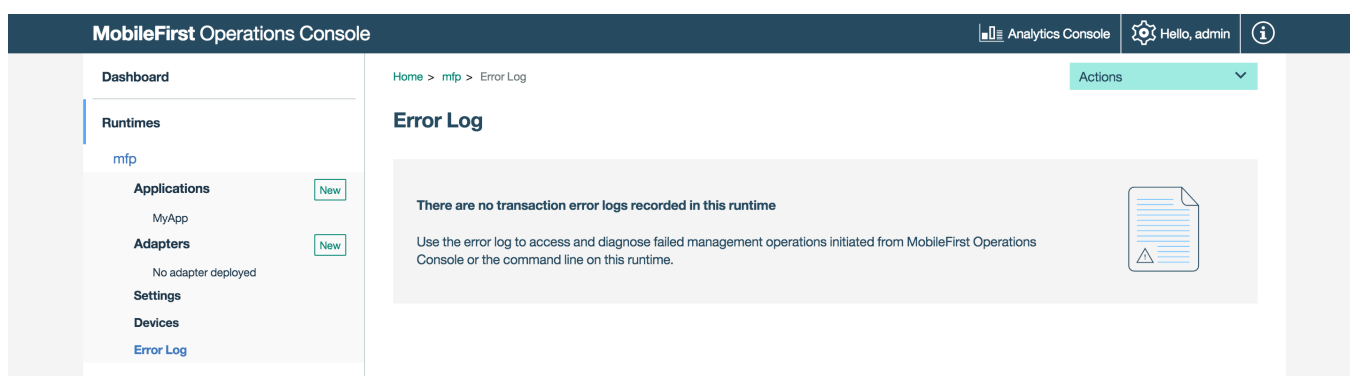
For more information, see the topic about client-side log capture configuration in the user documentation.



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.

The screenshot shows the MobileFirst Operations Console interface. The left sidebar contains navigation links: Dashboard, Runtimes (selected), Applications, Adapters, Settings, Devices, and Error Log. The main content area is titled 'License Tracking' and shows the 'mfp' runtime. A message states: 'Please note that the report has not yet been run'. Below this, there are three sections: 'Application License Tracking' (Number of Applications: 0), 'Addressable Device License Tracking' (Number of Addressable Devices, Target Category Undefined: 0, Target Category B2C: 0, Target Category B2E: 0), and 'Client Device License Tracking' (Number of Server Installations in Cluster: Check your application server administrative console, Number of *Active Client Devices: 0).

Downloads

For situations where Internet connectivity is not available, you can download a snapshot of the various development artifacts of MobileFirst Platform Foundation from the Downloads page.

The screenshot shows the MobileFirst Operations Console interface. The left sidebar contains navigation links: Dashboard, Runtimes (selected), Applications, Adapters, Settings, Devices, and Error Log. The main content area is titled 'Downloads' and shows the 'Tools' tab. It lists three items for download: 'MobileFirst Platform Development CLI' (Download this thing and then run `npm install -g mfpdev-cli.tgz`), 'Adapter Tooling' (Everything you need to develop adapters using Maven), and 'MobileFirst Extension for OAuth Security' (You can protect your resources that are running on WebSphere® Application Server or WebSphere Application Server Liberty servers with OAuth-based IBM MobileFirst™ Platform Foundation security). Each item has a 'Download' button with a download icon.