

# Troubleshooting Analytics

## Overview

Find information to help resolve issues that you might encounter when you use the MobileFirst Foundation's Operational Analytics.

### There is no data in the analytics console

Check the following possibilities.

- Verify that your apps are set to point to the MobileFirst Server, which forwards the logs to the MobileFirst Analytics Server. Ensure that the following values are set in the `mfpclient.plist` (iOS), `mfpclient.properties` (Android), or `config.xml` (Cordova) files.

```
protocol = http or https
host = the IP address of your MobileFirst Server
port = the HTTP port that is set in the server.xml file for reporting analytics
wlsServerContext = by default "/mfp/"
```

- Ensure that your MobileFirst Server is pointing to your MobileFirst Analytics Server.

```
/analytics-service
/analytics
```

- Check that you are calling the send method.

- iOS:

- Objective-C: `[[WLANalytics sharedInstance] send];`
- Swift: `WLANalytics.sharedInstance().send()`
- Android: `WLANalytics.send();`
- Cordova: `WL.Analytics.send();`
- Web: `ibmmfpanalytics.send();`

### Why is there crash data in the Crash Overview table, but nothing in the Crash Summary table?

The crash logs must be sent to the server once the app is again running. Verify that your apps are sending logs after a crash. To be safe, send logs on app start-up to ensure that any previously unsent information is reported.

### Why is there no data in the Server Usage Flow graph or the Network Request graph?

Configure your apps to collect analytics on the Network device event.

```
ibmmfpanalytics.logger.config({analyticsCapture: true});
```

- For cross-platform apps that use Cordova, follow the iOS or Android guides, as the configurations are the same as for native apps.
- To enable the capture of network analytic data in iOS, add the following code in your Application Delegate `application:didFinishLaunchingWithOptions` method.

#### Objective-C

```
WLANalytics *analytics = [WLANalytics sharedInstance];  
[analytics addDeviceEventListener:NETWORK];
```

#### Swift

```
WLANalytics.sharedInstance()  
WLANalytics.sharedInstance().addDeviceEventListener(NETWORK)
```

- To enable the capture of network analytic data in Android, add the following code in your Application subclass `onCreate` method.

#### Java

```
WLANalytics.init(this);  
WLANalytics.addDeviceEventListener(DeviceEvent.NETWORK);
```

## Why is there no data for app sessions?

Configure your apps to collect analytics using the Lifecycle device event listener.

- For cross-platform apps that use Cordova, follow the iOS or Android guides, as the configurations are the same as for native apps.
- To enable the capture of network analytic data in iOS, add the following code in your Application Delegate `application:didFinishLaunchingWithOptions` method.

#### Objective-C

```
WLANalytics *analytics = [WLANalytics sharedInstance];  
[analytics addDeviceEventListener:LIFECYCLE];
```

#### Swift

```
WLANalytics.sharedInstance()  
WLANalytics.sharedInstance().addDeviceEventListener(LIFECYCLE)
```

- To enable the capture of network analytic data in Android, add the following code in your Application subclass `onCreate` method.

#### Java

```
WLANalytics.init(this);  
WLANalytics.addDeviceEventListener(DeviceEvent.LIFECYCLE);
```

*Last modified on*