



Tivoli Endpoint Manager

BigFix Dashboard

Helping you monitor and control your Deployment.

By Daniel Heth Moran

Version 1.1.0

<http://bigfix.me/dashboard>

Copyright Stuff

This edition first published in 2012.

© Copyright 2012 Daniel Heth Moran

Content may not be reproduced or replicated without express written consent of Daniel Heth Moran.

For any/all inquiries, contact: danielheth@hotmail.com

Or tweet: @DanielHeth

URL: <http://danielheth.com> or <http://bigfix.me>

BigFix and Tivoli Endpoint Manager are registered trademarks of IBM International Corporation. Other product and company names mentioned herein may be the trademarks of their respective owners. Where those designations appear in this guide, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The author and publisher have taken care in the preparation of this guide, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

About

The BigFix Dashboard was originally setup to provide "Ops" style big screen dashboards. The data being displayed would relate directly to one or more deployments and provide a fast and easy way to asses the status of those deployments.

BigFix Dashboard has since grown into a back end service, an iPhone/iPad app, and a graphical Dashboard. We even have integrations into Leftronic.com dashboards which can be displayed across the world.

My goals with this project was to provide the status of essential services, general counts and the ability to run relevance remotely. I believe these have been accomplished and in grand fashion.

In the following pages you'll learn how to install, configure and access the deployment status information. If you have any questions, please contact me at danielheth@hotmail.com.

Change List

Every release has little bug-fixes and added features. This is a list of those changes:

1.0.2 -> 1.1.0

- Dashboard app changed to fix reloading components if server is not available.
- Added additional code to catch and display descriptive error messages
- Modified the way components are displayed. If the back-end service XML is NOT available, your components will not disappear, they will simply turn red indicating no status is available.
- Other minor bug fixes

1.0.0 -> 1.0.2

- Renamed project from BigFix Admin to BigFix Dashboard to reduce confusion related to the official Admin Tool.

1.0.0 -> 1.0.1

- Added SQL module to allow admins to configure SQL statement calls and retrieve information like Travel Time.

Installation

Installation of the back end service is relatively easy. It can be accomplished in one of a few ways.

Automatic Installation

I have created several TEM tasks which make installation easy to do. You can download the essential tasks here:

<http://bigfix.me/dashboard>

Import the tasks into your deployment, likely under a custom site called "BigFix Dashboard". Then launch the one called "BigFix Dashboard (Backend Service) v1.1.0 Available!". Target your root server and it takes care of setting up a basic configuration.

NOTE: The back end service can be installed onto any server and can remotely determine the status of your deployment but will require additional configuration.

Manual Installation

To manually install the back end service you will need to download the installers file from <http://bigfix.me/dashboard>. It contains the MSI installation file. Simply install the program on to the root server and the basic configuration will be setup automatically.

Configuration

I have designed the BigFix Dashboard to be broken down into "components". Each component is a separate thread which monitors the "item" it is configured to watch. All of these components are gathered together and returned to the clients as a simple XML document. You can see an example of this document here: <http://bigfix.me:3001/>

Each component is configured within the registry at the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\BigFix\Dashboard\Settings\[xxx]
```

Where the [xxx] is each component that is defined. All components share the following Values:

(Default)	Defines the "type" of component: service, relevance, registry, or general
Category	Defines how the endpoint will be sorted on the various display apps: general, service, or other
Desc	Provides the display app user with a description of this module and its importance.
Title	A short title also used for the various display apps.
Hidden	This will prevent the components status from being included in the status XML. Typically this type of component is used for providing an additional data feed for a Leftronic.com dashboard. (optional, Default is False)
ReadOnly	This property determines if the client can manipulate the status. Currently only services can be manipulated... IE started/stopped. But only if this property is present and set to "False". (Optional default is True).
Interval	You can configure the thread timer interval using this property. It should be specified in milliseconds. (optional default is 900,000 or 15min).

Properties

I've built in some properties that let you easily configure and troubleshoot your BigFix Dashboard Backend Service.

DefaultCMD

The default command is a registry value which determines what it will do when an end-user goes directly to the URL. In my example deployment (<http://bigfix.me:3001>) the user will see the status xml file. This is because the DefaultCMD registry value contains "status" (without "). This value sits in the following key: `HKEY_Local_Machine\SOFTWARE\Wow6432Node\BigFix\Dashboard` If you want the user to receive a blank page simply delete the "status" from this Value.

Port

I have also provided administrators the ability to configure the back end service on any desired port. By default the port number is 3001. You can change that by putting a Value in the above registry location called "Port". Enter the number of the port you'd like to change the service to.

You will need to restart the service if it is already running.

WebReports

Relevance components are configured to run session relevance against your WebReports server. This means we will need to know the URL, Username and Password that this will be done as. Please add the following three keys to the above registry location.

WRUrl	The url of your web reports server. Example: <code>http://localhost:80/webreports?wsdl</code> In this example my root server also hosts the web reports server.
WRUsername	The username to be used for running session relevance.
WRPassword	The related password for the username entered above.

NOTE: This information is stored in plain text so the username and password being used should not be an admin and have simple read-only access to web reports.

Debug

On occasion it may be necessary to troubleshoot the service to determine what it is doing. You can add the "Debug"="True" Value to the registry and it will provide additional logging within the Windows computer's Application Event Log.

For your convenience, I've included a task for easily configuring the Debug Logging feature. Download the task from <http://bigfix.me/dashboard>.

Components

In addition to these standard Values, each "type" of component has additional Values that are required.

For each component-type, I have created a task which will configure the proper values. Download those from <http://bigfix.me/dashboard>.

Service Components

Service components are designed to provide you with status of the specified service. You can monitor your WebReports service, FillDB, etc... The goal here is to design a monitoring and control ability while remote via the iPhone/iPad app or via the desktop Dashboard app.

ServiceName	Name of the service to be monitored.
Machine	Host name or IP of the machine running the service. (<i>optional</i> , default is localhost)

Relevance Components

The Relevance Component is one which lets you execute session relevance on the Web Reports server you specified in Configuration->Properties->WebReports of this document. This relevance statement should be simple in that it returns one value or string. This "status" string will be forwarded to the various apps that consume the backend service's XML.

Relevance	Insert the "simple" relevance here which will return one value. An example would be: number of bes computers
------------------	--

Registry Components

Registry Components allow you to monitor the registry anywhere within the LocalMachine HKEY. You specify the sub-key and the value of the registry setting. We will monitor it and provide that value to the various apps.

Key	This is the sub-key of LocalMachine that is to be monitored. Example: SOFTWARE\ (<i>Wow6432Node\</i>)BigFix\Dashboard
Value	The value to be monitored. Example: DefaultCMD

General Components

General Components are very special type of component. It doesn't actually do anything but read in the "Status" registry value. It works like the Registry component but is locked down and only reads that "Status" value from its own registry key location.

SQL Components

SQL Components allow you to run a query against a SQL Server and include the result in the status XML. Below are the values related to this component:

SQLCmd	This represents the actual SQL command that will be issued to the SQL Server.
SQLServer	The SQL Server we will connect to in order to run the SQLCmd. (Optional, Default = localhost)
SQLDatabase	Database you'll run the SQLCmd against. (Optional, Default = BFEnterprise)

NOTE: We are using "integrated security" as part of opening the SQL Server connection. Thus the username your back-end service is running as must have permissions into the specified server.

iPhone App

The iPhone app was created to provide a mobile "dashboard". It is available in the Apple App Store here:

<http://itunes.apple.com/us/app/bigfix-admin/id533908493?mt=8>

Installation is extremely easy. Follow the link from your "i" device and install this free app. It is published under my "company" called DanielHeth. The name of this app is "BigFix Admin".

Note: I will be renaming this application for the next release once I've added additional features and re-released via Apple's publishing procedure.

Settings

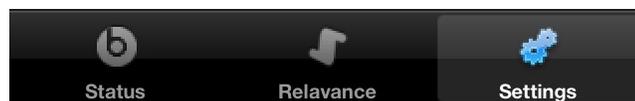
On the "Settings" tab enter the URL and port of your deployment. It will look something like this:



BigFix Admin URL:

Username:

Password:



In this image, you'll see that it is pointed to my personal test lab and will provide read-only access into that deployment.

After you have configured the settings properly, return to the "Status" tab and pull the list down to retrieve the latest status XML.

Usage

Once you have the iPhone/iPad app up and running you have several features to look into.

1. Status - Displays the status of all the defined components. If this is a "service" component and not set to read-only, you will be able to start/stop that service remotely. NOTE: Pull the list down to force a refresh of the server status. You can do this at any time to get the latest status information.
2. Relevance - Here you can run relevance commands if you have your WebReports server settings configured. (see configuration section of this guide). Simply enter your relevance statement and hit Go. The results will be displayed in the list view.
3. Settings - The settings tab provides an easy way to configure your app to point to your deployment. Currently the Username/Password authentication does not work but is expected to be functional with the next release.

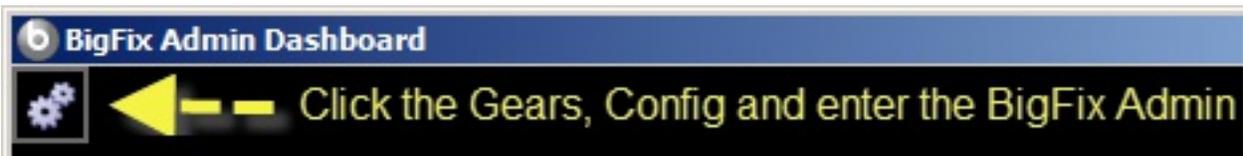
Dashboard App

The Dashboard application can be downloaded from <http://bigfix.me/dashboard>

Settings

Upon launch of the BigFix Dashboard application, you will see indicators leading you to the configuration section.

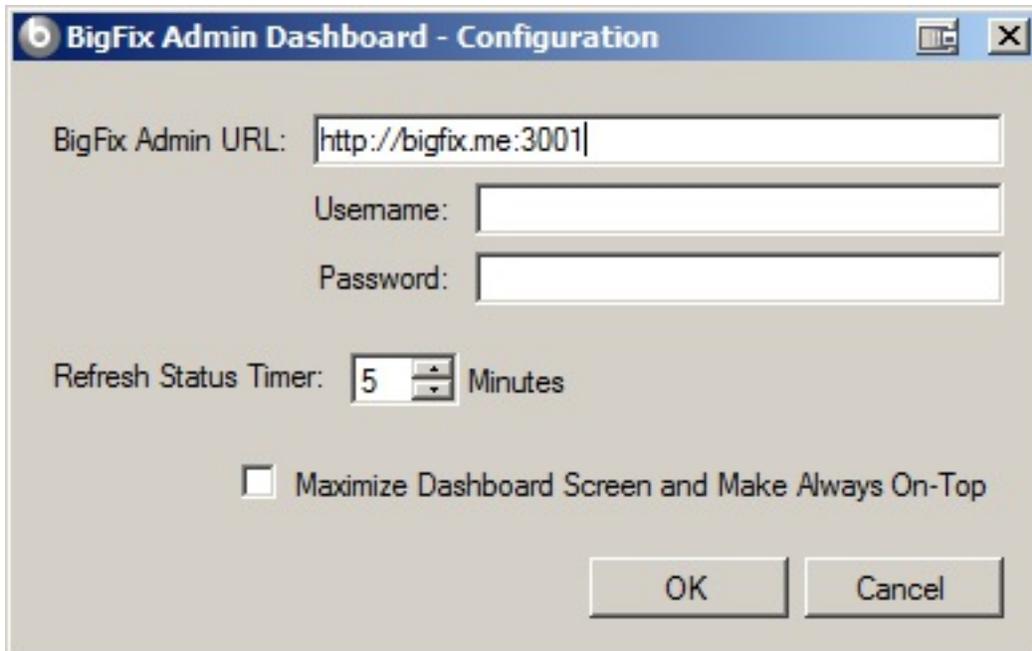
Basically click the Gears icon in the upper left of the window.



This will reveal additional controls for configuring your server connection and dashboard view.



First thing to do is configure the Server URL and authentication settings. Click the "Config" button and type in the appropriate information. NOTE: You can use the settings as follows for a read-only view of my personal lab infrastructure.



BigFix Admin Dashboard - Configuration

BigFix Admin URL:

Username:

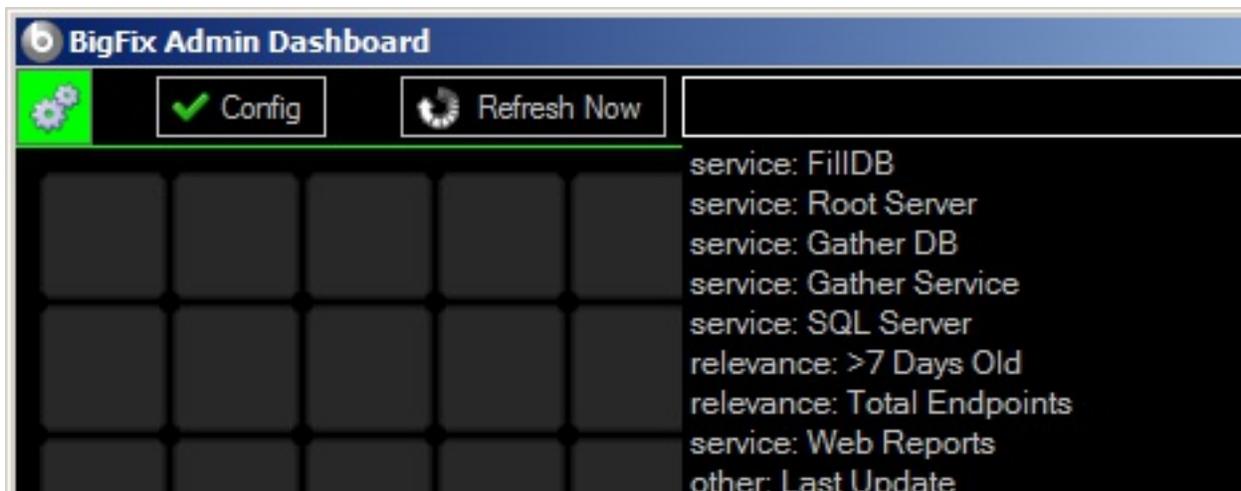
Password:

Refresh Status Timer: Minutes

Maximize Dashboard Screen and Make Always On-Top

OK Cancel

After hitting OK, click the "Refresh Now" button to reach out to your BigFix Dashboard back end service to retrieve the latest status XML. It will then populate the dropdown dialog box and allow you to start adding components.



BigFix Admin Dashboard

Config Refresh Now

service: FillDB
 service: Root Server
 service: Gather DB
 service: Gather Service
 service: SQL Server
 relevance: >7 Days Old
 relevance: Total Endpoints
 service: Web Reports
 other: Last Update

Choose the component you'd like to see and click the "Add" button to add that item to your dashboard. Since the component can only be added once, it will be removed from the drop-down list.



You can now drag-n-drop this component anywhere in the grid you'd like. You can also resize this component by click-n-hold the bottom right corner. If you no longer want to see this component, simply click in the upper right corner.

If the component has additional configuration options, you can click the wrench in the upper left corner. NOTE: None of the components in version 1 of this app have additional options.

When you are done adding your components, click the Gears button again and you will turn off edit-mode.

You may resize the window to the desired size at any time even if edit-mode is turned off. This will let you easily put the view onto a secondary monitor or have a tiny monitoring window you pull up when needed.

Integrations

Integrations are a nice addition to the back service. They allow you to feed data to various services to enhance your ability to monitor your infrastructure. The following are a few of the integrations that have been included with the BigFix Dashboard.

Leftronic Integration

Leftronic.com is a startup company specializing in "Ops" style dashboards that can be accessed and displayed anywhere. Since this project aligns very closely with that concept, I have created a simple integration method for feeding their dashboards data to be displayed.

Their dashboard widgets each receive data via SOAP submissions. Each of the dashboards you create is assigned a unique "AccessKey" and each Widget gets a unique "StreamName".

To feed their dashboard... simple add the following two Values to the component of your choice:

AccessKey	Defines the unique dashboard that we are sending this information to.
StreamName	Defines the unique widget that will display the data being sent.

If both keys are present, then the Leftronic integration code will be triggered and the "status" value will be sent to the dashboard widget.

To learn more about creating Leftronic.com dashboards visit their main website <http://leftronic.com/> or their API documentation at <https://www.leftronic.com/api/>

Wrap it up

The BigFix Dashboard is a very simple way of keeping tabs on your deployment.

This project will grow as admins and users submit their suggestions to: danielheth@hotmail.com.

Also email me if you have any additional questions or comments.

All of the various components are available for download from <http://bigfix.me/dashboard>

- Automatic Installation via BigFix using various Tasks.
- Backend Service and Dashboard GUI - Manual Installer MSI Files.
- iPhone/iPad App: <http://itunes.apple.com/us/app/bigfix-admin/id533908493?mt=8>