IBM MessageSight

MonitoringClient Guide

IBM® MessageSight™ V1 provides a large set of built-in monitoring functionality out of the box; queries from the admin CLI and UI monitoring graphs and tables provide a detailed window into the system.

With IBM MessageSight V1.0.0.1, monitoring becomes even more powerful through the pub/sub External Monitoring feature. With the V1.0.0.1 release, external monitoring clients (JMS, MQTT) can access MessageSight monitoring data from MessageSight by subscribing to the **$SYS/ResourceStatistics** topic tree.

This document describes the *MonitoringClient* sample: a web application that establishes an MQTT connection to MessageSight over a WebSocket, subscribes to the external monitoring topic, organizes and renders the data.

Usage

Step 0: Configure MessageSight

The sample attempts an unsecured WebSocket MQTT connection to MessageSight without a username and password. Make sure that an Endpoint is configured:

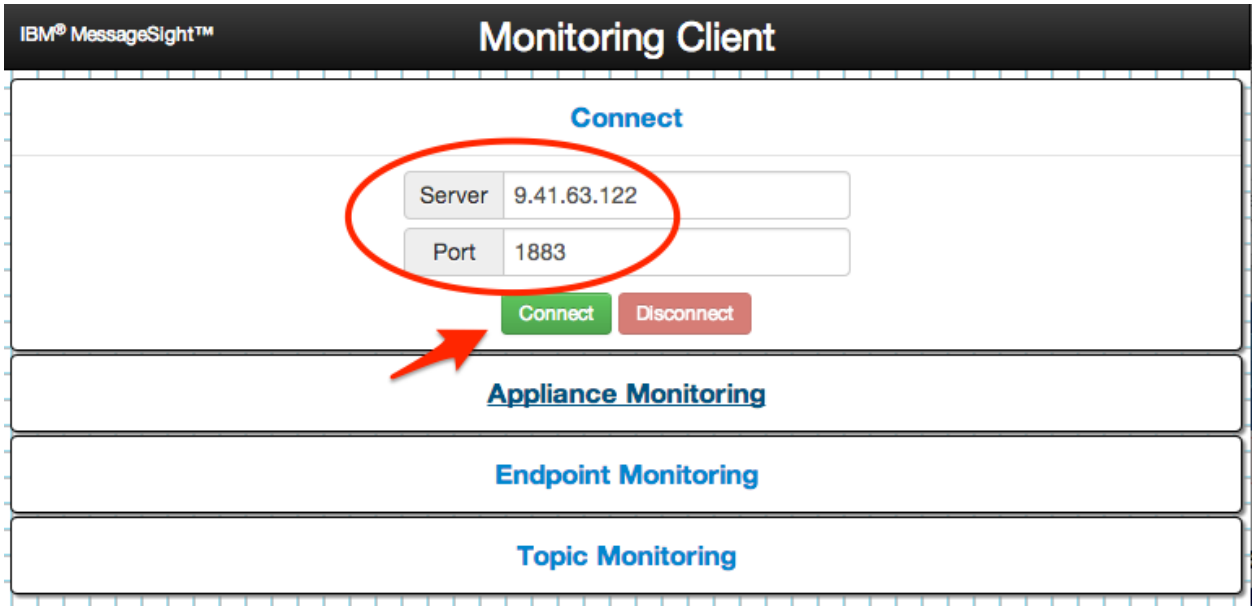
- Without an attached Security Policy
- With a Connection Policy permitting access to all incoming MQTT clients
- With a Messaging Policy permitting subscriptions on **$SYS/ResourceStatistics**

(with a default configuration of the MessageSight appliance, *DemoMqttEndpoint* will fit these criteria)

Step 1 – Connect to MessageSight

Enter the IP and port of the MessageSight endpoint configured in step 0, then press Connect.

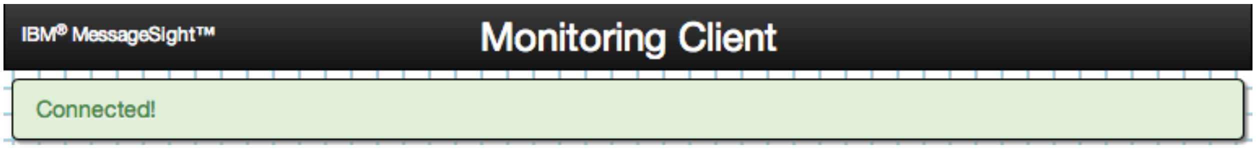
© Copyright IBM Corporation 2013



Server 9.41.63.122

Port 1883

After connecting, the monitoring client will automatically subscribe to the MessageSight external monitoring topic - **$SYS/ResourceStatistics** - and begin receiving data.



Connected!

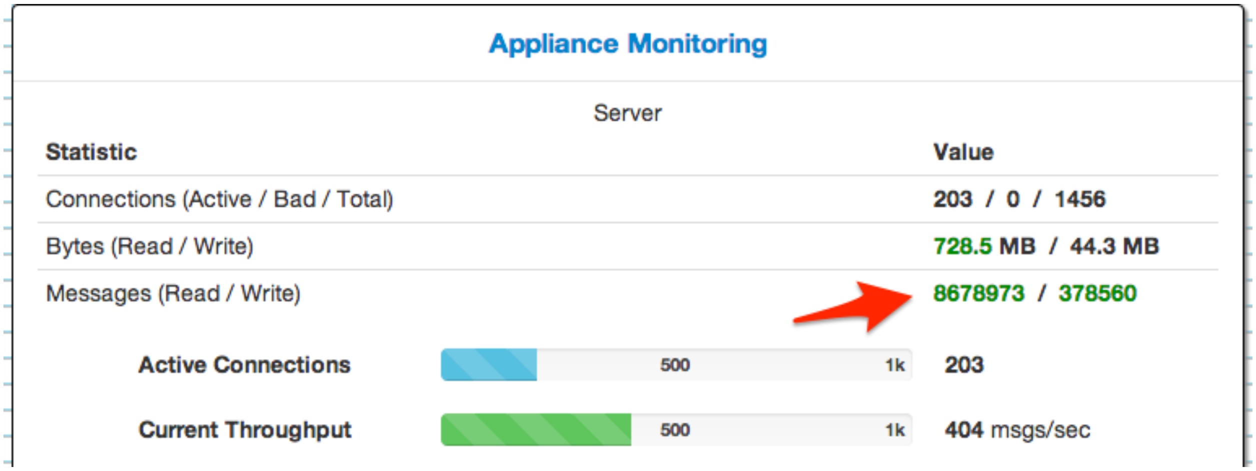
Step 2 – View *Appliance* data

After connecting, the application will automatically switch to the **Appliance Monitoring** accordion pane. This pane shows three sets of monitoring data:

1. **Server.** [collected by aggregating all Endpoint data (Step 3)] Shows global connection/data data, along with live meters indicating the current throughput and number of active connections. These meters have an adaptive maximum value, where "100%" is the next-highest order of magnitude. For example: when there are 131 active connections, the max value in the meter is 1000.
2. **Store.** Shows the memory and disk space used by MessageSight's persistent store.
3. **Memory.** Shows memory usage statistics.

Statistic values in this section will flash green/red with each update if the change in value is positive/negative.

© Copyright IBM Corporation 2013

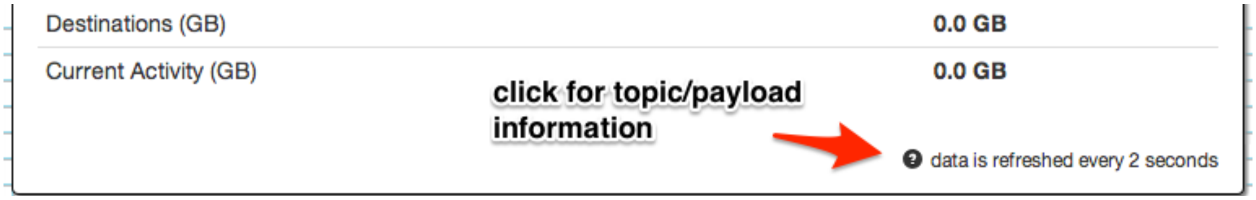


Appliance Monitoring

Server

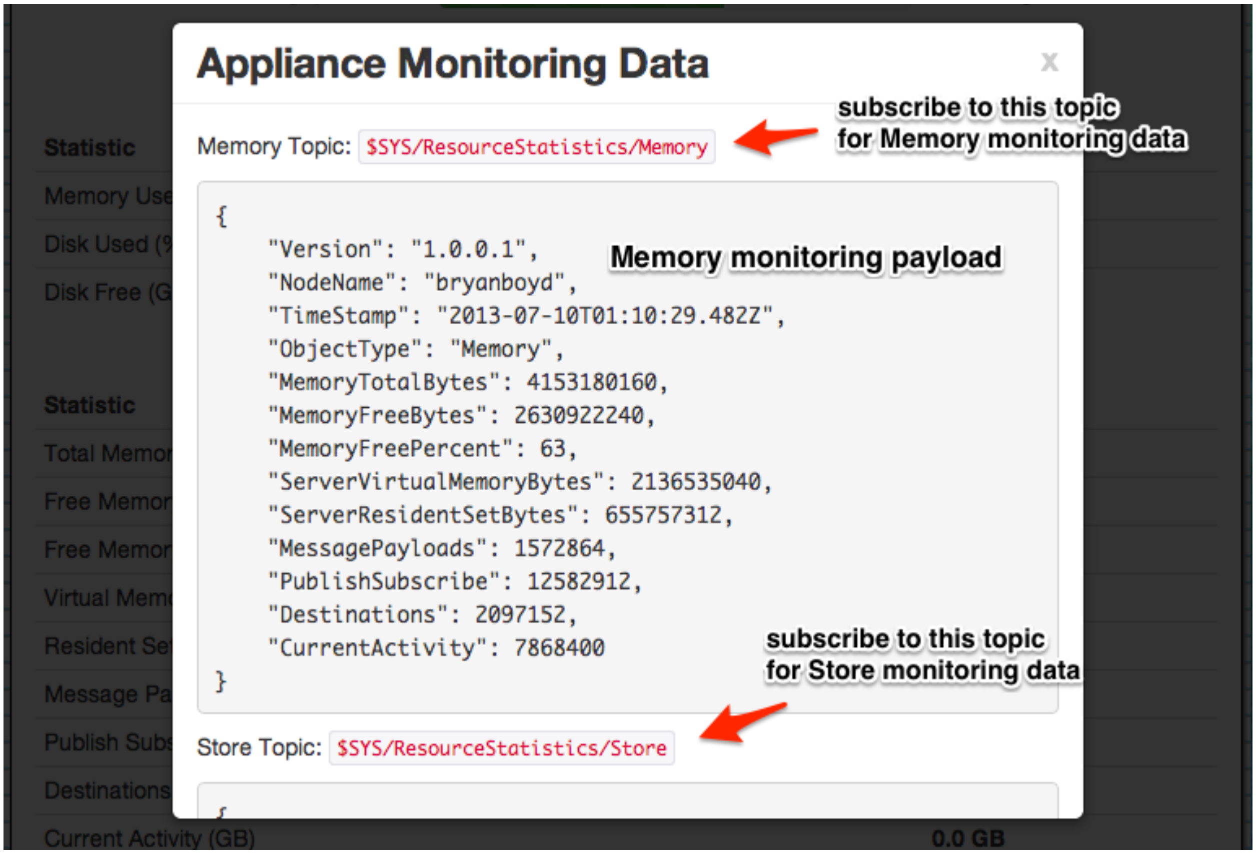
Statistic	Value
Connections (Active / Bad / Total)	203 / 0 / 1456
Bytes (Read / Write)	728.5 MB / 44.3 MB
Messages (Read / Write)	8678973 / 378560

For more information about the Store and Memory monitoring data, click on the question mark at the bottom-right of the pane. The MQTT topic for subscription and the most recent JSON payload will be shown.



Destinations (GB)	0.0 GB
Current Activity (GB)	0.0 GB

click for topic/payload information



Appliance Monitoring Data

Memory Topic: $SYS/ResourceStatistics/Memory

subscribe to this topic for Memory monitoring data

Memory monitoring payload

```
{
    "Version": "1.0.0.1",
    "NodeName": "bryanboyd",
    "TimeStamp": "2013-07-10T01:10:29.482Z",
    "ObjectType": "Memory",
    "MemoryTotalBytes": 4153180160,
    "MemoryFreeBytes": 2630922240,
    "MemoryFreePercent": 63,
    "ServerVirtualMemoryBytes": 2136535040,
    "ServerResidentSetBytes": 655757312,
    "MessagePayloads": 1572864,
    "PublishSubscribe": 12582912,
    "Destinations": 2097152,
    "CurrentActivity": 7868400
}
```

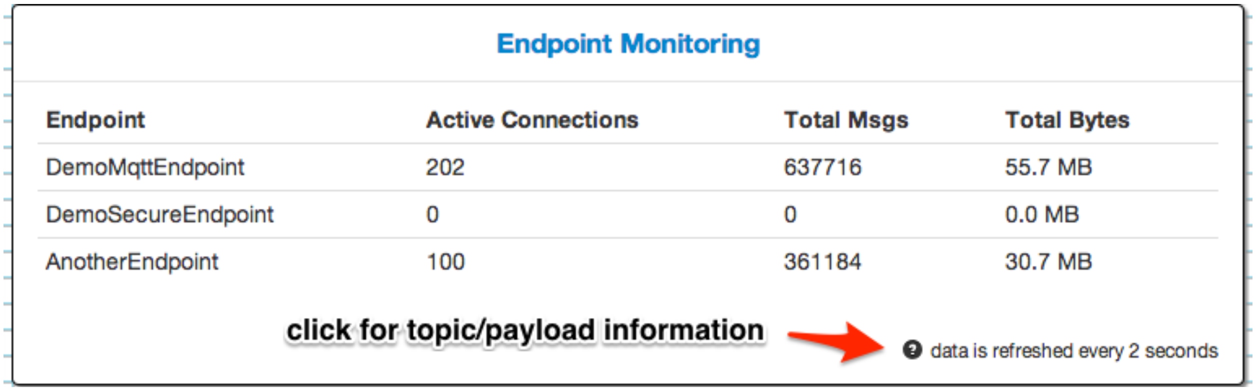
subscribe to this topic for Store monitoring data

Store Topic: $SYS/ResourceStatistics/Store

© Copyright IBM Corporation 2013

Step 3 – View *Endpoint* and *Topic* data

Switch to the **Endpoint Monitoring** accordion pane. This pane lists all Endpoints configured on MessageSight, along with the current number of connections and total data handled (sent + received). As in the Appliance Monitoring pane, you can click on the question mark to view subscription topic and payload information.

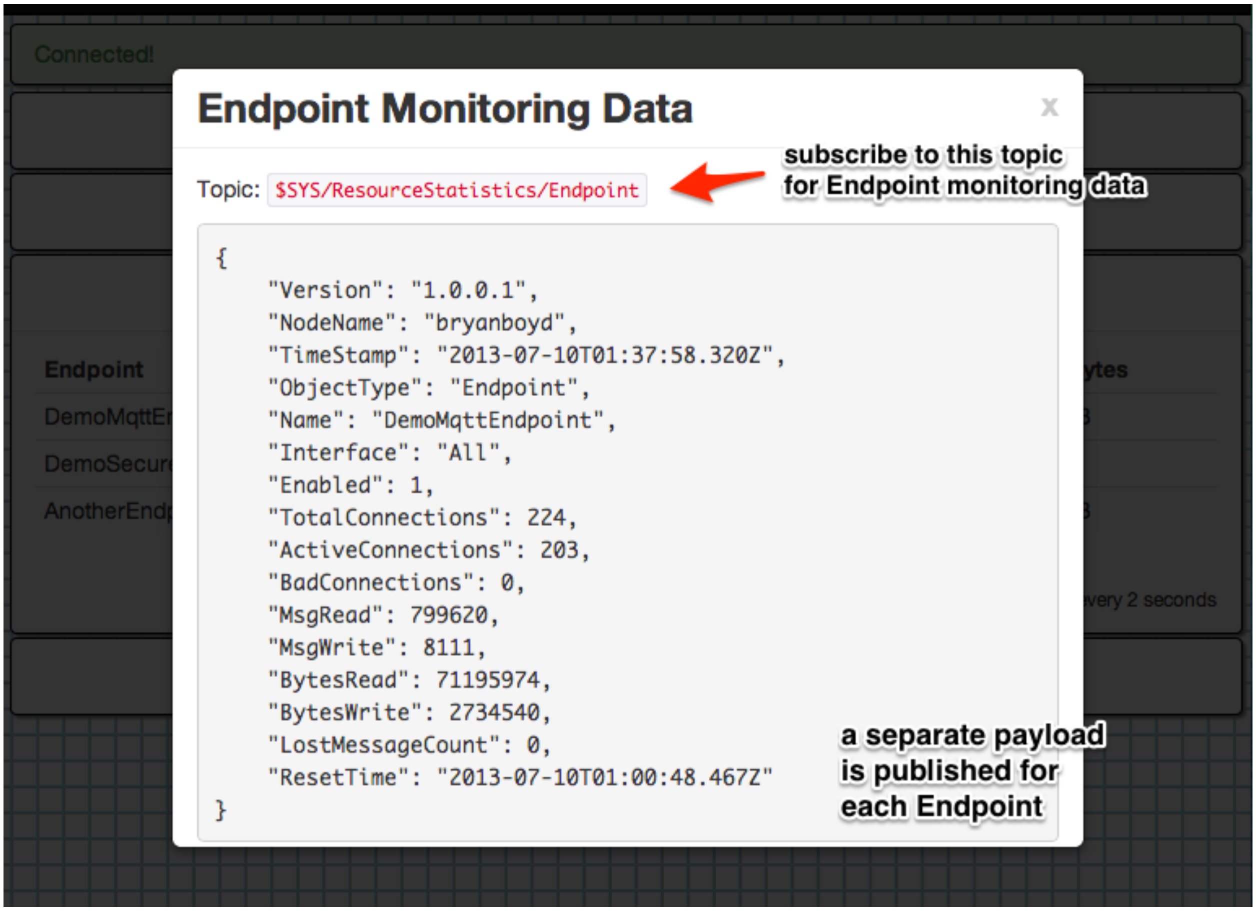


Endpoint Monitoring

Endpoint	Active Connections	Total Msgs	Total Bytes
DemoMqttEndpoint	202	637716	55.7 MB
DemoSecureEndpoint	0	0	0.0 MB
AnotherEndpoint	100	361184	30.7 MB

click for topic/payload information

? data is refreshed every 2 seconds



Endpoint Monitoring Data

Topic: $SYS/ResourceStatistics/Endpoint

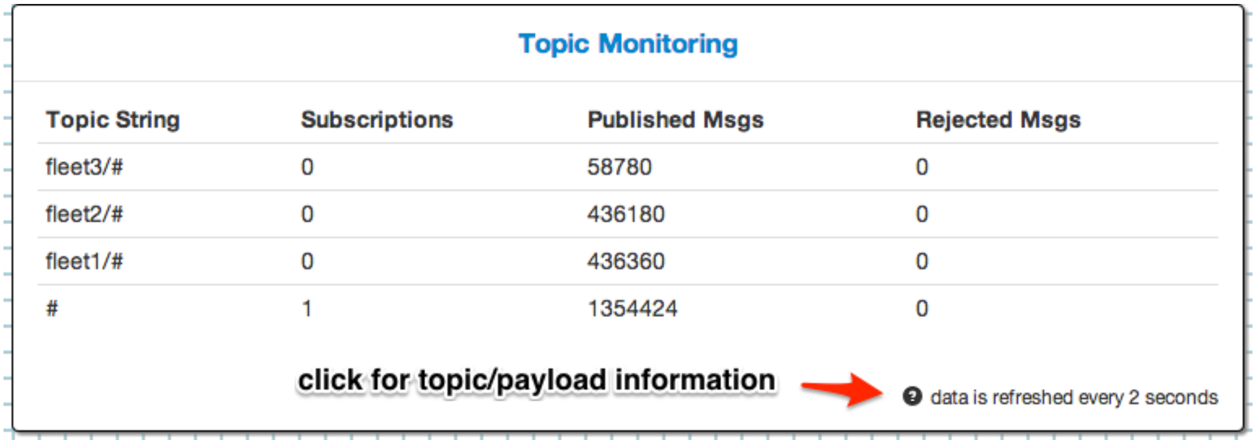
subscribe to this topic for Endpoint monitoring data

```
{
    "Version": "1.0.0.1",
    "NodeName": "bryanboyd",
    "TimeStamp": "2013-07-10T01:37:58.320Z",
    "ObjectType": "Endpoint",
    "Name": "DemoMqttEndpoint",
    "Interface": "All",
    "Enabled": 1,
    "TotalConnections": 224,
    "ActiveConnections": 203,
    "BadConnections": 0,
    "MsgRead": 799620,
    "MsgWrite": 8111,
    "BytesRead": 71195974,
    "BytesWrite": 2734540,
    "LostMessageCount": 0,
    "ResetTime": "2013-07-10T01:00:48.467Z"
}
```

a separate payload is published for each Endpoint

© Copyright IBM Corporation 2013

Now, switch to the **Topic Monitoring** accordion pane. This pane shows all **Topic Monitors** (counters set on a topic tree) that have been configured on MessageSight, with the number of subscribers, published, and rejected message counts. As with the other panes, you can click on the question mark to view subscription topic and payload information.

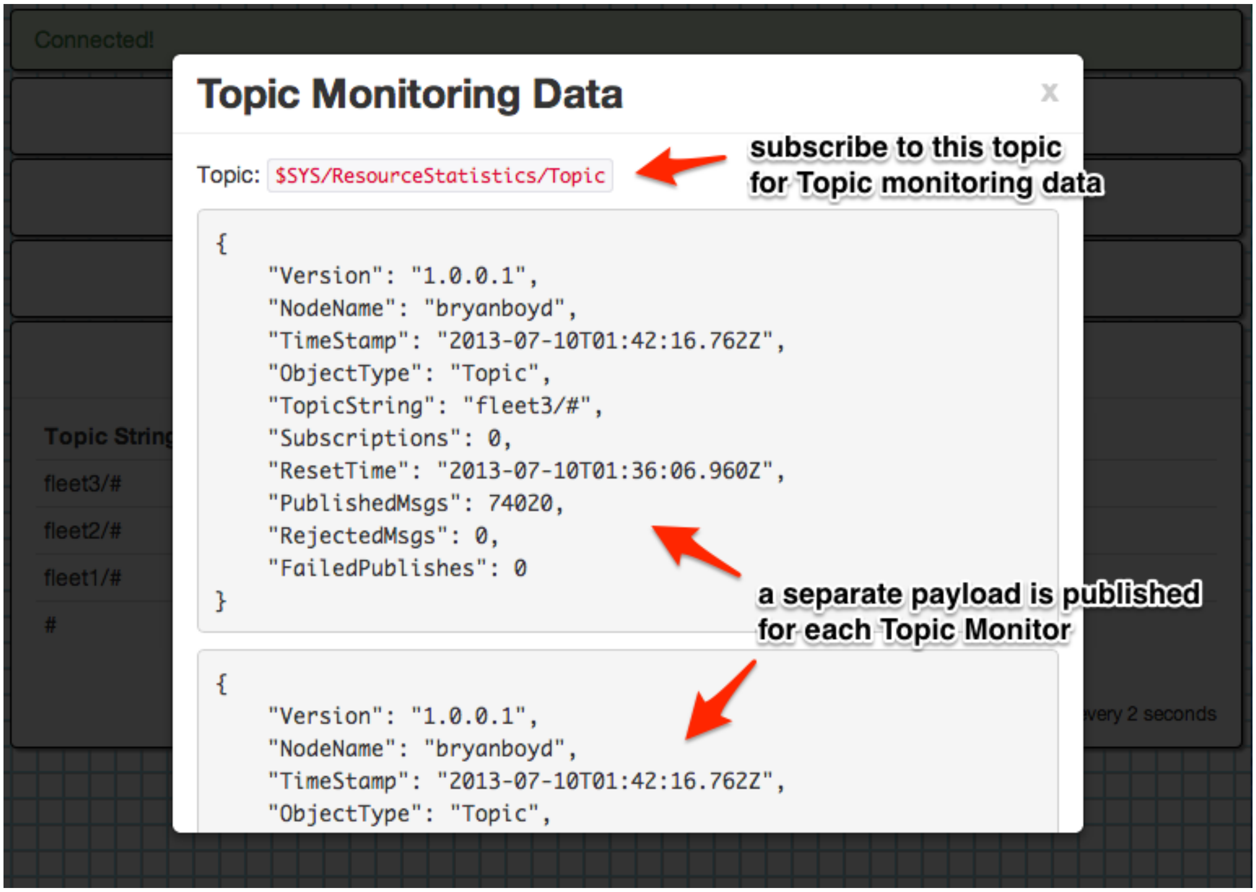


Topic Monitoring

Topic String	Subscriptions	Published Msgs	Rejected Msgs
fleet3/#	0	58780	0
fleet2/#	0	436180	0
fleet1/#	0	436360	0
#	1	1354424	0

click for topic/payload information

❓ data is refreshed every 2 seconds



Topic Monitoring Data

Topic: $SYS/ResourceStatistics/Topic

subscribe to this topic for Topic monitoring data

```
{
    "Version": "1.0.0.1",
    "NodeName": "bryanboyd",
    "TimeStamp": "2013-07-10T01:42:16.762Z",
    "ObjectType": "Topic",
    "TopicString": "fleet3/#",
    "Subscriptions": 0,
    "ResetTime": "2013-07-10T01:36:06.960Z",
    "PublishedMsgs": 74020,
    "RejectedMsgs": 0,
    "FailedPublishes": 0
}
```

a separate payload is published for each Topic Monitor

```
{
    "Version": "1.0.0.1",
    "NodeName": "bryanboyd",
    "TimeStamp": "2013-07-10T01:42:16.762Z",
    "ObjectType": "Topic",
```

© Copyright IBM Corporation 2013

Implementation

User Interface

jQuery and Bootstrap are used by MonitoringClient for layout, styling, dynamic actions, convenience methods, etc.

Messaging

MonitoringClient utilizes the MQTT-over-WebSocket JavaScript library (*mqttws31.js*) provided in IBM's Mobile Messaging & M2M Client Pack

Most of the *MonitoringClient* source code (*js/MonitoringClient.js*) simply serves to update the web application monitoring data view. The basic function of the monitoring client can be implemented in three steps.

Step 1 – Connect to MessageSight

```
$(“#connectButton”).click(function(event) {
    var server = $(“#connectServer”).val();
    var port = $(“#connectPort”).val();
    MonitoringClient.connect(server, port);
});
```

The “Connect” button in *MonitoringClient* has a document ID of **connectButton**. We define an onClick event function to locate the Server and Port input text, then call the *MonitoringClient* connect() function.

Step 2 – Subscribe to monitoring topic

```
function connect(server, port) {
    try {
        client = new Messaging.Client(server, port, clientId);
    } catch (error) { … }

    client.onMessageArrived = onMessage;
    client.onConnectionLost = function() { … }

    var connectOptions = new Object();
    connectOptions.useSSL = false;
    connectOptions.cleanSession = true;
    connectOptions.keepAliveInterval = 3600;

    connectOptions.onSuccess = (function(mqttclient) {
        return function() {
            …
            var monitoringTopic = “$SYS/ResourceStatistics/#”;
            mqttclient.subscribe(monitoringTopic);
            …
        }
    })(client);
    connectOptions.onFailure = function() { … }
```

© Copyright IBM Corporation 2013

```
    client.connect(connectOptions);
}
```

The **connect** function instantiates an MQTT client (*Messaging.Client)* and attempts a connection to MessageSight. We define connection options and callbacks for the client, including **onMessageArrived** (invoked when client receives a message) and **onSuccess** (invoked when a connection is established). In **onSuccess** we do the work of subscribing to the monitoring topic; **onMessage** is defined in the next step.

Step 3 – Process monitoring messages, update UI

```
function onMessage(msg) {
    var topic = msg.destinationName;
    var payload = JSON.parse(msg.payloadString);
    var dataType = topic.split(“/”)[2];   /* $SYS/ResourceStatistics/<dataType> */

    switch (dataType) {
        case: “Store”:
            updateStoreData(payload);      break;
        case: “Memory”:
            updateMemoryData(payload);     break;
        case: “Endpoint”:
            processEndpointData(payload);  break;
        case: “Topic”:
            processTopicData(payload);     break;
    }
}
```

Our **onMessage** callback first parses the topic and JSON payload out of the message. The monitoring data type is determined by the 3rd field in the topic tree. For Store and Memory data, we just overwrite the value in the UI. Endpoint and Topic data requires a different approach, as multiple Endpoint and Topic messages (one for each configured MessageSight object) are received during every 2-second monitoring update interval. **processEndpointData** and **processTopicData** (not shown) do the work of combining monitoring data from the same update interval together for a batch update of the UI.

© Copyright IBM Corporation 2013

Exercises

For further exercise, extend *MonitoringClient* by implementing the following function:

1. In the Appliance Monitoring section, add a meter indicating the current throughput in units of MB/sec.

2. In the Appliance Monitoring section, add "average throughput over time" statistics for various intervals (last minute, 5 min, 15 min, etc.) by collecting and averaging throughput data while the monitoring client is connected.

Notices
IBM, the IBM logo, ibm.com, and MessageSight are trademarks of IBM Corporation, registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legals/copytrade.shtml. Other product and service names might be trademarks of IBM or other companies.

© Copyright IBM Corporation 2013