Unzip bridge slapper to a directory of your choice. Open a command window and 'cd' to the directory where you unzipped the files to. You can start bridge slapper with the following command:

```
java –jar pacbridge-slapper-2.0.0.jar [--config=<config-filename>] [--name=<instance-name>]
```

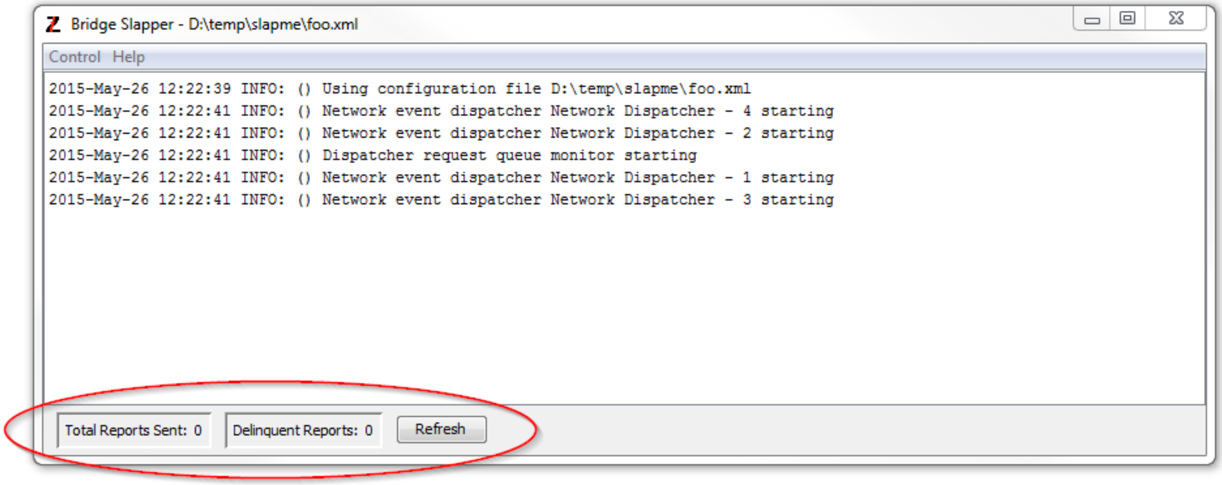
--config Optional. Allows you to specify a configuration file. By default, it will search for, and use a configuration file called `bridgeslapper.xml`.

--name Optional. Allows you to specify a name that will appear in the title bar of the main window. If not supplied, the name will show the full path of the configuration file.

Operation

Bridge slapper works by using *device templates* to create a number of FIG device simulations that operate like real FIG devices. They open a connection to the bridge and then send reports at defined intervals.

Each report that is sent to the bridge will have a unique event ID associated with it. This event ID is stored in the bridge and later used to correlate with site reports published by the bridge. Bridge slapper will connect to the same JMS server as the one used by the bridge and will listen for site reports. Whenever a site report is received, it will extract the event ID and remove it from its list of event ID. At the bottom of the Bridge Slapper UI is a count of the total number of reports published by Bridge Slapper and the number of delinquent reports. A report is considered *delinquent* if no site report is published with that event ID.



```
2015-May-26 12:22:39 INFO: () Using configuration file D:\temp\slapme\foo.xml
2015-May-26 12:22:41 INFO: () Network event dispatcher Network Dispatcher - 4 starting
2015-May-26 12:22:41 INFO: () Network event dispatcher Network Dispatcher - 2 starting
2015-May-26 12:22:41 INFO: () Dispatcher request queue monitor starting
2015-May-26 12:22:41 INFO: () Network event dispatcher Network Dispatcher - 1 starting
2015-May-26 12:22:41 INFO: () Network event dispatcher Network Dispatcher - 3 starting
```

Total Reports Sent: 0 | Delinquent Reports: 0

The values do not update in real time. Users can press the "Refresh" button to update the values as the simulation runs. These to values are counts of message from the start of Bridge Slapper. Stopping and starting a simulation will not reset these values.

Starting a simulation is done by opening the "Control" menu and clicking the "Start" menu option. Stopping the simulation is done by clicking the "Stop" menu option.

Configuration

The configuration file contains information that tells Bridge Slapper how to connect to the bridge and how to connect to the JMS server. For example:

```
<BridgeSlapper>

    <Jms>
        <QueueManagerName>QM_csmqdev1</QueueManagerName>
        <HostName>csmqdev1</HostName>
        <ClientId>pacbridge_id</ClientId>
        <RawDataDestinationName>topic://scada/system/events</RawDataDestinationName>
    </Jms>

    <Bridge>
        <Address>192.168.169.64</Address>
        <Port>3100</Port>
    </Bridge>
      . . .
</BridgeSlapper>
```

If you plan to run multiple copies of Bridge Slapper, each instance must be assigned a unique JMS `ClientId`.

Device Template Configuration

A FigDevice element defines a template for creating device simulations within BridgeSlapper. Each element must have a count attribute that defines the number of simulations that will be created for this template. Optionally, you can also specify a `startDelaySeconds` attribute. See below for description of how the `startDelaySeconds` is used by each device simulation.

The FigDevice accepts one of two possible sub elements:

> `<FixedReport…>` defines a simulation that sends in reports containing a fixed number of readings (IO Points)
>
> or
>
> `<RandomReport…>` defines a simulation that sends in reports containing a random number of readings (IO Points)

Both of these elements have a required attribute that specifies the type of interval to the use. It can be either “`fixed`”, meaning the interval between each report will be the same; or “`random`” where reports are generated randomly.

Random report generation requires to additional elements that specify the min and max number of IO points to add to a report. Here are some examples:

```
<FigDevice count="5">
    <FixedReport intervalType="fixed">
        <IntervalSeconds>30</IntervalSeconds>
        <NumberOfReadings>20</NumberOfReadings>
</FixedReport
```

Specifies a temple for 5 device simulations that will generate reports containing 20 readings per report. The reports will be generated and sent in at 30 second intervals. The interval timer will start counting when the simulation is started so the first report will be generated 30 seconds after the start of the simulation.

```
<FigDevice count="10">
    <FixedReport intervalType="fixed" startDelaySeconds=”30”>
        <IntervalSeconds>30</IntervalSeconds>
        <NumberOfReadings>20</NumberOfReadings>
    </FixedReport>
</FigDevice>
```

Specifies a temple for 10 device simulations that will generate reports containing 20 readings per report. The reports will be generated and sent in at 30 second intervals. The interval timer will start counting when the simulation is started. However, because the startDelaySeconds attribute is specified, the interval timer will not start until 30 seconds after the start of the simulation. That means once the simulation is started, the first report will be sent 60 seconds after the start of the simulation.

```
<FigDevice count="10" startDelaySeconds=”30”>
    <FixedReport intervalType="random">
        <IntervalSeconds>30</IntervalSeconds>
        <NumberOfReadings>20</NumberOfReadings>
    </FixedReport>
</FigDevice>
```

Specifies a template for 10 device simulations that will generate reports containing 20 readings per report. Since the interval type is “random”, reports will be sent at random intervals every 0-30 seconds. However, because the startDelaySeconds attribute is specified, the interval timer will not start until 30 seconds after the start of the simulation.

```
 <FigDevice count="5”>
    <RandomReport intervalType="fixed">
        <IntervalSeconds>60</IntervalSeconds>
        <MinNumberOfReadings>5</MinNumberOfReadings>
        <MaxNumberOfReadings>20</MaxNumberOfReadings>
    </RandomReport>
</FigDevice>
```

Specifies a template for 5 devices that will generate reports containing between 5 and 20 readings per report every 60 seconds.

```
<FigDevice count="5”>
    <RandomReport intervalType="fixed" startDelaySeconds=”120”>
        <IntervalSeconds>60</IntervalSeconds>
        <MinNumberOfReadings>5</MinNumberOfReadings>
```

```
            <MaxNumberOfReadings>20</MaxNumberOfReadings>
        </RandomReport>
    </FigDevice>
```

Specifies a template for 5 devices that will generate reports containing between 5 and 20 readings per report every 60 seconds. Since the interval type is “random”, reports will be sent at random intervals every 0-30 seconds. However, because the startDelaySeconds attribute is specified, the interval timer will not start until 120 seconds after the start of the simulation.