DataBot TODO

**Next Steps:**

* **Environment Variable Support:** go to YamlConfigurationFileTest. insureEnvironmentVariablesAreResolved() and build the context and the implementation so the test can pass.
* **IMPORTANT on PUBLIC RELEASE**: based the latest release on the wrong util – qualify all dependencies for NU 4.7.12.

Implement the latest bits to get access to environment variables from the YAML config file.

* Explore <https://prometheus.io/>
* <https://www.influxdata.com>
* <https://graphiteapp.org>
* <https://kb.novaordis.com/index.php/Grafana> or <https://kb.novaordis.com/index.php/Kibana>
* <https://blog.outlyer.com/top10-open-source-time-series-databases>

# Current

* HornetQ ObjectName Issue. I attempted to read JMS Destination metrics from JBoss/HornetQ over JMX and I used the following (no quotes):

metrics:

- ${active-hq}/org.hornetq:module=JMS,name=incoming,type=Queue/MessageCount

It did not work. However, when I use the value returned verbatim by cljmx, it worked:

metrics:

- ${active-hq}/org.hornetq:module=JMS,name="incoming",type=Queue/MessageCount

Investigate and clarify why. Shouldn’t ObjectName key/value pairs be impervious to “”?

* **JDG7 failure**. 10/15/16 Does not work with JDG 7:

@Timer-0 07:14:05,153 WARN [DataCollectionTimerTask] failed to collect data: org.jboss.as.cli.CommandContextFactory.newCommandContext(Ljava/lang/String;ILjava/lang/String;[CZZI)Lorg/jboss/as/cli/CommandContext;

* (?) Longs do not display correctly because the field is interpreted as Map.
* (...) **-f malfunctions**. -f should also send the results to stdout and avoid creating the CSV file. Currently it does not.
* **Sensitivity to $JBOSS\_HOME.** If using JBoss controllers as metric sources, databot is sensitive to the value of JBOSS\_HOME. Investigate why is that and fix.
* **Configuration File Classpath Support**
  + **Environment Variable Support in configuration file Classpath.** Currently the wrapper script does not resolve environment variables.
  + **Eliminate Redundant Classpath Elements.** Currently the wrapper script will add to classpath a JAR every time it sees is, even if it seen it before. Filter out the JARs that already exist.
* **Exiting after one execution.** DataBot does not exit after just one execution. Investigate the state of the threads and figure out who keeps it running. This is after making the stream consumer threads daemons. To replicate: run a data collection sweep with “sampling.interval: 0”.
* **Reduce JBoss CLI client console verbosity**. Currently the JBoss CLI client sends this to stdout. Make it not to, or at least to send it to file only:

WARN: can't find jboss-cli.xml. Using default configuration values.

INFO xnio: XNIO Version 3.0.16.GA-redhat-1

INFO nio: XNIO NIO Implementation Version 3.0.16.GA-redhat-1

INFO remoting: JBoss Remoting version 3.3.9.Final-redhat-1

* **OS Metrics from JVM**. Some useful metrics (OpenFileDescriptorCount, MaxFileDescriptorCount, etc.) can be obtained from the JVM. Test, document in NOKB and use them:

conf:

- include:

domain: java.lang

bean:

- java.lang:type=OperatingSystem

- java.lang:type=Threading

attribute:

OpenFileDescriptorCount:

alias: jmx.OpenFileDescriptorCount

metric\_type: counter

MaxFileDescriptorCount:

alias: jmx.MaxFileDescriptorCount

metric\_type: counter

TotalStartedThreadCount:

alias: jmx.TotalStartedThreadCount

metric\_type: counter

PeakThreadCount:

alias: jmx.PeakThreadCount

metric\_type: counter

* **Network Statistics.** Explore using [[iftop]] to gather up/down bandwidth usage for each host:port pair.
* For four CLI metrics (2 for one instance and 2 for another, I get 4 cli.xml warnings. Normally I should get only 2 because I should share the clients.

[fsgapp@ln001xsjdg0003 etc]$ WARN: can't find jboss-cli.xml. Using default configuration values.

@Timer-0 17:28:55,620 INFO [xnio] XNIO Version 3.0.14.GA-redhat-1

@Timer-0 17:28:55,623 INFO [nio] XNIO NIO Implementation Version 3.0.14.GA-redhat-1

@Timer-0 17:28:55,634 INFO [remoting] JBoss Remoting version 3.3.5.Final-redhat-1

WARN: can't find jboss-cli.xml. Using default configuration values.

WARN: can't find jboss-cli.xml. Using default configuration values.

WARN: can't find jboss-cli.xml. Using default configuration values.

* Disconnect the JBossCliMetricSource instances on databot shutdown.
* <span id="n3jHG32">'''Support for password externalization'''. Then update [[Os-stats\_Metric\_Reference#JBoss\_CLI\_Metrics]] with details.</span>
* Warn if the collection time takes so long that it interferes with sampling interval length. Tests.
* Implement <tt>databot --help metrics</tt> that introspects and displays the list of built-in metrics.

# When Publicly Released

**Yaml configuration errors must carry line number.** Yaml configuration errors must carry line number (and possibly position in line).

* **Low Priority Defect**: declaring a JMX metric in-line as follows:

-jmx://localhost:9999/java.lang:type=Threading/ThreadCount

without declaring the associated metric source with a “classpath:” element that gives to JBoss-specific implementation makes the generic JMX metric source start, but run into trouble because the proper jboss-specific protocol is not used. This is confusing, make if fail more clearly:

@DataBot Timer Thread 15:24:31,311 ERROR [DataCollectionTask] source jmx://localhost:9999 collection failed:

io.novaordis.events.api.metric.MetricSourceException: io.novaordis.jmx.JmxException: failed to create the underlying JMX remoting connector

at io.novaordis.events.api.metric.jmx.JmxBus.start(JmxBus.java:128)

at io.novaordis.events.api.metric.MetricSourceBase.collectMetrics(MetricSourceBase.java:91)

at io.novaordis.databot.task.SourceQueryTask.call(SourceQueryTask.java:110)

at io.novaordis.databot.task.SourceQueryTask.call(SourceQueryTask.java:38)

at java.util.concurrent.FutureTask.run(FutureTask.java:266)

at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)

at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)

at java.lang.Thread.run(Thread.java:745)

Caused by: io.novaordis.jmx.JmxException: failed to create the underlying JMX remoting connector

at io.novaordis.jmx.JmxClientImpl.connect(JmxClientImpl.java:176)

at io.novaordis.events.api.metric.jmx.JmxBus.start(JmxBus.java:122)

... 7 more

Caused by: java.net.MalformedURLException: Unsupported protocol: jmxmp

at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359)

at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269)

at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:229)

at io.novaordis.jmx.JmxClientImpl.connect(JmxClientImpl.java:163)

... 8 more

The same defect was recorded previously as “If I specify a JMX metric without specifying the corresponding source, the JMX custom library is missing from the classpath and I get the exception below. Solutions: 1) better error message that suggests what’s missing AND 2) Introduce configuration file variables so I can define the source with the classpath and use a variable to refer to it from the metric definition.”

* Add a “max executions” configuration element. The internal machinery is already implemented.
* **events-agent**. Investigate projects/events-agent and consolidate with DataBot, if possible. Get rid of events-agent.