



AP/KAF 610.003.1

AutoPilot[®] M6 Plug-in for Kafka

Version 6.1.0

Installation and User's Guide

CONFIDENTIALITY STATEMENT: THE INFORMATION WITHIN THIS MEDIA IS PROPRIETARY IN NATURE AND IS THE SOLE PROPERTY OF NASTEL TECHNOLOGIES, INC. ALL PRODUCTS AND INFORMATION DEVELOPED BY NASTEL ARE INTENDED FOR LIMITED DISTRIBUTION TO AUTHORIZED NASTEL EMPLOYEES, LICENSED CLIENTS, AND AUTHORIZED USERS. THIS INFORMATION (INCLUDING SOFTWARE, ELECTRONIC AND PRINTED MEDIA) IS NOT TO BE COPIED OR DISTRIBUTED IN ANY FORM WITHOUT THE EXPRESSED WRITTEN PERMISSION FROM NASTEL TECHNOLOGIES, INC.

© 2018–2022 Nastel Technologies, Inc. All rights reserved.

PUBLISHED BY:

RESEARCH & DEVELOPMENT
NASTEL TECHNOLOGIES, INC.
88 SUNNYSIDE BLVD, SUITE 101
PLAINVIEW, NY 11803

COPYRIGHT © 2018–2022. ALL RIGHTS RESERVED. NO PART OF THE CONTENTS OF THIS DOCUMENT MAY BE PRODUCED OR TRANSMITTED IN ANY FORM, OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF NASTEL TECHNOLOGIES.

DOCUMENT TITLE: **NASTEL AUTOPILOT® M6 PLUG-IN FOR KAFKA INSTALLATION AND USER'S GUIDE**

VERSION: **6.1.0**

DOCUMENT RELEASE DATE: **MAY 2022**

NASTEL DOCUMENT NUMBER: **AP/KAF 610.003.1**

CONFIDENTIALITY STATEMENT: THE INFORMATION WITHIN THIS MEDIA IS PROPRIETARY IN NATURE AND IS THE SOLE PROPERTY OF NASTEL TECHNOLOGIES, INC. ALL PRODUCTS AND INFORMATION DEVELOPED BY NASTEL ARE INTENDED FOR LIMITED DISTRIBUTION TO AUTHORIZED NASTEL EMPLOYEES, LICENSED CLIENTS, AND AUTHORIZED USERS. THIS INFORMATION (INCLUDING SOFTWARE, ELECTRONIC AND PRINTED MEDIA) IS NOT TO BE COPIED OR DISTRIBUTED IN ANY FORM WITHOUT THE EXPRESSED WRITTEN PERMISSION FROM NASTEL TECHNOLOGIES, INC.

ACKNOWLEDGEMENTS:

THE FOLLOWING TERMS ARE TRADEMARKS OF NASTEL TECHNOLOGIES CORPORATION IN THE UNITED STATES OR OTHER COUNTRIES OR BOTH: TRANSACTIONWORKS, M6 AUTOPILOT, AUTOPILOT/IT, AUTOPILOT/ENTERPRISE, M6 FOR WMQ, AUTOPILOT/WMQ, M6 WEB SERVER, M6 WEB CONSOLE, AUTOPILOT/WEB, MQCONTROL, MQCONTROL EXPRESS, AUTOPILOT/TRANSACTION ANALYZER, AUTOPILOT/WAS, AUTOPILOT/TRANSACTION MONITOR, AUTOPILOT/OS MONITOR.

THE FOLLOWING TERMS ARE TRADEMARKS OF THE IBM CORPORATION IN THE UNITED STATES OR OTHER COUNTRIES OR BOTH: IBM, MQ, MQSERIES, WEBSPIHERE, WEBSPIHERE MQ WIN-OS/2, AS/400, OS/2, DB2, AND AIX, z/OS.

THE FOLLOWING TERMS ARE TRADEMARKS OF HEWLETT-PACKARD IN THE UNITED STATES OR OTHER COUNTRIES OR BOTH: OPENVIEW, HP-UX.

COMPAQ, THE COMPAQ LOGO, ALPHASERVER, COMPAQ INSIGHT MANAGER, CDA, DEC, DECNET, TRUCLUSTER, ULTRIX, AND VAX REGISTERED IN U.S. PATENT AND TRADEMARK OFFICE. ALPHA AND TRU64 ARE TRADEMARKS OF COMPAQ INFORMATION TECHNOLOGIES GROUP, L.P. IN THE UNITED STATES AND OTHER COUNTRIES.

SNMPC, SNMPC, WORKGROUP, AND SNMPC ENTERPRISE ARE TRADEMARKS OF CASTLE ROCK COMPUTING IN THE UNITED STATES OR OTHER COUNTRIES, OR BOTH.

SUN, SUN MICROSYSTEMS, THE SUN LOGO, IFORCE, JAVA, NETRA, N1, SOLARIS, SUN FIRE, SUN RAY, SUNSPECTRUM, SUN STOREDGE, SUNTONE, THE NETWORK IS THE COMPUTER, ALL TRADEMARKS AND LOGOS THAT CONTAIN SUN, SOLARIS, OR JAVA, AND CERTAIN OTHER TRADEMARKS AND LOGOS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF ORACLE CORPORATION AND/OR ITS AFFILIATES.

INSTALLANYWHERE IS A REGISTERED TRADEMARK OF ZEROG SOFTWARE IN THE UNITED STATES OR OTHER COUNTRIES, OR BOTH.

THIS PRODUCT INCLUDES SOFTWARE DEVELOPED BY THE APACHE SOFTWARE FOUNDATION ([HTTP://WWW.APACHE.ORG/](http://www.apache.org/)). THE "JAKARTA PROJECT" AND "TOMCAT" AND THE ASSOCIATED LOGOS ARE REGISTERED TRADEMARKS OF THE APACHE SOFTWARE FOUNDATION

INTEL, PENTIUM AND INTEL486 ARE TRADEMARKS OR REGISTERED TRADEMARKS OF INTEL CORPORATION IN THE UNITED STATES, OR OTHER COUNTRIES, OR BOTH

MICROSOFT, WINDOWS, WINDOWS NT, WINDOWS XP, .NET, .NET FRAMEWORK AND THE WINDOWS LOGOS ARE REGISTERED TRADEMARKS OF THE MICROSOFT CORPORATION.

UNIX IS A REGISTERED TRADEMARK IN THE UNITED STATES AND OTHER COUNTRIES LICENSED EXCLUSIVELY THROUGH X/OPEN COMPANY LIMITED.

"LINUX" AND THE LINUX LOGOS ARE REGISTERED TRADEMARKS OF LINUS TORVALDS, THE ORIGINAL AUTHOR OF THE LINUX KERNEL. ALL OTHER TITLES, APPLICATIONS, PRODUCTS, AND SO FORTH ARE COPYRIGHTED AND/OR TRADEMARKED BY THEIR RESPECTIVE AUTHORS.

SCO CUSA, SCO DOCTOR, SCO DOCTOR FOR NETWORKS, SCO DOCTOR LITE, SCO GLOBAL ACCESS, SCO MPX, SCO MULTIVIEW, SCO NIHONGO OPENSERVICES, SCO OK, THE SCO OK LOGO, SCO OPENSERVICES, SCO OPEN SERVER, SCO PORTFOLIO, SCO POS SYSTEM, SCO TOOLWARE, AND THE WORLD NEVER STOPS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF CALDERA INTERNATIONAL, INC. IN THE U.S.A. AND OTHER COUNTRIES, ALL RIGHTS RESERVED.

ORACLE® IS A REGISTERED TRADEMARK OF ORACLE CORPORATION AND/OR ITS AFFILIATES

OTHER COMPANY, PRODUCT, AND SERVICE NAMES, MAY BE TRADEMARKS OR SERVICE MARKS OF OTHERS.

Table of Contents

CHAPTER 1: INTRODUCTION	1
1.1 HOW THIS GUIDE IS ORGANIZED	1
1.2 HISTORY OF THIS DOCUMENT	1
1.2.1 <i>User Feedback</i>	1
1.3 RELATED DOCUMENTS	1
1.4 INTENDED AUDIENCE	2
1.5 SYSTEM REQUIREMENTS.....	2
1.5.1 <i>Platforms</i>	2
1.5.2 <i>Other Requirements</i>	2
1.6 TECHNICAL SUPPORT.....	2
1.7 CONVENTIONS	2
CHAPTER 2: ABOUT AUTOPILOT/KAFKA EXPERT	3
2.1 INTRODUCTION	3
CHAPTER 3: INSTALLATION & CONFIGURATION	5
3.1 INSTALLATION PREPARATION	5
3.1.1 <i>Installation Materials</i>	5
3.1.2 <i>Licensing Information</i>	5
3.2 INSTALLATION	5
3.2.1 <i>Before Installing AP/Kafka</i>	5
3.2.2 <i>AutoPilot Open Source Messaging Installation</i>	5
3.2.3 <i>Installing the Kafka Expert from the Nastel Delivery Center</i>	6
3.3 CONFIGURATION: USING STREAM-JMX FOR APACHE KAFKA MONITORING	7
3.3.1 <i>General Stream-JMX Configuration</i>	7
3.3.1.1 System Properties Used.....	8
3.3.1.2 Program Arguments Used	8
3.3.1.3 JMX Sampling Agent (Sampler) Options	8
3.3.1.4 Configure tnt4j.properties	9
3.3.2 <i>Connection Stream-JMX to Local or Remote Kafka JMX Service</i>	10
3.3.2.1 To connect to local JVM process	11
3.3.2.2 To Connect to JMX Service Over URL	12
3.3.2.3 Enable remote JMX connection over RMI.....	13
CHAPTER 4: AUTOPILOT INTEGRATION	15
CHAPTER 5: AP/KAFKA EXPERT METRICS	25
5.1 KAFKA SERVERS.....	26
5.2 JAVA.LANG	27
5.3 KAFKA.CLUSTER	28
5.4 KAFKA.NETWORK.....	29
5.5 KAFKA.LOG.....	30
CHAPTER 6: AP/KAFKA SAMPLE POLICIES	31
6.1 KAFKA DASHBOARD.....	31

6.1	KAFKA RESOURCES	32
APPENDIX A: REFERENCES		33
A.1	NASTEL DOCUMENTATION.....	33
APPENDIX B: CONVENTIONS		35
B.1	TYPOGRAPHICAL CONVENTIONS.....	35
B.2	NAMING CONVENTIONS	35

Figures

FIGURE 3-1. DETAIL OF INSTALLED LIBRARY LIST.....	6
FIGURE 3-2. INSTALLED CONFIGURATION	6
FIGURE 4-1. MODIFY KAFKA MONITOR	15
FIGURE 4-2. CREATE KAFKA MONITOR – GENERAL TAB	16
FIGURE 4-3. CREATE KAFKA MONITOR – FACT OPTIONS TAB	17
FIGURE 4-4. CREATE KAFKA MONITOR – LOGGING TAB.....	18
FIGURE 4-5. CREATE KAFKA MONITOR – RECORDING TAB.....	19
FIGURE 4-6. CREATE KAFKA MONITOR – RESTART-RECOVERY TAB.....	20
FIGURE 4-7. CREATE KAFKA MONITOR – SECURITY TAB.....	21
FIGURE 4-8. CREATE KAFKA MONITOR – STREAMING OPTIONS TAB.....	22
FIGURE 4-9. CREATE KAFKA MONITOR – TCP OPTIONS TAB.....	23
FIGURE 4-10. CREATE KAFKA MONITOR – UDP OPTIONS TAB.....	23
FIGURE 5-1. KAFKA EXPERT FACTS	25
FIGURE 5-2. SERVER METRICS	26
FIGURE 5-3. JAVA.LANG	27
FIGURE 5-4. KAFKA.CLUSTER	28
FIGURE 5-5. KAFKA.NETWORK	29
FIGURE 5-6. KAFKA.LOG.....	30
FIGURE 6-1. KAFKA DASHBOARD.....	31
FIGURE 6-2. KAFKA RESOURCES	32

Tables

TABLE 1-1. DOCUMENT HISTORY	1
TABLE 4-1. KAFKA MONITOR – GENERAL PROPERTIES	16
TABLE 4-2. KAFKA MONITOR – FACT OPTIONS PROPERTIES	17
TABLE 4-3. KAFKA MONITOR – LOGGING PROPERTIES	18
TABLE 4-4. KAFKA MONITOR – RECORDING PROPERTIES	19
TABLE 4-5. KAFKA MONITOR – RESTART-RECOVERY PROPERTIES	20
TABLE 4-6. KAFKA MONITOR – SECURITY PROPERTIES	21
TABLE 4-7. KAFKA MONITOR – STREAMING OPTIONS PROPERTIES	22
TABLE A-1. NASTEL DOCUMENTATION	33
TABLE B-1. TYPOGRAPHICAL CONVENTIONS	35
TABLE B-2. AUTOPILOT RELATED NAMING CONVENTIONS	35

Chapter 1: Introduction

Welcome to the *Nastel AutoPilot® M6 Plug-in for Kafka Installation and User's Guide*. The expert is compatible with Apache Kafka. This guide describes installation and use of the Kafka expert. This plug-in is designed to work with AutoPilot, its components, and other plug-ins, and run simultaneously without interference or performance degradation.

1.1 How This Guide is Organized

- [Chapter 1:](#) Identifies the users and history of the document. System requirements for this plug-in are outlined. All other system and platform information is listed in the AutoPilot Installation and User's Guides.
- [Chapter 2:](#) Contains a brief description of AP/Kafka Expert.
- [Chapter 3:](#) Provides instruction for new installations of the AP/Kafka Expert.
- [Chapter 4:](#) Provides instruction for using the AP/Kafka Expert.
- [Chapter 5:](#) Describes some of the AP/Kafka Expert metrics collected by the expert coming from the Kafka.
- [Chapter 6:](#) Provides examples of sample policies.
- [Appendix A:](#) Provides a detailed list of all reference information required for the installation of AutoPilot.
- [Appendix B:](#) Contains conventions used in AutoPilot and documents typographical conventions.

1.2 History of This Document

Release Date:	Document Number	For AutoPilot Version	Summary
February 2018	AP/KAF 610.001	AP 6.0 or higher	Original issue
August 2018	AP/KAF 610.002	AP 6.0 or higher	General update
January 2021	AP/KAF 610.003	AP 6.0 or higher	Updates to 3.3.1, 3.3.1.4, 3.3.2 and 3.3.2.3.
May 2022	AP/KAF 610.003.1	AP 6.0 or higher	Changed title to <i>AutoPilot® M6 Plug-in for Kafka Installation and User's Guide</i>

1.2.1 User Feedback

Nastel encourages all Users and Administrators of AutoPilot to submit comments, suggestions, corrections and recommendations for improvement for all AutoPilot documentation. Please send your comments via email to: support@nastel.com. You will receive a written response, along with the status of any proposed change, update, or correction.

1.3 Related Documents

The complete listing of related and referenced documents is listed in [Appendix A](#) of this guide.

1.4 Intended Audience

The AP/KAFKA Guide is intended for use by installers and administrators of Nastel's AutoPilot with Apache Kafka and related components.

1.5 System Requirements

This section defines system and platform prerequisite support requirements for AP/Kafka.

1.5.1 Platforms

AP/KAFKA expert is compatible with the following platforms:

- Windows NT/2000 or later/XP
- Unix (Solaris, AIX, HP-UX, Linux)

1.5.2 Other Requirements

AP/Kafka requires the following conditions:

- AutoPilot 6.0 or higher
- Java Run Time Environment 1.7.x (JRE 1.6.x) or higher
- Apache Kafka
- Target operating system environment
- Installer may need administrative privileges for the target platform
- Since communication between Kafka and AutoPilot is done via JMX it is necessary to have a proper installed configuration for operation of the expert. (Refer to [section 3.3](#).)

1.6 Technical Support

If you need additional technical support, you can contact Nastel Technologies by telephone or by e-mail. To contact Nastel technical support by telephone, call **800-963-9822 ext. 1**, if you are calling from outside the United States dial **001-516-801-2100**. To contact Nastel technical support by email, send a message to support@nastel.com. To access the Nastel automated support system, go to <http://support.nastel.com/>. (A user name and password are required.) Contact your local AutoPilot Administrator for further information.

1.7 Conventions

Refer to [Appendix B](#) for conventions used in this guide.

Chapter 2: About AutoPilot/Kafka Expert

2.1 Introduction

The AP/Kafka Expert was designed to monitor and manage your Apache Kafka and related components. This information is processed by the AP/Kafka Expert and integrated into the AutoPilot infrastructure. Communication with the Kafka Broker is via JMX either locally or remotely across a network.

Information includes:

- JVM Metrics
- Broker Metrics
- Topic Information
- Producers Details
- Consumer Details
- Zookeeper Metrics.

This information can be combined with information provided by AutoPilot for other components, such as the operating system or log files to get further insight into the performance and operation of the Apache Kafka environment.

This Page Intentionally Left Blank

Chapter 3: Installation & Configuration

3.1 Installation Preparation

This section contains general information related to preparing for and installing the AP/Kafka Expert software.

3.1.1 Installation Materials

Installation can be performed from installation media or by download through the Nastel Resource Center.

Prior to installation, review all text files and installation procedures provided on the Nastel Resource Center. It is recommended that all installation related materials are printed to allow the installer to review them prior to installation, and better to follow the detailed instructions within.

3.1.2 Licensing Information

A copy of the standard Licensing Agreement is imbedded in the installation software and is provided on the Resource Center. The formal licensing agreement has been furnished in the purchase agreement package.

3.2 Installation

This section provides instructions for installing AP/Kafka Expert on the compatible platforms. Review all installation-related materials prior to commencing installation procedures. Reviewing materials will allow installers to pre-determine installation options and familiarize themselves with associated requirements.

3.2.1 Before Installing AP/Kafka

The Kafka expert uses the open source messaging installation package (AP_OPMSMSG).

If you have previously installed this package, it is not necessary to repeat this installation and you can proceed to configuring the Kafka server ([section 3.3](#)).

If you have not previously installed the expert, install it using package manager similar to the instructions below. Then transfer the open source message components to the server(s) where Kafka is to be managed from, if different than the AutoPilot server. Additional steps on configuring these services specific to Kafka usage is outlined below.

3.2.2 AutoPilot Open Source Messaging Installation

Nastel AutoPilot Open Source Messaging supports Apache Kafka and other open source messaging services.

Following installation of AutoPilot Open Source Messaging (the package AP_OPMSMSG), the following files are placed in the AutoPilot Home directory (e.g., /opt/nastel/AutoPilotM6):

- AP_OSMsg_readme.txt
- tnt4j-stream-jmx-version.tar.gz
- tnt4j-stream-jmx-version.zip


Use zip or gunzip to extract tnt4j-stream-jmx to a folder on your system (e.g., /opt/nastel/AutoPilotOPMSMSG).

Review the README.md file for general configuration instructions.


Review [section 3.3](#), Using Stream-JMX for Apache Kafka Monitoring, for specific instructions for configuring tnt4j-stream-jmx for Apache Kafka.

3.2.3 Installing the Kafka Expert from the Nastel Delivery Center

1. Download the AP/Kafka Expert Plug-in from the Nastel Delivery Center (<http://data.nastel.com/ap/>). A user name and password are required.
2. Save your work and logoff AutoPilot.

	NOTE:	There are no specific logoff procedures required to exit AutoPilot Console.
---	--------------	---

3. Stop the Nodes and/or Domain Servers that will be updated as specified in the AutoPilot User's Guide.
4. Copy the open source messaging installation package `AP_KAFKA_version.pkg` into the `[AUTOPILOT_HOME]\updates` directory.
5. At the command prompt run:
`[AUTOPILOT_HOME]\bin\pkgman ..\updates\AP_KAFKA-version.pkg`
6. Verify the plug-in installation by running:
`[AUTOPILOT_HOME]\bin\pkgman -info`
 Verify `AP_Kafka-Plugin (AP_Kafka-<version_number>.pkg)` is listed.

	NOTE:	Make sure there are no errors posted at the bottom of the screen.
---	--------------	---

Package	Version	Size(KB)	Time
AutoPilot M6(NA)	6.0	NA	2018-02-12 21:52:36
JRE(NA)	1.7.0_60	NA	2017-06-12 15:23:04
AIM-Plugin(NA)	6.0.32	2548	2016-05-05 22:55:52
ServiceUpdate(AP60_SU26.pkg)	6.0.26	5958	2017-12-14 10:00:55
AP CAAPM RP(AP_CA_APM-6.21.pkg)	6.2.1	4809	2017-07-28 14:26:05
zExpert-Plugin(AP_ZXP-1.0.1.1.pkg)	1.0.1.1	25	2017-08-10 11:24:30
AP OpenSourceMsg(AP_OPMSG-1.0.0.pkg)	1.0.0	57084	2018-02-12 20:58:44
AP Kafka-Plugin(AP_Kafka-6.1.0.pkg)	6.1.0	90	2018-02-12 21:52:25

Figure 3-1. Detail of Installed Library List

7. Restart the AutoPilot services and login to the Enterprise Manager. The following expert and managers should be configured.

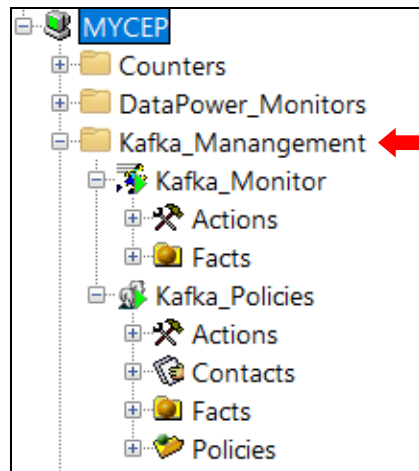


Figure 3-2. Installed Configuration

3.3 Configuration: Using Stream-JMX for Apache Kafka Monitoring

There are several deployments options available for monitoring Apache Kafka with Stream-JMX:

- Run Stream-JMX as a -javaagent
- Attach Stream-JMX as an agent to a running Apache Kafka service
- Connect Stream-JMX over JMXConnector to a local or remote Apache Kafka service.

The third method, using either a local or remote connection, is the only one covered in this document.

3.3.1 General Stream-JMX Configuration

Stream-JMX has configuration properties that allow you to configure the JMX sampler.

It is possible to configure the JMX sampler using System Properties or Program Arguments. Depending on the sampling environment used, it may be easier to configure the JMX sampler using one approach or another.

When both definitions are available, the System Property value is assigned first and then the Program Argument value.

JMX sampler configuration properties are:

- `trace` – flag indicating whether the sample listener should print trace entries to print stream. Default value = `false`
- `forceObjectName` – flag indicating to forcibly add `objectName` attribute if such is not present for a MBean. Default value = `false`
- `compositeDelimiter` – delimiter used to tokenize composite/tabular type MBean properties keys. Default value = `\`
- `useObjectNameProperties` – flag indicating to copy MBean `ObjectName` contained properties into sample snapshot properties.

To configure Stream-JMX using system properties, refer to System Properties Used ([section 3.3.1.1](#)).

To configure Stream-JMX using program arguments, refer to Program Arguments Used ([section 3.3.1.2](#)).

To configure `tnt4j.properties`, refer to the section, Configure `tnt4j.properties` ([section 3.3.1.4](#)).

3.3.1.1 System Properties Used

- `tnt4j.config` – defines TNT4J properties file path.
Example: `-Dtnt4j.config=". \config\tnt4j.properties"`
- `com.jkoolcloud.tnt4j.stream.jmx.agent.trace` – defines whether to dump trace data to application console output. Default value = `false`.
Example: `Dcom.jkoolcloud.tnt4j.stream.jmx.agent.trace=true`
- `com.jkoolcloud.tnt4j.stream.jmx.agent.forceObjectName` – defines whether to forcibly add `objectName` attribute if such is not present for MBean. Default value = `false`.
Example: `Dcom.jkoolcloud.tnt4j.stream.jmx.agent.forceObjectName=true`
- `com.jkoolcloud.tnt4j.stream.jmx.agent.compositeDelimiter` – defines delimiter used to tokenize composite/tabular type MBean properties keys. Default value = `\`.
Example: `Dcom.jkoolcloud.tnt4j.stream.jmx.agent.compositeDelimiter=.`
- `com.jkoolcloud.tnt4j.stream.jmx.agent.useObjectNameProperties` – defines whether to copy MBean `ObjectName` contained properties into sample snapshot properties. Default value = `true`.
Example: `Dcom.jkoolcloud.tnt4j.stream.jmx.agent.useObjectNameProperties=false`

3.3.1.2 Program Arguments Used

To define stream JMX sampler configuration property use program argument `-slp:`. One argument defines one property. To define multiple properties use as many argument definitions as there are required properties. For example:

```
-slp:trace=true
-slp:forceObjectName=true
-slp:compositeDelimiter=.
-slp:useObjectNameProperties=false
```

3.3.1.3 JMX Sampling Agent (Sampler) Options

Agent options are defined using format: `mbean-filter!exclude-filter!sample-ms!init-delay-ms`

- `mbean-filter` – MBean include name filter defined using object name pattern:
`domainName:keysSet`
- `exclude-filter` – MBean exclude name filter defined using object name pattern:
`domainName:keysSet`
- `sample-ms` – MBeans sampling rate in milliseconds
- `init-delay-ms` – MBeans sampling initial delay in milliseconds. Optional, by default it is equal to 'sample-ms' value

The Default sampling agent options value is: `*:*!!30000`

3.3.1.4 Configure tnt4j.properties

The following properties are required for stream-JMX to publish Kafka metrics to AutoPilot Kafka_Expert. These properties are part of tnt4j.properties. The stanza used for Stream-JMX sources are mandatory.

```
#####
;Stanza used for Stream-JMX sources
#####
{
  source: com.jkoolcloud.tnt4j.stream.jmx
    source.factory: com.jkoolcloud.tnt4j.stream.jmx.source.JMXSourceFactoryImpl
    source.factory.GEOADDR: New York
    source.factory.DATACENTER: HQDC
    source.factory.SERVICE: $sjmx.serviceId ← example1 for Kafka processID/service name
#
  source.factory.SERVICE:
@bean:org.apache.activemq:type=Broker,brokerName=localhost/?BrokerId ← example2 for Kafka
processID/service name
#
  source.factory.SERVER: @sjmx.serverName ← example1 for Kafka node/server name
#
  source.factory.SERVER: @sjmx.serverAddress ← example2 for Kafka node/server name

  source.factory.SERVER: @bean:java.lang:type=Runtime/?Name ← example3 for Kafka node/server name
; This RootFQN definition is for single VM monitoring
  source.factory.RootFQN: SERVICE=?#SERVER=?#DATACENTER=?
; This RootFQN definition is for multi VM monitoring: SERVICE token shall be defined next to VM
#source.factory.RootFQN: SERVICE=?#DATACENTER=?
  source.factory.RootSSN: tnt4j-stream-jmx
#####
; Event Sink configuration for streaming name=value pairs over socket
#####
  event.sink.factory.EventSinkFactory: com.jkoolcloud.tnt4j.sink.impl.SocketEventSinkFactory
;--- If socket sent data should no be logged anywhere else
  event.sink.factory.EventSinkFactory.LogSink: null
;--- If socket sent data should be logged to file
##event.sink.factory.EventSinkFactory.LogSink: file:./logs/tnt4j-stream-jmx_samples_socket.log
  event.sink.factory.EventSinkFactory.Host: localhost ← hostname of AutoPilot Kafka_Expert
  event.sink.factory.EventSinkFactory.Port: 6061 ← TCP port of AutoPilot Kafka_Expert
```

3.3.2 Connection Stream-JMX to Local or Remote Kafka JMX Service

The following OS shell script files are provided for connecting Stream-JMX to a local or remote Kafka JMX service:

- bin/stream-jmx-connect.bat or
- bin/stream-jmx-connect.sh

Windows

rem using URL

```
/bin/stream-jmx-connect.bat service:jmx:rmi:///jndi/rmi://localhost:9999/jmxrmi
```

rem using URL with connection parameters

```
/bin/stream-jmx-connect.bat service:jmx:rmi:///jndi/rmi://localhost:9999/jmxrmi -
ul:admin -up:admin -cp:java.naming.security.authentication=simple -
cp:java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory
```

rem using process name part

```
/bin/stream-jmx-connect.bat server.properties
```

rem using pid

```
/bin/stream-jmx-connect.bat 1553
```

Unix/Linux

using URL

```
./bin/stream-jmx-connect.sh service:jmx:rmi:///jndi/rmi://localhost:9999/jmxrmi
```

using URL with connection parameters


```
./bin/stream-jmx-connect.sh service:jmx:rmi:///jndi/rmi://localhost:9999/jmxrmi -
ul:admin -up:admin -cp:java.naming.security.authentication=simple -
cp:java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory
```

using process name part

```
./bin/stream-jmx-connect.sh server.properties
```

using pid

```
./bin/stream-jmx-connect.sh 1553
```

	<p>NOTE:</p>	<p>Kafka does not provide J2EE implementation, thus you need only stream-jmx core jar in classpath when sampling Kafka metrics over JMX. Executable OS shell run script files uses only core as MODULE_SET variable value</p>
---	---------------------	---

- Windows
 - set variable in batch script bin/stream-jmx-connect.bat
 - set MODULE_SET=core
- Unix/Linux
 - set variable in shell script bin/stream-jmx-connect.sh
 - export MODULE_SET=("core")

3.3.2.1 To connect to local JVM process

Command line to connect local JVM process JMX looks like this:

```
java -Dtnt4j.config=. \config\tnt4j.properties -
Dcom.jkoolcloud.tnt4j.stream.jmx.agent.trace=true -classpath "tnt4j-stream-jmx-core-
0.7-all.jar" com.jkoolcloud.tnt4j.stream.jmx.SamplingAgent -connect -
vm:server.properties -ao:*:*!10000
```

System properties `-Dxxxxx` defines Stream-JMX configuration. For details, refer to General Stream-JMX Configuration ([section 3.3.1](#)).

SamplingAgent arguments `-connect -vm:server.properties -ao:*:*!10000` states:

- `-connect` – defines that SamplingAgent shall connect to running JVM process over JMXConnector (RMI) connection.
- `-vm:server.properties` – is JVM descriptor. In this case it is running JVM name fragment `server.properties`. But it also may be JVM process identifier - PID. **Mandatory argument.**
- `-ao:*:*!10000` – is JMX sampler options stating to include all MBeans and schedule sampling every 10 seconds. Sampler options are optional. Default value = `*:*!30000`. Initial sampler delay can be configured by adding numeric parameter `*:*!30000!1000` defining initial sampler delay as 1 second. Default sampler delay value is equal to sampling period value.
- `-slp:` – any JMX sampler configuration property. Refer to Program Arguments Used ([section 3.3.1.2](#)) for details.

3.3.2.2 To Connect to JMX Service Over URL

Command line to connect remote JMX service looks like this:

```
java -Dtnt4j.config=.\\config\\tnt4j.properties -
Dcom.jkoolcloud.tnt4j.stream.jmx.agent.trace=true -classpath " tnt4j-stream-jmx-
core-0.7-all.jar" com.jkoolcloud.tnt4j.stream.jmx.SamplingAgent -connect -
vm:service:jmx:[JMX_URL] -ul:admin -up:admin -ao:*:*!10000 -cri:30 -
cp:java.naming.security.authentication=simple -
cp:java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory
```

System properties `-Dxxxxx` defines Stream-JMX configuration. For details, refer to General Stream-JMX Configuration ([section 3.3.1](#)).

```
SamplingAgent arguments -connect -vm:service:jmx:[JMX_URL] -ul:admin -up:admin -
ao:*:*!10000 -cri:30 -cp:java.naming.security.authentication=simple -
cp:java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory states:
```

- `-connect` – defines that `SamplingAgent` shall connect to running JMX service over JMXConnector (RMI) connection.
- `-vm:service:jmx:[JMX_URL]` – is JMX service URL to use for connection. **Mandatory argument.** Full URL may be like:
`service:jmx:rmi:///jndi/rmi://localhost:9999/jmxrmi`
- `-ul:admin` – is user login. In this case it is admin. User login argument is optional.
- `-up:admin` – is user password. In this case it is admin. User password argument is optional.
- `-ao:*:*!10000` – is JMX sampler options stating to include all MBeans and schedule sampling every 10 seconds. Sampler options are optional - default value is `*:*!30000`. Initial sampler delay can be configured by adding numeric parameter `*:*!30000!1000` defining initial sampler delay as 1 second. Default sampler delay value is equal to sampling period value.
- `-cp:java.naming.security.authentication=simple -`
`cp:java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory` - is JMX connector parameters definitions in properties format `key=value`. JMX connector parameters are optional and can be defined multiple times; as many as there are required JMX connector parameters. Refer to [Java API Context class documentation](#) for available properties naming.



NOTE:

If you are using some API extending JNDI, check documentation if it provides some additional connection configuration properties.

- `-cri:30` – is connection retry interval in seconds. In this case it is 30 seconds between connect retry attempts. Connection retry interval is optional. Default value = 10 sec. Special values are:
 - 0 indicates no delay between repeating connect attempts.
 - -1 indicates no repeating connect attempts shall be made at all and application has to stop on first failed attempt to connect.
- `-slp:` – any JMX sampler configuration property. Refer to Program Arguments Used ([section 3.3.1.2](#)) for details.

3.3.2.3 Enable remote JMX connection over RMI

- Kafka by default does not allow remote JMX connections over RMI. To enable it you need to:
 - Define it when running Kafka broker. Windows example:

```
set JMX_PORT=9999 bin/kafka-server-start.bat config/server-1.properties
set JMX_PORT=10000 bin/kafka-server-start.bat config/server-2.properties
```
 - Define it within run script kafka-server-start.sh. Unix/Linux example:
Add following options in the startup script for each Kafka server:


```
export KAFKA_JMX_OPTS="-Dcom.sun.management.jmxremote=true -
Dcom.sun.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.ssl=false
-Djava.rmi.server.hostname=KAFKA-SERVER1"
export JMX_PORT=9999

export KAFKA_JMX_OPTS="-Dcom.sun.management.jmxremote=true -
Dcom.sun.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.ssl=false
-Djava.rmi.server.hostname=KAFKA-SERVER2"
export JMX_PORT=9999
```

This Page Intentionally Left Blank

Chapter 4: AutoPilot Integration

As noted above, during installation, a Kafka monitoring node and a policy manager with a set of policies were configured automatically. As such, only minor configuration should be required. If needed, you can edit the monitor properties as follows:

1. Open the AutoPilot Console.
2. Click the **Deployment Tool**  to display Directory Viewer (if not already displayed).
3. Right-click **Kafka_Monitor** and select **Properties**.

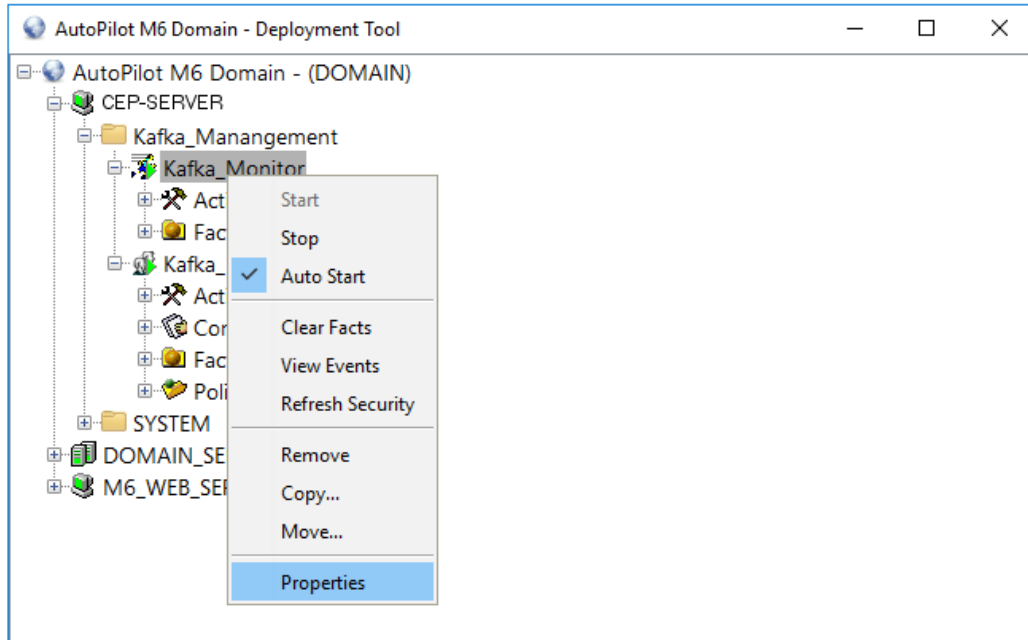


Figure 4-1. Modify Kafka Monitor

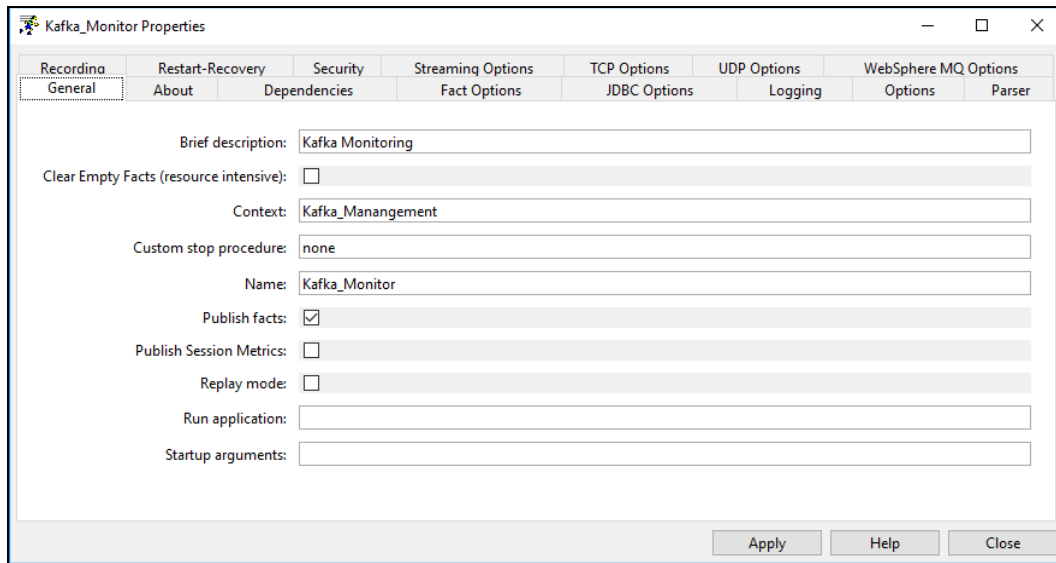


Figure 4-2. Create Kafka Monitor – General Tab

- On the **General** tab, the following fields could be changed as required. Other parameters are not applicable to Kafka monitoring.

Table 4-1. Kafka Monitor – General Properties

Property	Description
Brief description	Short description of the service.
Context	User-defined category that will be registered in the Domain Server. Context is displayed as folder icon under each Managed Node.
Name	Name that uniquely identifies the service in the Domain Server. Enter or modify the Service Name as required, or in accordance with local guidelines. Variations of names are used when deploying services on multiple Nodes. No spaces or blanks are recommended in Service Name formats. For example, KAFKA_Monitor.

- The following properties are available for the Kafka expert. Review (if updating existing expert) or configure data elements as follows.

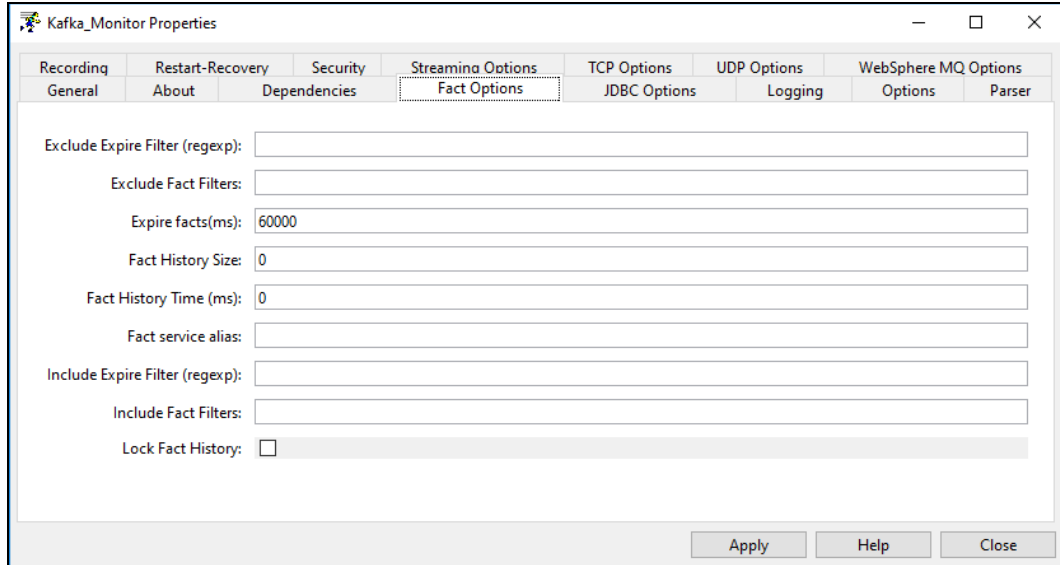


Figure 4-3. Create Kafka Monitor – Fact Options Tab

Table 4-2. Kafka Monitor – Fact Options Properties	
Property	Description
Exclude Expire Filter (regexp)	Facts that match the specified regular expression are not expired.
Exclude Fact Filters	Comma separated list of fact paths to exclude during publishing. For example: *SYSTEM*, *FactName*
Expire facts(ms)	User-defined time in which facts that have not been updated within a specific time automatically expire (in milliseconds). The default is 0, which means never expire. However, in most applications, 0 should not be used. In cases where certain data is no longer published, if 0 is used, these facts will never expire. It is recommended that this value be 50% larger than the sample rate.
Fact History Size	Automatically maintains the specified number of samples for each published fact in memory.
Fact History Time	Automatically maintain fact history not exceeding specified time in milliseconds.
Include Fact Filters	Comma separated list of fact paths to include during publishing. For example: *SYSTEM*, *FactName*
Fact service alias	If supported by the expert, specifies the alternative service name that the expert will publish its facts under.
Include Expire Filter (regexp)	Facts that match the specified regular expression are expired.
Lock Fact History	Enables/disables history collection after accumulating the first history batch up to Fact History Time or Fact History Size which ever limit is reached first. If disabled newer history samples replace older on a rolling basis.

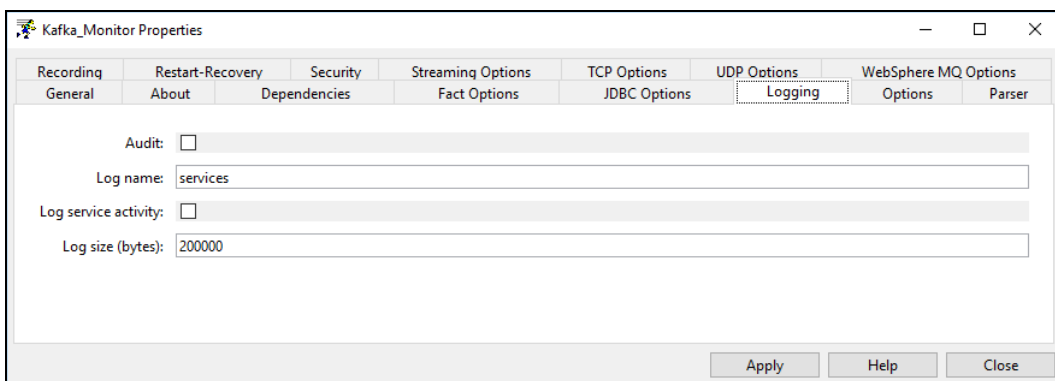


Figure 4-4. Create Kafka Monitor – Logging Tab

Table 4-3. Kafka Monitor – Logging Properties

Property	Description
Audit	Enable/Disable service audit trace. Default is disabled.
Log name	Log name associated with the service. The default name is Services, but may be changed as required.
Log service activity	Enable/Disable service activity trace. Default is disabled.
Log size (bytes)	Log size in bytes. Real log size is the maximum value of the server.log.size and logsize.

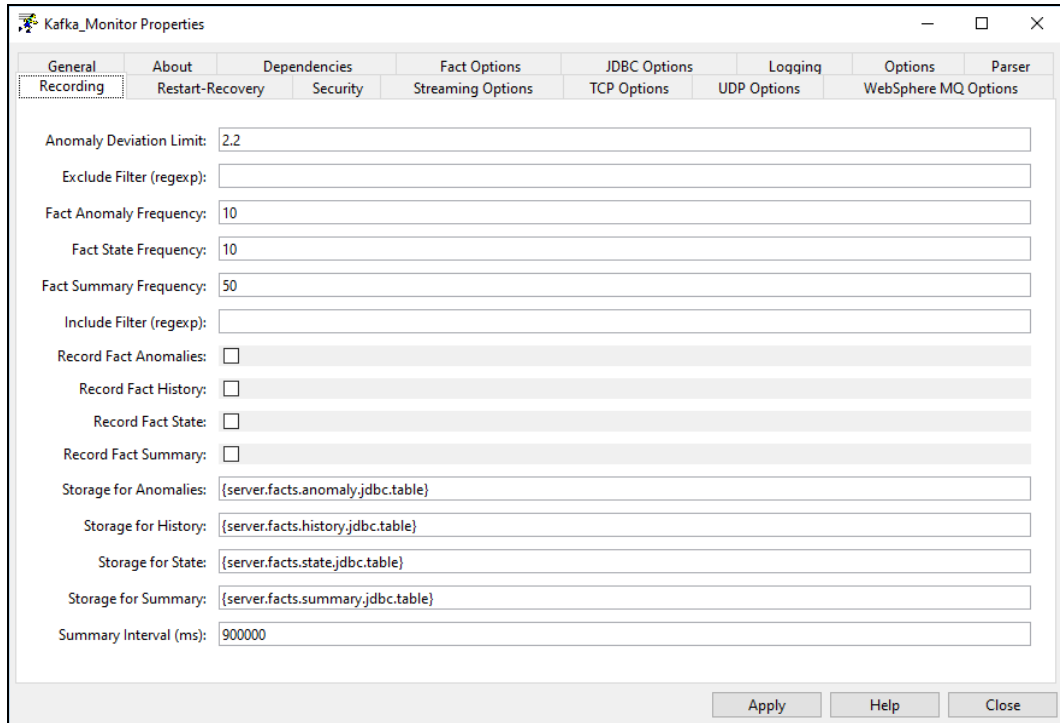


Figure 4-5. Create Kafka Monitor – Recording Tab

Table 4-4. Kafka Monitor – Recording Properties

Property	Description
Anomaly Deviation Limit	The number of standard deviations from the mean at which the value is considered an anomaly. For example, a value of 2.2 is 2.2 standard deviations. Requires fact recording to be configured (although not actually recording).
Exclude Filter (regexp)	A regular expression filter to exclude certain facts from being written to the database. Facts have the format <code>expert\class\instance\leaf=value</code> such as in the example <code>Servers\Linux\Serv7\processes=40</code> .
Fact Anomaly Frequency	The frequency of fact updates at which anomaly calculation is done. A value of 10 indicates every 10 th sample. A value of 1 would analyze every fact update to determine if it was an anomaly.
Fact State Frequency	If Record Fact State is enabled, the value entered here specifies how often the Fact State is updated.
Fact Summary Frequency	If Record Fact Summary is enabled, used to write an intermediate summary record every X th update to the fact during the Summary Interval. In this example, every 50 th update to the fact an intermediate summary record is recorded. This is done to avoid waiting 15 minutes for a summary record to appear in the summary table.
Include Filter (regexp)	A regular expression filter to include certain facts being written to the database. Same format as described for the exclude filter.
Record Fact Anomalies	If enabled, records every fact anomaly into the Anomaly database. The exclude/include filters are respected. Requires fact recording to be configured (although not actually recording).
Record Fact History	If enabled, records every fact change into the History database. The exclude/include filters are respected. To define database tables and set AutoPilot options, refer to <i>AutoPilot M6 User’s Guide</i> section 4.5.4.1.
Record Fact State	If enabled, records the last value published (current state) into the state database and restores that value when the CEP Server is stopped and restarted. The exclude/include filters are respected. To define database tables and set AutoPilot options, refer to <i>AutoPilot M6 User’s Guide</i> , section 4.5.4.1.

Table 4-4. Kafka Monitor – Recording Properties

Property	Description
Record Fact Summary	If enabled, records summary record at the interval designated in the Summary Interval (ms) field into the Summary database. The exclude/include filters are respected. To define database tables and set AutoPilot options, refer to <i>AutoPilot M6 User’s Guide</i> , section 4.5.4.1.
Storage for Anomalies	Database table where the Fact Anomalies data is stored.
Storage for History	Database table where the Fact History data is stored.
Storage for State	Database table where the Fact State data is stored.
Storage for Summary	Database table where the Fact Summary data is stored.
Summary Interval (ms)	If Record Fact Summary is enabled, designates the interval of time in ms for which baseline numbers for each numeric fact are computed. Summary Interval is only in affect when CEP instance is running in record mode (ATPNODE –record). Default 900000 is 15 minutes, which means maintain a baseline of statistics for each numeric fact for a period of 15 minutes and write a record to the database. At the end of interval fact statistics is reset and the baseline collection starts again.

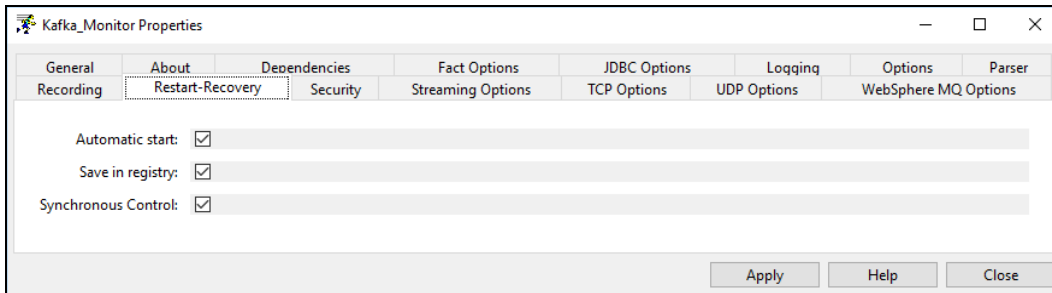


Figure 4-6. Create Kafka Monitor – Restart-Recovery Tab

Table 4-5. Kafka Monitor – Restart-Recovery Properties

Property	Description
Automatic start	Enable/disable automatic start.
Save in registry	Persistent services are saved in <code>Registry.xml</code> file. Default is enabled.
Synchronous Control	Enable/Disable synchronous service initiation.

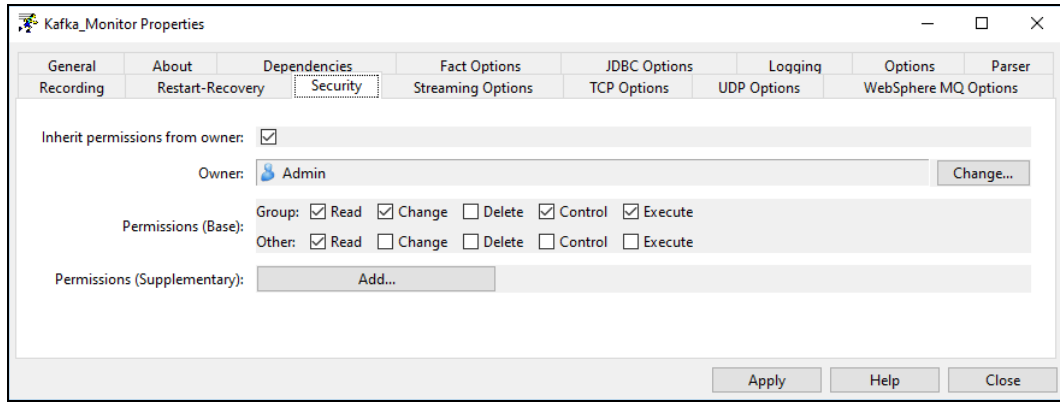


Figure 4-7. Create Kafka Monitor – Security Tab

Table 4-6. Kafka Monitor – Security Properties		
Property	Description	
Inherit permissions from owner	Enable/disable inherit permission from owner’s permission masks. Default is enabled.	
Owner	User that owns the object.	
Permissions	Permissions for users in the same group and users in other groups. Enable/disable as required.	
	Group:	Others:
Read	Group members may read/view attributes of an object.	Other users may read/view attributes of an object.
Change	Group members may change the attributes of an object.	Other users may change the attributes of an object.
Delete	Group members may delete the object.	Other users may delete the object.
Control	Group members may execute control actions such as start, stop, and disable.	Other users may execute control actions such as start, stop, and disable.
Execute	Group members may execute operational commands on the object.	Other users may execute operational commands on the object.

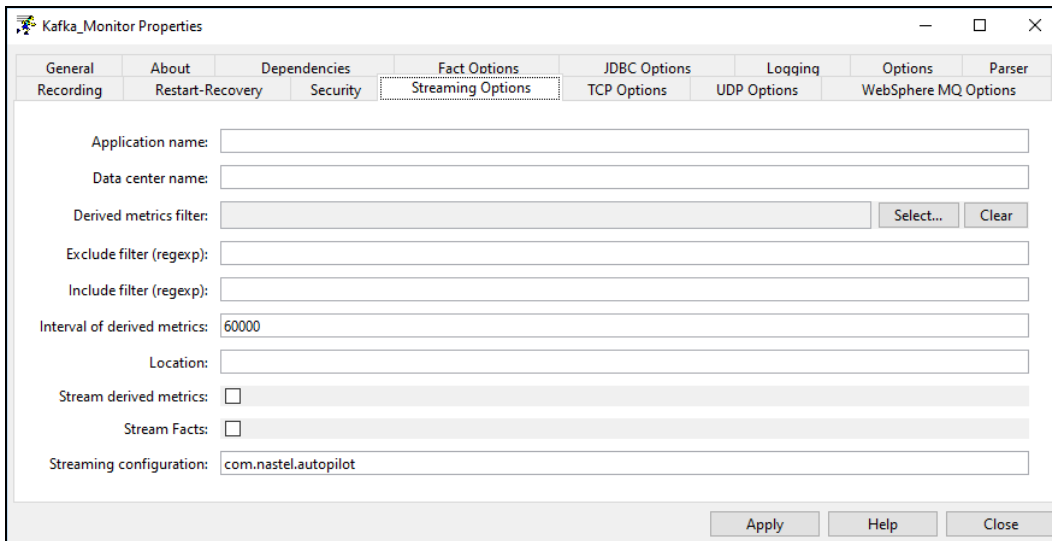


Figure 4-8. Create Kafka Monitor – Streaming Options Tab

Table 4-7. Kafka Monitor – Streaming Options Properties

Table 4-7. Kafka Monitor – Streaming Options Properties	
Application name	Sets application name if different from the default set in the tnt4j.properties file.
Data center name	Sets data center name if different from the default set in the tnt4j.properties file.
Derived metrics filter	Create or select filter.
Exclude filter (regex)	Ignore facts that match specified regular expression; that is, do not stream facts that match the regex.
Include filter (regex)	Only stream the facts that match specified regular expression.
Interval of derived metrics	Set interval.
Location	Sets server location if different from the default set in the tnt4j.properties file.
Stream derived metrics	Enable/disable derived metrics streaming.
Stream Facts	Enable/disable fact streaming (requires TNT4J streaming framework).
Streaming configuration	Indicates where the data streams. This value must match a stanza in the tnt4j.properties file. The default is com.nastel.autopilot .

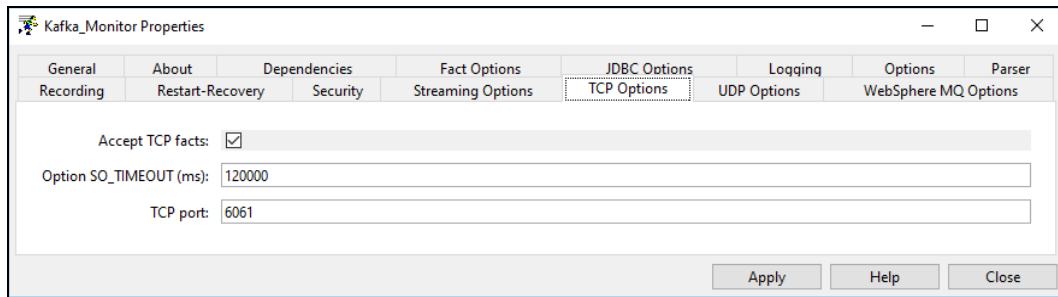


Figure 4-9. Create Kafka Monitor – TCP Options Tab

TCP is a reliable data connection to ensure facts will be published, but there is a slight performance hit due to the extra networking overhead required.

1. Click *TCP Options* tab, to enable an M6 Process Wrapper to receive TCP data.
2. Check *Accept TCP Facts* checkbox , and then enter a port. The port will also have to be specified in the application sending the fact data.

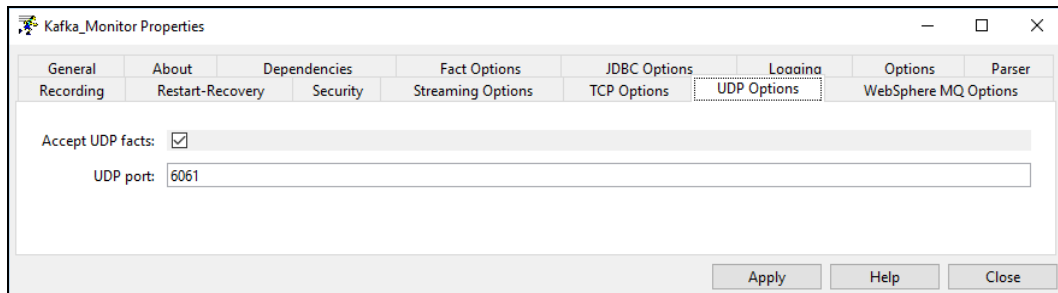


Figure 4-10. Create Kafka Monitor – UDP Options Tab

UDP is a less reliable data connection used mainly for speed and application-to-application decoupling. While this is the fastest protocol with the least amount of network overhead, there is potential that data could be lost since there is no acknowledgment/hand shaking between the sender and receiver.

1. Click *UDP Options* tab to enable an M6 process wrapper to receive UDP data.
2. Check *Accept UDP Facts* checkbox and then enter a port. The port will also have to be specified in the application sending the fact data.

This Page Intentionally Left Blank

Chapter 5: AP/Kafka Expert Metrics

This section describes some of the AP/Kafka Expert metrics collected by the expert coming from the Kafka. The data presented is collected using JMX services and can be modified as discussed in the configuration section above. The facts produced are samples only.

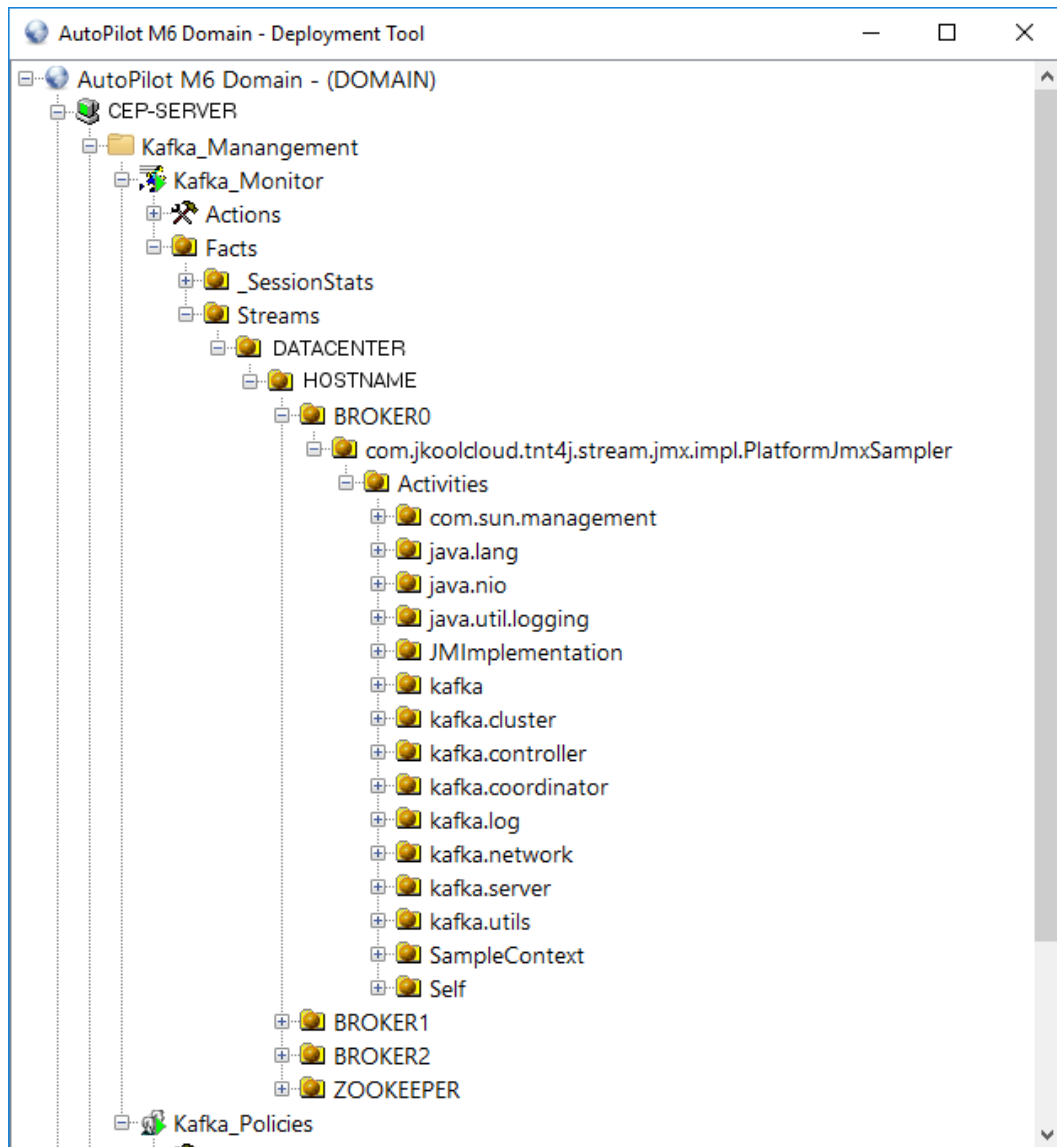


Figure 5-1. Kafka Expert Facts

5.1 Kafka Servers

Kafka servers provide a variety of metrics about the health of the server.

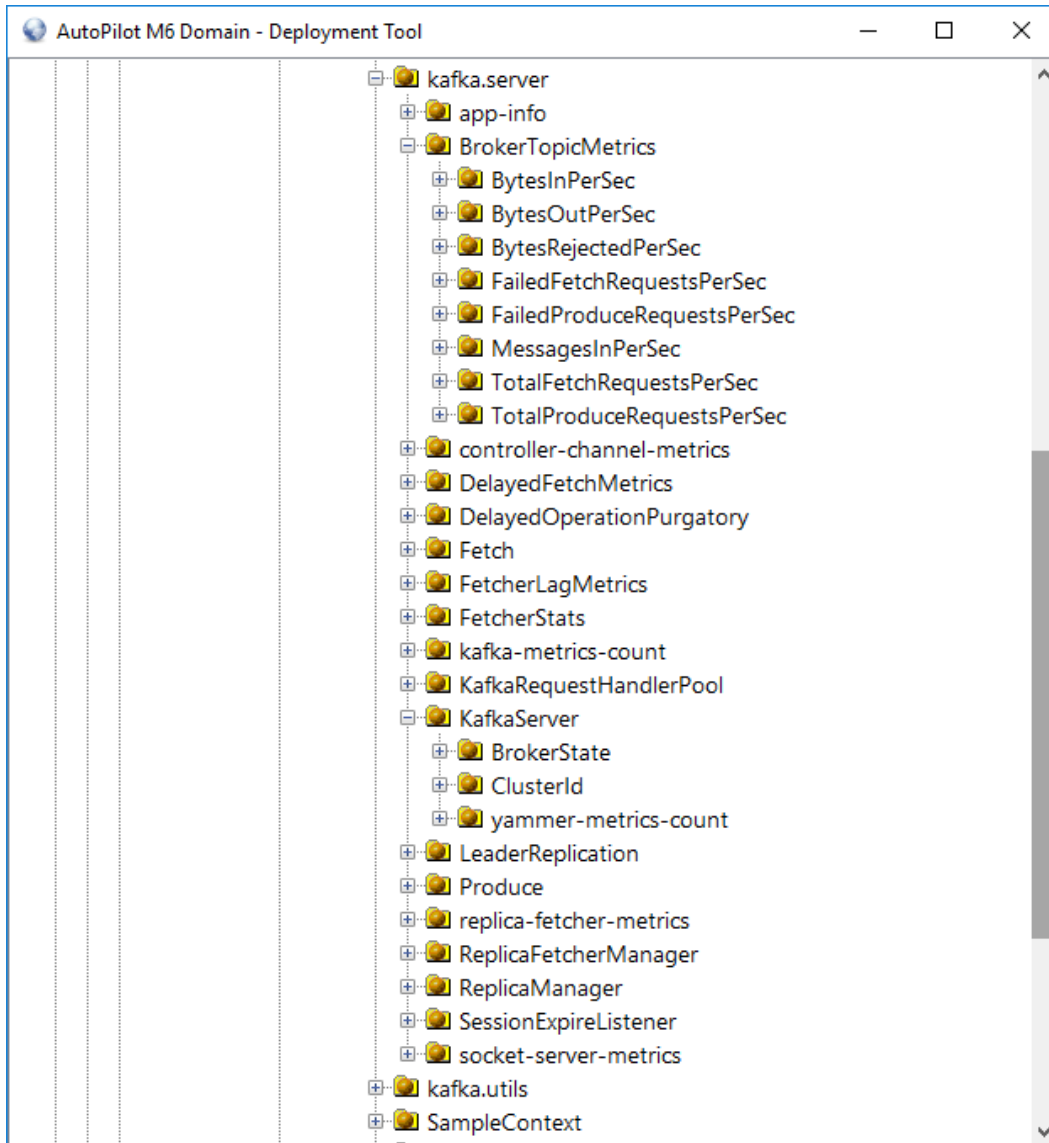


Figure 5-2. Server Metrics

5.2 java.lang

These metrics are specific to the Java JVM, including memory usage and GC information.

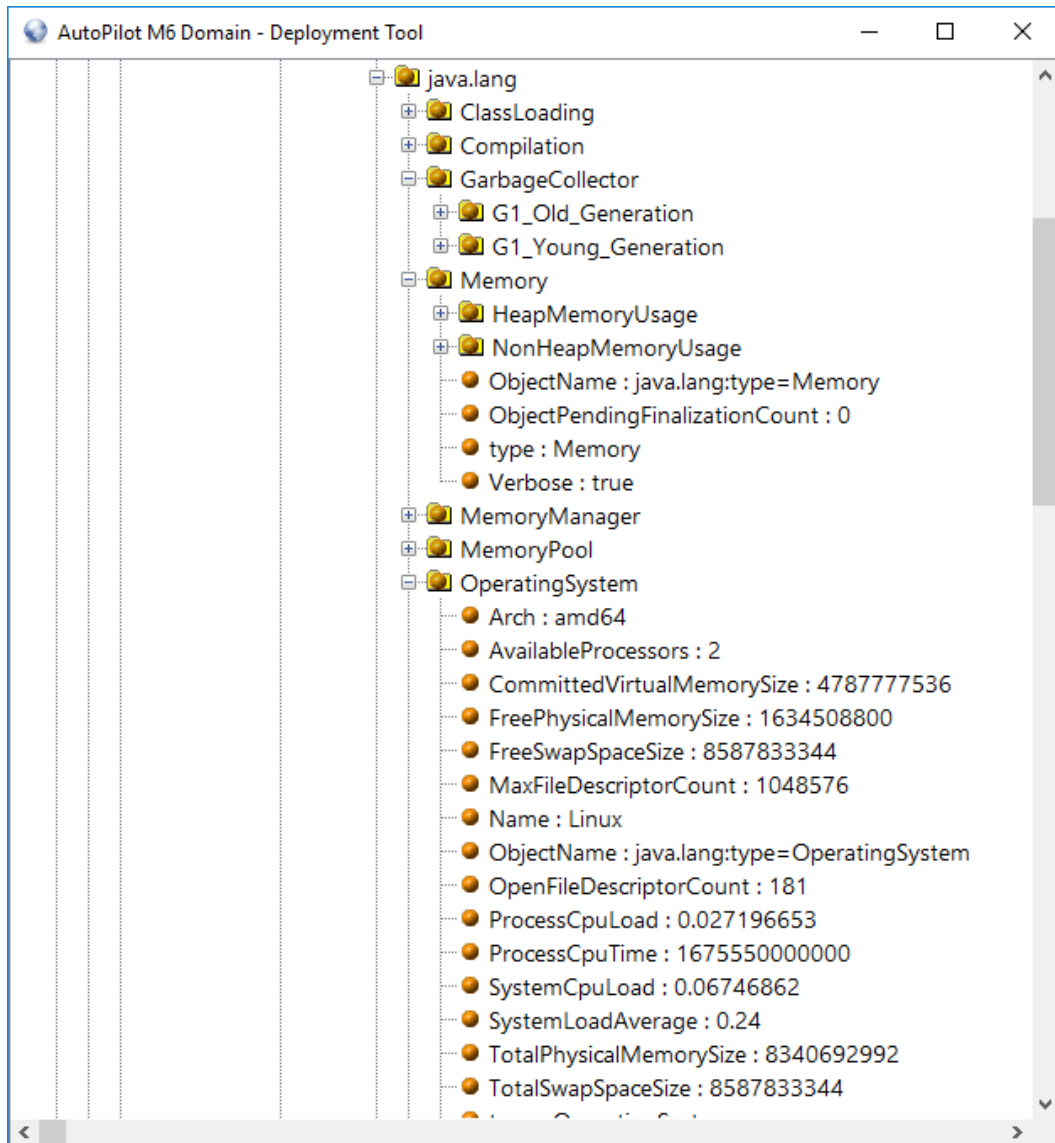


Figure 5-3. java.lang

5.3 kafka.cluster

Identifies key performance indicators about the health of the cluster, specifically identifying under replicated partitions.

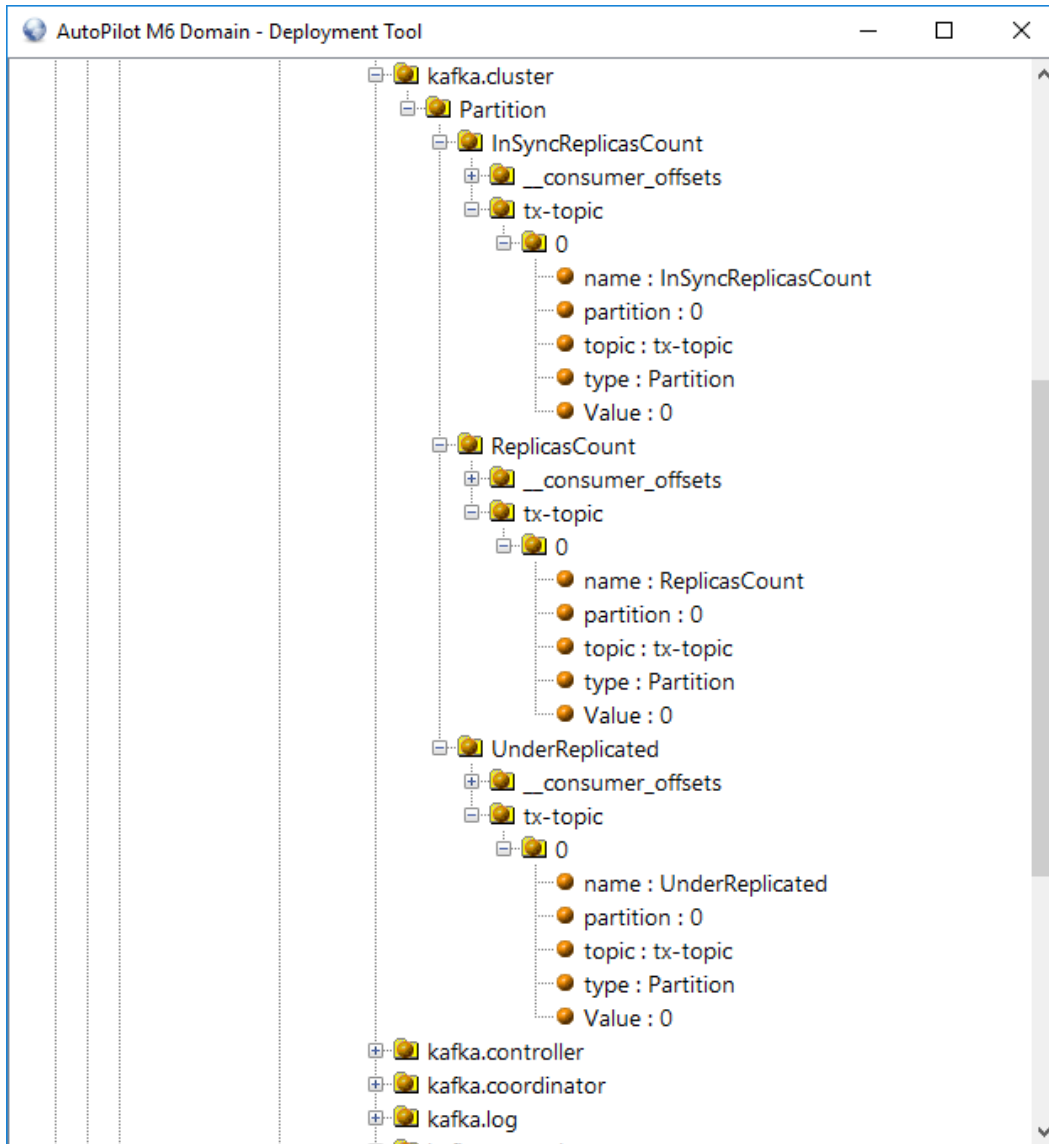


Figure 5-4. kafka.cluster

5.4 kafka.network

The network is a key contributor to the health of the Kafka environment. These indicators provide details about the requests and timings

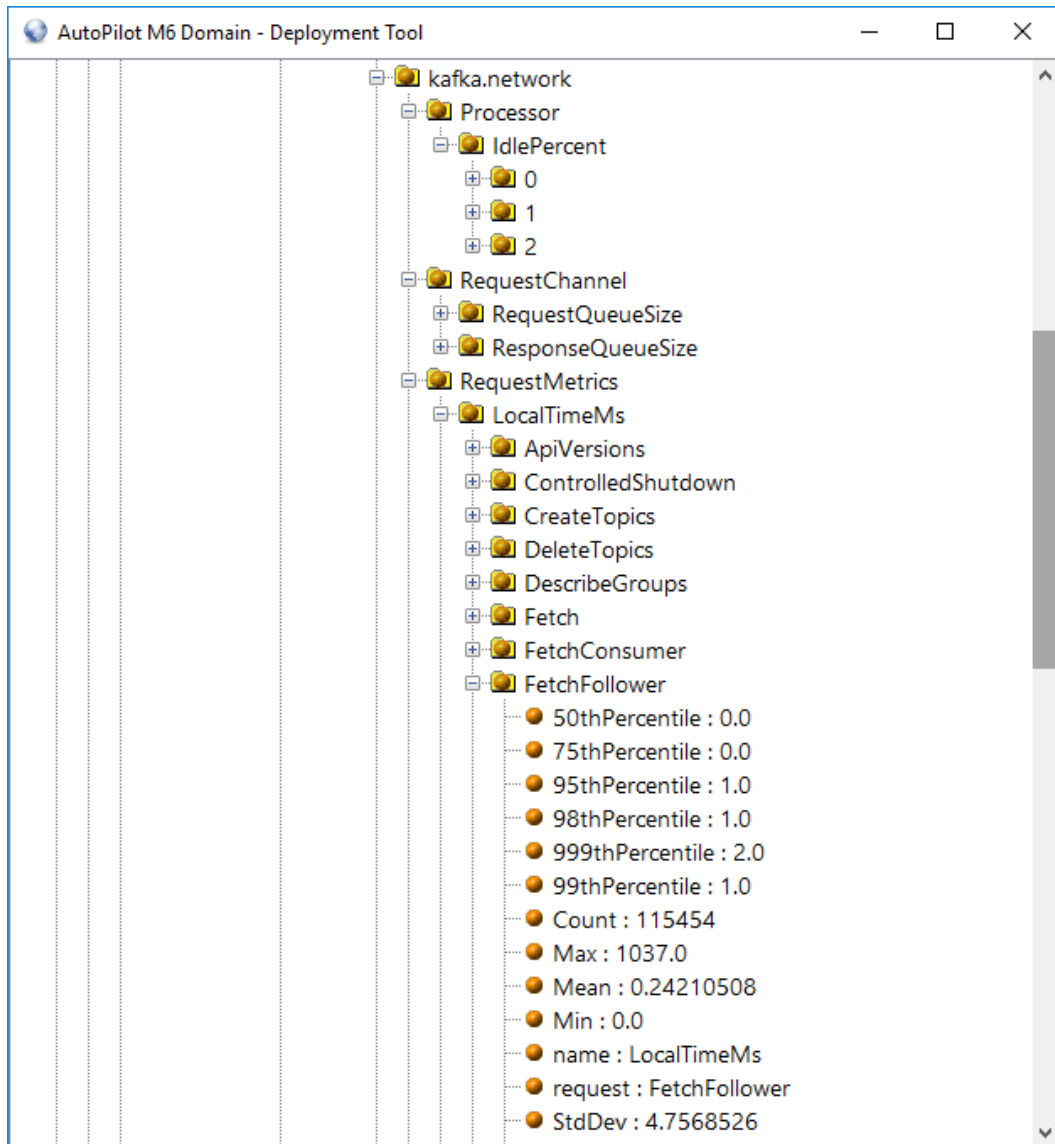


Figure 5-5. kafka.network

5.5 kafka.log

The logging subsystem can have a direct impact on the ability of Kafka to complete tasks in a timely manner to ensure replication is completing as needed. The indicators provide details on the timings of key Kafka functions.

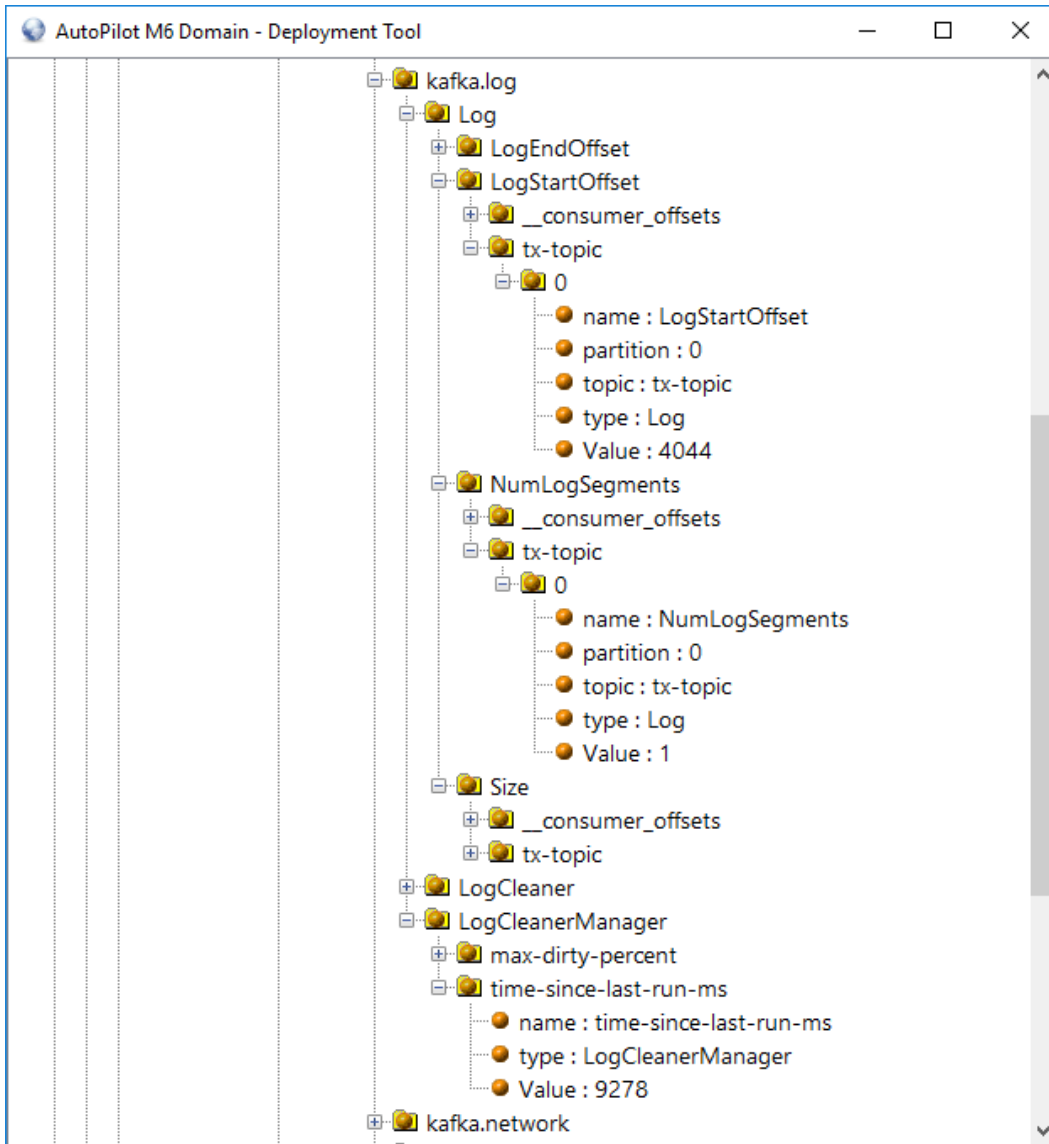


Figure 5-6. kafka.log

Chapter 6: AP/Kafka Sample Policies

This Kafka plugin provides a set of out-of-the-box policies which analyze the data collected by the Kafka Monitor. This section outlines the key policies and use cases for these policies. Additional out of the box policies are included.

6.1 Kafka Dashboard

This dashboard provides insight into the overall health of the server.

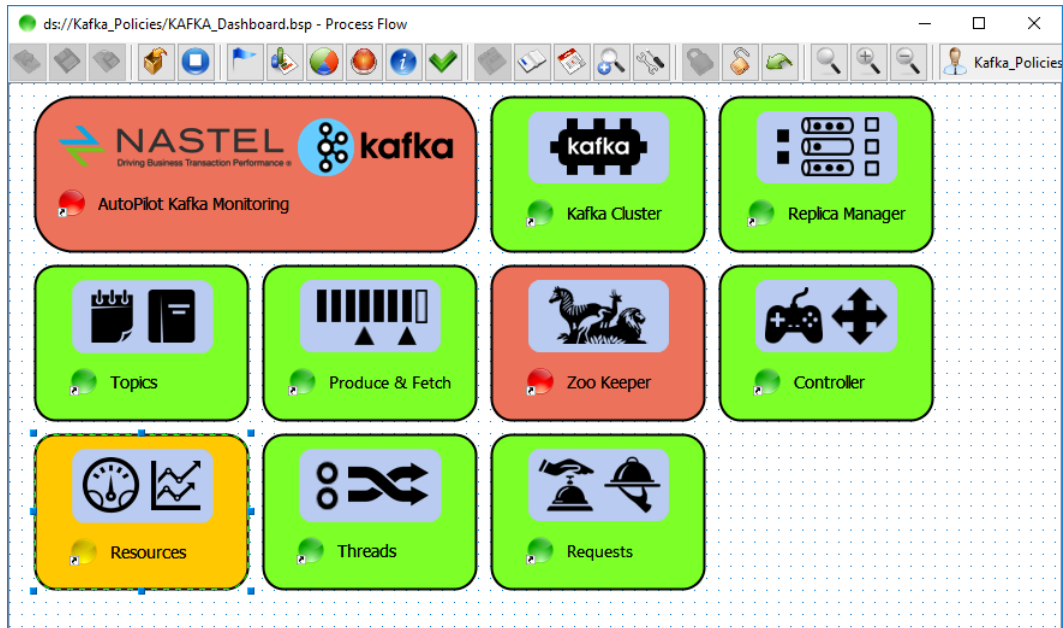


Figure 6-1. Kafka Dashboard

6.1 Kafka Resources

This example analyzes the tracking of JVM memory over time.

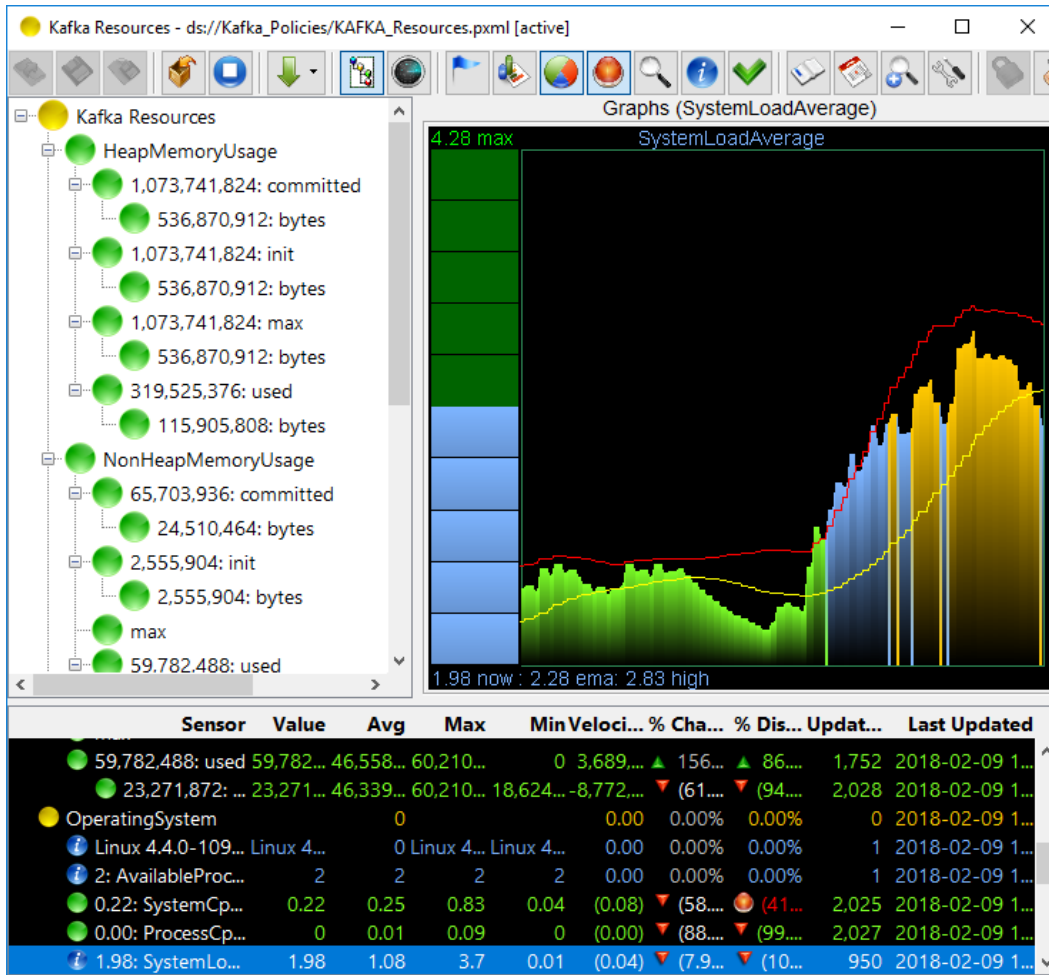


Figure 6-2. Kafka Resources

Appendix A: References

A.1 Nastel Documentation

Table A-1. Nastel Documentation	
Document Number (or higher)	Title
M6/INS 625.001	<i>AutoPilot M6 Installation Guide</i>
M6/USR 625.001	<i>AutoPilot M6 User's Guide</i>
M6WMQ 600.004	<i>AutoPilot M6 Plug-in for WebSphere MQ</i>
M6WMQ/ADM 658.002	<i>AutoPilot M6 for WebSphere MQ Administrator's Guide</i>
M6WMQ/INS 658.002	<i>AutoPilot M6 for WebSphere MQ Installation Guide</i>
M6/OSM 600.002	<i>AutoPilot M6 Operating System Monitors Installation and User's Guide</i>
AP/TEMS 110.004	<i>AutoPilot Plug-in for TIBCO EMS</i>
AP/OR 100.006	<i>AutoPilot/Oracle Plug-in Guide</i>
AP/IT JMX 430.001	<i>AutoPilot/JMX Plug-in Guide</i>

This Page Intentionally Left Blank

Appendix B: Conventions

B.1 Typographical Conventions

Convention	Description
<u>Blue/Underlined</u>	Used to identify links to referenced material or websites. Example: support@nastel.com
Bold Print	Used to identify topical headings and to identify toggle or buttons used in procedural steps. Example: Click EXIT .
<i>Italic Print</i>	Used to place emphasis on a title, menu, screen name, or other categories.
Monospaced Bold	Used to identify keystrokes/data entries, file names, directory name etc.
<i>Monospaced italic</i>	Used to identify variables in an address location. Example: [C:\AutoPilot_Home]\documents. Where the portion of the address in the brackets [] is variable.
Monospaced Text	Used to identify addresses, commands, scripts etc.
Normal Text	Typically used for general text throughout the document.
Table Text	Table text is generally a smaller size to conserve space. 10, 9, and 8 point type is used in tables throughout the AutoPilot product family of documents

B.2 Naming Conventions

Naming conventions have been adjusted to accommodate IBM's re-naming of MQSeries products to WebSphere MQ.

Nastel has adapted AutoPilot products to reflect IBM's product naming changes. In the redesign of AutoPilot, we have also better defined many elements within the AutoPilot product line.

Old Name	New Name
AutoPilot/MQSI	AutoPilot for WebSphere MQI
MQSeries Plug-in for AutoPilot	WebSphere MQ Plug-in for AutoPilot
MQControl	AutoPilot for WebSphere MQ
MQSeries	WebSphere MQ (IBM)

This Page Intentionally Left Blank