INDUSTRIAL DATABASE TECHNOLOGY

Canary System Admin Version 20

The Canary System consists of multiple pieces of software that help you accomplish three main objectives:

Collection & Storage Of Process Data
Adding Context To Your Data
Data Analytics and Connectivity

This guide will focus on the necessary information to install, configure, and administer the Canary System. For additional information, it is recommended to use the help files within the Canary Admin which can be accessed using F1 or the Canary Knowledge Base available at https://help.canarylabs.com

Canary Terminology

Canary System - The three main components of the Canary System are Data Collection, Data Storage, and Data Analytics

Sender - A service that collects, compresses, and encrypts data locally before sending it to the Canary Receiver.

Receiver - A service that moves the data received from the Sender to the Historian for permanent storage. Receiver can communicate with multiple Senders and allows for logging redundancy.

Store and Forward (SaF) - The ability of the Sender to buffer data locally when the Receiver or Historian is unavailable, preventing data loss.

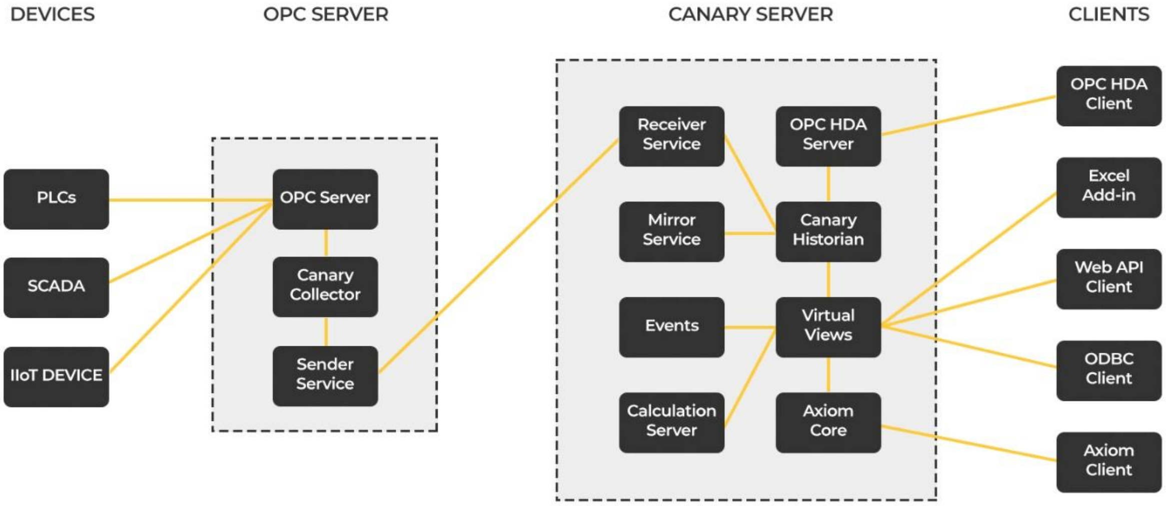
DataSets - A group of logically associated tags stored within the Canary Historian.

Tag - A single stream of time-series data, also referred to as points, channels, or items.

Historical Database File (HDB) - Within each DataSet, the Historian writes tag properties and TVQs to an HDB file. HDB files are segmented by a time-period, typically daily.

TVQ - For each tag value change, Canary records a Timestamp, a Value, and Quality. Other historical databases and SCADA platforms may refer to these as VTQs, VQTs, or Value-Timestamp Pairs.

Sample Architecture



Best Practices

It is always recommended to install the Canary Sender Service as local to the data source as possible. Additionally, whenever possible, collect data directly from an MQTT broker or OPC Server decoupled from a SCADA System or DCS.

The Canary Sender Service will communicate via WCF and use Store and Forward technology between the Sender and the Canary Receiver Service. The Receiver Service is always local to the Canary Historian. The Sender automatically buffers data as it is logged in local memory while the Receiver and Historian complete the task of committing the records to disk. Should the Receiver or Historian not be available to the Sender Service, the data will automatically be written to local disk and stored in the following folder – C:\ProgramData\Canary Labs\Sender\Buffer.

Upon reconnection to the Receiver Service or Historian, the cached data will automatically backfill to the Historian, without any admin interaction. Should the machine the Sender Service is installed on and buffering to be restarted, the cached data will persist.

System Specifications

The Canary Historian must be installed on a Windows server running at least Windows 7 or a VM running Server 2012 or newer. It is recommended that the server have the following resources as a minimum:

- 4 Core 2.20 GHz Processor
- 16 GB Memory
- 500 GB HD

Depending on the number of tags you wish to store, how often those tag values update, and the client demands on the server, you may need to scale these resources. Typically, we suggest these benchmarks based on TVQ update frequency:

System	10K Updates/Sec	50K Updates/Sec	300K Updates/Sec	1M+ Updates/Sec
Processor	4 Core 2.2+ GHz	4 Core 2.4+ GHz	8 Core 2.4+ GHz	10 Core 2.4+ GHz
Memory	8 GB	8 GB	16 GB	32 GB
Storage*	1 TB	4 TB	10 TB	20+ TB

*Storage capacity will change based on data types stored, desired historical availability, and other varying factors.

Software Installation

Historian Server:

Download and the following software on the Canary Historian:

- CanaryHistorian-ServerApps-x64-19.*.*.exe
- Axiom_x64-19.0.0.exe

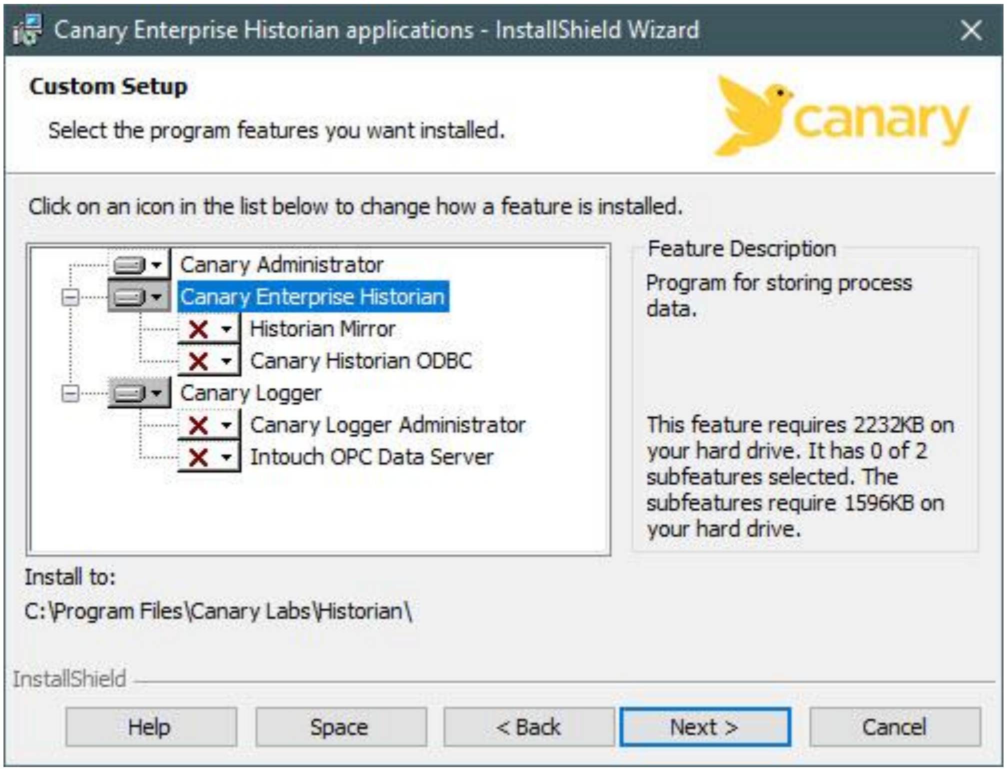
Begin by installing the EntHistorian-Serverapps-x64-19.*.* first.

Right-click on the executable and select “Run as Administrator”

The installer will require you to also install Microsoft Visual C++ 2015 Updates as well as .Net Framework 4.7.x if they are not already present

After accepting the Terms of Agreement, select only the following components for installation:

- Canary Administrator
- Canary Enterprise Historian



Canary Enterprise Historian applications - InstallShield Wizard

Custom Setup

Select the program features you want installed.

Click on an icon in the list below to change how a feature is installed.

- Canary Administrator
- Canary Enterprise Historian
  - ☒ Historian Mirror
  - ☒ Canary Historian ODBC
- Canary Logger
  - ☒ Canary Logger Administrator
  - ☒ Intouch OPC Data Server

Feature Description

Program for storing process data.

This feature requires 2232KB on your hard drive. It has 0 of 2 subfeatures selected. The subfeatures require 1596KB on your hard drive.

Install to:

C:\Program Files\Canary Labs\Historian\

Click “Next” to complete the installation.

To install the Axiom Core on the Historian Server, right-click on Axiom_x64-19.0.0.exe and select “Run as Administrator”.

Follow all installer instructions and complete the installation.

Restart the server.

OPC DA Collector:

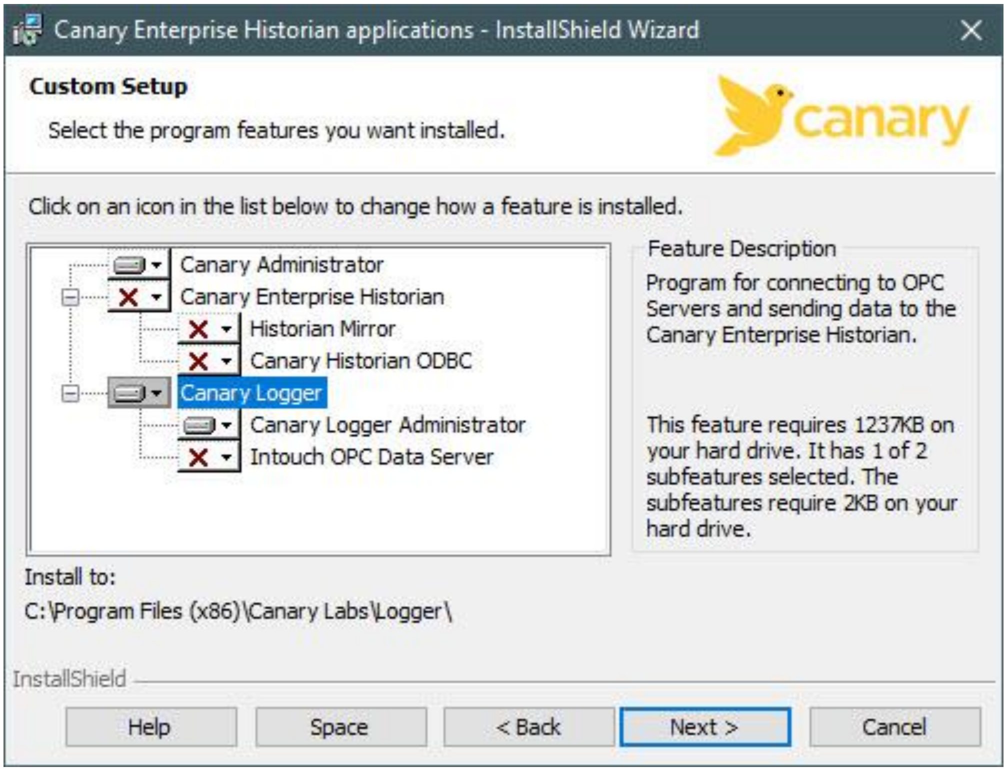
Download and install the following software on the OPC DA Server:

- CanaryHistorian-ServerApps-x64-19.*.*.exe

Right-click on the executable and select “Run as Administrator”.

After accepting the Terms of Agreement, select only the following components for installation:

- Canary Administrator
- Canary Logger
- Canary Logger Administrator



Canary Enterprise Historian applications - InstallShield Wizard

Custom Setup

Select the program features you want installed.

Click on an icon in the list below to change how a feature is installed.

- Canary Administrator
- ✗ Canary Enterprise Historian
  - ✗ Historian Mirror
  - ✗ Canary Historian ODBC
- Canary Logger
  - Canary Logger Administrator
  - ✗ Intouch OPC Data Server

Feature Description

Program for connecting to OPC Servers and sending data to the Canary Enterprise Historian.

This feature requires 1237KB on your hard drive. It has 1 of 2 subfeatures selected. The subfeatures require 2KB on your hard drive.

Install to:

C:\Program Files (x86)\Canary Labs\Logger\

Click “Next” to complete the installation.

OPC UA Collector:

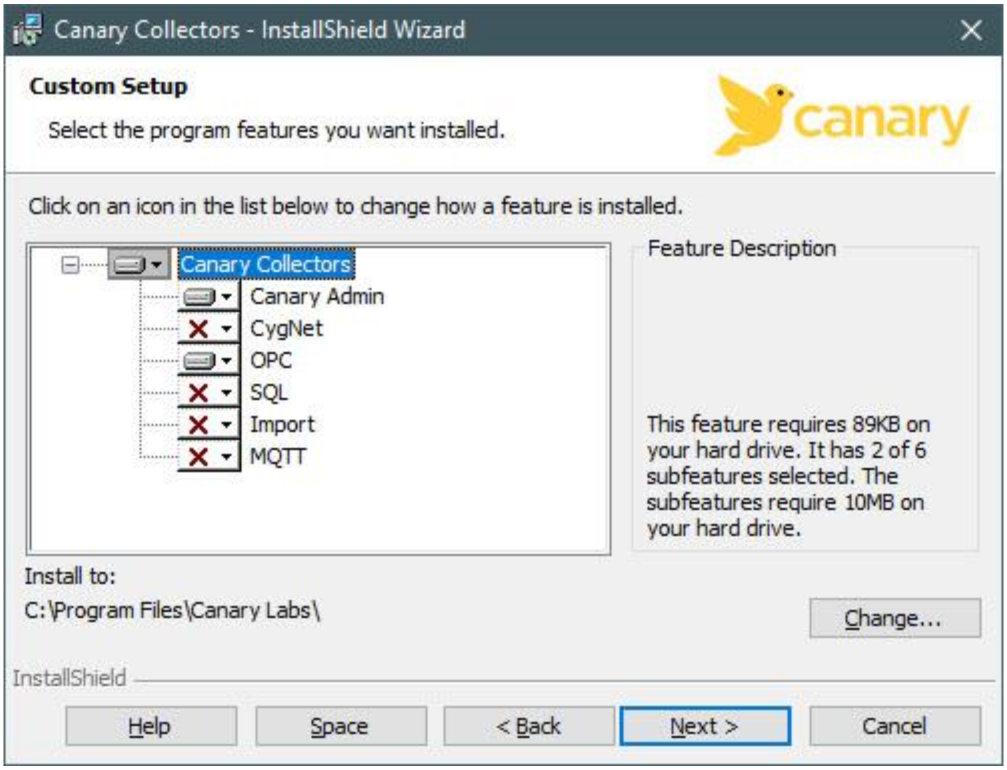
Download and install the following software on the OPC UA Server:

- Collectors-x64-19.*.*.exe

Right-click on the executable and select “Run as Administrator”

After accepting the Terms of Agreement, select only the following components for installation:

- Canary Collectors
- OPC


Canary Collectors - InstallShield Wizard

Custom Setup

Select the program features you want installed.

Click on an icon in the list below to change how a feature is installed.

- ▭ Canary Collectors
  - ▭ Canary Admin
  - ✗ CygNet
  - ▭ OPC
  - ✗ SQL
  - ✗ Import
  - ✗ MQTT

Feature Description

This feature requires 89KB on your hard drive. It has 2 of 6 subfeatures selected. The subfeatures require 10MB on your hard drive.

Install to:

C:\Program Files\Canary Labs\

Click “Next” to complete the installation.

MQTT Sparkplug B Collector:

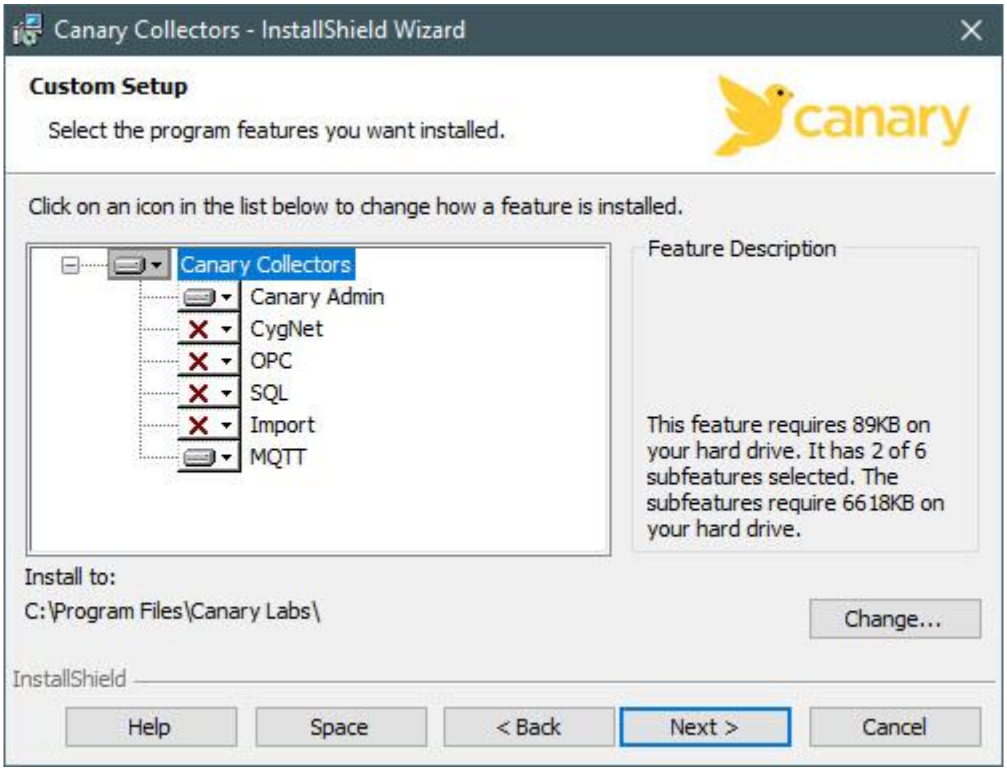
Download and install the following software on the OPC UA Server:

- Collectors-x64-19.*.*.exe

Right-click on the executable and select “Run as Administrator”

After accepting the Terms of Agreement, select only the following components for installation:

- Canary Collectors
- MQTT



Canary Collectors - InstallShield Wizard

Custom Setup

Select the program features you want installed.

Click on an icon in the list below to change how a feature is installed.

- Canary Collectors
  - Canary Admin
  - ✗ CygNet
  - ✗ OPC
  - ✗ SQL
  - ✗ Import
  - MQTT

Feature Description

This feature requires 89KB on your hard drive. It has 2 of 6 subfeatures selected. The subfeatures require 6618KB on your hard drive.

Install to:
C:\Program Files\Canary Labs\

Click “Next” to complete the installation.

Licensing the Historian Server

To license, you will need a serial number. The S/N should be in the following format:

C007-XXX-XXXX

Access the Canary Historian server.

Launch the application “Canary Admin”.

Select the Licenses Panel.

The main window will display any licenses that you currently have. To the right are two windows, the top window is for manually licensing if you do not have internet access, the bottom right window is for adding license with internet access.

Adding Licenses Automatically (requires direct internet access)

Fill out the fields for “Name”, “Company”, and “Email” based on how you would like the software licensed.

Insert your serial number in the “Serial Number” field.

Select a Quantity of “1”.

Check the boxes for “Axiom” and “Historian”.

In the bottom right corner, select “Get License”.

Your system is now licensed.

Adding Licenses Manually (no internet access required)

Copy the License Code.

Navigate to www.canarylabs.com/support/license on a resource that has internet access.

Fill out the fields for “Name”, “Company”, and “Email” based on how you would like the software licensed.

Insert your serial number in the “Serial Number” field.

Insert the License Code.

Select a Quantity of “1”.

From the License Products dropdown, select “Axiom” and “Historian”.

Click “Send”.

You will be provided with a License Key. Copy this key into the License Key box on the Canary Admin’s Licenses panel.

Click “Accept Key”.

Your system is now licensed.

The Canary Admin Tool

The Canary Admin tool is used by system administrators to manage the Canary System. The default installation of the Canary Admin tool is C:\Program Files\Canary Labs\Canary Admin\CanaryAdmin.exe.

Once opened, admin features are grouped by blue tiles, each explained in this document. Additionally, the Canary Admin can be used to connect to various Canary servers by using the drop down menu in the upper right hand corner.

Admin Tile

The Admin tile contains configuration settings to limit access to users with rights to Administer Canary Labs programs.

The following endpoints are provided for local and remote access. The default remote port number 55273 can be changed if it conflicts with existing applications.

Net.Pipe - Anonymous (Local Only) - Anonymous pipes provide interprocess communication on a local computer. This endpoint is always enabled and provides any interactive user access.

Net.TCP - Windows - When connecting remotely to another Canary Admin Service this endpoint will be tried first using the local users windows credentials.

Net.TCP - Username - If the windows credentials fail then the communication will fail over to allow the user to enter credentials for remote connection.

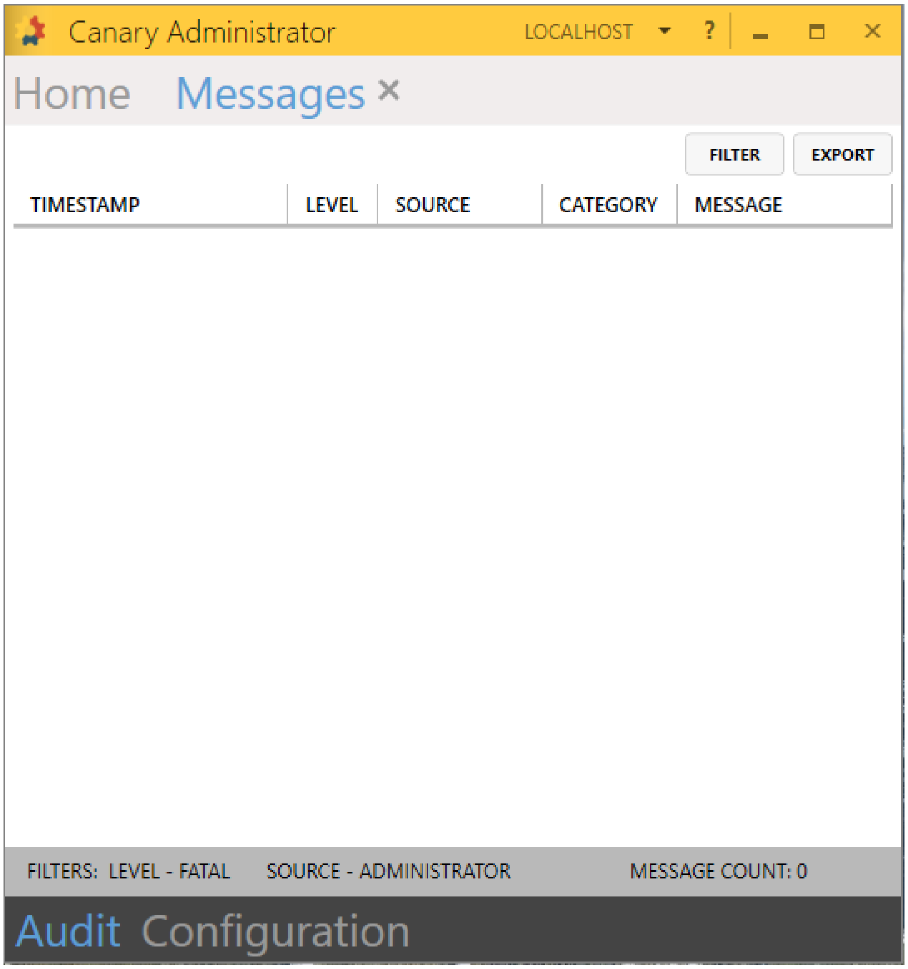
By default, Administrators have full access. Users and/or groups can be added or removed to allow access.

Settings

Persist Last Connection – When checked the Canary Admin will reconnect to the last Admin service connected. Unchecked the Canary Admin will default to the localhost.

Messages Tile

Messages encompasses information from all Canary Labs processes. It provides a sequential view of events as they happen to troubleshoot any connection, data logging or viewing problems.



TIMESTAMP	LEVEL	SOURCE	CATEGORY	MESSAGE

FILTERS: LEVEL - FATAL SOURCE - ADMINISTRATOR MESSAGE COUNT: 0

Two sub menus are presented at the bottom of the screen, “Audit”, and “Configuration”.

The “Audit” page provides a view of the messages, updated every 5 seconds with the latest messages at the top. Messages are displayed in columns and rows. Each message row will have a column for Timestamp, Log Level, Source, Category, and Message.

At the top right is a button labeled “Filter”. To configure filters, click the “Filter” button. Once filters are created, the Status bar at the bottom (medium grey) shows all filters currently applied.

The “Export” button is used to create a subset of the message database. Filtering of message type, date range and number of events can be applied.

The exported file can then be viewed with a Canary Admin by support staff or emailed to Canary Labs Customer Care (support@canarylabs.com).

The following column headers are displayed when messages exist.

Timestamp - Date and time of the message.

Level - various message levels exist.

Debug: Must be enabled from Configuration > Message Level. Provides additional debugging information in addition to the Standard messages. Usually enabled while contacting Canary Labs CustomerCare.

Trace: Must be enabled from Configuration > Message Level. Provides the most verbose log including Debug and Standard messages. Usually enabled while contacting Canary Labs CustomerCare.

Info: Messages indicating a normal sequence of events.

Warn: Messages indicating a possible problem. For instance: If a DataSet has not received any data in the past 1 minute.

Error: Messages indicating a configuration problem or possible loss of data. For instance: Was not able to identify a valid DataSet Name, or Timestamp moved in a backward direction.

Fatal: Messages indicating an exception or security problem. For instance: Service encountered an internal error and had to close, or Access is Denied.

Source - depicts the origination of the message. Available sources are as follows.

Administrator: Canary Admin Service

Axiom: Canary Axiom Core

AxiomClient: Axiom Desktop Client

AxiomView: Axiom Desktop Client

EventService: Canary Event Service

Historian: Canary Labs Enterprise Historian

License: Licensing for any products.

Logger: Canary Labs Logger

Mirror: Canary Historian Mirror

Plugin: Plugins used by Views

Receiver: Canary SAF Receiver Service

SAF Import: Canary Import Service

Sender: Canary SAF Sender Service

Views: Canary Web Service

CygNetCollectorService: Canary CygNet Collector Service

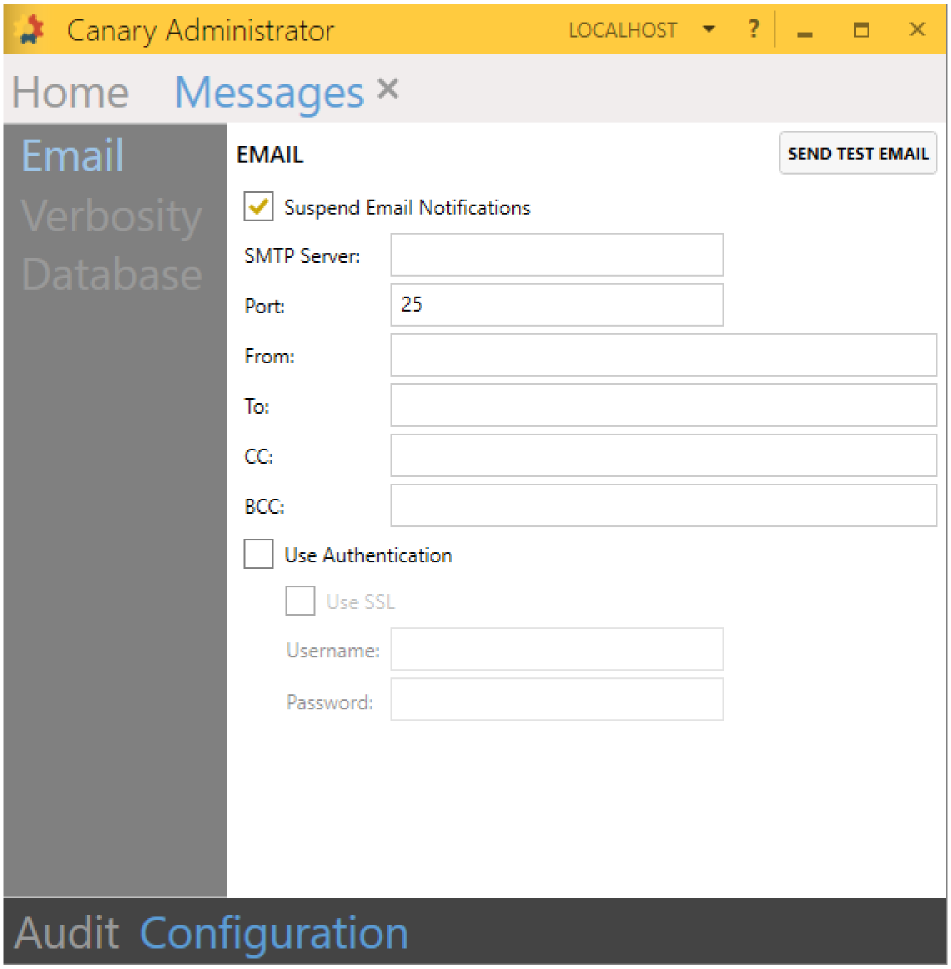
OPC Collector: Canary OPC UA Collector

SAF_SQLCollectorService: Canary SAF SQL Collector Service

Category - An event or object related to the source for filtering.

Message - Description of the event.

Below the message table is the Status Bar (medium grey). The Status Bar displays any filters that have been applied as well as the total Message Count. When first opened, the system will show the first 1000. Scrolling to the bottom will retrieve additional messages.



EMAIL

☑ Suspend Email Notifications

SMTP Server:

Port: 25

From:

To:

CC:

BCC:

□ Use Authentication

□ Use SSL

Username:

Password:

The “Configuration” tab allows setup of email notification and the message level for each source. To setup your email server, select “Email” at the left. The following will need completed.

Suspend Email Notifications - Checking this will disable all email notification until it is unchecked.

Send Test Email - Sends a test email message using the configured settings. A confirmation message is displayed indicating the status of the test.

SMTP Server - The name of the mail server that can send out mail through the SMTP protocol. If you do not know the name of your SMTP mail server, contact your Network Administrator for this information.

Port - Port 25 is the default for no authentication and port 587 when using authentication, but these are dependent on the SMTP server.

From - Required in email address format.

- When not using authentication this addres could contain the name of the machine to indicate where the message originated from, for example "Mymachine@canarylabs.com".
- When "Use Authentication" is checked this email must match the authentication, Username. To track the source of the email the address can be formatted as "Mymachine <ValidEmail@mailserver.com>".

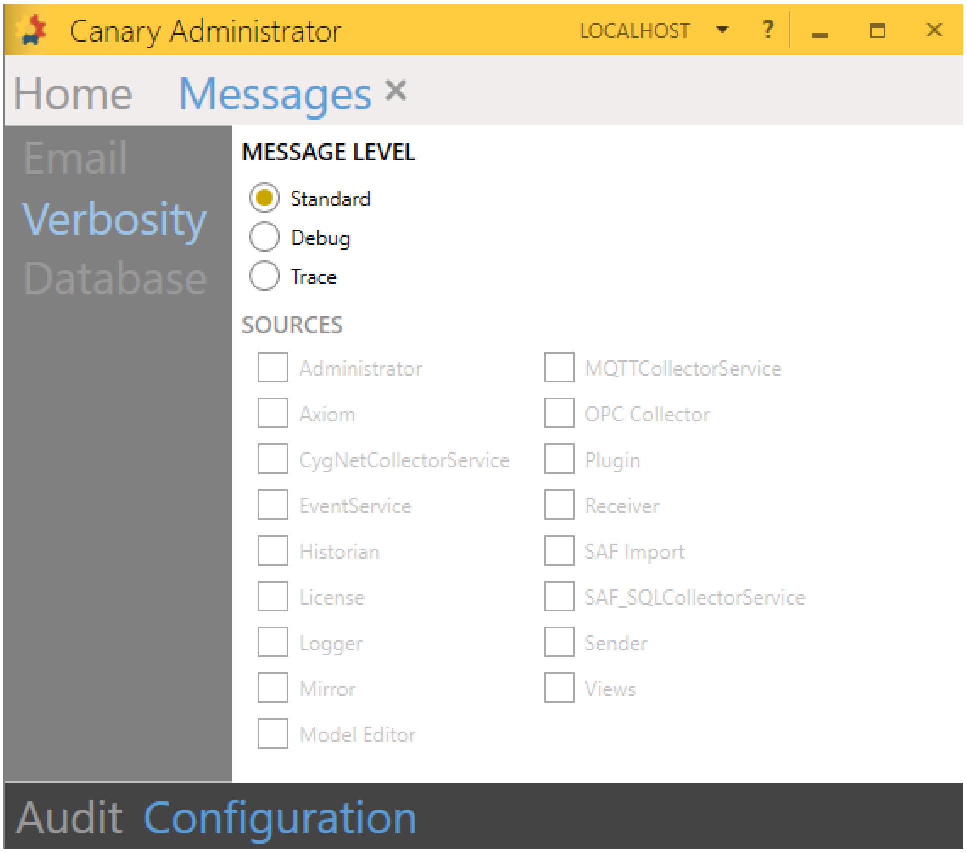
To - A valid email address. Multiple emails can be entered by separating with commas.

CC - Carbon Copy email address.

BCC - Blind Carbon Copy email address.

Use Authentication - Email credentials for SMTP servers requiring authentication.

> *Use SSL* - Most SMTP servers using Authentication will support SSL for encryption.



MESSAGE LEVEL

● Standard
○ Debug
○ Trace

SOURCES

□ Administrator
□ Axiom
□ CygNetCollectorService
□ EventService
□ Historian
□ License
□ Logger
□ Mirror
□ Model Editor
□ MQTTCollectorService
□ OPC Collector
□ Plugin
□ Receiver
□ SAF Import
□ SAF_SQLCollectorService
□ Sender
□ Views

Additionally, at left is the "Verbostiy" menu. This may be configured to control/filter both the level of messages as well as their sources.

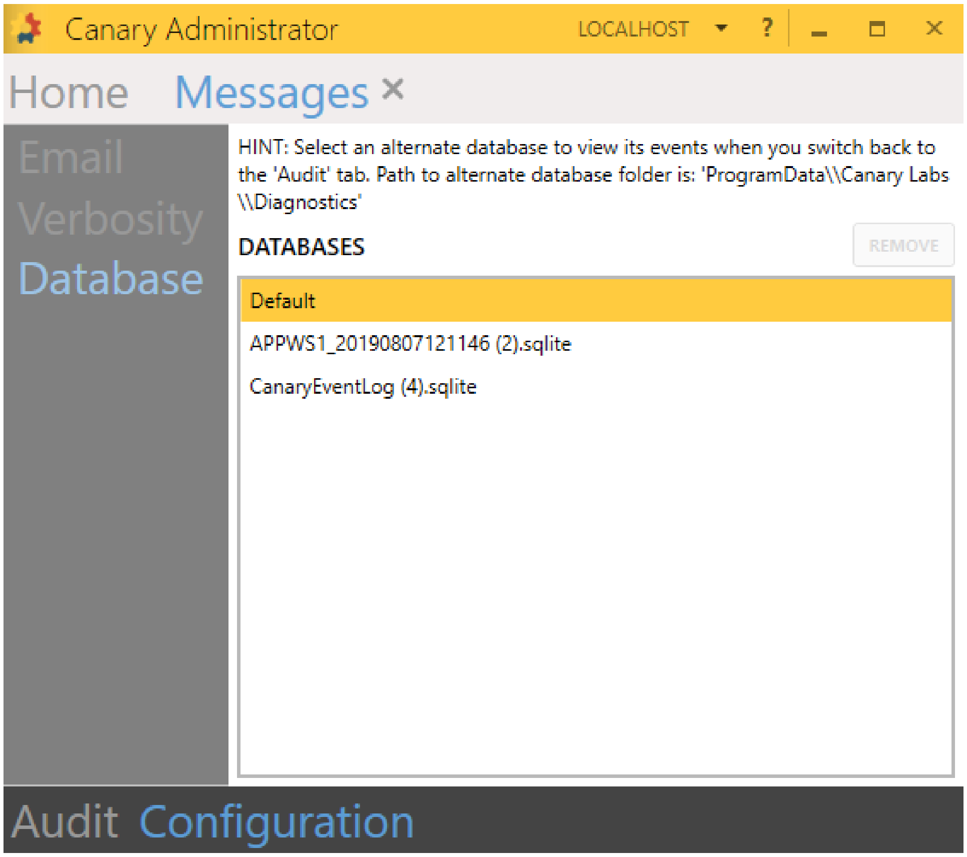
Message Level - Debug or Trace messages will consume extra resources to write out the message log. Be sure to set back to Standard for normal operation.

Standard: Info, Warn, Error and Fatal errors

Debug: Includes Standard plus Debug messages written into the code.

Trace: Function calls and results depending on application. Currently the Views application has Trace capability. Includes Standard, Debug and Trace messages. Very verbose.

Sources - The Sources available will vary depending on products installed. All sources will log messages at the Standard Level. The sources checked will also log messages for Debug or Trace, whichever is selected.



HINT: Select an alternate database to view its events when you switch back to the 'Audit' tab. Path to alternate database folder is: 'ProgramData\\Canary Labs\\Diagnostics'

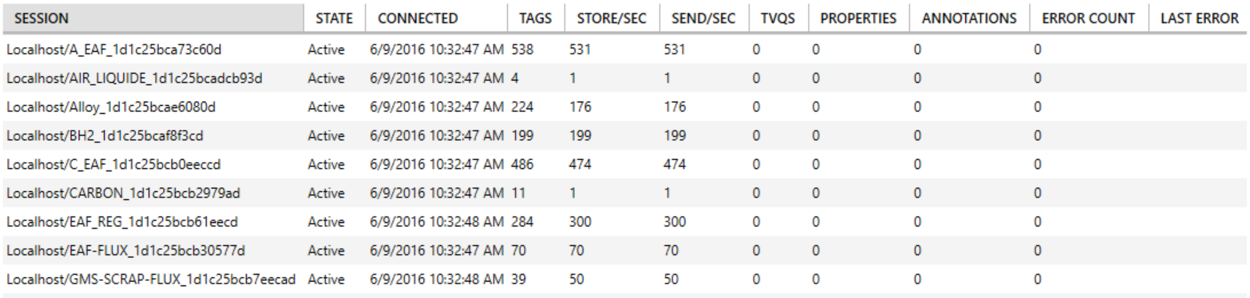
DATABASES

Default
APPWS1_20190807121146 (2).sqlite
CanaryEventLog (4).sqlite

Finally, the "Database" menu is a diagnostic tool mainly used by Canary Support. It will be viewable only when a Message database has been exported into the default location. Allows support staff or Canary Labs Customer Care to select an alternate database.

Sender Tile

Status



SESSION	STATE	CONNECTED	TAGS	STORE/SEC	SEND/SEC	TVQS	PROPERTIES	ANNOTATIONS	ERROR COUNT	LAST ERROR
Localhost/A_EAF_1d1c25bca73c60d	Active	6/9/2016 10:32:47 AM	538	531	531	0	0	0	0	
Localhost/AIR_LIQUIDE_1d1c25bcadcb93d	Active	6/9/2016 10:32:47 AM	4	1	1	0	0	0	0	
Localhost/Alloy_1d1c25bcae6080d	Active	6/9/2016 10:32:47 AM	224	176	176	0	0	0	0	
Localhost/BH2_1d1c25bcaf8f3cd	Active	6/9/2016 10:32:47 AM	199	199	199	0	0	0	0	
Localhost/C_EAF_1d1c25bcb0eeccd	Active	6/9/2016 10:32:47 AM	486	474	474	0	0	0	0	
Localhost/CARBON_1d1c25bcb2979ad	Active	6/9/2016 10:32:47 AM	11	1	1	0	0	0	0	
Localhost/EAF_REG_1d1c25bcb61eecd	Active	6/9/2016 10:32:48 AM	284	300	300	0	0	0	0	
Localhost/EAF-FLUX_1d1c25bcb30577d	Active	6/9/2016 10:32:47 AM	70	70	70	0	0	0	0	
Localhost/GMS-SCRAP-FLUX_1d1c25bcb7eecad	Active	6/9/2016 10:32:48 AM	39	50	50	0	0	0	0	

Opening the Sender tile will display all the local sessions that are currently paired with a collector service. Each session is given a unique name made up of the computer and log session names plus an ID. Also displayed is the following:

Sender States – One of various conditions as outlined below.

Active – Sender is connected to Receiver and transferring data.

Inactive – Connected but has not received data from logger in the last 5 minutes.

Connecting attempt # - The Sender will try to make a connection with a Receiver every 5 seconds.

Waiting (send tags) – The Sender is limiting how many tags to connect at one time so as not to overload the system.

Waiting (buffered data) – Buffered data in another session must complete before this session can send data.

Waiting (timeout) – A timeout has occurred waiting for a response from the Receiver.

Forward failed – Data was sent to the Receiver, but the confirmation of receipt was not returned.

Reconnect – A Receiver service is not available. Possibly the service is stopped or does not exist.

Connected - Date/Time when the session last made connection with the Receiver or "Disconnected".

Tags - Total tags included in the session.

Store/Sec - TVQs per second being written to the Sender buffer.

Send/Sec - TVQs per second being sent to a Receiver.

TVQs - TVQs currently stored in the Sender buffer.

Properties - Tag Properties currently stored in the Sender buffer. There are usually about 6 properties per tag, sent only once, when the log session connects with the Sender.

Annotations - Annotations from a custom sender client currently stored in the Sender buffer. Not used by the Canary Labs Logger.

Error Count - Number of errors logged since the session was created.

Last Error - The last Sender error logged to the Message log for each session.

Selecting a session and clicking the “Purge” button can cause loss of data. Only do this if the buffered data is no longer wanted. A confirmation dialog displays to force the session to close and delete all buffered data.

Configuration

The Sender service uses the following endpoints for local and remote access:

Net.Pipe – Anonymous (Local Only) – Anonymous pipes provide interprocess communication on a local computer. This endpoint is always enabled and provides any interactive user access.

Https – Username (Web API) – Secure endpoint for custom applications to write data into the historian. The default remote port number of 55254 can be changed if it conflicts with existing applications.

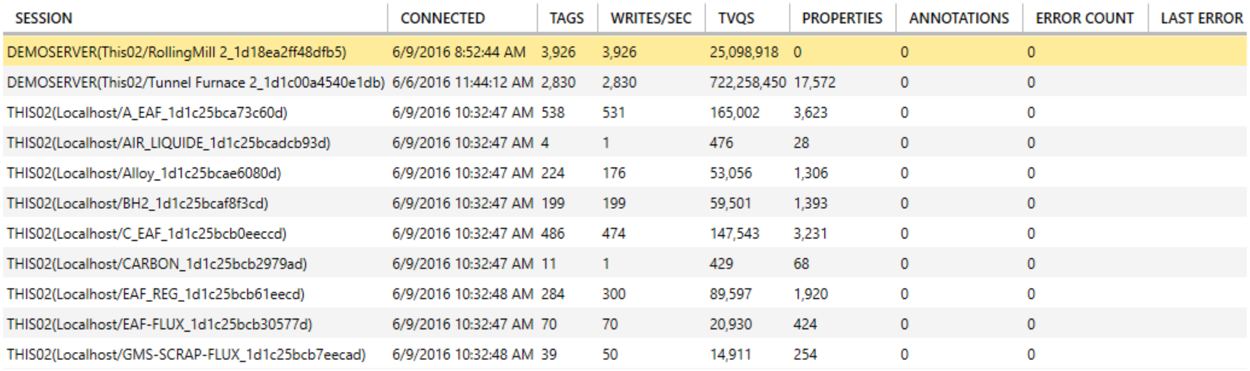
Http – Anonymous (Web API) – Unsecure endpoint for custom applications to write data into the historian. The default remote port number of 55253 can be changed if it conflicts with existing applications.

Users or groups are granted access to connect to the Sender through the *Https – Username* endpoint via the Access tab. The Allow panel displays the users/groups who have access. The Deny panel displays the users/groups who are specifically denied.

Receiver Tile

Status

Opening the Receiver tile will display all the sessions that are currently logging to the local historian. Each session is given a unique name made up of the Sender source followed by the Sender session name. Also displayed is the following.



SESSION	CONNECTED	TAGS	WRITES/SEC	TVQS	PROPERTIES	ANNOTATIONS	ERROR COUNT	LAST ERROR
DEMOSERVER(This02/RollingMill 2_1d18ea2ff48dfb5)	6/9/2016 8:52:44 AM	3,926	3,926	25,098,918	0	0	0	
DEMOSERVER(This02/Tunnel Furnace 2_1d1c00a4540e1db)	6/6/2016 11:44:12 AM	2,830	2,830	722,258,450	17,572	0	0	
THIS02(Localhost/A_EAF_1d1c25bca73c60d)	6/9/2016 10:32:47 AM	538	531	165,002	3,623	0	0	
THIS02(Localhost/AIR_LIQUIDE_1d1c25bcadcb93d)	6/9/2016 10:32:47 AM	4	1	476	28	0	0	
THIS02(Localhost/Alloy_1d1c25bcae6080d)	6/9/2016 10:32:47 AM	224	176	53,056	1,306	0	0	
THIS02(Localhost/BH2_1d1c25bcaf8f3cd)	6/9/2016 10:32:47 AM	199	199	59,501	1,393	0	0	
THIS02(Localhost/C_EAF_1d1c25bcb0eeccd)	6/9/2016 10:32:47 AM	486	474	147,543	3,231	0	0	
THIS02(Localhost/CARBON_1d1c25bcb2979ad)	6/9/2016 10:32:47 AM	11	1	429	68	0	0	
THIS02(Localhost/EAF_REG_1d1c25bcb61eecd)	6/9/2016 10:32:48 AM	284	300	89,597	1,920	0	0	
THIS02(Localhost/EAF-FLUX_1d1c25bcb30577d)	6/9/2016 10:32:47 AM	70	70	20,930	424	0	0	
THIS02(Localhost/GMS-SCRAP-FLUX_1d1c25bcb7eecad)	6/9/2016 10:32:48 AM	39	50	14,911	254	0	0	

Session - A unique name beginning with the Sender source followed by the Sender session name.

Connected - Date/Time when the session last made connection with the Sender session.

Tags - Total tags included in the session.

Writes/Sec - TVQs per second being written to the Historian.

TVQs - TVQs written since the last connection with the Sender session.

Properties - Tag properties sent since last connection. There are usually about 6 properties per tag. The Sender will only send properties once when a session is created, not after a reconnect.

Annotations - Count of annotations being sent through a custom sender client. Not used by the Canary Labs Logger.

Error Count - Number of errors logged since the session was created.

Last Error - The last Receiver error logged to the Message log for each session.

Redundancy

By logging the same tags from two locations to the same DataSet assures the data will be written if one location goes down. This requires the timestamps be normalized to the same interval from both log sessions. The Redundant tab is only visible if redundant tags exist in the Receiver sessions.

Configuration

The Receiver service uses the following endpoints for local and remote access:

> *Net.Pipe – Anonymous (Local Only)* – Anonymous pipes provide interprocess communication on a local computer. This endpoint is always enabled and provides any interactive user access.
>
> *Net.TCP - Username* – The Sender will try the Anonymous endpoint first. If it is not enabled then the Sender will try this secure endpoint using the credentials specified in the Sender credentials consisting of an Identity name and GUID. The default remote port number of 55256 can be changed if it conflicts with existing applications.
>
> *Net.TCP - Anonymous* – Checking the Anonymous endpoint will allow any Sender to connect. A Sender outside a firewall could still be forced to

use the secure endpoint by only opening the secure port 55256 for access. The default remote port numbers of 55255 can be changed if it conflicts with existing applications.

Users or groups are granted access to connect to the Receiver through the *Net.TCP – Username* endpoint via the Senders tab. The Allow panel displays the users/groups who have access. The Deny panel displays the users/groups who are specifically denied.

Data Collection

Methodology Overview

Canary collects data local to the source using industry standard formats like OPC UA/DA, MQTT, SQL, CSV, or through manual data entry. Canary also can collect data directly from SCADA systems or via a .NET or Web API. Canary Collectors are used to create logging sessions. Each logging session is configured based on the data source and allows for tag naming, dead-banding, data transformation, and additional configuration.

Once data logging is configured, Canary's Sender Service encrypts, compresses, and caches the data locally. Sender will then push the data packet across the network to the storage portion of the Canary system. You can use multiple logging configurations and Sender instances across your company without worrying about rising costs because both are unlicensed. In addition, each Sender can push data to additional Canary storage systems creating redundancy and fast disaster-recovery.

To move data for long-term storage Canary uses a Store and Forward Service comprised of the Sender and Canary's Receiver Service. The Receiver Service accepts data from multiple Senders and moves that data into the Canary Historian database. If Sender and Receiver loose contact, or if Historian is offline, Sender will buffer all data locally. Once the Historian's availability returns, all buffered data backfills automatically without any data loss.

Configuring OPC UA Logging Sessions

Create a Logging Session

From the OPC Server, launch the application Canary Admin.

Open the OPC Collector panel.

Select “Configuration” from the menu across the bottom of the Canary Admin.

The Configuration page features three windows, “Sessions”, “Session Settings”, and “Tags”.

From the Sessions window, select “New Session”.

In the “Session Settings” window, create a Name for the session.

Determine which historian or historians you wish to push data to and enter those names in the “Historian” section. This can be either the machine name or the IP address. If logging to multiple historians, separate their names with a comma. (ie *Historian1, Historian2*).

Select the DataSet you wish to log to from the dropdown menu or enter the name of the DataSet manually that you wish to create.

“New File” is used to create a logging file revision. It instructs the Historian to release any tag licensing for tag names that do not appear in this new session. You may leave it unchecked for a new session, but should always be checked it if you are removing tags from a previous session.

Enter the machine name or URL for the OPC server. If you are local to the machine, you may also use “localhost”.

Specify the host port.

Apply your changes in the red bar at the top of the page.

Configure Tags

Click “Browse” in the Tags window.

From the popup window, browse the OPC server and highlight the tags you wish to log by using the mouse and the Shift key. Once selected, you may choose which properties you would like to include by checking the available properties listed in the “Properties” window.

Use the “Apply” button to select the tags you wish to log. When finished, click “OK”.

A table has been created that will list all tags you have selected. The first column of the table is empty. Clicking on a cell in the first column will highlight the entire corresponding row.

When a row is selected, you may edit each cell within the “Tag Settings” window.

You can also delete an entire row by first selecting the row, and then clicking “Delete” from the top right corner of the “Tags” window.

Clicking the “New” button will create a new row which can then manually be configured within the “Tag Settings” window.

You may also use the “Tag Properties” feature to add metadata or property columns.

Once clicked, a side window appears that features a dropdown of standard properties. You may select and add these properties or manually enter your own desired properties. Once your property list has been created, click “Apply”.

You must finalize all additions and modifications by clicking the Apply button within the red bar at the top of the page.

Tag Configuration with Microsoft Excel

You have the option to copy logging data from Microsoft Excel and can paste it into your logging session. To do this, follow these steps.

Create columns in Excel that match the order of columns as seen in the logging session (‘Canary Tag Name’, OPC Node ID’, ‘Type’, ‘Deadband’, ‘Transform’, and any additional meta data properties you desire).

Once your table is complete in Excel, select the row numbers you wish to copy, thus highlighting all your logging configuration columns without including the column headers. Press CTRL+C.

In the Canary Admin navigate to your OPC Collector logging session configuration menu. Click the “PASTE NEW” button at the top of the screen to paste all copied rows and columns into the logging configuration.

If you make an error or need to reconfigure, you can click the grey square to the left of “GROUP” to highlight the entire table and then click the “REMOVE” button to delete your work. Additionally, you can select a single row by clicking the empty box to the left of the Group Name or multiple rows using the SHIFT key for multiselect.

Create Groups

You can create multiple logging groups within a single session by using the “New Group” button within the “Sessions” window.

Clicking “New Group” will create another group within the session. When the group is selected, you may edit its Name, Sample Interval, Deadband Type, and choose to Normalize Time in the “Group Settings” window.

Finalize all modifications by clicking “Apply” at the top of page.

Start/Stop Logging Session

To begin logging data, click on the “Status” on the bottom menu.

You will see a grey panel for each session you have created.

Click the play icon to begin logging. The panel will turn blue when logging has begun. To stop the session, click the stop logo.

Clicking the panel once will highlight it yellow and display a session summary in the “Session” window on the bottom left of the Canary Admin.

Double clicking the session panel will show all logging groups within the session.

Double clicking a group will allow you to visually monitor the logging session as well as filter the columns that are shown using the “Modify Grid” feature.

Configuring OPC DA Logging Sessions

Create a Logging Session

From the OPC Server, launch the application Logger Administrator.

A logging session will automatically cache data to local memory. To adjust the memory threshold, change the value in the “Minimum available memory” field.

Click “File” and select “New Log Session”

A new tab named “Untitled” will be created next to the “General” tab.

Name your logging session in the “Name” field.

Enter the machine name or IP address for your Canary Historian in the “Historian Computer” field. If logging to multiple historians, separate their names with a comma. (ie *Historian1, Historian2*).

“Create new file when logging starts” is used to create a logging file revision. It instructs the Historian to release any tag licensing for tag names that do not appear in this new session. You may leave it unchecked for a new session but should always be checked it if you are removing tags from a previous session.

Edit and Create Tag Groups

Group 1 has automatically been created and a tab for Group 1 is visible under the “Name” field. To create additional groups, right-click the Group 1 tab and select “Add New Group”.

Edit the “Group Name” as needed.

Use the “Computer” field to enter the machine name or IP address of the OPC Server if not local. If local, leave blank.

Select the OPC Server of choice by clickin ... he button.

Once the OPC Server is selected, you can browse for tabs by using Ctrl+B or right-clicking within the main dark grey window and selecting “Browse OPC Server”.

A new “Add Trends” window will appear.

First, use the dropdown at the top of the window to select the DataSet in which you wish to log data. Note, if no DataSets are visible, you will need to close the “Add Trends” window and create a DataSet on the Canary Historian Server.

You may choose to prepend the tag name with it’s browse path by selecting the “Prepend browser path” box.

Browse through the OPC Server structure in the “Server Nodes” window to find your tags. Once tags are visible in the window on the right, select the tags you wish to log using the mouse and Shift key.

Use the “Apply” button to select the tags you wish to log. When finished, click “OK”.

A table has been created that will list all tags you have selected. The first column of the table is empty. Clicking on a cell in the first column will highlight the entire corresponding row.

Any cell may be edited by double-clicking on the cell.

You can also delete an entire row by first selecting the row, and then pressing the delete key or by right-clicking and selecting “Delete Selected Trend(s)”.

Once all tags are selected you can apply your changes at the top of the Logger Administrator by clicking the “Apply” icon.

Next, save your logging configuration by clicking the “Save” icon.

Finally, click “Start” to log data.

Configuring MQTT Sparkplug B Logging Sessions

Create a Logging Session

The DB Browser for SQLite or comparable SQLite database tool is required. You can download a free version from this link: https://sqlitebrowser.org/

The Canary MQTT collector utilizes an SQLite database for configuration settings. It is located at:

C:\programdata\canary labs\logger\storeandforward\mqttcollector\mqttcollector.sqlite

The collector will automatically create the database configuration file upon first run.

To configure the MQTT Collector, open the database file with DB browser. There are three separate tables that will need to be configured, “Servers”, “Subscriptions”, and “SubscriptionSenderSessions”.

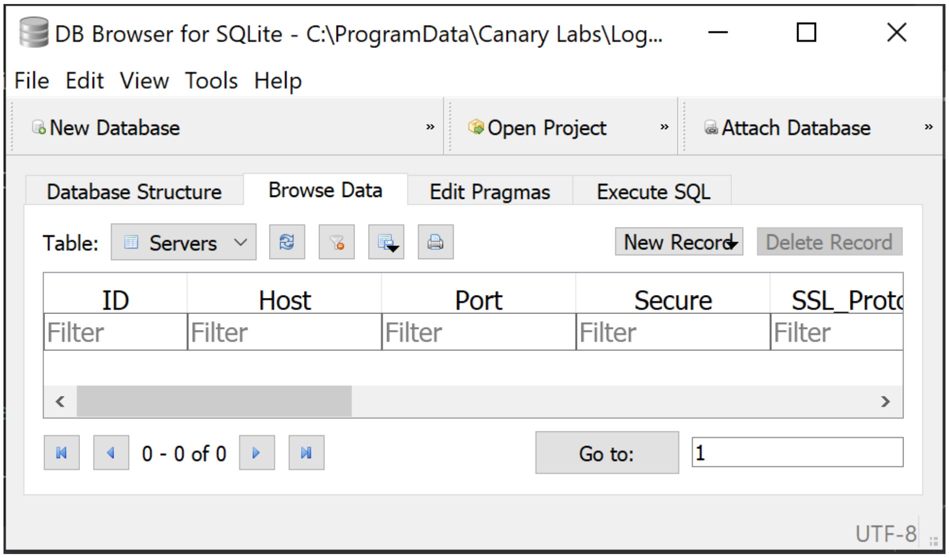


Table: Servers

ID	Host	Port	Secure	SSL_Proto
Filter	Filter	Filter	Filter	Filter

Go to: 1

SERVERS

Click on the “Browse Data” tab and select the “Servers” table in the drop down list.

Click on “New Record”.

Enter information for each of the fields:

ID - Auto incremented field (leave default).

Host - (MQTT Server): IP address, machine name, or domain name. An example would be Mqtttest.canarylabs.com or 192.168.0.1.

Port - (MQTT Server Port): Port number the MQTT server is listening on. Standard ports are 1883 unsecure and 8883 for secure.

Secure - Whether or not security is being used (1 = true, 0 = false).

SSL_Protocol - Type of security being used (0 = none, 1 = SSLv3, 2 = TLSv1.0, 3 = TLSv1.1, 4 = TLSv1.2).

Username - Username to use when connecting.

Password - Password to use when connecting.

Enabled - Controls whether the connection is enabled. Set to 0.

SessionKeepAlive - Leave default (00:00:30) = 30 seconds. Canary sender session keep alive.

ServerReconnect - Leave default (00:01:00) = 1 minute

IgnoreSSLErrors - Set to 1 if using security and a self-signed certificate on the MQTT server. Otherwise, set to 0.

CleanSession - Set to 1 (default).

SessionKeepAlive - 0-65535 seconds: Leave set at 0.

Click File->Write Changes.

SUBSCRIPTIONS

Select the Subscriptions table.

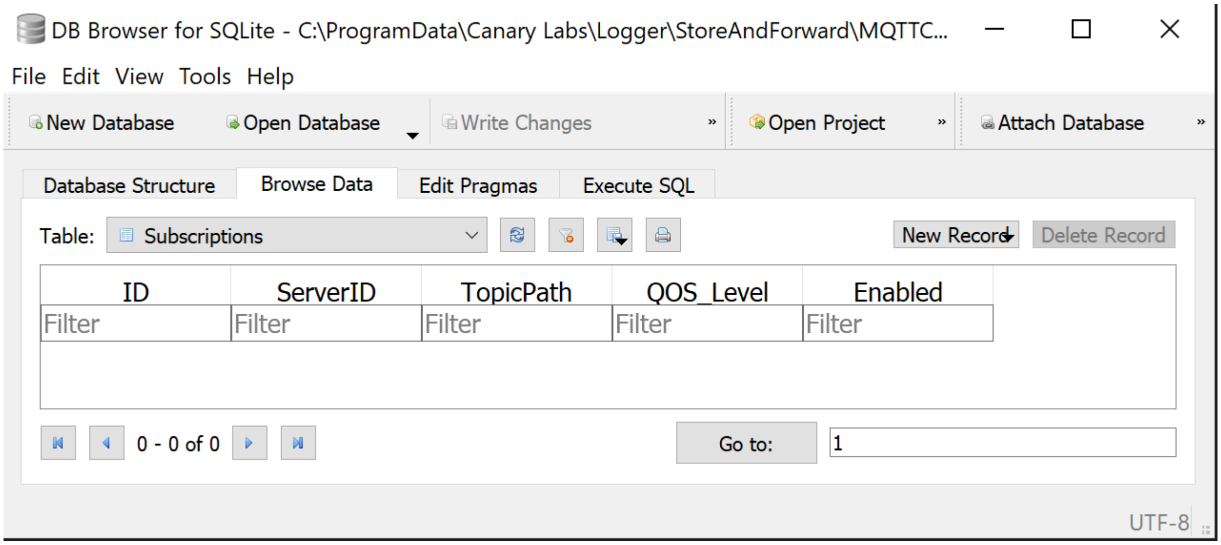
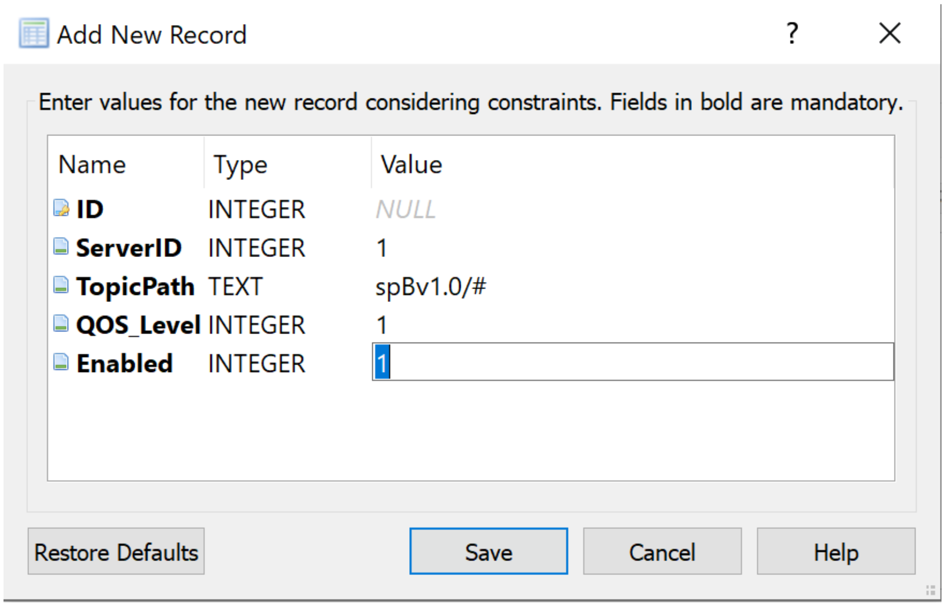


Table: Subscriptions

ID	ServerID	TopicPath	QOS_Level	Enabled
Filter	Filter	Filter	Filter	Filter

Go to: 1

Click ‘New Record’.



Enter values for the new record considering constraints. Fields in bold are mandatory.

Name	Type	Value
ID	INTEGER	*NULL*
ServerID	INTEGER	1
TopicPath	TEXT	spBv1.0/#
QOS_Level	INTEGER	1
Enabled	INTEGER	1

Configure the subscription to the MQTT server.

ID - Auto incremented field (leave default).

ServerID - This references the Servers table ID field of the record added to the servers table.

TopicPath - This is the MQTT topic that you would like to subscribe to. All Sparkplug B topics start with spBv1.0/. The hash ‘#’ stands for multi-level wildcard whereas the ‘+’ is a single level wildcard. Sparkplug B topic names are in the format spBv1.0/<GroupName>/<MessageType>/<NodeName>/<DeviceName>.

Example:

spBv1.0/# subscribes to all tags on the MQTT server

spBv1.0/Group1/+/Node1/# subscribes to all tags under node 'Node1'

spBv1.0/Group1/+/Node1/Device1 subscribes to all tags under device 'Device1'

QOS_Level - Corresponds to the publish subscribe model in MQTT. Set this to 0.

Enabled - Leave default, 1.

Click File->Write Changes.

SUBSCRIPTIONSENDERSESSIONS

Select the SubscriptionSenderSessions table.

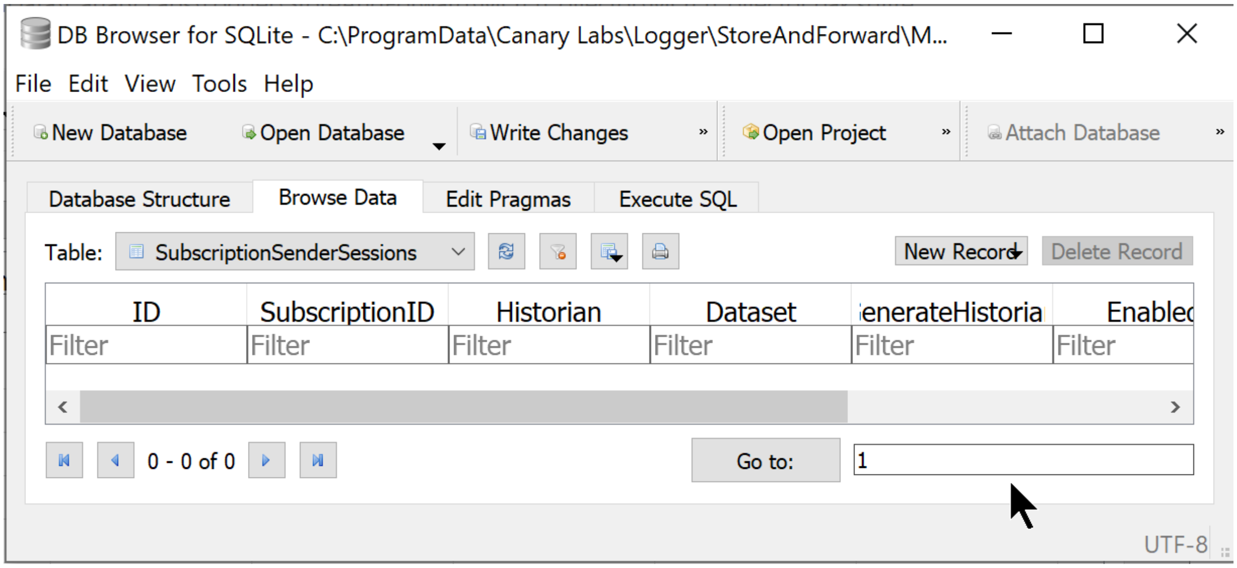

DB Browser for SQLite - C:\ProgramData\Canary Labs\Logger\StoreAndForward\M...

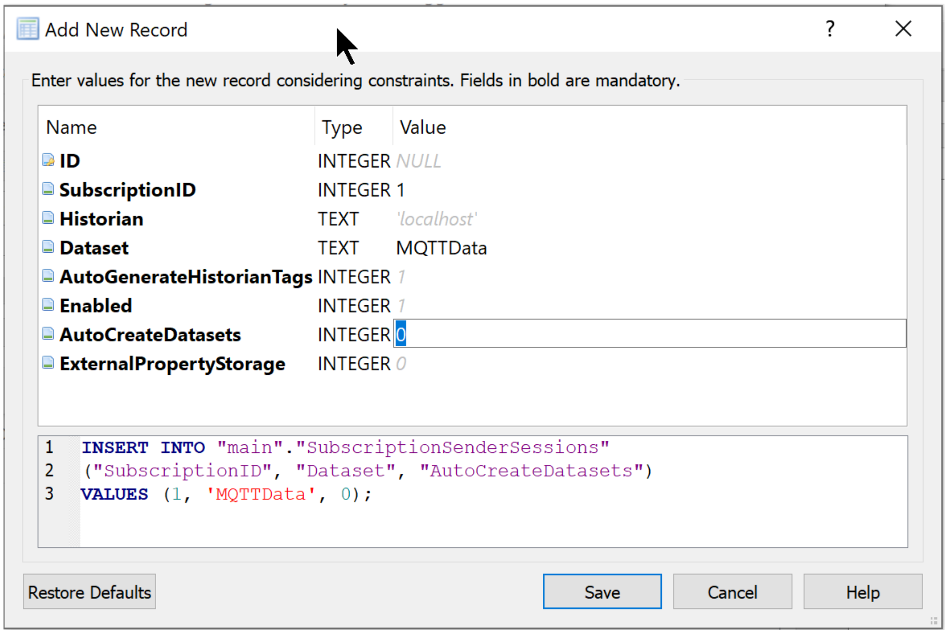
Table: SubscriptionSenderSessions

ID	SubscriptionID	Historian	Dataset	ienerateHistoria	Enablec
Filter	Filter	Filter	Filter	Filter	Filter

0 - 0 of 0

Go to: 1

Click 'New Record'.



Enter values for the new record considering constraints. Fields in bold are mandatory.

Name	Type	Value
ID	INTEGER	*NULL*
SubscriptionID	INTEGER	1
Historian	TEXT	*'localhost'*
Dataset	TEXT	MQTTData
AutoGenerateHistorianTags	INTEGER	*1*
Enabled	INTEGER	*1*
AutoCreateDatasets	INTEGER	0
ExternalPropertyStorage	INTEGER	*0*

```sql
INSERT INTO "main"."SubscriptionSenderSessions"
("SubscriptionID", "Dataset", "AutoCreateDatasets")
VALUES (1, 'MQTTData', 0);
```

Configure the Canary target for the subscribed tag data.

ID – Auto increment (leave defaulted).

Subscription ID – This references the Subscriptions table ID field for the subscription that was created previously.

Historian – Canary Historian instance, example 'localhost'

DataSet – Canary DataSet into which the data will be logged.

AutoGenerateHistorianTags – Not used, leave defaulted to 1k.

Enabled – Not used, leave defaulted to 1.

AutoCreateDataSets – If set to 1 the DataSet will be created in the Canary Historian if it does not exist. Recommended to set to 1.

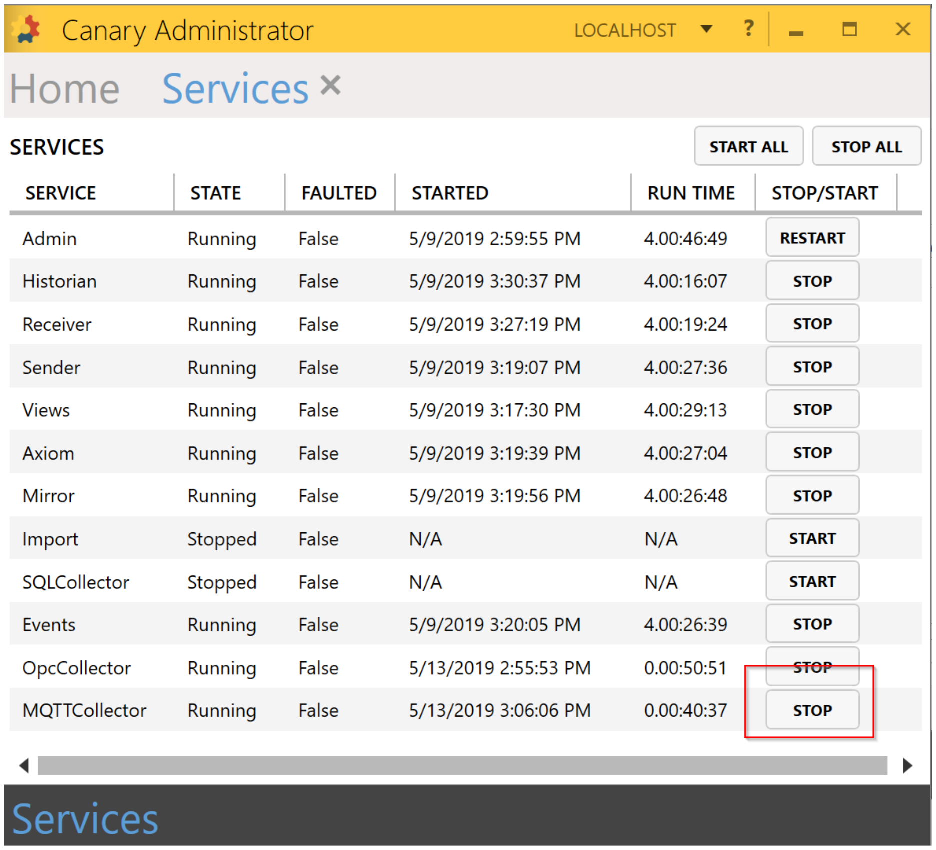
ExternalPropertyStorage – Leave defaulted to 0.

To log the same subscription data to multiple historians, add a record for each historian to the SubscriptionSenderSessions table that references the subscription.

Click on File->Write Changes in DB Browser.

Open the Canary Admin.

Within the "Services" panel, stop and start the MQTT collector from the Canary Admin.



SERVICES

SERVICE	STATE	FAULTED	STARTED	RUN TIME	STOP/START
Admin	Running	False	5/9/2019 2:59:55 PM	4.00:46:49	RESTART
Historian	Running	False	5/9/2019 3:30:37 PM	4.00:16:07	STOP
Receiver	Running	False	5/9/2019 3:27:19 PM	4.00:19:24	STOP
Sender	Running	False	5/9/2019 3:19:07 PM	4.00:27:36	STOP
Views	Running	False	5/9/2019 3:17:30 PM	4.00:29:13	STOP
Axiom	Running	False	5/9/2019 3:19:39 PM	4.00:27:04	STOP
Mirror	Running	False	5/9/2019 3:19:56 PM	4.00:26:48	STOP
Import	Stopped	False	N/A	N/A	START
SQLCollector	Stopped	False	N/A	N/A	START
Events	Running	False	5/9/2019 3:20:05 PM	4.00:26:39	STOP
OpcCollector	Running	False	5/13/2019 2:55:53 PM	0.00:50:51	STOP
MQTTCollector	Running	False	5/13/2019 3:06:06 PM	0.00:40:37	STOP

Return to the home screen in the Canary Admin, and click on the "MQTT Collector" tile.

Each logging session will be displayed. Click on the "Enable" button on the connection to start logging.

If tag data is being published, the Tags and TVQs columns will reflect the activity.

Data Storage

Methodology Overview

Canary has designed its solution to eliminate the need of a database administrator to manage the Historian. To achieve this, Canary uses a simple but elegant method to segment the data in two ways. First is the creation of DataSets, or a group of tags with logically associated data. Second is the Historical Database (HDB) file, segmented by a time period. These two organizational structures allow for database expansion without effecting performance or requiring any management of size or duration.

Generally, a new HDB file is created daily. Once closed, each HDB goes through loss-less compression to minimize Historian's storage requirements. Canary stores data using lossless data compression to ensure data written to the Historian is identical to the data later read from it. Canary algorithms achieve a data compression rate beyond sixty percent.

Each HDB file contains tag names, tag properties, and TVQ units comprised of a timestamp, a value, and a quality score. Values can include Booleans, Integers, Floats, Doubles, or Strings. Canary writes all three TVQ components together ensuring the value is always paired with the correct timestamp and quality.

The data storage structures of the Canary Historian are designed to maximize read speeds rather than write speeds. Reads are prioritized because data is only written once but read many times. To provision read performance beyond 4,000,000 updates per second, HDB files are organized so TVQs are physically stored with their associated tag, rather than the traditional method of storing TVQs in a time block. As a result, queries made to the Historian on a tag-by-tag basis are optimized for both high tag count and high-speed performance.

A single historian can be scaled to store values for two million individual tags. The Historian performs continuous write speeds at 1,500,000+ updates per second and achieves updates rates as fast as ten milliseconds.

Creating DataSets

DataSets are the way tags are organized. DataSets can be created by clicking on the Historian Panel within the Canary Admin. When logging tags, each DataSet will have it's own historical database file, called an HDB file.

Begin by clicking on the "Historian" panel in the Canary Admin.

At the bottom of the window, click "Configuration".

Click "New" to create your first DataSet.

Name your DataSet and confirm your path (default is C:\Historian Data)

The "Settings" window can now be configured. Once settings are correct, click "Create" to build your DataSet.

SETTINGS

Validate - Leave "Validate" checked. This feature will automatically validate the DataSets HDB file when it closes.

Email if no data received… - Choose "Email if no data received…" if you want to receive an email should the DataSet stop receiving data.

Roll-Over - By default, the HDB file will roll-over at midnight or when it reaches 512MB (which ever happens first). You can adjust this feature, but it is recommended to leave it set to "Daily".

Roll-Up - Smaller systems may choose to Roll-Up their collection of daily HDB files into weekly, monthly, or quarterly HDB files. There is no performance advantage to doing so and it is a best practice to leave this feature set to "Never".

Delete - You may automatically have HDB files deleted after a set period of time. The default is set to "Never". If you choose to delete files, this is a permanent action and there is no recovery. A better option would be to move files "off-line" and is discussed in the Managing DataSets documentation. Canary recommends leaving this set to "Never".

Creating Mirror Sessions

You may automate the movement of DataSets between multiple Canary Historians using the Mirror feature. Mirrors allow for scheduled snapshots of a DataSet to be pulled by another Historian.

Click the “Mirror” panel in the Canary Admin.

Click “Configuration” at the bottom of the window.

Select “New” at the top of the page.

In the blank space beside “Browse” enter the machine name or IP address you wish to mirror data from and click “Add”.

Browse the structure for the desired DataSet and select it with a single-click.

Once selected, information pertaining to the DataSet will display below and customizable settings will appear in the “Settings” window to the right.

SETTINGS

Source Address - Name or IP address of the source Historian computer. If access is to be secure then the secure port number of the Views endpoint will need to be appended to the name or IP address.

Source Historian – Computer name of the source Historian computer.

Source DataSet – Name of the DataSet to mirror from the Source Historian.

Destination DataSet - A default DataSet name is generated when a DataSet is selected. This name can be edited to be exactly the same as the source or changed to something different.

Destination Path - This is the directory location where the new DataSet is to be created.

Snapshot Time – Determines how of the the HDB files are synched. “Frequency” allows for normalized snapshot intervals from 15 minutes to 12 hours, while “Specified” snapshots are meant to occur at scheduled times. Multiple times may be entered by separating them with a comma (,).

Snapshot Files – Allows the user to choose the amount of HDB files to snapshot when starting a Mirror. The default of 'All' will transfer the entire historical record from the Source DataSet.

Snapshot Throttling – Allows for the control of the snapshot transfer over the communication line. If unchecked, the block size will be 256K with 0 interblock delay. The "Compress" feature is disabled, and the "Snapshot On Start" and "Snapshot On Reconnect" features are enabled.

Block Size (KB) and Block Delay (ms)

Network Speed	Ms / Default 256K Block	Comm Usage
1 GB	7-9 ms	85-90%
100 Mb	19-21 ms	95%
10 Mb	340-350 ms	99%

Depending on the comm link you may wish to tune down the snapshot. By setting the "delay between blocks" to the approximate time to download a single block, the Comm Usage will be decrease by about half. The overall time to download the HDB file will increase accordingly.

By decreasing the size of the blocks, you will shorten the time to transfer a single block, but you will increase the number of blocks needed to transfer the entire file. Theoretically this allows more network traffic to interweave with the snapshot blocks. On a slow communication link, reduce the block size to 8 or 16K bytes. The ratio is linear. By cutting the block size in half, the time per block will be about half.

On a 10Mb network, setting the block size to 16K, changes the single block time down to 20ms. Then changing the delay to 20ms changes the Comm Usage % to under 50%.

Compress: Adds additional message compression (similar to zipping). This compression will require additional time to compress before transmitting and then uncompressing afterward. On faster networks, the "message compression" time can take longer than the time saved by reducing the network traffic.

Snapshot On Start: On a critical control network, it may be important to control the exact time that the snapshot occurs. Uncheck this option and set the specific time to snapshot during off hours.

Snapshot On Reconnect: By default the Mirror will automatically Snapshot the files should communication be lost for any length of time. Unchecking this box will disable this functionality especially for a critical control network. If tags are added to the source dataset, or the source dataset is rolled-over, a snapshot will occur unless the “Snapshot on Reconnect” is unchecked.

Real-Time Frequency – Determines how often the mirror updates data values between snapshots. From Never to 60 seconds.

Track Performance – Checking this box will create a tag in the {Diagnostics} DataSet for this Mirror Session which tracks the communication speed.

Once all desired settings are configured, click “Create” in the upper portion of the window.

Click the “Status” option at the bottom of the screen to see the newly created Mirror Session.

Managing the Historian

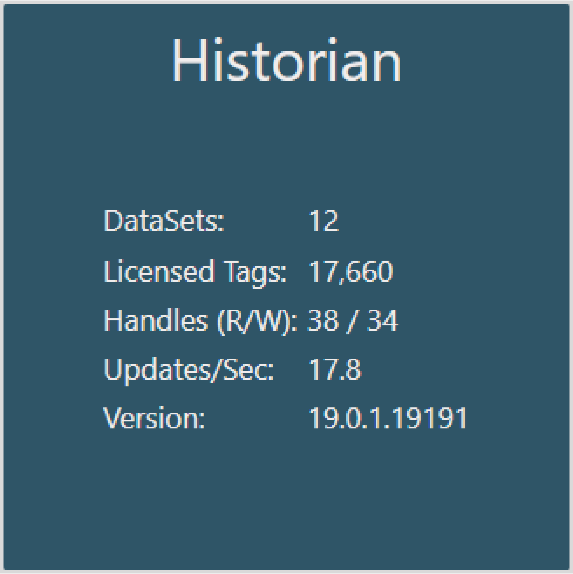
DataSets

All tags written to the Canary Historian are organized into DataSets. DataSets can be created and managed by using the Canary Admin tool.

From the main screen of the Canary Admin, the Historian panel displays several KPIs you may find helpful.

These include the total number of DataSets you have created, the number of tag licenses you are consuming, the handle count for tags being read by clients (R) and number of

tags being actively updated (W), the total updates per second of tags being written, and the version information. By clicking on the Historian panel, you can access all DataSets and their HDB files.



Historian

DataSets: 12
Licensed Tags: 17,660
Handles (R/W): 38 / 34
Updates/Sec: 17.8
Version: 19.0.1.19191

Once you have opened the Historian panel, you will see three sections, a window titled “HISTORIAN”, another window titled “DATASET”, and panels on the right 2/3rds of the window representing each DataSet.

For larger systems, use the Range Selector at the top of the page to view more DataSets if they do not all display. Adjusting the grips at either end will increase the visible range of DataSets. Sliding the yellow bar left or right will adjust the overall view.

HISTORIAN

The “HISTORIAN” panel in the upper left corner displays overall Canary Historian properties and disk space.

DataSet Count - Total DataSets including {Diagnostics}.

Licensed Tags - Tags being counted against the license.

Updates/Sec - Total updates per second being written by the historian.

Readers - Number of tags being read by Historian for clients.

Writers - Number of tags actively being updated.

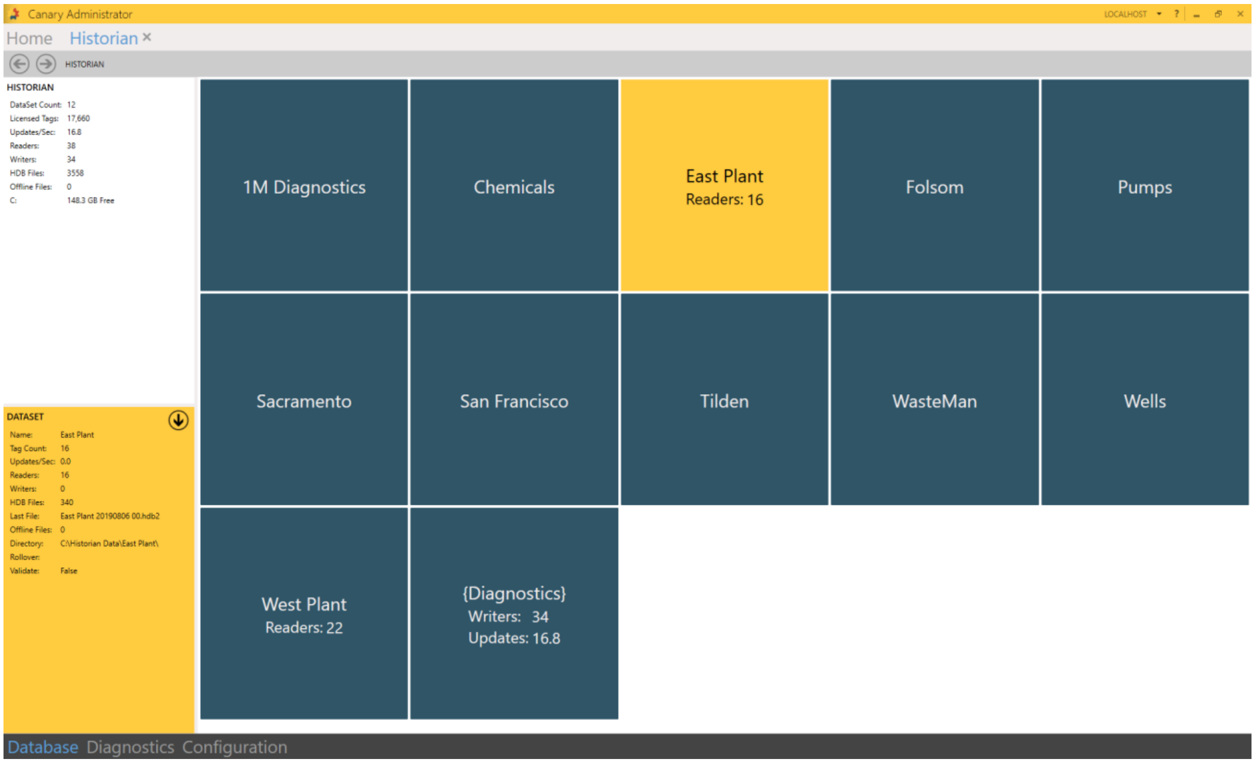
HDB Files - Total number of hdb files in all DataSets.

Offline Files - Number of files offline in all DataSets.

Disks - Free disk space per drive.

DATASET

The "DataSet" panel in the lower left-hand corner contains information for a DataSet once it has been selected. To do so, simply single-click a DataSet. The "Down Arrow" icon contains a list of executables for the DataSet.



HISTORIAN

DataSet Count:	12
Licensed Tags:	17,660
Updates/Sec:	16.8
Readers:	38
Writers:	34
HDB Files:	3558
Offline Files:	0
C:	148.3 GB Free

DATASET

Name:	East Plant
Tag Count:	16
Updates/Sec:	0.0
Readers:	16
Writers:	0
HDB Files:	340
Last File:	East Plant 20190806 00.hdb2
Offline Files:	0
Directory:	C:\Historian Data\East Plant\
Rollover:	
Validate:	False

1M Diagnostics	Chemicals	East Plant Readers: 16	Folsom	Pumps
Sacramento	San Francisco	Tilden	WasteMan	Wells
West Plant Readers: 22	{Diagnostics} Writers: 34 Updates: 16.8			

Name - The Name of the DataSet precedes all .hdb files and is used in any ItemID.

Tag Count - Total number of tags being logged in this DataSet based on the most current file.

Updates/Sec - When logging is active this will show the total updates per second for all tags in this DataSet.

Readers - Number of tags being read from this DataSet (each tag will be a Reader).

Writers - Number of tags being written to this DataSet (each logging tag is a Writer).

HDB Files - Number of .hdb2 files in this DataSet.

Offline Files - The number of files in this DataSet with an offline extension because background validation found errors in the file.

Directory - Path where the files are stored for this DataSet.

Rollover - If logging continues this will show the date/time the DataSet is configured to Rollover (create a new file). If logging is stopped and resumed after the time passes then Rollover will occur when logging starts.

Validate - True or False if background validation should run on this DataSet. Controlled by a checkbox on the Historian > Configuration > DataSets tab for each DataSet.

DOWN ARROW

When a DataSet is selected and you click on the Down Arrow you have the following options available to you.

Roll-Over - Forces a file Roll-Over to happen immediately.

Roll-Up - Forces a Roll-Up if it is configured for this DataSet and files are available to act on.

Refresh File List - Refreshes the list of .hdb2 files displayed if a file has been copied into or removed through the directory.

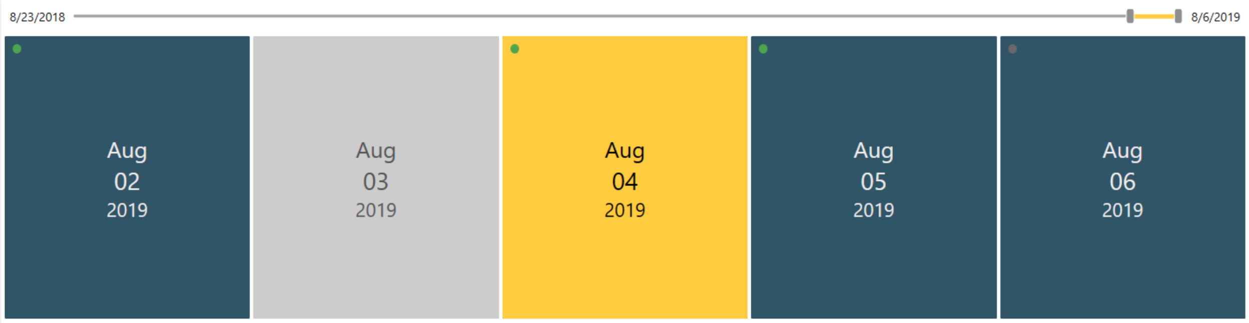
Validate All Files - Performs a validation on all files in the DataSet.

Managing HDB Files

DataSets and their corresponding HDB files can be managed from the “Historian” panel found within the Canary Admin.

Click on the “Historian” panel to see a list of your DataSets.

Double-click on a DataSet to see a list of that DataSets’ HDB files.



The above graphic illustrates several HDB file states.

Files that are “online” and validated (Aug 02 & Aug 05) are colored blue with a green dot in the upper left-hand corner.

Files that are “offline” (Aug 03) are grey.

Files that are yellow have been selected by clicking on them.

Files that are “online” and have not been validated (Aug 06) are colored blue with a grey dot in the upper left-hand corner.

Use the Range Selector at the top of the page to find a series of HDB files from certain dates. Adjusting the grips at either end will increase the visible range of HDB files. Sliding the yellow bar left or right will adjust the overall date range.

Select an HDB file with a single click. You may also select a range of HDB files using either the Shift or Ctrl keys.

The “File” section in the lower left hand corner will now provide information on the selected HDB file.

FILE

Created - Creation date/time of the selected file.

Validated - If the file is validated the date/time is displayed.

Tag Count - Number of tags in the selected file.

File Path - Full path to the location of the selected file.

File Size - Size of the file on the disk.

Embedded Size - Files open for writing will be larger due to allocated space. This will display the size of the file with the extra space released.

Rolled Over - True or False depending on whether the file rolled over to a new file (tag key is copied to the next file) or a new file was created when logging started.

Offline - True or False depending if background validation detected errors in the file.

Status - Current state of the file. A status of 'OK' is normal. A status of '???' means the file was not properly closed. Running the Validation process will properly close the file if possible and take the file offline if not. The Bad status means the HDB file has some type of corruption but has not been validated to take offline.

DOWN ARROW

Different HDB options will appear based on the status of the HDB file you have selected.

Online files allow for the following options:

Validate – Runs the validation on the selected file or files.

Take Offline – Adds the offline extension to the file which will prevent it from being accessed for reading or writing.

Offline files allow for the following options:

Validate – Runs validation without comparing timestamps to adjacent files.

Put Online – Removes the offline extension to allow access for reading and writing.

Recover – To be used when a file is taken offline to fix a validation issue. This option will try to recover any usable data without risking the existing data by automatically executing the following steps. (Caution, this process can create large files requiring extra disk space and can take several minutes to hours to complete depending on the size of the offline file).

1. Dump all the data and tag information to a file.
2. Transpose to another file into the proper time sequence.
3. Import the data into a temporary DataSet.
4. Insert the new file into the current DataSet.
5. Clean up the temporary files created.
6. Rename the offline file with a .old extension.

Delete – Executes the permanent deletion of the HDB file. This cannot be undone.

Managing Tag Records

Individual tags can be managed from the “Historian” panel found within the Canary Admin.

Click on the “Historian” panel to see a list of your DataSets. Double-click the DataSet and then the HDB file for the tag you wish to manage.

A tag list of all available tags will populate in the upper left-hand corner. The search box may be used to quickly find the desired tag. Type any part of a tag name in the field to filter the list below.

Select a tag by clicking on its name.

The most recent raw data values will display in the large window on the right. The “Trend” feature will provide a basic trend window.

Use the Range Selector at the top of the page to see raw values from different time periods within the current HDB file. Sliding the gray grip left or right will adjust your placement in time.

The “Properties” window in the lower left-hand corner will display any metadata associated with the tag.

Additionally, the Down Arrow may be used to reveal additional tag options.

DOWN ARROW

Rename – This option allows you to rename the tag. You may choose to apply the changes to only the current HDB file, all HDB files in the DataSet, or only HDB files sequentially forward or behind the current file.

Data Context

Canary Views and Asset Models

To read data from the Historian, Canary uses the Views Service. Views connects to the Historian and allows for client interaction via Windows Communication Foundation (WCF). Access to the Historian, DataSets, or even individual tags can be limited with Views security parameters. Using Windows Active Directory, individuals or user groups can have inherited or explicit permissions for data access.

You have the ability to create multiple virtualized views of your physical data. This allows for tag regrouping, structure changes, name aliasing, and the organization of similar tags by asset.

Creating Views

To create a new View, click the "Views" panel in the Canary Admin. All current Views will be displayed on the right. By default, you will see a view with the same name as your Historian. This view represents the physical data structure as it is being written to the Canary Historian.

Click the Down Arrow within the "Views" window in the upper left hand corner, and select "Create Asset Model".

The "Create Model" window will appear. Begin by giving your model a name.

Then select the View from which you want to base this model. Note, once a new View has been created, future Views may be based off of it. This allows for a "master" that will automatically update "children".

Enter the DataSet(s) you wish to reference. Multiple DataSets may be entered and should be separated with a semicolon ";" followed by a space.

If you would like to use MetaData to build your new View, indicate so by entering the corresponding fields here. Again, use the semicolon followed by a space.

Click “Next”.

The window at the right side of the screen has now updated to two sections. The top section will be used to build rules using regular expressions. The bottom section will show all the tags that are currently contained within your model and can be expanded by clicking the arrow inside the yellow bar.

Click “Add” to create your first rule.

Choose between a “Model Rule” or an “Asset Rule”. A Model Rule is used to match and then replace tag names or structure. An Asset Rule is used to group like tags into an asset.

If you are unfamiliar with regular expressions or need further assistance, online tutorials can be found. We recommend https://www.regexr.com as a resource.

As you create rules, they will display in the top of the window and information regarding the number of matches, duplicates, instances, exclusions, and response time will populate.

Additionally, the lower portion of the window will update with your new rules. On the right you will see the physical name of the tag as it appears in the Historian. On the left, you will see the modified tag name as it lives in your virtualized View. If you have begun grouping tags into Assets, these will also appear.

Rules may be drug up and down the rule stack by clicking and dragging the rule name. Rules are applied in order from top down.

Additionally, rules may be edited by highlighting the rule and clicking the “Edit” or “Remove” button.

Once an Asset has been created, you may highlight the Asset and click the “Rule Details” button to receive information regarding each Asset, including duplicates and missing tag data on an asset-by-asset level.

Once the View is built to your satisfaction, click “Create” in the red bar at the top of the page. You may now click the “Close” button and browse your new View.

Controlling View Access

Complete access to the Views Service can be managed from the Canary Admin.

Click on “Views” panel and then select “Configuration” from the bottom menu.

Select “Access” from the left menu.

Four windows appear, Allow, Allow (Secure Endpoint), Deny, and Deny (Secure Endpoint).

For each category, you may use the “Add” or “Remove” buttons to grant or deny access to the Views Service for an individual Windows user or group.

You must click “Apply” after adding users for the changes to take effect.

You may wish to limit users/groups to certain Views to better enable their workflow. Beyond just Views, you may also restrain permissions based on Assets, DataSets, and even individual tags. This can be done by clicking on the “Views” panel within the Canary Admin.

At the bottom of the page, select “Security”.

On the far left, select “Settings” and ensure you have checked “Security Enabled”. If not, select it and click “Apply”.

Select “Permissions” in the far left panel.

Select the View you wish to give permission to by clicking it. If there are Assets or DataSets within the View they can be shown by expanding the View.

Select either the View, DataSet, Asset, or tags you wish to give permissions to, and click the “Add” button in the upper right-hand corner. A new window will appear.

Click the “user” button to select a user or group from the Windows user selection dialog. The selected user will be added to the filed and the “OK” button enabled.

Adjust the user’s Access rights. Options include ReadWrite (full access), Read (access limited to viewing only), Write (limited to writing only), and None (user is denied all access).

Click “ok” when finished and then click “Apply” in the red bar at the top of the page.

Adjusting Views Configurations

The Views web service has several connection points depending if the client is local or remote and the type of security required. Remote listening ports have default values but can be changed to allow user defined ports.

Click the “Views” panel in the Canary Admin.

Click “Configuration” at the bottom of the page.

Select the endpoint port you wish to enable or disable, or adjust the port accordingly.

Net.Pipe - Windows (Local Only) - Used when the connection is from a client on the local machine. Windows Authentication uses logon information of the current user.

Net.Pipe - Anonymous (Local Only) - Allows any local user access without credentials.

Net.TCP - Windows - Uses the windows credentials of the client user to make a secure connection through the default port 55234.

Net.TCP - Username - Uses username/password credentials entered by the client to make a secure connection through the default port 55234.

Net.TCP - Secure - Uses the Mirror Credentials from the destination computer to make a secure connection through the default port 55234.

Net.TCP - Anonymous - Allows any remote user access without credentials through the default port 55231.

Https - Username - Uses username/password credentials entered by a web client to make a secure connection through the default port 55232.

Http - Anonymous - Allows any remote web user access without credentials through the default port 55230.

The Canary Calculation Server

The Calculation Server allows you to create calculated tags using tags within the Canary Historian. Creation of calculated tags occurs within the Canary Admin under the Calculation tile.

Once defined, calculated tags are logged to a DataSet in the same manner as any other tag. You must have the Sender Service installed on the server for Calculation Server to function.

Creating Calculated Tags

Begin by launching the Canary Admin and selecting the Calculation panel.

Click 'ADD' in the upper right-hand corner.

Use the 'Name' field to give your calculation a name.

If applying a calculation to a group of assets, select the Virtual View from the 'Asset Path' drop down.

Now select the specific asset from 'Asset Type'.

The 'Meta' field can be used to add properties to the new calculated tag. Formatting should be as follows:

> *PropertyName::PropertyValue*
>
> You can place multiple properties within this cell, just separate each property using a double semicolon.
>
> *PropertyName::PropertyValue;;PropertyName2::PropertyValue2*
>
> Additionally, you can pass existing properties from the source tag by using the following format:
>
> *PropertyName::[SourceTagName].SourceTagPropertyName*

The 'Backfill To' feature allows you to historically backfill calculations. Simply enter a date you wish to backfill to. The 'Date Range' will display the oldest and newest timestamps for any tag used in the calculation. Use the 'Remove

Old Data' if you are editing an existing calculation and wish to write over the previous calculations data. If checked, when you apply the calculation, all previous backfilled data will be updated with the new calculation results.

'Frequency' is configured to determine how often the calculation executes. You can adjust this to occur based on the following:

> Value Change – the calculation only operates when one of the tags contained within it change depending on whether 'Any tag values change' or 'Only when specific tags change' is then selected. Until the calculation has been built, you will not have tags in the list to choose from for the 'Only when specific tags change' feature.
>
> Periodic – The calculation occurs on a set time interval. You customize the interval, time offset, as well as delay. Each input should use a time configuration such as 5s, 5m, 5h where 's' equals seconds, 'm' equals minutes, and 'h' equals hours. To set a calculation to run every hour, simply enter 1h
>
>> Additionally, use the 'Offset' to adjust the trigger time forward or backward from the interval. To run 5 minutes after the hour enter 5m
>>
>> Delay is helpful for giving your tags an opportunity to collect all their values from the field prior to running the calculation. You can delay the calculation from running while without changing the interval or offset. For instance, if you have an hourly calculation but know your network reliability is an issue, delay the calculation by 30 minutes with a 30m input. The calculation will not run until 30 minutes after the hour, but will use the value at the top of the hour when it does run.

Build the Expression

Once the top section is configured, you can begin to build the expression for the calculation. To begin, click the '+' on the right-hand side to populate a new expression field.

The ‘Details’ dropdown allows you to choose how multiple expression fields display.

First, set your variable by entering an alpha character in the first box of the expression field (left of the = sign).

Then, to the right of the equals sign, enter the function, tag name(s), and/or operators to define your expression.

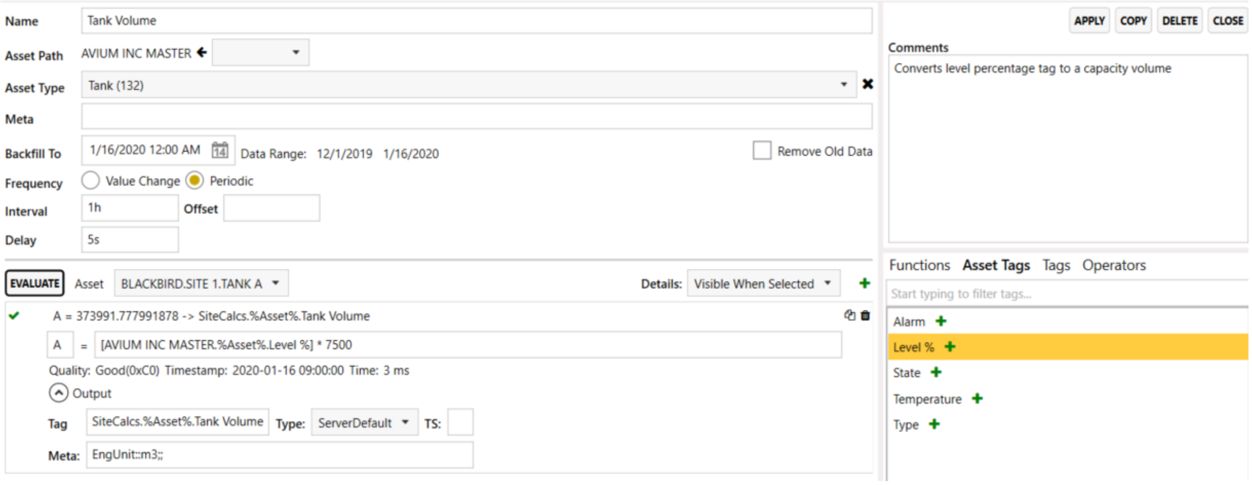
If you have selected an asset model and an asset type, you will have a fourth tab on the right side of your expressions made available to you, ‘Asset Tags’. With your cursor in the expression field, you can click on the green plus sign beside any function, tag, asset tag, or operator in the right window and they will be inserted automatically. Note, when using functions, once highlighted, the function formatting is displayed below the window. Aggregates handle the tag names slightly differently and need to the square brackets removed.

You can create multiple expression lines and can use variables from above expressions in expressions below.

Once your expressions are complete, use the ‘Tag’ field of the final expression to dictate your output tag name and location. You must first have a DataSet created and include it in the tag name path. It is recommended that calculated tags be logged to a unique DataSet that has only calculated tags within it. If you are using asset tags, you must place an asset wildcard, %Asset% between the DataSet name and the calculated tag name. This wildcard will copy the asset instance for each asset and place it in the tag name. This feature allows you to create a single calculation and apply it to every asset instance.

Several examples of screenshots have been included for you to reference when building your calculations.

Basic Asset Calculation Using Asset Tags and Operators



Name: Tank Volume

Asset Path: AVIUM INC MASTER

Asset Type: Tank (132)

Meta:

Backfill To: 1/16/2020 12:00 AM Data Range: 12/1/2019 1/16/2020 □ Remove Old Data

Frequency: ○ Value Change ● Periodic

Interval: 1h Offset:

Delay: 5s

Comments: Converts level percentage tag to a capacity volume

Asset: BLACKBIRD.SITE 1.TANK A Details: Visible When Selected

✓ A = 373991.777991878 -> SiteCalcs.%Asset%.Tank Volume

A = [AVIUM INC MASTER.%Asset%.Level %] * 7500

Quality: Good(0xC0) Timestamp: 2020-01-16 09:00:00 Time: 3 ms

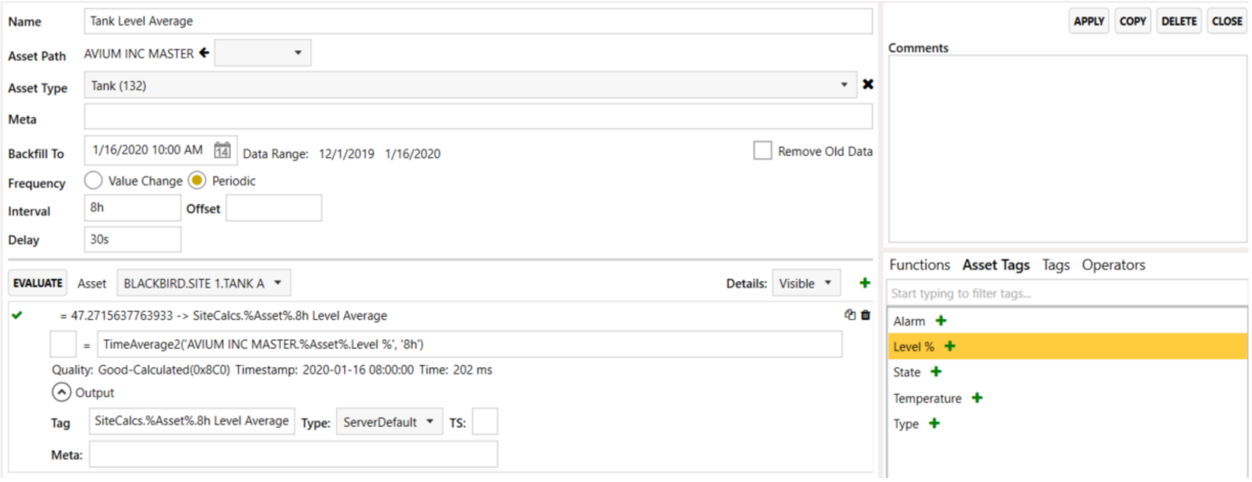
Output

Tag: SiteCalcs.%Asset%.Tank Volume Type: ServerDefault TS:

Meta: EngUnit::m3;;

Here we are simply multiplying the Level % tag by 7500 to convert to a tank volume

Asset Calculation Using Aggregated Function



Name: Tank Level Average

Asset Path: AVIUM INC MASTER

Asset Type: Tank (132)

Meta:

Backfill To: 1/16/2020 10:00 AM Data Range: 12/1/2019 1/16/2020 □ Remove Old Data

Frequency: ○ Value Change ● Periodic

Interval: 8h Offset:

Delay: 30s

Comments:

Asset: BLACKBIRD.SITE 1.TANK A Details: Visible

✓ = 47.2715637763933 -> SiteCalcs.%Asset%.8h Level Average

= TimeAverage2('AVIUM INC MASTER.%Asset%.Level %', '8h')

Quality: Good-Calculated(0x8C0) Timestamp: 2020-01-16 08:00:00 Time: 202 ms

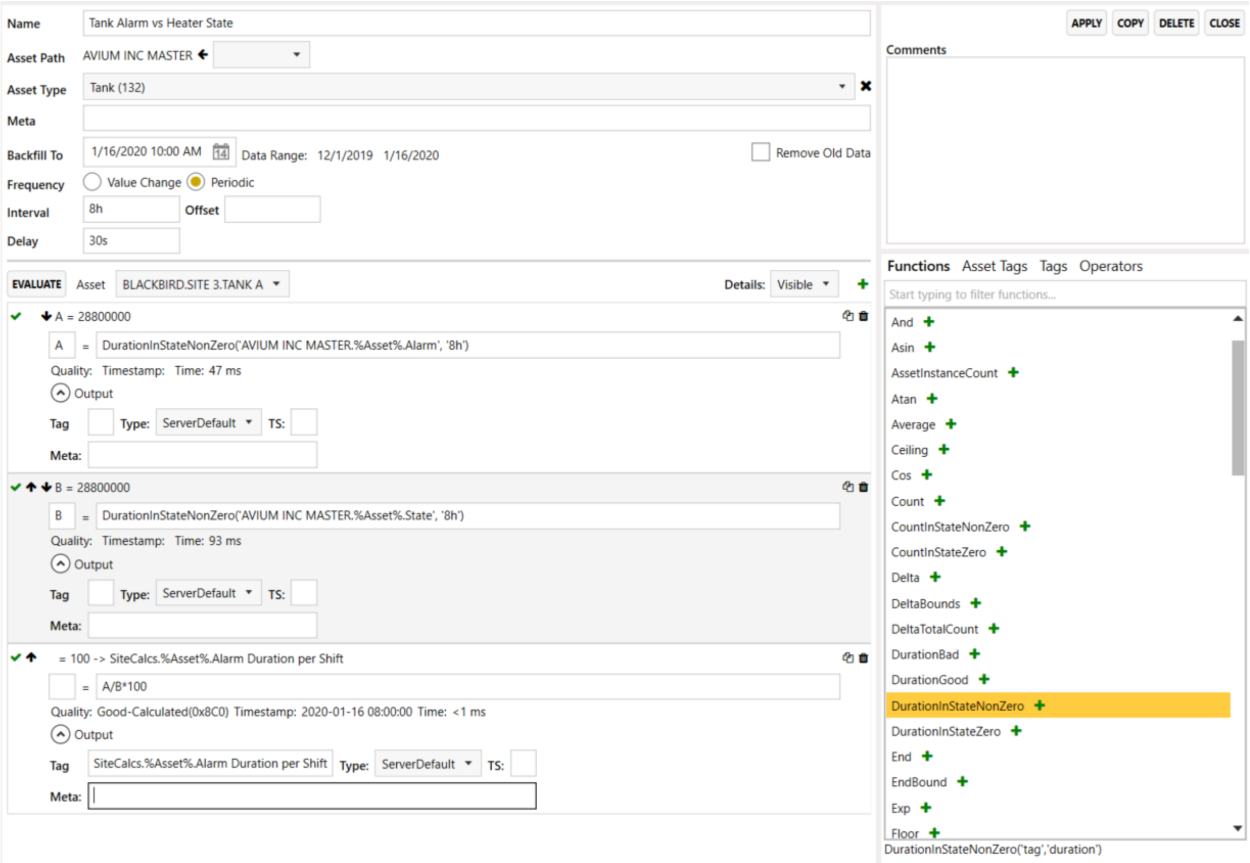
Output

Tag: SiteCalcs.%Asset%.8h Level Average Type: ServerDefault TS:

Meta:

The TimeAverage2 aggregate has been used. The proper format is to remove the square brackets from around the asset tag and wrap it in single quotes. Then place a comma afterwards with a time interval to average within another set of single quotes.

Multiple Variable Calculation



Name: Tank Alarm vs Heater State

Asset Path: AVIUM INC MASTER

Asset Type: Tank (132)

Meta:

Backfill To: 1/16/2020 10:00 AM Data Range: 12/1/2019 1/16/2020

□ Remove Old Data

Frequency: ○ Value Change ● Periodic

Interval: 8h Offset:

Delay: 30s

Asset: BLACKBIRD.SITE 3.TANK A

Details: Visible

✓ A = 28800000

A = DurationInStateNonZero('AVIUM INC MASTER.%Asset%.Alarm', '8h')

Quality: Timestamp: Time: 47 ms

Output

Tag: Type: ServerDefault TS:

Meta:

✓ B = 28800000

B = DurationInStateNonZero('AVIUM INC MASTER.%Asset%.State', '8h')

Quality: Timestamp: Time: 93 ms

Output

Tag: Type: ServerDefault TS:

Meta:

✓ = 100 -> SiteCalcs.%Asset%.Alarm Duration per Shift

= A/B*100

Quality: Good-Calculated(0x8C0) Timestamp: 2020-01-16 08:00:00 Time: <1 ms

Output

Tag: SiteCalcs.%Asset%.Alarm Duration per Shift Type: ServerDefault TS:

Meta:

Comments:

DurationInStateNonZero('tag','duration')

We have established A as the duration of time (in milliseconds) that the Alarm has been in nonzero form. B is the duration of time (in milliseconds) that the State has been in nonzero form. The final expression establishes a percentage by dividing the Alarm time by the State time and multiplying by 100

Once finished, click 'Evaluate' to ensure your calculation is functioning properly. You can then click 'Apply' at the top right of the window.

From the main Calculation window, you will still need to run your calculation by clicking the 'play' button to the left of it's name. You may also use the 'Start All' feature.

You can edit your calculation by clicking the 'pencil' icon.

Calculation Status

When a calculation is highlighted, you can view its status in the yellow 'CALCULATION STATUS' window. When a calculation is first enabled, it may take more than two minutes to see the Calculation Status update and show active calculations running, depending on available server resources.

Creating and Monitoring Events

The Canary Historian includes an Event Service that can be used to monitor the status of tag values and create a database of events around those rules.

To create a new Event, select the “Event” panel in the Canary Admin.

Click “Configuration” from the bottom menu.

Select “New” at the top of the page.

Name your Event and click “add”.

A “Settings” window is now visible.

SETTINGS

Description - A text field describing the event.

Source - Required. Minimum requirement is a View on the local machine. This could be the local Historian or an Asset Model view. If all the tags used in the configuration are from the same DataSet or Asset then this could be added to the source, separated by a period (*View.DataSet*).

Asset Type – Optional. When the source is an Asset Model then this field is a defined Asset Type within the model. An Asset Type will search through all instances of this asset type.

Start Trigger/End Trigger - This is an expression that evaluates to True or False. If the Start Trigger evaluates to TRUE, then this is defined as the beginning of an Event. Then the End Trigger is evaluated on the data to find where an Event ends. If the End Trigger Expression is empty, the reverse of the Start Trigger Expression becomes the End Trigger.

> An expression may look like *[DO] > 6.0*. This expression will evaluate the data stream of the tag looking for the DO tag value to be above 6.0. For tag values the names must be enclosed within square brackets [ ]. The tag is read from the specified View. The full tag name is the Source Path or Asset Instance with the tag name from the expression appended to the end.
>
> Multiple tags can be incorporated into the expression: *[DO] > 6.1 && [Cycle] < 150*

Quality and Property values can also be used in expressions. Functions are used to access these values. An expression to get the quality of the current value may look like *Quality('Tag001')*. Properties are expressed as *Property('Tag001', 'Property Name')*. Since the function is not accessing the value the tag name is enclosed in single quotes ' ' and the parameters are included within the parenthesis ( ).

The AND and OR functions can be used to evaluate multiple expressions. For instance *AND([Tag001], [Tag002], [Tag0003])*, would evaluate to TRUE only when all the tag values do not equal 0.

Expression - The evaluation process is case sensitive

- Logical Operators: or, ||, and, &&
- Relational Operators: =, ==, !=, <>, <, <=, >, >=
- Math Operators: +, -, *, /, %
- Bitwise Operators:& (bitwise and), | (bitwise or), ^(bitwise xor), << (left shift), >>(right shift)
- Unary Operators: !, not, -, ~ (bitwise not)
- Primary Operators: ( )
- Values:Integers, Floating point numbers, Scientific notation, Dates and Times, Booleans, Strings
- Functions: Abs, Acos, Asin, Atan, Ceiling, Cos, Exp, Floor, IEEERemainder, Log, Log10, Max, Min, Pow, Round, Sign, Sin, Sqrt, Tan, Truncate, Quality, Property, AND, OR

Start Time-Shift/End Time-Shift - The start and end time shifts only apply to the way the event is viewed in Axiom. This is how much data will be shown on the trend graph before (start time shift) and after (end time shift) the duration of the event.

Min Duration - If "*00:00:00*" (hh:mm:ss) all events found will be reported. If set, the event must last for the minimum duration before it will be reported. The input can be 00:10:00 for a 10-minute duration, or 10m.

Notify Email - This requires SMTP Server and email addresses to be setup under Messages > Configuration > Email. When a Start Trigger evaluates to TRUE and the Min Duration has elapsed (an event has occurred), an email will be sent to the email recipients.

Backfill Duration - Events are processed on real time data from the time you save the event config until you disable or delete it. Backfilling is an additional feature that if configured (backfill duration > 0) upon save of the config, the software will save any events that have occurred in the past by loading data from the historian from NOW – Backfill duration until NOW.

Properties - Properties allow for additional raw values or the result of calculations (equations) to be stored with the event.

> Name - A text field describing the property.
>
> Equation - The library that we use supports all of the functions in System.Math (https://msdn.microsoft.com/en-us/library/system.math_methods(v=vs.110).aspx). It also supports simple math operators (+, -, *, /). Full documentation on this library can be found here: https://github.com/MichaelAguilar/NCalc. Operators supported: https://github.com/ncalc/ncalc/wiki/Operators. In addition to those, we have also exposed our aggregate function names from the views service such as TimeAverage2().
>
> *Example:*
>
> *Name*: *CombinedAveragePressures*
>
> *Equation*: *TimeAverage2([PressureTag1]) + TimeAverage2([PressureTag2])*
>
> In the property equations, tag names must be surrounded by square brackets. This also applies to the start and end expressions.

Associated Tags - A list of tags related to the event. They may include tags used in the expressions above or additional tags which can be added to Axiom when viewing the event.

Once configuration is complete, click “Apply” in the upper red bar to start your Event.

All Event data is written to an included SQL lite database. When an Event begins to backfill for the first time, allow time for the service to run. Depending on the amount of data, and the time requested for backfill, this could take several minutes to a few hours to complete.

Accessing Events

Once configured, Events may be monitored by clicking the “Events” panel in the Canary Admin.

All Events will show as a tile on the right side of the “Database” screen.

Double-click an Event to see the table, including data regarding which tag or Asset was the Event instance associated with, the start and stop time, total duration, and any defined properties or associated tags.

Additionally, the Canary Excel Add-in can also display this information.

Data Analytics

Creating Axiom Dashboards

Axiom can be run on either a modern web browser or as a Windows application.

The web browser version of Axiom includes a built-in dashboard editor tool than can be used to create rich HTML5 dashboards for display in either the web browser or Windows application versions of Axiom.

This documentation will explain in detail each component of the dashboard editor.

To create a new Axiom application, launch Axiom from a modern web browser. To connect, enter the IP address of the machine where the Axiom Core is installed. If the Axiom Core is installed locally, enter “localhost”.

From the Load Application menu, a new application can be built by selecting the “New” button in the lower right-hand corner.

A New Application window will appear.

By default, three pre-existing templates, Phone, Tablet, and Desktop, are listed as potential application sizes.

To create custom application sizes, enter Custom Width and Custom Height values at the bottom of the screen. If you would like to save this layout as a template, click the Plus icon at the top of the page. Name the screen and click OK. The new template will appear in the list.

The Rotate icon can be used to switch the width and height of the application. This can be useful when designing for either a portrait or landscape layout on a smartphone. Highlight either the template or adjust the Custom Width and Custom Height values as needed. Click the Rotate icon and the width and height will switch.

Either highlight an existing template or adjust the Custom Width and Custom Height values.

Click OK to begin the new application.

Screen Layout

A new Axiom application loads with the Editor menu open on the left side of the screen. Edit mode can be toggled by clicking the “X” to the right of “Editor”. To enter back into Editor mode, click on the Pencil Icon.

The Editor is displayed on the left side of the browser and features four collapsible sections labeled:

• Screens

• Controls

• Tree

• Properties

The middle of the screen contains the workspace. The desired screen dimensions are shown with dashes. If the workspace sizing needs to be adjusted, this can be done during the design process.

Across the top of the Axiom application window are the Editor menu, Application title (defaulted as ‘New”), a Save icon, and three vertical dots referred to as a Vertical Ellipsis.

Save - The application name will show "New" until the application is named and saved by clicking the disk icon.

Vertical Ellipsis - Displays a menu of options including:

• Applications - Returns to the Load Application dialog.

• Preferences - Allows users to change the default settings of charts. o Preference Level - A user with Administrator rights in Axiom can set the default preferences for all users. Under the "User" level, a user can set preferences which will be applied when they connect to Axiom using "Windows" or "User/password" credentials. Note: If a user is connected through the "Anonymous" endpoint, setting preferences will affect any other users connecting through this endpoint.

o Theme - 3 themes are available to alter the look of Axiom; Dark, Light and Canary.

o Chart Preferences – Allows for the customization of new charts when they are created but will not update existing charts. Property preferences include these elements: ▪ Chart

▪ Legend

▪ Cursor

▪ Floating Label

▪ Trend

o Reset - Sets all Preferences back to the factory defaults.

• About - Displays About dialog with version and copyright information.

• Sign Out - Disconnects from the Axiom Core and returns to the login screen.

Axiom Editor

The Axiom Editor consists of four collapsible sections:

• Screens

• Controls

• Tree

• Properties

Each section can be expanded or collapsed by clicking on the arrow to the right of the section name.

The Editor can be toggled in and out of edit mode by clicking the "X" to the right of "Editor". To enter back into Editor mode, click on the Pencil Icon.

Screens

Controls can be placed on a screen to create a dashboard or HMI.

Round Arrow icon - Allows for a one of multiple screens to be selected and then navigated to. Select the desired screen from the list of screens below the arrow. Then, click the arrow to load that screen. The same outcome can be reached by clicking on the screen name as it appears in the navigation list across the top of the display. The first screen will always be the startup screen when the application loads.

Plus - Click the plus icon to add additional screens to the application. Navigation buttons are added automatically with the screen name. The current screen will be highlighted.

Trash - Select a screen to delete and click the trash can to remove it.

Drag grip - Screens can be reordered by clicking the 2 line grip and dragging to a new location.

Screen Properties

When a screen has been physically selected, the following properties are available for customization.

Name - The name used by the application and script to reference this screen.

Width - Total width of the application. Changing one screen width will change all additional screen widths within the application.

Height - Total height of the application. Changing one screen height will change all additional screen heights within the application.

Text - The name used within the navigation bar to access and address this screen.

Asset - Clicking the Asset icon will show the available views. A view is the associated tags to an object (type of equipment or location) defined in Canary Views within the Canary Admin. Once an Asset has been associated with a screen, the Asset icon will appear beside the Vertical Ellipsis. Selecting the Asset Icon will allow for the user to apply one of the linked assets to display on the screen. Note, for this feature to work, when the controls are associated to tags using the SourceTag feature, the tag path must include the corresponding Asset Model.

ShowNavigation - Unchecking this will remove the button at the top of the application. Navigation would then need to be managed through a button on a visible screen or script.

Script - Creates an "OnScreenVisible()" and "OnScreenInvisible()" script stub for this screen and opens the script editor to add event scripting.

Controls

An application is built by arranging controls on each screen and connecting them to tags from the Historian Views service. A list of the available controls includes:

Asset Label
Asset Template
Button
Circular Gauge
Donut Gauge
Grid
Iframe
Image
Label
Linear Gauge
Panel
Paragraph
Spark Chart
Symbol
Trend Graph
Value Box

Each control can be placed by using the drag grip to select the control from the menu and drop it onto the screen. Controls can be placed within Panels, as well as be placed onto of one another to created layers. When selected, each control will allow for sizing adjustment by dragging the square at either the corners or sides. Selecting the control will also allow for the Properties of that control to be customized.

Properties

Each control and screen have a unique set of properties that can be adjusted to allow for customization. A definition of properties for each screen and control are as follows:

Screen

When a screen has been physically selected, the following properties are available for customization.

Name - The name used by the application and script to reference this screen.

Width - Total width of the application. Changing one screen width will change all additional screen widths within the application.

Height - Total height of the application. Changing one screen height will change all additional screen heights within the application.

Text - The name used within the navigation bar to access and address this screen.

Asset - Clicking the Asset icon will show the available views. A view is the associated tags to an object (type of equipment or location) defined in Canary Views within the Canary Admin. Once an Asset has been associated with a screen, the Asset icon will appear beside the Vertical Ellipsis. Selecting the Asset Icon will allow for the user to apply one of the linked assets to display on the screen. Note, for this feature to work, when the controls are associated to tags using the SourceTag feature, the tag path must include the corresponding Asset Model.

ShowNavigation - Unchecking this will remove the button at the top of the application. Navigation would then need to be managed through a button on a visible screen or script.

Script - Creates an "OnScreenVisible()" and "OnScreenInvisible()" script stub for this screen and opens the script editor to add event scripting.

Asset Label

Shows the current instance of an Asset. The Asset, property of the Screen or Panel must be set for the label to display. The example below shows "Basin1" as an asset instance on our machine.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels. Label height is fixed depending on the TextSize.

AssetBranchCount – Determines how many sections of the asset branch to display, working right to left.

TextSize - Text size has 4 options; Small, Normal, Large or ExtraLarge.

HorizontalAlign - Position of the text within the label; Left, Center or Right

Navigate – Links the label to another application or trend chart. Will pass the selected asset to the next application if built on the same model.

NavigateTarget - Choose between launching in a new window or using the existing window.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates an "AssetLabel1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Asset Template

Creates a new panel for each of the indicated asset instances. Drop onto your screen and you are prompted to assign an asset type. A unique panel will appear for each instance of that asset type. Place other objects onto the first panel to duplicate those objects for each asset instance.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels. Button height is fixed.

Height - Height of the control in pixels.

AssetType – The asset that has been assigned to the template

AssetFilter – Allows for input above the asset template that will filter the results based on most current tag values.

TemplateRefreshInterval – Designates how often the template should refresh and update the filter.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "CircularGauge1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Button

Provides OnClick event for navigation and scripting.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels. Button height is fixed.

Text - Text displayed on the face of the button.

Navigate – Links the button to another screen of the existing application, another Axiom application or trend chart. Will pass the selected asset to the next application if built on the same model.

NavigateTarget - Choose between launching in a new window or using the existing window.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Button1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Circular Gauge

Circular gauge with a needle pointing to the value of the source tag. Dual or multiple nested gauges can be displayed by stacking smaller gauges on top of larger ones.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

ScaleLow - The lowest value on the circular scale.

ScaleHigh - The highest value on the circular scale.

MajorInterval - The difference between each value on the scale.

StartAngle - The degrees clockwise from straight up, where the gauge will start.

SweepAngle - The number of degrees out of 360 the gauge will cover.

Reverse - Changes the default rotation of low to high being clockwise to counter clockwise.

Color – Selects the color of the needle

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

TransparentBack… - Changes the gauge background to show the underlying color.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "CircularGauge1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "CircularGauge1_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Donut Gauge

A donut ring showing the value of the source tag in the center. The ring changes color to show the value location between the scale low and high values.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

ScaleLow - The lowest value at the top of the donut.

ScaleHigh - The highest value to complete the donut in a clockwise direction.

Color – Selects the color of the fill.

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "DonutGauge1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "DonutGauge1_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Grid

The Grid control functions as a container for other controls. Controls are added left to right in a row. By selecting the controls in the first row and then using Ctrl+v to copy the controls to new rows a table of controls can be built with row stripping to aid in reading.

In the view below a symbol, label and value box were added to the top row and then copied. A scroll bar will appear if there are more rows than fit the grid.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

AssetType – The asset that has been assigned to the template

AssetFilter – Allows for input above the asset template that will filter the results based on most current tag values.

TemplateRefreshInterval – Designates how often the template should refresh and update the filter.

Columns - A column is required for each control across a row.

ColumnWidths - Column Widths... displays a dialog for setting column widths. By default they are set to Auto which will divide the the grid evenly. The values for each column are a percentage of the grid width.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Grid1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Iframe

Displays web content within the control.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

Source - Enter a URL for a web page.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

Image

A control to display image files.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

ImageSource - Choose Image..., opens a dialog to browse for image files.

HorizontalAlign - Position of the image within the control; Left, Center or Right.

VerticalAlign - Position of the image within the control; Top, Middle or Bottom.

Scale- Determines how the image should fill the space.

- None - the image is not resized.
- Uniform - the aspect ratio of the image is maintained until one direction matches the control size.
- UniformToFill - the aspect ratio of the image is maintained until the both directions of the control are filled.
- Fill - the image is stretched to fill the control.

Repeat - The image is tiled until the control is filled. If the control is filled, then the image will not tile.

- None - image is not tiled.
- RepeatX - image is tiled only horizontally.
- RepeatY - image is tiled only vertically.
- RepeatXY - image is tiled in both directions.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Image1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Label

A single line field for displaying text.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels. Label height is fixed depending on the TextSize.

Text - Text displayed in the label.

TextSize - Text size has 4 options; Small, Normal, Large or ExtraLarge.

HorizontalAlign - Position of the text within the label; Left, Center or Right.

Navigate – Links the button to another screen of the existing application, another Axiom application or trend chart. Will pass the selected asset to the next application if built on the same model.

NavigateTarget - Choose between launching in a new window or using the existing window.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Label1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Linear Gauge

A vertical or horizontal fill gauge, changes color to show the value of the source tag location between the scale low and high values.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

Display – Toggles between Standard, Cutaway, and Fill. Standard holds the fill to a set width, cutaway will fill the entire gauge with the fill line, and fill preserves the scale but uses the rest of the available gauge space as fill.

Orientation - Can be vertical or horizontal. Auto will choose, depending on the width and height.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

ScaleLow - The lowest value on the linear scale.

ScaleHigh - The highest value on the linear scale.

MajorInterval - The difference between each value on the scale.

Color – Selects the color of the fill.

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "LinearGauge1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "LinearGauge1_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Panel

Provides a container for grouping controls. Controls contained in the panel will also be linked to the Asset of the panel. Copying a panel will copy all the controls within it.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

Asset - Clicking the Asset icon will show the available views. A view is the associated tags to an object (type of equipment or location) defined in Canary Views in the Canary Admin.

AssetPickerDisplay – Toggles between a modal display featuring an icon in the menu bar, an inline list of assets, or no selector option.

AssetPickerWidth – Applies to the Inline option for AssetPickerDisplay.

AssetBranchCount – Limits the number of branches from the right to the left of an asset when it is child to a parent asset in the Inline asset picker display.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag. Assigning a tag to a panel allows the designer to set limits which can change the color of the panel.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

Color – Selects the color of the panel.

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

TransparentBack... - Changes the panel background to show the underlying color. Checking this will disable the limit colors.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Panel1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "Panel1_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Paragraph

Provides a multiline text field.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels. Label height is fixed depending on the TextSize and carridge returns (enter).

Text - Text displayed in the paragraph. Clicking "enter" will add a new line to the control.

TextSize - Text size has 4 options; Small, Normal, Large or ExtraLarge.

HorizontalAlign - Position of the text within the label; Left, Center or Right.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Paragraph1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Spark Chart

A simple trend line with configurable duration. Often used with another control showing a current value. Shows the user the past trend of the value. The trend is in "Live Mode" and autoscales to the duration values.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag.

Duration - Clicking the Duration time will launch a time picker window to select a time with the format; d.hh.mm.ss.

PlotColor – The color of the trend. Pick from the color picker or enter hexadecimal values.

Scale Max / Min – Allows the scale to be set to fixed points.

ShowScale – toggles scale visibility on or off.

Visible – Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "SparkChart1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Symbol

Over 3800 symbol graphics are available for designing dashboards and HMI screens. The symbol picker will open when the symbol control is added to the screen or “Choose Symbol” is clicked.

Name – The name used by the application and script to reference this control.

X – Number of pixels from the left edge of the containing panel or screen.

Y – Number of pixels from the top edge of the containing panel or screen.

Width – Width of the control in pixels.

Height – Height of the control in pixels.

SymbolSource- Click "Choose Symbol..." to display the Symbol Picker dialog.

Symbols are separated into groups to help choose from the 1000s of symbols.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trash can will delete the reference to the tag. Assigning a tag to a symbol allows the designer to set limits which can change the color of the symbol.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

Color – Selects the color of the symbol

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

FlipHorizontal - Flips the symbol over from right to left.

FlipVertical - Flips the image upside down.

Rotation- Rotates the symbol clockwise by degrees.

Stretch - Will determine how the symbol fills the control box

- Fill - will stretch the symbol in both direction to fill the size of the control box.
- Uniform - will stretch but maintain the aspect ratio until the symbol reaches either edge of the control box.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "Symbol7_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "Symbol7_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Trend Graph

Adds a self-contained Axiom Trend to the screen. The Trend Graph can be blank for adhoc trending or specify a default Axiom Chart to open.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Height - Height of the control in pixels.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

Value Box

Displays real time values with applied formatting with optional pretext or post text descriptions.

Name - The name used by the application and script to reference this control.

X - Number of pixels from the left edge of the containing panel or screen.

Y - Number of pixels from the top edge of the containing panel or screen.

Width - Width of the control in pixels.

Format - Standard numeric format strings are used to format common numeric types in the Value Box. A pretext can be added before the "{" and a post text can be added after the "}", as in "Flow {0:G} GPH"

TextSize - Text size has four options; Small, Normal, Large, or ExtraLarge.

HorizontalAlign - Position of the text within the label: Left, Center, or Right.

SourceTag - Clicking the trend icon launches the tag browser to select a tag for value input. Clicking the trashcan will delete the reference to the tag. Assigning a tag to a value box allows the designer to set limits which can change the color of the value box.

ValueTransform – Allows for the tag value displayed to be transformed using basic mathematical operators.

StartTime – Allows for time shifting of the data displayed.

Interval – Used along with Aggregate feature, will determine the time interval for the aggregate.

Aggregate – Applies the selected aggregate to the Interval selected and displays the value.

Color – Selects the color of the backfill

Limits – User can input various limits as well as designate colors for each limit

ShowLimits – will either add the limits as a band on the gauge or change the entire fill color to represent state.

Visible - Unchecking this will hide the control. A script would be required to toggle the visibility during runtime.

OnClick - An event that occurs when the control is clicked. Clicking "<>" Creates a "ValueBox1_Click()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

OnValueChange - An event that occurs when the SourceTag value changes. Clicking "<>" Creates a "ValueBox1_ValueChange()" script stub for this control and opens the script editor to add event scripting. Clicking the trash can will delete only the reference to the script, not the script.

Standard Numeric Format Strings

Standard numeric format strings are used to format common numeric types. A standard format string takes the form {A:xx} where A is a single alphabetic character called the format specifier and xx is an optional integer called the precision specifier.

The format specifier must be one of the built-in format characters as outlined below. The precision specifier can range from 0 to 99 and controls the number of significant digits or zeros to the right of a decimal. The format string cannot contain white spaces.

For example, the format string {C:02} will display the number as a currency with three decimal places.

The following describes the standard numeric format strings. Note that the result string produced by these format specifiers is influenced by the settings in the Regional Options control panel. Computers using different settings will generate different result strings.

C – Currency

The number is converted to a string that represents a currency amount. The conversion is controlled by the currency format information of the NumberFormatInfo object used to format the number. The precision specifier indicates the desired number of decimal places. If the precision specifier is omitted, the default currency precision given by the NumberFormatInfo is used.

E or e - Scientific / exponential

The number is converted to a string of the form “-d.ddd…E+ddd” or “-d.ddd…e+ddd”, where each ‘d’ indicates a digit (0-9). The string starts with a minus sign if the number is negative. One digit always precedes the decimal point. The precision specifier indicates the desired number of decimal places. If the precision specifier is omitted, a default of six digits after the decimal point is used. The case of the format specifier indicates whether to prefix the exponent with an ‘E’ or an ‘e’. The exponent always consists of a plus or minus sign and a minimum of three digits. The exponent is padded with zeros to meet this minimum, if required.

F or f - Fixed-point

The number is converted to a string of the form “-ddd.ddd…” where each ‘d’ indicates a digit (0-9). The string starts with a minus sign if the number is negative. The precision specifier indicate the desired number of decimal places. If the precision specifier is omitted, the default numeric precision given by the NumberFormatInfo is used.

G or g - General

The number is converted to the most compact of either fixed-point or scientific notation, depending on the type of the number and whether a precision specifier is present. If the precision specifier is omitted or zero, the type of the number determines the default precision, as indicated by the following list.

Byte or SByte: 3

Int16 or UInt16: 5

Int32 or UInt32: 10

Int64 or UInt64: 19

Single: 7

Double: 15

Decimal: 29

Fixed-point notation is used if the exponent that would result from expressing the number in scientific notation is greater than -5 and less than the precision specifier; otherwise, scientific notation is used. The result contains a decimal point if required and trailing zeros are omitted. If the precision specifier is present and the number of significant digits in the result exceeds the specified precision, then the excess trailing digits are removed by rounding. If scientific notation is used, the exponent in the result is prefixed with ‘E’ if the format specifier is ‘G’, or ‘e’ if the format specifier is ’g’. The exception to this preceding rule is if the number is a Decimal and the precision specifier is omitted. In that case, fixed-point notation is always used and trailing zeros are preserved.

N or n – Number

The number is converted to a string of the form “-d,ddd,ddd.ddd…”, where each ‘d’ indicates a digit (0-9). The string starts with a minus sign if the number is negative. Thousand seperators are inserted between each group of three digits to the left of the decimal point. The precision specifier indicates the desired number of decimal places. If the precision specifier is omitted, the default numeric precision given by the NumberFormatInfo is used.

P or p – Percent

The number is converted to a string that represents a percent as defined by the NumberFormatInfo.PercentNegativePattern property or the NumberFormatInfo.PrecentPositivePattern property. If the number is negative, the string produced is defined by the PercentNegativePattern and starts with a minus sign. The converted number is multiplied by 100 in order to be presented as a percentage. The precision specifier indicated the desired number of decimal places. If the precision specifier is omitted, the default numeric precision given by NumberFormatInfo is used.

Example Outputs

{0:C2} = “$0.00”

{0:C0} = “$0”

{0:E2} = “0.00E+000”

{0:E4} = “0.0000E+000”

{0:F3} = “0.000”

{0:F1} = “0.0”

{0:N2} = “0,000.00”

{0:N0} = “00,000,000”

when value is 0.102

{0.P1} = “10.2%”

{0:P3} = “10.200%”

when value is 10.2

{0:0.0##%’} = “10.2%”

{0:0.000%’} = “10.200%”

When value is 10.250

{0:0.0##’%’} = “10.25%”

Different formatting can be applied to a string based on whether the value is positive, negative, or zero. To produce this behavior, a custom format string can contain up to three sections separated by semicolons:

• One section - The format string applies to all values

• Two sections - The first section applies to positive values and zeros, and the second section applies to negative values. If the number to be formatted is negative, but becomes zero after rounding according to the format in the second section, then the resulting zero is formatted according to the first section.

• Three sections - The first section applies to positive values, the second section applies to negative values, and the third section applies to zeros. The second section might be left empty (by having nothing between the semicolons), in which case the first section applies to all nonzero values. If the number to be formatted is not zero, but becomes zero after rounding according to the format in the first or second section, then the resulting zero is formatted according to the third section.

Note: This type of formatting ignores any preexisting formatting associated with a number when the final value is formatted. For example, negative values are always displayed without a minus sign when section separators are used. If you want the final formatted value to have a minus sign, you should explicitly include the minus sign as part of the custom format specifier. The following example illustrates how section separators can be used to produce formatted strings.

For the following format string: {0:$#,##0.00;(0:$#,##0.00);Zero}

If value is 19.95 the output is “$19.95”

If value is -19.95 the output is “($19.95)”

If value is 0 the output is “Zero”

For the following format string:

{0:RUNNING;;STOPPED}

If value is 0 the output is “STOPPED”

If values is not 0 the output is “RUNNING”

Additional Resources

An in-depth collection of videos can be found on th

Excel Add-in

The primary purpose of the Add-in is to easily extract data from the Canary Historian for use in Excel reports.

Most of the functions used return data as an array. If selecting a cell and inputting a function excel may return multiple cells of data. This can overwrite any data already on the spread sheet. Cells within an array formula cannot be changed. Edits to the formula can also be made by clicking the "fx" button (immediately to the left of the formula). This will show the various input parameters of the specified function.

Functions

There are several functions available in the Excel Add-in. Each function can be found at the top of the worksheet under the “Canary Labs” ribbon bar.

Nearly all functions will require a list of tags, otherwise known as an ‘Input Range’. To create this list of tags, use the function ‘Tag List’.

Tag List

Provides a list of tags in either a column or across a row. The window on the right provides a list of available Views from the dropdown labeled ‘Search Path’. By default, all available tags will appear in the second window. If you select a specific View from the ‘Search Path’ dropdown, the list of tags below will narrow. You can continue to drill down to DataSets or Assets from within the View selected, and likewise, narrow down your tag list.

Choose where to output your list of tags on your document using the ‘Output Location’ feature. Move your cursor to the cell you wish to begin your output in and then click ‘Set’.

Finally, choose whether to list the tag names vertically (in a column) or horizontally (across a row) using the ‘Output Vertically’ selection tool.

Click ‘Apply’ to populate your tag list.

Set Endpoint

Syntax: CanarySetEndpoint() is used to set the endpoint address of the Canary Add-In to point to a Views Web Service. This syntax will populate a box with empty text fields to direct traffic to appropriate historian. The Historian box is for the machine name or IP address where the Views Web Service is installed. The type 'Anonymous' is the default with the least security. 'Windows' uses the current windows user credentials and 'Username' enables the User and Password fields to enter a user with security rights to access this server. The port is set automatically by the 'Type' unless a different port has been specified. User allows username with rights to access this server. Password is used in associated with above user.

> *Note: the endpoint should be set by entering the name or IP address of a server that is running the views web service. If the port number is different from the original 55231, the port should be specified.*

Tag Properties

Purpose: This function gets the properties and property values for a set of tags. There are 7 default properties that a tag will have when it is logged via the standard Canary Logger. These are description, engineering units, high scale, low scale, high limit, low limit and sample interval. However, a tag can have other properties defined by the logging source.

Fx Syntax: Canary3GetTagProperties (TagNames, OutputVertically)

Inputs:
TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is selected. This option is controlled by the Input Range text box. Setting the input range will automatically update the output location.
OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the Canary3GetTagProperties function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.

The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Tag Properties" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Tag Data Context

Purpose: This function gets the data context for a set of tags. The data context information consists of four fields: latest timestamp, latest value, latest quality, and oldest timestamp.

Fx Syntax: Canary3GetTagContext (TagNames, ShowHeaders, DisplayFields, OutputVertically)

Inputs:
TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is

selected. This option is controlled by the Input Range text box.
ShowHeaders: *(True or False).* If True, a heading for each display field will be included in the output. This option is controlled by the Show Headers check box.
Display Fields: If the display field name is checked, it will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options.
OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the Canary3GetTagDataContext function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.

The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Tag Data Context" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Raw Data

Purpose: This function gets the raw data for a set of tags. The raw data is a series of TVQs (timestamp, value, quality). If the data changes a new TVQ is generated. If the tags were polled for updates every one second, but the value only changed every 2 minutes, there would be a TVQ every 2 minutes and not every second. When getting data from multiple tags, the timestamp sequence for each tag will be different.

Fx Syntax: Canary3GetRawData (TagNames, StartDate, StartTime, EndDate, EndTime, SortOrder, ShowHeaders, DisplayFields, OutputVertically)

Inputs:
TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is selected. This option is controlled by the Input Range text box. Setting the input range will automatically update the output location.
StartDate: The start date can be an absolute date (2/15/2019) or a relative time (NOW-5h) or a cell reference ('Sheet6'!$G$12). The absolute date can be typed in or the Calendar control may be used by pressing the down arrow at the end of the text box. A relative time can also be typed. See Relative Time Formats. If a relative time is specified, then the StartTime does not need to be used. To set a cell reference, select the desired cell and press the "Set" button.
StartTime: The start time can be an absolute time (10:15:00 AM) or a cell reference ('Sheet6'!$G$12). To set a cell reference, press the "Set" button with the desired cell selected.
EndDate: The end date and time should be later than the start date and time. The EndDate can be an absolute date (2/15/2019) or a relative time (NOW-5h) or a cell reference ('Sheet6'!$G$12). The absolute date can be typed in or the Calendar control may be used by pressing the down arrow at the end of the text box. A relative time can also be typed. See Relative Time Formats. If a relative time is specified, then the StartTime does not need to be used. To set a cell reference, select the desired cell and press the "Set" button.
EndTime: The end date and time should be later than the start date and time. The end time can be an absolute time (10:15:00 AM) or a cell reference ('Sheet6'!$G$12). To set a cell reference, select the desired cell and press the "Set" button.

SortOrder: The data will be returned in Ascending or Descending order depending on the selection.
ShowHeaders: *(True or False).* If True, a heading for each display field will be included in the output. This option is controlled by the Show Headers check box.
DisplayFields: If the display field name is checked, that field will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options.

- TimeStamp as Text - TVQ TimeStamp is returned as a formated string down to one second resolution (1/1/2017 12:00:00 AM).
- TimeStamp as Number - TVQ TimeStamp is returned as an Excel DateTime number (needs to have Excel formatting applied to make it readable).
- Value if Good - The TVQ value if the quality is good, otherwise blank.
- Value - The TVQ value.
- Data Type - The TVQ value, data type.
- Decimal Quality - The TVQ quality (in decimal).
- Hex Quality - The TVQ quality (in hexadecimal).
- Readable Quality - The TVQ quality - in a readable format - of the different bit meanings.
- Annotation Entered - If an annotation is present at this time, the TimeStamp the annotation was entered by the user.
- Annotation Message - If an annotation is present at this time, the annotation message.
- Annotation By - If an annotation is present at this time, the person who entered the annotation.

OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the Canary3GetRawData function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: If any of the Annotation fields are checked and none of the TVQ fields (value and quality), then the output will just contain the annotations and will not display timestamps of the TVQs.

When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.

The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Raw Data" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Processed Data

Purpose: This function gets the processed data for a set of tags. The processed data is a series of TVQs (timestamp, value, quality) and there will be a TVQ for each aggregate interval between the start time and end time. The timestamp column(s) will be common across all tags.

Fx Syntax: Canary3GetProcessedData (TagNames, StartDate, StartTime, EndDate, EndTime, SortOrder, Aggregate, AggregateInterval, ConsolidateRows, ShowHeaders, DisplayFields, OutputVertically)

Inputs:
TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is selected. This option is controlled by the Input Range text box. Setting the input range will automatically update the output location.
StartDate: The start date can be an absolute date (2/15/2019) or a relative time (NOW-5h) or a cell reference ('Sheet6'!$G$12). The absolute date can be typed in or the Calendar control may be used by pressing the down arrow at the end of the text box. A relative time can also be typed. See Relative Time Formats. If a relative time is specified, then the StartTime does not need to be used. To set a cell reference, select the desired cell and press the "Set" button.
StartTime: The start time can be an absolute time (10:15:00 AM) or a cell reference ('Sheet6'!$G$12). To set a cell reference, press the "Set" button with the desired cell selected.
EndDate: The end date and time should be later than the start date and time. The EndDate can be an absolute date (2/15/2019) or a relative time (NOW-5h) or a cell reference ('Sheet6'!$G$12). The absolute date can be typed in or the Calendar control may be used by pressing the down arrow at the end of the text box. A relative time can also be typed. See Relative Time Formats. If a relative time is specified, then the StartTime does not need to be used. To set a cell reference, select the desired cell and press the "Set" button.
EndTime: The end date and time should be later than the start date and time. The end time can be an absolute time (10:15:00 AM) or a cell reference ('Sheet6'!$G$12). To set a cell reference, select the desired cell and press the "Set" button.
SortOrder: The data will be returned in Ascending or Descending order depending on the selection.
Aggregate: There are currently 43 different aggregates supported by the Views Service. Select the desired aggregate from the drop-down list. This field also supports a cell reference that contains one of the defined aggregates. To set a cell reference, make sure the selection box has focus and the text area is empty, "CTRL-click" on the desired cell and a cell reference will be written into the input parameter.
AggregateInterval: This the time span between TVQs. A time span entry of "00:01:00" or "1m" is one minute. A relative time can also be typed. See Relative Time Formats.
ConsolidateRows: *(True or False).* This option will consolidate rows of data

when there are 10 or more consecutive rows where the data is the same. A special row will be inserted showing the number of rows where the data was exactly the same.
ShowHeaders: *(True or False).* If True, a heading for each display field will be included in the output. This option is controlled by the Show Headers check box.
DisplayFields: If the display field name is checked, that field will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options.

- TimeStamp as Text - TVQ TimeStamp is returned as as formated string down to one second resolution (1/1/2017 12:00:00 AM).
- TimeStamp as Number - TVQ TimeStamp is returned as an Excel DateTime number (needs to have Excel formatting applied to make it readable).
- Value if Good - The TVQ value if the quality is good, otherwise blank.
- Value - The TVQ value.
- Data Type - The TVQ value data type.
- Decimal Quality - The TVQ quality (in decimal).
- Hex Quality - The TVQ quality (in hexadecimal).
- Readable Quality - The TVQ quality - in a readable format - of the different bit meanings.

OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the

Canary3GetProcessedData function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.

The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Processed Data" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Value At

Purpose: This function finds TVQs that meet a set of criteria for a set of tags. If found there will be one TVQ (timestamp, value, quality) for each tag.
Fx Syntax: Canary3GetValueAt (TagNames, StartDate, StartTime, SeekDirection, Expression, ShowHeaders, DisplayFields, OutputVertically)
Inputs:

TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is selected. This option is controlled by the Input Range text box. Setting the input range will automatically update the output location.
StartDate: The start date can be an absolute date (2/15/2019) or a relative time (NOW-5h) or a cell reference ('Sheet6'!$G$12). The absolute date can be typed in or the Calendar control may be used by pressing the down arrow at the end of the text box. A relative time can also be typed. See Relative Time Formats. If a relative time is specified, then the StartTime does not need to be used. To set a cell reference, select the desired cell and press the "Set" button.
StartTime: The start time can be an absolute time (10:15:00 AM) or a cell reference ('Sheet6'!$G$12). To set a cell reference, press the "Set" button with the desired cell selected.

SeekDirection: Select the direction (Forward or Backward) from the drop down list.
Expression: The expression is a powerful search expression. The drop down list will provide some clues as to how to construct an expression. There are four predefined expressions:

- First Value - return first TVQ encountered.
- value > {*number*} - value is greater than a specified number.
- value < {*number*} - value is less than a specified number.
- value >= {*number*} - value is greater than or equal to a specified number.
- value <= {*number*} - value is less than or equal to a specified number.
- value = {*number*} - value is equal to a specified number.
- value <> {*number*} - value is not equal to a specified number.
- quality = {*number*} - quality is equal to a secified number.
- Quality is Good - return first TVQ will a good quality.
- Quality is Bad - return first TVQ with a bad quality.
- Quality is NoData - return first TVQ will a "NoData" quality.

The expression uses two key words: value and quality. A valid expression could be (value > 55.2) or (value < 10.2) or (value = 25). Using the quality keyword an expression might be (quality = 0xC0) or (quality <> 0xC0). The expression will accept both decimal and hexadecimal number (preceded by 0x). The expression will also accept (value > 33.3 && value < 44.4) to find a value that is in a certain range. An expression can also mix both quality and value keywords like (value > 55.2 && quality = 0x2C0). The comparative operators are >, <, >=, <=, =, <>. The '&&' is the AND operator and || is the OR operator. The expression can only be applied to a single TVQ.
ShowHeaders: *(True or False).* If True, a heading for each display field will be included in the output. This option is controlled by the Show Headers check box.
DisplayFields: If the display field name is checked, that field will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options.

- TimeStamp as Text - TVQ TimeStamp is returned as as formated string down to one second resolution (1/1/2017 12:00:00 AM)

- TimeStamp as Number - TVQ TimeStamp is returned a Excel DateTime number (needs to have Excel formatting applied to make it readable)
- Value if Good - The TVQ value if the quality is good, otherwise blank
- Value - The TVQ value
- Data Type - The TVQ value data type
- Decimal Quality - The TVQ quality (in decimal)
- Hex Quality - The TVQ quality (in hexadecimal)
- Readable Quality - The TVQ quality - in a readable format - of the different bit meanings

OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the Canary3GetValueAt function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.
The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Value At" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Last Value

Purpose: This function returns the most recent TVQ for a set of tags.

Fx Syntax: Canary3GetLastValue (TagNames, ShowHeaders, DisplayFields, OutputVertically)
Inputs:
TagNames: A reference to a range of cells containing a list of tag names. The list of tag names can span multiple views. The tag names range is set to the range of cells that are selected when the Ribbon button for this function is selected. This option is controlled by the Input Range text box. Setting the input range will automatically update the output location.
ShowHeaders: *(True or False).* If True, a heading for each display field will be included in the output. This option is controlled by the Show Headers check box.
Display Fields: If the display field name is checked, that field will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options.

- TimeStamp as Text - TVQ TimeStamp is returned as as formated string down to one second resolution (1/1/2017 12:00:00 AM).
- TimeStamp as Number - TVQ TimeStamp is returned a Excel DateTime number (needs to have Excel formatting applied to make it readable).
- Value if Good - The TVQ value if the quality is good, otherwise blank.
- Value - The TVQ value.
- Data Type - The TVQ value data type.
- Decimal Quality - The TVQ quality (in decimal).
- Hex Quality - The TVQ quality (in hexadecimal).
- Readable Quality - The TVQ quality - in a readable format - of the different bit meanings.

OutputVertically: *(True or False).* Output Vertically causes the output rows and columns to be swapped. On the pane there is a check box for this located just above the "Output Location" input.

Output Location: This is the location where the output will be written to. It is the cell in the upper left corner of the result. The number of cells needed for the result will depend on the input parameters. The number of columns and rows returned is dependent on the number of tags and display fields. The default Output Location cell is set to the cell just below the left most cell of the input range (if the input range is horizontal) or just to the right of the top most cell of the input range (if the input range is vertical.)

OK / Apply Buttons: Either button will take the inputs and create the Canary3GetLastValue function with appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Remarks: When using input parameters referencing other cells, if the input cell changes it will automatically update the resulting data.

The Input Pane can be closed at any time by clicking on the X in the upper right corner. To re-open the Pane, click on the Canary Ribbon "Last Value" button. If the cell containing an existing Canary3 function has the focus and the user selects the corresponding function button from the Canary Ribbon, the input controls in the Pane will be set according to the parameters of the Canary3 function.

Find Events

Syntax: Canary2FindEvents (Name, EventType, SearchFrom, SearchTo, SearchMode, SortOrder, Duration, Properties, Source, Notify, DisplayFields) returns the Events based on the search criteria. The event type box will return any event type that matches the specified event type. If no event type is selected, will match on all event types. The event type can be selected from the drop-down list. Search from and search to can be an absolute date or a relative one. If the field is empty, the Search From is defined as the earliest possible time and the Search Toas the latest possible time. An absolute time can by typed or selected via the calendar that is activated by clicking on the down arrow at the end of the field. A relative time can also be typed. The Search Mode controls how the SearchFrom and SearchTo are compared against the time span of the event. The Range is define as the time period between Search From and Search To. Search Mode can be one of the following options:

- Active during Range: Find all events that are active at any time during the range.
- Starts in Range: Find all events that started in the range.
- Ends in Range: Find all events that ended in the range.
- Started & Ended in Range: Find all events that both started and ended in the range.
- In Process: Find all events that are currently in process and have not ended

A maximum of 1000 events will be returned in a single search. Use the Sort Order setting to return descending or ascending Event Start times. Duration returns any event duration that matches the specified condition. Example conditions are <=15 minutes or > 5 hours. The operators are <, <=, >, >=, and =. The length is relative with the unit's options as seconds, minutes, hours and days. The units can be abbreviated to 1 character. If duration is blank it will match on all event durations. This input parameter can also reference a cell. The properties field are defined during the creation of the Event. The Property conditions will return any event that satisfies all property conditions. There can be multiple property conditions, so if all conditions are true for an event it will be returned. To Add a condition, press the Add button below the Properties List. A pop-up dialog will guide you through building a property condition. To remove a condition, select the condition in the list, and press the Remove button. To edit an existing condition, double click on the condition. A more advanced property condition can be constructed by selecting the 'between' operator. This will allow a search for a property value that is greater the X and less than Y. Source will return an event that contains a string. There can be more than one string to match on separated by spaces. If blank will match on all names. This input parameter can also reference a cell. Once the tags have been selected a field containing headers appears. This will have numerous optional information displays to pick from. If the display field name is checked, that field will be included in the output results. The output location will return the inputs in a specified location. The ok/apply buttons create the Canary2FindEvents with the appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

AdHoc Events

Syntax: Canary2SearchForEvents (ViewName, TagPrefixOrAssetType, SearchFrom, SearchTo, StartTriggerExpression, EndTriggerExpression, MinDuration, MaxEvents, EventProperties, SortOrder, SortBy, ShowHeaders, DisplayFields, ShowStats) searches for Events and returns ones matching the search criteria. View text field selects available View from the drop-down list. This list is read from the Views, web service on the machine the Endpoint is set to. Use the Expression base path to see tags used in expression. If searching an asset for an event, apply it in this field. Search from and search to can be an absolute date or a relative one. If the field is empty, the Search From is defined as the earliest possible time and the Search Toas the latest possible time. An absolute time can by typed or selected via the calendar that is activated by clicking on the down arrow at the end of the field. A relative time can also be typed. The start and end trigger expressions evaluate to True or False. If the Start Trigger evaluates to TRUE, then this is defined as the beginning of an event. The End Trigger is evaluated on the data to find where an event ends at. If the End Trigger Expression is empty, the reverse of the Start Trigger Expression becomes the End Trigger. An expression may look like [DO] > 6.0. This expression will evaluate the data stream of the tag (starting at Search From) looking for the DO tag to be above the value of 6.0. Tag names are enclosed within square brackets [ ]. The tag is read from the specified view. The full tag name is the Expression Base Path or Asset Instance with the tag name from the expression appended to the end.

Multiple tags can be incorporated into the expression: [DO] > 6.1 && [Cycle] < 150

Expression The evaluation process is case sensitive

- Logical Operators: or, ||, and, &&
- Relational Operators: =, ==, !=, <>, <, <=, >, >=
- Math Operators: +, -, *, /, %
- Bitwise Operators:& (bitwise and), | (bitwise or), ^(bitwise xor), << (left shift), >>(right shift)
- Unary Operators:!, not, -, ~ (bitwise not)
- Primary Operators: ( )
- Values:Integers, Floating point numbers, Scientific notation, Dates and Times, Booleans, Strings

- Functions: Abs, Acos, Asin, Atan, Ceiling, Cos, Exp,
  Floor, IEEERemainder, Log, Log10, Max, Min, Pow, Round, Sign, Sin, Sqrt, Tan,
  Truncate

If using min event duration and none is selected, all events will be found. If set, the event must last for the minimum duration before it will be reported. The input can be 00:10:00for a 10-minute duration, or 10m. A absolute maximum of 10,000 events will be returned in a single search. The search will stop when the maximum number of events is reached. Use the max. number of events to help limit some of the data. When an event is found, you can define properties that are computed across the range of the event. The property must be given a name and an equation. For example, Avg may be the property name and TimeAverage2([DO]) may be the equation. In this case, when the end of an event is found, it will compute the TimeAverage2 aggregate for the DO tag across the entire time span of the event. The equation functionality is the same as used in the expressions. All available Canary Aggregate functions can be used. An equation of Maximum2([DO])+10.0 will find the maximum value across the interval and then add 10 to it.

A special condition occurs when an Aggregate name contains ActualTime. For example, MaximumActualTime2. In this case, 2 property columns are returned, one for the max value and another for the time when the max value occurred.
<!--[if !supportLineBreakNewLine]-->
<!--[endif]-->

Sort order will display results in Ascending or Descending order. Sort by displays results by fields or properties. If a display field name is checked, that field will be included in the output results. The order of the display field names is the order of the fields in the output result. To rearrange the order, click a field and then press the up/down buttons along the right side. You can also drag and drop fields to rearrange the order. The function parameter is a comma separated string of selected options. If show headers is checked, a heading for display fields will be included in the output. Progress report location outputs cell location where statistics will be shown. This setting is optional. If left blank, no statistics will be shown during the load of ad hoc events.
To set a new cell reference:

- Select the cell and press the 'Set' button beside the field or

- Select the cell, click into the text box, and press CTRL+Insert on the keyboard

The output location will return the inputs in a specified location. The ok/apply buttons create the Canary2SearchForEvents with the appropriate parameters. The function will then retrieve and display the results. The "OK" button will close the Input Pane while the Apply button will leave it open.

Relative Time Format

The format for the relative time is: keyword+-offset+-offset where keyword and offset are as specified in the table below. White space is ignored. The time string must begin with a keyword. Each offset must be preceded by a signed integer that specifies the number and direction of the offset. If the integer preceding the offset is unsigned, the value of the preceding sign is assumed positive (beginning default sign is positive). The keyword refers to the beginning of the specified time period. DAY means the timestamp at the beginning of the current day (00:00 hours, midnight), MONTH means the timestamp at the beginning of the current month, etc.

For example, DAY -1D+7H30M could represent the start time for data request for a daily report beginning at 7:30 in the morning of the previous day (DAY = the first timestamp for today, -1D would make it the first timestamp for yesterday, +7H would take it to 7 a.m. yesterday, +30M would make it 7:30 a.m. yesterday (the + on the last term is carried over from the last term).

Similarly, MO-1D+5h would be 5 a.m. on the last day of the previous month, NOW-1H15M would be an hour and fifteen minutes ago, and YEAR+3MO would be the first timestamp of April 1 this year.

In handling a gap in the calendar (due to different numbers of days in the month, or in the year), when one is adding or subtracting months or years:

- Month: if the answer falls in the gap, it is backed up to the same time of day on the last day of the month.
- Year: if the answer falls in the gap (February 29), it is backed up to the same time of day on February 28.

Note that the above does not hold for cases where one is adding or subtracting weeks or days, but only when adding or subtracting months or years, which may have different numbers of days in them.

Time Intervals can be absolute (00:01:00) or relative (1m). Intervals of 1 hour or greater can be affected by Daylight Savings Time changes. One day in the spring will have 23 hours and one day in the fall will have 25 hours. If an interval is passed as relative, then the timespan is passed to the Views Service as a negative timespan which does special "relative calculations" to account for transition in Daylight Saving Time and months.

- Relative aggregate intervals can use the short, offset notation and imply a single unit if not specified (d = 1d).
- For "relative calculations" to know what time zone the client is in, a coded number is passed in the sub-second portion of the timespan.
- Requesting an aggregate for YEAR-1Y till YEAR with an aggregate interval of MONTH will return a value for the 1st of each month over the previous year.

Note that all keywords and offsets are not case sensitive, but using upper case is a best practice.

Keyword	Description
NOW	Current time
SECOND	The start of the current second
MINUTE	The start of the current minute
HOUR	The start of the current hour
DAY	The start of the current day
WEEK	The start of the current week (Sunday)
MONTH	The start of the current month
YEAR	The start of the current year

OFFSET	Description
S	Offset from time in seconds
M	Offset from time in minutes
H	Offset from time in hours
D	Offset from time in days
W	Offset from time in weeks
MO	Offset from time in months
Y	Offset from time in years

Version Info

Syntax: CanaryVersion() returns versions the web service and client.

Aggregate List

Syntax: Canary2GetAggregates() gives a list of aggregates supported by the Views Web Service. It will list all Aggregates along with the description in the next cell column.

Relative Time Format

The format for the relative time is: keyword+-offset+-offset where keyword and offset are as specified in the table below. White space is ignored. The time string must begin with a keyword. Each offset must be preceded by a signed integer that specifies the number and direction of the offset. If the integer preceding the offset is unsigned, the value of the preceding sign is assumed positive (beginning default sign is positive). The keyword refers to the beginning of the specified time period. DAY means the timestamp at the beginning of the current day (00:00 hours, midnight), MONTH means the timestamp at the beginning of the current month, etc.

For example, DAY -1D+7H30M could represent the start time for data request for a daily report beginning at 7:30 in the morning of the previous day (DAY = the first timestamp for today, -1D would make it the first timestamp for yesterday, +7H would take it to 7 a.m. yesterday, +30M would make it 7:30 a.m. yesterday (the + on the last term is carried over from the last term).

Similarly, MO-1D+5h would be 5 a.m. on the last day of the previous month, NOW-1H15M would be an hour and fifteen minutes ago, and YEAR+3MO would be the first timestamp of April 1 this year.

In handling a gap in the calendar (due to different numbers of days in the month, or in the year), when one is adding or subtracting months or years:

- Month: if the answer falls in the gap, it is backed up to the same time of day on the last day of the month.
- Year: if the answer falls in the gap (February 29), it is backed up to the same time of day on February 28.

Note that the above does not hold for cases where one is adding or subtracting weeks or days, but only when adding or subtracting months or years, which may have different numbers of days in them.

Time Intervals can be absolute (00:01:00) or relative (1m). Intervals of 1 hour or greater can be affected by Daylight Savings Time changes. One day in the spring will have 23 hours and one day in the fall will have 25 hours. If an interval is passed as relative then the timespan is passed to the Views Service as a negative timespan which does special "relative calculations" to account for transition in Daylight Saving Time and months.

- Relative aggregate intervals can use the short, offset notation and imply a single unit if not specified (d = 1d).
- For "relative calculations" to know what time zone the client is in, a coded number is passed in the sub-second portion of the timespan.
- Requesting an aggregate for YEAR-1Y till YEAR with an aggregate interval of MONTH will return a value for the 1st of each month over the previous year.

Note that all keywords and offsets are not case sensitive, but using upper case is a best practice.

Keyword	Description
NOW	Current time
SECOND	The start of the current second
MINUTE	The start of the current minute
HOUR	The start of the current hour
DAY	The start of the current day
WEEK	The start of the current week (Sunday)
MONTH	The start of the current month
YEAR	The start of the current year

OFFSET	Description
S	Offset from time in seconds
M	Offset from time in minutes
H	Offset from time in hours
D	Offset from time in days
W	Offset from time in weeks
MO	Offset from time in months
Y	Offset from time in years

Data Access

Methodology Overview

The Views Web Service provides a general purpose interface to access data from the Canary Enterprise Historian. It is built on the “Windows Communication Foundation” (WCF) technology from Microsoft, providing several communication bindings for connection. With security enabled, access to sensitive data can be restricted to authorized users.

Alhough the Views Web Service communicates with the Enterprise Historian through COM (if local) or DCOM (if remote), Microsoft WCF technology does not use DCOM, so it is “friendly” for IT, providing simplified access through networking firewalls for clients to connect.

The Views Web Service runs on a system with network connections to one or more Canary Historians. On the local computer, a client connects to an “endpoint” like net.tcp://localhost:55231/historianservice, with “localhost” being the name of the computer on the network and 55231 being the port for net.tcp - anonymous (55230 is the default port for http - anonymous). The Web Service communicates to the Canary Enterprise Historian, retrieving the data and then properly formatting the data to communicate to the client.

Web Services are designed to communicate over the internet, so the Views Web Service can connected through an internet URL allowing a client anywhere on the Internet to get Canary historical data, through a simple client application.

Web Service Clients

- Canary Admin - allows a user to browse the historical data, retrieving both raw and processed data.
- Canary Excel Add-in - uses Microsoft Excel to generate powerful reports.
- Axiom
- Browser - a web page app designed for displaying trends or Axiom Views from the Axiom Core on web pages.
- Desktop - a desktop client for displaying trends or Axiom Views from the Axiom Core on PCs.

- Canary ODBC - provides the ability to make SQL queries from the data tables of the Enterprise Historian.
- Historian Mirror - provides customers with the ability to transport realtime and snap shot images of data from a primary historian to a secondary historian.
- Views plugins - plugins provide extended capability to the Views Web Service such as customized tag views or batch capability.

- Asset Model - a customized view of like tags from multiple like machines or processes.
- Events - scans tags in the Historian for events meeting the conditions of an event rule.

Additionally, custom user applications may use the following methods to access data from the Canary Historian.

OPC HDA

By default each Canary Historian includes an OPC HDA server that reads directly from the Historian record. Note, the OPC HDA server does not sit above the Views Service. This means you will not have access to any virtualized Views or Asset Models that you have created.

ODBC Connector

Licensed separately, the ODBC Connector reads data from the Views Service and allows client tools to query the Canary Historian using SQL.

Canary Web API

Licensed separately, the ViewsSDK allows clients to make API calls through the Views Service. Documentation and examples may be found here: https://readapi.canarylabs.com/