CYGNET

CygNet v9.5 Release Notes

Release Date: February 5, 2021

This document describes new features and changes to CygNet Software since the v9.4 release.

For instructions on updating your host, refer to the ***CygNet v9.5 Upgrade Procedure*** document.

Contents

Product Lifecycle 5

CygNet Lifecycle Notice 5

Upgrade Assistance 6

CygNet Documentation 6

Highlights in v9.5 7

CygNet Core 7

Canvas HMI Client 7

CygNet Bridge API 8

CygNet Dispatch 9

CygNet Measurement 9

CygNet EIEs 9

Link Service 9

Well Test 9

Changes in v9.5 10

CygNet Core 10

CygNet Clients — Canvas, Canvas View, and Canvas View Lite 10

Canvas 10

Canvas View 12

Canvas View Lite 12

Alarm Notifier 12

Canvas Controls 13

Canvas Control Events 13

Alarm Grid 14

Button 15

Chart 15

Combo Box 16

CygNet Grid 16

Detail Control 17

Donut 17

Edit Box 18

Heat Map 18

Image Control 19

Linear Gauges (Horizontal and Vertical) 19

Navigator 20

Nested View 20

Object Container 20

Relative Facility Tree 21

Screen and Object 21

Shape Control 22

- Sparkline 22
- Tag Chooser 22
- Text Tool 23
- Tile View 23
- Tree Map 24
- Value Indicator 24
- Canvas Scripting 24
- Find and Replace 25

CygNet Clients — Other 25

- CygNet Client Installer 25
- CygNet Studio 25

CygNet Bridge 26

CygNet Bridge API 26

CygNet Dispatch 26

CygNet Measurement (FMS) 27

CygNet Services 28

- Change Affecting All CygNet Services 28
- Changes Affecting Many CygNet Services 28
- Current Value Services (CVS) 29
- Database Services (DBS) 29
- Device Definition Service (DDS) 30
- Facility Service (FAC) 30
- General Notification Service (GNS) 31
- OPC Interface Service (OPCIS) 31
- Point Service (PNT) 31

Enhanced Alarm Configuration (EAC) 32

EIEs – Communication Devices 32

- MQTT Comm EIE 32
- OPC Comm EIE 33

EIEs – Device Template Files 33

EIEs – Remote Devices 33

- Changes affecting all EIE Remote Devices 33
- Allen Bradley CIP EIE 33
- Amocams 700 EIE 34
- DNP3 Emerson EIE 34
- Emerson ROCPlus EIE 34
- eProd EIE 35
- Lufkin MPC/RPC EIE 35
- Changes Affecting some Modbus EIEs 35
- Modbus Realflo EIE 35

All OPC EIEs (OPC EIE, OPC Lufkin EIE, OPC Weatherford EIE, and OPC Comm EIE) 35

OPC Lufkin EIE 36

OPC Weatherford EIE 36

Totalflow EIE 36

Link Service 37

ODCB Driver 39

Replication 39

Security 40

Utilities 40

CygNet Message Sniffer Lite 40

CygNet Service Migration 40

CygNet ServiceMon Administration 40

DAT to CAPRI and CAPRI to DAT 41

Group Manager 41

MSS Viewer 41

Point Configuration Manager 41

Well Test 41

Product Lifecycle

For more information on the lifecycle of CygNet components, refer to the *CygNet Software Product Lifecycle Matrices* on the Software Support portal (login required).

CygNet Lifecycle Notice

CygNet Software follows a fixed product lifecycle policy and offers three years of support (two years of mainstream support followed by one year of limited support). Please contact your Account Manager or CygNet Sales (via email at CygNet Sales) for more information about your support options.

The following table lists the dates of support for the currently supported versions of CygNet Software.

CygNet Version	Release Date	Limited Support Begins	Limited Support Ends (End of Life)
9.5	February 5, 2021	February 5, 2023	February 5, 2024
9.4	February 28, 2020	February 28, 2022	February 28, 2023
9.3	July 19, 2019	July 19, 2021	July 19, 2022
9.2	December 14, 2018	December 14, 2020	December 14, 2021
9.1	July 20, 2018	July 20, 2020	July 20, 2021

The following CygNet versions have reached lifecycle milestones with the release of CygNet v9.5:

- **CygNet v9.3** will enter Limited Support on July 19, 2021 and will reach its End of Life on July 19, 2022.
- **CygNet v9.2** entered Limited Support on December 14, 2020 and will reach its End of Life on December 14, 2021.
- **CygNet v9.1** is in Limited Support effective July 20, 2020 and will reach its End of Life on July 20, 2021.
- **CygNet v9.0** reached its End of Life on December 19, 2020.

The following Windows operating systems are no longer supported with the release of CygNet v9.5:

- **Windows 7**
- **Windows 8**
- **Windows Server 2008**
- **Windows Server 2008 R2**

Refer to the Microsoft Lifecyle Policy for information about Windows operating system support.

Upgrade Assistance

Upgrade assistance is provided through prepaid professional service hours provided with your annual services subscription or through time-and-materials consulting services. If you need assistance in planning, upgrading, or deploying this release, please contact CygNet Support for more information about these options. Contact CygNet Support at the Software Support portal (login required), via phone 1-866-4CYGNET (1-866-429-4638) or via email at CygNet Support.

CygNet Documentation

The ***CygNet Help*** for CygNet v9.5 is available from https://softwaredocs.weatherford.com/cygnet/95/. Best viewed in Microsoft Edge or Google Chrome browser. Microsoft Internet Explorer 11 is not supported.

Highlights in v9.5

This section highlights some of the major new features and enhancements in the v9.5 release. Please see Changes in v9.5 below for a detailed list of changes. Refer to the *CygNet Help* for user assistance on these enhancements, modifications, and other updates.

CygNet Core

CygNet v9.5 has been enhanced to support 41 additional facility attributes, including 30 new 10-character table attributes, 10 new 80-character text attributes, and one large 300-character text attribute for use as a facility comment. All CygNet applications, clients, user interfaces, utilities, APIs, metadata, and schemas have been modified to reflect the new facility attributes. See the full note here.

A new database retrieval mechanism has been added for all database services to fetch data in a single message, improving speed and application performance. See the full note here.

Performance of filter rule resolution and creation of group node rule-based hierarchies has been improved in CygNet v9.5. See the full note here.

The maximum number of individual Enhanced Alarm Configuration (EAC) expressions per condition has increased to 50 from 20. See all EAC notes here.

Special consideration must be made when replicating between CygNet v9.5 and an older CygNet host. Refer to **Appendix A: Mixed-Mode Hosts** in the ***CygNet v9.5 Upgrade Procedure*** for more information about replicating in a mixed-mode environment and applying a recommended patch to allow an older Facility service and clients to interact with a v9.5 Facility service.

Canvas HMI Client

Several new and enhanced controls, configuration and scripting options, and improved usability features have been implemented in Canvas v9.5. See the full release notes section for more information about each major highlight listed below.

The Canvas applications now support 64-bit operations by default. Canvas, Canvas View, and Canvas View Lite detect the platform where the client is started and will run as 64-bit on a 64-bit machine and 32-bit on a 32-bit machine. See the full note here.

The Canvas applications (Canvas, Canvas View, and Canvas View Lite) now support three user interface color themes: Dark, Light, and Blue. The default theme for all applications is Dark. See the full note here.

A new color configuration option, Auto, has been added to allow screen portability between Canvas clients that use different themes. A screen created in Canvas using one theme will display properly when viewed in an instance of Canvas using a different theme, avoiding display problems such as white text over a very light background, etc. See the full note here.

A runtime Historical Playback feature is now available allowing you to replay historical data in all open screens with controls that show current values. See the full note here.

You can now add and remove controls to a Canvas screen via script at runtime to create dynamic templated screens without affecting time-to-load or screen performance. You can also now script a popup to appear on the screen at the mouse location. See the full notes here and here.

The Canvas screen (and object) now supports a context menu at runtime. From the menu you can copy the screen path, access the screen's defined facility configuration, show the default chart, and add your own custom scripted context menu items. See the full note here.

Custom scripted context menu items are also now supported by many controls. See the full notes here: Alarm Grid, Button, CygNet Grid, Detail, Donut, Image, Linear Gauges (Horizontal and Vertical), Shape, Text Tool, and Value Indicator.

Several controls now support configuration of multi-line tooltips, which can contain static text, dynamic text via tokens, and blank lines. See the full notes here: Button, Detail, Donut, Image, Linear Gauges (Horizontal and Vertical), Shape, Text Tool, and Value Indicator.

Several enhancements have been made to the Chart control, including:

- multiple y-axes are now supported
- configurable quick range buttons which can be used to quickly view predefined windows of time on the chart
- greater control over the buttons that appear on the Chart's runtime toolbar
- default charts can now display trends with historical rollups. Four rollup series are supported (Real Time (no rollup), Calc Min, Calc Max, and Calc Mean). The type and point appearance on a default chart are now configurable.
- See the Chart (and Default Chart) notes here.

You can now enter multi-line text in the Edit Box in run mode. The Vertical Alignment is also configurable: Top, Center, Bottom, or Stretched. See the full note here.

The Image now supports a CygNet-aware image, which can change based on the point state or the alarm condition of the associated point. The control will be hidden if the configured point is in a point state that doesn’t match any configured images for the control. Image sizing enhancements have been implemented allowing you to control the size of the image independently from the size of the control, display a smaller image in one case, and a larger, different image in the case of an alarm. See the full notes here.

The Grid controls now support grid row summaries for point columns. The summary types available are: Sum, Min, Max, MinMax, Count, and Mean. See the full notes here: Alarm Grid and CygNet Grid.

The Object Container control now includes a pagination interface allowing you to page through a large number of objects. See the full note here.

A new navigation control, the Relative Facility Tree, is available for displaying a hierarchical tree of the resolved facilities based on your relative facility definitions. See the full note here.

The Shape control now supports configuration over how the control behaves when it is resized providing greater flexibility when scripting the control. See the full note here.

The Value bounds configuration has been expanded to include Minimum Setpoint and Maximum Setpoint values for the lower and upper value bounds for an associated analog point for range-based controls. See the full note for the Donut, Linear Gauges (Horizontal and Vertical), and the Value Indicator.

The design-time Layers pane has been enhanced to provide greater control over the layers on your Canvas screens. See the full note here.

CygNet Bridge API

New methods have been added to CygNet Bridge API for interacting with CygNet facility attribute records and point configuration records. Methods have been added to support data interaction with the CygNet Flow Measurement Service (FMS). See the notes here.

CygNet Dispatch

A few changes have been made in Dispatch in v9.5: added new FMS configuration file keywords, added a "Node Description" field at Job creation, and enhanced the Job Report template. See the notes here.

CygNet Measurement

Several enhancements have been made to CygNet Measurement features, including additions to allow interoperability with CygNet Bridge API, support for new FMS Batch process variables, additional FMS Service Info items to monitor device information and licensing, enhancements to gas analysis records visibility, expanded liquid data support including data normalization and liquid configuration editing, an added wet flowing conditions recalculation feature in the Configuration control, improved performance and viewing options in the History Graph control, expansion of what data can be included in Missing Data reports, additional Smart Groups filtering, and added support for the CFX 8.5.0 standard for batch and periodic data for the optional features utilizing that format.

CygNet Measurement includes these and many other improvements in the v9.5 release. See the full list of CygNet Measurement release notes here.

CygNet EIEs

Enhancements have been made to multiple CygNet EIEs to add features bolstering functionality, including improvements for OPC server connections and OPC EIEs utilizing Matrikon SCADA Modbus OPC servers, Allen Bradley CIP EIE string support, eProd EI packet sizes, DNP3 support, and more.

Other enhancements, modifications, and fixes have been added to several remote device EIEs and device template files in this release, as described here.

Link Service

Several major enhancements have been made to the CygNet Link service and MQTT publishing support in v9.5, including:

- MQTT optimization by publishing rollup history is now available. You can publish only the data that interest you, and publish less often at longer intervals using historical rollups from the CygNet VHS.
- Link configuration has been simplified in v9.5. We've added a rules engines that allows you to target multiple facility/UDC combinations with filtering rules. You can now publish multiple points by creating custom filter rules, via two new members, facilityFilter and pointFilter, which are supported in the publish object of the CVS endpoint. Wildcards are also supported in the facilityFilter and pointFilter rules configuration.
- We've added improvement to help avoid MQTT topic collisions for duplicated facility IDs, which are not always unique across CygNet services. You can now specify the environment in which Link is running: "Edge" for Link running on a ForeSite Edge device, and "SCADA" for Link running in a SCADA data center.
- Optional point validation has been added to detect missing and disabled points, and other security errors. Strict point validation and strict point value validation rules have been added to provide better error handling.

See the full list of Link release notes here.

Well Test

The CygNet Well Test Module has been updated in the v9.5. See the full list of Well Test release notes here.

Changes in v9.5

This section describes enhancements, modifications, and fixes to existing components in CygNet v9.5.

CygNet Core

The following changes have been made to the core CygNet application in CygNet v9.5.

Enhancements

- Performance of **filter rule** resolution and creation of **group node** rule-based hierarchies has been improved in CygNet v9.5. Rules are used extensively in CygNet, including in the following applications and components: CAS control, Enterprise Operations, Canvas controls, CygNet Studio controls, General Contract Monitor, Group Manager, Link MQTT publisher, Measurement Smart Groups, and VHS Recovery utility. Note that when exporting rules from the rule filter configuration dialog box to an XML file, the order of XML attributes may have changed in some applications, although the core structure is unchanged.

- CygNet v9.5 requires the following minimum version of the **Microsoft Visual C++ Redistributable Package**:

  - Microsoft Visual C++ 2015-2019 Redistributable (x64) - v14.28.29325 or later
  - Microsoft Visual C++ 2015-2019 Redistributable (x86) - v14.28.29325 or later

  Check whether the most current version of the package is already installed in the Control Panel/Programs/Programs and Features. If not, you will need to update to v14.28.29325 or later.

  These redistributable packages are required to run C++ applications that are developed using Visual Studio and link dynamically to Visual C++ libraries. The packages must be installed via the CygNet Client Installer or updated via CygNet Host Updater once installed.

Fix

- Fixed a caching issue where clients encountered an exception when unsubscribing from a point or facility tag.

CygNet Clients — Canvas, Canvas View, and Canvas View Lite

The following changes have been made to the Canvas client applications in CygNet v9.5.

Canvas

Application-Wide Enhancements

- The Canvas clients now support **64-bit** operations by default. In CygNet v9.5 Canvas, Canvas View, and Canvas View Lite detect the platform where the client is started and will run as 64-bit on a 64-bit machine and 32-bit on a 32-bit machine. Running as a 64-bit application significantly improves memory allocation and application performance.

- A runtime **Historical Playback** feature has been added to Canvas to allow you to replay historical data in all open screens with controls that show current values. You can continuously stream historical data, or step incrementally through the data, or jump back to the most current timestamp at any time.

- The **management of the global settings** (fetching data, caching, messaging, warnings) has been enhanced to improve performance of the Canvas applications.

- The Canvas applications (Canvas, Canvas View, and Canvas View Lite) now support three **color themes** for the user interface: Dark, Light, and Blue. Configuration is via settings on the Canvas backstage view. For Canvas you can change the theme at any time while the application is running and the new theme is saved in the Canvas Workspace settings file. For Canvas View and Canvas View Lite the theme is stored in the global settings file (.gsf). These applications must be restarted to pick up the new theme from the .gsf. The default theme for all applications is Dark.

- As part of Canvas' support for **themes**, a new color configuration option has been added, **Auto**. Where applicable, color-related properties will default to this configuration, allowing them to automatically follow the theme of the current client rather than fixing them to the one used when the screen was created. By using **Auto** for the color configuration, a screen developer can work in their preferred theme without affecting the presentation of the final screen when displayed by the user—avoiding display problems such as white text over a very light background, etc.

  Note the following:

  - If a color property already supported a specific color source (e.g. **<Self>** or **Point State**), then **Auto** has been added to that list of options.

  - If a color property did not have a specific source, then a new source property has been added (e.g. **BackgroundColorSource**, **GridColorSource**, **TextColorSource**, etc.) and **Auto** and **<Self>** are now the options.

  - Not all colors used by Canvas controls are theme-specific; some are **static**. Examples include: Chart and Sparkline series colors, Donut and Value Indicator alarm range colors, Heat Map and Tree Map range colors.

  - The following controls are affected by this enhancement:

    Alarm Notifier
    Alarm Grid
    Button
    Chart (background color only)
    Combo Box
    CygNet Grid
    Detail
    Donut
    Heat Map (background color only)
    Linear Gauges (Horizontal and Vertical)
    Object Container
    Screen (background and layout grid colors only)
    Shape
    Sparkline (background color only)
    Text Tool
    Tile View (background and detail colors only)
    Tree Map (background color only)
    Value Indicator

- The **Layers** pane had been enhanced to provide greater control over the layers on your Canvas screen. Interface elements have been added to allow you to:

  - Hide/show all layers or a selected layer
  - Lock/unlock all layers or a selected layer
  - Select all controls on a layer
  - Delete a selected layer
  - Rename a selected layer

- The **blinking interval** for controls that blink when an associated point is in alarm and unacknowledged is now configurable in the Canvas settings in the Backstage view. The controls that blink are: Alarm Grid, Button, CygNet Grid, and the Text Tool. Note that making the blink rate too fast may cause performance issues on your screens.

Application-Wide Fixes

- Fixed an issue with the **Navigation by Facility** feature where a popup message is displayed indicating that the configured facility attribute for the screen path for a facility is blank or otherwise invalid, when the facility navigation mode for the screen or Search Box is "Script event".
- Fixed an issue where opening a screen containing a **custom control** caused an exception. Canvas will now find custom control plugins in any Windows file-system folder accessible to Canvas.
- Fixed an issue with the Canvas **toolbar** where the Display tag chooser button and the Display layers toolbar button remained highlighted after the corresponding panes were closed.
- Fixed an issue with the **mouse cursor** in Script View when deleting an event via the Delete Event prompt where the mouse cursor was permanently set to busy.
- Fixed an issue with Canvas where **pasting** any text in the Design view rendered the screen unusable (unable to save or run). Previously pasted text was ignored or Canvas created a new text tool with the pasted text.
- Fixed an issue when using an **event's hyperlink** for a copied control generated an unexpected error.

Canvas View

Enhancement

- A runtime **Historical Playback** feature has been added to Canvas View to allow you to replay historical data in all open screens with controls that show current values. You can continuously stream historical data, or step incrementally through the data, or jump back to the most current timestamp at any time.

Canvas View Lite

Enhancement

- A runtime **Historical Playback** feature has been added to Canvas View Lite to allow you to replay historical data in all open screens with controls that show current values. You can continuously stream historical data, or step incrementally through the data, or jump back to the most current timestamp at any time.

Alarm Notifier

Enhancement

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Alarm Notifier's background and text color properties. See the full note here.

Canvas Controls

The following changes apply to all or many Canvas controls.

Enhancements

- Several Canvas controls now support **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property. The following controls support tooltips: Button, Detail, Donut, Image, Linear Gauges (Horizontal and Vertical), Shape, Text Tool, and Value Indicator.

- You can now optionally lock a control to the screen to avoid accidental movement via the **Lock** (IsLocked) property. A locked control cannot be moved via click and drag, or nudged with the arrow keys, or resized. Locking is also available on the small Settings pane in design mode.

Modification

- Control names can now include an **underscore**. Valid characters are A-Z, 0-9, and underscore (_). Special characters and spaces are not allowed.

Fixes

- Fixed an issue where an **empty tooltip** caused a control to fail to display. Tooltips are configurable on all single-value controls: Button, Detail, Donut, Image, Linear Gauges (Horizontal and Vertical), Shape, Text Tool, and Value Indicator.

- Fixed an issue where **modifying multiple controls** unexpectedly linked their properties. Previously when attempting to configure a single Point configuration property with multiple controls selected, the individual configuration elements became linked, such that when only a single control was selected and one of the Point configuration properties was modified, the change was applied to all the previously selected controls.

- Fixed an issue where **switching focus** from a property dialog to the screen when multiple controls are selected caused an error and failure to focus on the screen's property pane. Now when switching to the screen from a property dialog, the property pane will update to the screen, and only the screen will have focus.

Canvas Control Events

Fix

- Fixed an issue where some control **events** were inaccessible, could not be added to script, and generated a validation error. The event names were changed to past tense to comply with Microsoft naming conventions. The following controls and events have been changed:
  - Chart - **DataLoadCompleted**
  - Heat Map - **SelectionChanged**
  - Sparkline - **DataLoadCompleted**
  - Tab Control - **TabSelectionChanged**
  - Tag Chooser - **SelectionChanged**
  - Text Tool - **PointConfigurationChanged** and **PointValueChanged**
  - Tree Map - **SelectionChanged**
  - Value Indicator - **PointValueChanged**

Any older screens that use the previous event names (DataLoadComplete, PointConfigurationChange, PointValueChange, SelectionChange, and TabSelectionChange) will continued to work in Canvas v9.4 and later. A **warning** in script in the development environment will notify you that the old event name is obsolete.

Alarm Grid

Enhancements

- Alarm Grid **row summaries** are now available for Point columns. Supported summary types are: Sum, Min, Max, MinMax, Count, and Mean. You can optionally append a label to provide context for the summary row value. Summary rows will be recalculated when a value in a cell is updated. However, recalculation does not occur immediately or for every update, as it is delayed for the duration of the Row summary update rate property setting, allowing for other updates to come in that may affect the recalculated values. Any recalculation requests that come in during the delay will be ignored. No matter how often values change, Canvas will not recalculate the summary row value more often than the set rate. Note that there may be performance implications if the update rate is set too low. Conversely, if it is set too high, the summaries may become out-of-date until the delay has passed. The Export script method has been updated to include row summaries.

- The Alarm Grid now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Alarm Grid's cell background, text, and border color properties. See the full note here.

Fixes

- An Alarm Grid that includes a **facility column type** can now process and display new alarms. Previously a grid with a configured facility column was unable to post new alarms.

- The Alarm Grid will now **populate all columns** for a new alarm. Previously a row for the new alarm would display, showing only point- and facility-based data. Alarm-based columns remained empty for the new alarm record.

- Fixed an issue with the Alarm grid where the first row of the control did not have **focus** after the Focus() method was called.

- Fixed an issue with Alarm Grid **SelectedRows** script property. Previously the Objects.AlarmGrid.SelectedRows was returning a list of CygNet Grid data rows.

- Fixed an issue when **filtering** the Alarm Grid, where the grid failed to update after changing the filter parameters and then refreshing the grid.

- Fixed an issue with the context menu for the Alarm Grid. The context menu options have been changed to the following: **Acknowledge alarm** is disabled if the alarm has been acknowledged already. **Clear alarm** is disabled if the point is still in alarm or it has not yet been acknowledged. **Force clear alarm** ignores those two conditions and is always enabled. If the alarm has already been cleared it no longer appears in the grid. Previously the Alarm Grid popped a confusing error message when attempting to acknowledge an alarm that had already been acknowledged or when clearing an alarm before it had been acknowledged.

Button

Enhancements

- The Button now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Button now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Button's background, text, and border color properties. See the full note here.

Chart

Enhancements

- The Canvas chart now supports **multiple y-axes** and any single y-axis can support multiple series. The user interface for configuring series and y-axes is combined under a single Series Configuration dialog box, as these elements are now logically grouped. Each chart series is automatically assigned to a y-axis. Chart screens made for an older version of Canvas will be converted to be compatible with Canvas v9.5.

- **Default or on-demand charts** now optionally display **trends with historical rollups**. Four rollup trend series will be shown on the default chart for the selected point: Real Time (no rollup), calculated Min, calculated Max, and calculated Mean. The rollup properties are configured in the default chart settings on the screen. Some screen property categories have been reorganized and renamed to improve usability.

- The Canvas chart has been enhanced to allow configuration of the buttons that appear in the chart's toolbar in run mode. A new set of toolbar properties is available in the chart's property pane. In addition to specifying which toolbar buttons are visible, you can now configure any number of **quick range buttons**, which can be used to quickly view predefined windows of time on the chart. The quick range button's time unit, size of the unit, label, and tooltip are configurable. When clicked the button will reset the time range presented in the chart based on the button's configuration. For example, for a button configured for 8 hours, the chart's time range will be set so that it begins eight hours from now with the left edge of the chart representing now. For a chart configured to be in live update mode, the chart will update to show this eight-hour window. The quick range only persists for the current runtime session.

- Several script events have been added to the chart to allow **interaction** between a chart and other charts or controls on a screen:

  - **Live Update Changed** — Called when the chart's live update setting changes
  - **Pan Offset Changed** — Called when the chart's pan offset values change
  - **Zoom Changed** — Called when the chart's zoom values change

- Several configuration properties have been added to the **default or on-demand chart**, allowing a more **customized appearance**. The new properties will be used by each point (in a series) added to the default trend at run-time and include the following:

  - **Series type** (DefaultSeriesType) Options include Bar, Line, LineArea, Spline, SplineArea, Step, and StepArea
  - **Show points** (DefaultShowPoints)
  - **Point size** (DefaultPointSize)
  - **Show point labels** (DefaultShowPointLabels)

  - **Label background color** (DefaultLabelBackgroundColor)
  - **Label connector color** (DefaultLabelConnectorColor)
  - **Label text color** (DefaultLabelForegroundColor)

  Previously, when a point was added to the default chart, the values for these properties were hard-coded and not available for configuration.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Chart's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. The default colors used for a Chart series are drawn from a set palette and are therefore not theme-specific. See the full note here.

Fixes

- Fixed an issue with the Canvas Chart where the **DataLoadCompleted** event fired multiple times during screen load. The event now only fires when historical data has finished loading for all series.

- **Point color** is now displayed correctly on each series in a chart with multiple trend lines. Previously, for a chart with multiple series and each series configured to show points, all series displayed the same color for all points even when different colors were configured.

- A chart series now resolves data for **relative facilities** as expected. Previously a chart with a single series configured to use a relative facility failed to render correctly.

- A chart displaying multiple series now resolves data for **relative facilities** as expected. Previously a chart showing multiple series configured to use relative facilities failed to render correctly. Also, the user interface for configuring relative facilities was defective when trying to configure multiple series with relative facilities in the same session.

Combo Box

Enhancement

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Combo Box's background and text color properties. New properties, **Background color source** (BackgroundColorSource) and **Text color source** (TextColorSource), have been added. See the full note here.

CygNet Grid

Enhancements

- CygNet Grid **row summaries** are now available for Point columns. Supported summary types are: Sum, Min, Max, MinMax, Count, and Mean. You can optionally append a label to provide context for the summary row value. Summary rows will be recalculated when a value in a cell is updated. However, recalculation does not occur immediately or for every update, as it is delayed for the duration of the Row summary update rate property setting, allowing for other updates to come in that may affect the recalculated values. Any recalculation requests that come in during the delay will be ignored. No matter how often values change, Canvas will not recalculate the summary row value more often than the set rate. Note that there may be performance implications if the update rate is set too low. Conversely, if it is set too high, the summaries may become out-of-date until the delay has passed. The Export script method has been updated to include row summaries.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the CygNet Grid's cell background, text, and border color properties. The default color used for a History column's sparkline is drawn from a set palette and is therefore not theme-specific. See the full note here.

- The CygNet Grid now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

Fixes

- Fixed an issue where the CygNet Grid could not **resolve a facility** sourced from the screen.

- Fixed an issue with the CygNet Grid where double-clicking on a non-facility cell failed to return the **associated FacilityTag**. Now the grid will check the type of column and return the appropriate FacilityTag object from either the point tag or facility tag.

Detail Control

Enhancements

- The Detail control now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Detail control now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- As part of Canvas’s support for **themes**, a new color configuration option, **Auto**, has been added to the Detail's background and text color properties. The default color used for a Detail's Sparkline is drawn from a set palette and is therefore not theme-specific. See the full note here.

Fix

- Fixed an issue with the Detail control so that the control and the Canvas object displayed when the control is flipped display the **correct facility**. Previously when a Detail control was flipped to display an object, and the facility sent to the Detail control changed (for example, when sent from a tag chooser or other control), the flipped object did not get updated with the new facility.

Donut

Enhancements

- The **Value bounds configuration** for the Donut has been expanded to include Minimum Setpoint and Maximum Setpoint values for the lower and upper value bounds for an associated analog point. The Minimum and Maximum values are configured on the Analog page of the PNT editor.

- The Donut now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Donut now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Donut's background, text, ring, empty ring, and marker color properties. The default colors used for a Donut's alarm range colors are drawn from a set palette and are therefore not theme-specific. See the full note here.

Fixes

- Fixed a display issue for a Donut with dynamic value bounds configured where the **text color or background color** failed to display the point state color, if configured, and the alarm range ring failed to display, if configured.

- Fixed an issue with the **display of range color** on the Donut. Previously when a point value was above the maximum or below the minimum value of the control, the range color wasn't set, even if configured to do so.

Edit Box

Enhancement

- The Edit Box now supports **multi-line text entry** in run mode only. Two new properties have been added to support this modification:

    - **Accepts return** (AcceptsReturn) – indicates that the edit box will accept the return key to create a new line in the text block in run mode. A scroll bar will be added if the number of lines exceeds the height of the edit box.

    - **Vertical alignment** (VerticalAlignment) – determines the vertical alignment of the text in the edit box. Options are: Top, Center (default), Bottom, and Stretch.

Fix

- Fixed an issue with the Edit Box where the control did not have **focus** after the Focus() method was called.

Heat Map

Enhancement

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Heat Map's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. The default colors used for a Heat Map's range colors are drawn from a set palette and are therefore not theme-specific. See the full note here.

Image Control

Enhancements

- The Image control has been modified to display an image based on the **point state or the alarm condition** of the associated point. The point state of the associated point is evaluated to determine which image to display. A new property Image resolution mode (ImageResolutionMode) has been added to configure this option.

    - If point state mode is selected, all possible point states are included

    - If alarm condition mode is selected, only point states marked explicitly as an alarm condition in the CVSMetadata are included (point states with the alarm condition attribute set to "true")

  Previously the control displayed an image based only on point state. This new behavior mirrors the CygNet Studio Image tool.

- The Image control has been modified to **hide the control** if the configured point is in a point state that doesn't match any configured images for the control. The control will then update to display the configured image if the point enters a state matching the point scheme, point type, and point state. This functionality is similar to the **Hide invalid tag** property (available on single-value controls, the button, and the image), which hides a control if it fails to resolve to a valid or known tag.

- Image sizing properties have been added to allow you to control the size of an image independently from the size of the control, display a smaller image in a nominal case, and a larger, different image in the case of an alarm. A **Size mode** property (ImageSizeMode) has been added so that bitmap and vector images will scale without distortion. **Size mode** options include Auto and Stretch. Additionally read-only **Image height** (ImageHeight) and **Image width** (ImageWidth) properties have been added to display the raw image dimensions for informational purpose only. Other image properties have been renamed. Refer to the ***Canvas Help*** for more information about these properties.

- The Image control now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Image control now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

Linear Gauges (Horizontal and Vertical)

Enhancements

- The **Value bounds configuration** for the Horizontal and Vertical Linear Gauges has been expanded to include Minimum Setpoint and Maximum Setpoint values for the lower and upper value bounds for an associated analog point. The Minimum and Maximum values are configured on the Analog page of the PNT editor.

- The Linear Gauges now support **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- The Linear Gauges now support **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Linear Gauges' background, text, empty bar, and bar color properties. See the full note here.

Navigator

Fixes

- Fixed an issue with the Navigator control where the menu did not launch in **expanded mode**. Previously the menu would display as expected in design mode, but when in run mode the menu displayed in compact mode by default.

- Fixed an issue where **closing a screen** containing a Navigator control caused an exception. A Navigator screen now works as expected.

Nested View

Fixes

- Fixed a **scaling issue** with the Nested View control. Previously when loading a screen into the nested view via script, the objects in the new screen failed to resize properly, even if the horizontal and vertical resize mode of all objects in the child screens were set to Proportional.

- Fixed an issue with the Nested View so that it **better handles empty screen configurations** when its CygNet configuration is changed.

Object Container

Enhancements

- The Object Container now supports a navigation interface to page through large numbers of objects. If **paging** is enabled a paging interface is added to the bottom of the control. A toolbar indicates the total number of pages, with arrows and buttons for paging through the objects in run mode. The number of objects per page is determined by the Object limit property. If paging is disabled a message indicates that the maximum number of objects is reached, and the total number of objects.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Object Container's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. See the full note here.

Relative Facility Tree

Enhancement

- A new navigation control, the **Relative Facility Tree**, has been added to Canvas. This control displays a hierarchical tree of CygNet facilities based on the relative facility settings defined for Canvas. The control can be used as the primary navigational control on your screens to drive other controls by passing in facilities, allowing the creation and implementation of templated screens. The control presents a navigation model that closely resembles field operations without the overhead of a group hierarchy. You can build a similar navigation tree with the Tag Chooser using the Group hierarchy type; however, the Relative Facility Tree goes directly against the Facility service to build its tree, without dealing with Group Service nodes. Another advantage of this control over the Tag Chooser is that the Relative Facility Tree supports three different selection actions (similar to the Search Box): as a simple facility sender to other controls; an action to navigate by facility to other associated screens by script, hyperlinking, or using the Navigator control; or selection via custom script execution.

Screen and Object

Enhancements

- The Canvas screen (and object) now displays a right-click **runtime context menu**. The fixed menu items are:

  - **Copy full path** – Select to copy the full screen path to the clipboard

  - **Facility configuration** – Select to open the screen's defined facility configuration

  - **Show default chart** – Select to navigate to the default chart associated with the screen

  The screen also supports **scripted context menu items** allowing customized screen behavior. Each custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- **Default charts** now optionally display trends with historical **rollups**. Four rollup trend series will be shown on the default chart for the selected point: Real Time (no rollup), calculated Min, calculated Max, and calculated Mean. The rollup properties are configured in the Default Chart settings on the screen. Some screen property categories have been reorganized and renamed to improve usability.

- A method has been added to the Canvas screen, **ShowPopup**, that creates a popup at the mouse location containing an object specified in the parameters. The path can be either a Canvas screen file (.can) or Canvas object file (.cob).

- As part of Canvas’s support for **themes**, a new color configuration option, **Auto**, has been added to the Screen's background and layout grid color properties. New properties, **Background color source** (BackgroundColorSource) and **Grid color source** (GridColorSource), have been added. See the full note here.

Shape Control

Enhancements

- The Shape control now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- The Shape control now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Shape's fill, text, and line color properties. See the full note here.

- The Shape control now exposes the **Stretch** property, allowing configuration over how the control behaves when it is resized, and providing greater flexibility when using the control in a scripted configuration. Four options are available:

  - **None** — The shape preserves its original size.
  - **Fill** — The shape is resized to fill the destination dimensions. The aspect ratio is not preserved. Fill is the default option.
  - **Uniform** — The shape is resized to fit in the destination dimensions while it preserves its native aspect ratio.
  - **UniformToFill** — The shape is resized to fill the destination dimensions while it preserves its native aspect ratio. If the aspect ratio of the destination rectangle differs from the source, the shape is clipped to fit in the destination dimensions.

Sparkline

Enhancement

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Sparkline's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. The default color used for a Sparkline is drawn from a set palette and is therefore not theme-specific. See the full note here.

Fix

- Fixed an issue with the Sparkline where the **DataLoadCompleted** event fired multiple times during screen load. The event now only fires when historical data has finished loading for all series.

Tag Chooser

Enhancement

- An **Ignore UDCs** (IgnoreUdc) property has been added to specify a comma-separated list of UDCs to hide when displaying points in the tag chooser.

Fixes

- Fixed an issue with the Tag Chooser in pane mode where the **pane failed to persist** and display the configured hierarchical settings (facilities and points) from session to session. Previously after restarting Canvas, the Tag Chooser pane area was empty. Only the SiteService remained specified in the configuration area; all other metadata information (facility filter and facility levels) failed to persist.

- Fixed a **performance** issue with the Tag Chooser. Previously the control was performing unnecessary service status checks when initializing.

Text Tool

Enhancements

- The Text Tool now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Text Tool now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- You can now configure the width of a text tool's border with the **Border width** (BorderWidth) property.

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Text Tool's background, text, and border color properties. See the full note here.

Fixes

- Fixed an issue with the Text Tool's **context menu** not displaying when running a screen using the legacy run mode.

- Fixed a bug with the Text Tool **border not blinking** when it is configured to do so.

- Fixed an issue with Canvas where **pasting** any text in the Design view rendered the screen unusable (unable to save or run). Previously pasted text was ignored or Canvas created a new text tool with the pasted text.

- Fixed an issue where the Text Tool's **context menu** did not display properly.

Tile View

Enhancement

- As part of Canvas's support for **themes**, a new color configuration option, **Auto**, has been added to the Tile View's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. Additionally, the **Auto** option has been added to the Tile View's embedded Detail control's background and text color properties. The default color used for the embedded Detail control's Sparkline is drawn from a set palette and is therefore not theme-specific. See the full note here.

Tree Map

Enhancement

- As part of Canvas’s support for **themes**, a new color configuration option, **Auto**, has been added to the Tree Map's background color property. A new property, **Background color source** (BackgroundColorSource), has been added. The default colors used for a Tree Map's range colors are drawn from a set palette and are therefore not theme-specific. See the full note here.

Value Indicator

Enhancements

- The **Value bounds configuration** for the Value Indicator has been expanded to include Minimum Setpoint and Maximum Setpoint values for the lower and upper value bounds for an associated analog point. The Minimum and Maximum values are configured on the Analog page of the PNT editor.

- The Value Indicator now supports **scripted context menu items**, in addition to the standard context menu items that display in run mode, allowing customized screen behavior. A custom context menu item will call a **Custom Context Menu Action** event into script along with other relevant information in the context of the click, for example, facility or point tag. The custom context menu options will appear at the bottom of the fixed context menu items, separated by a line. You can also add a separator line between custom menu items, and nested items are supported.

- The Value Indicator now supports **multi-line tooltips**. Type any text, select tokens, and press Enter to separate the lines. If more than five lines are entered, a scroll bar will appear in design mode. The scroll bar does not display in run mode. Multi-line tooltips are also supported in Style Sheets via the **TooltipTokenizedString** property.

- As part of Canvas’s support for **themes**, a new color configuration option, **Auto**, has been added to the Value Indicator's background, text, marker border, and marker fill color properties. The default colors used for a Value Indicator's alarm range colors are drawn from a set palette and are therefore not theme-specific. See the full note here.

Fix

- Fixed an issue with the **display of range color** on the Value Indicator. Previously when a point value was above the maximum or below the minimum value of the control, the range color wasn't set, even if configured to do so.

Canvas Scripting

Enhancement

- You can now easily add (and remove) controls to a screen via script at runtime. Two new methods (Screen.**AddControl** and Screen.**RemoveControl**) are available to support this feature. Dynamic screen creation can help improve screen templating, and limit screen size and time-to-load.
  - By adding scripted controls you can take screen templating to a new level via dynamic screens. For example, you could create a well-pad screen that queries facilities and facility attributes and builds itself on-the-fly.
  - You could use custom annotations to build a dynamic screen without increasing screen size or time-to-load. For example, you could create a dense piping and instrumentation diagram (P&ID) type screen with hundreds of text tools that creates (and removes) additional visual elements for alarm indication, without doubling or tripling the number of controls on the screen.

Fix

- Fixed an issue when **renaming similarly named controls** in script. Previously, when renaming a control such that the new name matches a substring of an existing control, then the original control rename will be processed without issue, but the additional control will have its event handler and corresponding implementation modified, breaking the relationship between the control and its corresponding region.

Find and Replace

Fix

- Fixed **Find and Replace** for pre-v9.4 Canvas files. String replacement is now possible in older files.

CygNet Clients — Other

The following changes have been made to the following CygNet client applications in CygNet v9.5.

CygNet Client Installer

Modification

- Removed the warning note about the **GMR** product (discontinued effective with the CygNet v8.5.0 release) that was previously displayed when using the CygNet Client Installer (CInstall.exe) to launch CygNet applications and utilities.

CygNet Studio

Enhancements

- Enhanced the capabilities of the **CygNet Studio screen export** from .csf file to .xml, to include additional properties for a variety of tools and controls. The command-line format: CStudio.exe /xml<myscreen>.csf is used to yield an export file in the format: <myscreen>.xml. Impacted screen elements include TheFrame/TheView, and multiple objects including the Trend, Button, Check Button, Combo Box, Edit Box, Image, Radio Button, Box, Ellipse, Line, Text, and Linear Gauge CygNet Studio tools, and the CxHmiCas3 (Alarm), CxHmiGenericGrid (Generic Grid), CxGrpTree (Group Tree), and Nested View ActiveX controls.

- Enhanced the Group Navigation Bar control in CygNet Studio so that when defining Group Hierarchy levels, a "**Display Text**" selector is now provided and attribute selection options for label usage have been expanded.

Fixes

- Fixed an issue with the Radio Button tool in CygNet Studio, so that the **ListItems** property now functions as expected in cases where the ListItems value equals zero or is not present. Previously this situation could cause the client to fail.

- Updated the ***CygNet Help*** to describe the "obsolete" status of the **TimeStamp2** property for both the RtREx and SRtREx objects used in CxCvs libraries.

- Updated the ***CygNet Help*** to describe the **GetRowIndexAdjustedForGroups** method as "obsolete" for the **Generic Grid** control in CygNet Studio, and removed the method from the table of available methods for the control.

- Fixed formatting strings when displaying **PNT floating point attribute values** in some client applications. Previously, some attributes contained a decimal even if the value was not a decimal value.

CygNet Bridge

Enhancements

- For users of CygNet Bridge, modified the **Feature** selection page of the installer to allow specifying an alternative Port binding value.

- For users of CygNet Bridge, enhanced **validation of user-entry fields** in the installer to help streamline the process.

CygNet Bridge API

Enhancements

- Added new CygNet Bridge API methods for interacting with CygNet **facility attribute records**. Methods provided with the Base license type now include the ability to create, update, or delete facility attribute records.

- Added new CygNet Bridge API methods for interacting with CygNet **point configuration records**. Methods provided with the Base license type now include the ability to retrieve, create, update, or delete point configuration records.

- Enhanced the CygNet Bridge APIs to include a variety of new methods to support **data interaction** with the CygNet Flow Measurement Service (FMS) including numerous facilities and points operations.

- Refer to the **CygNet Bridge API Help Page** in the main ***CygNet Help*** for offline user assistance on these new methods.

CygNet Dispatch

The following changes have been made to CygNet Dispatch in CygNet v9.5.

Enhancements

- For users of CygNet Dispatch, added **new FMS configuration file keywords** (in a "Dispatch" section) to allow administrators to manage the amount of jobs data transferred in a single CygNet Dispatch synchronization action.

- For users of CygNet Dispatch, added a "**Node Description**" field at Job creation, to provide additional job sorting mechanisms within the system.

- For users of CygNet Dispatch, enhanced the **Job Report template**, by adding a Cell Name field to the report definition to accommodate running external queries.

CygNet Measurement (FMS)

The following changes have been made to CygNet Measurement in CygNet v9.5.

Enhancements

- Enhanced the **Export: Normalization View Data CSV** command to support normalized liquid device data.
- Updated the optional **Export: Flow-Cal CFX Data** command to support exports to CFX file version 8.5.0, for batch and periodic data.
- Expanded the **Import: History CSV** command to support liquid device Nodes with periodic history data.
- Added new **FMS Info Items** to the FMS service, so that CygNet Measurement administrators can monitor current status values for ensuring licensing compliance. The new info items provided can be used to track the numbers of active devices, inactive devices, meters licensed, and the percentage of licensed meters being used.
- Enhanced the CygNet Bridge APIs to include a variety of new methods to support **data interaction** with the CygNet Flow Measurement Service (FMS) including numerous facilities and points operations.
- Added a new option to **Normalization View** definitions, to support only complete spans containing no missing data.
- Added support for new **Batch data items** to define several custom numeric values, to allow seamless display of custom items in the Raw Data control and Batch reports, and to export the values via the optional Export: Flow-Cal Data CFX command.
- Added the ability in the Configuration control to view complete **Gas Analysis records**, regardless of status value. Previously some records were not displayed, preventing visibility into the full range of records.
- Added the ability to the Configuration control to **recalculate gas data** based on wet flowing conditions, so that relative density, heating value volume, and history volume/energy/mass outputs can be adjusted to account for water content., regardless of status value. Previously some records were not displayed, preventing visibility into the full range of records.
- Improved performance of the **History Graph** control, and implemented an automatic "daily" record span view for device data requests exceeding 31 days.
- For users of CygNet Dispatch, modified the **Event Review** dialog that appears when reviewing calibration events in the Jobs control, to allow screen resizing and improve button access.
- Modified the **Generate Smart Groups** command to allow the option of generating groups based on specific smart group definitions.
- Added the ability to the **Gas Analysis Report** to optionally filter reported results by the 'Status' value of the reported gas analysis records.
- Modified the **Missing Data Report** to be able to include estimated and unavailable data in published results.

Fixes

- Fixed an issue with **configuration validation** so that, in circumstances where the Orifice Plate Material type is not specified in the configuration transaction, the corresponding coefficient value (Orifice Thermal Expansion Coefficient) is correctly used for validation instead.

- Fixed an issue in the **Balance** control so that, when viewing historical data for periodic liquid meter Nodes, Gas Equivalent Volume (GEV) calculation results are shown correctly whether configuration data is present for the time period or not.

- Fixed an issue with the **Close Period** control to display a more accurate informational message, and to properly recognize the current date value, when attempting to close a period in circumstances such as the control remaining open for multiple days.

- Fixed an issue in the **Dashboard** control for v9.4, so that service is reliable when performing incremental updates. Previously a crash could occur in certain circumstances.

- Fixed a keyboard shortcut in the **Jobs** control for the "Precalculate calibration data" command.

- Fixed an issue with the **Update Normalization Views** command, so that no change occurs when ID-based update requests find no records requiring normalization. Previously the index could be incorrectly reset to zero in such circumstances.

- Fixed an issue with the optional **Export: Flow-Cal Data CFX** command so that mapped Latitude process variable values are now successfully included in the exported records.

- Fixed a mapping issue with the optional **Export: Flow-Cal CFX Data** command, for exports to CFX file versions 8.0.6 and 8.3.1, so that Relative Density events appear correctly.

- Fixed an issue with both the optional **Export: Flow-Cal Data CFX** and **Export: Flow-Cal Transaction Queue** commands so that, for cases containing split records where the record count differs between periodic metering and quality data types, the data is processed successfully and an explanatory error message is logged. Previously the export was not successful in such circumstances.

- Fixed an issue with the **Gas Analysis report** so that all gas analysis records found within the requested date range are correctly included in the published report, and no report is generated when no gas analysis records are present. Previously only a single record may have been reported when multiple records were present, and null values reported unnecessarily.

CygNet Services

The following changes have been made to CygNet services in CygNet v9.5.

Change Affecting All CygNet Services

Fix

- Fixed an issue where **services failed to shutdown** gracefully if the send queue reaches capacity. Previously this could be observed if the Active messages counter permanently sticks above 1 (Info Item: SRVNET_ACT_MSGS (Act Msg), UDC: SVMAMSG).

Changes Affecting Many CygNet Services

Fix

- Fixed a **memory leak** in all services except the ARS, CAS, and VHS, which improves message handling and service performance. Previously the leak behavior exhibited in services whose Active Messages counter exceeded 10 in bursts (Info Item: SRVNET_ACT_MSGS (Act Msg), UDC: SVMAMSG).

Current Value Services (CVS)

For notes about the Enhanced Alarm Configuration (EAC) feature, see Enhanced Alarm Configuration.

Modifications

- Modified the way the Current Value Services (CVS) enforce affected configurable bit calculation type behavior when the **Unreliable** bit changes. Previously the CVSs did not perform any config-bit calculations when the only change to the real-time record was that the Unreliable bit changed from unset to set. For example, when the OPCIS loses connectivity to its configured OPC Server, it will mark all of its points' statuses as Unreliable without changing the points' values. Prior to this fix, any configurable bits that referenced the Unreliable bit were skipped. Now, those configurable bits will be correctly processed.

  Additionally, the following alarm calculation types will ignore any points marked as Unreliable, even if the point value changed, and if the value was Unreliable because it was clipped by the value range determined by the Minimum or Maximum values:

  - CHANGE_VAL_PCT
  - CHANGE_VAL_POS_PCT
  - CHANGE_VAL_NEG_PCT
  - CREEP_AVG
  - CREEP_VAL
  - CREEP_VAL_POS
  - CREEP_VAL_NEG
  - DELTA_OVER_TIME
  - DELTA_OVER_TIME_POS
  - DELTA_OVER_TIME_NEG

  CREEP_VAL_POS and CREEP_VAL_NEG are now documented in the ***CygNet Help***.

- The **Change Rate** (CHANGE_RATE) alarm calculation has been modified to set the alarm bit when the maximum allowed number of changes occurs, even if the full time interval has not completed. Previously the alarm calculation was not triggered correctly when a longer time interval was defined.

Fixes

- Fixed an issue preventing some **hyperpoint scripts** from loading when the script is abnormally terminated.

- Fixed an issue for a CVS point where the evaluation of a **configurable point status bit** failed when the point was previously set to Unreliable. Configurable point states will now be evaluated when a point is also set to Unreliable.

Database Services (DBS)

Enhancements

- When replicating between any two CygNet systems and a **mismatch in the database version** between a replicating DBS and source DBS is detected, the replicating service will now requeue all updated records for a later resync to ensure consistency in data records. A new info item, **REPL_DBS_RECHECK_KEY**, (Nice Name: Repl DDL recheck keys) has been added to track the number of keys that were modified in the source database if the source database has a larger database version number (DDL_VERSION_NUM info item). After normal replication up-to-date checks, the replicating service will recheck those records to make sure there is no mismatch.

  This enhancement is related to changes in CygNet v9.5 to support additional facility attributes and updates to the Facility service database definition (FAC DDL), but applies to all DBS services when replicating between one CygNet version and another.

- A new **database** definition (DDL) **retrieval mechanism** has been added for all database services to allow client-side components that support custom queries to fetch database table and field information in a single message, improving data retrieval speed and application performance.

Device Definition Service (DDS)

Enhancement

- A **Quality** column has been added to the DDS Configuration History Header, and a new DDS index containing the Quality field, as well as the first 17 characters of the 64-character 'status_message'. This summary field indicates the OPC quality status of the DEIDs in the data-group transaction with four possible values returned:

  - G — All Good
  - B — All Bad
  - P — Partially Good
  - U — Unknown

  The DDS Transaction Viewer (DDSViewer.exe) and the transaction XML have been modified to include the Quality field.

  The quality field is visible only for the CygNet OPC EIEs (OPC EIE, OPC Lufkin EIE, and OPC Weatherford EIE). All other EIEs will show *Unknown* for data-group transaction quality. See OPC EIEs for other related changes in v9.5.

  After upgrading to v9.5 the DDS database file *must* be reindexed. Refer to the "Reindex the DDS" step in the ***CygNet v9.5 Upgrade Procedure*** for instructions.

Facility Service (FAC)

Enhancements

- CygNet v9.5 has been enhanced to support many **more facility attributes**. The following attributes have been added to the Facility service:

  - **Table attributes** — we have doubled the number of table-driven attributes, adding 30 new 10-character table attributes (300 bytes) for a total of 60 table attributes

  - **Text attributes** — we have added support for more blob path values, adding 10 new 80-character free-form text attributes (800 bytes) for a total of 42 text attributes

  - **Comment attribute** — we have added one large 300-character text attribute for use as a simple facility comment

  All CygNet applications, clients, user interfaces, utilities, APIs, metadata, and schemas have been modified to reflect the new facility attributes.

  The Facility service database definition (FAC DDL) has been updated. After upgrading to v9.5, and the FAC.exe runs for the first time, the Facility service database is immediately "converted" and an earlier FAC.exe will not be able to read the modified database. See Replication for a related note about replicating between a v9.5 system and a CygNet host running and earlier version.

- We increased the maximum record length for a CygNet DBS record to support the increase in the number of facility attributes. As a result **increased memory usage** may occur requiring the allocation of additional system resources to the host where the FAC service is running. In testing we have observed that a v9.5 FAC service may use 20MB to 40MB more bytes than a v9.4 FAC service. We recommend monitoring the PRIVATE_BYTES and VIRT_BYTES performance info items to determine whether an increase in memory is required.

Modification

- The Facility service database definition has been updated to support additional facility attributes. Please note the following:
  - A .compact file generated from a v9.5+ **DatToCapri** utility against a v9.5+ Facility service database cannot be restored with an earlier version of **CapriToDat**.
  - A v9.4 and earlier **DatToCapri** utility cannot read a v9.5+ Facility service database.

Fix

- The Facility service now **fully synchronizes data** with the Point service and the Device Definition service. Previously, if the Facility service isn't running when new points or devices are created or updated, the Facility service will never fully synchronize with those facility references, unless they are updated again.

General Notification Service (GNS)

Fix

- When performing a **backup of the GNS database**, the ELS Event record now displays an appropriate message and category. Previously the ELS record indicated that there was a Category of ERROR even if the backup was reported as successful. Fixed an issue with the Close Period control to display a more accurate informational message, and to properly recognize the current date value, when attempting to close a period in circumstances such as the control remaining open for multiple days.

OPC Interface Service (OPCIS)

Fix

- Fixes a 40-byte **memory leak** with each health check in the OPC Interface Service. Shut-down time for services with many groups has also been significantly reduced.

Point Service (PNT)

For notes about the Enhanced Alarm Configuration (EAC) feature, see Enhanced Alarm Configuration.

Fix

- Fixed an issue in the **Point query engine** so that client and API queries now return expected results that match a specified filter. Previously when retrieving point tags via CxScript.Points.GetPointTagList using certain wildcarded queries on any of the following PNT header record properties: FACILITYID, INDEXED1, INDEXED2, INDEXED3, SERVICE, SITE, SYSTEMNAME, or UNIFORMDATACODE, and the "!!USEEXTENDEDFILTER" option is included in the filter string, the query returned an empty list.

Enhanced Alarm Configuration (EAC)

Enhancement

- The maximum number of individual **Enhanced Alarm Configuration** (EAC) expressions per condition has increased to 50 from 20.

Fixes

- Buttons on the Enhanced Alarm Settings dialog box are now enabled when initially **adding expressions**. Previously, when adding initial expressions, the buttons were disabled.

- When **changing point types** and importing them into the Enhanced Alarm Settings dialog box, the point types are now displaying correctly. Previously, importing point types that were changed did not match their exported settings.

- **String and value inputs** are now limited to a length of 16 and display an error when an entry exceeds 16 characters. Previously, an error message failed to display.

- **Moving expressions** between levels in the Enhanced Alarm Settings dialog box now display consistent behavior. Previously, when moving expressions, the operators could change to a different operator.

- Fixed an issue that could cause the **internal processing queue** for EAC expression evaluations to build without bound resulting in new values associated with EAC enabled points to not be processed.

- Fixed an issue with EAC conditions set for **asynchronous alarm calculation types**. Previously, the Timestamp property would always be the current time, rather than the last time the value was updated.

- Fixed an issue with the copy points for a facility option in CygNet Explorer for source points with enhanced alarm(s) configured. Previously when attempting to copy the EAC segment to a new point via **Copy Points For Facility** the action failed.

- Fixed an issue so that editing, saving, and **exporting** EAC settings are persisted and exported as expected. Previously when exporting settings after an edit and save operation the EAC feature failed to persist changes to the database and export the updated settings properly.

EIEs – Communication Devices

The following changes have been made to CygNet communication devices in CygNet v9.5.

MQTT Comm EIE

Fix

- Fixed a rare issue with the MQTT Comm EIE to handle a specific rarely occurring **comm reconnection circumstance**. Previously, in an environment with frequent reconnection, a comm trying to reconnect could encounter a race condition between threads which could cause a UIS crash.

OPC Comm EIE

Enhancements

- Modified the device editor for the OPC Comm EIE to **remove** the (unused) "**Device not ready events**" setting and **add** the (new) "**Advanced options**” setting containing an "Enhanced connection validation" check box. This change reflects an enhancement to the OPC server reconnection logic to reduce disconnection occurrences, detect when connections are lost, and automatically re-establish the connection and subscriptions when disconnection does occur. When the new option is selected, every health check will verify that the OPC connection is still active by performing a read from cache for one data item from every group the selected device is subscribed to, which helps to keep the connection from becoming inactive thereby reducing disconnect/reconnect occurrences. Also see OPC EIEs.

- Enhanced the **TCP/IP Listen mode** to support DNP3.

Fix

- Fixed an issue for the OPC Comm EIE so that upon **reloading of a comm device**, its active connected remote devices are found, and any subscriptions are reestablished for groups with asynchronous callbacks. Also see OPC EIEs.

EIEs – Device Template Files

The following change has been made to CygNet device template files in CygNet v9.5.

Important: *If a device template file has been updated for this release, we strongly recommend that you obtain the applicable v9.5 sample device template file, edit it to retain customizations you added to your pre-v9.5 in-use template, and replace your pre-v9.5 in-use template with the version v9.5 sample template. Do not simply replace your pre-v9.5 in-use template with that provided on the source image because you will lose any template customizations that you previously made. Refer to the* ***Device Template Files*** *sections of the CygNet Help for detailed information about modifying templates.*

EIEs – Remote Devices

The following changes have been made to CygNet remote devices in CygNet v9.5.

Changes affecting all EIE Remote Devices

Fix

- Updated the reporting of **byte-related comm statistics** to reflect actual bytes received, particularly in cases where remote devices have multiple data groups subscribing to the same MQTT topic. Bytes shown in the Communications Data Viewer also now reflect actual bytes received, and show the same bytes only once if they are sent to multiple data groups. Previously bytes were counted for each data group subscribed to the same topic.

Allen Bradley CIP EIE

Enhancement

- For the Allen Bradley CIP EIE devices using "Native" protocol, sending of **string values** is now supported.

Fix

- Modified the Allen Bradley CIP EIE to **improve performance** of response processing and reduce memory consumption for in-progress messages.

Amocams 700 EIE

Enhancements

- Created a new ***CygNet Help*** topic, "Generic Template-Driven Data Groups", on implementing generic messages (**GenMsg**), command (**GenCmd**), and scans (**GenScan**). These data groups use standard message formats with opcodes, data items and offsets supplied in the template file.

- Created a new ***CygNet Help*** topic, "Opcode and Data Group Mapping" for the Amocams 700 EIE that **maps data groups** to their corresponding opcode numbers and message class.

Fix

- Fixed an issue for the Amocams 700 EIE to allow the value of a custom item in the "FMS Legacy Configuration Data" (**GmrConfig**) data group to be a string or other non-float value. Previously, the **customItem** element did not recognize strings.

DNP3 Emerson EIE

Enhancement

- Modified the DNP3 Emerson EIE device template files to **reduce complexity** and define a **configurable data group** (CfgDg). The changes appear in both the DNP3Emerson_FM2x.dtf and DNP3Emerson_FB3000.dtf EIE template samples provided in the CygNet source files. Use only the new device template file format going forward, as the manner of calculating units has also been affected.

Emerson ROCPlus EIE

Fixes

- Fixed an issue with the Emerson ROCPlus EIE when **polling** the History Point Configuration data group (**HPtCfg800**) in circumstances where a field device has a history segment that contains more than 199 configured points; up to 240 points are now supported. Previously an exception was logged and a UIS crash could occur in such cases.

- Fixed an issue with the Emerson ROCPlus EIE to use the **user-supplied data type** to calculate field length. Previously, the device was not able to send a single TLP UIS command for a user-defined point when the TLP was not found in the internal point tables.

- Fixed an issue with the Emerson ROCPlus EIE to correct the length on the **contract hour parameter** in order to poll the "Advanced Pulse Module" (**APM800**) data group. Previously, the device failed to poll the **APM800** data group due to an inconsistency in the contract hour parameter length.

- Fixed an issue with the Emerson ROCPlus EIE so that certain **MVS calibration events** (e.g. DP, SP, TMP) appear as the correct (calibration) event type in FMS.

Modification

- Reordered Old and New "**verify calibration**" event values for Emerson ROCPlus events logs, to align with Actual and Expected values in the Emerson calibration reports.

eProd EIE

Enhancement

- Modified the CygNet eProd EIE to allow **decreasing the packet size** when polling Dynagraph Card (DynaCard) data, to accommodate a broader range of communication constraints. This is accomplished by limiting the number of points requested per message with the field device, via usage of an optional UIS command parameter.

Lufkin MPC/RPC EIE

Modification

- Modified dynacard transactions to use the retrieved time as the **card timestamp** ("STime" and "STimeD" DEIDs). Previously there was no card timestamp included on dynacard transactions.

Changes Affecting some Modbus EIEs

Modification

- The data group attributes "**arrsize**" and "**arrRegOff**" can now be inherited from dgElements. Previously these attributes had to be set on each data group element.

Modbus Realflo EIE

Modification

- For the Modbus Realflo EIE, modified the device template file to **rename data group items** that represent observed density for liquid meters. The DEIDs were changed from **SpcGrv** to **Density** to better reflect the actual data. Removed the attribute **unitsCat**="density" from data group items that represent relative density. Data group items that represent observed density continue to include **unitsCat**="density".

All OPC EIEs (OPC EIE, OPC Lufkin EIE, OPC Weatherford EIE, and OPC Comm EIE)

Enhancements

- Modified the device editor for the OPC Comm EIE to **remove** the unused "**Device not ready events**" setting and **add** the new "**Advanced options**” setting containing an "Enhanced connection validation" check box. This change reflects an enhancement to the OPC server reconnection logic to reduce disconnection occurrences, detect when connections are lost, and automatically re-establish the connection and subscriptions when disconnection does occur. When the new option is selected, every health check will verify that the OPC connection is still active by performing a read from cache for one data item from every group the selected device is subscribed to, which helps to keep the connection from becoming inactive thereby reducing disconnect/reconnect occurrences. Also see OPC Comm EIE.

- OPC EIEs now support a new "**accessPathSeparator**" attribute that can be added to the data group in the device template file (DTF) to customize the OPC tag separating character occurring between the OPC tag elements for "Access path" (as set on the remote device editor) and "itemId" (as set on DEIDs in the DTF). If a value is specified the resultant separator is as user-entered, if no value is specified the separator "." (period) is used, and if the value "None" is specified no separating character is used.

- A new field is now visible in the OPC EIEs for **OPC quality status**. Also see Device Definition Service (DDS) for more information about this quality field.

Fixes

- Fixed an issue for the OPC EIE, OPC Lufkin EIE, OPC Weatherford EIE so that upon **reloading of a comm device**, its active connected remote devices are found, and any subscriptions are reestablished for groups with asynchronous callbacks. Also see OPC Comm EIE.

- Improved handling of the situation where an **OPC connection error** occurs when creating or adding items to an OPC group. The OPC connection will now automatically disconnect and be reestablished on the next reconnect interval in this circumstance, so that all valid OPC groups will be connected properly and respond as expected. Previously such connection errors would require a manual reset to fix the behavior of affected groups.

- A processing issue for **points mapped to data groups** in the OPC EIEs has been fixed, so that if group creation fails on the OPC server all points associated with that group are marked as unreliable. This change fixes an issue in the OPC EIEs when using quality mapping, so that point status bits for reference DEIDs properly update using the OPC quality bit mapping of the source DEID, as defined in the "qualityMapping" section of the device template file. Also see Device Definition Service (DDS) for more information about this quality field.

OPC Lufkin EIE

Enhancement

- Provided the ability for the OPC Lufkin EIE to retrieve data (including dynagraph cards) when utilizing the **Matrikon SCADA Modbus OPC** server.

OPC Weatherford EIE

Enhancement

- Provided the ability for the OPC Weatherford EIE to retrieve data (including dynagraph cards) when utilizing the **Matrikon SCADA Modbus OPC** server.

Fix

- Fixed an issue for the OPC Weatherford EIE for the "Dynagraph Card" (**DynaCard**) and the "Event Directory" (**EvtDir**) data groups so that polling for either data group **returns the correct date**. Previously, when polling for either data group an incorrect date could be returned.

Totalflow EIE

Fix

- For the Totalflow EIE, when **mapping data elements** on the "Register - Device Specific" (**RgstrData**) data group, if the data element ID exceeds 10 characters (the maximum number allowed), an error message will display.

Link Service

The following changes have been made to the Link Service for MQTT publishing in CygNet v9.5.

Enhancements

- CygNet Link v9.5 is backwards compatible with v9.4. Your **v9.4 appsettings file** will work with v9.5, however, you will need to modify your appsettings file to take advantage of the new features described below.

- Link **configuration** has been **simplified** in v9.5. The CVS endpoint no longer needs to be connected to a PNT endpoint in order for Link to function properly.

- All CygNet service **endpoints** now work with a protocol type of "CygNet". Previously supported protocol types ("CygNetDevice", "CygNetHistorian") have been deprecated in v9.5 but are supported for backwards compatibility.

- You can now target multiple points by creating **custom filter rules**. Two new members, **facilityFilter** and **pointFilter**, are supported in the publish object of the CVS endpoint in the appsettings file.

    - **facilityFilter** can be used alone, or with **pointFilter**, in a single publish object
    - **pointFilter** can be used instead of, or in conjunction with **pointTag**, in a single publish object

- **Wildcards** are now supported anywhere in a point configuration in the **facilityFilter** and **pointFilter** members so that you can target multiple points in a single line and publish all points that match the string. For example, (facility_id='RPOC*' and facility_desc='RTU*'). Wildcards are not supported in the **pointTag** member.

- An **environment** mode has been added to the gateway details in the Link configuration file to indicate the environment in which Link is running. Two modes are possible: "Edge" for Link running on a ForeSite Edge device and "SCADA" for Link running in a SCADA data center. Edge is the default value. If the environment property is missing from the configuration; the mode will be set to "Edge." Some additional modifications have been made for SCADA mode: in order to avoid device name collisions between facilities that share the same ID (but on a different site.service), Link will add the facility's site.service to the device ID. For the DDS endpoint, Link will add the facility's CVS service rather than the DDS service. Refer to the ***Link Help*** for more information about device names in SCADA mode.

- In order to differentiate at the MQTT topic level if the **data is DynaCard**, the device ID for the DDS endpoint and VHS endpoint will be appended with a suffix "?<DATA_TYPE>", where <DATA_TYPE> is variable depending on the type of data being published. The possible <DATA_TYPE> variables are:

    - DT — DynaCard transaction (published by the DDS endpoint)
    - DC — DynaCard data (published by the VHS endpoint)

    The CVS endpoint will not append a suffix to the device ID. Refer to the ***Link Help*** for more information about device ID formatting for DynaCard data.

- You can now limit CygNet point data by optionally publishing rollup history using the VHS. A **historicalRollup** parameter has been added to the CVS endpoint publish object to specify the rollup type, units, period and topOfUnit options. Link supports the following rollup types: None (raw data), ThinMedian, ThinMinMax, and ThinLast.

- **Point validation** has been enhanced in Link to detect missing and disabled points, and other security errors. A new strictPointValidation property has been added to the "data" object for the CVS endpoint to monitor for these types of errors. If set to true, Link will generate an error if the point is missing, disabled, or inaccessible. If set to false, Link will publish the metric, except the value will be null and there will be a property called "Error" with a value of "PointNotFound." The default value is true.

- Link now applies the following logic when the **strictPointValidation** and **strictPointValues** properties are configured in the appsettings file.
  - When **strictPointValidation** is true and
    - if a point is missing prior to birth, Link will not birth the device and polling will not happen
    - if the point becomes missing after birth, Link will publish an error and continue polling the other points on that device
  - When **strictPointValidation** is false, Link will publish an error and successfully birth the device and poll other points on the device
  - When **strictPointValues** is true and
    - if a point has the wrong value type prior to birth, Link will not birth the device and polling will not happen
    - if the point has the wrong value type after birth, Link will publish an error and continue polling the other points on that device
  - When **strictPointValues** is false, Link will publish whatever the value is regardless of type

  strictPointValidation also applies to history rollup, when rollup is defined in appsettings file, but a point is not configured to report to VHS.

  Note that the **strictPointValues** logic described above does not apply to history rollup in v9.5.

  Refer to the ***Link Help*** for more information about how Link handles point and point value validation.

- A data limit feature has been added to Link to indicate the maximum number of points that Link can publish at any point in time. The optional setting, **pointLimit**, is added to the "data" definition for each CVS service endpoint in the appsettings.json file. The recommended maximum amount of data that Link can publish at any point in time is 200,000 points across all CVS service endpoints in each Link instance. If you plan to publish more than 200,000 points, ensure that the system has sufficient CPU and memory to process the total amount of data, as attempting to publish too many points can severely impact a system's performance.

Modifications

- The "**provides**" property has been removed from the Link appsettings.json file, since it served no purpose. Other properties describe the type of data provided by each service endpoint.

- Link now accepts **multiple CygNet tag formats** in the appsettings file. To preserve backwards compatibility, the initialization process allows Link to read older tag formats, if used. Supported tag formats include:

Real-time Tag Formats	DynaCard Tag Formats
<facilityid>:<udc> (legacy)	<site>.<service>:<facilityid> (legacy)
<facilityid>.<udc>	<site>.<service>::<facilityid>
<site>.<service>::<facilityid>.<udc>	

- Link will now publish an **NDEATH** message on every shutdown.

- Link will now publish the current product version in the **NBIRTH** payload message.

- If a real-time value is not found, Link will now publish uninitialized points with a null value. Previously, Link waited for all real-time values to come in before publishing a **DBIRTH** message.

- The client ID maximum character limit has been increased from 23 (MQTT v3.1 spec) to 65535 (MQTT v3.1.1 spec). The previous limit did not work with a custom 64-bit client ID. **clientId** is specified in the protocol object of the **publisher** endpoint in the appsettings file.

- The **MQTT character set restriction** has been removed from the client ID. Some brokers might reject client IDs that use characters beyond the MQTT specification.

ODCB Driver

Fixes

- Updates have been made to the 32-bit and 64-bit CygNet ODBC drivers to **better support DDS transaction retrieval**. The 64-bit CygNet ODBC installer has also been updated. Specifically we fixed queries for DBS index tables against columns that are numeric, such as facility_ordinal and data_group_ordinal:

  AUD: ad_index, aud_id, op_index, ts_index, vr_index

  BSS: a1_index, a2_index, b1_index, f1_index, f2_index, f3_index, size, u1_index, u2_index

  DDS: c2_index, c4_index, c5_index, c6_index, data_group_ordinal, dg_index, facility_ordinal, facility_oridinal_obs, fd_index, fe_index, status_code, tf_index, tt_index, tx_index

  ELS: f1_index, f2_index, priority, x0_index, x1_index, x2_index, x7_index, x8_index, x9_index

  PNT: i1_index, point_scheme, v2_index

  Note that these updates may negatively affect queries that were specifically redesigned to work around the issue this fix corrects.

- Fixed an issue where the CygNet ODBC driver failed without warning when querying a point from the VHS that contained **special characters**.

Replication

Enhancements

These enhancements are related to changes in CygNet v9.5 to support additional facility attributes and updates to the Facility service database definition (FAC DDL). See Facility Service for the related note.

- Special consideration must be made when replicating between CygNet v9.5 and an older CygNet host. Refer to **Appendix A: Mixed-Mode Hosts** in the ***CygNet v9.5 Upgrade Procedure*** for more information about replicating in a mixed-mode environment and applying a recommended patch to allow an older Facility service and clients to interact with a v9.5 Facility service.

- When replicating between any two CygNet systems and **a mismatch in the database version** between a replicating DBS and source DBS is detected, the replicating service will now requeue all updated records for a later resync to ensure consistency in data records.

  A new info item, **REPL_DBS_RECHECK_KEY**, (Nice Name: Repl DDL recheck keys) has been added to track the number of keys that were modified in the source database if the source database has a larger database version number (DDL_VERSION_NUM info item). After normal replication up-to-date checks, the replicating service will recheck those records to make sure there is no mismatch.

  This enhancement applies to all DBS services when replicating between one CygNet version and another.

Security

Fix

- **Point-level Security Application Override performance improvement**. Although the recommended use of point-level security application overrides is that they "should be used sparingly and judiciously" (refer to the ***CygNet Help***), this recommendation has no enforcement and so the use of point-level security application overrides can inadvertently become excessive. A system with less than 1000 point records assigned an application override is considered reasonable. However, when the number of point records assigned an application override exceeds 10,000, significant performance degradation can occur when resolving security access in various client applications, most notably, loading a trend in CygNet Studio. This fix improves perceived performance of long running security checks by running them in the background instead of blocking the updating of the user interface until the checks complete. The existing recommendation of sparse and judicious use of point-level security application overrides is still encouraged.

Utilities

The following changes have been made to CygNet utilities in CygNet v9.5.

CygNet Message Sniffer Lite

Enhancement

- The CygNet Message Sniffer Lite utility has been enhanced to include **delta times** to help identify where message slowdowns occur. New diagnostic columns added to the CygNet Message Viewer dialog box include: Server Time To Ack (ms), Server Handling Time (ms), Client Time To Ack (ms), Client Delta (ms). The Request Size and Response Size columns have been split into two separate columns.

  Two additional options have been added to the message viewer context menu to display and highlight duplicate messages:

  - **Display Duplicate Messages** — Opens another CygNet Message Viewer dialog box containing all messages that are duplicates. A duplicate message is one where the request and response data are the same.

  - **Highlight Duplicates of Selected Message** — Selects and highlights all messages in the current view that are duplicates of the highlighted message

CygNet Service Migration

Modification

- The CygNet Service Migration Utility has been updated to **remove** items that have reached their **End-of-Life**.

CygNet ServiceMon Administration

Enhancement

- Messages have been optimized in the ServiceMon Administration Utility to improve performance when **creating SVCMON points**.

Fix

- Fixed an issue on the **Timer Config** page when adding a timer where the assigned timer did not match the timer that was selected.

DAT to CAPRI and CAPRI to DAT

Modification

- The Facility service database definition has been updated to support **additional facility attributes**. Please note the following:
  - A .compact file generated from a v9.5+ **DatToCapri** utility against a v9.5+ Facility service database cannot be restored with an earlier version of **CapriToDat**.
  - A v9.4 and earlier **DatToCapri** utility cannot read a v9.5+ Facility service database.

Fix

- Fixed an issue where the **DatToCapri** utility failed to export the database tables used for GNS plugins, GNS broadcasts, and NOTE metadata.

Group Manager

Fix

- Fixed an issue with the Group Manager where the utility was failing to **build a hierarchy** correctly. Previously some nodes in a built group hierarchy were missing leaf nodes.

MSS Viewer

Fix

- Fixed an issue for the **MSS Viewer** utility so that it correctly displays the string displayed on the UIS Command Task page of the Edit Schedule Task dialog box of the MSS. Previously, the MSS Viewer utility could crash or display incorrect information when trying to access the tool.

Point Configuration Manager

Fix

- Fixed an issue with updating existing points using the **Point Configuration Manager** utility, so that point values for cells containing edited values are changed successfully while values for cells left blank will remain unchanged.

Well Test

Enhancements

The CygNet Well Test Module and supporting documentation have been updated in the following ways in v9.5:

- **Support for Multiple Headers:** The CygNet Well Test Module has been enhanced to support the configuration of multiple headers for device-based configurations. In an environment where there are more wells than there are ports on a single MSV, cascaded or parallel MSVs are needed, and the CygNet Well Test Module needs to understand where wells are connected.

  A new configurable element, the sub-header, has been added to the header template in the Well Test configuration control (and to the XML), to indicate when testing wells are physically connected to multiple MSVs or other switching devices. A new sub-header configuration grid is available on the Templates page to define the cross-reference between the unique position of the header template to the sub-header and sub-header position where the well is connected.

The changes to the **Templates** and **Headers** configuration have been documented in the ***Well Test Help***.

- The ***Well Test Help*** has been updated to clarify the following **Test Action Type – Condition Action and Sleep Time Actions** details:

  The **Condition** action is evaluated every time the **Sleep timer** has elapsed. If the condition is true then the **Test** action will stop and the next action in the well test command sequence is executed. The Condition actions are optional:

  - If the condition is true and there are Condition actions defined, those actions will execute and then move on to the next action in the well test command sequence
  - If the condition is true and there aren't any Condition actions, CygNet simply moves on to the next action in the well test command sequence

- An issue retrieving a well test with the **CygNet.API.WellTest** when calling GetWellTestsForWell has been fixed. Previously requesting a well test with a UTC start and end date offset yielded no data.

- The **Canvas Well Test Control** design properties have been supplemented to include the following properties, which allow for the creation of an operator-focused screen that can limit well test configurations:

  - Show general – Shows or hides the **General** configuration page
  - Show headers – Shows or hides the **Headers** configuration page
  - Show templates – Shows or hides the **Templates** configuration page

Modifications

The CygNet Well Test Module and supporting documentation have been updated in the following ways:

- The **Exit** action type has been enhanced to allow the well test to exit the entire action sequence. An **Exit command** check box has been added to the Exit action configuration (and xml) to allow the well test to exit out of the entire action sequence.

  - If selected, when the Exit action is performed, the entire command sequence will halt at that point.
  - If not selected, when the Exit action is performed, the test just exits to the next item in the command sequence.

  For example, say you have a Condition in a Monitor or Test action that when met you want to exit the action and continue in the sequence, you would not select the Exit command option. However, if the sequence is in a Monitor or Test action and checking for the communication status of a UIS command, and you want to quit the entire command if the UIS command isn't successful, then you would select the Exit Command option.

- The **Script** action type has been enhanced tell the Well Test module how to process the well test results. A **Process output** check box has been added to the Script action configuration (and xml) to allow for the execution of script *without* processing returned data into a well test record. When this option is selected, the Well Test module will attempt to process the output as a well test result.

Fix

- Fixed an issue when adding a well to a **CygNet well test configuration** using the Canvas Well Test Configuration screen. The Facility Tag chooser for selecting a well facility now works as expected.