





CygNet v9.9 Release Notes

Release Date: April 2, 2025

This document describes new features and changes to CygNet Software since the v9.8 release.

For instructions on updating your host, refer to the  [CygNet v9.9 Upgrade Procedure](#)  for more information.

Copyright © 2025 Weatherford
All rights reserved

Contents

Product Lifecycle	3
CygNet Lifecycle Notice	3
Upgrade Assistance	4
CygNet Documentation	4
Highlights in v9.9	5
CygNet Core	5
CygNet EIEs	5
CygNet Measurement	6
CygNet OPC UA Suite	6
Canvas Client and CygNet Thin Web HMI Client	7
Changes in v9.9	8
Alarms	8
CygNet Clients — Canvas	8
CygNet Clients — CygNet Explorer	10
CygNet Dispatch	10
CygNet Measurement (FMS)	10
CygNet OPC UA Suite	14
CygNet Services	15
Most Services	15
Universal Interface Service (UIS)	15
EIEs – Communication Devices	15
MQTT Comm EIE	15
EIEs – Device Template Files	15
EIEs – Remote Devices	16
Allen Bradley CIP EIE	16
DNP3 Emerson EIE	16
IoT CygNet Link EIE	17
IoT EIE	17
IoT Sparkplug EIE	17
Lufkin SAM EIE	17
Totalflow EIE	18
ODBC Driver	18
Replication	19
Scripting	20
Utilities	20
Well Test	21
CygNet Online Help	25

Product Lifecycle

For more information on the lifecycle of CygNet components, refer to the **CygNet Software Product Lifecycle Matrices** on the [Weatherford Software Support portal](#) (login required) under **CygNet Software > Maintenance & Support Info**.

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). Contact your Weatherford Account Manager or CygNet Sales (via email at [CygNet Sales](#)) for more information about your support options.

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

CygNet Version	Release Date	Limited Support Begins	Limited Support Ends (End of Life)
9.9	April 2, 2025	April 4, 2027	April 4, 2028
9.8	March 15, 2024	March 15, 2026	March 15, 2027
9.7	March 27, 2023	March 27, 2025	March 27, 2026

The following CygNet versions have reached lifecycle milestones with the release of CygNet v9.9:



- **CygNet v9.8** will enter Limited Support on March 15, 2026, and will reach its End of Life on March 15, 2027.
- **CygNet v9.7** entered Limited Support on March 27, 2025, and will reach its End of Life on March 27, 2026.
- **CygNet v9.6** reached its End of Life on March 2, 2025.

The following CygNet components have reached lifecycle milestones with the release of CygNet v9.9:


- **CygNet v9.6 Online Help** will enter Limited Support with the release of CygNet v9.9. It will reach its **End of Life** with the release of CygNet v10.0 and will be removed from the documentation website.
- **CygNet v9.5 Online Help** entered Limited Support with the release of CygNet v9.8. It has reached its **End of Life** with the release of CygNet v9.9 and has been removed from the documentation website. An offline version of the CygNet v9.5 Online Help can be downloaded from the [CygNet Download Site](#) (login required).

Refer to the [Microsoft Lifecycle Policy](#) for information about Windows operating system support.

Upgrade Assistance

Upgrade assistance is available 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 [Weatherford Software Support portal](#)  (login required), via phone at 1-866-4CYGNET (1-866-429-4638), or via email at [CygNet Support](#) .

CygNet Documentation

Refer to the [CygNet Help](#)  for CygNet v9.9 for user assistance. The online help is best viewed in Microsoft Edge or Google Chrome browser. Microsoft Internet Explorer 11 is not supported.

Highlights in v9.9

This section highlights some of the major new features and enhancements in the v9.9 release. Please see [Changes in v9.9](#) 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 Services enhancements:

- **CygNet Messages** for several services (ARS, CVS, DBS, and VHS) have been updated to enhance performance and reduce impact of network latency.
- In the UIS we have modified the data type of several **communication and remote device statistics** from i4 to ui8 to allow larger, non-negative numbers before the numbers wrap around at the data type limit.
- See the notes [here](#).

CygNet Replication — The process by which replicated database services perform a **full resync process** has been significantly changed in CygNet v9.9. The new replication full resync approach prioritizes full resync speed over low network bandwidth considerations and has shown dramatic improvements in full resync duration times with little additional impact to network bandwidth. For example, a PNT with 200K points that previously took over 40 minutes to perform a full resync now completes the full resync in under 40 seconds. See the note [here](#).

Several enhancements have been made to the **CygNet.NET API** to aid programmatic access to devices and data group transactions, and CygNet Notes. See the note [here](#).

A 64-bit version of the **CygNet Facility Import/Export** (FacLoad64.exe) utility is now available. See the note [here](#).

The **Telerik UI for WPF** libraries (used by Canvas, CygNet Explorer, and Dispatch) have been updated to version 2024.3.924.462 to prevent multiple security vulnerabilities detected in prior versions of the dlls.

Several enhancements, modifications, and fixes have been made to the **CygNet Well Test** module to improve performance and usability. See the note [here](#).

CygNet EIEs

Several **Allen Bradley CIP EIE** fixes have been applied:

- Fixed an invalid header error when writing data to routed field devices using the PCCC protocol.
- Fixed an issue with sending data groups to routed devices.
- Fixed an issue where configurable data groups did not always load the data group definition correctly on UIS startup.
- Fixed an issue where editing a DTF could break configurable data group polling on any device using that DTF.
- Fixed some UI issues in the remote device editor for configurable data groups.
- See the notes [here](#).

The maximum number of instances of a data group for the **DNP3 Emerson EIE** is now DTF driven. See the note [here](#).

A new EIE, the **IoT CygNet Link EIE**, is now available to support bidirectional communication between a CygNet enterprise system and CygNet running on a ForeSite Edge device communicating with a remote device in the field. The IoT CygNet Link EIE is an IoT Sparkplug-based EIE that communicates with an instance of Weatherford Link via an MQTT server. See the note [here](#).

In the **Lufkin SAM EIE**, an optional DTF attribute on the DateTime "Date and Time" data group called includeDayOfWeek="true" has been added, which allows users to include or exclude register 41431 when getting or sending this data group to a Lufkin SAM device. See the note [here](#).

The **MQTT Comm EIE** now prevents duplicate client IDs and enhances robustness for blank IDs by offering an option to append the database key to the ID. Other MQTT Comm EIE enhancements include: message handling has been greatly improved; the normal comm message queueing mechanism now occurs; and all messages get queued appropriately allowing comm tuning, memory management, and enhanced logging and statistics. See the notes [here](#).

The **Totalflow EIE** now supports the configuration of Enhanced meter mode in Totalflow devices for AGA3, AGA7 and API Linear Liquid measurement applications. Several data groups have been modified or added to support enhanced meter mode configuration, enhanced alarm logging, and data retrieval from meters configured for enhanced mode.

The **Totalflow EIE** now uses version 3.16 of the ABB Totalflow toolkit API (tcidll.dll and tcidll64.dll). See the notes [here](#).

CygNet Measurement

An array of useful enhancements, modifications, and fixes is included in the v9.9 release for CygNet Measurement. Highlights are listed below, and further details can be found [here](#).

- **Database upgrade performance** — improved efficiency of upgrade processes from versions prior to v9.8, for implementations processing liquid device data.
- Added a new **Command Queue** visibility feature.
- Added a new **Liquid Configuration** report.
- **Detect New Devices commands** — expanded device detection criteria.
- **Variable job schedules** — added "daily" or "native" record span options.
- **Update Normalization Views commands** — improved performance when writing to the database.
- **Command log messaging** - improved messaging for Detect New Devices, and in other locations.

CygNet OPC UA Suite





The existing CygNet OPC UA Server has been updated and is now integrated into a new **CygNet OPC UA Suite**, which includes additional tools for managing access to CygNet data via the OPC UA protocol. The CygNet OPC UA Suite consists of the following components:

- **CygNet OPC UA Server** — The CygNet OPC UA Server exposes real-time and historical CygNet data using the OPC UA specification.
- **CygNet OPC UA Client Service** — The CygNet OPC UA Client Service transfers real-time values from an OPC UA Server to a CygNet CVS.
- **CygNet OPC UA Config Manager** — The CygNet OPC UA Config Manager is a web-based application used to configure and manage the CygNet OPC UA Server and CygNet OPC UA Client Service.
- **CygNet OPC UA Web API** — The CygNet OPC UA Web API provides programmatic access to the CygNet OPC UA components.

See the note [here](#).

Canvas Client and CygNet Thin Web HMI Client

All changes made to Canvas and the CygNet Thin Web Client since CygNet v9.8 are recorded in the following documents:

- Canvas changes in CygNet v9.9 since the release of CygNet Thin Web Client v1.6 on November 15, 2024 are listed [here](#).
-  [CygNet Thin Web Client v1.6 Release Notes](#) 
-  [CygNet Thin Web Client v1.4 Release Notes](#) 

Changes in v9.9

This section describes enhancements, modifications, and fixes to existing components in CygNet v9.9.

Alarms

Fix

- The **Deviation for Time Alarm Calculations** now work as expected. Fixed an issue when calculating the DEVIATION_PCT_FOR_TIME and DEVIATION_VAL_FOR_TIME variations, where the wrong value was being converted to an absolute value for threshold comparison. This caused negative deviations to not be triggered properly. Also made the <= or >= comparisons explicit.

CygNet Clients — Canvas

All changes made to Canvas since CygNet v9.8 are described in the following documents:

-  [CygNet Thin Web Client v1.6 Release Notes](#) 
-  [CygNet Thin Web Client v1.4 Release Notes](#) 

Modifications and Fixes

The following changes have been made to the Canvas client since the release of CygNet Thin Web Client v1.6 on November 15, 2024.

Canvas and Canvas.View Clients

- The **Telerik UI for WPF** libraries have been updated to version 2024.3.924.462 to prevent multiple security vulnerabilities detected in prior versions of the dlls.
- Fixed some **vertical alignment issues** with the right-side toolbar in the Canvas and Canvas.View **Backstage View** caused by an upgrade to the 2024 **Telerik UI for WPF** libraries.
- Fixed some display issues with the **layout grid** in the screen design view in Canvas caused by an upgrade to the 2024 **Telerik UI for WPF** libraries.
- Removed an **unnecessary horizontal scrollbar** that was appearing in the Canvas Properties pane after an upgrade to the 2024 **Telerik UI for WPF** libraries.
- Fixed some **padding issues** with the Canvas workspace caused by an upgrade to the 2024 **Telerik UI for WPF** libraries. After the upgrade the Workspace contained extra padding in some elements (Toolbox, Design view, and Properties pane). This is now fixed.

Canvas Controls

Alarm Notifier

- Hid the **color picker** when either the Background color or Text color source is set to Auto in the Canvas Design view. When color is set to Auto the color picker should not be visible.

Button

- Fixed an issue where the **Color configuration properties** (Background, Text and Border) failed to persist on a screen in both design and run mode when the color was configured in a Button style sheet in the Canvas Backstage view.

DDS Button

- Fixed an issue where the **Color configuration properties** (Background, Text and Border) failed to persist on a screen in both design and run mode when the color was configured in a Button style sheet in the Canvas Backstage view.

DDS Edit Box

- **Validation** has been added to ensure that the DDS Edit Box supports **all data types of values** configured for the data group element IDs (DEIDs). The Canvas UI has been modified to support limiting the range of values accepted for any DEID.

Heat Map

- Fixed some **resizing issues** with the Heat Map in the Canvas view caused by an upgrade to the 2024 **Telerik UI for WPF** libraries. After the upgrade the Heat Map did not scale properly and became unresponsive when resizing at runtime. This is now fixed.

Navigation Button

- Fixed an issue where the **Color configuration properties** (Background, Text and Border) failed to persist on a screen in both design and run mode when the color was configured in a Button style sheet in the Canvas Backstage view.

SetPoint Button

- Fixed an issue where the **Color configuration properties** (Background, Text and Border) failed to persist on a screen in both design and run mode when the color was configured in a Button style sheet in the Canvas Backstage view.

CygNet Clients – CygNet Explorer

The following changes have been made to the CygNet Explorer client application in CygNet v9.9.

Enhancement



- The **Telerik UI for WPF** libraries have been updated to version 2024.3.924.462 to prevent multiple security vulnerabilities detected in prior versions of the dlls.

Fix

- Fixed an issue where the **CygNet Service Information** dialog box for the PNT (Service Details), and in other clients where this dialog box is visible, failed to display data in any tab and clicking **Get Svc Info** generated a count error.

CygNet Dispatch

The following changes have been made to CygNet Dispatch in CygNet v9.9.

Important: **CygNet Bridge v4.8** is required to run Dispatch with CygNet Measurement in CygNet v9.9. Refer to the  [CygNet v9.9 System Requirements](#)  for more information about additional version compatibility details.

Enhancements

- The **Telerik UI for WPF** libraries have been updated to version 2024.3.924.462 to prevent multiple security vulnerabilities detected in prior versions of the dlls.
- For **CygNet Dispatch** users, added options to the variable job schedule feature when averaging by volume, to allow selection of "daily" or "native" record spans in addition to the previously supported "hourly" record span.

CygNet Measurement (FMS)

The following changes have been made to CygNet Measurement in CygNet v9.9.

Enhancements

- For CygNet Measurement upgrades from versions prior to v9.8, **improved database upgrade performance** for implementations processing liquid device data, saving time and memory resources.
- Improved **unit conversions** in FMS to streamline the processing of absolute and pressure data values.
- Added a **Command Queue** feature, accessible from the Commands menu, to provide a view into current command processing activity. The new functionality provides monitoring of in-progress and queued FMS commands, and allows optional cancellation of pending command actions.
- Expanded the device detection criteria used by **Detect New Devices** commands to determine automatic creation of new FMS Nodes, by adding more facility filtering options to the previously supported Facility ID filter.
- Improved performance of the **Update Normalization Views** command when modifying base conditions for gas devices with the "Include archived data" option selected.

- Increased performance of the **Update Normalization Views** command by improving database processes to gain speed and efficiency.
- Added a "Node name" detail to **command log** entries, so that Node name/ID information is displayed for successful transactions, as was previously displayed for unsuccessful transactions.
- Added a **Liquid Configuration** report template with support for liquid data item tokens (LiquidConfigurationTemplate.xlsx) to the FMS service, to allow display and publishing of configuration reports for liquid devices.
- For **CygNet Dispatch** users, added options to the variable job schedule feature when averaging by volume, to allow selection of "daily" or "native" record spans in addition to the previously supported "hourly" record span.

Modifications

- Modified the **FMS service** start-up files to incorporate the (previous) CxFmsImportCsvHistory.dll and CxFmsImportCsvConfig.dll file content into the (streamlined) FMS.exe file, in order to consolidate installation processes.
- Modified command log messaging for **Detect New Devices** actions to improve formatting, better describe error details, and remove unnecessary warnings.
- Modified the calculation process in the **Configuration** control, used when calculating total composition for Gas Quality values, to initially use 7 significant digits so that values are better aligned for comparison.

Fixes

- Optimized memory consumption when updating the database for **FMS services** versioned prior to v9.8.
- Fixed an issue with **upgrades** from prior versions to CygNet Measurement v9.8, for those using a SQL database and having pulse-based devices, so that Pulse Count data values are retained correctly. Previously these values were no longer appearing after the upgrade.
- Improved performance of CygNet Measurement systems with extremely large FMS databases running the **Service Monitoring Service** (SVCMON) with FMS data points (e.g., SVMFMSDBSI – Size of Database), to reduce the appearance of superfluous errors in the **fms.log** file.
- Fixed an issue with the processing of **transaction status values** in circumstances where the service shuts down while a transaction is in progress, so that the "in progress" status is now accurately reflected and additional transactions for the involved Node are no longer blocked.
- Fixed an issue with **Import: CSV** commands so that requests including data types not supported for the Node are not processed.
- Fixed an issue with **Export: CSV** commands to correctly display liquid quality data item values. Previously some values could be shown incorrectly or not displayed at all.
- Fixed an issue with **Export: CSV** commands so that fields with blank values are handled consistently.
- Fixed an issue with the **Import: History CSV** command to successfully import gas quality data values.
- Fixed an issue with the **Import: History CSV** command to process data imports without service interruption. Previously this was not the case in certain circumstances.

- Fixed an issue with the processing of **Import: History CSV** commands so that large CSV import files are imported completely and successfully. Previously such imports could be truncated after the first 1000 records.
- Fixed an issue with the optional **Export: Flow-Cal Transaction Queue** command so that, when the Config data requirement parameter is set to the "always export" value, exports containing no configuration data succeed as expected.
- Fixed an issue with the optional **Export: Flow-Cal Transaction Queue** command by improving memory usage efficiency during export operations.
- Modified the messaging displayed when **Request Current Configuration** commands include liquid product data, to accurately reflect the scope of the action.
- Fixed an issue with the **Generate Smart Groups** command so that Nodes are correctly assigned to groups in cases where a facility is linked to multiple device Nodes and the Smart Group definition includes the "device type" value for group determination.
- Fixed an issue with the **Validate Data** command to ensure that Liquid Quality data is retained properly. Previously the liquid configuration record could be missing this data after running a validation command.
- Fixed an issue with the processing of **Update Normalization Views** commands to improve the efficiency of data queries for liquid Nodes.
- Fixed an issue in **Configuration** reports so that the [FirstRow] token works as expected, as it does for other FMS reports.
- Fixed an issue with **Liquid Device QTR** reports to correctly display liquid quality data item values. Previously some values could be displayed for the wrong Node.
- Fixed an issue in the **Liquid Device QTR** Report to properly display orifice, cone, and pipe thermal expansion coefficient values as expected when the corresponding data item tokens are used. Previously these values did not appear in the report.
- Fixed an issue affecting **Balance Reports** to correctly calculate Gas Equivalent Volume (GEV) values in cases where configuration changes include liquid quality data value changes.
- Fixed an issue with the processing of date ranges when purging records so that **Batch records** are consistently included as intended. Previously some records could have been omitted from the purge action.
- Fixed an issue with the display of **Dew Point** configuration data values, so that units are shown uniformly. Previously Dew Point was appearing as unitless.
- For CygNet Measurement upgrades from versions prior to v9.8, fixed an issue with the **Balance control** in FMS Explorer, so that liquid data can be loaded into the control correctly.
- Fixed an issue and improved performance in the **Configuration control**, so that tabbing/editing functions behave as expected for Device Nodes when using the Misc property page.
- Fixed an issue in the **Configuration control** to improve performance when updating configuration data via the Mass Update feature. Previously FMS Explorer could close unexpectedly in this circumstance.
- Fixed an issue in the **Configuration control** so that, for liquid device Nodes, Product Index data item cells can be successfully edited when the value is null or not previously present.
- Fixed an issue in the **Configuration control** so that editing a "null" Contract Hour value for a liquid device Node succeeds as expected. Previously the FMS Explorer application could stop running in this circumstance.

- Fixed an issue in the **Configuration control**, Configuration view, so that using the Mass Update feature when the selection contains flow computer values (Brand, ID, Model, Software Version) maintains the data item strings as expected. Previously mass updates containing these values could cause the data item character strings to be truncated.
- Fixed an issue in the **Dashboard** control so that, when viewing data in "Station view" data mode, general group Nodes only display their station Node members, and provide explanatory messaging for the user if no station Node members exist. Although general groups can contain other device Nodes, only station-level data records are eligible to be shown in Station view. Previously some liquid device Nodes could errantly appear when Station view was selected in the control.
- Fixed an issue with the display of missing data in the **Dashboard** control when viewing data for time ranges greater than 100 days, so that only missing data is displayed. Previously collected data would mistakenly appear as missing in this circumstance.
- Fixed an issue in the **History Grid control**, occurring in specific circumstances after editing data quality records, to correct the logging of an errant exceptions message.
- Fixed an issue so that rapid reloading of **FMS controls** won't cause potential memory errors.


CygNet OPC UA Suite


The following changes have been made to CygNet OPC UA in CygNet v9.9.


Enhancement

- The existing CygNet OPC UA Server has been updated and is now integrated into a new **CygNet OPC UA Suite**, which includes additional tools for managing access to CygNet data via the OPC UA protocol. The CygNet OPC UA Suite consists of the following components:
 - **CygNet OPC UA Server** — The CygNet OPC UA Server exposes real-time and historical CygNet data using the OPC UA specification.
 - **CygNet OPC UA Client Service** — The CygNet OPC UA Client Service transfers real-time values from an OPC UA Server to a CygNet Current Value Service (CVS).
 - **CygNet OPC UA Config Manager** — The CygNet OPC UA Config Manager is a web-based application used to configure and manage the CygNet OPC UA Server and CygNet OPC UA Client Service.
 - **CygNet OPC UA Web API** — The CygNet OPC UA Web API provides programmatic access to the CygNet OPC UA components.

The **CygNet OPC UA Suite** installed on CygNet v9.9, v9.8, or v9.7 requires **CygNet Bridge API v4.8** to operate.

The **CygNet OPC UA Suite** is a separately licensed component from existing CygNet SCADA software components. Contact your Weatherford Account Manager or CygNet Sales (via email at [CygNet Sales](mailto:CygNet_Sales) ) for more information about obtaining and licensing the CygNet OPC UA Suite and CygNet Bridge products.

Download the **CygNet OPC UA Suite Installer** from the [CygNet Download Site](#)  (login required).

Refer to the [CygNet OPC UA Suite](#)  section in the **CygNet Help** for more information.

CygNet Services

The following changes have been made to CygNet services in CygNet v9.9.

Most Services

Enhancement

- Added and updated several CygNet service messages (ARS, CVS, DBS, and VHS) to **enhance performance** and reduce impact of network latency.

Universal Interface Service (UIS)

Enhancement

- Modified the **data type** of several **communication and remote device statistics** from i4 to ui8 to allow larger, non-negative numbers before the numbers wrap around at the data type limit. This also includes the system UDCs: SYCSBSNT, SYCSBRCV, SYCSTOT, SYCSFTOT, SYCSFCRC, SYCSFNOR, SYCSBRCV, SYCSFOTH, SYCSSTN, SYCSS1P and SYCSSNP.

EIEs – Communication Devices

The following changes have been made to CygNet communication devices in CygNet v9.9.

MQTT Comm EIE

Enhancements

- MQTT Comm EIE now **prevents duplicate client IDs** and enhances robustness for blank IDs by offering an option to **append the database key to the ID**. When Append unique ID is selected, the database key, preceded by an underscore, is added to the end of the user-provided client ID. If the client ID is left blank, it will automatically be set to [site].[service]:[commId] or if Append unique ID is selected it will automatically be set to [site].[service]:[commId]_[dbKey].
- **Message handling** for the MQTT Comm EIE has been greatly improved. Now the normal comm message queueing mechanism occurs, all messages get queued appropriately allowing comm tuning, memory management, and enhanced logging and statistics.

EIEs – Device Template Files

Important: *If a device template file has been updated for this release, we strongly recommend that you obtain the applicable v9.9 sample device template file, edit it to retain customizations you added to your pre-v9.9 in-use template, and replace your pre-v9.9 in-use template with the version v9.9 sample template. Do not simply replace your pre-v9.9 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 in the **CygNet Help** for detailed information about modifying templates.*

EIEs – Remote Devices

The following changes have been made to CygNet remote devices in CygNet v9.9.

Allen Bradley CIP EIE

Fixes

- Fixed an **invalid header error** for Allen Bradley CIP EIE when writing data to routed field devices using the PCCC protocol. Previously data bytes were missing in the CIP PCCC message for Send commands when routing was utilized.
- Fixed an issue with **sending data groups to routed Allen Bradley CIP devices**. Previously, if the data group elements plus the routing information needed to be split into multiple CIP packets, the maximum CIP message size could be exceeded.
- Fixed an issue where **configurable data groups** did not always load the data group definition correctly on UIS startup.
- Fixed an issue where editing an Allen Bradley CIP DTF could break **CfgDg** "Configurable Data Group" polling on any device using that DTF.
- Fixed some **user interface** issues in the remote device editor for the **CfgDg** "Configurable Data Group":
 - When adding or editing a data group element, the **Data type** drop-down menu was missing the Boolean, i1, and ui1 data types. It is also possible to edit the XML directly to add these items, but all data types are now visible in the menu with this fix.
 - If the configurable data group was set for **Can write via UIS command** the DEIDs did not show in the UIS command component parameters; the parameters now display as expected.
 - When adding or editing a data group definition, the **IO service** drop-down menu was empty. The menu now defaults to <None> and accurately represents what has been set previously for the data group. It is also possible to edit the XML directly to populate this menu, but the IO service options (Tag, TagFragmented, or None) are now visible in the menu with this fix.


DNP3 Emerson EIE

Modification

- Removed the hard-coded **maximum number of instances** of a data group allowed in the DNP3 Emerson EIE so that the maximum number is now DTF driven. The only limit is the value specified in the **maxCnt** attribute.

IoT CygNet Link EIE

Enhancement

- A new EIE, the **IoT CygNet Link EIE**, is now available to support bidirectional communication between a CygNet enterprise system and CygNet running on a ForeSite Edge device communicating with a remote device in the field. The IoT CygNet Link EIE is an IoT Sparkplug-based EIE that communicates with an instance of **Weatherford Link** via an **MQTT server**. The purpose of the EIE is to mirror a remote device on a CygNet system where Link is running. One side of the mirror is an IoT CygNet Link remote device running on a CygNet enterprise system; the other side of the mirror is another CygNet remote device using a standard CygNet EIE running on an Edge device. Link acts as a gateway or conduit through which the IoT CygNet Link device sends commands to the CygNet instance running on the Edge device, which then performs on-demand polling of data groups and send requests and values to a remote device in the field. Link will then publish data group transactions back to the MQTT server and back to the CygNet enterprise system for processing and display. The IoT CygNet Link EIE gets and sends data to and from a Link-enabled device using the standard CygNet data group transaction / UIS command access model. Note the following:
 - The IoT CygNet Link EIE requires **Weatherford Link 2.1** for interoperability.
 - Refer to the **IoT_CygNet_Link_WellPilotRPOC.dtf** in the `\Samples\EIETemplates\Samples` folder of the CygNet Source for an example device template file.
 - Refer to the [IoT CygNet Link EIE](#)  section in the **CygNet Help** for more information.

IoT EIE

Fix

- The remote device **data group values** (Group ID, Edge node ID, Device ID, QoS, Use comm default option, and Payload format) entered in the **Data Group Properties** dialog box from the device's **Data Group** tab are now saved into metadata as expected. Previously these values were not getting saved without clicking Apply in the **Data Group** tab.

IoT Sparkplug EIE

Fixes

- Command parameters in the **CmdDev** "Command - Device" and **CmdNode** "Command - Edge Node" data groups are now being validated. Previously a UIS command could be published with invalid parameters.
- The remote device **data group values** (Group ID, Edge node ID, Device ID, QoS, Use comm default option, and Payload format) entered in the **Data Group Properties** dialog box from the device's **Data Group** tab are now saved into metadata as expected. Previously these values were not getting saved without clicking Apply in the **Data Group** tab.

Lufkin SAM EIE

Enhancement

- Added a new **optional DTF attribute** on the **DateTime** "Date and Time" data group called `includeDayOfWeek="true"`, which allows users to **include or exclude register 41431** when getting or sending this data group to a Lufkin SAM device. An earlier Lufkin firmware used register 41431 for the day of the week value, although that register was later changed to be "reserved". When the Lufkin Well Manager 2.0 devices were released, register 41431 was changed to indicate the time zone value. To keep legacy behavior in place, the Lufkin Sam EIE will use register 41431 for day of the week, unless `includeDayOfWeek` is set to "false". Any Lufkin Well Manager 2.0 devices should use a DTF with `includeDayOfWeek="false"` on the **DateTime** data group. The sample **LufkinSam.dtf** has been updated with the new attribute.

Totalflow EIE


Enhancement

- The **CygNet Totalflow EIE** now supports the configuration of **Enhanced meter mode** in Totalflow devices for AGA3, AGA7, and API Linear Liquid measurement applications.

Several changes have been made to the **Totalflow EIE** and the sample **TotalFlow_G4.dtf** to support an enhanced meter mode and all customers who want to use enhanced mode should incorporate the changes into their own Totalflow DTFs. Changes to support enhanced mode are backwards compatibility with prior versions so that users who do not need to use enhanced mode do not need to make any changes to their DTFs after upgrading to CygNet v9.9.

The following changes have been made to the Totalflow **data groups** to support data retrieval from meters configured for enhanced mode:

- **Config** "Meter Configuration" — Added several data group elements to configure and monitor enhanced meter mode configuration, general information, data item calculation, and enhanced mode alarm thresholds. Added two attributes (**Undefined Sentinel Value** and **Undefined Sentinel Description**) to signal to the Totalflow EIE how to handle *Undefined* alarm configuration parameters received from meters enabled with enhanced mode.
- **FMSAlarm** "FMS Alarms" — When a meter has enhanced mode enabled, FMS Alarms uses the new **HistAlm** "History - Alarm Log Enhanced" data group to get alarm data. Non-enhanced meters continue to get their alarm data from the meter's **HistLog** "History - Log Period" data group.
- **HistAlm** "History - Alarm Log Enhanced" — Added a new data group to manage information from the enhanced alarm logs. Updated the **View Data** dialog box to display enhanced alarm logs.
- **HistLog** "History - Log Period" — Added a data group element to retrieve an **Indicated Volume** data item for AGA7 and API Liquid applications only. IV/UVolume is used when the meter is in enhanced mode, instead of the ACF (Uncorrected volume) data group element. No change to volume retrieval in the **HistLog** "History - Log Period" data group for AGA3.

Refer to the sample **TotalFlow_G4.dtf** for usage examples, found in **CygNet\Samples\EIETemplates\Release** in the CygNet source files. Refer to the [Device Template Files](#)  sections in the **CygNet Help** for detailed information about modifying templates.

Refer to the [Totalflow EIE](#)  section in the **CygNet Help** for more information about this enhancement.

- The Totalflow EIE now uses version 3.16 of the **ABB Totalflow toolkit API** (tcidll.dll and tcidll64.dll). Contact your ABB Totalflow representative for additional information about changes to the tcidll.dll.

ODBC Driver

The following change has been made to CygNet ODBC driver in CygNet v9.9.

Fix

- Fixed an issue with the **CygNet ODBC driver** where queries referencing literal **QueueKey** values failed to return the correct data. Also fixed **LIKE** processing on QueueKeys greater than level one and a few **TopKey** formatting issues.

Replication

Enhancement

- The process by which replicated database services perform a **full resync process** has been significantly changed in CygNet v9.9. **Prior** to this change, the full resync process was as follows:
 1. The replicated service determines that a full resync is needed or it is told to perform a full resync via an info item-based command and the current timestamp is recorded to the REPL_RESYNC_START info item. Additionally, the value "2: Full sync" is recorded to the REPL_STATUS info item.
 2. The replicated service reads the keys for all top-level database records and stores them in an internal list or "queue".
 - a. A maximum of 1000 database keys are queued up at any one time.
 - b. The number of database keys in the queue is represented by the value of the REPL_RESYNC_Q_COUNT info item.
 - c. When the last database key has been added to the queue, the current timestamp is recorded to the REPL_RESYNC_ALL_QUEUED info item.
 3. A separate processing thread removes each database key from the queue, one at a time, and reads that record and all of its child segment records, updating the replicated service's database files accordingly.
 4. When all records have been processed from the queue, the current timestamp is recorded to the REPL_RESYNC_END info item. On success, the REPL_STATUS info item is reset to "1: Waiting".

The retrieval of all database records on a full resync was intentionally designed to be gradual so as to not impact networks with limited bandwidth. The consequence of this design decision is that, depending on network latency, the full resync process for a PNT with 200K points, for example, could take up to 40 minutes to complete. Unfortunately, when the failover features were added, which depend heavily on replicated services being fully synchronized, the gradual resync mechanism, and the potentially very long full resync times, is possible and problematic.

The **new replication full resync approach** implemented in CygNet v9.9 prioritizes full resync speed over low network bandwidth considerations. Highlights of the new full resync approach are as follows:

1. The full resync process is initiated as before but instead of reading and queueing up the top-level database keys, all top-level and child database records are retrieved immediately.
2. The records are retrieved using high-capacity 64 KB compressed CygNet messages for maximum throughput and minimum effect from network latency.
3. A new info item, **REPL_RESYNC_CNTDN - Repl Resync Countdown** has been added to indicate progress of each top-level database key processed.
4. Two info items related to the now defunct queueing of top-level database keys have been obsoleted, **REPL_RESYNC_Q_COUNT** and **REPL_RESYNC_ALL_QUEUED**. The info items are no longer updated, although they both still show up in the info item list (e.g., in the **Service Details** dialog box) for backwards compatibility.

Testing of the **new full resync approach** has shown dramatic improvements in **full resync duration times** with little additional impact to network bandwidth. As an example, a PNT with 200K points that took over 40 minutes to perform a full resync now completes the full resync in under 40 seconds.

Scripting

The following enhancements have been made to the **CygNet .NET API** libraries in CygNet v9.9.

Enhancements

- Added a variation of the **GetDataGroupTransactions** method to the **CygNet.API.Devices** library, to retrieve a list of data group transactions that were registered between two specified dates. The maximum number of records retrieved by default is 100. The list can be filtered by device id, device type, data group type, ordinal and by the status of the transaction. The wildcards '*' and '?' are supported to match case-insensitivity for the string parameters.
- Added a **GetDataGroupTransactionsData** method to the **CygNet.API.Devices** library, to retrieve a list of data group transactions and their associated data by a given transaction key list.
- Added a **GetAndSendData** method to the **CygNet.API.Devices.PollingClient** library to support sending and retrieving data from a device via script.
- Enhanced the CygNet.API library to include **CygNet.API.Notes** to aid in programmatic access to the CygNet Note Service (NOTE). Interfaces are available to create notes, delete notes, get note metadata, retrieve and filter notes between a specified time range, read ranged and TBD notes, set note metadata, and update notes via script in the connected Note service.
- The **CygNet.API.Devices** and **CygNet.API.Notes** assemblies are documented in a standalone help file, **CygNet\Bin\CygNet.API.chm**.

Utilities

The following change has been made to CygNet utilities in CygNet v9.9.

Enhancement

- A 64-bit version of the **CygNet Facility Import/Export** (FacLoad64.exe) utility is now available. It can be installed using the CygNet Client Installer from the APP/UTILS directory.

Well Test

The following notes describe enhancements, modifications, and fixes to the **CygNet Well Test** module in v9.9.

Miscellaneous Well Test Enhancements

Well Test performance and action execution has been improved in the **CygNet Well Test** module.

- Added support for up to 1000 simultaneous well tests.
- Well Test actions have been reworked to execute more efficiently.
- Added support for multiple headers in configuration.
- Added support for multiple points in configuration.
- Added the ability to sequence wells in a header dynamically.
- Changed the name of the process that shows in Windows **Task Manager** to "**CygNet Well Test Service**" from "Well Test Service" so that it is easier to find in the **Processes** list.
- Stopping the CygNet Well Test Service from the Windows **Services** Console takes less time and no longer leaves the process running in the background.
- The CygNet Well Test Service is now **64-bit**. Consequently, any custom script actions created that use CygNet COM automation DLLs will require 64-bit versions of those DLLs. The **CxScript64.dll** has been included in the CygNet Well Test version management process to account for this when using the sample well test scripts.
- The CygNet Well Test Service now creates more detailed and useful **log entries** written to its primary log with a file name similar to "**CygNetWellTestService0xx.csv**".
 - Because the service is now built in a 64-bit configuration, this log file is written to the **C:\Windows\System32** folder; when built under 32-bit it was written to the **C:\Windows\SysWOW64** folder.
 - Up to 10 log files are created, each with a maximum size of ~10MB and with file names "CygNetWellTestService001.csv" to "CygNetWellTestService010.csv". When log file #010 fills up, it cycles back to log file #001.
 - If you do not see these log files being generated, it is likely due to limited permissions for the user account under which the CygNet Well Test Service is running. Running the service under an account with elevated permissions should fix the problem.
- Updated Well Test Canvas screens are available in the **Samples\Well Test** folder:
 - **Well Test Configuration.can**
 - **Well Test Status.can**

Other Enhancements

- Improved the method for determining if the **Microsoft script engine** has exited after it has executed a user-provided VBScript file. The old method would occasionally cause an infinite wait. Now it will wait at most 10 seconds for the Microsoft script engine to exit.
- Changed the convention for **storing Well Test XML results in VHS blobs**:

Previously, the Well Test results process was the following:

- Retrieved the Well Test XML results produced by the configured VBScript file.
- Extracted the %STATUS% and %DATETIME% values from the results.
- Attempted to write the XML directly to the VHS by creating a new HistoryEntry for the point with the configured UDC (e.g., "WELLTEST").
 - Set the value of the HistoryEntry to the value of the Well Test result %STATUS% indicator (typically similar to "1:Good") and set the timestamp of the entry to the extracted %DATETIME% value.
 - If the VHS point did not yet exist, created a new HistoryName in the VHS only and wrote the new HistoryEntry and associated blob to the new VHS point.
 - This technique created an orphaned point in the VHS, meaning it had no representation in a corresponding PNT or CVS, which violated CygNet best-practices.
- If a HistoryEntry with the same timestamp already existed for the HistoryName, the new Well Test result did not get recorded.

The **new** Well Test results process is as follows:

- Retrieve the Well Test XML results produced by the configured VBScript file.
- Extract *only* the %DATETIME% value from the results.
- Look for the point with the configured UDC (e.g., "WELLTEST") in its associated CVS (UIS, HSS, etc.).
 - If not there, log an error and *stop* processing, since the point must be pre-configured in the PNT like all other CygNet Well Test points.
- Update the current value of the point in the CVS, setting both the value and the timestamp to the %DATETIME% value.
 - The value is in the form "YYMMddhhmmssxxxx" (e.g., 2503250805040000).
 - This is the same convention used by the new method for storing a Header's **PendingWellQueue** and **ActiveWell/ActionQueue** in VHS blobs, described [below](#).
 - If a valid %DATETIME% value cannot be extracted from the Well Test XML results, the current system time is used instead.
- Wait for that value to propagate to the VHS and, when it arrives, update the VHS blob associated with that entry.
 - The %STATUS% value is still maintained within the Well Test XML results and is still displayed properly in the Canvas **Well Test Detail.can** screen.

- If a HistoryEntry with a matching timestamp already exists, overwrite the existing VHS blob with the new Well Test XML results (the last one in wins).
- Removed use of **PNT HyperPoint script text** for each Headers' **PendingWellQueue** and **ActiveWell/ActionQueue** configuration in favor of VHS blob storage.
 - Using the PNT service's HyperPoint script text field for rapidly changing entries was an inefficient choice and had negative consequences for the associated CVS, as well as for any point data caches being maintained on every CygNet client in the enterprise.
 - The Well Test Configuration point *continues* to use the PNT HyperPoint script text for its configuration because it does not change frequently.
 - Migrating an existing CygNet system to this new model includes the addition of the two separate points, described below, as well as changing the configuration of all existing QUEUE points by setting the "HyperPoint Enabled" attribute to N for "no" (see the PNT Editor, General Properties). This can be done conveniently in bulk using the **CygNet Point Configuration Manager** (PointCfgMgr.exe) utility.
- Implemented separate points for **PendingWellQueue** and **ActiveWell/ActionQueue**.
 - The previous use of the QUEUE Header point to store this very volatile Well Test state had to be abandoned because the current value of the QUEUE point is already being used to drive the Well Test command process.
 - The configuration for both the **PendingWellQueue** XML and **ActiveWell/ActionQueue** XML previously resided in the same HyperPoint script text of the Header's QUEUE point. They have been split apart and are now stored in separate points' VHS blobs. This was done to improve change synchronization between the CygNet Well Test Service itself, and any external user of the CygNet.API.WellTest, such as the Canvas **Well Test Status.can** screen.
 - The **ActiveWell/ActionQueue** settings are now managed exclusively by the CygNet Well Test Service and cannot be modified by the CygNet.API.WellTest. This approach guarantees that the CygNet Well Test Service always knows what the latest configuration is and that nothing has gone behind its back and changed it.
 - However, because the **PendingWellQueue** is often changed by users of the Canvas **Well Test Status.can** screen, additional safeguards have been implemented to minimize the chance that the CygNet Well Test Service might be working with a list of pending wells that are not the latest ones stored for that Header. There are still small, but possible, scenarios whereby a user of the Canvas **Well Test Status.can** screen could make changes, just at the wrong instant, which would not be immediately picked up by the CygNet Well Test Service. But the likelihood of this occurring has been dramatically reduced compared with the old HyperPoint script text model.
 - Removed the obsolete **Header.ClearActionQueue()** method from the CygNet.API.WellTest now that only the CygNet Well Test Service affects the contents of the persisted **ActionQueue**.
 - The following updates have been made to the Canvas **Well Test Status.can** screen:
 - Added new **Header Actions** label and buttons for the newly added Well Test commands: **Schedule**, **Continuous**, and **Stop Continuous**.
 - Added new **Pending Well Queue Actions** label and relabeled buttons: **Add One to Start**, **Add One to End**, **Add All to End**.
 - Added **Read Queue** and **Clear Queue** buttons.

- Fixed issues related to properly updating the **Pending Well Queue** when it is changed by the screen or by the CygNet Well Test Service. Added a **Refresh** button to refresh the **Pending Well Queue Actions**.
 - Implemented better detection and reporting of errors generated by calls to the `CygNet.API.WellTest`.
- The Well Test **Header** configuration and **QUEUE** execution have been enhanced to allow the specification of a **Sequence number** or a **Sequence UDC** to manage the order of wells in the test on the fly. Previously **Continuous** mode would add wells to the Queue automatically based on the order in the well test configuration; however, that was not always the order in which they need to be tested. This enhancement adds the ability to specify a Sequence number in the well test configuration or a Sequence UDC to take a value from a point value. When using **Continuous** or **Schedule** mode, the test will now follow the sequence order.
 - Two columns (a **Sequence** number and a **Sequence UDC** (SEQ)) have been added to the **Wells** configuration on the **Headers** tab of the Canvas **Well Test Configuration.can** screen to specify the sequence order for each well in the Header. The Wells Queue will be updated using the current value of Sequence or Sequence UDC before starting the next test, so that the Queue can change dynamically each time a new test for any position on the Header is started. Similar to the way **Enabled UDC** works, if both Sequence and Sequence UDC have assigned values, Sequence UDC takes precedence.
 - A new method, **Well.GetSequence()**, has been added to the `CygNet.API.WellTest` to return the sequence order.
- Added a way to exit **Continuous** mode without having to abort the current well in test. A new reserved Queue command, **StopContinuous**, has been added, which will set the `ContinuousTesting` flag to false. When this flag is set to false and no other action or command is issued, the wells in the Queue will continue to be tested and once completed the Queue will go to an Idle state. Once the Header is in its Idle state, the next time a command is processed, it will read the new configuration prior to execution. This modification will allow for the user to change any well test configuration required and be able to stop Continuous mode gracefully to allow for the new changes to be read.
- The **Set Point** action when the **Value type** is set to *PointValue*, *PointAttribute*, or *FacilityAttribute* now works as expected. The Canvas **Well Test Configuration.can** screen and the `CygNet.API.WellTest` have been enhanced to allow resolution of the source point for the Set Point action via configuration fields. Previously these fields were missing from the UI and the API. For each Value type the following fields have been added:
 - **PointValue** — UDC and Relative facility link
 - **PointAttribute** — Point attribute, UDC, and Relative facility link
 - **FacilityAttribute** — Facility attribute, and Relative facility link


Fixes

- Well Test **recovery** has been fixed so that when the CygNet Well Test Service is stopped and restarted testing will resume on the well and action currently in test.
- Protected repeated **Start** commands for a Header already in a Start state in **Continuous** or **Schedule** mode. Previously this condition could put the Well Test into an unknown state. The CygNet Well Test Service now prevents a `START`, `CONTINUOUS`, or `SCHEDULE` request from processing while one of those commands is already being processed.
- Fixed the CygNet Well Test Service so that it does not require a restart every time a new Header is added, disabled, or enabled. Previously after configuring a new Header for an existing template, then triggering a **Start** action, the action Queue became stuck in pending. The default status for a newly added Header is Idle.

- Fixed an issue with the **Monitor** action not observing the **Exit Command** action. Now the configuration will exit the complete sequence of actions for the well and move on to the next well in the Queue.
- Fixed an issue with a **Header** action not working in a **SleepTimer** action. A **Test** command has a Sleep Timeout action that executes every time the configured sleep time expires. Previously, if you configured a Header action and referenced it to a Header command, that action never executed.
- Cleaned up the **Idle** state so that the Queue will reset to Idle after a **Stop, Abort, or Complete** command. Previously, a Queue would go to an Idle state when all actions for the Queue completed. However, if a Stop, Abort, or Complete action interrupted that completion, the Queue point didn't get set to Idle.
- Fixed the **Stop, Abort, and Complete** commands while in the **Test** and **Monitor** actions. Previously the **Stop** command failed to stop execution if the well test configuration had nested **Timeout** or **SleepTimer** actions. Multiple nested actions increased the delay to stop execution. Now when **Stop** is selected the well test will stop immediately even if the configuration has **Timeout** or **SleepTimer** actions.
- Fixed the **Stop** command while in **Test** and **Monitor** actions. Previously the Stop command failed to clear the action Queue. After adding wells to a Queue and starting the Queue, a subsequent STOP command didn't completely flush the action Queue.

CygNet Online Help

Notice of Status Change (NSC)

- **CygNet v9.6 Online Help** will enter Limited Support with the release of CygNet v9.9. It will reach its **End of Life** with the release of CygNet v10.0 and will be removed from the documentation website.
- **CygNet v9.5 Online Help** entered Limited Support with the release of CygNet v9.8. It has reached its **End of Life** with the release of CygNet v9.9 and has been removed from the documentation website. An offline version of the CygNet v9.5 Online Help can be downloaded from the [CygNet Download Site](#)  (login required).