CygNet Thin Web Client v1.0 Release Notes

Release Date: July 31, 2023

The CygNet Thin Web Client v1.0 is compatible with Canvas in CygNet v9.7.

This document describes features included in the CygNet Thin Web Client v1.0.

For instructions on updating your host, refer to the *CygNet v9.7 Upgrade Procedure* for more information.

Contents

CygNet Thin Web Client Overview

The **CygNet Thin Web Client** (TWC) is CygNet's browser-based Human-Machine Interface (HMI) client application, which allows users to view operational screens and workflows in a web browser using sophisticated web-based technologies. The Thin Web Client is a Single Page Application (SPA) that is built using customer provided and built screens. Those screens are constructed using the design and screen building components of the Canvas desktop client. Canvas is used to layout HMI components onto a screen (a page of controls) and configure a logical workflow between screen controls and related screens. When published, those screens are converted and compiled into the SPA that resides on the main TWC server.

This document supports CygNet Thin Web Client v1.0 and Canvas in CygNet v9.7.

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) under **CygNet Software** > **Maintenance & Support Info**.

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 at 1-866-4CYGNET (1-866-429-4638), or via email at CygNet Support.

CygNet Documentation

Refer to the *CygNet Help* for user assistance. The online help is best viewed in Microsoft Edge or Google Chrome browser. Microsoft Internet Explorer 11 is not supported.

Refer to the following sections of the *CygNet Help* for further user assistance:

- Canvas Help
- CygNet Thin Web Client Help

CygNet TWC Sample Screen Library

A set of sample Canvas screens is available for download from the CygNet Download Site (login required). These screens can be modified to work against your CygNet system to create TWC screens and applications.

Major Features in CygNet TWC v1.0

This section describes major features in CygNet Thin Web Client (TWC) v1.0.

General TWC Features

License Requirements

- The CygNet Thin Web Client is a component of the CygNet SCADA platform licensed for use by CygNet Software and as such requires a **CygNet.lic** file containing an enabled CygNet TWC key in order to use the product. The CygNet.lic file must be loaded in the license master ARS on the same domain where the TWC services are running. A time-based trial license is also available. Contact your Account Manager about acquiring the necessary license file.

User and Screen Access

- Access to the TWC web view, published applications, and Canvas screens is determined by security settings in the **CygNet Access Control Service** (ACS):
    - User access to the TWC web view is controlled by a new **TWC** security application and an **ACCESS** event. The TWC security application and event stored in the ACS is specified in the TWC configuration file (configuration.json).
    - Screen access is controlled by the **BSS** security application and an **ACCESS** event (This is not new. Access to Canvas screen files stored in the BSS have always had security permissions applied to them.) The TWC applies this permission to users attempting to access BSS screens in the web view.

TWC Services

- The **port numbers** where the TWC services run are configurable. Edit the Config/configuration.json file and change the port numbers assigned to the TWC.PublishingService.Server and/or the TWC.Service.Server parameters as necessary. Restart the affected server(s).
- The Publishing service converts and publishes **multiple applications** efficiently by publishing multiple applications in a single action.
- If a Canvas screen contains a mix of supported and unsupported controls, the Publishing service will process the screen successfully, displaying the supported controls and **ignoring the unsupported ones**. The unsupported controls will be identified in the web view.
- If the CygNet **ACS permissions** on a Blob file (or folder) change, the Publishing service will automatically detect this and re-convert the screen even if the screen itself didn't change. A **Force publish** action may be required to get the web server to recognize the new permissions.

TWC Installation and Upgrade

- A Thin Web Client installer (**TWC.Installer.SetUp.exe**) is available to install the two TWC services (Main web service and Publishing service). The installer is also used to upgrade your TWC services. See the instructions in the Thin Web Client online help about using the installer and applying additional steps after an upgrade.
- The Thin Web Client installer adds the user specified during installation to the **Windows Logon as a Service** policy, ensuring the TWC services can run under the supplied user account.

- Users may need to manually update the required **Collected-Packages.zip** file after an upgrade or other TWC server configuration change. The Publishing service uses a required Collected-Packages.zip file, which contains various software packages, to compile an application during the publishing process. One of the included software packages contains information about the TWC configuration. If your TWC server configuration changes (e.g., after un upgrade), you may need to update the Collected-Packages.zip file with a version generated by the TWC installer. Refer to the Thin Web Client online help for instructions on how to do this.

TWC User Interface

Context Menu

- The TWC supports its own **context menu** in the web view. To activate the context menu click anywhere within a control and the menu will appear on the right side of the browser window so as not to block the main view of your screen. From the context menu you can do the following:
    - view **Current Value** attributes for the selected point
    - **Acknowledge** an alarm if the point is in an alarm state
    - view **Alarm Summary** values.

Canvas Backstage View

Web Settings Configuration

- A required Canvas patch for v9.7 users (**Patch-2023-07-31-Canvas**) to update your 9.7 system is available from the CygNet Download Site (login required).
- The Canvas **Backstage** view has been enhanced to support required **CygNet Thin Web Client** configuration:
    - A **Target environment** settings is available on the **General Settings** page to indicate whether Canvas is in *Native* or *Web clien*t mode. *Native* mode provides full base Canvas functionality. *Web client* mode provides the controls and properties supported by the TWC web view. When in *Native* mode the Web Settings tab is not visible.
    - A **Web Settings** page is available to configure, publish, and monitor TWC applications. This page consists of the following sections:
        - **Publishing Service** — Access the Publishing service, which can be refreshed at any time.
        - **Application** tab — Configure settings for each TWC application: name, start page, menu items, Blob folders, and folder management. This is where you publish (or force publish) the active application.
        - **Server** tab — Perform the following actions: a) publish multiple applications together in a single action after a server upgrade; b) republish previously published applications (soft publish) after a server upgrade; and c) assign a short name to each application Blob folder.
        - **Publishing** tab — View publishing status and error information for TWC applications as they are converted, compressed, and saved ready for viewing in a web browser.

Canvas Controls

All TWC Controls

- A required Canvas patch for v9.7 users (**Patch-2023-07-31-Canvas**) to update your 9.7 system is available from the CygNet Download Site (login required).

- Configuration properties that are not supported in TWC screens are hidden when the **Target environment** is *Web client*.

- **Auto color configuration** is not supported as a color source in control properties in screens to be used in a TWC application because TWC does not support themes in the web view. If a property specifies **Auto** for color configuration, the screen will use the **<Self>** color that is specified in the Canvas file during the publishing process.

Alarm Grid

- The **Alarm Grid** supports the creation of customizable and interactive tables for the display and management of large amounts of alarm data in columns and rows. The main TWC features include:

    - Static alarm filter support
    - Configurable alarm toolbar filters in runtime
    - Configurable columns — alarm, facility, and point
    - Self and point state color support — background and text
    - Facility sender
    - Facility receiver.

- The Alarm Grid (like the CygNet Grid and Navigation Button) supports **hyperlinking without script**, allowing you to double-click on a cell or row, open another screen, and send the selected facility to the receiving screen. Hyperlinking from an Alarm Grid is supported for inclusion on screens viewed in the Thin Web Client, although the TWC only supports *Open*, *Open modal*, and *Replace* hyperlink modes.

- A dynamic run-mode filtering option has been added to the Alarm Grid. A new property, **Alarm toolbar filters**, has been added, which allows you to add a toolbar containing custom filters to narrow the alarms presented in the grid.

- The Alarm Grid supports configuration as a **SiteService** or **facility receiver** similar to the CygNet Grid. A new property, **Row configuration**, has been added, so that you can indicate where the Alarm Grid is receiving its data. Data can be sourced either from a SiteService (received from the screen or the Common Alarm Service (CAS)) or from facilities (received from the screen or from another control on the screen).

- Unacknowledged alarms **blink** as expected in the TWC web view in the Alarm Grid (background and text) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

Chart

- The **Chart** provides a customizable means to visualize point values and historical CygNet data in a line chart. The main TWC features include:

    - Facility receiver
    - Single y-axis
    - Series type of Line only
    - Series relative facility support.

- A Chart will display **x-axis ticks and labels** that correspond to the date range configured for the chart, e.g., for every second, minute, hour, day, or year. Additionally, the x-axis label contains only the relevant part of the timestamp, e.g., the day will not be included if the date range is seconds or minutes. Previously, if you were displaying data for a day, the chart only showed a single tick for that day, but now the chart shows the hours within that day.

CygNet Grid

- The **CygNet Grid** supports the creation of customizable and interactive tables for the display and management of large amounts of data in columns and rows. The main TWC features include:
    - Configurable columns — facility and point
    - Configurable column colors — self and point state color support for background and text
    - Facility sender
    - Facility receiver
    - Built-in hyperlink support on double-click — modes are *Open*, *Modal open* (popup window), *Replace*.

- Unacknowledged alarms **blink** as expected in the TWC web view in the CygNet Grid (background and text) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

- Numerous enhancements have been made to the CygNet Grid to **improve performance** and data loading in the web view.

Donut

- The **Donut** displays real-time point data on a circular linear scale. Data is represented in a ring that fills the donut in a clockwise direction to a value between a lower and upper bound. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute in three lines of text
    - Self and point state color support — background, text, ring, and empty ring
    - Can display alarm ranges
    - Facility receiver.

- The **value ring and alarm ranges** on the Donut display as expected in Canvas and in the Web client view. If any misconfiguration in the Value bounds for the control is detected, the value ring and alarm ranges are hidden, and an "Invalid bounds" label will display under the control. Several enhancements have been made to ensure that the Value bounds properties are handled correctly, including:
    - Added logic to detect any misconfiguration in the **Value bounds** settings. Invalid bounds include any of the following conditions:
        - the lower value bound is greater than the upper value bound
        - the lower value bound is equal to the upper value bound
        - the lower and upper value bounds are zero
        - either the lower value bound or the upper value bound is non-numeric.
    - Added a new property in the Bounds category in Canvas, **Hide invalid bounds**, to hide the value ring and alarm ranges if any misconfiguration described above is detected. This property allows you to hide the value ring and alarm ranges and show a warning label if invalid value bounds are detected. The options are:
        - If **Hide invalid bounds** is not selected, Canvas will hide the donut's value ring and alarm ranges and show an "Invalid bounds" label. This is the default setting.
        - If **Hide invalid bounds** is selected, Canvas will hide the donut's value ring and alarm ranges and hide the "Invalid bounds" label.

Note that any configured **Text** properties will continue to display on the donut even if the value bounds are invalid.

Image

- The **Image** control can be used to display images on your Thin web Client screens. The main TWC features include:

    - Support for common image formats (.bmp, .png, .jpg, .jpeg, .gif, .svg)
    - Facility receiver
    - Point state image support.

Linear Gauges

- The **Linear Gauge** controls display real-time point data on a horizontal or vertical object. Two separate controls are supported: the **Horizontal Linear Gauge** and the **Vertical Linear Gauge**. Data is represented by a bar that fills the control to a calculated value between a lower and upper bound. The main TWC features include:

    - Single point control
    - Display any associated facility or point attribute
    - Facility receiver
    - Self and point state color support — background, text, bar, and empty bar.

- The **value bar and scales** (primary and secondary) on the two Linear Gauges display as expected in Canvas and in the Web client view. If any misconfiguration in the Value bounds for the control is detected, the value bar and scales are hidden, and an "Invalid bounds" label will display along the hidden bar on the control. Several improvements have been made to ensure that the Value bounds properties are handled correctly, including:

    - Added logic to detect any misconfiguration in the **Value bounds** settings. Invalid bounds include any of the following conditions:

        - the lower value bound is greater than the upper value bound
        - the lower value bound is equal to the upper value bound
        - the lower and upper value bounds are zero
        - either the lower value bound or the upper value bound is non-numeric.

    - Added a new property in the Bounds category in Canvas, **Hide invalid bounds**, to hide the value bar and scales if any misconfiguration described above is detected. This property allows you to hide the value bar and scales and show a warning label if invalid value bounds are detected. The options are:

        - If **Hide invalid bounds** is not selected, Canvas will hide the linear gauges' value bar and scales and show an "Invalid bounds" label. This is the default setting.
        - If **Hide invalid bounds** is selected, Canvas will hide the linear gauges' value bar and scales and hide the "Invalid bounds" label.

    Note that the configured **Text** property will continue to display on the linear gauges even if the value bounds are invalid.

Navigation Button

- The **Navigation Button** can be used to add buttons to your screens that implement hyperlinking to other Canvas screens *without* scripting. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute
    - Facility receiver
    - Self and point state color support — background and text
    - Built-in hyperlink support on double-click — modes are *Open*, *Modal open* (popup window), *Replace*
    - Static image support.
- Unacknowledged alarms **blink** as expected in the TWC web view for the Navigation Button (text) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

Nested View

- The **Nested View** shows another screen, meaning you can show a screen within another screen. This allows for more templated screens or dynamic screens, which can be swapped out at runtime based on user selection. The main TWC features include:
    - Facility receiver
    - Border color support
    - Nest linked screen in parent screen
    - Nested screens must be stored in a BSS folder.
- A **BorderColor** property has been added to the Nested View control in Canvas to specify the color used for the border of the control.

Screen

- The Canvas **Screen** is where you create flexible and interactive diagrams and layouts for your TWC users. By configuring a screen with TWC-supported controls, you can create a data-rich visualization of devices, facilities, and other elements in your CygNet environment, conveying information about the state of a process via charts, grids, gauges, images, and other data-visualization tools, and design workflows in run mode. The CygNet screen is where you create the HMI for your users in the field and is analogous to TheFrame/TheView in CygNet Studio.

SetPoint Button

- The **SetPoint Button** can be used to add buttons to your Thin Web Client screens that perform a setpoint command to change to a new point value without scripting. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute
    - Facility receiver
    - Self and point state color support — background, text, and border
    - Static image support
    - Built-in setpoint command (no script needed)
    - Configurable dialog box that prompts user to enter a new point value.
- Unacknowledged alarms **blink** as expected in the TWC web view for the SetPoint Button (text and border) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

Tag Chooser

- The **Tag Chooser** displays a hierarchical tree of CygNet facilities, allowing the creation and implementation of templated screens in Canvas. It can be used as the primary navigational control on your screens to drive other controls by sending facilities and point values to receiving controls. The main TWC features include:
    - Facility-based hierarchy support
    - Support for initial filter
    - Facility sender
    - Color support — background and text.
- Two color properties have been added to the Tag Chooser to allow the setting of **foreground color**; the new properties are **Text color** and **Text color source**.
- A facility-based Tag Chooser will display the expected hierarchical configuration of CygNet facilities organized by level in the web view. This type of tag chooser uses facility attributes, such as text-based or table-based attributes (which are determined by global attributes configured in the FAC service), to group individual facilities into a hierarchy. Generally, facility attributes that represent individual facilities, such as Facility ID and Facility Description, are used as leaf nodes in the hierarchy—the last level. However, other facility attributes that may contain multiple facilities, such as Field or Area, can also be used as the leaf node in a hierarchy. Previously, in certain configurations of the tag chooser and selection of a tag chooser node, not all the facilities were being sent to receiving controls (such as the CygNet Grid). Only, the first facility was being sent. That is now fixed.

Text Tool

- The **Text Tool** can be used to display text on your Canvas screens. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute
    - Facility receiver
    - Self and point state color support — background, text, and border.
- Unacknowledged alarms **blink** as expected in the TWC web view for the Text Tool (background, text, and border) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

UIS Command Button

- The **UIS Command Button** can be used to add buttons to your screens that send any kind of UIS command for a specified facility without scripting. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute
    - Facility receiver
    - Self and point state color support — background, text, and border
    - Static image support
    - Built-in UIS command (no script needed)
    - a configurable dialog box that prompts user to enter parameter value:
        - Configuration includes the following options: the name of the UIS command as defined in the DDS, the command's parameters and values, a prompt so a user can enter parameter value(s), the ability to set a default parameter value, a prompt to send or cancel the command, a status point to monitor the status of the command, the option to enable or disable the button while the command is in progress, and an option to display a confirmation message.

- Unacknowledged alarms **blink** as expected in the TWC web view for the UIS Command Button (text, and border) when configured to display unacknowledged alarms. Alarm acknowledgment is handled via the TWC context menu. Note that while blinking is supported for unacknowledged alarms, the blink rate is not being read from the Global Settings file (.gsf) in this release.

Value Indicator

- The **Value Indicator** adds visual context to real-time point and alarm data. A data range is represented on a vertical bar that fills the control between an upper and lower bound. The main TWC features include:
    - Single point control
    - Display any associated facility or point attribute for three lines of text
    - Facility receiver
    - Self and point state color support — background, text, and marker fill color
    - Can display alarm ranges.

- The **value bar, alarm ranges, and marker** on the Value Indicator display as expected in Canvas and in the Web client view. If any misconfiguration in the Value bounds for the control is detected the value bar, alarm ranges, and marker are hidden, and an "Invalid bounds" label will display along the hidden bar on the control. Several improvements have been made to ensure that the Value bounds properties are handled correctly, including:
    - Added logic to detect any misconfiguration in the **Value bounds** settings. Invalid bounds include any of the following conditions:
        - the lower value bound is greater than the upper value bound
        - the lower value bound is equal to the upper value bound
        - the lower and upper value bounds are zero
        - either the lower value bound or the upper value bound is non-numeric.
    - Added a new property in the Bounds category in Canvas, **Hide invalid bounds**, to hide the value bar, alarm ranges, and marker if any misconfiguration described above is detected. This property allows you to hide the value bar, alarm ranges, and marker and show a warning label if invalid value bounds are detected. The options are:
        - If **Hide invalid bounds** is not selected, Canvas will hide the value indicator's value bar, alarm ranges, and marker and show an "Invalid bounds" label. This is the default setting.
        - If **Hide invalid bounds** is selected, Canvas will hide the value indicator's value bar, alarm ranges, and marker and hide the "Invalid bounds" label.

  Note that the configured **Text** property will continue to display on the value indicator even if the value bounds are invalid.