

---

# Cylon Unitron UC32 Driver for Tridium Niagara

User Guide

[baudrate.io](https://baudrate.io)

niagara<sup>4</sup>

February 23, 2026

# Contents

- 1 Introduction** **2**
- 2 Requirements** **2**
- 3 Quick Start** **2**
- 4 Cylon Network** **3**
- 5 Cylon Devices** **4**
- 6 Cylon Points** **4**
  - 6.1 Point Import from CSV File . . . . . 5
  - 6.2 Point Discovery . . . . . 7
- 7 Schedules** **8**
- 8 FAQ** **11**
  - 8.1 I cannot discover any devices . . . . . 11
  - 8.2 Points are not reading . . . . . 11
  - 8.3 Schedule export is not working . . . . . 12

## 1 Introduction

Cylon<sup>1</sup> Unitron UC32 is a well-known BMS that includes a wide range of programmable controllers for HVAC control and building automation. There are two types of Unitron devices: \* field controllers for direct digital control \* communication controllers for information exchange, global schedules and LAN connection

Cylon driver for Niagara is designed to enable communication between Cylon controllers and Tridium Niagara powered devices. It provides an efficient solution for retrofit of building management systems, facilitates seamless integration of multiple protocols and allows convenient makeover of front-end software.

## 2 Requirements

- Niagara-powered device with software v4.8 or later, including JACE, Supervisor, or their OEM versions
- Cylon driver module and license
- UC32.net communication controller with TCP/IP connectivity

Niagara Supervisor or JACE is connected to a communication controller UC32.net via TCP/IP network. The communication controller is connected to field controllers via RS-485 port.

## 3 Quick Start

1. Copy cylon-rt.jar to both JACE and Workbench
2. Add **Cylon Network** to JACE station
3. Enter license code in **License** property under **Cylon Network**
4. Enter IP address and port in **Tcp Config** property; default port number is 4950
5. Open **Cylon Network** and **Discover** devices – it should find one communication controller and field controllers if available
6. Add controllers to the station
7. *For each field controller* open **Points** extension, **Discover** points or **Import File**, add points to the station
8. *For each global schedule* create a **Boolean Schedule** in the station, then open communication controller **Schedules** extension, use **Import File** to add **Schedule Export** to the station and

---

<sup>1</sup>All trademarks or registered trademarks are the property of their respective owners

select **Boolean Schedule** in **Supervisor Ord**; run **Read from controller** action and check if **Boolean Schedule** is filled.

## 4 Cylon Network

**Cylon Network** contains many standard Niagara properties, as well as a few Cylon-specific:

- **License** – the code that allows the driver to run on your host.
- **Tcp Config / Ip Address** – IP and port (4950 by default) of UC32.net communication controller.
- **Tcp Config / Site** – Unitron site number.
- **Tcp Config / UC32net Address** – communication controller address. Both **Site** number and **UC32net Address** can be found in Unitron project or on UC32.net display.
- **Poll Scheduler** – specifies polling rates for periodic data reading:
  - Fast Rate: 5 seconds
  - Normal Rate: 10 seconds
  - Slow Rate: 30 seconds

In most cases only **License** and **Tcp Config / Ip Address** values need to be changed.

The screenshot displays the configuration window for 'CylonNetwork (Cylon Network)'. The interface is organized into a tree view on the left and a corresponding property grid on the right. The 'Tcp Config' section is expanded, showing the following properties:

Property	Value
Status	{ok}
Enabled	true
Fault Cause	
Health	Ok [31-Aug-17 2:59 PM BST]
Alarm Source Info	Alarm Source Info
Monitor	Ping Monitor
Tuning Policies	Tuning Policy Map
Poll Scheduler	N Poll Scheduler
Tcp Config	Cylon Tcp Comm Config
Fault Cause	
Ip Address	192.168.1.151:4950
Ip Address	192.168.1.151
Port	unspecified 4950 [-1 - 65536]
Site	0
Uc32net Address	1
License	MCwCFAk5uC8sq4V/TusZs5zU/B9zRkJxAhQv9N7V

**Figure 1:** Cylon Network properties

## 5 Cylon Devices

There are two types of Unitron devices:

1. **Cylon Comm Device** – a communication controller with LAN connection. There is only one per network and always with address 0. This device has **Schedules** extension for global schedules.
2. **Cylon Device** – a field controller with **Points** extension.

Device properties:

- **Address** – unique address of the controller on the Unitron network.
- **Poll Frequency** – polling rate: Fast, Normal, or Slow, corresponding to the **Poll Scheduler** rates.

The screenshot shows the 'Cylon Discovery' window with a 'Success' status. It displays two tables: 'Discovered' and 'Database'.

Discovered			2 objects		
Model	Address	Serial			
UC32.net Lite, Internal Keypad, 1.01.52 (Apr 12 2010)	0	CNET831256G			
UC32.24 6. 1.6 13/11/07	1	CU24842123C			

Database						2 objects
Name	Type	Exts	Address	Model	Serial	
UC32.net.000	Cylon Comm Device		0	UC32.net Lite, Internal Keypad, 1.01.52 (Apr 12 2010)	CNET831256G	
UC32.24.001	Cylon Device		1	UC32.24 6. 1.6 13/11/07	CU24842123C	

**Figure 2:** Cylon device discovery

## 6 Cylon Points

Unitron field controllers points are identified by a type and an address from 1 to 1024 (some models less). Types can be:

1. Analog hardware

2. Digital hardware
3. Analog software
4. Digital software

Hardware points are physical inputs and outputs, depending on controller model and configuration. Software points are variables, they can be writable – also called setpoints – or read-only.

Writing to a software point *overwrites* its value, i.e., the “old” value will be lost. Writing to a hardware point *overrides* its value, so one can return it to the previous automatic value.

Each point proxy extension contains:

- **Point Type** – Analog Hardware, Digital Hardware, Analog Software, or Digital Software.
- **Address** – point address within the controller (1–1024).
- **Poll Frequency** – polling rate: Fast, Normal, or Slow, corresponding to the **Poll Scheduler** rates.

Each point is polled by JACE when it is required. Polling frequencies are specified for each point individually and reference polling rates defined in Poll Scheduler as in most Niagara drivers. This allows finding an optimal solution when reading a lot of points as often as suitable.

There are two ways to get Unitron points into Niagara: **Import File** and **Discover**.

## 6.1 Point Import from CSV File

**Import File** is a preferred method as it creates all points with names, units, limits and boolean facets. In order to utilize it, one should have access to Unitron project and export its points to CSV files in Cylon Database Interface software. Then in device **Points** extension press Import File and choose exported CSV.

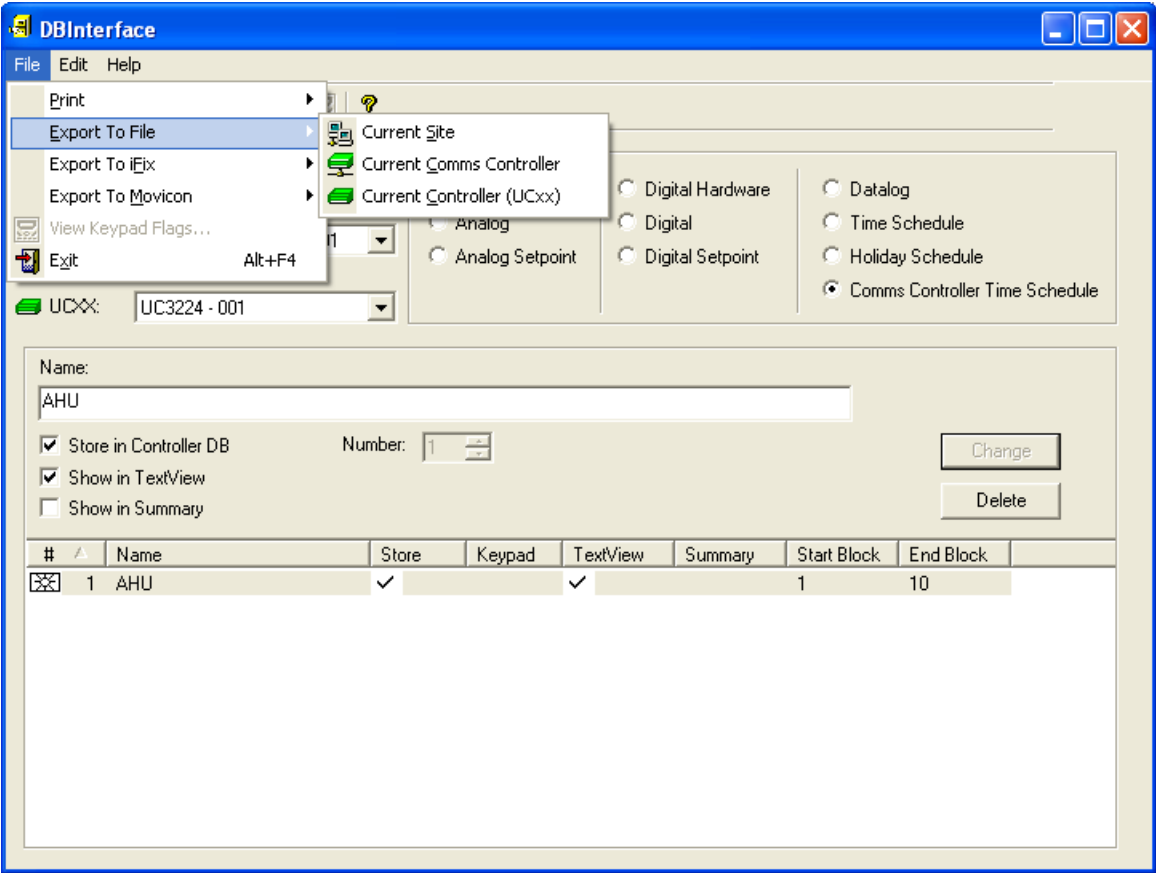


Figure 3: Export points to CSV file

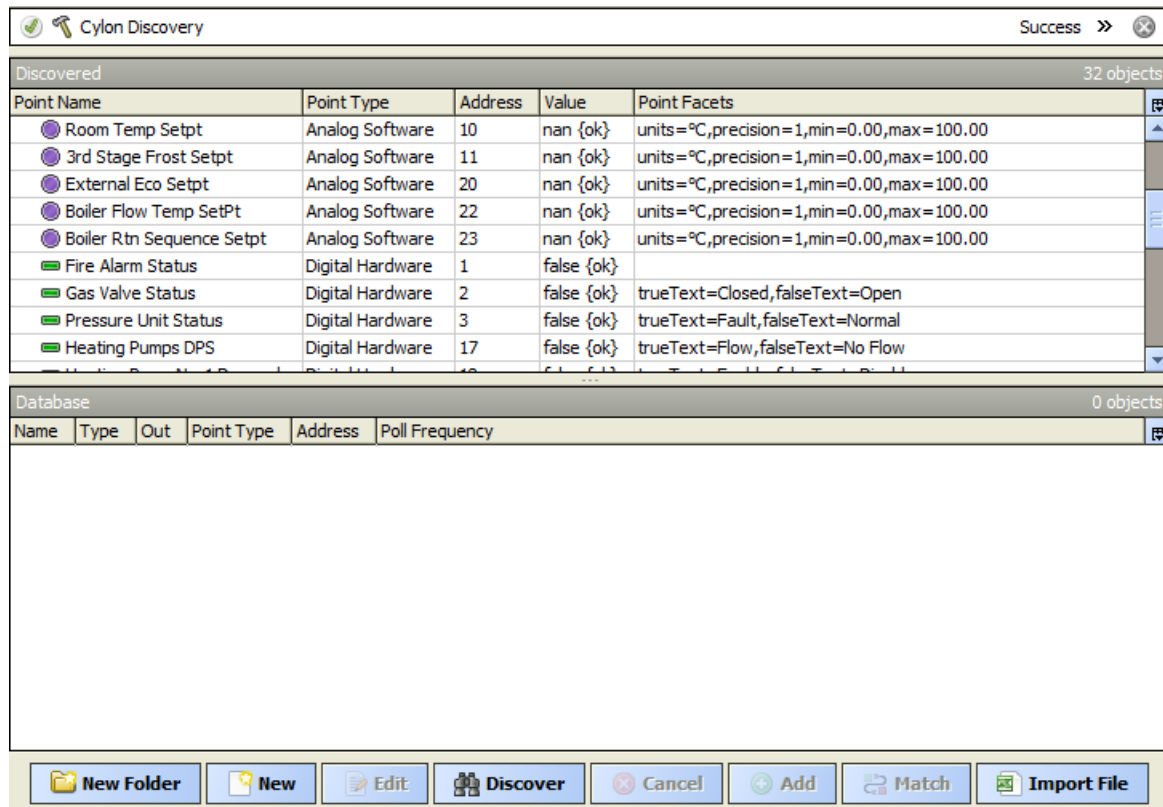
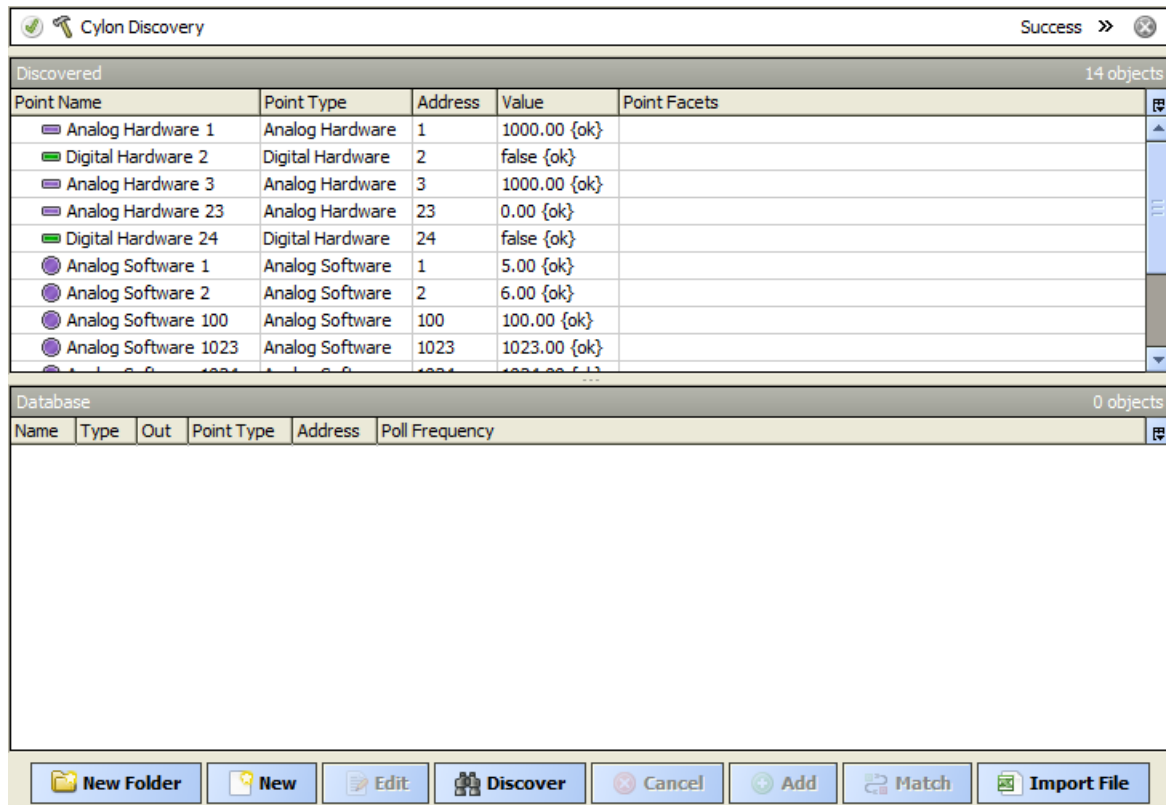


Figure 4: Import points from CSV file

## 6.2 Point Discovery

**Discover** is another method that allows retrieving some point information directly from the device. It can be used if Unitron project files are inaccessible. There are a few limitations in this method: read-only software points cannot be discovered and should be created manually; no point names or facets are available. In general this method still allows retrieving enough information to build all graphics, although it will require more investigation and engineering.



**Figure 5:** Point discovery

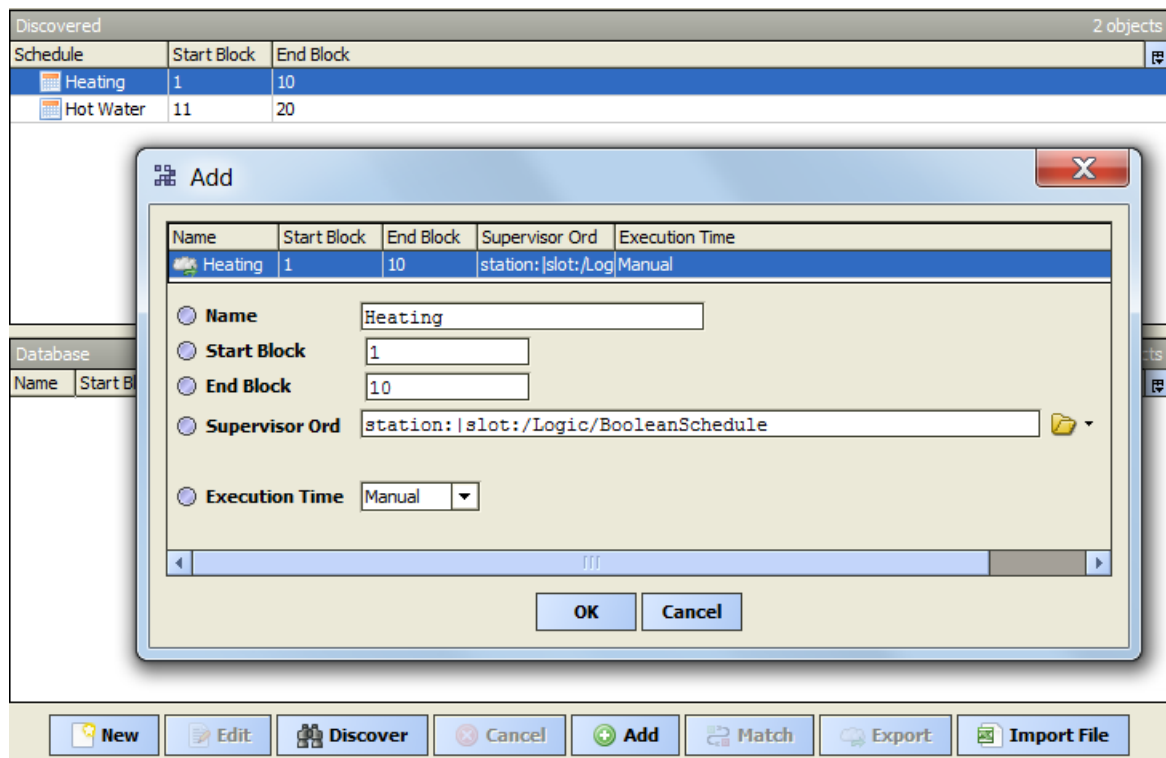
## 7 Schedules

Communication controllers store global time schedules and send them to field controllers. Each schedule occupies a number of *blocks* in its memory. Each *block* represents one weekly period for each day or one special date range.

Niagara can read a schedule from the Cylon device, map it into a standard **Boolean Schedule** and then write it to controller when necessary. These are the steps:

1. Create **Boolean Schedule** somewhere in Niagara station.
2. Open communication controller **Schedules** extension, **Import File** or create schedule manually (you will have to know start and end blocks, which this schedule occupies).
3. Select **Boolean Schedule** in **Supervisor Ord** property.
4. When the schedule is created, right-click on it and run **Read from controller** action. Open **Boolean Schedule** and check if it is filled properly.
5. Change some periods in **Boolean Schedule**, save it and then **Export** it back to controller. It can be done with the **Execute** action, **Export** button, periodically using **Execution Time** trigger or

by demand.



**Figure 6:** Schedule creation

Cylon global schedules are not identical to Niagara **Boolean Schedules**, so they cannot be mapped fully. There are a few limitations that should be taken into account, namely:

- when Cylon schedule exception dates are read into Boolean Schedule, all days will have the same time period
- when **Boolean Schedule** is written into controller, only **Day** and **Date Range** special events are used, both with exactly one time period; dates cannot have wildcards (\*)

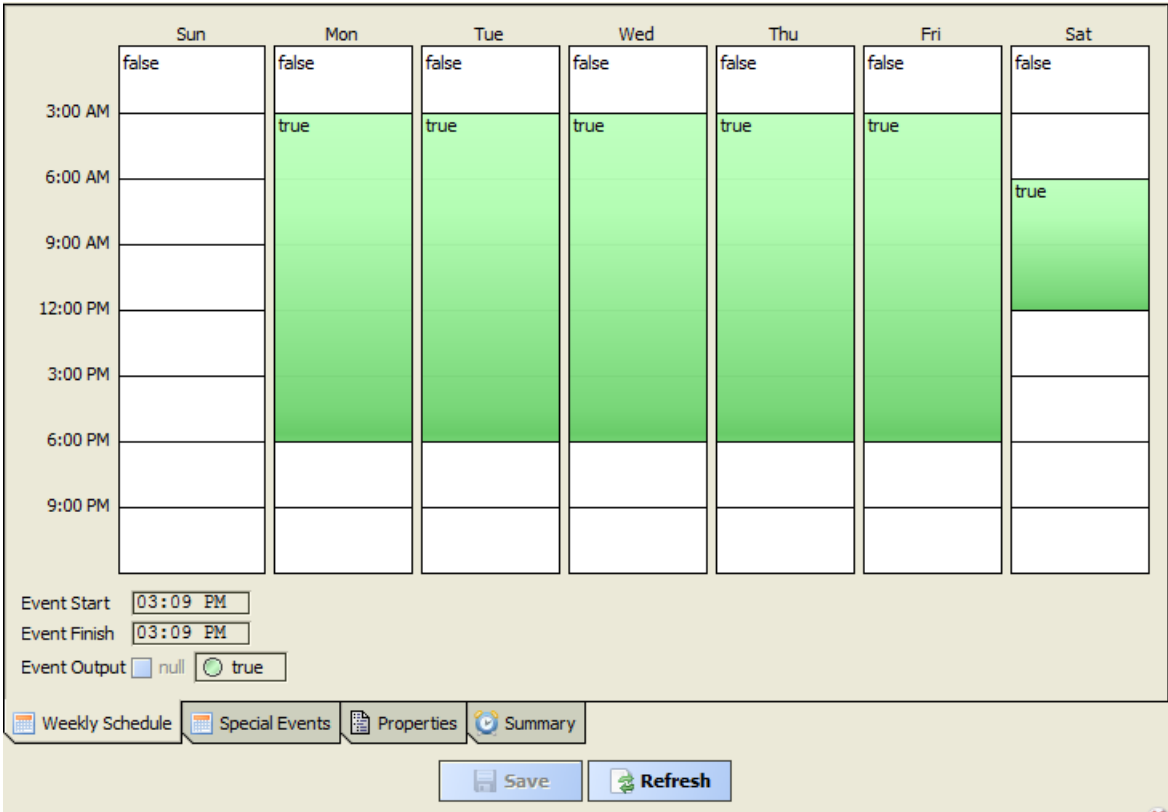


Figure 7: Weekly schedule

The screenshot displays a web-based interface for managing special days. At the top, there are navigation buttons: 'Prev Page', 'Prev Month', 'Today', 'Next Month', and 'Next Page'. Below these are three calendar grids for August, September, and October 2017. The 'Holidays' table is as follows:

Name	Summary
Holidays	Date Range: 24 Aug 2017 - 15 Sep 2017
New Year	Date: 1 Jan 2018

To the right of the table is a time slot selection grid:

Time	Schedule
3:00 AM	Unscheduled
6:00 AM	
9:00 AM	true
12:00 PM	
3:00 PM	
6:00 PM	
9:00 PM	

Below the table are controls for 'Event Start' (03:10 PM), 'Event Finish' (03:10 PM), and 'Event Output' (null, true). At the bottom, there are buttons for 'Add', 'Edit', 'Priority', 'Rename', and 'Delete', along with tabs for 'Weekly Schedule', 'Special Events', 'Properties', and 'Summary'. 'Save' and 'Refresh' buttons are at the very bottom.

Figure 8: Special days schedule

## 8 FAQ

### 8.1 I cannot discover any devices

- Verify the IP address and port (default 4950) in **Tcp Config** are correct.
- Check network connectivity: ping the UC32.net communication controller from the Niagara host.
- Verify the **Site** number and **UC32net Address** match the Unitron system configuration.
- Ensure the license is valid.

### 8.2 Points are not reading

- Verify the device is online (device status should show **ok**).
- Check that the point type and address match the controller configuration.
- If using **Discover**, remember that read-only software points cannot be discovered and must be created manually.

### 8.3 Schedule export is not working

- Ensure the schedule start and end blocks are correct.
- Verify the **Boolean Schedule** is linked in the **Supervisor Ord** property.
- Run **Read from controller** first to verify communication before attempting an export.
- Remember that only **Day** and **Date Range** special events with one time period are supported for export.