

Polisy/eISY Installation and User Notes

Contents

(Re) Installing the ISY Launcher.....	2
Password Notes	3
Updating the Polisy OS/modules	4
Installing Nodes.....	5
Using Nodes in the Polisy Admin Console	9
One-Time Publishing the node data	9
What to do if the Node Data does NOT appear.....	10
Using the Node data in your Programs.....	11
Harmony Hub Node – Important Information.....	12
OpenWeatherMap Node – Important Information.....	13
PG3 Node Logs – Important Information.....	13
Resetting Nodes Programmatically.....	13
Understanding Program Flow	14
Schedule conditions	15
Condition conditions.....	16
Node Delay Complications.....	17
Under the Hood Explanation	18
Thread Termination on Event becoming false.....	19
Solution	20
Known Issues as of 4/2022	21
Using Admin Console to set Z-Wave device clock fails.....	21
Notifications (with Variables) Fail to preserve Variable States	22
Notifications using “alert.event” variable don’t work properly	23
Notify "Content" drop down is not populated at startup.....	24
Admin Console, program delete function is non-intuitive	24

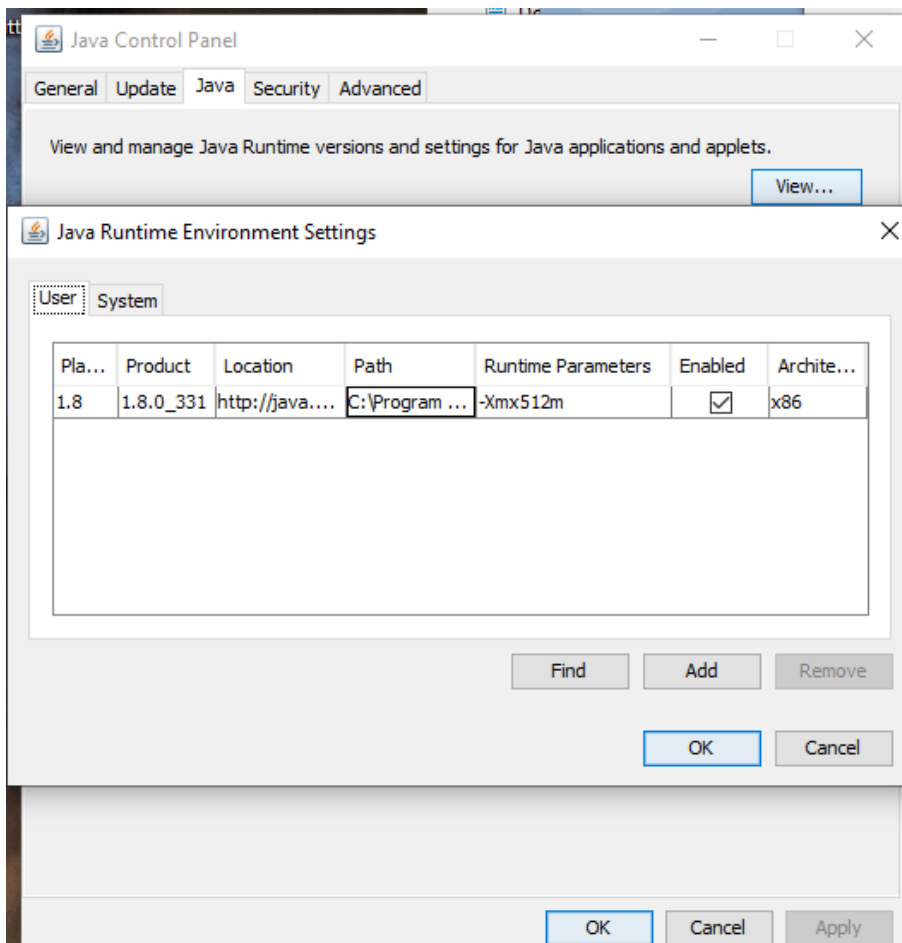
(Re) Installing the ISY Launcher

You will need the ISY Launcher installed to be able to access and configure the Polisy device.

Follow the instructions on the UDI web site here: [Universal Devices, Inc. Wiki \(universal-devices.com\)](http://universal-devices.com)

It is critical that you clear the java cache – do not skip this step – especially when re-installing after updating modules.

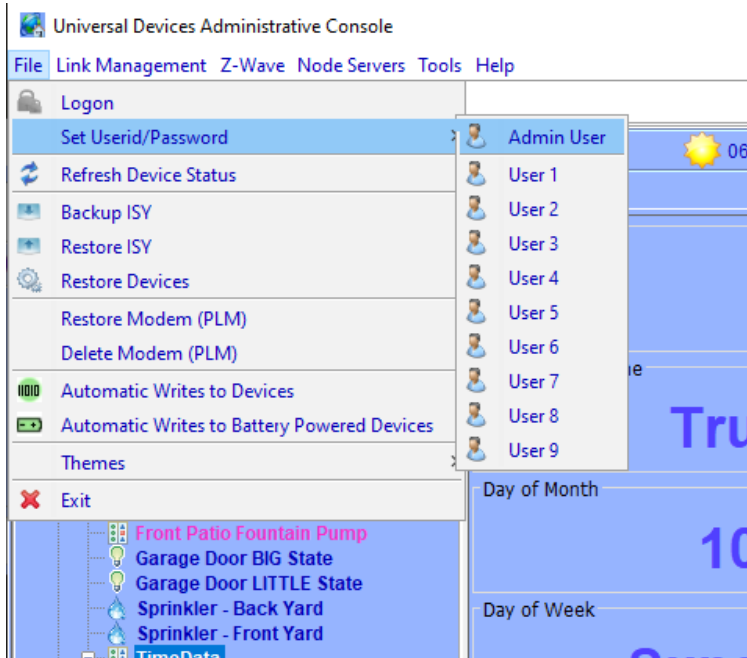
It is also critical that you increase the amount of memory available to the Java Runtime (JRE) per the [wiki page](#). Make sure that your runtime parameters include (as shown below): -Xmx512m.



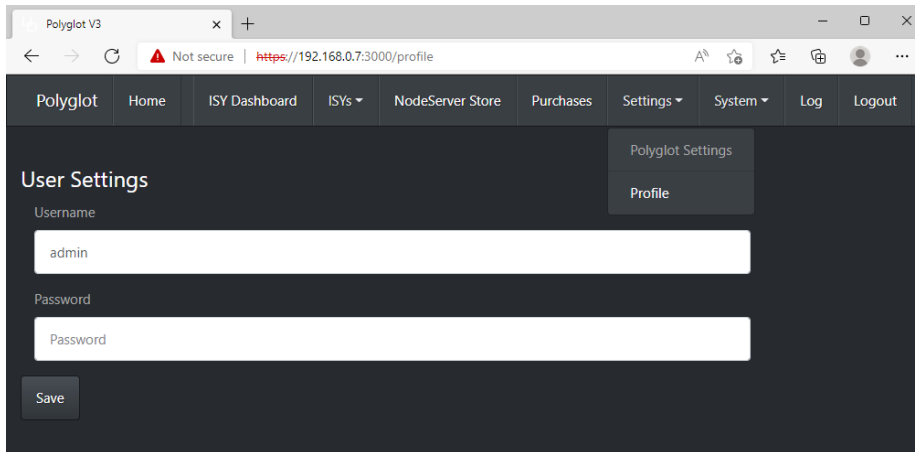
Failure to do this may result in random functions hanging for long periods of time.

Password Notes

If you change the admin name/password in the Admin Console:



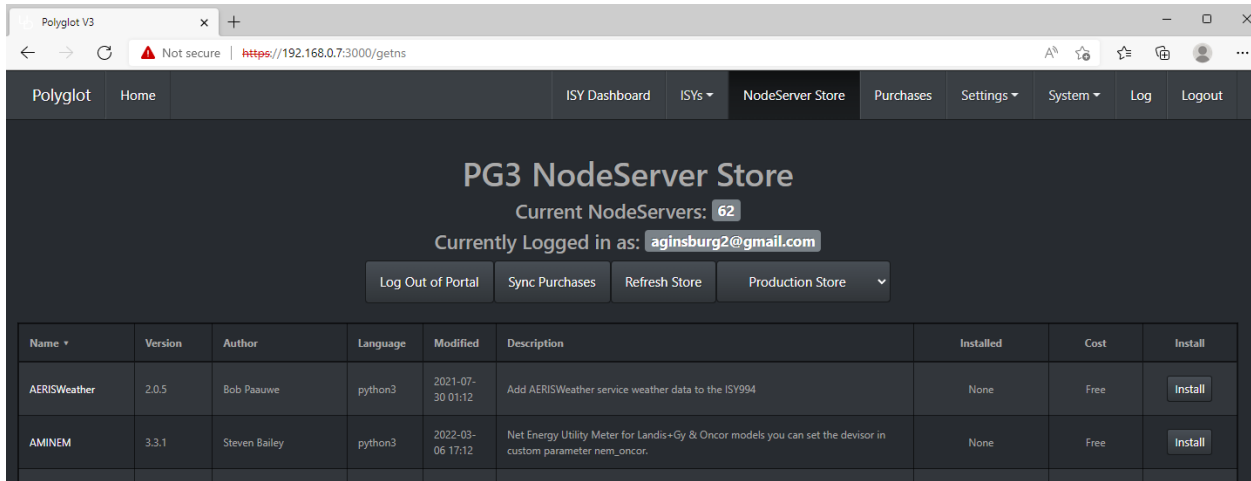
You need to ALSO change it in the PG3 configuration:



Otherwise installation of nodes will fail to be able to configure the Admin Console and inform it that new nodes are available.

Installing Nodes

Open PG3, and click the “Refresh Store” button followed by the “Sync Purchases” to make sure the data displayed is up to date. You would think this would be built in, but it is not.



The screenshot shows the PG3 NodeServer Store interface. At the top, there is a navigation bar with links for Polyglot, Home, ISY Dashboard, ISYs, NodeServer Store, Purchases, Settings, System, Log, and Logout. The main content area displays the store title, the number of current nodes (62), and the user's login information (aginsburg2@gmail.com). Below this, there are buttons for Log Out of Portal, Sync Purchases, Refresh Store, and Production Store. A table lists available nodes with columns for Name, Version, Author, Language, Modified, Description, Installed, Cost, and Install.

Name	Version	Author	Language	Modified	Description	Installed	Cost	Install
AERISWeather	2.0.5	Bob Pasauwe	python3	2021-07-30 01:12	Add AERISWeather service weather data to the ISY994	None	Free	Install
AMINEM	3.3.1	Steven Bailey	python3	2022-03-06 17:12	Net Energy Utility Meter for Landis+Gy & Oncor models you can set the divisor in custom parameter nem_oncor.	None	Free	Install

Find the node you are interested in by looking at the descriptions. If it is free, or you have already purchased it, you can click the “Install” button on the right to install it.

If you need to purchase it first, then click the “Purchase” button on the right. This will walk you through some screens where you can put in your credit card information. Eventually, when you are done, it will return to the screen above, and the “Purchase” button will have changed to an “Install” button. If this does not occur, then try pressing the “Sync Purchases” button at the top.

Once you click the “Install” button you will see a screen as follows:

Install AMINEM?

NodeServer:

Name	AMINEM
Language	python3
Version	3.3.1
Price	Free

Target ISY:

Alias	ISY
UUID	00:0db9:59:3f:dc
Firmware	5.4.2

Do you wish to install this NodeServer on to the target device?

NOTE: If you are trying to upgrade a NodeServer that is already installed, stop and re-start it from the dashboard. It will automatically load the newest version available.

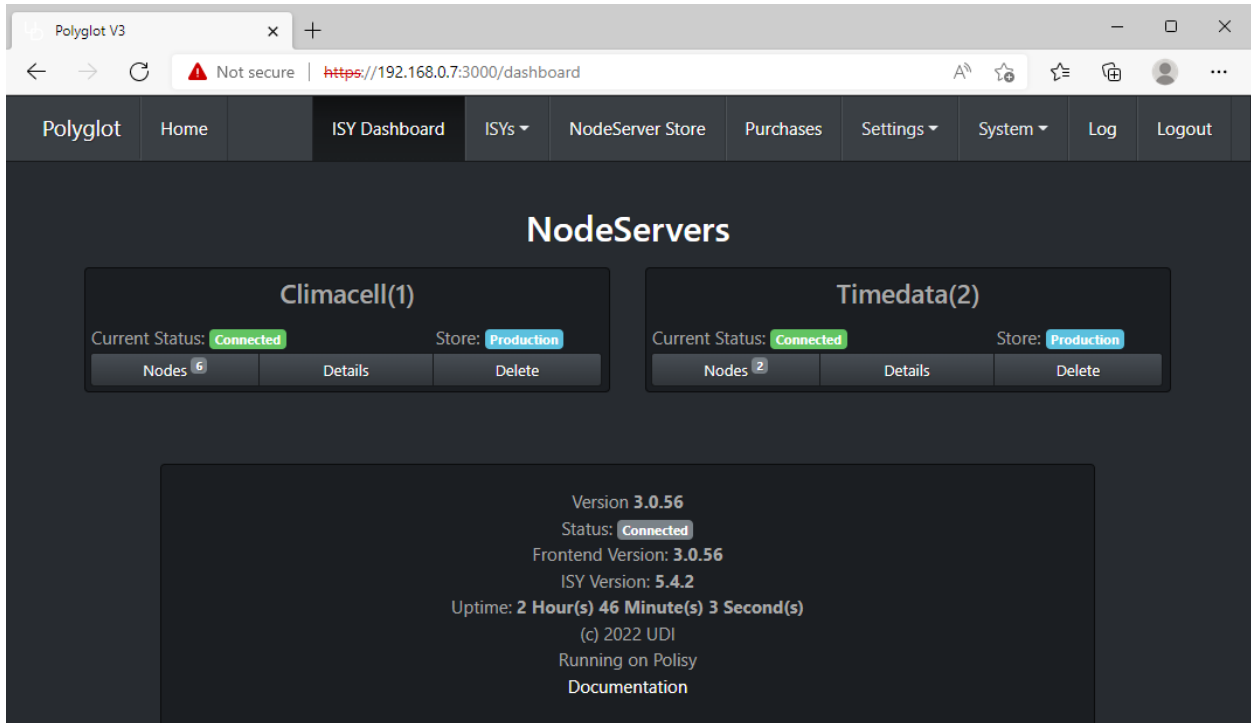
Select Available Slot:

Select Slot

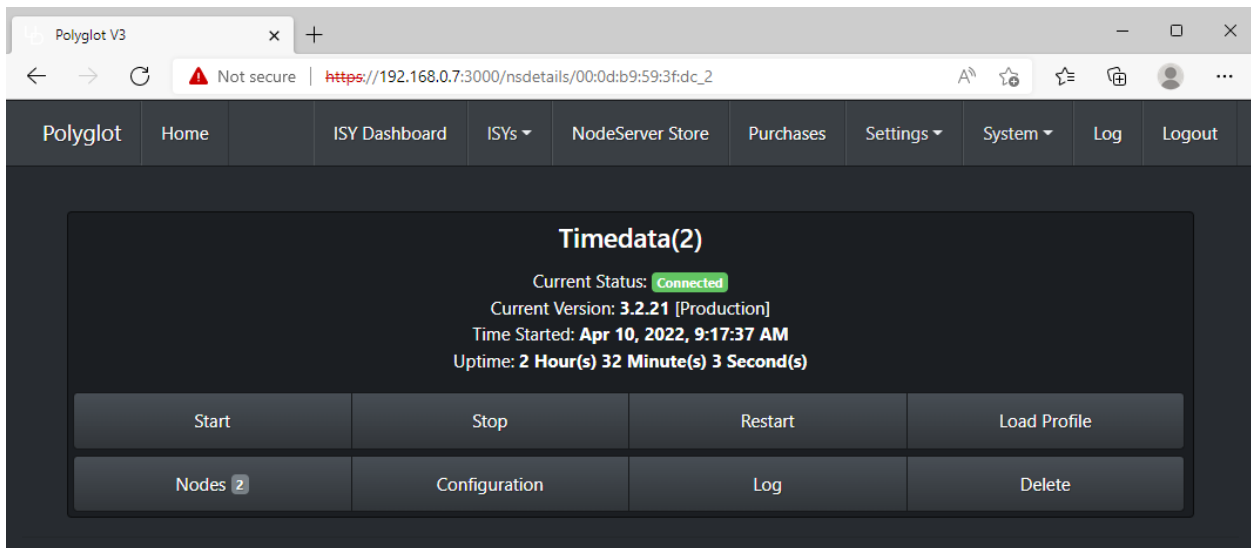
Install Cancel

You now have to decide which node “Slot” (1-25) you want to use for this node. Only one node can occupy a slot at any time. The drop down will show you all the currently “empty” slots you can choose from. One slot is no different from another slot so it is personal preference where you want to put it. When you have decided, press the “Install” button and the node will be installed.

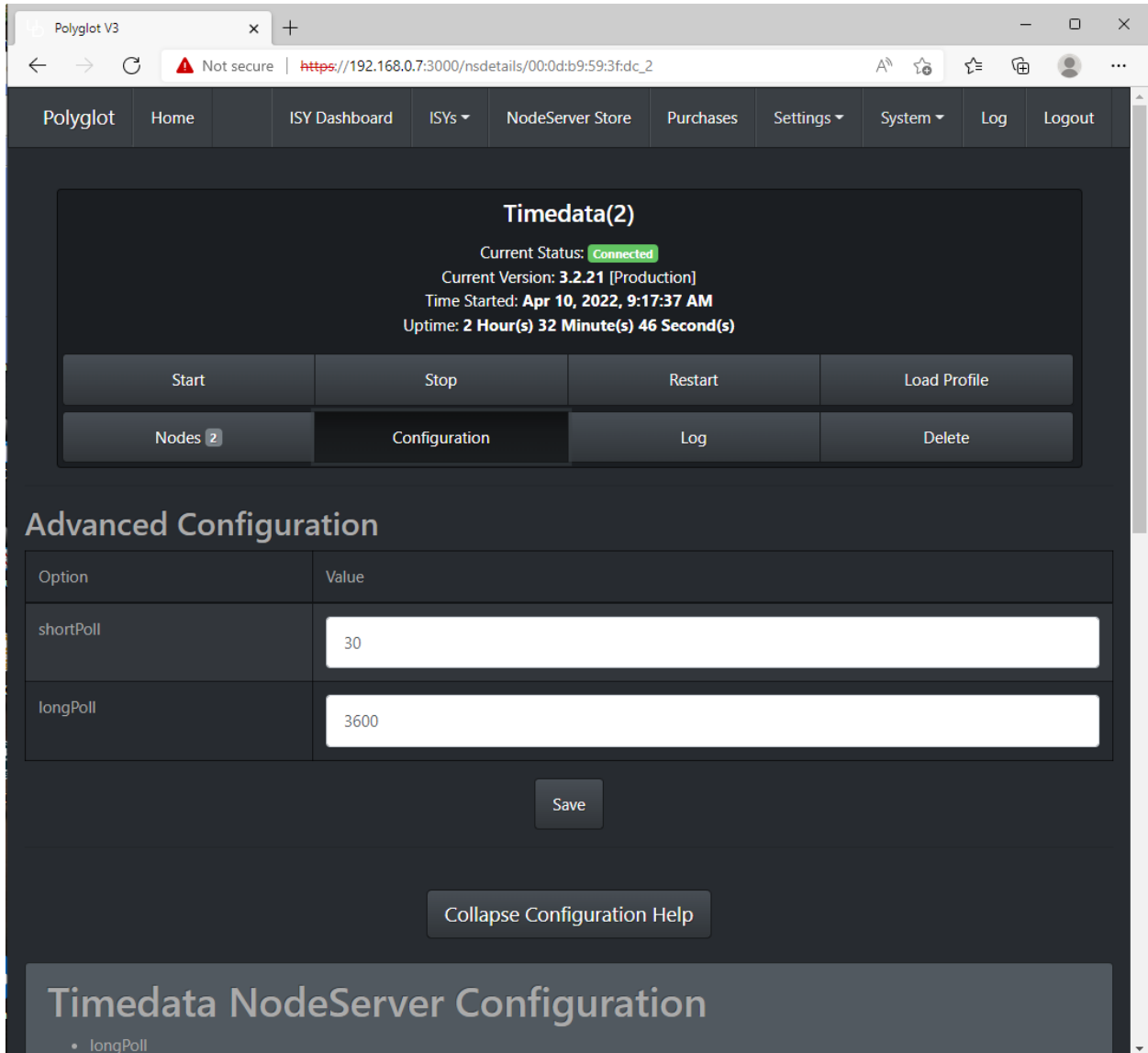
Once the node has been installed you will need to configure it. Go to the “ISY Dashboard” and find the node you just installed in the list of NodeServers and click on its “Details” button.



You will a screen like the following:



Click on the “Configuration” button and you should see a screen like the following:



You now have to configure your node. Each node will have different requirements. Once you have entered the configuration values, make sure to press the “Save” button.

To get more information about the configuration you can go back to the “NodeServer Store” and click on the node “Name” (in the left hand column) which will bring you to a web site describing details about that node.

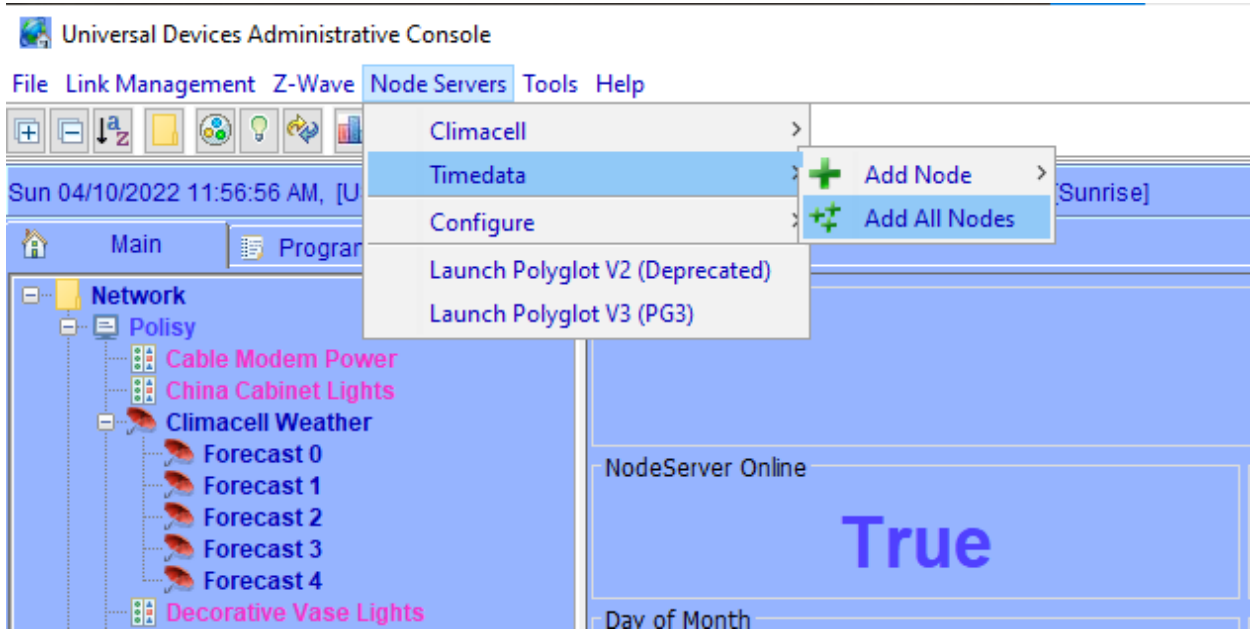
If, after configuring the node, its “Current Status” is not “Connected” then something is wrong and you will need to troubleshoot it or contact support.

Using Nodes in the Polisy Admin Console

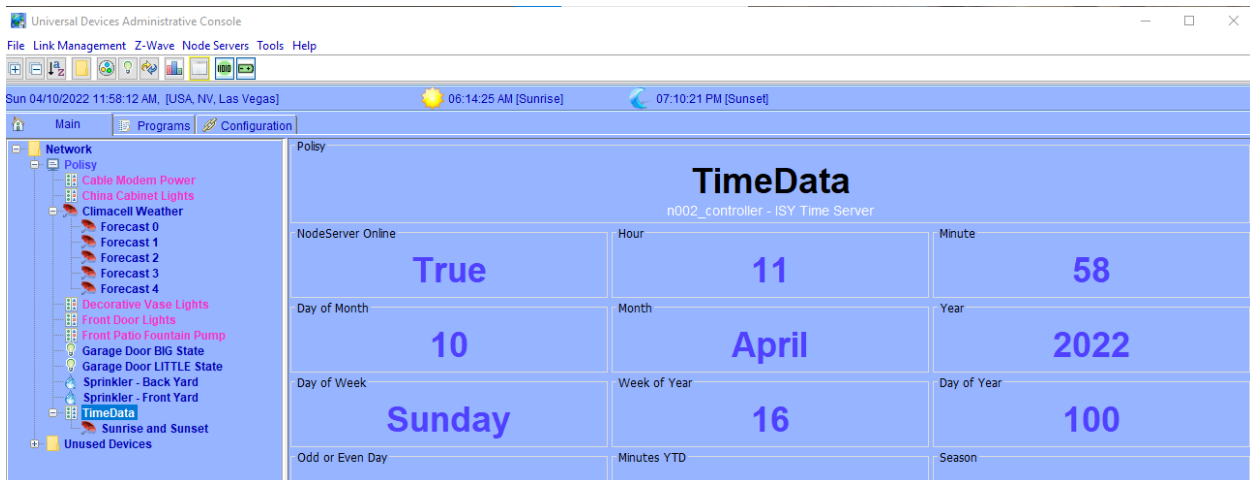
One-Time Publishing the node data

Once you have a node installed (through PG3) you can start using its features in the Admin Console. Note that this only has to be done once.

The most common way to do this is to simply add all of the 'node's nodes to the Admin Console as follows:



After doing this, you should see something appear in Polisy node network (on the left) – in this case “TimeData.” If you click on this, you will see in the right pane all of the data that the node has published to the Polisy device.



What to do if the Node Data does NOT appear

If the node was not properly installed, configured, or for some reason crashed, no node data will appear.

If this occurs you will need to go back to the “ISY Dashboard” in PG3 and check its status – it must be “Connected” to work. Additionally, you can check the “Log” which often shows errors that point you in the right direction (ex: configuration data missing, etc.)

The screenshot shows a web browser window with the URL https://192.168.0.7:3000/nsdetails/00:0d:b9:59:3f:dc_2. The dashboard has a navigation bar with links: Polyglot, Home, ISY Dashboard (active), ISYs, NodeServer Store, Purchases, Settings, System, Log, and Logout. The main content area displays the following information for 'Timedata(2)':

- Current Status: **Connected**
- Current Version: **3.2.21** [Production]
- Time Started: **Apr 10, 2022, 9:17:37 AM**
- Uptime: **2 Hour(s) 50 Minute(s) 35 Second(s)**

Below this information is a grid of control buttons:

Start	Stop	Restart	Load Profile
Nodes 2	Configuration	Log	Delete

The 'Log' button is highlighted. Below the grid is a section titled 'Real-time Timedata log file' with buttons for 'Scroll to Bottom', 'Download Log', 'Download Log Package', and 'Info'. There is also a checkbox for 'AutoScroll' which is checked. The log file content is as follows:

```
2022-04-10 12:00:13,076 MQTT udi_interface.interface INFO interface:_message: Successfully set controller :: GV1 to 0 UOM 0
2022-04-10 12:00:13,158 MQTT udi_interface.interface INFO interface:_message: Successfully set controller :: GV9 to 143280 UOM 0
2022-04-10 12:00:13,258 MQTT udi_interface.interface INFO interface:_message: Successfully set controller :: GV14 to 2388 UOM 0
2022-04-10 12:01:13,003 MQTT udi_interface.interface INFO interface:_message: Successfully set controller :: GV1 to 1 UOM 0
2022-04-10 12:01:13,098 MQTT udi_interface.interface INFO interface:_message: Successfully set controller :: GV9 to 143281 UOM 0
```

Using the Node data in your Programs

You can then use this data in any of your programs as shown in the example below where the program is checking the “TimeData” node’s “Month” node to see if it is currently “November”:

The screenshot displays the Universal Devices Administrative Console interface. The top status bar shows the date and time: Sun 04/10/2022 12:00:44 PM, [USA, NV, Las Vegas], with sunrise at 06:14:25 AM and sunset at 07:10:21 PM. The main window is titled 'Programs' and shows a list of programs on the left, including 'Time To Run Sprinklers Improved' which is highlighted. The right pane shows the configuration for this program, titled 'Program Content for 'Time To Run Sprinklers Improved''. The configuration is as follows:

```
If (
  (
    On Fri
    Time is 12:00:00PM
    And (
      'TimeData' Month is November
      Or 'TimeData' Month is December
      Or 'TimeData' Month is January
      Or 'TimeData' Month is February
    )
  )
  Or (
    On Mon, Fri
    Time is 9:00:00AM
    And (
      'TimeData' Month is March

```

Below the program content, there is an 'Add To Program' section with tabs for 'Schedule', 'Condition', 'Action', and 'Comment'. The 'Condition' tab is active, showing a configuration for a condition:

Or [Status] [TimeData] [Month] is [November]

Buttons for 'Add to 'If'', 'Update', 'Remove Line', 'Move Line Up', and 'Move Line Down' are visible.

Harmony Hub Node – Important Information

Installing this node is a nightmare of poorly documented requirements and broken links.

You MUST do the following steps BEFORE you install the node. If you have already installed the node, you can probably do the following and then re-start the node, but I haven't tried it.

First of all XMPP must be enabled on your Harmony Hub. You must follow these steps:

1. You MUST download the Harmony App to your phone – it will not work from the PC Desktop Harmony App!
2. You MUST navigate a cryptic set of menus to find where to enable this as follows:
 - a. Hit the 3 bar icon in the top left of the app (the menu button essentially)
 - b. Press “Harmony Setup”
 - c. Press “Add/Edit Devices & Activities”
 - d. Press “Remote & Hub”
 - e. Press “Enable XMPP”
 - f. Agree to all of the warnings and disclaimers about security breeches
 - g. Now XMPP will be enabled, and the “Enable XMPP” option will change to “Disable XMPP”

Make sure that your Harmony Hub is assigned a static IP address in your router before you install the node.

If successful, you should see something like the following (different of course for “your” Harmony setup):



IMPORTANT NOTE: after the ISY has been restarted – you MUST “restart” this node (through PG3) otherwise it will cease to function properly.

OpenWeatherMap Node – Important Information

After the ISY has been restarted – you MUST “restart” this node (through PG3) otherwise it will cease to function properly. It continues to report its status as running, but none of the weather values will ever change again. It is as though the connection to the weather service server has broken and it does not know how to re-establish it.

PG3 Node Logs – Important Information

Logs are rotated every day at midnight.

Past logs can be obtained using ssh. They are located in `/var/polyglot/pg3/ns/<ns folder>/logs`

Resetting Nodes Programmatically

There is currently no built-in way to programmatically restart a node server.

However, know that a service can be restarted using this command:

```
sudo service uuid_slot onerestart
```

Example, to restart a nodeserver in slot 1 of my loX: `sudo service 0021b9025fc2_1 onerestart`

If you want to trigger this using an http call, chatgpt can help you build a python script that could run on your system.

Understanding Program Flow

The most confusing thing to understand is how the program flow works – and I used to write low level hardware drivers for a living! After much trial and error, here is what I have found to be useful.

All programs are designed as:

If (some set of conditions) then (do some actions) else (do other actions)

The confusing part is “when” will the “set of conditions” be evaluated.

There are two different types of conditions: “Schedule” and “Condition” as seen below – which operate quite differently.

The screenshot displays a software interface for configuring a program. The top section, titled "Program Content for 'New Program'", shows an "If" statement with the condition "Time is 5:41:00PM". The "Then" block contains the action "\$Test += 1", and the "Else" block contains "\$Test -= 1". Below this is the "Add To Program" section, which includes four buttons: "Schedule", "Condition", "Action", and "Comment". The "Schedule" button is circled in red. Underneath, the "Schedule" configuration panel is visible, featuring a dropdown menu set to "And", a row of checkboxes for days of the week (All, Saturday, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday), a time selection area with dropdowns for "Time is", "05", ":", "45", ":", "00", "PM", and "Time", and a checked "Daily" checkbox.

Schedule conditions

Schedule conditions are based on time/date criteria, and sadly you cannot use partial time/date data (such as if 'month' == 'january') – you need to use the TimeData node to do this.

A schedule condition:

- is evaluated ONCE when the condition initially becomes true
- using “Time is” will never terminate the thread
- using “From/To” or “From/For” will terminate the thread when the “To/For” become true

Program Content for 'New Program'

```
If
  Time is 5:41:00PM
Then
  $Test += 1
Else
  $Test -= 1
```

Remove Line

Add To Program

Schedule Condition Action Comment

Schedule

And

On Days: All Saturday Sunday Monday Tuesday Wednesday Thursday Friday

Time is 05 : 41 : 00 PM Time

Daily

In the example above, the 'schedule' condition (Time is 5:41:00PM) will be evaluated once at 5:41:00PM causing the “Then” logic to run and increment the “Test” variable.

Note that the “Else” will never run in this case because the only time the condition is evaluated is when it is true.

Condition conditions

Condition conditions are either “static” (Comment, Variable if the variable is an ‘integer’ and NOT a state) ‘or “events” (all other cases.)

A Condition condition:

- that is an EVENT is evaluated EVERY TIME the condition changes state (i.e. has a new value)
- that is STATIC is NEVER evaluated; it only participates when something else is evaluated.

The screenshot displays two parts of a software interface. The top part, titled "Program Content for 'New Program'", shows a code editor with the following content:

```
If  
  'TimeData' Minute < 27 minute  
Then  
  $Test += 1  
Else  
  $Test -= 1
```

Below the code editor is a "Remove Line" button. The bottom part of the screenshot is a dialog box titled "Add To Program". It has four tabs: "Schedule", "Condition", "Action", and "Comment". The "Condition" tab is selected. Under the "Condition" heading, there is a configuration area. It starts with an "And" dropdown menu. To its right is a dropdown menu with "Status" selected. Below this is a list of options: "Status", "Control", "Comment", "Program", "X10", "Variable", and "Module". To the right of the list is a text input field containing "TimeData". Below this field is a configuration row with a "Minute" dropdown, a "<" button, a ">" button, and a "27 minute" dropdown.

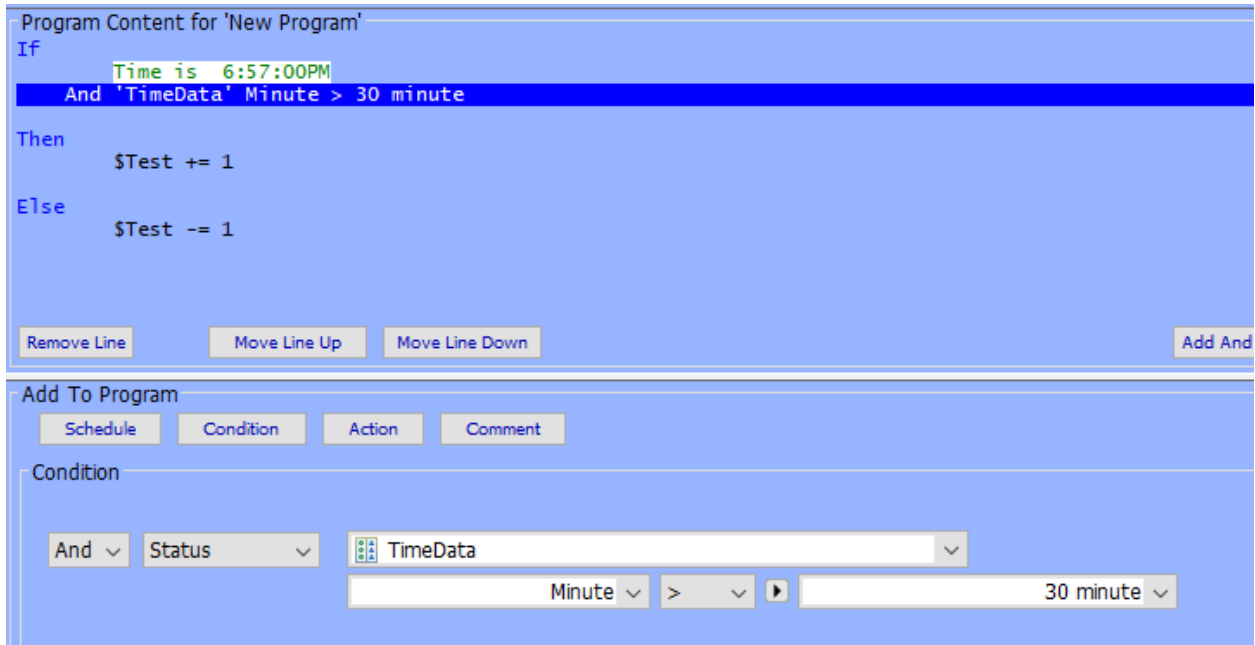
In the example above “TimeData Minute” is an event published by the TimeData node every minute when the minute value changes. Therefore, it will be evaluated every minute.

In the example above, the “Test” variable will increment at the beginning of every new minute from 00-26, and decrement at the beginning of every new minute from 27-59.

Node Delay Complications

This can get very confusing when you start combining “schedule” conditions with node data “events.”

To illustrate, take this simple example which “looks like” it will increment “Test” by one at 6:57:00PM, but the reality is quite different.



Program Content for 'New Program'

If

Time is 6:57:00PM

And 'TimeData' Minute > 30 minute

Then

\$Test += 1

Else

\$Test -= 1

Remove Line Move Line Up Move Line Down Add And

Add To Program

Schedule Condition Action Comment

Condition

And Status TimeData Minute > 30 minute

What actually happens is that:

1. The Test variable “appears” to be unchanged at 6:57:00PM
 - a. But , in reality, it very quickly incremented, then decremented by one
2. Every other start of a new minute, the Test variable is decremented by one

Item (1) is very perplexing, until you understand what is going on under the hood as shown below.

Under the Hood Explanation

In this case, you have to realize that the “TimeData Minute” event is computed from (and thus takes time to calculate) the ISY time used in the schedule condition (Time is 6:57:00PM.)

So what is really happening under the hood is the following:

- At 6:57:00pm,
 - the schedule condition (Time is 6:57:00PM) is evaluated (causing the ‘if’ to be run) and found to be true
 - TimeData Minute (which is still 56 at this point) is > 30
 - So both conditions in the ‘if’ being true, the ‘then’ code is run and Test is incremented
- At 6:57:00pm + TimeData node time to process the time update (about 1-2 seconds)
 - TimeData increments its Minute event value from 56 to 57
 - This causes the Condition condition (‘TimeData’ Minute > 30 minute) to be evaluated (causing the ‘if’ to be run again) and found to be true
 - However, because of the node processing time, the schedule condition (Time is 6:57:00PM) is no longer true – it is now 6:57:01
 - So only one condition is true in the ‘if’ and the ‘else’ code runs and Test is decremented
- At 6:58:00pm + TimeData node time to process the time update (about 1-2 seconds)
 - TimeData increments its Minute event value from 57 to 58
 - This causes the Condition condition (‘TimeData’ Minute > 30 minute) to be evaluated (causing the ‘if’ to be run again) and found to be true
 - the schedule condition (Time is 6:57:00PM) is no longer true – it is now 6:58:01
 - So only one condition is true in the ‘if’ and the ‘else’ code runs and Test is decremented

Thread Termination on Event becoming false

You have to be very careful when using “Wait” delays inside your logic because your thread can be terminated when the “if” condition becomes false.

In the following example, the intention is to accumulate the “Rain Rate” value every ten minute for the first 50 minutes of each hour, starting at the beginning of the hour.

```
Program Content for 'Log RainMM'  
If  
  '"TimeData" Minute is 0 minute'  
Then  
  $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
  Repeat 5 times  
    Wait 10 minutes  
    $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
  Repeat 1 times  
    $RainSampleMM /= 6  
    $RainMM += $RainSampleMM  
    $RainSampleMM = 0
```

Remove Line

Add To Program

Schedule Condition Action Comment

Condition

And Status TimeData Minute is 0 minute

The problem here is that during the first “Wait 10 minutes” command, the “if” condition becomes false (TimeData Minute changes from 0 to 1) which causes the entire thread to be terminated. This results in only one sample (instead of the 6 samples spaced out every 10 minutes.)

You might think the way to solve this to expand the time check as follows:

```
Program Content for 'Log RainMM'  
If  
  And '"TimeData" Minute >= 0 minute  
  And '"TimeData" Minute <= 50 minute'  
Then  
  $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
  Repeat 5 times  
    Wait 10 minutes  
    $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
  Repeat 1 times  
    $RainSampleMM /= 6  
    $RainMM += $RainSampleMM  
    $RainSampleMM = 0  
Else  
  - No Actions - (To add one, press 'Action')
```

However, this causes an even more insidious problem. Now (because TimeData Minute is an event that changes every minute) every minute, for the first 50 minutes of every hour the following will occur:

- The code will run and start waiting in the first “Wait 10 minutes” command
- After 60 seconds, TimeData Minute will update to its new value which causes the following:
 - The code thread that had run and was waiting will be terminated
 - A new code thread will be run, which will again run into the same problem while it is waiting.

So, instead of accumulating the Rain Rate every 10 minutes it will be accumulated every minute!

Solution

The way to solve this is to break the code into two programs as follows:

```
Program Content for 'Time To Log RainMM'  
If  
    'TimeData' Minute is 0 minute  
Then  
    Run Program 'Log RainMM' (Then Path)  
Else  
    - No Actions - (To add one, press 'Action')
```

```
Program Content for 'Log RainMM'  
If  
    - No Conditions - (To add one, press 'Schedule' or 'Condition')  
Then  
    $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
    Repeat 5 times  
        Wait 10 minutes  
        $RainSampleMM += 'Climacell Weather' Rain Rate mm/hr  
    Repeat 1 times  
        $RainSampleMM /= 6  
        $RainMM += $RainSampleMM  
        $RainSampleMM = 0  
Else  
    - No Actions - (To add one, press 'Action')
```

Now the first program runs at the top of every hour and “runs” (in another thread) the second program which will run to completion without being terminated.

Known Issues as of 4/2022

Using Admin Console to set Z-Wave device clock fails

Using the “Set Time” function fails to take the Time Zone into effect and always sets the time to GMT. This has been reported and should be fixed in a future version.

The screenshot displays the Admin Console for a thermostat device. The main title is "Thermostat Bedrooms" with the device ID "ZW005_1 - Thermostat". The interface is divided into several sections:

- Temperature:** 75°F
- Heat Setpoint:** 71°F
- Cool Setpoint:** 74°F
- Energy Save Heat Setpoint:** 62°F
- Energy Save Cool Setpoint:** 85°F
- Mode:** Cool
- Fan Mode:** Auto
- Fan Mode Override:** No Override
- Fan State:** Off
- Heat/Cool State:** Idle

Below these settings, there are input fields for "Heat Setpoint", "Cool Setpoint", "Energy Save Heat Setpoint", and "Energy Save Cool Setpoint", each with a dropdown menu showing the current value. To the right, there is a "Membership" section showing "Thermostat Bedrooms" as an "Is Responder to".

At the bottom of the interface, there are three buttons: "Query", "Set Time", and "Write Changes". The "Set Time" button is circled in red.

Notifications (with Variables) Fail to preserve Variable States

Code such as the following will report the wrong value in the notification (var.1.7 is \$EvaporationMM):

```
Program Content for 'Test'
If
- No Conditions - (To add one, press 'Schedule' or 'Condition')

Then
  $EvaporationMM = 11
  $RainMM = 9
  Send Notification to 'Default' content [Entry 3]
  $EvaporationMM -= $RainMM
  $LastEvapMM = $EvaporationMM

Else
- No Actions - (To add one, press 'Action')
```

Custom Notification 'Evaporation' [3]

From: First-Name Last-Name:Email-Address (Omit to use default)

Subject: Evaporation Update

Body: Evaporation since last irrigation cycle is \${var.1.7}mm
Rain since last irrigation cycle is \${var.1.5}mm

Text HTML XML

Alert | Default Subject | Add Variable

Ok | Cancel

Instead of sending '11' the notification will report '2' (11 – 9.)

This has been reported to the vendor and I have not heard back from them yet.

To work around this problem, simply use a temp variable and display its value instead of the original variable that gets modified. Or, alternatively, place a short (a few seconds) “Wait” action after the “Notification” and before your other code – this allows the notification to run before the other code is run.

Notify "Content" drop down is not populated at startup

Notify "Content" drop down is not populated with custom content (and only shows "Default") when the Admin Console is initially started. Example: start Admin Console, create a new program and immediately add a "Notify" command to the "Then" section - only "Default" will show up in the Content drop down. To resolve this, you have to exit out of the "Programs-Details" tab, switch to the "Configuration-Customizations" tab, and then switch back to the "Programs-Details" tab - now the "Content" drop down is populated with all the custom content.

Admin Console, program delete function is non-intuitive

When making changes in the "Program-Details" tab all changes (adding programs, editing programs, ...) appear in the left of the tab with a visual indication that they have been modified. Then when the "Save Changes" button is pressed, the visual indication goes away informing the user that the change has been processed and accepted.

The "Delete Program" function does not follow this pattern and is confusing to the user. When a program is deleted, it is immediately removed from the left of the tab leaving the user with the impression that it has actually been removed, AND that there is no need to press the "Save Changes" button. In fact, the "Save Changes" button DOES need to be pressed, and if the user exits and later comes back, they are confused as to why the deleted programs are still there.

I suggest that either: (a) the program is actually deleted when the "Delete Program" button is pressed, or (b) the program is NOT removed from the tab when the "Delete Program" button is pressed, instead it would be flagged with some visual cue (perhaps a red 'X') indicating that it is flagged for removal.

This will be resolved in when the Admin Console is replaced with the next generation UI.