



Make the Invisible Visible™

syncromesh 

Integrating Syncromesh IoT network with ICONICS

An ICONICS Whitepaper

April 2021

Version 1.0

Contents

1	Introduction	4
2	Architecture	5
3	Azure infrastructure setup.....	6
4	Device setup on Azure IoT Hub.....	6
5	IoT Subscriber Connection	7
6	AssetWorX configuration	8
6.1	Create the equipment class	8
6.2	Create telemetry data properties	9
6.3	Update class parameter definitions.....	9
6.4	Instantiate equipment	10
7	AnalytiX-BI configuration	11
7.1	Create the Data Flow	11
7.2	Create the Data Model.....	12
8	BridgeWorX64 configuration	13
8.1	Create the Transaction.....	13
8.2	Configure the Transaction Template	14
8.3	Trigger transaction with IoT Hub messages.....	16
9	Visualize the IoT device data.....	18
10	Conclusion.....	18

Copyright and Confidentiality

By accessing and using the instructions within (the “instructions”), you acknowledge and agree, on your behalf and on behalf of the person, entity or other organization on whose behalf you are accessing the instructions, that neither ICONICS nor any of its partners mentioned herewith and service providers, including, without limitation, any system integrator or independent software vendor: (1) makes any representations or warranties of any kind, either express, implied, statutory or otherwise with respect to the instructions, including the accuracy, completeness or usefulness thereof; and (2) shall be liable for damages of any kind, under any legal theory, arising out of or in connection with your election to follow or use, or inability to follow or use, the instructions, including any direct, indirect, incidental, special, punitive or consequential damages, or for loss of use, loss of profits, loss of data, loss of business, or loss of privacy or security, even if foreseeable, arising out of or in connection with your election to follow or use, or inability to follow or use, the instructions. You further acknowledge and agree that your use of the instructions, whether directly or indirectly, is at your own risk and that you expressly assume all risk in connection with your use of the instructions. If you do not agree to the foregoing, you may not access or use the instructions.

Copyright © 2021, ICONICS, Inc. All rights reserved.

Authors

- Zhi Wei Li, Director of Innovation and Incubation Solutions, ICONICS Inc.
- Mark Weinstein, Head of Sales and Delivery, Cognian

1 Introduction

The Internet of Things (IoT) is augmenting our understanding of the world around us by connecting the physical aspects of our environment with the digital platforms that simplify our lives.

To realize the transformative power of IoT, it must be possible to deploy sensors quickly and flexibly in an environment. In support of flexible deployments of IoT sensors, a key infrastructure to have is a communication network that spans the environment, allowing easy integration of IoT sensors that communicate wirelessly.

Cognian Technologies is a fast-growing Australian property technology company whose mission is to make existing buildings smart buildings, and smart buildings smarter by adopting an open, secure, and scalable human-centric approach.

Australian patented Syncromesh is a wireless canopy device that turns every building into a smart building with management of cost, carbon emissions, safety, and well-being at its core. The unified data connectivity platform enables remote and autonomous control over smart building capabilities such as access, occupant capacity, heating/cooling, air quality, and lighting control.

Syncromesh enables open systems that avoid technology vendor lock-in and deliver the flexibility to add the features and services their customers need throughout the life cycle of their buildings. In parallel, it ensures system security and data privacy.

Syncromesh has unlocked the smart building retrofits market while also remaining a highly attractive solution for new builds. The Syncromesh platform lowers installation costs and enables wireless lighting control, location-based services, air quality control, space management and utilisation, environmental monitoring, temp and environmental monitoring, people counting, and mobile device control for new and existing buildings.

The integration of Syncromesh with ICONICS combines the best of IoT connectivity with an enterprise grade monitoring and analysis platform, creating a robust and open IoT solution stack, ready to realize your IoT vision.

2 Architecture

The architecture of this solution is shown in the diagram below:

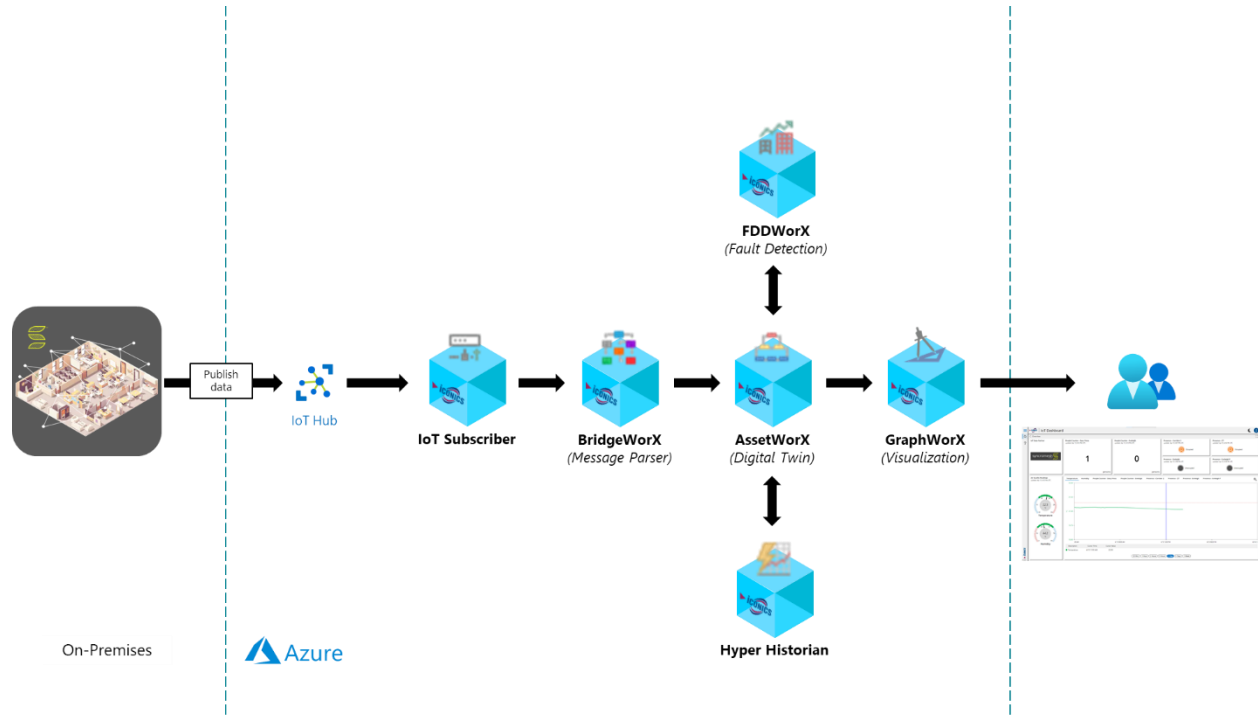


Figure 1 - Syncromesh ICONICS integration architecture

There are two primary Azure components of the architecture:

- Azure IoT Hub
- ICONICS Suite Virtual Machine

All ICONICS capabilities are available in the ICONICS Suite Virtual Machine.

The FDDWorX™ capability shown in the architecture diagram can optionally be added to the solution to provide fault detection based on the ingested IoT data. Learn more about FDDWorX here:

https://docs.iconics.com/V10.97/GENESIS64/Help/ICONICS_Product_Help.htm#Apps/WBDT/FDDWorX/FDDWorX_Overview.htm

3 Azure infrastructure setup

For this solution, we need to first set up the following Azure infrastructure components:

- Azure IoT Hub
- ICONICS Suite Virtual Machine

To provision the Azure IoT Hub and Device Provisioning Service, please follow the instructions in this Microsoft article:

<https://docs.microsoft.com/en-us/azure/iot-dps/quick-setup-auto-provision>

For this solution, the IoT Hub must be at least of the S1 tier.

To provision the latest version of ICONICS Suite Virtual Machine from the Azure Marketplace, please follow the instructions in this ICONICS article:

<https://iconics.com/Documents/Whitepapers/Creating-Azure-Virtual-Machine-Image-Definition>

Proceed to the next section once the necessary Azure components are successfully created.

4 Device setup on Azure IoT Hub

An IoT device must first be created in the Azure IoT Hub to represent the Syncromesh network's gateway. IoT devices on the Syncromesh network will publish data via this gateway device into the IoT hub.

In your IoT hub, follow the steps below to add a new IoT device:

1. Click on the IoT devices link on the side panel
2. Click on + New at the top
3. Enter a desired Device ID
Note this Device ID as it will be needed later.
4. Click Save

Click on the newly added device and get the Primary Connection String to program into the Syncromesh gateway for data publishing.

5 IoT Subscriber Connection

To be able to receive data published by IoT devices via the Syncromesh network into IoT Hub, we must subscribe to the IoT Hub from GENESIS64™.

Create a subscription with the following steps:

1. From the IoT Hub that was set up in section 2, obtain the following:
 - a. Event Hub Connection String
 - Found in the **Built-in endpoints** section of the Hub → **Event Hub-compatible endpoint**
 - b. Hub Owner Connection String
 - Found in the **Shared access policies** section of the Hub → **iothubowner** → **Connection string-primary key**
2. Open ICONICS Workbench
3. In the Workbench project, expand **Internet of Things**
4. Right click on **Subscriber Connections** → **Add Subscriber Connection**
5. Give the subscriber connection a name, e.g., name of the hub
6. Ensure the connection type is **Azure IoT Hub**
7. Change the **Default Decoder** to be **JSON**
8. In the **IoT Hub Settings** section, enter the **Event Hub Connection String** and **Hub Owner Connection String** information from step 1
9. Click **Apply**

Start the **Subscriber** service from the **Home** ribbon

The configuration of IoT Subscriber Connection is now complete.

6 AssetWorX configuration

Before we can utilize the data published by IoT sensors connected to the Syncromesh gateway, we need to create digital twins of the sensors in AssetWorX™, ICONICS' digital twin platform. This provides a place to store the most recent value for each published metric.

6.1 Create the equipment class

We start by creating an equipment class that serves as the template of each IoT device. Follow the steps below to create an equipment class:

1. In the ICONICS Workbench project, expand **Assets**
2. Right click on **Equipment classes** → **Add Folder**
3. Name the folder **Syncromesh** and click **Apply**
4. Right click on the newly created folder and select **Add Equipment class**
5. Name the equipment class as per the device's purpose, e.g., **DoorSensor**, **PresenceSensor**, etc.
6. In the Display Name field, enter the parameter **/?DisplayName?/**
7. Click **Apply**
8. Right click on the equipment class and select **Add Equipment class property**
9. Name this property **is-Syncromesh**
10. Select the **Real Time Data** tab and configure the following fields:
 - a. Set the **Source type** to **Static value**
 - b. Set the **Data type** to **Boolean**
 - c. Set the **Value** to **True**
11. Click **Apply**
12. Repeating steps 8 to 11, create these additional properties in each IoT device class:

Name	Source Type	Data Type	Value
Gateway	Static value	String	/?GatewayID?/
Topic	Static value	String	/?Topic?/
DeviceType	Static value	String	Match the name of the equipment class

6.2 Create telemetry data properties

In each IoT device class, based on the telemetry published, a data property that matches the telemetry must be created. Create the telemetry properties with the following steps:

1. Right click on the respective IoT device equipment class and select **Add Equipment class property**
2. Name the property based on the expected telemetry, **e.g., doorCounter, Temperature, etc.**
3. Select the **Real Time Data** tab and configure the following fields:
 - a. Set the **Source type** to be **Static value**
 - b. Set the **Data type** field to be **Float32**
 - c. Set the **Value** field to be **0**
 - d. Check the **Use database cache** checkbox
4. Click **Apply**
5. To easily have a historical log of the telemetry in Hyper Historian, select the **Historical Data** tab and configure the following fields:
 - a. Set the **Source type** to **Hyper Historian Tag**
 - b. Under **Properties** section, set the **In Group** field to an appropriate collector group
6. Click **Apply**
7. Repeat these steps for every telemetry expected

6.3 Update class parameter definitions

Before the IoT device class can be used to create device instances, we must first update its parameter definitions. Follow the steps below to update the parameter definitions:

1. Right click on the respective IoT device class
2. Select **Update parameter definitions**
3. Uncheck **Simulate operation**
4. Click **OK**
5. Once the update process is complete, click **Close**

In each IoT device class's **Parameters** tab, you should at least see the following 3 parameters:

Parameter Name	Default Value	Parameter Description
+ Click here to add new item		
DisplayName		
GatewayID		ID of the Syncromesh gateway in IoT Hub
Topic		Topic of the published telemetry

Figure 2 - IoT device class parameters

6.4 *Instantiate equipment*

Now that we have an equipment class that describes the IoT device, we can instantiate the IoT device into the AssetWorX equipment hierarchy. Follow the steps below to create equipment based on the IoT device equipment class:

1. Under the **Assets** provider tree, right click on **Equipment** → **Add Equipment**
2. Name the equipment **Syncromesh**
3. Right click on **Syncromesh** → **Add Equipment from library**
4. In the **Equipment Class Library** dialog, browse to and select the desired IoT equipment class in the Syncromesh folder
5. In the **Instantiation Options** dialog:
 - a. For the **Instance Name**, enter the device ID of the IoT device on the Syncromesh network
 - b. For the **DisplayName** parameter, in the **Value** column, enter a user-friendly name for the device
 - c. For the **GatewayID** parameter, in the **Value** column, enter the IoT Device ID of the Syncromesh gateway device as seen in the IoT Hub
 - d. For the **Topic** parameter, in the **Value** column, enter the topic identifier of the IoT device as published by the Syncromesh gateway
 - e. Click on **Instantiate**
 - f. Click **Close** once successful
6. You should now see the equipment created with the entered display name with the equipment properties as defined in the class
7. Later, the values of each equipment property will be updated by the BridgeWorX transaction that parses the published JSON message

The configuration of AssetWorX is now complete.

To bulk create multiple IoT devices based on the respective equipment class, use Bulk Asset Configurator. Learn more about the Bulk Asset Configurator here:

https://docs.iconics.com/V10.96.2/GENESIS64/Help/ICONICS_Product_Help.htm#Com/Tools/BulkAsset/About_the_Bulk_Asset_Configurator.htm


7 AnalytiX-BI configuration

In order to map the metrics published by each IoT device to their respective digital twin in AssetWorX, we need to match the IoT Device ID in the incoming message with the AssetWorX equipment whose name matches the Device ID.

To achieve this, we will use AnalytiX®-BI to extract the list of AssetWorX equipment that are IoT devices. The extracted list will later be used by the message parsing transaction in BridgeWorX™64 to match the IoT Device ID in the published message to the respective AssetWorX equipment. The transaction will then update the equipment property per device that matches the metric parsed from the message.

7.1 Create the Data Flow



We start by creating a data flow to bring in the list of IoT devices that are part of the Syncromesh network. Follow the steps below to configure the necessary data flow:

1. In the ICONICS Workbench project, expand **AnalytiX → BI Server**
2. Right click on **Data Flows → Add Data Flow Folder**
3. Give the folder a name, e.g., Syncromesh
4. Right click on the folder → **Add Data Flow**
5. Give the Data Flow a name, e.g., Devices
6. In the **Data Flow** tab, under **Steps**, click on the top where it says **+ Click here to add new step**
7. From the menu, select **Data Sources → Asset Property Values**
8. In the **Asset Property Values Properties** dialog in the middle, define the configuration fields as follows:
 - Set the **Property-Read batch size** to **500**
 - Set the **Property-Read timeout** to **5** seconds
 - Set the **Property Name Filter** to *****
9. In the **Data Sources** dialog in the middle:
 - Click on **Click here to add new item**
 - Click on the tag browser button, , to the right of the highlighted field
 - Browse to the folder that contains the Syncromesh IoT devices under **My Computer → Assets**
 - Add a ***** at the end so all Syncromesh IoT devices under this folder will be brought in
10. Click on **Apply Changes**
11. In the **Data Flow Preview** table on the right, remove all columns **except** the following:
 - **AssetPropertyName**
 - **AssetPropertyPath**
 - **AssetPropertyValue**
12. From the **Steps** section, add a **Transform Column** step
13. In the **Transform Column Properties** dialog, set the following:
 - Set the **Column Name** to be **AssetPropertyPath**
 - Click on the **Configure the expression** link
 - Enter this expression to extract the Device ID used as the equipment name:
`replace({{AssetPropertyPath}},{{AssetPropertyName}}, "")`
 - Click **OK**
14. Click on **Apply Changes**
15. From the **Steps** section, add a **Transpose** step

16. For the Pivot Column Name field, select **AssetPropertyName**
17. For the Values Column Name field, select **AssetPropertyValue**
18. Click on **Apply Changes**
19. From the **Steps** section, add a **Filter** step
20. Set the filter expression to be this:
x={{Topic}}!=null
21. Click on **Apply Changes**
22. From the **Steps** section, add a **Filter** step
23. Set the filter expression to be this:
x={{is-Syncromesh}}=="True"
24. Click on **Apply Changes**
25. From the Data Flow Preview table, remove the **is-Syncromesh** column
26. From the Data Flow Preview table, click on the column header for **AssetPropertyPath** column and change the column name to **AssetPath**
27. Click **Apply** to save this data flow

7.2 Create the Data Model

The Data Flow above provides a list of the Syncromesh IoT devices that are configured in AssetWorX. To allow the message parsing transaction to query this list, it must first be cached in a Data Model. Create the Data Model with the following steps:

1. In the ICONICS Workbench project, expand **AnalytiX → BI Server**
2. Right click on **Data Models → Add Data Model**
3. Name the **Data Model** as **SyncromeshDevices**
4. Click **Apply**
5. Right click on the new Data Model → **Add Data Table**
6. Name the Data Table as **Devices**
7. In the Data Table configuration dialog, enter the following information:
 - a. For the **Data Source** field, click on the data browser icon, , on the right
 - b. Browse to the Data Flow created before under **My Computer → AnalytiX → BI Server → Data Flows**
 - c. Select the Data Flow
 - d. Click **OK**
 - e. Click away from the **Data Source** field
 - f. The schema of the Data Flow would be brought in and a preview of the data table is shown in the Runtime Preview section at the bottom
 - g. Select the **Refresh** tab
 - h. In the **Refresh Triggers** section, click on the column header that says **+ Click here to add new item**
 - i. Click on the  button
 - j. In the Data Browser, browse under **My Computer → Actions → Triggers → Date/Time Triggers**
 - k. Select the **Every5Minutes** item
 - l. Click **OK**
 - m. For the **Refresh Type**, set it to **Overwrite with new data**
 - n. Click **Apply**
8. Start the **Point Manager** service from the **Home** ribbon.

The configuration of AnalytiX-BI is now complete.

8 BridgeWorX64 configuration

BridgeWorX64 is ICONICS' transactional workflow module. In this solution, a transaction will be configured in BridgeWorX64 to parse the incoming messages from Syncromesh IoT devices.

The messages published by Syncromesh IoT devices will be first received by the IoT Subscriber Connection created in a prior section. The message parsing transaction would then subscribe to the IoT Subscriber Connection to receive the published messages.

The message parsing transaction would be triggered to perform its message parsing duties every time a new message is received in the IoT Hub. Since the transaction will receive all messages from any IoT device of the specified IoT Hub, the logic of the workflow is designed as such:

1. Get the count of devices that match the incoming message's Device ID from the AnalytiX-BI Data Model
2. If the count is more than 0, that means it is a valid Syncromesh IoT message and is configured in AssetWorX. Message should be parsed. Otherwise, exit the transaction
3. Parse the JSON message for all of the specified metrics
4. Write each parsed metric to its respective equipment property of the matching AssetWorX equipment

8.1 Create the Transaction

Create the message parsing transaction with the following steps:

1. In the ICONICS Workbench project, expand **Bridging → Transactions**
2. If there is no existing configuration under Transactions, create a new configuration with the following steps:
 - a. Right click on **Transactions → Add Configuration**
 - b. Give the configuration a name
 - c. Check **Active Configuration**
 - d. For the **Temporary Directory** field, provide a file location
 - e. Click **Apply**
3. Right click on the active configuration → **Add Transaction**
4. Give the transaction a name
5. Check off **The transaction is enabled**
6. In the **Transaction Template** section's **Selected Template** field, click on the + button to the right to create a new transaction template
7. Give the transaction template a name in the **Template Name** field
8. Click **Apply**

- The transaction settings should look like this:


Figure 3 - Transaction settings

- In the **Transaction Template** section, click on the edit button  to edit the template.

8.2 Configure the Transaction Template

The Transaction Template is where the logic of the transaction workflow is defined.

Follow the steps below to define the logic necessary to parse the incoming messages for Syncromesh IoT devices:

- From the **Activities Library** on the left, drag in the **Real Time Input** activity
- Select the new activity and edit its properties as follows:
 - Name the activity **Get device count**
 - In the **Data Sources** section, click on the **Click here to add new item** header
 - Click on the  button on the right that brings up the expression editor
 - Enter the following expression:


```
"bi:Models:SyncromeshDevices(Select count(Devices.Gateway) as DeviceCount WHERE Devices.Gateway = '"+{{triggerVariable:Sender}}+"') [DeviceCount][0]"
```
 - Click **OK**
- From the **Activities Library** on the left, drag in a **Condition** activity
- Select the new activity and edit its properties as follows:
 - Name the activity **Should process**
 - In the **Condition Settings** section, click on the **Edit Expression** link
 - In the expression editor, paste in this expression:


```
{{PrevActivityOutput:Row[0].Col[Value]}}>0
```
 - Click **OK**
- From the **Activities Library** on the left, drag in a **JSON Content Reader** activity
- Select the new activity and edit its properties as follows:
 - Name the activity **Decode_Syncromesh**
 - In the **JSON Reader Settings** section, set the following settings:
 - Set the **JSON Source Type** to be **Expression**
 - Click on the **Edit Expression** link
 - In the expression editor, paste in this expression:


```
{{triggerVariable:Message}}
```
 - Click **OK**

- v. Set the **JSON Parse Mode** to be **Match Predefined Columns By Names**
- c. In the **Data Schema** section, click on the **Configure** link
- d. Define the columns list as follows:

Configure the columns						
		#	Name	Display Name	Data Type	Output
+ Click here to add new item						
	▲ ▼	0	topic		String	<input checked="" type="checkbox"/>
	▲ ▼	1	telemetry.temperatureC		String	<input checked="" type="checkbox"/>
	▲ ▼	2	telemetry.pressure		String	<input checked="" type="checkbox"/>
	▲ ▼	3	telemetry.humidity		String	<input checked="" type="checkbox"/>
	▲ ▼	4	telemetry.doorCounter		String	<input checked="" type="checkbox"/>
	▲ ▼	5	telemetry.presence		String	<input checked="" type="checkbox"/>

Figure 4 - IoT data mapping

- e. Add any additional telemetry identifiers as necessary with the same syntax of prefixing it with **"telemetry."**
 - f. Click OK
7. From the **Activities Library** on the left, drag in a **Real Time Input** activity
 8. Select the new activity and edit its properties as follows:
 - a. Name the activity **Get AssetPath**
 - b. In the **Data Sources** section, click on the **Click here to add new item** header
 - c. Click on the **Σ** button on the right that brings up the expression editor
 - d. Enter the following expression:


```
"bi:Models:SyncromeshDevices(Select Top 1 Devices.AssetPath Where Devices.Topic = '"+{{PrevActivityOutput:Row[0].Col[topic]}}+"') [AssetPath] [0]"
```
 - e. Click **OK**
 9. From the **Activities Library** on the left, drag in a **Dynamic Tag Writer** activity
 10. Select the new activity and edit its properties as follows:
 - a. Name the activity **Update_Device_Data**
 - b. In the **Destination Dynamic Tags** section, add the list of tags as follows:

Point Name Expression	Output Expression
x="ac:"+"{{PrevActivityOutput:Row[0].Col[Value]}}+"TemperatureC"	x={{ActivityOutput:Decode_Syncromesh.Row[0].Col[telemetry.temperatureC]}}
x="ac:"+"{{PrevActivityOutput:Row[0].Col[Value]}}+"Humidity"	x={{ActivityOutput:Decode_Syncromesh.Row[0].Col[telemetry.humidity]}}
x="ac:"+"{{PrevActivityOutput:Row[0].Col[Value]}}+"Pressure"	x={{ActivityOutput:Decode_Syncromesh.Row[0].Col[telemetry.pressure]}}
x="ac:"+"{{PrevActivityOutput:Row[0].Col[Value]}}+"doorCounter"	x={{ActivityOutput:Decode_Syncromesh.Row[0].Col[telemetry.doorCounter]}}
x="ac:"+"{{PrevActivityOutput:Row[0].Col[Value]}}+"Presence"	x={{ActivityOutput:Decode_Syncromesh.Row[0].Col[telemetry.presence]}}

11. Now that all necessary activity blocks are configured, link the activities into a workflow as shown below:

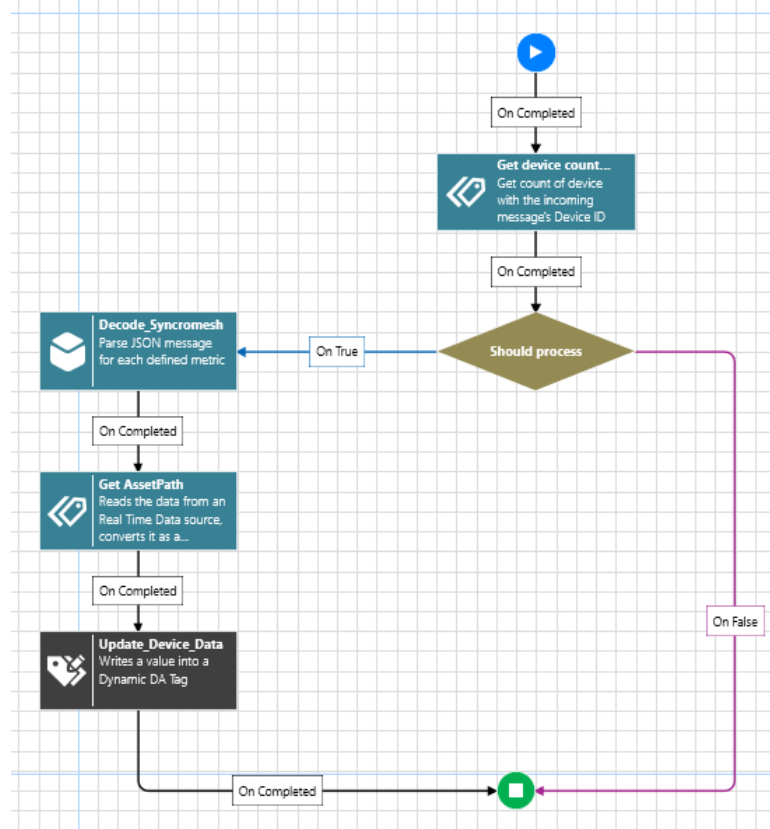



Figure 5 - IoT parsing transaction diagram

12. Click **Apply** when complete

8.3 Trigger transaction with IoT Hub messages

To trigger the transaction based on messages published to IoT Hub, it needs to be configured to subscribe to the IoT Hub via the IoT Subscriber Connection defined before. Follow the steps below to connect the transaction to the IoT Hub:

1. In the ICONICS Workbench project, under **Bridging → Transactions**
2. Expand the active configuration
3. Right click on the message parsing transaction item created before and select **Edit**
4. In the transaction configuration dialog, click on the **Alarm Subscriptions** tab
5. Check off **Trigger transaction using alarm source**
6. In the **Alarm Subscription** section, define the following:
 - a. Click on the **Click here to add new item header**
 - b. Click on the browse button, 
 - c. In the Data Browser, browse under **My Computer → Internet of Things**
 - d. Select the subscriber connection configured before
 - e. Click **OK**
 - f. Click on the **Click to configure fields and mapping** link at the **Alarm Subscription** header
 - g. In the popup dialog, select **RawMessageType** on the left

- h. Check off the first row, which should have **Sender** in its **Field Name**, like so:

Event/Alarm Type	Field Name	Field Type
<input checked="" type="checkbox"/> RawMessageType	Sender	String

Figure 6 - IoT Event Server Subscription Fields

- i. Click **OK**
- j. Click **Next**
- k. Click **Finish**
- l. Click **Apply**

Start the **Point Manager and Scheduler** service from the **Home** ribbon.

The configuration of BridgeWorX64 is now complete.

When the Syncromesh gateway publishes its IoT data to IoT Hub, it will be received by BridgeWorX64 and trigger the transaction to parse the JSON message and update the appropriate equipment property in AssetWorX for the respective device.

Since we also configured the equipment properties to log data into Hyper Historian, we will also have the historical trends of all published metrics.

9 Visualize the IoT device data

With the IoT device data now available as AssetWorX equipment properties, we can easily create an IoT data dashboard utilizing GraphWorX™64 and its collection of templates, controls, and symbols. The examples below illustrate what a dashboard that visualizes the current value and historical trend of each IoT data could look like:

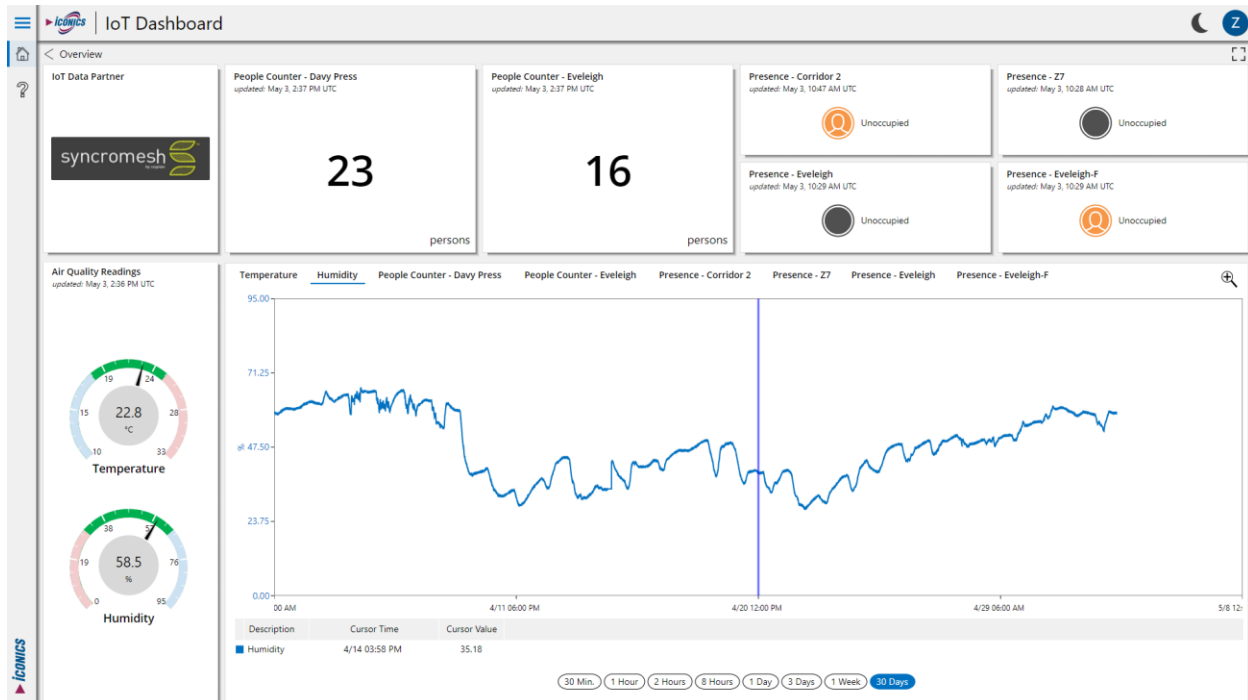


Figure 7 - Sample IoT Dashboard

10 Conclusion

An IoT mesh network like Cognian Syncromesh is a fundamental infrastructure component that should be part of any IoT strategy to streamline the deployment of IoT devices throughout a portfolio. Having such a network also standardizes the way IoT data is published and managed.

Leverage ICONICS GENESIS64, Cognian Syncromesh, and Azure IoT technologies to easily deploy and monitor an IoT mesh network solution and gain a better understanding of your environment.



Founded in 1986, ICONICS is an award-winning independent software provider offering real-time visualization, HMI/SCADA, energy management, fault detection, manufacturing intelligence, MES, and a suite of analytics solutions for operational excellence. ICONICS solutions are installed in 70 percent of the Global 500 companies around the world, helping customers to be more profitable, agile and efficient, to improve quality, and to be more sustainable.

ICONICS is leading the way in cloud-based solutions with its HMI/SCADA, analytics, mobile and data historian to help its customers embrace the Internet of Things (IoT). ICONICS products are used in manufacturing, building automation, oil and gas, renewable energy, utilities, water and wastewater, pharmaceuticals, automotive, and many other industries. ICONICS' advanced visualization, productivity, and sustainability solutions are built on its flagship products: GENESIS64™ HMI/SCADA, Hyper Historian™ plant historian, AnalytiX® solution suite, and MobileHMI™ mobile apps. Delivering information anytime, anywhere, ICONICS' solutions scale from the smallest standalone embedded projects to the largest enterprise applications.

ICONICS promotes an international culture of innovation, creativity, and excellence in product design, development, technical support, training, sales, and consulting services for end users, systems integrators, OEMs, and channel partners. ICONICS has over 375,000 applications installed in multiple industries worldwide.

ICONICS Sales Offices



World Headquarters

100 Foxborough Blvd.
Foxborough, MA, USA, 02035

☎ +1 508 543 8600
✉ us@iconics.com



European Headquarters

Netherlands

☎ +31 252 228 588
✉ holland@iconics.com

Australia

☎ +61 2 9605 1333
✉ australia@iconics.com

Canada

☎ +1 647 544 1150
✉ canada@iconics.com

China

☎ +86 10 8494 2570
✉ china@iconics.com

Czech Republic

☎ +420 377 183 420
✉ czech@iconics.com

France

☎ +33 4 50 19 11 80
✉ france@iconics.com

Germany

☎ +49 2241 16 508 0
✉ germany@iconics.com

India

☎ +91 265 6700821
✉ india@iconics.com

Italy

☎ +39 010 46 0626
✉ italy@iconics.com

Middle East

☎ +966 540 881 264
✉ middleeast@iconics.co

Singapore

☎ +65 6667 8295
✉ singapore@iconics.com

UK

☎ +44 1384 246 700
✉ uk@iconics.com



For more, visit iconics.com

© 2021 ICONICS, Inc. All rights reserved. Specifications are subject to change without notice. AnalytiX and its respective modules are registered trademarks of ICONICS, Inc. GENESIS64, GENESIS32, Hyper Historian, IoTWorX, KPIWorX, MobileHMI, WebHMI and their respective modules, OPC-to-the-Core, Make the Invisible Visible, and ICONICS company logo, are trademarks of ICONICS, Inc. Other product and company names mentioned herein may be trademarks of their respective owners.

Gold

Microsoft Partner

Six-time Partner of the Year Winner

