



IoTWorX Quick Start



APPLICATION NOTE

June 2017

Description: This document describes IoTWorX™ and how to configure it.

OS Requirements:

- *IoT Gateway:* Windows 10 IoT or Windows 10 Enterprise
- *Remote PC:* Vista x64, Windows 7 x64, Windows 8 x64, Windows 10 x64, Windows Server 2008 x64, Windows Server 2008 R2 x64, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016

General Requirement: MS Azure Portal account + Basic SQL Azure/IoT knowledge.

Introduction

IoTWorX bridges the gap between an on-premises communications networks and an Azure cloud-based deployment platform. The IoT gateway, which can be considered an “Edge Device”, provides data connectivity between on-premises end devices and the cloud. Targeting process, factory, and building automation as well as other industries, IoTWorX installs with on-premises communications includes BACnet, SNMP, Modbus, and OPC. The cloud communication path is highly secure and is based on Advanced Message Queuing Protocol (AMQP) and a publish/subscribe (“pub/sub”) mechanism.

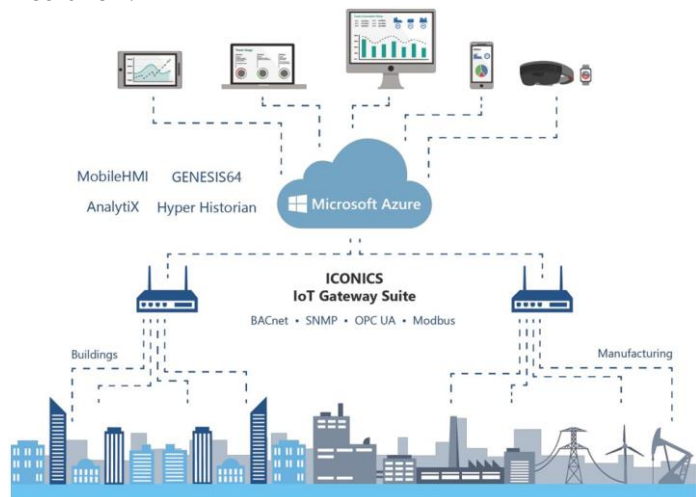


Figure 1 - IoTWorX Architecture

Benefits

IoTWorX provides manufacturers and facility managers with a flexible platform to create Internet of Things applications. ICONICS offers several key IoT technologies, including rich connectivity to things (OPC, OPC UA, BACnet, SNMP, Modbus), secure cloud communications, and built-in real-time

visualization and analytics. With simple setup and configuration, users can easily create remote monitoring and analytics solutions that meet their innovative business requirements for collaboration and shared insight across widely dispersed assets. The ICONICS IoT solution leverages the Azure cloud to provide global visibility, scalability, and reliability. Connect to standard ICONICS applications in the cloud such as GENESIS64™, Hyper Historian™, AnalytiX®, and more. Optionally integrate Microsoft Azure services such as Power BI and Machine Learning to provide greater depth of analyses.

Prerequisites and Use

Initial Resource Setup

1. Create an IoT Hub in Azure Portal. Copy the **iothubowner** “Connection String—primary key”. This will be used later.

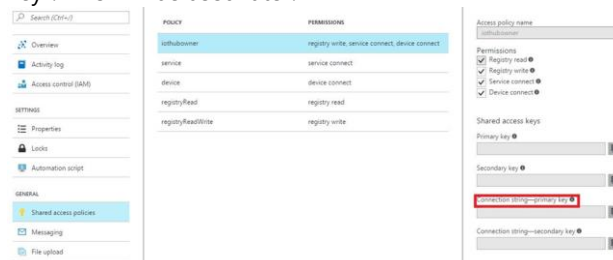


Figure 2 - IoT Hub in Azure Portal

2. Create a SQL server and SQL database in Azure Portal. Copy the connection string (make sure to fill in username and password) and keep available. The string can be found by clicking on the SQL database, and then clicking **Show database connection strings** in the **Essentials** section of the **Overview** screen. Copy the “ADO.NET” string.

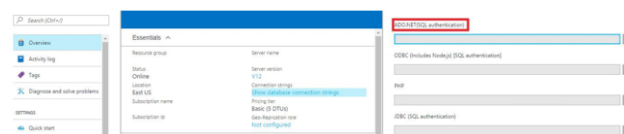


Figure 3 - SQL Azure in Azure Portal

3. Configure your SQL Azure firewall to allow client IP “69.26.123.193”, as well as the IP addresses of your IoT gateway device and Workbench server.
4. Install **IoTWorX** on the gateway device.
5. Install ICONICS Suite on a remote PC. This PC does not need to be on the same network as the gateway device, though ICONICS does recommend installing on an Azure VM within the same Azure resource group as the IoT Hub.



IoTWorX Quick Start

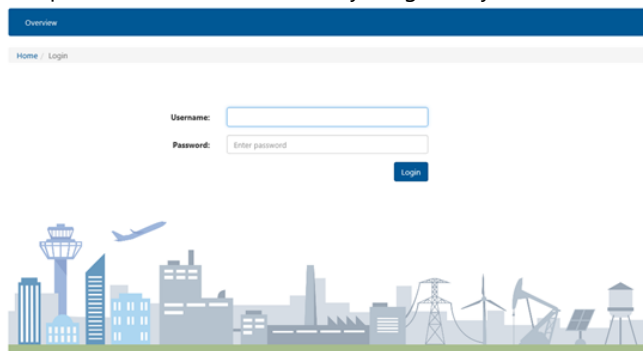


APPLICATION NOTE

June 2017

Device Provisioning

1. On either the gateway or a remote PC on the same network as the gateway, open a web browser and go to <http://<gateway>:1447/>, replacing "<gateway>" with the computer name or IP address of your gateway device.



© copyright ICONICS 2017

Figure 4 - The IoTWorX Web Interface

2. Login with credentials (user: *admin*, password: *iconics*).
3. Upon logging in for the first time, make sure to change your password to a secure password
4. Click on the **Provisioning** link at the top of the page.
5. Select the option "**I want to use my private IoT Hub**" and click **Next**.

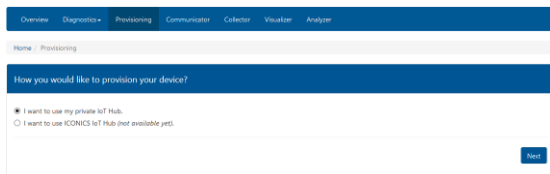


Figure 5 - Selecting IoT Hub During Provisioning

6. Select the option "**I want to register this IoT Gateway as a new device in my IoT Hub**" and click **Next**.

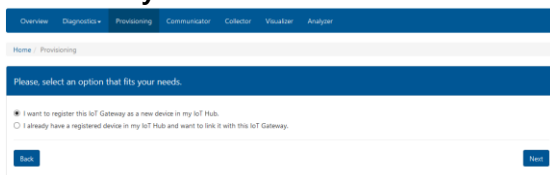


Figure 6 - Registering IoT Gateway as a New Device

7. Copy over the IoT Hub Connection String from the **Azure Portal**.
8. Name the gateway device in the Azure IoT Hub by providing a Device ID (an alphanumeric name without spaces).
9. If the device has been previously provisioned, the device requires a unique name (a name not previously used).
10. Copy over the Azure SQL connection string from the **Azure Portal**.

11. Click **Test SQL** button to test SQL connection. "**SQL database connection test succeeded**" appears, indicating that you are ready to provision your device.

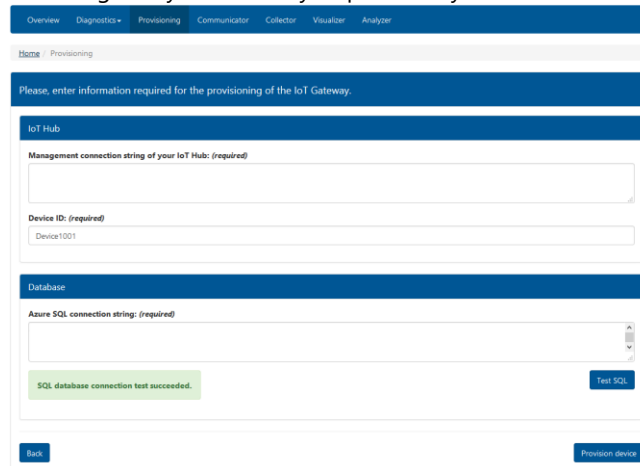


Figure 7 - Provisioning interface

12. Click **Provision Device**. This step may take several minutes to complete. Do not refresh the page during this time.

NOTE: If you are using B0 or S1 tier Azure SQL database, the provisioning may take longer. Please wait up to 15 minutes while the necessary database tables are created and the device finishes provisioning.

13. Once the device is provisioned, the message "**IoT Gateway device is provisioned. See the provisioning info below**" appears. Click the **Overview** tab to confirm that your device provisioned correctly.

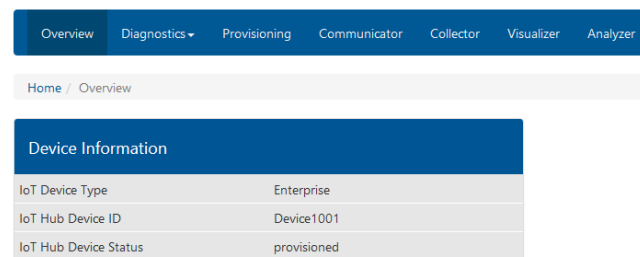


Figure 8 - IoT Gateway Device Successfully Provisioned

14. Once provisioning is complete, restart the device by clicking on the **Diagnostics** dropdown menu at the top of the interface. Select **Maintenance** and click **Restart IoT Gateway** (Diagnostics > Maintenance > Restart IoT Gateway).

IoT Publish List Configuration

1. On the remote PC with the ICONICS Suite installed, open Workbench and select the **Project** tab in the top ribbon.



IoTWorX Quick Start



APPLICATION NOTE

June 2017

- Create a new IoT project by clicking **New Project** and selecting the **IoT** project from the dropdown.
- Select **Create a New Connection** and fill out the required fields. Use the same IoT Hub Connection String from the provisioning process. Provide a project name and description and click **OK**.
- Under the new IoT project added to the Project Explorer, you see the newly provisioned device (alongside any other devices connected to the Azure IoT Hub). If the device icon is red, or if the device does not appear, the device may not have finished its restart. Right-click on the project name and select **Update IoT Devices** to refresh the list.
- Once the gateway is up and running, the device has a blue icon, indicating it is online. Within "Subscriber Connections" under "MyProject" (not the IoT Project), a subscription to the IoT Hub has been added.
- Expand **Internet of Things** under your device in your IoT project. Right-click on **Publish Lists** and select **Add Publish List (Real Time)**. Name the publish list.
- In the **Published Points** tab, add points to be exposed over the cloud by adding items to the list and using the data browser to navigate to the appropriate points. Give each point a short "Publish Name", as this reduces messages, minimizing Azure costs. Click **Apply** to save changes.

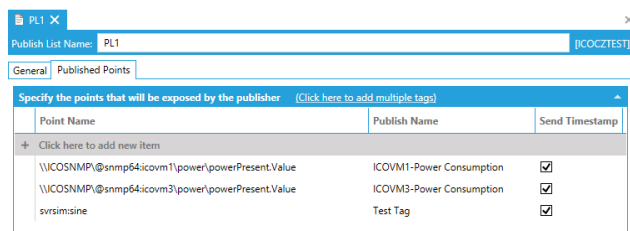


Figure 9 - Published Points

- Open the "Publisher Connection" form. Ensure the connection is enabled by checking the **Enable this connection** box, and, under "Real Time Publish List", select the Publish List you created in the previous step. Apply your changes.

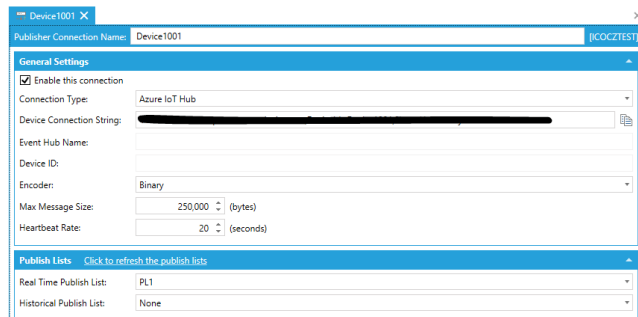


Figure 10 - Publisher Connection Form

- Back on your remote PC, open GraphWorX64. In the Data Browser on the left, expand **My Computer** and select **Internet of Things**. The IoT Subscription you created appears. If you expand this further, your device TestTag becomes visible.

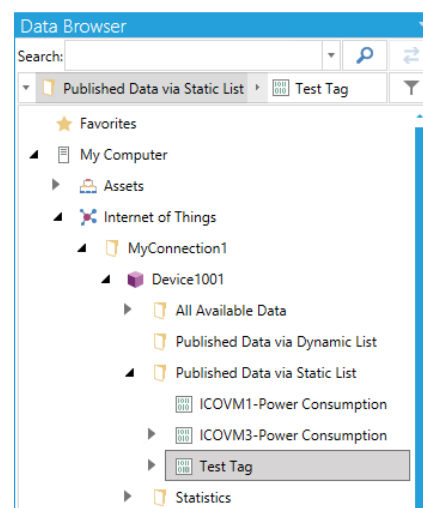


Figure 11 - Browsing IoT Subscription

- Open the Symbol Library in the left-hand pane by clicking **Symbols** and search for "KPI Gauge". Drag KPI Gauge 1 into the canvas and set the gauge's DataSource to your IoT tag using the Data Browser. Enter Runtime to activate your tag.
- You can also see the "All Available Data" folder within the Data Browser. This folder exposes the entire address space and is referred to as the "Dynamic Publish List". If you go back to the Publish List we created in Workbench, you can use tags from this dynamic publish list to add to the list we defined earlier (known as the "Static Publish List").

IoT Collector

IoTWorX can serve as a remote collector for a remote Hyper Historian server.

- Switch to your Remote PC and create a new Publish List (Historical) and add tags similar to the ones you created in real-time publish list.

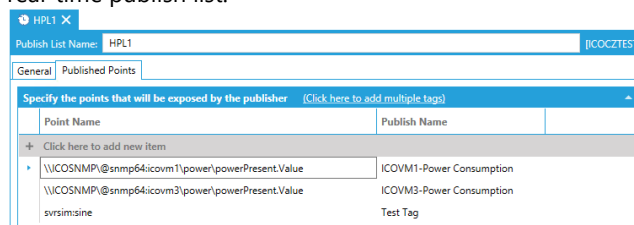


Figure 12 - Historical Publish List



IoTWorkX Quick Start



APPLICATION NOTE

June 2017

2. Create a new Publisher Connection.
 - a. Change the Connection Type to "Platform Services".
 - b. Specify the name of the machine where Hyper Historian is running (Publish Machine Name).

NOTE: In case Hyper Historian Logger is running on a Cloud virtual machine you need to specify the fully qualified domain name or public IP of that VM).

- c. Select the **Historical Publish List** created in the previous step.

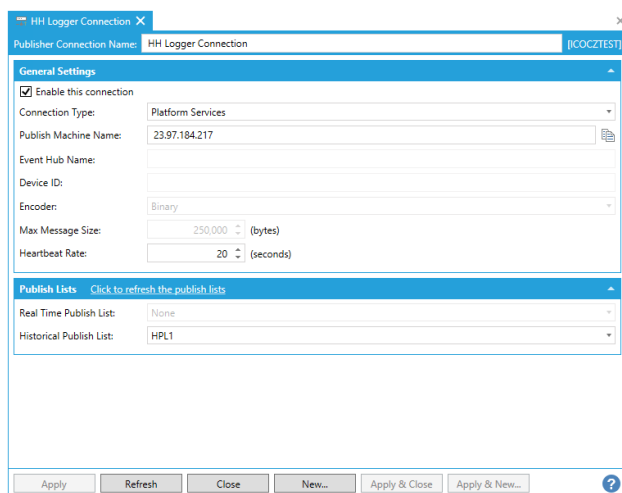


Figure 13 - Publisher Connection for Hyper Historian Logger

3. On your Hyper Historian server, go to the Hyper Historian configuration in Workbench.

- a. Expand **Data Collections** and create your configuration in root (under folders). Tag name(s) needs to be equal to Historical Public List tag name(s). The tag names are case sensitive.
 - b. Create a new tag. Uncheck the **Is Collected** checkbox and click **Apply**.

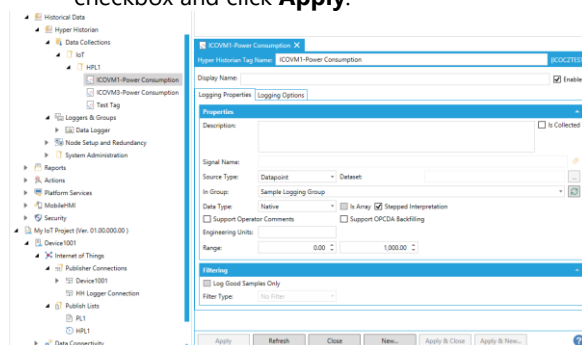


Figure 14 - Hyper Historian Logger Configuration using IoT Collector

4. Confirm inbound traffic on **Port 8778** on the firewall is allowed.
5. Start Hyper Historian service.
6. Restart the IoT Publisher on your gateway.
7. Historical data is now logged from the gateway. To confirm this, create a TrendWorX64 Viewer in GraphWorX64 connected to the tag created in step 3.

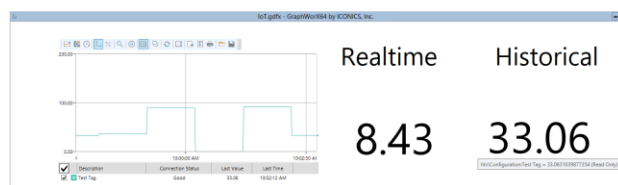


Figure 15 - Historical Data from IoTWorkX in GraphWorX64