1. Welcome Root . . . 2
1.1 Welcome . . . 3
1.1.1 Introducing Ignition . . . 6
1.1.2 Quick Start Guide . . . 12
1.1.2.1 Quick Start - Perspective Session . . . 42
1.1.2.2 Quick Start - Vision Client . . . 59
1.1.3 Installing and Upgrading Ignition . . . 80
1.1.3.1 Command Line Installations . . . 87
1.1.3.2 ZIP File Installations . . . 89
1.1.3.3 Installing or Upgrading a Module . . . 90
1.1.3.4 Ignition 8 Upgrade Guide . . . 95
1.1.4 Vision and Perspective — Which Visualization System Is Best for Me? . . . 103
1.1.5 Support, Feedback, and Resources . . . 106
1.1.6 A Vision-Oriented Guide to Perspective . . . 108
1.1.7 Licensing and Activation . . . 112
1.1.7.1 Emergency Activation . . . 121
1.1.8 Maker Edition . . . 123
1.1.9 Ignition Edge . . . 125
1.1.9.1 Edge Gateway . . . 129
1.1.9.2 Edge Panel . . . 135
1.1.9.3 Edge IIoT . . . 139
1.1.9.4 Edge Compute . . . 144
1.1.9.5 Edge Sync Services . . . 146
1.1.9.6 Edge EAM . . . 151
2. Modules Overview . . . 152
2.1 Core Modules . . . 154
2.1.1 Vision Module . . . 156
2.1.2 Perspective Module . . . 160
2.1.3 OPC UA Module . . . 162
2.1.3.1 Driver Modules . . . 164
2.1.4 Tag Historian Module . . . 166
2.1.5 SQL Bridge Module . . . 168
2.1.6 Reporting Module . . . 171
2.1.7 Alarm Notification Module . . . 174
2.1.8 SMS Notification Module . . . 177
2.1.9 WebDev Module . . . 179
2.1.10 Enterprise Administration Module . . . 180
2.1.11 SFC Module . . . 182
2.2 Additional Modules . . . 184
2.2.1 Voice Notification Module . . . 186
2.2.2 OPC COM Module . . . 188
2.2.3 OPC COM Tunneller Module . . . 189
2.2.4 Web Browser Module . . . 190
2.2.5 Twilio Alarm Notification . . . 191
2.2.6 Serial Modules . . . 192
2.2.7 SECS/GEM Module . . . 193
3. System Architectures . . . 194
3.1 Standard Architecture . . . 200
3.2 Scale Out Architecture . . . 202
3.3 Hub and Spoke Architecture . . . 205
3.4 Edge Architectures . . . 207
3.5 IIoT Architecture . . . 212
3.6 Enterprise Architecture . . . 214
3.7 Redundancy Architecture . . . 215
3.8 Cloud Based Architecture . . . 220

Welcome Root

Welcome

Welcome to Ignition by Inductive Automation

Ignition is not like any other HMI / SCADA system you've seen before! Ignition is a single install, runs from a single location, is server based, and is sold by the server not by the client. It uses current web-based technology and allows you to bring the IT department and the plant floor closer than ever. Ignition uses Java for fast secure runtime clients, and that means it can be used on any system, any version. Windows 10, Mac OSX, Linux or Cloud servers. You can switch between systems or use one type for the server and a combination of types for your runtime clients. If an IT department wants to use Windows servers to host and the Operations groups want to use their new MacBooks, Ignition can handle it.

What can Ignition do for you? It has built-in, drag and drop functionality for just about anything you can imagine. Ignition does HMI/SCADA controls, Dashboards, Historical Trending, Database access, Reporting, Alarming, Security, Sequential Function Charts, Redundancy, Failover control, Enterprise Administration, and more. Ignition does all this well because of its **Modular Architecture**. You choose the features and functionality you need. Want to try some of the other features? You always have a full version of Ignition with a resettable timer to try new things. Play around as much as you like and decide for yourself what you want.

About this User Manual

This Welcome section provides a broad overview and information relating to modules, architectures, installation, and a Quick Start Guide. Th e sidebar on the left is a table of contents organized so you can navigate easily and quickly through Ignition's features, modules, functions, and so forth.

In the Appendix you'll find an exhaustive list of all the Components, Expressions, and Scripting Functions in Ignition.

Take a look at the navigation section on the left and poke around, or use the search bar at the top of the navigation.

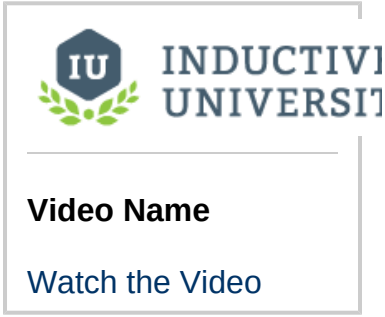
Ignition 8.0

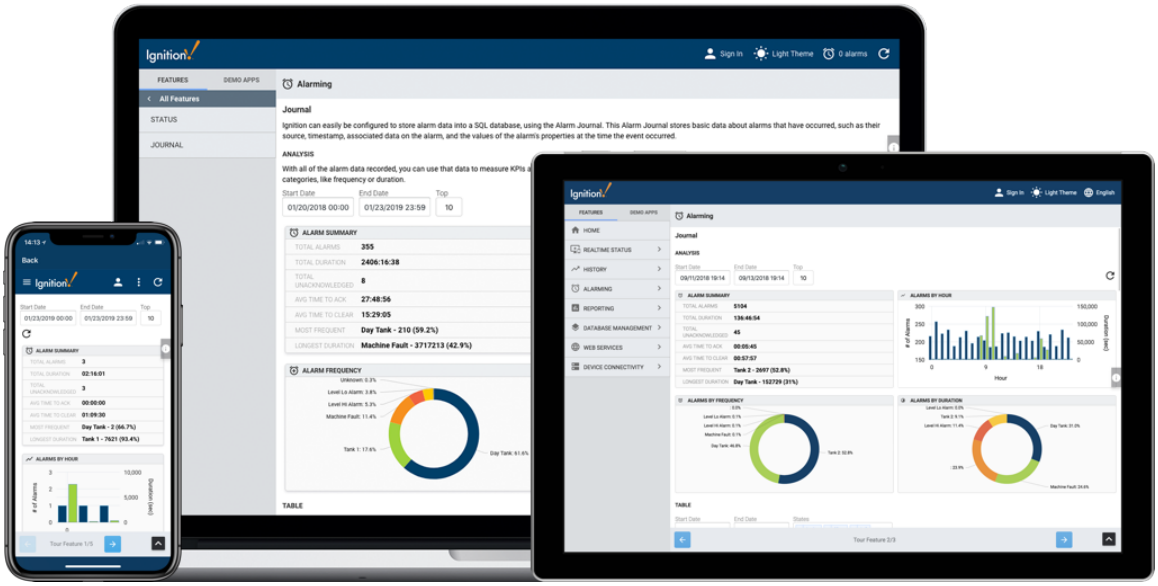
This User Manual is specifically for Ignition 8.0. Newer major versions of the software will have separate User Manuals that focus on newer features. You can also tell which User Manual version you're looking at by the Image in the Sidebar.

Inductive University

Inductive University is a free online learning platform designed to help you master Ignition. You can watch over 500 training videos and participate in our credential program to test your knowledge. It's the first step to joining our ranks of Certified Integrators.

This User Manual section is closely tied with Inductive University so you can have a conceptual understanding from this manual and a visual walk-through of the feature in the videos. Throughout this online manual, you will find links to the various Inductive University videos on the right side of the page. These videos correspond to the information in that page or section. This way, you can get a conceptual understanding from the information provided here, and a visual walk-through of the feature by watching the videos. The video links will look like the one to the right of this paragraph.





Ignition Basics

Ignition is completely different from what you are used to using. See what Ignition is and some of the many things it can do for you. Learn the basics and how it can work for you.

Introducing Ignition

Quick Start Guide

New to Ignition? Get started by getting Ignition running on your computer in minutes with our Quick Start Guide. Download Ignition and evaluate the full version right away.

Quick Start Guide

Installing and Upgrading

You can install Ignition in under 3 minutes; staying current is just as easy. No hassles, no secret handshakes, only one lightweight install and you can have clients running everywhere.

Installing and Upgrading Ignition

System Architectures

Ignition can support many different types of architectures from single standalone systems to vast enterprise level structures. Learn about the types of architectures and how they work.

System Architectures

Modules

Are you ready to **really** get to know Ignition? Modules are the building blocks of Ignition. They are the reason it can do so much and be so flexible. Adding the functionality you need is simple with Ignition Modules.

Modules Overview

New in this Version

Are you looking for something that has changed? Or maybe you want to see the things we have been adding to Ignition? Ch eck out the New in this Version page for a list of improvements for each version.

New in this Version

Partner Programs

Inductive Automation is proud to have a Partner Program that allows others to create modules to enhance Ignition's functionality. You can learn more about Sep asoft's MES modules and Cirrus Link's MQTT modules here.

SDK Documentation

Documentation for the Ignition Module SDK. Anyone looking to create a new module should start here.

Ignition™, Ignition Edge™, and Inductive Automation® are trademarks of Inductive Automation.

All other product or company names that may be mentioned in this publication are tradenames, trademarks, or registered trademarks of their respective owners.

Ignition User Manual © 2015-2020 Inductive Automation
All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.
All Inductive Automation products are trademarks or registered trademarks of Inductive Automation. Other brand and product names are trademarks or registered trademarks of their respective holders.
Every effort has been made to make this user manual as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. Inductive Automation shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in his manual.

Introducing Ignition

What Is Ignition?

Ignition SCADA by Inductive Automation® combines an unlimited licensing model, with instant web-based deployment, and the industry-leading toolset for supervisory control and data acquisition (SCADA) — all on one open and scalable universal platform. Ignition is The New SCADA because it solves all the major pain points of old SCADA. Ignition empowers your business to easily control your processes, and track, display, and analyze all your data, without limits. Ignition is designed from the ground up to be approachable and easy to get started with while it is flexible and capable of scaling up to the largest projects. Ignition is unlimited, so for the low cost of one server license you can seamlessly connect all your data, design any kind of industrial application with ease, and instantly web-deploy unlimited clients to anyone, anywhere — all from one universal platform for industrial automation.

Ignition has a core focused on everything you need and nothing that you don't.

	Server Based One computer controls everything in Ignition. That means you only install in one place and connect everything else to that server. No separate install for the Designer. You can have as many Clients (runtimes) and Designers as you want, no installation necessary.
	Simple Licensing One license for the server gives you unlimited access. Unlimited Clients, unlimited Tags, and unlimited Projects. With unlimited run-time clients at no additional cost, you can get your important data and analytics to your entire team, across your whole company. Ignition allows you to create and use as many tags as you need for devices, OPC servers, and anything else, without limits.
	Web-based Deployment Ignition is installed, deployed, and managed using standard web technologies. Connect and change settings, update projects, and create new Tags, all from any computer on the network.
	Web-launched Designer and Clients Ignition's Designer tool and runtime clients and sessions are launched using the web. Run a large amount of clients across your enterprise at the same time. In initial benchmark testing, Ignition Perspective can simultaneously launch 200 clients with ease.
	Built-in Security Security is built in to every aspect of Ignition. You can manage users, edit screen permissions down to the individual Tag or component level, and even secure certain assets in your projects.
	Scalable Modular Architecture Create a scalable and centrally managed SCADA system using Ignition's web-based server architecture. Do more with unlimited connections to PLCs, databases, web-based deployment, and web-launched clients. Learn more about Ignition's System Architectures. Projects can be made small and expanded out indefinitely as your business grows, or you can have smaller installations connected to and interacting with other larger systems.
	Cross Platform Ignition works with any major operating system, even iOS and Android. Ignition is built on Java, which means you can install and run it on any modern operating system. Ignition will run on Windows, Mac OS X, and Linux. Backups of your system are even cross platform, you can load a backup from any system.

On this page

...

- What Is Ignition?
- New in This Release
- What Can it Do?
  - Web-Based HMI and SCADA Applications Deployment
  - High Performance Historian
  - Reporting Engine
  - Alarming
- Communications Hub
  - Mix and Match
- Understanding Ignition's Flow

New in This Release

- Ignition 8 features a new poll-free protocol that updates tag data instantly for both Vision and Perspective clients. As a result, Vision clients in Ignition 8 are drastically faster while using dramatically less CPU power.
- For the new Vision Module, we've created new native Vision client launchers for Windows, Mac, and Linux, so you can easily launch Vision clients on any major operating system without needing to install and update Java.
- Ignition Vision is perfect for dedicated plant-floor displays and HMIs, while Ignition Perspective is perfect for mobile-first industrial applications. You can use them on their own or together to form the industry's most powerful, cutting-edge visualization system.
- Vision client screens will look even better in Ignition 8, thanks to a new design and new support for high-resolution displays.
- Fonts have also been improved with more choices and better rendering so your screens look sharper and more stylish than ever.
- Ignition 8 has an improved concurrent editing system which allows multiple people to work on the same project at the same time without locking each other out of resources. This will help your team work faster and more collaboratively.
- In Ignition Perspective, you can use powerful and flexible CSS3 styles to change the appearance and position of anything in your application. By combining styles into themes, you'll be able to apply and edit styles across multiple applications in an instant.

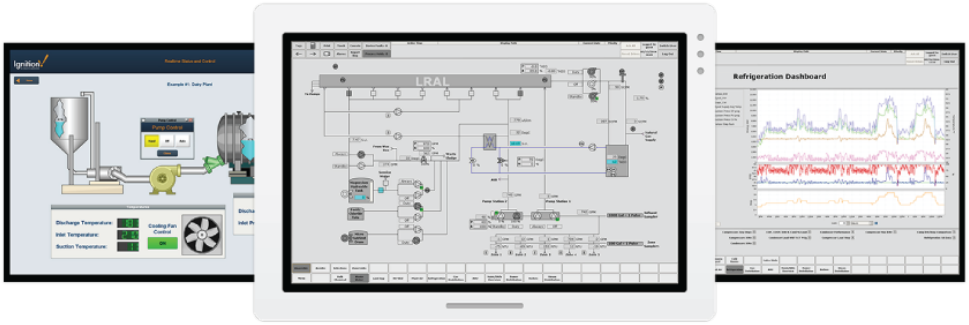
What Can it Do?

Because of its modular nature, Ignition can do a lot or a little and be completely customized to your needs. A lot of the basic functionality is shared between modules and stored in the Gateway to make everything as smooth and seamless as possible. Some of these basics are database connections, device connections, Tags, Security profiles, and Alarming. Because these are all shared resources, you don't have to worry about managing multiple connections for the different modules in Ignition: it just works! You can build database applications using Igniti on. You can create POS, CRM, and inventory tracking systems. You can bring in data from any SQL database in your plant and interact with other existing ERP and Access systems.

Modules provide the core functionality for Ignition where the Gateway provides the backbone of shared resources. There are no limitations to Ignition, here are **just a few** of the things Ignition can do for you:

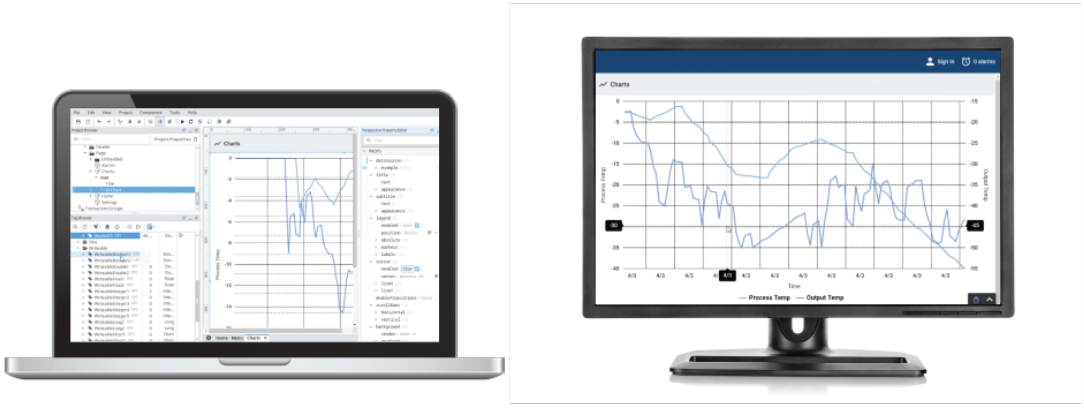
Web-Based HMI and SCADA Applications Deployment

Using web technologies, you can access PLCs and connect to SQL databases on any platform. Ignition provides the common HMI/SCADA capabilities, but in an easier and more efficient way. Create runtime clients and sessions that show current or past data, interact with your other systems and devices, create displays and controls for anything wired into your system. See and manage your entire system from one place.



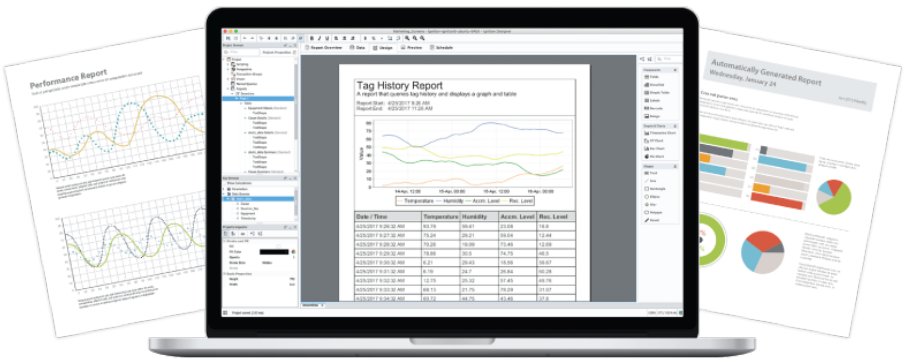
High Performance Historian

You can use any SQL database with Ignition's high-performance historian. It's simple to set up and Ignition will monitor and control the data for you. The Historian includes partitioning and uses compression algorithms for fast data storage and retrieval.



Reporting Engine

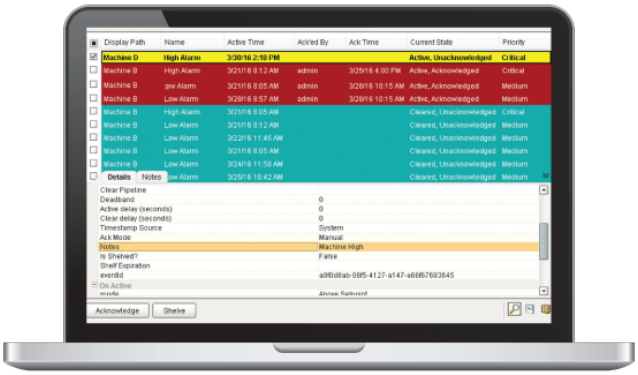
You can create dynamic PDF reports that users can see and change with on the fly. Give users the ability to change date ranges or select past reports without any changes to the system. Create schedules to email or save compliance reports when you need them.



Alarming

You can use Ignition's state-of-the-art alarming engine to create alarms for any condition you can imagine. Alarms can be recorded, acknowledged, or dealt with by sending out notifications. Email, SMS and voice messages can be sent to individuals or whole lists of users

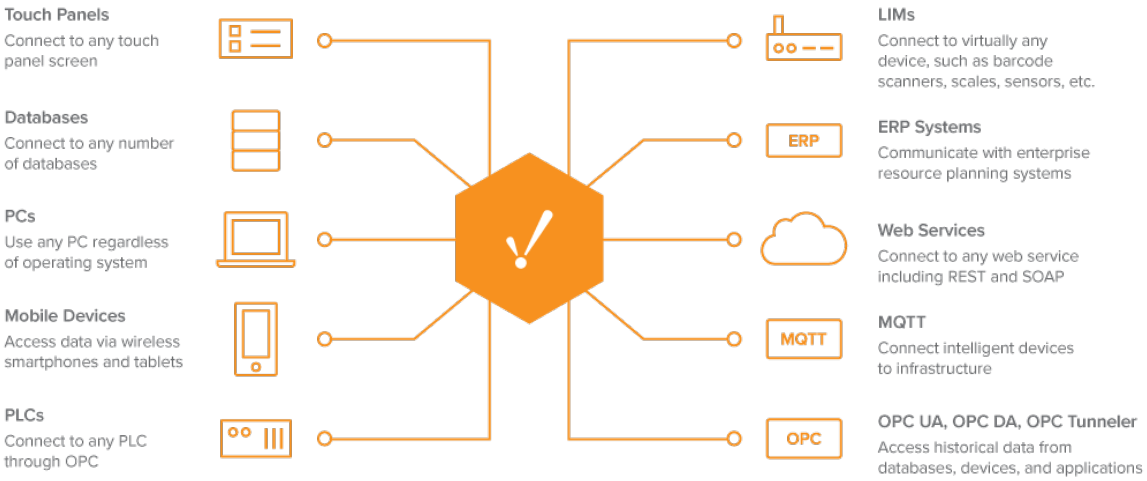
based on any logic you need.



Communications Hub

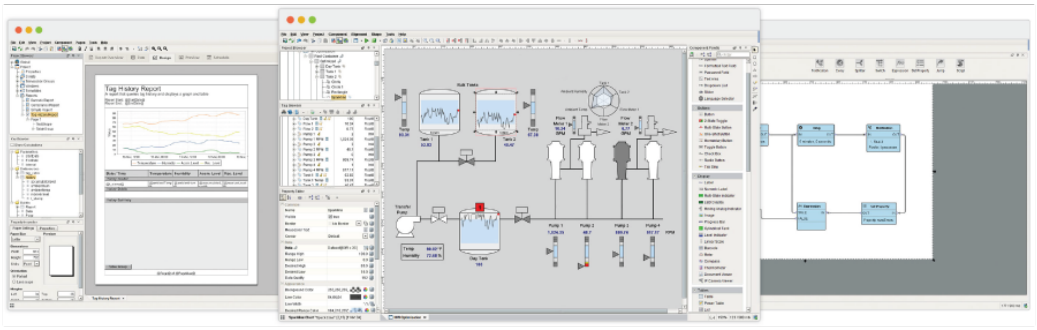
You can use Ignition as an effective communication hub in your network. You can use some or all of these features to make Ignition move data back and forth throughout your facility. Connect everything together with:

- OPC-based and communicates with virtually any OPC server
- SQL-based (JDBC) and connects to any number of Databases
- Built-in PLC drivers and connects to any PLC (Allen-Bradley, Siemens, Modbus, etc.) through OPC
- Supports any web services and launches anywhere
- Connects with other enterprise systems (ERP)
- Connects to devices such as barcode scanners, scales, and sensors
- Connects to any mobile device such as wireless smartphones and tablets
- Connects to any touch panel screen



Mix and Match

You can mix and match all of these features in any way you want!

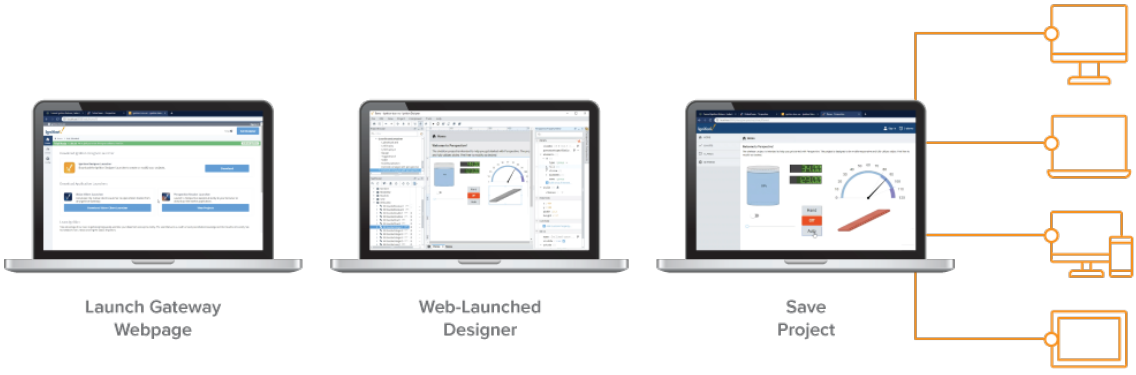


Understanding Ignition's Flow

Ignition is simple to set up and use, unlike anything else you've encountered in the HMI/SCADA world before. Getting installed and running is easy, and once you have your project set up it's extremely simple to start using. After installation, setting up and using a project follows this simple set of steps:

- **Launch the Gateway Webpage**
  The Gateway Webpage is where all of your information is stored, and where all of the general configuration happens. Here you can set up your Device connections, your Database Connections, alter your security, make backups, and set up redundancy.
- **Launch the Designer**
  Once your connections are in place, you can start using them to create a project. Opening a Designer is as simple as clicking a link in the Gateway. No installation is necessary, and you can even launch the Designer from any other computer on the network. Add Tags, query databases, show status, history trends, and alarms in your project.
- **Save your project**
  When you save, Ignition is sending all your data and all your changes back to the Gateway. Everything is centrally stored in the Gateway so you never have to worry about managing several versions or trying to combine many changes into one. You can even launch many Designers concurrently to allow multiple people to work on the same project at the same time. Projects aren't limited in scope, and you can create anything you can think of. HMI, SCADA, Reporting, Historian, and Alarming.
- **Launch Clients**
  Launch one Client or 20! Clients launch the same way as the Designer: simply click a link in the Gateway, no install is necessary. All of your clients talk directly to the Gateway to get new information on the fly, so there's no need to worry about them ever getting out of sync or loading changes individually to Clients.

Ignition's modular platform can support requirements of any type by selecting the modules for your needs and building processes suited for your business. Just create your project, it's all managed in one central platform and can be launched anywhere you want.



Related Topics ...

- System Architectures
- Quick Start Guide
- Installing and Upgrading
- Core Functions and Features

Quick Start Guide

Quick Start Guide

Are you new to Ignition? Here are a few simple steps to get going right away! In very little time, you can get a sample project up and running.

When getting started with Ignition, it helps to have a general idea about the major steps involved in designing your SCADA/HMI projects. Although there are many creative and innovative ways of using Ignition to do almost anything you want, we outlined some simple steps that can guide you in setting up a project.

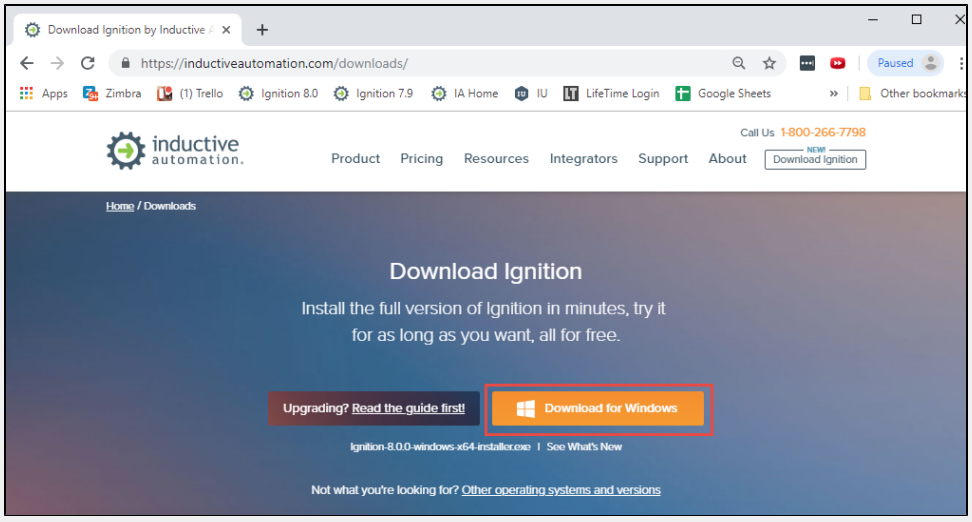
1. Download and Install Ignition

Installation is simple and can be done in under 3 minutes! Let's get started and install now.

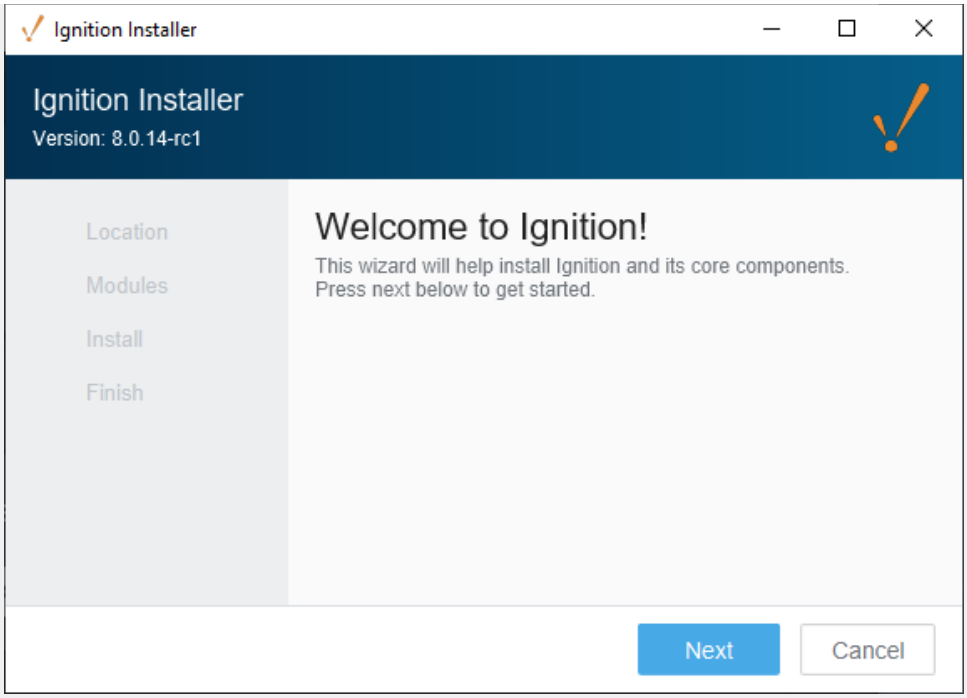
Before you install Ignition, make sure you meet the minimum system requirements:
• Dual-Core Processor (64 bit) • 4GB RAM • 10GB free HD space Also, ensure you have Administrator privileges to run the Ignition installer.

To Download and Install Ignition on Windows

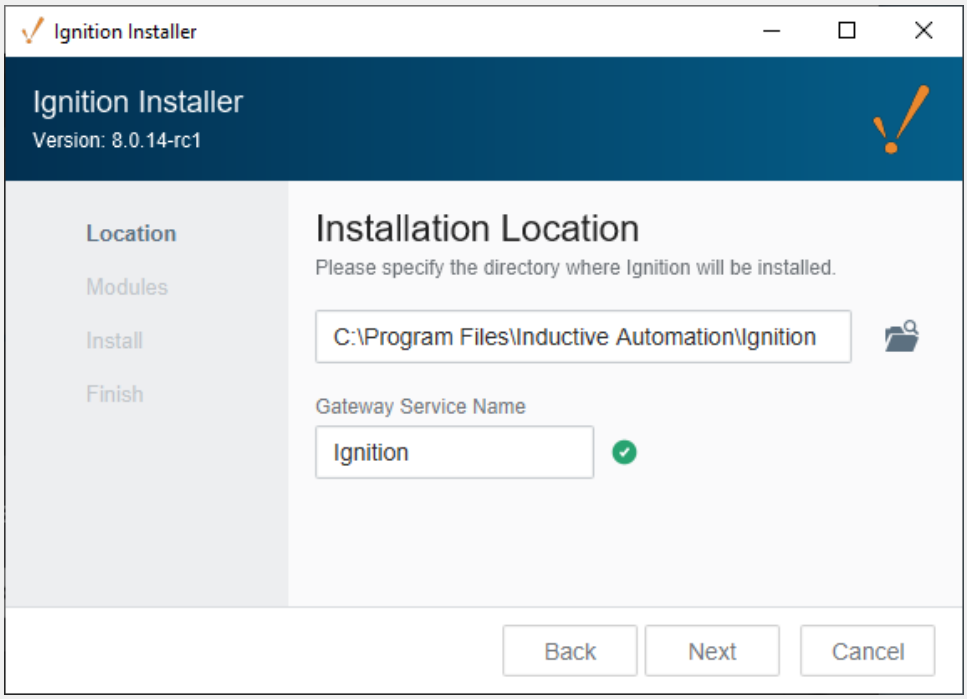
1. Go to the Inductive Automation website at www.inductiveautomation.com
2. Select **Download Ignition** in the upper right corner of the page. The browser will determine what operating system you're running and will bring up **Download for Windows.**
3. Click **Download for Windows.** If you want to see what other operating systems are supported, click on the Other operating systems and versions link.



4. Run the downloaded file **Ignition-X.X.X-windows-x64-installer.exe** (found on the lower-left of the window if using Chrome), or go to your downloads folder and double-click the file to start installation.
5. The **Ignition Installer** window welcomes you to the Installer Wizard. To being the installation, click **Next**.

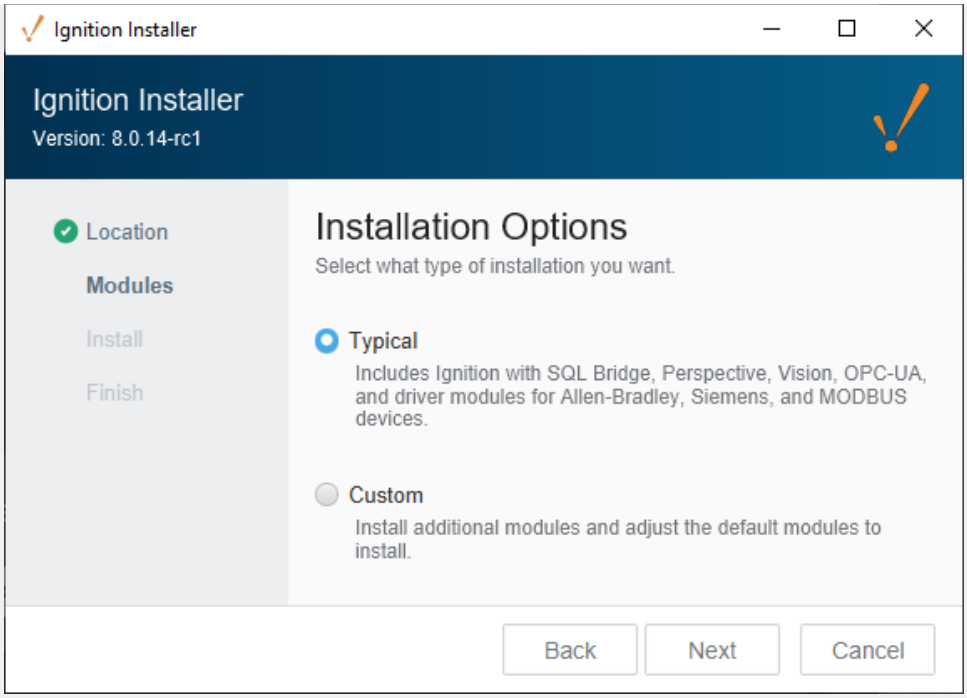


6. The **Installation Location** window will be displayed. By default, Ignition is installed under your Program Files folder: **Program Files\Inductive Automation\Ignition**. This is usually a good choice. You also have the option to choose another location. Simply click on the folder on the right side of this field to browser for another folder location. You can accept the default Gateway Service Name or enter another name. Click **Next**.



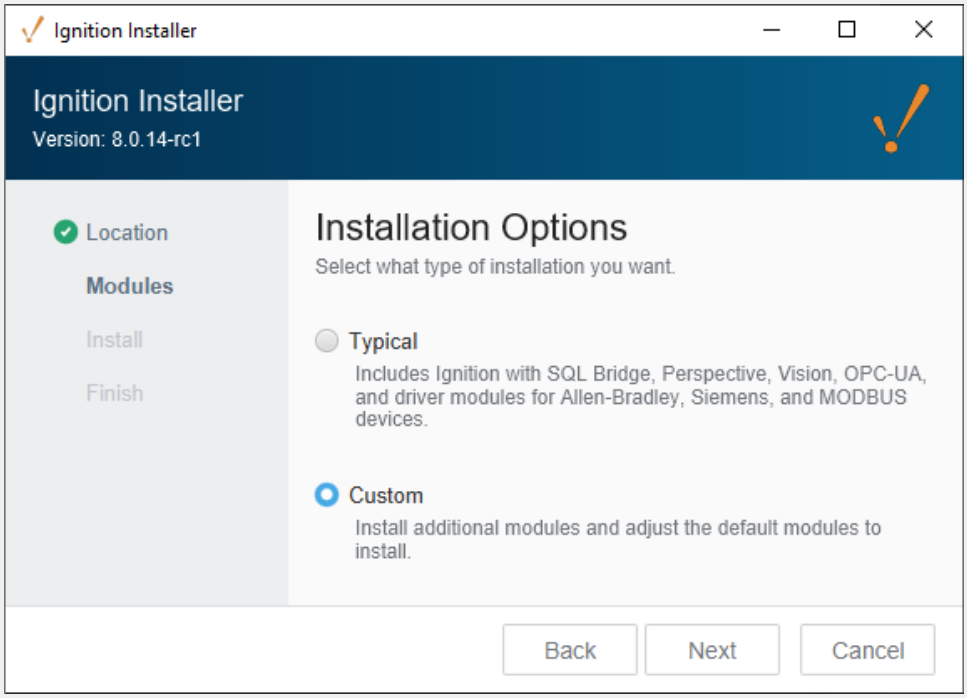
7. The **Installation Options** window will be displayed. Select either the **Typical** or **Custom** mode. The **Typical** installation includes Ignition with most of the most commonly used modules. The **Custom** installation lets you control and individually select the modules that you want installed. (Both options are shown in this step). If you selected a **Typical** installation, click **Next**.

   Typical Installation

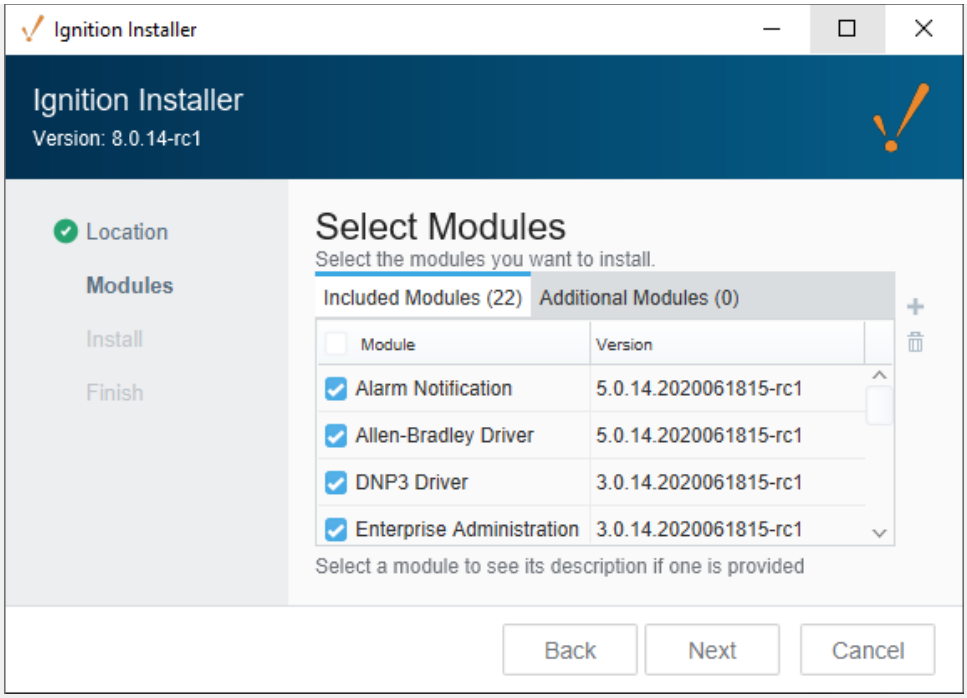


If you select **Custom** mode, check the box for the modules you want to install and uncheck those modules you don't want installed as part of your Custom Ignition installation. To see a brief description of the module, click on the module name.

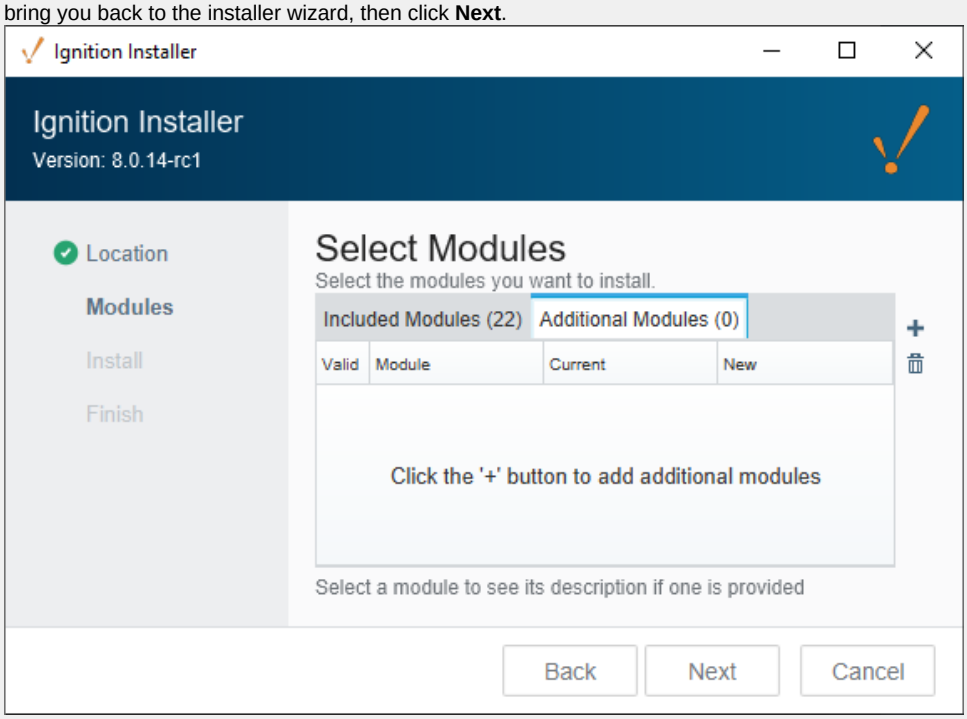
Custom Installation



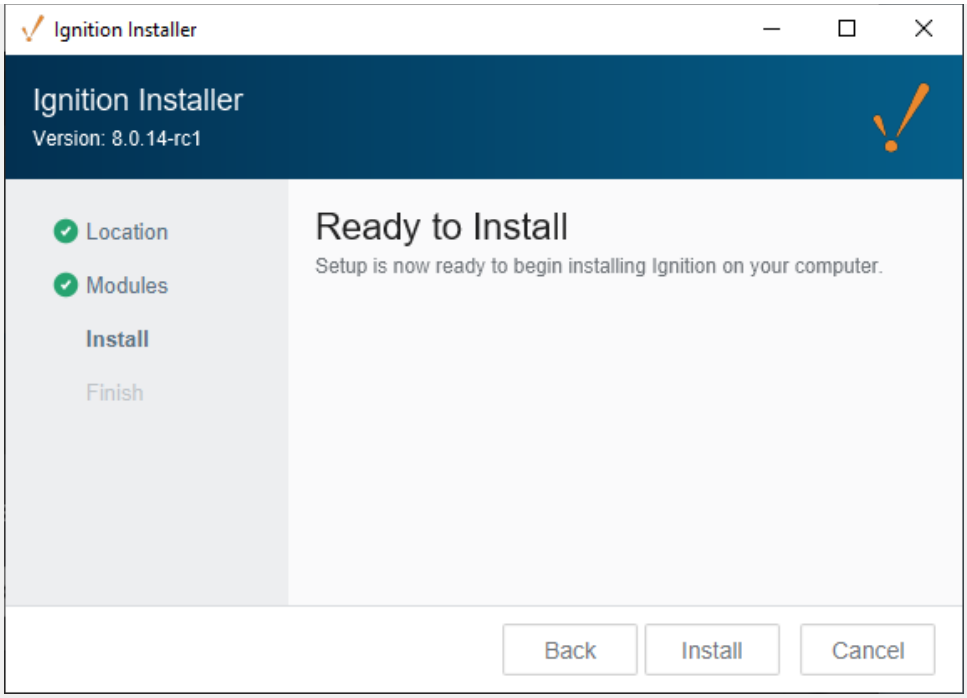
Select the modules to install, and then click **Next**.



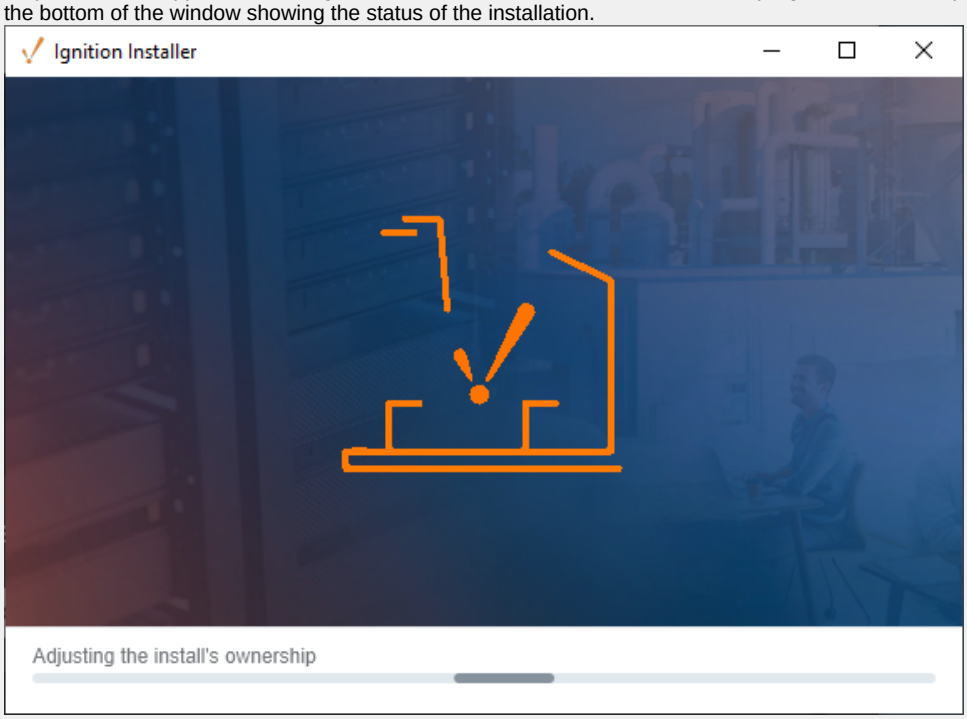
If you want to install additional modules, select the **Additional Modules** tab and the plus + icon. This will open your browser. Navigate to the location of your modules, select the modules you want to install, and click **Open**. This will bring you back to the installer wizard, then click **Next**.



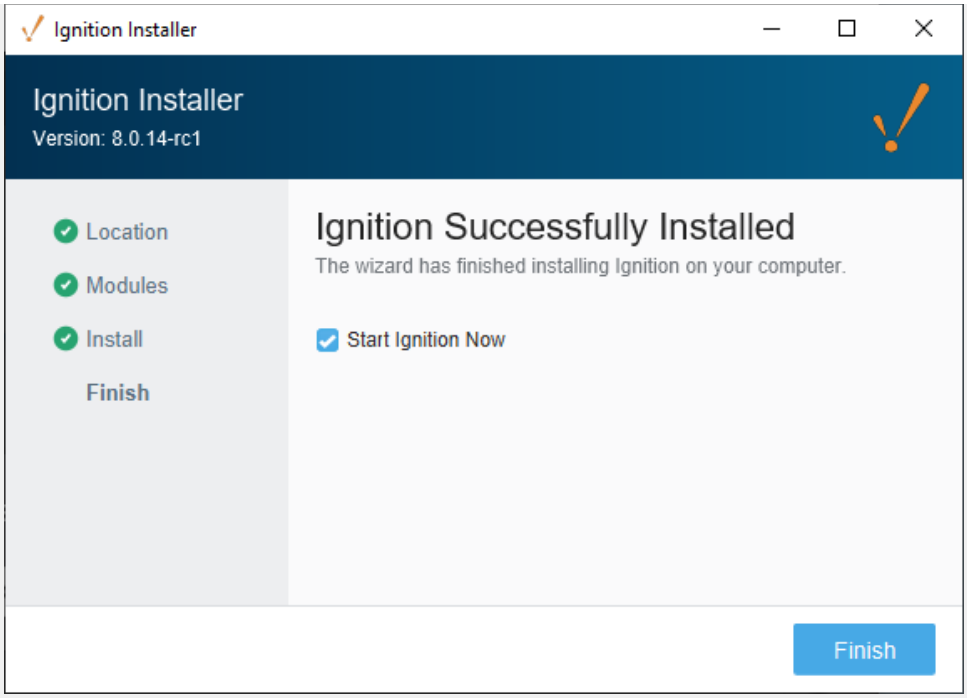
8. The wizard shows Ignition is now **Ready to Install**. If you decide to change your selections, click **Back** to go back and make your changes. Click **Install** to continue.



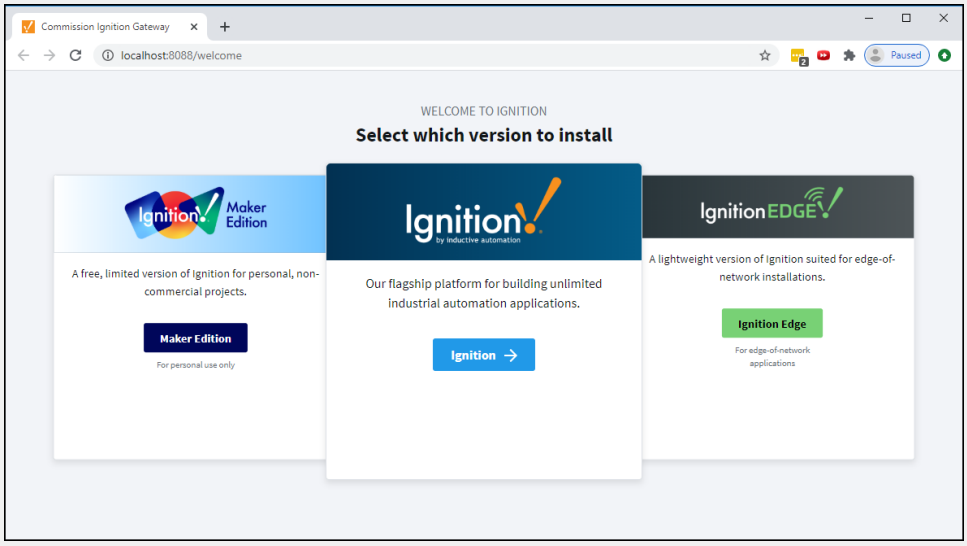
9. A splash screen appears indicating the installation of the selected modules is in progress. There is a progress bar at the bottom of the window showing the status of the installation.



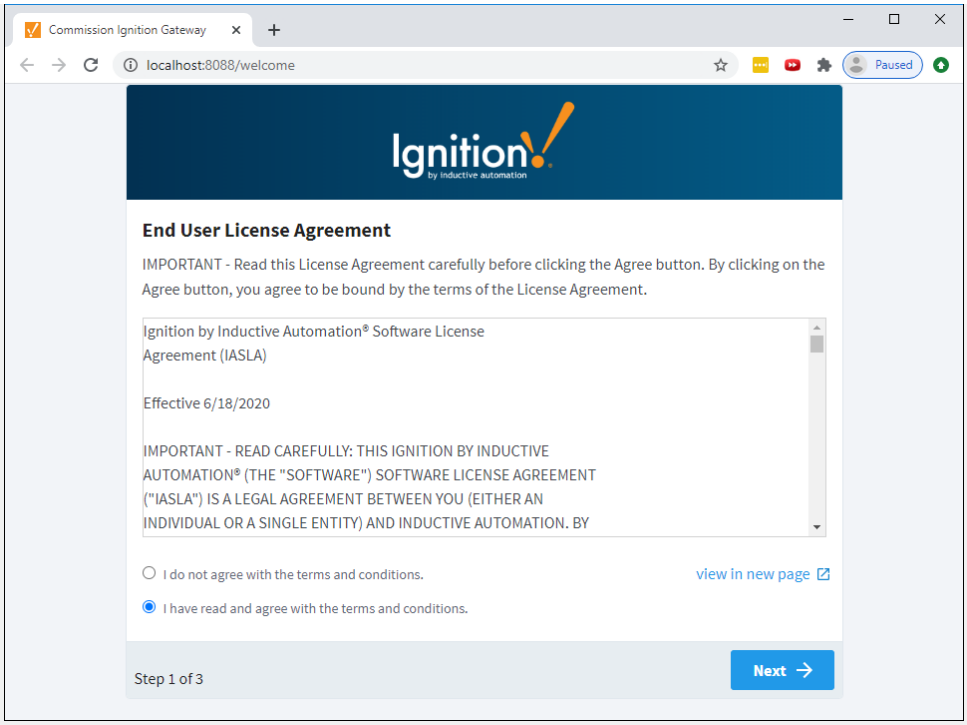
10. Once the Installer window shows that the Ignition installation is successfully installed, make sure that the checkbox for **Start Ignition Now** is selected, then click **Finish**.



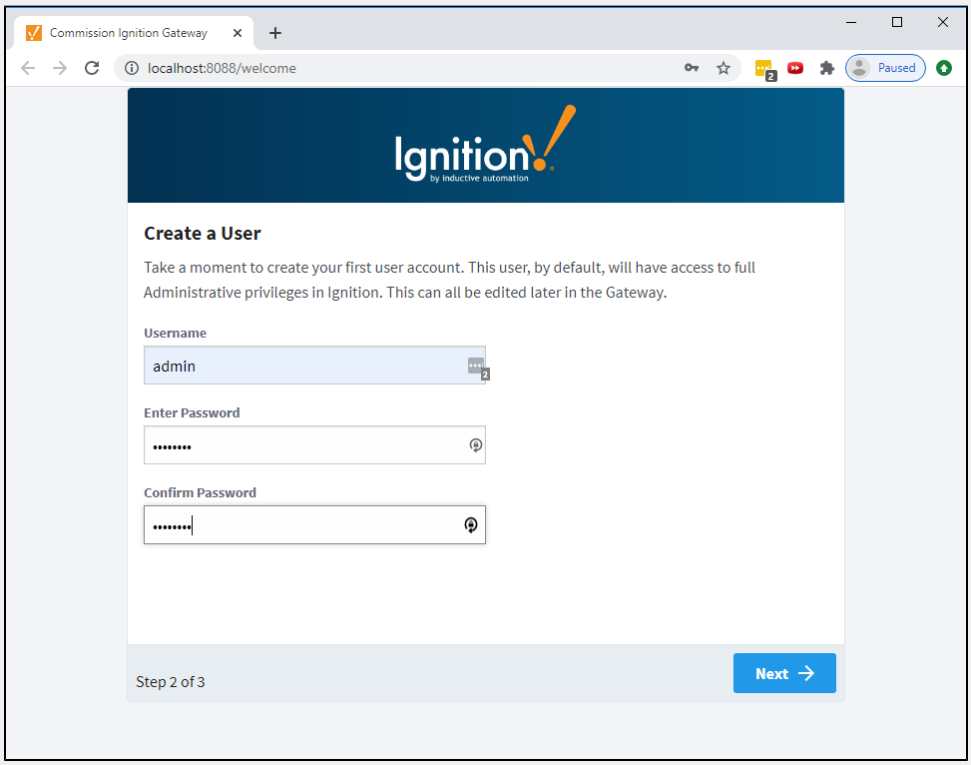
11. Once Ignition is installed, your default web browser will open and you will be greeted with a **Welcome to Ignition.** You will be presented with a message to select a version to install. Select **Ignition.**



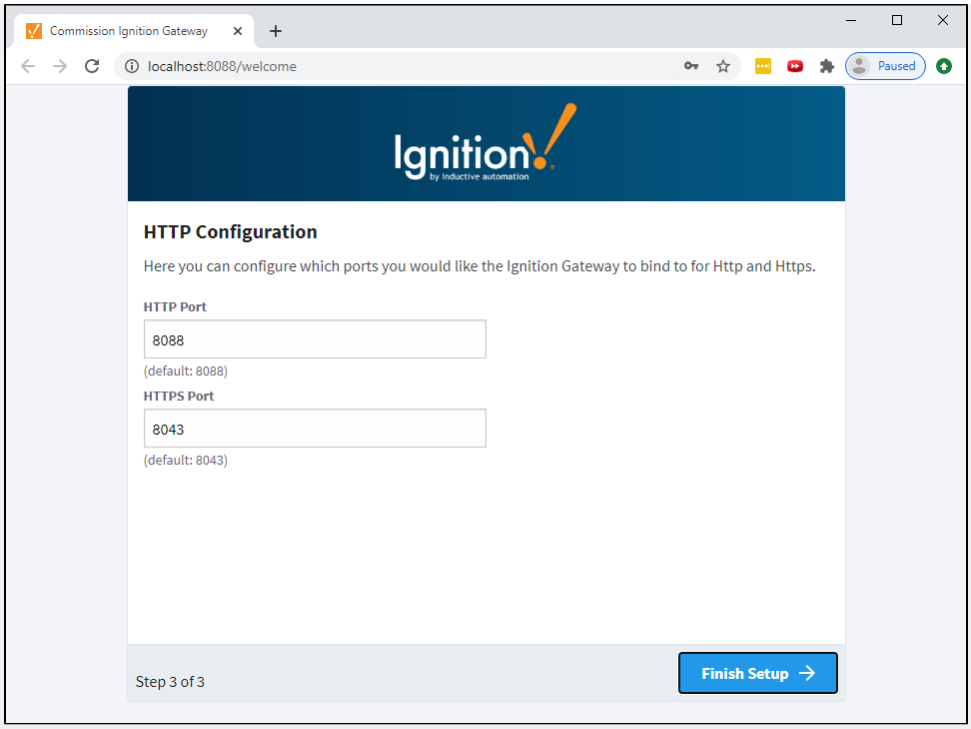
12. The **End User License Agreement (EULA)** window will be displayed. Click the button acknowledging you read and agree with the terms and conditions, then press **Next**.



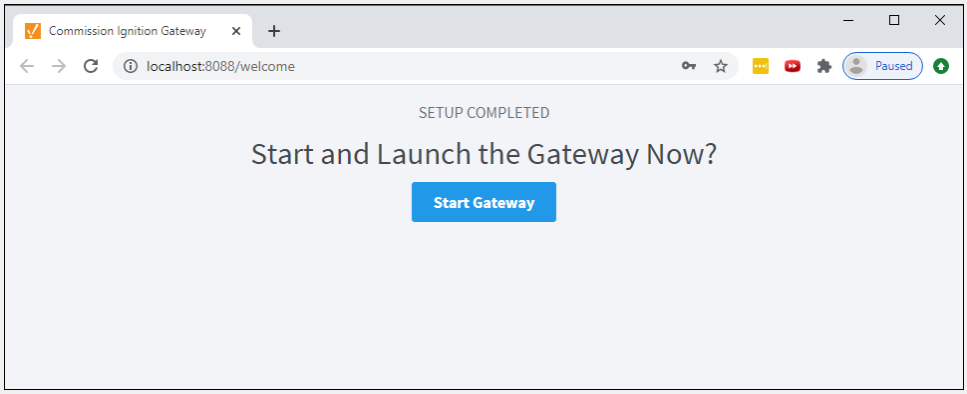
13. An **Account Setup** window will open for you to enter your first user. This first user will be the main / administrator account with full privileges in Ignition. Enter a **username** and **password**, and click **Next**.



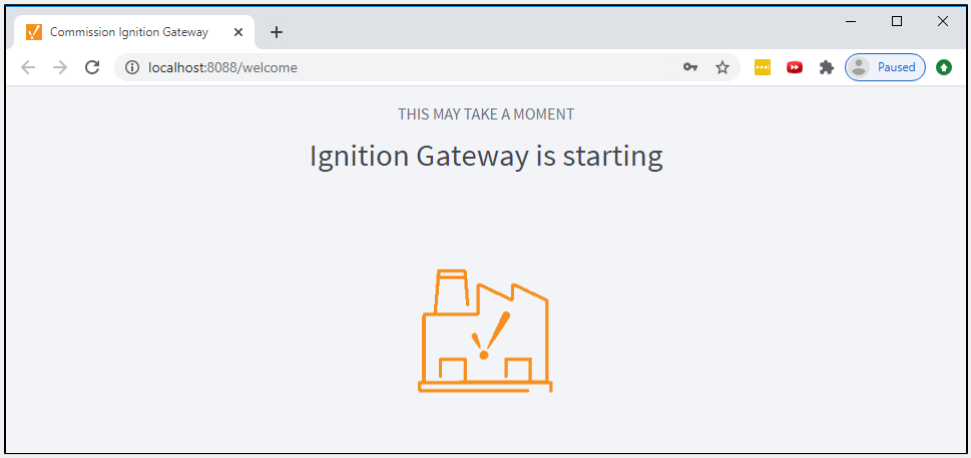
14. The **HTTP Configuration** window will open for you to configure the HTTP and HTTPS Ports. The default HTTP Port is 8088, and the default HTTPS Port is 8043. Click **Finish Setup.**



15. The **Setup Completed** window will be displayed, and now you can **Start the Gateway**.

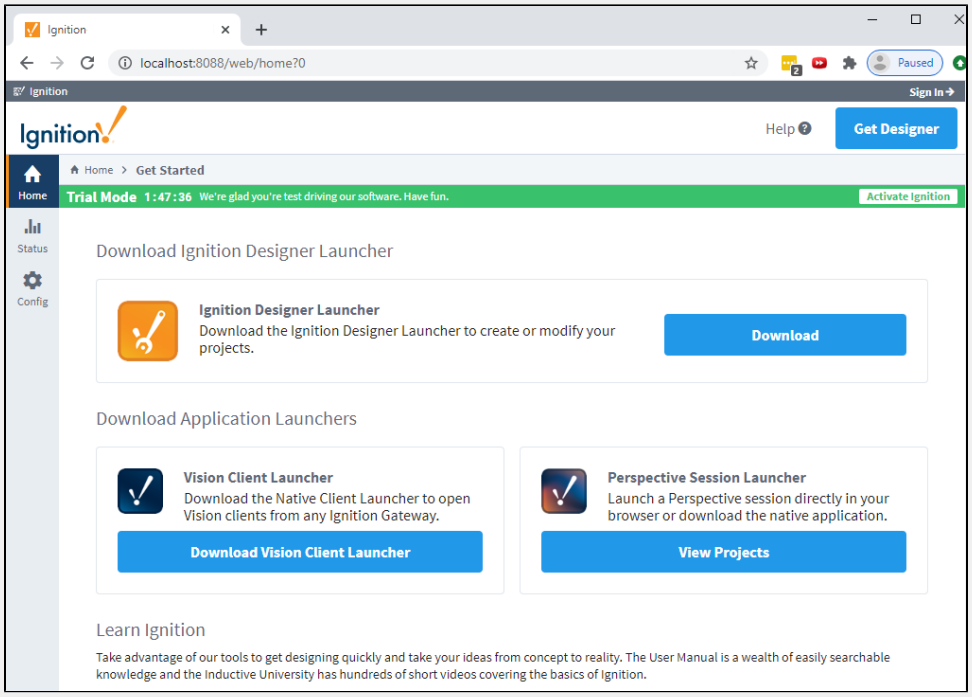


A splash screen informs you that Ignition is starting. This may take a few minutes to finish



16. That's it! Ignition installation is complete and the Gateway is launched. Your web browser opens the Gateway Homepage at **http://localhost:8088.** Sign in using the credentials for the administrator or user with full privileges for

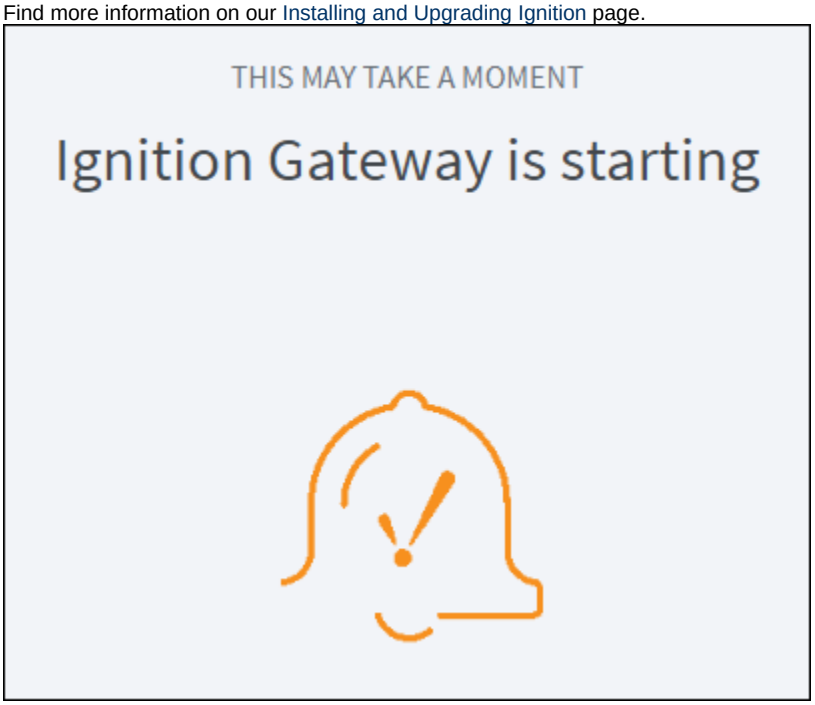
the Gateway as shown in Step 13 above.



Ignition installs itself as a Window Service (**Start > Control Panel > Administrative Tools > Services > Ignition Gateway**), so it starts automatically when your computer starts up.

To learn more, go to Windows - Install!

Find more information on our Installing and Upgrading Ignition page.



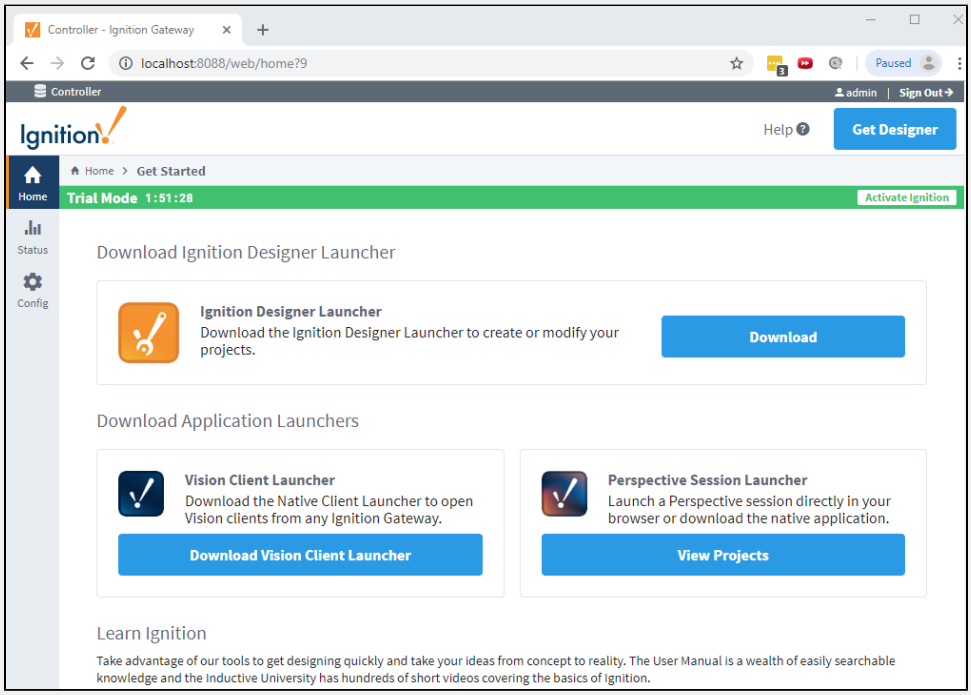
2. Launch the Gateway Webpage

The Ignition Gateway is a web server. When it is running, you access it through a web browser in just two steps.

If you're not already logged into Ignition, enter the credentials for the administrator or user with full privileges for the Gateway.

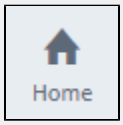
1. Figure out the IP address of the computer you installed Ignition on. If this computer is the one Ignition is installed on, then you can use **localhost** instead of the IP.

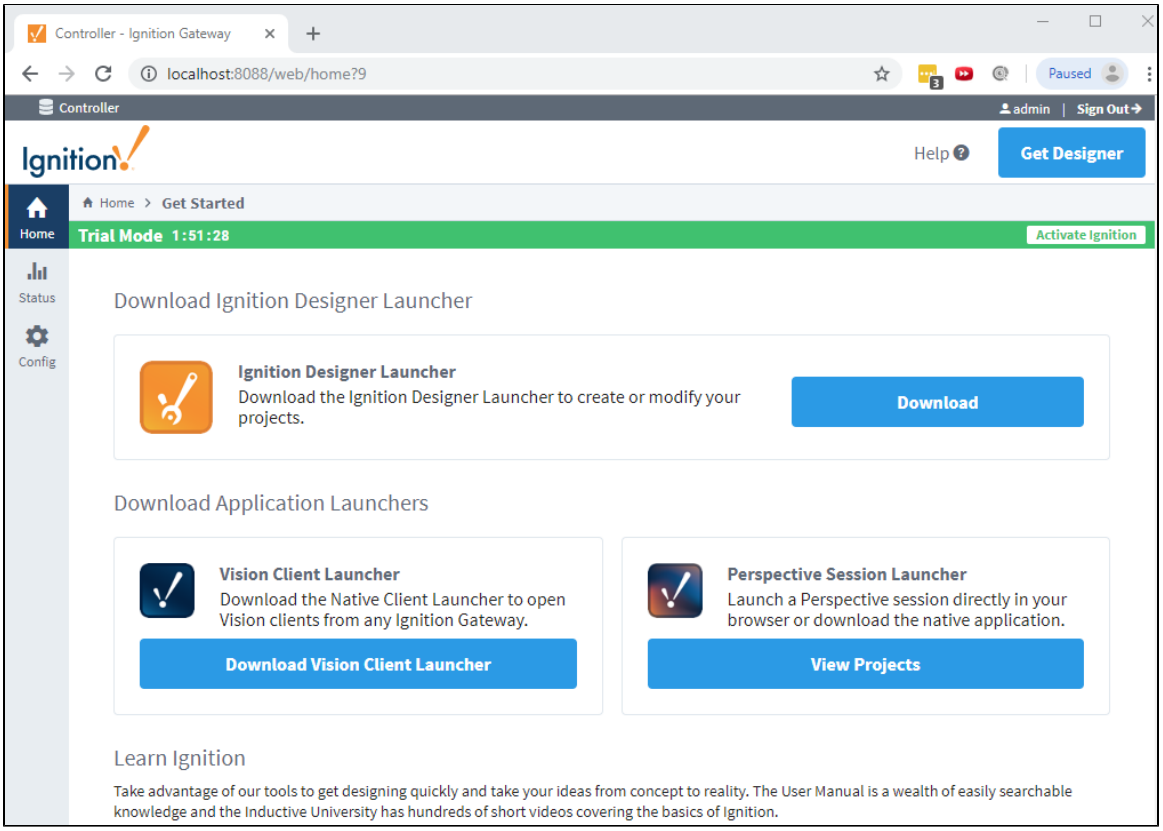
2. Open up a web browser and go to this address: **http://ipaddress:8088** and it will bring up the Gateway webpage. If you are using localhost, it will look like this: **http://localhost:8088.**



   The first time you go to the Gateway webpage, it shows you the **Designer**, **Perspective Projects**, and **Vision Client Launchers**.

3. To navigate back to the Gateway Homepage, click the **Home** button.





3. Connect to a Device

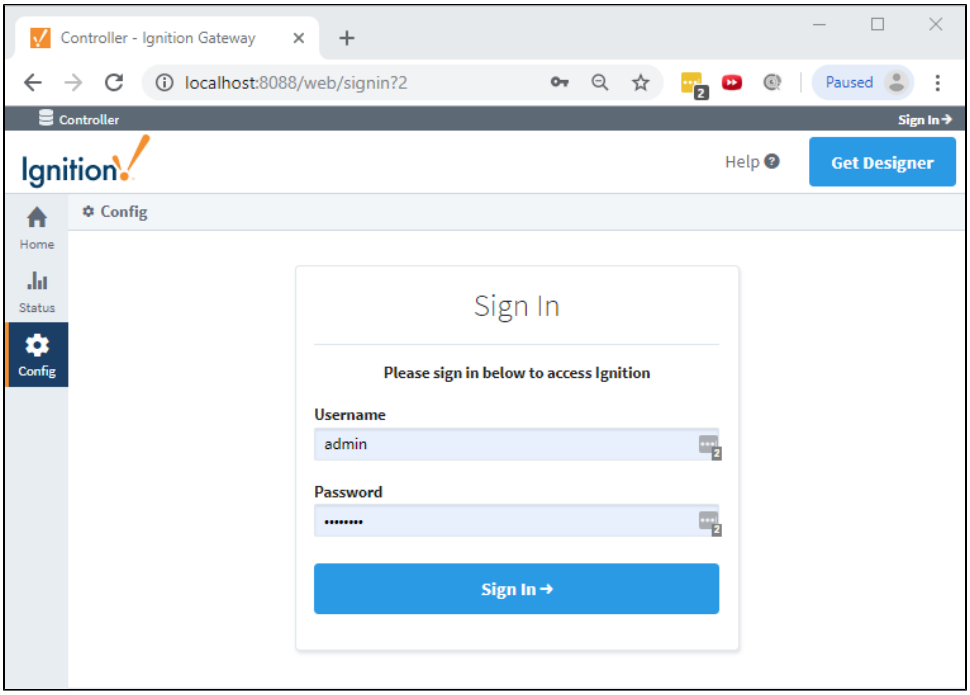
With the Ignition OPC UA module and device drivers installed, connecting to a device is simple.

With the Ignition OPC UA module and device drivers installed, connecting to a device is simple. The Programmable Device Simulator, as described below, includes Generic, Dairy and SLC simulator devices. It provides functionality that allows you to create your own simulator program with outputs you define. For more in depth information, refer to Programmable Device Simulator.

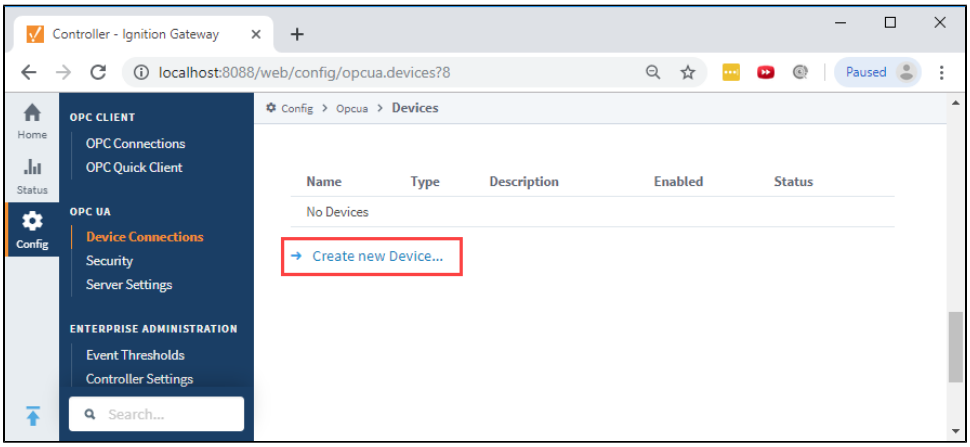
Using the Simulator in Ignition version 8.0.7 or before

If you are using the Simulator in Ignition version 8.0.7 or before, click here for the documentation. If you delete your existing simulator (prior to Ignition 8.0.8), adding the new Programmable Device Simulator and loading the same program, will allow addresses within the device to map existing addresses.

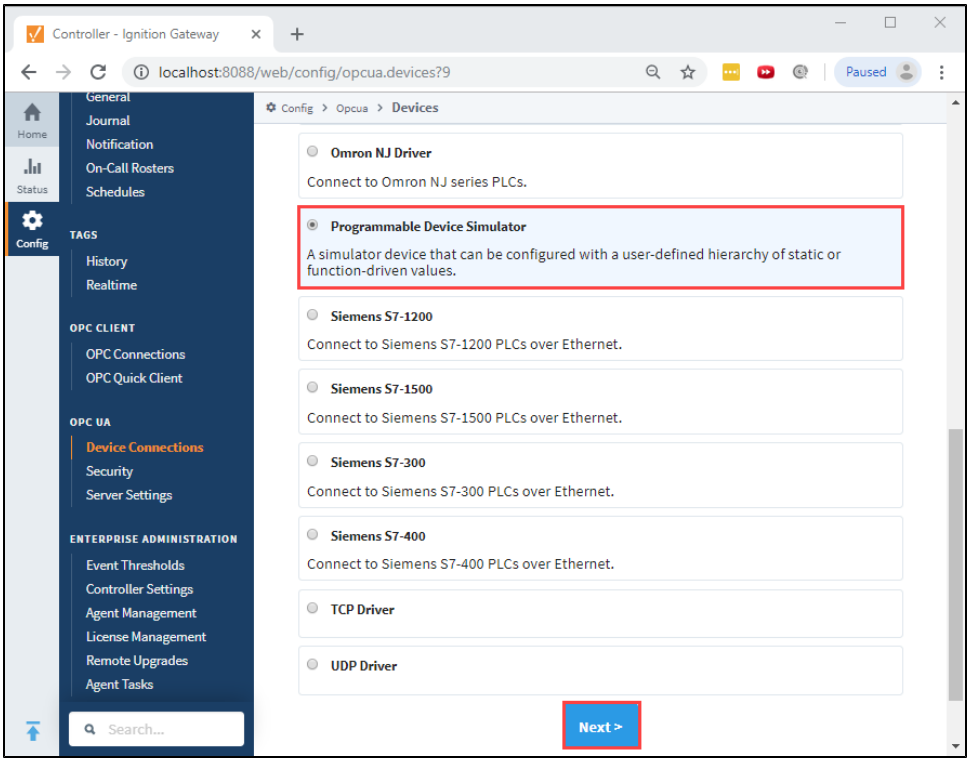
1. If you are not already logged into Ignition, go to the Gateway webpage. Enter the credentials for the administrator or user with full privileges for the Gateway, and click **Login**, and click the **Config** tab.



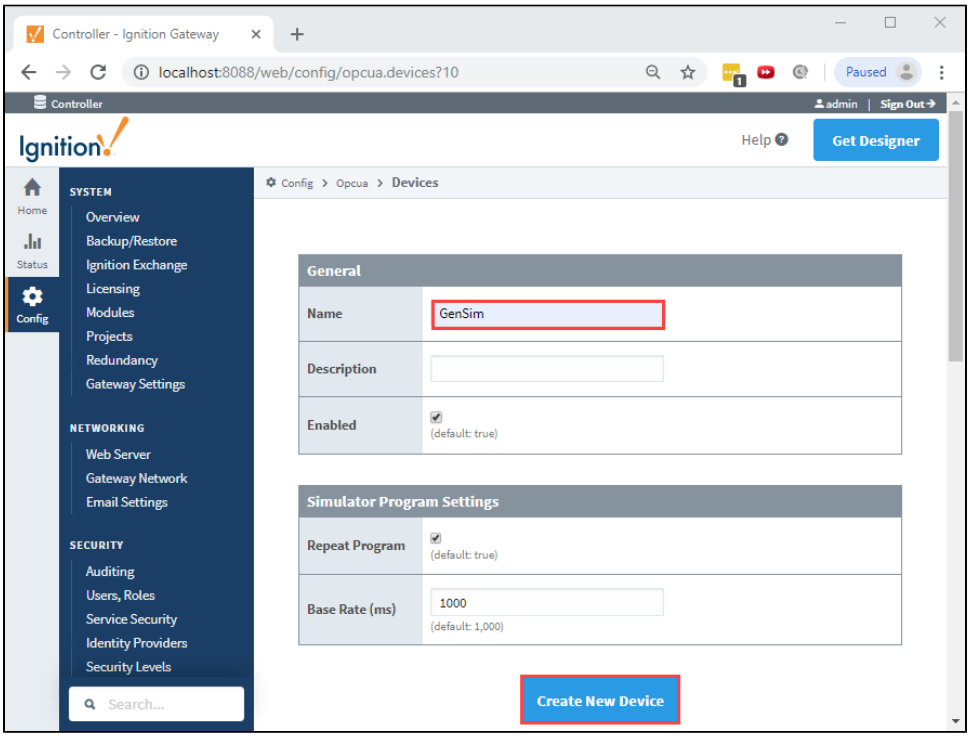
2. Scroll down to the **OPC UA** section on the left and click the **Device Connections** link.
3. To add a new Device Connection, click the **Create new Device...** link.



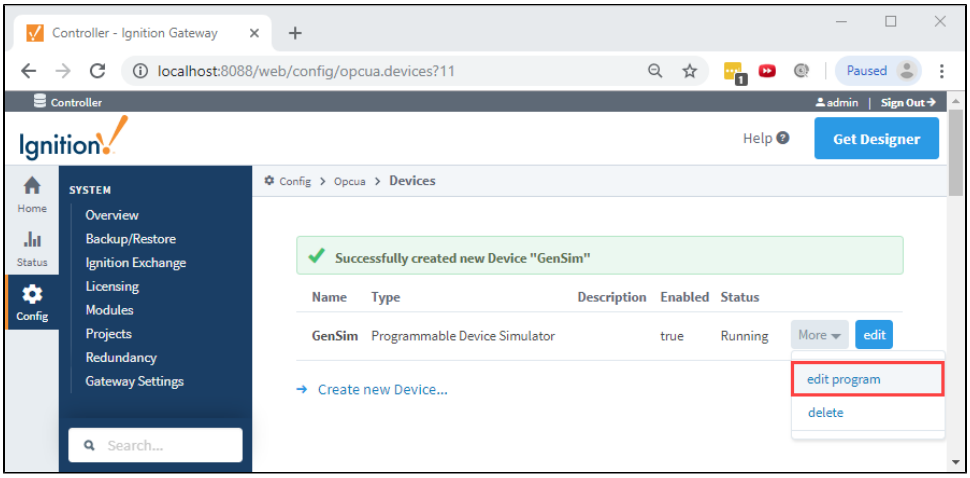
4. Select the **Programmable Device Simulator** and click **Next**.



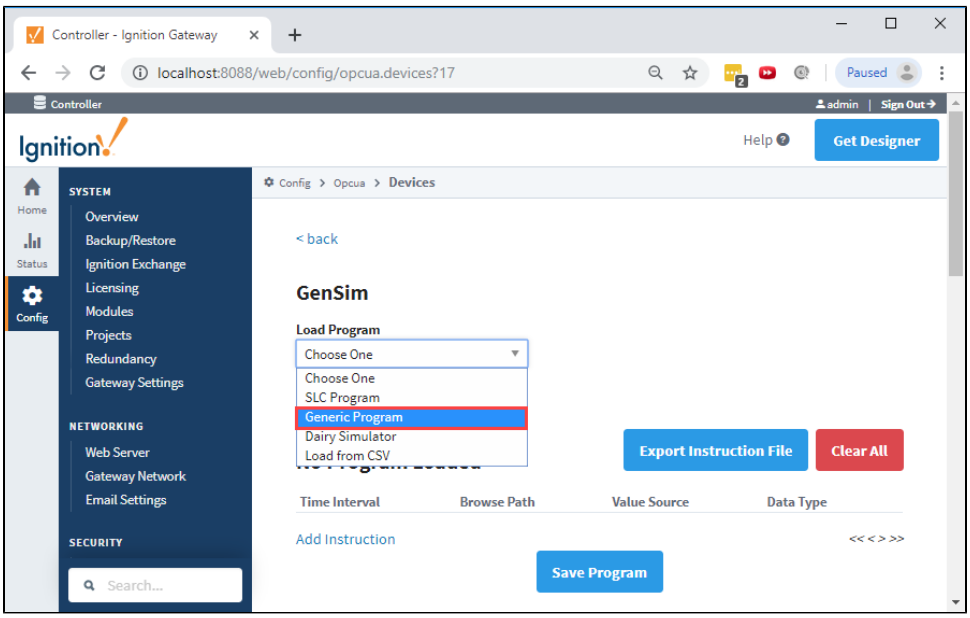
5. Give the new connection a name and click **Create New Device**.



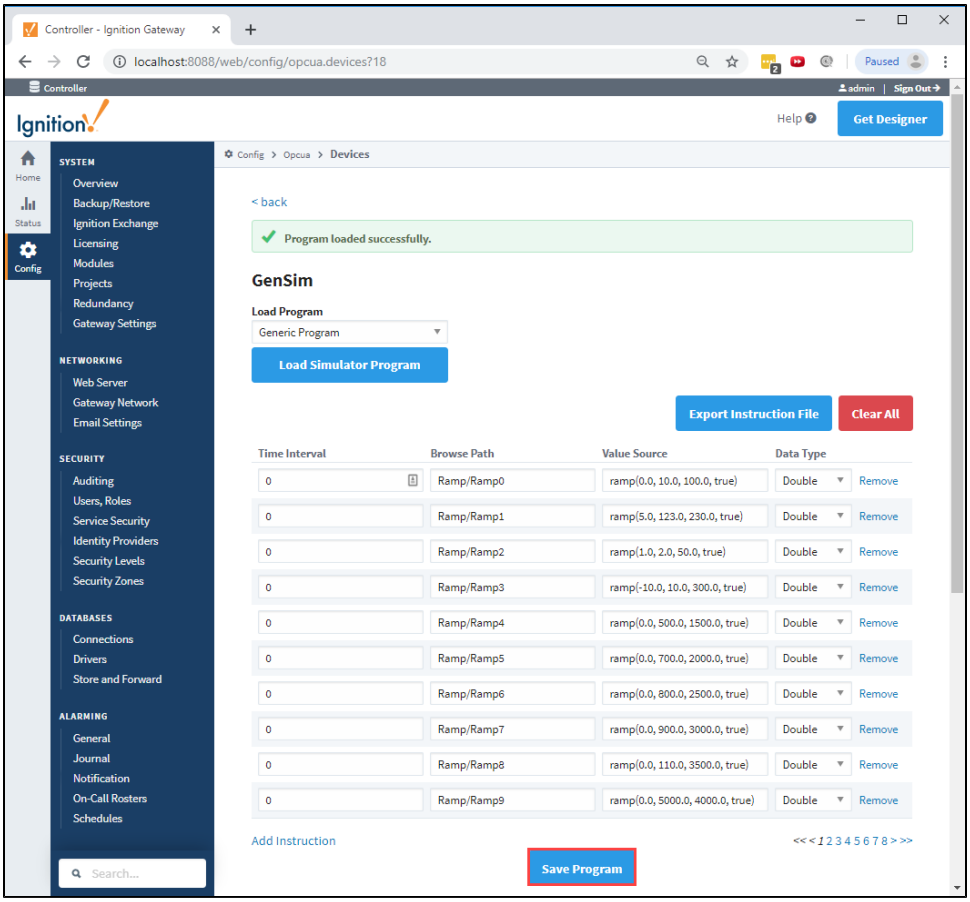
6. The window will refresh and you'll see your device was successfully created with a status of "Running". Now you can select the program for the simulator by clicking **More > edit program**.



7. From the **Load Program** dropdown, select a program (i.e., Generic Simulator Program) and click the **Load Simulator Program** butt on.



8. You will see a number of instructions which expand multiple pages. This is how the simulator creates Tags and sets their values. Click the **Save Program** button.

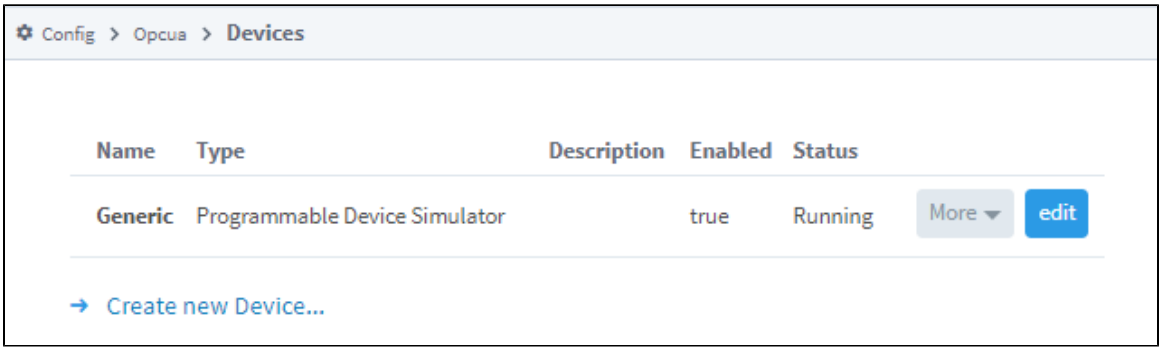


9. You have the option to modify the program before or after saving if you want to change some of the functionality. If you want to add an instruction, click the **Add Instruction** link under the last instruction on the screen. You can also **Remove** the instruction if necessary.

Ignition can connect to many different devices natively with the built-in OPC UA Server.

- Allen Bradley Ethernet
- Modbus
- Siemens
- UDP and TCP Driver
- DNP3
- Omron NJ Driver
- Programmable Device Simulator
- Third Party OPC Servers

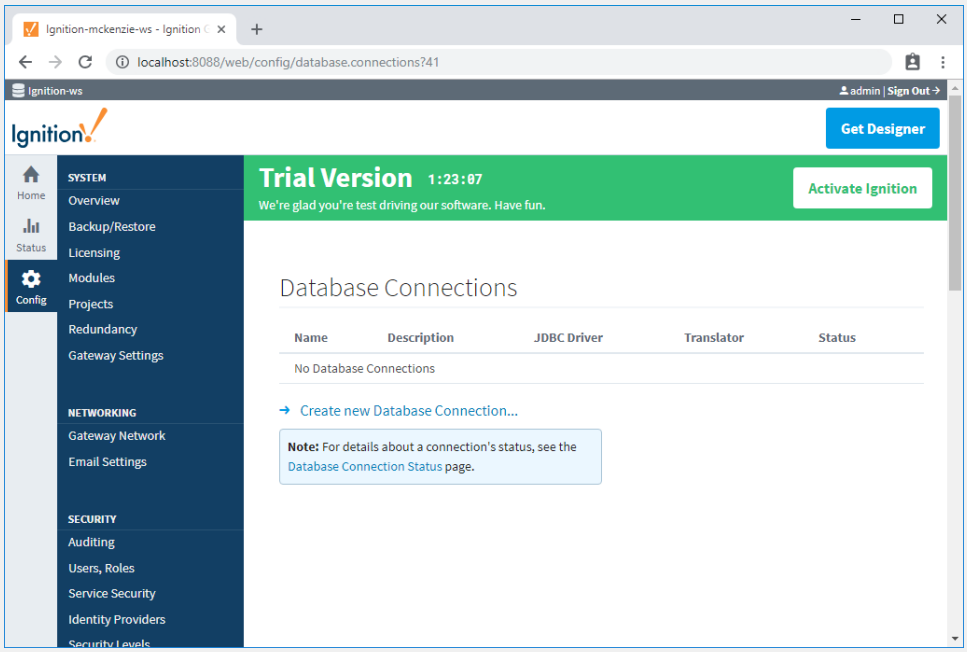
To learn more, go to the OPC UA and Devices pages!

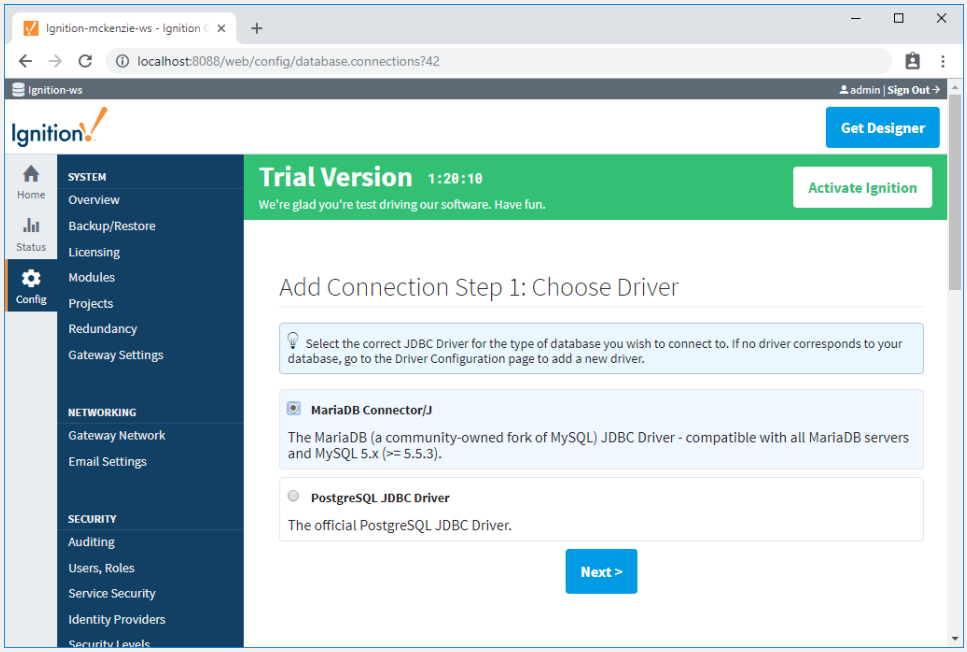


4. Connect to a Database

After you install your database, you can connect it to Ignition.

1. If you are already logged into Ignition, then go to the **Config** section of the Gateway Webpage, otherwise login as the user with full / administrative privileges.
2. From the **Config** section, navigate to the **Databases > Connections** page.
3. Click the **Create new Database Connection...** link.
4. Select the **MariaDB Connector/J** option and click the **Next >** button. The MariaDB connector will work for many types of MySQL installations. It is recommended to get the official driver, see the JDBC Drivers and Translators page for more information.

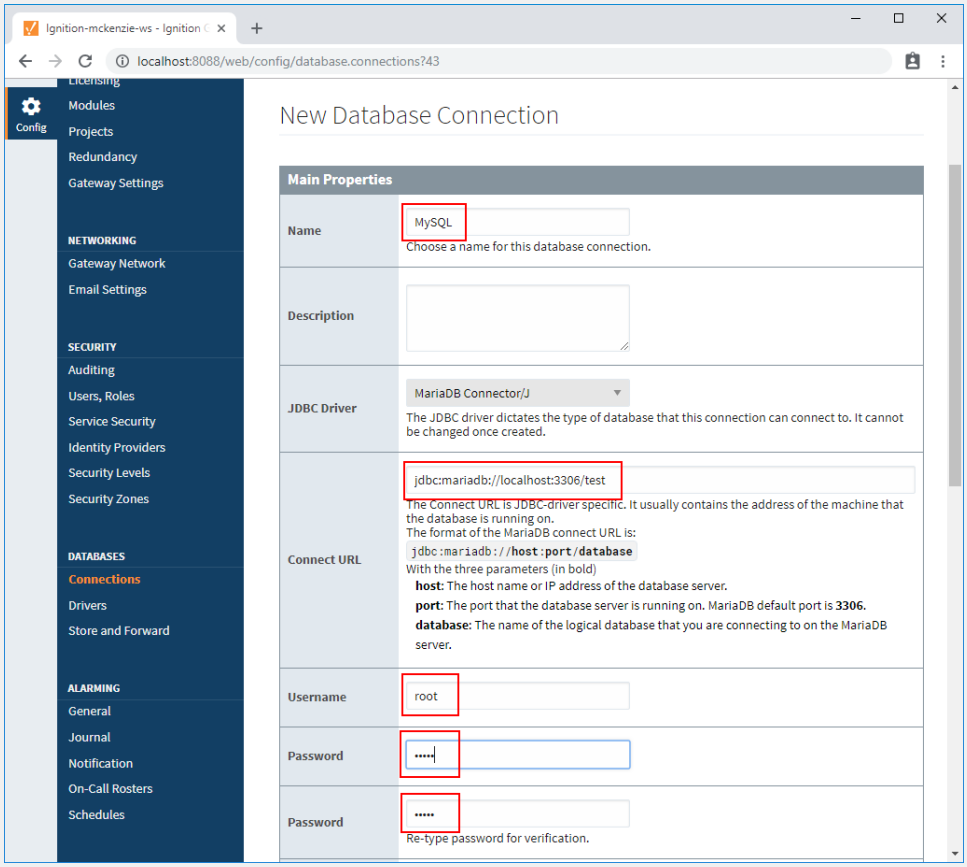




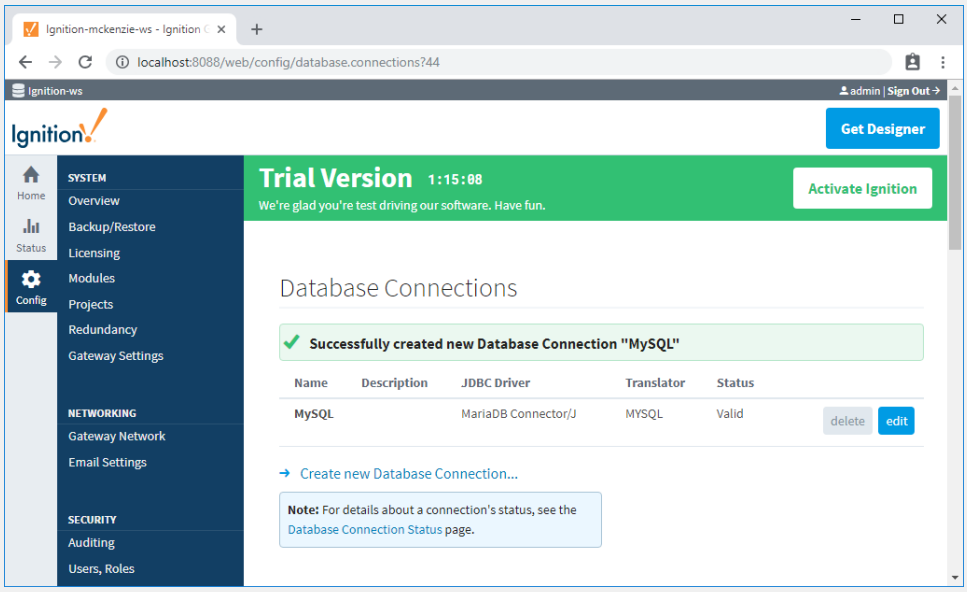
5. Enter a **Name**, **Connect URL**, **username**, and **password**.
   If you just installed MySQL on the same machine as Ignition and it has a schema called **test** you will be able to use the following Connect URL:
   Connect URL: **jdbc:mysql://localhost:3306/test**

The username and password should have been set when MySQL was installed, but **root** is often the username for test systems.

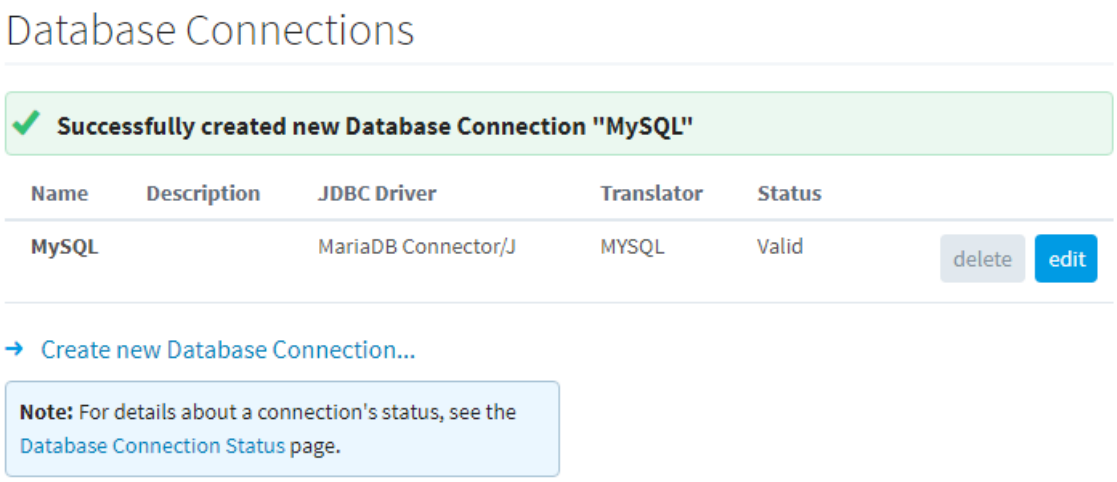
Scroll to the bottom and click the **Create new Database Connection** button.



A "success" message will appear when the database connection is created.



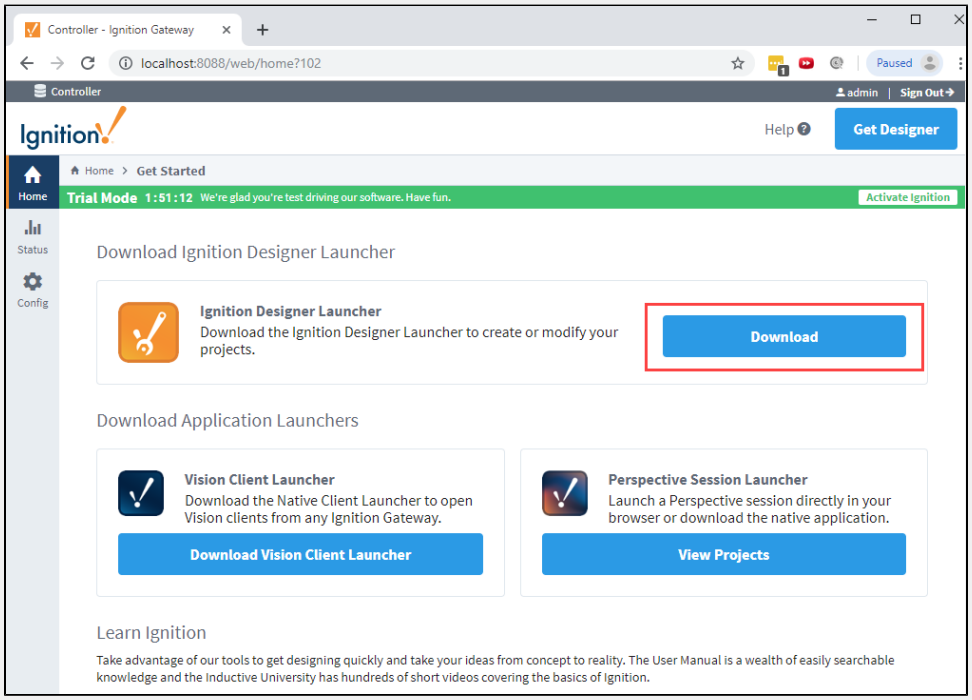
To get detailed descriptions for a number of different database connections, see Connecting to Databases. You can also go to Database Connections to learn how databases are used in Ignition and how to get started.



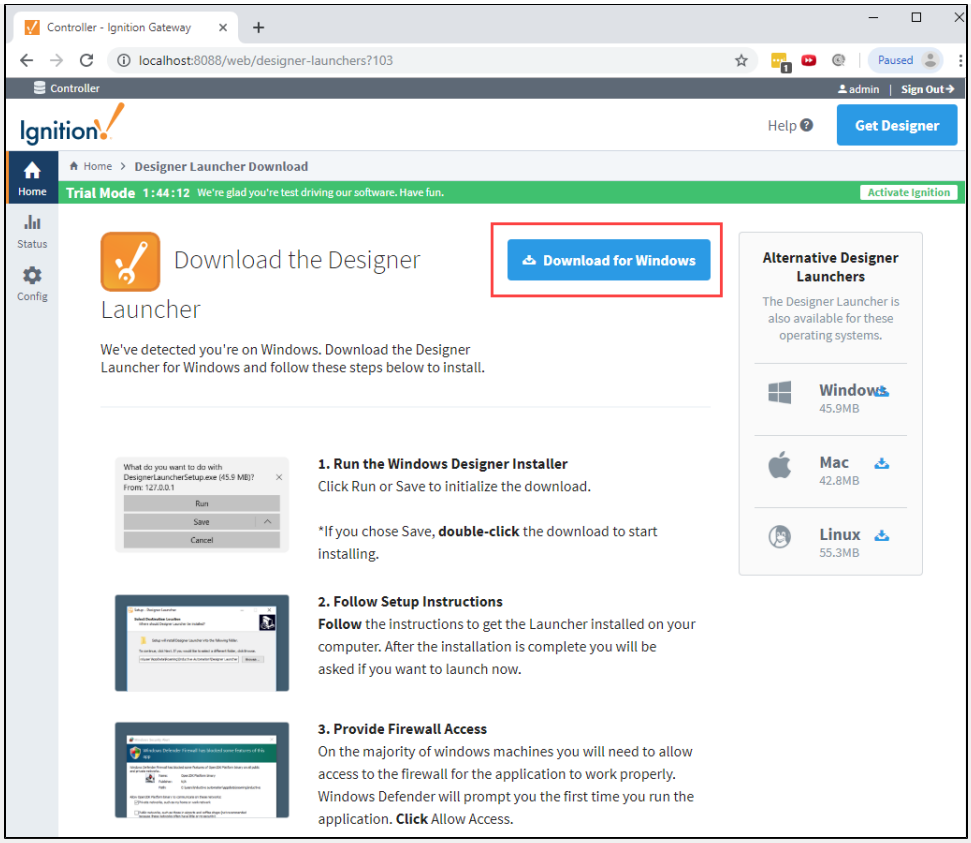
5. Open the Designer

Now that you set up a device (Step 3) and database (Step 4) connections, let's open the Designer and create a project.

1. If you are already logged into Ignition, then go to the Homepage of the Gateway Webpage, otherwise login as the user with full / administrative privileges.
2. From the Gateway webpage, click the **Download** button for the Designer.

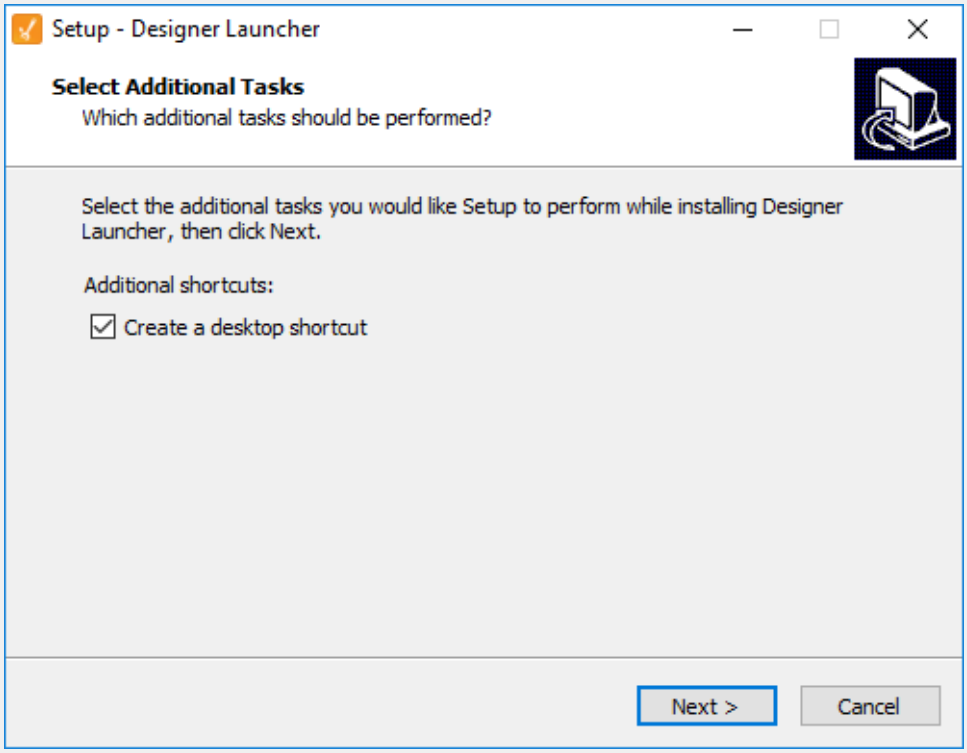


3. Select the **Download** button for the platform you are running on: Windows, Mac OS, or Lunix. In this example, we are running on Windows.

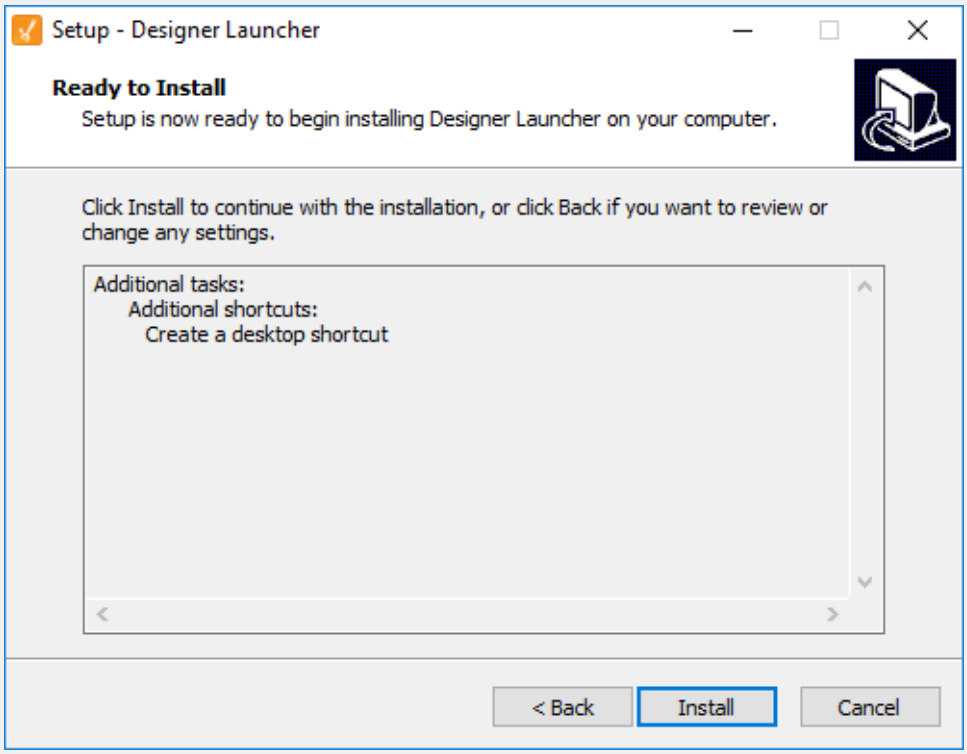


4. Run the downloaded file (**DesignerLauncherSetup.exe**) found on the lower-left of the window, or go to your downloads folder and double-click the file to run it.

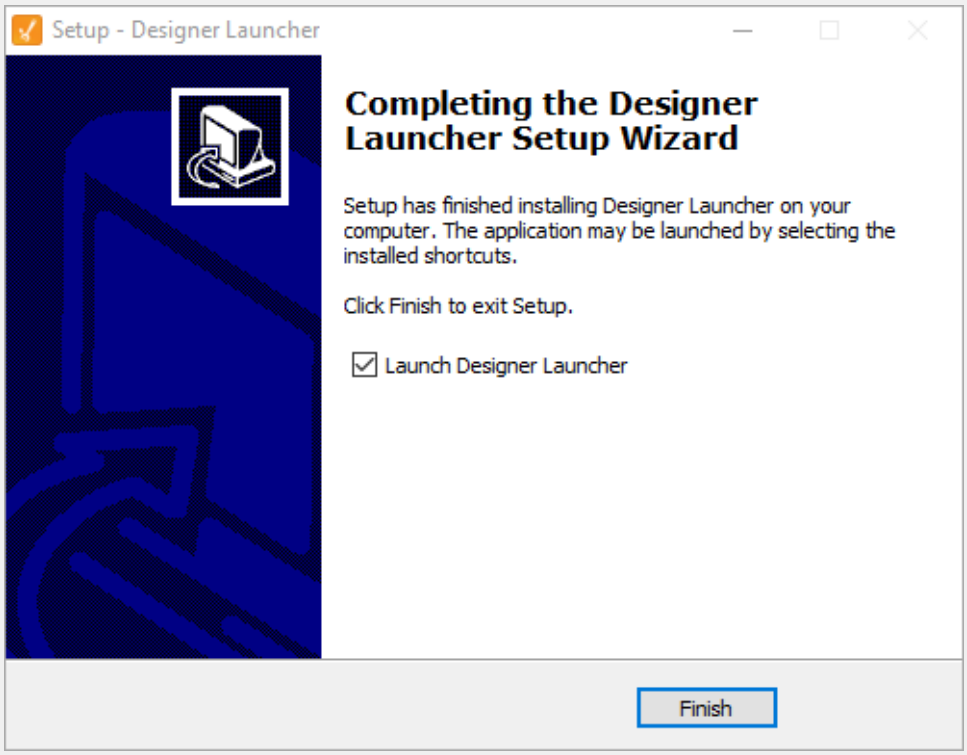
5. The Select Additional Tasks window will open. Here, you can select additional tasks that you would like to setup to perform while installing the DesignerLauncher. It's a good idea to **Create a desktop shortcut** so it's easily accessible when you want to open the Designer Launcher. Click **Next**.



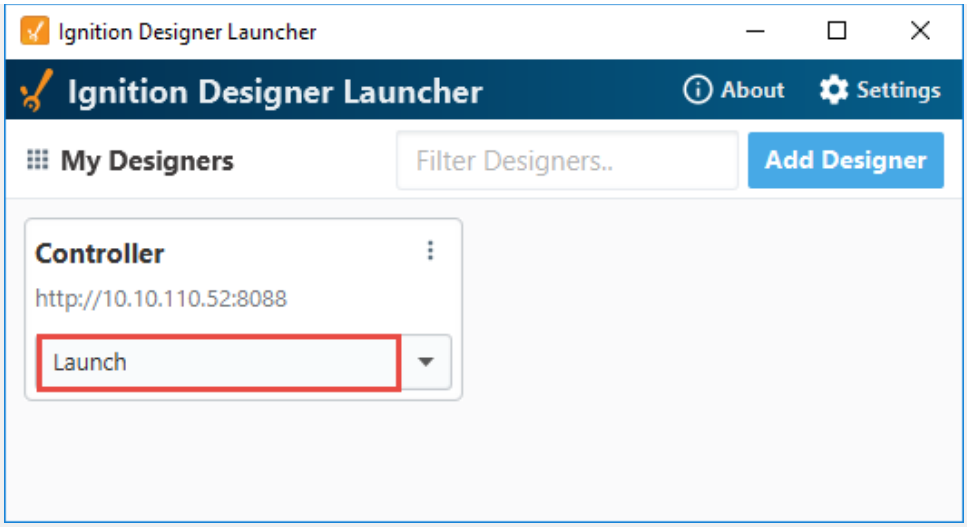
6. Now, you're Ready to Install the Designer Launcher on your computer. Click **Install** to continue.



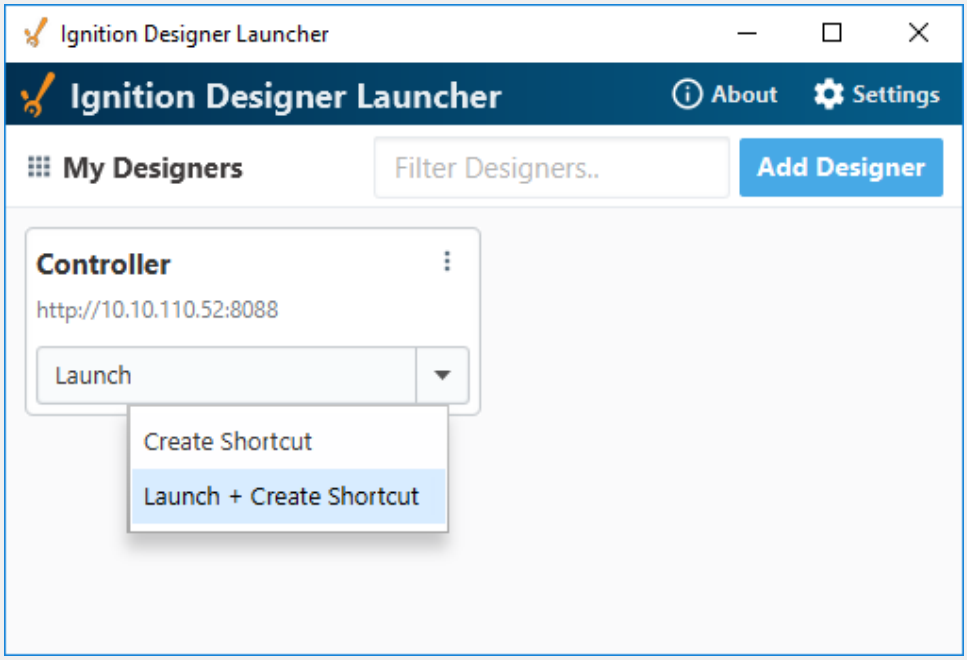
7. Once the Designer Launcher is installed, complete the setup by checking the box to **Launch Designer Launcher** and click **Finish**.



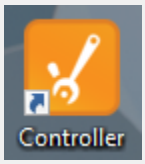
8. The Ignition Designer Launcher window will open. Here you can select any Designers you have permission to launch. If you are running this for the first time, by default, you will only see one Designer. Click **Launch**.



a. You also have the option to **Create Shortcut**, or **Launch + Create a Shortcut** on your desktop by clicking the dropdown.

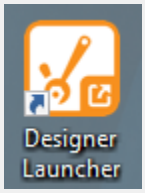


If you Create a Shortcut, it will look like the image below with the name of your Designer (i.e., **Controller**) on it like the icon below. Click on the shortcut icon to launch your Designer.



Since a Designer Launcher icon was created in Step 7, it is aready on your desktop. By clicking the Designer Launcher icon, it will open the Ignition Designer Launcher window and display all your Designers. To learn

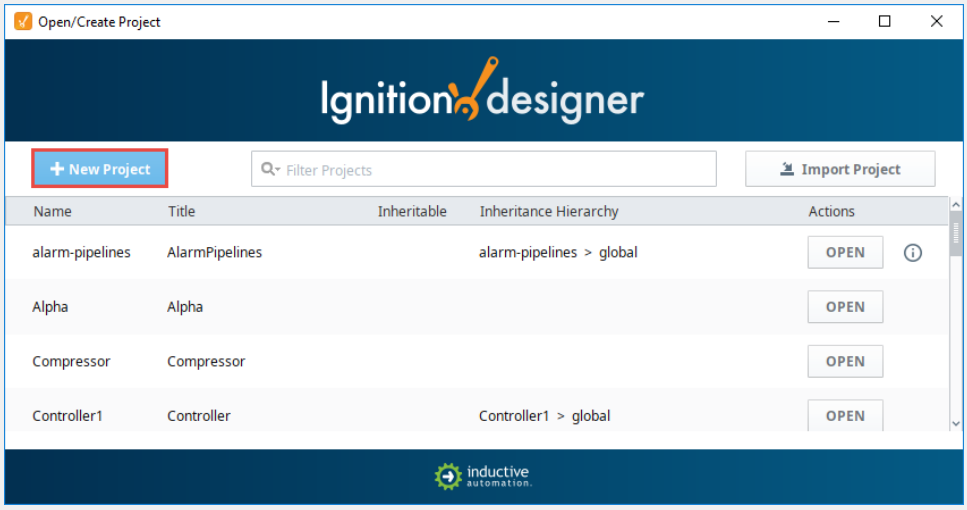
more, refer to Designer Launcher.



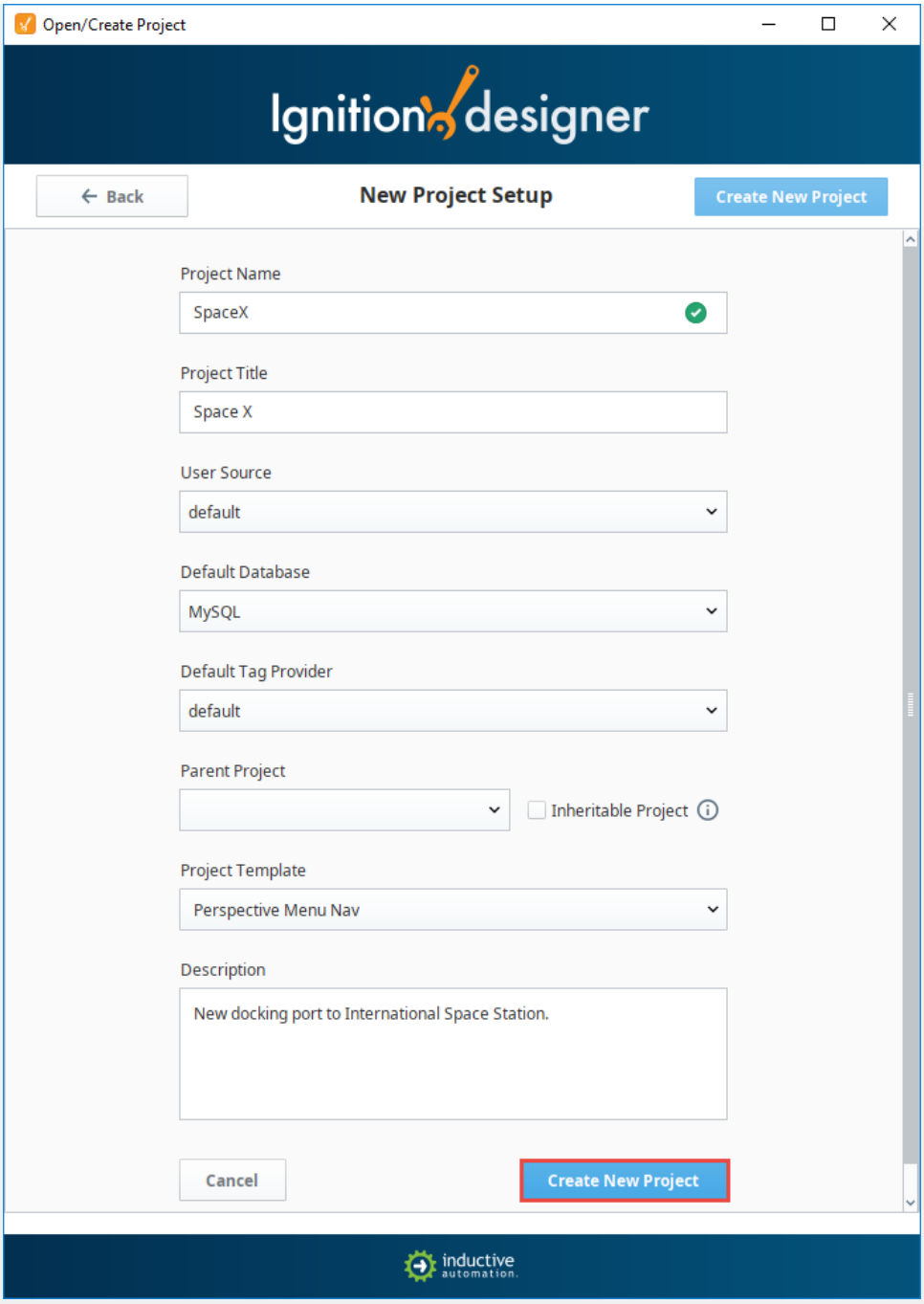
9. The Designer starts up and the login window is displayed. Enter the credentials for the administrator or user with full privileges for the Designer, and click **Login**. It should be the same as the login for the Gateway **Config** section.

> ⚠ Login credentials are set in the Config section of the Gateway Webpage, under Security > Users, Roles.

10. The **Open/Create Project** window is displayed giving you two options, to create a new project or open an existing one. Let's create a new project by clicking **+ New Project.**



11. This opens the **New Project Setup** window.



12. Enter the **Project Name** (required) and any other configuration settings you need for your project. Most settings are optional. Refer to the New Project Setup Properties Table below for a description of each property.

> ⚠ It is not advisable to change the Project Name after it's been created, instead, change the Title property in **Config > Projects** if you want to change how the project appears later.

New Project Setup Properties

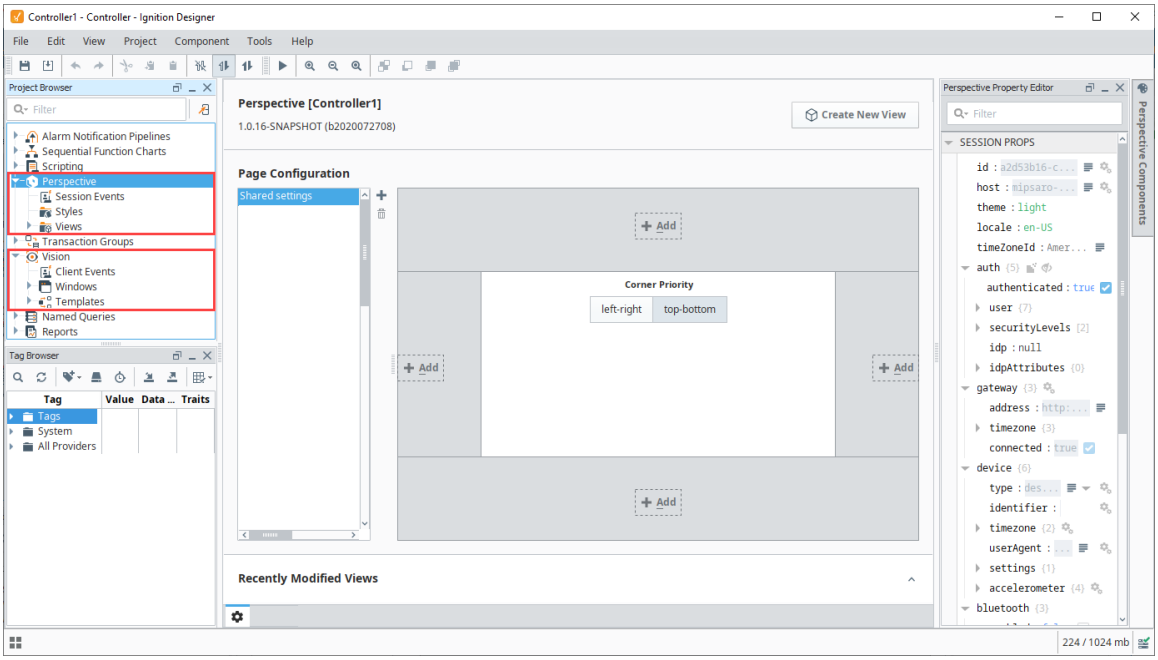
Property	Description
Project Name	The Project Name can only consist of alphanumeric characters and the '_' (underscore) character. Spaces and other special characters are not supported. Note: It is not advisable to change the Project Name after it's been created, instead, change the Title

	property in **Config > Projects** if you want to change how the project appears later.
Project Title	This is the name that will be displayed on the launch page of the Gateway and in the runtime Client or Session, (optional) There are no restrictions on using special characters or spaces. If you do not specify a title, the project name will be displayed on the Gateway launch page and in the runtime.
User Source	Controls access to the project. If this field is left blank, the **'default'** user source is set. For more information or to add users, see Security.
Default Database	Select a Default Database (optional). Any queries to the database will use this default database connection unless explicitly specified otherwise.
Default Tag Provider	Select a default Tag Provider (optional). Most installations will likely only have one provider, but in situations where there are more than one, this is the provider that will be used unless explicitly specified otherwise.
Parent Project	Each project may have a parent project, and will inherit all of the resources of that parent project, (optional). Click the dropdown list to see all the available options.
Project Template	Select a Project Template (optional). There are several pre-built project templates focused on navigation that support either Perspective or Vision: Perspective Menu Nav, Perspective Web Nav, Vision 2-Tier Tab Nav, Vision Tab Nav, and Vision Tree Nav. Click the dropdown to see all the available options.
Description	Enter a description of the project (optional). Note that once a project is created, this description can be viewed on the Open/Create Project screen when you hover over the Information icon.

13. Click on **Create New Project**.
14. When you open the Designer interface for the first time, the Perspective Page Configuration opens if you have the Perspective Module installed. If you don't have the Perspective Module installed, the Vision Designer window will open.

    You can begin designing in either Perspective or Vision from the Designer interface, as shown below. Expand the Perspective and Vision folders and you'll notice that the folder contents are dfferent. In Perspective, you add components to Views. In Vision, you add components to Windows. To learn how to add components and bring them to life, go to Quick Start - Perspective Session and Quick Start - Vision Client.

Learn more about the Designer, Perspective Designer Interface, and Vision Designer Interface.



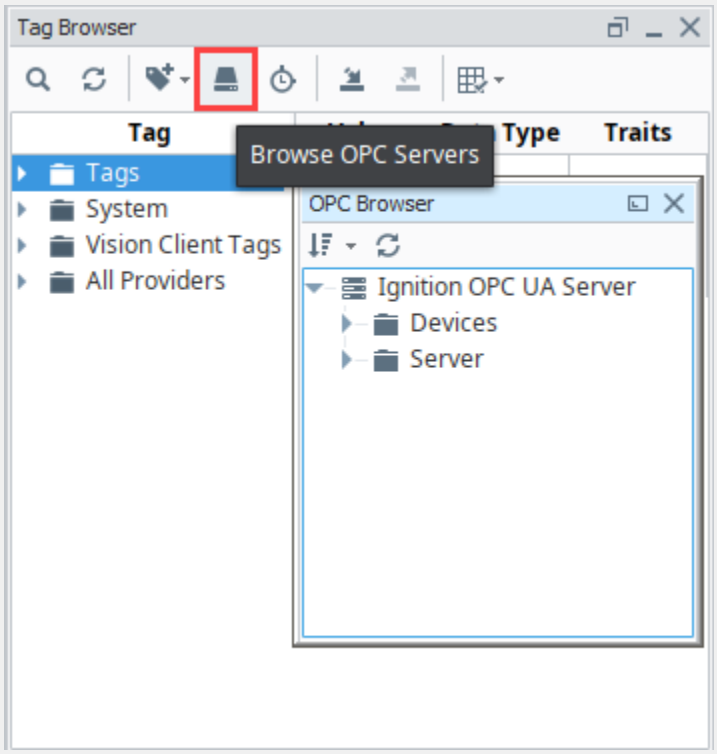
6. Create Tags

With the Designer open, a great first step is to create some Tags. You'll use these Tags for realtime status and control and to store history with the Tag Historian. You can use Drag and Drop to bring Tags into Ignition with the Designer. Once there, Tags are all configured in the **Tag Browser** panel.

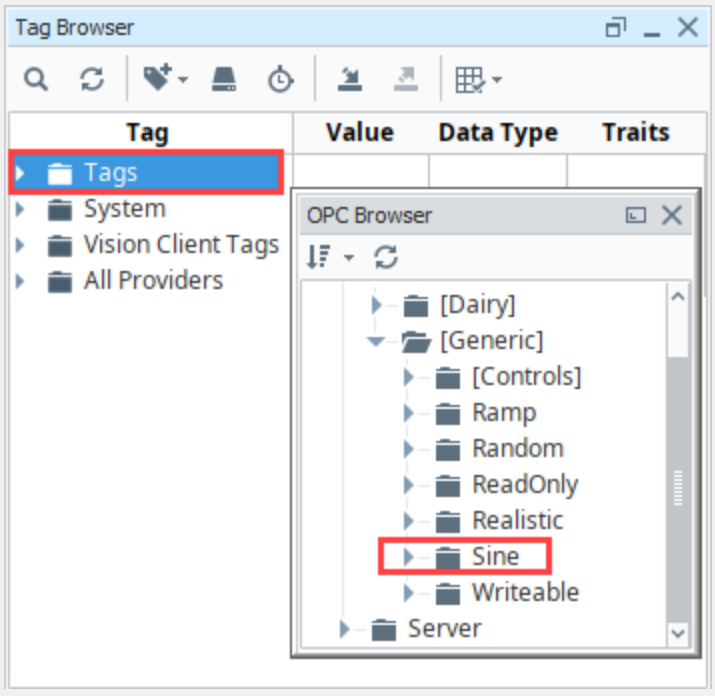
If you are connected to a device in Step 3, the easiest way to create some Tags is by dragging the Tags from the OPC Browser.

On the left side of the Designer you will see 3 panels, the **Project Browser**, the **Tag Browser**, and the **Property Editor**.

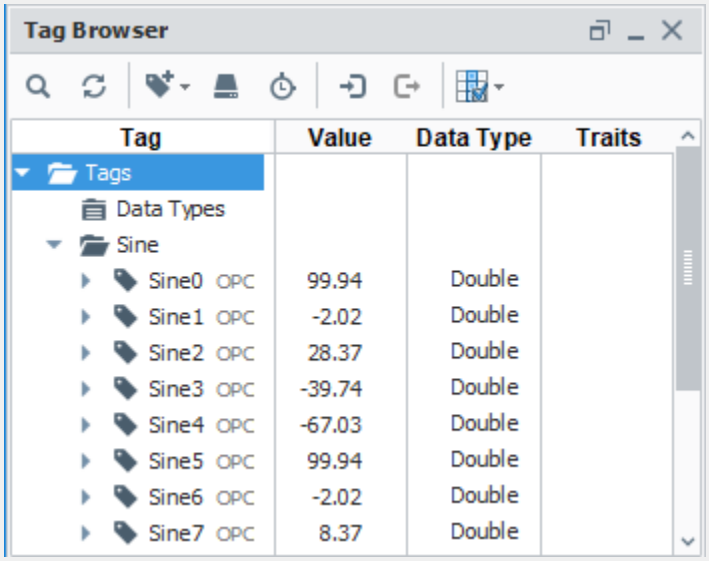
1. In the **Tag Browser**, select the **Tags** folder then click the **Browse OPC Servers** icon. The OPC Browser will be displayed and you can browse all of your OPC connections.



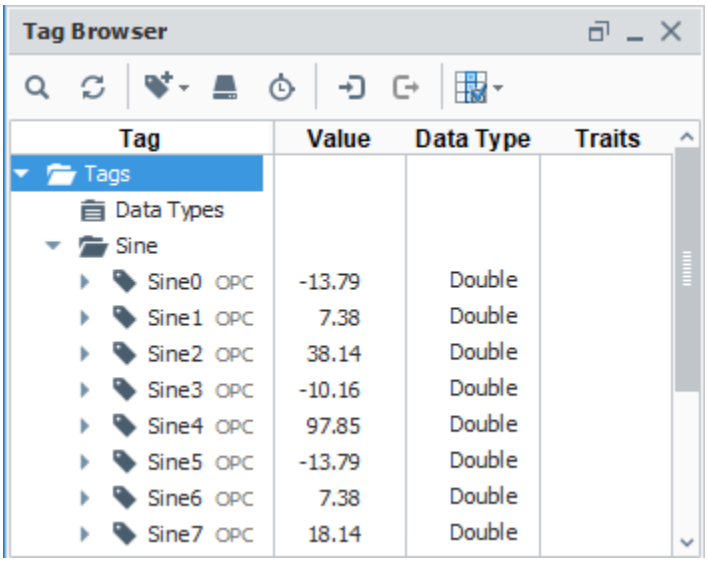
2. By default, you have a connection to the internal **Ignition OPC-UA Server.** Expand the Devices folder to find your list of devices, and expand your device folder to see some Tags inside. From the **OPC Browser**, highlight the Tags you want (for example, as in the image below, the entire **Sine** folder is selected) and drag it to the **Tags** folder in the **Tag Browser**. You can drag individual Tags or whole folders. When you drag folders, Ignition keeps the same hierarchy as the PLC.



3. That's it! You now have some Tags. Expand the **Tags** folder and the new **Sine** folder to see what was added. You can see their values come in and start updating automatically. By default, they update at a 1 second rate.



Learn more about Tags!

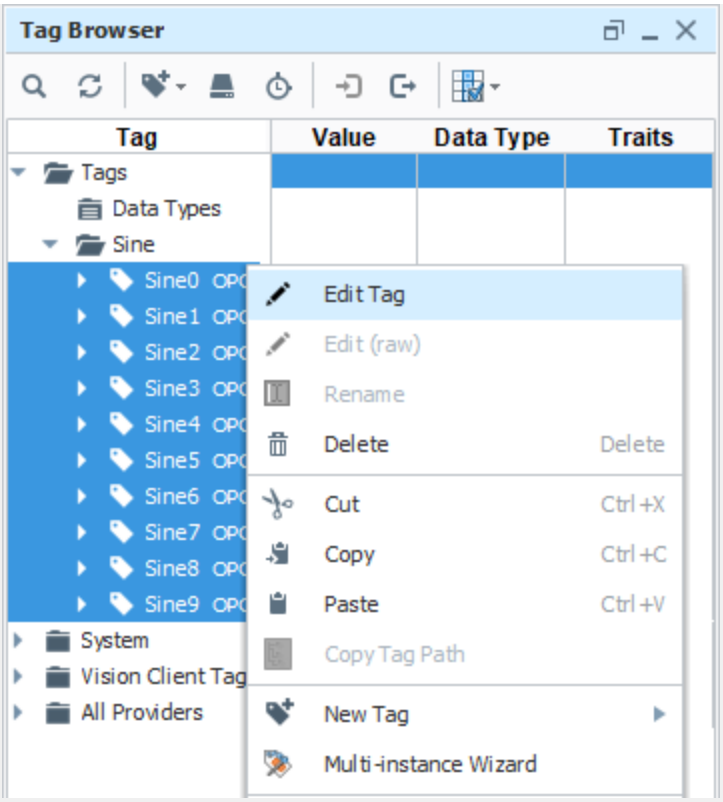


7. Add History to Tags

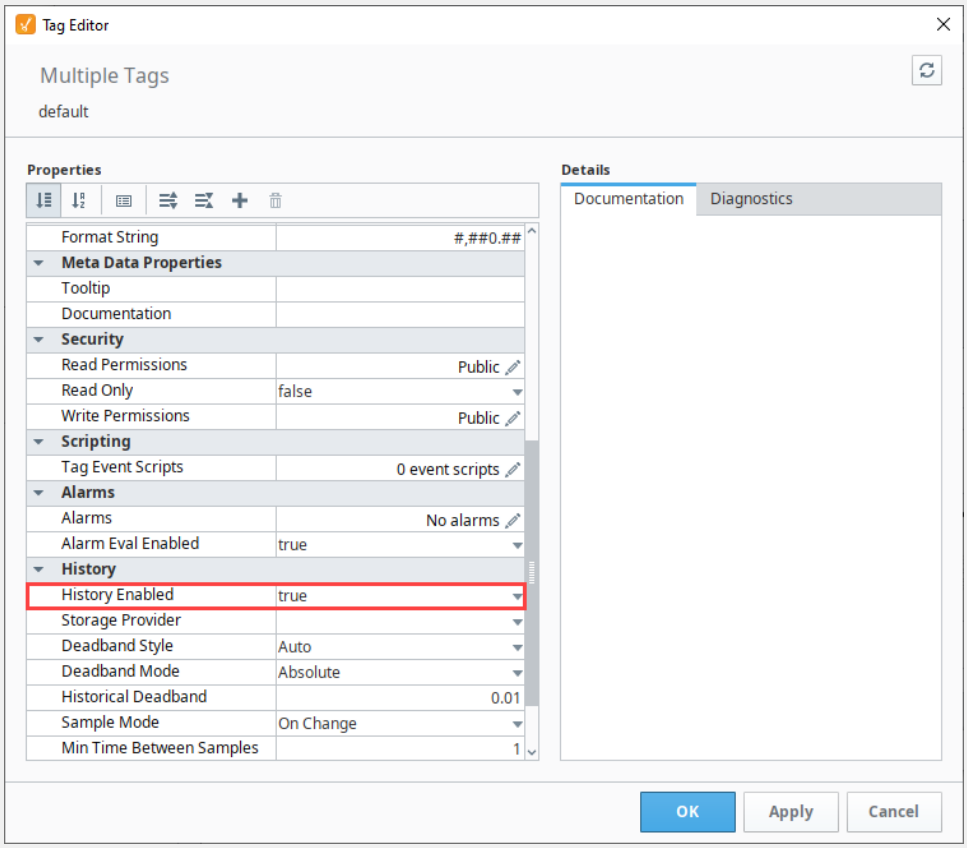
Storing history for your Tags is simple, and once that's done we can display it on screen in a variety of ways. If you are connected to a database in Step 4, you are ready to add history to some of our Sine Tags.

1. In the Designer, go to the **Tag Browser** and select one or more Tags.
   For example, select all the **Sine** Tags in the Sine folder by holding the **Ctrl** key and clicking on several Tags.

2. Right-click on the selected Tags, and then select **Edit tag(s)**.
   The Tag Editor window is displayed. Here, you can edit the Tag and change the name, data type, scaling options, metadata, permissions, history, and alarming.



3. Scroll down to **History** on the left side of **Tag Editor**, and set **History Enabled** to **'true'**.

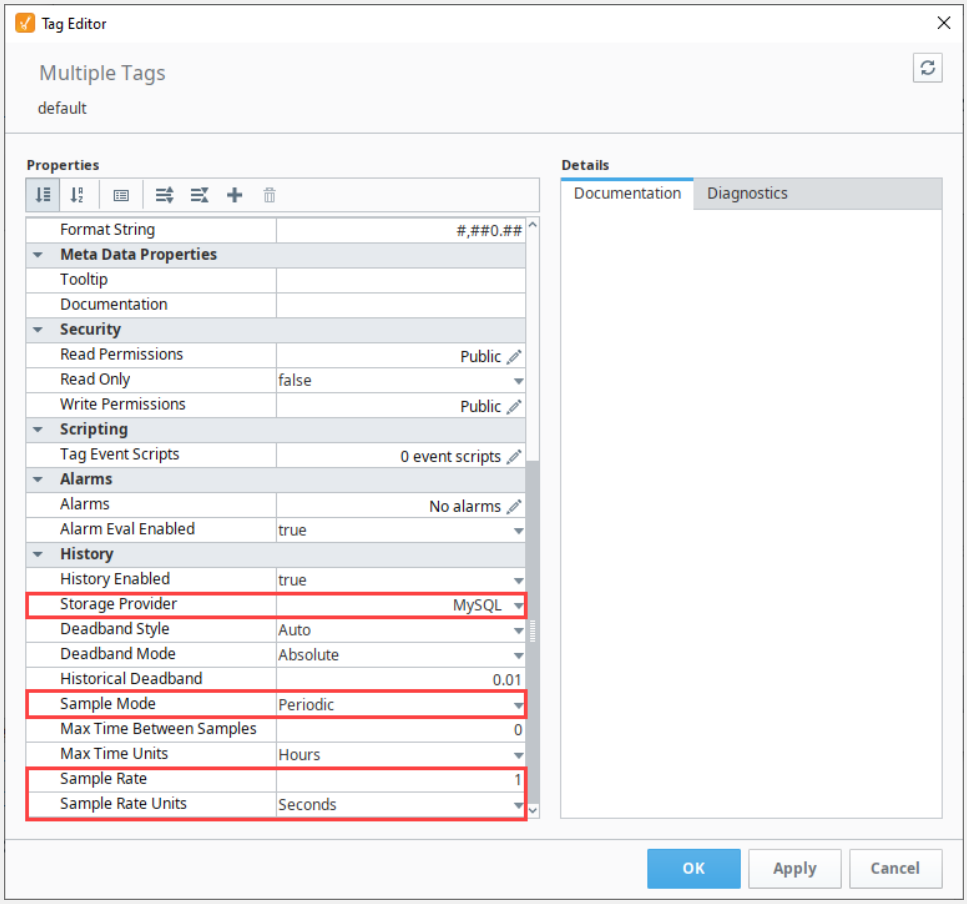


4. Choose a database from the **Storage Provider** dropdown, like the one we set up earlier.

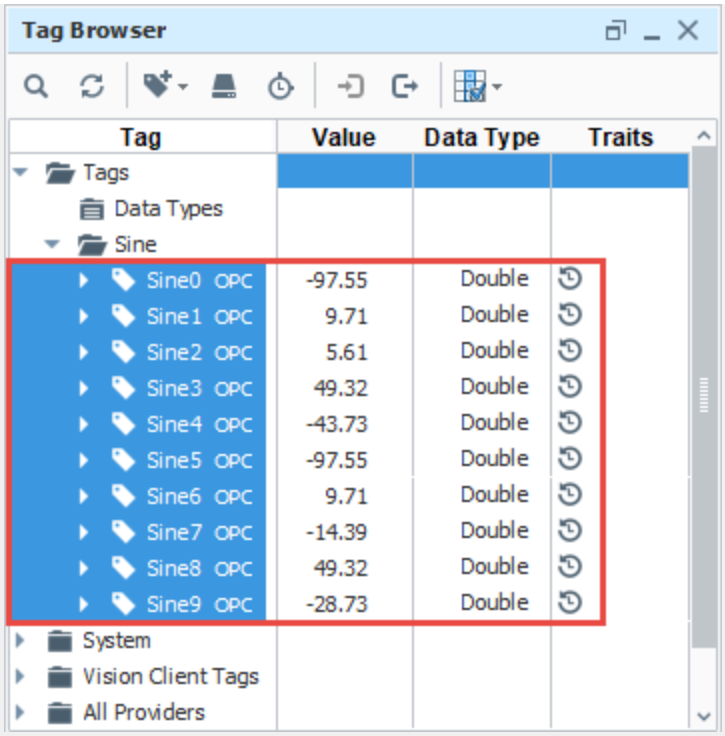
5. To set the rate that data is logged:
    a. Change the **Sample Mode** to **Periodic.**

b. Set the **Sample Rate** to **1.** (Data will be logged every **1** second).
c. Verify that the **Sample Rate Units** is set to **Seconds** (which is the default).

6. Click **OK**.

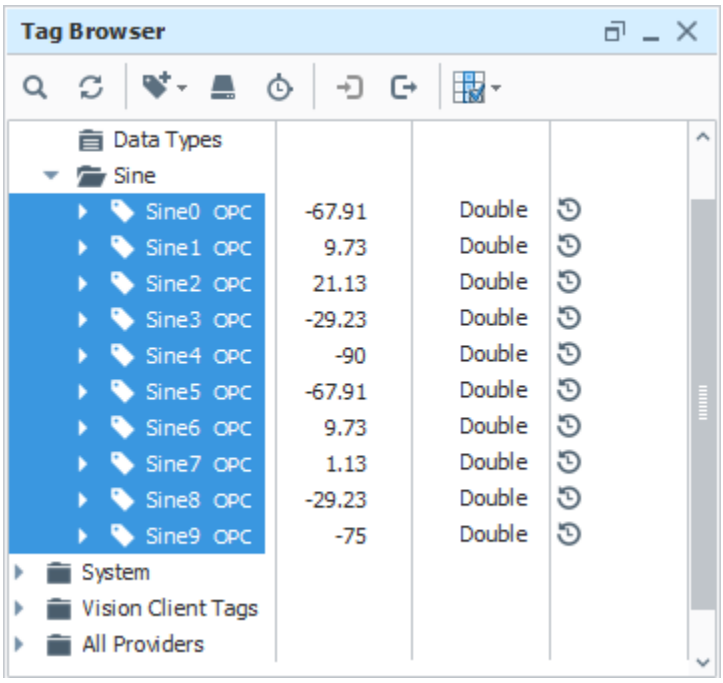


7. Now look in the **Tag Browser**, to the right of each **Sine** Tag that is storing history, an icon ( ) appears letting you know it is setup.



If you were to look in your database, you can see all the tables and data Ignition created for you.

Learn more about Tag Historian!



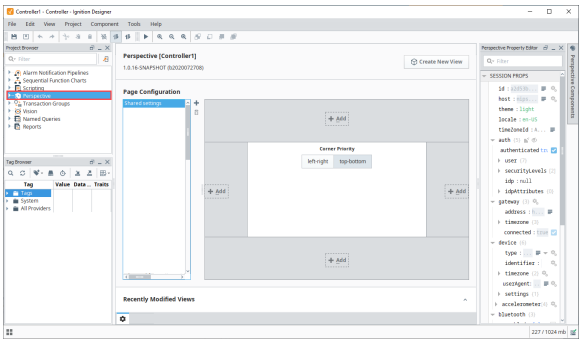
8. Adding Components

Building your industrial application starts by adding components. Whether you're using the Perspective Module or the Vision Module, adding components is easy, but they each have their own visualization system. It's for this reason, the Quick Start Guide dedicated separate Perspective and Vision pages for how to add components and configure live values. Click the link below to navigate to the respective module's page.

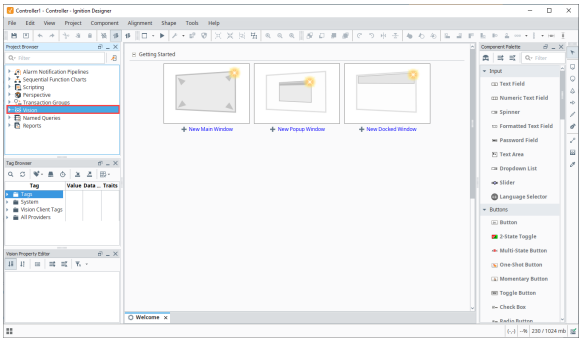
- Quick Start - Perspective Session
- Quick Start - Vision Client

When you open the Designer interface for the first time, the Perspective Page Configuration opens assuming you have the Perspective Module installed, as shown in the Perspective Session image below. If you don't have the Perspective Module installed, the Vision Designer window will open.

Creating a Perspective Session



Creating a Vision Client



Related Topics ...

- Gateway Backup and Restore
- Licensing and Activation
- Alarming
- Security
- Localization and Languages

In This Section ...

Quick Start - Perspective Session

Before we talk about how to add components to Perspective Views, it's important to first understand all the objects that comprise a Perspective Session. It will help you determine your design strategy within a browser-based design environment and how components inside a View will behave before you ever drag a component into a View.

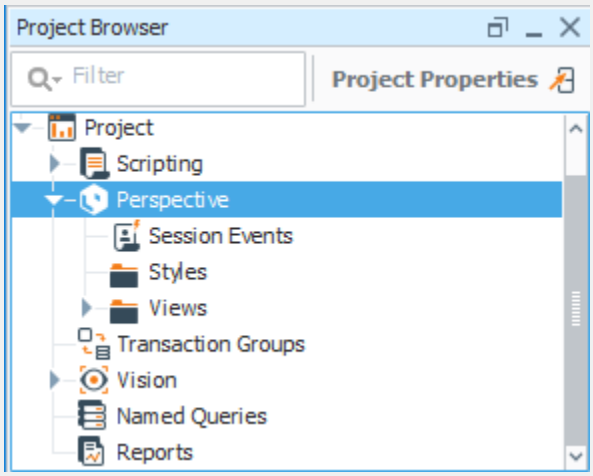
With that in mind, we recommend reading the Perspective Sessions section of the manual to fully understand how all the objects that make up a Perspective Session work together to create dynamic pages. You may also want to peruse the page on new navigation strategies that are offered in Perspective. This Quick Start guide can be completed without reading any other topics.

On this page we demonstrate how to add components to a View and bring them to life by showing live values, displaying Tag History data on a chart, configuring pages for navigating within a session, and launching a Perspective session.

1. Creating Views and Adding Components

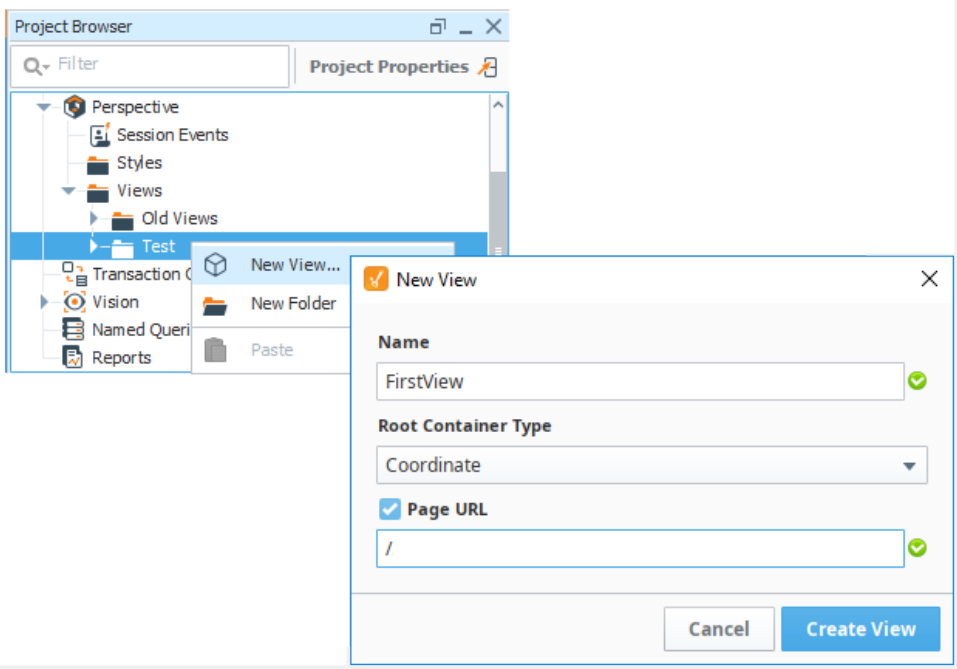
Views are the primary unit of design and can be displayed across the top, bottom or sides of a page, or even a whole page in a Perspective Session. You can drag and drop components into a View, and make them display text, show Tag Values, and display Tag History data.

1. With the Perspective Designer open, go to the **Project Browser**, and expand **Perspective.**

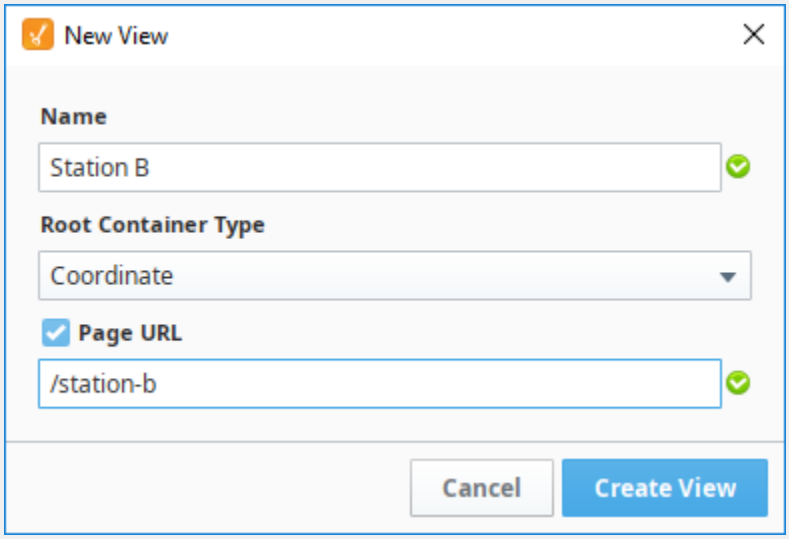


2. Before you create a view, you have the option to create folders for your views which is a good way to organize your views. In this example, a Test folder was created to store views.
    a. Click on the **Views** folder and right click on **New View...** to create a View. When you enter the **View Name**, you'll notice that the **Page URL** will also be entered at the same time, unless it is your very first view in your project, then the page will be mounted to the Page URL **"/"** or **"root"** as shown in the image below. **Note:** The **Page URL** will only get created if you check the **Page URL** checkbox.
    b. Select the **Layout** from the dropdown. Each view contains one root level container which proivdes a way of laying out and organizing child components. This example uses the **Coordinate** layout type which is the simplest type of view, but you can also choose from Tab, Breakpoint, or Flex containers.
    c. Make sure your **Page URL** is checked if you want the Page URL to be created at the same time your view is created.

> **Match the URL with the Folder Structure**
>
> Perspective will match the **Page URL** with the View's folder structure upon creation. In the example below, the FirstView was created in the Test folder. Perspective knows to match the URL to the View.

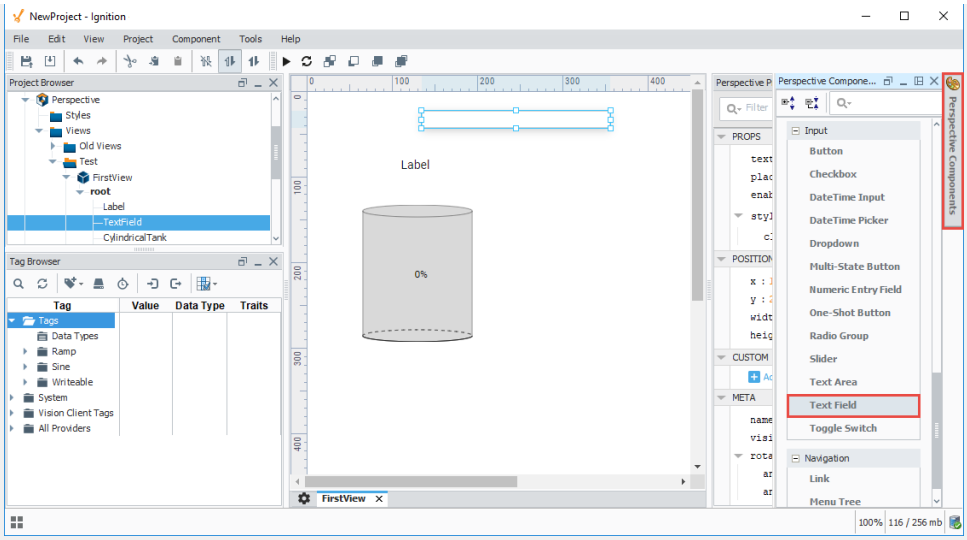


Note: If your View Name contains a space, such as "Station B", the Page URL will replace the space with a hyphen. Notice how the Page URL automatically gets filled in when the View Name is typed in.

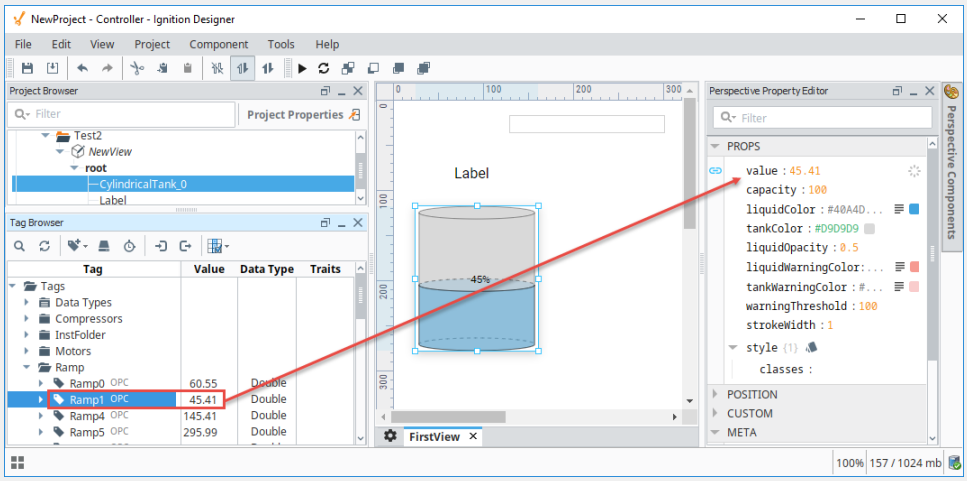


d. Click **Create View** when you have everything filled in.

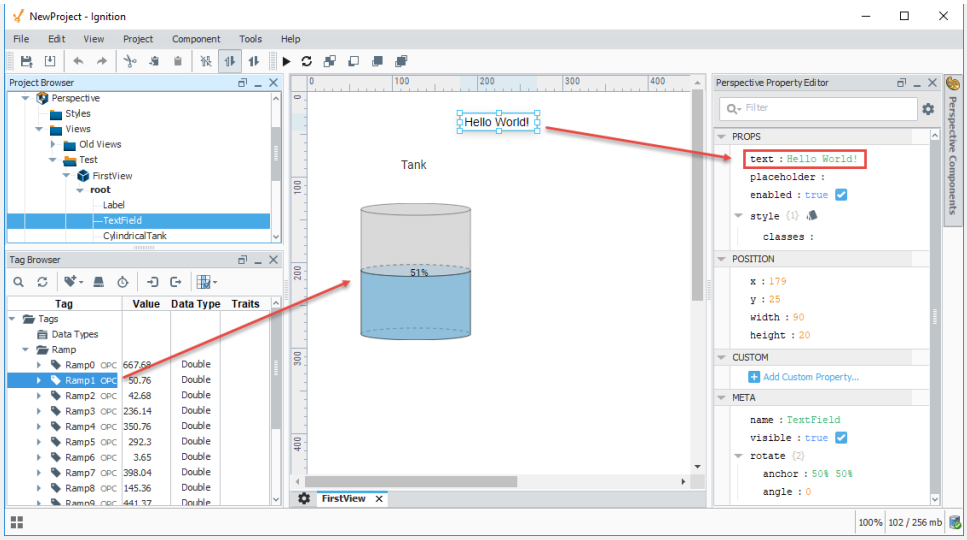
3. The Designer will refresh and now you're ready to begin adding components into your view. On the right side of the Designer, click on the grayed out **Perspective Components** banner to open the **Components Palette**. Drag the **Text Field**, **Label**, and **Cylindrical Tank** components into the View.



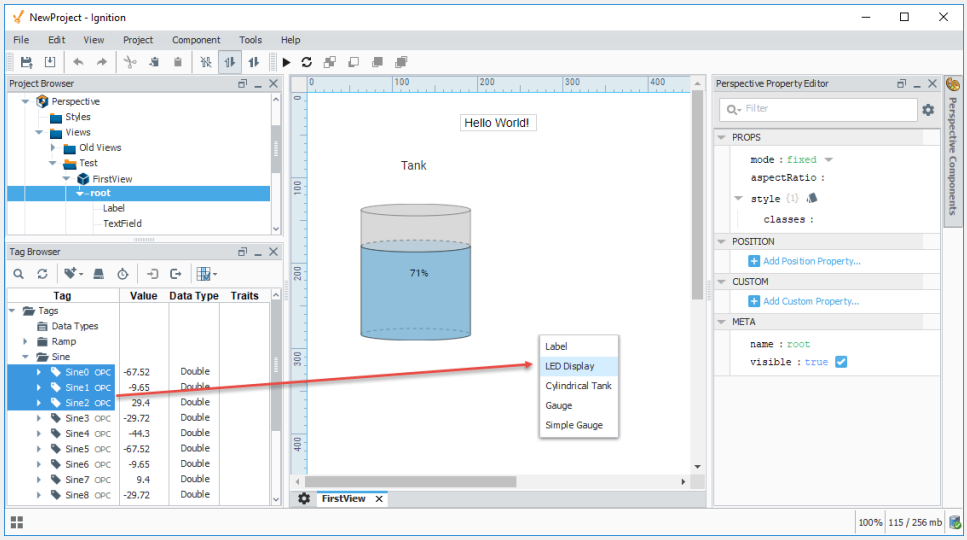
4. To add Tag values to each of the components, click on the component and enter the value in the **Perspective Property Editor**:
    a. **Cylindrical Tank** - Drag a Tag from your Tag Browser on to the **value** property. This example uses a generic simulator Tag, **Ramp1.** (**Note:** This requires that you are connected to the Generic Simulator. Refer to the Quick Start Guide for Connecting to a Device).
    b. **Label** - Enter **'Tank'** next to the **text** property.
    c. **Text Field** - Enter **'Hello World!'** in the **text** property. While the component is selected you can also click and drag the arrow handles to resize them.



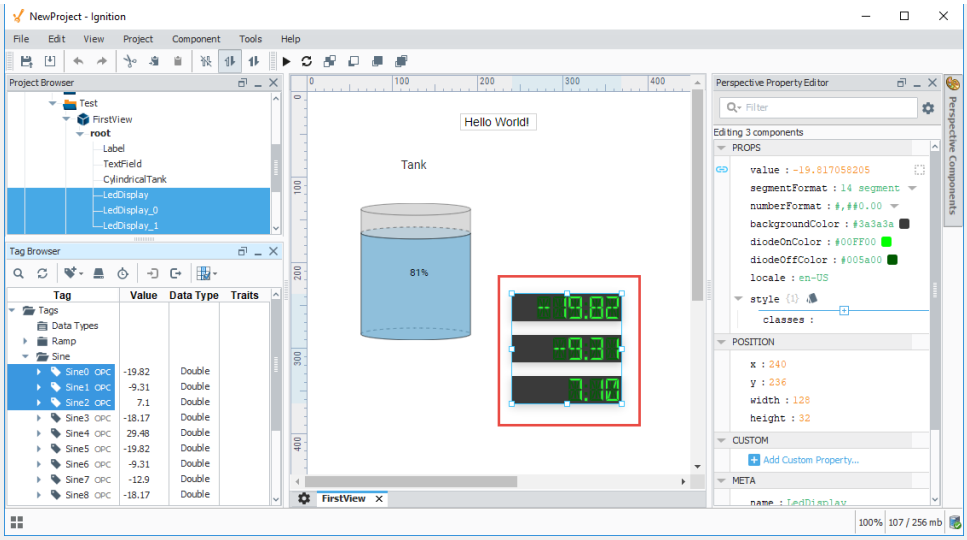
Once all the values are added to the components, they will look like the following image in your View.



5. Another way and the simplest way to add a Tag to component is to go into the **Tag Browser,** select the Tag you want and drag it into your view. A popup will appear giving you several selections for components to use. In this example, select a few Tags and drag them into the view. You will see a popup asking you what type of component you want to use to show these Tag values. Select **LED Display**.

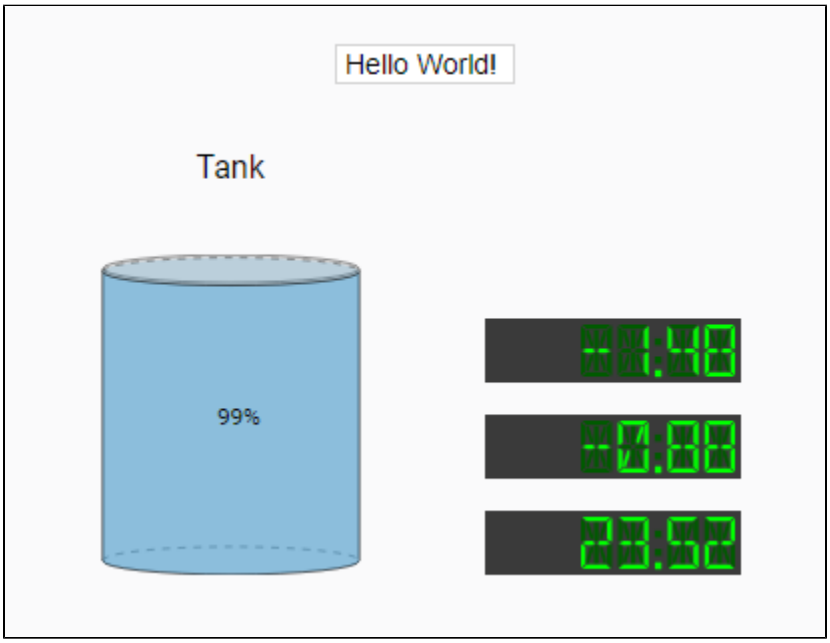


Three LED Displays will be created displaying live values from the three Tags you selected.



6. Don't forget to **Save** your project by selecting **File > Save** in the menubar.

To learn more about bindings, go to the Bindings in Perspective page.

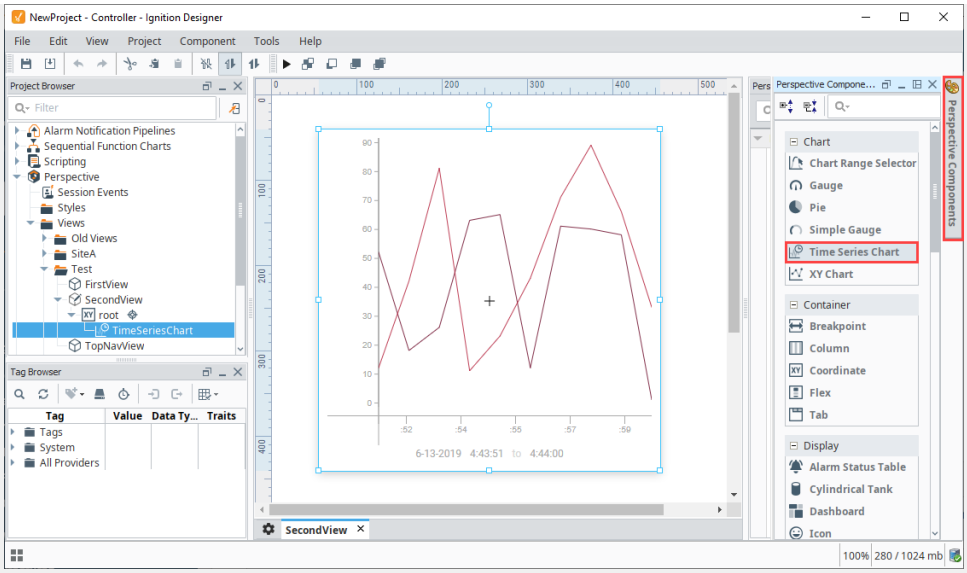


2. Showing Tag History in a Time Series Chart

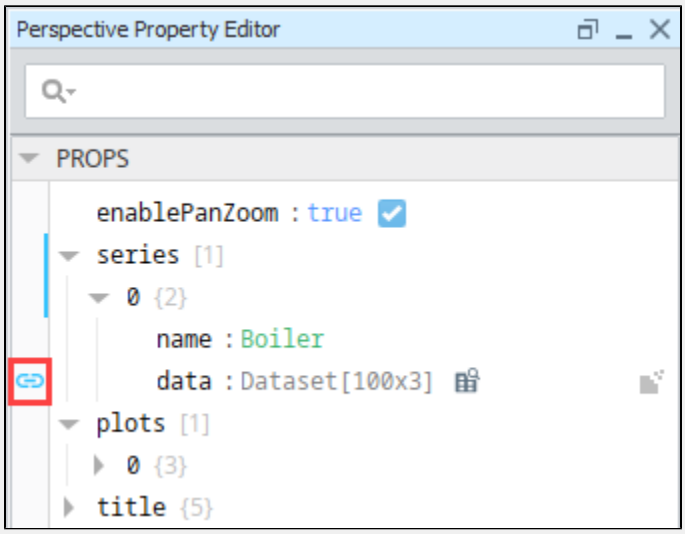
Displaying Tag History data using an Time Series Chart. This requires that you have Tag History already set up. Refer to the Quick Start Guide, Add History to Tags.

Let's create a second view (i.e., **SecondView**), add an Time Series Chart component to show history on the values of the LED Displays.

1. Right click on the Views Test folder and create a new view (i.e.,**SecondView**).
2. Once you have your second view created, drag an **Time Series Chart** component on to your view.

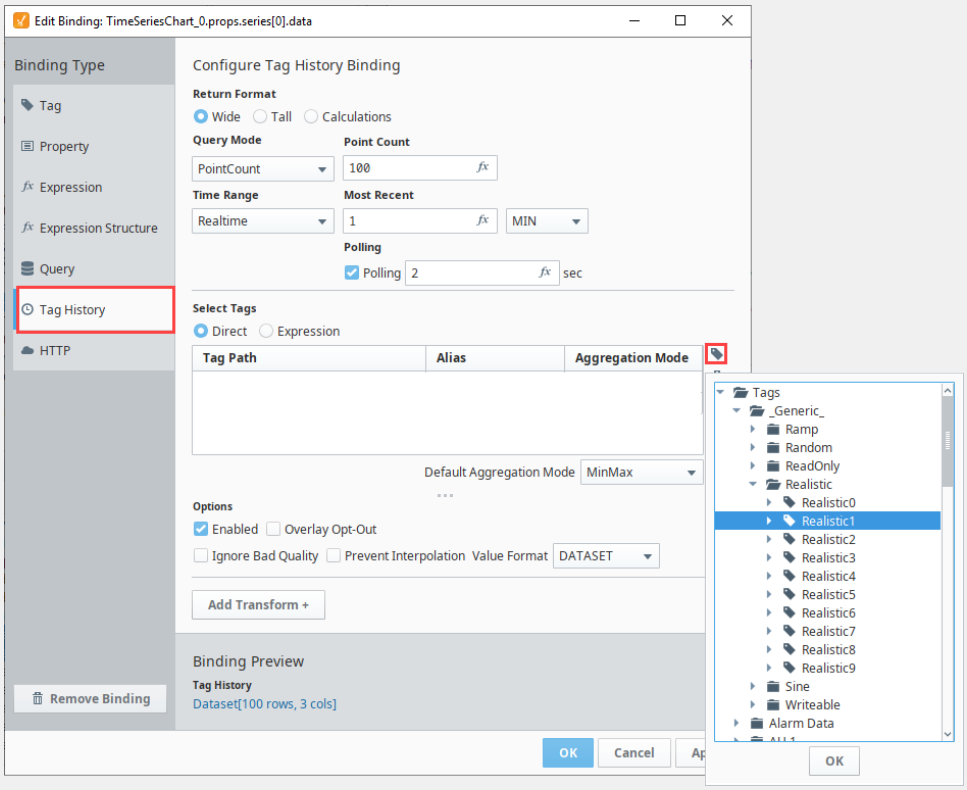


3. With the Time Series Chart selected, click on the chain link **Binding icon** under the **series > 0 > data** property.



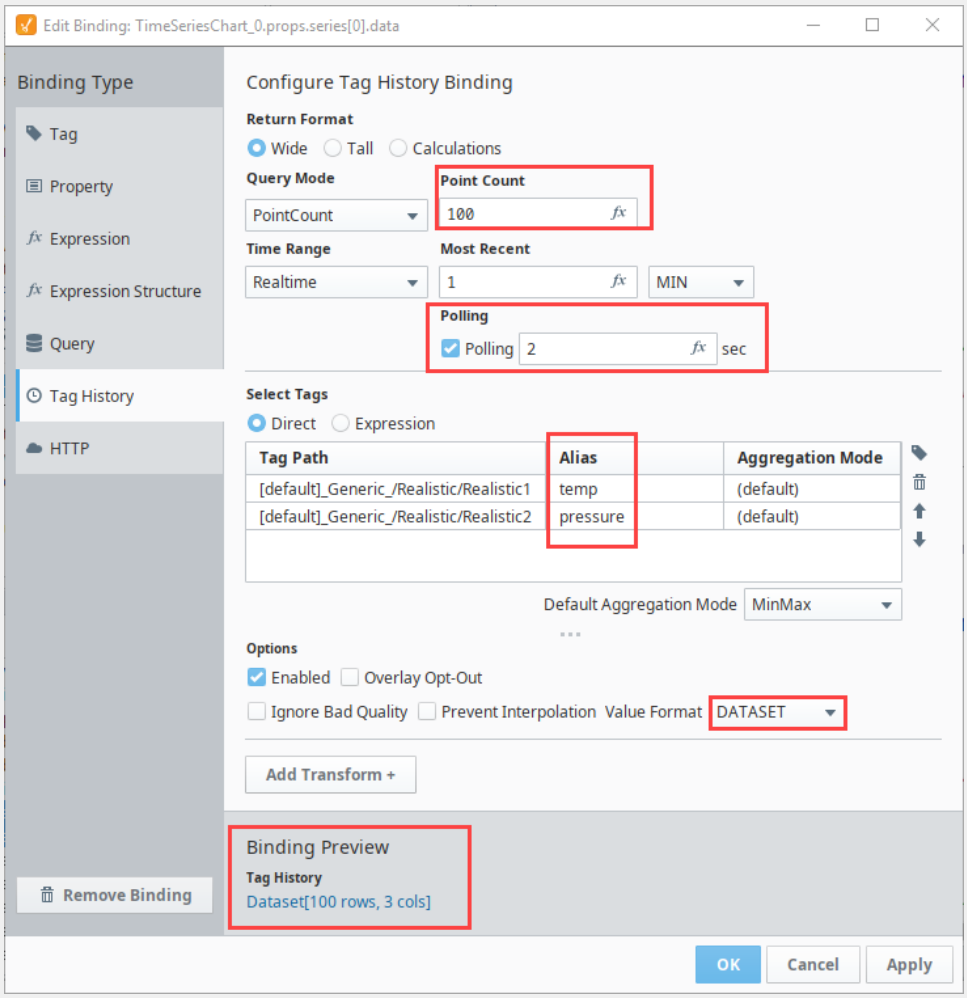
4. Select the **Tag History** binding type.

5. From the **Select Tags** section, click on the Tag Browse icon on the far right. Browse to select your Tag path. You can select multiple Tags if you want. We selected **Realistic1** and **Realistic2** from the Generic Simulator device.



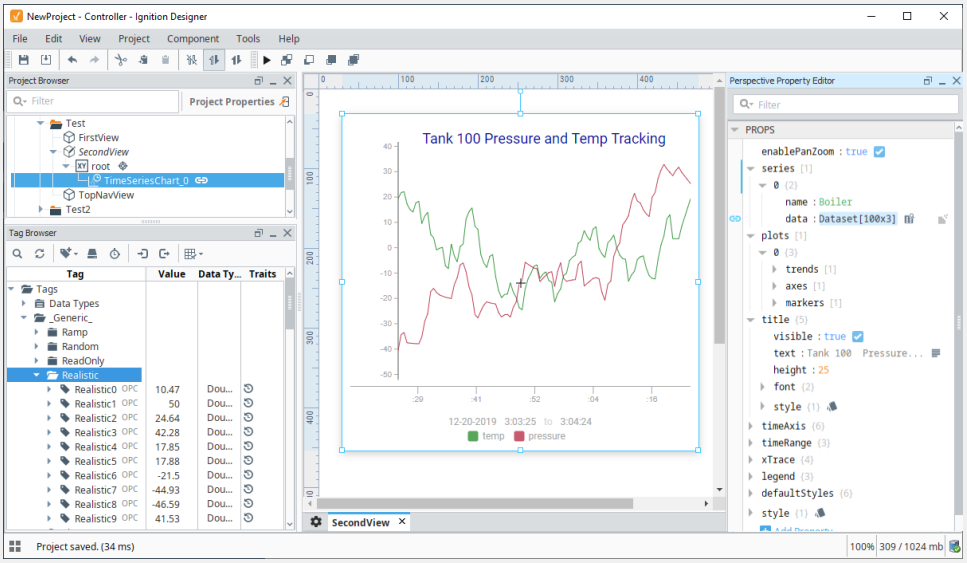
6. Now let's configure the following Tag History binding settings:
    a. Double click in the Alias column for each Tag to add an Alias
        i. Add the **Alias** for **Realistic1** as **temp.**
        ii. Add the **Alias** for **Realistic2** as **pressure.**
    b. Set the **Point** count to **100.**
    c. Set **Polling** to **2.**
    d. The Tag History binding returns a **Dataset**. You will see the format type in the Binding Preview in the lower left.

7. Click **OK**.

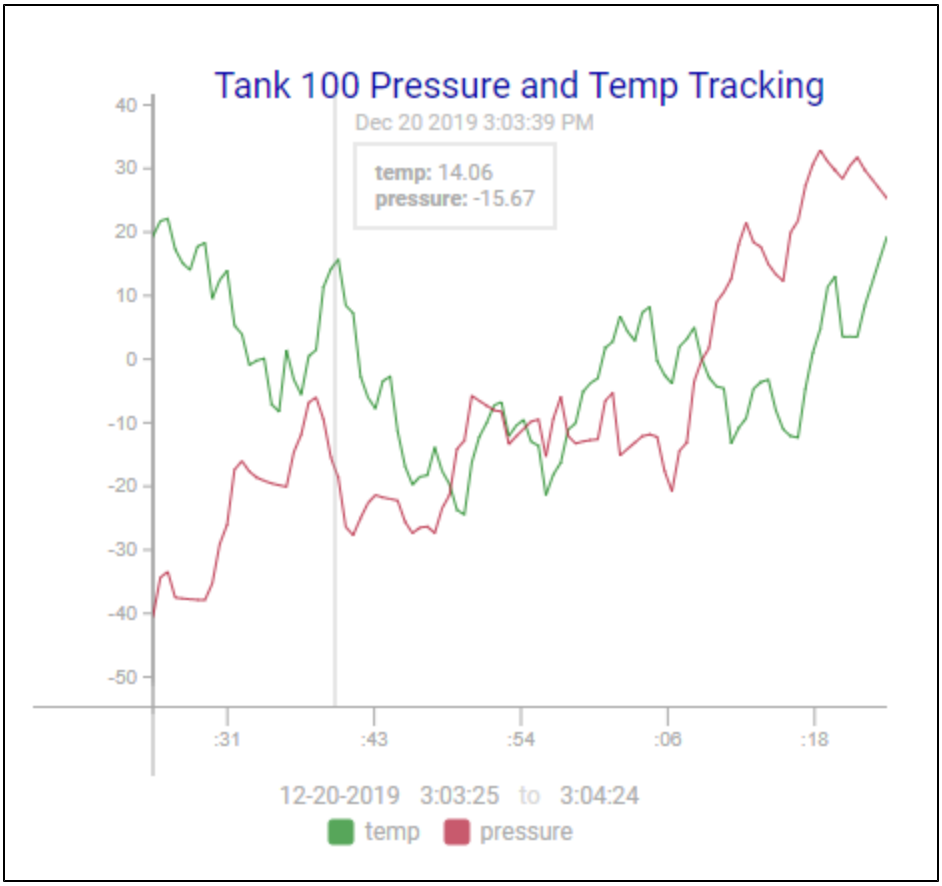


8. Now you have Tag History data in your Time Series Chart.
9. To add a title, scroll ot the **Title** object in the PROPS section and set the following values:

    a. **title.visible** - Set to **true**.
    b. **title.text** - Enter a title: **Tank 100 Pressure and Temp Tracking**.



10. **Save and Publish** your project.

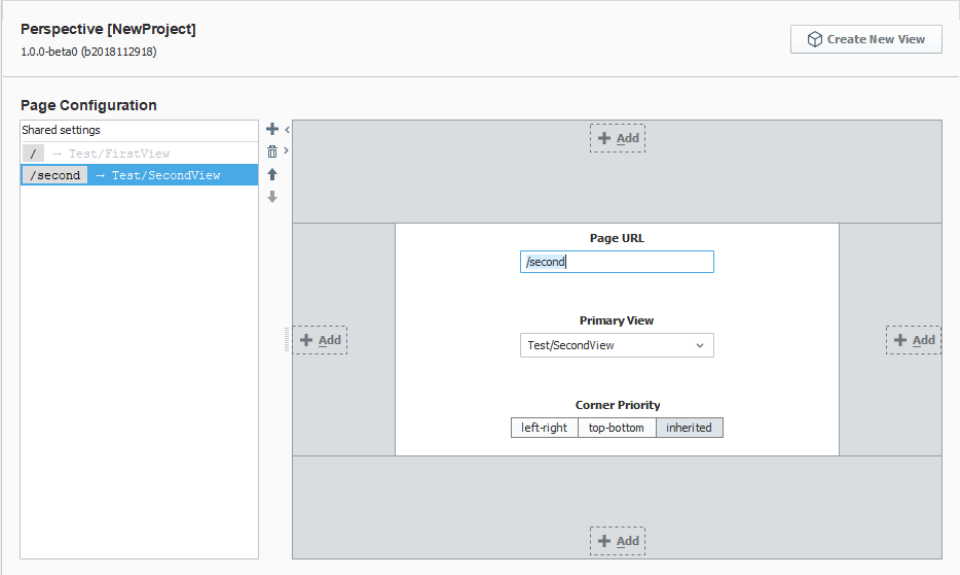


3. Navigating in a Perspective Session

One of the most important aspects when launching a Perspective Session is having a solid navigation design so the user knows where they are, where they've been, and where they are going. How you configure your page settings dictate how the browser navigates and displays each page of your Perspective Session.

A Page is the main navigational element in a Perspective Session and represents a collection of views that are displayed in a browser window. When we created the two views in the Creating Views and Adding Component section above, we also configured the Page URL associated with each new view. You have the option of configuring the Page URL in a separate step if you don't want to do it at the time you create your new view. This is addessed in Pages in Perspective.

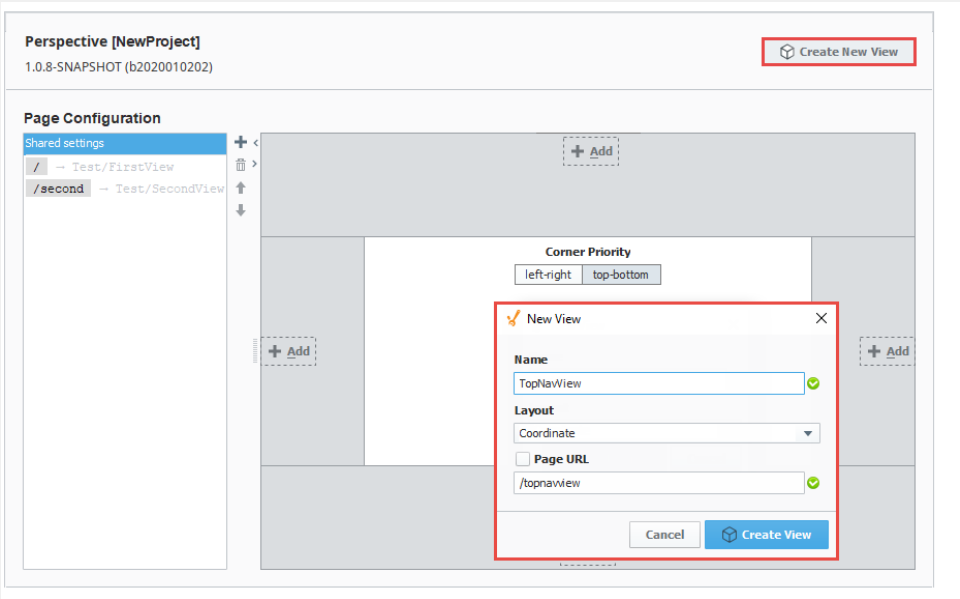
1. Open the **Page Configuration** by clicking the Tool icon ( ) tab at the bottom left of your Designer workspace. You'll notice that the two views (i.e., FirstView and SecondView) also have their Page URLs created. This was done at the same time we created each view.



> **First View mounted to the root**
>
> You'll notice the first View created in a project is automatically mounted to the root. The FirstView defaulted to "/" which denotes a "root" page.

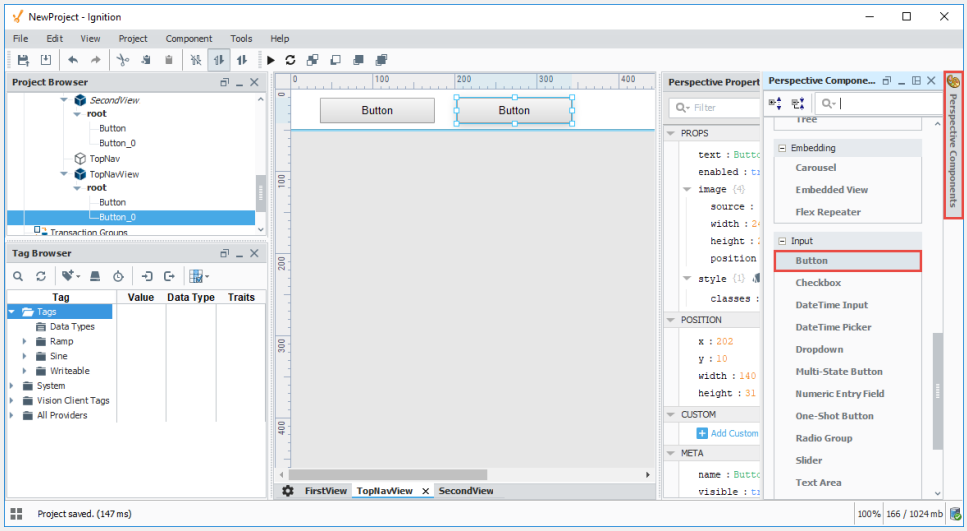
2. Now, let's create a navigation view (i.e., **TopNavView**) to be shared by all pages in our project. This page will contain components that will link to different pages allowing users to navigate from one from one page to another in a Perspective Session.

    a. Select the **Shared Settings** option under Page Configuration.
    b. At the top of the Page Configuration, click on **Create New View** (i.e., **TopNavView**).
    c. Use the default **Layout** (i.e., **Coordinate**).
    d. *Do not* check the **Page URL** checkbox to create the URL page. This isn't necessary because it is a shared setting.
    e. Click **Create View**.

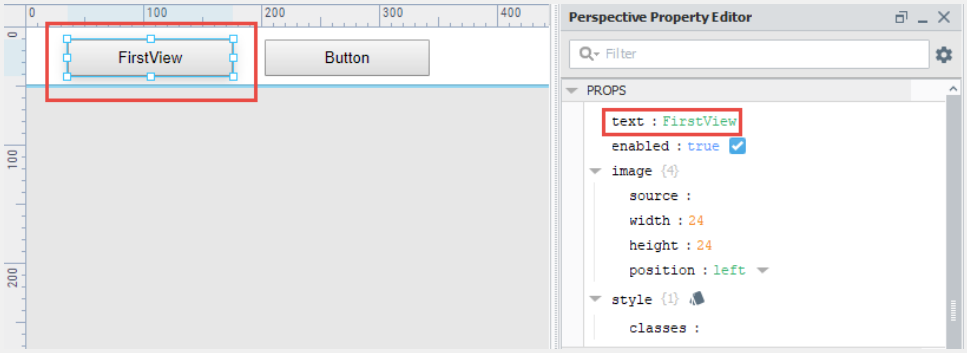


2. Go to the **Project Browser** and select your new view (i.e, **TopNavView**). This will show the handles for resizing the view. Resize the root container so the view looks like a banner or bar at the top of the workspace.

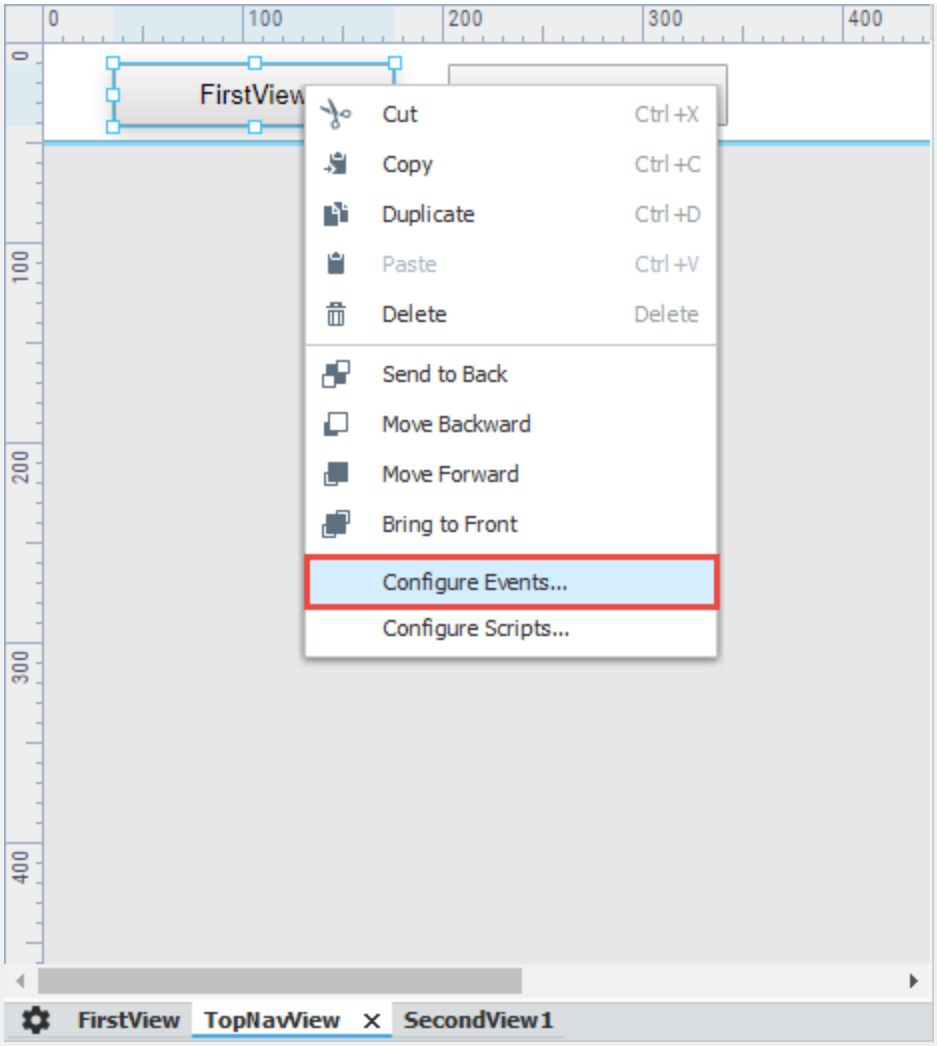
3. Add two Button components to navigate between pages.
You can drag two **Button** components from the Perspective Component Palette or drag one in and duplicate (Ctrl D) the first Button.



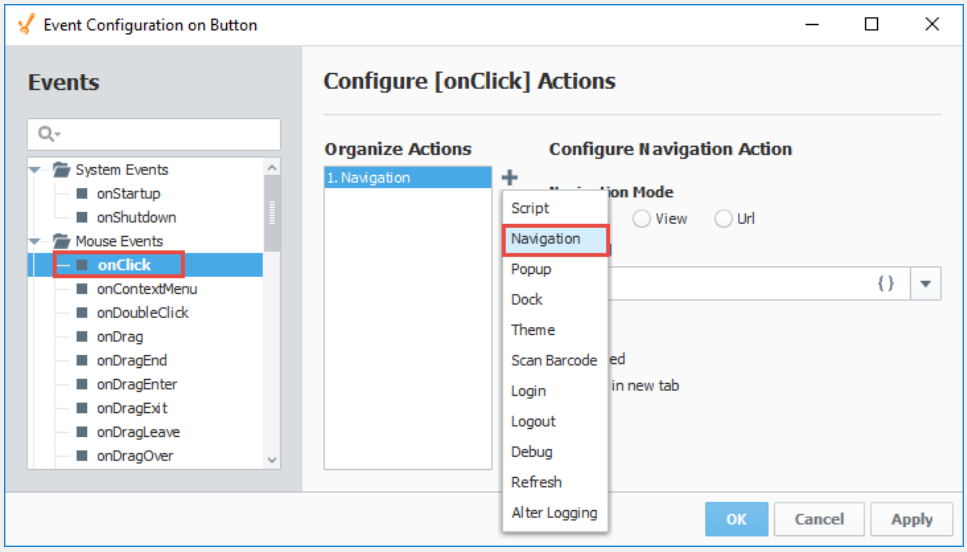
4. Assign each Button component unique text. Select the first Button component, and change the **'text'** property to **'First View'** in the **Perspective Property Editor**.



With the **FirstView** button still selected, right click and choose **Configure Events...** (You can also click **Components** o n the top menubar and choose **Configure Events...**).

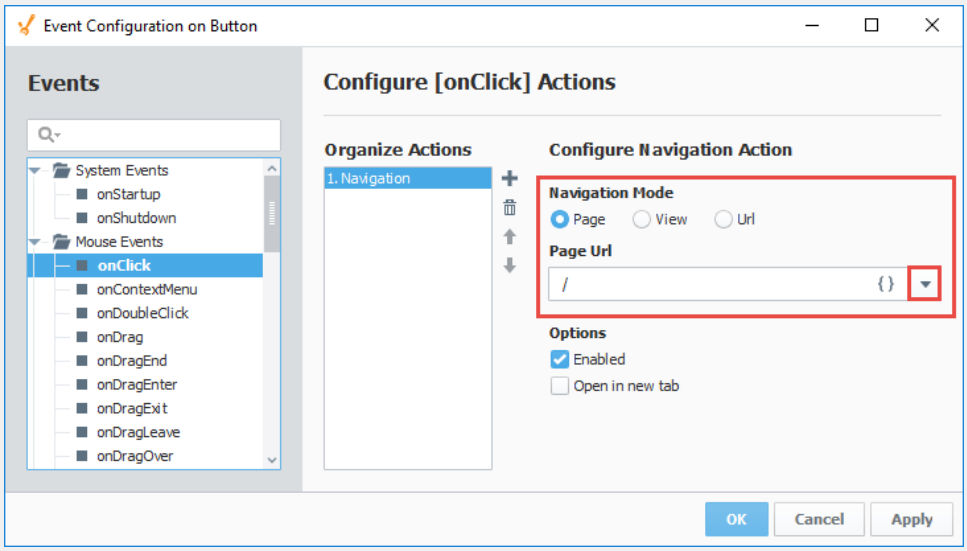


5. This opens the **Event Configuration on Button** window.
   a. **Expand Mouse Events**, and select **onClick**.
   b. **Under Organize Actions**, click the (➕) icon and select **Navigation** from the dropdown.



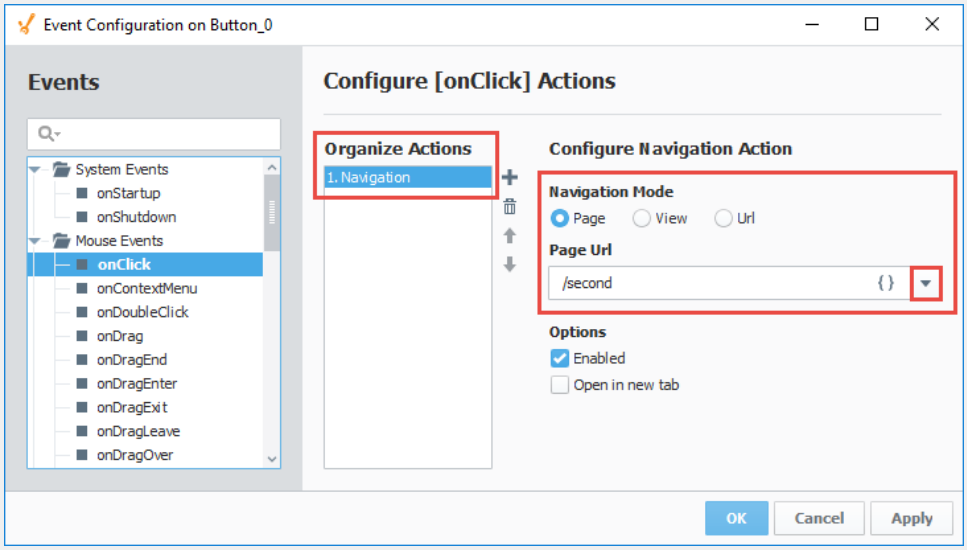
   c. Under **Navigation Mode**, click **Page**, and select the **"/"** from the dropdown which is the **Page Url** for the **First View**.

d. Click **OK**.



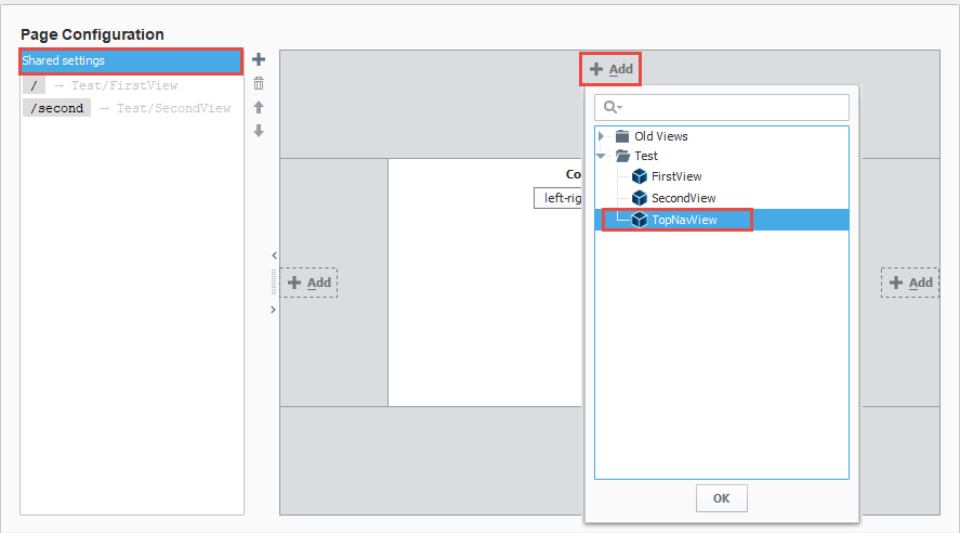
6. Now let's repeat Steps 3 and 4 for the second Button component:
   a. Select your **second** Button component, and change the **'text'** property to **'SecondView'** in the **Perspective Property Editor**.
   b. With the **SecondView** button still selected, right click and choose **Configure Events...**

7. The **Event Configuration on Button_0** window will open.
   a. **Expand Mouse Events**, and select **onClick**.
   b. **Under Organize Actions**, click the ( ) icon and select **Navigation** from the dropdown.
   c. Under **Navigation Mode**, click **Page**, and select the **"/second"** from the dropdown which is the **Page Url** for the **SecondView**.
   d. Click **OK**.

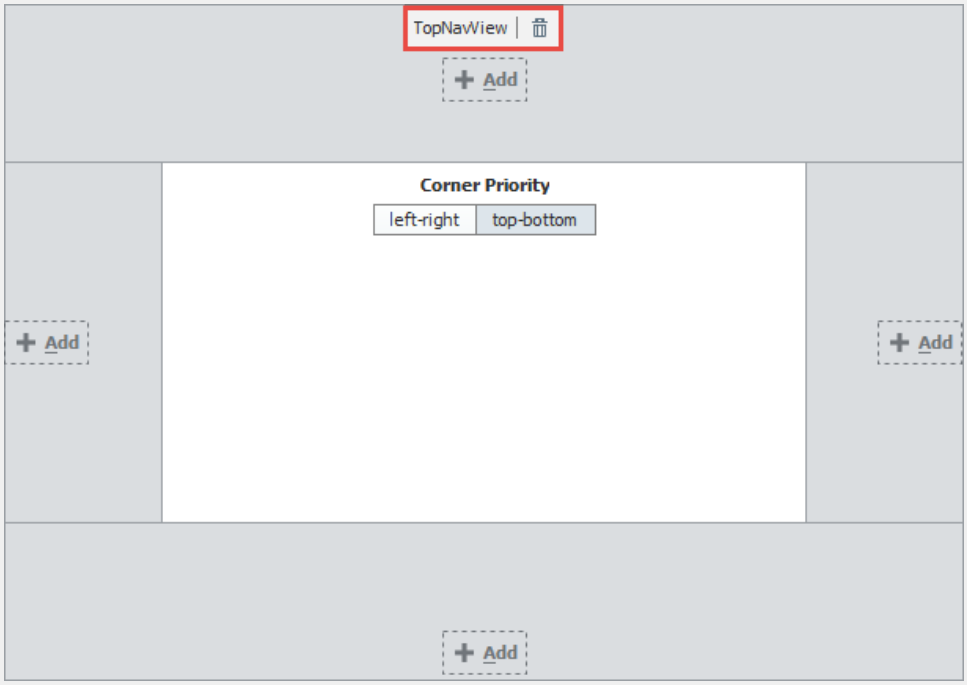


8. Lastly, set the navigation view (i.e., **TopNavView**) as a docked view so it appears on the top of every page in your project.
   a. Click on the Tool icon ( ) at the bottom of your Designer workspace to open the **Page Configuration**.
   b. Select **Shared Settings** under Page Configuration, click on the **Add** button at the **top** of the Page Layout. Select your navigation view (i.e., **TopNavView**) from the dropdown, and press **OK.**

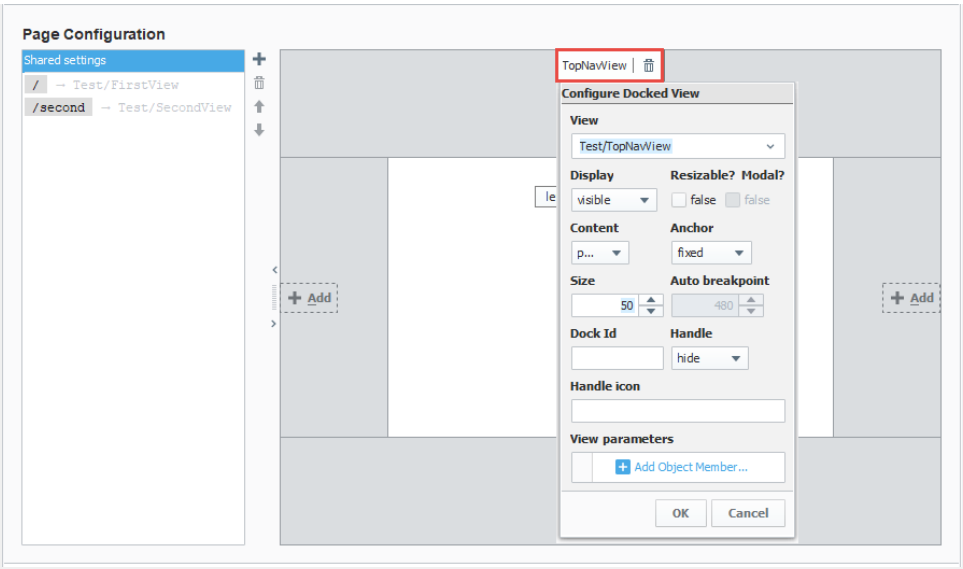
> If the dropdown is empty or you are missing views, the **pages** (not views) were never created. Go back to Step 2 in Navigating in a Perspective Session and configure your pages. Once your pages are configured, you will see them in the dropdown list.



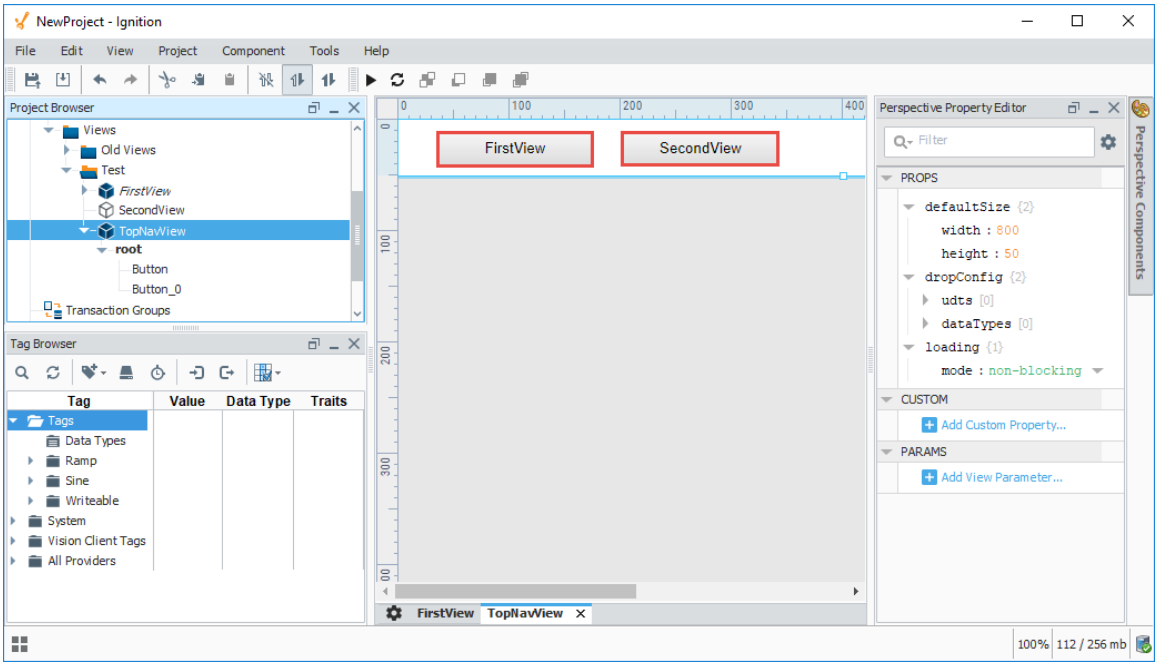
Your navigation view (i.e., **TopNavView**) will appear at the top of your page layout.



9. You can edit any of the docked view properties by clicking on the name of view in the page layout. Edit any of the properties and click **OK**. Regardless of your **TopNavView's** height, the **Size** property here will determine the height of the view in your session.



10. When you're finished setting up your Page Configuration, **Save and Publish** your project. Now the **TopNavView** will be shared on all the pages of your Perspective Session.



4. Launching a Perspective Session

Once your project is created and pages are configured, you're ready to launch a Perspective Session.

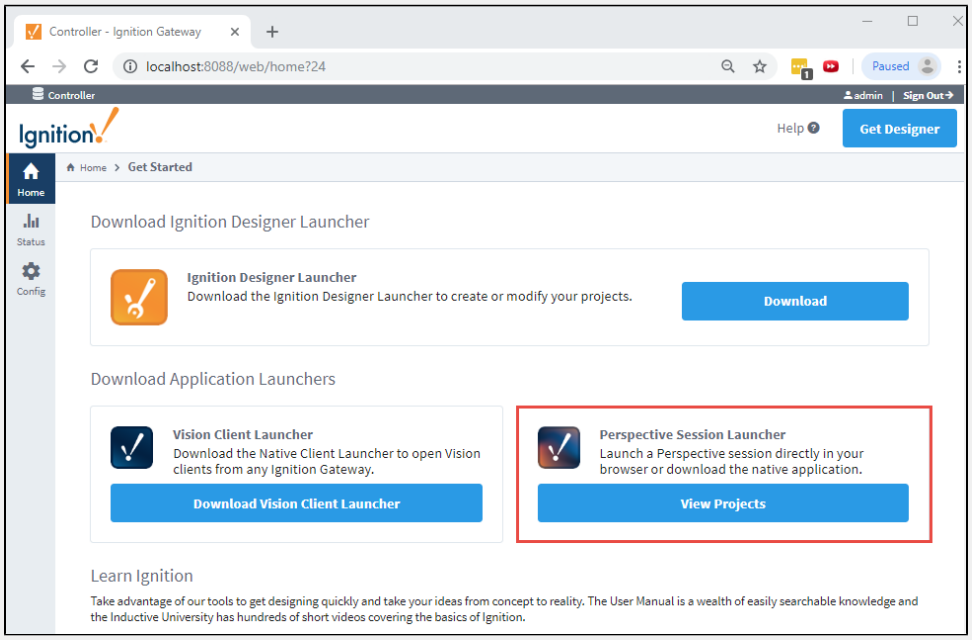
There are two ways to launch a Perspective Session:

1. From the **Gateway Webpage** home, click the **Projects** button below the Perspective logo. All the projects in your Gateway will be displayed. Click the **Launch Project** button for a project.
2. Manually enter the web address of the project in your browser.
3. From the **Designer** on the menubar, click **Tools > Launch Perspective**. This method is great if you are both a designer and a user.

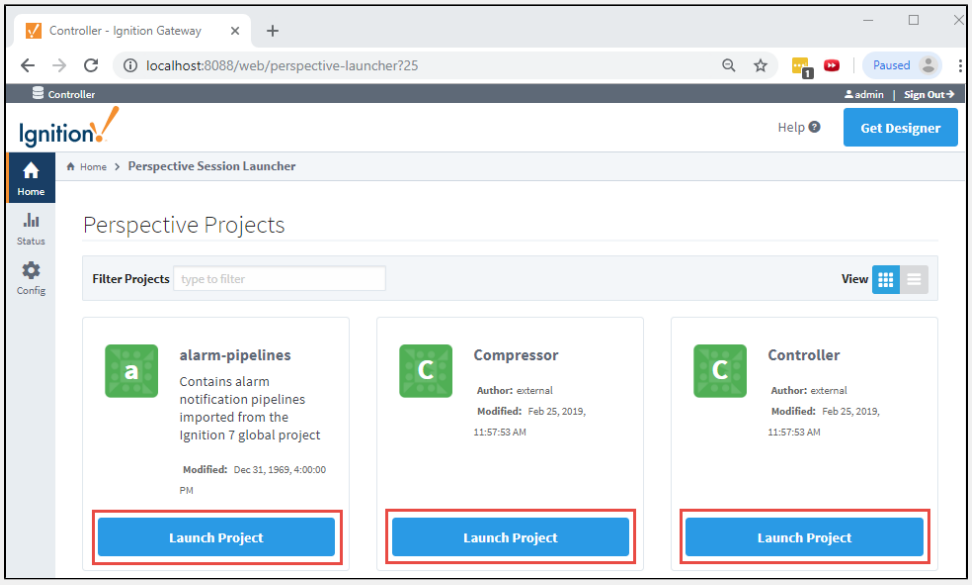
See Launching a Perspective Session for more information.

Gateway Webpage

1. Open a new browser window and enter the **IP address** of the Ignition project Gateway. You can also use **'localhost: 8088'**.
2. On the Gateway Webpage you will see the Get Started page. Under the Perspective logo, click **View Projects**.

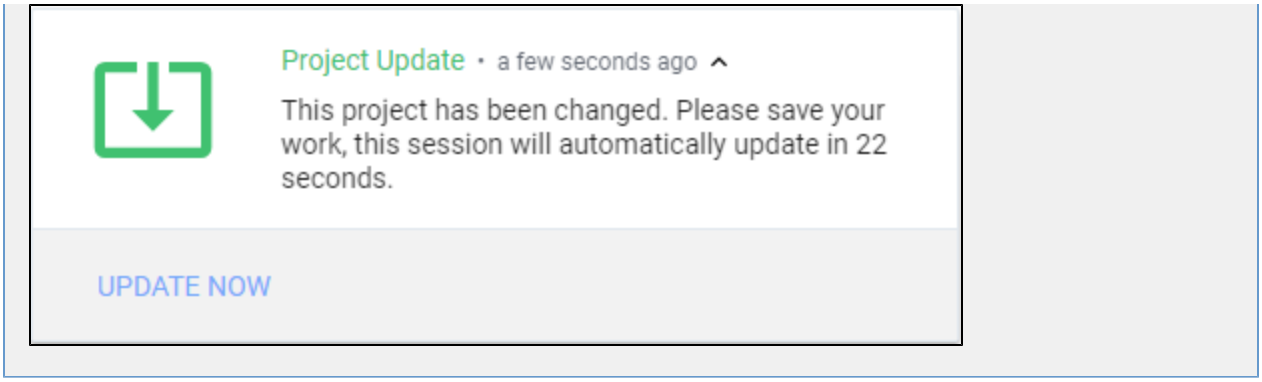


3. The window will refresh with a list of Perspective projects. Click the **Launch Project** button to launch the project in a Perspective Session.



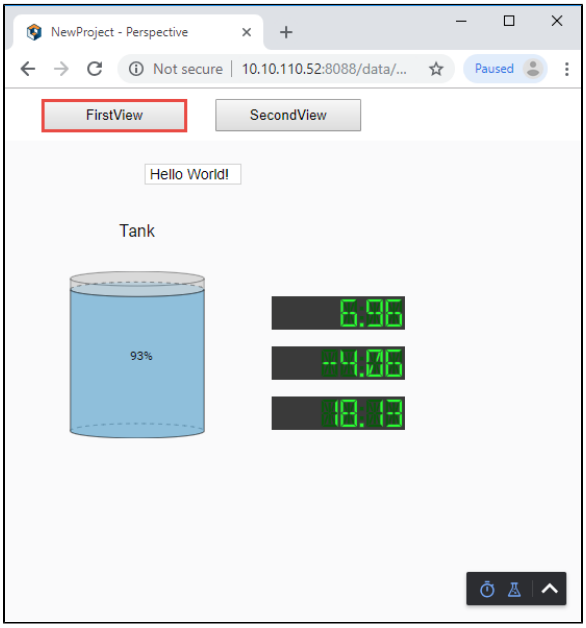
Project Updates

If you have a Perspective Session open and a change was made in the Designer that was saved and published, an Update Notification window will appear in your session. Your session will automatically update in 30 seconds or you can click **Update Now**.



Here is what your Perspective session will look like in your browser window. Toggle between the **FirstView** and **SecondView** to switch between pages in your session. The FirstView displays the components and the SecondView displays the XY Chart.

FirstView



SecondView

Related Topics ...

- Perspective Designer Interface
- Perspective Sessions
- Navigation Strategies in Perspective
- Views and Containers in Perspective
- Pages in Perspective

Quick Start - Vision Client

Overview

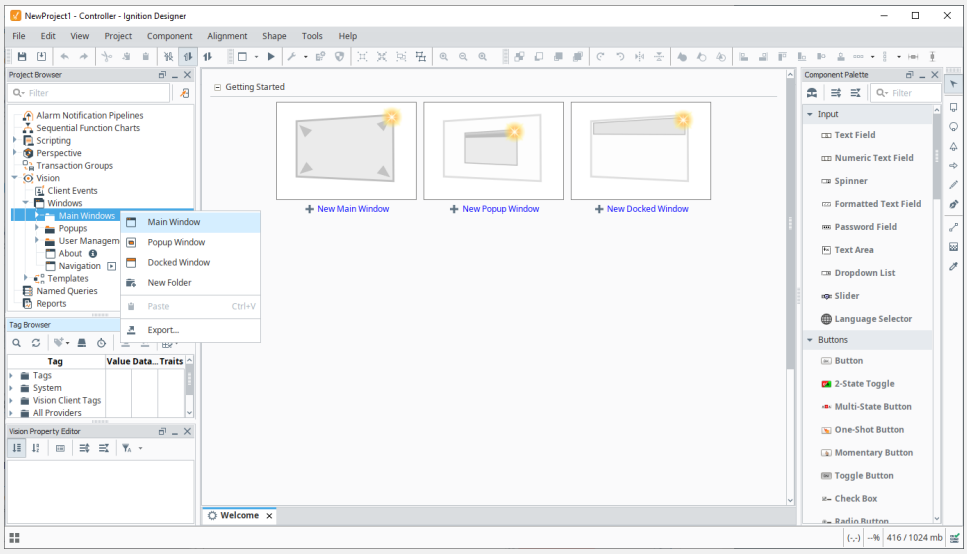
Vision Windows are the basic building blocks for all your Vision projects. Each window can contain any number of components that can display Tag values, run scripts, write values to the database, and display Tag History data on an Easy Chart. When you publish your project, these windows are loaded into the Vision Client where any number of windows can be opened at one time.

On this page, we'll walk you through how to add components to a Vision Window and bring them to life by showing live values, displaying Tag History data on an Easy Chart, and launching a project in a Vision Client.

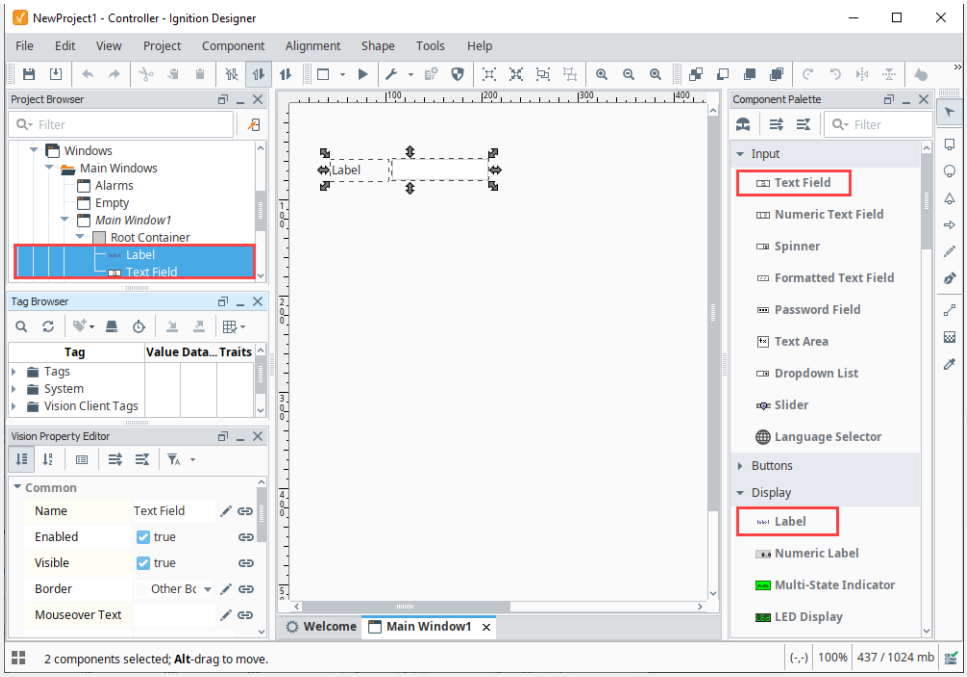
1. Adding Components to Vision Windows

Windows are the primary unit of design for Vision Clients and can be displayed as a whole page, as a popup, or docked across the top, bottom or sides of a Client. Components can do everything from displaying text, to showing Tag values, to displaying reports. You can bind components to each other, to Tags, to values in the database, or any combination. Let's drag a few components into your project windows and show some live values .

1. From the Vision Designer, go to the **Project Browser** and right-click on **Windows** to create a **Main Window** . You will be prompted to enter a window name (i.e., **Main Window1**). **Note:** You have the option to create folders for organizing your windows. This first window you create will automatically open when a client starts, but that can be changed. We will talk more about that later.

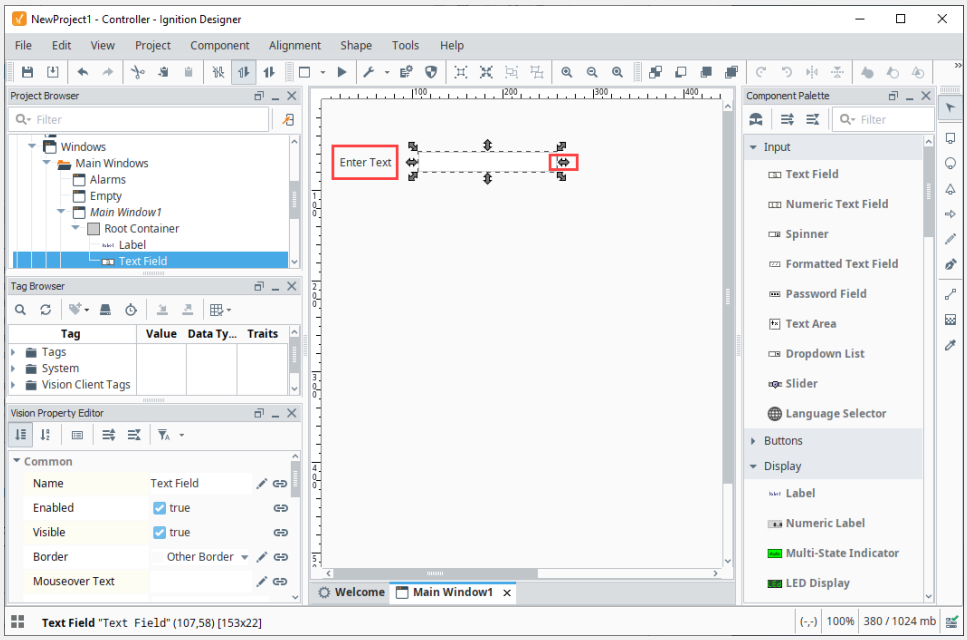


2. The window will refresh and you can begin clicking and dragging components from the **Component Palette** on the right onto your window. Add a **Label** and a **Text Field** component.

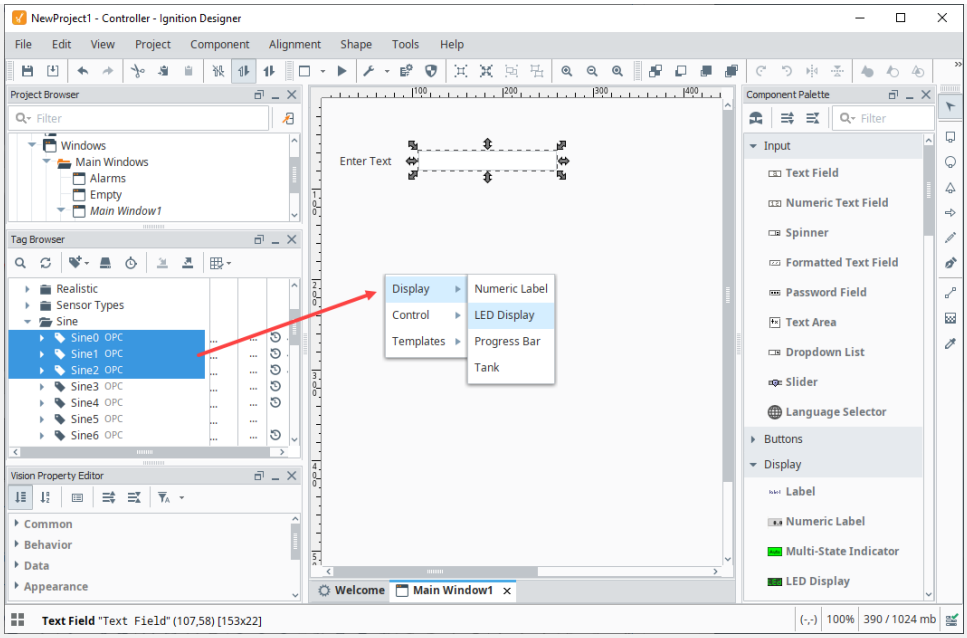


3. Double-click on the **Label** component to type directly into it, or go to the **Text** property in the **Property Editor,** and rename it to say "**Enter Text**."

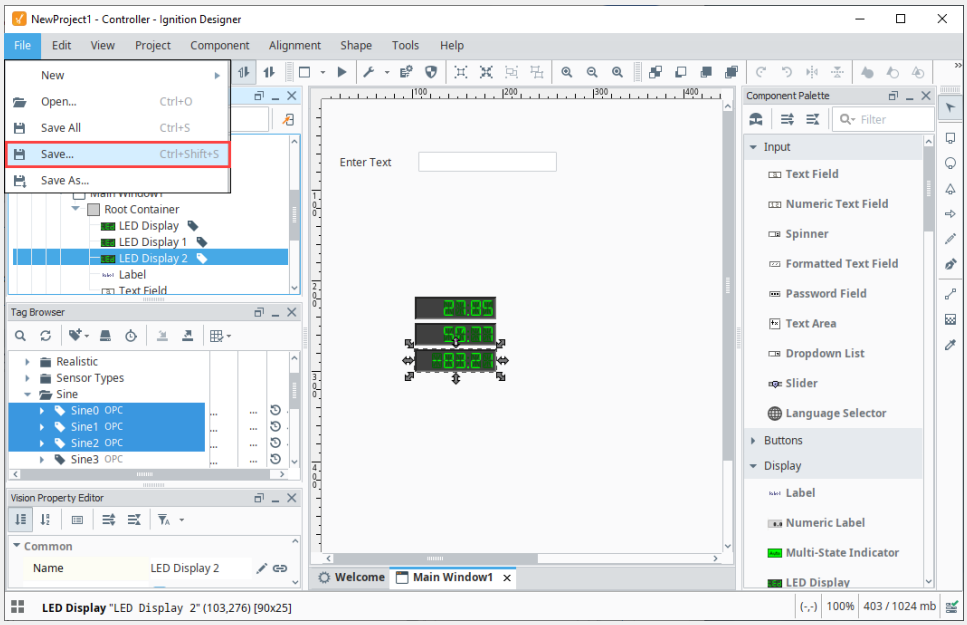
4. Select the **Text Field** component and click-and-drag the arrow handles to resize the component.



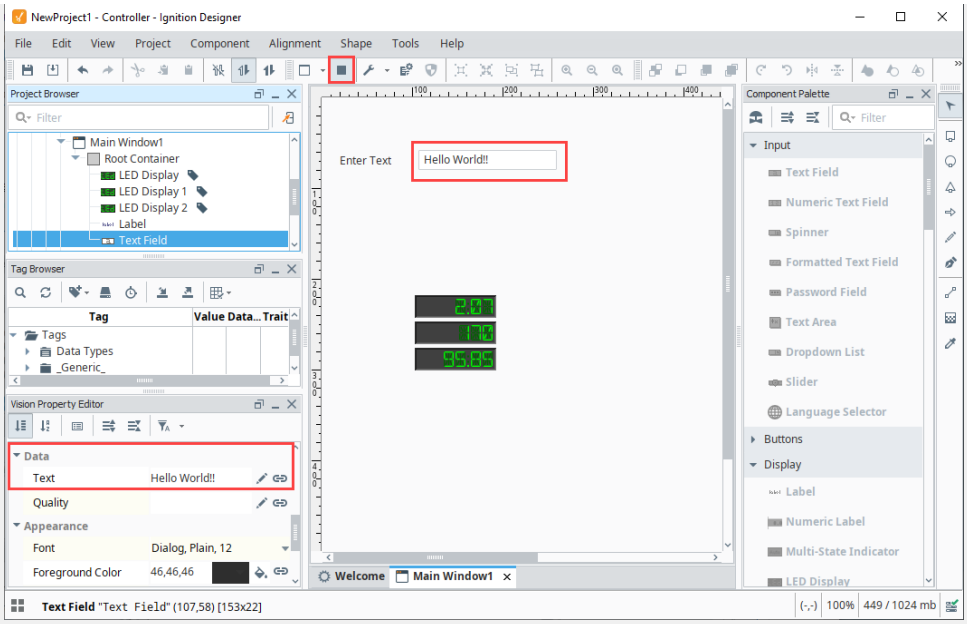
5. Now let's add some Tag values to the window. (**Note:** This step assumes you are already connected to Generic Simulator. Refer to the Quick Start Guide for Connecting to a Device).
The simplest way to add Tag values is to go into the **Tag Browser** and drill down to the Tag(s) you want and drag them onto the window. Select a few Tags (i.e., **Sine Tags**) and drag them into the window. You will see a popup asking you what type of components you want to use to show these Tag values. Select **LED Display**.



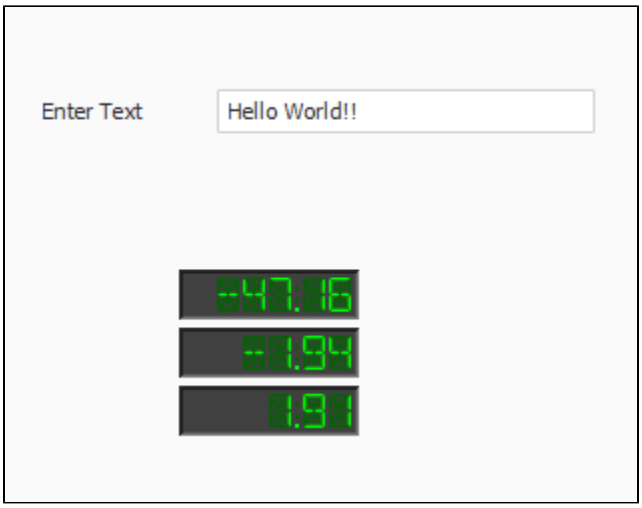
6. Now you can see some live values on your window! Don't forget to **Save** your project.



7. You can Preview your project in the Designer by clicking the **Preview Mode** button in the top menubar. This allows you to interact with your windows as if you were using a Vision Client. "The Preview Mode button changes between a gray triangle and a gray square depending on which mode you are currently in. Look for the gray triangle and click on it to go into Preview Mode. In this example, we entered "Hello World!! in the Text Field. You can see Property Editor, "Hello World!! also appears in the Text property of the Text Field.



Learn more about Working with Vision Components and Property Bindings!

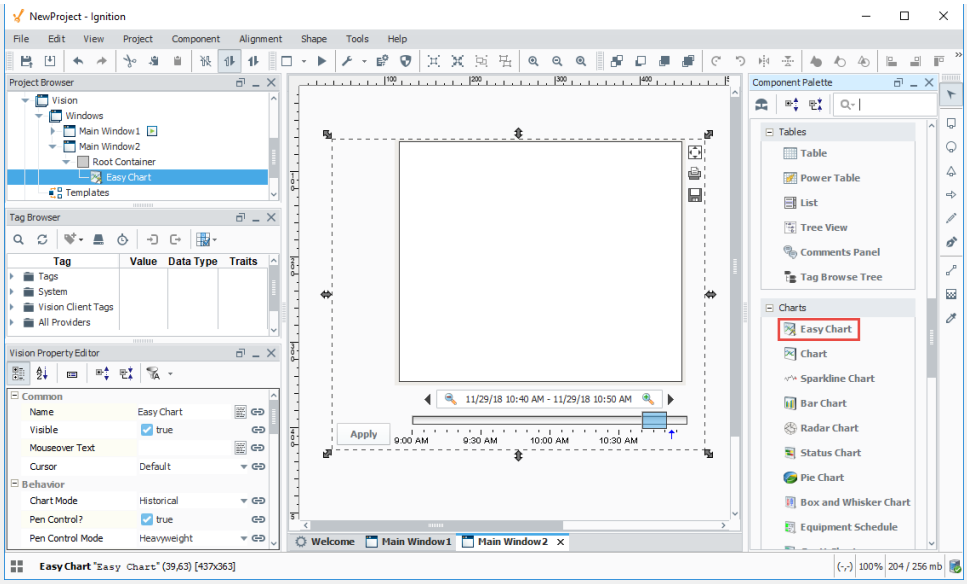


2. Showing Tag History in an Easy Chart

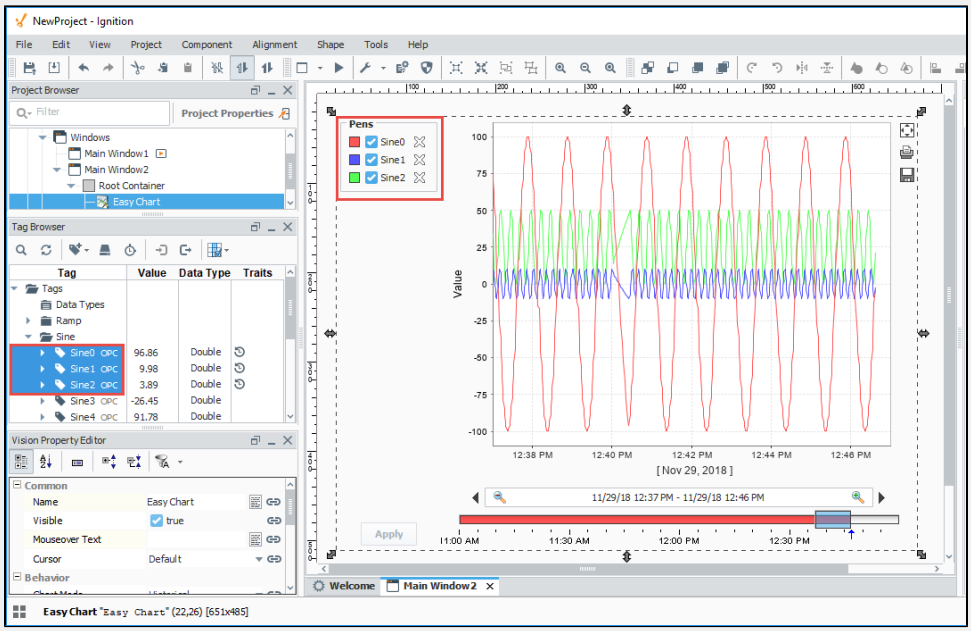
Display Tag History using an Easy Chart. Showing Tag History requires that you have Tag History already setup. Refer to the Quick Start Guide, Add History to Tags.

In this example, create a second Main Window (i.e., **Main Window2**), add an Easy Chart component to show history on the values of the LED Displays.

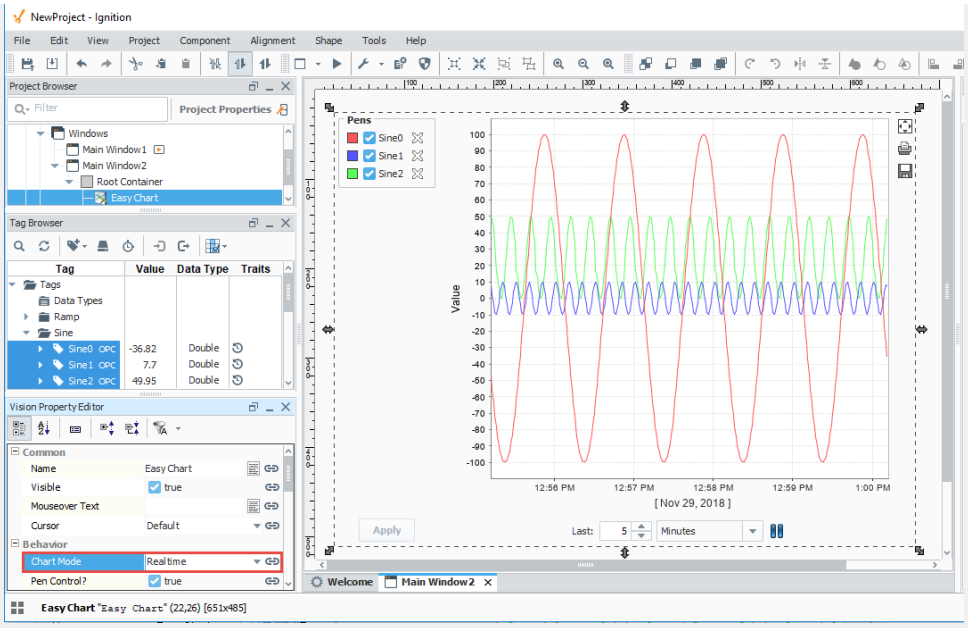
1. Once you have your second Main Window created, drag an **Easy Chart** component on to your window (i.e., **Main Window2**).



2. In the **Tag Browser**, the first three Sine Tags are logging data. Drag these Sine Tags to the Easy Chart. The Easy Chart will immediately fetch the results and trend the history. When you first drag Tags to the Easy Chart, not much data will be displayed because you just started logging data.

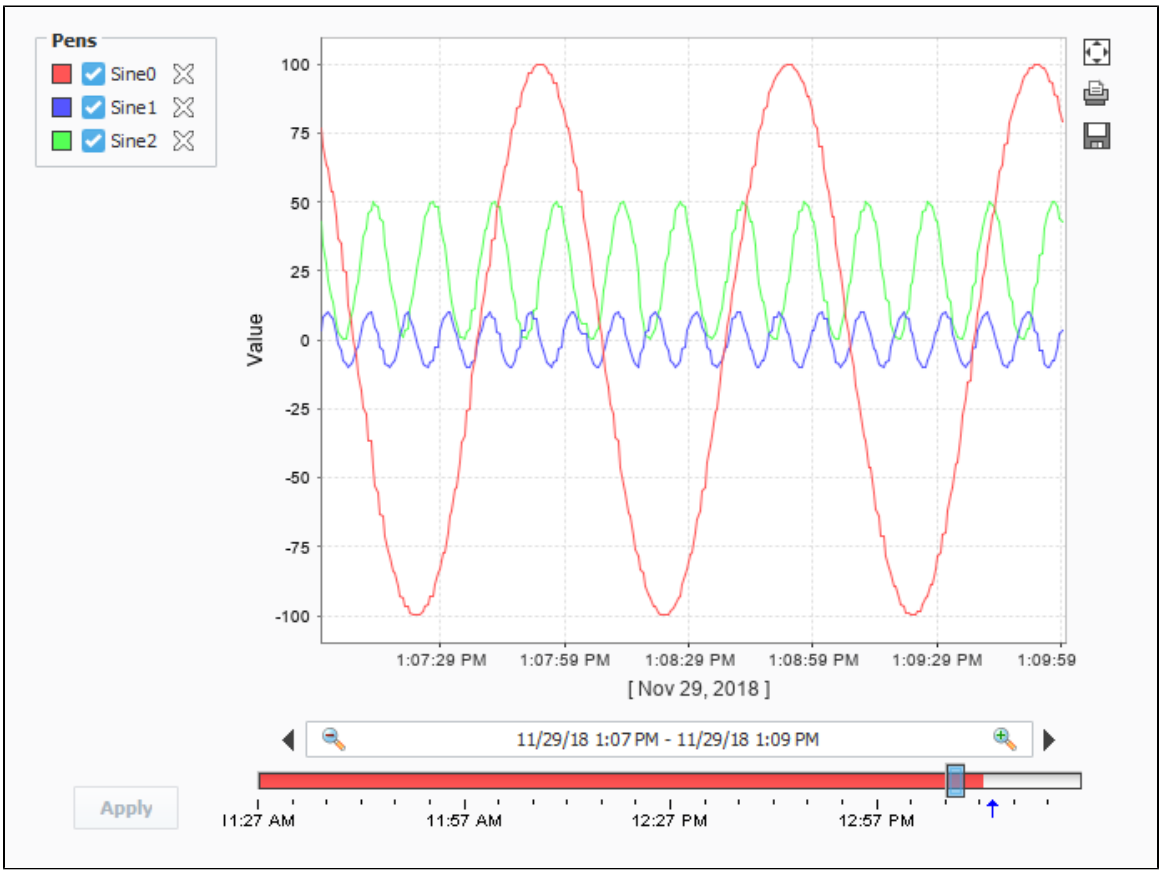


3. To show more data, set the **Chart Mode** property from Historical to **Realtime** in the **Property Editor**.



4. From the top menubar, save your project by clicking **File > Save**.

Learn more about the Vision Easy Chart!

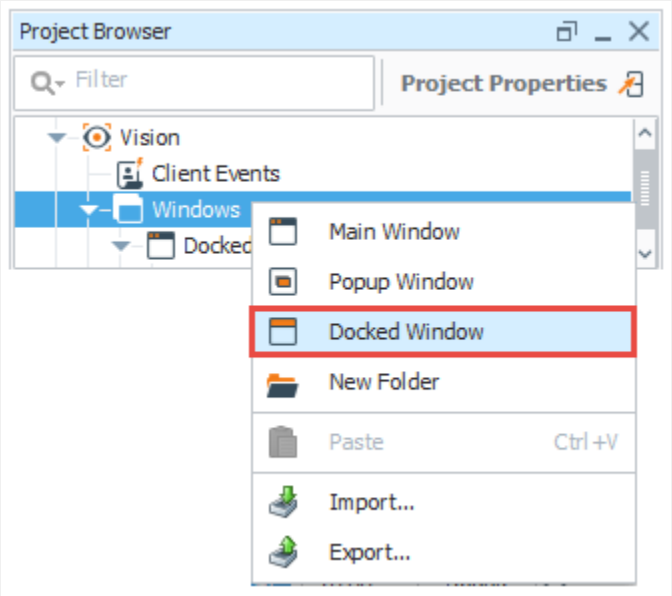


3. Navigating in a Vision Client

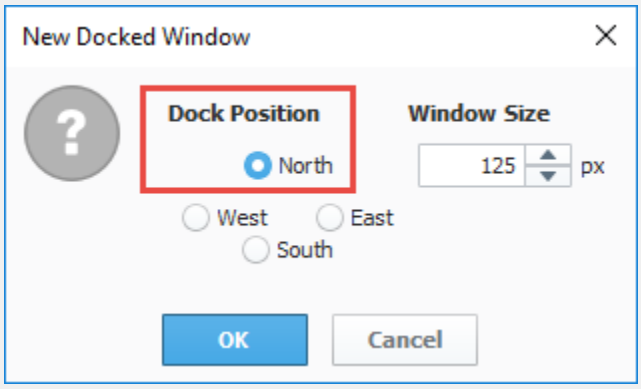
It's important to have a navigation strategy that allows you to navigate seamlessly between different windows in the Vision Client. You can choose from several types of navigation strategies: Tab Strip, Tree View, Menubar, Forward and Back Buttons, and Drill Down. The Naviation Strategies in Vision page provides some things to consider to help you determine the best navigation strategy for your project and your users.

In this example, we are going to use a simple Docked window and two Button components to switch between the two windows we created in the previous sections.

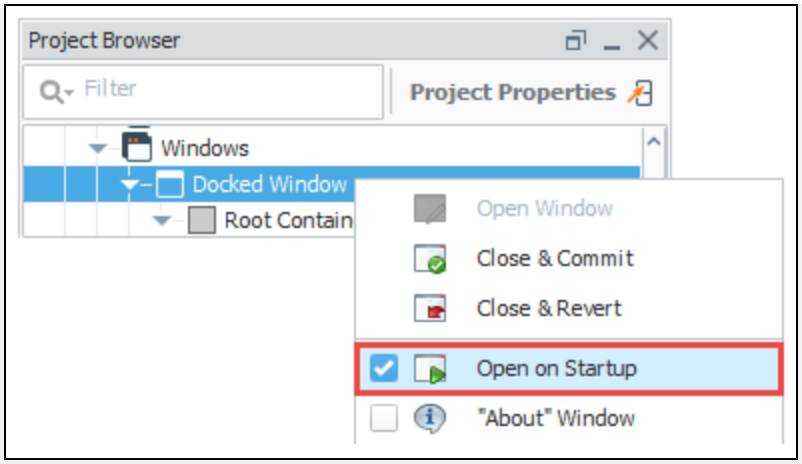
1. First, let's create our docked window. In the **Project Browser**, right click on **Windows** and select **Docked Window**.



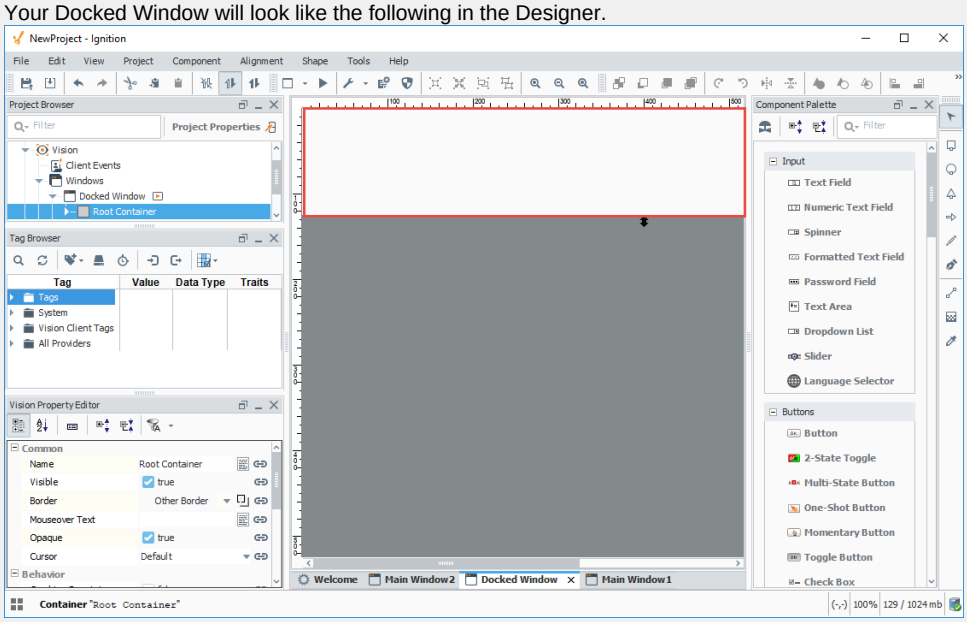
2. A New Dock Window will appear for you to select the **Docked Position**. You can choose to display a Docked Window across the top, sides or bottom of your Vision window. In this example, we selected **North.** Click **OK**.



3. Right click on your new Docked Window and check **Open on Startup**. Your Docked window will now open on startup in the client and will be shown above all main windows in the Vision Client. You can do the same to any window to toggle this option on any window. For now we only want two windows with this set, the first window from above (Main Window 1) and this Docked Window.

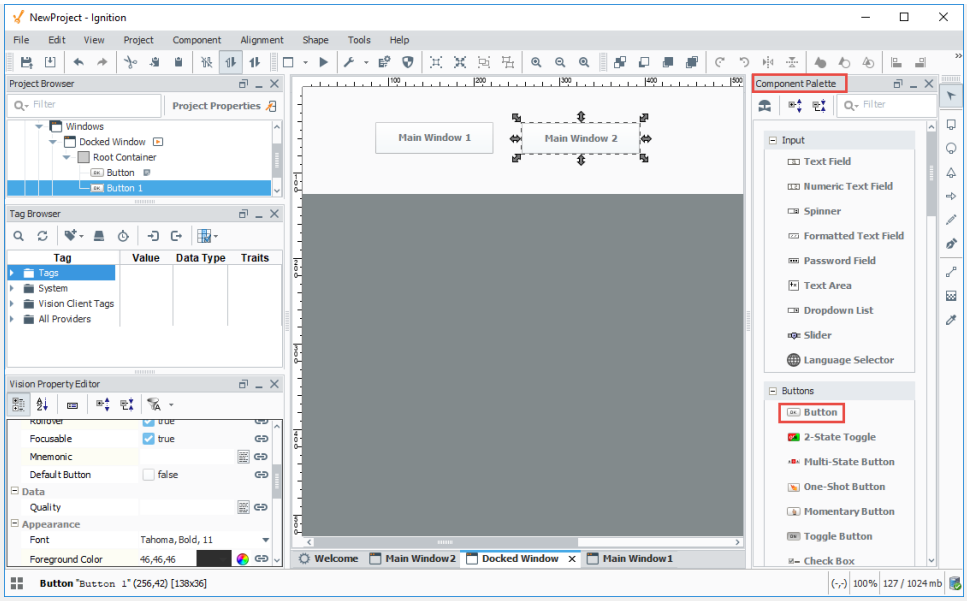


Your Docked Window will look like the following in the Designer.

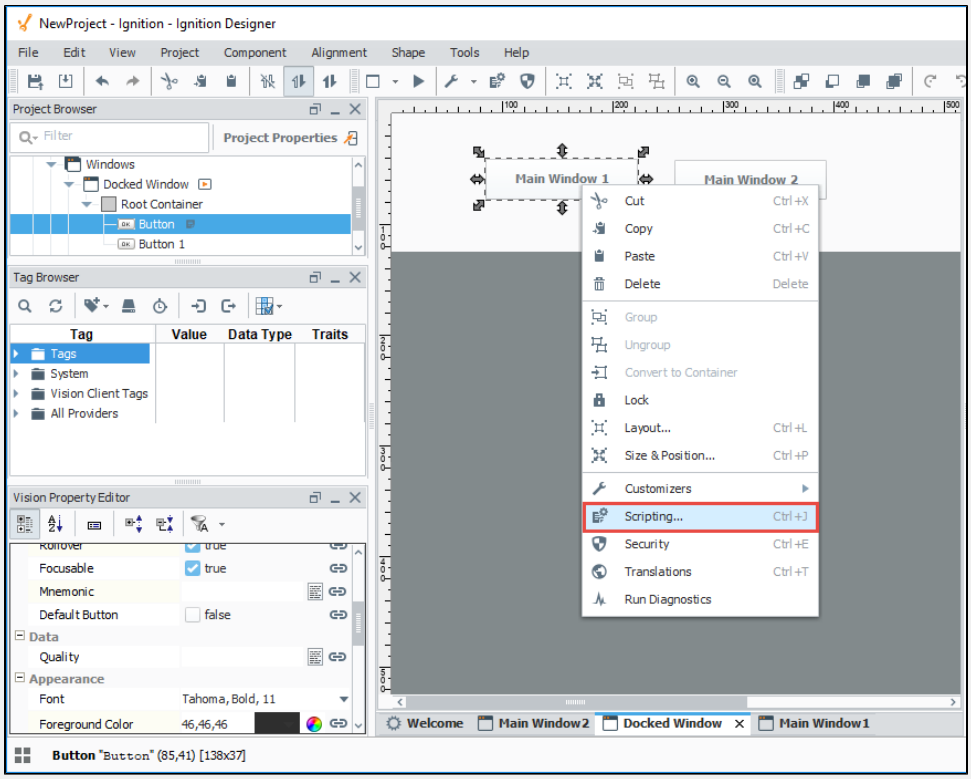


4. Now, let's drag a couple of **Button** components from the Component Palette to the Docked Window, or you can duplicate (Ctrl D) the first button to create the second button. We'll use these Button components to navigate from one window to another in the Vision Client.

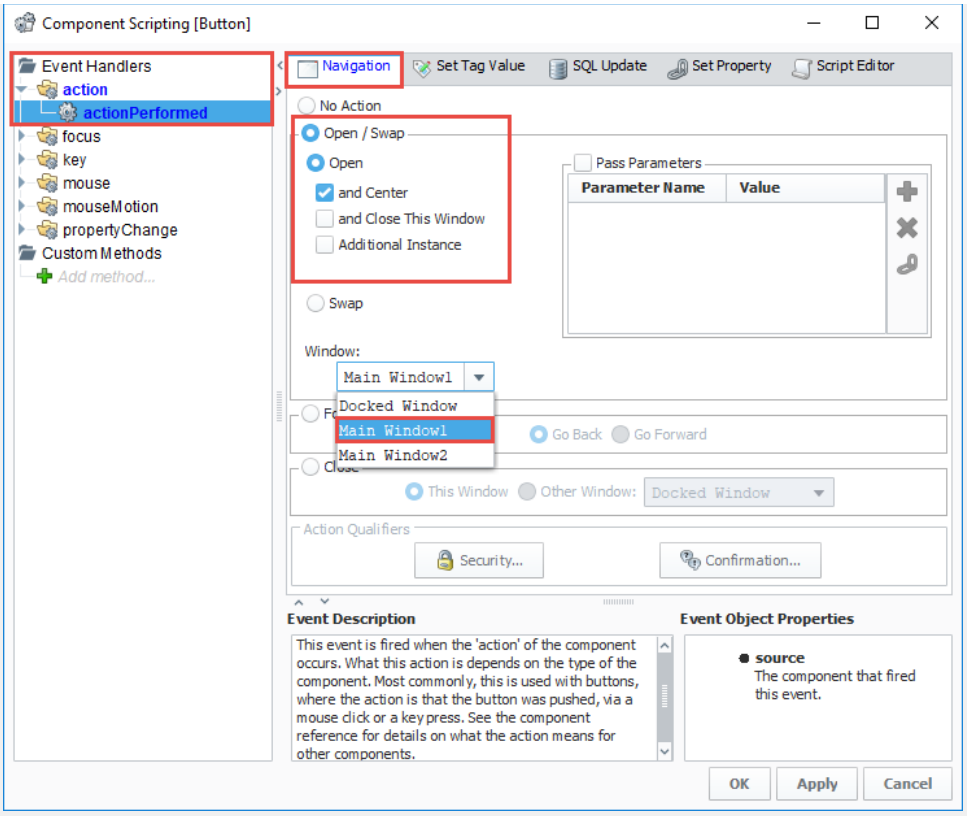
5. Change the **text** property of the Buttons to something meaningful (i.e., **Main Window 1** and **Main Window 2**).



6. Now you need to add scripting to each of your Buttons so they know what window to open when pressed. Right click on your first Button component (i.e., **Main Window 1**) and select **Scripting.**

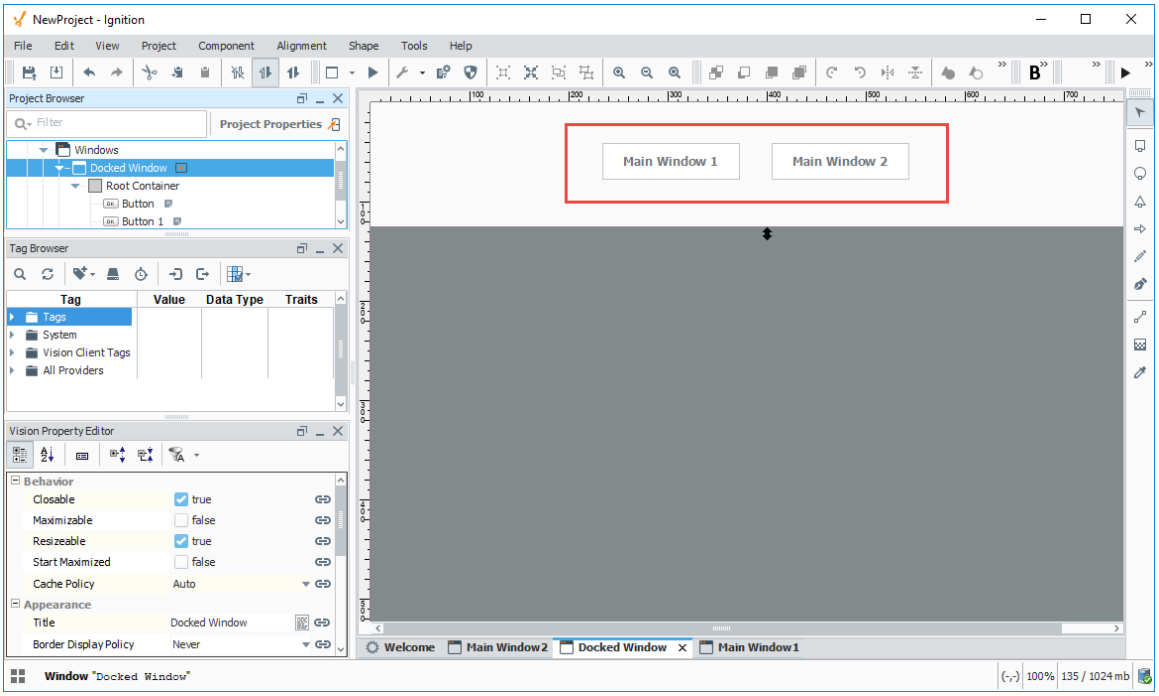


7. This opens the **Component Scripting [Button]** window. Here you have to tell the buttons to swap one window for the other when the Button is pressed. In other words, it closes the current window you have open, and then opens the other window in its place. Use the following steps to apply your script to your first button (i.e., **Main Window 1**):
    a. Select the **Main Window 1** button.
    b. Under **Event Handlers**, expand the **action** folder, and select **actionPerformed**.
    c. Select the **Navigation** tab, select **Open / Swap**, then **Open**, then **and Center**.
    d. From the **Window** dropdown, select the **Main Window1**.
    e. Click **OK**.



8. Repeat Step 6 for your second button (i.e., **Main Window 2**):
   a. Select the **Main Window 2** button.
   b. Under **Event Handlers**, expand the **action** folder, and select **actionPerformed**.
   c. Select the **Navigation** tab, select **Open / Swap**, then **Open**, then **and Center**.
   d. From the **Window** dropdown, select the **Main Window2**.
   e. Click **OK.**

9. The image below shows how your Docked Window will look in the Designer. Remember to **Save** your project.



4. Launching a Vision Client

Now that we created a project with a few windows and some basic navigation, let's launch a Vision Client. There are three ways to launch a Vision Client:

1. From the Gateway webpage under the **Vision Client Launcher** logo, click the **Downloads** button. This method must be used after you create your first project, typically on a new installation of Ignition and a new release of the Vision Client Launcher application.
2. From the **Vision Client Launcher** executable or a desktop shortcut, if you have used them previously.
3. From the **Designer** on the menubar, click **Tools > Launch Project > Windowed (Published)**. This method is great if you are both a designer and a user.

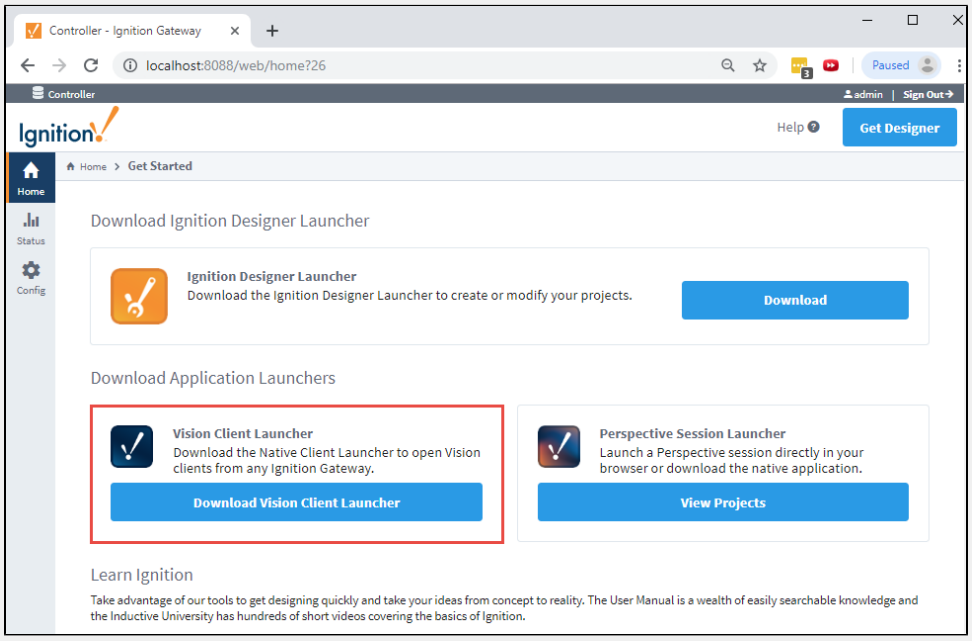
> The following feature is new in Ignition version **8.0.10**
> Click here to check out the other new features

Ignition 8.0.10 introduced a Vision Client Launcher option that can be installed as an "all user" application on a Windows operating system providing launcher acess to all users on the same machine from a single installation. The first time the Designer Launcher is installed, you will be prompted to select your install mode: "Install for me only" or to "Install for all users." If the Vision Client Launcher was already installed, subsequent install attempts are treated as an upgrade and you will not be prompted to select the install mode. This new option will only appear on new launcher installations or after uninstalling a prior version of the installer.

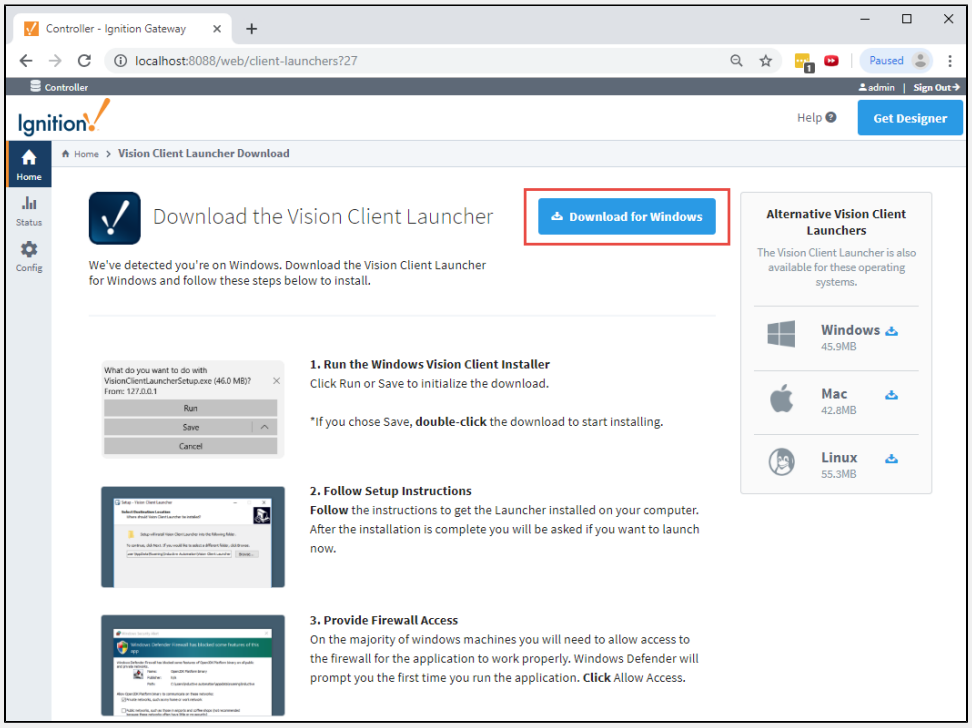
> **Vision Client Launcher Destination Folder**
>
> When installing for "all users," the installer will be placed in the following folder: "%Ignition Installation Directory%\Program Files\Inductive Automation"
> When installing for "me only," the installer will be found under the "\AppData\Roaming\Inductive Automation" folder.

Launching a Vision Client from the Gateway Webpage

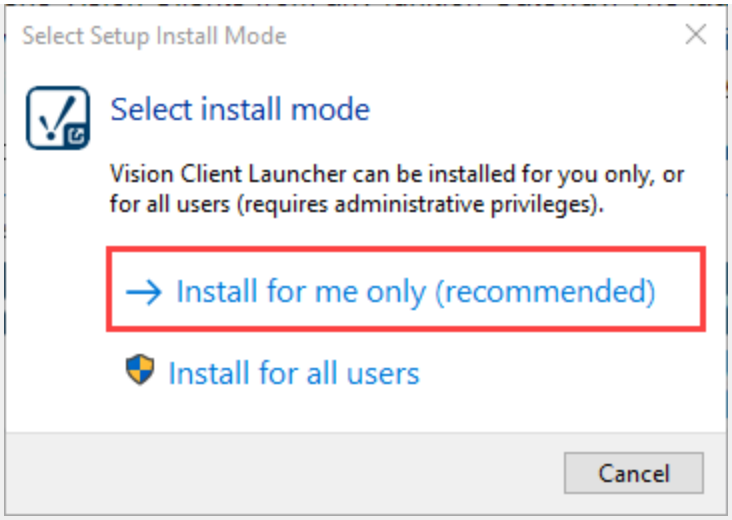
1. Open a new browser window and enter the **IP address** of the project Gateway . You can use **'localhost:8088'**.
2. If you're not already logged into Ignition, enter the credentials for the administrator or user with full privileges for the Gat eway. You must have a project created. (The project in this example is called NewProject). If this is your first time launching a Vision Client after installing and designing in Ignition, use the Vision Client Launcher on the Gateway webpage.
3. From the **Home** tab on the Gateway webpage, click **Download Vision Client Launcher**.



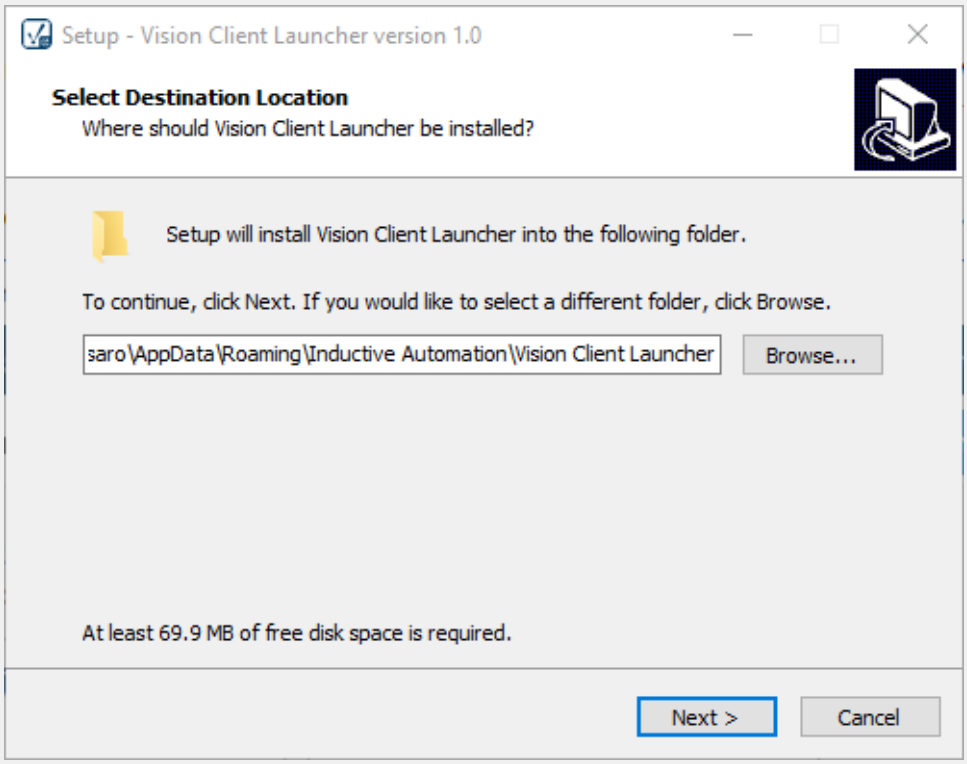
4. Select the **Download** button for the platform you are running on: Windows, Mac OS, or Linux. This example shows the installation steps for a Windows operating system.



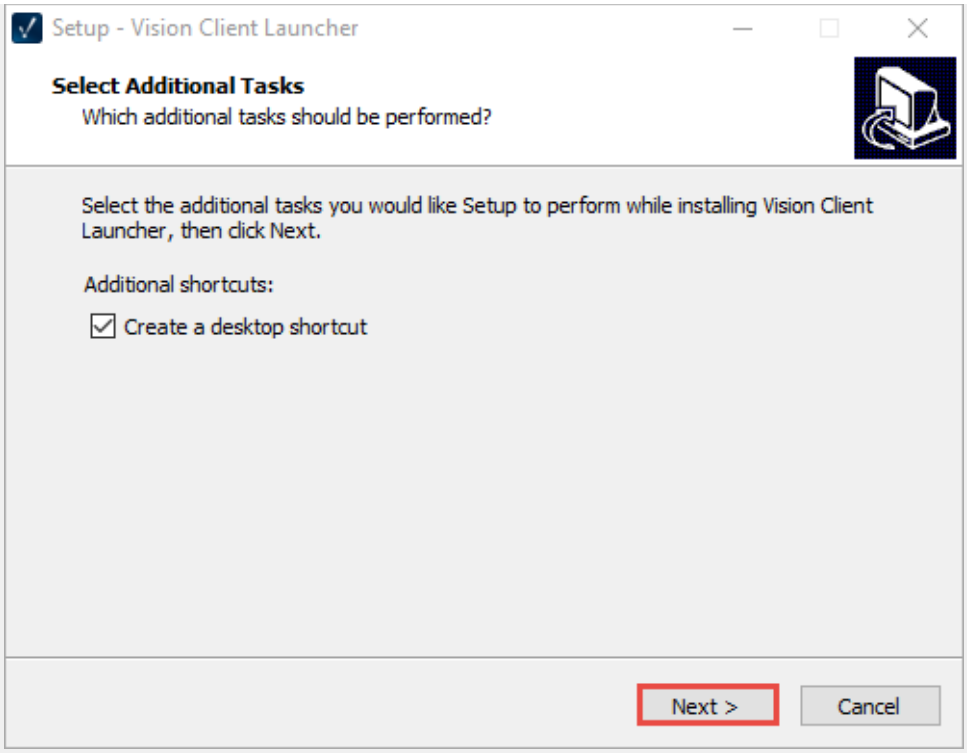
5. Run the downloaded file **VisionClientLauncherSetup.exe** (found on the lower-left of the window if using Chrome), or go to your Downloads folder and double-click the file to run it.
6. If this is the first time installing the Vision Client Launcher , you will be prompted to select an install mode. Select either **"Install for me only,"** or **"Install for all users."**



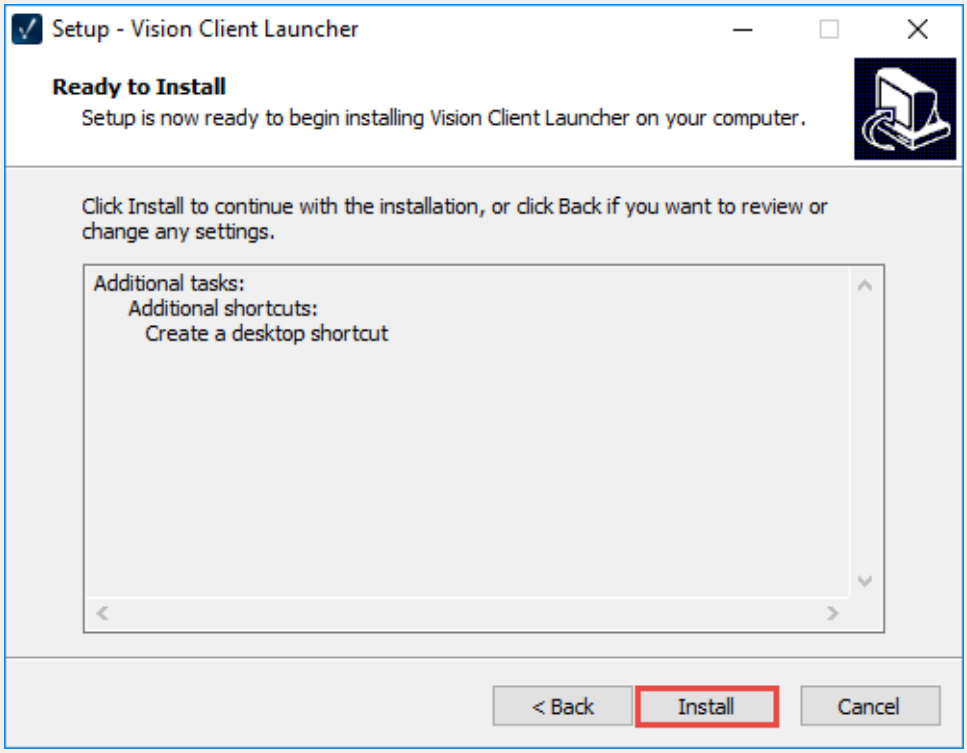
7. The Install Wizard will select a default destination location folder for the Vision Client application files. If you are "installing for me only", the destination folder will be different from the "all users" destination folder. You also have the option to select a different folder. Click **Next**.



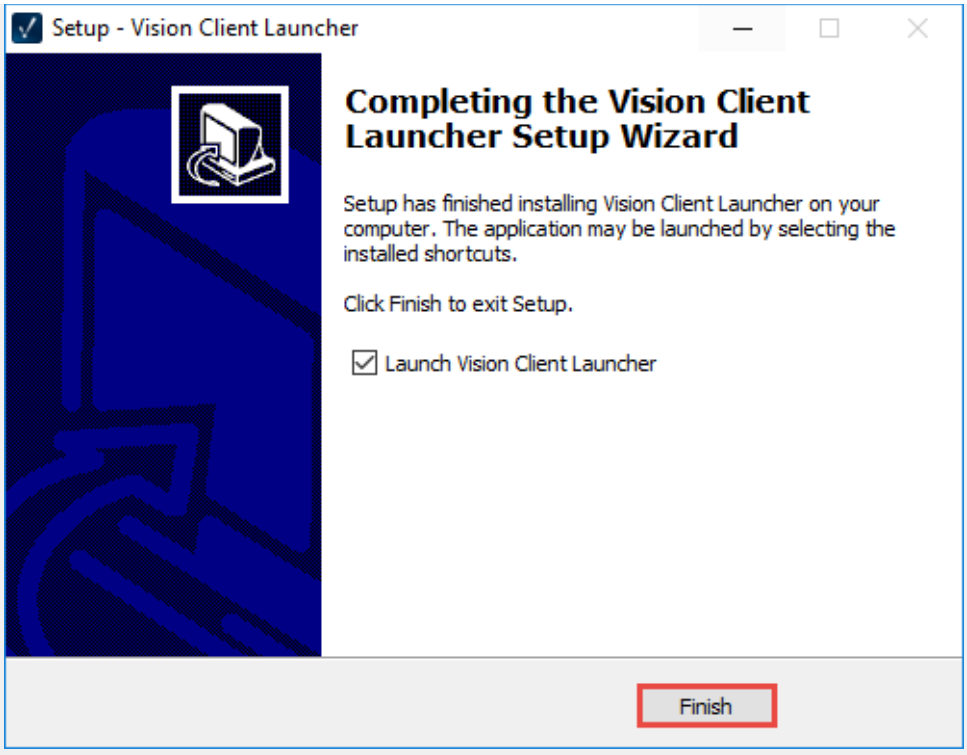
8. The Setup - Vision Client Launcher wizard will open. You'll notice that a desktop shortcut is checked and will be created so you don't have to run the Vision Client Launcher each time you want to open a client. Click **Next**.



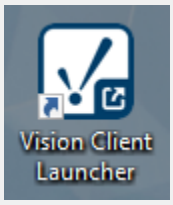
9. The Ready to Install window will open. Click **Install**.



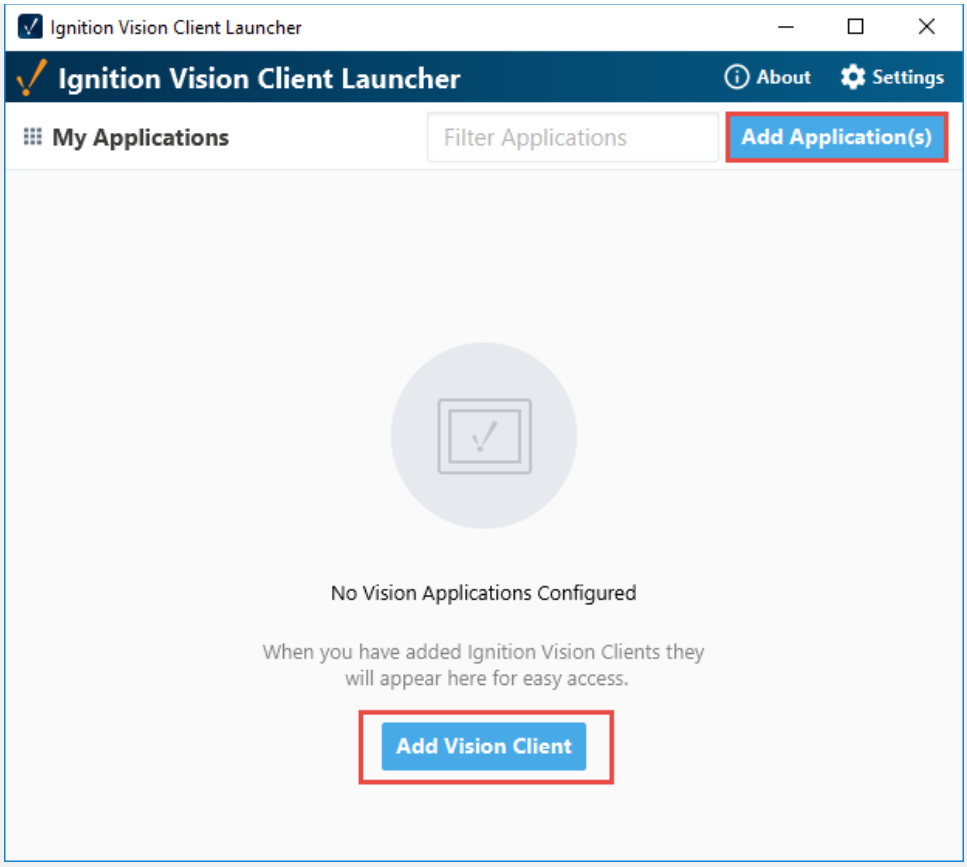
10. When the Vision Client Launcher completes the install, click **Finish** .



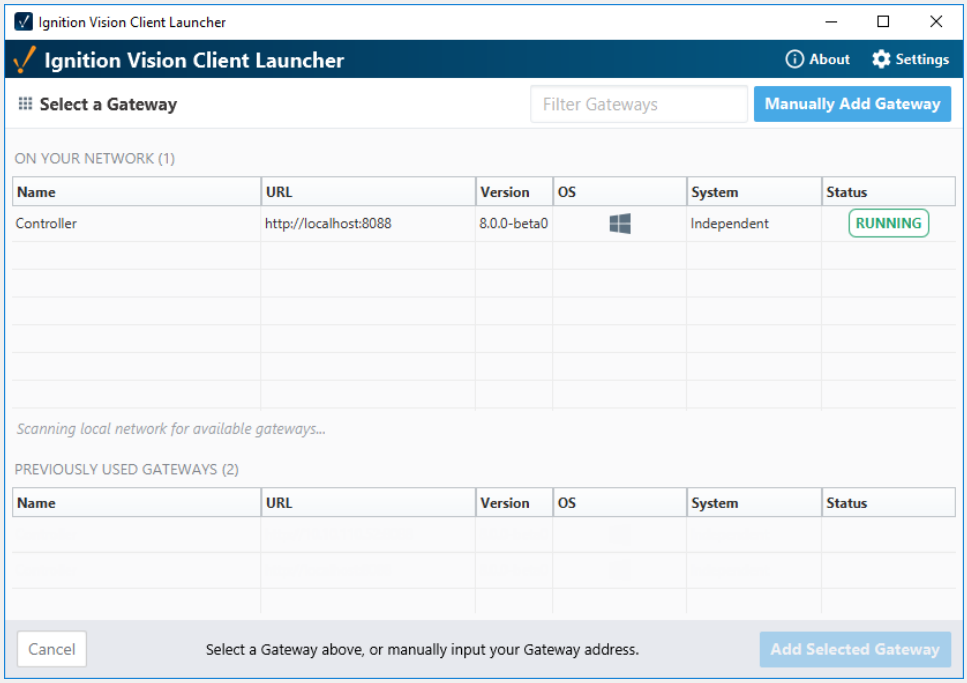
11. The Vision Client Launcher shortcut icon will be placed on your desktop and looks like the following.



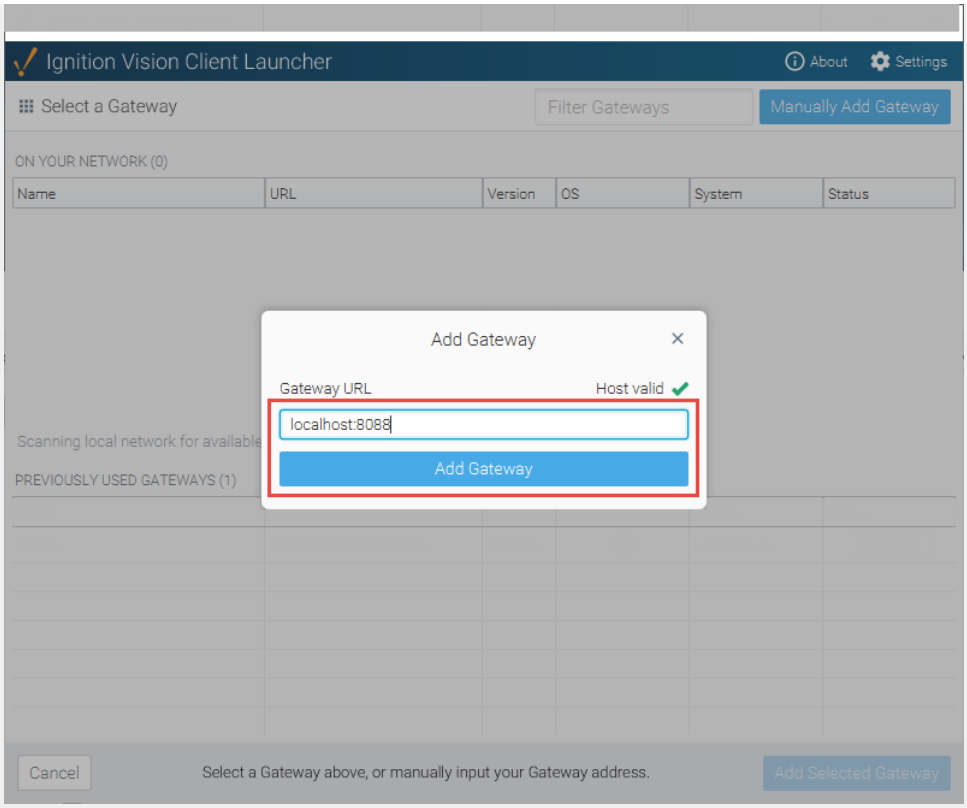
12. The Ignition Vision Client Launcher will open, but you won't have any Vision Clients (applications) available if this is your first time using the Vision Client Launcher. Click **Add Vision Client or Add Applications(s)**. Both buttons will take you to the same window to browse for local and remoter Gateways.



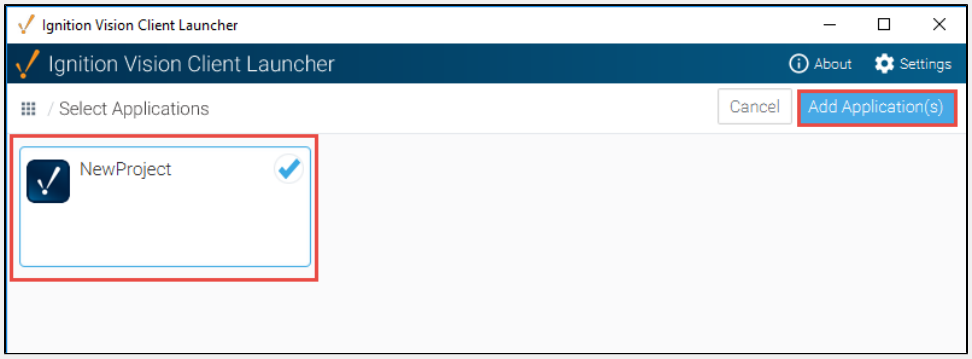
13. The window will refresh and show your local Gateway. It will also check for other Gateways on your network. Select a Gateway and click the **Add a Selected Gateway** button.



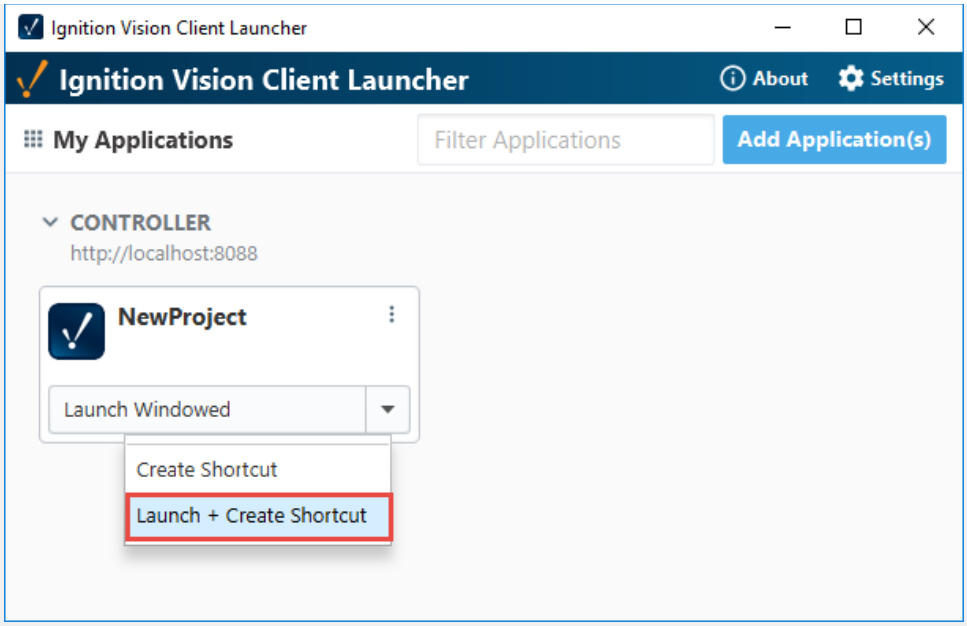
14. If there is an issue automatically detecting your Ignition Gateway, you can manually add your Gateway. Select **Manually Add Gateway** . Enter your **Gateway URL** or **IP address** (i.e., **localhost:8088**), and click **Add Gateway** .



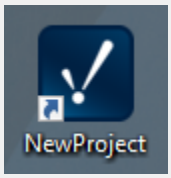
15. Once you add your Gateway, the Vision Client Launcher will automatically find all your Vision projects. Select your Vision project(s) and click **Add Applications**.



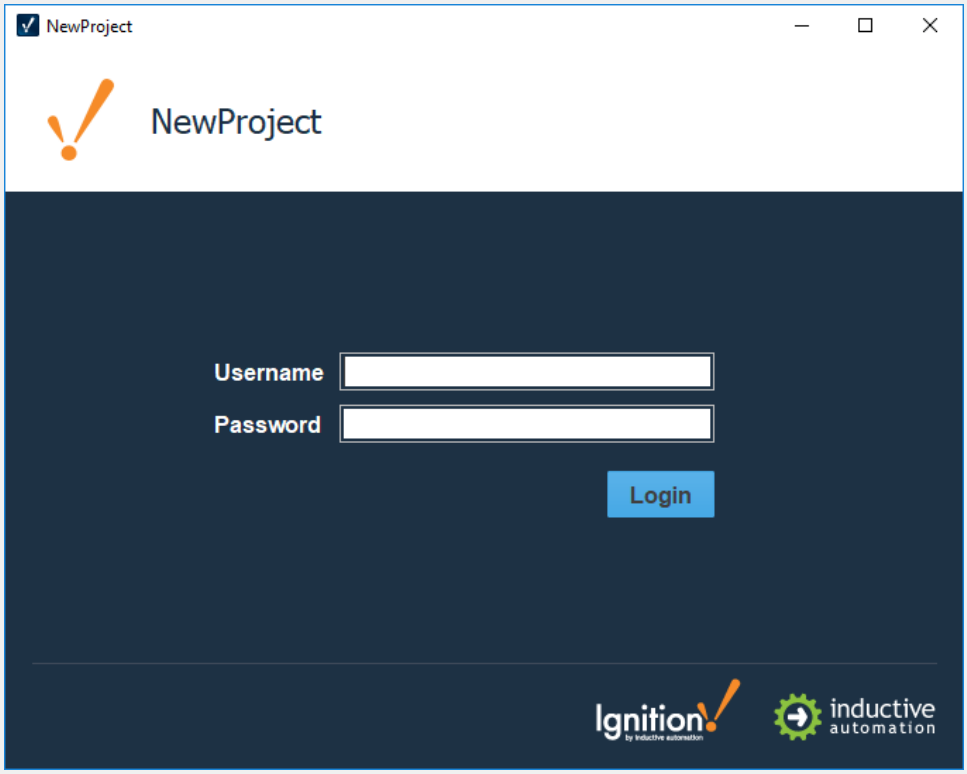
16. The following image is what you will see the next time the Client Launcher is opened. The Vision Client Launcher is installed and opens the same way you open any program. Simply launch the client by pressing on the **Launch Windowed** button, or you can click the down triangle button and select **Launch + Create Shortcut**. The project shortcut will be placed on your desktop so you don't have to run the Vision Client Launcher each time you want to launch a project.



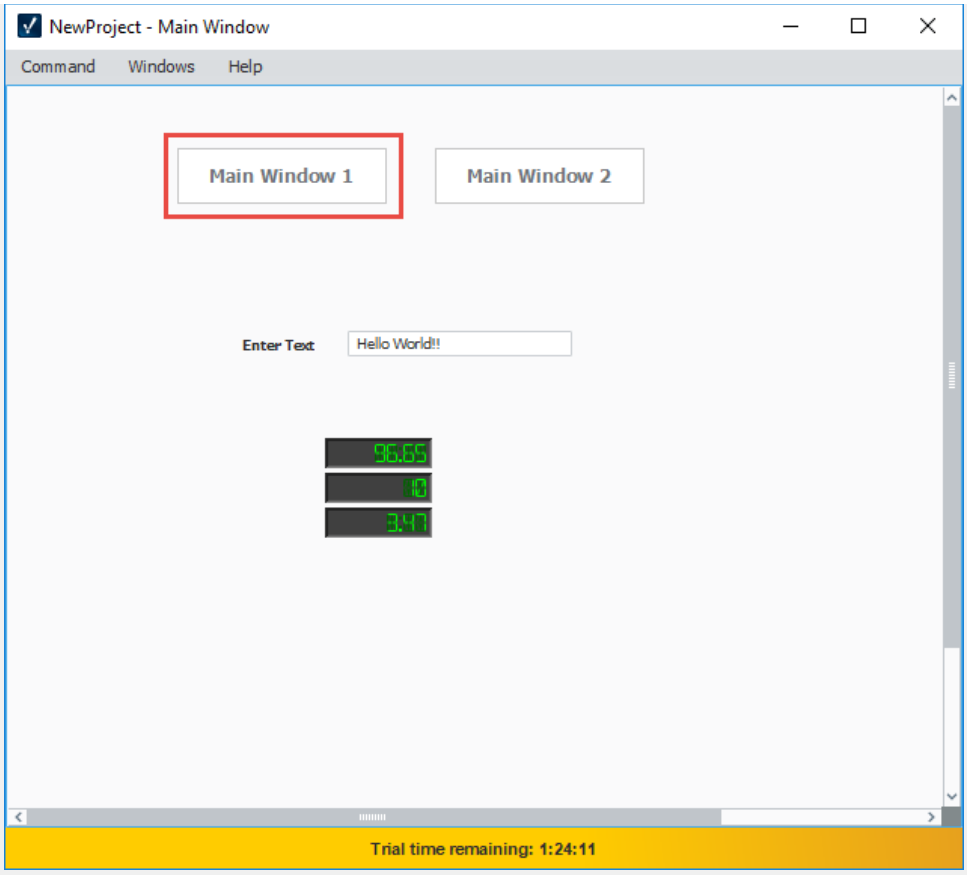
Next time you can double click the shortcut to open your project in the Vision Client.



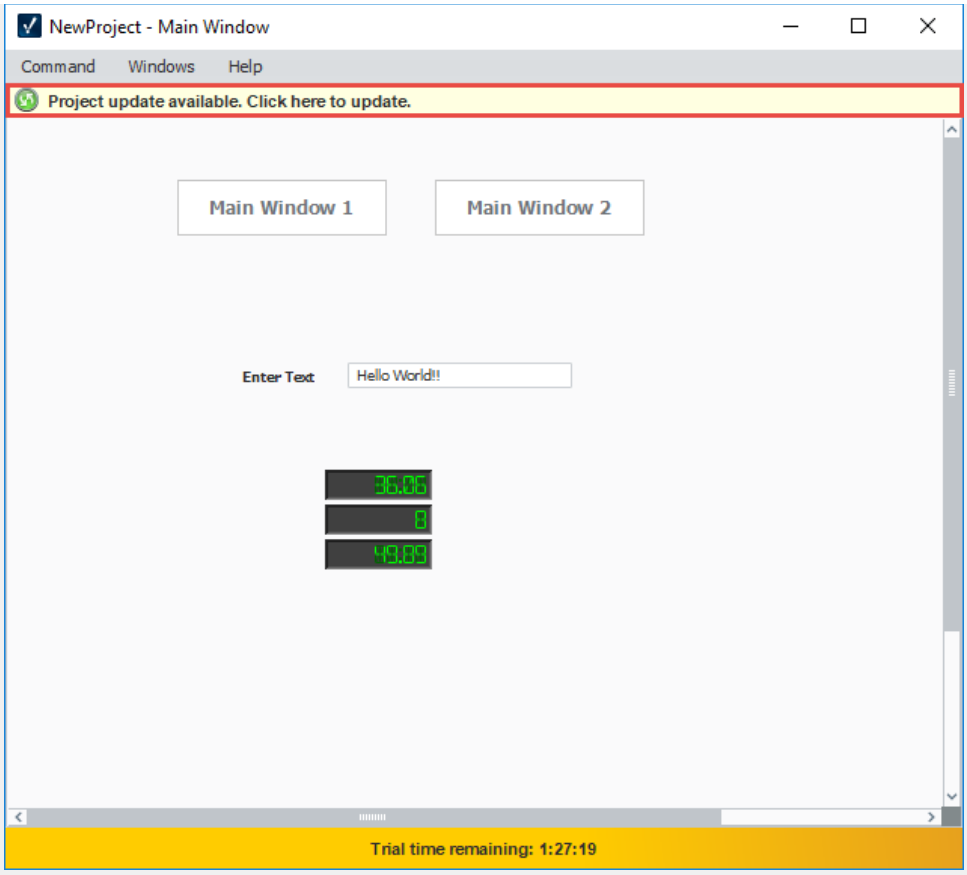
17. Login to the Vision Client using the user credentials you set up when Ignition was installed.



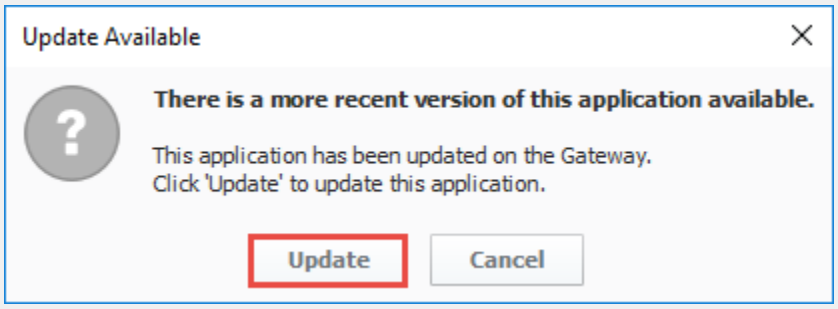
18. After you log in, you can see your Vision Client window displaying your project. Below, we are showing **Main Window 1** which opens up on startup. Try clicking on **Main Window 2**.



19. Go back into the Designer, make a change to the window, and click **File > Save and Publish**. Your Vision Client will show a notification banner stating that there is an update to the project.



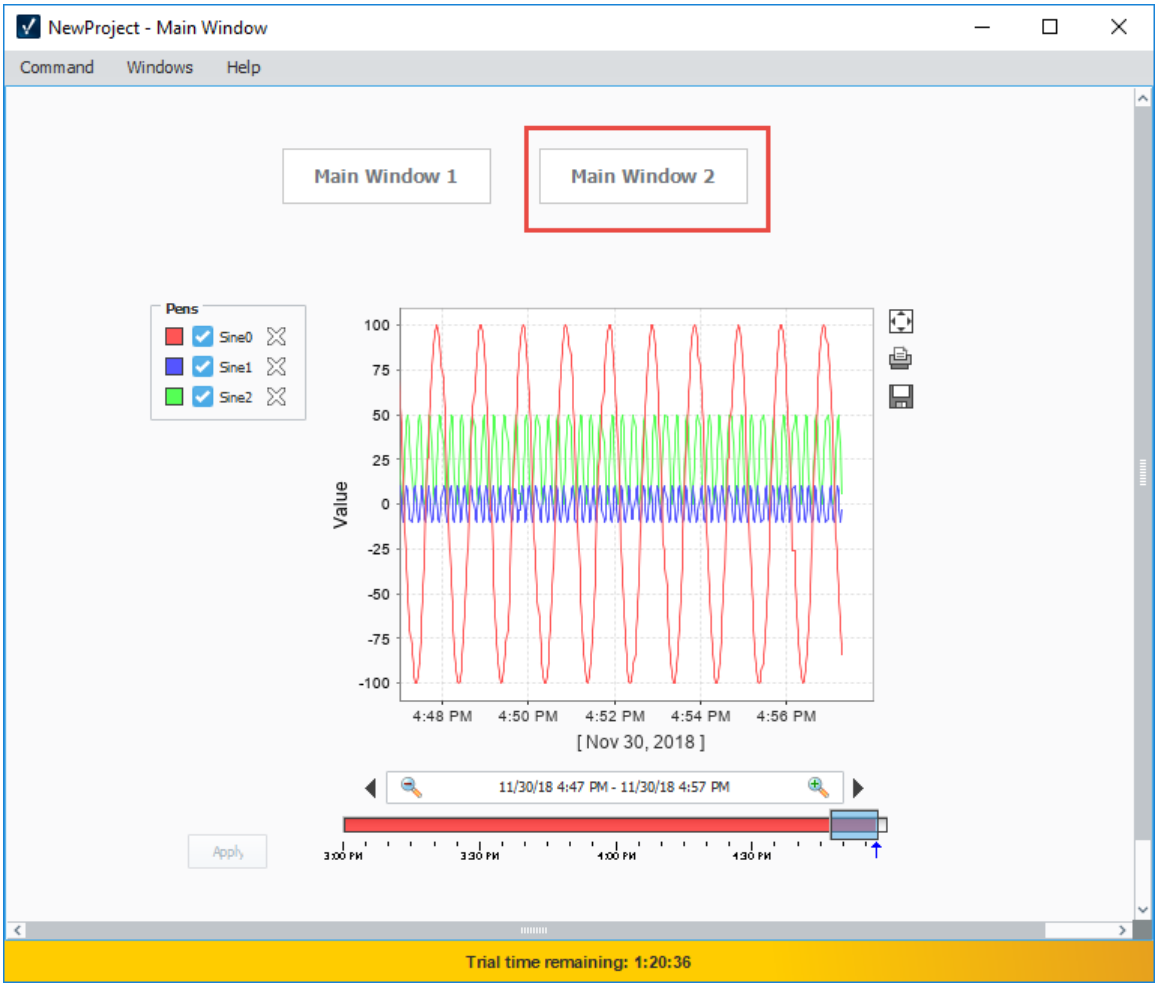
20. Click on the notification banner and a dialogue box will open stating there is a more recent version of your application available. If you want to run the update, click **Update**.



21. The Vision Client will then update itself!

You can launch as many Clients as you want! If you have other computers on the same network as the Gateway computer try launching on them too. Make sure that your Gateway computer doesn't have a Firewall enabled, or it is allowing traffic on port 8088 - the default port for the Ignition web server. The address to use on other computers is **http://ipaddress:8088** where ipaddress is the IP address of the computer Ignition is installed on.

Learn more about Vision Client Launcher!



Related Topics ...

- Component Events
- Using the Vision Easy Chart
- Configuring Tag History
- Binding Types in Vision

Installing and Upgrading Ignition

The installation process for Ignition is designed to be as simple as possible. You can find installers on our Downloads page: https://inductiveautomation.com/downloads/ignition/.

The Downloads page has the dedicated Installers (called "installer" for all major operating systems) and ZIP files. Both options install Ignition, but the installers are generally easier to use. Download an installer and head to the appropriate section below. For information on ZIP file installation instead, refer to the ZIP File Installations page.

Make sure your system meets our minimum system requirements (also found on our Downloads page).

> ⚠ If you are installing or upgrading Ignition to a version prior to 8.0.14, please see the following link.

On this page

...

- Install Ignition
- Start or Stop Ignition
- Upgrade Ignition
  - Updating License for an Upgrade
  - Upgrading a Redundant Pair
- Reset the Password
- Uninstall Ignition

Install Ignition

Installing on a Headless Server?
The steps below detail installing Ignition in a graphical environment. If you're attempting to install Ignition on a headless server, you can use the same installer, but you'll likely want to refer to the Command Line Installations page for details on command line arguments, which allow you to do things like run the installer in unattended mode.

Installing on Headless Linux Distros

When installing Ignition on a headless Linux Distro, you'll want to install the fontconfig package, as some Ignition features use the package. This must be installed prior to the Gateway for everything to work correctly.

```
sudo apt-get install fontconfig
```

1. Download an installer from our Downloads page: https://inductiveautomation.com/downloads/ignition/.
2. Run the installer and follow the steps in the installation wizard.

Linux Installations

Make the Installer Executable

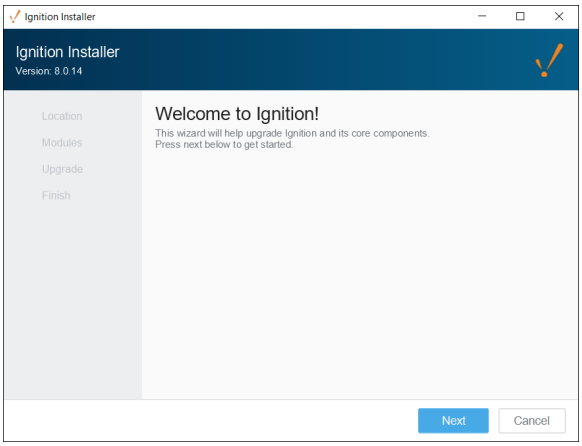
Before you can run the installer on Linux, you must first make the installer executable. From terminal:

```
$ chmod +x ignition-x.x.x-linux-xxxinstaller.run
```

Now, you can run the installer:

```
$ ./ignition-x.x.x-linux-x64-installer.run
```

> ⚠ Your desktop environment may have some graphical interface to make the installer executable. Check your desktop's documentation for more details.



In most cases you can use the default settings in the installer. This section describes what each setting does.

Default Installation Directories

This determines the installation directory. You're free to pick the directory you wish. The default directories for each operating system are listed below.

Windows

```
C:\Program Files\Inductive Automation\Ignition
```

MacOS

```
/usr/local/ignition
```

Linux

```
/usr/local/bin/ignition
```

Gateway Service Name (Windows Only)

This field allows you to choose the service name of Ignition, as it appears in the operating system's service listing.

Installation Options

Allows which modules are included in your base installation. Your choices are **Typical** and **Custom**. The Typical option will use a select list of commonly used modules. The Custom option allows you to install specific modules. In most installations the Standard option will suffice, but do note you can always add modules and remove modules after the installation process.

Start Ignition Now

Once the installer finishes, you'll be provided with an option to start the Gateway immediately. In most cases this is the desired option. You would only opt out of this choice if you didn't want the Gateway to start once the installer finishes.

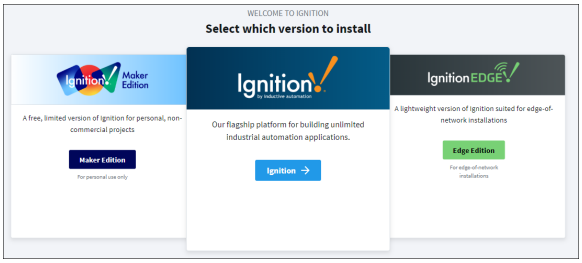
Once finished, a web browser will open, and start off the System Commissioning process.

Install As a Service (Linux and MacOS only)

Allows you to install Ignition as a service. For Windows systems, Ignition is installed as a service by default as it is a requirement for Ignition to work on a Windows server.

3. After the installation wizard finishes, a web browser will open and start the commissioning process. At this point you'll need to select your edition.

The following feature is new in Ignition version **8.0.14**
Click here to check out the other new features



Standard Ignition - Unlimited edition of our flagship platform. Designed for commercial use.

Ignition Edge - A lightweight version of Ignition suited for edge-of-network installations.

Maker Edition - A free, limited version of Ignition for personal, non-commercial projects.

4. Follow the steps in the commissioning process to complete installation.

The System Commissioning process is a series of steps that further configure your Gateway. This process begins after installation or upgrading. In addition, it can be re-triggered via a password reset with the Gateway Command-Line Utility.

Activate License (Maker Edition Only)

When installing Maker Edition, you'll be asked to provide a License Key (an 8 digit string) and an Activation Token (a considerably longer string). Both of these can be acquired for free from your Inductive Automation account: https://account.inductiveautomation.com/.

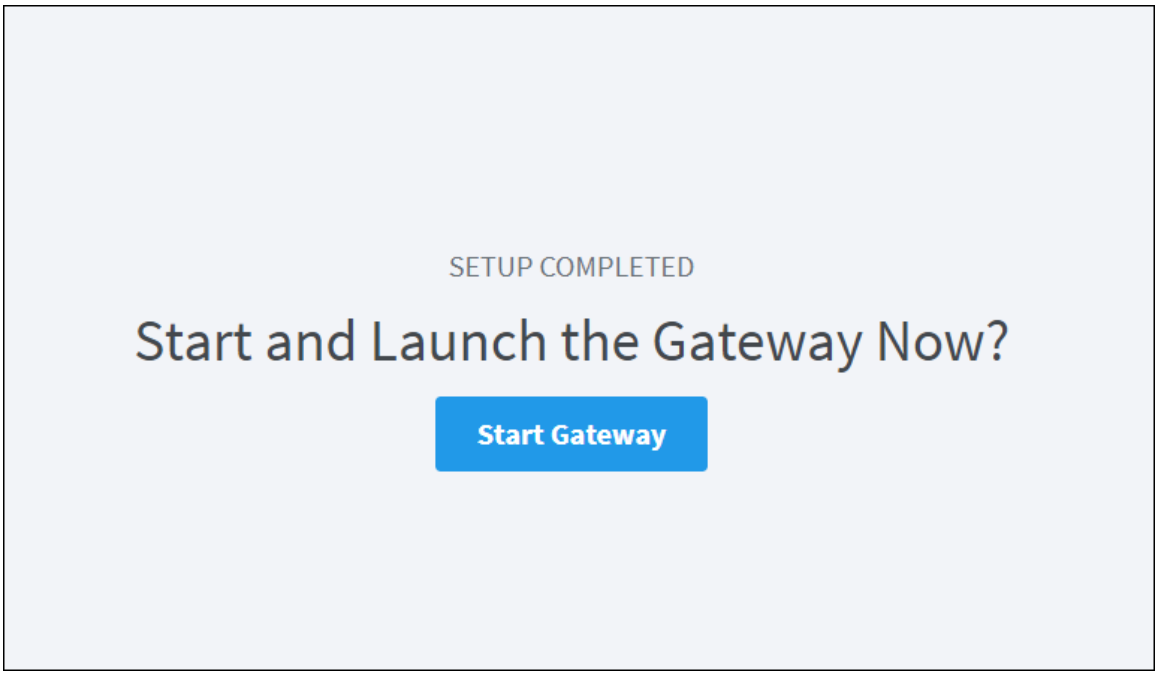
Create a User

This step asks you to create an initial user. This user will have full access to Ignition, although you're free to modify this user later. You'll need this user account to make changes to Ignition at all, so make sure you remember these credentials. Otherwise you'll need to perform a password reset: see the Gateway Command-Line Utility page.

HTTP Configuration

Allows you to specify the ports Ignition uses. By default, HTTP is **8088**, and HTTPS is **8043**.

Once you finish the commissioning process, your service will start.



Start or Stop Ignition

After installing with the Installer, Ignition will have a service representing it on the operating system. By default Ignition will start when the system starts, but you can manually start and stop the system with the following terminal commands:

Windows

The examples below assume "Ignition" is the service named used during installation (which also happens to be the default service name). You may need to modify the examples below using your system's service name.

```
net start Ignition
net stop Ignition
```

Mac

```
/usr/local/ignition/ignition.sh start
/usr/local/ignition/ignition.sh stop
/usr/local/ignition/ignition.sh restart
```

Linux

The examples below use the default installation directory

```
/usr/local/bin/ignition/ignition.sh start
/usr/local/bin/ignition/ignition.sh stop
/usr/local/bin/ignition/ignition.sh restart
```

Upgrade Ignition

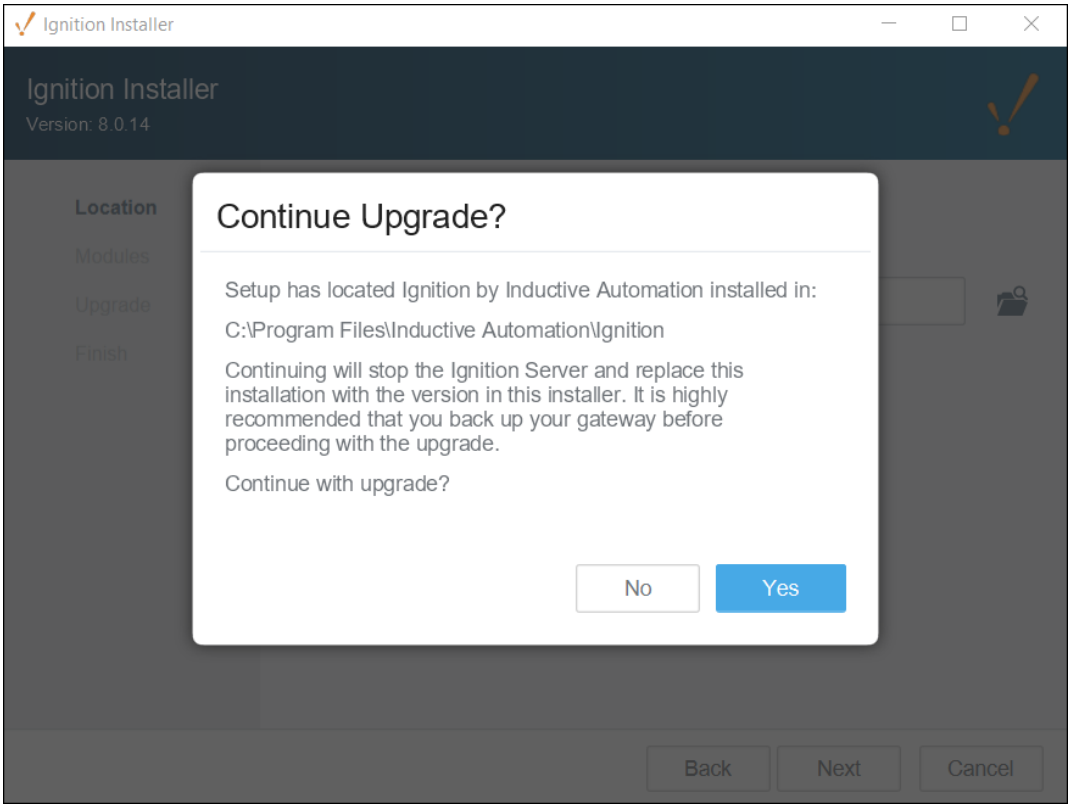
Upgrading Ignition is as easy as running the installer. When you run the installer from the same directory as a previous install, our installers are able to detect that Ignition is already installed. If Ignition is already installed, the installer will ask if you would like to proceed with the upgrade. A wizard takes you through the upgrade in a similar workflow as installation workflow.

As of 8.0.14, Ignition installers can be used to upgrade any edition of Ignition (i.e., Edge, Maker, or Standard) with the exception of ARM as that will require a zip installer for upgrading.

Linux Upgrade from 8.0.13 and Prior

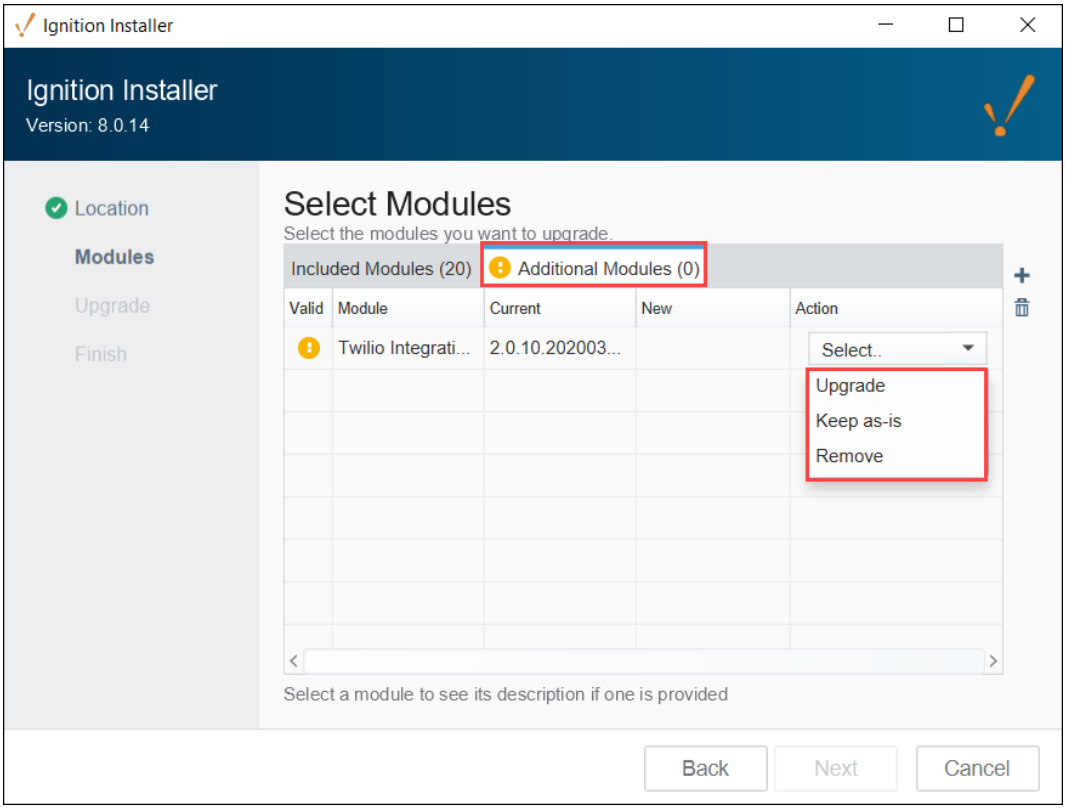
Older versions of the Linux installer would place files across multiple different directories. Running an 8.0.14+ installer will move all those files to the specified installation directory.

During the upgrade process, the Gateway will have to restart, so make sure to account for downtime when upgrading.



When you upgrade Ignition, you'll also be given the option to update any additional modules you currently have installed, including Third-Party modules. On the Select Modules screen, there is a tab for the modules included in your version of Ignition, and a tab for Additional

Modules you may be using. Under the Action column, you can choose to upgrade or remove the module or leave it unchanged.



Updating License for an Upgrade

If you are updating from one major version to the next (i.e, upgrading from version 8.0.# to version 8.1.#) , you will need an updated license. You can do this before upgrading so there is no unlicensed time after the upgrade.

1. Let us know.
   You can reach Inductive Automation by phone at **(800) 266-7798**.
   Some Major upgrades include new modules or have modified the functionality of pre-existing modules. If your license needs to be updated, call us and we can guide you through it. If you are updating from one version to the next, or from 7.0.0 all the way to the latest version, we can talk you though what is different and what new functionality is available. See the New in this Version page for more specifics about what was updated in the various versions.

   > **Note:** If your license is covered by a support plan, you have Upgrade Protection which automatically keeps your platform and module versions up to date in our system. Refreshing your license on your Gateway will synchronize the license on your server with the updated license on our activation servers.

2. Reload your license.
   Once your license is updated in our system, go to your Config section of the Gateway webpage, and click System > Licensing. Click on the Update License icon .
   For for a detailed guide on reloading a license, see updating a License on this page. If you are coming from an older version, the Gateway may look different. Don't worry, all this functionality is still there, but you may have to un-activate then re-activate instead if the Update License button is missing.

Upgrading a Redundant Pair

There are a few extra things to consider when upgrading a redundant pair of Gateways. When performing the upgrade locally, we recommend upgrading the Master first. This allows the Backup to take over when the Master is down for an upgrade, and the Master can resume operations when it comes back online. At this point the Backup can be upgraded without interruption. The same considerations mentioned above should also be applied to the redundant pair, but you only need a Gateway Backup file from the Master Ignition Gateway.

If the upgrade will be performed remotely using the Enterprise Administration Module, then the Backup Gateway should be upgraded first. This allows the EAM controller to manually tell clients/sessions to communicate with the newly upgraded Backup Gateway, even though the Master will be a different version, allowing time to upgrade the Master.

Reset the Password

To reset the main password for the user with full privileges, use the Gateway Command-line Utility (gwcmd) utility. The 'gwcmd' utility triggers system commissioning, and prompts the user to reset the username and password for the main user (with full privileges) after the Gateway restarts. Linux and Mac OS platforms also use the 'gwcmd' utility for password reset. For more information, see the Gateway Command-line Utility page.

Uninstall Ignition

> ⚠ Uninstalling Ignition will remove any projects or Gateway level configurations you made. If you wish to preserve anything, make sure to take a Gateway backup first.

To uninstall Ignition, run the uninstaller file. This file can be found in the main Ignition installation directory, for example:

Windows
`C:\Program Files\Inductive Automation\Ignition\Uninstall.exe`

Mac
`/usr/local/ignition/Uninstall.app`

Linux
`/usr/local/bin/ignition/Uninstall.run`

Command Line Installations

Command Line Installations

Command Line Installations

The content on this page applies to Ignition installer versions 8.0.14 and up. Documentation on command line installations for earlier 8.0 versions can be found on the legacy 8.0 install pages:

Windows - Install

Linux - Install

Mac OS X - Install

Command line argument keys can be used to modify the installation process. Each operating system supports launching with arguments that contain some small modifications to ensure they are passed through to the installer:

- Windows: a `--%` arg should go before installer args
- Mac: requires `open -a` command with `--args` before the args to pass to the installer
- Linux: a `--` arg should go before installer args

Arguments should be quoted.

Arguments Table

The table below lists the supported arguments.

Command Line Argument Key	Description	Possible Values	Example
unattended	The unattended install mode to run the installer in. It runs with all default values unless otherwise specified The unattended upgrades remove known IA modules that are deprecated, leaves third party as they are, and upgrades currently installed bundled modules. Additionally, unattended upgrades do not trigger the Ignition service to start automatically. The Ignition service **must** be started by a user. The argument also works with the uninstaller, allowing for unattended uninstalls.	• graphical - run with full size installer window but no interaction • minimal - shows a progress bar only unless errors occur • text - feedback on the command line only, no GUI • none - completely silent, no feedback except for the log file when complete regardless of success or fail	"unattended=minimal"
textMode	Whether to run this installer in *interactive* text mode regardless of graphical support. Note that **unattended** supersedes this value.	true - run in text mode	"textMode=true"
location	Overrides the default installation directory, allowing you to specify where Ignition should be installed.	a fully qualified file path	"location=/usr/local/ignition"
logLevel	The logging level for the installer. The default is INFO, however more detail can be provided, revealing more information about the install process, as wells as the installer's internal operations.	Options are (from most granular to least granular): • trace • debug • info • warn • error	"logLevel=trace"

user	Sets the ownership of the install to this user.	The username (as recognized by the operating system) that should be the owner of the installation.	"user=myUser"
serviceName	Sets the service name for the gateway. Defaults to "Ignition". When running the graphical installer, the service is installed immediately on Windows and set for optional future installs on Linux/Mac When using this key with the unattended key during an upgrade, the existing service on windows will be removed, and replaced by a new service.	my-gateway	"serviceName=my-gateway"

Example Commands

The following examples demonstrate how arguments can be used when running the installer from. For the purposes of these examples, any mention of a user directory will use "**yourUser**", and the Ignition version number in the installer name will be "**8.0.14**". These should be replaced by appropriate values when attempting to run the installer on your system.

Windows

```
# Runs the installer on command line only and requires user interaction
C:\Users\yourUser\Downloads\ignition-8.0.14-windows-x64-installer.exe --% "textMode=true"

# Runs the installer on command line, changes the default install directory, and does not requires user
interaction
C:\Users\yourUser\Downloads\ignition-8.0.14-windows-x64-installer.exe --% "unattended=text" "location=C:
\Ignition"
```

Mac

```
# Runs the installer on command line only and requires user interaction
open -a /Volumes/IgnitionInstaller/ignition-8.0.14-osx-installer.app --args "textMode=true"

# Runs the installer on command line, changes the default install directory, and does not requires user
interaction
open -a /Volumes/IgnitionInstaller/ignition-8.0.14-osx-installer.app --args "unattended=text" "location=
/Users/yourUser/Desktop/test"
```

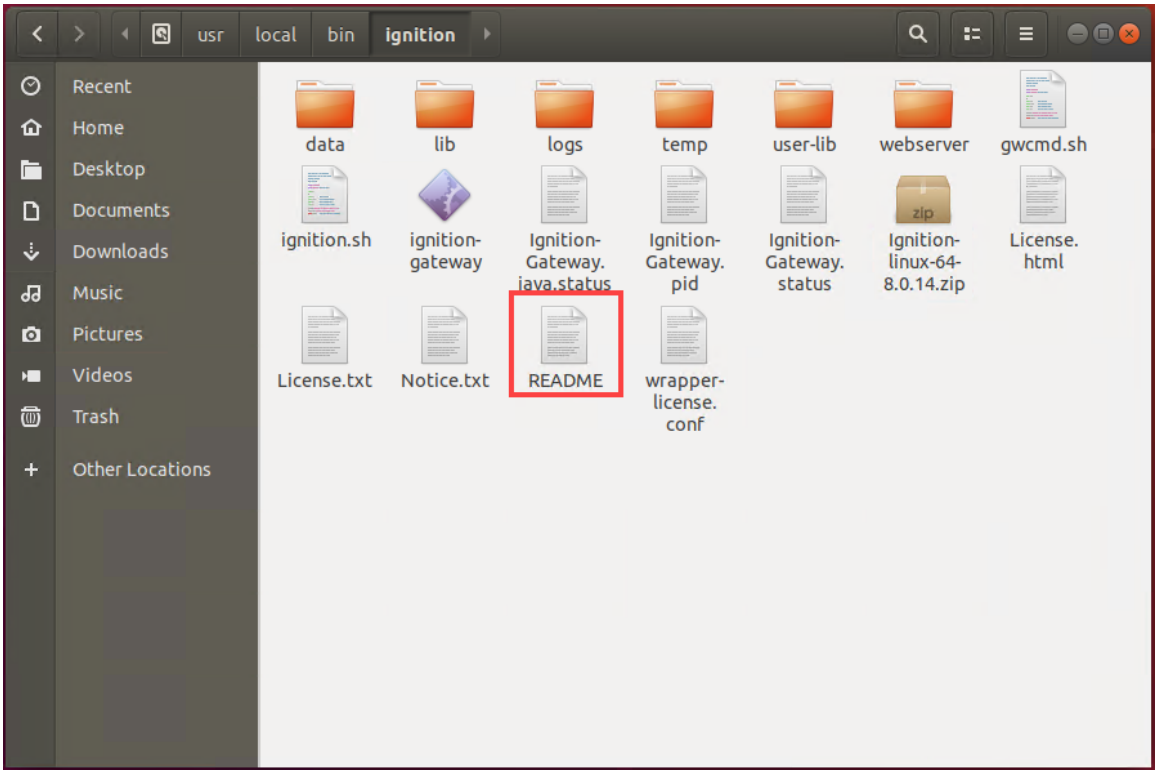
Linux

```
# Runs the installer on command line only and requires user interaction
"/home/yourUser/Downloads/ignition-8.0.14-linux-x64-installer.run" -- "textMode=true"

# Runs the installer on command line, changes the default install directory, and does not requires user
interaction
"/home/yourUser/Downloads/ignition-8.0.14-linux-x64-installer.run" -- "unattended=text" "location=/home
/yourUser/Desktop/test"
```

ZIP File Installations

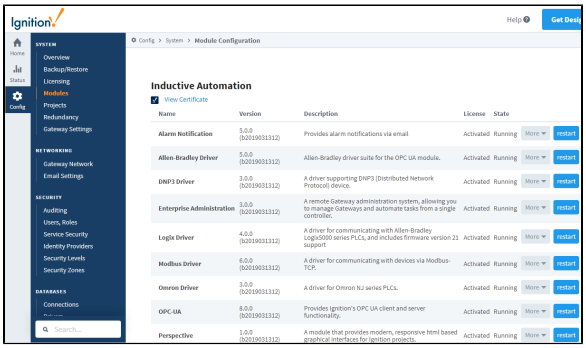
Aside from the graphical installers, our downloads page features ZIP files that can be used to install and run Ignition. The ZIP file installation process differs from the graphical installation process. Using the ZIP files to install involves following the included **README** file, which contains instructions on how to make use of the ZIP file contents.



Installing or Upgrading a Module

Module Configuration

All module configuration is done from Gateway Webpage in the **Config > Modules** section. From here you can add, remove, and restart modules. Modules integrate their settings into the Gateway configuration tree, and therefore do not offer settings in this section.



On this page

...

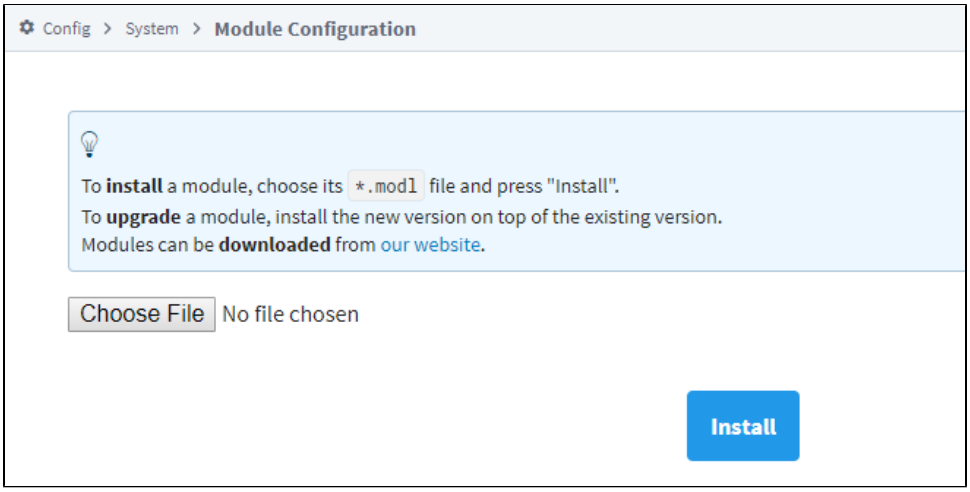
- Module Configuration
- Install or Update a Module
- Uninstall a Module
- Restart a Module
- Module Status

Installing or Upgrading a Module

Watch the Video

Install or Update a Module

1. On the Gateway Webpage, select **Config > Modules** to open the **Module Configuration** page.
2. Scroll to the bottom on the list, find the blue arrow, and click the **Install or Upgrade a Module** link.



3. Click **Choose File**, select a `.modl` file that you have previously downloaded.

   Go to the Inductive Automation downloads page. Search for the module you want and save it to your Downloads folder on your computer.

   Third Party Modules
   All third party Ignition modules require the Ignition platform to be installed.

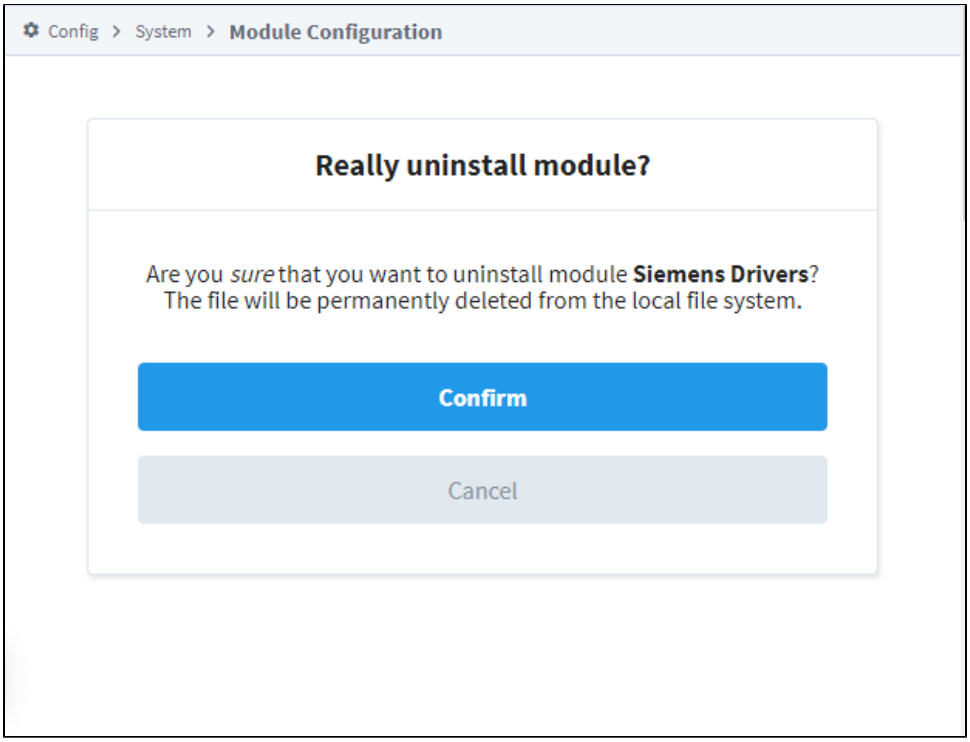
   To install third party modules:
   - Install Ignition: See Installing and Upgrading Ignition
   - Once Ignition is installed, download the module and install it in the Ignition Gateway: See Installing or Upgrading a Module

SepaSoft, Inc. MES Modules for Ignition		
OEEDT Installer Module	**OEEDT-Installer-module.modl** (8MB)	Version: 1.9.0.1610111600
SPC-Installer-module	**SPC-Installer-module.modl** (10MB)	Version: 1.9.0.1610111606
Instrument Interface-module	**Instrument-Interface-module.modl** (1MB)	Version: 1.9.0.1610111601
Recipe-Installer	**Recipe-Installer-module.modl** (8MB)	Version: 1.9.0.1610111609
Productionsimulator-Installer	**Production-Simulator-module.modl** (58KB)	Version: 1.9.0.1610111559
Trace-Installer	**Trace-Installer-module.modl** (7MB)	Version: 1.9.0.1610111555
Web Service-module	**Web-Service-module.modl** (2MB)	Version: 1.9.0.1610111611
Barcode Scanner-module	**Barcode-Scanner-module.modl** (61KB)	Version: 1.9.0.1610111610

   Go back to the **Install or Upgrade Module** screen and browse your Downloads folder for your module, select it, and click **Open**. Ignition will copy the path to the module to the Browse field and all that's left for you to do is click **Install**.
4. Click **Install**. When the page reloads you can now see the module you installed in the list of modules.

Uninstall a Module

1. On the Gateway Webpage, select **Config > Modules** to open the **Module Configuration** page.
2. Scroll to the the module you want to uninstall.
3. To the right of module name, click the **More button** and select uninstall. A confirmation message appears.



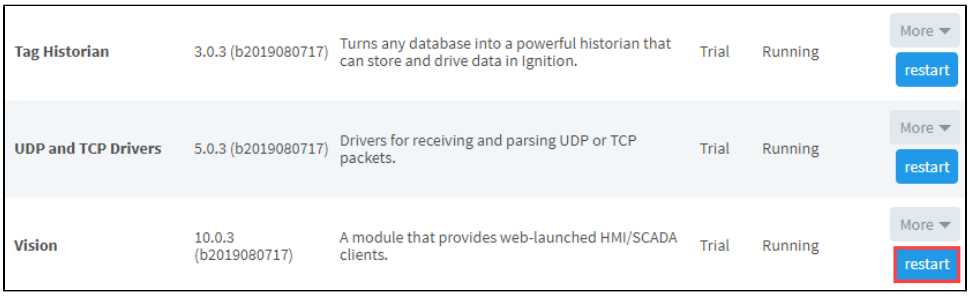
4. Click **Confirm**. The module is no longer installed and is removed form the list.

Modules are hot-swappable, this means these actions can be performed while the system is running. The isolated nature of modules ensures that performing one of these actions only affects that particular module, and any modules which depend on it. For example, uninstalling the SQL Bridge module will not affect any running Vision module clients.

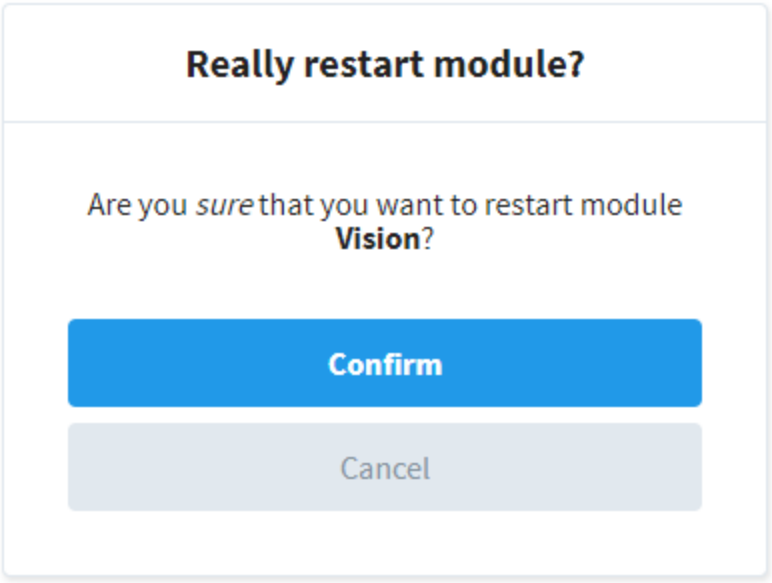
Restart a Module

Because of the isolated nature of modules, the other modules are not affected by the restart (unless they depend on that particular module).

1. On the Gateway Webpage, select **Config > Modules** to open the **Module Configuration** page.
2. Scroll to the the module you want to restart.
3. To the right of module name, click the **restart** button.



4. Click Confirm at the confirmation message.



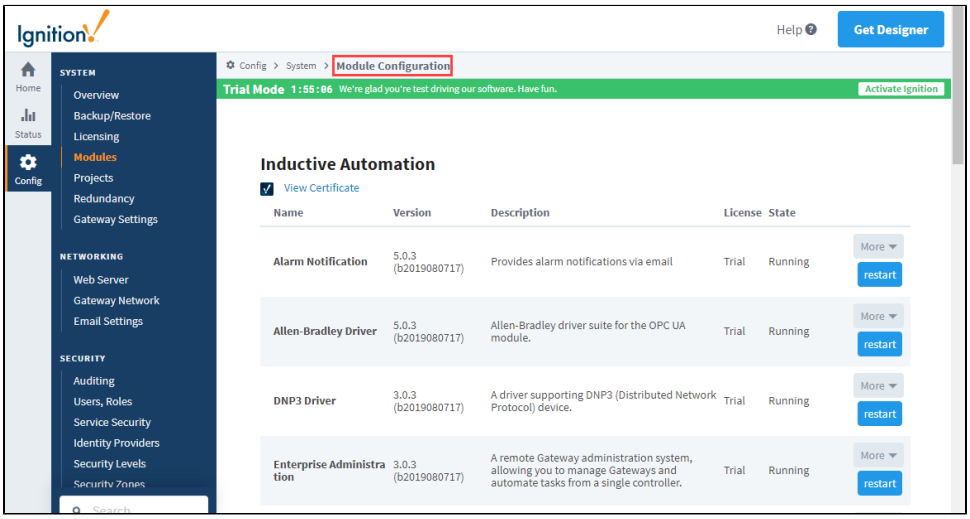
Ignition will give you a confirmation message when the module has restarted.

Module Status

The installed module list on the **Module Configuration** page also provides some basic information about the **State** of the module. The **Versi on**, **License**, and **State** columns are all displayed in the list. Module licensing is performed centrally in **System > Licensing**, so the values here are only for information purposes.

The module State can be either Running or Loaded:

- Running indicates the module programs are actively performing tasks, for example, the Alarm Notification modules shows as Running because it is continually checking for condition changes that need to set the alarm on or off.
- Loaded indicates the module program or resources are available but nothing is running, for example, the Symbol Factory module makes all the symbol images available for your use.



Related Topics ...

- Licensing and Activation

Ignition 8 Upgrade Guide

Upgrading from 7 to 8

Introduction

Ignition 8.0 is a major update to the Ignition platform, representing over 18 months of work across virtually all subsystems and modules. Throughout the process, Inductive Automation has strived to maintain its commitment to backwards compatibility and a straight-forward upgrade process. However, given the vast scope of changes in 8.0, there are inevitably special considerations that need to be evaluated and actions taken in order to successfully upgrade an existing system. This document seeks to identify those issues and provide guidance for users who are migrating from an Ignition 7 system. To discover the new features of Ignition 8, please see the What's New section.

> ⚠ ***In some cases, it is not advisable or possible to upgrade from Ignition 7 to 8 at this time. Please read this guide carefully before upgrading a production gateway!***

Should I Upgrade?

It may not be advisable for all users to immediately upgrade production systems to 8.0. It is highly recommended that all users who wish to upgrade should first install Ignition 8 on a separate computer and validate all of the functionality required by their system before deploying it into production. Please read this guide in its entirety to understand the changes introduced in Ignition 8 and evaluate if they will impact your installation(s).

Expected Downtime

If you are upgrading from Ignition 7.9 or before to Ignition 8, there are some changes that may require a downtime window, even when using Ignition's Redundancy. These changes are covered in the rest of the upgrade guide. For example, for anyone using Vision, all Vision Clients will need to be updated to run using the new Vision Client Launcher, which will need to be installed on each client PC.

Changes to the Release Cycle

Updates to Ignition 8 will be made according to a new release cadence that targets smaller, more frequent, and more predictable releases. The numbering methodology will continue to be the same as before (eg. 8.0.1, 8.0.2), but will now occur at regular intervals. At any given time, multiple branches will be available on the downloads page: Stable, RC, and Nightly Beta.

As part of this effort, we will also be improving the Change Log section of the website, and will strive to provide more information to users in regards to upcoming changes and known issues.

Please see the More Information section at the end of this document for links to various resources where update information will be published.

Using Perspective without Upgrading

The Ignition Perspective module is one of the most exciting new features of Ignition 8, offering an entirely new and innovative way to visualize data and deploy applications to mobile devices. Many users will want to begin leveraging this module right away, and will feel compelled to upgrade for that purpose.

It is important to understand how various Ignition features, in particular the gateway network and remote services, can be leveraged to allow immediate usage of Ignition Perspective without the potential risk or disruption necessary to upgrade a running production gateway.

As part of the base platform, the Gateway Network allows you to create connections between gateways, including between 8 and 7. The Remote Tag Provider allows you to use tags from one

On this page ...

- Upgrading from 7 to 8
  - Introduction
- Should I Upgrade?
  - Expected Downtime
  - Changes to the Release Cycle
- Using Perspective without Upgrading
- Important Changes
  - Discontinued Features
    - Vision Mobile Module Discontinued
    - External Database "SQLTags" Providers No Longer Supported
    - 32 bit Support
    - Project Rollback
    - OPC UA Server Stale Threshold
  - Currently Incompatible Changes
    - Sepasoft Modules
    - Cirrus Link Modules
    - 3rd-party Modules
    - Staging & Publishing
    - OEM Lock
    - OPC UA Server Exposed Tags Auditing
    - OPC UA Server "ReadOnly" Role
  - Changes That Require User Action
    - Look and Feel changes
    - Redundancy Connection Configuration
    - Replacement for the system.tag.browseConfiguration Function

gateway on another, including both realtime and historical values. Remote Tag Providers created in Ignition 8 targeting Ignition 7 gateways will support reading, writing, real-time subscriptions, and tag history queries. Tags on a version 7 gateway cannot be modified or edited from a version 8 remoted tag provider. However edits and modifications from their host gateway/designer will be reflected in both gateways immediately.

Therefore, to develop and deploy Perspective applications with minimum impact on your production system you can:

1. Install Ignition 8 onto a different computer or VM instance.
2. Limit the installation to the Perspective module (though, of course, other modules can be used)
3. Create a gateway network connection between the Ignition 8 gateway and the Ignition 7 gateway.
4. Create a Remote Tag Provider on the Ignition 8 system for each provider you'd like to access.

Unless previously changed, remember that the Ignition 7 gateway will employ the default "service security" settings for the remote tag gateway network service. This will allow read only data, but if you want to support writes, you'll need to change the policy by going to Configure>Security /Service Security in the gateway.

- Changes to Be Aware Of
  - Introducing Ignition Perspective
  - Introducing Identity Providers
  - System Commissioning
  - Java Web Start Discontinued
  - Embedded Java
  - Project Inheritance & "global" upgrade logic
  - Project Resource Storage
  - Designer Concurrent Editing and Conflict Resolution
  - Deprecated system.tag Functions
  - Tag System Changes
  - UDT Performance
  - UDT Parameter Usage
  - Quotes Taken Literally in String UDT Parameters
  - The instanceName UDT Parameter
  - Script Based Editing
  - OPC UA Certificate Management
  - OPC UA Default Configuration Changes
  - OPC UA Discovery with Third-Party Clients
  - OPC UA Exposed Tags NodeIds
  - Changes to system.alarm.listPipelines()
  - Changes to Gateway Command Utility (GCU)
- More Information

Important Changes

This section describes four levels of changes that users should be aware of when upgrading a system:

- Discontinued features
- Currently incompatible changes
- Changes that require user action
- Changes to be aware of

Discontinued Features

These are areas that have been removed or discontinued, as new methodologies have been adopted to replace the functionality.

Vision Mobile Module Discontinued

The Vision Mobile Module is no longer supported in Ignition 8. The new Ignition Perspective Module provides a robust, modern solution for mobile visualization. Users should migrate their mobile visualization needs to be based on Perspective applications going forward.

A few important points:

- The Mobile Module will continue to be supported on Ignition 7.7 and 7.9 for the remainder of their respective Long Term Support (LTS) terms.
- All customers who own the Mobile Module and have current TotalCare Upgrade Protection contracts are entitled to a license of Perspective Mobile upon upgrade.

External Database “SQLTags” Providers No Longer Supported

The “Database Provider” and “Database Driving Providers” are no longer supported in Ignition 8. These systems were very early ways to bridge outside data into Ignition’s tag system and have been replaced over the years with much easier and faster methods of data integration.

32 bit Support

Support for 32-bit platforms other than ARM has been discontinued.

Project Rollback

The Project Rollback feature has been discontinued in Ignition 8.

OPC UA Server Stale Threshold

The Stale Threshold setting on Ignition's OPC UA server has been retired. The setting was originally used in earlier versions of Ignition as a means to help some of our drivers determine when values should be marked as stale. Our drivers have since gaining their own methods to determine staleness. The setting has little use in Ignition 8, so it was removed.

Currently Incompatible Changes

These are areas that are not yet fully available or compatible in 8.0.0, but that we expect to progress soon. Please see the references in “More Information” to track how these develop.

Sepasoft Modules

Please follow this link to Sepasoft’s website for information about module compatibility with Ignition 8. https://help.sepasoft.com/docs/display/MHD/Ignition+8+Readiness

Cirrus Link Modules

Please follow this link to Cirrus Link’s website for information about module compatibility with Ignition 8. https://docs.chariot.io/display/CLD80/Ignition+Compatibility+with+Cirrus+Link+Modules

3rd-party Modules

Please consult individual module authors concerning upgraded module availability. Modules built for Ignition 7 ***will not work*** in Ignition 8 without modification.

Staging & Publishing

Ignition 8 features a completely revamped project system. This new project system does not have different “staging” and “published” versions of each project. When you upgrade from Ignition 7, any projects that were split into separate staging and publish versions will have their stagi ng versions imported into Ignition 8. This is to prevent any loss: the staging version is always at least as and usually more up-to-date than the published project.

Staging and publishing was not included in Ignition 8’s project subsystem for two reasons:

1. Staging & Publishing was not consistently implemented across various modules, leading to confusion about which types of resources participated in it.
2. Projects are not self-contained: they rely on tags and external connections to devices, databases, etc. For this reason, the staging & publishing feature was only effective in limited circumstances.

We plan on bringing back the idea of staging and publishing in the near future, except that it will be a more comprehensive approach that will affect entire gateways. Many users already use multiple gateways to create separate environments: this is a concept that we can make more powerful by building support for it into the platform. In the meantime, you can emulate the old staging & publishing system by simply having a “staging” copy of your project and either move resources between projects or copy & overwrite them in order to “publish” your changes.

OEM Lock

The “OEM-Lock” feature that was used to encrypt project resources by authors is not currently supported in 8.0. An updated version of this feature will be introduced at some point. Project authors that used this feature may migrate their projects to 8.0, as the owner license will decrypt Ignition 7 resources, but they should be aware that the resources will currently remain unencrypted.

End users who upgrade OEM-Locked systems to 8.0 will not be able to view the project resources.

OPC UA Server Exposed Tags Auditing

When a tag made available via the “Exposed Tags” feature of Ignition’s OPC UA server is written to by an OPC UA client the write is not currently entered into the audit log. This will be re-implemented in a future version.

OPC UA Server “ReadOnly” Role

The undocumented “ReadOnly” role no longer has any effect. First class support for role-based access control will be implemented in a future version.

Changes That Require User Action

These changes will require manual action on the part of the system administrator/project author. In other words, these are features or situations that have a direct corresponding solution, but for various reasons cannot be automatically upgraded.

Look and Feel changes

The general look and feel of the Ignition Designer and Vision Client has been updated. This may result in subtle changes to the appearance of Vision Clients depending on the components used by a given project. It is not possible to revert to the old “look and feel” as it’s not compatible with Java 11.

This means screens that have been developed in version 7 may look slightly different after upgrade. Manual configurations to Vision resources, such as manually selecting a font or background color for a component, will be preserved upon upgrade. However, visual properties left with default values may appear differently after upgrade.

It is highly advised to upgrade version 7 projects to version 8 on a test server and verify the look of the project, before upgrading the production servers.

Redundancy Connection Configuration

Redundancy communication is now accomplished through a Gateway Network connection between the master gateway and the backup. This mechanism provides greater security and performance than the previous communication channel, but will require the system administrator to re-establish the connection.

The settings for the communication channel are still configured in the gateway under Configure>Redundancy, and are still established from the Backup to the Master, but must be modified before the backup will successfully connect to the master after upgrade.

Replacement for the system.tag.browseConfiguration Function

The tags system in Ignition changed dramatically in Ignition 8, which makes the Ignition 7 function system.tag.browseConfiguration obsolete. The function has been removed from the code base, so scripts using the old browseConfiguraiton function will break upon upgrade. The functionality has been replaced by system.tag.getConfiguration.

Changes to Be Aware Of

Ignition 8 has many additional changes and improvements that system administrators should be aware of. Although none of them present a significant issue, some may directly affect end users (for example, launching clients), and some can be leveraged to improve the security and performance of your system.

Introducing Ignition Perspective

Ignition Perspective is an entirely new visualization system, built from the ground up to allow you to create modern, high performance, and highly functional pure-web applications that can be viewed anywhere. Learn more about Perspective by visiting the Perspective pages in the User Manual, or the Inductive University.

Introducing Identity Providers

Ignition 8 adds support for connecting to SAML and OpenID Connect Identity Providers. In addition, Ignition can act as an Identity Provider, using a User Source as a backing source of users. On upgrade, a new Ignition Identity Provider will be created that points to a single user source, allowing users in that user source to authenticate against Perspective sessions. You can learn more about this on the Configuring Identity Providers page.

System Commissioning

Upon installing Ignition, the user is now presented with a “system commissioning” web based walkthrough. This process must be completed for new installations, and requires that the administrator create a root username and password combination, specify the ports that Ignition will use, and accept the appropriate licenses.

As of Ignition 8.0, there is no longer a pre-defined default “admin” user. System administrators should choose a secure root username and password for optimal system security.

Java Web Start Discontinued

Support for Java Web Start has been discontinued by Oracle and the OpenJDK community as of Java 11. This technology has been used in Ignition since its original release to launch the Designer and Vision clients, though we have also offered alternative launching tools for years.

In anticipation of this change, the designer and client launching tools have been greatly improved in Ignition 8. In addition to a fresh and more functional UI, the launch tools now manage the Java runtime environment used by the client (see next point), and can help you track many projects and gateways. There are separate launchers for Vision Clients and the Designer, representing the natural distinction between the target users of these products, and the launchers can be downloaded for Windows, Mac, and Linux directly from the gateway homepage.

Java Web Start is no longer supported, please use the Designer and Client launchers to launch those applications.

Embedded Java

Ignition no longer requires Java to be installed on the gateway or client machines. Native versions of the Ignition installer and launchers have been made available for 64-bit Windows, Mac, and Linux, as well as ARM, for use in edge devices. These distributions include embedded JREs (Java Runtime Environment) that are provided by Inductive Automation and managed by Ignition.

The native client launchers manage retrieving the current version of the JRE as necessary for the client and designer runtimes. Updates to the embedded JREs are managed by Inductive Automation and are made available through updates to Ignition. The client launchers will automatically update their local versions as necessary, ensuring that Vision Clients and the Designer are always running the latest version available on the Gateway.

Ignition no longer requires Java to be installed on server or client machines. System administrators should monitor Inductive Automation for important system and security updates, and can sign up for important notifications at http://support.inductiveautomation.com

Project Inheritance & “global” upgrade logic

The project system in Ignition 8 has been improved and now features a more flexible inheritance model. Each project may have a “parent” project, and will inherit all of the resources of that parent project. That project may in turn have its own parent project, and so in this way complex hierarchies of re-usable resources may be designed. Within each project, inherited resources may be used by other, “local” resources. For example, an inherited Vision template could be embedded in a window, or an inherited script could be executed by a button. Inherited resources may also be overridden, allowing them to be re-defined when needed. Conversely, resources in parent projects may be marked as “not overridable” to ensure that children cannot alter them, when appropriate.

This new inheritance system makes Ignition 7’s “global” project concept obsolete. In order to maintain backwards compatibility, the following will happen when an Ignition 7 Gateway is upgraded to Ignition 8:

- The resources from the Ignition 7 special “[global]” project will be copied into an Ignition 8 standard, inheritable project called “global”.
- All other projects will have their parent project set to this new “global” project.
- All “shared” scripts are now simply project-level scripts in a top-level script package called “shared”, stored in the new “global” project. This allows all existing scripts to function without edits.
- All “project” scripts are now simply project-level scripts in a top-level script package called “project”.
- All Vision templates stored in the old global area are moved into a folder called “SharedTemplates/” in the new global project. The vision module will automatically convert a template path like “[shared]MyTemplate” as “SharedTemplates/MyTemplate” so that no further action is needed.
- If any run-always sequential function charts are present in the global area, a new project called “run-always-charts” will automatically be created upon upgrade, and these charts will be moved into this project. This is because resource types that execute automatically like run always SFCs and Transaction Groups do not run in inheritable projects.
- If any alarm pipelines exist, a project called “alarm-pipelines” will be created and all alarm pipelines will be moved into this project. Tags can target pipelines in any project, but in the absence of a specific project, the pipeline is assumed to be in this special upgrade project.
- There is a new gateway-wide setting called “Gateway Scripting Project”. This setting lets you specify one project which all scripts that are outside the scope of a project will execute against. For example: Tag Change Scripts and Tag expressions using the runScript expression. On upgrade, this will be set to the new “global” project so that Tag scripts can continue to access shared.* scripting functions.

Project Resource Storage

The new project subsystem in Ignition 8 uses a new file-based storage strategy, instead of the internal-database strategy used in Ignition 7. This has several significant benefits over the previous methodology:

1. All resources can now be tracked with industry standard source control tools, such as Git. Resources can be tracked individually, and changes to disk caused by source control operations (checkout, pull, etc) will be applied automatically to the running system.
2. Project resources can be transferred between systems with simple file system operations. System administrators can use a wide range of tools to synchronize all types of resources between Gateways.
3. Project backups are simply standard zip files organized in the same way as the file system.

Designer Concurrent Editing and Conflict Resolution

Ignition 8’s Designer uses a new lock-free strategy for handling concurrent editing. We believe that this strategy is significantly more robust and will allow for more designers to work together on projects.

The Ignition 7 locking strategy (besides being frustrating when users would lock resources and then go home for the day!) is incompatible with the new filesystem storage strategy. Since changes can occur in the filesystem at any time, it isn’t possible to maintain exclusive edit locks in the Designer.

In Ignition 8, you may open and modify any resource in your project. You will be notified if the system detects that you and another user are both editing the same resource, or if the resource you’re editing has already been modified remotely. If you modify the resource anyway, and someone else has modified it first, this is called a conflict. The system automatically detects conflicts when you try to save. If any are detected, you are given the opportunity to resolve these conflicts by choosing whether to use your changes, accept the changes that happened elsewhere, or cancel your save and figure out what to do in another way.

Deprecated system.tag Functions

The system.tag.read and system.tag.readAll functions were deprecated. They has been replaced by system.tag.readBlocking.

The system.tag.write and system.tag.writeAll functions were deprecated. They have been replaced by system.tag.writeAsync.

The system.tag.writeSynchronous and system.tag.writeAllSynchronous functions were deprecated. They have been replaced by system.tag.writeBlocking.

Tag System Changes

The Tag system has undergone significant changes in Ignition 8. The majority of these changes should go unnoticed upon upgrade (apart from obvious UI changes), and most systems should immediately benefit from improved performance, faster edits, and expanded functionality. Under the hood, however, many technical aspects have changed. Understanding these changes are important for design and troubleshooting in Ignition 8.

UDT Performance

The tag system has been redesigned to optimize for UDT oriented design. In Ignition 7, certain aspects of UDT performance led some system designers to favor standard Tags over UDTs for some applications. Such systems should be re-evaluated in Ignition 8, as use of User Defined Types will lead to more efficient memory usage and lighter weight change processing.

UDT Parameter Usage

As mentioned in the section Changes That Require User Action , the way that Parameter references are applied to Tags has changed. Instead of being replaced in a “pre-compilation” stage of Tag execution, they are now treated as true property references. This means that they can be modified, and those modifications propagate dynamically to referencing properties, causing those to be updated in turn.

An important consequence of these changes is that syntax for parameter references in Expressions has changed in a subtle, but crucial way. In Ignition 7, due to the replacement strategy, the following would have been a valid expression:

```
“The value of the parameter is: {ParamRef}”
```

In Ignition 8, this value would no longer be a valid reference. It is counter-intuitive to all other uses of reference in Ignition, and has been replaced with a more familiar syntax:

```
concat(“The value of the parameter is: “, {ParamRef})
```

All existing references will be upgraded automatically to the new syntax upon system upgrade. Designers should be aware of this change when creating new Tags.

Quotes Taken Literally in String UDT Parameters

In addition, string UDT parameters that contained quotation marks in 7.9 will now be represented literally. In 7.9 the quotation marks were omitted. After upgrading, the quotation marks will be included in the UDT parameter value.

The instanceName UDT Parameter

The built-in `instanceName` UDT parameter works differently in nested UDTs. Referencing the parameter in a nested UDT will return the name of the nested UDT. In 7.9 the parameter would always return the name of the root UDT.

Script Based Editing

The various scripting functions for editing Tags (system.tag.editTag(s), addTags) have been deprecated and replaced by system.tag. configure() . The previous functions will continue to work, but designers should learn the new function, which is far more intuitive and powerful than the previous functions.

Tags are now defined as JSON objects, which consist of properties, arrays, and sub-objects. The system.tag.configure function can take either a String document definition, or a JSON object that defines one or more Tags. Overrides for UDTs are created by simple redefinition of properties, and complex structures like Event Scripts and Alarm configurations will be merged with inherited definitions.

OPC UA Certificate Management

Prior to Ignition 8, Ignition’s built-in OPC UA client implicitly trusted the certificate of any server it connected to, and Ignition’s built-in OPC UA server implicitly trusted the certificate of any client connecting to it. This is no longer true. New certificate management pages for the client and server have been added to the configuration section of the Gateway under “OPC UA > Security”. From this UI, trusted certificates can be imported and quarantined certificates can be marked as trusted. Ensuring the remote certificate is trusted is required for all secured inbound and outbound connections.

On upgrade, all secured inbound and outbound connections other than the default “loopback” connection will be faulted until the remote certificate is explicitly marked as trusted.

OPC UA Default Configuration Changes

New installations have some new OPC UA related defaults:

- The server binds only to localhost (i.e. no remote connections allowed)
- The server uses port 62541 instead of 4096
- The server only allows secured connections (SecurityPolicy Basic256Sha256)
- The default “loopback” connection is called “Ignition OPC UA Server” (there is no longer a “-” between “OPC” and “UA”)

Upgrades are not affected by these changes; they retain their existing configuration.

OPC UA Discovery with Third-Party Clients

Ignition's OPC UA server disallows anonymous access on new installations, which can make the server endpoint difficult to discover by UA clients. To help with this, Ignition's UA server runs a separate unsecured endpoint to allow discovery. The discovery endpoint is accessible by appending `/discovery` to the end of the URL:

```
opc.tcp://192.168.2.134:62541/discovery
```

OPC UA Exposed Tags NodeIds

When using the OPC UA Exposed Tags feature, the NodeId returned by the browse has changed slightly. In prior versions the NodeId would follow a pattern like: `[prov]Path/To/Tag`

As of 8.0+ the pattern now includes a slash after the provider: `[prov]/Path/To/Tag.`

Both patterns still function as expected for the purposes of Reads, Writes, and Subscriptions. However a NodeId returned by a browse will always use the newer pattern.

Changes to system.alarm.listPipelines()

The listPipelines function was changed to search a single project for pipelines. Upgrading from Ignition 7 to Ignition 8 will migrate all alarm pipelines into a project named "alarm-pipelines".To maintain backwards compatibility, system.alarm.listPipelines() will check for a project named "alarm-pipelines", but now accepts a project name as a parameter, allowing the function to check for pipelines in any project.

Changes to Gateway Command Utility (GCU)

As of Ignition 8.0, the Gateway Command Utility is deprecated. Functions similar to the ones found in the Gateway Command Utility are now run through the command line. You can find a list of available commands on the Gateway Command-line Utility page.

More Information

To learn more and to stay up to date on updates to Ignition 8, check out the following resources:

- Inductive Automation’s website: many product pages have been updated with information about Ignition 8 and the Perspective Module.
  - The Blog has new information concerning Ignition 8 coming over the next few months.
  - The downloads page has been modified, and will offer new nightly builds of the latest Ignition 8 updates, in addition to stable and RC versions.
- support.inductiveautomation.com
- forum.inductiveautomation.com - The online forum is a great place to provide feedback and to learn about what other users are encountering with Ignition 8.
- Inductive University has many new videos covering a wide range of Ignition 8 topics. New videos will also be released regularly.
- This User Manual is being continually updated with new content as Ignition evolves.

Related Topics ...

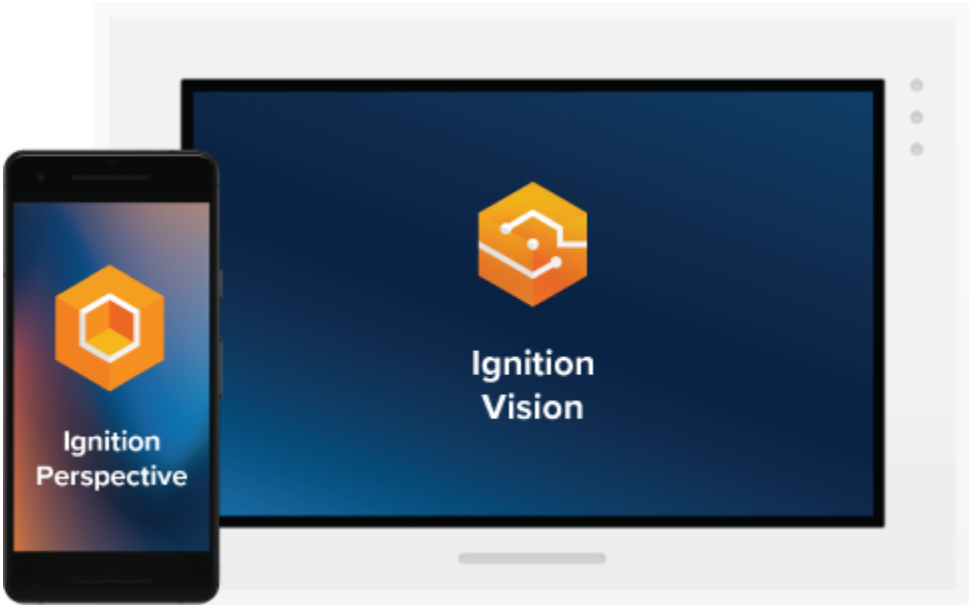
- Installing and Upgrading

Vision and Perspective — Which Visualization System Is Best for Me?

Ignition's Perspective Module is not replacing the Vision Module — in fact, the modules complement each other and can work together. Vision is the preferred module for building traditional industrial applications for the plant floor, and Perspective is preferred for mobile-first applications.

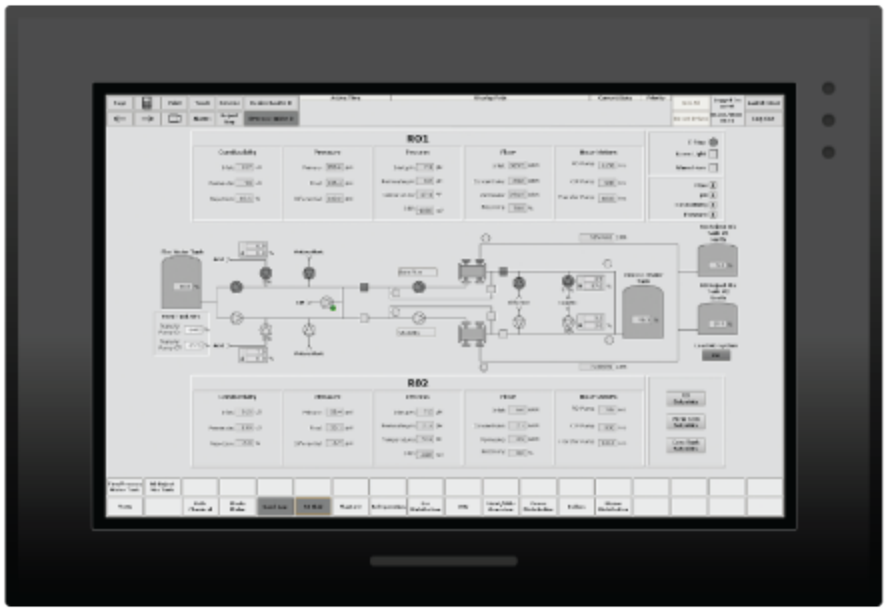
You don't have to choose between one module or the other; in fact, we think that you should use both. Perspective is for mobile-first projects and applications, optimized for web browsers. Vision, on the other hand, is a tried and tested solution for designing traditional plant-floor displays, HMIs, and desktop screens. We continue to recommend Vision for any projects that don't require browser/mobile-first functionality. Vision and Perspective modules complement one another's functionality, and when used together, form the most powerful visualization system built specifically for the industrial space.

When you start a new project, there are many things to consider, such as who the users will be, how much data you need, what kind of time and resources you have to work with, visualization needs, and so forth. Deciding whether to use Vision or Perspective really comes down to which module best fits your project at the visualization level.



When It's Best to Use Vision

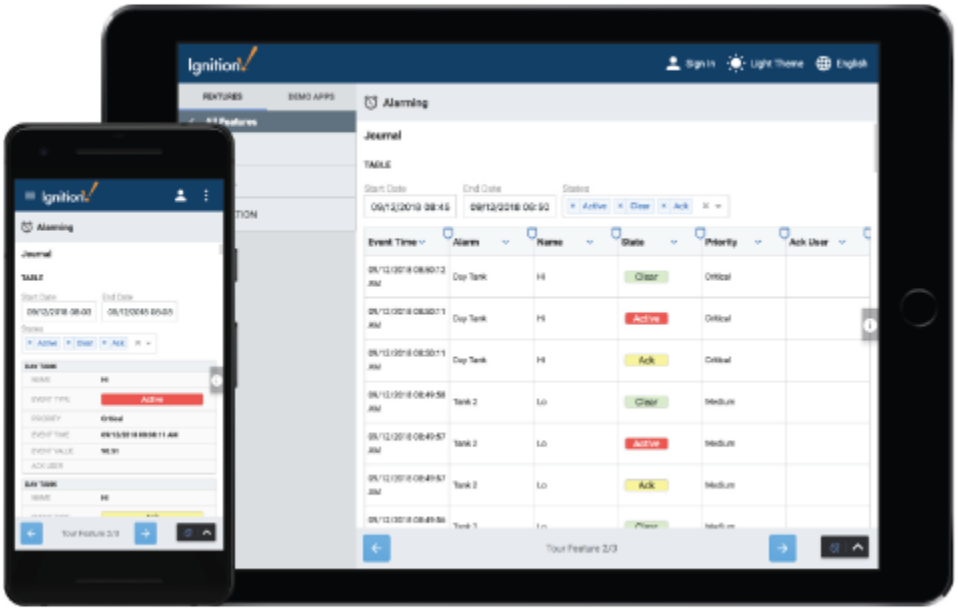
Vision is and has been for many years a top-class desktop application. It is the best choice for traditional industrial plant-floor and desktop screens, standalone HMIs, and the like. If you need a full, production-ready application right now, then it is best to choose Vision. If you're using multi-monitor or multi-desktop workstations, dedicated control panels and applications, desktop-dedicated or dedicated-access applications, terminals, or parallel screens, then Vision is the best way to go. In Ignition 8, the Vision Module got one of its biggest updates since it was first released in 2010, allowing you to make your HMIs and plant-floor dashboards faster, easier to deploy, and more beautiful than ever.



When It's Best to Use Perspective

If you need to build mobile-responsive applications, then Perspective is the recommended way to go. If your application needs to run on a mobile OS, namely iOS or Android, then choose Perspective. If you need Two-Factor Authentication or federated identities for the application you're building, choose Perspective.

With Perspective, you can automatically adapt to fit any screen size using mobile-optimized container types. It provides the ability to use your device's sensors and intuitive touch commands, as well as message handling, flexible property bindings, CSS3 styles, and more. For example, with an application built in Perspective, you could augment alarm information by taking a picture of the faulty machine. You could scan a QR code by a machine and have the SCADA system tell you what to do, or take readings or photos in the field and tag them with GPS and other sensor data. There are so many new possibilities.



Should I Upgrade?

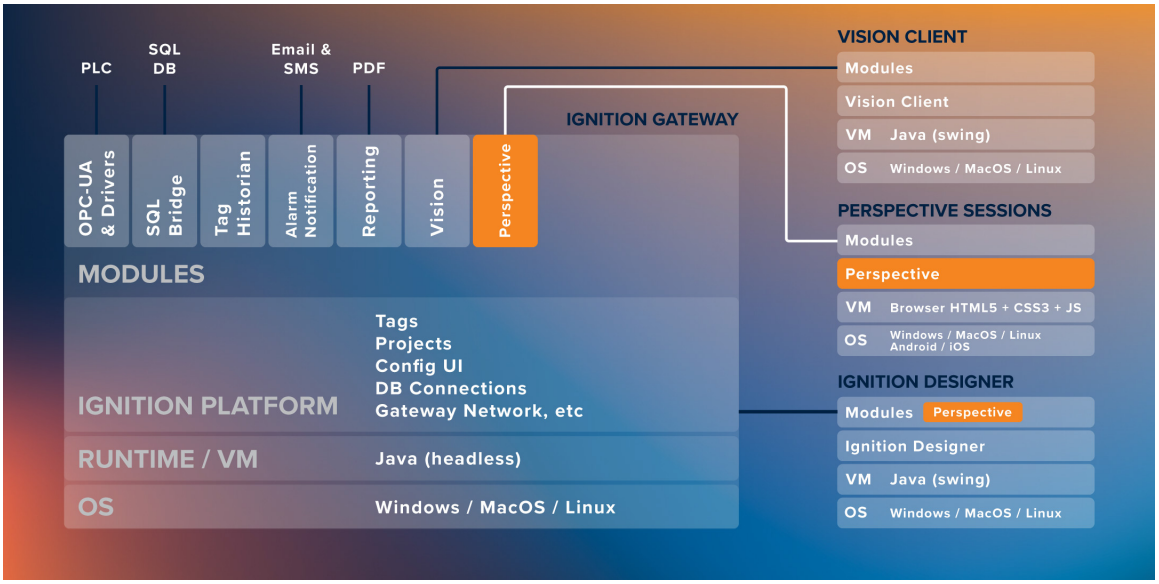
There are many considerations for using Vision or Perspective. Perspective contains familiar elements like the drag-and-drop Designer, property binding system, and scripting engine. Refer to the Ignition 8 Upgrade Guide for complete information. See also A Vision-Oriented Guide to Perspective.

Behind the Scenes

A “Client” in Vision is called a "Session" in Perspective, and the two actually have very similar tech stacks. At the bottom of the stack is the OS. In addition to supporting Windows, macOS, and Linux, Perspective also supports Android and iOS. The next part of the stack is the web browser, which is very similar to the runtime/VM layer in the other stacks because it provides an environment to execute code, generate user interfaces, and communicate back to the Gateway. On top of that runs the Perspective session application itself, and then modules can augment the features of Perspective.

In the topology of a typical Ignition installation, there’s the Ignition Gateway, the Ignition Designer, and Vision clients. At the base of the Gateway is the operating system, which can be Windows, Mac, or Linux. Above the OS is the Java runtime/VM layer, which is what Ignition runs on. Above the runtime/VM layer is the Ignition platform, which provides many features and services like Tags, projects, database connections, the Gateway Network, and so on. At the top of the tech stack are modules, which let you do a number of useful things like connect to devices, use databases effectively, send alarm notifications, generate reports, and, in the case of the Vision Module, create applications which run as Vision clients.

As you can see, Perspective is just another module. It installs into the Gateway and provides a design experience in the Designer just like other Ignition modules do.



Related Topics ...

- Perspective
- Vision

Support, Feedback, and Resources

Support Team

Can't find it here? Go to the Support homepage for one-on-one help from our support team. Submit a Ticket, and one of our Support Engineers will follow up with you quickly.

You can reach us during business hours 8am-5pm PST at 1-800-266-7798. Support charges may apply. 24-hour support is also available, at an additional fee.

E-mail support is available at support@inductiveautomation.com.

Long-Term Support (LTS)

At Inductive Automation, we are constantly improving our software, both in features, functionality and reliability. We provide Long-Term Support (LTS) for certain versions of the Ignition software for a period of five years from the date of its original release. The current release branch and non-expired LTS releases fall into the Active category. This means that the software is being actively developed/patched and the IA support team can provide full customer support with issues related to these releases.

For those Ignition versions that expired, there are no new releases or patches for any issues that may arise in expired versions. You can see in the table below that LTS is currently provided for versions 7.8 through 8.0, and versions 7.7 and prior, new development and patches are discontinued.

The table below shows the original release dates, LTS status, and end date of Active support status.

Version	Date Released	Is LTS	Date Active Support Lapses
8.0	Apr-08-2019	No	(8.1 Release)
7.9	Dec-12-2016	Yes	Dec-12-2021
7.8	Oct-15-2015	No	Dec-12-2016
7.7	Jul-16-2014	Yes	Jul-16-2019
7.6	May-7-2013	No	Jul-16-2014
7.5	Jun-14-2012	Yes	Jun-14-2017
7.4	Mar-29-2012	No	Jun-14-2012
7.3	Oct-5-2011	Yes	Oct-5-2016

To learn more about our product support in detail, refer to our Support Policy.

Resources

Inductive Automation Blog

Our company blog features announcements, Ignition tips, and a more personal look at the folks behind the scenes of Inductive Automation.

Knowledge Base

In our Knowledge Base, you can search and view articles created by our support team for troubleshooting, known problems, and workarounds.

Community User Forum

On this page

...

- Support Team
  - Long-Term Support (LTS)
- Resources
  - Inductive Automation Blog
  - Knowledge Base
  - Community User Forum
  - More Resources on our Website
- Comments, Feedback, or Ideas?
  - Feedback on the User Manual
  - Feedback on the Ignition Software
  - Feedback on the Inductive University
  - Other Feedback

Our User Forum is *the* place to get help from our passionate community of Ignition users. The Forum is one of the most effective ways to get help from others. It is always available and is actively patrolled by Inductive Automation staff and many knowledgeable users. Chances are you will find your question already answered in an existing post, but if not you can sign up and ask away.

More Resources on our Website

- Ignition How-Tos: Step by step guides
- White Papers and Articles: What's happening in the industry and with Inductive Automation
- Webinars on Demand: Watch recordings of our past live webinars
- Tip Sheets: Tips and references put together for you

Comments, Feedback, or Ideas?

We encourage your feedback on both the Ignition Software and Ignition User Manual. We strive to keep in step with our community and grow the software to meet your needs. If you have something you think we should add or improve, let us know!

Feedback on the User Manual

We encourage you to give us feedback on the Ignition wiki User Manual as this will help us improve the documentation over time. You can email us at training@inductiveautomation.com.

Feedback on the Ignition Software

How can we improve Ignition? Our goal is to provide you a powerful yet easy-to-use SCADA/HMI software that does everything you need. Let us hear from you on our features and ideas page. We love to hear any ideas you have so we can continue to improve Ignition.

Feedback on the Inductive University

Do you have a question about any of our videos? You can email iu_support@inductiveautomation.com for answers or suggestions on new videos and improvements.

Other Feedback

Do you have something else to tell us? Not sure where to send it? You can send an email to us at training@inductiveautomation.com, and we will get you pointed in the right direction

Related Topics ...

- Installing or Upgrading a Module
- Quick Start Guide
- Quick Start - Vision Client
- Quick Start - Perspective Session

A Vision-Oriented Guide to Perspective

An Intro to Perspective

On this page

...

- An Intro to Perspective
- Perspective Design Considerations
  - Designing Your Main Views
  - Docked Views and Pages
  - JSON Component Properties
  - Bindings and Transforms
  - Inherited Projects
  - Components
- Comparison of Perspective and Vision Functionality
- New Features in Perspective

Perspective is a flexible and versatile approach to HMI/SCADA design. Whereas Vision clients run on any platform that supports Java, Perspective *Sessions* run anywhere that supports a web browser. Additionally, sessions can run in the Ignition Perspective App. This opens the door for a new level of support for mobile devices and tablets. Perspective's features follow a responsive design paradigm. This added dynamic and flexibility means you'll need put a little more planning into your projects.

The purpose of this guide is to describe the core functionality of Perspective in ways that make sense for Vision users. For a step-by-step guide in starting your first Perspective project, check out our Quick Start Guide.

The largest difference between Vision and Perspective is the way they do layout and project design. Perspective is primarily designed to allow you to create mobile responsive interfaces. Perspective is an Ignition module, using the same platform as the Vision module. It has access to all of the same systems like Tags, database connections, reports, and security (though security got a few updates in Ignition 8 too).

As we developed Perspective, we took the familiar aspects of using the Ignition Designer and provided more tools and flexibility. You will find many similarities between how you design a Vision project and a Perspective project. Anyone familiar with Vision can start designing quickly. For example, you are still working with containers, components, properties, bindings, Python scripting, Tags, and databases. While it is visually different from Vision, you will find the interfaces in Perspective familiar, but updated.

In order to take full advantage of the new systems in Perspective, there are a few things that you need to think about in a different way from your past Vision projects. But first, there are a few terms that we need to define in Perspective. We will be using them a lot to talk about the differences, so please familiarize yourself with them before continuing. A more complete set of terms is provided below.

- **View**: Think of this as a Vision Template, Window, and Container all rolled into one. You can put components in it, you can pass parameters into it, and you can nest them inside each other. Each view has a Layout type.
- **Layout**: There are several types of containers in Perspective. Each has its own set of position attributes for the components inside it, which is similar to the Vision Relative vs Anchored constraints. There is more than just X, Y, width, and height now.
- **Page**: This is a new concept for Perspective. Instead of having a Vision Client with multiple windows open at the same time, you open a single Page at a time and navigate by switching pages. Each page has its own main View and any docked Views you want.
- **Session**: A session is the Perspective equivalent to a Vision Client, except it runs in the browser instead of using Java. You can have multiple browser tabs open using the same session.
- **StyleClasses**: Style Classes are new too. These are groups of settings like text color, text size, and margins. They use CSS and can be bound to components for very dynamic styling across your project.

Perspective Design Considerations

With Perspective, it's more important than ever to have a plan before you start designing. Here is a short list of the things you want to think about before starting to build you visualization system. Note that this assumes you already have your Tags, database, and other Gateway items taken care of.

- Make a flowchart of your project. How will users navigate, and what is the tree structure for your pages and popups?
- Make a visual outline of each page. What docked windows do you want? What will you navigation look like?
- Get an idea of what windows you want, and how you want them to look.
- Will your project be mobile responsive? Plan the look for both a large and small version of each window.

Once you have an idea of what your session will look like, you can start designing.

Designing Your Main Views

Deciding on headers, tab strips, navigation trees or other methods of navigation is just a start in Perspective.

If you want to **keep things simple**, you can use a Coordinate View. This will feel extremely similar to a Vision window since all the components have an X, Y, width, and height. Just drag your components onto the View and use the handles to stretch them to the size you want. By default, all components will be locked in using those location properties. These views will behave similar to a Vision window with all components anchored only to the top and left. You can change the Mode property in the root container to "Percent" to make the components behave similar to the Relative layout mode in Vision. Many will use this with a single docked view for navigation. This will create a project very similar to the Vision Single Tier Nav project template.

Views can be **nested** inside each other to create more complex structures. If you want a more structured layout, you can use the Flex View wi th other views inside it. A Flex View creates a row or column structure out of your components. For example, you could use use a Flex View with a header at the top, and a coordinate container filling the rest of the view. This is another simple structure that creates a similar structure to a Vision window with an anchored header and relative layout components in the main space.

If you want to make your project **mobile responsive**, there are other types of Views that you can use. The Column View allows you to add your components, then arrange them differently for three unique sizes of screen. The Breakpoint View allows you to show two completely different views based on the size of the screen.

And of course you can nest all of the different types of views in any combination you want. However you choose to set up your project, you will want to figure out which containers and combinations of containers will give you the look you want. Try looking at your favorite webpages for inspiration, there are a lot of great (and terrible) User Interfaces out there to give you ideas. For example, you can use a Breakpoint View that when small, contains a Flex View that is a single column of components, and when large, contains a Coordinate View that shows a diagram of your entire facility.

Docked Views and Pages

A session, much like a client, is made up of more than one view, and you will often see multiple views at the same time. In Vision, you have multiple windows open, and certain windows are designed to be docked or floating. In Perspective you have Pages instead. A Page is a view or set of views that open and close together. Most commonly, a page will have a main View and a docked view.

Navigation in Perspective means moving from one page to another, not opening and closing multiple windows. This allows you to use the forward and back buttons on your browser as a part of your navigation strategy.

There is a shared space that all pages inherit from, so you can set up docked views there, and any page that is open will automatically have those docked views open as well.

There are many settings for your docked views. You can set these views to be expandable, auto hide depending on session width, and a number of other options.

JSON Component Properties

Perspective components have a slightly different type of property. In Vision, we used a flat property structure where everything was a basic data type like integer or string, except for datasets. If you wanted a dynamic number of properties, you had to create a dataset, then use scripting to pull the values out. In Perspective, all of the properties on a component or container are JSON objects.

This means the properties have different types. They can be a Value (int, float, string, etc), an Array (a numbered set of sub-properties), or an Object (a complex set of sub-properties with key:value pairs). For arrays and objects, the sub members can be any of the three types.

Datasets can now be stored on components as an array of objects, where each object is a row, and each object has several values that make up the columns.

Bindings and Transforms

There are more options for bindings that were not possible in Vision, and this is only partly due to the new property structure. For example, there is an HTTP binding that allows you to directly connect to a web service and download or upload data.

On top of that, any binding can have transforms on it. A transform is an expression, map, or script that takes in the value of the binding and changes the output. If you chain together multiple transforms, the output of the previous transform is in input of the next. Think of number-to-color translations. Instead of creating a custom property with an expression on it and then binding a color to that custom property, you can do it all in one binding. Just create an expression binding, then add a map transform to change the output value to a color.

Inherited Projects

Another design consideration is inheritance. Outside of the look and feel of a perspective session, in Ignition 8 you can have multiple projects that inherit resources. If you have views or scripts that you want every project to be able to use, you can create a master project and tell all other projects to inherit from that. You can then use the resources or allow designers to overwrite them as needed.

Components

Because of all the differences between Perspective and Vision, the two modules necessarily have a different set of components. There is no plan to ever have a one-to-one parity between components, and not all of the Vision components will make their way into Perspective components.

Some components are new to perspective like the Menu tree and the Link components. The Link component was not feasible in Vision, but with Perspective Sessions living in a browser it makes sense to have that component.

The library of components in Perspective is growing continually. For example, an Alarm Status Table similar to the one in Vision is currently under development. Make sure to go to the forum, the ideas portal, or download a nightly build if you want to see what is coming up.

Comparison of Perspective and Vision Functionality

Here is a quick comparison of summary of basic Perspective concepts, and how they are similar to / differ from their Vision analogs.

Perspective Concept	Similar Vision Concept	Perspective Description
Session	Client	A Perspective Session is a running instance of an application, much like a Vision client. Whereas Vision clients run as independent Java programs on a user's machine, a Perspective session runs natively in a web browser (or the Ignition Perspective App). Notably, a session can run across multiple pages in the same web browser. This is similar to how sessions function elsewhere on the Internet: log into your favorite shopping website and then open several new tabs, each one knows about your current shopping cart.
Page	**Desktop (A.K.A multi-monitor clients)**	**A page in Perspective equates to a single page in a web browser. Pages are the main navigational unit in a session, and consist of one or more views. Each page is associated with a URL, which means the Forward and Back buttons in a web browser can be used to navigate to pages that have already been visited. Multiple pages can be open as part of the same Session, similar to how multiple desktops in a Vision Client may be open simultaneously.** **Views can be docked to specific edges of the page, or be used as a "Main View". Each page consists of at least a main view, but multiple views can be configured on a page. A page has specific regions where you can place instances of your views.**
View	Window and Template	A View is the primary unit of design for Perspective. Perspective Views fill the roles of both Windows and Templates in Vision. Thus, you could think of a view as a window that can be nested inside other windows, or as a template that can be maximized or docked. Views have a root container, much like windows in Vision. Nesting one view inside another requires the Embedded View component. Parameters may be defined on the view, and then passed into the embedded view There are also components that may dynamically create view instances (like the Flex Repeater component). In this way, views act like templates just as they do in the Vision module. Parameters can be passed into a view from an external source.
Container	Container + Layout Constraints	Containers are objects that contain components. You can nest one container inside of another. In Perspective, the way that components reposition and resize is managed on the container, and every component contains only simple information about its size and position, which is interpreted based on the container's specifications. There are several container types, and each type is defined by the layout type it uses.
Component	Component	Components, like in Vision, are displays, buttons, charts, labels, and other objects that display information to the user viewing the session. Components are the elements displayed in the Component Palette in the Designer.
Property	Property	

		Properties serve as a place to change how a component looks or behaves, or store specific pieces of information. In Perspective, the property tree of a component is a JSON object, and as a result, there are only three configurable data types: **value**, **object**, and **array**. Rather than characterize the expected format of the property's data (e.g., integer, boolean, or string), these data types control the structure of the property tree. No further configuration of data types is required.
Session Property	Client Tag	Every session has a configurable collection of properties that can be managed from the designer. They can be referenced from any view in the project, both from property/expression bindings and from scripting.
Events and Actions	Component Scripting	Event and Actions are configured similarly to the Component Scripting section in Vision. However, Perspective offers more configurable events, along with more possible responses to these events (Actions).
Styles	Component Style Properties /Customizer	Styles in Perspective use CSS, and are configurable from the properties on each component. However, Perspective also offers the ability to configure style classes, which allow you to reuse configured styles across many different components. Styles can also be configured to change based on more advanced properties, like whether a mouse is hovering over the component, or how wide the viewport of the viewing device is.

New Features in Perspective

In addition to an entirely new set of components, Perspective offers a variety of new features in the realm of Project design.

Perspective Term	Description
Transform	A transform lies between a binding and the property it modifies, and provides an opportunity to change the value or format of the binding's output. For example, if a tag binding yields 0 for Normal and 1 for Faulted, we could use a transform to map 0 to Green and 1 to Red.
Component Messaging	Perspective offers the ability to communicate between *components*, using a similar style of messaging that one might use to communicate between clients in Vision. You can send messages from any component using the system.perspective.sendMessage function, and you can configure message handlers on any component. This is useful for controlling the behavior of one component from another in a different view.

Licensing and Activation

How Licensing Works

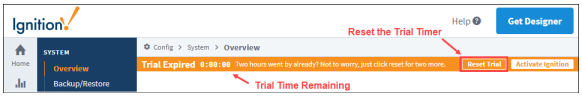
Ignition's licensing is unique and easy to use because Ignition is licensed by the server, not the client. You only need one license for your server; any clients are automatically included. In addition to that, an Ignition license is unlimited and sold based on which modules you want. There are unlimited clients, Tags, and projects. Buy only the modules you need, and don't worry about running into limits. If you want to test other modules, you don't need to do anything extra because of our built-in Trial mode. All unlicensed modules can be reset in 2 hour trial mode.

Trial Period

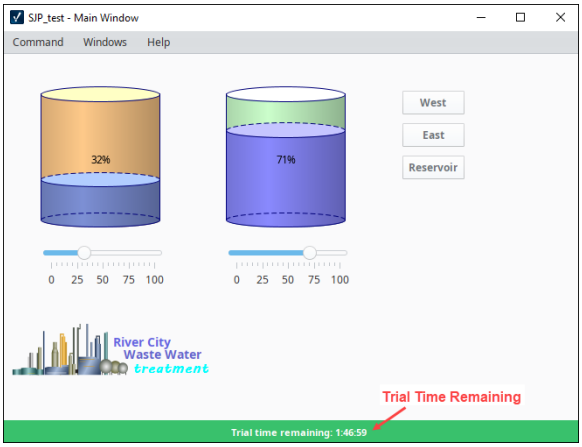
Our goal at Inductive Automation is to provide an easy way to access and learn Ignition. We want everyone to have the opportunity to try Ignition and start working with it with no restrictions. You can download Ignition from our webpage, install it, and start using it for 2 hours at a time. At the end of the 2 hour time period, all modules will stop running, but don't panic, you can reset the timer to run for another 2 hours. You can reset the timer as many times as you want, so go for it! The Gateway Webpage and the Designer are not affected by this trial, so you can develop for as long as you want without interruption. If you do have a license, any unlicensed modules will run in this Trial mode, but licensed modules will never timeout.

You can re-start the Trial period by logging into the Gateway, and clicking **Reset Trial** to enable another 2 hours of execution. The Trial Timer may be restarted any number of times. Depending on the module, you may need to take some additional actions. For example, the Vision Clients requires you to log out and back in again in order to continue the Trial.

In the Gateway, the Trial Time banner is displayed near the top of the screen. To reset the trial timer, click the Reset Trial button:



In a Vision Client (for Edge or Ignition Standard), the banner appears near the bottom of the screen.



On this page

...

- How Licensing Works
- Trial Period
- How to Activate Ignition Edge and Ignition Standard Edition
  - Online Activation Example
  - Offline Activation Example
- How to Activate Maker Edition
  - Key and Token
  - Activation
- Adding Multiple Licenses to a Single Gateway
- Module Status
  - License Incomplete
- Updating a License
  - Two Methods of Updating a License
- Emergency Activation
  - How to Activate in Emergency Mode
- Unactivating a License

INDUCTIVE UNIVERSITY

About Licensing

Watch the Video

INDUCTIVE UNIVERSITY

About the Trial Period

Watch the Video

How to Activate Ignition Edge and Ignition Standard Edition

When you purchase a license for Ignition Standard edition or Edge, you receive a CD-Key, a six-digit code that identifies your purchase. Use this CD-Key to activate the software online through the Ignition Gateway. If you later want to add any additional modules, your account is updated and you can re-use your existing CD-Key to activate the new features. You can also una ctivate your CD-Key, and reuse it to activate Ignition on a different machine as many times as you want.

Online Activation

Watch the Video

You can activate your license in two ways:

- **Online Activation** - from the **System > Licensing** section on the Gateway Webpage, your request to activate your Gateway is activated over the internet.
- **Offline Activation** - if you don't have an Internet connection, you can follow the manual activation process in the Offline Activation section.

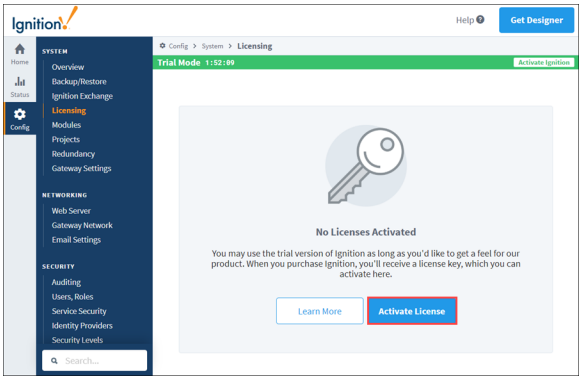
Offline Activations

Watch the Video

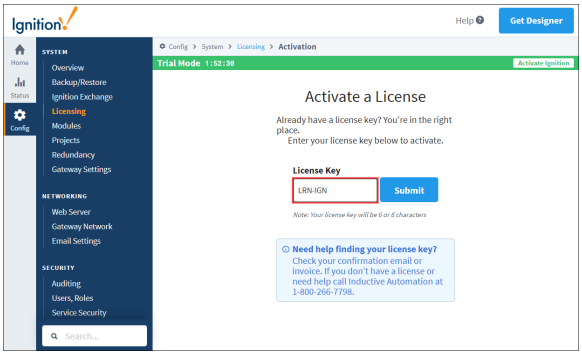
Online Activation Example

To activate a license online, use the following steps:

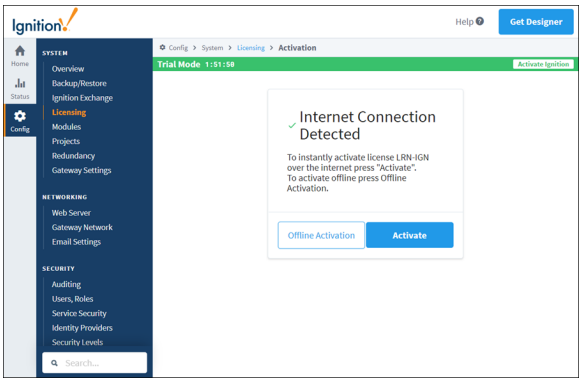
1. Go to the Gateway Webpage and select the **Config** section.
2. From the menu on the left, select **System > Licensing.** The Licensing page will appear. If you already have a license key, click the **Activate License** button.



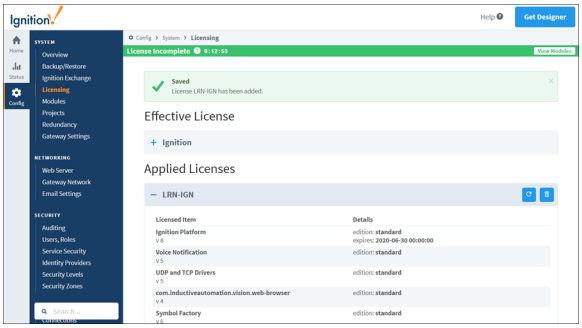
3. The **Licensing / Activate Online** page will be displayed. Enter your **License Key** and click **Submit**.



a. If you are connected to the Internet, click the **Activate** button.
b. If you want to activate the license offline, click the **Offline Activation** button.



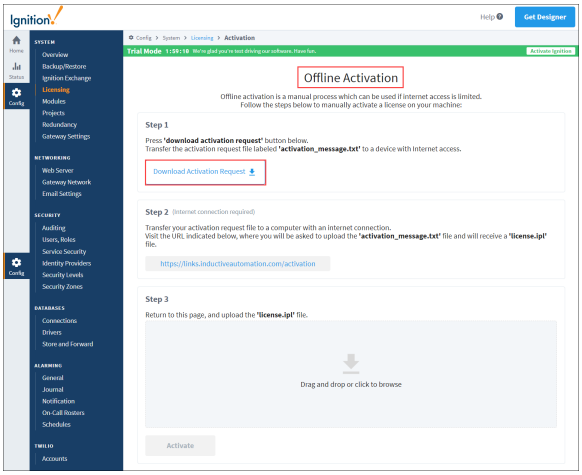
4. The **Licensing** page will refresh and your **Current License** will be successfully activated.



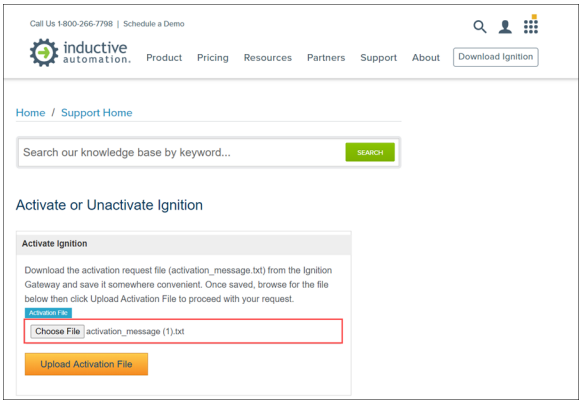
Offline Activation Example

When you do not have an internet connection, you can do the following steps to activate your license manually:

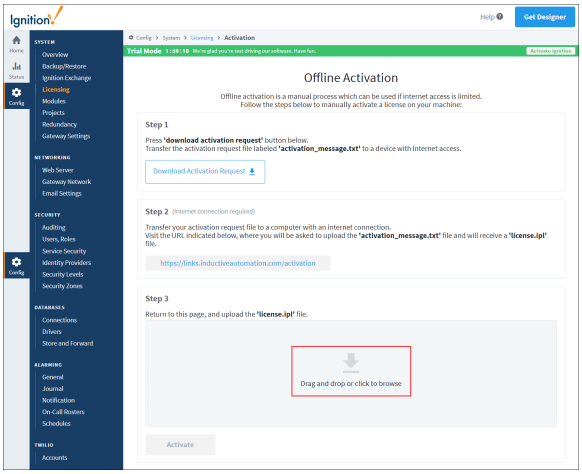
1. Go to the **Config** tab on the Gateway Webpage.
2. Scroll down to **System > Licensing.** The Licensing Activation page appears.
3. Click the **Offline Activation** button.
4. Click the **Download Activation Request**. An activation request file, called `activation_message.txt` is generated and downloaded.



5. Take the `activation_message.txt` file to a machine with Internet access and go to https://links.inductiveautomation.com/activation.
6. Click **Choose File** and select the `activation_message.txt` file.



> Should any troubles occur with the upload process, please contact our support department.

7. Once you upload the `activation_message.txt` file, a license file called `license.ipl` is generated.
8. Bring the license.ipl file back to the computer on which you're licensing Ignition. Back on the Offline Activation page, upload the `license.ipl` file and click **Activate** to finish the process.
9. 

10. The Licensing page will refresh showing your license was successfully activated.

11. Click **View Modules** to see a list of all your activated modules.

> The following feature is new in Ignition version **8.0.14**
> Click here to check out the other new features

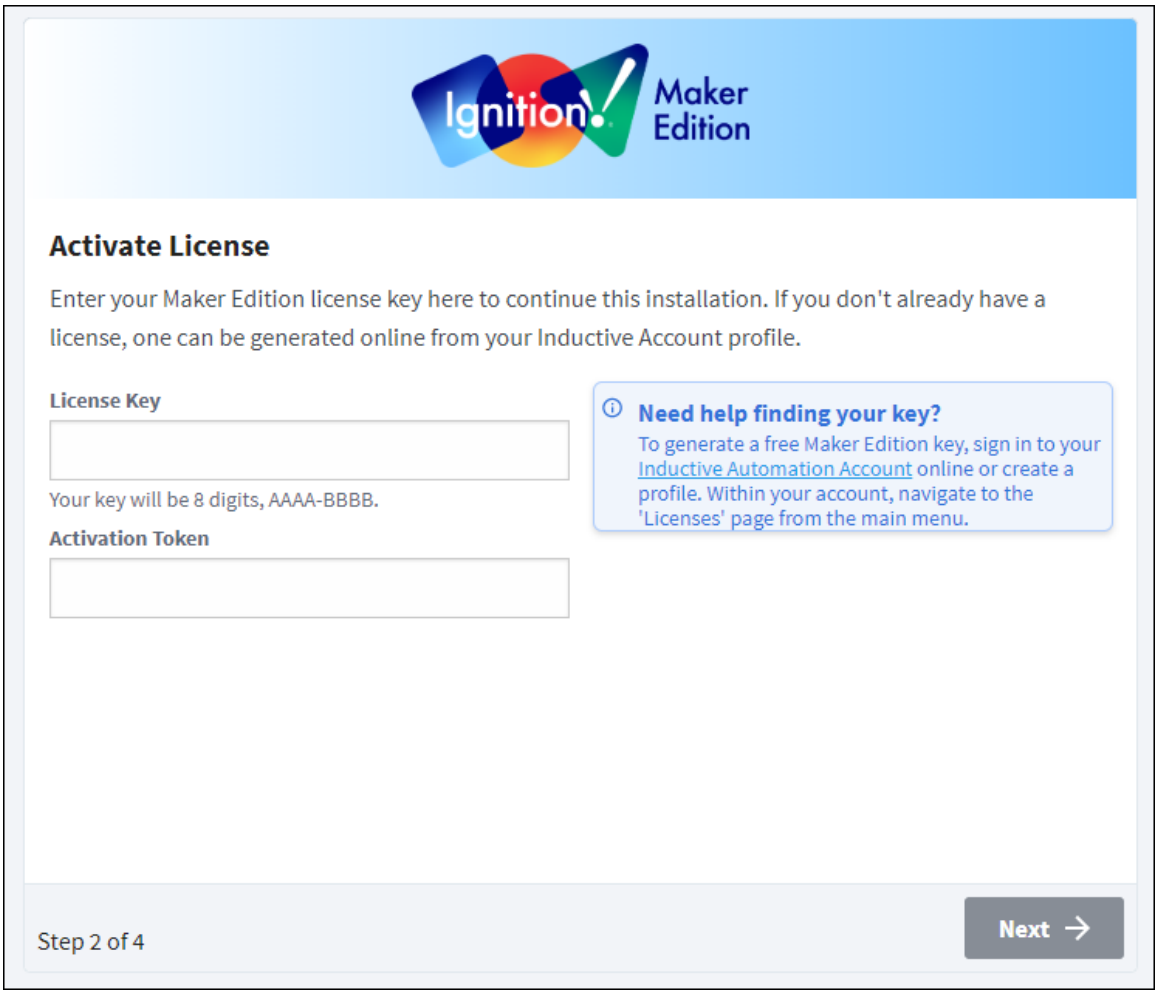
How to Activate Maker Edition

Key and Token

When a license is purchased you receive a Key (an eight-digit code) and a token that identify your purchase. Use this Key to activate the software online through the Ignition Gateway. You can also unactivate your license key, and reuse it to activate Ignition on a different machine as many times as you want.

Activation

When you download Ignition, you have the choice of Ignition Standard, Edge, or Maker Edition. You activate the license for Maker Edition during the installation process. When you choose Maker Edition. the installation wizard will prompt you for a license key and token. If you need help finding your key, sign in to your Inductive Automation account.

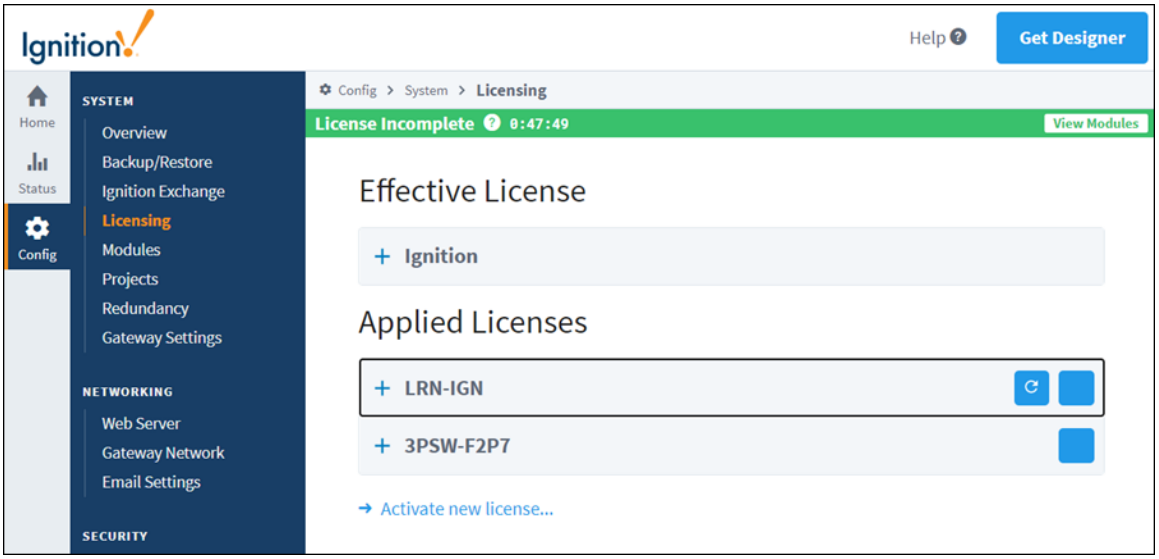


Adding Multiple Licenses to a Single Gateway

In order to better support our community of third-party module authors, we also allow for multiple license keys to be installed on a single Gateway. A third-party module author can issue a license key for their module directly to a customer, whereby, they can immediately do an install of the module. To learn more, check out the Third-Party Module Showcase to find and purchase modules that extend Ignition's functionality.

It is important to note that there may only be one license on it with a platform version per Gateway. The platform will look similar to the picture below. If you try and activate a second license with a platform onto a Gateway that already has a license with a platform, the new license will overwrite the previous license.

In the image below, two licenses have been applied to this Gateway, but only one is active.



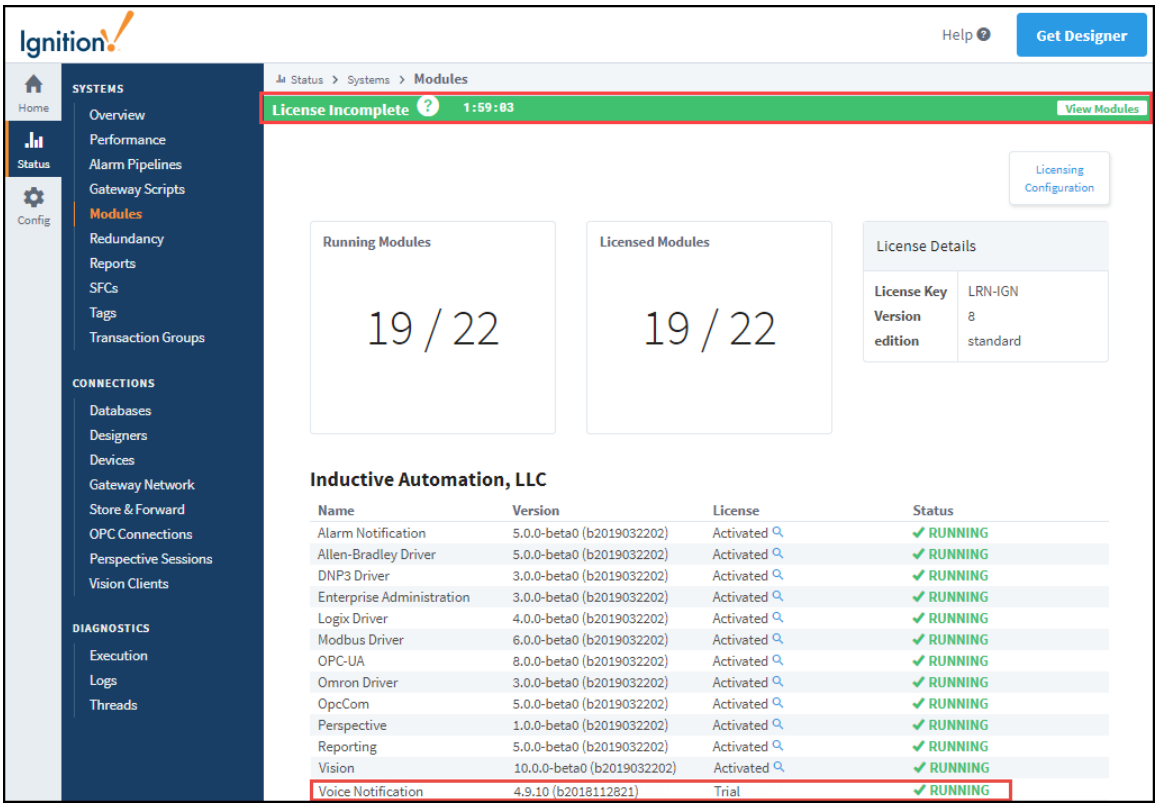
Module Status

It's important to know which version of a module you have installed and which version of the module you are licensed for. You could have a license for another version of a module so it's not going to work correctly until you have the correct version of the license. To verify the versions on your installed modules, go to the **Modules** page in the **Status** section of the Gateway Webpage or by clicking **View Modules** in the green banner.

Here you can see all the modules that are currently running with their version numbers and license status. The licenses are either in **Trial, Activated,** or **Free** mode. On this page, you can add or remove modules from your Ignition. If you are licensed for a module you are running, it will run in **Activated** mode. If you are not licensed for a module you are running, it will run in **Trial** mode until the Trial time is expired.

License Incomplete

Don't be alarmed by a 'License Incomplete' message on the green banner. Click on the View Modules link within the banner. It opens an informational message box showing modules that are installed. Some of the modules are in Trial Mode.



Updating a License

If you want to add more modules to your existing license, just call Inductive Automation Support and we can add whatever you need. Once we upgrade your license key to add or remove modules, your license needs to be updated. You update a license using the same license key from the **Config > Licensing** section on the Gateway webpage.

Your license needs to be reloaded once it has been upgraded by Inductive Automation to include new modules, removed modules, or update allowable module versions. The same license key is used for the license upgrade, so there is no need to unactivate the license. You can update your license using the same license key by reloading the license.

When updating a license from the Gateway, there are two methods of activation: with an Internet connection and without an Internet connection.

Two Methods of Updating a License

You can update your license in two ways:

- **With an Internet Connection** - from the **System > Licensing** section on the Gateway Webpage, update your Gateway license over the Internet.
- **Without an Internet Connection** - if you don't have an internet connection, you can follow the offline update process in the Offline Activation section.

Emergency Activation

In cases where you may have a hardware or OS failure and you cannot unactivate a license, Ignition provides an Emergency Activation mode. In this mode, you can temporarily activate your license for 7 days giving you enough time to contact Inductive Automation Support.

How to Activate in Emergency Mode

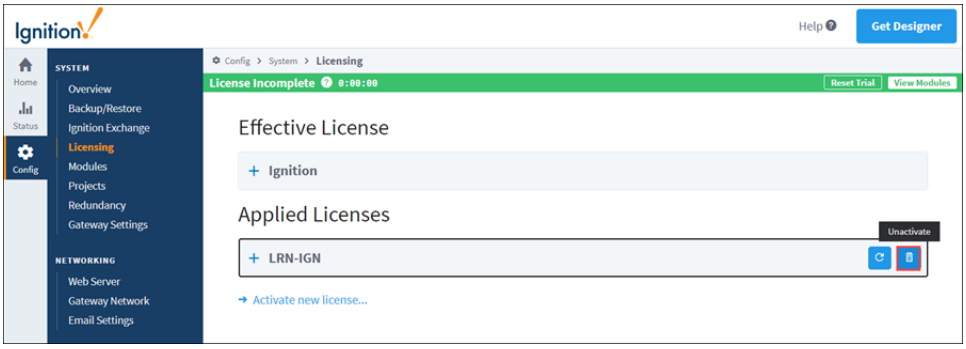
Activating your license in emergency mode is exactly the same as activating with your normal license, you don't have to do anything different because Ignition handles it all for you. See Licensing and Activation page for details on how to activate. The Gateway will know to run in emergency activation mode and it will display a timer stating how many days, hours, and minutes you have remaining in the Emergency Activation Mode banner. Any time before it expires, you can contact Inductive Automation Support to get your license fixed.

Unactivating a License

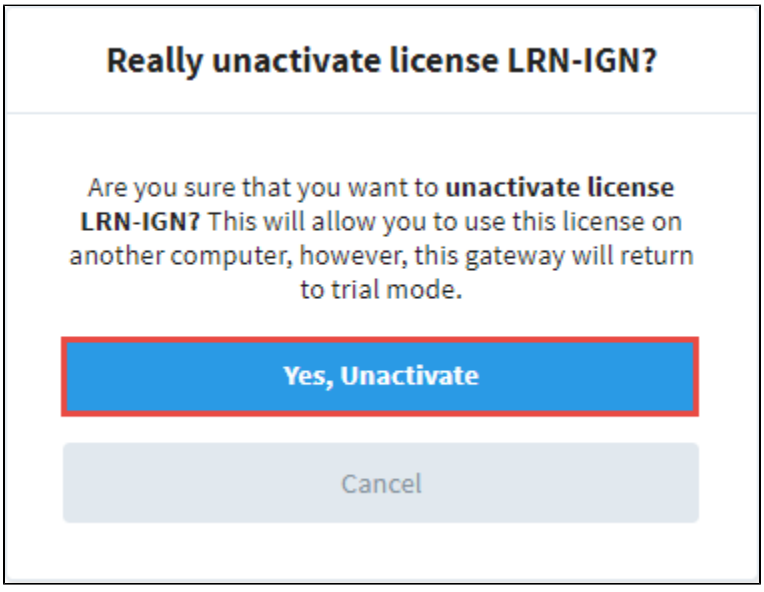
For a given license key, a limited amount of simultaneous activations are allowed at a given time. If you want to activate Ignition on a different server, you must first unactivate it on the current server. You can unactivate the license on one Gateway, and then activate it on a different Gateway if needed. Unactivation occurs immediately over the Internet, and makes this license available for activation on another machine.

To unactivate the Gateway, do the following steps:

1. From the **Config** section of Gateway, go to **System > Licensing**. The **Licensing** page is displayed and you can see the currently installed license license key.
2. Click on **Unactivate License** icon.



3. A Licensing / Confirm Unactive window will appear asking you to confirm the unactivation. Click the **Yes, Unactivate** button. It may take a minute or so for the request to finish.



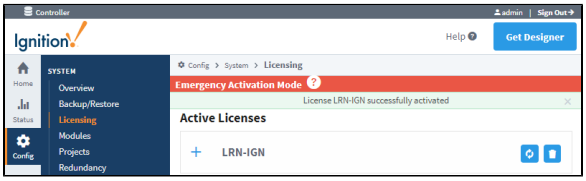
4. The unactivation request will be sent to Inductive Automation's licensing servers, and the license will again be available for activation on another Gateway.

In This Section ...

Emergency Activation

What Is Emergency Activation?

After activating your License Key in Ignition, you may see a banner that says Ignition is in **Emergency Activation Mode**.



What Does this Mean?

This means the License Key that you used has already been activated. Either it was used previously on another computer, or something went wrong during activation and it registered twice. Fortunately, you have plenty of time to fix this problem. Emergency Activation Mode means **you are fully licensed for 7 days**, so even at night or over the weekend, you can still run without interruption until it is fixed.

On this page

...

- What Is Emergency Activation?
  - What Does this Mean?
  - How Do I Fix it?
  - Why Does this Exist?
- Quick Disaster Recovery Plan
  - What You Need Before an Emergency Happens?
  - Emergency Restore Steps

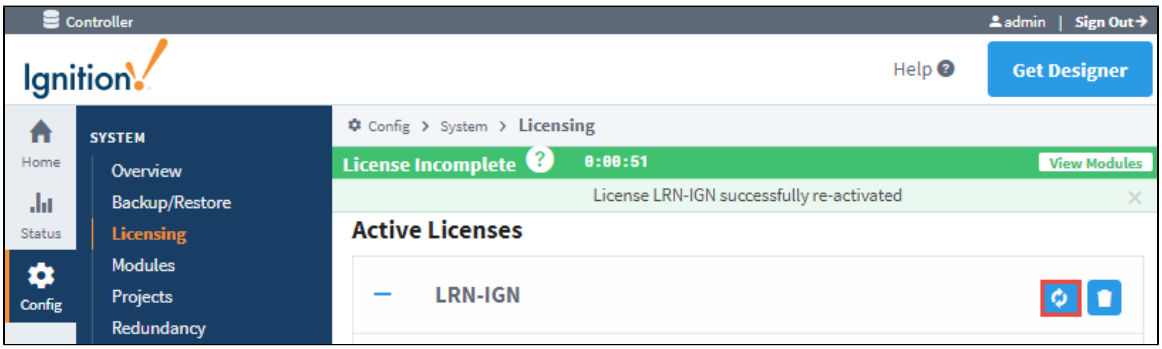
INDUCTIVE UNIVERSITY

Emergency Activation

Watch the Video

How Do I Fix it?

In order to fix your License Key, you must **contact Inductive Automation**. You can reach us by email or by phone and we will walk you through fixing your license. It is usually a very quick procedure that involves you re-activating a license after we fixed your License Key from our end. Once your license key is fixed, press the **Refresh** button and your license will be re-activated.



Why Does this Exist?

Basically, computers sometimes fail and you need to get running again, fast.

When you initially activate your copy of Ignition, it will run as long as your computer does. As we all know, sometimes there are hardware failures or your computer just stops working. There are many things you can do to minimize this, but once it happens you need to be able to get Ignition up and running again quickly. This Emergency Activation Mode allows you to do this without having to involve Inductive Automation until after your facility is back on track.

Quick Disaster Recovery Plan

What You Need Before an Emergency Happens?

For full steps to get Ignition running for the first time, see our Quick Start Guide. If you already have Ignition installed and running in your facility, it's easy to get back to running quickly. Just make sure you have the following stored on a computer or shared drive **that is not your Ignition Gateway**:

- A Gateway Backup. If you haven't already, you should set up Scheduled Backups.
- Your Ignition License Key. Make sure you store the 6 digit code somewhere that you can easily retrieve it if your computer fails.
- The Ignition software Installer. In an emergency, it's best to have the installer for the version you are currently using. You can always download an archived version from our website, but having one on hand is preferred. *Pro tip*: Edit the installer filename to include your License Key so it's easily available.

Emergency Restore Steps

1. Find another computer or create a new Virtual Machine. You need to get something up and running to put Ignition on first. We recommend having something on standby that is the same as the original computer. Further considerations:
    a. Set your new computer to the same IP Address of the original Ignition Gateway. This way your clients won't need to do anything special to start running again.
    b. Make sure your firewall is set correctly. At minimum, you must have the Ignition port open (8088 by default). In an emergency situation, you may want to just disable the firewall temporarily.
2. Install Ignition on the new computer.
3. Load a Gateway Backup. Everything in Ignition is in the Gateway Backup, once it's loaded, you are running again!
4. Activate your License. Your existing License Key will only work normally the first time it is used. After that, if you try to use it again, it will instead go into an **Emergency 7-Day Trial**. This gives you plenty of time to get back up and running before you have to deal with the License Key, even if your failure is on the weekend or outside Inductive Automation's business hours.
5. Contact Inductive Automation. You can reach us by email or by phone (**1-800-266-7798**), and we will walk you through fixing your license.

Maker Edition

What Is Ignition Maker Edition?

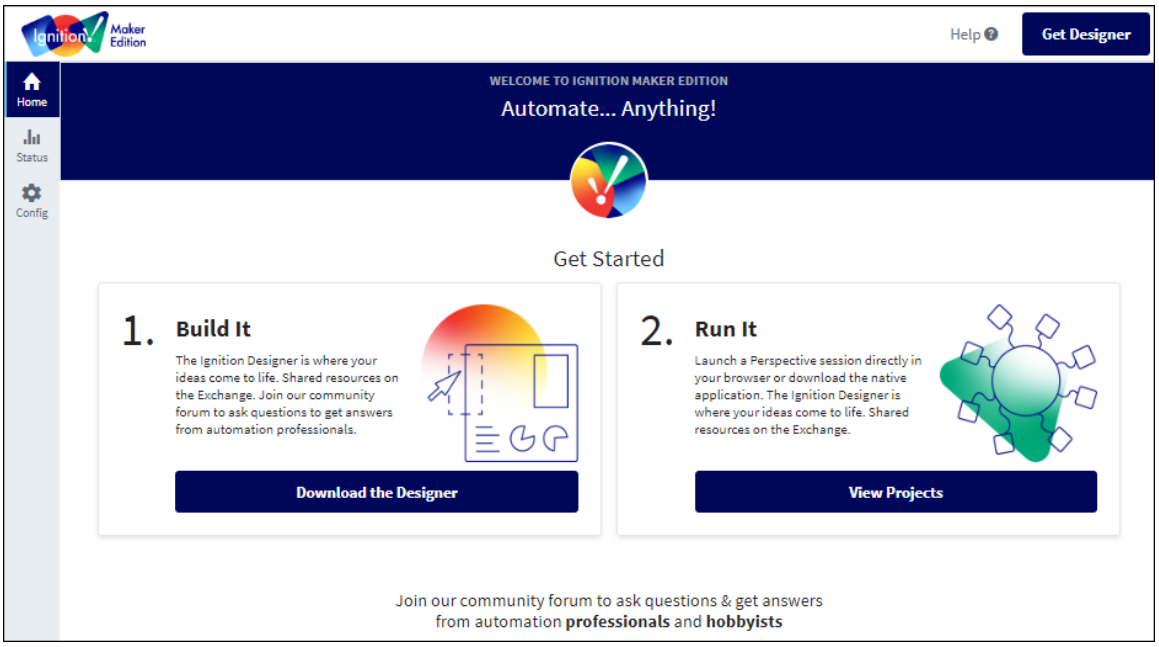
Ignition Maker Edition is a community version of Inductive Automation's Ignition SCADA application. The Maker Edition enables hobbyists, and those folks who want to explore and get inspired to build and automate their own projects in their home. Ignition Maker Edition is a free license edition that can be strictly used for non-commercial and personal educational use. Users are free to make use of all the modules and features of the product while creating their own project.

> Note that for educational institutions, Inductive Automation has other options for a classroom setting or senior projects. Please contact our sales team for more information on these options.

INDUCTIVE UNIVERSITY

Getting Started with Maker

Watch the Video



What Can Maker Edition Do?

Maker Edition can do almost anything standard Ignition can do. You can seamlessly connect all your data, design any kind of application with ease, and instantly web-deploy clients to anyone, anywhere — all from one platform.

How Maker Edition Compares to Standard Edition

Make Edition differs from Standard Edition Ignition in the following ways:

- Maker licenses are free. Get one now in your IA account.
- Maker Gateways can support a maximum of 10 Perspective sessions.
- Maker Gateways can run up to 10,000 Tags, not counting Tags in remote Tag providers.
- In regards to redundancy, Maker Gateways can only be set to the Independent mode.
- Limited module support: Maker Gateways are designed to only run appropriate modules.

The table below represents Inductive Automation modules that are ready to use on Maker Gateways. Modules that aren't allowed to run on Maker will be removed during Maker's commissioning process.

Module Name	Description

Alarm Notification	Provides alarm notifications via email.
Allen Bradley Driver Suite	Allen-Bradley driver suite for the OPC UA module.
Logix Driver	A driver for communicating with Allen-Bradley Logix5000 series PLCs, and includes firmware version 21 support.
Modbus Driver	A driver for communicating with devices via Modbus-TCP.
Omron Driver	Drivers for Omron PLCs.
OPC UA	Provides Ignition's OPC UA client and server functionality.
Perspective	A module that provides modern, responsive html based graphical interfaces for Ignition projects.
Reporting	Adds reporting capability.
Serial Support (Gateway)	Adds platform-level serial comm support and system.serial functions to gateway-scoped scripts.
SFC	Sequential Function Charts are customizable logic execution environments.
Siemens Driver	Siemens S7-300, S7-400 and S7-1200 drivers.
SQL Bridge	An OPC-to-SQL data logger and transaction manager.
Tag Historian	Turns any database into a powerful historian that can store and drive data in Maker Edition.
Twilio	Send SMS alarm notifications without a physical device allowing you to send SMS message via python script.
UDP/TCP Driver	Drivers for receiving and parsing UDP or TCP packets.
WebDev	Allows you to create and serve basic web resources and respond to web requests through scripting.

Ignition Edge

> The following feature is new in Ignition version **8.0.10**
> Click here to check out the other new features

Ignition Edge was revised and expanded significantly as of release 8.0.10. Deprecated pages for earlier functionality can be found here.

Welcome to Edge

Ignition Edge is a lightweight, lean version of Ignition with software solutions designed for devices used in the field and OEM devices at the edge of the network. Ignition Edge solutions work on a wide variety of operating systems such as Linux, Windows (any version), and macOS. With support for ARM processors, Ignition Edge can run on devices like a Raspberry Pi or the latest generation of edge-of-network devices.

Core functionality for Edge includes the following:

- All Inductive Automation OPC UA drivers (AB, Siemens, Modbus, Omron, DNP3, UDP /TCP, serial, BACnet)
- One-week data buffer
- Two device connections
- One-way email notification
- Gateway Network connections (actual Gateway Network services are enabled with IIoT, EAM, and Sync Services)
- Limited to one project, one Tag Provider, and one Historical Tag Provider

Edge Onboard

Ignition Edge comes pre-installed on a range of edge-of-the-network devices that are IIoT ready out of the box. Edge can also be installed on virtually any industrial device.

Licensing

The new Edge version of Ignition has a separate license that works in the same way as a traditional Ignition license. The Edge license will not work if you attempt to activate it on an Ignition Gateway as it is only set up to license an Edge Gateway. However, you can activate a full Ignition license on an Edge Gateway, but this is not recommended due to the limitations of the Edge platform. Additionally, once a license has been activated on an Edge Gateway, the trial time for any unlicensed Edge plugins will be removed, and you will not be able to access them.

On this page

...

- Welcome to Edge
  - Edge Onboard
  - Licensing
- Core Functionality
  - OPC UA Module and Drivers
  - One Week of Internal Storage
  - Projects
  - Redundancy
- Edge Plugins
- Edge Gateway
- Ignition Edge Compared with Ignition
  - Database Access
  - Gateway Scripting
  - Gateway and Project Backups
  - Third-Party Modules

INDUCTIVI
UNIVERSIT

Edge Architecture

Watch the Video

Core Functionality

The following features are available on all Edge Gateways, license or unlicensed.

OPC UA Module and Drivers

Edge can create OPC UA client connections. The internal OPC UA server in the Edge Gateway acts only as a client and not as a server. This means Edge can connect to other outside OPC servers as a client to fetch data, but can not act as an OPC UA server for outside programs to connect to. Adding the IIoT plugin extends this functionality.

One Week of Internal Storage

An Edge installation can store certain historical data inside Ignition.

- **Tag History -** Up to one week or 10 million data points of history can be stored locally. Pruning will automatically happen, and the historian is automatically enabled and not configurable. An Edge Gateway can only have one realtime Tag provider and one Edge historian. You can learn more about Tag history here.
- **Alarm Journal** - The alarm journal will store all alarm events locally for one week of events. There are limited configuration options; the Alarm Journal can be accessed via the Edge Alarm Journal profile. The Edge alarm journal and audit log can sync under EAM.
- **Audit Logs -** Like the Alarms, the Audit log is limited to one week of local storage. It cannot be configured.

Projects

Edge installations have just one project. This project is created automatically if it did not exist, and it cannot be removed. The default project name is "Edge", however as of version 8.0.8, the Edge project can be renamed. For more information, see Edge Gateway Config section.

Redundancy

Two Edge Gateways can be set up with redundancy. An Edge Gateway can only failover to another Edge Gateway (not a standard Ignition Gateway).

Edge Plugins

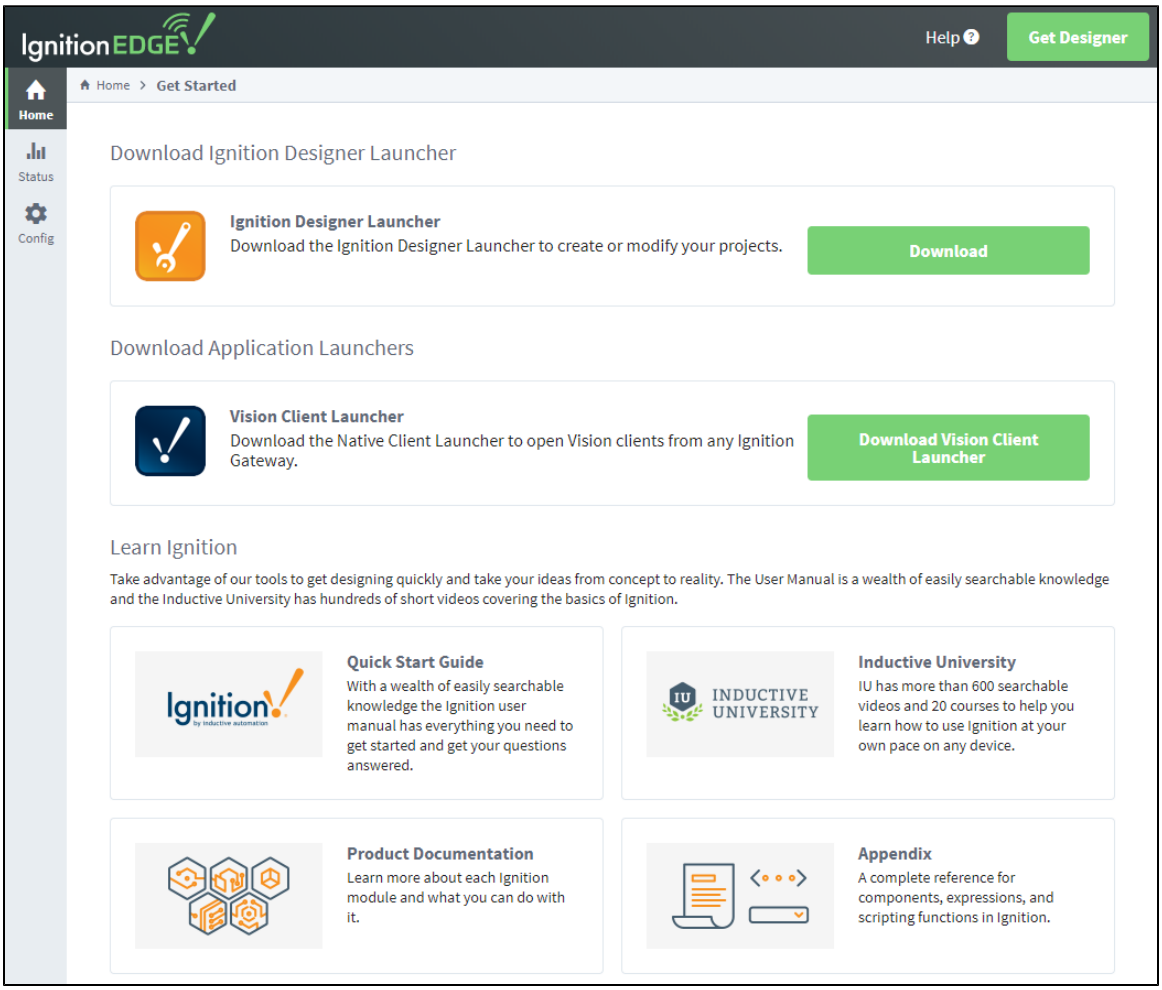
Plugins extend Edge's capabilities, each adding a unique set of features. There are five Ignition Edge plugins available: **Panel**, **IIoT**, **Compute** , **Sync Services**, and **EAM**. Plugins can be mixed and matched to create the system you need. Please note that a license is required to activate each of the Ignition Edge Plugins. Each Edge plugin offers specific functionality that works best in certain scenarios, but because any combination of plugins can be installed, an Edge Gateway may fill multiple roles and help bridge different architectures.

	Ignition Edge Panel creates a standalone HMI at the edge of network. Either Vision or Perspective may be used. The Web Browser module is included with Vision.
	Ignition Edge IIoT turns any device into an Edge Gateway that publishes data to an MQTT broker. (IIoT was formerly Edge MQTT.) The plugin allows access to the MQTT Transmission, Opto22 SNAP PAC driver, and the Emerson ROC driver modules (manual installation required). In addition, Ignition Edge IIoT exposes Edge's OPC UA server to external OPC UA clients.
	Ignition Edge Compute enables you to remotely run scripts and create REST APIs for interfacing with third-party applications. The plugin allows access to Python scripting, Gateway Event Scripts, and the Web Dev module. The Compute Plugin is new for release 8.0.10.
	Ignition Edge Sync Services synchronizes data from the edge of the network to a central Ignition server. The plugin allows access to Sync Services, remote Tags and alarm pipelines, Tag history sync, alarm history sync, and audit log sync.

EAM	Ignition Edge EAM brings diagnostics, automatic backup and recovery, central licensing, and project synchronization to your edge-of-network device. The EAM plugin allows this Edge Gateway to be an Agent in an EAM architecture.

Edge Gateway

All Edge installations are a limited version of the Ignition platform for a significantly reduced cost. This helps keep things simple, and these installations always work with other full versions of Ignition if you need additional functionality. In an Edge installation there are a lot of visual changes to the Gateway Webpage, and some pages have been removed to avoid confusion. You can see below that after installing Edge, the traditional orange Ignition logos are replaced with new green Edge logos so a Gateway can be easily identified. For more information, see Edge Gateway.



Ignition Edge Compared with Ignition

Because Edge is a lean solution, it does have some functionality limitations compared to full Ignition. Ignition and Ignition Edge share the same basic platform, but Edge is a leaner version of Ignition made specifically for use in on edge-of-network devices. Ignition comes with unlimited Tags, Clients, and database connections, while Ignition Edge comes with unlimited Tags, two Clients (one local and one remote) and no database connectivity.

Database Access

Database access is disabled. This means that any database queries or bindings will not work, and the user sources that require a database connection will not be available. The Status and Configure pages of the Gateway Webpage have been updated to remove these features and reduce confusion. Despite this restriction, some "historical" storage can happen inside of Ignition.

Gateway Scripting

All Gateway level scripting is disabled including Gateway Event Scripts and Tag Event scripts. This does not affect project or client scoped scripts. If you wanted to take advantage of Gateway Events Scripts and Tag Events Scripts with an Edge installation, you can do so through the use of the Compute Plugin. Keep in mind that you will not see the “Gateway Events” menu item in the Designer on unlicensed Gateways. Applying a compute license will require a Designer restart before the menu item will appear.

Gateway and Project Backups

> This feature was changed in Ignition version **8.0.15**:

A Gateway backup made in an Edge installation will work in other Edge installations, or if needed, Standard Edition Gateways as well. Although restoring a Standard Edition Gateway backup is possible in an Edge installation, it is not recommended due to the limitations of the Edge platform. You can, however, still export tags, windows, and other project resources in the Designer from Standard Edition Gateways and import them into an Edge project, or vice versa.

Third-Party Modules

Edge's core functionality and plugins largely determine which modules run on the platform. Third-party modules will not run on an Edge Gateway, with the exception any third-party module explicitly stated by a plugin, such as the Edge IIoT Plugin.

In This Section ...

Edge Gateway

The following feature is new in Ignition version **8.0.10**
Click here to check out the other new features

Ignition Edge was revised and expanded significantly as of release 8.0.10.

The Edge Gateway has many of the same options as the regular Ignition Gateway Webpage along with a few Edge -specific settings. There are three tabs on the left: Home, Status, and Config.

Home

The first time you go to the Edge Gateway Webpage, it shows you several steps to help you get started. Once you're up and running, the **Home** tab lets you download the Designer Launcher or Vision Client Launcher, or view a Perspective project. There are some resource links to help you get started with Ignition quickly: Inductive University, Quick Start Guide or Product Documentation where you can learn about Ignition modules, as well as the Appendix which contains a complete reference for components , expressions, and scripting functions.

When you're first introduced to Ignition, the Homepage is the landing page. It's where you'll find the Designer Launcher, Vision Client Launcher, and Perspective Sessions.

- The Designer Launcher locates all Gateways that are available on your local network. Once you open the Designer, you can access the Edge project.
- The Vision Client Launcher opens Vision Clients from any Ignition Gateway.
- The Perspective Sessions opens Perspective sessions from this Gateway.

If you're running Perspective as the visualization method for your project, you can launch a Perspective Session from the Designer, from the Gateway Webpage, or by entering the web address of the project in your web browser.

On this page

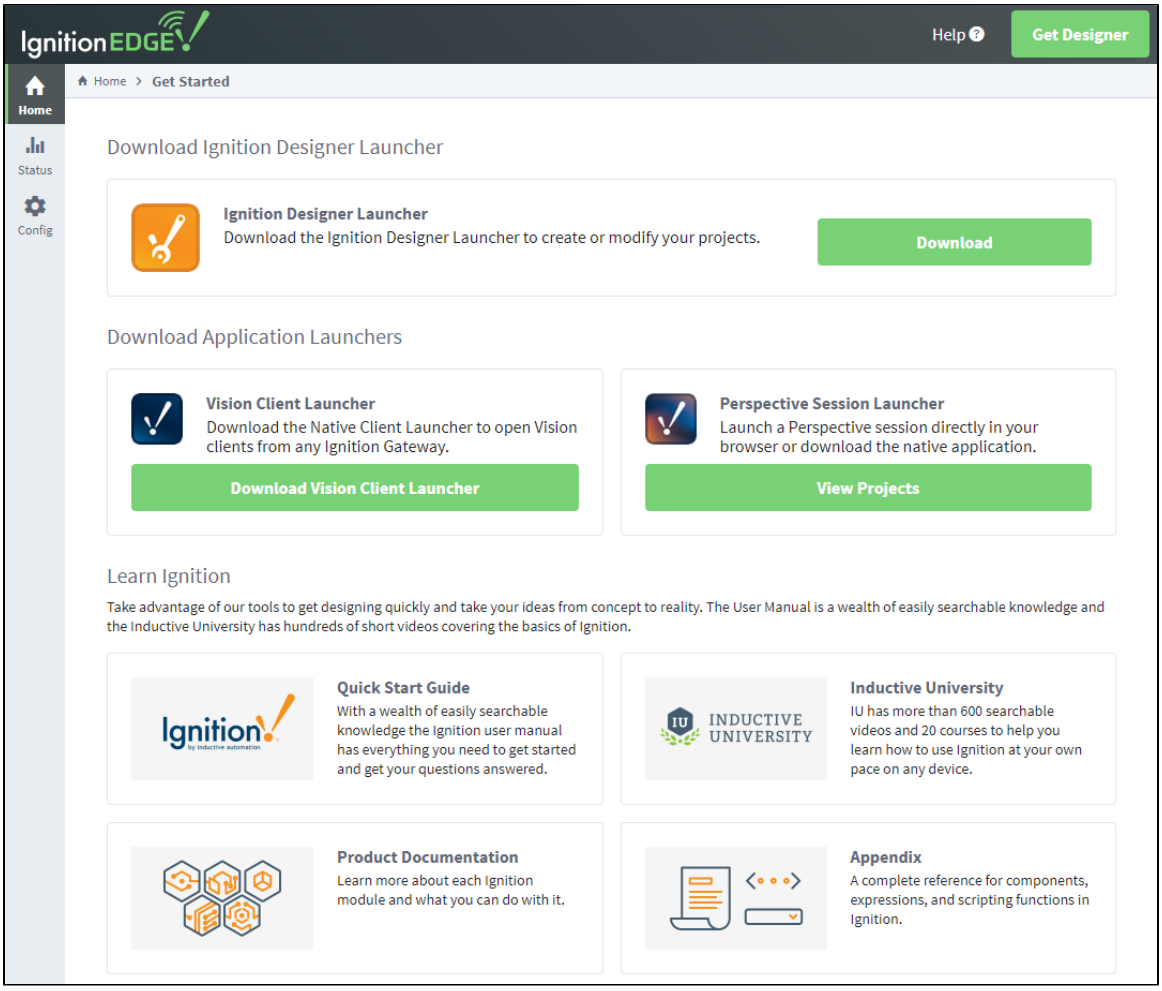
...

- Home
- Status
- Config
  - Edge Gateway Settings

INDUCTIVI
UNIVERSIT

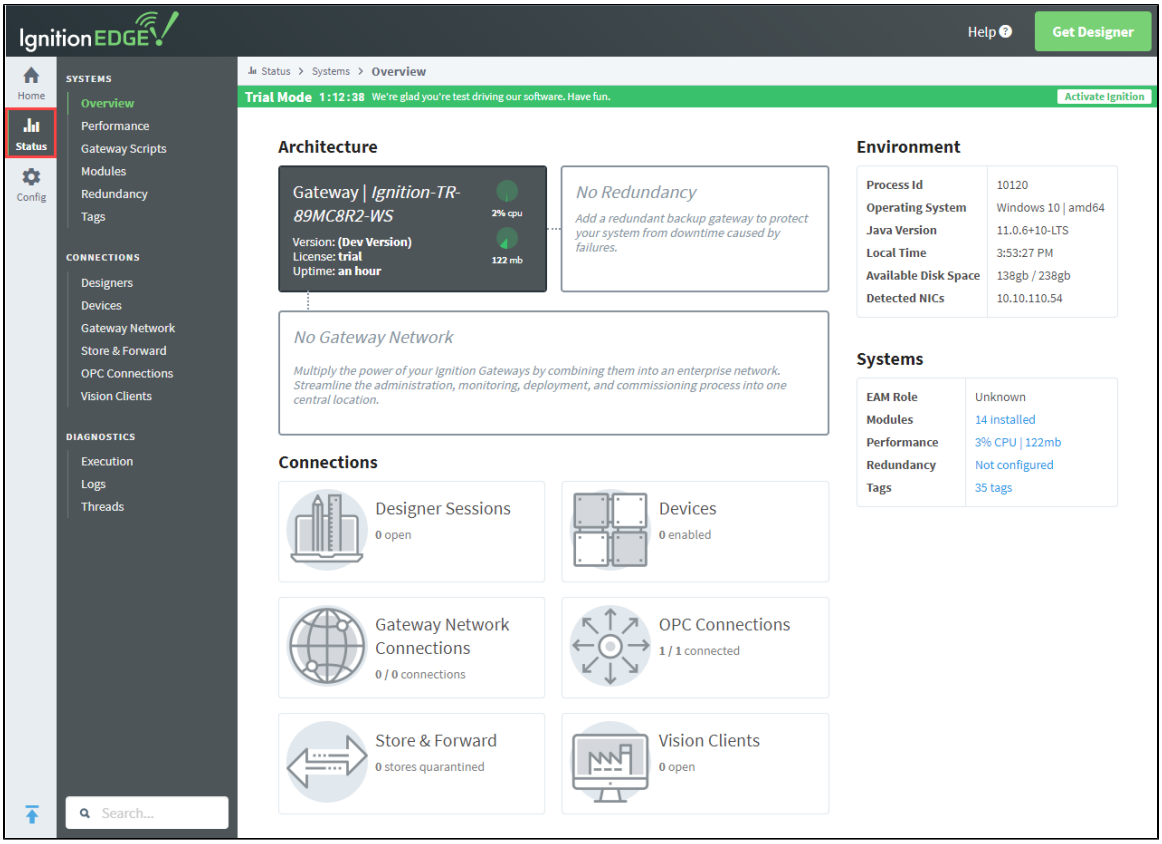
Ignition Edge Configuration

Watch the Video



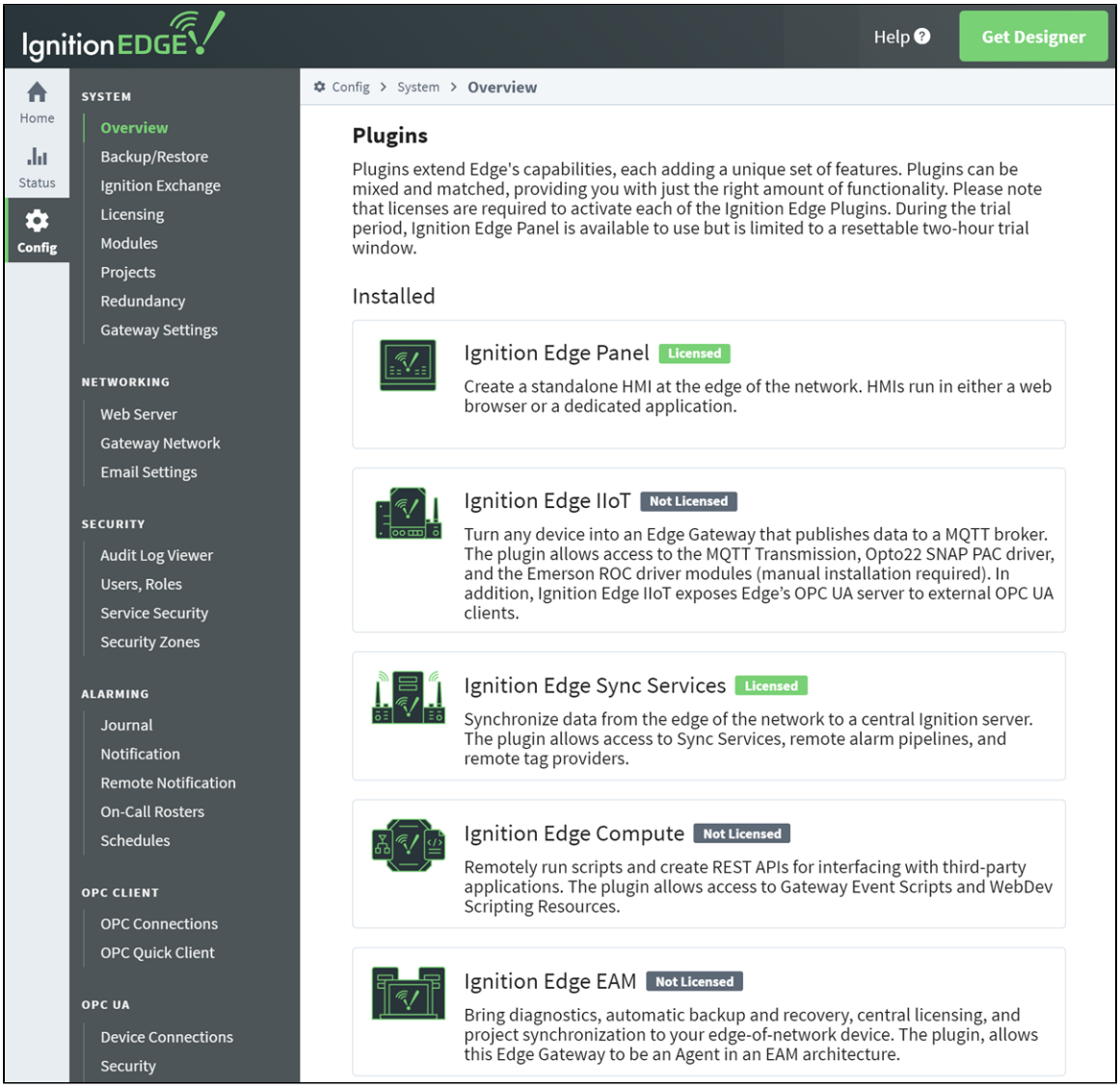
Status

The Status tab provides an overview of all of the systems in your Gateway while also enabling you to drill down into specific systems and get an in-depth view of what is currently happening. For complete information, see Ignition Gateway Status.



Config

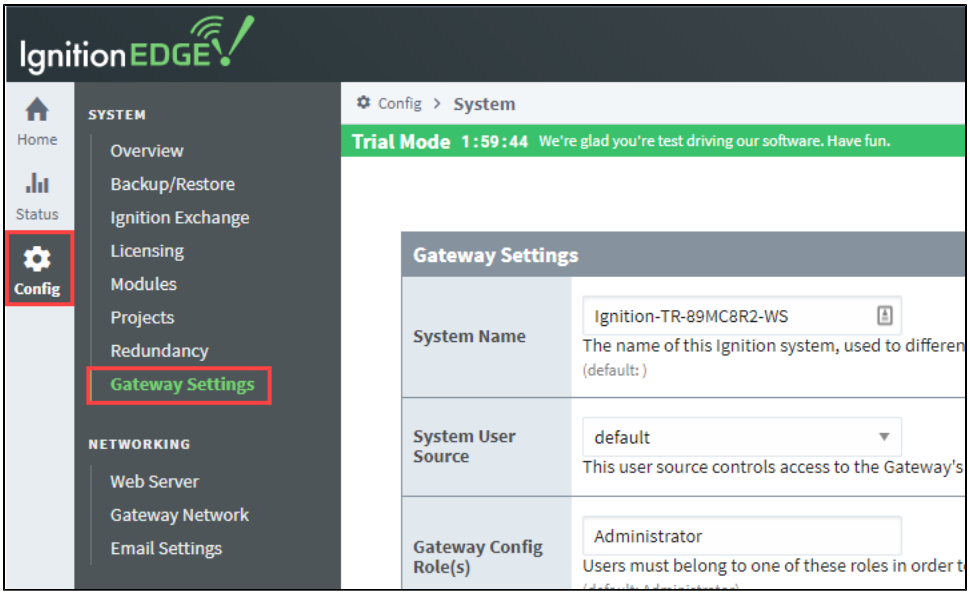
The Config tab provides access to configuration options for Gateway settings. This is where most of the settings that affect the whole Gateway are set up. Note that some settings such as database connections are not available for Edge. For complete information, see Ignition Gateway Config.



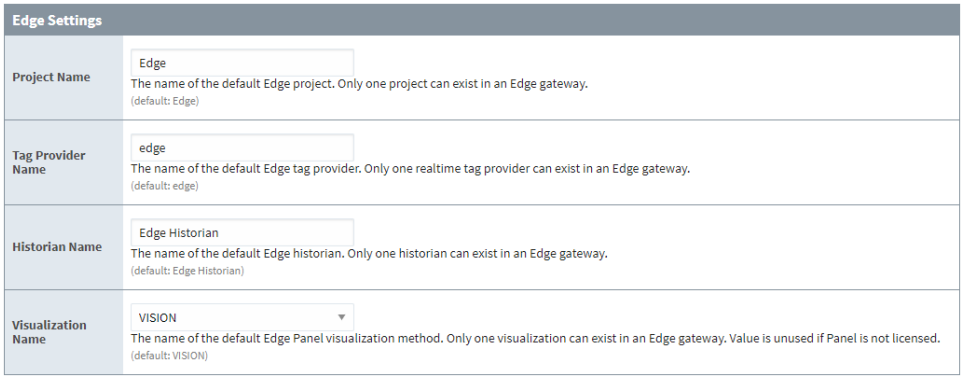
Edge Gateway Settings

As of version 8.0.10, there is an additional set of Gateway Settings for Edge that are not available in regular Ignition. These settings enable you to modify the names of the Edge project, Edge Tag provider, and Edge historian. You can also set the visualization method (Vision or Perspective) here if you are using the Edge Panel plugin.

1. To get to the settings, choose **Config > Gateway Settings** on the Edge Gateway Webpage.



2. Scroll down to the **Edge Settings** section.



3. Change the settings as desired, then click **Save Changes** at the bottom of the screen.

The following table describes the Edge settings.

Edge Setting	Description
Project Name	The following feature is new in Ignition version **8.0.8** Click here to check out the other new features The name of the Edge project. Default is "Edge." Only one project can exist in an Edge Gateway. As of release 8.0.8, you can change the name of this project with this setting.
Tag Provider Name	The name of the Edge Tag provider. Only one provider can exist in an Edge Gateway.
Historian Name	The name of the Edge Historian. Only one historian can exist in an Edge Gateway.
Visualization Name	The following feature is new in Ignition version **8.0.10** Click here to check out the other new features The name of the default Edge Panel visualization method (Vision or Perspective). Only one visualization system can be used in an Edge gateway. This value is unused if the Edge Panel plugin is not licensed. Default is Vision.

Related Topics ...

- Edge Panel
- Edge IIoT
- Edge Compute
- Edge Sync Services
- Edge EAM

Edge Panel

Ignition Edge was revised and expanded significantly as of release 8.0.10. Deprecated pages for the plugins can be found here.

Ignition Edge Panel enables standalone HMI functionality for one local client and one remote client at the edge of the network. Ignition Edge Panel gives you a choice between the Vision or Perspective modules. Ignition Edge Panel comes standard with the Ignition Designer. Panel provides up to two launchable clients for your Edge project, one local and one remote can be open at the same time.

Ignition Edge Panel includes one week of data buffering for trending and local client fallback for mission-critical applications. The Panel plugin provides a remote terminal for operators, usually to act as a backup in case there is a network failure.

Ignition Edge Panel can send one-way alarm notifications by email through your company's SMTP server, so you can stay alert to what's happening at the edge of the network.

During the trial period, Ignition Edge Panel is available to use but is limited to a resettable two-hour trial window.

On this page ...

- Visualization Module
  - Running Vision on Panel
  - Running Perspective on Panel
- One-Way Alarm Notification

Visualization Module

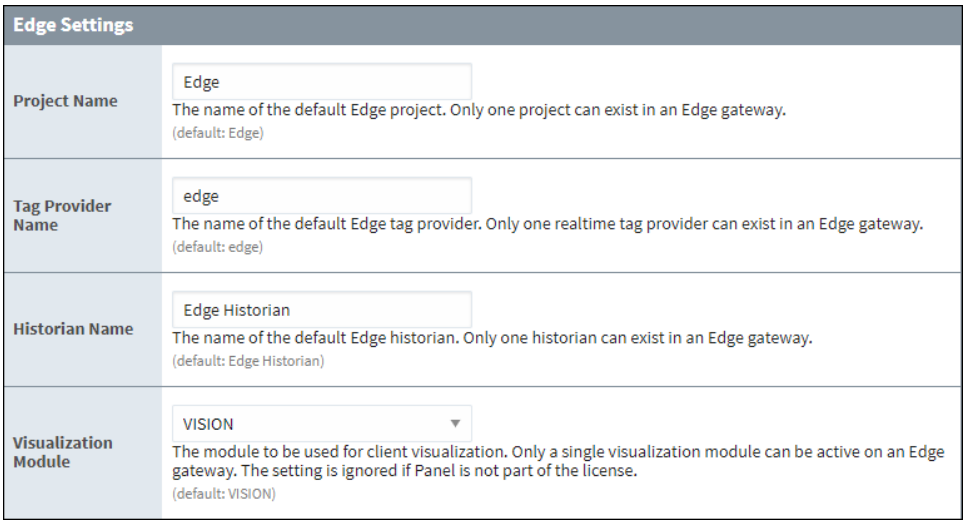
An Edge Gateway with the Panel plugin can launch either Perspective sessions or Vision clients, but not both. Ignition Edge Panel comes standard with the Ignition Designer.

Licensed Edge Gateways that don't have the plugin will refuse to launch either runtime. In addition, switching visualization to one module will fault the other, preventing the Designer from accessing resources to the faulted module (may require a Designer restart).

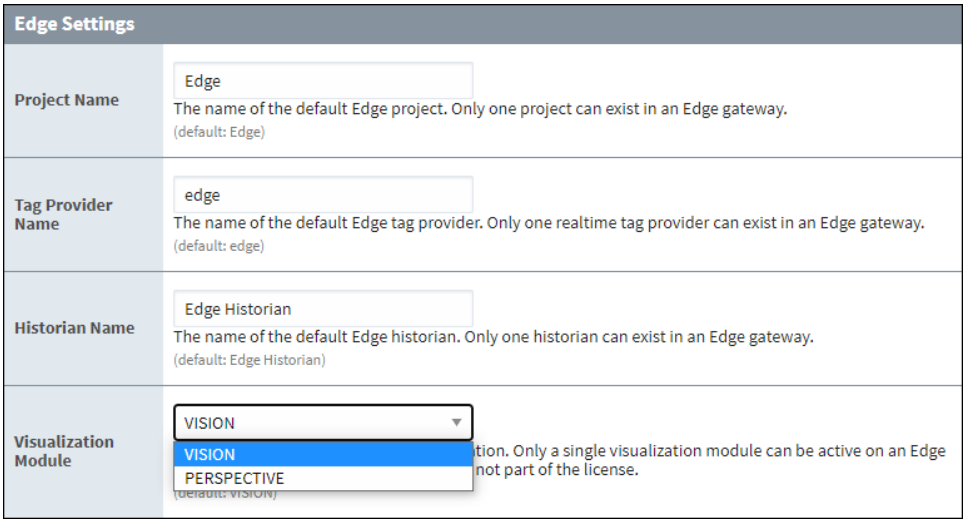
By default, the visualization module is Vision. To change the visualization module, do the following:

1. One the Edge Gateway go to **Config > Gateway Settings**.

2. Scroll down to the **Edge Settings** section.



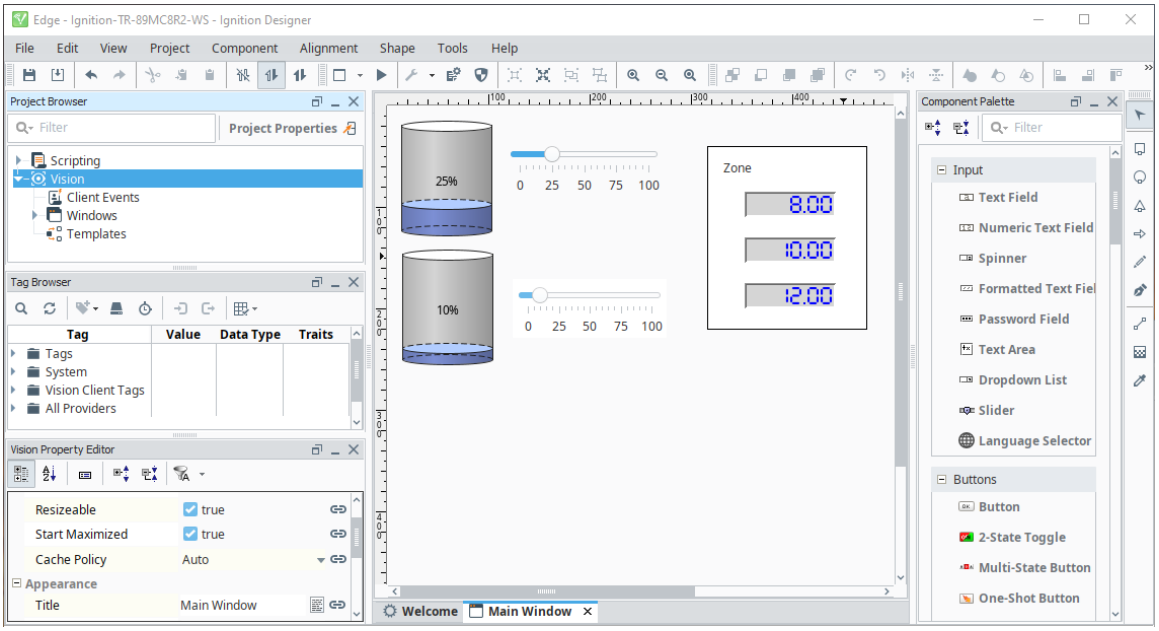
3. Under Visualization Module, click the **Expand** ▼ arrow then select the visualization module you want (Vision or Perspective).



4. Click **Save Changes.**

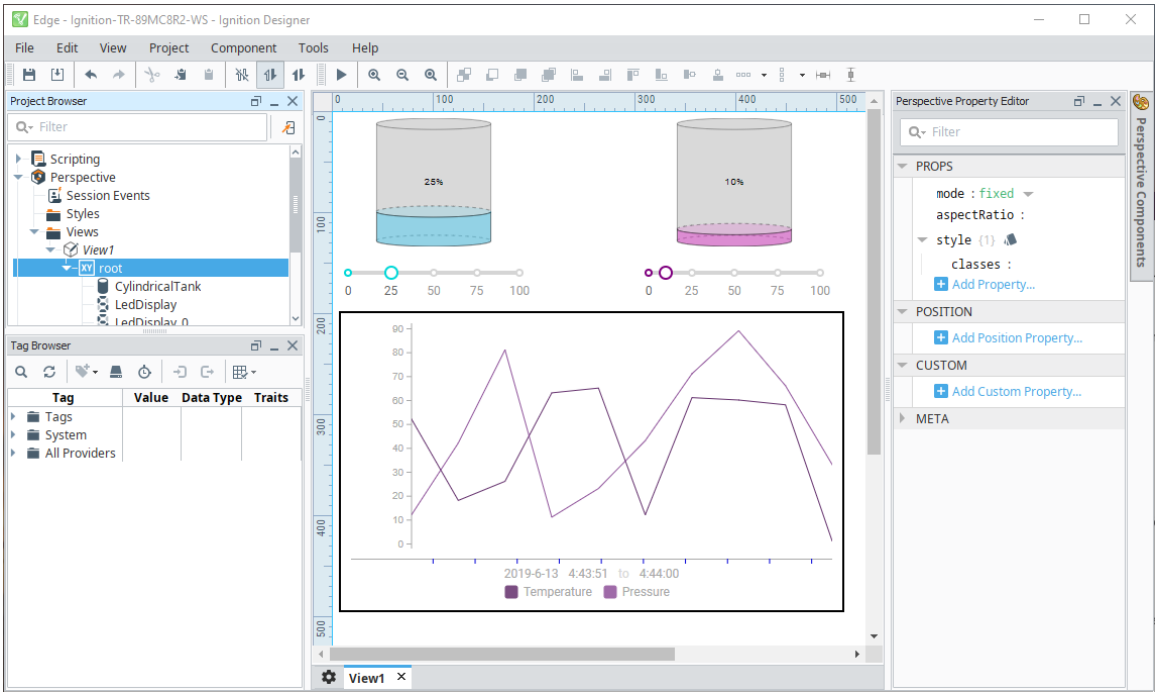
Running Vision on Panel

When the visualization module is set to Vision, only two clients are allowed: one local and one remote. The Web Browser Module is included with Vision. The following is an example of Vision running in Designer on Edge:



Running Perspective on Panel

When the visualization module is set to Perspective, only two sessions are allowed. The following is an example of Perspective running in Designer on Edge:



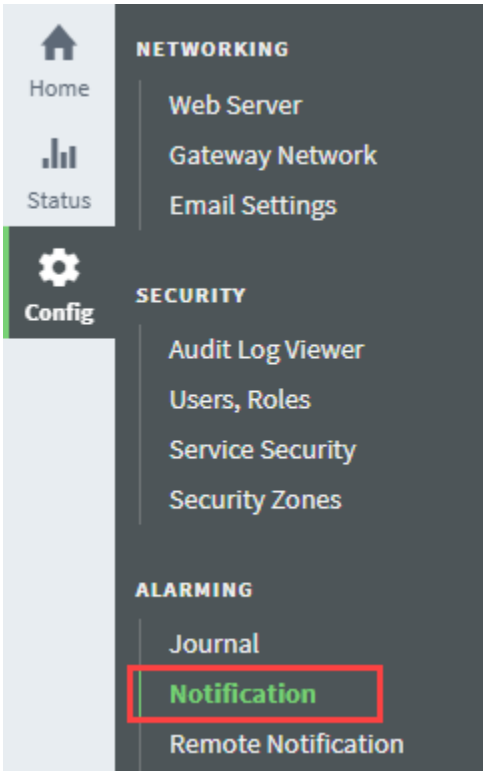
One-Way Alarm Notification

Ignition Edge Panel can send one-way alarm notifications by email through your company's SMTP server, so you can stay alert to what's happening at the edge of the network. Panel provides alarm notification in a limited form. This means the Edge Gateway has only one email notification profile and no alarm pipelines. Ignition Edge Panel includes one week of data buffering for trending and local client fallback for mission-critical applications.

Data buffer is limited to 10 millions rows, does not include database support.

To set up alarm notification, do the following:

1. On the Edge Gateway Webpage select **Config > Alarming > Notification**.



2. On the Alarming Notification screen, choose the SMTP profile and On-Call Roster.
3. Set the Alarm Criteria as desired.
4. Click **Save**.

Property Name	Description
Enabled	Whether notification is enabled. (Default is true.)
SMTP Profile	The SMTP profile that will be used to distribute the notifications.
On-Call Roster	The Roster of users that will be notified when an alarm meets the criteria set in this profile.
Minimum Priority	For a Notification to be sent from this Notification Profile, its priority must be equal to or higher than the priority specified on this property. (Diagnostic, Low, Medium, High, Critical)
Maximum Priority	For a Notification to be sent from this Notification Profile, its priority must be equal to or lower than the priority specified on this property. (Diagnostic, Low, Medium, High, Critical)

For more information, see Simple One-Way Email Notification Profile.

Edge IIoT

The following feature is new in Ignition version **8.0.10**
Click here to check out the other new features

The IIoT plugin (formerly known as MQTT) allows this copy of Edge to push data directly into a Broker using the fast and lightweight MQTT data transfer protocol. In effect, this makes your computer (and whatever devices are attached to it) into a remote MQTT publisher. If you are already rolling out new smart devices that connect to your Broker, this allows you to connect all of your legacy devices into the same system.

IIoT plugin also allows UA clients to connect to the UA server on Edge.

Edge IIoT activates the following modules:

- MQTT Transmission
- Opto22 SNAP PAC driver
- Emerson ROC driver
- ABB TotalFlow

On this page ...

- Remote Tag Provider
- Expose Edge's UA Server
- Install MQTT modules to Activte IIoT Plugin
- Install IIoT Opto22 SNAP Pac Driver or the Emerson ROC Driver

Remote Tag Provider

With Edge IIoT installed, Ignition Gateways will be able to create remote Tag providers, targeting a provider on an Edge Gateway: either the system provider or the one realtime Tag provider.

For more information about remote Tag providers, see Tag Providers.

The Edge Sync Services plugin also unlocks this feature.

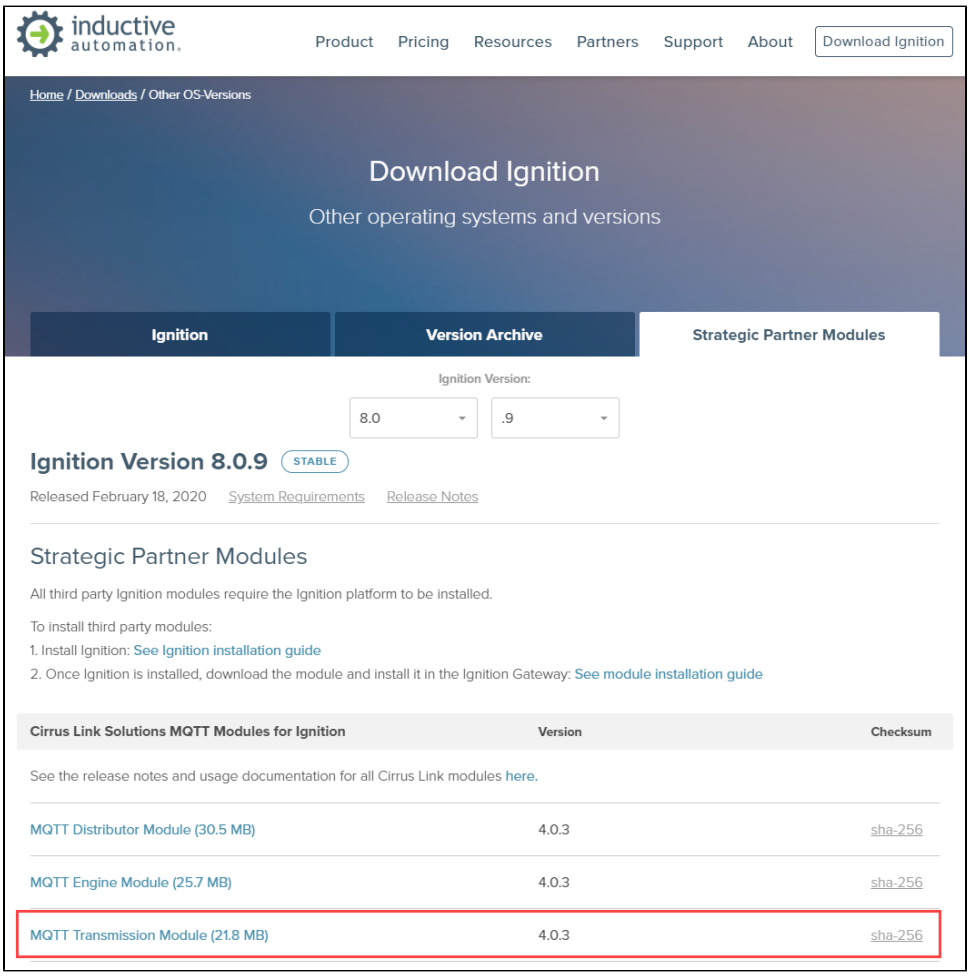
Expose Edge's UA Server

The IIoT plugin enables UA clients to connect to the UA server on Edge. Applying this plugin will require that the OPC UA module is restarted before the UA server will be available publicly.

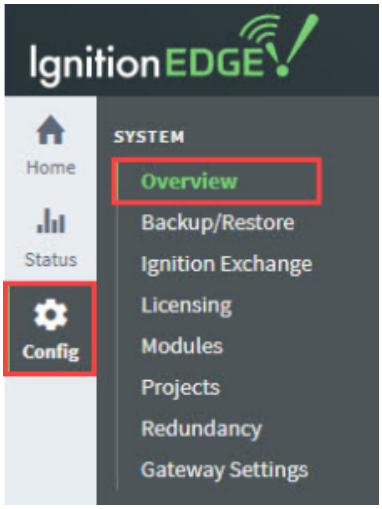
Install MQTT modules to Activte IIoT Plugin

After you install Edge, you need to download and install the MQTT Transmission module from Cirrus Link. You can use the free trial as long as you want so you can build and test full solutions before you buy.

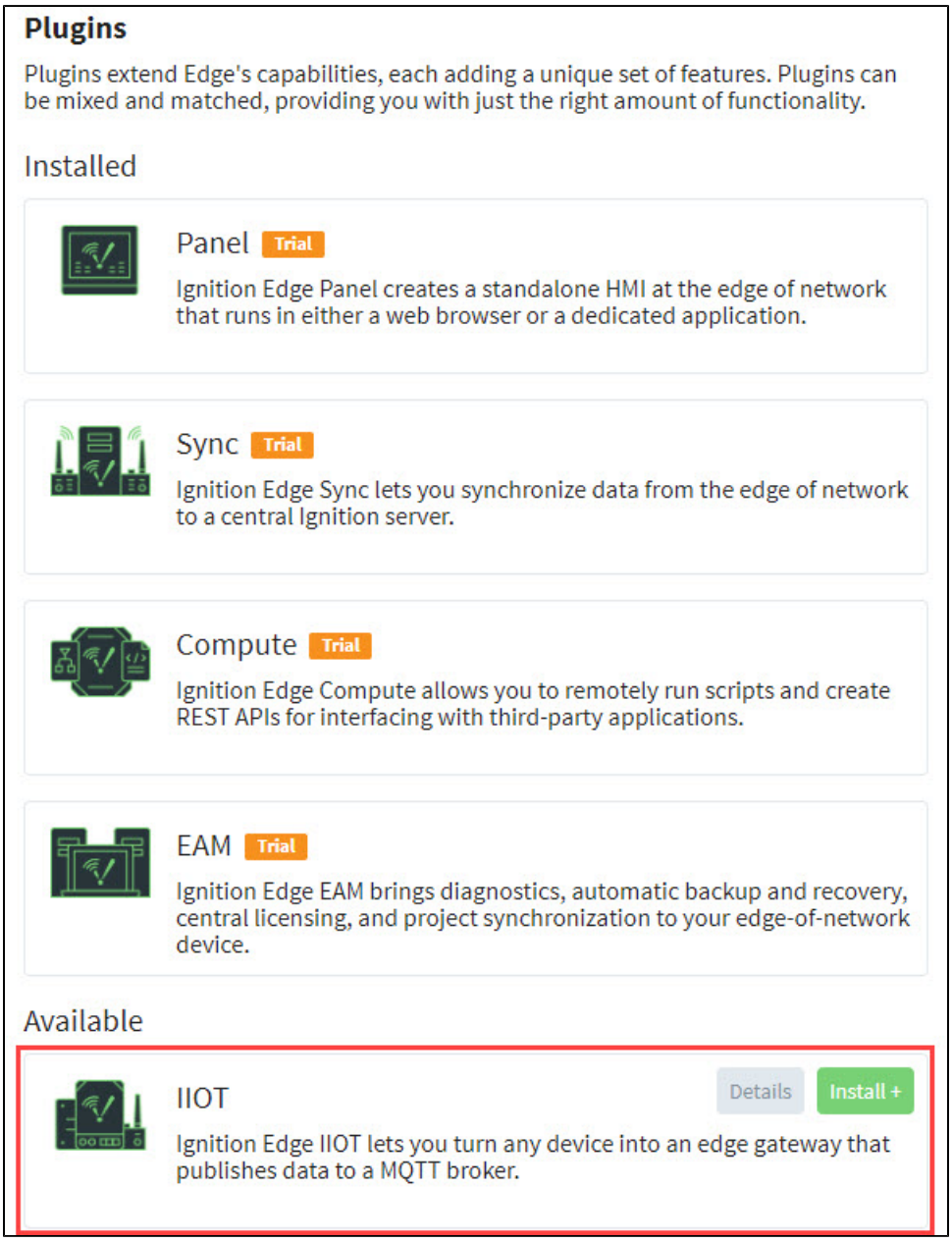
1. To install the Ignition Edge IIoT plugin, first download the MQTT Transmission module from the Inductive Automation website.



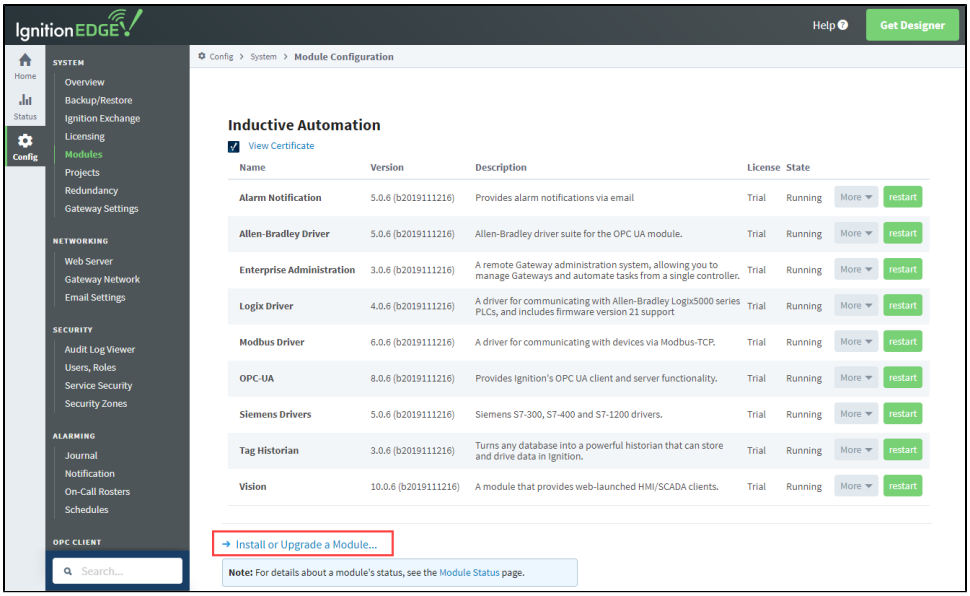
2. Next go to the Config tab of your Edge Gateway.



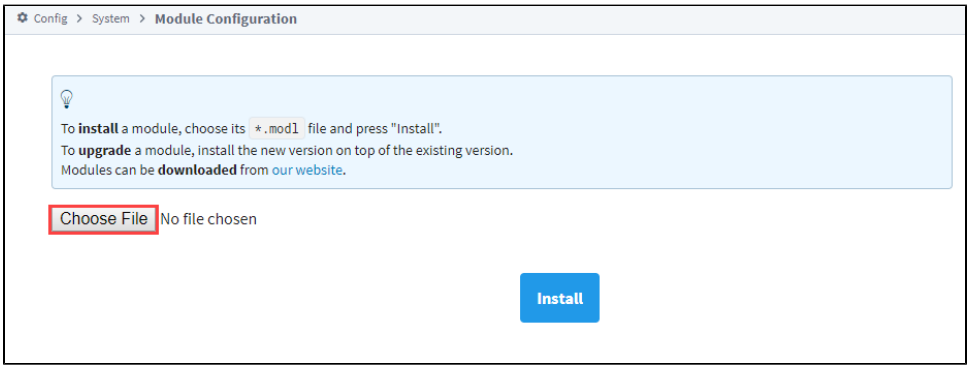
3. On the Overview page, scroll down to the MQTT and click the **Install+** button.



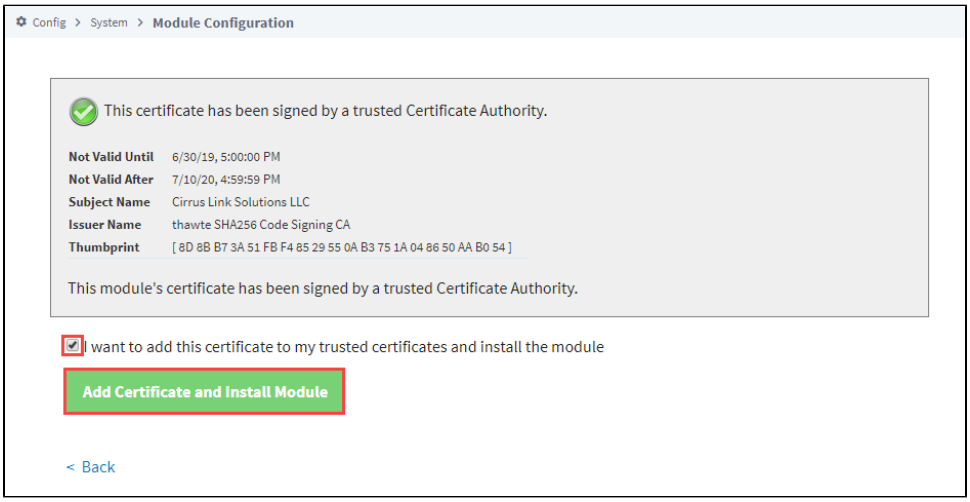
4. Scroll down and click on **Install** or **Upgrade a Module...**



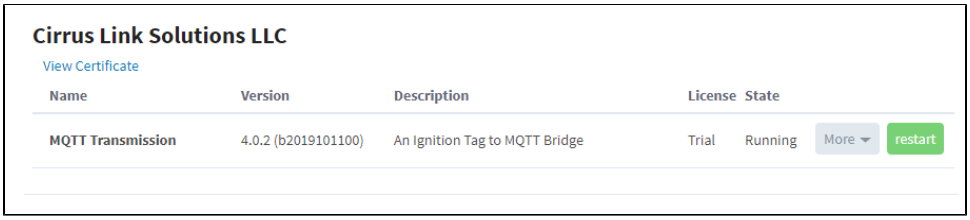
5. Next click on **Choose file**.



6. Navigate to the folder where you downloaded MQTT-Transmission-signed.modl, select the file, and click **Open**.
7. Click the **Install** button.
8. At the License agreement, select **I accept the terms in the License Agreement**, then click the **Accept License** button.
9. At the Trusted Certificates window, click **I want to add this certificate to my trusted certificates and install the module**.
10. Click the **Add Certificate and Install Module** button.



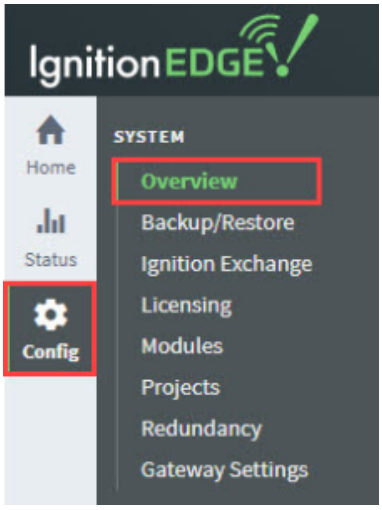
11. You'll see a message indicating the module was successfully installed. And the Module will show up on the Module Configuration page in the Edge Gateway now.



For more information, check out the documentation on the Cirrus Link website https://docs.chariot.io/display/CLD.

Install IIoT Opto22 SNAP Pac Driver or the Emerson ROC Driver

1. To install the Opto22 SNAP Pac driver or Emerson ROC driver, first download the driver you want from the Inductive Automation website.
2. Next go to the Config tab of your Edge Gateway.



3. Scroll down and click on **Install** or **Upgrade a Module...**
4. Next click on **Choose file**.
5. Navigate to the folder where you downloaded the driver(s) and select the file.
6. Click the **Install** button.
7. At the License agreement, select **I accept the terms in the License Agreement**, then click the **Accept License** button.
8. You'll see a message indicating the driver module was successfully installed. And the module will show up on the Module Configuration page in the Edge Gateway now, for example the following image shows the Opto22 SNAP Pac driver installed.

Edge Compute

The following feature is new in Ignition version **8.0.10**
Click here to check out the other new features

Ignition Edge Compute is a new plugin for Ignition Edge release 8.0.10. It brings true edge computing to your network. Compute enables you to program directly against the web server in the Ignition Edge Gateway. You can remotely run scripts and create REST APIs for interfacing with third-party applications. It also provides Gateway Network Messaging.

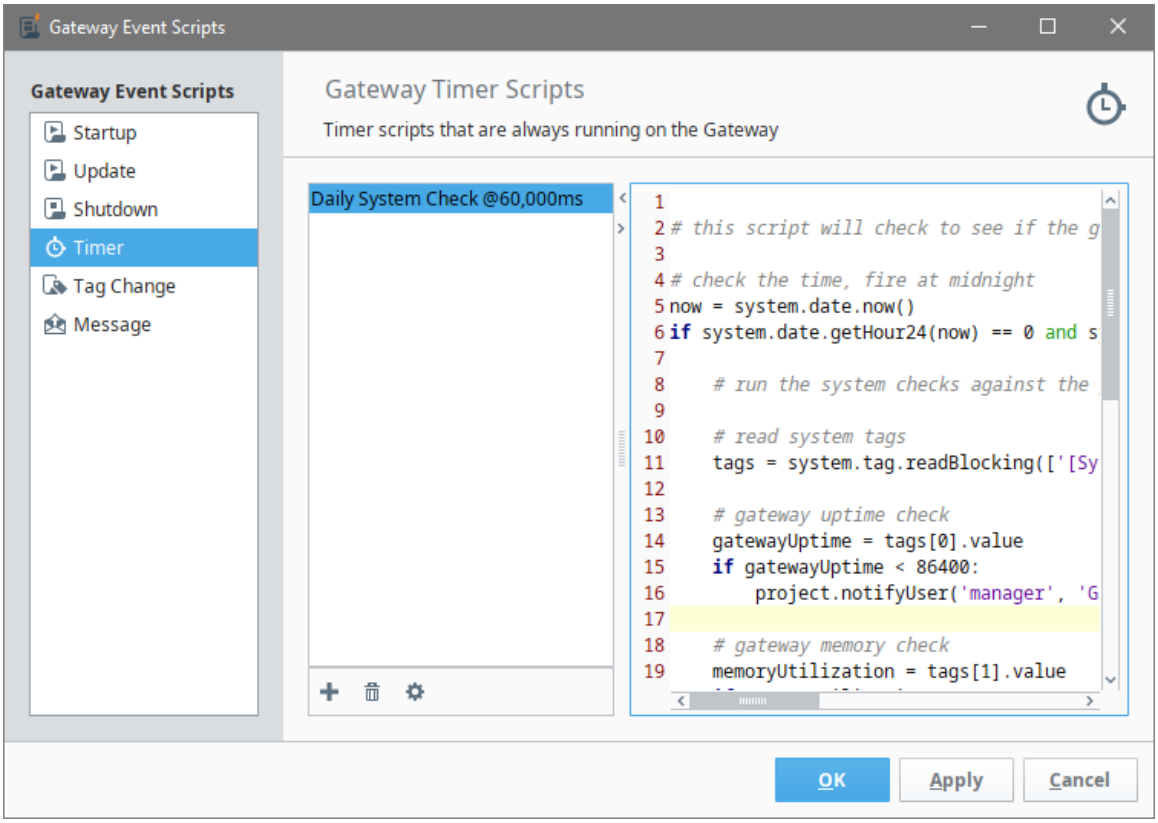
Ignition Edge Compute gives your edge devices the ability to run scripts without the need to depend on the central server. Python is an easy to read, versatile programming language. Even if you don't know it at all, you will probably be able to understand a basic Python script. See Python Scripting for complete information.

On this page ...

- Scripting in Edge Compute
- Edge Gateway Network Messaging
- WebDev Module

Scripting in Edge Compute

Gateway events are unlocked when the Compute plugin is applied. This allows you to use any of the Gateway scoped Event scripts for your project. You will not see the “Gateway Events” menu item in the Designer on unlicensed Gateways. Applying a compute license will require a Designer restart before the menu item will appear.



Edge Gateway Network Messaging

Edge Compute provides access to call message handlers on other Gateways. This allows you to send data to other Gateways to be dealt with. You can access this functionality through the following system.util messaging functions:

- system.util.sendMessage
- system.util.sendRequest
- system.util.sendRequestAsync

WebDev Module

Edge Compute comes with the WebDev module, which enables you to directly program against the web server inside the Ignition Edge Gateway and systems running Vision Clients. This gives you the ability to host web pages or files, and create full-fledged REST APIs that allow external systems to interact with the Ignition server.

Edge Sync Services

The following feature is new in Ignition version **8.0.10**
Click here to check out the other new features

Ignition Edge Sync Services acts as a limited remote server that synchronizes data from the edge of the network to a central Ignition server.

- Remote tags
- Tag history sync
- Alarm journal sync
- Audit log sync
- Remote alarm notification profiles

Sync Services can be used with the EAM plugin or separately. For an example of architecture using Sync Services, see Edge Architectures.

On this page ...

- Agent Gateway
- Store and Forward
- Data Synchronization
- Sync Reset
- Remote Alarm Notification Profiles
- Remote Tag Provider

Agent Gateway

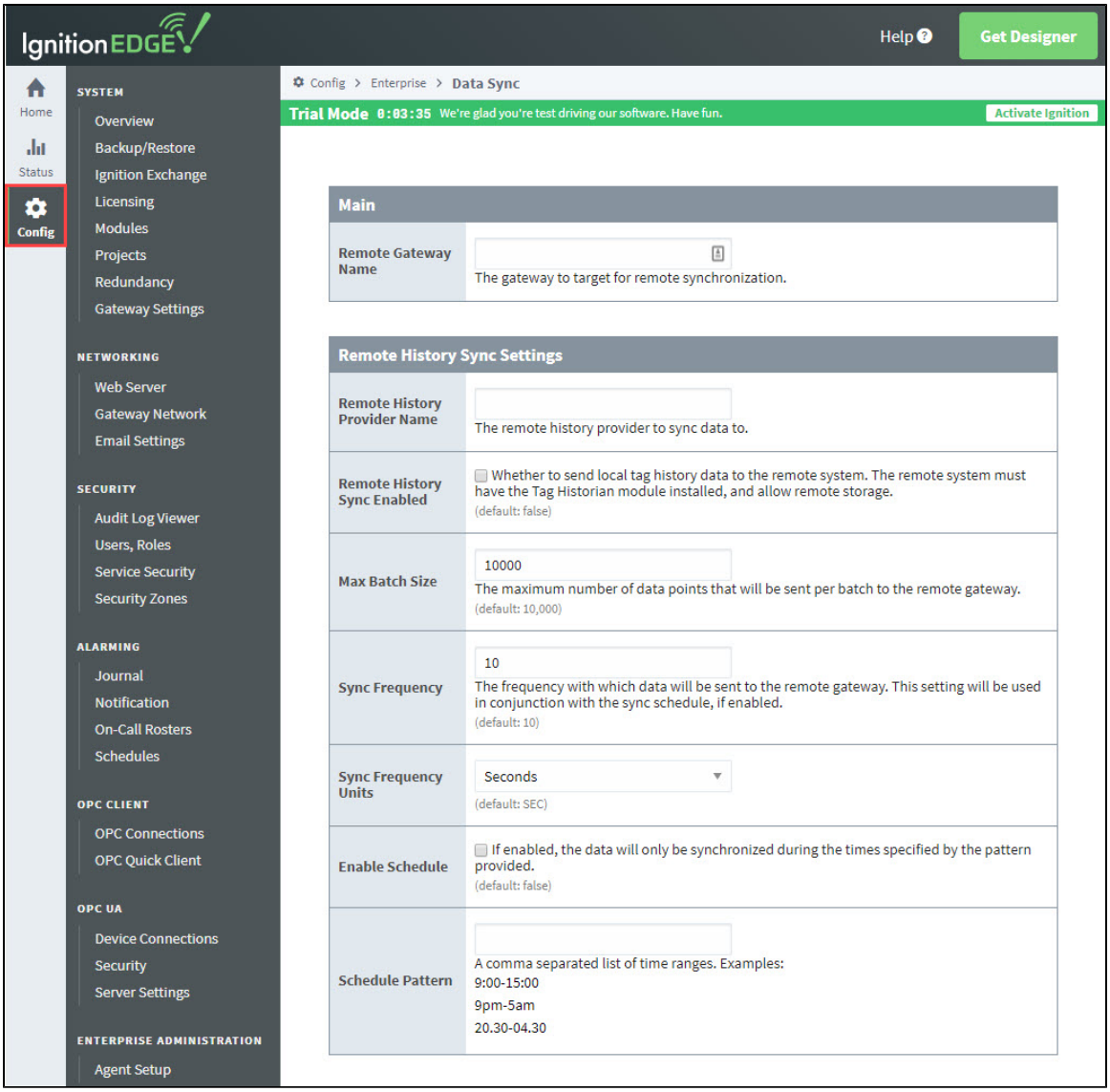
Ignition Edge Sync Services, acting as an Agent Gateway in a secure, high-performance Ignition Gateway Network, can share information with other Ignition Gateways through Distributed Services, including distributed tags, history, and remote alarming.

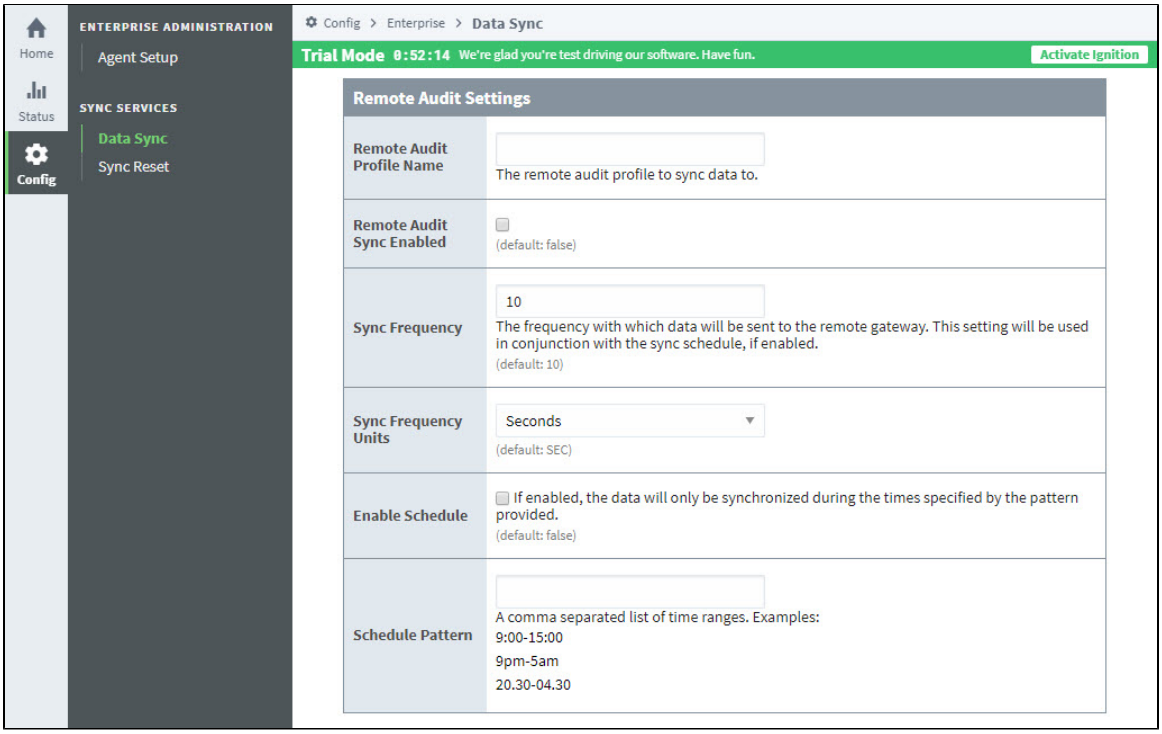
Store and Forward

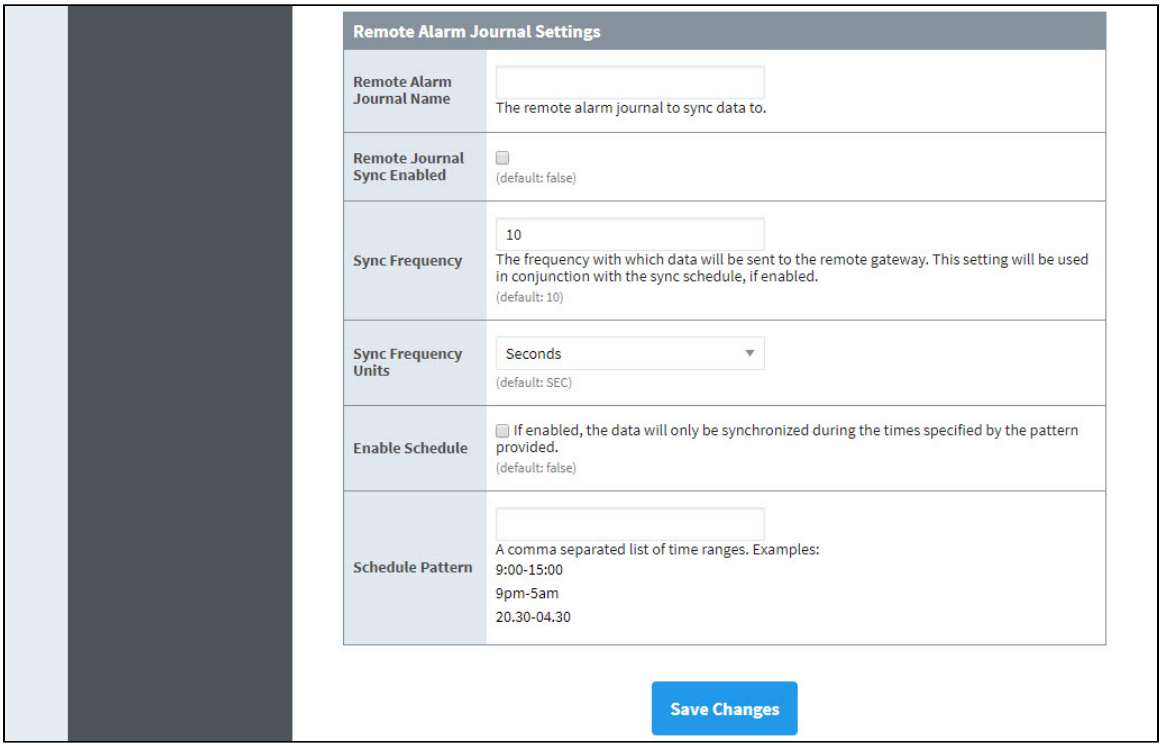
Ignition Edge Sync Services comes with up to one-week of data buffering so it can be used to store-and-forward your data to a central server in case of network failure.

Data Synchronization

The Sync Services plugin introduces Tag History Synchronization. Tag History Synchronization allows you to sync your history data from the Edge Gateway to a full Tag History Provider on an Ignition Gateway, allowing you to save past the one week data limit. The Edge Gateway will still only see the one week of data that it can hold internally, but it will allow you to store the data in a more permanent database that can be viewed from other Ignition systems. This is setup by navigating to the Tag History Sync page link at the bottom of the Edge Gateway Configure page.


IgnitionEDGE
Help
Get Designer
Home
Status
Config
SYSTEM
Overview
Backup/Restore
Ignition Exchange
Licensing
Modules
Projects
Redundancy
Gateway Settings
NETWORKING
Web Server
Gateway Network
Email Settings
SECURITY
Audit Log Viewer
Users, Roles
Service Security
Security Zones
ALARMING
Journal
Notification
On-Call Rosters
Schedules
OPC CLIENT
OPC Connections
OPC Quick Client
OPC UA
Device Connections
Security
Server Settings
ENTERPRISE ADMINISTRATION
Agent Setup
Config > Enterprise > Data Sync
Trial Mode 0:03:35 We're glad you're test driving our software. Have fun.
Activate Ignition
Main
Remote Gateway Name
The gateway to target for remote synchronization.
Remote History Sync Settings
Remote History Provider Name
The remote history provider to sync data to.
Remote History Sync Enabled
Whether to send local tag history data to the remote system. The remote system must have the Tag Historian module installed, and allow remote storage.
(default: false)
Max Batch Size
10000
The maximum number of data points that will be sent per batch to the remote gateway.
(default: 10,000)
Sync Frequency
10
The frequency with which data will be sent to the remote gateway. This setting will be used in conjunction with the sync schedule, if enabled.
(default: 10)
Sync Frequency Units
Seconds
(default: SEC)
Enable Schedule
If enabled, the data will only be synchronized during the times specified by the pattern provided.
(default: false)
Schedule Pattern
A comma separated list of time ranges. Examples:
9:00-15:00
9pm-5am
20.30-04.30





The following table describes the tag history synchronization properties.

Main	
Settings	
Remote Gateway Name	The gateway to target for remote synchronization.

Remote History Sync Enabled	
Remote History Provider Name	The remote history provider to sync data to.
Remote History Sync Enabled	Whether to send local tag history data to the remote system. The remote system must have the Tag Historian module installed, and allow remote storage. Default is false.
Max Batch Size	The maximum number of data points that will be sent per batch to the remote gateway.
Sync Frequency	The frequency with which data will be sent to the remote gateway. This setting will be used in conjunction with the sync schedule, if enabled.
Sync Frequency Units	Units are defined as milliseconds, seconds, minutes, hours, days, weeks, months, and years.
Enable Schedule	If enabled, the data will only be synchronized during the times specified by the pattern provided.
Schedule Pattern	A comma separated list of time ranges that will determine when data will be sent over the network.
Remote Audit Settings	
Remote Audit Profile Name	The remote audit profile to sync data to.
Remote Audit Sync Enabled	Whether Remote Audit Sync is enabled. Default is false.
Sync Frequency	The frequency with which data will be sent to the remote gateway. This setting will be used in conjunction with the sync schedule, if enabled.
Sync Frequency Units	Units are defined as milliseconds, seconds, minutes, hours, days, weeks, months, and years.
Enable Schedule	If enabled, the data will only be synchronized during the times specified by the pattern provided.
Schedule Pattern	A comma separated list of time ranges that will determine when data will be sent over the network.
Remote Alarm Journal Settings	
Remote Alarm Journal Name	The remote alarm journal to sync data to.
Remote Journal Sync Enabled	Whether Remote Alarm Journal settings are enabled. Default is false.
Sync Frequency	The frequency with which data will be sent to the remote gateway. This setting will be used in conjunction with the sync schedule, if enabled.
Sync Frequency Units	Units are defined as milliseconds, seconds, minutes, hours, days, weeks, months, and years.
Enable Schedule	If enabled, the data will only be synchronized during the times specified by the pattern provided.
Schedule Pattern	A comma separated list of time ranges that will determine when data will be sent over the network.

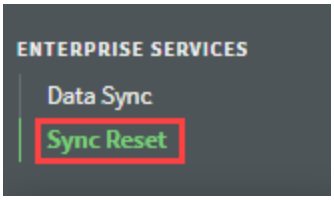
Tag History Sync is unique in that if the connection to the remote Ignition Gateway is severed, the Edge Gateway will use its week of storage as a Store and Forward buffer, allowing you to store up to a week of data before data is lost. Once the connection is restored, the Edge Gateway will send data over based on the max batch size and data frequency until all previous data is sent.

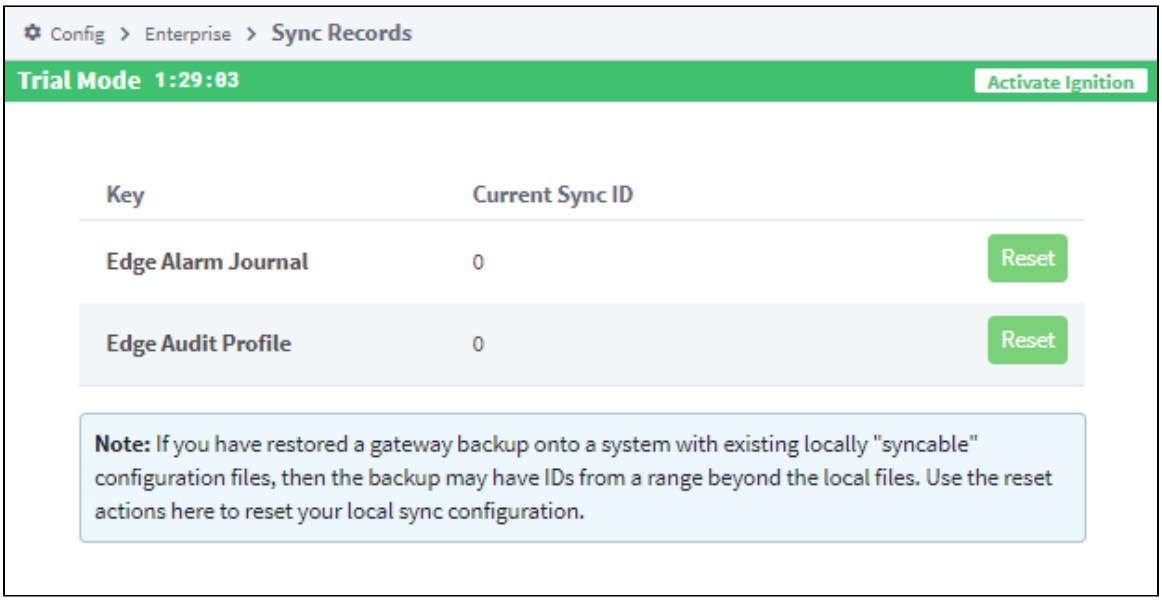
> The following feature is new in Ignition version **8.0.7**
> Click here to check out the other new features

Sync Reset

Added a way to reset synchronization state for internal historian providers.
If you restored a Gateway backup onto a system with existing locally "syncable" configuration files, then the backup may have IDs from a range beyond the local files. Use the Reset Sync actions to reset your local sync configuration.

To do a **Sync Reset**, go to Edge Gateways under **Config > Enterprise Services > Sync Reset**.





Remote Alarm Notification Profiles

Sync Services allows remote alarm notification profiles for remote pipelines only

- No alarm pipelines locally
- Requires a remote alarm notification profile be configured on the Edge Gateway.

For more information, see Alarm Notification Profiles.

Remote Tag Provider

With Edge Sync Services installed, Ignition Gateways will be able to create remote tag providers, targeting a provider on an Edge Gateway: either the system provider or the one realtime tag provider.

⚠ The Edge IIoT plugin also unlocks this feature.

Edge EAM

The following feature is new in Ignition version **8.0.10**
Click here to check out the other new features

The EAM plugin allows an Edge Gateway to act as an agent in EAM architecture. The Edge Gateway can only be an agent, not a controller. EAM enables you to bring diagnostics, automatic backup and recovery, central licensing, and project and Tag synchronization to your edge-of-network device.

The EAM plugin activates the Enterprise Administration module. Some features of Ignition Edge EAM require the Enterprise Administration module to be installed on the central Ignition Gateway.

For an example of architecture using EAM, see Edge Architectures.

On this page ...

- Gateway Backup and Recovery

Gateway Backup and Recovery

The EAM enables automated backup and quick recovery for all Gateways through a Gateway backup archive.

Modules Overview

Start Small or Large, and Grow From There

Get the functionality you need now, with the freedom to seamlessly add on to your system and match your company's ever-growing needs. The key to Ignition's flexibility is its modular architecture, which means you can buy different modules to perfectly fit your company's requirements.

Ignition is a platform that allows you to put together any system you want, and its modules are the building blocks. You can load all of them for a complete system that can do anything, or pick and choose only the ones you need right now.

On this page

...

- Start Small or Large, and Grow From There
- What are Modules?
  - Why Modules?
- What Can Modules Do?
- Third-Party Modules
- Can I Create My Own Modules?

What are Modules?

Modules are additional files that allow Ignition to do more. Much like adding a new tool to the workbench, adding new modules extends Ignition's capabilities while offering new ways to approach and solve problems. Most of the main features of Ignition are actually provided by different modules such as the Perspective, Vision and SQL Bridge modules.

Modules integrate seamlessly into the system and provide things like new Designer workspaces, new Gateway settings, and new drivers. Newly installed modules immediately integrate with other modules, so installation is quick and painless.

About Ignition's Modules

Watch the Video

Why Modules?

The modules in Ignition offer many benefits.

- **Flexible Licensing**
  Only license the modules that you need, saving money and reducing complexity compared to big monolithic applications that try to do everything. At the same time, the modules have been designed to offer a broad swath of functionality, to avoid having too many pieces.
- **Hot-Swappable**
  Modules can be dynamically loaded and unloaded, allowing you to install, remove, and upgrade them without affecting other parts of the system. This can have huge implications for big projects where up-time is important.
- **Increase System Stability**
  Building modules on a common platform means fewer bugs, better isolation, and all around increased stability.
- **Pluggable Module Architecture**
  The pluggable module architecture allows quick integration of new modules into the Ignition platform. From time to time new modules will be released which add additional features. Third-party modules that provide a wide range of functionality are also available.

What Can Modules Do?

Modules are what drive Ignition's functionality and allow for unlimited possibilities. Here are some common examples of what modules can do:

- **Create Screens for all Occasions**
  Dashboards, detail windows, and historical displays can be created with the Vision module. Read real-time values from any OPC-UA server with the OPC-UA module, and display the data for all users to see. Interactive screens with user-driven forms can also be created, so users can easily write back to a PLC and store data in a database.

- **SCADA on the Go**
  Perspective applications can be viewed on mobile devices with the Ignition Perspective App. Travel through the plant floor with quick access to all your applications. Break free from attached panels and monitors by viewing projects from tablets, phones, or any other mobile device.
- **Lightning-Speed Alarm Detection**
  Long gone are the days when an operator must monitor every aspect of the system to look for problems. Ignition can generate alarms when certain conditions are met, such as when temperature readings reach a certain point, or communication with a device has dropped out. Alarms can be easily displayed with pre-built components, and alarm events can be stored into any SQL database. Additionally, with the Notification modules, Ignition can generate messages when an alarm triggers. Send emails, SMS texts, or phone calls to operators once an issue is detected. Users will always know as soon as a problem occurs.
- **Easy Data Collection**
  With either the SQL Bridge or Tag Historian module, Ignition makes data logging a breeze! History collection can be enabled in just a few steps, and the data can be feed into other parts of Ignition, such as charts, tables, and even reports! All history is stored in a SQL database, so there is no need to worry about proprietary systems locking away the data. History can easily be accessed by third-party software such as ERP or SAP systems.
- **Dynamic Reports**
  Instead of printing individual tables or charts, the Reporting module creates powerful, visually stunning, reports on-demand. Historical data recorded by the Tag Historian or SQL Bridge module can be inserted into a report, but data from any SQL database can easily be visualized and presented in a report. Reports can also be scheduled to run automatically, and be delivered in a number of ways such as emails, or saving to a network drive.

Third-Party Modules

In addition to the modules provided by Inductive Automation, several third-party companies have developed their own modules using our Module Software Development Kit. These modules add even more possibilities and functionality to help you design any project.

- **MES**
  Sepasoft offers a suite of Manufacturing Execution Systems (MES) modules that allow you to track OEE, implement a SPC system, and track resources! The core purpose of these modules is to spot deficiencies in your process, and increase profitability by detecting problem areas, or catching problems before they become an issue.
- **MQTT**
  Modules from Cirrus Link bring the power of MQTT into Ignition. These modules expose massive amounts of data-points as tags, and allow Ignition access to data that would otherwise be lost.

Modules from our Strategic Partners can be found on our downloads page.

Can I Create My Own Modules?

Yes! Do you want to help make Ignition better or create something custom for yourself? Check out our SDK documentation where you will find all of the resources necessary to develop your own module. Modules can be as simple as adding scripting functions or components, or as complex as creating whole new systems for Ignition.

In This Section ...

Core Modules

Build any kind of HMI (human machine interface) or SCADA (supervisory control and data acquisition) system using Ignition's powerful offering of HMI/SCADA modules.

What Are Core Modules?

Core Modules are commonly used modules included in a typical installation. Most systems have at least one of these modules installed. Each core module encapsulates a core SCADA or HMI principle such as data-logging or process visualization. These modules are the foundation upon which great projects are built.

SQL Bridge

Bridge the gap between OPC data and SQL databases and get a highly efficient SQL-based Tag historian and transaction manager. Easily log data, call stored procedures, and synchronize data bi-directionally.

Perspective

Create beautiful, mobile-responsive industrial applications that run natively on any device type such as smartphones, tablets, touchscreens, laptops, and desktop computers. It automatically adapts to fit screens of any size and orientation.

Vision

Use this module to launch an unlimited number of powerful Ignition Vision clients anywhere on the network. Display charts, graphs, trends, and HMI screens.

Tag Historian

Turn a SQL database into a high-performance time-series Tag historian.

Reporting

Create dynamic, database-driven PDF reports with pixel-perfect layout - with ease.

OPC UA

Connect to most major PLCs (Programmable Logic Controllers) with the Ignition OPC-UA Module. This module serves as a totally cross-platform OPC-UA server with an open, plugable driver system.

Alarm Notification

Configure the logic for how, why and when alarm notifications are delivered, manage alarm notification for groups of users, and send notifications via email.

SMS Notification

Add text alerts with the SMS Notification Module for even more options.

Sequential Function Charts

Streamline the creation, organization, and visualization of robust logic systems.

Web Dev

Create web-pages that are hosted on Ignition's web server.

Enterprise Administration

The EAM provides a secure and intuitive way to manage many Ignition installations from one location.

In Ignition Version 8, the Symbol Factory module has been combined into the Vision and Perspective modules. This functionality does not exist as a separate module anymore.

In This Section ...

Vision Module

On this page

...

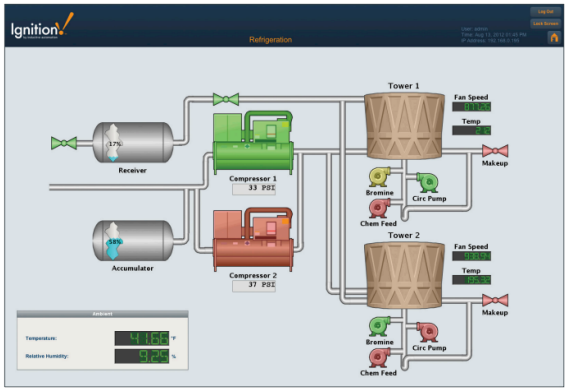
- HMI/SCADA Visualization
  - Get Real-Time & Historical Data on Control Screens
  - Real-Time Status & Control
  - Alarming
  - Analyze Real-Time and Historical Data
  - Database Front-End
  - Designer Included
  - Vector-Based 2D Drawing Tools
  - Image Library
  - Easily Create Animations
  - Speed Development With Templates
  - Python Scripting
  - Secured Via SSL
  - Unlimited Clients & Instant Deployment
  - Update All Screens With One Click
  - View Anywhere, Any Time
- Looking for Less?
- Want to Learn More?

HMI/SCADA Visualization

Get Real-Time & Historical Data on Control Screens

The Vision Module delivers a dynamic, powerful, and beautiful HMI experience unlike any other. Display real-time and historical data on everything from status and control screens to interactive charts and tables so you can analyze data at a glance and quickly act on it. Since all clients are connected to a central server, everyone in your company will stay up to date and can easily access the real-time data they need.

You'll never be limited by how many Clients (runtimes) or screens you can use because the Ignition Vision Module includes unlimited clients per central server. Every member of your company can have insight into operations at no additional cost.



Real-Time Status & Control

Ignition's real-time status and control functionality, coupled with advanced analytic capabilities provided by Ignition's SQL database integration, gives you the power to quickly analyze and easily display the status of your facility.

Alarming

Set up screens to display real-time alarms from Ignition's Alarming system, so your operations team can respond quickly to resolve the issue.

Analyze Real-Time and Historical Data

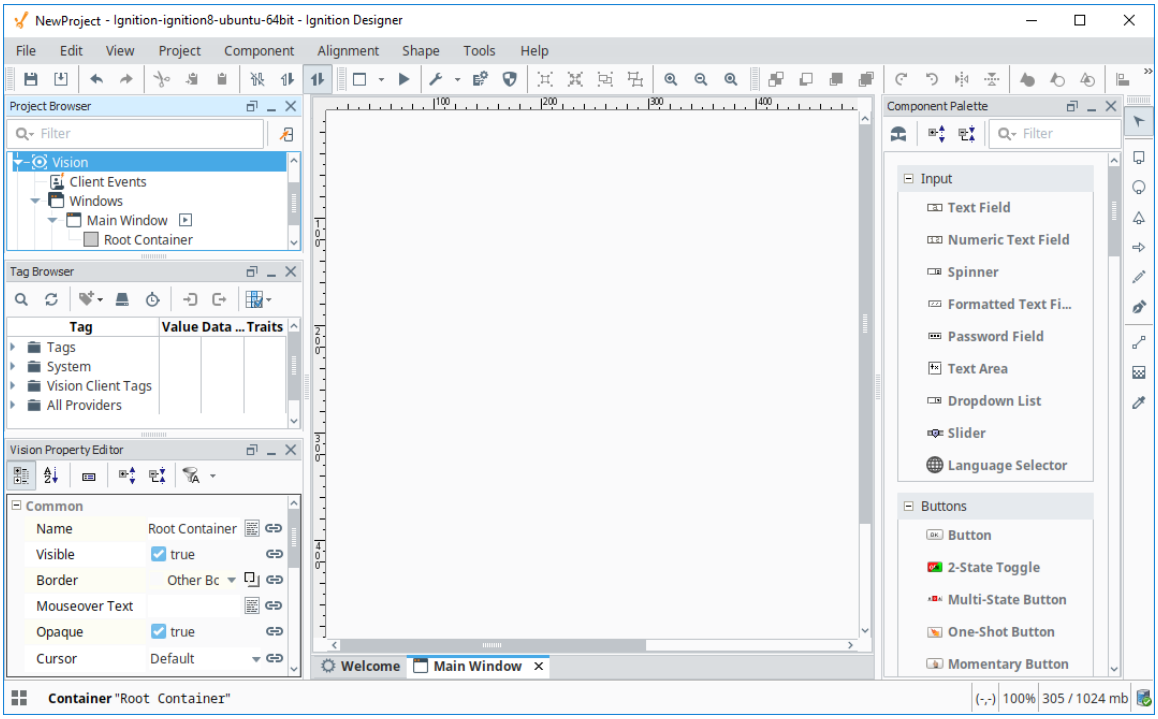
Included customizable charts, tables and graphs display your real-time and historical data, making at a glance analysis easy. You'll be able to quickly set up executive dashboards, track key performance indicators (KPIs), and view trends.

Database Front-End

Easily integrate data from SQL databases with your SCADA system. Data entered by operators can be written directly to any database. The powerful property binding system makes it easy to display information from databases on any screen.

Designer Included

Clients are configured through the included Ignition Designer. Drag-and-drop configuration makes screen development intuitive; simply drag Tags and drop them onto components to instantly bind data to inputs, displays, tables, charts, and more.



Vector-Based 2D Drawing Tools

Create distinctive screen designs and layouts with shape-building tools for lines, polygons, rectangles, arrows, and circles. Vector graphics let you create precision shapes and Bézier curves that look great on any size monitor; shapes scale to any size without losing image quality.

Image Library

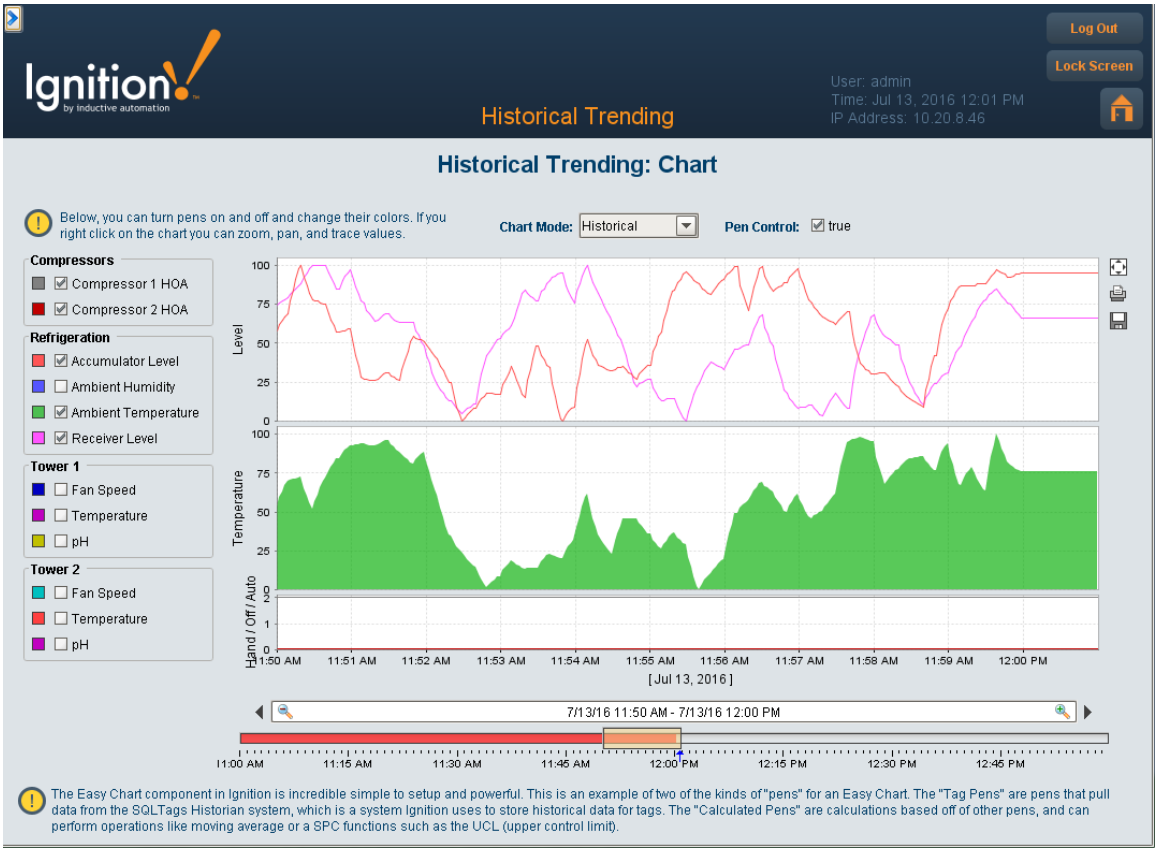
The designer integrates Symbol Factory 2.5 from Software Toolbox. The intuitive interface allows you to browse and search through nearly 4,000 high quality vector graphics symbols. Once you find the symbol you're looking for, simply drag-and-drop it onto your design window to use it right away.

Easily Create Animations

State dependent settings and dynamic positioning lets you bind shape properties to data values so you can easily create animations.

Speed Development With Templates

Develop large projects at record speeds by using component templates. Instances of each template can be placed on as many screens as you want. Any changes made to the template are automatically updated to any instance of that component, saving a huge amount of tedious development time.



Python Scripting

Advanced functionality can be achieved using the built-in Python scripting engine. Python is an extremely popular, powerful, and easy to learn scripting language. Python is highly regarded for its readability, making your projects easy to maintain.

Secured Via SSL

Every client you launch from the Vision Module is secured using SSL protocols. In order to keep your data secure, authentication groups can be used to strictly limit access to certain areas of a project to only the users with appropriate access levels.

Unlimited Clients & Instant Deployment

With the Ignition Vision Module installed on your server, you can launch as many clients as you want, instantly. No individual client install required!

Update All Screens With One Click

With Ignition, managing five clients is just as easy as managing 50. All clients can be updated instantly by making changes to the master project screen on your server, and then clicking the publish button.

View Anywhere, Any Time

View your HMI / SCADA system from anywhere, any time, on any machine. Ignition uses your computer's web browser to launch rich Java applications – not thin web pages – which are fast and cross-platform, making accessibility easy for anyone on your network.

Looking for Less?

Some systems only require a handful of clients at any given time. In cases like these the **limited version** of Vision is a perfect fit: all the functionality of the full version but at a greatly reduced cost! Contact our Sales department for more details.

Want to Learn More?

For more information on the Vision module, please see the Vision section of this user manual.

Perspective Module

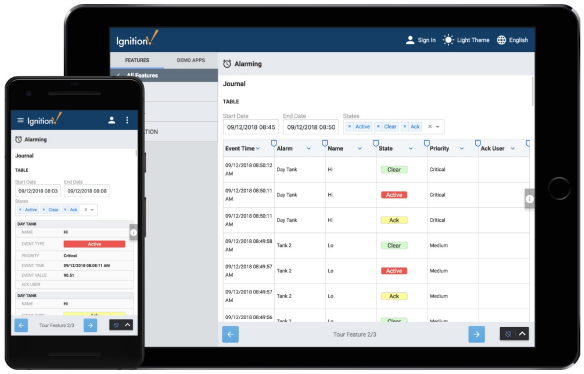
On this page

...

- Mobile Responsive Design
  - Device Compatibility
  - Run as a Native App
  - Responsive Design
  - Designed for Touch
  - CSS and Themes
  - Security via SSL

Mobile Responsive Design

The Perspective Module raises the bar for visualizing your control applications in Ignition. Not only does it leverage the popular standard HTML5/CSS3 technology for its UI layer, but it also allows users to see and control their application on any device, from a small smartphone to an oversized desktop, in an optimized format. HTML5/CSS3 allows screens to display gracefully on any device, regardless of size or scale. Multi-touch responsiveness allows for greater flexibility of use than ever before. The Perspective Module marks the beginning of truly mobile-optimized, touch-responsive, easily accessible applications for monitoring, control, analysis and data gathering in industrial systems.



Online Demo Home Page

The Demo Project uses the Perspective Module to show off many of the key features of Ignition. Click the link below to begin.

Go to the Demo Page

What is Ignition?

This is a short video that goes over the basics of Ignition and how it works.

Watch the Video

Device Compatibility

The Perspective Module creates applications that are compatible with any device, including all mobile touchscreen devices like smartphones and tablets, and any desktop running any current web browser, on any major operating system: Windows, macOS, and Linux, iOS, and Android.

Run as a Native App

Run your Perspective Sessions in a web browser or in a native App on your device. Apps can be downloaded on Android or iOS devices and allow you to natively run a Perspective session on your phone or tablet. This gives you the ability to use more of your device, like using the camera to scan barcodes or the built-in GPS to verify a location.

Responsive Design

Applications designed with the Perspective Module respond to changes in screen size and orientation, and scale elegantly when launched on any size mobile multi-touch device. Perspective provides several container types for maximum design flexibility, each of which employs a different layout technique. Designers can compose these container types together, providing a very flexible system that can handle any kind of responsive design.

Designed for Touch

Perspective employs multi-touch technology, making it possible to work with TouchPad and touch-screen interfaces found on smartphones, tablets, laptops, and desktops. This allows users to work with familiar tools for panning, zooming, and scrolling.

CSS and Themes

The Perspective module takes advantage of its roots in web technology to give you easy to configure themes and styles using familiar CSS settings. Configure a few themes, use them throughout your projects, and swap between them however you'd like. It's easy to let your users change text and colors whenever they like.

Security via SSL

Clients launched from the Perspective Module are secured using cutting-edge encrypting technologies and communications protocols to provide the most secure web-based data transfer. It also has strongly enforced “guest mode” access to prevent against unauthorized writes on the Gateway.

OPC UA Module

On this page

...

- Cross-Platform OPC UA Functionality
  - OPC UA Server Functionality
  - Built-in Device Drivers
  - Redundancy Support
  - Public Driver API
  - Driver Modules
- Ready to Get Started?

Cross-Platform OPC UA Functionality

The Ignition OPC UA module offers OPC UA server functionality with a variety of device drivers and a robust, open driver API.

OPC UA Server Functionality

This module turns Ignition into an OPC UA server, capable of handling connections from any OPC UA compatible client. Outgoing connections can also be made to 3rd-party OPC UA servers. In this way Ignition functions as both an OPC UA server and an OPC UA client.

Built-in Device Drivers

The OPC UA module offers a number of device drivers for common protocols out of the box, and is easily expandable thanks to the hot-swappable module architecture in Ignition. New drivers can be downloaded and installed in minutes without requiring a system restart or otherwise affecting other parts of the Ignition platform.

Redundancy Support

The OPC UA module ties into the Ignition redundancy in order to provide efficient access to device data along with failover redundancy, with no additional configuration.

Public Driver API

Anyone can create new drivers thanks to the open driver API, and users can download and install drivers created by other developers. This is the first time such an API has been made publicly available for a product like Ignition. For more information about creating drivers for Ignition, visit the Inductive Automation website.

Driver Modules

Drivers for the OPC UA module are deployed as modules themselves. While they don't add a visible element to the system, they are loaded and upgraded in the same manner as other Ignition modules. A list of available drivers can be found on the Driver Modules page.

Ready to Get Started?

Documentation on the OPC UA module can be found in the OPC UA and Device Connections page.

In This Section ...

Driver Modules

Available Drivers

The following drivers are available for Ignition's OPC UA module.

On this page

...

- Available Drivers
- Allen-Bradley Ethernet
- Modbus
- Siemens
- UDP and TCP
- DNP3
- Omron

Allen-Bradley Ethernet

The Allen-Bradley brand of programmable logic controllers (PLC) is a popular choice in many industries. Ignition can connect to the following Allen-Bradley controllers:

PLC	Capability
Logix Driver	Connect to Allen-Bradley Logix family devices, including devices with firmware v21+.
ControlLogix	Connect to ControlLogix firmware v20 and prior processors.
CompactLogix	Connect to CompactLogix firmware v20 and prior processors.
MicroLogix	Connect to MicroLogix 1100, 1200, 1400 and 1500 series PLCs.
PLC5	Connect to PLC5s via Ethernet.
SLC	Connect to SLC 5/05s via Ethernet.

See the following links for detailed step-by-step instructions on how to connect to these PLCs:

- Connecting to CompactLogix
- Connecting to ControlLogix
- Connecting to Logix
- Connecting to MicroLogix
- Connecting to PLC5
- Connecting to SLC
- Allen-Bradley Connection Paths

Modbus

The Modbus Driver module allows the Ignition OPC UA server to communicate with any device that supports the Modbus protocol, that is, the Modbus TCP and Modbus RTU over TCP protocols.

The Modbus Driver module can connect directly to devices that support Ethernet communications or it can also connect to Modbus devices through a Gateway device.

See the following sections for detailed step-by-step instructions:

- Connecting to Modbus Device
- Modbus Addressing
- Modbus Address Mapping

Siemens

The Siemens Drivers module provides support for connecting to the following PLCs via TCP/IP using the S7 protocol:

- S7-300

- S7-400
- S7-1200

For step-by-step instructions on how to connect to Siemens Drivers, see Siemens.

UDP and TCP

The UDP driver is a strictly passive listener. The UDP driver is configured to *listen* to one or more ports on a given IP address.

The TCP driver is configured to *connect* to one or more ports on a given IP address. This driver can listen, as well as write back to the connected port.

Rules are configured that dictate how the incoming data is interpreted.

For step-by-step instructions on how to connect to TCP Drivers, see UDP and TCP Driver.

DNP3

The DNP3 module allows the Ignition OPC UA server to communicate with other devices that support the DNP3 protocol (DNP3 outstations).

See the following sections for detailed step-by-step instructions, see DNP3

Omron

The Omron Driver module allows the Ignition OPC UA server to communicate with NJ-series controllers.

More information on the Omron driver can be found on the Omron NJ Driver page.

Tag Historian Module

The Tag Historian Module for Ignition allows you to turn a SQL database into a high-performance time-series Tag historian. It requires minimal work to implement and is simple to use, even if you aren't very experienced with SQL databases. The power, ease of use, and incomparable price-to-performance ratio of the Tag Historian Module make it the clear choice for storing your organization's valuable historical information.

On this page ...

- Store History with a Click
- Create Tables Effortlessly
- Easy Query Binding
- Powerful Historian Engine
- Make Data More Accessible
- Real-Time Charts and Trends
- Great Performance and Value
- Powered by Ignition
- Learn More

Store History with a Click

With the Tag Historian Module, you can store history on an Ignition Tag to your database with the click of a mouse - no complex configuration, tuning, or data modeling required.

Create Tables Effortlessly

The module creates and manages tables for you, so you don't need to be knowledgeable about database management to use it.

Easy Query Binding

Use the data intuitively throughout Ignition with simple, visual query binding screens – no need to write SQL or complex queries.

Powerful Historian Engine

The module's high-powered historian engine automatically supports compression, partitioning, interpolation, and aggregation for any SQL database.

Make Data More Accessible

By recording your data to a SQL database, the module makes your process data available to any application with database access.

Real-Time Charts and Trends

Combine with the Vision Module to simply pull the information you need into individual charts and graphs.

Great Performance and Value

By leveraging the power of SQL databases, the Tag Historian Module delivers tremendous performance at a fraction of the price of traditional historian products.

Powered by Ignition

The Tag Historian Module is a SCADA module for Ignition, the powerful HMI / SCADA software from Inductive Automation. It integrates seamlessly with the Ignition platform and with all other modules in the Ignition SCADA suite. Because the module is built on the power of Ignition, it shares the same advantages, such as cross-platform compatibility, unlimited free clients, robust out-of-the-box SQL database support, and fast installation. Its ability to leverage the full power of Ignition places it in a class above any other Tag historian on the SCADA software market.

Learn More

Read more about Tag Historian in the Tag Historian section of the manual.

SQL Bridge Module

High-Powered Data Acquisition

Share Plant Floor Data with Any Database

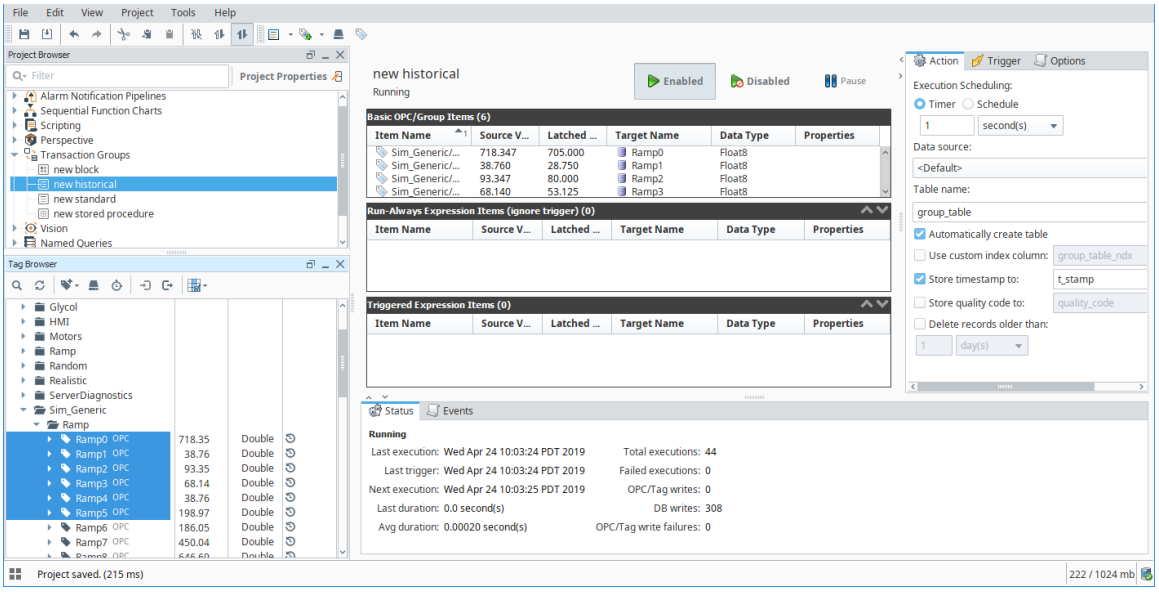
The Ignition SQL Bridge Module brings the power of structured query language (SQL) databases to your industrial process. The module allows you to leverage Ignition's unlimited Tags and database connections at an astonishingly low price — even if you don't know much about SQL databases.

Think of the SQL Bridge Module as a "Swiss Army Knife" for integrating programmable logic controllers (PLCs) and SQL databases in unlimited ways. Move data bidirectionally, log huge amounts of data easily, synchronize PLCs with databases, create powerful recipe and batching systems, and sequence entire processes – your imagination is the limit.

On this page

...

- High-Powered Data Acquisition
  - Share Plant Floor Data with Any Database
  - Move Data Bidirectionally
  - Log Data Easily
  - Synchronize PLCs with Databases
  - Build Recipe & Batching Systems
  - Sequence Your Processes
  - Flexible and Easy to Use
  - Powered by Ignition
- Looking for No-Frills Data Logging?
- Want to Learn More?



Move Data Bidirectionally

The module uses transaction groups that function as a bidirectional data bridge between an OPC server and a SQL database. Transaction groups can perform many data-centric tasks, such as event logging, vertical and horizontal data storage, inserting rows into a table, and storing data to a specific row.

Log Data Easily

The easiest, most powerful data logger on the market includes triggers, handshakes, expressions, and other items that help you store data in the exact way you need. Triggers offer endless data-storage possibilities such as storing production summaries, logging the sequence of event information, and tracking downtime events. Simply map your process data to any table structure and let the module do the work of creating and managing new tables or writing to existing tables.

Synchronize PLCs with Databases

Extend your PLC address space into your SQL database with bidirectional mirroring, which immediately reflects any value change in the PLC or database end in the opposite end. Bidirectional mirroring enables real-time control via common web technologies and other database-compatible front-ends. Set the update rate for rows/fields (down to sub-second rates), and create two-dimensional arrays by adding rows. The SQL Bridge Module is the first and only software in the industry to offer this functionality.

Build Recipe & Batching Systems

By extending your process into the database, the full bidirectional access of the SQL Bridge Module enables you to create a robust, customized recipe and batching system. Use the database to store a virtually unlimited number of recipes, and modify recipes however and wherever you want, while tracking all changes historically. Set up reliable database-to-PLC transactions using triggers and handshakes. Leverage stored procedures to perform complex tasks or replace aging transaction systems.

Sequence Your Processes

Flexible triggers and database tables enable you to easily sequence entire processes. Use dynamic references to step the index to a row of values that are then written to the PLC. Create complex triggers, which reference multiple process variables and expressions, by using a simple expression language. Use rows/fields as timers and counters by leveraging the unique built-in hour meter and event meter functionalities.

Flexible and Easy to Use

Across the board, SQL Bridge is the clear winner for tying industrial systems into SQL databases. Online editing lets you make changes on the fly, without stopping your process or affecting other parts of Ignition. You can stop locking data away in proprietary boxes because all data is open and accessible in the database. Use flexible data modeling without a lot of work: the module automatically creates tables and columns in any format. The built-in table browser allows you to work entirely within the Ignition Designer, so there is no need for external programs.

Powered by Ignition

The SQL Bridge Module is a SCADA module for Ignition, the powerful HMI, SCADA, MES and MQTT software from Inductive Automation. It integrates seamlessly with Ignition and all other modules in the Ignition SCADA suite. Because the module is built on the power of Ignition, it shares the same advantages, such as cross-platform compatibility, unlimited free clients, robust out-of-the-box SQL database support, and fast installation. Leveraging the full power of Ignition, it is a data management tool unlike any other in the SCADA software market.

Looking for No-Frills Data Logging?

SQL Bridge Limited provides an affordable way to create Historical Groups, and record history into any SQL database. Speak to our Sales department for more details.

Want to Learn More?

For more information on the SQL Bridge Module, please see the SQL Bridge (Transaction Groups) section of the user manual.

Reporting Module

On this page

...

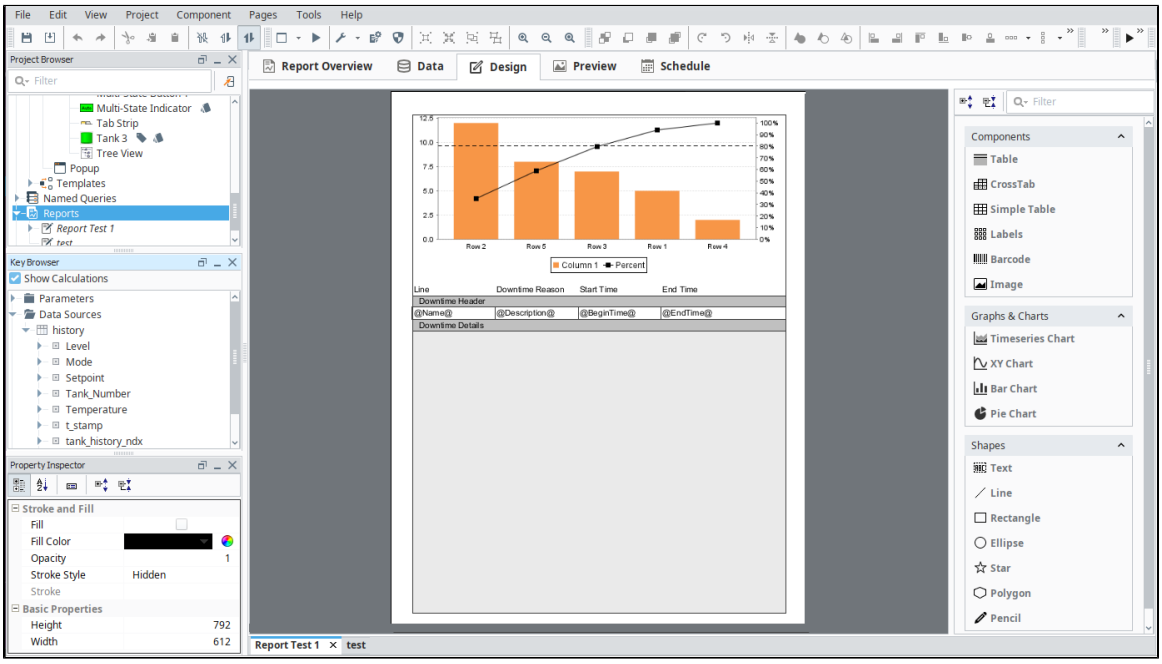
- Dynamic Report Generation
  - Quickly and Easily Create Rich, Dynamic PDF Reports
  - Select Data from Multiple Sources
  - Design Reports with an Easy-to-Use Interface
  - Schedule Reports to Run Automatically
  - Print, Send, or Save Automatically
  - Powerful Query Tools
  - Powered by Ignition
- Unlimited Options, Limited Price
- Want to Learn More?

Dynamic Report Generation

Quickly and Easily Create Rich, Dynamic PDF Reports

Now it's easier than ever to make and distribute dynamic, professional reports in Ignition. The Ignition Reporting Module is a standalone reporting solution that simplifies and enhances the reporting process from beginning to end. Generate reports from existing files or create them from scratch. Pull data from a variety of sources, and design reports with a simple to use interface. Add charts, graphs, tables, crosstabs, shapes, and other components for visual impact. Save your reports in PDF, HTML, CSV, and RTF file formats, and set up automatic scheduling and delivery.

Use all of these powerful capabilities to quickly create many common types of reports such as production management, efficiency monitoring, downtime tracking, SPC, QA, OEE management, and historical data analysis.



Select Data from Multiple Sources

The Reporting Module makes it easy to define the data behind your reports. Specify any number of report parameters and data sources in your reports.
Access information from any SQL database. Also, you can also easily query data from Ignition's Tag historian and event data from the alarm journal.

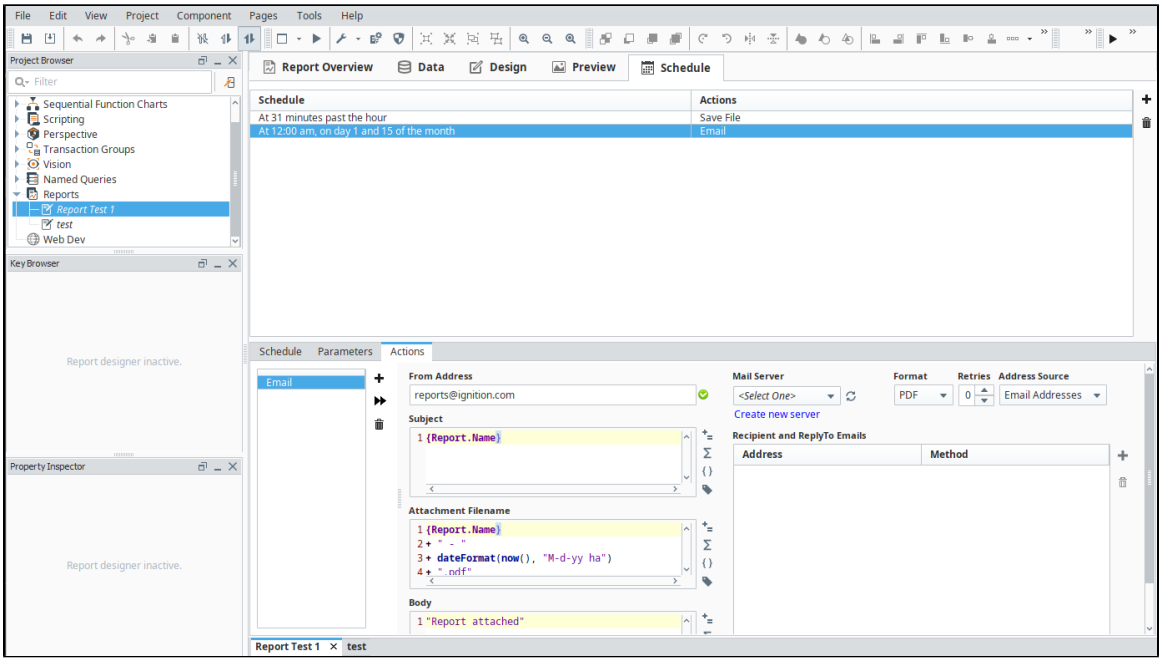
Design Reports with an Easy-to-Use Interface

Design your reports easily inside Ignition's Designer application. The interface is far more user friendly than those of other reporting software, because it enables you to design reports with the same intuitive feel and rapid application development tools that you get when designing applications in Ignition. This allows you to design your reports using the drag-and-drop method.

Schedule Reports to Run Automatically

The Reporting Module allows you to schedule reports within the Ignition Gateway. You can schedule a report to run on a regular basis; for example, on every Monday at 5:00 p.m. Then, your report will run as scheduled whether or not any Ignition clients are open.

You also have the option to set up triggered execution, which means that reports run on-demand when triggered by an event such as a Tag change, a shift change, the finish of a production run, etc.

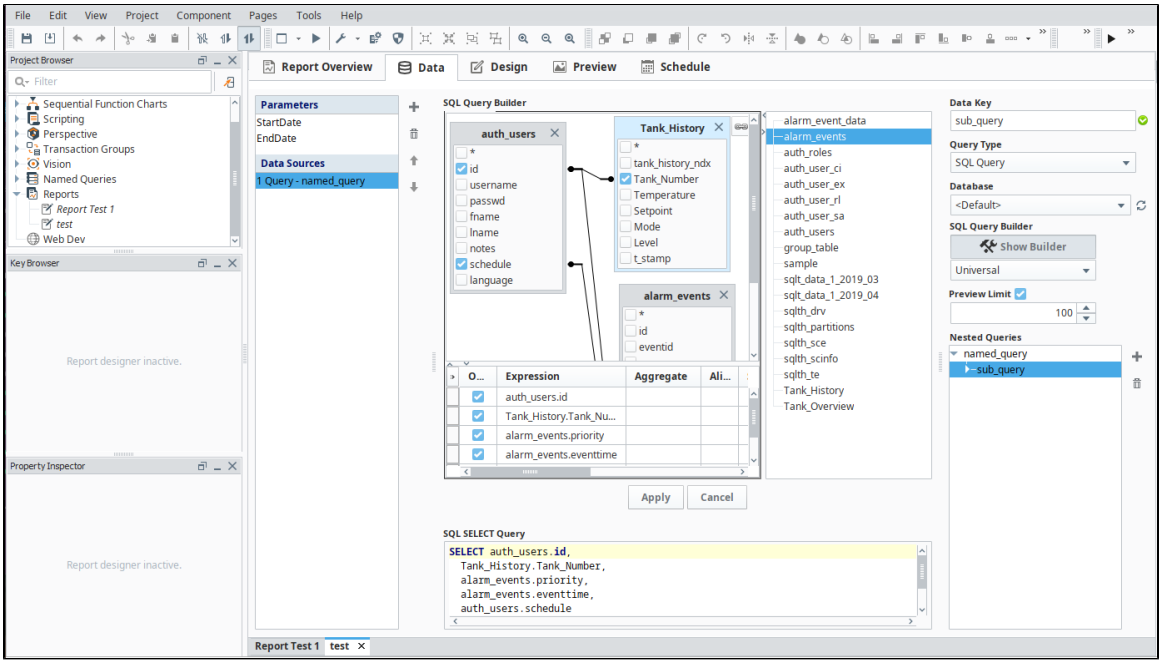


Print, Send, or Save Automatically

Scheduled reports can be automatically delivered in many ways: print, email, save to a file server, upload to an FTP server, or use scripting to customize the method of delivery. Have your reports distributed in the way you want, every time, automatically.

Powerful Query Tools

The Reporting Module has a powerful drag-and-drop query interface that allows you to build complex SQL queries easily. The new query structure allows you to create complex nesting which can correlate relational data and process historian data in a seamless fashion.



Powered by Ignition

The Reporting Module is a SCADA module for Ignition, the powerful software from Inductive Automation. The Reporting Module integrates seamlessly with Ignition and with all other Ignition modules. Because the module is built upon the power of Ignition, it shares the same advantages, such as cross-platform compatibility, unlimited free clients, robust SQL-database support, and fast installation. Leveraging the full power of Ignition, the Reporting Module is unmatched by any other SCADA reporting product on the market.

Unlimited Options, Limited Price

The limited version of the Report Module provides an affordable way to create and schedule just a few reports. Speak to our sales department for more details.

Want to Learn More?

For more documentation on the Reporting module, please see the Reporting section.

Alarm Notification Module

On this page

...

- Alarm Notification
  - Manage SCADA Alarm Notifications Any Way You Choose
  - Control Your Alarm Notification Logic
  - Manage Users and Schedules to Ensure Timely Delivery
  - 2-Way Email Notification
- Get Started

Alarm Notification

Manage SCADA Alarm Notifications Any Way You Choose

The Ignition Alarm Notification Module enables you to configure the logic for how, why and when alarm notifications are delivered, manage alarm notification for groups of users, and send notifications via email. Notifications can be easily be acknowledged by the recipient via email.

Control Your Alarm Notification Logic

Ignition's alarm notification pipeline feature is an innovative tool that lets you easily chart out the routing of your alarms. Make your alarm notification logic as simple or as multifaceted as you want. Prevent alarm flooding by directing the right alarms to the right people at the right time.

With Ignition and the Alarm Notification Module, you'll have greater control of what happens between when an alarm or alert goes active and when it is sent out to personnel, and of who gets alarm notifications and when.

By using a simple drag-and-drop interface, you can build various types of alarm notification logic, including:

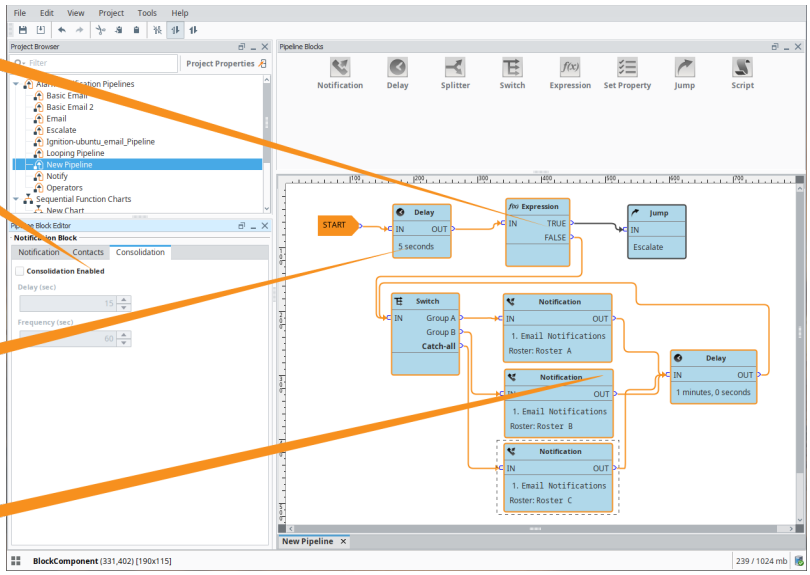
- **Delay**: Hold off on sending notifications until nearby operators are given time to deal with the problem first.
- **Escalation**: Elevate alarm notification when operators are not handling alarms.
- **Consolidation**: Combine multiple alarms into a digested message.
- **Selection**: Distribute different alarms to different groups of contacts.

Escalation:
Send higher priority alarms to a different contact or group.

Consolidation:
Combine multiple alarms into a digested message.

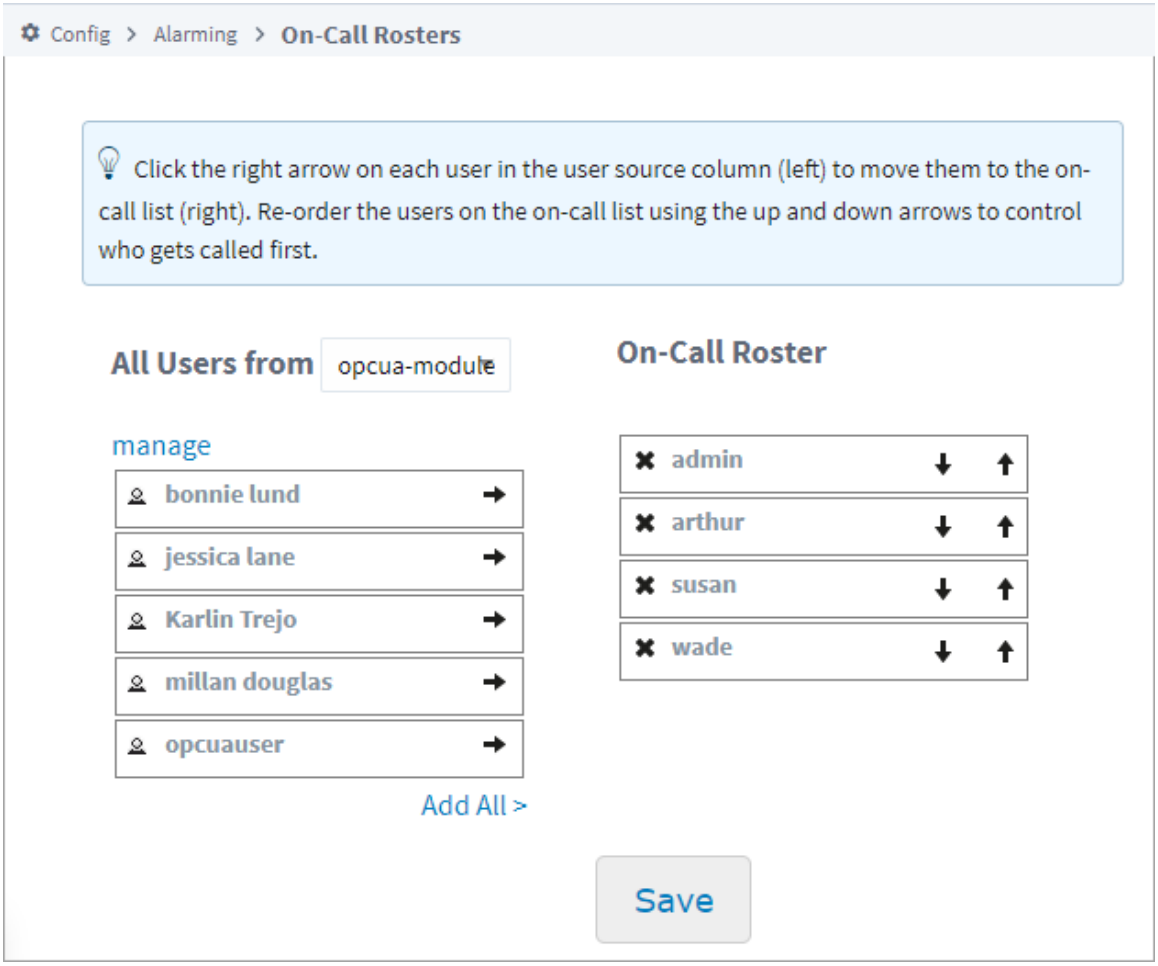
Delay:
Hold off on sending notifications until nearby operators are given time to deal with the problem.

Selection:
Distribute different alarms to different groups of contacts.



Manage Users and Schedules to Ensure Timely Delivery

Make sure the right people – and especially those on shift – get notified quickly. The Alarm Notification Module lets you organize individuals into groups based on their roles with the On-Call Roster feature.



When an alarm notification needs to be sent, it goes to a designated call roster and then the system evaluates individual schedules so that only those currently on shift will be notified. You can easily change schedules, assign roles and specify who should be notified when alarms pass through different notification blocks in the pipelines.

The Alarm Notification Module lets you edit rosters in the client in real time and to edit individual schedules right in the authentication profile. With permission roles, you can let individuals edit their own information on-the-fly to account for irregular time away from work so that alarms will skip over them and be sent quickly to an available person.

2-Way Email Notification

With the Alarm Notification Module, users in your organization can receive and acknowledge alarm notifications by email.

Get Started

Learn more about alarm notifications in the Alarm Notification section.

SMS Notification Module

On this page

...

- SMS Alarm Notification
  - Add SMS Functionality to Your Alarming System
  - Notify Users Quickly with a Text Message
- Get Started

SMS Alarm Notification

Add SMS Functionality to Your Alarming System

Quickly let key people know about any alarm by sending a text message to any cell phone. The Ignition SMS Alarm Notification Module enables both alarm notification and acknowledgment via text messages. You can enable phone-call alerts by adding the Voice Notification Module to your Ignition system.

Notify Users Quickly with a Text Message

By adding the SMS Notification Module to Ignition, you can send and acknowledge alarm notifications via text messaging. SMS notification is an efficient and immediate method that makes it easier for users to respond to alarms no matter where they are.

This module requires the Alarm Notification Module and a cellular modem.

Get Started

Read more about this module on the SMS Notification Profile page.

WebDev Module

Use Web Services in Ignition

On this page

...

- Use Web Services in Ignition
  - Host a Web Page on an Ignition Gateway
  - Configure Custom Web Service Calls
  - Barcode Scanners and Web Enabled Devices

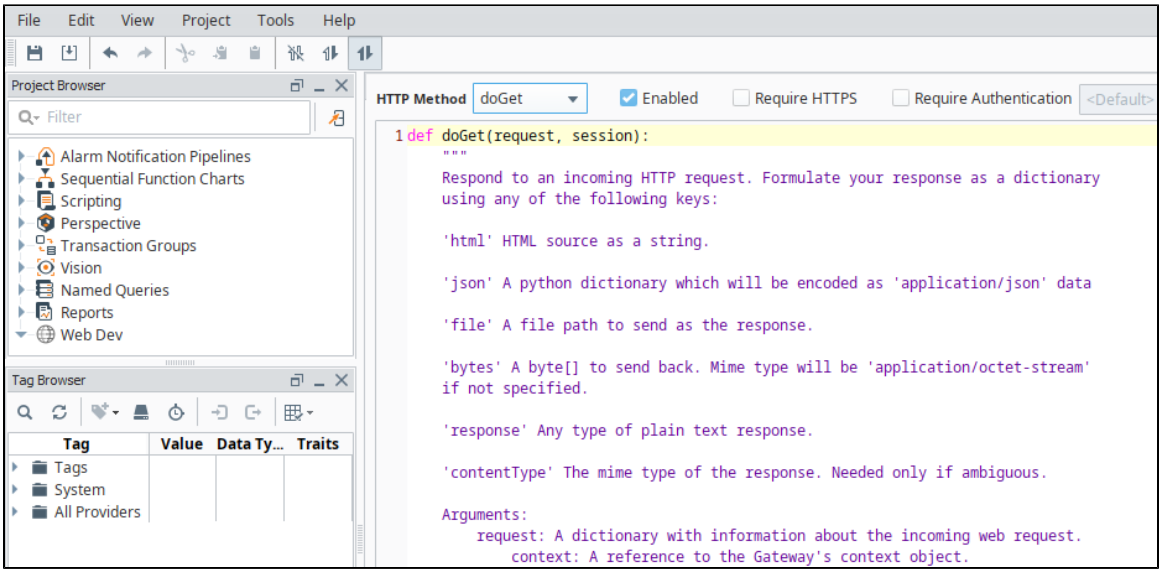
Host a Web Page on an Ignition Gateway

This module enables you to directly program against the web server inside the Ignition Gateway. Webpages can be built by hand using a combination of Python programming and static web resources such as images, CSS files, javascript files, and html files. You can design and host custom web pages directly against the Ignition service.

Configure Custom Web Service Calls

External systems can interact with the Ignition service through web service calls using the WebDev module. Allow SAP and ERP systems to request data directly from Ignition by developing a custom, full-fledged, API.

More Information on the WebDev module can be found on the WebDev page.



Barcode Scanners and Web Enabled Devices

Many newer barcode scanners are web-enabled and don't use serial ports to connect. For these types of scanners you can use the WebDev Module to write whatever connect functions that you need. This opens up a whole new world of possibilities for connecting to various mobile devices!

Enterprise Administration Module

Enterprise Administration

Enhance the reach of Ignition with the Enterprise Administration Module (EAM)

The EAM provides a secure and intuitive way to manage many Ignition installations from one location. It is ideal for large enterprises that deploy multiple Gateways across vast geographical distances, but even companies with a few Ignition Gateways on a single plant floor can benefit from its ability to synchronize projects, monitor performance, and automate backup and recovery from a central location.

On this page ...

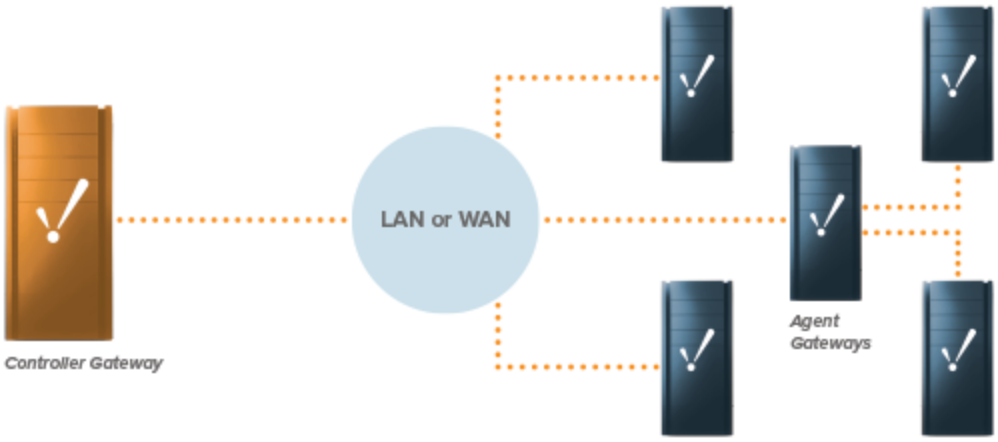
- Enterprise Administration
  - Enhance the reach of Ignition with the Enterprise Administration Module (EAM)
  - Manage Multiple Gateways from One Gateway
  - Synchronize Projects and Resources
  - Communicate Over a Secure Network
  - Monitor Agent Health and Performance
  - Automate Gateway Backup and Recovery
  - Powered by Ignition
- Get Started

Manage Multiple Gateways from One Gateway

With the EAM Module installed, an Ignition Gateway can function as the central Controller Gateway that connects with, monitors, and manages any number of remote Agent Gateways. Use the Controller Gateway to coordinate and automate many administrative tasks for Agent Gateways, including:

- Assigning, updating, activating, and unactivating licenses
- Deploying new and updated modules
- Disaster Recovery
- Remote Backups

All of these actions can be executed immediately or scheduled for later.



Synchronize Projects and Resources

The EAM gives you to the ability to design resources and projects centrally in the Controller Gateway and send them out to the Agent Gateways.

Communicate Over a Secure Network

The EAM is built on a high-performance, high-security communication network that facilitates simultaneous access to multiple Gateways. Connect multiple Ignition Gateways using the Gateway Network and all copies of Ignition will be able to communicate freely.

Monitor Agent Health and Performance

In the Controller Gateway, the EAM provides health indicators for Agent Gateways and alarms for impending lifecycle events such as Agent Gateway failure. Memory usage notifications, warnings, and error events are also provided.

Automate Gateway Backup and Recovery

The EAM enables automated backup and quick recovery for all Gateways through a Gateway backup archive.

Powered by Ignition

The EAM is a SCADA module for Ignition, the powerful HMI / SCADA software from Inductive Automation. The EAM integrates seamlessly with Ignition and all other modules in the Ignition SCADA suite. Because the module is built on the power of Ignition, it shares the same advantages, such as cross-platform compatibility, unlimited free clients, robust out-of-the-box SQL database support, and fast installation. Leveraging the full power of Ignition, it is an enterprise management tool unlike any other in the SCADA software market.

Get Started

Read more about EAM in the Enterprise Administration section.

SFC Module

On this page

...

- Sequential Function Charts
  - Streamline the Creation, Organization, and Visualization of Robust Logic Systems
  - Visual Logic Programming
  - Chart Monitoring
  - Build Robust Logic
  - Multiple Chart Instances
  - Long-Running Charts
  - Powered by Ignition
- Learn More

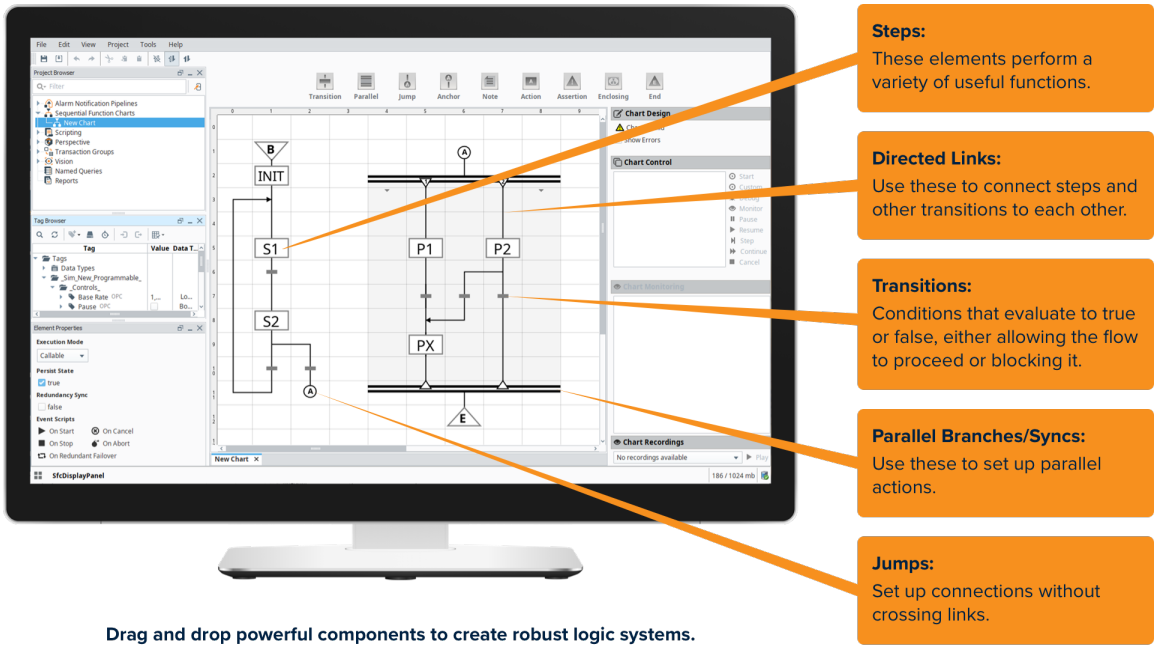
Sequential Function Charts

Streamline the Creation, Organization, and Visualization of Robust Logic Systems

The Ignition SFC Module's powerful visual programming language allows you to write complex logic in a simpler, more intuitive way. The SFC Module works seamlessly with Ignition, and represents a leap forward in the capabilities of the software.

Visual Logic Programming

The SFC Module's visual, drag-and-drop programming tool makes the process of writing complex logic much smoother than using the basic Python scripts included in Ignition. Its standardized programming method, part of the IEC 61131-3 specification, will be familiar to many users who have experience in programming PLCs. With a simple visual interface, you can use steps, transitions, parallel blocks and other drag-and-drop elements to create complex charts. SFCs within Ignition can make use of scripts, Tags, and other elements already available in the Ignition platform.



Drag and drop powerful components to create robust logic systems.

Chart Monitoring

You can monitor charts visually in the Ignition designer and clients, watching each chart execute its logic, down to which specific step is currently running. You can even inspect the chart's internal variables as it runs. The visual nature of SFCs make them much easier to troubleshoot than Python scripts.

Build Robust Logic

Now it takes far less time to build any type of robust logic your enterprise may need: long-running tasks, parallel tasks, tasks that pause for certain conditions before proceeding, loops, and sub-charts.

Multiple Chart Instances

The module allows you to instantiate multiple instances of any chart. Each instance can be configured with different starting parameters, allowing the instances to be independent from each other.

Long-Running Charts

Charts may be designed to run for a very long time. They can even keep running if a Gateway is restarted by pausing on shutdown and resuming on startup. SFCs also support redundant Ignition setups to ensure the chart keeps running even if the primary Gateway fails.

Powered by Ignition

The SFC Module is a SCADA module for Ignition, the powerful, award-winning, HMI / SCADA software from Inductive Automation. The SFC Module integrates seamlessly with Ignition and with all other modules in the Ignition SCADA suite. Because the module is built upon the power of Ignition, it shares the same advantages, such as cross-platform compatibility, unlimited free clients, robust out-of-the-box SQL database support, and fast installation. Leveraging the full power of Ignition, it is a SCADA logic engine unlike any other on the market.

Learn More

Documentation on the SFC module is available in the Sequential Function Charts section.

Additional Modules

Enhance Ignition's capabilities beyond traditional SCADA and HMI limits. These additional modules provide Ignition with even more functionality. Take your project to the next level with these powerful modules.

Voice Notification

Enable both alarm notification and acknowledgment via a phone call.

OPC COM

Integrate Ignition with Classic COM-based OPC Servers.

OPC COM Tunneller

Exposes OPC servers accessed through the OPC-COM module to Ignition's OPC-UA server. Allowing you to easily expose tags from a DA server to any UA server.

Web Browser

View web-pages directly from Ignition clients.

Serial

Adds platform-level serial comm support and serial-specific scripting functions.

Twilio Alarm Notification

Adds integration with Twilio via an SMS Alarm Notification Profile and scripting functions to send SMS.

SECS/GEM

Communicates with semiconductor fab equipment.

In This Section ...

Voice Notification Module

On this page

...

- Text to Speech Alarm Notification
  - Send Notifications Via a Phone Call
  - Alarm Notification Module Compatibility
  - Two-Way Acknowledgement
  - Custom Notification Messages
  - Audit Log Integration
  - Multi-Language Support

Text to Speech Alarm Notification

Send Notifications Via a Phone Call

Ensure no serious issue is overlooked by speeding alarm notification delivery to the right people as soon as possible. The Ignition Voice Notification Module uses state of the art text-to-speech technology to add phone-call notification to your alarming / alerting process.

Set up voice notification for alarms in Ignition with the Voice Notification Module. Use text-to-speech (TTS) technology to notify users of an alarm via a phone call and to allow users to acknowledge it with a simple key press. The VoIP-over-SIP voice notification system can be tied in with a VoIP phone system or online VoIP service, or bridged to a standard phone line using an existing phone gateway.

The Voice Notification module adds the ability to deliver alarm notifications to users via telephone, using any SIP compatible phone system. Messages are constructed in text, and are delivered through a high quality text-to-speech engine.

Alarm Notification Module Compatibility

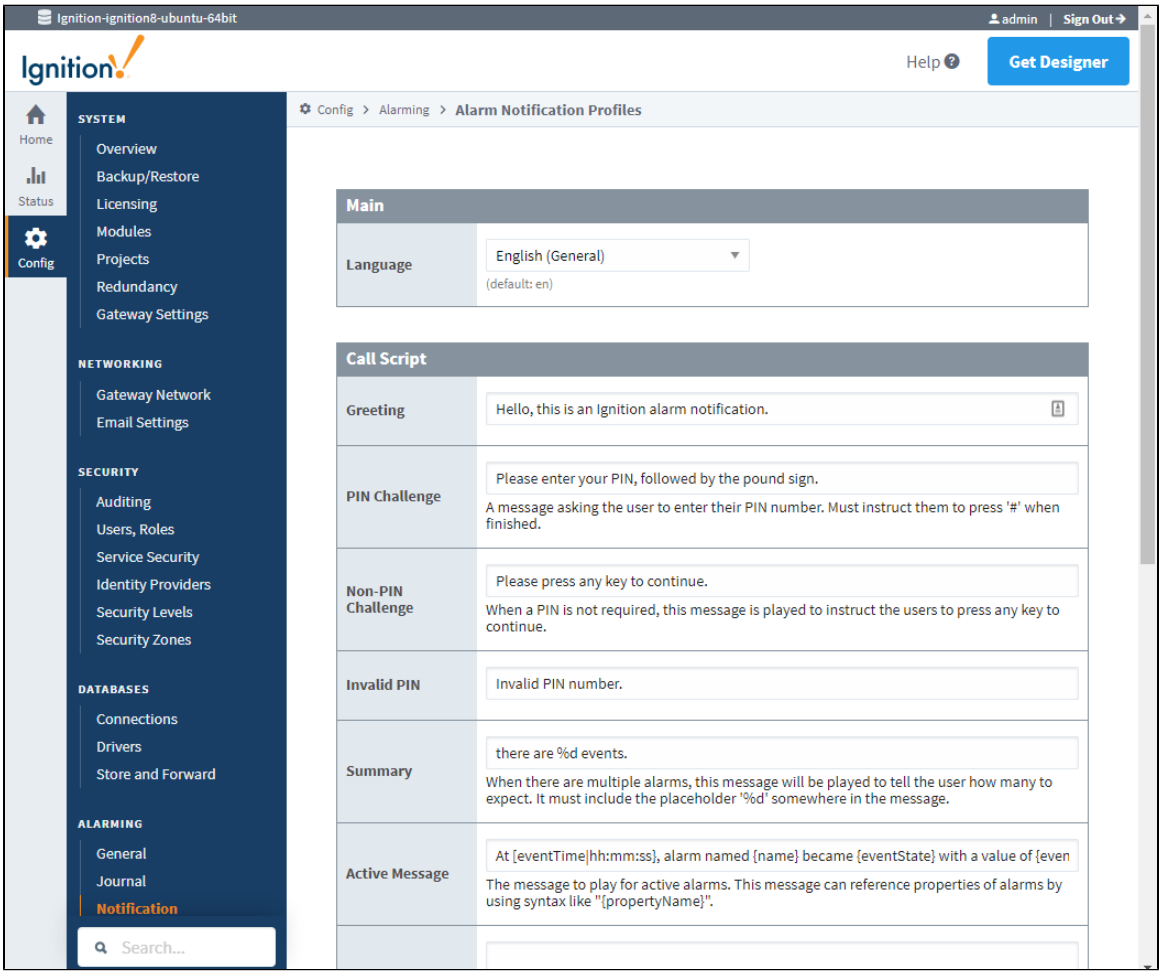
The Voice Notification module is fully compatible with the Alarm Notification module, so all pre-existing on-call rosters and pipelines will work with Voice notification.

Two-Way Acknowledgement

The Voice Notification Module can be configured to allow users to quickly acknowledge alarms. With the Two-Way property enabled, the notification is sent out a response code in the message. The user simply replies with the response code and the alarm will be acknowledged.

Custom Notification Messages

The High-Quality Text-to-Speech engine allows you to customize the alarm message the user hears, so you are not limited to a number of pre-recorded messages.



Audit Log Integration

The Voice Notification module can be configured to log successful message delivery and user acknowledgement into any connected database. Once recorded, a simply query will return all notification activity.

Multi-Language Support

Write your messages in any of the provided languages. A list of the available languages can be found on the Voice Languages page.

> This module requires the Alarm Notification module and some SIP-compatible VoIP service, such as your office VoIP server, Skype Connect or a IP Telephony Gateway appliance to bridge to a standard phone line (not included).
>
> Examples of a IP Telephony Gateway appliance are: Atcom IP04 (the number indicates the phone lines, Ignition can only use 1 line at a time), Grandstream HT-503, GXW410x (It is the FXO port that connects to the phone line).

OPC COM Module

On this page

...

- Connect to Legacy Devices
  - Connecting to Classic OPC Servers
  - Using Data from Classic OPC Servers

Connect to Legacy Devices

The OPC-COM module offers Ignition the ability to connect to legacy ("classic") COM-based OPC-DA servers. It supports OPC-DA 2.0 and 3.0.

> **Tip**
>
> The OPC Com Tunneller Module simplifies connecting to remote OPC-COM servers.

Connecting to Classic OPC Servers

With the OPC-COM module installed, there will be a new option for COM-based OPC servers when creating a server connection in the Gateway. The OPC-COM module is primarily intended for use with local OPC servers, although it also provides basic support for remote connections.

Even when connecting locally, the application may run into the traditional difficulties of connecting to windows based OPC servers. DCOM security settings on the machine can interfere with connections, and the OPC Core Components package (from the OPC Foundation) must be properly installed before connections can be established.

Using Data from Classic OPC Servers

After a connection to a server has been defined, the server will appear along side of other OPC ser vers (both COM and UA based) in the OPC Tag Browser. You can use these tags like any other ones - bring them into Ignition's Tag Browser, use them in Transaction Groups, and so on.

OPC COM Tunneller Module

On this page

...

- Create Remote Connections through OPC-UA
  - Simplify COM Connections
  - Modernize COM Servers

Create Remote Connections through OPC-UA

Simplify COM Connections

The OPC COM Tunneller module eliminates many difficulties associated with establishing remote OPC-COM connections. The OPC COM Tunneller module exposes OPC COM connections to Ignition's OPC UA server, allowing you to easily access them over the network and eliminate the need to use a Remote DCOM connection.

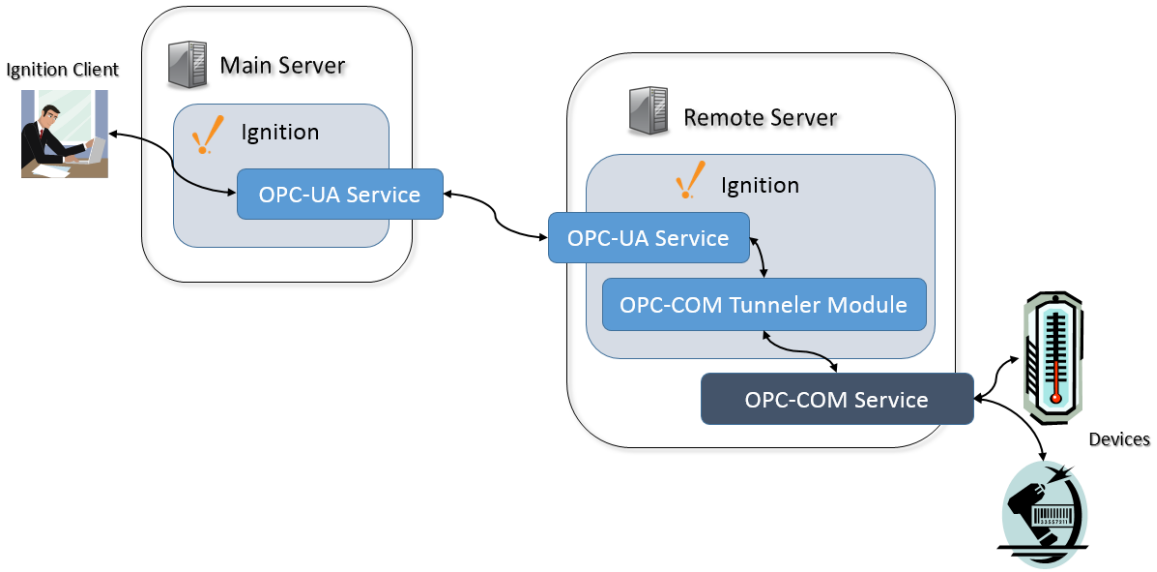
Modernize COM Servers

Ignition manages the OPC COM connection in the context of the OPC UA standard resulting in easier access over the network, eliminating the need to establish remote DCOM connections, and incorporating the security benefits of the TCP standard such as authentication and encryption.

OPC Com Tunneller

Learn how to configure the OPC Com Tunneller module.

Link to Knowledge Base Article



Web Browser Module

On this page

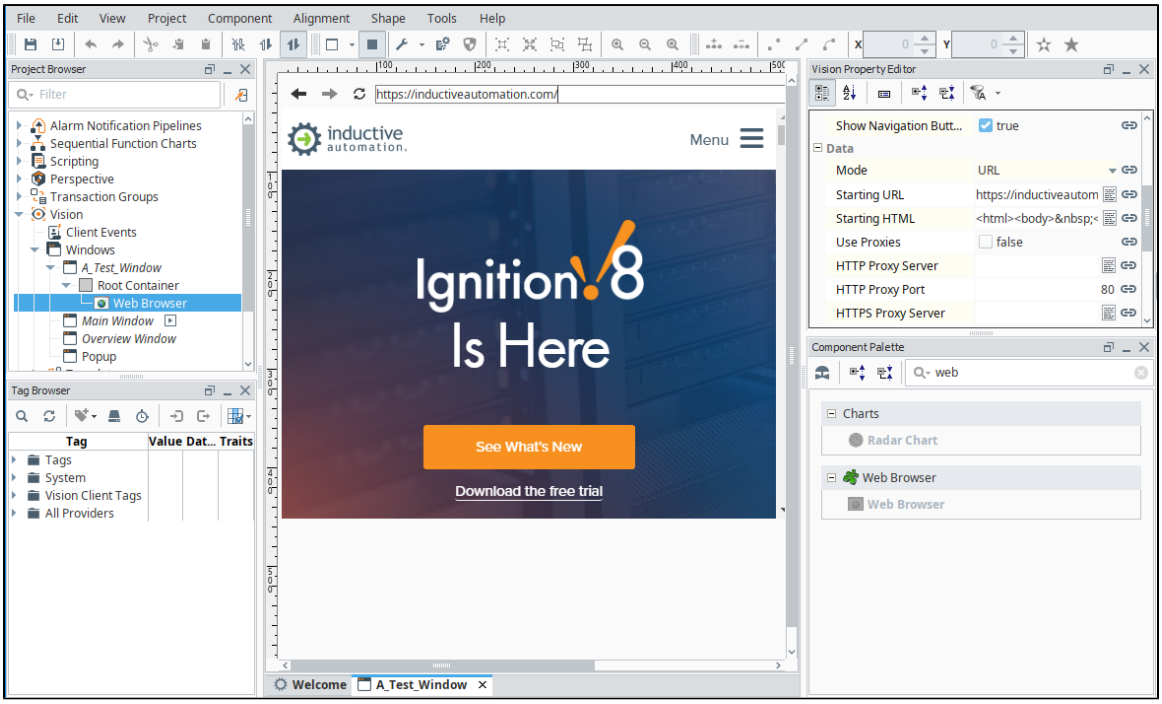
...

- Access the Web Directly From a Client

Access the Web Directly From a Client

The Web Browser module adds a web-browser component to the Vision module. This component allows you to embed other webpages from the internet or from your local intranet inside of your projects. The browser is based on the open source project Chromium, and supports modern web technology such as HTML5, CSS, and Javascript.

More information on the Web Browser Module can be found on the Web Browser Component page.



Twilio Alarm Notification

On this page

...

- Send SMS Messages through Scripts
  - Avoid the Hardware Hassle

Send SMS Messages through Scripts

The Twilio Alarm Notification module adds integration with Twilio via an SMS Alarm Notification Profile. Send SMS alarm notifications without a physical device. This module unlocks the system.twilio functions. These allow you to send SMS message via python script. See the system.twilio section of the Appendix for more details.

Avoid the Hardware Hassle

The Twilio Integration and Alarm Notification module does not require additional hardware, so there is no need to worry about firmware updates or hardware failure.

More information on using the Twilio Integration and Alarm Notification module can be found on the Twilio SMS Notification page.

⚠ The Twilio module is dependent on the Alarm Notification module, you must have both to send alarms.

Serial Modules

On this page

...

- Serial support
  - Multiple Scopes

Serial support

The Serial modules add serial comm support and system.serial functions to Python scripts. These functions allow you to open, read from, write to, and close serial ports. More information on these function can be found in the system.serial section of the Appendix. This module provides access to the serial port on the Ignition Gateway and systems running Vision Clients.

Multiple Scopes

There are two variants of the Serial Module. Both perform similar functionality, but at different scopes:

- The Serial Support Client module allows the system.serial functions to be accessed from client based scripts, as well as connect to serial devices plugged into the computer running the client.
- The Serial Support Gateway module allows the system.serial functions to be accessed from Gateway based scripts, as well as connect to serial devices plugged into the Gateway server.

SECS/GEM Module

On this page

...

- Communicating with Semiconductor Fab Equipment

Communicating with Semiconductor Fab Equipment

Ignition's SECS/GEM module is a highly specialized module that provides the ability to communicate with semiconductor fab equipment using the SECS-II, GEM, and HSMS standards. The SECS/GEM module is implemented as a host that can talk to one or more fab tools via equipment connections configured within the Ignition Gateway. It also provides the capability to create and configure equipment simulators that can be used for testing applications which is a big plus when getting started with the SECS/GEM module. Connections are made over Ethernet using TCP/IP (HSMS protocol) or over a direct serial connection (SECS-I protocol) using a RS-232 cable. Once installed, the SECS/GEM functionality is fully integrated into the Configure section of the Gateway webpage where you can configure connections to the equipment, and setup equipment simulators.

The SECS/GEM module provides the framework to create and configure a common set of scripting functions to control equipment behavior and communication capabilities that monitor and support semiconductor manufacturing equipment. Through scripting, it provides the flexibility to start and stop equipment processing, set equipment configuration parameter values, collect data, send alarms to the host, and select recipes, as long as the host and OEM adhere to SECS/GEM standards.

Communication between Ignition and the equipment is in the form of SECS messages defined by the SECS-II standard. SECS messages are typically sent and received using scripting functions within Ignition using the SECS/GEM defined messages (SECS Definition Language (SDL)) that are loaded into the equipment. SECS messages that contain data can also be written to Tags, updating the Tag's values as new messages come in. This functionality is used to capture data coming back from equipment, such as tool traces within S6F1 messages. The default SDL file that is included with the SECS/GEM module is already configured to create Tags for S6F1 messages sent to the module from the tool. For other messages, the SECS/GEM module can be configured to update Tags by adding some additional parameters to the equipment's SDL file.

To learn more, refer to the SECS/GEM section.

System Architectures

Overview

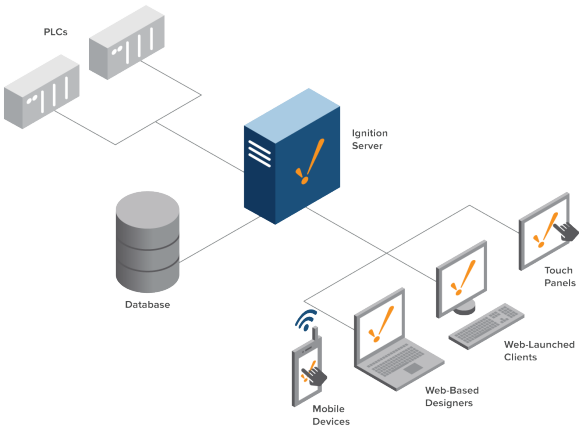
Ignition is flexible enough to meet the demands of any enterprise infrastructure large, small, or somewhere in-between. You can deploy Ignition at one site, multiple sites, or host it in the Cloud. This section displays several common architecture examples. Many are simply examples, but they are a great starting point. Additionally, these example architectures can be modified or scaled out to accommodate a number of different requirements.

Click on each architecture to learn more about it.

On this page

...

- Overview
  - Standard Architecture
  - Scale Out Architecture
  - Hub and Spoke Architecture
  - IIoT Architecture
  - Enterprise Architecture
  - Redundancy Architecture
  - Cloud Based Architecture



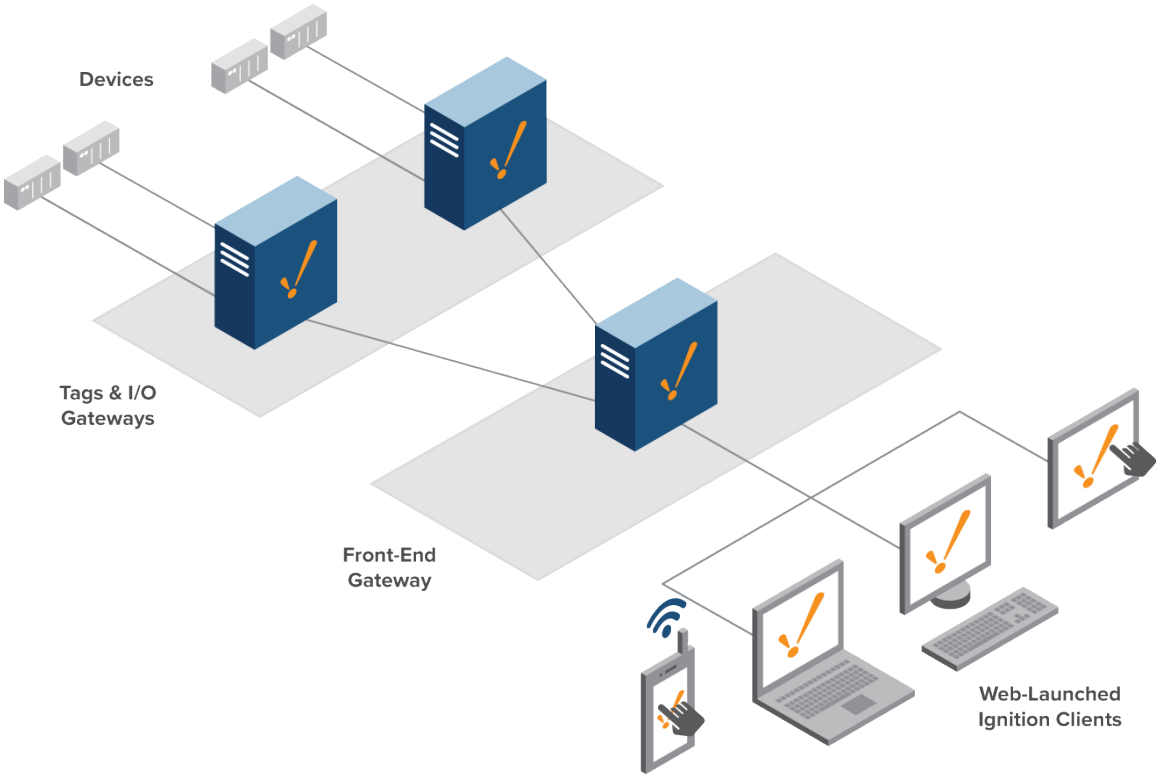
Standard Architecture

INDUCTIVE UNIVERSITY

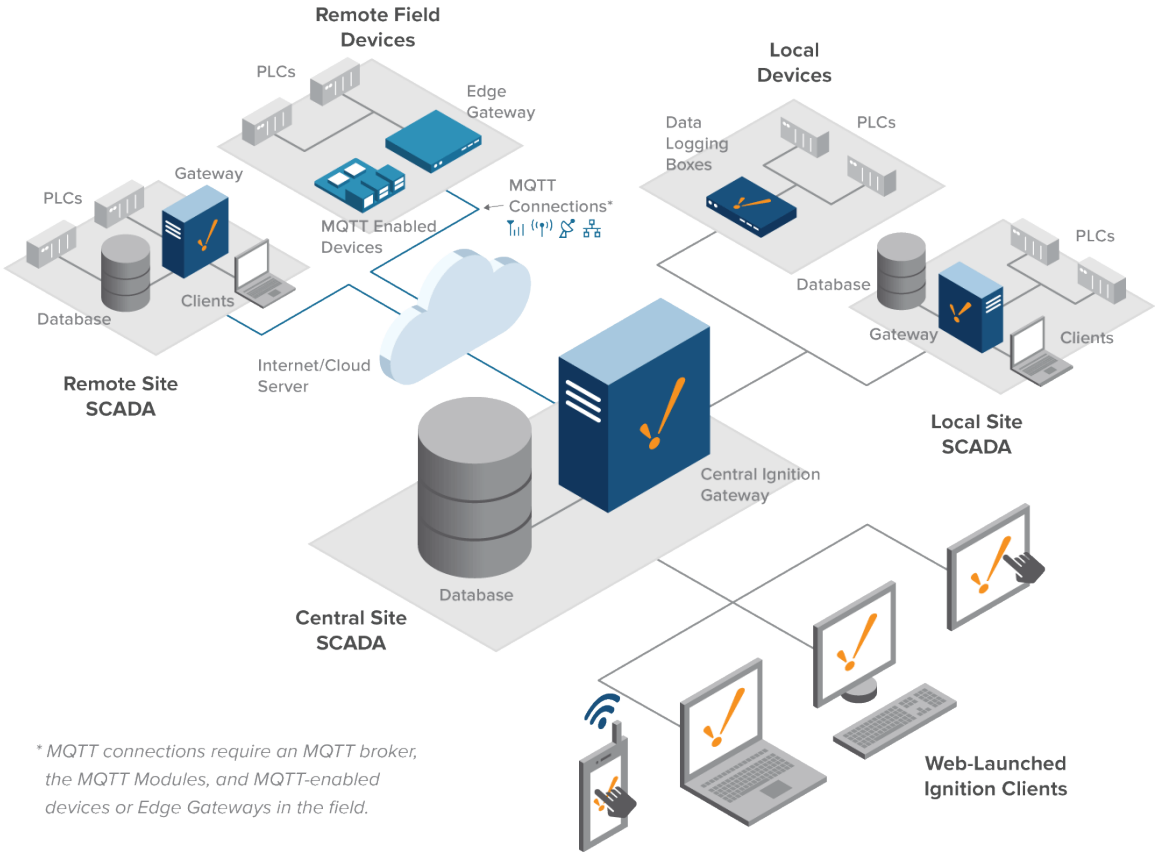
Ignition System Architecture

Watch the Video

Scale Out Architecture

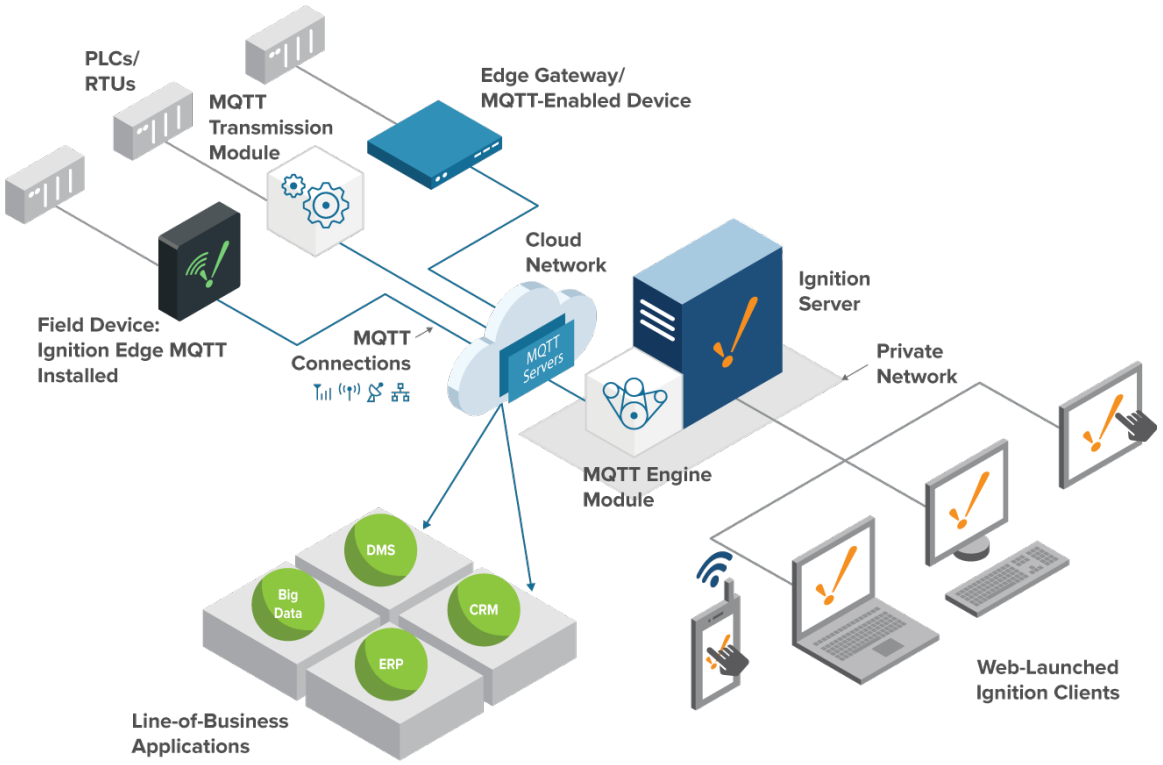


Hub and Spoke Architecture

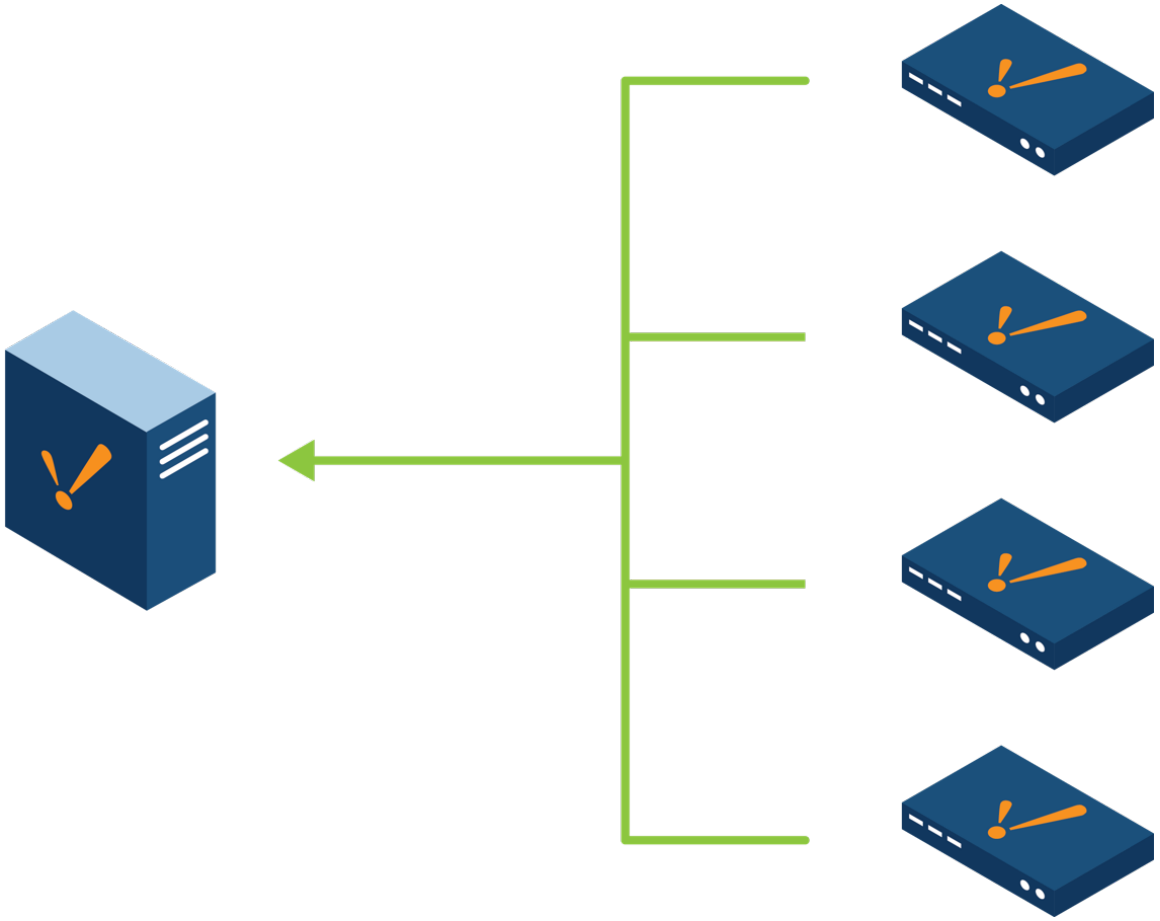


** MQTT connections require an MQTT broker, the MQTT Modules, and MQTT-enabled devices or Edge Gateways in the field.*

IIoT Architecture

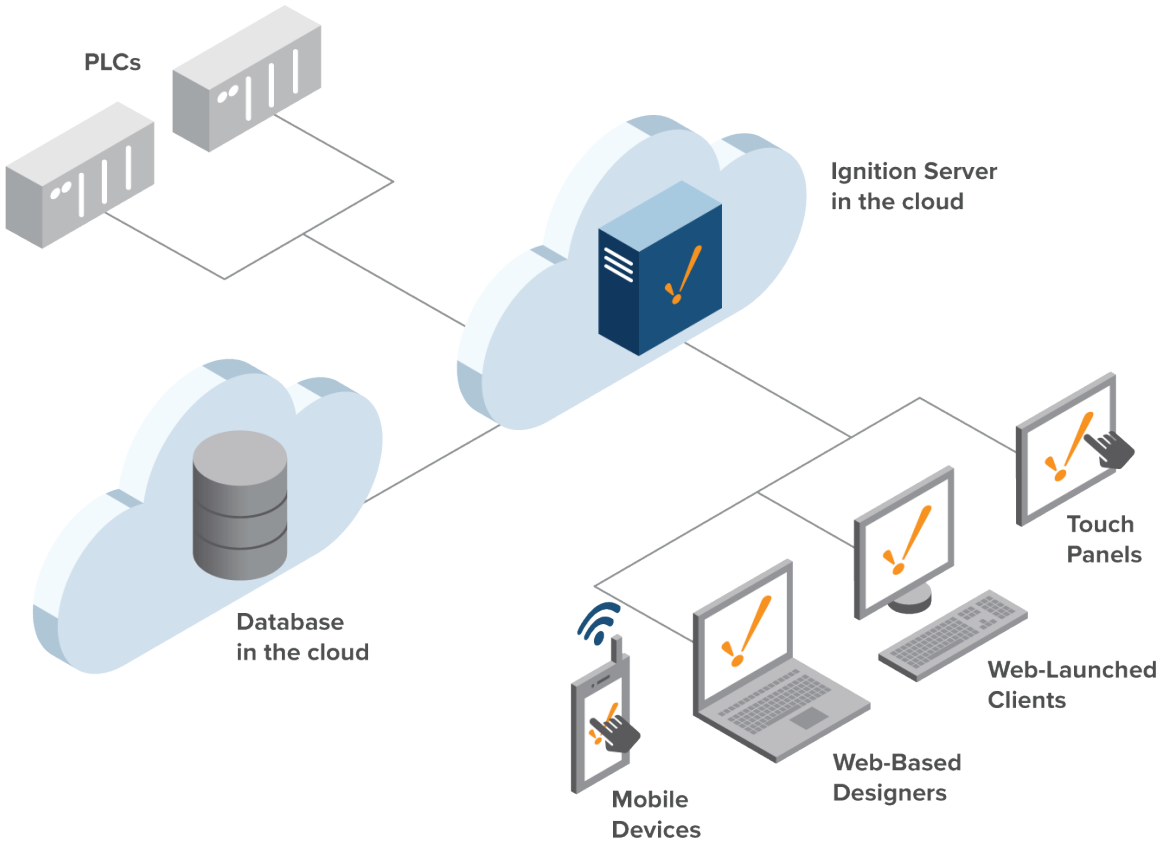


Enterprise Architecture



Redundancy Architecture

Cloud Based Architecture



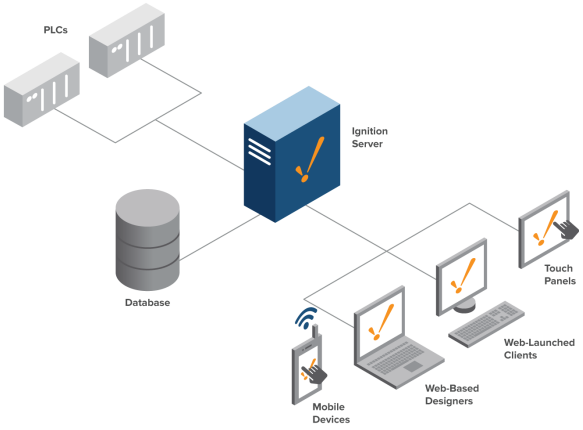
In This Section ...

Standard Architecture

Single Server Install

Ignition's most basic architecture is a standard single installation. This is very simple to setup and it provides the powerful functionality of the Ignition Gateway and it can serve as a starting point for other more complicated architectures.

The Standard Architecture consists of a single Gateway that can connect to multiple PLCs, databases, and other devices, all with a single licensed install. With the Ignition licensing system, adding new devices, databases, and clients is easy. Simply connect them to the Ignition Gateway, no licensing changes required. The Ignition Gateway can launch multiple clients on everything from touch panels to workstations and even mobile devices, all from one Ignition installation.



On this page

...

- Single Server Install
  - One Server, Multiple Networks

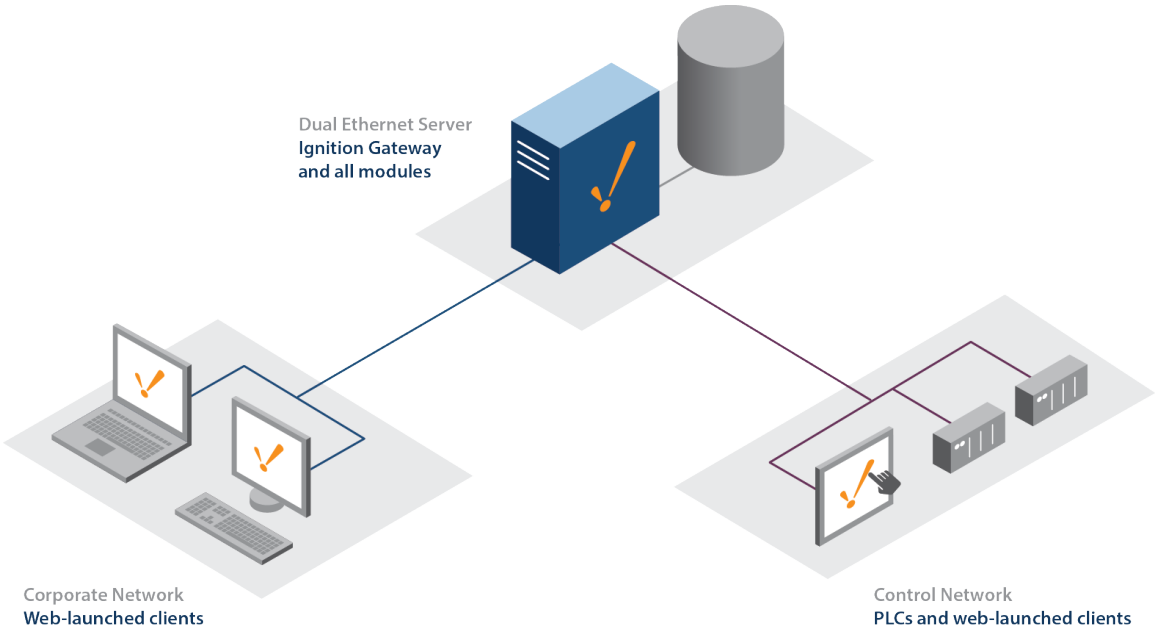
INDUCTIVI UNIVERSIT

Standard Architecture

Watch the Video

One Server, Multiple Networks

The Ignition Gateway supports dual-NIC servers, and can act as a bridge between multiple networks, or communicate with multiple sites over a corporate WAN. Since clients talk to databases and PLCs through the Gateway, clients can be launched from both a corporate network and an isolated control network, and provide full access to both. Built-in security settings can restrict project access to users on different networks, either by restricting certain things in a project, or denying access to a whole project based on user role and network location.



Related Topics ...

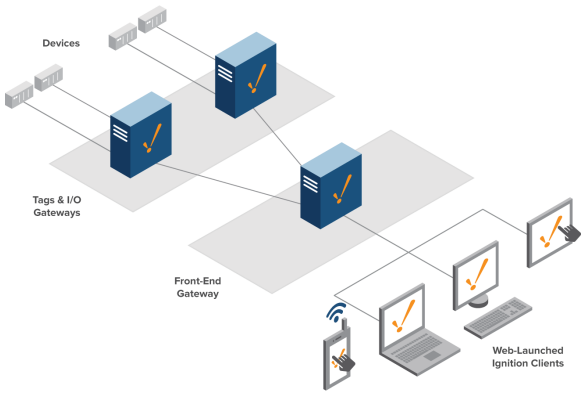
- Scale Out Architecture
- Redundancy Architecture

Scale Out Architecture

Scaling Out Ignition for Larger Systems

In larger systems, it is often easier to have multiple Ignition installations to help split the load between the front-end tasks and the back-end tasks. This is perfect for single large systems that aren't split up into different sites. In those cases, the Hub and Spoke Architecture is usually a better fit.

In the Scale Out Architecture, we have at least one Gateway that handles back-end communications. The back-end Gateway deals with all PLC and device communications. The front-end Gateway handles all of the Clients, serving up the data pulled from the back-end Gateway. This is made possible through the Gateway Network, connecting Gateways to each other, and allowing Tags to be shared through remote Tag providers.



On this page

...

- Scaling Out Ignition for Larger Systems
  - Load Balancing
  - Gateway Types
  - Database Placement
  - Scaling Out the Databases
- Why Use the Scale Out Architecture?
  - Decentralized Servers
  - Pooled Resource Usage
  - Easy Growth

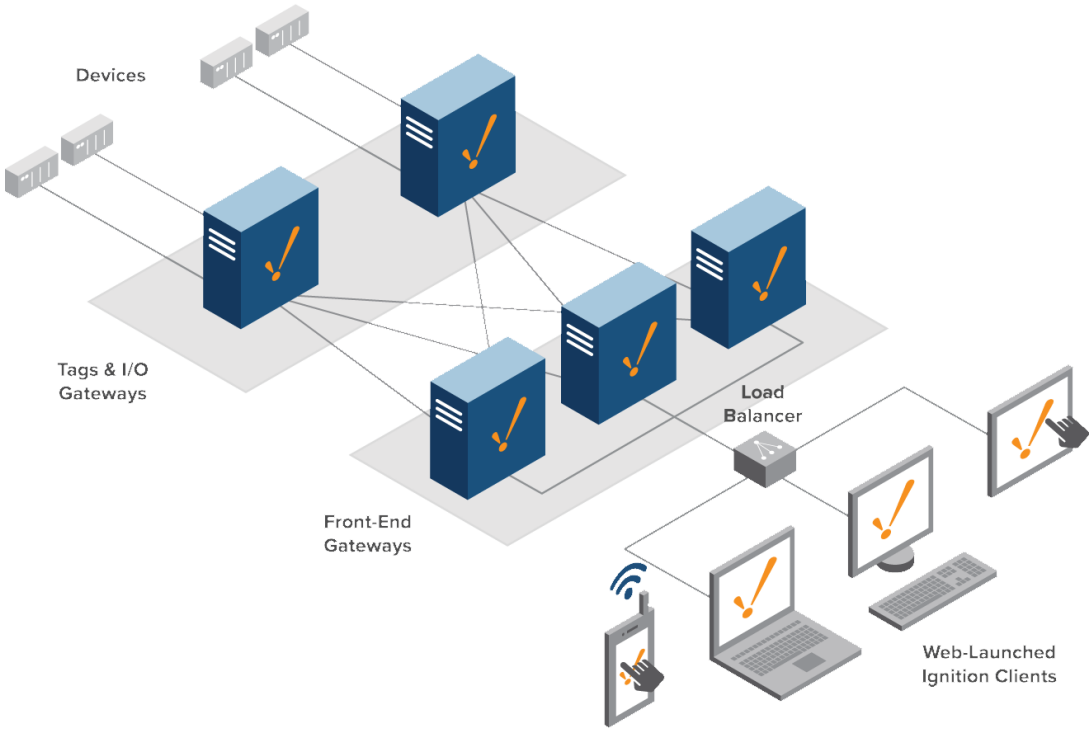
INDUCTIVE UNIVERSITY

Scale Out Architecture

Watch the Video

Load Balancing

The best thing about the Scale Out Architecture is that it is easy to scale up Ignition as your system grows. In the image below, we added more front-end Gateways to help handle an increase in clients, and a Load Balancer to automatically distribute the clients between them.



Gateway Types

When utilizing Scale Out, an Ignition Gateway is given focus due to its placement. There are two possible types:

- **Tags & I/O**: Also known as a **Back-End Gateway**, is responsible for communicating with the PLCs, and executing Tags. They share their local Tags with the Front-End Gateways via a Remote Tag Provider, and are generally responsible for recording history. Additionally, alarms can be added to Tags locally, and exposed elsewhere in the system.
- **Front-End Gateway**: Sometimes called a **Visualization Gateway**, this Gateway is responsible for hosting the clients, allowing the Back-End Gateways to focus solely on PLC communication and history.

Database Placement

With this type of architecture, both types of Gateways will need database access: the Tag & I/O Gateways need to be able to store data, while the Front-End Gateways need to query the data out.

Scaling Out the Databases

Multiple on-site databases may also be utilized with this architecture. This typically involves configuring a database cluster.

Why Use the Scale Out Architecture?

Scale Out is utilized in large systems to distribute the workload across multiple servers. This is in contrast to a 'Scale Up' ideology where you have a single powerful server manage the whole system.

Decentralized Servers

In this architecture, each server only deals with a small part of the overall system. Should a server fault, only that part of the system is hindered, allowing other parts in organization to continue uninterrupted. Applying Redundancy to the the Tag & I/O servers is ideal to maximize uptime.

Pooled Resource Usage

Since multiple servers are working together bottlenecks are reduced. A massive amount of work can be performed in this system without slowing down other servers.

Easy Growth

Improving performance in this architecture is simple: add another Gateway where appropriate. Because the system is spread out on purpose, it's easy to determine which kind of Gateway to add. A Back-End Gateway would help if you're adding new PLCs, and a Front-End Gateway will help when more clients need to be active at a time.

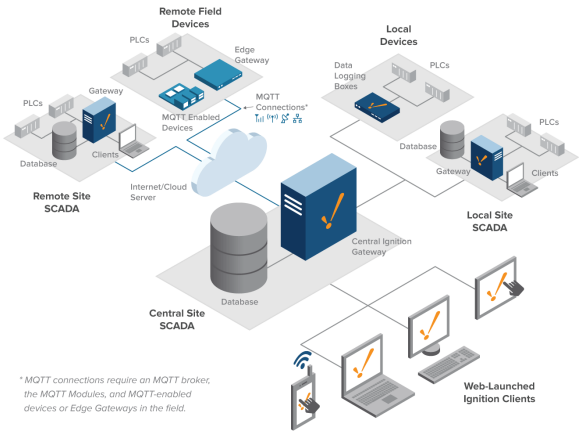
Related Topics ...

- Hub and Spoke Architecture

Hub and Spoke Architecture

Centrally Managed Systems

The Hub and Spoke Architecture involves multiple Ignition installations (the Spokes) communicating and forwarding data points to a centralized Ignition installation (the Hub). The Hub and Spoke Architecture is unique in that you start with a single Ignition install at a central site. Ignition can then be installed at as many other remote or local sites as needed. These sites can then be customized or configured to suit that specific site's needs.



On this page

...

- Centrally Managed Systems
  - Reliable Remote Data Logging
  - Local and Remote Logging
  - Local Client Fallback
  - Remote Site Alarming

INDUCTIVE UNIVERSITY

Hub and Spoke Architecture

Watch the Video

Reliable Remote Data Logging

Because connections between a database and remote PLCs can be unreliable, we can install either the SQL Bridge module or Tag History module. The Store-and-Forward system then ensures that data is never lost. The system may still be accessed through the OPC-UA server built into ignition, or database-based Tags may be used to communicate current status. By making use of Ignition's Store-and-Forward system, data loss is prevented, ensuring that history always reaches its destination in the database.

The Hub and Spoke Architecture is ideal for centralizing historical data from multiple remote sites, and allows access to locally collected data via Ignition's Gateway Network.

Local and Remote Logging

The Hub and Spoke Architecture can be modified to store data at each spoke, in addition to the hub by splitting the Tag history. Redundant data is stored at each site as well as the hub database to protect against hard drive failure.

Local Client Fallback

Typically, the Hub and Spoke Architecture does not include the Vision Module at the remote sites. You can add the Vision Module at each spoke to take advantage of local client fallback. This way each site can continue to operate at full efficiency in the event communication with the hub Gateway drops. Users at each spoke will never lose visibility of the system. Vision Limited licenses help lower costs on these spoke.

Remote Site Alarming

By adding the Alarming Notification module to a remote site, you can notify users when a problem occurs at each spoke. It is best to handle alarms at each remote site rather than centrally to avoid any problems in case the connection to the central location is lost.

Related Topics ...

- IIoT Architecture

Edge Architectures

Overview

On this page

...

- Overview
- Panel
- Edge IIoT
- Edge Sync Services
- Edge EAM

The Ignition Edge system is unique in that it can be very flexible and fill in the gaps in any system architecture. Ignition Edge is a smaller version of the standard Ignition install. Edge is used to sit at the edge of a network and do a few specific tasks like client failover or push MQTT data.

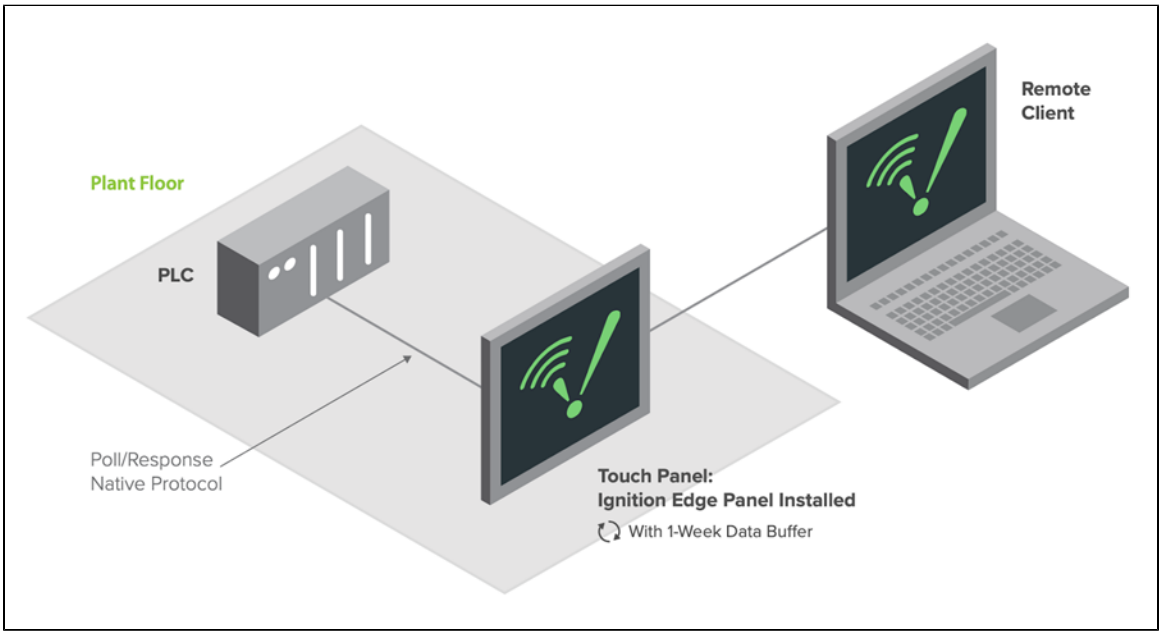
Each Edge plugin (Panel, IIoT, Compute, Edge Sync Services and EAM) offers specific functionality that works best in certain scenarios, but because any combination of plugins can be installed, an Edge Gateway may fill multiple roles and help bridge two different architectures. The diagrams show some architecture options using Edge. Additionally, these architecture diagrams show an Edge Gateway paired with an Ignition Gateway, which shows how an Edge system architecture can be added on to any of the other Ignition architectures.

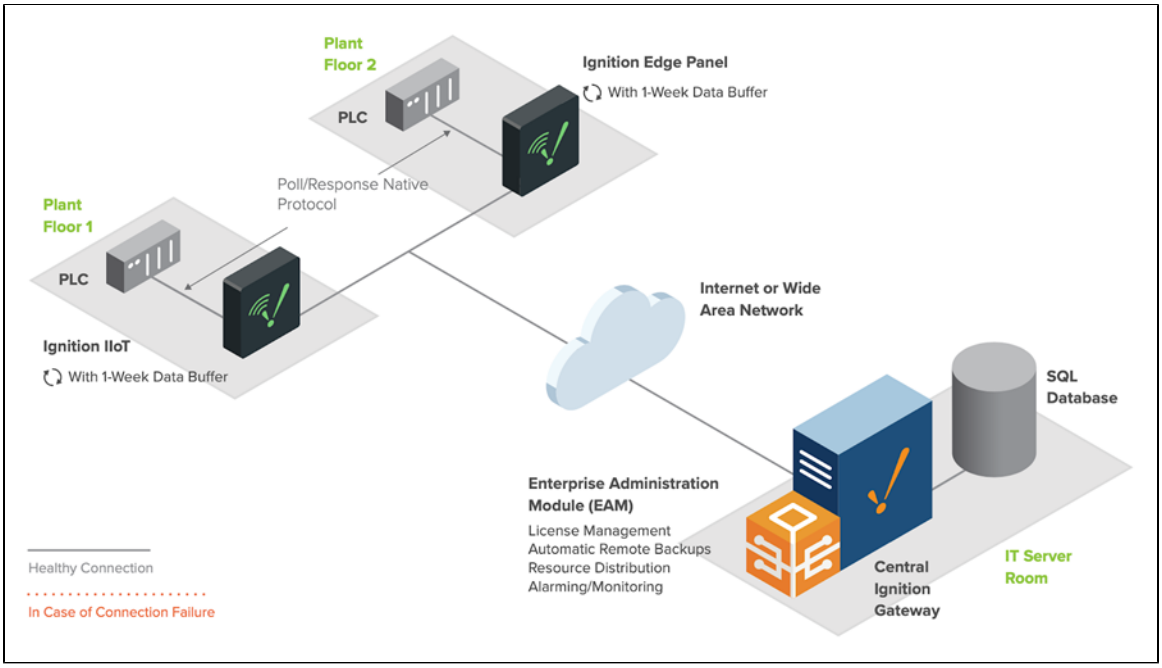
> The following feature is new in Ignition version **8.0.10**
> Click here to check out the other new features

Ignition Edge was revised and expanded significantly as of release 8.0.10. For more information, see Ignition Edge.

Panel

Edge with the Panel plugin can act as a stand-alone panel by simply connecting to a local PLC and building a basic screen. Paired with a central Ignition Gateway, the Edge can also act as a local client fallback, providing control on site in the event of a lost network connection.

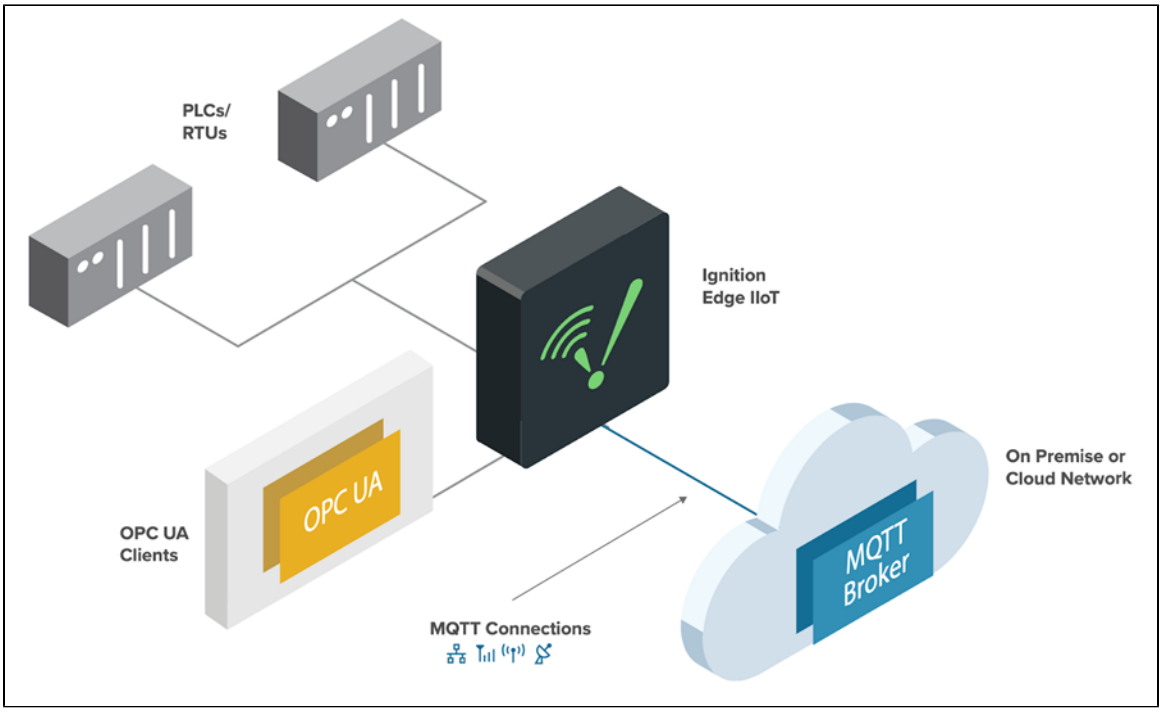




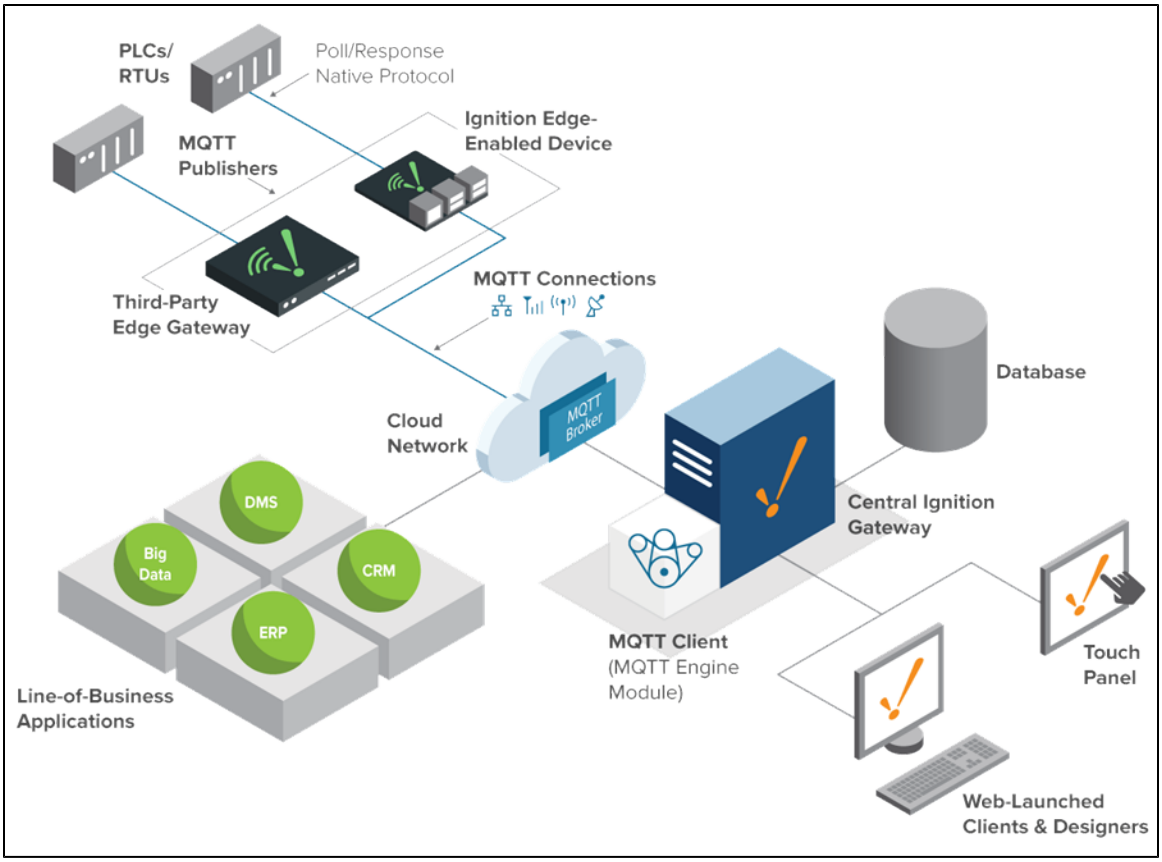
Edge IIoT

Edge with the IIoT plugin (formerly MQTT) provides a simple way of setting up an MQTT Publisher next to a PLC, allowing it to connect to a greater IIoT architecture. Here you can see multiple types of devices using Ignition Edge to connect in the network.

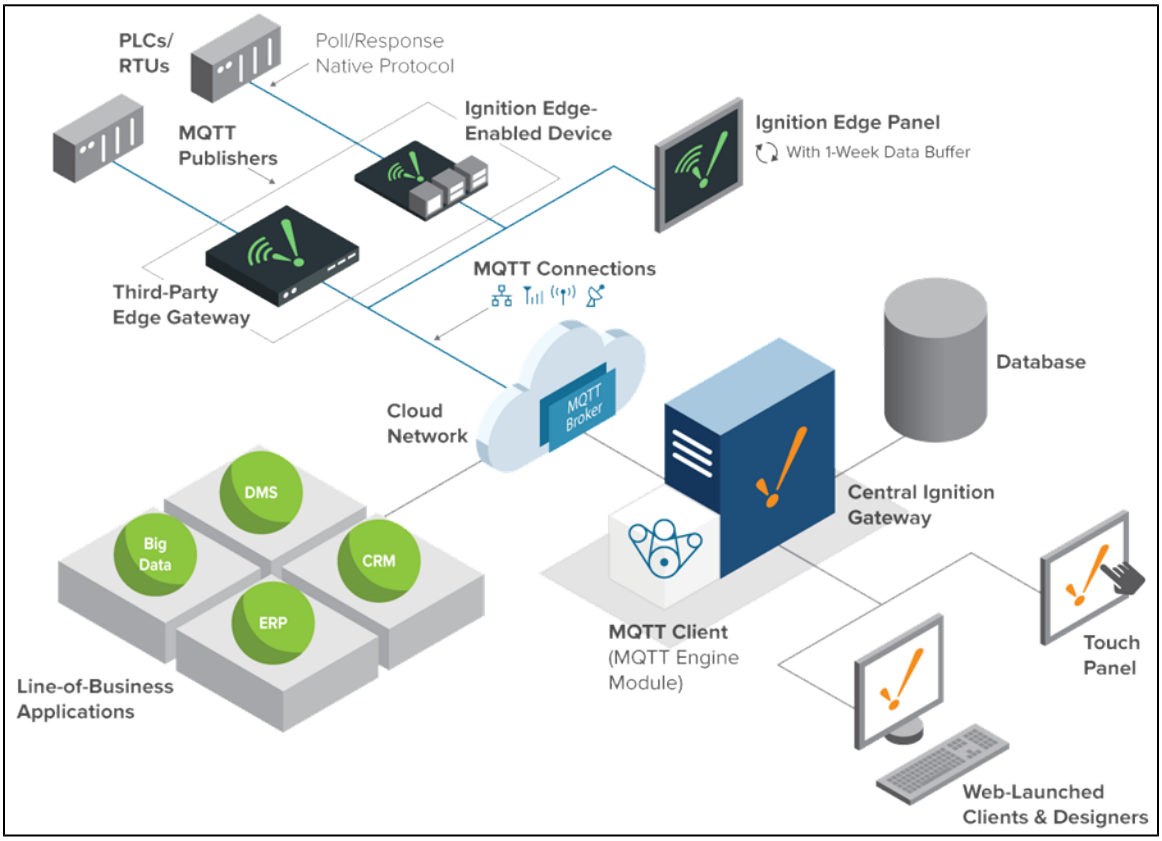
Edge IIoT as a standalone edge gateway to publish device data from the edge of the network to an MQTT broker:



Edge IIoT as an edge gateway to publish device data from the edge of the network to an MQTT broker:

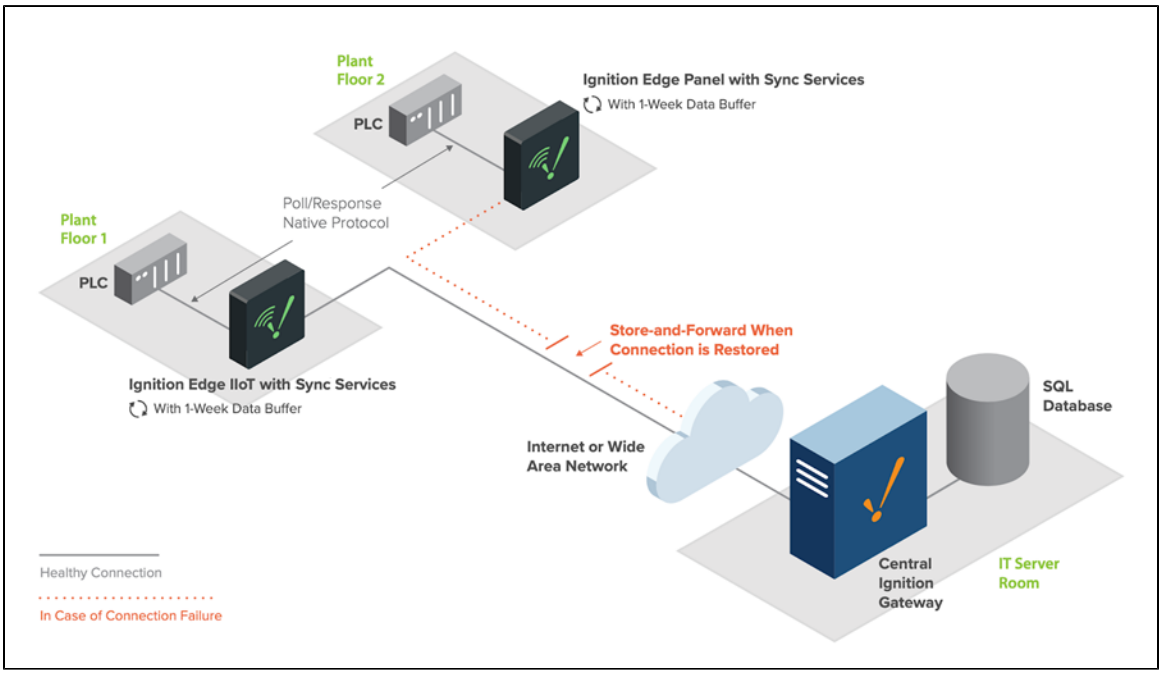


Edge IIoT and Edge Panel together to add a local client at the edge of the network and to publish data to an MQTT broker.



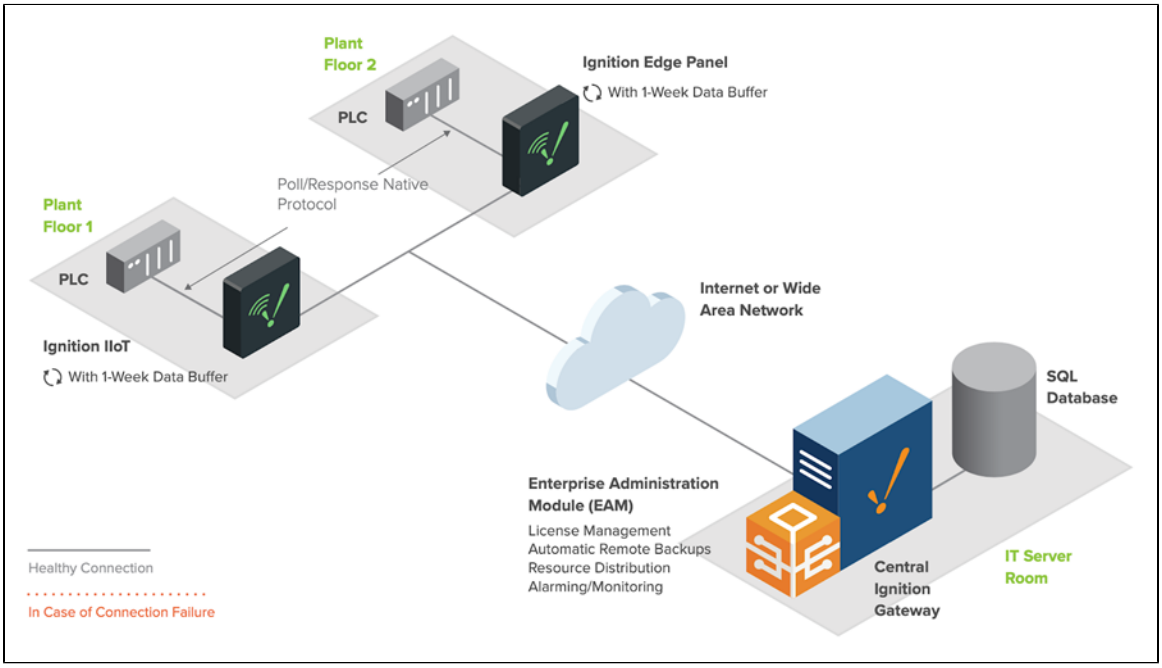
Edge Sync Services

Ignition Edge Sync Services acts as a limited remote server that synchronizes data from the edge of the network to a central Ignition server.



Edge EAM

Edge EAM acts as a remote EAM Agent Gateway and can easily combine with any other Edge edition for central management.



Related Topics ...

- Redundancy Architecture
- IIoT Architecture

IIoT Architecture

Overview

Ignition can collect data from any devices at the edge of the network, publish that data to a central broker, and push that data to subscribed industrial and line-of-business applications. Ignition IIoT can connect to PLCs in the field through the use of the MQTT Transmission module, field devices with Ignition Edge MQTT installed, and/or MQTT-enabled edge gateways and field devices that use the Cirrus Link Sparkplug MQTT specification. This data is published to an MQTT broker, this broker can be located on-premise, in the cloud, or a hybrid of the two. The MQTT Engine module located on an Ignition Gateway can subscribe to any data published from the broker. This data can be used in any Ignition application.

On this page

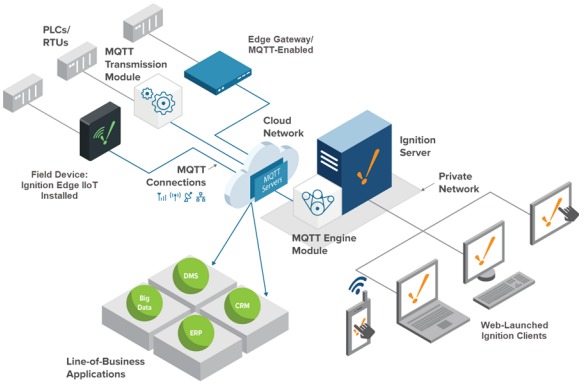
...

- Overview
- IIoT Architecture Example
- IIoT Architecture Objects
  - Field Device
  - MQTT Server (Broker)
  - Subscriber

IIoT Architecture Example

IIoT architecture typically involves the use of the MQTT protocol to communicate with a large number of devices. MQTT is a lightweight and secure protocol that makes use of a unique publish /subscribe transportation method. Additional information on the MQTT protocol and modules can be found in the Cirrus Link documentation.

The Standard Architecture can easily be expanded with the MQTT modules to collect data from numerous Edge of Network devices. Data collected by the MQTT Engine can be historized and presented in a client with the Vision module, or via reports with the Reporting module.



INDUCTIVE UNIVERSITY

IIoT Architecture

Watch the Video

IIoT Architecture Objects

The following are key pieces in the IIoT Architecture.

Field Device

A Field Device of some sort can be used to connect to a PLC. This device also needs to be able to publish Tag values to an MQTT server. Any of the following hardware can act as a Field Device:

- Ignition Edge with the MQTT plugin
- Ignition server with the MQTT Transmitter Module
- Third-party MQTT enabled device

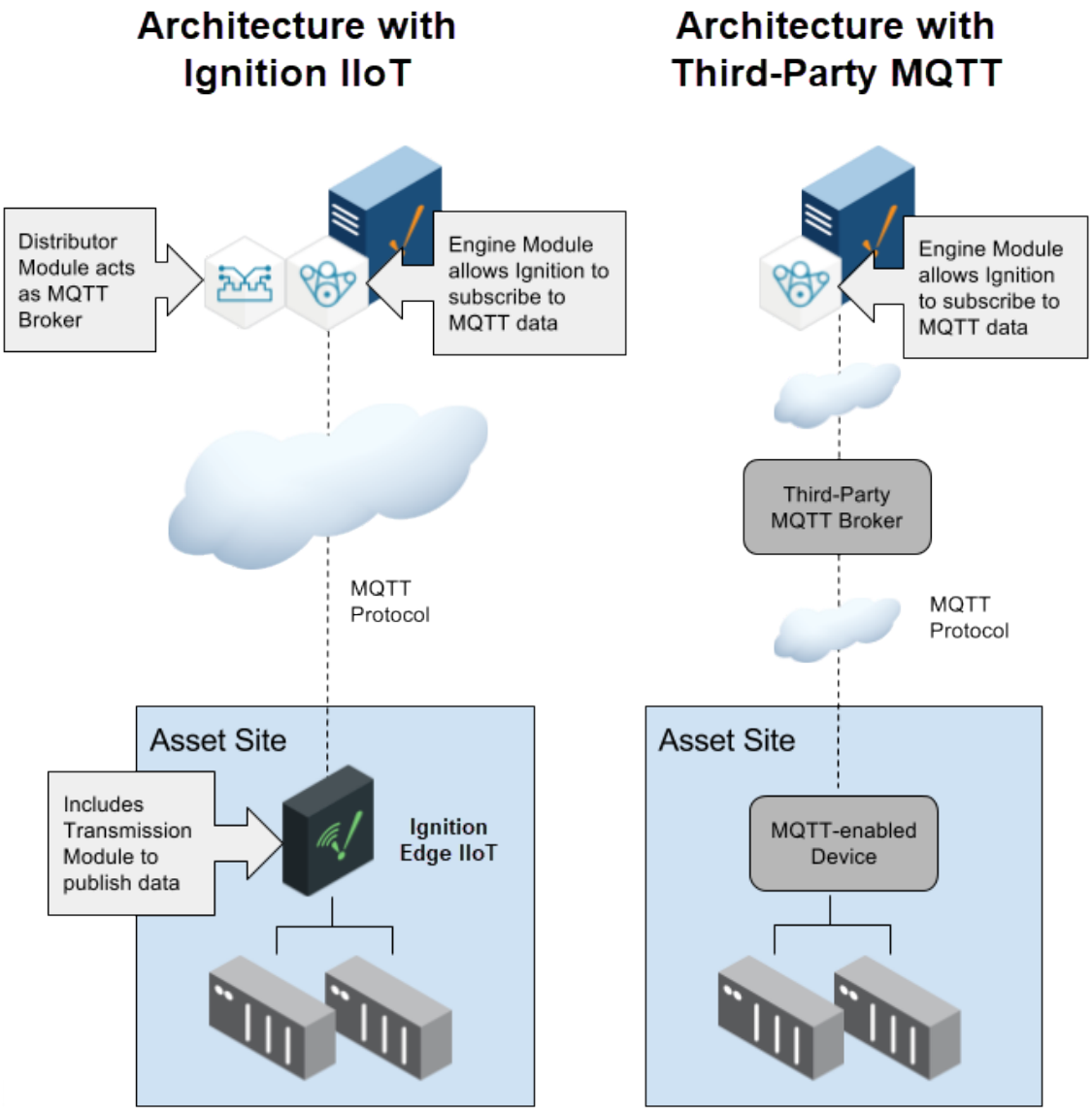
MQTT Server (Broker)

An MQTT Broker is a server that can communicate with your Field Devices via MQTT. This can be an Ignition Gateway with the MQTT Distributor Module installed, or a third-party server, such as a Chariot MQTT Server.

Subscriber

A subscriber is some application that subscribes to topics in the Broker. In regards to Ignition, this is the Gateway with the the MQTT Engine Module installed.

More details on the MQTT modules can be found in the Cirrus Link documentation.



Related Topics ...

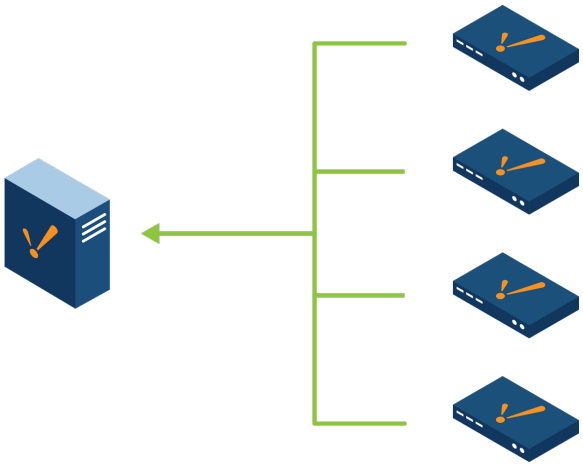
- Enterprise Architecture
- Edge Architectures

Enterprise Architecture

Overview

Many types of architecture involve multiple Ignition servers. Eventually, some assistance overseeing each server can be helpful. Enterprise Architecture typically involves having a mechanism to monitor and manage multiple Ignition Gateways. This involves utilizing the Enterprise Administration Module (EAM).

EAM specifies one Gateway as the Controller, and the others as Agents. Connections between these Gateways take advantage of the Gateway Network, which is a secure method multiple Gateways can use to share information.



On this page

...

- Overview
- Why Use Enterprise Architecture
  - System Management
  - Project Synchronization

Why Use Enterprise Architecture

System Management

Monitoring the health of several systems can be time consuming. EAM streamlines system management by having each Gateway report to a single Controller Gateway. This Controller can then generate alarms when problems occur, or assist in recovering from a hardware failure. Agent events can be stored in a SQL database, and easily incorporated into an Ignition Report. The best part is that the EAM can easily be incorporated into any architecture where multiple Gateways are present.

Project Synchronization

When multiple Gateways are used to launch clients, such as in the Hub and Spoke Architecture, standardizing the look and feel of projects across both Gateways is important for a consistent organization. With EAM, the Controller can synchronize resources across multiple Agents, allowing development to occur on one Gateway, and updates pushed out to other Agent automatically.

Related Topics ...

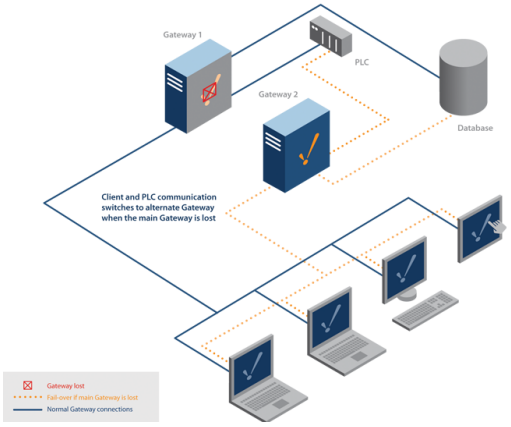
- Edge Architectures

Redundancy Architecture

Mission Critical - Gateway Redundancy

Using Redundancy, two Ignition installations can be linked together, so if one server fails, the other takes over and continues executing. All of the clients connected will be redirected to the backup machine, and historical data will continue to be logged. The transition is seamless, meaning processes will never be prevented from executing due to one server going down.

Any single installation in the System Architectures can be replaced with a redundant pair of Ignition Gateways. If you have a dedicated backup computer, Inductive Automation offers a cheaper Backup-specific license.



On this page

...

- Mission Critical - Gateway Redundancy
- Redundancy Architecture Examples
  - Standard Architecture
  - Hub and Spoke
  - Scale Out Architecture
  - Scale Out Architecture with a Load Balancer

Redundancy Architecture Examples

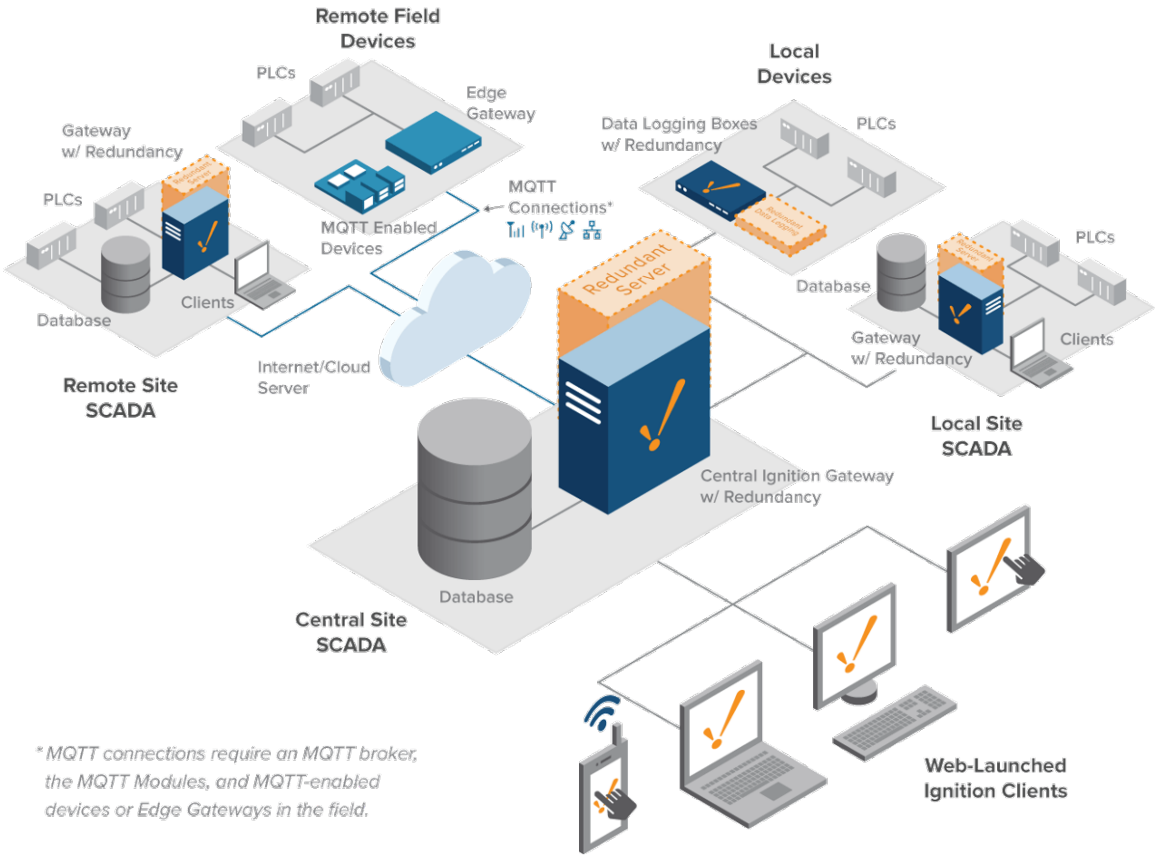
In this section, we will take a look at several other types of architectures that have been enhanced by redundancy.

Standard Architecture

Reliability in the Standard Architecture can dramatically be improved by adding a redundant server. Should the primary server fail, the redundant server can seamlessly take over, allowing production to continue while the primary is brought back online.

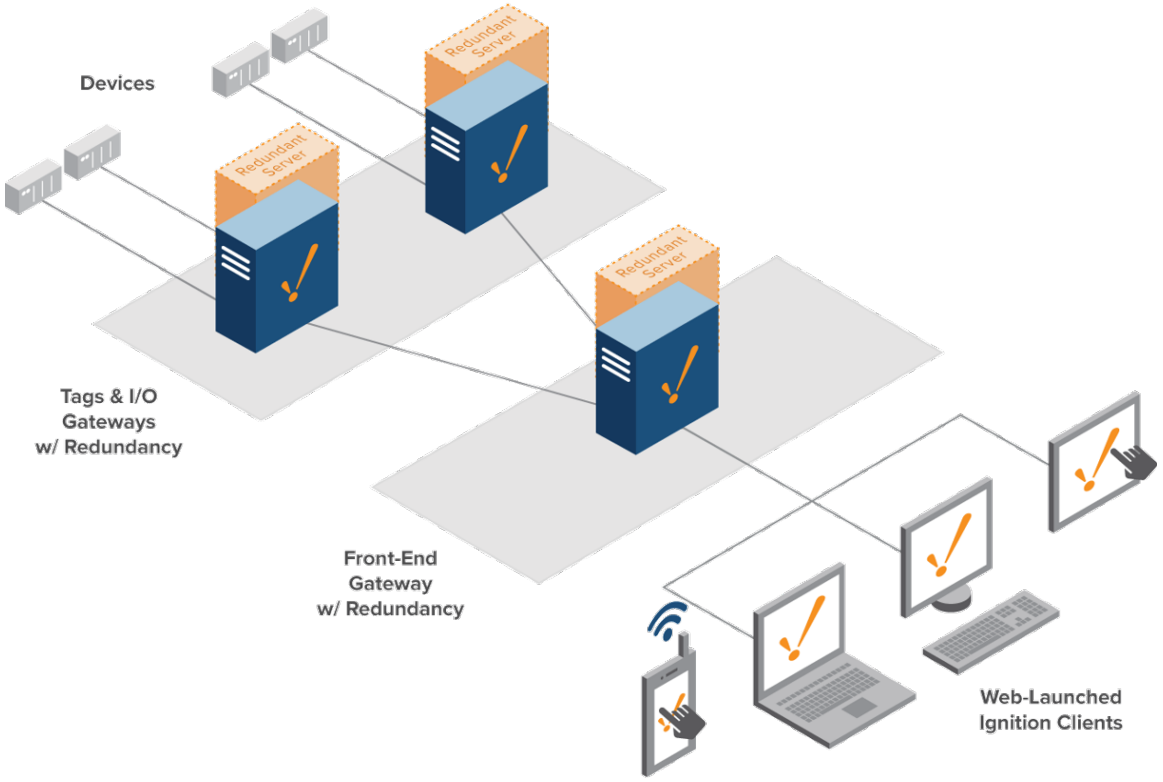
Hub and Spoke

The Spokes that contain an Ignition Gateway can be thought of as individual instances of the Standard Architecture. Incorporating a backup server at each of these Spokes will ensure uptime is maintained locally. The Hub can also utilize a backup server, which is especially useful if clients are also launched from it.



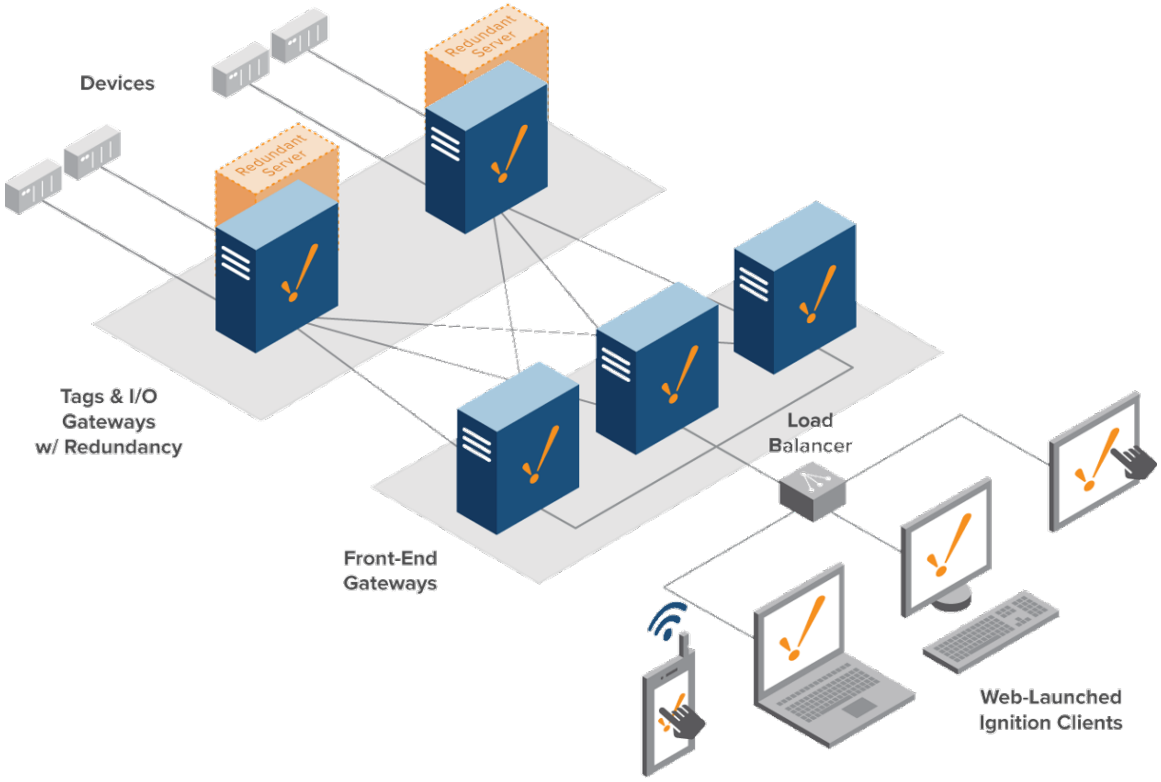
Scale Out Architecture

A server fault using the Scale Out Architecture, while not fatal to the whole system, can cripple the system. Without a Load Balancer, a Backup Gateway should be utilized at each Gateway. These means each Gateway has a failsafe ready to take over in the event of a failure.



Scale Out Architecture with a Load Balancer

When utilizing a Load Balancer, redundancy should not be applied to the Front-End Gateways: the Load Balancer is responsible for distributing the workload amongst the front-end, so a backup Gateway is not required. However, an extra Front-End Gateway could be incorporated to act as a failover if you plan on having three Front-End Gateways active at any time, then a fourth Gateway should be available in case one server faults.



Related Topics ...

- Cloud Based Architecture

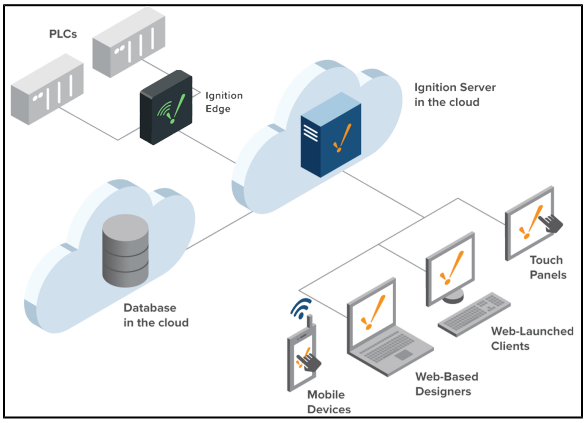
Cloud Based Architecture

Hosted Ignition Server

Ignition server can be hosted in the Cloud. Inductive Automation does not provide cloud services, but there are many great providers out there, such as Amazon EC2, or Windows Azure. The server can connect to the PLCs at the remote or customer sites via Round Robin Poll, Cell Tower, Satellite, or Secure VPN. The clients are connected to the Ignition server via Internet. Liberal use of Ignition's security features protects each customer from accessing another customer's projects or data.

Architecture Example

In this example, we see the Standard Architecture, but with the Ignition Server and Database both running on cloud servers. Note, that the Database and Ignition are on different cloud servers. This is because cloud service providers generally have specialized services that are designed explicitly for hosting SQL databases, as well as more general purpose services such as running Ignition. An installation of Ignition Edge is placed local to the PLCs, and is responsible for forwarding data to the cloud.



On this page

...

- Hosted Ignition Server
- Architecture Example
  - Why Use a Cloud Based Architecture?
  - Does the Database Always Have to be in the Cloud?
  - Are Cloud Based Servers Secure?
  - Should I have a Redundant Ignition Server in the Cloud?
  - What if I Require an On-Site Server with Redundancy?
- Cloud Service Suggestions
  - Amazon Web Services
  - Microsoft Azure

Why Use a Cloud Based Architecture?

This type of architecture is ideal in smaller organizations since you don't have to worry about maintaining a server. Additionally, there is no on-site IT staff required to monitor the system.

Does the Database Always Have to be in the Cloud?

If Ignition is hosted in the cloud, then it is very common to have your database also hosted in the cloud. However, many of our customers find that also having a controls database on-site to store temporary or in-progress data useful. This is great when you want to have data stored locally.

Are Cloud Based Servers Secure?

Cloud based servers are increasingly common, but it is important to educate your organization on the risks associated with them. If your organization is interested in a cloud based solution, we recommend following the guidelines in the Ignition Security Hardening Guide, as it addresses many potential security issues for both cloud and non-cloud based servers. Additionally, if offers recommendations on how to better protect your system.

Should I have a Redundant Ignition Server in the Cloud?

If you are using redundancy or are concerned about uptime, you should be using Ignition redundancy regardless of whether your Ignition is in the cloud. While most cloud providers offer built-in redundancy, this is often much slower than Ignition's redundancy failover time. Typically Cloud Server redundancy has a new computer instance spun up (started) in the event the original fails. This will often take a couple minutes where Ignition's redundancy (with default settings) takes around 20 seconds to switch over. There are other options to make a Cloud Server start faster, but currently they all slow down the CPU substantially to make it possible.

What if I Require an On-Site Server with Redundancy?

If you are using Ignition's Redundancy with a Cloud Server, then typically we recommend that both of the servers are run side-by-side. This means either both in the cloud or both on site. One warning: If you have both instances of Ignition in the cloud but if your connection to the internet goes down, then you lose access to both instances of Ignition.

Cloud Service Suggestions

Many customers find the following cloud based service suggestions helpful:

Amazon Web Services

- Ignition Hosting: AWS EC2 Instance types. The M4 and M5 (including the M5a and M5n series) instance types generally meet most customers needs in regards to hosting Ignition.
- Database Hosting: AWS Relational Database Service.

Microsoft Azure

- Ignition Hosting: Azure for Discrete Manufacturing
- Database Hosting: Azure SQL Database

Related Topics ...

- Standard Architecture