

# Deploying Applications

When you are finished developing and testing your project, you can deploy the runtime application so it can be used by end-users. Projects can be deployed to run locally on a stand-alone computer or embedded device, in a client-server distributed architecture, or on the cloud. This chapter discusses the various methods for deploying these applications.

---

## Local and Distributed Systems

FactoryStudio supports many deployment scenarios such as:

- Embedded devices with no local user interface.
- Stand-alone panels or computers with local user interface.
- Distributed systems with many client stations accessing a server.
- Redundant servers with multiple users connected.

The setup procedures for each scenario have many common steps. For all systems, FactoryStudio uses a client-server architecture.

The server components are the project file and the modules that run the server side tasks, such as data-acquisition, alarms, and data logging. The client components are the graphical user interfaces and related scripts.

When you have a local or stand-alone project, the server and the client components run on the same machine.

FactoryStudio's client technologies simplify deployment because you only need to install the project on the server computer. The client's will use the project from the server.

In order to setup the server components or the stand-alone configurations, refer to the following sections: "Product Installation on the Target Computer", "License and Project Settings Verification", and "Installing the Project File".

Setting up the client can be slightly different according to its type. FactoryStudio supports the following client technologies:

- Windows rich clients
- Windows smart clients
- Windows web-based clients
- HTML5 clients
- iOS clients

For more on these client types, see "Remote Client Users Setup" below.

If you need to use Active-X, COM, and JavaScript, you can access the runtime application using the DataAccess API. This is a COM interface that provides integration with Active-X, JavaScript on web pages, or legacy programming languages such as VBScript. Contact support for assistance.

For redundant scenarios, see "Deploying Redundant Systems" below.

---

## Product Installation on the Target Computer

A FactoryStudio license must be installed on a standalone computer or on the server computer in distributed systems. The section "Installing FactoryStudio" in the "Getting Started" guide describes the standard way to install and license FactoryStudio. All servers and clients require the 4.0 or higher version of the .NET framework.

If you use components that do not require the setup of the Windows Registry, copy the product files without running an installation. This is especially useful for embedding the software on devices. If needed, you may remove the project examples or engineering components and add custom protocols.

In distributed systems, the client computers do not require a license. They only need to connect to the server. For more information on the client setup, see "Remote Client Users Setup", below.

## License and Project Settings Verification

A FactoryStudio project is created targeting one specific Product Family and Product Model, as defined on the **Info > Project > Settings** page.

It is necessary to ensure that the license on the server computer is greater than or equal to the requirements of the project. Review the following checklist to ensure your license can handle your project.

- The **Family** defined in the license must be the same as the family in the project or a higher hierarchy family.
- **Enterprise** licenses can run all projects (Enterprise, HMI and OPCServer)
- **HMI** licenses can run HMI and OPC Server projects
- **OPCServer** licenses can run only OPC server projects
- **Express** licenses or projects are not authorized for production environments
- The **Model** in the license must support a number of communication points equal to or greater than the requirements of the project that will run on the server computer.

TWebServer or IIS must be installed and running on the server computer if the system will have remote users.

- The **License Type** on the target computer must be **Engineering** or **Runtime**. Licenses of the DEV (Development) type are only for the internal work of System Integrators and may not be used in production environments.
- If the project requires any additional interfaces, such as the OSIsoft<sup>™</sup> PI System or IEC protocols, the license on the target computer must be enabled for those interfaces.
- The number of enabled remote clients must meet the project requirements.

For more Information about product and license models, refer to "[Version s and Licenses](#)".

---

## Installing the Project File

A project is installed on the server as a single file, either the main configuration file (with the ".tproj" extension ) or a read-only file (with the ".trun" extension) if you have created one.

The Project Management utility allows you to connect with remote servers and download the project file to remote computers.

Although one file contains the entire project configuration, you should use the following checklist to ensure that any external dependencies are also taken care of.

- If the folder structure is different on the production computer and the computer used for your project's development, make sure that all of your project's file paths are correctly mapped to the production computer. FactoryStudio has many features and macros to define paths relative to the project location, product installation, or execution path. Whenever possible, avoid using fixed path locations in your projects.
- Any external WPF controls should also be copied to the target computer. For remote web access, these files should be located in the WpfControl folder and the utility that updates the web manifest must be executed.
- If the application references external DLL or .NET assemblies, ensure they are available and at the correct paths on the target computer.
- If the project uses Retentive values, you must decide if the target computer will create a new Retentive database or if you will copy one with predefined values.
- Enable the firewall to allow remote clients. Ports 3101 for startup (optionally port 3201 for Test Mode). For web and iOS clients, data web services are enabled on port 80.
- If the application uses external DLLs, WpfControls, configuration files, or embedded databases, make sure you copy the files to the target machine and double check if the PATHNAMES you used in the project configuration are compatible.

---

## Deploying Redundant Systems

You can configure application redundancy by configuring two computers to be used as servers. One computer will be the primary server and the other will be the secondary or hot standby. If the primary computer or the connection to the computer fails, the system automatically fails over to the secondary computer.

If you selected HMI as the Product Family, the redundancy configuration is not available.

**To configure redundancy:**

- Go to **Info >Project > Redundancy**.
- Enter or select the information, as needed.

Field	Description
Enable Configuration	Select to enable the redundancy configuration.
Primary Server IP and Port	Enter the IP address and port of the primary server.
Secondary Server IP and Port	Enter the IP address and port of the secondary server.
On Primary Startup	Select the option you want.
Historian Replication	Select how to handle historian replication.
Connection Timeout	Connection timeout time, in seconds. If reached, this will cause the system to switch to the secondary server.
Server Command Line	Read-only field populated based on the fields above. Click Copy to Clipboard to copy the command for use.
Rich Client Command	Read-only field populated based on the fields above. Click Copy to copy the command for use.
Smart Client URL	Read-only field populated based on the fields above. Click Copy to copy the command for use.

Web Client URL	Read-only field populated based on the fields above. Click Copy to copy the command for use.
----------------	--

**To automatically start the application on a Windows client:**

- Go to **Info > Project > Redundancy**.
- Enter or select the information, as needed.

Field	Description
Enable Configuration	Select to enable the configuration.
Primary Server IP and Port	Enter the IP address and port of the primary server.
Secondary Server IP and Port	Enter the IP address and port of the secondary server, if any.
On Primary Startup	Select the option you want.
Historian Replication	Select how to handle historian replication.
Connection Timeout	Connection timeout time, in seconds, to switch to secondary server.
Rich Client Command	Read-only field populated based on the fields above. Click <b>Copy</b> to copy the command for use.
Smart Client URL	Read-only field populated based on the fields above. Click <b>Copy</b> to copy the command for use.
Web Client URL	Read-only field populated based on the fields above. Click <b>Copy</b> to copy the command for use.

- Click the **Copy** button next to the type of the client you want to start automatically.