

Getting Started

The Ignition Platform

The "Ignition Platform", as we refer to it frequently throughout this document, is the base of the Ignition product and provides the framework upon which modules, including your own, can be built. When we refer to the platform, we are referring primarily to the publicly accessible functions that can be used by modules. Occasionally, though, we simply use it as a convenient way to describe Ignition independently from any particular modules.

The various sections and chapters of this guide outline the functionality provided by the platform, first at a high level, and then in further depth. Despite the fact that we refer to the Ignition Platform as a single unit, there are really three primary components to it. We call these components the *scopes* of the platform. The first two: the **gateway** and the **designer** are the most substantial. The third, **client**, scope is important in displaying information, but has a smaller architectural footprint. The structure of this document clearly reflects this main division, with separate chapters for programming for the gateway and for the designer. Understanding this structure will help you to better understand the modular layout of the SDK, with libraries generally split between Gateway, Designer, and Client. Some parts of the system apply to all three scopes, and we typically refer to these parts as the **common** dependencies.

Upon these foundations, we have additional Application Programming Interfaces (APIs) to target things like our Alarm Notifications system and Sequential Function Charts.

The role of the SDK is to provide access to the platform through these APIs. In simple terms, the API consists of code that is both understood by Ignition, and public for use in your code. It is the "common layer" between what you write and the private inner workings of the Ignition system. Through the course of this document, you'll learn about various interfaces and classes available in the API, and how you can use them to build your own custom modules.

Peeking at the Architecture

The following diagram gives a visual overview to some of the services and managers available in the gateway. Through this guide, we'll reference various aspects of this architecture to help you make the most of your module.

