



## Overview

The WASP3D SDK helps developers create custom applications for broadcast & interactive graphics. Live event graphics developers can use the API to data-bind WASP3D scenes to various sources of data, scoring applications, wire feeds and databases in order to send it to the WASP3D rendering engine - Sting Server for on-air playout. The WASP3D SDK consists of the following APIs:

### Scene API



The scene API helps developers control the WASP3D scene and its elements using their code. The behavior of any WASP3D scene and the objects and elements of the scene can be altered based on a frame or event triggered during playout.

### Playlist API



With the playlist API, developers can create custom WASP3D rundowns (playlists) that can be hosted in the WASP3D Sting Client. Integrating third party applications within the Sting Client helps developers to provide end users a unified experience without the need to switch between applications for different tasks.

### Shotbox API



The WASP3D Shotbox is a simple API that gives users a high level control over the Sting Server. The API provides classes for developers to load and meld data with WASP3D design templates on Sting Server. The API wraps the underlying protocol for communication through helper functions, enabling developers to create custom solutions for broadcast & interactive graphics. Multiple Sting Servers can be connected using different communication channels and applications can be created, from simple to complex, in managing multiple scenes on multiple Sting Servers from a single application using the WASP3D Shotbox API. Shotbox features include:

- Connect or disconnect from a Sting server on a WASP3D network.
- Load and unload WASP3D scenes on the Sting Server.
- Play and pause WASP3D scenes on the Sting Server.
- Update data of WASP3D scenes at run time on the Sting Server, ensuring on-air data changes.
- Receive update/acknowledgement information from the Sting Server.

### Workflow API



Using the Workflow API developers can create and manage WASP3D assets, rundowns (playlists), and programs (multiple rundowns z-layered and centrally controlled). Developers can populate WASP3D templates with data and add them as a WASP3D instances (items) in a rundown. Third party software that is non-MOS compliant can also use the Workflow API to create/populate/update the WASP3D rundowns.

### Automation API



The Automation API allows developers control over the WASP3D Sting Server (playout render engine) through TCP/IP based commands. Third party systems like automation systems can use the Automation API to load/control WASP3D scene on the Sting Server. Developers can also build html5 based applications using the built-in web socket support to control the playout of WASP3D templates.

## Add-In API



Developers can use the Add-In API to provide their design team (using Drone Designer) a "drag & drop" interface of data sets in the WASP3D templates. Reusing business logic components as add-ins simplifies the data-binding process for complex logic driven templates. Developers can also limit the scope of the add-in built to a scene level or a Sting Server level.