

DSix Scripting

About DSix Scripting

Description

This section introduces you to automating tasks in DSix with scripting commands. We will discuss the available scripting languages and how to install and select them. We will then cover some basic operations from the command line, such as setting variables and executing the simulation. We will then proceed to more complex operations by writing and executing script files. We will discuss startup scripts, and briefly cover setting a script to execute in response to an event, such as a button press or reaching a minimum altitude.

What you will learn

Upon completion of this section, you will be able to:

- Select a preferred **script language**
- **Execute scripts** and script files from the command line
- Set **startup scripts** for DSix or individual projects
- Set scripts to execute in response to **simulation events**

About DSix Scripting

DSix supports scripting services through the Microsoft Active Scripting standard. Active Scripting provides the link between DSix and a scripting engine, such as VBScript or Jscript.

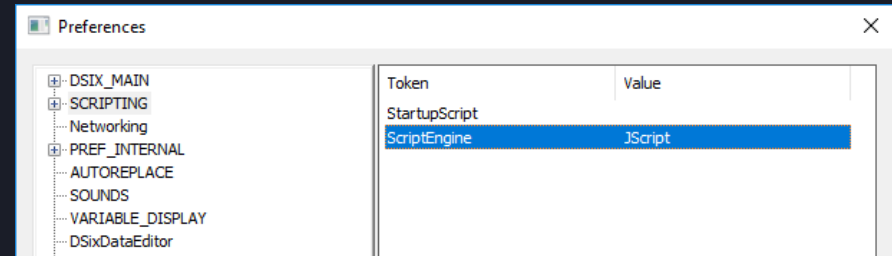
By default, DSix loads the **Jscript** engine which is similar in syntax to Java or C

DSix Scripting (General Information)

Supported Script Engines:

Jscript

VBScript



Supported Script Locations (*Search Path*):

Any Fully-Qualified Path

<Project Directory>\Scripts

<DSix Directory>\Scripts

DSix Stock Script Objects:

ModData

BARSettings

DSix Plots

Running Script Commands/Script Files

- ✈ DSix Script Command Window
- ✈ Automated Startup Scripts
- ✈ I/O Mapped Scripting
- ✈ Program Events

DSix Script Command Window

The DSix Script Command window is located at the bottom of the main DSix window. Commands may be entered manually, or by using the up and down arrows (↑ ↓) to scroll through recent script history.

Example:

Please see the file LoadBARJet.js in the DSix20\Scripts directory

Automated Startup Scripts

DSix provides a mechanism for the automated execution of script files upon application startup, and/or project load.

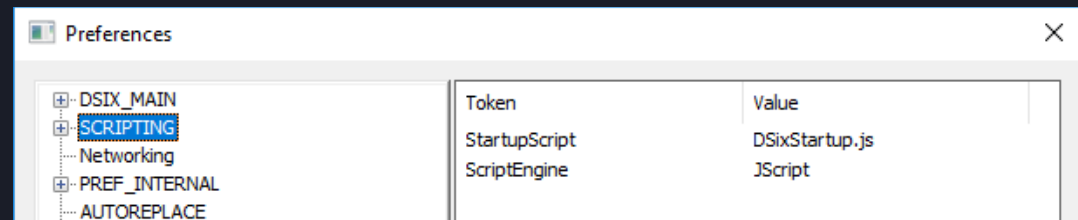
Startup Scripts:

- DSix Startup Script

- Project Startup Script

Automated Startup Scripts (DSix Application)

- ✈ Default Startup Script Location
<DSix Root>\Scripts
- ✈ Default Startup Script Name
DSixStartup.js or DSixStartup.vb
- ✈ Editable From the DSix Preference Editor



Example:

C:\ProgramFiles...DSix20\Scripts\DSixStartup.js

Automated Startup Scripts (Project Startup)

- Location
 <Project_TGT>\Scripts
- File Name
 <ProjectName>.js

Example: "BARJet.js"

C:\...\BARJet_TGT\Scripts\BARJet.js

or

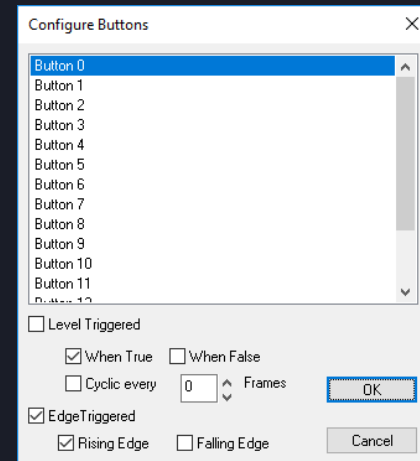
C:\...\BARJet_TGT\Scripts\BARJet.vbs

I/O Mapped Scripting

Scripts are included among the Event Types in I/O Devices.

Just as a defined I/O event may be mapped to Graphics, Sound, or Program events, I/O events may also be mapped to script files or commands.

Example: Tools .. I/O Devices .. Events Tab.
 Script, Read from File,
 Navigate to TriggerScript.js
 T.16000M::Buttons: Button 0
 Edge Triggered Falling



DSix Program Events

DSix Scripts (files or individual commands) may be set to execute when certain events occur in DSix, such as when a project is loaded, when the sim starts, steps or stops, or when saving or loading a project.

Available scripting events are defined as follows:

PRERESET

POSTRESET

PRESTEP

POSTSTEP

PROJECTSAVE

POSTRUN

SAFEMODEON

SAFEMODEOFF

IDLE

PROJECTLOAD

PROJECTCLOSING

DSIXCLOSING

DSix Program Event (Syntax)

- ✈ **SetEventString (*EventType*, *ScriptString*)**
Sets a script command to be executed in response to the indicated event.
- ✈ **SetEventFile (*EventType*, *ScriptFile*)**
Sets a script file to be executed in response to the indicated event.
- ✈ **SetEventString/File (*EventType*, "")**
An empty string argument will clear the script command/file
- ✈ **ClearEvents()**
Clears all event commands/files

DSix Program Event Example

File: Rename SetupPlotScript.js as the Project Startup
Script: BARJet.js

Responds to the events:

PRERESET with clearing the print window

POSTRUN with a PlotScript.js

Additional Script Objects

BARSettings

Provides access to the BARSettings File functionality

Example: `ExampleOfAvailableCommands.js`

Plots

Provides scripting support for the DSix Plots

Example: `PlotScript.js`



Flight Simulation Environment