AutomationDesk

Accessing MotionDesk

For AutomationDesk 6.5

Release 2021-A – May 2021

dSPACE

How to Contact dSPACE

Mail:	dSPACE GmbH Rathenaustraße 26 33102 Paderborn Germany
Tel.:	+49 5251 1638-0
Fax:	+49 5251 16198-0
E-mail:	info@dspace.de
Web:	http://www.dspace.com

How to Contact dSPACE Support

If you encounter a problem when using dSPACE products, contact your local dSPACE representative:

- Local dSPACE companies and distributors: http://www.dspace.com/go/locations
- For countries not listed, contact dSPACE GmbH in Paderborn, Germany.
  Tel.: +49 5251 1638-941 or e-mail: support@dspace.de

You can also use the support request form: http://www.dspace.com/go/supportrequest. If you are logged on to mydSPACE, you are automatically identified and do not need to add your contact details manually.

If possible, always provide the relevant dSPACE License ID or the serial number of the CmContainer in your support request.

Software Updates and Patches

dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit http://www.dspace.com/go/patches for software updates and patches.

Important Notice

Contents

About This Document

Content

This document gives you information on how to access MotionDesk via AutomationDesk.

Required knowledge

Working with AutomationDesk requires:

- Basic knowledge in handling the PC and the Microsoft Windows operating system.
- Basic knowledge in developing applications or tests.
- Basic knowledge in handling the external device, which you control remotely via AutomationDesk.

dSPACE provides trainings for AutomationDesk. For more information, refer to https://www.dspace.com/go/trainings.

Symbols

dSPACE user documentation uses the following symbols:

Symbol	Description
DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazard that, if not avoided, could result in property damage.
Note	Indicates important information that you should take into account to avoid malfunctions.
Tip	Indicates tips that can make your work easier.
	Indicates a link that refers to a definition in the glossary, which you can find at the end of the document unless stated otherwise.

Symbol	Description
🕮	Precedes the document title in a link that refers to another document.

Naming conventions

dSPACE user documentation uses the following naming conventions:

%name% Names enclosed in percent signs refer to environment variables for file and path names.

< > Angle brackets contain wildcard characters or placeholders for variable file and path names, etc.

Special folders

Some software products use the following special folders:

Common Program Data folder A standard folder for application-specific configuration data that is used by all users.

```
%PROGRAMDATA%\dSPACE\<InstallationGUID>\<ProductName>
```

or

```
%PROGRAMDATA%\dSPACE\<ProductName>\<VersionNumber>
```

Documents folder A standard folder for user-specific documents.

```
%USERPROFILE%\Documents\dSPACE\<ProductName>\
<VersionNumber>
```

Local Program Data folder A standard folder for application-specific configuration data that is used by the current, non-roaming user.

```
%USERPROFILE%\AppData\Local\dSPACE\<InstallationGUID>\
<ProductName>
```

Accessing dSPACE Help and PDF Files

After you install and decrypt dSPACE software, the documentation for the installed products is available in dSPACE Help and as PDF files.

dSPACE Help (local) You can open your local installation of dSPACE Help:

- On its home page via Windows Start Menu
- On specific content using context-sensitive help via **F1**

dSPACE Help (Web) You can access the Web version of dSPACE Help at www.dspace.com/go/help.

To access the Web version, you must have a *mydSPACE* account.

PDF files You can access PDF files via the icon in dSPACE Help. The PDF opens on the first page.

Basics and Instructions

Where to go from here

Information in this section

Basics on MotionDesk

Introduction

General information on MotionDesk.

Features of MotionDesk

MotionDesk is a dSPACE software for visualizing the movement of mechanical objects, such as car components or robot arms, in a virtual 3-D world.

It contains graphical tools for arranging static and movable objects in animatable scenes.

The movement of objects during the animation can be controlled online by a running simulation application or offline by reading motion data from a motion data file (MDF).

MotionDesk elements

Project You can group visualization tasks that belong together in a MotionDesk project.

Experiment This is the basis for carrying out an animation of one specific scene. To experiment with another scene, you have to add a second experiment to the project.

Scene The scene of an experiment specifies the static and movable objects that exist in the experiment's virtual world.

You can add 3-D objects to a scene from a library and parameterize them via their properties. Via the **Data Stream** property, you can connect a movable object to a data stream.

Additionally, you can add observers to a scene, each one specifying a certain viewpoint within the scene. You select up to four viewpoints to be displayed in parallel during the animation of the scene.

Activating MotionDesk experiments

Although a MotionDesk project can contain several experiments, you can only work with one experiment at a time. To select an experiment for further processing, you have to activate it.

Automating MotionDesk

You can write scripts to control MotionDesk via a COM-based automation interface. For more details, refer to MotionDesk Automation 🕮. AutomationDesk's **MotionDesk Access** library uses this interface to provide automation blocks for some basic MotionDesk features. The blocks offer only the subset of the automation interface that is necessary to animate an experiment's scene.

Demo projects

For an example of a MotionDesk project, refer to `<RCP_HIL_InstallationPath>\Demos\MotionDesk`.

Related topics

Basics

MotionDesk Automation
Introduction to MotionDesk (MotionDesk Basics 🕮)

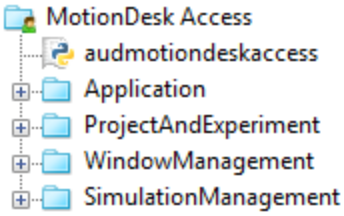
Overview of the MotionDesk Access Library

Introduction

The **MotionDesk Access** library is implemented as a custom library. This means that you can manage it like any other custom library. For example, you can open, close, export and import it. You can read the library's Python source code in the **audmotiondeskaccess** Python module at the library's top level.

Library overview

The blocks of the library are grouped in folders which represent the object that the blocks deal with.



Application This folder provides automation blocks to start and close MotionDesk. You can specify to save all project files when you close MotionDesk.

ProjectAndExperiment This folder provides automation blocks to open and close a MotionDesk project and to activate and close an experiment in it.

WindowManagement This folder provides an automation block to specify whether an animation is displayed in a window or in full screen.

SimulationManagement This folder provides automation blocks to control the animation of the activated experiment's scene.

For detailed information on the automation blocks, refer to Automation Blocks (AutomationDesk Basic Practices 🕮).

Access via Exec block

The `audmotiondeskaccess` Python module provides methods for the MotionDesk Access automation blocks. Each method has the same name as its block, its arguments correspond to the block's input data objects, and its return value corresponds to the output data object.

Example

```
import audmotiondeskaccess
audmotiondeskaccess.OpenProjectAndExperiment( \
    ProjectFile, "Experiment_002", SaveActiveProject="True")
```

Demo projects

For an example of automating MotionDesk access, refer to the AutomationDesk demo project at `<DocumentsFolder>\MotionDesk Access`.

Related topics

HowTos

How to Animate a MotionDesk Scene 10

References

Automation Blocks (AutomationDesk Basic Practices 🕮)

How to Animate a MotionDesk Scene

Objective

You can animate the scene of a MotionDesk experiment.

Generic steps

MotionDesk must be opened and an experiment must be activated.

Then the contained scene can be animated. The movement of its objects can be controlled offline by reading a motion data file (MDF) or online by a running simulation.

After the use case is finished, MotionDesk must be closed.

Preconditions

- MotionDesk must be installed on the host PC. You need the same licenses as to execute the tasks manually with MotionDesk.
- The MotionDesk experiment must be completely specified, including the scene, views and selected data source, so that it is ready to be animated.
- If the MotionDesk experiment is configured to get the motion data from a running simulation, the related simulation application must be loaded to a registered platform and started. You can implement this, for example, by using automation blocks from the **XIL API Convenience** library. For instructions, refer to How to Download and Start a Simulation Application (AutomationDesk Accessing Simulation Platforms 🕮).
- If the MotionDesk experiment is configured to get the motion data from an MDF file, this file must exist.
- The following information is required as input data:
  - The name and path of the MotionDesk project file (CDP) you want to open
  - The name of the MotionDesk experiment you want to access

Method

To animate a MotionDesk scene

1. Add the following data objects to your project to parameterize the input data:
   - A **File** data object

     In the Data Object Editor, parameterize the **File** data object with the file name and path of the project file (CDP) of the MotionDesk project you want to open.
   - A **String** data object

     Parameterize the **String** data object with the name of the experiment you want to activate.
2. From the Library Browser, drag a **StartMotionDesk** block from the **MotionDesk Access** library to the Sequence Builder to get an instance of MotionDesk. If MotionDesk is already running, the existing instance is used.

   By default, the block's **Visible** data object is set to **`True`**. This causes the graphical user interface of MotionDesk to be displayed.

3 Drag an **OpenProjectAndExperiment** block to your sequence. This opens the specified project, activates the specified experiment and displays the unanimated experiment's scene.

4 In the Data Object Editor, set the block's **ProjectFile** and **InitialExperiment** data objects as references to the project-specific File and String data objects that contain the MotionDesk project file and the experiment name.

5 Drag a **StartAnimation** block to your sequence. This starts displaying the scene that is contained in the currently activated experiment.

6 Add the blocks to be executed during the animation, i.e., your use case, to the sequence.

7 Drag a **StopAnimation** block to your sequence. This stops the animation.

8 Drag a **CloseProjectAndExperiment** block to your sequence. This closes the active experiment and the MotionDesk project. The display of the scene ends.

9 Drag a **CloseMotionDesk** block to your sequence. This closes the existing MotionDesk instance.

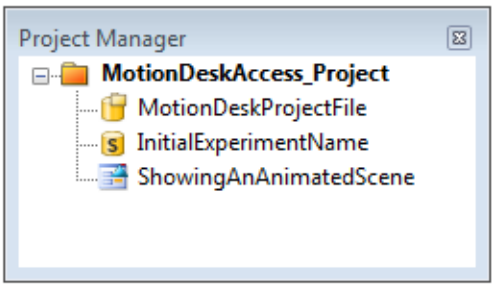
Note

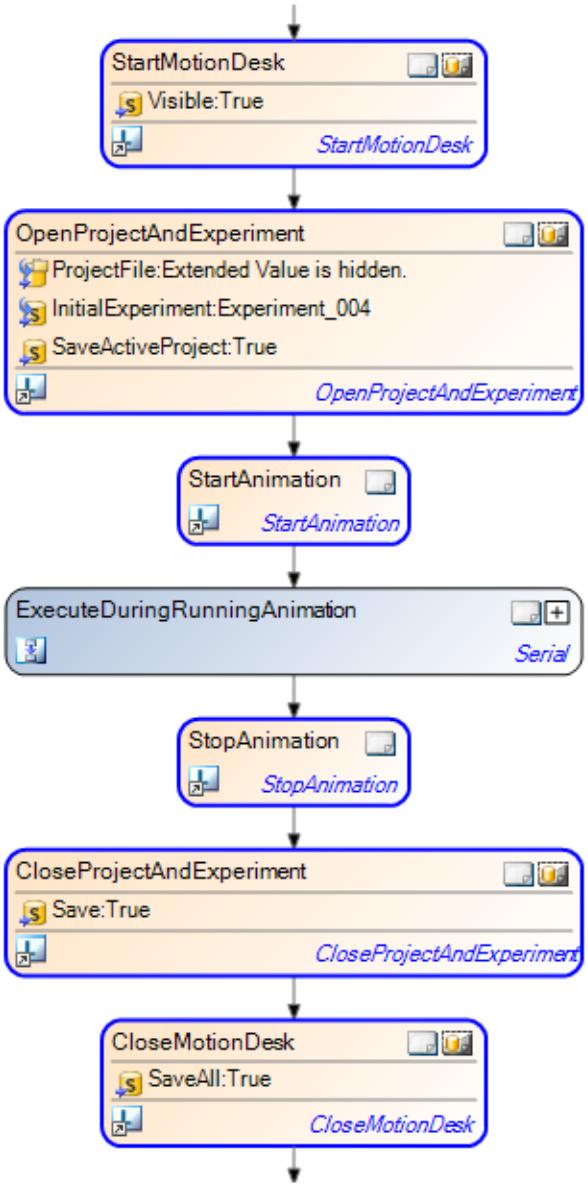
This will also close a MotionDesk instance that you started manually.

Result

You created a sequence to animate the scene of a MotionDesk experiment.

The resulting sequence looks like this:





When you run the sequence, a MotionDesk instance is created or accessed to open the specified project and to activate the specified experiment. Then the contained scene is animated as specified in the experiment.

In this example, the ExecuteDuringAnimation block represents your use case. After this block is executed the animation is stopped. MotionDesk access is finished by closing the project with its experiment and by closing MotionDesk.

Tip

In this example, the duration of the animation is determinated by the time that is needed to process the **ExecuteDuringAnimation** block. The duration can be controlled in the following ways:

- For a fixed duration of the animation, you can add a **Sleep** block to the **ExecuteDuringAnimation** block and specify the duration in the **Sleep** block's **Time** data object.
- For a duration of the animation that depends on variables of the running simulation application, you can use the blocks of the **XIL API Convenience** library to read the variable values. For more information, refer to Accessing Simulation Platforms via the XIL API Convenience Library (AutomationDesk Accessing Simulation Platforms 🕮).
- For a duration of the animation that depends on parameters of Automotive Simulation Models (ASM) of the running simulation application, you can use the blocks of the **ModelDesk Access** library to use the variable values. For more information, refer to AutomationDesk Accessing ModelDesk 🕮.

Related topics

Basics

Basics on MotionDesk 7
Overview of the MotionDesk Access Library 8

References

CloseMotionDesk 18
CloseProjectAndExperiment 20
OpenProjectAndExperiment 19
StartAnimation 22
StartMotionDesk 17
StopAnimation 23

Reference Information

Automation Blocks

Introduction

The Signal-MotionDesk Access library is a custom library. It is write-protected to prevent modifications to its blocks.

Using MotionDesk Access library features in Python scripts

You can use functions and other definitions of the MotionDesk Access library in Python scripts after you imported the `audmotiondeskaccess` module to the current namespace.

Where to go from here

Information in this section

Application

Introduction

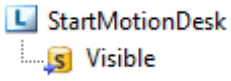
The Application folder in the MotionDesk Access library provides blocks to access the MotionDesk application.

Where to go from here

Information in this section

StartMotionDesk

Graphical representation



Purpose

To start MotionDesk.

Description

This block creates an instance of MotionDesk. If a MotionDesk process is already running, the existing process is used. You can specify to open MotionDesk in visible or invisible mode.

Note

If MotionDesk is not available, for example, if it is not installed, AutomationDesk throws an exception.

Data objects

This automation block provides the following data object:

Name	In / Out	Data Type	Default Value	Description
Visible	In	String	"True"	Lets you specify the visible mode of the application: ▪ True MotionDesk starts with the user interface displayed. ▪ False MotionDesk starts in hidden mode.

Related topics

HowTos

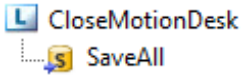
How to Animate a MotionDesk Scene 10

References

CloseMotionDesk 18

CloseMotionDesk

Graphical representation



Purpose

To close MotionDesk.

Description

This block exits MotionDesk. You can specify whether to save the modifications made during automated access. If MotionDesk is already closed, AutomationDesk starts the MotionDesk application in invisible mode to close it correctly afterwards.

This block also closes an MotionDesk instance, if it was opened manually.

Data objects

This automation block provides the following data object:

Name	In / Out	Data Type	Default Value	Description
SaveAll	In	String	"True"	Lets you specify whether to save your modifications: ▪ True Modifications are saved before closing MotionDesk. ▪ False Modifications are discarded without confirmation.

Related topics

HowTos

How to Animate a MotionDesk Scene 10

References

StartMotionDesk 17

ProjectAndExperiment

Introduction

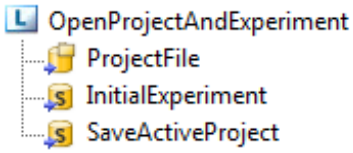
The ProjectAndExperiment folder in the MotionDesk Access library provides blocks to access MotionDesk's projects and experiments.

Where to go from here

Information in this section

OpenProjectAndExperiment

Graphical representation



Purpose

To load a project and activate an experiment.

Description

This block loads the specified project. If the project contains several experiments, you can specify which experiment is to be activated. Because you can load only one project at the same time in MotionDesk, any already loaded project is closed, even if it is identical to the one to be started. You can specify whether to save the project before it is closed.

If you have not used the **StartMotionDesk** block beforehand, MotionDesk is automatically started in invisible mode.

If the specified project file is not available, AutomationDesk throws an exception.

Data objects

This automation block provides the following data objects:

Name	In / Out	Data Type	Default Value	Description
ProjectFile	In	File	" "	Lets you specify the MotionDesk project file (CDP) to be loaded.
InitialExperiment	In	String	" "	Lets you optionally specify the experiment to be activated.
SaveActiveProject	In	String	"True"	Lets you specify whether to save an already loaded project if it differs from the specified one. ▪ True Modifications in the already loaded project are saved.

Name	In / Out	Data Type	Default Value	Description
				▪ False Modifications in the already loaded project are discarded.

Related topics

HowTos

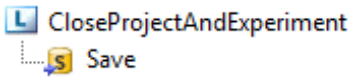
How to Animate a MotionDesk Scene 10

References

CloseProjectAndExperiment 20

CloseProjectAndExperiment

Graphical representation



Purpose

To close a MotionDesk project.

Description

This block closes the active MotionDesk project. You can specify whether to save modifications in the project before closing it. If the project is already closed, the block executes with no action.

Data objects

This automation block provides the following data object:

Name	In / Out	Data Type	Default Value	Description
Save	In	String	"True"	Lets you specify whether to save modifications in the active experiment and project before closing it. ▪ True Modifications are saved. ▪ False Modifications are discarded.

Related topics

HowTos

How to Animate a MotionDesk Scene..... 10

References

OpenProjectAndExperiment..... 19

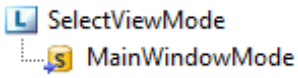
WindowManagement

Introduction

The WindowManagement folder in the MotionDesk Access library provides blocks to control the appearance of MotionDesk's animations.

SelectViewMode

Graphical representation



Purpose

To select the size of the displayed animation.

Description

This block lets you specify whether an animation is displayed in a window or in full screen.

Before you execute this block, the related project and experiment must be activated by using the OpenProjectAndExperiment block and the experiment's scenes must be loaded.

Data objects

This automation block provides the following data object:

Name	In / Out	Data Type	Default Value	Description
MainWindowMode	In	String	"Normal"	Lets you select one following view modes: ▪ Normal Animations are displayed in a window. ▪ FullScreen Animations are displayed in full screen.

Related topics

References

OpenProjectAndExperiment....19
Sleep (AutomationDesk Basic Practices 🕮)
StartAnimation....22

SimulationManagement

Introduction

The SimulationManagement folder in the MotionDesk Access library provides blocks to control the execution of MotionDesk's animations.

Where to go from here

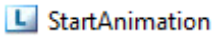
Information in this section

StartAnimation....22
To start an animation.

StopAnimation....23
To stop a running animation.

StartAnimation

Graphical representation



Purpose

To start an animation.

Description

This block starts the animation of the experiment that you opened before by using the OpenProjectAndExperiment block.

Data objects

None

Related topics

HowTos

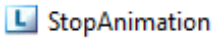
How to Animate a MotionDesk Scene..... 10

References

StopAnimation..... 23

StopAnimation

Graphical representation



Purpose

To stop a running animation.

Description

This block stops an animation that you started by using the StartAnimation block.

Data objects

None

Related topics

HowTos

How to Animate a MotionDesk Scene..... 10

References

StartAnimation..... 22

Automation

Basics on Automating the Access to MotionDesk

Introduction

AutomationDesk provides a COM-based API to automate the handling of AutomationDesk.

Related information

The AutomationDesk COM API provides no specific objects for accessing MotionDesk. You can only use the basic automation features, such as executing a project via script.

For information on the available objects with their properties and methods, refer to Basic Interface (AutomationDesk Automation 🕮).

For basic information and instructions, refer to Basics and Instructions on page 7.

Limitations

Limitations When Using the MotionDesk Access Library

Required MotionDesk version	It is recommended to use MotionDesk 3.5 or later when you want to remote-control it via the MotionDesk Access library.

A

C

D

L

M

O

S