ControlDesk

Project and Experiment Management

For ControlDesk 7.4

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 introduces you to ControlDesk's project and experiment management.

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.
	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

Introduction to Project and Experiment Management

Introduction

An experiment is the basis for carrying out a specific experiment task. A project manages different experiments belonging together. The **Project** controlbar in ControlDesk reflects the hierarchical project structure.

Where to go from here

Information in this section

Basics on Projects and Experiments

Introduction

In ControlDesk, an experiment is the basis for an experiment task and a project is a container for experiments. You handle experiments and projects with ControlDesk's **Project** controlbar.

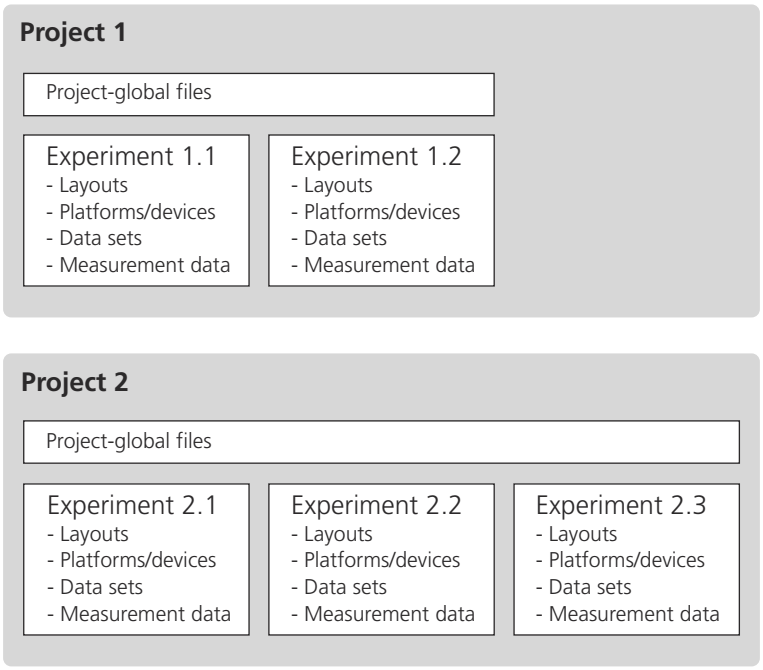
Experiment as the basis for an experiment task

An experiment is the basis for carrying out a specific experiment task, for example, the calibration of a cruise control. An experiment allows you to manage all the documents related to the task, such as

- Layouts – for visualizing variables
- Variable descriptions – specifying the variables of a platform/device
- Data sets – containing a set of ECU parameters
- measurement data files – containing the time traces of recorded variables

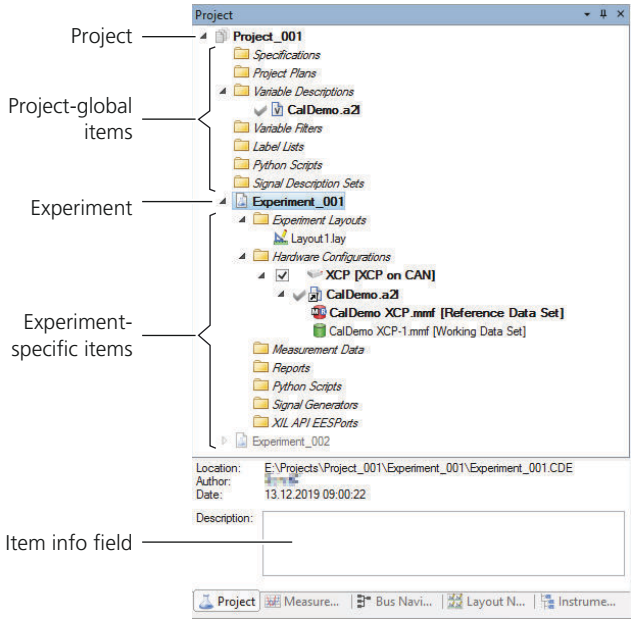
Project as the container for experiments

A project manages different experiments belonging together, such as the different tasks for calibrating a specific engine variant. It holds the experiments related to these tasks and items relevant for the entire project. Some of the project-global items, such as variable descriptions, are referenced in the experiments. Referencing global items allows you to reuse them in various experiments.



Handling projects with the Project controlbar

To handle projects and experiments, ControlDesk provides the Project controlbar. The Project controlbar gives you easy and intuitive access to all the experiments and project-global items of a project. The illustration below shows the Project controlbar with a project and the items managed by the project.



To handle projects, the Project controlbar provides a context menu with commands for copying, renaming, adding and removing items. The commands for handling individual items depend on the type of item.

Project root directories for grouping projects

Each ControlDesk project is related to a project root directory. This is a folder on your file system to which ControlDesk saves all project-relevant data, such as the experiments and documents of a project. Several projects can use the same project root directory.

Default project root directory ControlDesk will use the Documents folder as the default project root directory unless you specify a different one. ControlDesk saves all projects to the project root directory.

Specifying further project root directories You can specify further project root directories in addition to the default project root directory. This allows you to specify different destination directories for your projects, and to group projects. Refer to How to Specify a Project Root Directory on page 22.

Handling external documents in ControlDesk

Projects and experiments allow you to handle ControlDesk-specific documents such as layouts, data sets and measurement data files. You can also include external documents in a ControlDesk project. External documents are documents that cannot be modified with ControlDesk, such as DOC, PDF, XLS, or HTML files.

Accessing external documents via ControlDesk You cannot modify external documents via ControlDesk, but you can access them. Accessing external documents via ControlDesk means the following:

- External documents are displayed as items in ControlDesk's **Project** controlbar.
- You can open external documents in the corresponding programs from within ControlDesk. For example, ControlDesk opens Adobe® Reader® if you double-click a PDF file in the Project controlbar.

This allows you to include important documents such as project plans, specifications, and memos in a ControlDesk project or experiment, and access them from within ControlDesk.

Adding external documents To make external documents accessible via ControlDesk, you must add them to a ControlDesk project or experiment. For instructions, refer to How to Add External Documents to a Project or Experiment on page 29.

Related topics

Basics

Project Structure and File Storage 12

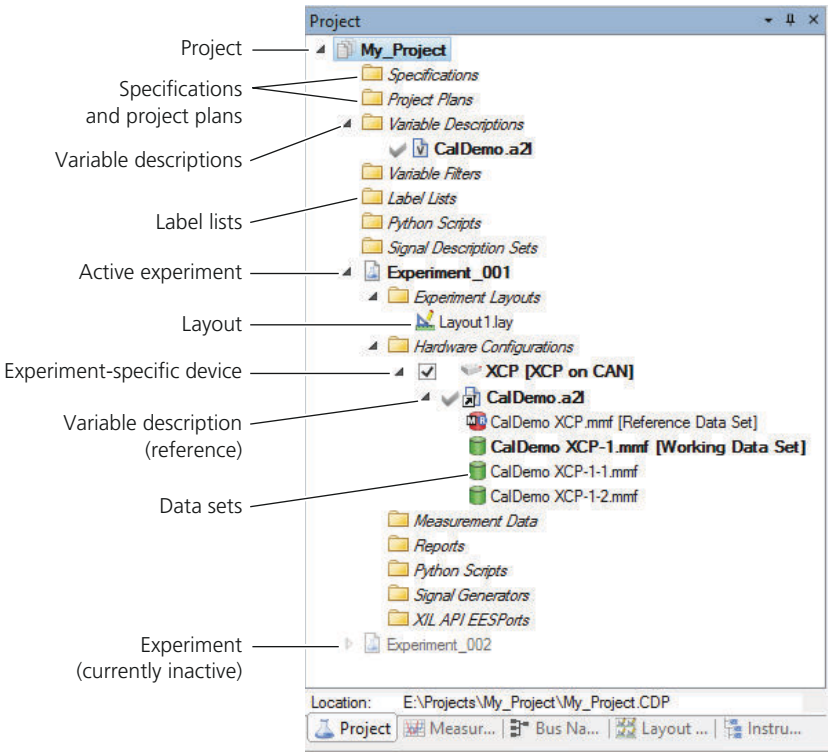
Project Structure and File Storage

Introduction

In ControlDesk, projects and experiments are structured hierarchically.

Hierarchical project structure

The structure displayed in ControlDesk's Project controlbar reflects the hierarchical project structure. The hierarchy helps you to organize experiment tasks.



A project contains:

Specifications and project plans Specifications and project plans can be used for collecting project-specific documents.

Variable descriptions The Variable Descriptions folder contains all the variable descriptions used in the experiments of a project. A variable description item in a specific experiment is always a reference to a variable description in this folder.

Label lists Label lists can be used for filter settings in variable lists.

Experiments A project contains one or more experiments. Only one experiment can be active at a time. An experiment contains:

- Layouts for visualizing variables
- Platforms and devices for interfacing, for example, dSPACE real-time hardware, VPUs, or ECUs
- References to variable descriptions specifying the variables of a platform/device
- Data sets containing a set of ECU parameters
- Measurement data files containing time traces of recorded variables

References and active items

Some of the experiment-specific items are references to project-global items. Reference items have an arrow in their symbol.

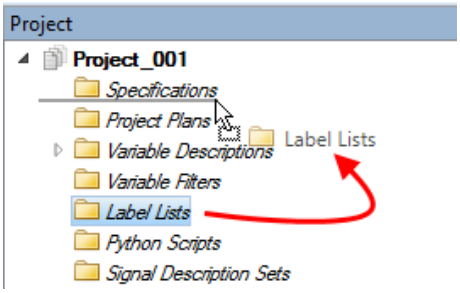
Symbol	Description
	Reference to a project global item (here: a variable description)

Items that are written in bold or marked by a checkmark are active.

Symbol	Description
Experiment_001	Active experiment
CalDemo.a2l	Reference to the active variable description

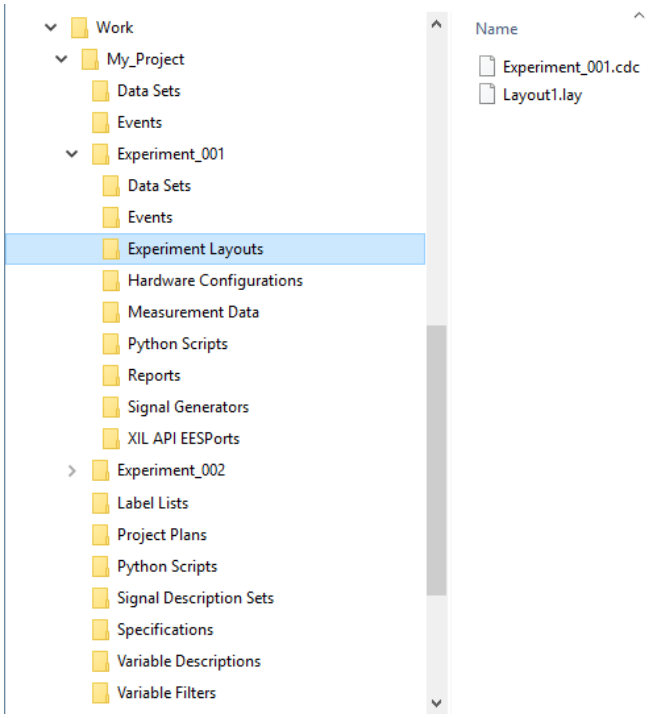
Sorting items

In the Project controlbar, you can sort items that are on the same hierarchy tree level by dragging them to new positions.



Filing according to the project structure

ControlDesk creates a folder structure for each project on your file system. The root of the folder structure is the project root directory. The project structure defines where ControlDesk stores the documents belonging to a project.



Tip

If you want to move a project to a different root directory, open the project. Then select File – Save As – Save Project As. A dialog opens, allowing you to save the project in a different root directory. Refer to Save as (Project/Experiment) on page 85.

Automatic file storage in component's folder

The components of ControlDesk such as the Data Set Management or Measurement Data Management "know" where to save files. ControlDesk automatically saves the files generated by these components to the corresponding folder. For example, when you create a layout, ControlDesk's Visualization component automatically saves it to the project's *Experiment Layouts* folder (see the illustration above). For this reason, you have neither to save files generated by ControlDesk to the file system manually, nor to specify file paths.

Tip

You can specify to save the project at regular intervals. If enabled, a self timer symbol is displayed next to the project. Refer to Project Page on page 72.

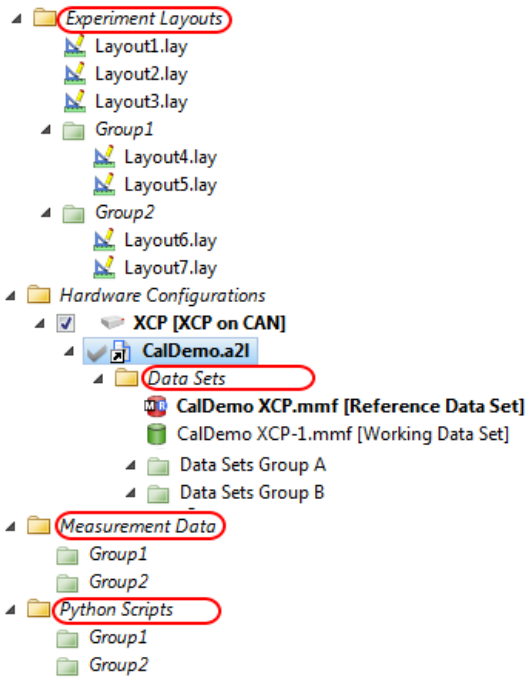
Adding folders and groups

Folders You can add new folders to the project structure and configure their settings. The added folders are stored in the file system. Refer to Add Folder on page 48.

Groups Groups are structuring elements that are only displayed in the project tree but not stored in the file system.

You can add groups to the following folders:

- Experiment Layouts
- Data Sets (data set grouping must be enabled)
- Measurement Data
- Python Scripts



Refer to Add Group on page 49.

Backup and export

You can back up projects and export experiments.

- Backing up a project allows you to save and transfer the whole content of a project in one ZIP file. Refer to Backup Project on page 50.
- Exporting an experiment allows you to save and transfer the content of an experiment in one DSA (dSPACE archive) file. Refer to Export (Experiment) on page 55.

XML project template

The structure of ControlDesk projects is based on the settings of the project template file `ControlDeskTemplate_eng.xml`. This predefines the structure of projects created with ControlDesk.

Default project template As long as you do not modify the project template, you work with the *default settings of the project template*. With the default settings, the structure of ControlDesk projects is as described in Hierarchical project structure on page 13.

Modifying the project template The ControlDesk Template Editor lets you modify the project template according to your needs. For details and instructions, refer to Modifying a Project Template on page 35.

Related topics

Basics

Basics on Projects and Experiments 10

Examples

Example of Project Structures 17

Example of Project Structures

Introduction

In ControlDesk, projects and experiments allow you to structure experiment tasks according to your needs. Below are two examples of applying projects and experiments in the calibration environment.

One ECU, different calibration stages

Imagine the following stages of calibrating a specific ECU:

- Calibration on the test bench
- Calibration in the test vehicle
- Calibration in the production vehicle

Common aspects of the calibration stages There is one specific ECU to be calibrated in the above calibration stages.

Differences in the calibration stages The above calibration stages usually are characterized by different ECU software revisions and layouts:

- Different ECU software revisions

  Suppliers of ECUs regularly provide new software revisions, that means, new code and data for an ECU. For this reason, ECU software revisions involved in different calibration stages differ.
- Different layouts

  Each calibration stage usually involves different parameters to be calibrated and variables to be measured. Layouts that visualize ECU parameters and measurement variables from different calibration stages therefore differ.

One ECU function, different ECUs

Imagine the calibration of a specific engine control function, for example, the cruise control. Engine control functions usually are implemented on different ECUs, each of which controls a certain engine variant, for example:

- ECU1 controlling a V6 gasoline engine.
- ECU2 controlling a V8 gasoline engine.
- ECU3 controlling a V8 diesel engine.

Common aspects of calibrating different ECUs Since the same control function is to be calibrated, similar parameters and measurement variables need to be visualized for different ECUs. Therefore, the same layouts may be reused for different ECUs.

Differences when calibrating different ECUs The calibration of different ECUs usually requires different calibration interfaces. For example, ECU1 may be calibrated via a memory emulator, whereas ECU2 and ECU3 are calibrated via a serial calibration protocol such as XCP. In other words, ECU interfaces for different ECUs often differ.

Related topics

Basics

Modifying a Project Template 35
Project Structure and File Storage 12

Defining Projects and Experiments

Where to go from here

Information in this section

How to Define a Project

Objective

The basis for whatever you do in ControlDesk is a project. A project manages different experiment tasks belonging together, for example, the different experiments to calibrate a specific engine variant. Defining a project therefore allows you to group ControlDesk experiments.

Preconditions

To define a project, the following preconditions apply:

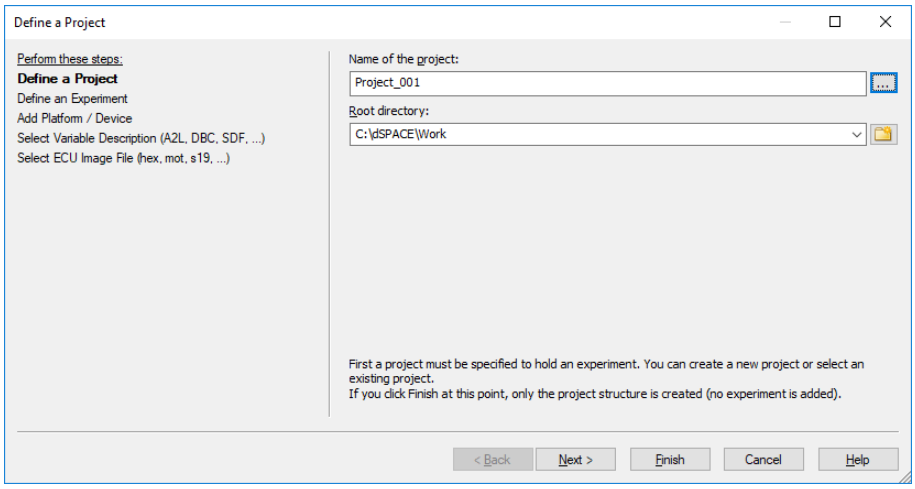
- At least one project root directory must be specified. For instructions, refer to How to Specify a Project Root Directory on page 22.
- No other project must be open.

Method

To define a project

1 On the File ribbon, click New – New Project + Experiment.
ControlDesk opens the Define a Project dialog.

2 In the **Name of the project** edit field, enter the project name.



3 From the **Root directory** edit list, select the desired project root directory.

If you want to specify a new project root directory, click  or edit the root directory directly in the edit field.

4 Click **Next>** or click **Finish**.

ControlDesk creates a project.

If you click **Next>**, ControlDesk opens the **Define an Experiment** dialog for you to define an experiment for the new project.

If you click **Finish**, ControlDesk creates an empty project, that is, a project without an experiment. You can define an experiment afterwards.

Result

You defined a new project.

Tip

- You can use ControlDesk events to create a project automatically when ControlDesk starts up. Refer to Creating a Project When ControlDesk Starts Up (ControlDesk Customization 🕮).
- If you want to move a project to a different root directory, open the project. Then select **File – Save As – Save Project As**. A dialog opens, allowing you to save the project in a different root directory. Refer to Save as (Project/Experiment) on page 85.

Next step

You can define an experiment for the new project. Refer to How to Define an Experiment on page 21.

Related topics

Basics

Introduction to Project and Experiment Management 10

References

Project Wizard 74

How to Define an Experiment

Objective

An experiment is the basis for carrying out a specific experiment task, for example the calibration of an idle speed control. An experiment allows you to manage all the files and data related to this task. You can organize your configuration tasks by defining experiments for them.

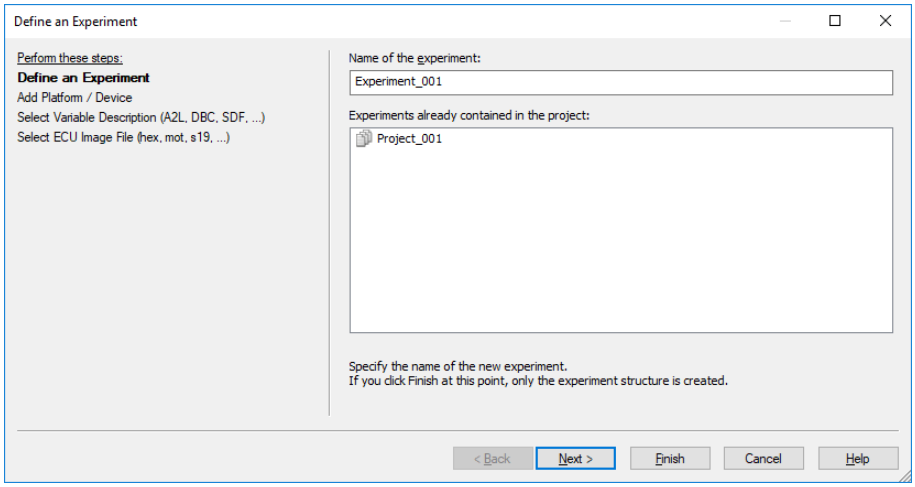
Preconditions

A project must be open. If no project is currently open, you have to open one (refer to Open Project + Experiment on page 64), or define a new project (refer to How to Define a Project on page 19).

Method

To define an experiment

1 On the File ribbon, click New – New Experiment.
ControlDesk opens the Define an Experiment dialog.



2 In the Name of the experiment edit field, enter the experiment name.

3 Click Next> or click Finish.
ControlDesk creates an experiment.

If you click **Next>**, ControlDesk opens the **Add Platform/Device** dialog that lets you add a platform/device to the new experiment.

If you click **Finish**, ControlDesk creates an empty experiment, that is, an experiment without a platform/device. You can add a platform/device afterwards.

Result

You defined an experiment.

Next step

You can add platforms/devices to the new experiment. Refer to:

- How to Add a Platform/Device to an Experiment (ControlDesk Platform Management 🕮)
- How to Add Multiple Platforms/Devices in SIL Testing Scenarios (ControlDesk Platform Management 🕮)

Related topics

Basics

Introduction to Project and Experiment Management 10

HowTos

How to Activate an Experiment 28
How to Add External Documents to a Project or Experiment 29

References

Project Wizard 74

How to Specify a Project Root Directory

Grouping projects in root folders

Each ControlDesk project is related to a project root directory. This is a physical directory on your file system. ControlDesk creates a folder structure beneath the project root directory and stores all the files of a project – layouts, data sets, etc. – to it. Specifying different project root directories therefore allows you to group ControlDesk projects on your file system.

Default project root directory

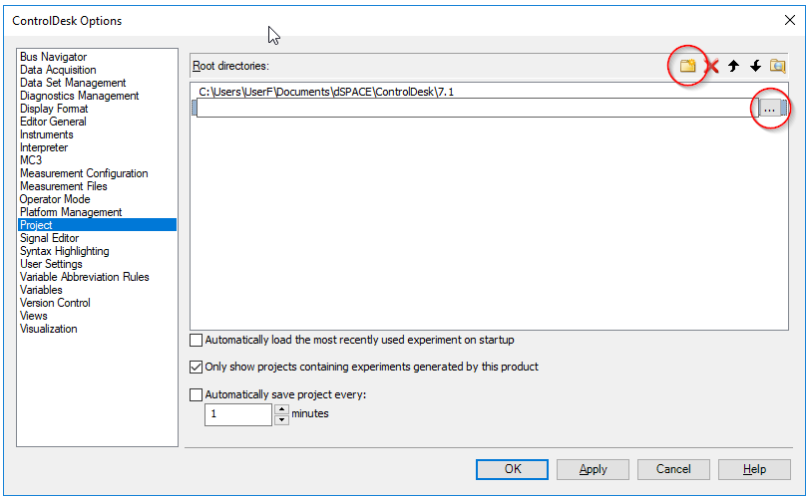
ControlDesk uses the Documents folder as the *default project root directory*.

Method

To specify a project root directory

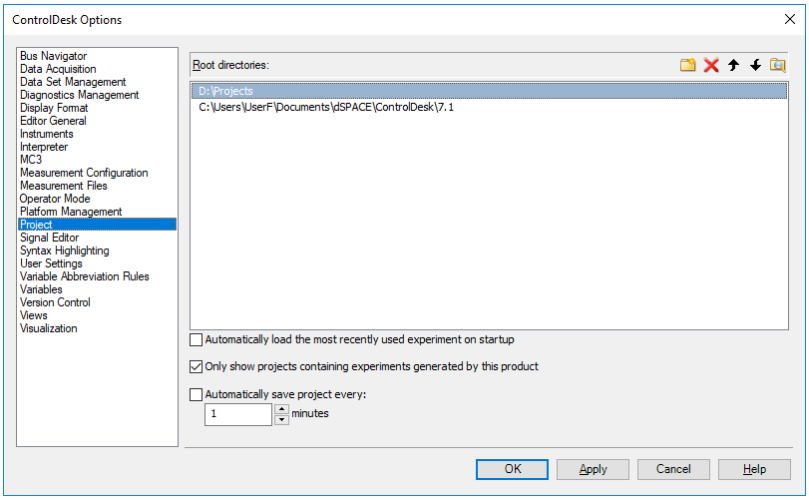
1 On the **File** ribbon, click **Options** to open the **ControlDesk Options** dialog.

2 On the Project page of the ControlDesk Options dialog, click , then click .



3 In the Browse for Folder dialog, select the new project root directory, or edit or paste the root directory directly in the edit field and click OK.

The ControlDesk Options dialog displays the new entry:



The new project root directory now is at the top of the list. Whenever you need to specify a project root directory, the entry at the top of this list is preselected.

4 If you want to change the preselection, select an entry in the list and click to move it to the top of the list.

5 Click OK to confirm your setting.

Result

You specified a new project root directory.

Tip

You can modify a project root directory by double-clicking and then editing its list entry in the ControlDesk Options dialog.

Related topics

Basics

Basics on Projects and Experiments..... 10

References

Project Page..... 72

Working with Projects and Experiments

Where to go from here

Information in this section

Information in other sections

How to Open a Project and Experiment

Objective

In ControlDesk, carrying out an experiment task is feasible only within an open experiment. You must therefore open an experiment and the project it belongs to before carrying out an experiment task.

Possible methods

Usually, you open projects and experiments that are stored in the project root directory. But you can also open projects/experiments that are stored outside of

it, for example, for quick access without changing the root directory or moving files to it.

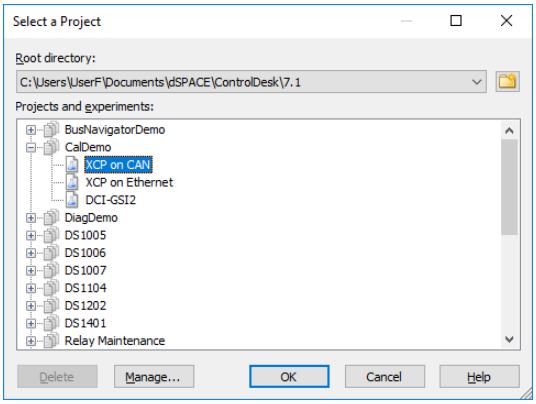
- To open a project/experiment that is stored in a project root directory, refer to Method 1.
- To open a project/experiment that is stored outside of the project root directory, refer to Method 2.

Method 1

To open a project and experiment that is stored in a project root directory

1 On the File ribbon, click Open – Open Project + Experiment, or press **Ctrl** + **Shift** + **O**.

ControlDesk opens the Select an Experiment dialog.



2 From the Root directory drop-down list, select the project root directory containing the project and experiment you want to open, or click  to define a new one.

3 In the Projects and experiments list, select the experiment you want to open.

4 Click OK.

ControlDesk

- Closes the current project,
- Opens the project of the experiment to be opened,
- Opens and activates the selected experiment.

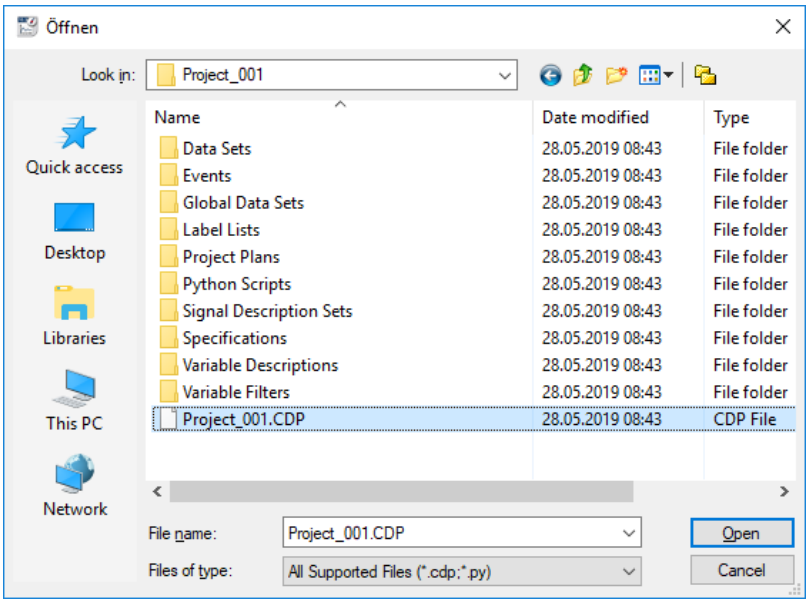
Method 2

To open a project and experiment that is stored outside of a project root directory

1 On the File ribbon, click Open – Open File or Project, or press **Ctrl** + **O**.

ControlDesk opens a standard Open dialog for you to browse for a project (CDP) file.

2 Select a CDP file and click Open.



ControlDesk opens the selected project. The path to the selected CDP file is not added to the list of project root directories.

3 Select an experiment and activate it via its context menu (see How to Activate an Experiment on page 28).

Result

You opened an experiment and the project it belongs to.

Tip

- Project overview: The fastest way to get an overview of the project structure is to select a project instead of an experiment. You can activate an experiment later on.
- Loading an experiment on startup: On the Project page of the ControlDesk Options dialog, you can activate Automatically load the most recently used experiment on startup. Remember that large projects will take a long time to load. Refer to Project Page on page 72.

Related topics

Basics

Introduction to Project and Experiment Management 10
Problem with Long Project Loading Times 100

References

Open Project + Experiment 64

How to Activate an Experiment

Objective

A ControlDesk project usually contains several experiments, each representing an entire experiment task. You can work with only one experiment – the *active* experiment – at a time. All the other experiments of the project are inactive. To work with another experiment of the project, you have to activate it first.

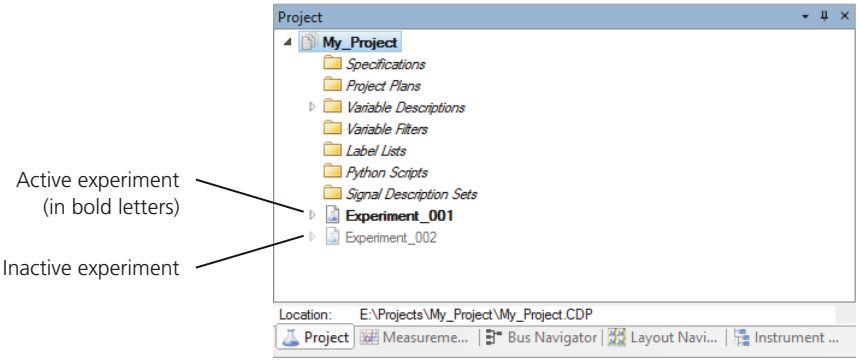
Restrictions

Activating an experiment is possible only within a currently open project. Refer to How to Open a Project and Experiment on page 25.

Method

To activate an experiment

1 In the Project controlbar, right-click the inactive experiment you want to activate.



2 From the context menu, select Activate.

ControlDesk activates the selected experiment, and deactivates the other.

Result

You activated an experiment.

Related topics

Basics

Introduction to Project and Experiment Management 10

References

Activate Experiment 47

How to Add External Documents to a Project or Experiment

Objective

To make external documents such as Microsoft Word or Adobe PDF documents accessible via ControlDesk, you must add them to the appropriate folder in the file system.

Accessing external documents via ControlDesk

Accessing external documents via ControlDesk means the following:

- External documents are displayed in ControlDesk's Project controlbar .
- You can open external documents in the corresponding programs from within ControlDesk. For example, ControlDesk opens Adobe Reader if you double-click a PDF file in the Project controlbar (if Adobe Reader is installed on your system).
- If you back up a project, the resulting ZIP file contains all external documents that are visible in the Project controlbar.

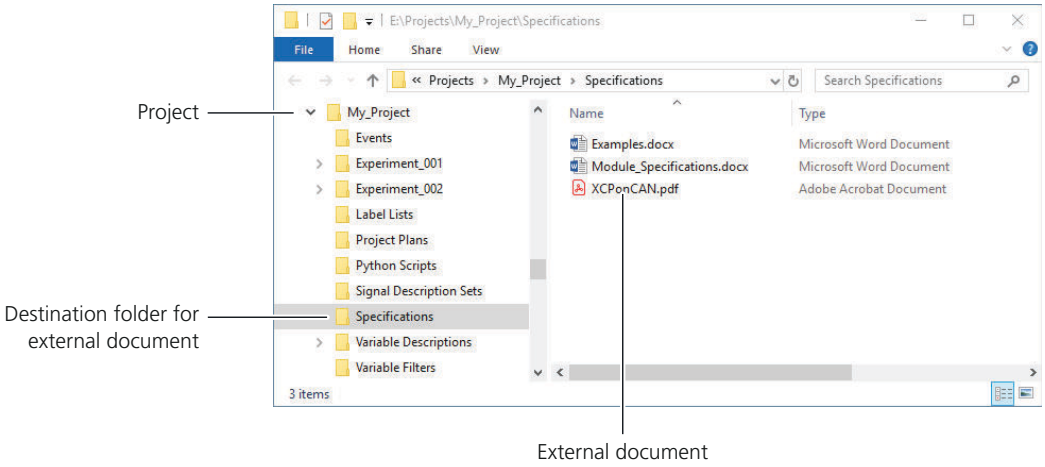
Restrictions

- The file formats of the external documents you want to make accessible via ControlDesk must be specified within the `ControlDeskTemplate_eng.xml` file. Refer to Modifying a Project Template on page 35.
- External documents can be viewed or edited only within the corresponding program, not within ControlDesk.

Method

To add external documents to a project or experiment

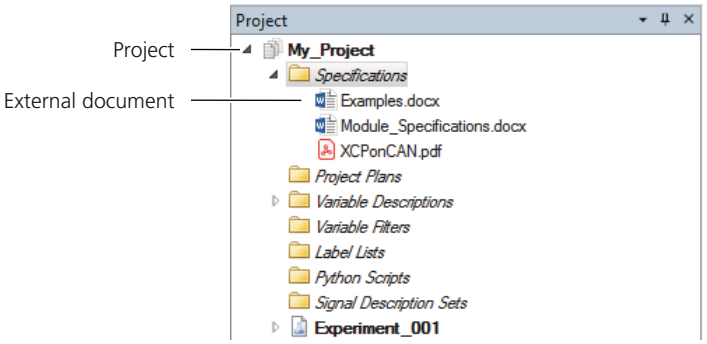
1. In the Project controlbar, right-click the folder, you want to add an external document to.
2. From the context menu, select Explore Folder.
   The destination folder is opened in the File Explorer.
3. In the file system, add the external document to its destination folder.



4 Check whether the file format of the external document is specified to be shown in the folder. To do so, right-click the folder in ControlDesk's **Project** controlbar and select **Configure**. If necessary, specify the file format in the **Documents to be shown in this folder** edit field.

5 In ControlDesk's **Project** controlbar, select **Refresh** from the project's context menu.

ControlDesk renews the directory structure shown in the **Project** controlbar, and displays the external document in its destination folder.



Result

You added an external document to a ControlDesk project. You can open the document via ControlDesk by double-clicking it.

Tip

If you want to add a new folder to a project, select **Add folder** from the context menu of the project.

Related topics

Basics

Introduction to Project and Experiment Management 10

References

Add Folder 48
Configure Folders / Configure (Folder) 53
Refresh 79

How to Configure Folder Settings

Objective

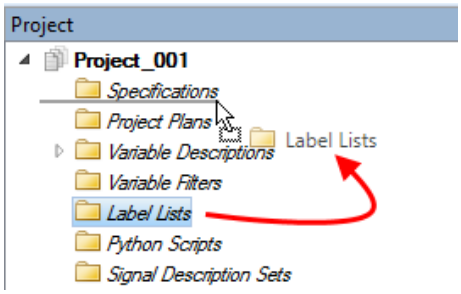
If you are familiar with the project structure, you can change folder settings to specify where files are stored and which folders and files are displayed in the **Project** controlbar.

Project Template Editor

Configured folder settings apply only to the currently open project and active experiment. The entries in the XML template file are not changed. To change folder settings globally for all new projects you define, use ControlDesk's Template Editor instead. Refer to Modifying a Project Template on page 35.

Rearranging folders

In the Project controlbar, you can rearrange folders that are on the same hierarchy tree level by dragging them to new positions.



Folder settings

You can configure the following settings for the currently open project and active experiment:

- You can specify which folders are displayed in the Project controlbar.
- You can specify which files of a folder are displayed in the Project controlbar.

Note

You should be familiar with the detailed structure of a ControlDesk project before changing folder settings.
Some settings are unique for the whole project and changing them in one folder can delete the settings of other folders without a warning.

New folder

If you add a new folder via the context menu of a project, ControlDesk automatically opens a dialog to configure the folder settings.

Tip

You can add groups to the Experiment Layouts folder, the Measurement Data folder, and the Python Scripts folder. Groups are structuring elements that are only displayed in the project tree but not stored in the file system. Refer to Add Group on page 49.

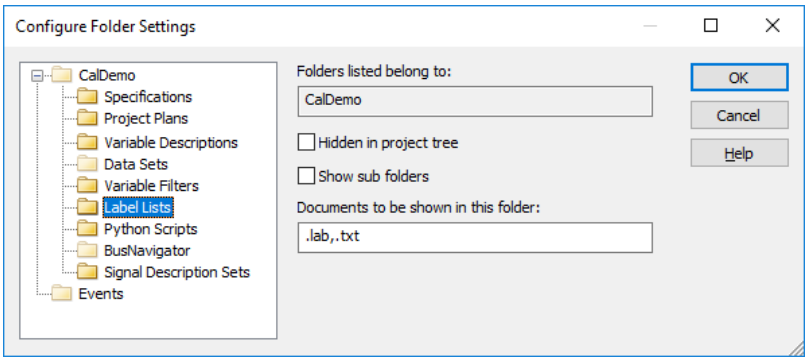
Method

To configure folder settings

1 In the Project controlbar, right-click a folder, the project, or the active experiment item.

2 From the context menu, choose **Configure** (**Configure folders** for the project or the active experiment item).

The **Configure Folder Settings** dialog opens. The selected folder is highlighted in the tree on the left side.



3 If you want to change the selection, click the desired folder.

4 Enter the desired settings for this folder.

Note

Some settings are unique for the whole project. Changing these settings can delete settings of other folders. For more information, refer to Configure Folders / Configure (Folder) on page 53.

Result

You configured the folder settings.

Related topics

Basics

Project Structure and File Storage 12

References

Configure Folders / Configure (Folder) 53

How to Delete a Project

Objective

In ControlDesk, you can delete one or more projects from the file system.

Method

To delete a project

1 On the File ribbon, click Open – Manage Projects.

2 Select one or more projects and click Delete.

You can select multiple projects by pressing **Ctrl** or **Shift** and clicking the projects.

Note

Keep in mind that ControlDesk deletes every file located in the folders of the project. This includes external documents added to the project.

Result

You have deleted one or more projects from the file system.

Related topics

Basics

Introduction to Project and Experiment Management 10

HowTos

How to Open a Project and Experiment 25

References

Open Project + Experiment 64

Recommendations for Working in a Team

Introduction

There are some recommendations for working in a team.

Recommendations

ControlDesk does not let you load and access a ControlDesk project from several PCs *simultaneously*.

Instead, the following practice is recommended:

- To let several users use the same variable description, store the variable description on a network drive.

  When the variable description on the network drive is updated, you can reload the variable description manually. Refer to Reload (Variable Description) (ControlDesk Variable Management 🕮).

As an alternative, you can enable ControlDesk to automatically reload a variable description after it was updated. Refer to Variables Page (ControlDesk Variable Management 🕮).

- To let several users work on the same layouts, perform the following steps:
  1. Export the layouts to a network drive, and remove them from the original experiment. Refer to Export Layout / Export (ControlDesk Layouting 🕮).
  2. Reimport the layouts *as a link*. Refer to Import Layout/Import (ControlDesk Layouting 🕮).
  3. You can then synchronize the settings of the layouts to be reused. Refer to Synchronize (Layout) (ControlDesk Layouting 🕮).

Modifying a Project Template

Introduction

The structure of ControlDesk projects is based on the project template file.

Where to go from here

Information in this section

Information in other sections

Modifying a Project Template

Introduction

The structure of ControlDesk projects is based on the project template file `ControlDeskTemplate_eng.xml` which comes with the ControlDesk installation. It predefines the structure of projects created with ControlDesk.

Default project template

As long as you do not modify the project template, you work with the default settings of the project template. With the default settings, the structure of ControlDesk projects is as described in Hierarchical project structure on page 13.

Modifying a project template

The ControlDesk Template Editor lets you modify a project template. This allows you to adapt the structure of ControlDesk projects, for example, to your company-specific structures of projects.

For instructions, refer to How to Modify the Project Template on page 37.

Projects created before and after modification

The structure of projects created before and after a template modification differs.

Projects created before template modification When you create a project, the current template settings are stored with it. For this reason, projects are not affected by later template modifications.

Projects created after template modification The projects you create *after* a template modification will be structured according to the modified template settings.

Note

To allow ControlDesk to work with a modified project template, ControlDesk must be restarted first.

Location of the project template file

The project template file provided with the ControlDesk installation is located in ControlDesk's Common Program Data folder (see Common Program Data folder on page 8). When you modify an existing project template file, the ControlDesk Template Editor saves the file to the user-specific Local Program Data folder (see Local Program Data folder on page 8).

Project template file used by ControlDesk If there is a project template file in the user-specific Local Program Data folder, ControlDesk uses this file. Otherwise, ControlDesk uses the project template file in ControlDesk's Common Program Data folder.

Tip

To make a modified project template available for *all* users, you have to copy it from the user-specific Local Program Data folder to ControlDesk's Common Program Data folder. This requires that you have write privilege for the Common Program Data folder.

Backup of the project template There can be only one `ControlDeskTemplate_eng.xml` file at a time in ControlDesk's Common Program Data folder and in the user-specific Local Program Data folder.

When you modify an existing project template, the ControlDesk Template Editor automatically backs up the old project template.

The backup file name indicates the backup date and time, for example, `ControlDeskTemplate_eng.xml.backup_2017_07_04(14-13-39)`.

Related topics

HowTos

How to Modify the Project Template....37

How to Modify the Project Template

Objective

You can adapt the structure of ControlDesk projects to company-specific structures of projects by modifying the project template.

Restrictions

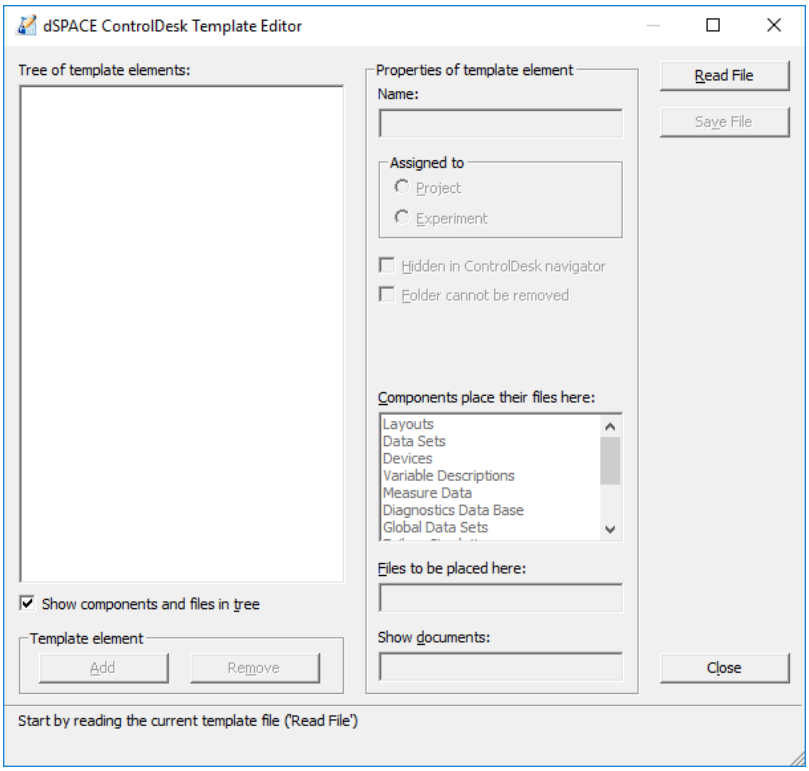
Modifying the project template affects only the structure of ControlDesk projects created *after* template modification. The structure of existing projects cannot be changed afterwards by a modified project template.

Method

To modify the project template

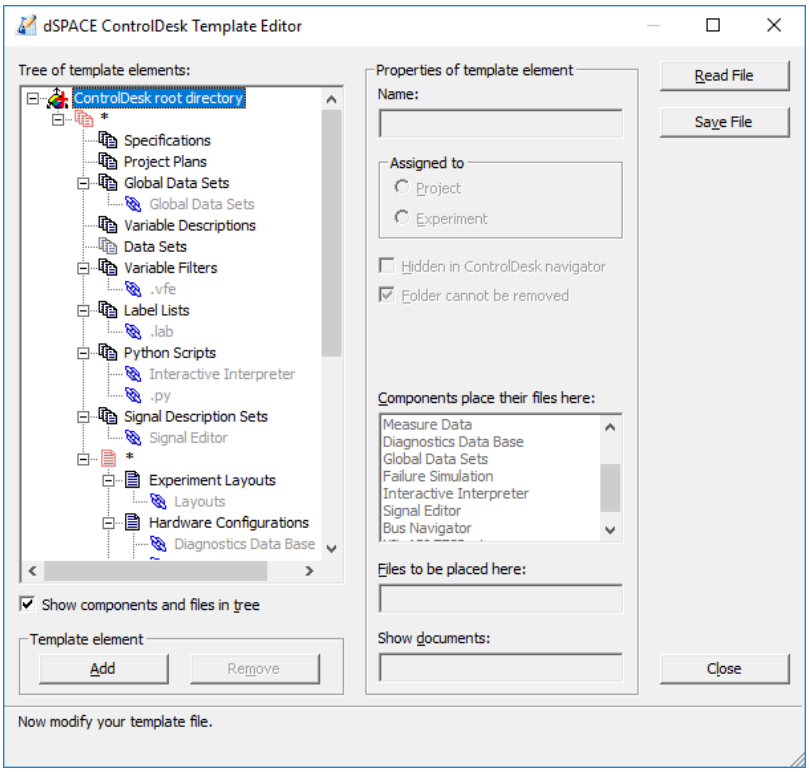
1 Click the Windows Start button and select dSPACE ControlDesk 7.4 – ControlDesk Template Editor.

The ControlDesk Template Editor opens.



2 Click Read file.

The ControlDesk Template Editor loads the current project template.



3 Select * in the tree of template elements. Click Add to add a new template element to the project template.

New elements have the symbol.

4 Name the new template element according to your requirements.

You may also enter * as the name. In this case, the folder that corresponds to the template element automatically gets the name of the project or experiment when you create a new project or experiment.

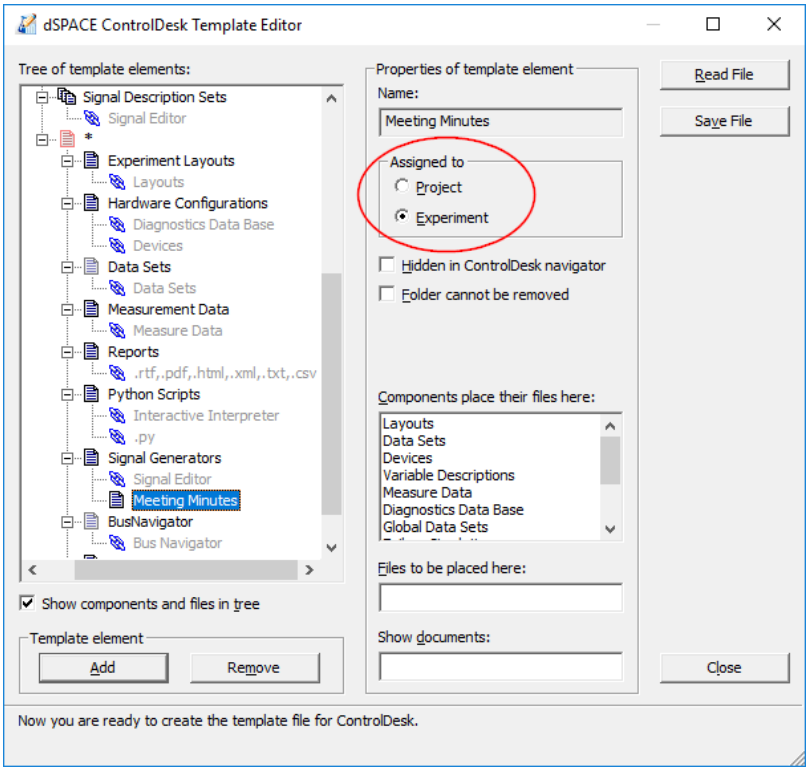
Tip

To rename a template element, press **F2** and enter the name according to your requirements.

5 Click Remove to remove obsolete template elements.

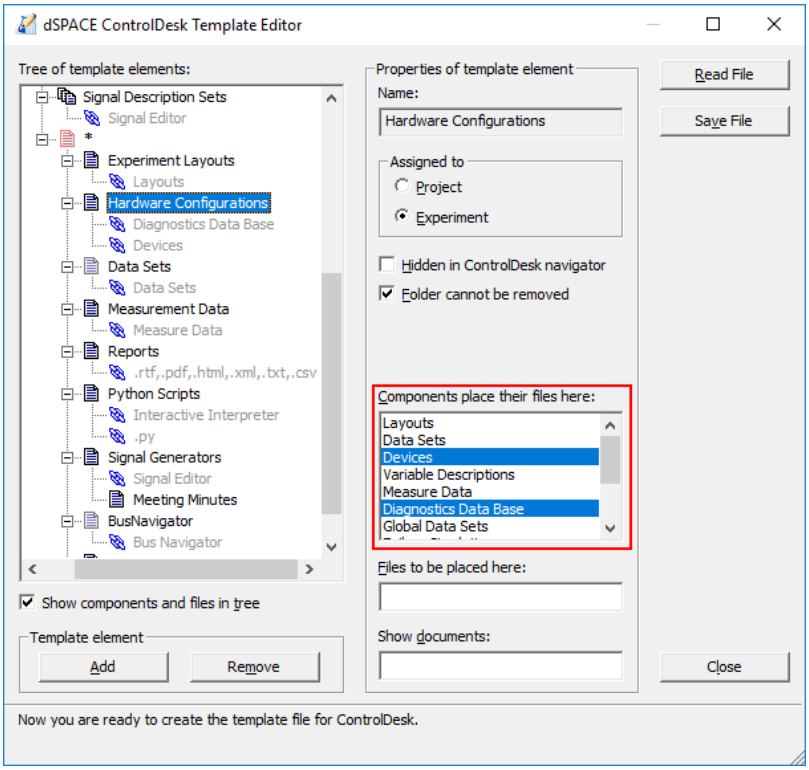
6 Repeat steps 3 … 5 until the template structure matches your requirements.

7 Assign each template element having the symbol to either the project or the experiment via the Assigned to buttons.



For each template element, ControlDesk will create a folder on your file system. The assignment specifies whether the folder will belong to the corresponding project or experiment.

8 Use Components place their files here to assign the files of a component to the selected template element. This will affect all files types handled by the selected component unless you specify another assignment via the Files to be placed here edit field.



9 Optionally, use Files to be placed here to type file name extensions for inclusion in the selected template element. You can do this regardless of what you defined in the previous step. Use this function if you want to store different file types of one component in different folders.

10 Optionally, use Show documents to display external documents (for example, PDF, DOC) in the selected template element.

11 Click Save File to save the project template.

Result

You modified the project template.

Next step

You can now define new projects that are structured according to the modified project template. Refer to How to Define a Project on page 19.

Related topics

Basics

Modifying a Project Template....35

References

ControlDesk Template Editor....51

Reference Information

Where to go from here

Information in this section

Project/Experiment-Related Properties

Item Info Properties

Purpose

Displays the information properties of the item that is selected in the Project controlbar. You can change the description text of some items.

Note

Whether the item information is available in the Properties controlbar depends on the specific item. You can always access the item information in the Item Info field of the **Project** controlbar. Refer to Project on page 69.

Properties

Author Displays the author that created the item.

Date Displays the creation date of the item.

Description Displays the description of the item. You can change the description text of some items.

Location (Available only for file-based items) Displays the storage location of the item.

Related topics

Basics

Basics on Projects and Experiments 10

References

Project 69

Project/Experiment-Related Commands

Where to go from here

Information in this section

Activate Experiment

Access

This command is available only for inactive experiments. You can access it via:

Ribbon	None
Context menu of	Project controlbar – experiments
Shortcut key	None
Icon	None

Purpose

To activate an experiment.

Description

Within a ControlDesk project, only one experiment can be active at a time. All the other experiments are inactive. To work with an experiment, it must be active.

Related topics

HowTos

Add Folder

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – project, experiments, and folders
Shortcut key	None
Icon	None

Purpose

To add a folder to the selected element.

Description

ControlDesk opens the **Create New Folder** dialog.

Tip

You can add groups to specific folders in the project structure. The groups are displayed in the project tree, but not stored in the file system. Refer to Add Group on page 49.

Create New Folder dialog

Name of the new folder Lets you enter a name for the new folder.

Location of the new folder Lets you specify where the new folder is stored.

OK Creates the new folder and opens the **Configure Folder Settings** dialog which lets you specify project-specific folder settings. Refer to Configure Folders / Configure (Folder) on page 53.

Related topics

Basics

Project Structure and File Storage 12

HowTos

How to Add External Documents to a Project or Experiment 29

References

Add Group 49

Add Group

Access

You can access this command via:

Ribbon	None
Context menu of	▪ **Project** controlbar – **Experiment Layouts** folder ▪ **Project** controlbar – **Measurement Data** folder ▪ **Project** controlbar – **Python Scripts** folder ▪ **Project** controlbar – **Data Sets** folder[1)] ▪ **Project** controlbar – group node
Shortcut key	`Insert`
Icon	None

[1)] The **Data Sets** folder can be shown or hidden. Refer to Data Set Grouping (ControlDesk Calibration and Data Set Management 🕮).

Purpose

To add a group (a structuring element that is displayed only in the project tree) to the selected folder.

Description

You can move elements of the parent folder to a group via drag and drop. Groups can be renamed via the context menu. You can build a structure by adding subgroups to a group.

For data sets, you must enable data set grouping first. The **Data Sets** folder is displayed and you can add groups and subgroups to it. Refer to Data Set Grouping (ControlDesk Calibration and Data Set Management 🕮).

Note

- A group is a structuring element that is only visible in the project tree. It is not added to the project structure in the file system. If you want to add folders to the project structure that are also stored in the file system, refer to Add Folder on page 48.
- If you remove a group, its elements (files and groups) are moved up one level.

Related topics

Basics

Project Structure and File Storage 12

References

Add Folder 48

Backup Project

Access

You can access this command via:

Ribbon	File – Save As
Context menu of	Project controlbar – project
Shortcut key	None
Icon	

Note

This command is not available in operator mode.

Purpose

To save either the loaded project and all experiments or the project and just the active experiment as a ZIP archive.

Result

Opens a standard Save As dialog that allows you to zip the loaded project.

Description

ZIP archives contain only files that are stored within the project tree. The files are archived with relative paths. They can thus be extracted anywhere. To load a project, either it should be located below a defined root directory, or a suitable root directory should be specified in the General Preferences. If a ZIP archive contains a file from a different file system, the absolute path is stored.

Save As dialog

To save the loaded project as a ZIP archive.

Backup only the active experiment Lets you specify that only the currently active experiment will be backed up.

Related topics

Basics

Project Structure and File Storage 12

ControlDesk Template Editor

Access	The ControlDesk Template Editor is available from the Start menu: Start – dSPACE ControlDesk 7.4 – ControlDesk Template Editor.
Purpose	To modify an existing project template.
Tree of template elements	**Show components and files in tree** Lets you select whether components are shown in your template tree.
Template element	**Add** Lets you add an element to the project template tree. **Remove** Lets you remove a selected element from the project template tree.
Properties of template element	**Name** Displays the currently selected template element. **Assigned to** Lets you assign the selected template element to a project or an experiment. For each template element, ControlDesk creates a folder on your file system. The assignment specifies whether the folder ▪ Will be created together with a project or together with an experiment ▪ Will belong to a project or an experiment **Hidden in ControlDesk navigator** Lets you hide the selected template element in ControlDesk's Project controlbar. Nevertheless, the corresponding folder will be created in the file system. **Folder cannot be removed** Lets you specify whether the folder can be removed in ControlDesk's Project controlbar. **Components place their files here** Lets you specify that the selected ControlDesk component(s) will save all their files to this folder on the file system and display them under this item in ControlDesk's Project controlbar. **Files to be placed here** Lets you specify one or more types of files to be saved in the selected template element. If you specify several file types, they must be comma-separated, for example, MF4, MAT, RDL. You can do this regardless of what you defined via Components place their files here. **Show documents** Lets you specify file types of external documents, that is, documents not handled by ControlDesk (e.g., PDF, DOC). They will be displayed with the selected template element.

File options

Read file Loads the project template for viewing and modification. For instructions, refer to How to Modify the Project Template on page 37.

Save file Saves the loaded project template file.

Close Closes the ControlDesk Template Editor.

Related topics

Basics

Modifying a Project Template....35

HowTos

How to Modify the Project Template....37

Examples

Example of Project Structures....17

Close Project + Experiment/ Close

Access

You can access this command via:

Ribbon	File
Context menu of	Project controlbar – project
Shortcut key	None
Icon	

Purpose

To close the currently loaded project and all experiments belonging to it.

Description

If the current project was changed since the last save operation, you are asked whether you want to save it.

Configure Folders / Configure (Folder)

Access

You can access this command via:

Ribbon	None
Context menu of	▪ Project controlbar – project or active experiment ▪ Project controlbar – folder in a project or active experiment
Shortcut key	None
Icon	None

Purpose

To configure project-specific folder settings.

Result

ControlDesk opens the **Configure Folder Settings** dialog.

Description

The structure of ControlDesk projects is based on an XML template file. This file determines, for example, where a ControlDesk component like the Data Set Management stores its files (automatic file storage).

The **Configure Folder Settings** dialog enables you to specify such settings for the selected folder in the current project.

The settings in the **Configure Folder Settings** dialog apply only to the currently open project. The entries in the XML template file are not changed. To change folder settings globally for all new projects you define, use ControlDesk's Template Editor instead. Refer to ControlDesk Template Editor on page 51.

Configure Folder Settings dialog

Lets you specify project-specific or experiment-specific folder settings.

A tree with the folders of the project or experiment is displayed on the left side. The selected folder is highlighted (you cannot edit its name here). You can change the selection in the tree.

Hidden in project tree Lets you specify whether to show the selected folder in the Project controlbar. Hiding a folder does not affect its other settings. They are valid even when the folder is hidden.

Show sub folders Lets you specify whether sub folders and their contents (documents and further subfolders) of the selected folder are shown in the **Project** controlbar. The documents to be shown in a subfolder are taken from the parent folder where you enable the **Show sub folders** option.

Note

Folders and their contents are included in a project backup (via ControlDesk's **Backup Project** command) only if the folders are displayed in the **Project** controlbar.

Documents to be shown in this folder Lets you specify a comma-separated list with file name extensions, for example: .doc,.txt,.pdf.

Only documents with one of the specified extensions are displayed in the **Project** controlbar, others are hidden.

Related topics

HowTos

How to Configure Folder Settings..... 30

Create Shortcut

Access

You can access this command via:

Ribbon	None
Context menu of	**Project** controlbar – experiment
Shortcut key	None
Icon	None

Purpose

To create a desktop shortcut to an experiment.

Result

A shortcut to the experiment is created on your desktop.

Description

You can create a desktop shortcut for each ControlDesk experiment. This allows you to open ControlDesk and quickly load a specific experiment.

Note

ControlDesk must be closed if you want to open an experiment via desktop shortcut.

Explore Folder

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – items
Shortcut key	None
Icon	None

Purpose

To open the folder of the selected item in the File Explorer.

Result

An Explorer Window opens showing the folder of the selected item.

Export (Experiment)

Access

You can access this command via:

Ribbon	File – Save As
Context menu of	Project controlbar – active experiment
Shortcut key	None
Icon	

Note

This command is not available in operator mode.

Purpose

To export the currently active experiment as a dSPACE archive (DSA) file.

Result

ControlDesk opens the **Save As dialog** to save the selected experiment and all the files belonging to it as one unit in a dSPACE archive file. The archive's file name extension is DSA.

Note

- Importing an experiment exported with a prior ControlDesk version fails if the experiment contains data that is incompatible with the current ControlDesk version. You also get a corresponding error message.
  To import such an experiment, you have to migrate the project containing the experiment. Refer to Migrating from Prior Versions of ControlDesk (ControlDesk Introduction and Overview 🕮).
- You can export an experiment from ControlDesk and import it in ConfigurationDesk. However, the experiment can be used in ControlDesk only if the ControlDesk version used for exporting the experiment and the ControlDesk version to be used after importing the experiment in ConfigurationDesk are identical.
  To complete the experiment import into ConfigurationDesk, you must activate the experiment in ControlDesk.

Save As dialog

Advanced Lets you select items to be included in the export.

You can exclude memory-intensive or obsolete items such as measurement files or reports from being exported.

Note

If you exclude items from being exported, ControlDesk does not check the rest of the experiment for consistency. You have to ensure experiment consistency yourself.

File Missing Dialog

Access

This dialog opens when ControlDesk opens a project and cannot find a referenced file. The following commands are involved:

- Open Project + Experiment on page 64
- Open Project + Experiment from Backup on page 67

File Missing dialog

To specify how to handle a missing file.

Ignore Opens the project without the file. The reference to the file remains in the project. Working with the project will be impaired to a greater or lesser degree depending on how important the file is.

Remove Removes the file from the project. Choose this option only if the file is obsolete.

Browse Opens a standard file dialog that lets you search for the file. Choose this option if you know that the file is available but has been moved or renamed.

Import - Experiment

Access

You can access this command via:

Ribbon	Open
Context menu of	Project controlbar – project
Shortcut key	None
Icon	None

Note

This command is not available in operator mode.

Purpose

To import an experiment saved as a dSPACE archive.

Result

A standard Open dialog is displayed to import the selected experiment. It is extracted below the selected project path. The imported files are added to the project. The imported experiment is not activated.

Note

Importing an experiment exported with a prior ControlDesk version fails if the experiment contains data that is incompatible with the current ControlDesk version. You also get a corresponding error message.
To import such an experiment, you have to migrate the project containing the experiment. Refer to Migrating from Prior Versions of ControlDesk (ControlDesk Introduction and Overview).

Manage Projects

Access

You can access this command via:

Ribbon	File - Open
Context menu of	None
Shortcut key	None

Icon	

Purpose

To delete projects.

Result

Opens the Manage Projects dialog, which lets you select a project to delete.

Manage Projects dialog

To specify project root directories and delete projects.

Root directory Lets you select the project root directory.

	To specify a new project root directory.

Projects and experiments Lets you browse in the list of projects and experiments available in the selected project root directory. The currently active project is not listed.

Symbol	Meaning
	ControlDesk project
	ControlDesk experiment

The context menu of a project provides the following commands:

Purpose	Refer to
To delete the selected project(s) from the file system.	Delete on page 58

Delete Deletes the selected project(s) from the file system. Multiple selection is possible by pressing **Ctrl** or **Shift** when clicking a project. ControlDesk only deletes those files that are associated with the project. Other files located in the folders of the deleted project are left untouched. This may result in folders remaining on your hard-disk. As a consequence, ControlDesk will refuse to create a new project with the name of the deleted project because the folder name is already in use.

Tip

- For multiple selection press **Ctrl** or **Shift** when clicking a project. Press **Ctrl** to select a number of single projects with the mouse or **Shift** to select a block of projects. Press **Ctrl** + **A** to select all the projects.
- Press **Delete** to delete the selected project(s).

Manage Opens the Project page of the ControlDesk Options dialog, where you can specify further project root directories. Refer to Project Page on page 72.

Manage Projects / Select a Project Dialog

Access

- The Manage Projects dialog opens when Manage Projects is invoked.
- The Select a Project dialog opens when Open Project + Experiment is invoked.

Manage Projects dialog

To specify project root directories and delete projects.

Root directory Lets you select the project root directory.

	To specify a new project root directory.

Projects and experiments Lets you browse in the list of projects and experiments available in the selected project root directory. The currently active project is not listed.

Symbol	Meaning
	ControlDesk project
	ControlDesk experiment

The context menu of a project provides the following commands:

Purpose	Refer to
To delete the selected project(s) from the file system.	Delete on page 59

Delete Deletes the selected project(s) from the file system. Multiple selection is possible by pressing **Ctrl** or **Shift** when clicking a project. ControlDesk only deletes those files that are associated with the project. Other files located in the folders of the deleted project are left untouched. This may result in folders remaining on your hard-disk. As a consequence, ControlDesk will refuse to create a new project with the name of the deleted project because the folder name is already in use.

Tip

- For multiple selection press **Ctrl** or **Shift** when clicking a project. Press **Ctrl** to select a number of single projects with the mouse or **Shift** to select a block of projects. Press **Ctrl + A** to select all the projects.
- Press **Delete** to delete the selected project(s).

Manage Opens the Project page of the ControlDesk Options dialog, where you can specify further project root directories. Refer to Project Page on page 72.

Select a Project dialog

To select a project and an experiment to be opened.

Tip

If you want to get an overview of the project structure, the fastest way is to select a project instead of an experiment. You can activate an experiment later on.

Root directory Lets you select the project root directory.

	Opens the Browse for Folder dialog for you to select a folder from the file system or create a new folder to be used as the project root directory.

Projects and experiments Lets you browse in the list of projects and experiments available in the selected project root directory.

Symbol	Meaning
	ControlDesk project
	ControlDesk experiment

Delete Deletes the selected project(s) from the file system. Multiple selection is possible by pressing **`Ctrl`** or **`Shift`** when clicking a project. ControlDesk cannot delete files that are used by other applications. This may result in folders remaining on your hard disk. As a consequence, ControlDesk will refuse to create a new project with the name of the deleted project because the folder name is already in use.

Manage Opens the Project page of the ControlDesk Options dialog, where you can specify further project root directories. Refer to Project Page on page 72.

OK Opens the selected project or the selected experiment and the project it belongs to.

This button is disabled as long as no project or experiment is selected in the Projects and experiments list.

Cancel Stops the project and/or experiment from being loaded.

Related topics

References

Manage Projects....57
Open Project + Experiment....64

New (Backstage View)

Access

You can access this ribbon group via:

Ribbon	File
Context menu of	None
Shortcut key	None
Icon	None

Purpose

To provide access to commands that allow you to define a new project and/or experiment.

Description

You have access to commands such as:

- New Project and Experiment / New Experiment on page 61

New Project and Experiment / New Experiment

Access

You can access this command via:

Ribbon	File - New
Context menu of	▪ Project controlbar – project ▪ Project controlbar (if no project is currently open)
Shortcut key	None
Icon	

Note

This command is not available in operator mode.

Purpose

To define a new project or a new experiment.

Description

No project is currently open ControlDesk opens the Define a Project dialog. You have to define a new project or open an existing one before you can define a new experiment.

A project is currently open ControlDesk opens the Define an Experiment dialog. This lets you define a new experiment within the open project.

Define a Project dialog

To define a new ControlDesk project.

Name of the project Enter a new project name in the edit field, or select an existing project via the Browse button. The name you enter must not contain a point or dot as the first or last character. By default, the project is named 'Project_nnn', where 'nnn' stands for a number that will be increased if the name already exists as a file or folder name in the project root directory.

Root directory Select a project root directory.

	Opens the Projects page of the ControlDesk Options dialog. You can specify further project root directories on this page.

Next > Opens the next dialog. This button is disabled as long as the Name of the project edit field is empty or invalid.

Finish Lets you finish project creation without having to define an experiment. This button is disabled as long as the Name of the project edit field is empty or invalid.

Cancel Closes the dialog without saving any of your settings.

Define an Experiment dialog

To define a new experiment within the current project.

Name of the experiment Enter the desired experiment name. By default, the experiment is named 'Experiment_nn', where 'nnn' stands for a number that will be increased if the name already exists as file name in the directory of the project. The name must be unique within the project, and must not contain any of the following characters: `* ? | < > : / \ "`.

Note

Do not specify an experiment name that is equal to the name of a file folder within the project that the experiment belongs to. Otherwise, automatic project migration might fail when you try to open the project in a later ControlDesk version.

Experiments already contained in the project If your project already contains experiments, they are displayed here (no changes possible).

< Back Opens the previous dialog. This button is disabled if a project is currently loaded.

Next > Opens the next dialog. This button is disabled or invalid as long as the Name of the experiment edit field is empty or invalid.

Finish Lets you finish experiment creation without having to add a platform/device to the experiment. This button is disabled or invalid as long as the Name of the experiment edit field is empty or invalid.

Cancel Closes the dialog without saving any of your settings.

Related topics

HowTos

Open (Backstage View)

Access

You can access this ribbon group via:

Ribbon	File
Context menu of	None
Shortcut key	None
Icon	None

Purpose

To provide access to commands that allow you, for example, to open a project.

Description

You have access to commands such as:

- Open Project + Experiment on page 64
- Open Project + Experiment from Backup on page 67
- Import - Experiment on page 57
- Open File or Project (ControlDesk User Interface Handling 🕮)
- Manage Projects on page 57

Open (External Document / ECU Image File Created in ControlDesk)

Access

You can access this command via:

Ribbon	None
Context menu of	▪ **Project** controlbar – external documents ▪ **Project** controlbar – ECU Image file
Shortcut key	`Enter`
Icon	None

Purpose

To open an external document in the associated application or to open the dSPACE ECU Flash Programming Tool using the ECU Image file as the flash data source.

Description

External documents are, for example, PDF or XLS files. This command opens the selected document in the corresponding program, for example, in Adobe Acrobat Reader or Microsoft Excel.

If the **Open** command is called from an ECU Image file that was created in the HEX, MOT or S19 file format in ControlDesk, the dSPACE ECU Flash Programming Tool opens. The ECU Image file is the default flash data source.

Note

The HEX, MOT, S19, and SREC file name extensions are automatically associated with the dSPACE ECU Flash Programming Tool, if you confirmed this when installing the tool. If the dSPACE ECU Flash Programming Tool does not start when you use the **Open** command on an ECU Image file, check whether the file name extension of the selected ECU Image file is associated with the dSPACE ECU Flash Programming Tool, that is, whether the dSPACE ECU Flash Programming Tool has been set for opening files of this type. If it is, a small icon indicating the dSPACE ECU Flash Programming Tool is displayed next to the ECU Image file in File Explorer.

Related topics

HowTos

How to Start the dSPACE ECU Flash Programming Tool (ControlDesk Platform Management 🕮)

Open Project + Experiment

Access

You can access this command via:

Ribbon	File - Open
Context menu of	Project controlbar (if no project is currently open)
Shortcut key	**Ctrl+Shift+O**
Icon	

Purpose

To open an experiment and the project it belongs to.

Result

Opens the Select a Project dialog that allows you to select an existing experiment to be opened.

Tip

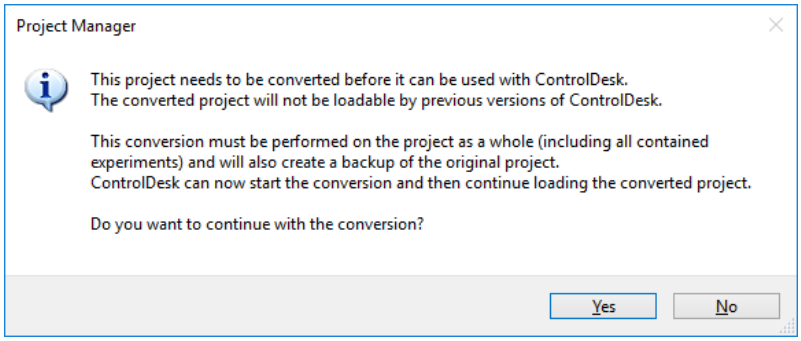
To open one of the most recent experiments that were open in ControlDesk, use the Recently used Projects and Experiments command.

Description

If another experiment is already open, it and the project it belongs to are closed. Then the experiment you selected is opened and activated.

When the experiment is opened, ControlDesk remains in offline mode.

Automatic migration of projects When you open a project last saved with ControlDesk 7.3 or earlier in ControlDesk 7.4, ControlDesk opens the following dialog that lets you convert the project and all its contained experiments.



Click Yes to confirm project conversion.

Tip

- Since the converted project cannot be loaded with ControlDesk 7.3 or earlier, a backup of the original project is stored in the project root folder during migration.
- Information on the conversion process is added to the dSPACE Log.

Select a Project dialog

To select a project and an experiment to be opened.

Tip

If you want to get an overview of the project structure, the fastest way is to select a project instead of an experiment. You can activate an experiment later on.

Root directory Lets you select the project root directory.

	Opens the Browse for Folder dialog for you to select a folder from the file system or create a new folder to be used as the project root directory.

Projects and experiments Lets you browse in the list of projects and experiments available in the selected project root directory.

Symbol	Meaning
	ControlDesk project
	ControlDesk experiment

Delete Deletes the selected project(s) from the file system. Multiple selection is possible by pressing **Ctrl** or **Shift** when clicking a project. ControlDesk cannot delete files that are used by other applications. This may result in folders remaining on your hard disk. As a consequence, ControlDesk will refuse to create a new project with the name of the deleted project because the folder name is already in use.

Manage Opens the Project page of the ControlDesk Options dialog, where you can specify further project root directories. Refer to Project Page on page 72.

OK Opens the selected project or the selected experiment and the project it belongs to.

This button is disabled as long as no project or experiment is selected in the Projects and experiments list.

Cancel Stops the project and/or experiment from being loaded.

File Missing dialog

To specify how to handle a missing file.

Ignore Opens the project without the file. The reference to the file remains in the project. Working with the project will be impaired to a greater or lesser degree depending on how important the file is.

Remove Removes the file from the project. Choose this option only if the file is obsolete.

Browse Opens a standard file dialog that lets you search for the file. Choose this option if you know that the file is available but has been moved or renamed.

Related topics

Basics

Problem with Long Project Loading Times..... 100

HowTos

How to Open a Project and Experiment..... 25

Open Project + Experiment from Backup

Access

You can access this command via:

Ribbon	File - Open
Context menu of	Project controlbar (if no project is currently open)
Shortcut key	None
Icon	

Note

This command is not available in operator mode.

Purpose

To open a project and experiment(s) from a backup file.

Result

Opens the Open dialog for you to select a backup file to be loaded.

ControlDesk then performs the following steps:

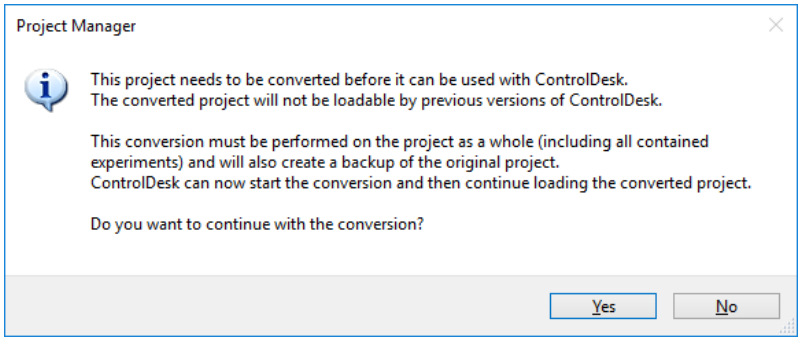
- The project in the selected backup file is loaded.
- The experiment that was active when the backup was created is activated.
- The layouts that were open when the backup was created are opened in the working area.

Description

If another experiment is already open, it and the project it belongs to are closed. Then the experiment you selected is opened and activated.

When the experiment is opened, ControlDesk remains in offline mode.

Automatic migration of projects When you open a project last saved with ControlDesk 7.3 or earlier in ControlDesk 7.4, ControlDesk opens the following dialog that lets you convert the project and all its contained experiments.



Click Yes to confirm project conversion.

Tip

- Since the converted project cannot be loaded with ControlDesk 7.3 or earlier, a backup of the original project is stored in the project root folder during migration.
- Information on the conversion process is added to the dSPACE Log.

Open dialog

Look in Lets you specify the directory where the backup file to be opened is stored.

File name Lets you specify the name of the backup file.

Files of type Lets you specify the file type of the backup file.

The project will be extracted to this root directory Lets you specify the root directory in which the project and experiment(s) are extracted. The project will be loaded from this root directory. Click  to open the Project Page on page 72 and create a new root directory.

Overwrite existing files Allows ControlDesk to overwrite files outside the project (referenced external files, for example, background images or sound files) without a confirmation prompt.

File Missing dialog

To specify how to handle a missing file.

Ignore Opens the project without the file. The reference to the file remains in the project. Working with the project will be impaired to a greater or lesser degree depending on how important the file is.

Remove Removes the file from the project. Choose this option only if the file is obsolete.

Browse Opens a standard file dialog that lets you search for the file. Choose this option if you know that the file is available but has been moved or renamed.

Related topics

HowTos

How to Open a Project and Experiment 25

Project

Access

You can access this command via:

Ribbon	View – Controlbar - Switch Controlbars – Project
Context menu of	None
Shortcut key	**Alt+Shift+2**
Icon	

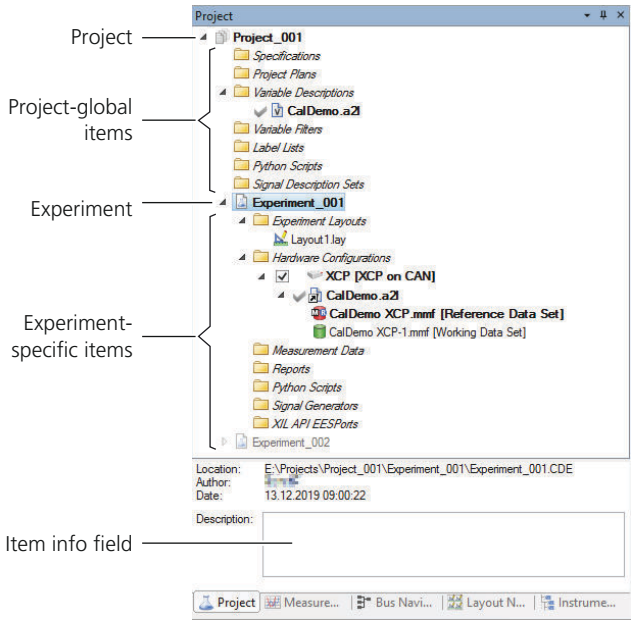
Purpose

To show or hide the Project controlbar.

Description

Project Controlbar A controlbar that provides access to projects and experiments and all the files they contain.

The Project controlbar consists of the following areas:



Management of a ControlDesk project Within the Project controlbar, you can manage the items belonging to the currently open project. Depending on the selected item, the Project controlbar provides a context menu, which allows

you to carry out tasks such as adding platforms/devices to an experiment, removing items from an experiment, copying and activating data sets.

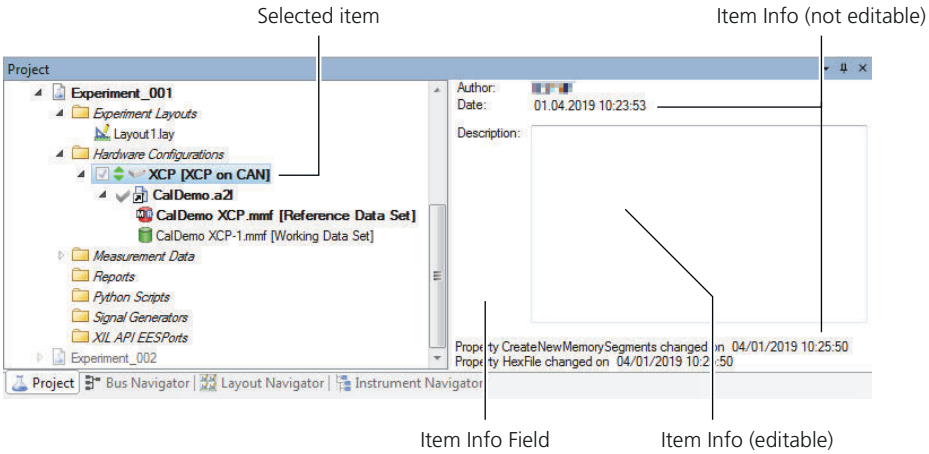
Item type and status The Project controlbar displays each item together with a symbol giving information on the item type and status:

Symbol	Meaning
Project	
	Project (only one project can be loaded at a time)
	Project with activated automatic saving at a specified time interval
	Folder with files belonging to the project
Experiments	
	Active and inactive experiments
	Folder with files belonging to the project or experiment
	Group (structuring element)
Platforms/Devices	
	Platform/Device is *connected*.[1)]
	Platform/Device is *disconnected*.[1)]
	A *measurement* is currently *running* on the platform/device.[1)]
	*Online calibration* is *started* for the platform/device.[1)]
	Platform/Device is *disabled*.[1)]
	Platform/Device is *unplugged*.[1)]
Layouts	
	Layout file belonging to the experiment
	Layout file that references an external file
	Layout file with locked layout editing
Variable descriptions	
	Active variable description (A2L, DBC, SDF, or LDF file)
	Inactive variable description (A2L, DBC, SDF, or LDF file)
Variable filters	
	Active variable filter
	Inactive variable filter

Symbol	Meaning
Data sets	
	Master data set (derived from ECU Image file)
	Reference data set
	Read-only data set
	Writable data set (complete)
	Writable sub data set (incomplete data set)
	Closed data set
Measurements	
	Measurement data file
	Link to measurement data file
Diagnostics database	
	Active ODX database
	Inactive ODX database
	Active ODX database (binary format)
	Inactive ODX database (binary format)
Logical links	
	Logical link is *connected*.[1)]
	Logical link is *disconnected*.[1)]
	*Online calibration* is *started* for the logical link.[1)]
	A *measurement* is currently *running* for the logical link.[1)]
Signal Editor	
	Signal description set
	Signal generator
Electrical Error Simulation	
	EESPort
	Error configuration
Failure Simulation	
	Failure simulation system

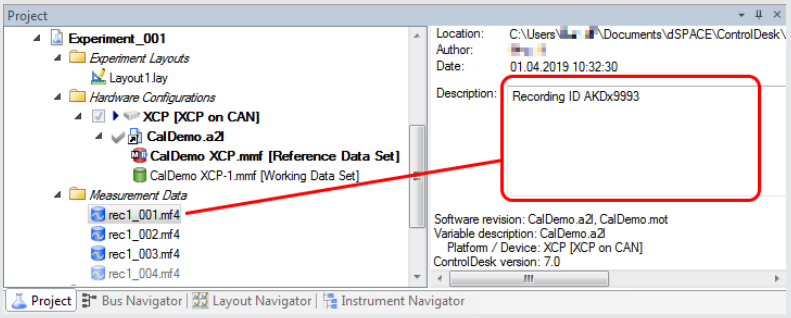
[1)] For details on platform/device states, refer to Basics of Platform/Device States (ControlDesk Platform Management 🕮).

Item info field Depending on the item selected in the Project controlbar, information such as the item's author and date is displayed in the Item info field of ControlDesk's Project controlbar.



Tip

You can edit the information displayed in the description area of the Item info field.



Related topics

Basics

Basics of Platform/Device States (ControlDesk Platform Management 🕮)

Project Page

Access

This page is part of the ControlDesk Options dialog.

The dialog can be opened via the **Options Command** (ControlDesk User Interface Handling 🕮).

Purpose

To specify project root directories and startup options.

Description

To define and work with projects and experiments in ControlDesk, at least one project root directory must be specified.

Dialog settings

Root directories Lets you specify one or more project root directories.

 `Insert`	To specify a new project root directory. A new line is added to the list of project root directories. You can enter a directory name in the edit field or select a directory via the Browse button. If you enter the path manually, the directory must already exist, it is not created by ControlDesk.
 `Delete`	To remove the selected directory from the list of project root directories.
 `Alt` + ↑	To move the selected directory up in the list of project root directories.
 `Alt` + ↓	To move the selected directory down in the list of project root directories.
	To browse the contents of the selected directory.
`F2`	To edit the selected directory in the list of project root directories.

Automatically load the most recently used experiment on startup Lets you specify whether to load the most recently used experiment when ControlDesk is started. The next time ControlDesk is started, the most recently used experiment is loaded, even if it was closed before ControlDesk was closed.

Starting an experiment via a desktop shortcut (see Create Shortcut on page 54) is not affected by this setting.

Note

Remember that large projects will take a long time to load.

Only show projects that contain experiments generated by this product (Intended for future use) Activates a filter when you browse for projects/experiments via Open Project + Experiment on page 64.

Automatically save project every <x> minutes Lets you specify to save the project at a regular time interval in minutes. If activated, a clock symbol is displayed in the Project controlbar next to the project item.

If you activate automatic saving and enter 0 for the minutes, the project is not saved and no clock symbol is displayed.

Related topics

References

Options Command (ControlDesk User Interface Handling 🕮)

Project Wizard

Access

ControlDesk's Project Wizard consists of a sequence of multiple dialogs. You access the wizard by one of the following commands:

- New Project and Experiment / New Experiment on page 61 (depending on whether a project is currently open)
- Insert Platform / Add Platform/Device (ControlDesk Platform Management 🕮)
- Add Variable Description (ControlDesk Variable Management 🕮)

Note

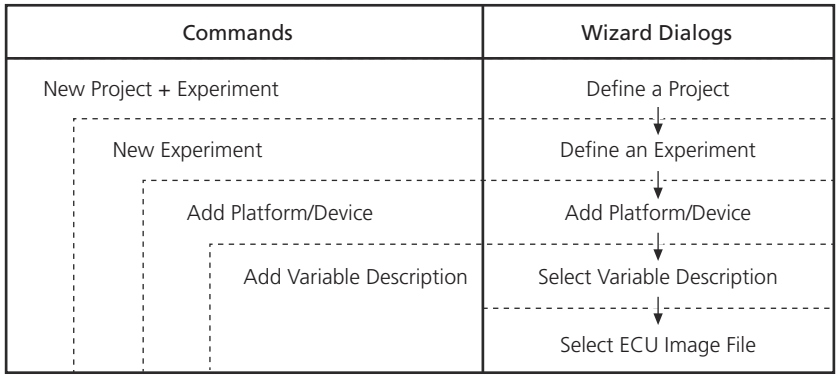
The Project Wizard is not available in the operator mode.

Purpose

To define a new project or experiment, add a platform/device to an experiment, or select a variable description for a platform/device.

Description

You get only the wizard dialogs relevant to the command that you selected. The following illustration shows the commands, and the relevant dialogs in the order in which they appear.



For details on the dialogs, refer to:

- Define a Project dialog on page 75
- Define an Experiment dialog on page 75
- Add Platform/Device dialog on page 76

- Select Variable Description dialog on page 77
- Select ECU Image File dialog on page 78

Define a Project dialog

To define a new ControlDesk project.

Name of the project Enter a new project name in the edit field, or select an existing project via the Browse button. The name you enter must not contain a point or dot as the first or last character. By default, the project is named 'Project_nnn', where 'nnn' stands for a number that will be increased if the name already exists as a file or folder name in the project root directory.

Root directory Select a project root directory.

	Opens the Projects page of the ControlDesk Options dialog. You can specify further project root directories on this page.

Next > Opens the next dialog. This button is disabled as long as the Name of the project edit field is empty or invalid.

Finish Lets you finish project creation without having to define an experiment. This button is disabled as long as the Name of the project edit field is empty or invalid.

Cancel Closes the dialog without saving any of your settings.

Define an Experiment dialog

To define a new experiment within the current project.

Name of the experiment Enter the desired experiment name. By default, the experiment is named 'Experiment_nn', where 'nnn' stands for a number that will be increased if the name already exists as file name in the directory of the project. The name must be unique within the project, and must not contain any of the following characters: `* ? | < > : / \ "`.

Note

Do not specify an experiment name that is equal to the name of a file folder within the project that the experiment belongs to. Otherwise, automatic project migration might fail when you try to open the project in a later ControlDesk version.

Experiments already contained in the project If your project already contains experiments, they are displayed here (no changes possible).

< Back Opens the previous dialog. This button is disabled if a project is currently loaded.

Next > Opens the next dialog. This button is disabled or invalid as long as the Name of the experiment edit field is empty or invalid.

Finish Lets you finish experiment creation without having to add a platform/device to the experiment. This button is disabled or invalid as long as the Name of the experiment edit field is empty or invalid.

Cancel Closes the dialog without saving any of your settings.

Add Platform/Device dialog

To specify the type and name of the platform/device to be added to the experiment.

Platform/device name Displays the default name for the platform/device, or lets you specify a different name to be used for the platform/device in the current experiment.

Note

The platform/device name must be unique within the experiment. It must differ from other platform/device names by at least one character. Differences in upper/lower case are not sufficient.

Supported Platform/Device Types list Displays all the platform/device types supported by ControlDesk and lets you select the platform/device type to be added to the experiment. The list is subdivided according to use cases.

You can limit the selection in the platform/device types list to platforms/devices you have a valid license for. Select Show Licensed Platforms/Devices Only from the list's context menu for this.

Available Platforms/Devices list Displays the available platforms/devices in drop-down lists according to platform/device types, and lets you select one of them to be assigned or added to the current experiment. The drop-down list for each platform/device type contains the platforms that were registered in the system, but are not assigned to any experiment in the project yet. If no assignable platform/device is available, the list is empty, and only the  --- entry is displayed.

Tip

If there are specifics to be noticed when adding a platform/device of a certain type (e.g., a device can be configured only after being added to the active experiment), ControlDesk displays a symbol next to the platform/device type. Move the mouse pointer over the symbol to open a tooltip with detailed information.

Name displayed in experiment Displays the name under which the platform/device is displayed in the experiment. The name consists of the following elements: platform/device name [platform/device type, assignment information].

Configure Opens the configuration wizard for the selected platform.

For devices, Multiprocessor System platforms and registered platforms, the Configure button is disabled. These platforms/devices can only be configured

after being added to the active experiment. To configure the platform/device later on or change the settings, choose Configure Platform/Device (ControlDesk Platform Management 🕮) from the platform's/device's context menu.

< Back Opens the previous dialog.

Next > Opens the next dialog.

Finish Confirms the adding of the platform/device to the experiment and closes the dialog without selecting a variable description for the platform/device. To select a variable description for the platform/device, click Next > instead.

Cancel Closes the dialog without adding a platform/device to the experiment.

Select Variable Description dialog

To select a variable description for a platform/device.

Variable description list Displays the variable descriptions (A2L, DBC, SDF, and LDF files) available in the `Variable Descriptions` folder of the ControlDesk project.

Variable descriptions that do not match the selected platform/device type or that are already assigned to another platform/device of the current experiment are not displayed in the list.

Import/Import from file Opens a dialog for you to import a variable description file (A2L, DBC, SDF, LDF file, …). The imported variable description file is added to the list of variable descriptions in the `Variable Descriptions` folder of the ControlDesk project.

If an AUTOSAR or FIBEX file contains the description of multiple clusters, ControlDesk lists all contained clusters. From this list, you can select the one that you want to import. Incompatible clusters are grayed out.

Tip

In the ControlDesk Options dialog, you can specify import options for variable descriptions in the SDF and FIBEX formats. Refer to Variables Page (ControlDesk Variable Management 🕮).

Remove (Available in the context menu of the variable descriptions in the list) Removes the selected variable description from the list of variable descriptions. Only variable descriptions that are not assigned to any platform/device in the project can be removed.

Variable description info field Displays information on the variable description selected in the variable description list.

< Back Opens the previous dialog.

Next > Opens the next dialog.

Finish Confirms the selection and closes the dialog without adding an ECU Image file to the selected variable description. To add an ECU Image file to the selected variable description, click Next > instead.

Cancel Closes the dialog without saving any of your settings.

Select ECU Image File dialog

(Available only for measurement and calibration devices) To specify an ECU Image file for the selected variable description.

Select ECU Image file Enter the path and name of the ECU Image file (HEX, MOT, S19, … file) or select it via the BROWSE button.

ControlDesk can handle ECU Image files that contain code and data or only data.

< Back Opens the previous dialog.

Finish Confirms the specified configuration and closes the dialog.

Cancel Closes the dialog without saving any of your settings.

Related topics

HowTos

How to Add a Platform/Device to an Experiment (ControlDesk Platform Management 🕮)
How to Add a Variable Description to a Platform/Device (ControlDesk Variable Management 🕮)
How to Define a Project....19
How to Define an Experiment....21

Recently used Projects and Experiments

Access

You can access recently used projects and experiments via:

Ribbon	File – Recently Used
Context menu of	None
Shortcut key	None
Icon	None
Others	Start Page

Purpose

To open one of the most recent experiments that were open in ControlDesk.

Result

If another experiment is already open, it and the project it belongs to are closed. Then the experiment you selected is opened and activated.

When the experiment is opened, ControlDesk remains in offline calibration mode.

Recent Projects and Experiments

List of experiments Lets you select one of the most recent experiments that were open in ControlDesk.

Open (Available from the context menu of list items) To open the selected experiment. You can also simply open an experiment by left-clicking it in the list.

Clear Recent Projects + Experiments List (Available from the context menu of list items) To clear the list of recently opened projects + experiments.

Remove from List (Available from the context menu of list items) To remove the selected experiment from the list of recently opened projects + experiments.

Size of recent experiments list Lets you specify the maximum number of list entries. You can specify a value in the range 4 ... 100.

Reset Sort Direction (Available from the context menu of the column header of the list of experiments) You can click the column headers to sort the experiments in ascending or descending order according to a column. To remove this sorting you can reset the sort direction to its default, which is according to the time the experiments were last opened, starting with the most recently opened experiment.

Visible Columns - Path/Opened/Modified/Version (Available from the context menu of the column header of the list of experiments) Lets you specify whether to display:

- The path to the experiment folder
- The point in time when the experiment was opened last
- The point in time when the experiment was modified last

Recent Files (not available on the Start page)

List of files Lets you select one of the most recent files that were open in ControlDesk. The list contains files that are not part of a project / experiment.

Size of recent files list Lets you specify the maximum number of list entries. You can specify a value in the range 4 ... 100.

Related topics

Basics

Remove from List.....81

Refresh

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – project

Shortcut key	None
Icon	None

Purpose

To renew the folder structure shown in the Project controlbar.

Result

The directory structure and included files are re-read and the view of the structure shown in the Project controlbar is refreshed. Newly added documents are shown.

Related topics

HowTos

How to Add External Documents to a Project or Experiment..... 29

Remove (from Project)

Access

You can access this command via:

Ribbon	None
Context menu of	▪ Project controlbar – inactive experiments ▪ Project controlbar – items of the active experiment (but not the active experiment itself, and not any active variable description and no data sets connected to either the working or reference page)
Shortcut key	`Delete`
Icon	None

Note

This command is not available in operator mode.

Purpose

To remove an item from the currently loaded project.

Description

The item is removed irretrievably from the project.

Tip

You cannot remove a whole project this way. You can delete a whole project via the context menu of the **Select an Experiment** dialog when you open a project.

Remove from List

Access

You can access this command via:

Ribbon	None
Context menu of	List items in the **Recently used Projects and Experiments** list
Shortcut key	None
Icon	None

Purpose

To remove the selected experiment from the list of recently opened projects + experiments.

Result

The selected experiment is removed from the list of recently opened projects + experiments.

Related topics

References

Recently used Projects and Experiments....78

Rename (Project/Experiment/Group)

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – project, experiment, group

Shortcut key	**F2**
Icon	None

Note

This command is not available in operator mode.

Purpose

To rename the selected item.

Description

If you select a project or an active experiment, ControlDesk closes it, before it is renamed. Afterwards, it is reloaded.

Result

For a group, the name of the item is displayed in an edit field that lets you enter the new name.

For a project or an experiment, one of the following **Rename** dialogs opens.

Rename Experiment dialog

Enter a new name for Lets you enter a new name for the experiment. The name must be unique within the project, and must not contain any of the following characters: `* ? | < > : / \ "`.

Note

Do not specify an experiment name that is equal to the name of a file folder within the project that the experiment belongs to. Otherwise, automatic project migration might fail when you try to open the project in a later ControlDesk version.

Rename Project dialog

Enter a new name for Lets you enter a new name for the project. The name must be unique within the selected project root directory and must not cause conflicts with existing folders. The name must not contain any of the following characters: `* ? | < > : / \ "`.

Save As (Backstage View)

Access

You can access this ribbon group via:

Ribbon	**File**
Context menu of	None

Shortcut key	None
Icon	None

Purpose

To provide access to commands that allow you, for example, to save a project under a new name.

Description

You have access to commands such as:

- Save as (Project/Experiment) on page 85
- Backup Project on page 50
- Export (Experiment) on page 55

Save As / Rename Dialog

Access

A **Save As** or **Rename** dialog opens for a project or an experiment if one of the following commands is invoked:

- Save as (Project/Experiment) on page 85
- Rename (Project/Experiment/Group) on page 81

Purpose

To rename a project or experiment or to save it under a new name.

Depending on the selected command one of the following dialogs opens:

- Rename Experiment dialog on page 83
- Rename Project dialog on page 84
- Save Experiment As dialog on page 84
- Save Project As dialog on page 84

Rename Experiment dialog

Enter a new name for Lets you enter a new name for the experiment. The name must be unique within the project, and must not contain any of the following characters: `* ? | < > : / \ "`.

Note

Do not specify an experiment name that is equal to the name of a file folder within the project that the experiment belongs to. Otherwise, automatic project migration might fail when you try to open the project in a later ControlDesk version.

Rename Project dialog

Enter a new name for Lets you enter a new name for the project. The name must be unique within the selected project root directory and must not cause conflicts with existing folders. The name must not contain any of the following characters: * ? | < > : / \ ".

Save Experiment As dialog

Enter a new name for Lets you enter a new name for the experiment. The experiment name must be unique in the active project. The name must not contain any of the following characters: * ? | < > : / \ ".

Save Project As dialog

Project name Lets you enter a new name for the project. The name must be unique in the selected project root directory. The name must not contain any of the following characters: * ? | < > : / \ ".

Project root Lets you select the project root directory.

	Opens the ControlDesk Options dialog with the Project page.

Related topics

References

Rename (Project/Experiment/Group) 81
Save as (Project/Experiment) 85

Save Project + Experiment / Save

Access

This command is available only if a project is open. You can access it via:

Ribbon	File
Context menu of	▪ Project controlbar – project ▪ Project controlbar – active experiment
Shortcut key	**Ctrl+Shift+S**
Icon	

Purpose

To save the loaded project and the active experiment.

Description

Tip

When you save a project last saved with ControlDesk 7.3 or earlier in ControlDesk 7.4, the project cannot be loaded with ControlDesk 7.3 or earlier. A backup of the original project is stored in the project root folder for this reason.

Result

ControlDesk saves the project.

Related topics

References

Save as (Project/Experiment)..... 85

Save as (Project/Experiment)

Access

You can access this command via:

Ribbon	File - Save As (project)
Context menu of	▪ Project controlbar – project ▪ Project controlbar – experiment
Shortcut key	None
Icon	

Note

This command is not available in operator mode.

Purpose

To save a project or an experiment under a new name.

Result

A Save As dialog opens for you to save the selected project or experiment under a new name.

Description

If you save a project or an active experiment, ControlDesk reloads it, then saves it under a new name.

> **Tip**
>
> When you save a project last saved with ControlDesk 7.3 or earlier in ControlDesk 7.4, the project cannot be loaded with ControlDesk 7.3 or earlier. A backup of the original project is stored in the project root folder for this reason.

Depending on the selected item, one of the following dialogs opens.

Save Experiment As dialog

Enter a new name for Lets you enter a new name for the experiment. The experiment name must be unique in the active project. The name must not contain any of the following characters: `* ? | < > : / \ "`.

Save Project As dialog

Project name Lets you enter a new name for the project. The name must be unique in the selected project root directory. The name must not contain any of the following characters: `* ? | < > : / \ "`.

Project root Lets you select the project root directory.

	Opens the ControlDesk Options dialog with the Project page.

Sort - Ascending

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – folder
Shortcut key	None
Icon	None

Purpose

To sort the selected folder or folder group alphabetically in ascending order.

Result

The selected folder or folder group is sorted alphabetically in ascending order.

Related topics

References

Add Folder....48
Sort - Descending....87

Sort - Descending

Access

You can access this command via:

Ribbon	None
Context menu of	Project controlbar – folder
Shortcut key	None
Icon	None

Purpose

To sort the selected folder or folder group alphabetically in descending order.

Result

The selected folder or folder group is sorted alphabetically in descending order.

Related topics

References

Add Folder....48
Sort - Ascending....86

Automation

Where to go from here

Information in this section

Programming ControlDesk Automation

Where to go from here

Information in this section

Information in other sections

Tool Automation Demos (ControlDesk Automation 🕮)
Demonstrate how to automate ControlDesk and use ControlDesk events.

Automating Project and Experiment Handling

Introduction

The main tasks of ControlDesk's project management are to define projects and experiments, and to open and activate them.

Unless otherwise indicated, the program listings below consist of excerpts from the `ProjectHandling.py` demo script. Keep in mind that these excerpts are not runnable by themselves.

Note

Initializing basic access to ControlDesk via the `Dispatch` class is necessary in all the automation scripts that are described in this and the following chapters on ControlDesk tool automation. It is omitted in the scripts for the sake of readability.
For details on initializing basic access to ControlDesk via the `Dispatch` class, refer to Automating the Start of ControlDesk (ControlDesk Automation 🕮).

Specifying a new project root directory

The following listing shows how to create a new project root directory called `DemoRoot` in your temporary system folder.

```
# Import: The tempfile module is used to access the temporary system folder.
# Import: The os module is used to access the dSPACE root directory.
import tempfile
import os
# Constant: A new root path in the temporary system folder.
PROJECTROOTPATH = os.path.join(tempfile.gettempdir(), "DemoRoot")
(...)
class MainDemoController(object):
    def __init__(self):
        # The project root where most parts of the demo are executed.
        self.DemoRoot = None
    (...)
    def Initialize(self):
        # Check if the project root doesn't already exist.
        if not self.ControlDeskApplication.ProjectRoots.Contains(PROJECTROOTPATH):
            # Create Demo Root Object.
            self.DemoRoot = self.ControlDeskApplication.ProjectRoots.Add(PROJECTROOTPATH)
        else:
            # Get demo root.
            self.DemoRoot = self.ControlDeskApplication.ProjectRoots.Item(PROJECTROOTPATH)
        # Activate new project root.
        self.DemoRoot.Activate()
```

Creating, closing, and deleting a project

The following listing shows how to create a project with the name `NewTestProject` and then close and delete it.

```
# The name of a project used during the demo.
PROJECTNAME = "NewTestProject"
(...)
class MainDemoController(object):
    (...)
    def CreateAndDeleteNewProject(self):
        # Create a new project with the given name.
        NewProject = self.ControlDeskApplication.Projects.Add(PROJECTNAME)
        (...)
        # Close project.
        NewProject.Close(False)
        # Delete project.
        self.ControlDeskApplication.ActiveProjectRoot.Projects.Item(PROJECTNAME).Remove(True)
```

Tip

You can use ControlDesk events to create a project automatically when ControlDesk starts up. Refer to Creating a Project When ControlDesk Starts Up (ControlDesk Customization 🕮).

Accessing an existing project

The following listing shows how to search the project root folders for the *CalDemo* project and load it.

```
# The name of the CalDemo project.
CALDEMOPROJECTNAME = "CalDemo"
(...)
class MainDemoController(object):
    (...)
    def LoadProject(self):
        # Check if the project exists.
        # Initialize root.
        CalDemoRoot = None
        # Iterate over all project roots to find the root with the project searched for.
        for root in self.ControlDeskApplication.ProjectRoots:
            # Check if the project was found.
            if root.Projects.Contains(CALDEMOPROJECTNAME):
                # Activate project root.
                root.Activate()
                # Set FoundRoot variable to true.
                CalDemoRoot = root
                # Stop the for loop, so the searched project root is stored in the Root variable.
                break
        # If the CalDemo project was found, load it.
        if CalDemoRoot:
            # Get project with given name.
            projectItem = CalDemoRoot.Projects.Item(CALDEMOPROJECTNAME)
            # Open project.
            activeProject = projectItem.Open()
```

Note

To modify a project, for example, to add experiments, it must be the active project.

After getting an existing project you must therefore activate it via the `Open` method as shown in the example above:

```
projectItem = ...
activeProject = projectItem.Open()
```

There is one exception to this rule: If you add a new project it is automatically active.

Creating a new experiment

The following listing shows how to add three new experiments to a project and activate the first (excerpts from the `ExperimentHandling.py` demo script).

```
# Constant: The name of a project used during the demo.
PROJECTNAME = "ExperimentHandlingDemoProject"
(...)
class MainDemoController(object):
    (...)
    def Initialize(self):
        (...)
        # Get project list.
        projects = self.ControlDeskApplication.Projects
        # Create a new project with the given name.
        self.Demoproject  = Projects.Add(PROJECTNAME)
        (...)
    def CreateThreeExperiments(self):
        # Create new experiment 'NewExpermentOne'.
        self.Demoproject.Experiments.Add("NewExperimentOne")
        # Create new experiments 'NewExpermentTwo'.
        self.Demoproject.Experiments.Add("NewExperimentTwo")
        # Create new experiments 'NewExperimentThree'.
        self.Demoproject.Experiments.Add("NewExperimentThree")
        # Activate experiment 'NewExperimentOne'.
        newExperiment = self.ControlDeskApplication.ActiveProject.Experiments.Item("NewExperimentOne")
        newExperiment.Activate()
```

Accessing an existing experiment

The following listing shows how to activate an existing experiment (excerpts from the `ExperimentHandling.py` demo script).

```
# Constant: The name of a project used during the demo.
PROJECTNAME = "ExperimentHandlingDemoProject"
(...)
class MainDemoController(object):
    (...)
    def OpenSpecificExperiment(self):
        (...)
        # Get the project.
        project = self.ControlDeskApplication.Projects.Item(PROJECTNAME)
        # Open the project.
        activeProject = project.Open()
        # Get an experiment.
        experiment = activeProject.Experiments.Item("NewExperimentOne")
        # Activate the experiment.
        activeExperiment = experiment.Activate()
```

Note

To work on an experiment, for example, to add layouts to it, it must be the *active* experiment.

After getting an existing experiment, you must therefore activate it via the `Activate` method as shown in the example above:

```
experiment = ...
activeExperiment = experiment.Activate()
```

There is one exception to this rule: If you add a new experiment it is automatically active.

Importing and exporting, renaming and removing of experiments

The following listing shows how to import/export, rename and remove an existing experiment (excerpts from the `ExperimentHandling.py` demo script).

```
# The experiment export path.
EXPERIMENT_EXPORT_PATH = os.path.join(PROJECTROOTPATH,"ExportedExperiment.DSA")
(...)
class MainDemoController(object):
    (...)
    def CommandsOnExperimentLevel(self):
        (...)
        # Export active experiment.
        self.ControlDeskApplication.ActiveExperiment.Export(EXPERIMENT_EXPORT_PATH)
        # Get the second experiment of the active project.
        secondExperiment = self.ControlDeskApplication.ActiveProject.Experiments.Item(1)
        # Activate second experiment.
        secondExperiment.Activate()
        # Rename active experiment.
        self.ControlDeskApplication.ActiveExperiment.Rename("RenamedExperiment")
        # Get first experiment.
        firstExperiment = self.ControlDeskApplication.ActiveProject.Experiments.Item(0)
        # Remove first experiment.
        firstExperiment.Remove(True)
        (...)
        # Import experiment from CalDemo.
        newProject.Experiments.Import(EXPERIMENT_EXPORT_PATH, "ImportedExperiment")
```

Related topics

Basics

Basics on Projects and Experiments.....10

Project and Experiment Handling

Where to go from here

Information in this section

Project and Experiment Handling Interfaces

Introduction

The main tasks of ControlDesk's project management are to define projects and experiments, and to open and activate them.

Project-Related Interfaces

Introduction

The main tasks of ControlDesk's project management are to define and open projects and handle files controlled by the project.

Description

The object that implements the *IXaProjectRoots* interface is used to handle the project root elements. A project root element is represented by a directory on your file system to which ControlDesk saves all the experiments and documents of a project. A project root element implements the *IXaProjectRoot* interface.

The Projects collection which implements the *IXaProjects* interface contains the projects in a project root element. A Project is used to handle the contained experiments and implements the *IXaProject* interface. Only one project can be active at the same time. The active project implements the *IXaActiveProject* interface.

The Project Management object implements the *IXaProjectManagement* interface. Use the properties defined in the interface to manipulate the settings of the Project Management.

Related interfaces

Interface	Description
IXaProjectRoots (refer to ProjectRoots / IXaProjectRoots <<Collection>> (ControlDesk Automation 🕮))	This interface is to access the project roots of the application.
IXaProjectRoot (refer to ProjectRoot / IXaProjectRoot <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a project root.
IXaProjects (refer to Projects / IXaProjects <<Collection>> (ControlDesk Automation 🕮))	This interface is to access the projects contained in a project root.
IXaProject (refer to Project / IXaProject <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a project, not necessarily active.
IXaActiveProject (refer to ActiveProject / IXaActiveProject <<Interface>> (ControlDesk Automation 🕮))	This interface is to access the currently active project.
IXaProjectManagement (refer to ProjectManagement / IXaProjectManagement <<Interface>> (ControlDesk Automation 🕮))	This interface is to access the variables management.
IXaFiles (refer to Files / IXaFiles <<Collection>> (ControlDesk Automation 🕮))	This interface is to access the files of a project or experiment.
IXaFile (refer to File / IXaFile <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a file of a project or experiment.
IEmEventContext (refer to EventContext / IEmEventContext <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a tool automation event context.

Related documentation

Topic	Description
Automating Project and Experiment Handling on page 90	The main tasks of ControlDesk's project management are to define projects and experiments, and to open and activate them.

Experiment-Related Interfaces

Introduction

The main tasks of ControlDesk's experiment handling are to define and activate experiments and handle files controlled by the experiment.

Description

The collection of experiments in a project contains all the experiments and implements the *IXaExperiments* interface. The Experiment object implements the *IXaExperiment* interface. Only one experiment can be active at the same time. The active experiment implements the *IXaActiveExperiment* interface.

Related interfaces

Interface	Description
IXaExperiments (refer to Experiments / IXaExperiments <<Collection>> (ControlDesk Automation 🕮))	This interface is to access the experiments of a project.
IXaExperiment (refer to Experiment / IXaExperiment <<Interface>> (ControlDesk Automation 🕮))	This interface is to access an experiment, which is not necessarily active.
IXaActiveExperiment (refer to ActiveExperiment / IXaActiveExperiment <<Interface>> (ControlDesk Automation 🕮))	This interface is to access the currently active experiment.
IXaFiles (refer to Files / IXaFiles <<Collection>> (ControlDesk Automation 🕮))	This interface is to access the files of a project or experiment.
IXaFile (refer to File / IXaFile <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a file of a project or experiment.
IEmEventContext (refer to EventContext / IEmEventContext <<Interface>> (ControlDesk Automation 🕮))	This interface is to access a tool automation event context.

Related documentation

Topic	Description
Automating Project and Experiment Handling on page 90	The main tasks of ControlDesk's project management are to define projects and experiments, and to open and activate them.

Troubleshooting

Where to go from here

Information in this section

Problem when Migrating Projects

Problem

When you open a project last saved with ControlDesk 7.0 or earlier in ControlDesk 7.1 or later, automatic project migration might fail.

Description

Automatic migration of projects might fail when the name of an experiment in that project is equal to the name of a file folder within the project.

Solution

To solve the problem:

1. Rename the experiment in ControlDesk 7.0 or earlier.
2. Open the project in ControlDesk 7.1 or later to initiate project migration.

Related topics

Basics

Basics on Migrating from Prior Versions of ControlDesk (ControlDesk Introduction and Overview 🕮)

References

Open Project + Experiment....64
Rename (Project/Experiment/Group)....81

Problem with Long Project Loading Times

Problem

A project needs an extremely long time to load.

Description

The content of each variable description you add to a project is stored in a Variable Database (VDB) file in the project's `Variable Descriptions` folder. The Variable Database grows with the project and some data becomes obsolete. This can lead to long project loading times.

Solution

To reduce the loading time of a project, you can compress the content of the Variable Database file. This removes unnecessary data from the database.

Open ControlDesk's **Interpreter** and type the following command:

```
Application.VariablesManagement.Compress(<project>)
```

Replace `<project>` with the path to the project's CDP file or with the project automation object. See the following examples:

- Using the path to the CDP file:

```
Application.VariablesManagement.Compress("E:\Projects\Project_005\
                                         Project_005.cdp")
```

- Using the project automation object:

```
ProjectRoot = Application.ProjectRoots.Item("E:\Projects")
Project = ProjectRoot.Projects.Item("Project_005")
Application.VariablesManagement.Compress(Project)
```

Note

The Variable Database you want to compress must not be opened in ControlDesk or another tool, such as the Variable Editor.

If exceptions occur, ControlDesk displays them in the **Interpreter**. For more information, e.g., the compression success, check the dSPACE Log file via **View – Show - dSPACE Log**.

Related topics

HowTos

How to Open a Project and Experiment..... 25

Limitations

Limitations for Project and Experiment Management

Project import supported for the last seven ControlDesk versions

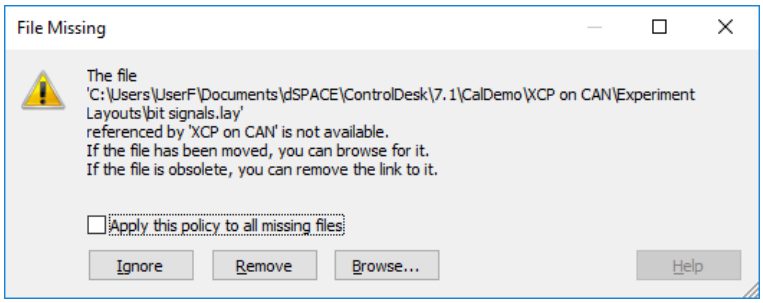
ControlDesk supports direct opening only for projects *last saved with one of the previous seven ControlDesk versions*.

To work with a ControlDesk project *last saved with a version earlier than the last seven ones*, open the project in one or more intermediate ControlDesk versions.

Closing an experiment without saving changes

When you remove an element such as a layout from an experiment, ControlDesk asks you whether to delete the related file from the hard disk. If you confirm, close the experiment without saving it, and then reopen the experiment, the element is *not* available in the experiment although you did not save the change you made to this experiment. This problem also occurs when you perform the steps above via ControlDesk's automation interface.

You get a warning like this:



Docked window state not stored

If your experiment contains windows in the *docked* window state, the state is not saved when you save the experiment. When you reopen the experiment, previously docked windows are floating in the main window (working area) of ControlDesk.

Related topics

Basics

Basics on Migrating from Prior Versions of ControlDesk (ControlDesk Introduction and Overview 🕮)

Glossary

Introduction

Briefly explains the most important expressions and naming conventions used in the ControlDesk documentation.

Where to go from here

Information in this section

Numerics

3-D Viewer An instrument for displaying items in a 3-D environment.

A

A2L file A file that contains all the relevant information on measurement and calibration variables in an ECU application and the ECU's communication interface(s). This includes information on the variables' memory addresses and conversion methods, the memory layout and data structures in the ECU as well as interface description data (IF_DATA).

Acquisition An object in the Measurement Configuration controlbar that specifies the variables to be measured and their measurement configuration.

Active variable description The variable description that is currently active for a platform/device. Multiple variable descriptions can be assigned to one platform/device, but only one of them can be active at a time.

Additional write variable A scalar parameter or writable measurement variable that can be connected to an instrument in addition to the main variable. When the value of the main variable changes, the changed value is also applied to all the additional write variables connected to the instrument.

Airspeed Indicator An instrument for displaying the airspeed of a simulated aircraft.

Altimeter An instrument for displaying the altitude of a simulated aircraft.

Animated Needle An instrument for displaying the value of a connected variable by a needle deflection.

Application image An image file that contains all the files that are created when the user builds a real-time application. It particularly includes the variable

description (SDF) file. To extend a real-time application, ControlDesk lets the user create an updated application image from a data set. The updated application image then contains a real-time application with an additional set of parameter values.

Artificial Horizon An instrument displaying the rotation on both the lateral and the longitudinal axis to indicate the angle of pitch and roll of a simulated aircraft. The Artificial Horizon has a pitch scale and a roll scale.

Automatic Reconnect Feature for automatically reconnecting to platform/device hardware, for example, when the ignition is turned off and on, or when the physical connection between the ControlDesk PC and the ECU is temporarily interrupted.

If the feature is enabled for a platform/device and if the platform/device is in the 'unplugged' state, ControlDesk tries to re-establish the logical connection to the platform/device hardware. After the logical connection is re-established, the platform/device has the same state as before the unplugged state was detected. A measurement started before the unplugged state was detected is resumed.

Automation A communication mechanism that can be used by various programming languages. A client can use it to control a server by calling methods and properties of the server's automation interface.

Automation script A script that uses automation to control an automation server.

Axis point object Common axis

B

Bar An instrument (or a value cell type of the Variable Array) for displaying a numerical value as a bar deflection on a horizontal or vertical scale.

Bitfield A value cell type of the Variable Array for displaying and editing the source value of a parameter as a bit string.

Bookmark A marker for a certain event during a measurement or recording.

Browser An instrument for displaying HTML and TXT files. It also supports Microsoft Internet Explorer© plug-ins that are installed on your system.

Bus communication replay A feature of the Bus Navigator that lets you replay logged bus communication data from a log file. You can add replay nodes

to the Bus Navigator tree for this purpose. You can specify filters to replay selected parts of the logged bus communication.

Bus configuration A configuration of all the controllers, communication matrices, and messages/frames/PDUs of a specific communication bus such as CAN. ControlDesk lets you display and experiment with bus configurations in the Bus Navigator.

Bus connection A mode for connecting dSPACE real-time hardware to the host PC via bus. The list below shows the possible bus connections:

- dSPACE real-time hardware installed directly in the host PC
- dSPACE real-time hardware installed in an expansion box connected to the host PC via dSPACE link board

Bus Instrument An instrument available for the Bus Navigator. It can be configured for different purposes, for example, to display information on received messages (RX messages) or to manipulate and transmit messages (TX messages). The instrument is tailor-made and displays only the message- and signal-specific settings which are enabled for display and/or manipulation by ControlDesk during run time.

Bus logging A feature of the Bus Navigator that lets you log raw bus communication data. You can add logger nodes on different hierarchy levels of the Bus Navigator tree for this purpose. You can specify filters to log filtered bus communication. The logged bus communication can be replayed.

Bus monitoring A feature of the Bus Navigator that lets you observe bus communication. You can open monitoring lists and add monitor nodes on different hierarchy levels of the Bus Navigator tree for this purpose. You can specify filters to monitor filtered bus communication.

Bus Navigator A controlbar for handling bus messages, such as CAN messages, LIN frames, and Ethernet packets.

Bus statistics A feature of the Bus Navigator that lets you display and log statistical information on the bus load during bus monitoring.

Bypassing A method for replacing an existing ECU function by running a new function.

C

Calculated variable A scalar variable that can be measured and recorded, and that is derived from one or more *input variables*.

The following input variable types are supported:

- Measurement variables
- Single elements of measurement arrays or value blocks
- Scalar parameters, or existing calculated variables

The value of a calculated variable is calculated via a user-defined *computation formula* that uses one or more input variables.

Calculated variables are represented by the  symbol.

CalDemo ECU A demo program that runs on the same PC as ControlDesk. It simulates an ECU on which the Universal Measurement and Calibration (XCP) protocol and the Unified Diagnostic Services (UDS) protocol are implemented.

The CalDemo ECU allows you to perform parameter calibration, variable measurement, and ECU diagnostics with ControlDesk under realistic conditions, but without having to have a real ECU connected to the PC. Communication between the CalDemo ECU and ControlDesk can be established via XCP on CAN or XCP on Ethernet, and UDS on CAN.

Tip

If communication is established via XCP on Ethernet, the CalDemo ECU can also run on a PC different from the PC on which ControlDesk is running.

The memory of the CalDemo ECU consists of two areas called memory page. Each page contains a complete set of parameters, but only one page is accessible by the CalDemo ECU at a time. You can easily switch the memory pages of the CalDemo ECU to change from one parameter to another in a single step.

Two ECU tasks run on the CalDemo ECU:

- ECU task #1 runs at a fixed sample time of 5 ms. In ControlDesk's **Measurement Configuration**, ECU task #1 is related to the time-based **5 ms**, **10 ms**, **50 ms** and **100 ms** measurement rasters of the CalDemo ECU.
- ECU task #2 has a variable sample time. Whenever the CalDemo ECU program is started, the initial sample time is 5 ms. This can then be increased or decreased by using the **dSPACE CalDemo** dialog.

  ECU task #2 is related to the **extEvent** measurement raster of the CalDemo ECU.

The CalDemo ECU can also be used to execute diagnostic services and jobs, handle DTCs and perform measurement and calibration via ECU diagnostics.

The CalDemo ECU program is run by invoking `CalDemo.exe`. The file is located in the `.\Demos\CalDemo` folder of the ControlDesk installation.

Calibration Changing the parameter values of real-time applications or ECU applications.

Calibration memory segment Part of the memory of an ECU containing the calibratable parameters. Memory segments can be defined as `MEMORY_SEGMENT` in the A2L file. ControlDesk can use the segments to evaluate the memory pages of the ECU.

ControlDesk lets you perform the calibration of:

- Parameters inside memory segments
- Parameters outside memory segments
- Parameters even if no memory segments are defined in the A2L file.

CAN Bus Monitoring device A device that monitors the data stream on a CAN bus connected to the ControlDesk PC.

The CAN Bus Monitoring device works, for example, with PC-based CAN interfaces such as the DCI-CAN2 or the DCI-CAN/LIN1.

The device supports the following variable description file types:

- DBC
- FIBEX
- AUTOSAR system description (ARXML)

CANGenerator A demo program that simulates a CAN system, that is, it generates signals that can be measured and recorded with ControlDesk. The program runs on the same PC as ControlDesk.

The CANGenerator allows you to use the CAN Bus Monitoring device under realistic conditions, but without having to have any device hardware connected to the PC.

The CAN (Controller Area Network) protocol is used for communication between the CANGenerator and ControlDesk. However, since the CANGenerator runs on the same PC as ControlDesk, ControlDesk does not communicate with the device via a real CAN channel, but via a *virtual CAN channel* implemented on the host PC.

You can start the CAN generator program by running `CANGenerator.exe`. The file is located in the `.\Demos\CANGenerator` folder of the ControlDesk installation.

Capture A data packet of all the measurement variables assigned to a measurement raster. The packet comprises the data that results from a single triggering of the raster.

CCP Abbreviation of CAN Calibration Protocol. This protocol can be implemented on electronic control units (ECUs) and allows users to access ECUs with measurement and calibration systems (MCS) such as ControlDesk.

The basic features of CCP are:

- Read and write access to the ECU memory, i.e., providing access for calibration
- Synchronous data acquisition
- Flash programming for ECU development purposes

The CCP protocol was developed by ASAM e.V. (Association for Standardization of Automation and Measuring Systems e.V.). For the protocol specification, refer to http://www.asam.net.

The following device supports ECUs with an integrated CCP service:

- CCP device

CCP device A device that provides access to an ECU with CCP connected to the ControlDesk PC via CAN, for example, for measurement and calibration purposes via CCP (CAN Calibration Protocol).

Check Button An instrument (or a cell type of the Variable Array) for displaying whether the value of a connected variable matches predefined values or for writing a predefined value to a connected variable.

cmdloader A command line tool for handling applications without using the user interface of an experiment software.

Common axis A parameter that consists of a 1-dimensional array containing axis points. A common axis can be referenced by one or more curves and/or maps. Calibrating the data points of a common axis affects all the curves and/or maps referencing the axis.

Common axes are represented by the symbol.

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>`

Computation method A formula or a table that defines the transformation of a source value into a converted value (and vice versa). In addition to the computation methods defined in the variable description file, ControlDesk provides the *__Identity* computation method which means the converted and the source value are equal.

Connected A platform/device state defined by the following characteristics:

- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- A platform/device must be in the 'connected' state before it can change to the 'measuring/recording' or 'online calibration started' state.
- Online calibration is impossible. ControlDesk did not yet adjust the memory segments containing calibration data in the platform/device and on the corresponding hardware. Offline calibration is possible.
- Platform/device configuration is not possible. However, you can invoke platform/device configuration for a platform/device that is in the connected state. ControlDesk temporarily sets the platform/device to the disconnected state.

The 'connected' platform/device state is indicated by the icon.

Connection mode dSPACE real-time systems can be installed within the host PC or connected to the host via a bus interface and/or via Ethernet. When the Ethernet is being used, different network clients might exist. The connection type being used and, in the case of Ethernet, the network client being used, determine the dSPACE systems that can be accessed.

Control primitive A special diagnostic communication object for changing communication states or protocol parameters, or for identifying (ECU) variants.

Controlbar A window or pane outside the working area. Can be docked to an edge of the main window or float in front of it. A controlbar can contain a

document, such as a layout, or a tool, such as the **Bus Navigator**. It can be grouped with other controlbars in a window with tabbed pages.

ControlDesk The main version of ControlDesk for creating and running experiments, and for accessing dSPACE real-time hardware and VEOS. The functionality can be extended by optional software modules.

ControlDesk - Operator Version A version of ControlDesk that provides only a subset of functionality for running existing experiments. The functionality can be extended by optional software modules.

ControlDesk Bus Navigator Module An optional software module for ControlDesk for handling bus messages, such as CAN, LIN, and FlexRay messages, frames, and PDUs and Ethernet packets.

ControlDesk ECU Diagnostics Module An optional software module for ControlDesk that facilitates the calibration and validation of ECU diagnostic functions.

ControlDesk ECU Interface Module An optional software module for ControlDesk for calibration and measurement access to electronic control units (ECUs). The module is also required for calibration and measurement access to virtual ECUs (V-ECUs) used in SIL testing scenarios.

ControlDesk Signal Editor Module An optional software module for ControlDesk for the graphical definition and execution of signal generators for stimulating model variables of real-time/offline simulation applications.

Controller board Single-board hardware computing the real-time application. Contains a real-time processor for fast calculation of the model and I/O interfaces for carrying out the control developments.

Conversion table A table that specifies the value conversion of a source value into a converted value. In the case of verbal conversion, the converted value is a string that represents one numerical value or a range of numerical values.

Conversion type The type of a computation method, for example a linear function or a verbal computation method.

Curve A parameter that consists of

- A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis.
- Another 1-dimensional array containing data points. The curve assigns one data point to each axis point.

Curves are represented by the  symbol.

D

DAQ module A hardware module for the acquisition of physical quantities

Data Cursor One or two cursors that are used to display the values of selected chart positions in a Time Plotter or an Index Plotter.

Data logger An object in the Measurement Configuration controlbar that lets you configure a data logging.

Data logger signal list A list that contains the variables to be included in subsequent data loggings on real-time hardware.

Data logging The recording of data on dSPACE real-time hardware that does not require a physical connection between the host PC and the real-time hardware. In contrast to flight recording, data logging is configured in ControlDesk.

Data set A set of the parameters and their values of a platform/device derived from the variable description of the platform/device. There are different types of data sets:

- Reference data set
- Sub data set
- Unassigned data set
- Working data set

DCI-CAN/LIN1 A dSPACE-specific interface between the host PC and the CAN/CAN FD bus and/or LIN bus. The DCI-CAN/LIN1 transfers messages between the CAN-/LIN-based devices and the host PC via the universal serial bus (USB).

DCI-CAN2 A dSPACE-specific interface between the host PC and the CAN bus. The DCI-CAN2 transfers CAN and CAN FD messages between the CAN-based devices and the host PC via the universal serial bus (USB).

DCI-GSI2 Abbreviation of *dSPACE Communication Interface - Generic Serial Interface 2*. A dSPACE-specific interface for ECU calibration, measurement and ECU interfacing.

DCI-GSI2 device A device that provides access to an ECU with DCI-GSI2 connected to the ControlDesk PC for measurement, calibration, and bypassing purposes via the ECU's debug interface.

DCI-KLine1 Abbreviation of *dSPACE Communication Interface - K-Line Interface*. A dSPACE-specific interface between the host PC and the diagnostics bus via K-Line.

Debug interface An ECU interface for diagnostics tasks and flashing.

Default raster A platform-/device-specific measurement raster that is used when a variable of the platform/device is connected to a plotter or a recorder, for example.

Deposition definition A definition specifying the sequence in which the axis point values of a curve or map are deposited in memory.

Device A software component for carrying out calibration and/or measurement, bypassing, ECU flash programming, or ECU diagnostics tasks.

ControlDesk provides the following devices:

- Bus devices:
  - CAN Bus Monitoring device
  - Ethernet Bus Monitoring device
  - LIN Bus Monitoring device
- ECU Diagnostics device
- GNSS device
- Measurement and calibration devices:
  - CCP device
  - DCI-GSI2 device
  - XCP on CAN device
  - XCP on Ethernet device

Each device usually has a variable description that specifies the device's variables to be calibrated and measured.

Diagnostic interface Interface for accessing the fault memory of an ECU.

Diagnostic job (often called Java job) Programmed sequence that is usually built from a sequence of the diagnostic service. A diagnostic job is either a single-ECU job or a multiple-ECU job, depending on whether it communicates with one ECU or multiple ECUs.

Diagnostic protocol A protocol that defines how an ECU communicates with a connected diagnostic tester. The protocol must be implemented on the ECU and on the tester. The diagnostics database specifies the diagnostic protocol(s) supported by a specific ECU.

ControlDesk's ECU Diagnostics device supports CAN and K-Line as the physical layers for communication with an ECU connected to the ControlDesk PC. For information on the supported diagnostic protocols with CAN and K-Line, refer to Basics of ECU Diagnostics with ControlDesk (ControlDesk ECU Diagnostics 🕮).

Diagnostic service A service implemented on the ECU as a basic diagnostic communication element. Communication is performed by selecting a service, configuring its parameters, executing it, and receiving the ECU results. When a service is executed, a defined request is sent to the ECU and the ECU answers with a specific response.

Diagnostic trouble code (DTC) A hexadecimal index for the identification of vehicle malfunctions. DTCs are stored in the fault memory of ECUs and can be read by diagnostic testers.

Diagnostics database A database that completely describes one or more ECUs with respect to diagnostics communication. ControlDesk supports the ASAM MCD-2 D ODX database format, which was standardized by ASAM e.V. (Association for Standardisation of Automation and Measuring Systems e.V.). For the format specification, refer to http://www.asam.net.

Proprietary diagnostics database formats are not supported by ControlDesk.

Diagnostics Instrument An instrument for communicating with an ECU via the diagnostic protocol using diagnostic services, diagnostic jobs, and control primitives.

Disabled A platform/device state defined by the following characteristics:

- No logical connection is established between ControlDesk and the platform/device hardware.
- When a platform/device is disabled, ControlDesk does not try to establish the logical connection for that platform/device. Any communication between the platform/device hardware and ControlDesk is rejected.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'disabled' platform/device state is indicated by the  icon.

Disconnected A platform/device state defined by the following characteristics:

- No logical connection is established between ControlDesk and the platform/device hardware.
- When a platform/device is in the disconnected state, ControlDesk does not try to re-establish the logical connection for that platform/device.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'disconnected' platform/device state is indicated by the  icon.

Display An instrument (or a value cell type of the Variable Array) for displaying the value of a scalar variable or the text content of an ASCII variable.

Documents folder A standard folder for user-specific documents.

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

DS1006 Processor Board platform A platform that provides access to a DS1006 Processor Board connected to the host PC for HIL simulation and function prototyping purposes.

DS1007 PPC Processor Board platform A platform that provides access to a single multicore DS1007 PPC Processor Board or a DS1007 multiprocessor system consisting of two or more DS1007 PPC Processor Boards, connected to the host PC for HIL simulation and function prototyping purposes.

DS1104 R&D Controller Board platform A platform that provides access to a DS1104 R&D Controller Board installed in the host PC for function prototyping purposes.

DS1202 MicroLabBox platform A platform that provides access to a MicroLabBox connected to the host PC for function prototyping purposes.

DsDAQ service A service in a real-time application or offline simulation application (OSA) that provides measurement data from the application to the

host PC. Unlike the host service, the DsDAQ service lets you perform, for example, triggered measurements with complex trigger conditions.

The following platforms support applications that contain the DsDAQ service:

- DS1007 PPC Processor Board platform
- DS1202 MicroLabBox platform
- MicroAutoBox III platform
- SCALEXIO platform
- VEOS platform
- XIL API MAPort platform

dSPACE Calibration and Bypassing Service An ECU service for measurement, calibration, bypassing, and ECU flash programming. The dSPACE Calibration and Bypassing Service can be integrated on the ECU. It provides access to the ECU application and the ECU resources and is used to control communication between an ECU and a calibration and/or bypassing tool.

With the dSPACE Calibration and Bypassing Service, users can run measurement, calibration, bypassing, and flash programming tasks on an ECU via the DCI-GSI2. The service is also designed for bypassing ECU functions using dSPACE prototyping hardware by means of the RTI Bypass Blockset in connection with DPMEM PODs. The dSPACE Calibration and Bypassing Service allows measurement, calibration, and bypassing tasks to be performed in parallel.

dSPACE Internal Bypassing Service An ECU service for on-target prototyping. The dSPACE Internal Bypassing Service can be integrated in the ECU application. It lets you add additional functions to be executed in the context of the ECU application without the need for recompiling the ECU application.

dSPACE Log A collection of errors, warnings, information, questions, and advice issued by all dSPACE products and connected systems over more than one session.

dSPACE system A hardware system such as a MicroAutoBox III or SCALEXIO system on which the real-time application runs.

Duration trigger A trigger that defines a duration. Using a duration trigger, you can, for example, specify the duration of data acquisition for a measurement raster. A duration trigger can be used as a stop trigger.

E

ECU Abbreviation of *electronic control unit*.

ECU application A sequence of operations executed by an ECU. An ECU application is mostly represented by a group of files such as ECU Image files, MAP files, A2L files and/or software module description files.

ECU calibration interface Interface for accessing an ECU by either emulating the ECU's memory or using a communication protocol (for example, XCP on CAN).

ECU diagnostics Functions such as:

- Handling the ECU fault memory: Entries in the ECU's fault memory can be read, cleared, and saved.
- Executing diagnostic services and jobs: Users can communicate with an ECU via a diagnostic protocol using diagnostic services, diagnostic jobs, and control primitives.

ControlDesk provides the ECU Diagnostics device device to access ECUs for diagnostic tasks. Communication is via diagnostic protocols implemented on the ECUs.

ECU diagnostics with ControlDesk are completely based on Open Diagnostic Data Exchange (ODX), the ASAM MCD-2 D diagnostics standard.

ControlDesk provides the Fault Memory Instrument and the Diagnostics Instrument for ECU diagnostics tasks.

ECU Diagnostics device A device that provides access to ECUs connected to the ControlDesk PC via CAN or K-Line for diagnostics or flash programming purposes.

ControlDesk provides the *ECU Diagnostics v2.0.2* device, which supports the ASAM MCD-3 D V2.0.2 standard.

ControlDesk supports the following ODX database standards:

- ASAM MCD-2 D V2.0.1
- ASAM MCD-2 D V2.2.0 (ISO 22901-1)

ECU flash programming A method by which new code or data is stored in ECU flash memory.

ECU Image file A binary file that is part of the ECU application. It usually contains the code of an ECU application and the data of the parameters within the application. It can be stored as an Intel Hex (HEX) or Motorola S-Record (MOT or S19) file.

EESPort Configurations controlbar A controlbar for configuring error configurations.

Electrical error simulation Simulating electrical errors such as loose contacts, broken cables, and short-circuits, in the wiring of an ECU. Electrical error simulation is performed by the failure simulation hardware of an HIL simulator.

Electrical Error Simulation port (EESPort) An *Electrical Error Simulation port* (EESPort) provides access to a failure simulation hardware for simulating electrical errors in an ECU wiring according to the ASAM AE XIL API standard.

The configuration of the EESPort is described by a hardware-dependent *port configuration* and one or more *error configurations*.

Environment model A model that represents a part or all of the ECU's environment in a simulation scenario.

The environment model is a part of the simulation system.

Environment VPU The executable of an environment model built for the VEOS platform. An environment VPU is part of an offline simulation application (OSA).

Error An electrical error that is specified by:

- An error category
- An error type
- A load type

Error category The error category defines how a signal is disturbed. Which errors you can create for a signal depends on the connected failure simulation hardware.

Error configuration An XML file that describes a sequence of errors you want to switch during electrical error simulation. Each error configuration comprises error sets with one or more errors.

Error set An error set is used to group errors (pin failures).

Error type The error type specifies the way an error category – i.e., an interruption or short circuit of signals – is provided. The error type defines the disturbance itself.

Ethernet Bus Monitoring device A device that monitors the data stream on an Ethernet network connected to the ControlDesk PC.

The device supports the following variable description file type:

- AUTOSAR system description (ARXML)

Ethernet connection A mode for connecting dSPACE real-time hardware to the host PC via Ethernet. The list below shows the possible Ethernet connections:

- dSPACE real-time hardware installed in an expansion box connected to the host PC via Ethernet.
- MicroAutoBox II/III and MicroLabBox connected via Ethernet.

Ethernet decoding A feature of the Bus Navigator that lets you view protocol data and raw data of an Ethernet frame.

Event An event that is triggered by an action performed in ControlDesk.

Event context The scope of validity of event sources and events. There is one event handler code area for each event context.

Event handler Code that is executed when the related event occurs.

Event management Functionality for executing custom code according to actions triggered by ControlDesk.

Event source An object providing and triggering events. *LayoutManagement* is an example of an event source.

Event state State of an event. ControlDesk provides the following event states:

- No event handler is defined
- Event handler is defined and enabled
- Event handler is defined and disabled
- Event handler is defined, but no Python code is available
- Event handler is deactivated because a run-time error occurred during the execution of the Python code

Expansion box A box that hosts dSPACE boards. It can be connected to the host PC via bus connection or via network.

Experiment A container for collecting and managing information and files required for a parameter calibration and/or measurement task. A number of experiments can be collected in a project but only one of them can be active.

Extension script A Python script (PY or PYC file) that is executed each time ControlDesk starts up. An extension script can be executed for all users or user-specifically.

F

Failure insertion unit Hardware unit used with dSPACE simulators to simulate failures in the wiring of an ECU, such as broken wire and short circuit to ground.

Fault memory Part of the ECU memory that stores diagnostic trouble code (DTC) entries with status and environment information.

Fault Memory Instrument An instrument for reading, clearing, and saving the content of the ECU's fault memory.

Firmware update An update for the firmware installed in the board's flash memory. Firmware should be updated if it is older than required by the real-time application to be downloaded.

Fixed axis An axis with data points that are not deposited in the ECU memory. Unlike a common axis, a fixed axis is specified within a curve or map. The parameters of a fixed axis cannot be calibrated.

Fixed parameter A parameter that has a fixed value during a running simulation. Changing the value of a fixed parameter does not immediately affect the simulation results. The affect occurs only after you stop the simulation and

start it again. A fixed parameter is represented by an added pin in its symbol, for example:  .

Flash job A specific diagnostic job for flashing the ECU memory. A flash job implements the process control for flashing the ECU memory, such as initialization, security access, writing data blocks, etc.

Flight recording The recording of data on dSPACE real-time hardware that does not require a physical connection between the host PC and the real-time hardware. In contrast to data logging, flight recording is not configured in ControlDesk but via RTI and RTLib.

Frame An instrument for adding a background frame to a layout, for example, to visualize an instrument group.

G

Gauge An instrument for displaying the value of the connected variable by a needle deflection on a circular scale.

Gigalink module A dSPACE board for connecting several processor boards in a multiprocessor system. The board allows high-speed serial data transmission via fiber-optic cable.

GNSS data Positioning and timing data that is transmitted by a Global Navigation Satellite System (GNSS), such as GPS, GLONASS, or Galileo. GNSS receivers use this data to determine their location.

GNSS device A device that provides positioning data from a GNSS receiver (e.g., a serial GPS mouse) in ControlDesk.

ControlDesk provides the *GNSS (GPS, GLONASS, Galileo, ...)* device that supports various global navigation satellite systems.

GPX file An XML file that contains geodata, such as waypoints, routes, or tracks. In ControlDesk, you can import GPX files to visualize GNSS positioning data in a Map instrument.

Group A collection of variables that are grouped according to a certain criterion.

H

Heading Indicator An instrument displaying the heading direction of a simulated aircraft on a circular scale.

Host service A service in a real-time application that provides measurement data from the application to the host PC.

The following platforms support applications that contain the host service:

- DS1006 Processor Board platform
- DS1104 R&D Controller Board platform
- MicroAutoBox platform
- Multiprocessor System platform

Index Plotter A plotter instrument for displaying signals that are measured in an event-based raster (index plots).

Input quantity A measurement variable that is referenced by a common axis and that provides the input value of that axis.

Instrument An on-screen representation that is designed to monitor and/or control simulator variables interactively and to display data captures. Instruments can be arranged freely on layouts.

The following instruments can be used in ControlDesk:

- 3-D Viewer
- Airspeed Indicator
- Altimeter
- Animated Needle
- Artificial Horizon
- Bar
- Browser
- Bus Instrument
- Check Button
- Diagnostics Instrument
- Display
- Fault Memory Instrument
- Frame
- Gauge
- Heading Indicator
- Index Plotter
- Invisible Switch
- Knob
- Multistate Display
- Multiswitch
- Numeric Input
- On/Off Button

- Push Button
- Radio Button
- Selection Box
- Slider
- Sound Controller
- Static Text
- Steering Controller
- Table Editor
- Time Plotter
- Variable Array
- XY Plotter

Instrument Navigator A controlbar that displays a tree with all the instruments of the active layout and all the variables that are connected to them. The Instrument Navigator's main function is easy selection of instruments in complex layouts.

Instrument script A Python script used to extend the functionality of an instrument.

Instrument Selector A controlbar that provides access to ControlDesk's instruments. The instruments can be placed on a layout via double-click or drag & drop.

Interface description data (IF_DATA) An information structure, mostly provided by an A2L file, describing the type, features and configuration of an implemented ECU interface.

Internal Interpreter ControlDesk's built-in programming interface for editing, running and importing Python scripts. It contains an Interpreter controlbar where the user can enter Python commands interactively and which displays output and error messages of Python commands.

Interpreter controlbar A controlbar that can be used to execute line-based commands. It is used by the Internal Interpreter to print out Python standard error messages and standard output during the execution or import of Python scripts.

Invisible Switch An instrument for defining an area that is sensitive to mouse operations.

IOCNET IOCNET (I/O carrier network) is a dSPACE-specific high-speed serial communication bus that connects all the real-time hardware in a SCALEXIO system. IOCNET can also be used to build a multiprocessor system that consists of multiple SCALEXIO processor hardware components.

K

Knob An instrument for displaying and setting the value of the connected variable by means of a knob on a circular scale.

L

Label list A list of user-defined variables that can be used for saving connected variables, etc.

Layout A window with instrument s connected to variables of one or more simulation models.

Layout Navigator A controlbar that displays all opened layout s. It can be used for switching between layouts.

Layout script A Python script used to extend the functionality of a layout .

Leading raster The measurement raster that specifies the trigger settings for the Time Plotter display. The leading raster determines the time range that is visible in the plotter if a start and stop trigger is used for displaying the signals.

LIN Bus Monitoring device A device that monitors the data stream on a LIN bus connected to the ControlDesk PC.

The LIN Bus Monitoring device works, for example, with PC-based LIN interfaces.

The device supports the following variable description file types:

- LDF
- FIBEX
- AUTOSAR system description (ARXML)

Load type The load type specifies the option to disturb a signal with or without load rejection.

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>`

Logical link A representation of an ECU specified in the diagnostics database. A logical link contains information on the ECU itself, and all the information required for accessing it, such as the diagnostic protocol used for

communication between the ECU and ControlDesk. Each logical link is represented by a unique short name in the ODX database.

Look-up table A look-up table maps one or more input values to one output value. You have to differentiate between the following look-up table types:

- A 1-D look-up table maps one input value to one output value.
- A 2-D look-up table maps two input values to one output value.
- An n-D look-up table maps multidimensional table data with 3 or more input values to one output value.

Look-up table is a generic term for curves and maps.

M

Main variable A scalar variable that is visualized in an instrument that can be used to change parameter values. In addition to the main variable, additional write variables can also be connected to (but not visualized in) the same instrument. When you change the value of the main variable in an instrument, the changed value is also applied to all the additional write variables connected to that instrument.

Map A parameter that consists of

- A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis.
- A 1-dimensional array containing the axis points for the y-axis. This array can also be specified by a reference to a common axis.
- A 2–dimensional array containing data points. The map assigns one data point of the array to each pair of x-axis and y-axis points.

Maps are represented by the  symbol.

Map file A file that contains symbols (symbolic names) and their physical addresses. It is generated during a build process of an ECU application.

Map instrument A customized Browser instrument. It uses an instrument script to open a web map and connect positioning data to the map. The Map instrument offers prepared connection nodes to connect variables with GNSS data.

Measurement Viewing and analyzing the time traces of variables, for example, to observe the effects of ECU parameter changes.

ControlDesk provides various instruments for measuring variables.

Measurement (variable type) A scalar variable that can be measured, including individual elements of a measurement array.

Measurement variables are represented by the  symbol.

Measurement array A 1-, 2-, or 3-dimensional array of measurement variables. In variable lists, ControlDesk displays entries for the measurement array itself and for each array element.

Measurement arrays are represented by the  symbol.

Measurement buffer A ring buffer that buffers measurement data at the start of a measurement. The measurement buffer size determines the amount of data that can be buffered. Earlier values are overwritten by later values when the buffer capacity is exceeded (buffer overflow).

Measurement Configuration A controlbar that allows you to configure measurement, recording and data logging.

Measurement Data API Application programming interface for accessing measurement data. The API lets the user access measurement data without having to use ControlDesk.

Measurement Data Pool A controlbar that provides access to measurement data recorded in measurement data files.

Measurement raster Specification of how often a value of a variable is updated during a measurement. A measurement raster can be derived from a measurement service.

Measurement service The generic term for the following services:

- CCP service
- DsDAQ service
- Host service
- XCP service

Measurement signal list A list containing the variables to be included in subsequent measurements and recording. The list is global for all platforms/devices of the current experiment. The measurement signal list is available in the configuration area of the **Measurement Configuration** controlbar.

Measurement variable Any variable type that can be measured but not calibrated.

Measuring/recording A platform/device state defined by the following characteristics:

- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- Online calibration is possible. Parameter values can be changed directly on the platform/device hardware.
- A measurement (or recording) is running.
- Platform/device configuration is not possible.

The 'measuring' / 'recording' platform/device state is indicated by the icon.

Memory page An area of a calibration memory. Each page contains a complete set of parameters of the platform/device hardware, but only one of the pages is "visible" to the microcontroller of the ECU or the real-time processor (RTP) of the platform hardware at a time.

ControlDesk supports platform/device hardware with up to two memory pages. These are usually the working page and the reference page. The parameter values on the two memory pages usually are different. ControlDesk lets you switch from one page to the other, so that when parameters are changed on one page, the changes can be made available to the ECU or prototyping hardware via a single page switch.

Messages controlbar A controlbar displaying a history of all error and warning messages that occur during work with ControlDesk.

MicroAutoBox III platform A platform that provides access to a MicroAutoBox III connected to the host PC for function prototyping purposes such as Bypassing.

MicroAutoBox platform A platform that provides access to a MicroAutoBox II connected to the host PC for function prototyping purposes such as bypassing.

Mirrored memory A memory area created by ControlDesk on the host PC that mirrors the contents of the available memory pages of calibration and prototyping hardware. For hardware with two memory pages, the mirrored memory is divided into a reference and a working page, each of them containing a complete set of parameters. When a calibration or prototyping platform/device is added to an experiment, ControlDesk initially fills the available memory pages of the mirrored memory with the contents of the ECU Image file (initial filling for calibration devices) or with the contents of the SDF file (initial filling for platforms).

- Mirrored memory for offline calibration

  Parameter values can even be changed offline. Changes to parameter values that are made offline affect only the mirrored memory.
- Offline-to-online transition for online calibration

  For online calibration, an offline-to-online transition must be performed. During the transition, ControlDesk compares the memory pages of the hardware of each platform/device with the corresponding pages of the mirrored memory. If the pages differ, the user has to equalize them by uploading them from the hardware to the host PC, or downloading them from the host PC to the hardware.
- Mirrored memory for online calibration

  When ControlDesk is in the online mode, parameter value changes become effective synchronously on the memory pages of the hardware and in the mirrored memory. In other words, parameter values on the hardware and on the host PC are always the same while you are performing online calibration.

Modular system A dSPACE processor board and one or more I/O boards connected to it.

Multi-capture history The storage of all the captures acquired during a triggered measurement. The amount of stored data depends on the measurement buffer.

Multi-pin error A feature of the SCALEXIO concept for electrical error simulation that lets you simulate a short circuit between three or more signal

channels and/or bus channels. The channels can be located on the same or different boards or I/O units. You can simulate a short circuit between:

- Channels of the same signal category (e.g., four signal generation channels)
- Channels of different signal categories (e.g., three signal generation channels and two signal measurement channels)
- Signal channels and bus channels (e.g., two signal generation channels, one signal measurement channel, and one bus channel)

Multiple electrical errors A feature of the SCALEXIO concept for electrical error simulation that lets you switch electrical errors at the same time or in succession. For example, you can simulate an open circuit for one channel and a short circuit for another channel at the same time, without deactivating the first error.

Multiprocessor System platform A platform that provides access to:

- A multicore application running on a multicore DS1006 board
- A multiprocessor application on a multiprocessor system consisting of two or more DS1006 processor boards interconnected via Gigalink.

ControlDesk handles a multiprocessor/multicore system as a unit and uses one system description file (SDF file) to load the applications to all the processor boards/cores in the system.

Multistate Display An instrument for displaying the value of a variable as an LED state and/or as a message text.

Multistate LED A value cell type of the Variable Array for displaying the value of a variable as an LED state.

Multiswitch An instrument for changing variable values by clicking sensitive areas in the instrument and for visualizing different states depending on the current value of the connected variable.

N

Numeric Input An instrument (or a value cell type of the Variable Array) for displaying and setting the value of the connected variable numerically.

O

Observing variables Reading variable values cyclically from the dSPACE real-time hardware and displaying their current values in ControlDesk, even if no measurement is running. Variable observation is performed without using a measurement buffer, and no value history is kept.

For platforms that support variable observation, variable observation is available for parameters and measurement variables that are visualized in single-shot instruments (all instruments except for a plotter). If you visualize a variable in a single-shot instrument, the variable is not added to the measurement signal list. Visualizing a parameter or measurement variable in a plotter automatically adds the variable to the measurement signal list.

ControlDesk starts observing variables if one of the following conditions is true:

- Online Calibration is started for the platform.

  All the parameters and measurement variables that are visualized in single-shot instruments are observed.
- Measurement is started for the platform.

  All the visualized parameters and measurement variables that are not activated for measurement in the measurement signal list are observed. Data of the activated parameters and measurement variables is acquired using measurement rasters.

ODX database Abbreviation of Open Diagnostic Data Exchange, a diagnostics database that is the central ECU description for working with an ECU Diagnostics device in ControlDesk. The ODX database contains all the information required to perform diagnostic communication between ControlDesk and a specific ECU or set of ECUs in a vehicle network. ControlDesk expects the database to be compliant with ASAM MCD-2 D (ODX).

Offline State in which the parameter values of platform/device hardware in the current experiment cannot be changed. This applies regardless of whether or not the host PC is physically connected to the hardware.

The mirrored memory allows parameter values to be changed even offline.

Offline simulation A PC-based simulation in which the simulator is not connected to a physical system and is thus independent of the real time.

Offline simulation application (OSA) An offline simulation application (OSA) file is an executable file for VEOS. After the build process with a tool such as the VEOS Player, the OSA file can be downloaded to VEOS.

An OSA contains one or more VPUs, such as V-ECUs and/or environment VPUs.

On/Off Button An instrument (or a value cell type of the Variable Array) for setting the value of the connected parameter to a predefined value when the button is pressed (On value) and released (Off value).

Online calibration started A platform/device state defined by the following characteristics:

- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- Online calibration is possible. Parameter values can be changed directly on the platform/device hardware.
- Platform/device configuration is not possible.

Before starting online calibration, ControlDesk lets you compare the memory pages on the platform/device hardware with the corresponding pages of the mirrored memory. If the parameter values on the pages differ, they must be

equalized by uploading the values from the hardware to ControlDesk, or downloading the values from ControlDesk to the hardware. However, a page cannot be downloaded if it is read-only.

The 'online calibration started' platform/device state is indicated by the symbol.

Operation signal A signal which represents the result of an arithmetical operation (such as addition or multiplication) between two other signals.

Operator mode A working mode of ControlDesk in which only a subset of the ControlDesk functionality is provided. You can work with existing experiments but not modify them, which protects them from unintentional changes.

Output parameter A parameter or writable measurement whose memory address is used to write the computed value of a calculated variable to.

P

Parameter Any variable type that can be calibrated.

Parameter (variable type) A scalar parameter, as well as the individual elements of a value block.

Scalar parameters are represented by the 🅿 symbol.

Parameter limits Limits within which parameters can be changed. Parameters have hard and weak limits.

- Hard limits

  Hard limits designate the value range of a parameter that you *cannot* cross during calibration.

  The hard limits of a parameter originate from the corresponding variable description and cannot be edited in ControlDesk.

- Weak limits

  Weak limits designate the value range of a parameter that you *should not* cross during calibration. When you cross the value range defined by the weak limits, ControlDesk warns you.

  In ControlDesk, you can edit the weak limits of a parameter within the value range given by the parameter's hard limits.

PHS (Peripheral High Speed) bus A dSPACE-specific bus for communication between a processor board and the I/O boards in a modular system. It allows direct I/O operations between the processor board (bus master) and I/O boards (bus slaves).

PHS-bus-based system A modular dSPACE system consisting of a processor board such as the DS1006 Processor Board and I/O boards. They communicate with each other via the PHS (Peripheral High Speed) bus.

Pitch variable A variable connected to the pitch scale of an Artificial Horizon.

Platform A software component representing a simulator where a simulation application is computed in real-time (on dSPACE real-time hardware) or in non-real-time (on VEOS).

ControlDesk provides the following platforms:

- DS1006 Processor Board platform
- DS1007 PPC Processor Board platform
- DS1104 R&D Controller Board platform
- DS1202 MicroLabBox platform
- MicroAutoBox platform
- MicroAutoBox III platform
- Multiprocessor System platform
- SCALEXIO platform
- VEOS platform
- XIL API MAPort platform

Each platform usually has a variable description that specifies its variables.

Platform trigger A trigger that is available for a platform and that is evaluated on the related dSPACE real-time hardware or VEOS.

Platforms/Devices controlbar A controlbar that provides functions to handle devices, platforms, and the applications assigned to the platforms.

Plotter instrument ControlDesk offers three plotter instruments with different main purposes:

- The Index Plotter displays signals in relation to events.
- The Time Plotter displays signals in relation to measurement time.
- The XY Plotter displays signals in relation to other signals.

Port configuration To interface the failure simulation hardware, an EESPort needs the hardware-dependent *port configuration file* (PORTCONFIG file). The file's contents must fit the connected HIL simulator architecture and its failure simulation hardware.

Postprocessing The handling of measured and recorded data by the following actions:

- Displaying measured or recorded data
- Zooming into measured or recorded signals with a plotter
- Displaying the values of measurement variables and parameters as they were at any specific point in time

Processor board A board that computes real-time applications. It has an operating system that controls all calculations and communication to other boards.

Project A container for collecting and managing the information and files required for experiment/calibration/modification tasks in a number of experiments. A project collects the experiments and manages their common data.

Project controlbar A controlbar that provides access to projects and experiments and all the files they contain.

Project root directory The directory on your file system to which ControlDesk saves all the experiments and documents of a project. Every project is associated with a project root directory, and several projects can use the same project root directory. The user can group projects by specifying several project root directories.

ControlDesk uses the Documents folder as the default project root directory unless a different one is specified.

Properties controlbar A controlbar providing access to the properties of, for example, platforms/devices, layouts/instruments, and measurement/recording configurations.

Proposed calibration A calibration mode in which the parameter value changes that the user makes do not become effective on the hardware until they are applied. This allows several parameter changes to be written to the hardware together. Being in proposed calibration mode is like being in the offline calibration mode temporarily.

Push Button An instrument (or a value cell type of the Variable Array) for setting the value of the connected parameter by push buttons.

Python Editor An editor for opening and editing PY files.

Q

Quick start measurement A type of measurement in which all the ECU variables configured for measurement are measured and recorded, starting with the first execution of an ECU task. ControlDesk supports quick start measurements on ECUs with DCI-GSI2, CCP, and XCP (except for XCP on Ethernet with the TCP transmission protocol).

Quick start measurement can be used to perform cold start measurements. Cold start means that the vehicle and/or the engine are cooled down to the temperature of the environment and then started. One reason for performing cold start measurements is to observe the behavior of an engine during the warm-up phase.

R

Radio Button An instrument for displaying and setting the value of the connected parameter by radio buttons.

Real-time application An application that can be executed in real time on dSPACE real-time hardware. A real-time application can be built from a Simulink model containing RTI blocks, for example.

Record layout A record layout is used to specify a data type and define the order of the data in the memory of the target system (ECU, for example). For scalar data types, a record layout allows you to add an address mode (direct or indirect). For structured (aggregated) data types, the record layout specifies all the structure elements and the order they appear in.

The `RECORD_LAYOUT` keyword in an A2L file is used to specify the various record layouts of the data types in the memory. The structural setup of the various data types must be described in such a way that a standard application system will be able to process all data types (reading, writing, operating point display etc.).

Record layout component A component of a record layout. A structured record layout consists of several components according to the ASAP2 specification. For example, the AXIS_PTS_X component specifies the x-axis points, and the FNC_VALUES component describes the function values of a map or a curve.

Recorder An object in the Measurement Configuration controlbar that specifies and executes the recording of variables according to a specific measurement configuration.

Recorder signal list A list that contains the variables to be included in subsequent recordings.

Recording Saving the time traces of variables to a file. Both measurement variables and parameters can be recorded. Recorded data can be postprocessed directly in ControlDesk.

A recording can be started and stopped immediately or via a trigger:

- Immediate recording

  The recording is started and stopped without delay, without having to meet a trigger condition.
- Triggered recording

  The recording is not started or stopped until certain trigger conditions are met. These conditions can be defined and edited in ControlDesk.

Reduction data Additional content in an MF4 file that allows for visualizing the MF4 file data depending on the visualization resolution. Reduction data therefore improves the performance of the visualization and postprocessing of measurement data.

Reference data set A read-only data set assigned to the reference page of a device that has two memory pages. There can be only one reference data set for each device. The reference data set is read-only.

Reference page Memory area containing the parameters of an ECU. The reference page contains the read-only reference data set.

Note

Some platforms/devices provide only a working page. You cannot switch to a reference page in this case.

Resynchronization Mechanism to periodically synchronize the drifting timers of the platform/device hardware ControlDesk is connected to. Resynchronization means adjustment to a common time base.

Roll variable A variable connected to the roll scale of an Artificial Horizon.

S

Sample count trigger A trigger that specifies the number of samples in a data capture.

A sample count trigger can be used as a stop trigger.

SCALEXIO platform A platform that provides access to a single-core, multicore or multiprocessor SCALEXIO system connected to the host PC for HIL simulation and function prototyping purposes.

SCALEXIO system A dSPACE hardware-in-the-loop (HIL) system consisting of at least one processing hardware component, I/O boards, and I/O units. They communicate with each other via the IOCNET. In a SCALEXIO system, two types of processing hardware can be used, a DS6001 Processor Board or a real-time industry PC as the SCALEXIO Processing Unit. The SCALEXIO system simulates the environment to test an ECU. It provides the sensor signals for the ECU, measures the signals of the ECU, and provides the power (battery voltage) for the ECU and a bus interface for restbus simulation.

SDF file The system description file that describes the files to be loaded to the individual processing units of a simulation platform. It also contains the variable description of the relevant simulation application.

The SDF file is generated automatically when the TRC file is built.

Segment The minimum part a segment signal can consist of.

There are different kinds of segments to be used in segment signals:

- Segments to form synthetic signal shapes (sine, sawtooth, ramp, etc.)
- Segments to perform arithmetical operations (addition, multiplication) with other segments
- Segments to represent numerical signal data (measured data)

Segment signal A signal consisting of one or more segments.

Selection Box An instrument for selecting a text-value entry and setting the respective numerical value for the connected variable.

Signal

- Representation of a variable measured in a specific measurement raster.
- Generic term for segment signals and operation signals.

  A signal is part of a signal description set which can be displayed and edited in the working area.

Signal description set A group of one or more signals.

A signal description set and its signals can be edited in the working area by means of the Signal Editor. Each signal description set is stored as an STZ file either in the **Signal Description Sets** folder or in the **Signal Generators** folder.

Signal Editor A software component to create, configure, display, and manage signals in signal description sets.

Signal file A file that contains the wiring information of a simulator and that is part of the standard dSPACE documentation of dSPACE Simulator Full-Size. Normally, dSPACE generates this file when designing the simulator. Before using a failure simulation system, users can adapt the signal file to their needs.

Signal generator An STZ file containing a signal description set and optional information about the signal mapping, the description of variables, and the real-time platform.

The file is located in the **Signal Generators** folder and used to generate, download, and control Real-Time Testing sequences, which are executed on the real-time platform to stimulate model variables in real time.

Signal Mapping A controlbar of the Signal Editor to map model variables to signals and variable aliases of a signal generator.

Signal Selector A controlbar of the Signal Editor. The Signal Selector provides signals and segments for arranging and configuring signal description sets in the working area.

SIL testing Abbreviation of *software-in-the-loop testing*.

Simulation and testing of individual software functions, complete virtual ECUs (V-ECUs), or even V-ECU networks on a local PC or highly parallel in the cloud independently of real-time constraints and real hardware.

Simulation application The generic term for offline simulation application (OSA) and real-time application.

Simulation system A description of the composition of V-ECU models, environment models, real ECUs, and their interconnections required for simulating the behavior of a system. A simulation system is the basis for the generation of a simulation application for a given simulator platform.

Simulation time group Group of platforms/devices in an experiment whose simulation times are synchronized with each other. If resynchronization is enabled, ControlDesk synchronizes a simulation time group as a whole, not the single members of the group individually.

Simulator A system that imitates the characteristics or behaviors of a selected physical or abstract system.

Single-processor system A system that is based on one dSPACE processor or controller board.

Single-shot instrument An instrument that displays an instantaneous value of a connected variable without keeping a value history. In ControlDesk, all instruments except for a plotter are single-shot instruments. For platforms that support the variable observer functionality, you can use single-shot instruments to observe variables.

Slave application An application assigned to the slave DSP of a controller or I/O board. It is usually loaded and started together with the real-time application running on the corresponding main board.

Slave DSP A DSP subsystem installed on a controller or I/O board. Its slave application can be loaded together with the real-time application or separately.

Slider An instrument (or a value cell type of the Variable Array) for displaying and setting the value of the connected variable by means of a slide.

Sound Controller An instrument for generating sounds to be played.

Standard axis An axis with data points that are deposited in the ECU memory. Unlike a common axis, a standard axis is specified within a curve or map. The parameters of a standard axis can be calibrated, which affects only the related curve or map.

Start trigger A trigger that is used, for example, to start a measurement raster. A platform trigger can be used as a start trigger.

Static Text An instrument for displaying explanations or inscriptions on the layout.

Steering Controller An instrument for changing variable values using a game controller device such as a joystick or a steering wheel.

Stimulation Writing signals to variables in real-time models during a simulation run.

Stop trigger A trigger that is used, for example, to stop a measurement raster.

String A text variable in ASCII format.

Strings are represented by the  symbol.

Struct A variable with the struct data type. A struct contains a structured list of variables that can have various data types. In ControlDesk, a struct variable can contain either parameters and value blocks or measurement variables and measurement arrays. ControlDesk supports nested structs, i.e., structs that contain further structs and struct arrays as elements.

Structs are represented by the symbol.

Struct array An array of homogeneous struct variables.

Struct arrays are represented by the symbol.

STZ file A ZIP file containing signal descriptions in the STI format. The STZ file can also contain additional MAT files to describe numerical signal data.

Sub data set A data set that does not contain the complete set of the parameters of a platform/device.

Symbol A symbolic name of a physical address in a MAP file. A symbol can be associated to a variable in the Variable Editor, for example, to support an address updates.

System variable A type of variable that represents internal variables of the device or platform hardware and that can be used as measurement signals in ControlDesk to give feedback on the status of the related device or platform hardware. For example, an ECU's power supply status or the simulation state of a dSPACE board can be visualized via system variables.

T

Table Editor An instrument for displaying and setting values of a connected curve, map, value block, or axis in a 2-D, 3-D, and grid view. The Table Editor can also display the values of a measurement array.

The Table Editor can be used for the following variable types:

- Common axis ( )
- Curve ( )
- Map ( )
- Measurement array ( )
- Value block ( )

Time cursor A cursor which is visible at the same time position in the following instruments:

- In all Time Plotters
- In all XY Plotters
- In all bus monitoring lists

You can use the time cursor to view signal values at a specific point in time. If you move the time cursor, all measured signals and the respective parameters are

updated. Instruments and bus monitoring lists display the values that are available at the selected time position.

Time Plotter A plotter instrument for displaying signals that are measured in a time-based raster (time plots).

Topology A description of the processor boards belonging to a multiprocessor system and their interconnections via Gigalinks. The topology also contains information on which Gigalink port of each processor board is connected to the Gigalink ports of other processor boards in the multiprocessor system.

Topology information is contained in the real-time application (PPC/x86/RTA) files of the multiprocessor system's processor boards.

TRC file A variable description file with information on the variables available in an environment model running on a dSPACE platform.

Trigger A condition for executing an action such as starting and stopping a measurement raster or a recorder.

The generic term for the following trigger types:

- Duration trigger
- Platform trigger
- Sample count trigger

Trigger condition A formula that specifies the condition of a trigger mathematically.

Triggered measurement The measurement of a measurement raster started by a platform trigger. The data flow between the dSPACE real-time hardware or VEOS and the host PC is not continuous.

U

Unassigned data set A data set that is assigned neither to the working page nor to the reference page of a platform/device. An unassigned data set can be defined as the new working or reference data set. It then replaces the "old" working or reference data set and is written to the corresponding memory page, if one is available on the platform/device.

Unplugged A platform/device state defined by the following characteristics:

- The logical connection between ControlDesk and the hardware was interrupted, for example, because the ignition was turned off or the ControlDesk PC and the hardware were disconnected.
- Before the state of a platform/device changes to 'unplugged', the platform/device was in one of the following states:
  - 'Connected'
  - 'Online calibration started'
  - 'Measuring' / 'Recording'

Tip

A device for which the connection between ControlDesk and the device hardware currently is interrupted is also set to the 'unplugged' state when you start online calibration if both the following conditions are fulfilled:

- The device's **Start unplugged** property is enabled.
- The **Start online calibration behavior** property is set to 'Ignore differences'.

This is possible for CCP and XCP devices. For details on the two properties listed above, refer to General Settings Properties (ControlDesk Platform Management 🕮).

- If the Automatic Reconnect feature is enabled for a platform/device and if the platform/device is in the 'unplugged' state, ControlDesk periodically tries to re-establish the logical connection for that platform/device.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'unplugged' platform/device state is indicated by the ⚠ icon.

Untriggered measurement The measurement of a measurement raster not started by a platform trigger. The data flow between the dSPACE real-time hardware or VEOS and the host PC is continuous.

User function An external function or program that is added to the ControlDesk user interface for quick and easy access during work with ControlDesk.

User Functions Output A controlbar that provides access to the output of external tools added to the **Automation** ribbon.

V

Value block A parameter that consists of a 1- or 2-dimensional array of scalar parameters.

In variable lists, ControlDesk displays entries for the value block itself and for each array element.

Value blocks are represented by the 🕮 symbol.

Value conversion The conversion of the original *source values* of variables of an application running on an ECU or dSPACE real-time hardware into the corresponding scaled *converted values*.

Variable Any parameter or measurement variable defined in a variable description. ControlDesk provides various instruments to visualize variables.

Variable alias An alias name that lets the user control the property of a segment by a model parameter of a real-time application.

Variable Array An instrument for calibrating parameters and displaying measurement variable values.

The Variable Array can be used for the following variable types:

- Measurement
- Measurement array
- String
- Struct
- Struct array
- Value
- Value block

Variable connection The connection of a variable to an instrument. Via the variable connection, data is exchanged between a variable and the instrument used to measure or calibrate the variable. In other words, variable connections are required to visualize variables in instrument.

Variable description A file describing the variables in a simulation application, which are available for measurement, calibration, and stimulation.

Variable Editor A tool for viewing, editing, and creating variable descriptions in the ASAM MCD-2MC (A2L) file format. The Variable Editor allows you to create A2L files from scratch, or to import existing A2L files for modification.

Variable Filter A variable filter contains the filter configuration of a combined filter, which is used to filter the variable list in the **Variables** controlbar using a combination of filter conditions.

Variables controlbar A controlbar that provides access to the variables of the currently open experiment.

V-ECU Abbreviation of *virtual ECU*.

ECU software that can be executed in a software-in-the-loop (SIL) testing environment such as a local PC or highly parallel in the cloud independently of real-time constraints and real ECU hardware.

Vehicle information The ODX database can contain information for one or more vehicles. Vehicle information data is used for vehicle identification purposes and for access to vehicles. It references the access paths (logical links) to the ECUs.

VEOS A simulator which is part of the PC and allows the user to run an offline simulation application (OSA) without relation to real time.

VEOS Player is the graphical user interface for VEOS.

VEOS platform A platform that configures and controls the offline simulation application (OSA) running in VEOS and that also provides access to the application's environment VPU.

VEOS Player An application running on the host PC for editing, configuring and controlling an offline simulation application (OSA) running in VEOS.

Verbal conversion A conversion in which a conversion table is used to specify the computation of numerical values into strings. The verbal conversion table is used when you switch the value representation from source to converted mode and vice versa.

Verbal conversion range A conversion in which a conversion table is used to specify the computation of a range of numerical values into strings. The verbal conversion range table is used when you switch the value representation from source to converted mode and vice versa.

View set A named configuration of the controlbars of ControlDesk. A view set has a default state and a current state that can differ from the default state. The configuration includes the geometry, visibility, and docking or floating state of controlbars.

Visualization The representation of variables in instruments:

- Measurement variables are visualized in instruments to view and analyze their time traces.
- Calibration parameters are visualized in instruments to change their values.

VPU Abbreviation of *virtual processing unit*. A VPU is part of an offline simulation application in VEOS. Each VPU runs in a separate process of the PC.

VPU is also the generic term for:

- V-ECUs
- Environment VPUs
- Controller VPUs
- Bus VPUs

W

Working data set The data set currently residing in the memory of a platform/device hardware. There can be only one working data set for each calibration platform/device. The working data set is read/write.

Working page Memory area containing the parameters of an ECU or prototyping hardware (memory page). The working page contains the read/write working data set.

If the platform/device also provides a reference page, ControlDesk lets you switch between both pages.

Writable measurement A scalar variable that can be measured and calibrated.

X

XCP Abbreviation of *Universal Measurement and Calibration Protocol*. A protocol that is implemented on electronic control units (ECUs) and provides access to ECUs with measurement and calibration systems (MCS) such as ControlDesk.

XCP is based on the *master-slave principle*:

- The ECU is the slave.
- The measurement and calibration system is the master.

The "X" stands for the physical layers for communication between the ECU and the MCS, such as CAN (Controller Area Network) and Ethernet.

The basic features of XCP are:

- ECU parameter calibration (CAL)
- Synchronous data acquisition (DAQ)
- Synchronous data stimulation (STIM), i.e., for bypassing
- ECU flash programming (PGM)

The XCP protocol was developed by ASAM e.V. (Association for Standardisation of Automation and Measuring Systems e.V.). For the protocol specification, refer to http://www.asam.net.

The following ControlDesk devices support ECUs with an integrated XCP service:

- XCP on CAN device
- XCP on Ethernet device

XCP on CAN device A device that provides access to an ECU with XCP connected to the ControlDesk PC via CAN. Using the XCP on CAN device, you can access the ECU for measurement and calibration purposes via XCP (*Universal Measurement and Calibration Protocol*).

XCP on Ethernet device A device that provides access to an ECU or V-ECU with XCP connected to the ControlDesk PC via Ethernet. The XCP on Ethernet device provides access to the ECU/V-ECU via XCP (*Universal Measurement and Calibration Protocol*) for measurement and calibration purposes.

XIL API EESPort Electrical Error Simulation port (EESPort)

XIL API MAPort platform A platform that provides access to a simulation platform via the ASAM XIL API implementation that is installed on your host PC.

XY Plotter A plotter instrument for displaying signals as functions of other signals.