DESITE BIM md pro DESITE BIM md DESITE BIM custom DESITE BIM share pro DESITE BIM share DESITE BIM touch

Manage your 3D building model interactively. **Right from the beginning.**

thinkproject

DESITE configuration with command line arguments

thinkproject

Generals

All desktop applications of the **DESITE** product family (md, md pro, custom, share, share pro) can be configured through command line switches. This means, that by starting the software some specific parameters could be passed to the executable file (.exe).

Example:

```
desitemd.exe "D:\Projects\SampleProject\mySampleProject.pfs"
```

```
desitemd.exe -srvPort 45456 -cltPort 45455 -lang "en"
```

The first example starts **DESITE** md and immediately loads the specified project. Passing files to applications in this way is supported by virtually all Windows programs. By the way, exactly the same thing happens when you double-click a PFS file - assuming, of course, that **DESITE** md is installed on your computer.

The second example is more **DESITE** specific. It configures the TCP ports for **DESITE's** remote interface. It also sets English as the interface language.

This guide describes what the parameters are, what they do, how they are passed to the application, and what differences there are between **DESITE** applications to consider.

1. Passing parameters
1.1 as command line argument
1.2 as configuration file

2. Parameter reference
2.1 file
2.2 srvPort, cltPort
2.3 showWidget
2.4 webFormLocations
2.5 unlockWebForms
2.6 lang
2.7 scriptDebug

3. Availability in **DESITE** applications

thinkproject

1. Passing parameters

1.1 Passing parameters as command line arguments

The parameters are typed with hyphen (-) after the program name (.exe), in order to be passed via command line.

Examples:

```
desitecustom.exe –lang "en"
```

```
desitecustom.exe –srvPort 45456
```

```
desitecustom.exe –unlockWebForms
```

(The meaning of these parameters is not of interest at this point, that will be covered in the parameter reference).

Some parameters require a value (like lang and srvPort here), some work like simple switches without a value (like unlockWebForms here). If a value is required, it is specified after the parameter with a single space as a separator. Note the data types of the values.

In the example, "en" is in quotes because it is a string, while 45456 is given without quotes because it is a port number. The arguments can of course be combined:

```
desitecustom.exe –lang "en" –srvPort 45456 -unlockWebForms
```

1.2 Passing parameters as configuration file

Alternatively, the parameters can be written to a config file (JSON format) and passed to the application using -configFile:

```
desitecustom.exe –configFile "E:\somePath\desiteConfig.json"
```

The configuration file could look something like this:

```
{
    "lang":"en",
    "srvPort":45456,
    "unlockWebForms":true
}
```

This would set the same options as we did on the previous page with command line arguments. The JSON keys are 1:1 like the command line arguments. However, there is one important difference with unlockWebForms: as a command line argument, no value is specified for this switch, in JSON the value is true.

Important: If -configFile is used, any other command line parameters that may be specified are ignored. This is to avoid ambiguities or conflicts.

2. Parameter reference

thinkproject

2.1 file

Load a project or model with -file "filepath".

Supported **DESITE** applications: md, md pro, custom, share, share pro (with the respective supported model formats).

Example:

```
desitemd.exe –file "D:\Projects\SampleProject\mySampleProject.pfs"
```

Note: This does the same thing as passing the filename directly:

```
desitemd.exe "D:\Projects\SampleProject\mySampleProject.pfs"
```

2.2 srvPort, cltPort

Set the TCP ports for the remote interface with -srvPort Number and/or -cltPort Number.

Supported **DESITE** applications: md, md pro, custom, share, share pro

Example:

```
desiteshare.exe –srvPort 45456 –cltPort 45455
```

DESITE applications can send and receive messages over local TCP sockets to communicate with other applications. By default, the server port (listener) is 45456 and the client port (sender) is 45455. Changing these values may be necessary if the default ports are not available in your network environment, or if multiple instances of **DESITE** applications are running simultaneously.

More details about the DESITE remote interface can be found in the corresponding documentation.

2.3 showWidget

Show a specific widget or window on application start with -showWidget
Supported **DESITE** applications: md, md pro, custom, share*, share pro*

Example:

```
desitecustom.exe –showWidget WebForms
```

Use this switch with one of the following parameters to open the respective widget:

- `WebForms`
- `Issues` (alternatively: `Viewpoints`)
- `ObjectInfo` (alternatively: `DataSheet`)
- `ProjectStructure`

Note: This does not necessarily ensure that the window appears visibly in the foreground. Depending on the last saved interface configuration, it could also be in an inactive tab.

This switch is the successor of `–showWebForms`. It will still work, but is deprecated. Use `–showWidget WebForms` instead.

* restricted, e.g. no WebForms

2.4 webFormLocations

Define form sources with -webFormLocations "url1[;url2]".
Supported DESITE applications: md, md pro, custom.

Example:

```
desitemd.exe –webFormLocations "https://www.thinkproject.com"
```

Since version 2.6 **DESITE** supports other sources as start page for the forms besides the traditional local DbInputForm.html. This can be especially useful if you need the same forms in many projects.

You can specify multiple URLs. When using the command line argument, set a semicolon as a separator. If a configuration file is used, use the JSON list notation (even with only one URL):

```
{
    "webFormLocations":["https://www.thinkproject.com"]
}
```

2.5 unlockWebForms

Grant a general WebForms unlock with -unlockWebForms.

Supported DESITE applications: md, md pro, custom

Example:

```
desitecustom.exe –unlockWebForms
```

Use this switch if you want to prevent the need to explicitly enable the use of WebForms and other browsers embedded in **DESITE** - a security feature introduced with version 2.6.1.

Note: For security reasons, we do not recommend routine use of this switch. It should only be used in WebForms development when the query would otherwise only interfere, or when you are really sure to open projects only from trusted sources.

2.6 lang

Set your language with -lang "langCode".
Supported DESITE applications: md, md pro, custom, share, share pro

Example:

```
desiteshare.exe –lang "en"
```

The language setting set in this way overrides the corresponding setting in the application configuration, but does not overwrite it (override, not overwrite).

For example, if English is set as the language in the program options, and you start with the -lang parameter "de", the interface for this session will be localized in German. If you then however, start the application again without the parameter, the English setting is applied again.

Valid values (**DESITE** 2.6): en, de, fr, it, es, nl, pl.

2.7 scriptDebug

Enable JavaScript/WebForms debugging with -scriptDebug [port].

Supported DESITE applications: md, md pro.

Example:

```
desitemd.exe –scriptDebug
```

When script debugging is active, JavaScript errors in macros will trigger a debug window that allows you to inspect local variables, set breakpoints, etc.

Furthermore, you can remote-debug WebForms. To do that, open up a separate Chromium-based browser (like Google Chrome or Microsoft Edge) and navigate to the following address:
http://localhost:1000 .
Choose the WebForm you wish to inspect, and then you can analyse and debug the code with the usual Chromium developer tools.

The default port for remote debugging is 1000. If you want another port for some reason, you can pass it as a parameter, e.g. "-scriptDebug 1234" and its corresponding address: http://localhost:1234 .

Please note that –scriptDebug can have negative effects on performance. You should therefore use it only while developing or testing your code, not in production.

3. Availability in **DESITE** applications

thinkproject

Availability in **DESITE** applications

	md pro	md	custom	share pro	share
file	✓	✓	✓	✓	✓
srvPort, cltPort	✓	✓	✓	✓	✓
showWidget	✓	✓	✓	🟠✓	🟠✓
webFormLocations	✓	✓	✓	(no web forms)	(no web forms)
unlockWebForms	✓	✓	✓	(no web forms)	(no web forms)
lang	✓	✓	✓	✓	✓
scriptDebug	✓	✓	✗	✗	✗