

MIKE SHE Toolbox

User Guide





PLEASE NOTE

COPYRIGHT

This document refers to proprietary computer software which is protected by copyright. All rights are reserved. Copying or other reproduction of this manual or the related programs is prohibited without prior written consent of DHI. For details please refer to your 'DHI Software Licence Agreement'.

LIMITED LIABILITY

The liability of DHI is limited as specified in Section III of your 'DHI Software Licence Agreement':

'IN NO EVENT SHALL DHI OR ITS REPRESENTATIVES (AGENTS AND SUPPLIERS) BE LIABLE FOR ANY DAMAGES WHATSOEVER INCLUDING, WITHOUT LIMITATION, SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES OR DAMAGES FOR LOSS OF BUSINESS PROFITS OR SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION OR OTHER PECUNIARY LOSS ARISING OUT OF THE USE OF OR THE INABILITY TO USE THIS DHI SOFTWARE PRODUCT, EVEN IF DHI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION SHALL APPLY TO CLAIMS OF PERSONAL INJURY TO THE EXTENT PERMITTED BY LAW. SOME COUNTRIES OR STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL, SPECIAL, INDIRECT, INCIDENTAL DAMAGES AND, ACCORDINGLY, SOME PORTIONS OF THESE LIMITATIONS MAY NOT APPLY TO YOU. BY YOUR OPENING OF THIS SEALED PACKAGE OR INSTALLING OR USING THE SOFTWARE, YOU HAVE ACCEPTED THAT THE ABOVE LIMITATIONS OR THE MAXIMUM LEGALLY APPLICABLE SUBSET OF THESE LIMITATIONS APPLY TO YOUR PURCHASE OF THIS SOFTWARE.'





CONTENTS

MSHE Tools: DATA ANALYSIS	7
1 Probability Curves	9
1.1 Setup Name	9
1.2 Time series file selection	9
1.3 Output Settings	9
1.4 Output file selection	10
1.5 Status	10
MSHE Tools: FILE CONVERTER	11
2 dfs2+dfs0 to dfs2	13
2.1 Setup Name	13
2.2 Files	13
2.3 Status	13
MSHE Tools: UTIL	15
3 Grid calculator	17
3.1 Setup Name	17
3.2 Calculation Setup	17
3.2.1 Expressions	18
3.3 Status	19
3.4 Command line usage	20
3.4.1 File Functions	20
3.4.2 A Sample Command Line Session	22
3.4.3 Running a script	23
3.4.4 Running from a pfs file	23
Index	25





MSHE TOOLS: DATA ANALYSIS



The Data Analysis Tools in the MSHE Toolbox include the facility

- Probability Curves



1 Probability Curves

The MIKE SHE Probability Curves Tool calculates the mean value, as well as the probability and duration curves for the specified period in a dfs0 file.

1.1 Setup Name

Specify a name for the setup. This allows you to save and retrieve the setup, either later, or from a batch file.

1.2 Time series file selection

Select the dfs0 file for which you want the probability and duration curves.

1.3 Output Settings

Output Settings

Periods:

Period Start: 01/02/1970 12:00:00

Period End: 01/02/2050 12:00:00

Interval: Monthly

Nr. of Periods: 12

Sub-Sets

☐ Manual Settings

Min. Value: 0

Max. Value: 0

Nr. Sub-sets: 100

File Start: 01/02/1970

File End: 01/02/2050

	Start	End
1	January	January
2	February	February
3	March	March
4	April	April
5	May	May
6	June	June
7	July	July
8	August	August
9	September	September
10	October	October
11	November	November
12	December	December

< Back Next > Cancel Help

The output settings dialogue allows you to define the period and interval for the statistic calculation. If you select Specified for the Interval, then you can define the number of periods and the start and end months.

The sub-sets option allows you to further refine the statistical breakdown.



1.4 Output file selection

Type in the name of the ASCII file where want the probability and duration curves written to.

1.5 Status

The Status page presents you with a summary of all the input parameters you have specified. Check that the parameters are correct. If not go back and change them.

Log/pfs file location

This is the directory where the output will be sent. A subdirectory will be created here based on the name of the setup.

Running the Tool

Finally, on the Status page, you can

Execute the setup, which will run the Tool with the current parameters, or

Finish the setup, which will save your setup definition in the current toolbox file.

The tool runs silently. If there are any errors, then the tool will simply not run.



MSHE TOOLS: FILE CONVERTER



The File Converter Tools in the MSHE Toolbox include the facility:

- dfs2+dfs0 to dfs2



2 dfs2+dfs0 to dfs2

This tool is used to build a time varying dfs2 file from a dfs2 grid code file and one or more dfs0 time series files. The time varying dfs2 file can be used for time varying gridded data items, including Precipitation Rate and Evapotranspiration.

2.1 Setup Name

Specify a name for the setup. This allows you to save and retrieve the setup, either later, or from a batch file.

2.2 Files

This dialogue contains the file name for the two input files and the output file.

The input file is a dfs0 file, usually with multiple data items. All the data items are expected to have the same units, but this is not checked.

The input dfs2 file is a grid file with grid codes corresponding to the item numbers in the dfs0 file. The dfs2 file must have an EUM type of Grid Codes with a unit of Integers.

If you are creating a distributed precipitation rate file, then the dfs0 items must be of type “Mean Step Accumulated”, with a EUM Type of Precipitation Rate and any valid unit, such as [inches/hour] or [mm/day].

The output file will be a dfs2 file with one grid for each time step in the dfs0 file. and the values in each grid point equal to the value in the dfs0 file at that time step.

2.3 Status

The Status page presents you with a summary of all the input parameters you have specified. Check that the parameters are correct. If not go back and change them.

Log/pfs file location

This is the directory where the output will be sent. A subdirectory will be created here based on the name of the setup.

Running the Tool

Finally, on the Status page, you can

Execute the setup, which will run the Tool with the current parameters, or



Finish the setup, which will save your setup definition in the current toolbox file.

The tool runs silently. If there are any errors, then the tool will simply not run.



MSHE TOOLS: UTIL



The Util Tools in the MSHE Toolbox include the facility

- Grid calculator



3 Grid calculator

The MIKE SHE Grid Calculator tool allows you to perform complex operations on .dfs2 and dfs3 grid files.

The Grid Calculator tool has been upgraded to handle time varying dfs files with multiple Items. The documentation also now includes examples on how to use the tool from a script.

The only limitation is that all the grid files must have the same grid dimensions and the same number of time steps.

3.1 Setup Name

Specify a name for the setup. This allows you to save and retrieve the setup, either later, or from a batch file.

3.2 Calculation Setup

Note all fields must be filled in.

	Variable Name	Data File
1	A	C:\Maps\Topography.dfs2
2	B	C:\Maps\Lower Level.dfs2
3		
4		
5		
6		
7		
8		

Output specification

Description: Thickness of Layer 1

File name: C:\Maps\thickness.dfs2

Item type: Layer Thickness Item unit: meter

Expression

A-B

Example: (Level1 - Level2) * Conduc

< Back Next > Cancel Help

In the first part of the dialogue, you need to specify the files and variable names used in the mathematical operation. These will be the variables used in the subsequent operation. **The variable names are case sensitive.**



Output Specification

In the second part, you must specify the output file name (including the .dfs2 or dfs3 file extension), as well as the EUM unit type and units for the resulting file. Although the EUM list is alphabetical, the list of EUM data types is very long, as it includes all of the available EUM types for all of the MIKE Zero products. If you are trying to create a particular input type for MIKE SHE, then you should look in the MIKE SHE Setup dialogue, to find out first what EUM data type is required and then find this type in the list.

3.2.1 Expressions

The last part of the dialogue is the mathematical expression you wish to evaluate. The standard order of operations is followed, including the use of nested brackets.

Operators

Valid operators are: +, -, *, /, and ^.

These (binary) operators can be applied to:

- Two scalars (single floating point numbers)
- Two arrays of equal structure (e. g. two variables containing data read from two dfs files, where the same spatial and temporal extent has been read from each file). If the two arrays do not match in size, an error will be reported.
- An arbitrary array and a scalar. In this case the operation will be performed on each element of the array and the scalar, so the result has the same dimensions and extents as the input array.

Functions

Valid functions with a single argument are:

- fabs - absolute value
- ceil - maximum value in the time series
- floor - minimum value in the time series
- log, and log10 - natural and base 10 logarithm
- sqrt - square root
- sin, cos, tan, asin, acos, atan - trigonometry functions

These (unary) functions can be applied to any array or scalar. The result has the same spatial/temporal dimensions and extent as the input.

Valid functions with two arguments are:

- fmin - minimum of two values
- fmax - maximum of two values



- `atan2` - arctangent function with two arguments

The same rules as for binary operators apply.

Pre-defined constants

The only pre-defined constant is

- $\pi = 3.141593$

Invalid operations and numbers

Mathematically invalid operations include, for example

- division by zero, `sqrt(-1)` and `log(0)`.

Any operation using an invalid number will yield an invalid number.

Invalid operations can be performed without causing the tool to fail, but the result will be an undefined/invalid number (including positive/negative infinity). This allows you to calculate derived values from input data without having to pre-process the input data. However, when writing the result to a dfs file any invalid number will be written as a "delete value", so there may be a "gap" in the result file even if all input data has valid values in that time step/location.

Note: All numbers assume the "." decimal separator, regardless of region settings on your computer.

3.3 Status

The Status page presents you with a summary of all the input parameters you have specified. Check that the parameters are correct. If not go back and change them.

Log/pfs file location

This is the directory where the output will be sent. A subdirectory will be created here based on the name of the setup.

Running the Tool

Finally, on the Status page, you can

- **Execute** the setup, which will run the Tool with the current parameters, or
- **Finish** the setup, which will save your setup definition in the current toolbox file.

The tool runs silently. If there are any errors, then the tool will simply not run.



3.4 Command line usage

To use the raster calc tool from the command line, open a windows command prompt (cmd.exe). Then execute MSheCalcApp.exe from your MIKE installation directory. An interactive shell will open waiting for commands to execute. To close the tool, press the enter key without any input in the current line.

For each operation the result is printed. For a scalar that is the scalar value. For a data object it is a sample of up to 4 time steps and the top 4 rows (highest row indices)/first 4 columns (lowest column indices). This is to resemble the ordering in the MZ grid editor.

Variables

Variables can be defined to hold scalar or array data. Variable names can consist of any sequence of alphanumeric ASCII characters where the first character has to be alphabetical. Variables can be used to hold scalar values, data objects read from files (see section "File Functions") or any intermediate result. The content of a variable can be written to a file.

3.4.1 File Functions

There are two functions available, one to read data from a file, the other to write data to a file.

The path to a file can be an absolute path or a path relative to the current working directory. The current working directory can be set before starting the raster calc tool by:

- In the windows command prompt switch to the drive,
- then cd to a directory - this will be your working directory.

```
ReadDfs(string filename, int item=0, int time-  
Start=0, int timeEnd=timeStart)
```

Reads data from a dfs file to an internal data object.

Parameters

- `filename`: Filename/path to the dfs file input.
- `item`: 0-based index of the dfs item to load. Default: 0 (the first item)
- `timeStart`: 0-based index of the first time step in the range of time steps to load. Default: 0 (the first time step in the file)
- `timeEnd`: 0-based index of the last time step in the range of time steps to load (inclusive). If the value is greater than the index of the last time step in the file, it will be replaced by the index of the last time step in the file. To make sure data is read in progressive order timeEnd should usually be greater or equal to timeStart. Default: timeStart (the one time step specified by timeStart will be loaded in this case)



Parameters that have a default value can be omitted, but if a parameter is specified, then all previous parameters have to be present, too.

Return value

A data object that can be assigned to a variable to be used in calculations.

Examples

Read the first item and first time step of a file and assign the content to the variable A:

```
A=ReadDfs("Grid1.dfs2")
```

Read the second item and first time step of a file and assign the content to the variable B:

```
B=ReadDfs("Grid1.dfs2",1)
```

Read the second item and first 3 time steps of a file and assign the content to the variable C:

```
C=ReadDfs("Grid1.dfs2",1,0,3)
```

```
CreateDfs(object o, string fileName, string itemName, int eumType, int eumUnit)
```

Writes an object to a dfs file.

Note: Any existing file will be overwritten without warning!

Parameters

- o: the data object to write to a dfs file. One of:
 - a defined variable
 - a valid ReadDfs statement
 - any valid calculation expression
- filename: Filename/path to the dfs file output.
- itemName: An arbitrary name for the dfs item
- eumType: the ID of the item type.
- eumUnit: the ID of the item unit.

The Item Type and Item Unit need to be valid, otherwise the creation of the dfs file will fail. The Item Type and Item Unit can be changed in the Grid Editor later if necessary.

The following properties of a result file cannot be set explicitly but will be assigned based on the input data:

- The data type
- The delete value



- The value type (e.g. instantaneous or accumulated)
- The spatial reference system/projection
- The dimension and spatial axis type
- The time axis type (e.g. equal calendar or equal time)

Return value

The object or expression passed as 1st parameter

Examples

Create a file with the contents of variable A. The file will be in the current working directory.

```
CreateDfs(A, "A_out.dfs2", "Water level", 10000, 1000)
```

Memory release

To release memory and un-define a variable, use the function:

```
Del(<variableName>)
```

3.4.2 A Sample Command Line Session

Assume you want to produce a flood hazard map containing the product of flooding depth and flow velocity. Your input file has (among others) the following items:

- The depth of overland water (m, first item, index 0)
- The overland flow in x-direction (m³/s, second item, index 1)
- The overland flow in y-direction (m³/s, third item, index 2)

The cell size is 500 by 500 m. Launch the tool from the directory where your input file is located. Then type the following commands:

- *d = ReadDfs("MyModel_overland.dfs2 ", 0, 1, 5)*
- *qx = ReadDfs("MyModel_overland.dfs2", 1, 1, 5)*
- *qy = ReadDfs("MyModel_overland.dfs2", 2, 1, 5)*
- *a = d * 500*
- *vx = qx / a*
- *vy = qy / a*
- *vabs = sqrt(vx^2 + vy^2)*
- *haz = vabs * d*
- *CreateDfs(haz, "MyModel_hazard.dfs2", "Flood Hazard", 100363, 4702)*



The resulting file will contain the flood hazard values for the second to 6th time step.

Note Due to the calculation of v_x and v_y , in this case, there will be no data values in cells where the depth is 0! If you prefer to have the value 0.0 in these cells instead an easy work around could be to calculate a such that it has a minimum value: $a = \text{fmax}(d * 500, + 1e-20)$. In this case, all calculations are valid and the hazard is calculated to be 0.0 with no effect on other values.

3.4.3 Running a script

The easiest way to run commands from a script is to use standard windows functionality. If you have a script file `postProc.txt` containing raster calc tool commands (e.g. those given in the "Sample Session" section) then you can feed those commands to the raster calc tool by doing the following:

- Open a windows command prompt and `cd` to the directory containing your data.

Then type the following:

- `C:\Your\Mike\InstallDir\bin\x64\MSheCalcApp.exe < postProc.txt`

The '`<`' will pipe the content of the file to the executable. The result will be the same as when typing each command of the file into the interactive shell.

3.4.4 Running from a pfs file

This is a way of executing the tool commands as defined in a GUI toolbox. The toolbox is stored as a *.mst file. The raster calc tool can take 2 additional arguments:

- The name of the *.mst file create by the GUI
- The name of the setup inside the *.mst file

If for an example you execute the following command:

```
MSheCalcApp.exe MST1.mst GridCalcTest
```

The commands stored in the setup called "GridCalcTest" in the file `MST1.mst` will be executed.





INDEX

