MSDA_XE

ADVANCED MANUAL

release date: 2012-08-26

TriOS Mess- und Datentechnik GmbH
Bürgermeister-Brötje-Str. 25
D- 26180 Rastede
Germany
Fon: +49 (0) 4402 69670-0
Fax: +49 (0) 4402 69670-20
support@trios.de
http://www.trios.de/

MSDA_XE
Advanced Manual

Content

1 Introduction

This manual explains the advanced feature of MSDA_XE which will be used only of advanced users.

2 Key

The following symbols / formats are used in the manual.

[Window-Name]

Menu-Entry

Link

Important Note !

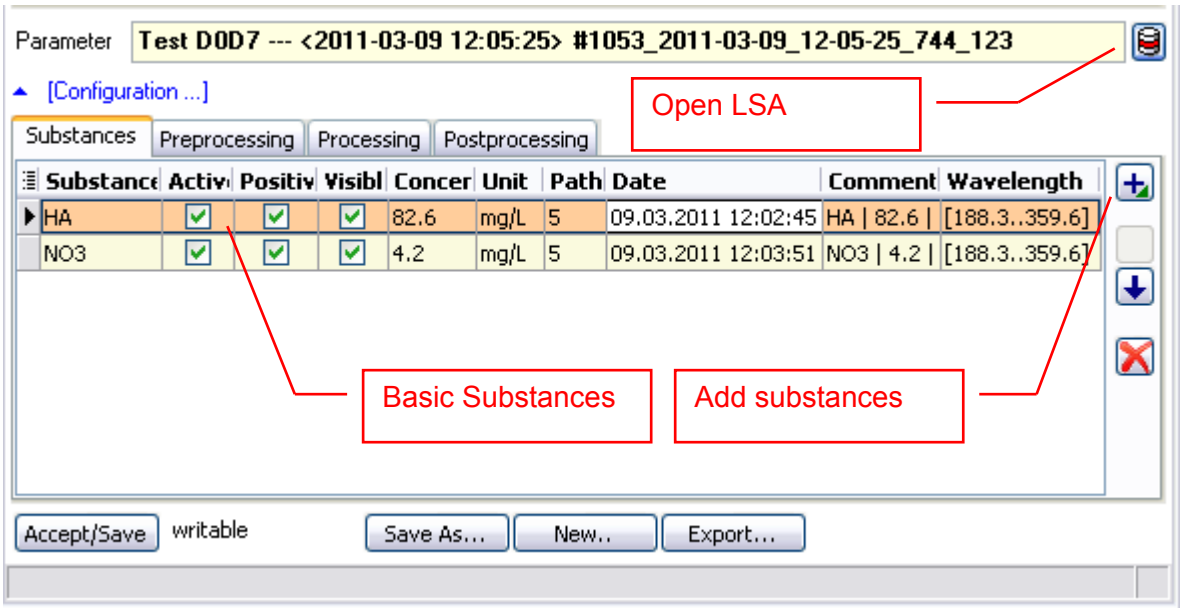
Hint !

3 Linear Substance Analysis Window

3.1 Introduction

The Linear Substance Analysis, henceforth called LSA is a complex function to calculate substance concentrations from the shape of the absorption spectrum. The LSA requires a set of basis spectra or substance bases (see 4 Concentration Assignment Window). The LSA finds the linear combination of basic spectra that most corresponds to the sample spectrum.

3.2 Group



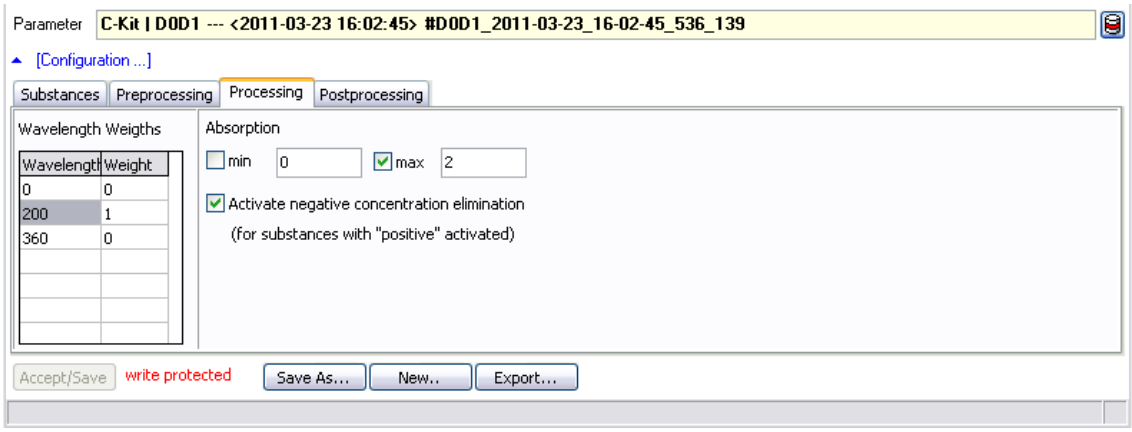
The LSA groups, if saved, will be listed in the database with the IDDataType SUBSTANALYSIS_PARAM. When the ProPS is delivered with a LSA it can be imported as file with the ending *.dat and selected with the button.

	Load LSA from database
New..	Create a new LSA
Accept/Save	Accept changes. Needed before new data is processed.
Export...	Export LSA as file
	Add substance basis
	Delete substance basis from the LSA group

	Change position in the LSA group

This function needs a special license key.

3.3 Processing



Wavelength Weights	Defining spectral ranges and their weight for calculations. The limitation of the wavelength range is necessary in most cases and depends on the reference substance and the sample. Insert a wavelength and a weight entry. All wavelength larger than the entry are used with this weight, up to the next entry. A wavelength interval with weight 0 is excluded from calculation. If intervals have larger weights than 1 they are fitted more precisely.
Absorption	Defining absorption range or upper absorption limit

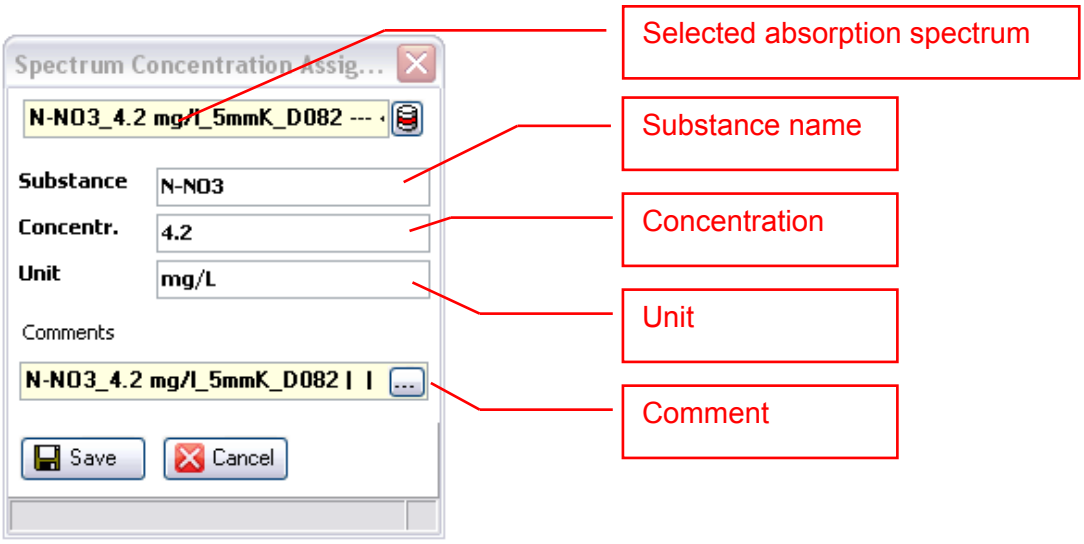
Examples for wavelength weights configuration.

Wavelength	**Weight**	**Used wavelengths**
No entry		Full range
200	1	From 200 nm to end of spectrum is taken
200 300	1 0	[200…300] taken
200 250 300 350	1 0 2 0	[200...250] weight 1, [250…300] weight 0 (not used), [300…350] weight 2 (Twice more accurate fit in this interval)

4 Concentration Assignment Window

To program a LSA it is necessary to select substance basis, which are either delivered from TriOS or can be named with the Concentrations Assignment Window.

A substance basis shall contain a single substance with a known substance concentration. The measured spectrum has to be selected in the concentration assignment window, therefore it needs to be saved in the database.



Procedure:

1. Select the absorption spectrum of the substance measurement
2. Type the substance name in
3. Type the concentration in (without units)
4. Type the unit in
5. Fill in the comment (optional, but useful)
6. Save

A data record (IDDataType=SubstAnalysis) will be created in the database. This data record contains an internal link to the absorption spectrum.

5 micro-/ enviroFlu - Calibration

With this function, TriOS fluorometer can be calibrated and new calibration coefficients can be saved on the sensors EEPROM.

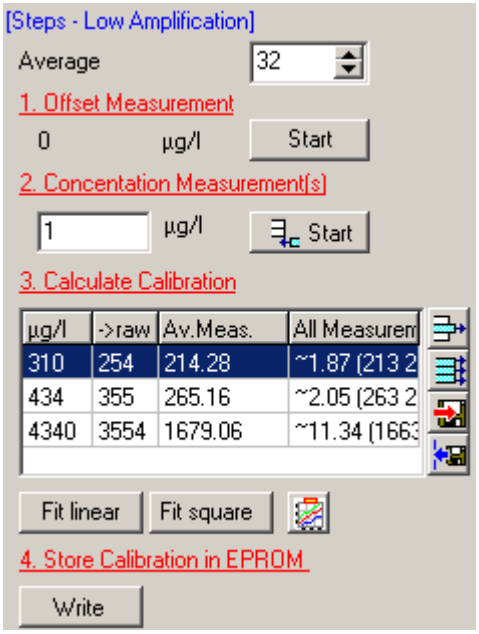
MSDA_XE
Advanced Manual

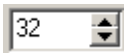
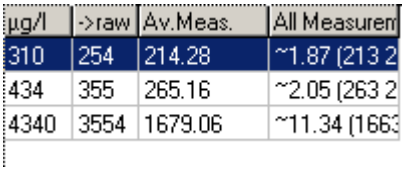
Steps 1-4, which are necessary for the calibration are listed below. At least one concentration measurement is needed for a linear fit. For a square fit, 2 values are needed at minimum. In both cases the offset needs to be calibrated at first.

To see this page in the window you must import the license Flu_Advanced.lic.

Read the sensors manual for more information.

Calibrating the sensors with this function will remove the factory calibration.



Average 	Additional software averaging of measurement values
1. Offset 	Measure and calculate calibration offset parameter while sensor is in fluorescence free medium (zero value)
2. Concentration 	Measurement of substance solution with known concentration(s). **The concentration field needs to be filled, before the measurement is started.**
3. Calculate Calibration	
	Table of calibration measurements from step 2.

µg/l	Concentration of sample
->raw	RAW counts of measurement
Meas.Av.	Average of RAW values of this sample (measured)
All Measurements	All raw value samples and its standard

MSDA_XE
Advanced Manual

	deviation (of the average count samples) This table is stored in the desktop automatically.
	Import table from file
	Export table to file (for later usage)
	Remove row of table
	Remove all rows of table
Fit linear	Calculate linear fit of all calibration measurements
Fit square	Calculate linear and square fit of all calibration measurements This button is shown only, when the device supports the current amplification.
	Show calibration points and the calculated fit line in a chart. A chart with the input “FluCtrl” needs to be opened at first.
4. Write Values Write	Write measured values in the EPROM

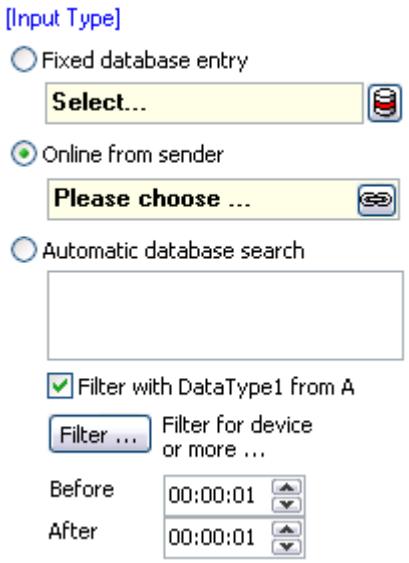
Both amplifications has be calibrated.

6 Double Spectrum Calculation

With this window you can calculate a new spectrum C based on the values of two other spectra (A and B). Different spectra may have different wavelength raster, therefore the algorithm synchronizes the wavelengths via a linear interpolation function. For every intensity pair A[Wavelength_i] and B[Wavelength_i] the method calculates a new value C[Wavelength_i]. You can configure standard formulas A/B and B/A or you can create your own formula with the integrated scripting language.

6.1 Input configuration

The input of the window configures the source for A. On every new receive spectrum A a new spectrum C will be calculated and stored in the database if enabled. Spectrum B can configured in three different ways:



1. Fixed database entry

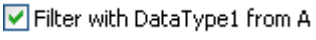
You choose one fixed spectrum from the database

2. Online from sender

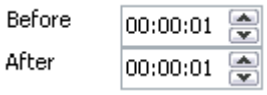
Spectrum B is variable and may change depend on the source. The source can be a Database Data Sender or a Device Control Window which sends spectra.

3. Automatic Database Search

If you have stored all the input spectra pairs (A B) in the database you should use this mode. The common aim is to find pairs with nearly the same timestamp. First you must click on 

to restrict the source for B. Configure the filter so that you see only allowed spectra for B. With 

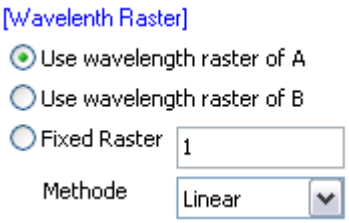
you can enhance the configured filter with an additional condition. F.e. if the current received spectrum A has DataType1=Calibrated the spectrum B must have this feature, too.

With this entry 

you configure what means “nearly the same timestamp”. Based on the timestamp from spectrum A the timestamps from B must be in the interval [Timestamp_A - Before … Timestamp_A + After]. It you have no matching spectrum B in the database you get an error message an the calculation will not performed. If you have more than one matching spectrum B in the database the best (nearest) will be taken.

6.2 Wavelength Raster

Here you can configure the calculation mode for the wavelengths synchronisation.



For the interpolation method you should use “Linear”. “Cubic Spline” gives strange result for very unstead values.

6.3 Formular

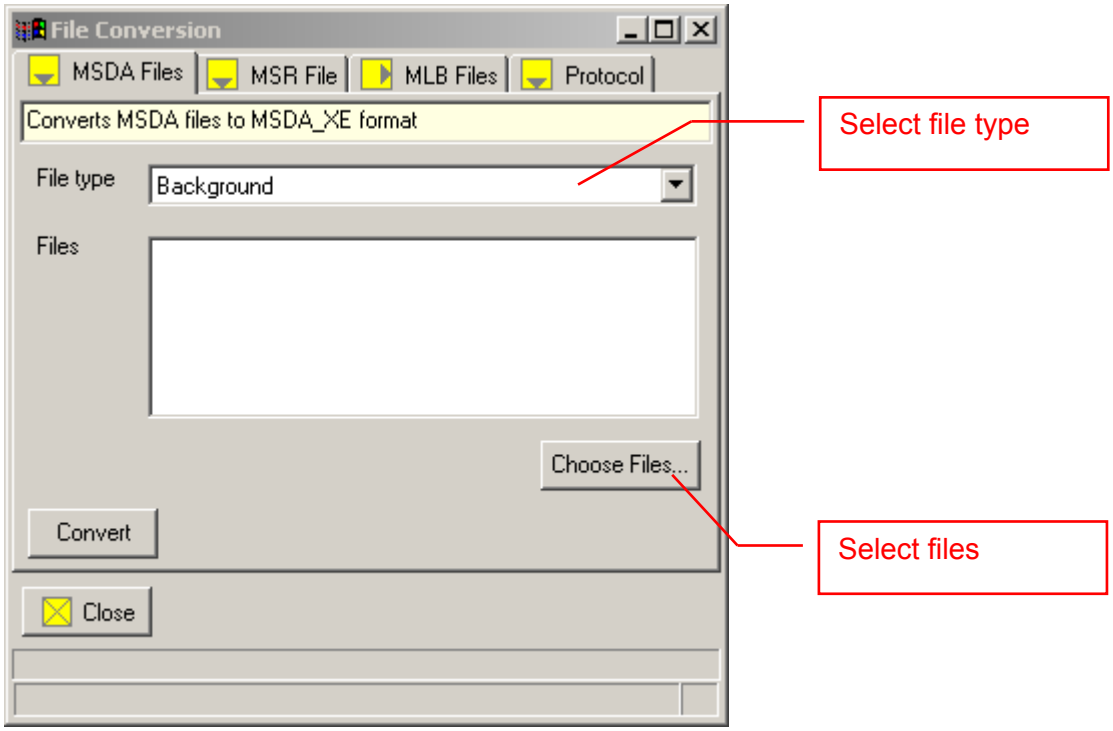
7 File Conversion

The old MSDA and the new MSDA_XE use different file formats. Therefore it is needed to convert some file types before importing them into the database of MSDA_XE. This can

be done with the file conversion window: **Main Menu \ Tools \ File Conversion**. The functions of the different tabs are described below.

7.1 Msda files

It is possible to load old files from the old software MSDA to the new file format of MSDA_XE. The software MSDA uses version numbers of 6 or less.



File conversion	
File type	Set file type of files to be converted.
Choose Files	Add one ore more files to convert
Convert	Start converting

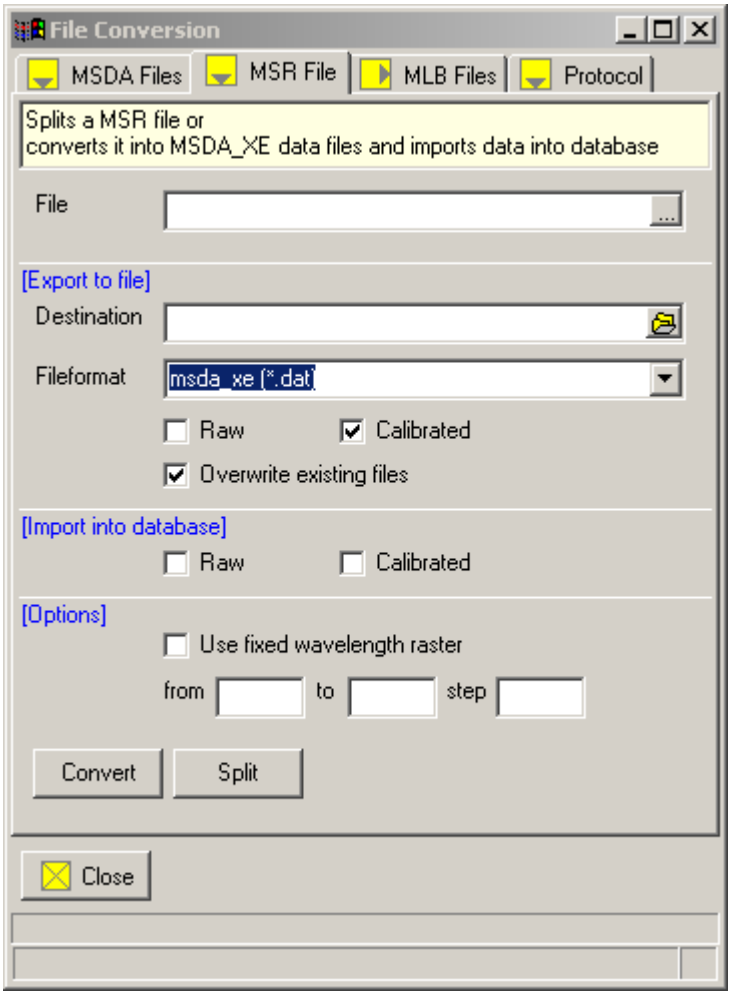
The files converted will be saved in the subfolder "*new*" within the programme folder.

For appropriate conversion, the file type must be correct! The software cannot identify different data types.

MSR

Import recorder files of the previous MSDA version here (version number less than 6.x).

Precedent software versions cannot import oversized MSR files. Thus, it is possible to split oversized MSR files.



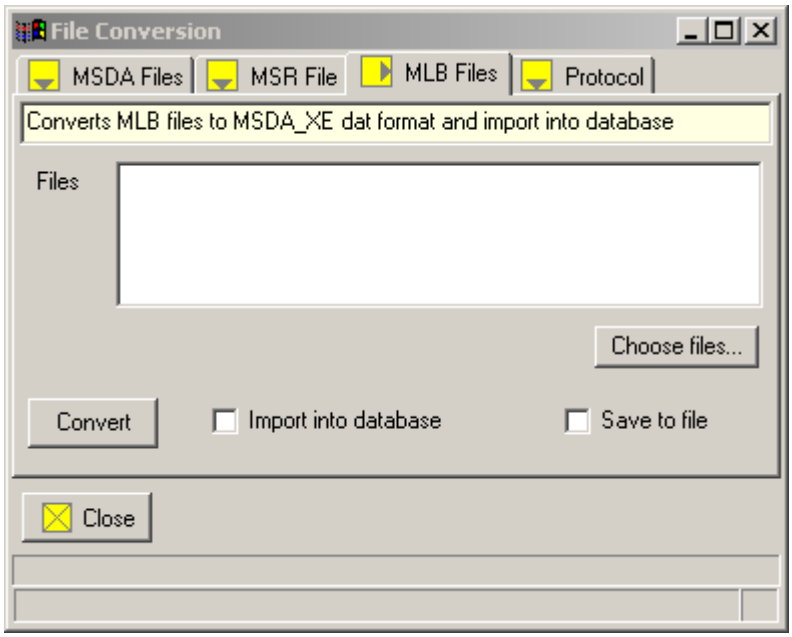
File conversion	
File	Select the file to be converted.
Export to file	
Destination	Destination path
File format	msda_xe (*.dat) - internal data exchanging format (ASCII) Matlab (*.mlb) - Matlab compatible format (ASCII) Raw (*.raw) - internal data format (binary)
Raw / Calibrated	Raw and/or calibrated data
Overwrite existing files	Overwrite an existing file

Import into database	
Raw / Calibrated	Raw and/or calibrated data
Options	
Use fixed wavelength raster	A fixed wavelength raster can be used for exporting
[Convert]	Converts the file
[Split]	Split the file into smaller parts. Useful with previous MSDA versions.

Corrupted data files can cause problems with conversion. For further help, please contact our software or support department (support@trios.de).

7.2 MLB Files

Spectra, which are stored in the MatLab file format can be imported by the MLB import tab.



Data reimport is optimized for the *.dat format. Please do only use the other file formats, if necessarily needed.

8 Calculator

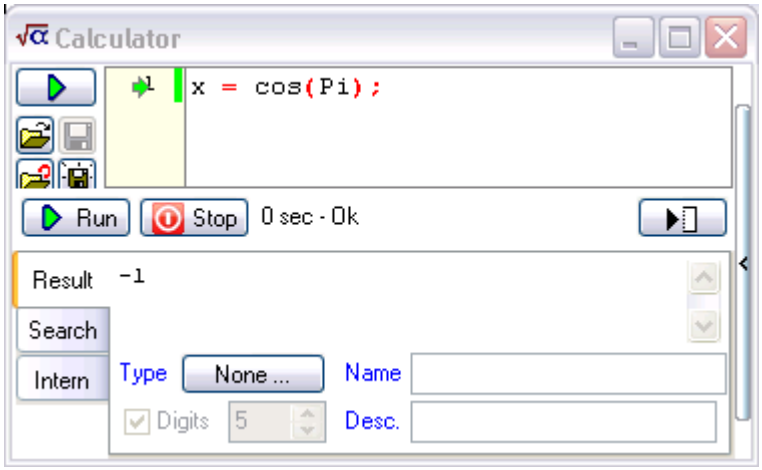
8.1 Introduction

A powerful scripting language for custom calculations is integrated in MSDA_XE. With the Calculator you develop scripting-code, run and test the code and send its results to other windows. Scripts are also used for the ValueCalculator window.

8.2 Inputs

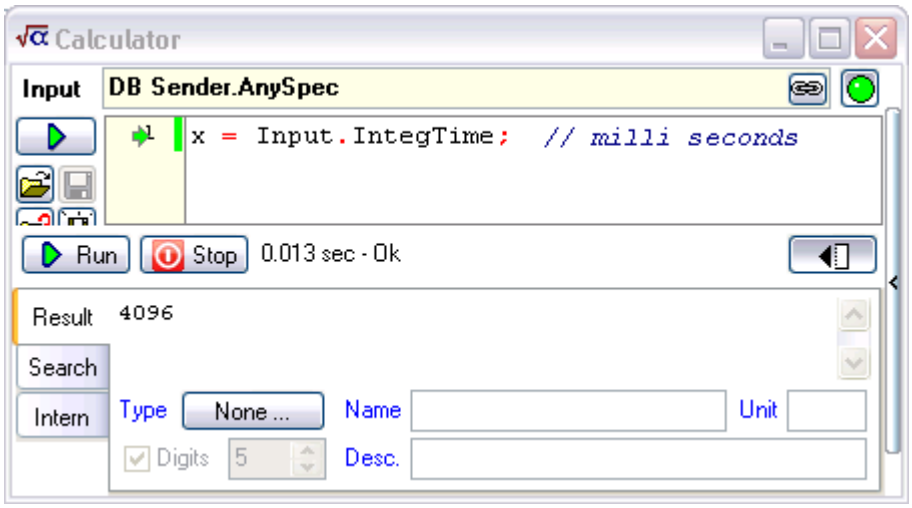
A scripts can be used in standard alone mode or as input dependent. The result of the script is always the last term, shown in the “Result” page.

Her you see a script with no input. Edit the code an press “Run”.

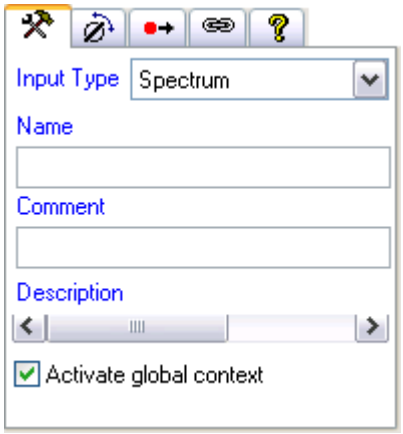


MSDA_XE
Advanced Manual

Here you see code for a “Spectrum” as input. The calculation will be started every time when the “DB Sender” sends new data.



To set a input you must configure the “Input Type” at first. The input choosing field will be shown at the top of the Calculator. Choose the input and enable the window.



The “Global context” is only needed if you want access other objects outside the calculator context (to control or get values from other windows). Without global context scripting execution is much more faster!

8.3 Templates

Scripts can be saved as templates and restored of the saved version.

	Save as new script.
	Save script – overwrite current template version. Only available if it was “saved as new” before.
	Load script template
	Restore script from last saved remplate version. Discard changes.

9 Scripting-Language

9.1 Introduction

The scripting-language of MSDA_XE based on the common C++ programming language. Do not be panic! You must not be a perfect software developer to understand the scripting-language. A simple number calculation as easy as in a pocked calculator.

Some mechanism we do not take from C++ to keep it simpler.

9.2 Commands / Comments

Command must be separated by a semicolon. Line breaks have not meanings.

```
sin(x);
sin(y);
```

You can make singe line comments or multi-line comments.

```
sin(x);         // sine function

/*  sin(x);
    sin(y);   this 2 lines are excluded from execution */
```

9.3 Variables / Data Types

You must not declare variables as needed in C++. Write an assignment and the new variable will be created automatically. Therefore variables can change its data type during scripting execution!

A variables name must start with a letter or a underscore "_". Numbers are only allowed at the following positions.

```
x  = 1;                 //  x will be an integer
x  = 1.0;               // x will be a float
x2 = "Hallo World!";    // x2 will be a string
```

The basic data types are:

bool	true or false
int	A 31-bit signed integer
float	A 8 byte floating point number with 15 digits up to 1e308 (This is the C++ data type double)
String	A text. Write a value in quotes "Hallo World!" Code a line break with "\n".
Date	Write it with an constructor Date(year, month, day)
Time	Write it with an constructor Time(hours, minutes, seconds)
DateTime	Write it with an constructor DateTime(year, month, day, hours, minutes, seconds)

Bool, int and float will be automatically casted in other types if needed.

bool -> int, float	true -> 1 false -> 0 sin(false) -> 0
int, float -> bool	Zero numbers are false, Not zero number are true,
int <->float	

9.4 Number Operators

+ - * /	Standard operators.

	Be aware the “*” hast a stronger binding priority that “+” for example. Look at the operator help panel at the right of the calculator window for binding priorities. If you not sure of it use brackets.
	Be aware of integer division: 1.0/2 -> 0.5 but 1/2 -> 0
`---`	Division operator with small binding priority. (Fraction line) 1 + 2 --- 3 -> 1
`**`	Power 2**3 -> 8
`+= -= *= /=`	x = x operator argument x+=1; -> x = x +1;
`++ --`	Increment/Decrement x++; -> x=x+1 x = 1; y = x++; -> x==2; y==1 x = 1; y = ++x; -> x==2; y==2

9.5 Boolean result operators

==	Comparision 1 == 2 -> false 1.5 == 1.5 -> true;
“and” resp. “&&”	Logical AND. true && true -> true true and false -> false
“or” resp. “||”	Logical OR false || true -> true false or false -> false
“not” resp. “!”	Logical NOT !true -> false not false -> true
!=	Not equal
> (>=)	Larger (or equal)
< (<=)	Smaller (or equal)

9.6 String

```
x = "Hallo";  x + " World!" + x.Count();
// -> "Hallo World!5"
```

9.7 Memberfunktions and Properties

Subobjects can be accessed via the dot notation:

Variable.Function() resp. Variable.Property

```
//The variable S should be a Spectrum
x = S.Longitude;
y = S.Integral(500, 600);  // Integral from 500 nm to 600 nm
S.Comment0 = "Test";
```

A detailed description of the data types and its properties and functions you find in Scripting-helpfile. Open it from MSDA_XE **Main Menu/Help/Scripting Interface** or with the help button in the Calculator.

The variable name for the current inputs is “Input”. Nevertheless the script gets the input-name space. Therefore you must not write “Input”.

Input.PathLength;

PathLength; // same as Input.PathLength

9.8 Conditions

```
if(Condition)
      DoSomeThing();

if(Condition)
{            // Bracket multiple commands in a conditions
      DoSomeThing1();
      DoSomeThing2();
}

if(Condition)
      DoSomeThing1();
else
      DoSomeThing2();

if(Condition1)
      DoSomeThing1();
else if(Condition)
      DoSomeThing2();
else
      DoSomeThing3();
```

9.9 Loops

for(Start; BreakCondition; Progress)

DoSomeThing();

```
for(x=0.0; x<1000; x=x+1)
{
      if(Condition1)
            continue;
            // next loop step - don't execute following code
      y=x**2;
      if(Condition2)
            break;          // leave loop
}

while(Condition)    // Executed only if Condition==true
{
      DoSomeThing();
      if(Condition2)
            break;              // leave loop
}

do           // Execute once at least
{
      DoSomeThing();
      if(Condition2)
            break;          // leave loop
}
while(Condition)
```

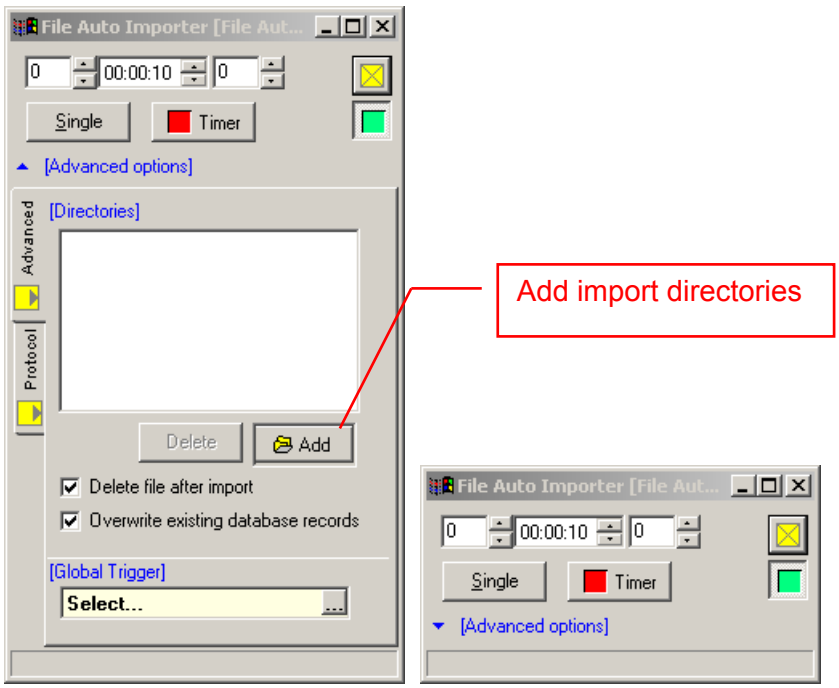
9.10 XY-Series

```
xy=xySeries();

for(x=0.0; x<=1; x+=0.001)
{
      xy.Add(x, sin(x*(2*Pi));
}
```

10 File Auto Importer

Choose [**Extras \ File Auto Importer**] from the Main menu for configuration of an automatic file import control. This control window imports data files (TriOS format) into the database. This window can be used as input for a chart as well.



File Auto Importer	
dd hh:mm:ss ms	Sets days, hours, minutes, seconds and milliseconds
Directories	List of import directories
[Add]	Add import directory
[Delete]	Delete import directory
Delete file after import	Delete imported file in directory
Overwrite existing database records	Records with the same ID will be overwritten
Global trigger	Choose [Extras / Timer] to configure a global trigger. Can be used for more than one device control at the same time.
Select…	Select a global trigger
Single	single measurement.
Timer	switch on/off local trigger.
	Enable/disable the File Auto Importer control.

11 Technical Information

11.1 Installation Files

All program files are stored in the installation folder, which can be found at "C:\Programs\TriOS GmbH\MSDA_XE"). It is possible to install different program versions of MSDA_XE, as long as they are installed in different folders.

The following files are installed:

Directory	Files	Function
MSDA_XE	MSDA_XE.exe	Executable program
	*.dll, *.bpl	Library files for "MSDA_XE.exe" These files must be located in the same directory as MSDA_XE.exe.
	msda_xe.ini	Configuration file: Contains information on - Serial Port configuration - Database connection - Devices connected last - Last size of dialog windows - Further program configurations This file must be located in the MSDA_XE.exe directory.
	Default.dsk	Standard Desktop file.
	trace.log	Protocol of all software processes. (needed by the software department, if bugs occur)
MSDA_XE/db	data.mdb	Standard Access Database file.
MSDA_XE/db	empty.mdb	Empty Access Database file. Can be copied and renamed for starting a new DB
MSDA_XE/New Licenses	*.lic	Different licenses files to configure capability of the software. In most cases sensor specific functions are activated by the import of the *.ini file
Examples	*.dsk	Example desktops
Help	Manual_msda_xe.pdf	Manual as PDF file
	Change.txt	Changes between the versions
	Current Notes.txt	Important changes in this version.

12 Dynamic Data Exchange

MSDA_XE supplies the Dynamic Data Exchange (DDE) Windows interface in order to control the program externally:

DDE-Configuration

Property	Value
Application	MSDA_XE
Topic	Topic (has only one topic)
Item	Item (has only one item)

Commands consist of the window name and a special window command separated by point ("WindowName.Command)".

E.g.:

```
ProPS_D013.Measurement()
ProPS_D013.SetIntegTimeNr(0)    // automatic integration time
ProPS_D013.SetIntegTime(4)      // 4 ms integration time
```

For further information on commands see **Help/DDE Commands** in the Main menu.

All data generated by MSDA_XE can be sent back to DDE interface and be processed by Data Table Window (see chapter Fehler: Referenz nicht gefunden Fehler: Referenz nicht gefunden).

The DDE interface can be checked with the software check_xe.

13 Contact

We are always working to improve our products. Please check our website for updates.

You have found an error in this program, or you would like to see some additional features enabled in a future version?

Feel free to contact our support team: support@trios.de

Our website: www.trios.de

TriOS Mess- und Datentechnik GmbH

Bürgermeister-Brötje-Str. 25

D- 26180 Rastede

Germany

fon +49 (0) 4402 69670 - 0

fax +49 (0) 4402 69670 -20