

EP4Model Manual

for Version 1.0 beta

Revision: 1

Authors: DH

Date: February 2009

Contents

General Introduction	2
The EP4 software family	2
Optical Modeling and Ellipsometry	3
Installation	4
File system	4
Users and session	4
The Graphical User Interface.....	5
Starting up the software.....	6
Creating materials	7
The Epsilon graph	7
Creating a new material	9
Effective Medium Approximation (EMA)	10
User-defined dispersion functions	10
Creating Samples	12
Multiple Samples	13
The Common Layerstack	13
Delta & Psi graphs properties.....	13
Fit & Data tree	15
Spectrum & AOI	15
Datasets	15
Fitting.....	18
Local and Global fitting.....	18
Selecting fit parameters	18
Fit GUI.....	19
Special graph functions	21
Multiplot.....	21
Annotations	21
Appendix.....	22
Dispersion functions.....	22
Licenses, Copyright, Trademarks.....	23
Release and upgrading notes	24
Known issues and workarounds.....	24
Index	25

General Introduction

The following chapters also serve the purpose of a tutorial, so they build up on each other. It is recommended that you go through the chapters and to follow the steps with the running software. If you want to look up something at a later time, you might use the index at the end to find the topic.

The EP4 software family

The software EP4Model is the first module of a series for our Imaging Ellipsometers EP3 and EP4. EP4Model allows you model complex thin film systems and to fit your measured data with the chosen model. As a special feature, EP4Model enables you to follow the effect of any parameter in the model on the resulting simulated data by shifting sliders in real-time. By this, finding the initial fitting values or value ranges of your model is greatly facilitated. Of course, one has to confirm the validity of this “chi-by-eye” approach by looking at the resulting fitting errors bars and statistical parameters supplied.

Another new concept introduced with EP4Model is “Distributed Software”, i.e. the Modeling package not necessarily has to reside on the same computer that controls the instrument. The EP4Model interfaces with the instrument using TCP/IP, thus it works on the LAN and you may install the software on your office PC.

More generally, this is part of the new strategy of the EP4 software family. Rather than integrating everything into a monolithic block of software, we will supply you with a couple of separated applications for each purpose. To this end, one may conceptually separate the process of a sample analysis with an EP3/EP4 instrument in several steps, for example:

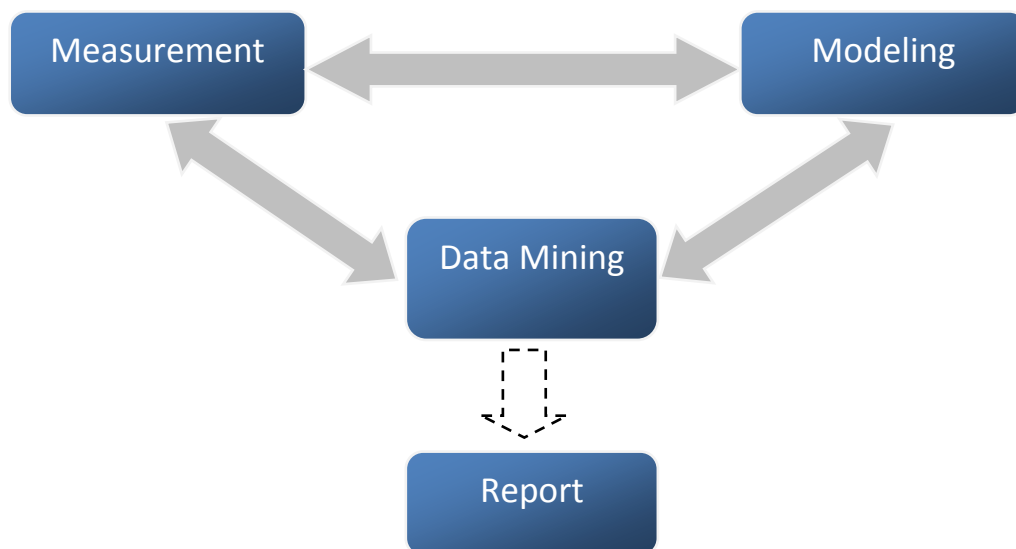


Fig. 1 Typical steps in ellipsometric analysis

As an example, consider a thickness map of a sample: the EP3 or EP4 software first has to generate one or more Δ or Ψ maps. As a next step, a Data Mining Tool can remove unmapped points, precisely align the maps and do a cluster analysis to locate different regions of the sample. This could be film covered and bare substrate areas. Then, the Data Mining sends the processed data to the Modeling,

including information about the proper optical model to apply. The Model, in turn, sends back fitted thickness results to the Data Mining, which displays the thickness map. Often the output of one module might be fed back for another round of measurements, etc. Finally, a report is produced (this will mainly be done in external software, e.g. as a PowerPoint presentation, therefore the dotted line).

The EP4Model software, by concept, does not provide data mining tools on its own. In particular, some functions of the EP3Model are not available: Batch fit or simultaneous MultiROI analysis conceptually belong to data mining (also to fit a Δ Map etc. belongs there) and have been omitted here.

The EP4DataMining application, which may be thought of as a toolbox of various data analysis tools, obviously plays a key role in our software development roadmap. Meanwhile, a simple direct interface to the instrument control software is provided.

Optical Modeling and Ellipsometry

Ellipsometry is a model based method. This means that one does not get a direct value of the wanted physical quantity, e.g. thickness or a refractive index. Instead, one measures the quantities Δ and Ψ which then have to be compared to the output of simulated data from an optical model describing the sample. In practice, a non-linear fitting algorithm is applied to match the parameters of the model to the data by computer. If the match is regarded as “good enough”, and the model sounds “reasonable”, the resulting parameters are taken as the “true values” for the sample.

Obviously, the application of proper models and the successful fitting of the data is most critical for a successful application of Ellipsometry. Especially beginners are often discouraged by the complexity of these procedures and the required understanding of the physical and mathematical limitations of the method. This is in particular true for research on non-standard, ever-changing samples.

With EP4Model we are trying to make modeling easier for the beginner as well as for the experienced user. A continuously expanding base of dispersion models and functionality will be provided with future updates of the software.

We will not provide generally available information about the method of Ellipsometry here. Instead we suggest the following excellent book on the topic:

Tompkins, Harland G. / McGahan, William A.
Spectroscopic Ellipsometry and Reflectometry A User's Guide
1999, John Wiley & Sons

Installation



This area is under construction.

(installer instruction, hardware requirements, LAN settings,...)

File system

The main directory into which the software has been installed will be referenced to as the “root directory” or root. In the root you will find –among others- the following sub-directories:

dispersions	contains the material database
\disp_defs	the definitions of the available dispersion functions
documentation	available documents, eg. this manual
program	Contains the Exe file, related .ini files and several supporting dirs
\model\model.ini.xml	this file holds user specific ini information
recipes	global recipes for samples that all user may access
users	contains a directory for each user
_default user	a default user dir with some example sessions
\demo@date_time	some example sessions

See below for more details about some of these directories.

Note that some .ini file, as well as all database files, session files etc. are XML files. You can open these in any Internet Browser and edit them, if needed, in any text editor.

Users and session

Imagine there is more than a single person in the lab using the instrument and the EP4Model software. What is specific to the individual user and what is global for all users? To keep it simple, we have decided to use the following structure:

- **Users** should create directories with their name in the root\users directory. “Users” may of course also be projects, e.g. “Bens_project1”, “Bens_project2”
- Currently, sub-dirs under \users are not supported (may change with future versions)
- You may also link to directories outside \users by creating an alias. It is recommended to rename the alias to “username.lnk” for correct operation. Aliases may point to remote network resources.
- In the user directory, the software stores a snapshot of the current **session** on shutdown of the program. A session contains the current **recipe**, a screenshot and *all measurement data files*. By this, you can continue working with a session on the next day. However, the old session will not be overwritten, but a new session will be generated every time you request or if you leave the software.
- Recipes of interest for more than the current session may be stored globally in the root\recipes directory.
- Material definitions, as parts of a recipe, may also be stored globally in the root\dispersions directory.

The Graphical User Interface

The main element of the EP4Model GUI is a *tree structure* on the left side of the window. The idea is that the entire information about an optical model should be visible in a single window. One should not need to toggle between multiple windows. The content of the entire tree structure, describing the sample as well as fit settings, *but excluding measured data*, is called a *recipe*. A *session* is a directory containing a recipe *plus measured data*. Make yourself familiar with some elements of the tree:

The screenshot shows the EP4Model 1.0 beta interface. The main window is titled "nanofilm SURFACE ANALYSIS" and shows a user session for "_DefaultUser: newicons based on: default.rcp" which is currently "offline". The interface is divided into several sections: MATERIALS, SAMPLES, and FIT & DATA. The MATERIALS section contains a table with columns for value, min, max, and unit. The SAMPLES section shows a tree structure of layers like Air, BK7_Cauchy, and SI_100. The FIT & DATA section lists various fit parameters and settings.

Materials lists all materials used in the model.

You may expand or contract sub-items or parameters of an object.

Samples containing the structure of one or more sample defined.

Fit & Data is containing all other information needed, especially a listing of the measured datasets and the fitted parameters.

User & Session indicator

Online indicator shows an existing link to the instrument's control software.

The tree table displays information about the tree object. Editing objects is done by small pop-up windows or larger "viewers" that open up on a mouse click.

Context menus, opening by clicking the right mouse button, will show you the options and functions.

Use the scroll bar if the recipe is too long, or contract some objects.

MATERIALS	value	min	max	unit
Air				
SI_100				
BK7_Cauchy				
Cauchy	#0			
A.n	1.5046	1.0000		
B.n	4200	0	6000	nm
C.n	0.000E+0	0.000E+0	1.000E+9	nm ⁴
A.k	0.000	0.000	1.000	1
B.k	0	0	6000	nm ²
C.k	0.000E+0	0.000E+0	1.000E+9	nm ⁴


SAMPLES	value	min	max	unit
sample				
Air				
layerstack				
BK7_Cauchy	#0			
thickness		10.0	100.0	nm
SI_100				

FIT & DATA	value	min	max	unit
spectrum		400.0	900.0	nm
AOI		40.000	90.000	°
AOI_offset	0	-1	1	
datasets				
manual input	sample	file		enabled
Fit for				
thickness	layer#0	sample	local	enabled
Fit settings				
iteration limit	200			
tolerance	1E-7			
use...	Delta&Psi			

Starting up the software

When you start the software by left-clicking on the EP4Model x.yy.exe, a progress bar will be shown. If there is no connection to referenced software modules on their host computers, for example the EP3 computer running the instrument control software, you will be prompted for it. You can choose to work offline, for example to run a simulation only or to continue a previous session, or switch on the other software modules and/or computers and retry to establish a connection.

For the next steps, it is OK to remain stand-alone without connection to the remote software modules.

 In some cases, you might see internal “LabVIEW” warning or error messages, indicating an exception not handled properly by EP4Model . Always select the “Continue” button then. Usually, you may proceed working with the program. If possible, please report the bug to [Nanofilm](#).

Creating materials

We will now explore the possibilities to create materials and study their optical properties.

After starting the software, use *File\Load Recipe* to load the Default recipe. The tree should look empty except the *Fit & Data* area. Go to the empty area below the *Materials* header and right-click to display the pop-up menu. Select Load Material and scroll down to the BK7 material (a standard optical glass). You will notice that there are several files starting with BK7 (make sure keep the listed sorted by Name, not by Type etc). What is the meaning of the different file extensions?

- **.disp.xml** files are tabulated data files from various sources. For a particular wavelength, the tabulated data is interpolated. The file includes reference information (or at least has a field for it). We will expand the database over time. Your contributions are welcome.
- **.mat.xml** files are materials defined by a dispersion function. Optical properties are calculated for each wavelength accordingly. The .mat.xml file is simply an extract from the materials tree structure, as we will see shortly.

Load the file BK7.disp.xml now. The material will appear in the tree (with a little ϵ icon). Left-click on the material: the **Epsilon graph** window opens to the right and you can see the optical dispersion of the BK7. Note that the wavelength scale is as given in the **spectrum** object lower in the tree. Note also that the graph has a header, indicating the currently **active object** (here the selected material).

Right-click the spectrum object to edit it: a little pop-up window appears with a slider control. Change the minimum limit of the spectral range by entering 300 instead of 400. Press <Enter> or left-click on the green button to activate your selection. The pop-up disappears and the Epsilon graph is now updated for the modified spectral range. By the way: note that the ϵ_2 dispersion is constant 0 even below 350nm, where BK7 is actually strongly absorbing. You should apply the BK7 material only in the visible.

The Epsilon graph

Let's explore the properties of this graph. Right-click anywhere in the area of the graph to show its context menu. *X-Axis Select* allows you to switch the physical unit of the X axis. Note that the X axis scales automatically for a loose fit to the data.

Using *Y-Axis Select* you may switch from the **complex dielectric constant**¹ ϵ to the **complex index of refraction** N with its real part n (usually referred to as the refractive index) and the absorption k .

Both representation are coupled:

$$\epsilon_1 = n^2 - k^2 \qquad \epsilon_2 = 2 * n * k$$

Another important feature of the graphs is the **Copy to Compare** function. Use this function store the current dispersion graph in memory as the **comparison graph** and compare it to another dispersion.

¹ This term is frequently used although it is not a constant. More accurate would be the term *relative permittivity*.

Click this function now (a checkmark) in the context menu will indicate that there is something in the memory. To change this to another graph, you need to call the function twice).

You may also change the appearance of the curve by clicking on the legend. A multitude of display style options are available there.

Some of the other functions will be explored later.

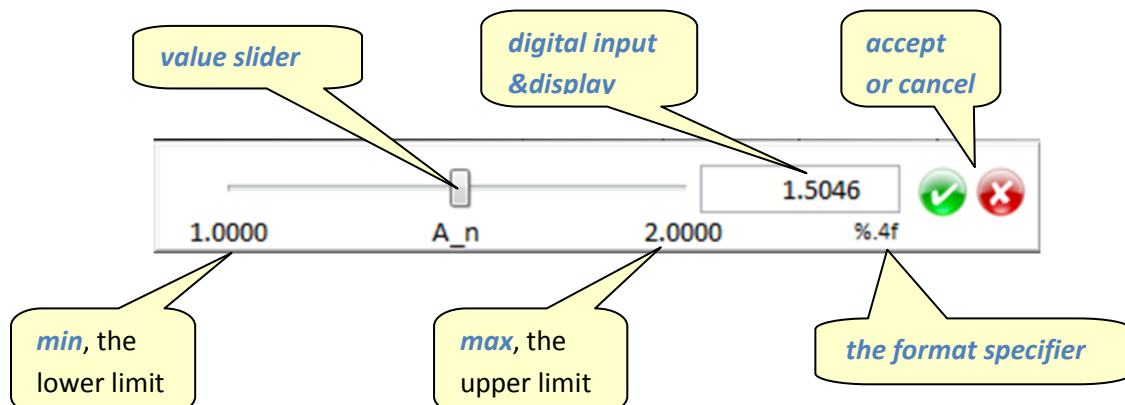
Let's go back to the materials. A right mouse click on the area beneath the BK7 dispersion let's you load the BK7_Cauchy.mat.xml. Opposite to the tabulated BK7 above here we can see a *dispersion term* of type "Cauchy" that describe the dispersion. For a discussion of dispersion functions and the underlying physics, please refer to the text books. See the [Appendix](#) for details about the dispersion functions. Here we note only that are a couple of parameters for this dispersion. Usually only the first 3 parameters are being used, describing the spectral behavior of the refractive index according to

$$n=A+B/\lambda^2+C/\lambda^4$$

In the unit column of the tree table you can see that the parameters (with an added subscript *_n*) are based on λ [nm].

If you click on the line "BK7_Cauchy" in the tree, the Epsilon graph will be updated. Provided that you had selected "Copy to Compare" when we discussed the tabulated BK7 dispersion, you might notice that there are small deviations to the Cauchy-based BK7 and high and low wavelengths. This shows the limits of the Cauchy formula, which is only an approximation.

It would be interesting to see how the Cauchy parameters change the dispersion form. Therefore, we left-click on the *A_n* parameter to get this pop-up window:



The value slider allows you to change the value of a parameter while following the dispersion in real-time. Alternatively you can enter the numbers into the digital display. Accept or cancel using the red or green buttons. However, entering <Enter> will also close the pop-up and accept the values.

The min&max values on the slider may (and should) be edited. They serve multiple purposes:

- limiting the slider movement to a convenient and reasonable range
- manually entered values are coerced to the limits
- providing the boundaries for *Multiplot*
- *providing the boundaries for subsequent fitting of the parameter, see below*

Move the slider to the left and right. You will notice that this shifts the dispersion curve up & down as expected from the formula. It should not change the shape, however, it may *appear* as if it does if you have the *Comparison Graph* enabled (yellow) and AutoScale Y is active. Therefore, switch it off in the graphs context menu and repeat shifting the slider (you'll have to close the slider pop-up first before you can make changes anywhere else). You may need to rescale the Y axis manually on the graph.

Repeating this on the B_n parameter does change the graph's shape, as expected from the formula, and at $B_n=0$ it flattens to a straight line (if $C_n=0$).

If you load the dispersion BK7_Sellmeier you could compare the previous BK7 dispersions to its Sellmeier formulation which is commonly used to describe the refractive index in the optical industry. It gives a better fit to the tabulated data compared to the Cauchy formula.

Creating a new material

To make some room for the next steps, contract the previous dispersion by a left-doubleclick on the dispersion name or by left-clicks on the little arrows on the left hand side of the dispersion. To create a custom material, right-click on the Materials section of the tree, beneath the last material and select *Create Material*.

A new material item will be created, named "material_n", where n may be an index depending on the number of previous "material" items. This *auto-numbering* is global also for other objects in the tree. Note that depending on previous create/delete operations the numbers may not be ordered. Rename the new material by *right-click/rename* and type „my_mat1“ into the pop-up window that appears.

Now, right-click "my_mat1" and select "add term". From the list of the available dispersion functions, select "Oscillator". As you see, this dispersion has 3 parameters: frequency (which corresponds to the resonance "frequency" of the oscillator, but given in eV), strength and damping. The Epsilon graphs should now display the dispersion function (depending of the default values of the parameters).

It is instructive to check the effect of the parameters. To this end, select *photon energy* as the X axis and then change the frequency parameter with the slider while observing the graphs. You will see that the maximum of the ϵ_2 curve move with the frequency. To display *cursors* select *Show Cursors* in the graphs menu and move the cursors to the maximum of the ϵ_2 curve. You will notice that energy value for the maximum is not equal to the *Frequency* parameter. This has to do with the damping. Change the *Damping* parameter to lower values and you will see that the maximum of the curve shifts towards the energy of the transition (you might change the lower limit of the damping to zero to track it down to the case of an undamped oscillator). Finally, at a reasonable damping value, observe the effect of the *Strength* parameter.

Before we proceed, adjust the following set of parameters: Frequency=1.5, Strength=10, Damping=0.5.

Next, right-click on my-mat1 or "Oscillator", select *add term* to add another Oscillator, which is being added to the dispersion. For this second oscillator, choose the following parameters: Frequency=3.3, Strength=40, Damping=1. You should now see curves with 2 distinct maxima of roughly equal height in ϵ_2 at around 400nm and 850nm. From this graph we understand that the resulting total dispersion

of a material in ϵ is the *sum* of the dispersion terms. Thus, we can construct material dispersions from a set of dispersion terms. Choosing the right terms is based on general knowledge about the material class (e.g. metal, semiconductor, dielectric) and experimental findings, for example absorption lines of the material.

If you click on the first or second “Oscillator” item, you will see the particular term as a thin line, while the thick line is the total dispersion. You may also display the other terms via the *right-click/show/all terms* option of the graphs.

Effective Medium Approximation (EMA)

Sometimes, a material is a mixture material of 2 or more other materials for which the dispersions (are at least the general types of their terms) are known. Such a compound material may often be described by an Effective Medium Approximation. There are several approaches depending on the geometry and the involved material classes. The most simple is a linear mixture of a *guest* material in a *host* material according to

$$\epsilon = \text{fraction} * \epsilon_{\text{guest}} + (1-\text{fraction}) * \epsilon_{\text{host}}$$

where fraction being a single parameter to describe the amount of guest in host.

Let’s explore this. Right-click on “my_mat1” material from the last paragraph and select *duplicate material*. A copy of my_mat1 will be created below (you might need to expand it with by double-click). Repeat this to get a second copy. Rename the first copy to “my_host”. Then, delete the *second* Oscillator term and. Rename the second copy to “my_guest” and delete the *first* dispersion term. So we have split the 2 previous terms into 2 new materials (confirm it by clicking on each material, observing the graphs).

Now, create a new, empty material and add the term “EM_linear”. Left-click on *host* parameter and select my_host, which will be entered into the tree. Likewise, select my_guest for the *guest* parameter. After this, the ϵ graphs will be shown.

Use the slider to change the parameter “*fraction (guest)*” and see what happens: for fraction=0 the dispersion resembles the pure host material, for fraction=1 the material becomes pure guest material. For fraction=0.5, the dispersion looks like the original my_mat1, however, divided by 2.

For details on the physical background of other available dispersion models, please consult the textbooks.

User-defined dispersion functions

Besides the pre-defined user dispersions it is also possible to define user dispersions. However, this is currently limited to simple formulae that may be given explicitly for ϵ_1 and ϵ_2 . You can open the GUI of the dispersion editor via the menu bar of the program Menu/Tools/Dispersion Editor.

The list of user-defined dispersion formulae will be stored and thus remain available for future work.

Please note that it is possible to write down formulae which are unphysical, for example lacking Kramers-Kronig consistency. However, there are cases when purely heuristical approximations are still justified. For example, in quality control a polynomial fit to data might provide figures of merit to decide whether a process is running within its limits.

User defined dispersions formula might cause the fits to take longer. Should you frequently use such a function, consider to contact us for implementation of this formula. Also if you need another dispersion formula for which this simple interface is not sufficient, please contact us for an implementation.

Creating Samples

A *sample* definition is the part of the recipe describing the composition of the sample. While the material definition results in a certain dispersion function, visualized in the ϵ graphs, the sample definition allows one to calculate the effect that this sample would have on incident polarized light at a given wavelength and for a certain *angle-of-incidence*² (*AOI*). In Ellipsometry this is given in the form of the two quantities Δ (*Delta*) and Ψ (*Psi*).

The general sample consists of the *ambient* medium (mostly air), the *layerstack* consisting of one or more layers and the *substrate*. The Sample area of the tree allows you to create sample definitions. Please note that EP4Model supports the definition of multiple samples simultaneously. Data may be fitted to multiple samples, too. We will come back to that later.

To create a new sample, right-click the sample area of the tree and select *create sample*. A template is created with the 3 sub-objects *ambient*, *layerstack* and *substrate* as explained above. Rename the same via right-click to *my_sample1*.

Now we need to fill in materials from above into the sample template. To start with a simple sample, make sure that you have loaded the following material: AIR, SiO₂ and Si_100 (which is 100-cut silicon). Next, right-click *ambient* and assign AIR to it from the selection list that pops up. Note that the label *ambient* is replaced by the material that you just selected. You will also notice that Δ and Ψ graphs are opening, however still empty. Note that the graphs have a header, indicating the currently *active object* (like for the ϵ graphs) which is the selected sample here. This is important to remember if multiple samples are defined (see below).

Likewise, assign the silicon to the substrate. The graphs will show the *simulated Δ & Ψ spectrum* already now, which means that we may of course also consider samples that are not covered with any layer, thus pure bulk substrates. Note that these values are calculated according to the *current AOI* value as provided in the Fit&Data part of the tree.

Add an SiO₂ layer to the layerstack. A *thickness* parameter is provided with some default values and limits.

For the exercise, add another SiO₂ layer and note the indices that are assigned to the layers, displayed in the *value* column of the tree. Now, change the thickness of the second layer to 100nm using the pop-up slider control. You can follow the change in the Δ & Ψ simulation upon moving the sliders.

The layers are sorted from the ambient side towards the substrate in our software. The first layer gets the index #0. Be careful about this convention when comparing to other software which might follow other conventions (in particular, the order of layers might be reversed)! To change the ordering of a set of given layers, right-click on the layer (not the thickness parameter) and select *swap with next*. You may also exchange the material of a layer at any time using *change material*.

² Being the effective or internal angle-of-incidence at the sample. If a prism or else is used as a light coupler, the external goniometer angles must be corrected using the law of refraction.

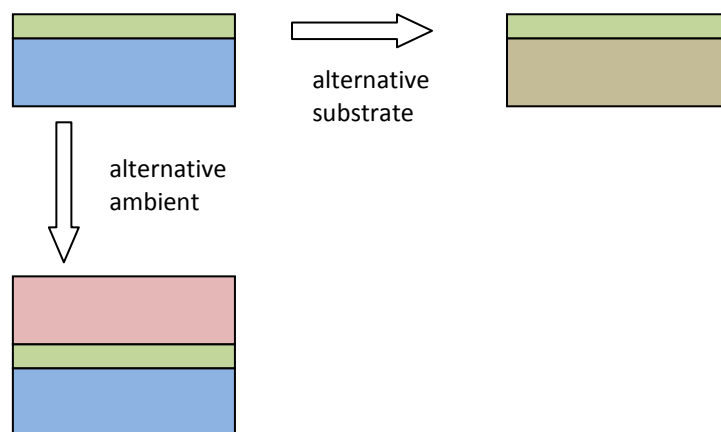
Because we have used two identical layer materials, there is no effect when we swap layer #0 with layer #1. You may also easily confirm that any combination of their thicknesses that adds up to the same total thickness will also give the same Δ & Ψ graphs. However, remember that in general a layerstack AB will NOT have the same optical effect as the inverted layerstack BA.

Multiple Samples

There are cases when it is advantageous to define multiple sample. For example the sample may be unknown and one would like to test various proposed structures or compare different material definitions as layers. Another very important case, especially in Imaging Ellipsometry, are inhomogeneous samples, i.e. where the layerstack is not always the same everywhere. One example of this type would be film-covered areas of a different molecular species (and hence optical properties and thicknesses) on the substrate. Here, intelligent data mining will be able to find out which sample applies to which region on the surface. Therefore, EP4Model supports simultaneous multiple sample analysis.

The Common Layerstack

A specific case of multiple sample analysis is present when a presumably identical layerstack is analyzed: if it is possible to coat two different substrates with the same layerstack, this adds another independent measurement and hence increases the chances to quantify parameters that otherwise are not accessible. Another option is to exchange the ambient.



While the material properties, i.e. the dispersion parameters, are automatically fit globally, if you enter the same material into all sample definitions, the thickness parameter of each sample would be regarded as an independent parameter. This means that we need a structure element that assigns a certain layerstack as globally valid for all samples using this *common layerstack*. In particular, there is only one thickness parameter for a layer in the common layerstack, and it has the same value for all samples.

A common layerstack is created just in the same way as a new sample, except that ambient and substrate are missing. Notice the special icon that marks it as global.

Delta & Psi graphs properties

Most of the right-click graphs properties are equivalent to the corresponding ones in the ϵ graphs. Differing options are:

- *Save Graph Data* will save the graphs to an XML file. The numeric graph data may be exported to other software using copy/paste when editing the files. Note that multiple graphs are saved in a sequential form (the reason for it is that graphs may actually have different lengths, so a 2-D table is not appropriate).
- *X-axis select*: in addition to changing the unit for the spectral axis, here you may also select the angle-of-incidence AOI as the X-axis. In this case, the *current wavelength* as given in the Fit&Data tree is being used for the calculation. You may also use one of the thickness parameters of the given layers as X-axis. Then the *current wavelength & AOI* are being used.
- *Unwrap Delta*: this option allows you to change the representation of the Delta values in the display. When the option is checked, Delta is unwrapped by adding multiples of 360° to reduce discontinuities in the graph. Note that this function influences only the display of the graph. During fits, data points are unwrapped in a different way.

Fit & Data tree

Everything that does not belong directly to the materials or sample definitions is found in the Fit & Data tree.

Spectrum & AOI

The spectrum object shows the *current wavelength* which is needed if plotting Δ & Ψ as a function of AOI or layer thickness. For these X-axes, the slider pop-up has an effect. If the X-axis is a spectral axis moving the slider obviously will not change the graphs. The spectral is automatically updated from the EP3/EP4 hardware if it is online. You may also select the option *Refresh from Instrument* to update or reset the spectral range at any time.

If you change the spectral range manually, also the ϵ and Δ & Ψ graphs will be adjusted accordingly.

Likewise, the AOI objects shows the *current angle-of-incidence* needed to calculate Δ & Ψ as a function of the spectrum or a layer thickness. Thus, AOI slider movements will not have a visual effect in the case of AOI chosen as X-axis.

Both spectrum and AOI will be disabled and grayed out if experimental data becomes available.

Valid experimental data provides its own spectral and angular information, which is being used as current values then. Thus, moving those sliders does not make sense in this case. You may re-activate the sliders if you (temporarily) disable all data (see below).

However: imagine you measured data at nominally 50° AOI. There still might be a small deviation, for example by a tilt of a sample which is difficult to align. This means that one needs a possibility to fine-tune the nominal AOI to the *real* AOI. Therefore, the additional parameter *AOI_offset* has been introduced. Most of the time it will be set to 0°, as the AOI reading will be accurate (actually, a special warning message informs you about a non-zero AOI_offset in the fit engine later). But if needed, this parameter may be adjusted or even fitted for to get better match of data and model.

Using a global AOI_offset parameter has the additional advantage that it is convenient if measurements have been performed under multiple AOI, because this angle offset should be the same for all datasets in the majority of cases.

Datasets

The EP4Model software, as a stand-alone application, receives data from the EP3/4View hardware control software directly or from EP4DataMining. Alternatively, data may be added by hand or imported from a spreadsheet table.

Data is stored internally as data objects added to the *datasets* part of the tree. *A data object may be a full multi-angle spectroscopic measurement (VASE) or a single Δ value measured with a single-wavelength system. You do not have to notify the system whether it was an angle-of-incidence spectrum, a VASE measurement or mixtures of it. All available datasets are pre-processed and merged into a consistent form of data for subsequent fitting.*

However, in the case of multiple samples, the software needs to know to which sample a certain measurement belongs. Thus, the sending software or the user (you) is required to *bind the data to a sample*:

2 samples are defined

4 data objects (=measurements), 2 of which are enabled, the other 2 are disabled

SAMPLES				
	value	min	max	unit
sample_1				
sample_2				

FIT & DATA				
	value	min	max	unit
spectrum		359.0	1000.0	nm
AOI		40.000	90.000	°
AOI_offset	0.000	-1.000	1.000	°
datasets				
✓ measurement_A	sample_2	edited		enabled
✓ measurement_B	sample_1	edited		enabled
✗ measurement_C	sample_1	edited		disabled
✗ EP3data (ROI)	sample_2	file		disabled

the data objects are bound to their respective samples

the source of the data is displayed here

Binding/unbinding as well as **enabling/disabling** of data is possible via right-click on the object.

As soon as data is enabled, it will appear in the $\Delta & \Psi$ graphs too for the sample it is bound to. Thus, data that belongs to sample_2 will not be shown if sample_1 is currently active. Click on the respective sample object to see simulation & actual data.

If you left-click on a data object, the **data editor** GUI will be opened, allowing you to edit and rename the data. You may also disable individual points, e.g. outliers may be reversibly removed from subsequent fits. It is possible to overwrite the errors send with the data with default **zone errors**, depending if it was a 1-, 2- or 4-zone measurement (see documentation of the instrument).

The data editor also allows you to manually create new data using the **Create Data** function of the **datasets** object.

edit to rename

apply default zone errors or switch back to the original errors (M)

disabled data points

measurement_A							zones
	Lambda (nm)	AOI (°)	Delta (°)	Psi (°)	Delta er		
✓	359.000	50.000	151.443	51.765	10.000	1	1.000
✓	423.100	50.000	133.182	41.392	10.000	2	1.000
✓	487.200	50.000	137.836	35.950	10.000	4	1.000
✗	551.300	50.000	140.522	34.001	10.000	M	1.000
✗	615.400	50.000	147.575	31.906	10.000		1.000
✓	679.500	50.000	147.590	31.517	10.000		1.000
✓	743.600	50.000	147.577	31.076	10.000		1.000

editable data array

Please note the following:

- currently no special security is build-in if samples are deleted to which data objects are bound to

- as data contains information about wavelength and AOI, these controls are disabled and grayed-out. If you *Disable All Data* (right-click the *datasets* object) , these objects become operational again
- editing data changes the source displayed in the tree from its original source to “edited” to indicate that this is not original data any more

Fitting

The main point of optical modeling is of course to convert the measured data Δ & Ψ into physical quantities like thickness or refractive index. This is –except from certain simple cases- done by changing the parameters until the model optimally matches the data. Obviously, choosing the correct model is a key to accomplish this task. The sliders of the model parameters and the real-time update of the graphs allows one to match the data under visual control³ or at least to come close to a match and leave the fine-tuning to the fitting engine. The visual approach is also helpful to find a proper model at all for an unknown sample by inspection of characteristic features of the data and simulation graphs. Such could be minima, maxima, slopes or phase jumps.

Local and Global fitting

During the automatic fitting process, the fitting engine tries to minimize a quantity that is a measure for the deviation of the data and the simulation. Depending on the starting values of the parameters, the fit might find more than one minimum within the limiting ranges of each parameter (as given in the tree and editable via the slider control). Such a local minimum will be reflected in a specific solution for the parameters. Another local minimum will be represented by a different solution.

If there are multiple local minima depends on the sample definition and the significance of the measured data. Therefore, it is important to reduce the limits of all parameters to reasonable ranges. This is true in particular as the Δ and Ψ functions may be periodic in the layer thickness. This means that there may be equivalent solutions if the limits are taken too large.

It is always a good idea to test the stability of a local fit by repeating it with another starting value, which is the current value of a parameter as given in the tree. Try values which are outside and inside the interval between the previous starting value and the found result. If all these tests lead to the same solutions, one can be pretty confident that this local minimum is the correct one.

Global fitting tries to overcome the uncertainty of local fitting by applying algorithms to make sure that one is not stuck in a (wrong) local minimum. As this involves a huge amount of calculations of the optical model, one has to consider global fitting carefully: in a multi-parameter fit global fitting may take infinitely. Thus, for those parameters that are already known with good confidence and appear to be “well-behaving”⁴, local fitting should be considered. EP4Model allows you to select global or local fitting individually for every parameter.



Global fitting is not supported in version 1.0 beta, but will be implemented with the next release.

Selecting fit parameters

Add a parameter for fitting by right-clicking the *Fit for:* object and selecting the parameter. **Only those parameters are available for which a sample is actually referenced by enabled data.** Additionally, already selected parameters will not appear on the list. To distinguish parameters with potentially equal names, a special index is added in this list. After selecting a parameter, this information is also displayed in the tree.

³ Sometimes and critically denoted as “chi-by-eye fitting”.

⁴ A rather vague term. Use the slider of this parameter and observe the simulation: if the curve just shifts slightly but not much changing shape the local fit will most find the correct minimum.

Fit GUI

Let's try to understand the fitting functions of the software better using an example. Click on the user&session selector and load the oldest (=original) *Fit_demo_1* session from the *_Default user* .

It is a simple single layer model with a custom material with *n_kfix* dispersion. As you can see the refractive index of the layer is set to 1.427 and the thickness is set to 130.5 nm. Also you will notice that there is already a data object and there are 2 fit parameters, namely thickness & refractive index.

Click on the sample object to show simulation and data. You may *Unwrap Delta* in the graphs. Obviously, the model does not match the data very well. Now, move the thickness slider and observe the simulation. It causes the curves mainly to shift horizontally, with a slight change in the shape.

Repeat this for the n parameter of the film: the curve changes more significantly in shape, especially Ψ . Try to match the data by eye iteratively using the n & thickness slider. It is easy to find out that a thickness of around 100nm and a refractive index of around 1.5 provides a good match.

Now, click on the *Fit for:* object or any of the fit parameters. The Fit GUI opens up. Click on the green arrow to start the fitting. In the result area you will see the best fit results and the errors of the parameters (1σ), as well as the **RMSE** (root mean square error). The RMSE is normalized in the following way⁵:

- RMSE=1 indicates that the fit is optimal taking into account the experimental errors provided with the data.
- RMSE<1, the fit is better than may be expected from the data errors. Thus, the instrument seems to be able to measure with better data quality than assumed.
- RMSE>1 indicates a sub-optimal match (which is the normal case). RMSE values below 2-3 may usually still be regarded as a good fit, depending on the complexity of the sample.

On the top of the window you can see the parameters *iteration limit* and *tolerance*. Both are break conditions for the fitting process. Tolerance is the relative change of the MSE for two subsequent iterations. If the fit improves by less than tolerance it will be stopped with success. If the number of iterations is exceeded and still *tolerance* is larger than provided, a *No optimum found* message will be displayed.

The **RMSE trace** graph is helpful to judge if there is still „reserve“ in the fitting: if there is still non-zero slope at the end, increase the iteration limit. On the other hand, if tolerance is set too low, you might get a *No optimum found* message even if the trace and the fit look nice. Increasing the iteration limit will let you reach a successful fit, but at the cost of calculation time. Reasonable values for iteration limit and tolerance are in the range of 400 and 1E-4, respectively.

The **correlation matrix** is another import measure of fit quality and significance. It provides information about the coupling between the fit parameters: if the effect of changing parameter A can be compensated completely by changing parameter B, then both A and B are completely correlated⁶. Thus, numerically there are other solutions as good as the found one, which is thus not significant.

⁵ in the limit of a large number of data point. For few data points, deviations will occur.

⁶ a negative correlation coefficient simply means that a positive change of A can be compensated by some negative change of B

From another point of view: the data is not significant to extract those 2 parameters from it. One must add more and independent information, for example by providing additional measurements at different AOI etc.

The coefficients of the correlation matrix are also used to calculate the error intervals for the parameters. Note that these coefficients are not constant, but “local” for the found result: for a different result the correlation situation may also be different. This makes sense, as there might well be a thickness range, for example, where the thickness parameter and the refractive index are more decoupled than in our example. In our case, there is a rather strong correlation between n & thickness, but the resulting errors on the parameters are acceptable.

Another important control is the *use delta&psi* control next to tolerance. By this you may globally disable or enable usage of Δ or Ψ information in the dataset. This is especially useful if one parameter is not significant for a specific sample. For example for a very thin film (a few nms), the Ψ measurement might become deteriorated by systematic errors larger than the effect of the thin film. In this case, one might consider using delta only.

For the current example, try out different starting values for n and thickness. You will notice that some combinations do not lead to a good fit.

Special graph functions

Multiplot

This function may be called directly from the tree when right-clicking on any of the numeric parameters like dispersion parameters or layer thicknesses. Multiplot draws multiple plots by altering the parameter automatically within the given limiting range in a number of discrete steps. The number of plots to draw may be changed via Menu/Tools/Options. Note that performing Multiplot on a dispersion parameter re-draws both ϵ and $\Delta\Psi$ graphs, if the material is being used in the current sample, while changing a thickness would only multiplot $\Delta\Psi$ graphs.

Different color options are available with the graphs right-click menus.

Annotations

Annotations are markers in a graph to identify a specific graph. Typically you would use them prior to printing a graph or when you make a screenshot. **Note that annotations are not permanent and will be lost as soon as the graph is being redrawn after changing a parameter etc.**


You may enter an annotation manually via *create annotation* the graphs menus. You are prompted to select several options and for the annotation text (“annotation name”):

- all manually created annotation are currently created in yellow, which is hard to see. You’ll have to change the color afterwards.
- *Lock name* will keep the position of the annotation text when you move the tip of the arrow. If not selected, both arrow and text will be moving jointly.
- to move the arrow, position the cursor over the tip until it changes its appearance. In this state a right-click allows you to change the attributes (especially its color).

With the graph option *Multiplot Annotations* you may create annotations automatically for each of the multiple graphs. All annotations will be created densely packed on one side of the graph. You will need to adjust each position manually for good readability. Some graph artifacts (varying line thicknesses and overlay effects) may be cured with the *Redraw graphs* option once all annotations are positioned.

Appendix

Dispersion functions

 This area is under construction.

A listing of the mathematical realization of various provided dispersion functions and short comments on their area of application.

Licenses, Copyright, Trademarks

Product and company names used within this manual are trademarks or trade names of their respective companies or owners.

LabVIEW™ is a trademark of National Instruments Corporation.

Windows™, Windows Vista™, WindowsXP™ and other names of the operating system, Word, Excel, PowerPoint are trademarks of Microsoft Corporation.

This software was built using the OpenG Array Tools software package under the BSD License applicable to the respective copyright holders of the individual LabVIEW VIs within that package.

<http://www.opensource.org/licenses/bsd-license.php>

The EP4Model software including all Add-on modules
Copyright© 2009 Halcyonics GmbH

Release and upgrading notes

Version 1.0 beta released February 2009

Known issues and workarounds

This is a list of some tips and tricks that might be helpful if the software is stuck or not performing as expected, as well as a (incomplete) list of some known bugs.

symptom	possible reasons and/or workaround
Data points are not shown in the Δ - Ψ graphs	Make sure that the data is bound to the right sample or left-click on the appropriate sample. Data is only shown for the sample it is bound to
Startup stuck when "loading previous session"	Might be a problem with the previous session files. If needed, stop the process EP4Model....exe in the Windows Task Manager. Then edit the file root\model\model.ini and delete everything between the tags <previous_session> and <\previous_session>, save the file and restart.
EM Material shows no dispersion	Check if you have erroneously deleted one of the materials that constitute the EM material
After startup, fit to existing data does not give correct results	The data is not properly updated and ordered. Move one of the parameter sliders and fit with a slightly different value. This should lead to a complete refresh of the internal data objects.
RMSE scaling for single or few data points might be mathematically inconsistent.	
MultiROI fits are not automatically done with separate results for every ROI. Instead, one has to manually fit every ROI. Otherwise, all ROI data are used to fit ONE result.	MultiROI fitting, similar to Mapping, is part of data mining. EP4Model does not have an option to display MultiROI-results, so MultiROI results need to send to a calling data mining tool anyway.
Deleting a sample which is assigned to data objects leads to multiple Lab view error messages	

Index

- .disp.xml 7
- .mat.xml 7
- active object 7, 12
- ambient 12
- angle-of-incidence 12
- Annotations 21
- AOI_offset 15
- auto-numbering 9
- Batch fit 3
- Binding/unbinding 16
- bugs 24
- Cauchy 8
- chi-by-eye fitting 18
- comparison graph 7
- complex dielectric constant 7
- complex index of refraction 7
- Copyright 23
- correlation matrix 19
- current angle-of-incidence 15
- current wavelength 15
- cursors 9
- data editor 16
- Data Mining 2
- Datasets 15
- Delta 12
- Delta & Psi graphs 13
- dielectric constant 7
- dispersion term 8
- Distributed Software 2
- Effective Medium Approximation 10
- EMA 10
- Enabling/Disabling 16
- Epsilon graph 7
- File system 4
- fit parameters 18
- index of refraction 7
- iteration limit 19
- Kramers-Kronig 11
- layerstack 12
- Licenses 23
- Local and Global fitting 18
- local minima 18
- manually creating data 16
- multiple samples 12
- Multiplot 21
- Multiplot Annotations 21
- parameter coupling 19
- Psi 12
- recipe 4, 5
- Redraw graphs 21
- Release notes 24
- RMSE 19
- RMSE trace 19
- root directory 4
- root mean square error 19
- Sellmeier 9
- session 4, 5
- simulation 12
- slider 8
- spectrum 7
- spreadsheet 15
- substrate 12
- thickness 12
- tolerance 19
- Trademarks 23
- tree 5
- Unwrap Delta 14
- upgrading 24
- use.. 20
- user 4
- Users 4
- VASE 15
- zone errors 16