# ERA 2.0

user manual

Legal notice

ERA 2.0 is supported by MS Windows 9x/2000/XP operating systems. It is distributed as a self-extracting archive containing all the files needed by ERA to run. Note however, that Borland Delphi compiler is necessary to compile models, so that ERA cannot be legally used without having a legal copy of Borland Delphi installed on your computer. Usually some versions of Borland Delphi can be downloaded for free from www.borland.com.

Installation

Simply run and extract the SFX archive into a folder of your choice. The install path must not contain spaces, i.e. it can be c:\era\, but not c:\Program Files\era\. This limitation is caused by Delphi compiler requirements. When it is done ERA can be run by era20.exe program.
When the program is run first time following configuration must be done: In the *Tools* menu choose *Preferences.* Following window will display:
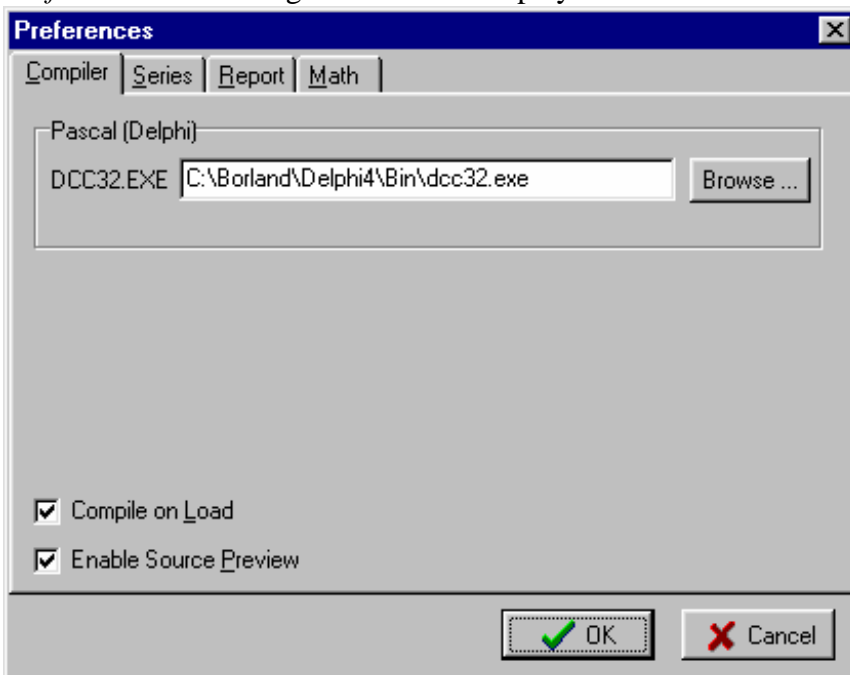
In the *Compiler* page of the window the path to the Delphi compiler dcc32.exe must be set. You can choose path to the dcc32.exe in the directory where you installed ERA if you have a legal copy of Delphi installed on your computer.
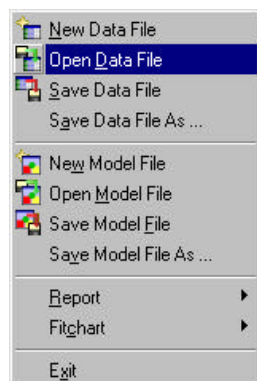
User Interface
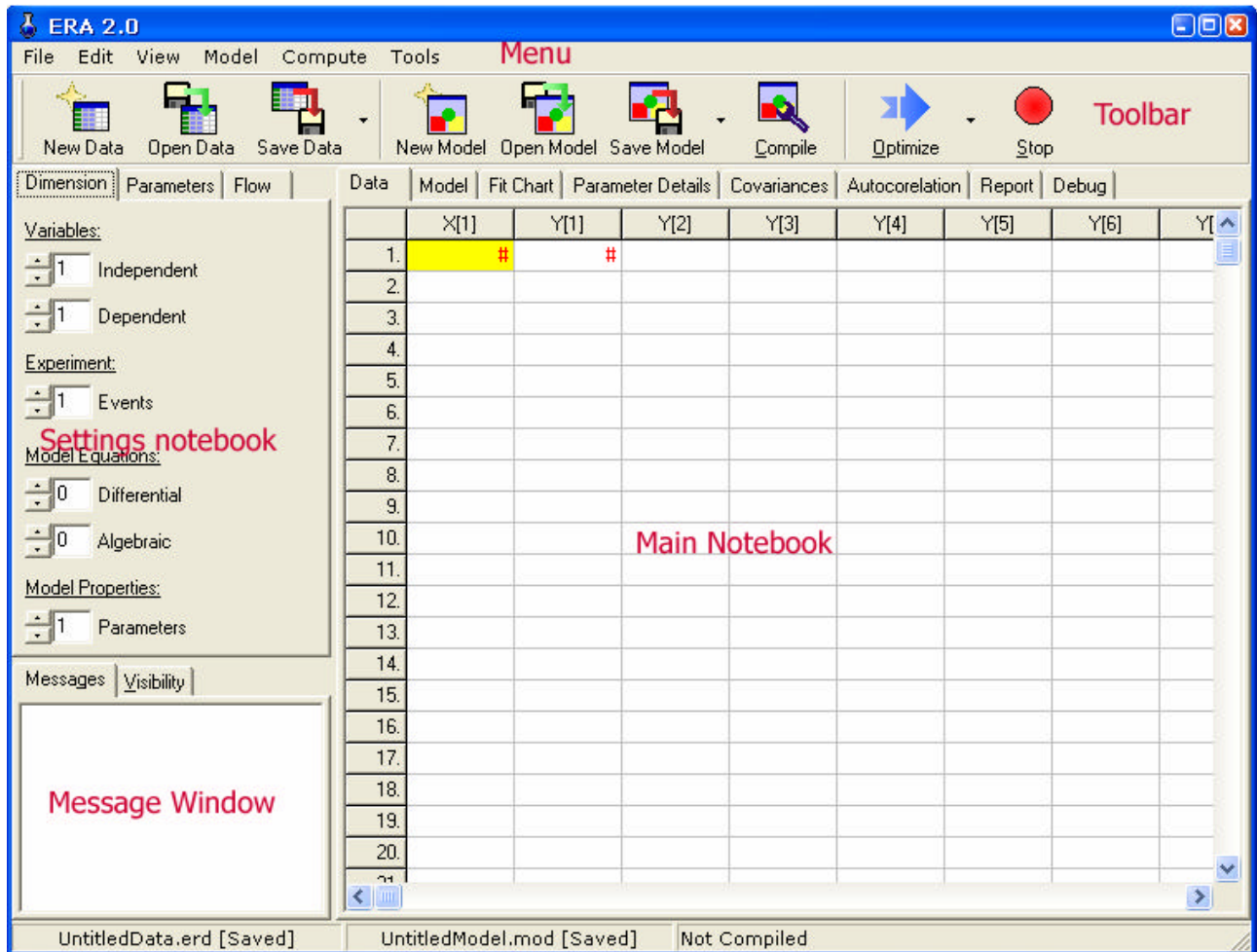
ERA can be controlled like most other Windows applications by *Menu* and *Toolbar*. Input and output information is exchanged between the user and program via *Settings notebook* and the *Main notebook*. Both notebooks have a number of pages. Message window is used for displaying non-critical messages during computation.
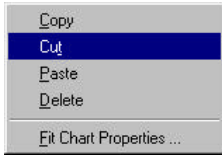
Menu

File Menu

File menu contains commands for data and model files management. *New Data File* and *New Model File* command are used to create a new table of experimental data and new model file. The files can be saved or opened by commands *Save Data File, Save Model File, Open Data File, Open model File, Save Data File As,* and *Save Model File As.*

Submenu *Report* contain commands for creating reports. Reports can be printed, previewed or saved in RTF format. Also *Fitchart* can be exported by *Export…* (as Enhanced Metafile) or printed by *Print…* command.

Edit Menu

Edit menu contains standard functions for clipboard support *Copy*, *Cut*, *Paste* a *Delete.* Their functions are somewhat bugged and they do not work in all controls. It is recommended to use shortcuts Ctrl+C, Ctrl+X, Ctrl+V and Del that works well.
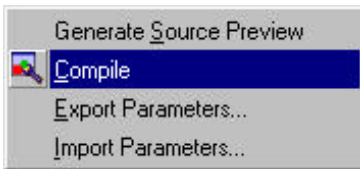
Fitchart Properties... command is used to display a dialog window, where the chart and axes descriptions and scales can be customized.

View Menu
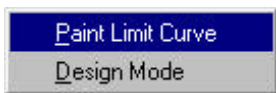
View menu contain two toggles that can be turned on or off. *Paint Limit Curve* toggles display of confidence limit curves *(see later). Design Mode* toggles the display of results of sequential design of experiments.

Model Menu

Model menu supports the development of the mathematical model. *Generate Source Preview* command create a source code of the model library in Pascal language from model equations written in the *Model* page in the *Main notebook* and places it in the page *Source* in the same notebook. *Compile* command does the same and then compile source code into the model DLL.

*Export Parameters...*command exports computed parameter values into the file. Reversibly the values can be imported by the *Import Parameters...*command.

Compute menu
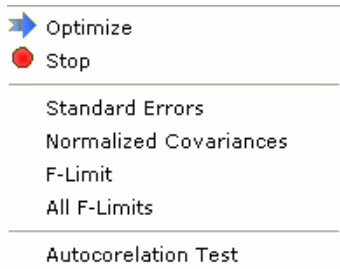
Compute menu collects commands that can be used to start computation functions provided by ERA. The most basic function is *Optimize* command, that starts optimization of parameters using objective function specified in the *Flow* page of the *Settings notebook*, current data and model, and takes parameter values specified in *Parameters* page of *Settings notebook* as initial estimates of parameters for non-linear optimization. The computation proceeds until it is stopped by *Stop* command.

Standard Errors and *Normalized Covariances* commands starts computations of standard errors of parameter estimates (optimization must be performed before these functions) and normalized covariances (correlation coefficients) among parameters. Confidence limits for all parameters can be computed by *All-F Limits* command. Single confidence limit can be computed by the command *F-Limit* if a valid cell in table on the *Parameter Details* page of the *Main* Notebook is selected (any cell in the Lo-F Limit or Hi-F Limit columns).

Tools Menu
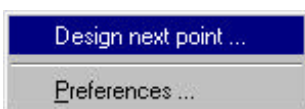
*Design Next Point...*command in the *Tools* menu displays a dialog window for the sequential design of experiments support. *Preferences* command brings on the preferences window used to change program settings. *Preferences dialog* is described in details further in the manual.

Toolbar

Toolbar contains shortcut commands to most frequently used commands in the menu.

Settings Notebook

This notebook consists of *Dimension, Parameters, and Flow* pages. Their description follows:

Dimension Page

Controls at the *Dimension* page must be used to specify the dimension of the problem. Each value can be set either by the couple of arrows in the left or edited by the attached edit box.

Variables section allows setting the number of *independent* variables and *dependent* variables }responses. Number of experiments can be specified in the *Experiment/Events* field. Number of algebraic and differential equations can be specified in *Differential* and *Algebraic* fields in *Model Equations* section and the number of parameters is specified in *Parameters* field.

Parameters Page

This page contains the actual parameter values. The number of active fields corresponds to the number of parameters specified in *Dimension* page. The parameter values on this page are bound to the values on the *Parameter Details* page in the *Main notebook* – the values at these two locations are always the same. They both are also bound to the plot displayed on *Fitchart* page – whenever parameter values update, either by user or the program, the fitchart redraws. Therefore, there are two uses of the parameter values on this page. It can be used for setting initial parameter estimates and after optimization optimal parameter values can be read there.

For international users: Numeric values must be given in proper format corresponding to that specified in windows regional settings (usually . or ,). This rule applies for all numeric fields with the only exception being the *Model* page of the *Main notebook,* where the numbers in model equations must be given with decimal point regardless the Windows settings. Scientific format (e.g. 1.256E-7) is also allowed.

Flow page

Flow page provides the status information about the computation flow. Depending on the setting of the *Optimization/F-test* toggle displays the info on optimization or computation of confidence limits (these two tasks are potentially time consuming). When in *Optimize* mode it display information about attained minimum of *Objective Function*, number of *Model Calls,* and *Successful* aproximations. It also allows to change objective function type to following options:

- Sum of Squares – minimizes the sum of squares of residual deviations, good for most applications
- Box-Draper determinant – minimizes complex objective function proposed in Box, G. E. P. & Draper, N. R. (1965) *Biometrika* **52**, 355. It can be used if the assumption of independent errors in experimental points is not valid.
- Weighted sum of squares – minimizes the weighted sum of squares of residual deviations. Weights are different for each response and equal to that response maximum. Therefore it may be usefull if the assumption of constant variance of relative errors is more valid than the assumption of constant variance of absolute errors

Main notebook

Data page

| Data | Model | Source | Fit Chart | Parameter Details | Covariances | Report |

	X[1]	Y[1]	Y[2]	Y[3]	Y[4]	Y[5]	Y[6]	Y
1.	0.00000	1.00000	0.00000	0.00000	0.00000			
2.	2.00000	0.89200	0.08400	0.01200	0.00700			
3.	5.00000	0.65100	0.29500	0.04200	0.02200			
4.	10.0000	0.44600	0.44500	0.05100	0.09600			
5.	20.0000	0.19300	0.54200	0.05800	0.22300			
6.	35.0000	0.06000	#	0.05700	0.43600			
7.	55.0000	0.00100	0.25900	0.02900	0.71100			
8.	90.0000	0.00000	0.02400	0.00000	0.97600			
9.								
10.								
11.					[yellow cell]			
12.								

Data page contain a table of experimental data. Column headers indicate the number of variable and whether it is independent (X) or dependent (Y) variable. The number of columns for independent variables corresponds to the number of independent variables specified on *Dimension* page of the *Settings notebook*. Cells that are active according to the dimension settings are marked by a # symbol. These cells can be edited. It is not necessary to fill all dependent variable cells the data matrix can be sparse and can contain missing data field (e.g. y[2] in 6. row on picture above).

ⓘ   It is important to have sorted data rows by ascending value of independent variable.

The data table support clipboard operations on cells and block of cells. For instance the data from Excel can be quickly copied to the data table via clipboard. When pasting clipboard content into the data table make sure that the upper left cell of the block which you wish to paste data into is selected AND focused:

	X[1]	Y[1
1.	0.00000	0.1!
2.	1.00000	0.2!

This is O.K.

	X[1]	Y[1
1.	0.00000	0.1
2.	1.00000	0.2

The data table is not focused (any other control is active)

	X[1]	Y[1
1.	0.00000	0.1
2.	1.00000	0.2

The grid is in the cell edit mode (clipboard will paste into this cell)

Model and Source pages

```
Data   Model   Source   Fit Chart   Parameter Details   Covariances   Report

language = pascal

declar
  r1, r2, r3, r4, r5 : Double;
end declar

static
end static

dynamic
r5 := y[1] + p[5]*y[2] + p[6]*y[3] + p[7]*y[4];
r1 := p[1]*y[1]/r5;
r2 := p[2]*y[1]/r5;
r3 := p[3]*y[2]*p[5]/r5;
r4 := p[4]*y[3]*p[6]/r5;
dy[1] := -r1 - r2;
dy[2] := r1 - r3;
dy[3] := r2 - r4;
```

Model page contains a field where the model must be specified. Model syntax is described in separate section.

```
Data   Model   Source   Fit Chart   Parameter Details   Covariances   Report

Library cinn;

uses Math;

type T20Vector = array[0..20] of Double;
type T50Vector = array[0..50] of Double;
type T20x500Matrix = array[0..500] of T20Vector;
type T50x500Matrix = array[0..500] of T50Vector;
type P20Vector = ^T20Vector;
type P50Vector = ^T50Vector;
type P20x500Matrix = ^T20x500Matrix;
type P50x500Matrix = ^T50x500Matrix;

procedure DynamicEquations(var p,x : P20Vector; var y,dy : P50Vect
var
```
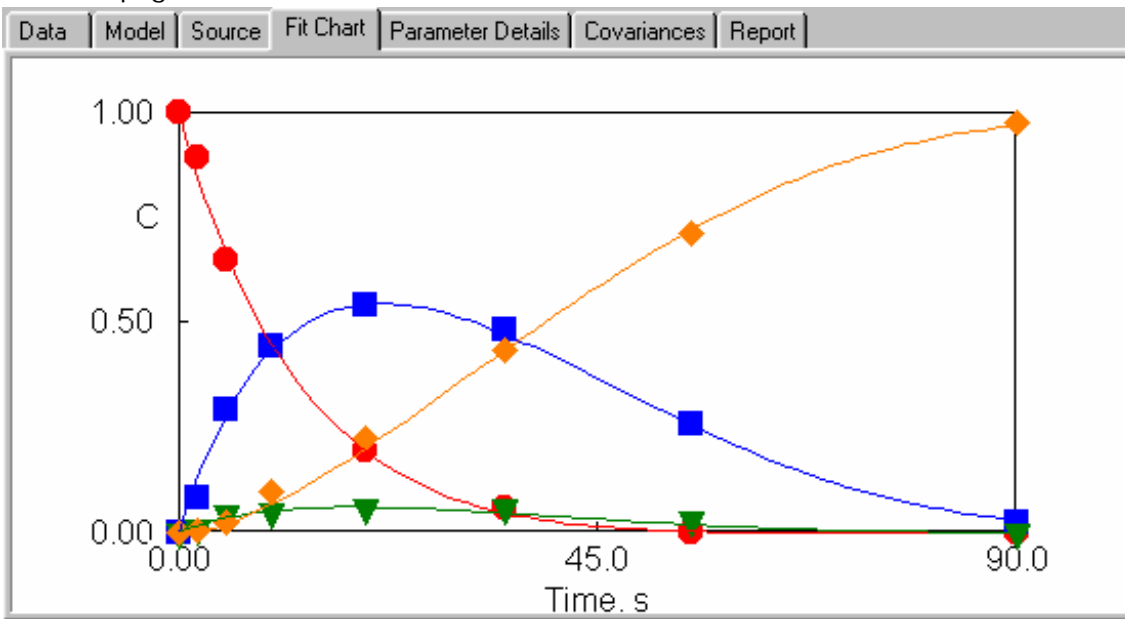
Source page contains preview of the PASCAL source generated by *Generate Source Preview* or *Compile* commands. It is only for information and debugging purposes and can be hidden in *Preferences*.

Fit Chart page

Fit Chart page shows the comparison of experimental data (points) and model solution (curves). Captions, fonts, and scales can be set by choosing *Fitchart Properties...* in the context (right-click) menu. Points and curves attributes can be set in *Peferences.*

Parameter Details page

Parameter Details page numerical input/output information about parameters. Columns indicate (from left to right) parameter value, allowed minimum and maximum, standard error and lower and upper confidence limits. First three columns are editable. The table support clipboard operations.

Covariances page

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
1.	1.00	-0.20	-0.48	0.50	0.87	-0.52	-0.07				
2.	-0.20	1.00	-0.49	0.02	0.05	0.07	-0.40				
3.	-0.48	-0.49	1.00	-0.75	-0.75	0.68	0.66				
4.	0.50	0.02	-0.75	1.00	0.70	-0.99	-0.25				
5.	0.87	0.05	-0.75	0.70	1.00	-0.69	-0.25				
6.	-0.52	0.07	0.68	-0.99	-0.69	1.00	0.20				
7.	-0.07	-0.40	0.66	-0.25	-0.25	0.20	1.00				
8.											
9.											
10.											
11.											

Covariance matrix determinant = 0.0000353

Covariances page display the matrix of correlation coefficients and its determinant after the *Computation | Normalized covariances* was performed. High correlations are marked in red.

Report page

```
17:57:46 - 31.10.2002

ERA 2.0 - Report

Model:   C:\era20\voda\Obr1PCEFO.mod
Data:    C:\era20\voda\Obr1PCEFO.erd

Objective Function = 0.0378140345469572

Parameter Settings:

        Minimum            Init. Est.       Maximum
1.      0.0000000          1.0000000        10000.000
2.      0.0000000          1.0000000        10000.000

Estimated Parameters:

        Value              Std. Error       Lo F-Limit       Hi F-Limit
1.      0.0337975          0.0000000        0.0337975        0.0337975
2.      0.9205499          0.0000000        0.9205499        0.9205499

Normalized Covariances:

        1.      2.
1.      0.00    0.00
2.      0.00    0.00

Residuals:

        Y1       Y2       Y3       Y4       Y5       Y6       Y7
1.      0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
2.      0.0388   0.0103   0.0059  -0.3558  -0.6641  -0.5541  -0.0180
3.      0.0345   0.0020   0.0015   0.3446   0.6433   0.5367   0.0354
```

Obr1PCEFO.mod [Saved]   Compiled

Report page contains the preview of report created by the *File | Report | Preview* command. Report can be printed or saved by commands of context menu.

How to create models

Mathematical model must be typed into the field on *Model* page of the *Main notebook*. The process of model development consists in three steps – writing source code, saving source code, and its compilation.

Writing source code

Currently the only supported model constructing language is derived from the PASCAL language. However, it is possible that different languages and compilers will be supported in future and therefore the language must be specified – in the very beginning of the source code must be written:

```
language=pascal
```

Next part of the source code allows to define user-defined variables. It is marked by following keywords:

```
declar
.
.
end declar
```

the list of declarations that follows must correspond to the PASCAL syntax – list of defined variables separated by comas, followed by a colon and type identifier and finalized by a semicolon. For instance three floating point variables a, b, c are defined by:

```
a, b, c : Double;
```

After the declaration section follows the section of static code

```
static
.
.
end static
```

and section of dynamic code

```
dynamic
.
.
end dynamic
```

Last two parts can contain model equations. Whenever the model is solved the static equations are evaluated at first and then dynamic equations are evaluated in all experimental points, as needed by integration method, or by demands of graphic engine.

> Static part of code is usually used only if it is necessary to access experimental data in the model, e.g. if you wish assign parameter to missing/unmeasured experimental value. The use of this section is quite rare and it is left empty in most cases

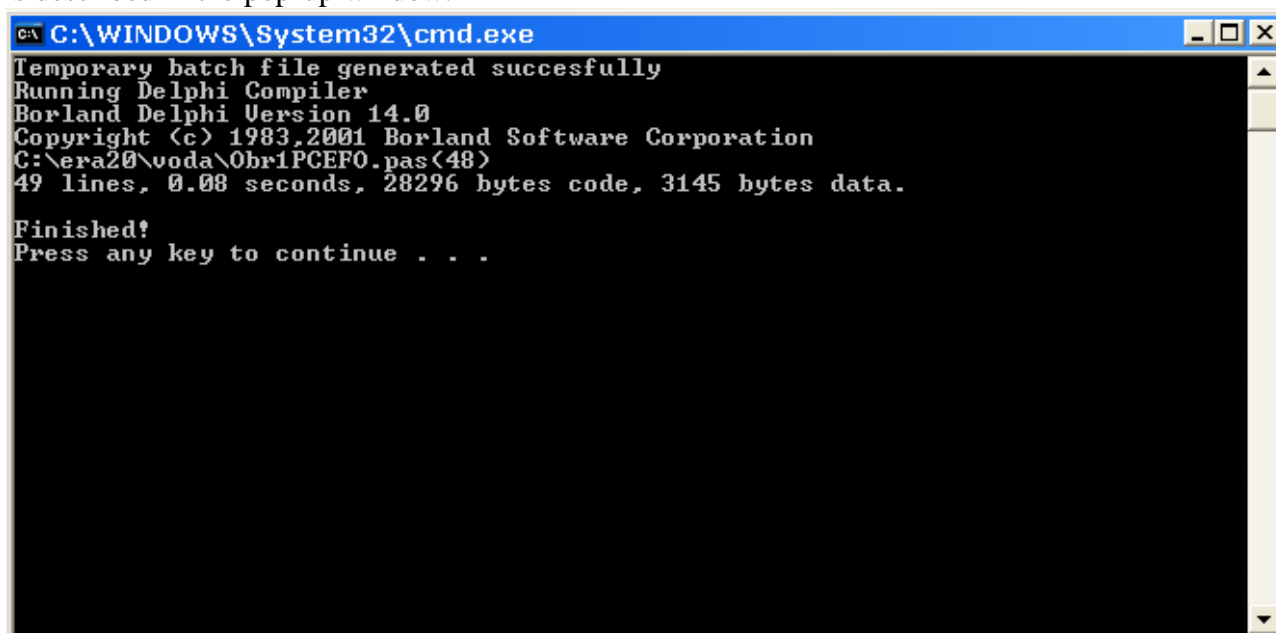> Section keywords cannot be omited, but the sections can be left empty.

In both the static and dynamic sections can be used user-defined variables. Except them the following predefined global variables can be used:

✧ In the static section:
  ✧ vectors p, x and y as symbols for parameters, independent variables, and dependent variables. Indices are given in square brackets, e.g. x[1].
  ✧ matrices xe and ye as symbols for matrices of experimental values, e.g. xe[1][1]. First index represents the row and second the column of the data matrix.
✧ In the dynamic section:
  ✧ vectors p, x, y, dy, as symbols for parameters, independent variables, dependent variables, and derivatives of dependent variable by x[1]. It means that dy[1] is shortcut for dy[1]/dx[1].

When writing equations the syntactic rules of PASCAL language must be followed. Each dependent variable must be either expressed by exactly one explicit algebraic equation, or specified by exactly one ordinary differential equation.

Saving and compiling model

Model must be saved by commands in the *File* menu. Model file name is used also as the name of model function and therefore it cannot contain spaces and must begin with letter or _ (it can contain digits, but cannot start with a digit). Saved model must be compiled by *Model | Compile* command. External compiler performs the model compilation. The information about the compilation progress is described in the pop-up window.

As soon as the message "Press any key to continue" appears, the compilation is done. The picture above shows a result of successful compilation. If it is not successful, the list of error messages is included among the messages. The window MUST BE CLOSED BY PRESSING ANY KEY after inspecting the messages. If the window is closed by close button the compiled model is not saved. Whenever is the model changed it must be saved AND re-compiled to any changes take effect.

Computations

All types of computations described in this section assume that data and model files are already set.

Parameter estimation

The *Compute | Optimize* command runs the parameter estimation. The estimation runs iteratively from the initial parameter estimates. If the estimation does not run well you can try to change initial estimates. Parameter estimation can be interrupted by the *Stop* command whenever you wish. Also it can be re-run at arbitrary time. The program provides ability to track changes in parameter estimates, objective function value and comparison of experimental data with model predictions throughout the whole optimization process. The updated information is displayed after each improvement of parameter estimates, which slightly slows whole process but provides detailed information about the optimization progress. The optimization must be stopped manually when the changes of objective function are less than desired precision. The description of computation algorithms can be found in paper Zámostný, P., Belohlav, Z., "A software for regression analysis of kinetic data", Comput. Chem. 23, 479 (1999).

Parameter reliability and correlation

The statistical significance and reliability of parameter estimates can be assessed according to parameters confidence intervals, which are for the k-th parameter computed by equation

$$L_k = b_k \pm t_{a/2}(n)\sqrt{s^2_{\exp} c_{kk}}$$

where c_{kk} is element of variance-covariance matrix of parameters $\boldsymbol{C}$, p_k estimate of k-th parameter, $t_{\alpha/2}$ critical value of Student's statistics for significance level α, and $s^2_{\exp}$ variance of responses. It is very closely bound with the computation of correlation coefficients (normalized covariances between parameters) from the non-diagonal elements of matrix $\boldsymbol{C}$. However, the equation assumes local linearity of the model around the optimum and therefore it is valid only if this assumption is accomplished (at least approximately). Since models seldom fill the given assumption, it is advisable to use more rigorous criterion.

Therefore the ERA package provides an interval estimate based on the real confidence region defined by likelihood criterion. The confidence region is the set of all $\boldsymbol{p}$ vectors matching the criterion bellow

$$\frac{\left[S(\boldsymbol{p}) - S(\boldsymbol{p}^{\mathrm{opt}})\right]/p}{S(\boldsymbol{p}^{\mathrm{opt}})/n - p} \leq F_a(p, n-p) \qquad [1]$$

where $\boldsymbol{p}^{\mathrm{opt}}$ is vector of optimal parameter values, p number of parameters, n number of experimental points, and $F_{\alpha}(p, n\text{-}p)$ critical value of F statistic for (1-α) significance level. Instead of construction of the whole confidence region or its surface, which are extremely time consuming tasks, only its projections to all parameter axes are computed. Marginal points of these projections are called confidence limits (Belohlav, Z., Zamostny, P., Kluson, P. & Volf, J. (1997) *Can. J. Chem. Engng.* **75**, 735). Confidence intervals are provided as standard deviations (half-size of confidence interval) due to their symmetry around optimal value. On the other hand, confidence limits need not be necessarily symmetric around optimum, so that both upper and lower confidence limits are provided.