

Example:

```
; open chart and run expert and/or script
Symbol=EURUSD
Period=H4
Template=popular.tpl
Expert=MACD Sample
ExpertParameters=macd.set
Script=period_converter
ScriptParameters=
```

f) Settings of the Strategy Tester Launch

- **TestExpert** - the name of the expert to be launched for testing. If this parameter has not been specified, no testing is launched.
- **TestExpertParameters** - the name of the file containing parameters (the \tester directory). The file can be created in the Properties window of the expert under test by clicking the "Inputs - Save" button. It is normally used to save parameters other than the default ones. Other parameters of the expert under test in the "Testing" and "Optimization" tabs (as well as in the "Inputs" tab if this parameter has not been specified) are filled up with the values automatically saved in the \tester\[the expert name].ini file after the latest test.
- **TestSymbol** - the name of the symbol used for the expert testing. If this parameter has not been specified, the latest value used in the tester is used.
- **TestPeriod** - the chart period (M1, M5, M15, M30, H1, H4, D1, W1, MN). If this parameter has not been specified, H1 is used.
- **TestModel** - 0, 1, or 2, depending on the testing model (Every tick, Control points, Openprices only). If this parameter has not been specified, 0 is used (Every tick).
- **TestOptimization** - enable/disable optimization. The values that can be taken are "true" or "false". If this parameter had not been specified, the "false" value is used.
- **TestDateEnable** - enable/disable the "Use date" flag. The values that can be taken are "true" or "false". If this parameter had not been specified, the "false" value is used.
- **TestFromDate** - the date, from which to start testing, appeared as YYYY.MM.DD. If this parameter has not been specified, this date is 1970.01.01.
- **TestToDate** - the date, on which to finish testing, appeared as YYYY.MM.DD. If this parameter has not been specified, this date is 1970.01.01.
- **TestReport** - the name of the test report file. The file will be created in the client terminal directory. A relative path can be specified, for example: tester\MovingAverageReport". If the extension has not been specified in the file name, the ".htm" will be set automatically. If this parameter has not been specified, the test report will not be formed.
- **TestReplaceReport** - enable/disable the repeated report file record. The values that can be taken are "true" or "false". If the "false" value is specified and a report file named in the same way exists already, the number in square brackets will be added to the file name. For example, "MovingAverageReport[1].htm". If this parameter had not been specified, the "false" value is used.
- **TestShutdownTerminal** - enable/disable shutdown of the terminal after the testing has been finished. The values that can be taken are "true" or "false". If this parameter had not been specified, the "false" value is used. If the user has pressed the "Stop" button, the value of this parameter will be flushed to "false" since the control has been given to the user.

Example:

```
; start strategy tester
TestExpert=Moving Average
TestExpertParameters=ma0.set
TestSymbol=EURUSD
TestPeriod=H1
TestModel=2
TestOptimization=false
TestDateEnable=true
```

```
TestFromDate=1970.01.01
TestToDate=2006.06.06
TestReport=MovingAverageReport
TestReplaceReport=false
TestShutdownTerminal=true
```

I. HISTORY CENTER

Technical analysis is the market movements research made in order to forecast future price movements. The market is often analyzed using charts. So it is very important to have available historical data for all symbols and timeframes used. Historical data are constantly formed and stored on the server. Connecting to it, the client terminal downloads all necessary data. They will be then used for drawing of charts, testing and optimization of Expert Advisors. To control historical data, the terminal has a special window named "History Center". This window can be opened by executing the command "Tools – History Center" or by pressing F2.

After the terminal has been shut down, all accumulated historical data will be stored in the "History Center". Sizes of files containing historical quotes do not exceed values defined in settings. If the amount of historical data accumulated exceeds the value set in the field of "Max. bars in history:", the oldest bars will be deleted when storing. For each timeframe, a separate history file is formed named as SSSSSPP.hst (where SSSSSS - symbol name, PP - timeframe in minutes) and saved in the /HISTORY. Later on, the saved data will be used to draw charts, as well as for testing of trading strategies.

In the "History Center" window, the available data can be changed. For this, it is necessary to select the desired symbol and timeframe in the left part of the window. The corresponding data will be loaded in form of a table. To add a record about a new bar, it is necessary to press the button of the same name, fill out all necessary fields in the new window and press "OK". After that, the new bar will appear in the history. One can modify the bar by selecting the corresponding record and pressing the "Modify" button. To delete a bar, it is necessary to select it and press the button of the same name.

a) Load of Historical Data

It is possible to load quotes for basic currency pairs starting with year 1999 from the historical data server. To do it, it is necessary to select the desired symbol and press "Download".

Attention: The loaded data can differ from historical data stored on the broker's trade server.

Upon pressing the button, data of M1 timeframe will be loaded. Other timeframes will be automatically recalculated from M1. At that, the time of the downloaded data will be automatically recalculated according to the active account time zone. When downloading historical data, it is recommended to control amount of bars in history and in charts.

Attention: The deeper is the history used, the more PC resources are needed.

Quotes are weekly updated on the server of historical data. Further, at restarts, only updated quotes will be downloaded.

b) Exports and Imports of Historical Data

Historical data can be exported into files formatted as CSV, PRN and HTM. For this, it is necessary to select the desired symbol in the left part of the "History Center" window and press "Export". Then it is necessary to select one of three file formats and specify the path of location on the hard disk.

Historical data as CSV, PRN, TXT, HTM and HST can also be imported into terminal.

Historical data in the file can be represented as follows (any other separator can be used instead of space):

- YYYY.MM.DD HH:MM O H L C V
- YYYY-MM-DD HH:MM O H L C V
- YYYY/MM/DD HH:MM O H L C V
- DD.MM.YYYY HH:MM O H L C V
- DD-MM-YYYY HH:MM O H L C V
- DD/MM/YYYY HH:MM O H L C V

First of all, it is necessary to select a symbol and a timeframe, for which the import will be performed, in the left part of the “History Center” window. Then it is necessary to set up import parameters by pressing “Import”:

- **Separator** – data separator in the file to be imported. Comma, semicolon, space or tabulation character can be used as separators;
- **Skip columns** – skip columns when importing. This can be helpful when the imported file contains more data types than necessary;
- **Skip lines** – skip rows (lines) when importing;
- **Time shift** – shift data by several hours in time;
- **Selected only** – import only selected data. Data are selected by lines using “Ctrl” and “Shift”;
- **Volumes** – enable/disable importing of volumes.

After historical data have been imported, they can be used to show charts and test Expert Advisors.

c) Historical File Format (HST Files)

The database header is the first

```
struct HistoryHeader
{
int    version;           // database version
char   copyright[64];    // copyright info
char   symbol[12];       // symbol name
int    period;           // symbol timeframe
int    digits;           // the amount of digits after decimal point in the symbol
time_t timesign;         // timesign of the database creation
time_t last_sync;       // the last synchronization time
int    unused[13];      // to be used in future
};
then goes the bars array (single-byte justification)

#pragma pack(push,1)
//---- standard representation of the quote in the database struct RateInfo
{
time_t ctm; // current time in seconds double open;
double low;
double high;
double close;
double vol;
};
#pragma pack(pop)
```

J. EXPORT OF QUOTES

Source data serving as a basis for the entire analytical work of the terminal user are those about security price changes. This information is provided by the brokerage company. Price data allow to draw symbol charts, research in financial markets, use various trading tactics, and make trade decisions. Quotes represent files with records in format of “SYMBOL, BID, ASK, DATE”

(security symbol, bid price, ask price, date and time) and income in the terminal automatically as soon as connection to the server has been established.

The terminal allows to export the current quotes to other programs in the real-time mode through "DDE" (Dynamic Data Exchange) protocol. This is a protocol of operational systems of MS Windows used for dynamic data exchange among various applications. Quotes are given through DDE only at incoming of new ticks (ADVISE mode), but not immediately on request (REQUEST mode) where the latest price is shown. N/A is shown on the first REQUEST, and after the new price has been income, quotes will appear.

To activate the export of quotes from the terminal through DDE, one has to enable the "Enable DDE server" option in the terminal settings.

Attention: History Data cannot be exported through DDE protocol. The current quotes are exported only when the client terminal is online.

DDE request formats and their possible results by the example of "DDE-sample.xls" file:

BID request: =	MT4 BID!USDCHF	result: 1.5773
ASK request: =	MT4 ASK!USDCHF	result: 1.5778
HIGH request: =	MT4 HIGH!USDCHF	result: 1.5801
LOW request: =	MT4 LOW!USDCHF	result: 1.5741
TIME request: =	MT4 TIME!USDCHF	result: 21.05.02 9:52
QUOTE request: =	MT4 QUOTE!USDCHF	result: 21.05.02 9:52 1.5773 1.5778 1.5776

Attention: For data to be shown properly in MS Excel, one has to enable "Tools – Options... – Translation – Translation formula entry" menu option of MS Excel.

K. GLOBAL VARIABLES

Several experts can be launched in the client terminal at the same time. Sometimes, there is a need them to interchange with information. To provide possibility of prompt transfer of moderate amounts of information among experts, as well as organize conflict-free simultaneous working of several experts, there are global variables in the terminal. Unlike variables claimed at a global level in the expert source code and available only within the corresponding module, global variables exists independently on experts. Their values are saved between terminal launches, unlike those of variables claimed at a global level (they are set at every expert launch and lost at expert remove). Global variables are available within four weeks since their last call from experts or manual modifying.

There is a special window in terminal that manages global variables. It can be opened by execution of the "Tools – Global Variables" menu command or by pressing of F3. All global variables, their values and times of their last calls are listed in a table in this window. Using buttons located in the right part of the window, one can add a new global variable or delete an existing one. To change the name or value of a global variable, one has to double-click with the left mouse button on the corresponding cell of the table. The last call time will be changed automatically for this variable.

L. CONTRACT SPECIFICATION

This message window allows to view securities contract specifications. The main parameters are grouped in table with following fields:

- **Spread** – difference between Bid and Ask prices in points;
- **Digits** – the amount of digits after decimal point in the price representation;
- **Stops level** – minimum distance to the current price in points at which Stop Loss and Take Profit orders can be placed;
- **Pendings are good till cancel** – forced closing of pending orders at the end of a session. "Yes" means that pending orders will not be closed forcibly;
- **Contract size** – one-lot price in deposit currency;
- **Tick price** – the size of minimal price change in quote currency;
- **Tick size** – minimal symbol price change interval in points;
- **Profit calculation mode** – accepted profit calculation technique (Forex, CFD, Futures);

- **Swap type** – rollover calculation type (in points, in deposit currency, or in per cents);
- **Swap long** – rollover size for a long position;
- **Swap short** – rollover size for a short position;
- **Margin calculation mode** – accepted free margin calculation technique (Forex, CFD, Futures)
- **Margin hedge** – size of margin for hedged positions.

The symbol specification window can be called by pressing of “Properties” button in the Market Watch window or the “Symbol properties” of the “Tester – Settings” window.

M. LANGUAGES SUPPORT

The Client Terminal interface is multilingual. It can be represented in any available language. This feature makes the program not only easy-to-use, but also allows to reduce the amount of errors that occur during trading because of lack of understanding of foreign terms and notions. The list of all available languages can be found in the “View – Languages” menu. One can switch one’s terminal to a language by selecting of this language in the list. The language can be changed actually only after restart of the terminal.

N. DATACENTER.INI

Administrators of dealing centers who would like to redirect the clients’ traffic forcibly have a new opportunity now. The Data Center used by the client terminal can be strictly defined using the configuration file named datacenter.ini. This file must be placed in the /config directory of the client terminal.

If the “Data Center auto configuration” is flagged in the server settings, there will be an attempt to open and analyze the datacenter.ini file. If a data center for the server will be found in this file, the connection will be performed exactly to it. No more than one data center can be defined for one server.

The string format of the datacenter.ini:
[server name][delimiter][data center address]

where:

- **server name** is the name of the corresponding .srv file;
- **delimiter** is a comma or any quantity of spaces;
- **data center address** is the IP address : port number.

Lines that start with a semicolon (;) are considered to be a comment and are not processed.

Exemplary datacenter.ini file:

```
;-----  
;comment  
;-----  
MetaQuotes-demo1 192.168.1.4:443  
MetaQuotes-demo2, 192.168.1.4:444  
;-----  
MetaQuotes-demo3 192.168.1.5:445  
MetaQuotes-demo4, 192.168.1.5:446
```