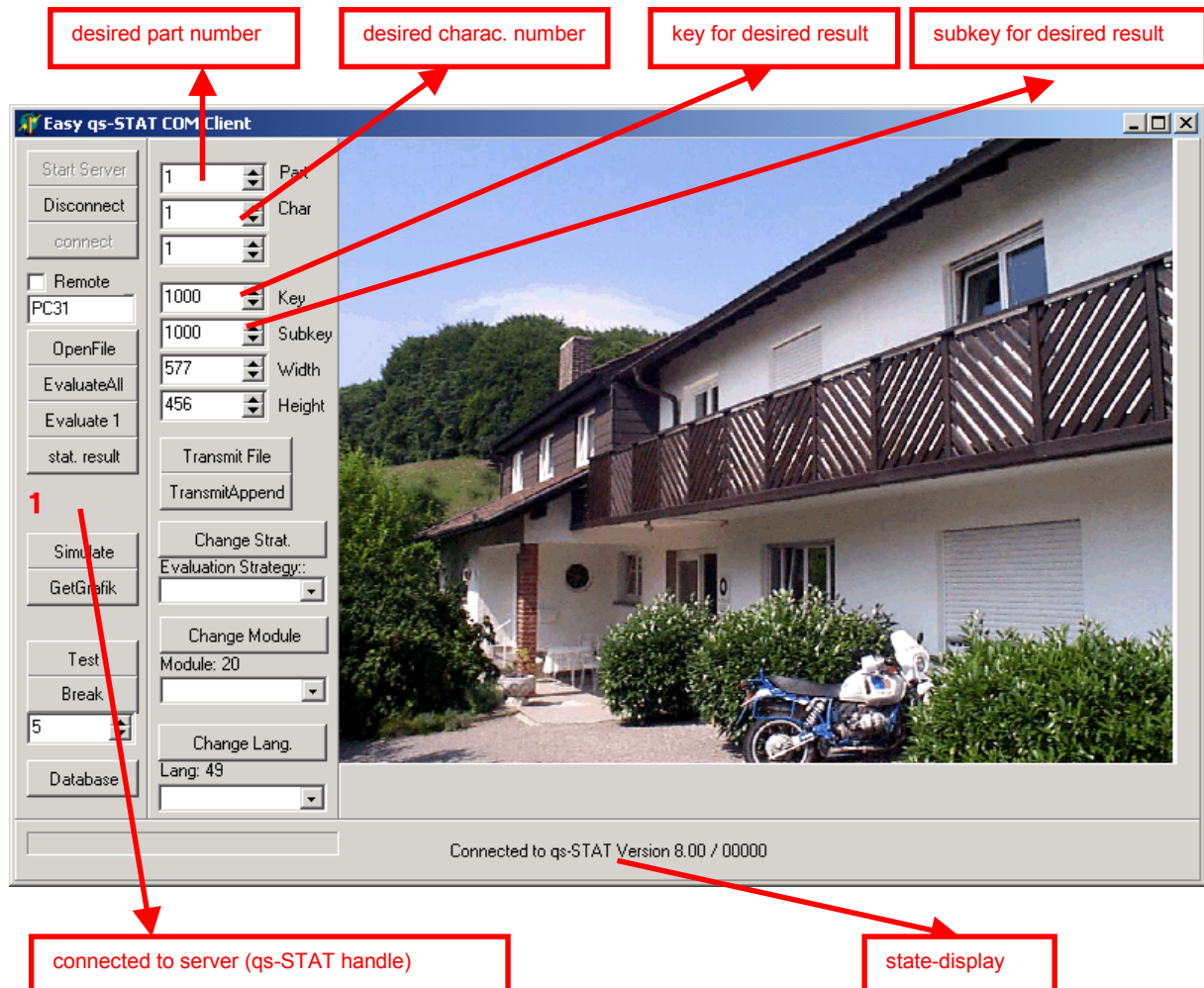


1 Delphi Client

1.1 General remarks

This sample shows, how to integrate the qs-STAT server into a Delphi source project. It was built for Borland Delphi 5.0. It shows the functionalities of the Offline server.



1.2 Usage

1. Press the „Start Server“ Button
The Client creates an instance of the COM object.
Wait until a messagebox „Server is initialized“ appears.
2. Press the „Connect“ Button
The Client connects itself to the server (a new Client instance is created inside the server).
Wait until the display changes from „Discon.“ to a numerical value.
3. Press the „Open File“ Button
The client forces the server to open a data set (in this case in form of a file based on the qs-STAT dataformat)
4. Press the „Evaluate All“ Button

The client forces the server to evaluate the loaded data set completely, according to the current qs-STAT module.

5. Get any results „Stat result“, „GetGrafik“

1. „Stat Result“:

The Client asks for a statistical result according to the value in the „Key“-Box for part x characteristic y, which are preseted in the boxes „Part“ and „Char“.

1000 average
 1100 median
 2000 variance
 etc.

2. „GetGrafik“

The Client asks for a grafic according to the value in the „Key“box for part x characteristic y, which are preseted in the boxes „Part“ and „Char“.

Grafics supported:

3100 value chart
 3200 value plot
 3300 Histogram
 etc.

1.3 Functionalities / used server methods

The following table illustrates, which server methods are called at the corresponding control on the client screen:

Button	Server methods
Start server	Creating an instance of the COM object
Remote	try to create a DCOM object on the PC named in the related box
Connect	ITQSstatRemoteControl.ClientConnect
Disconnect	ITQSstatRemoteControl.ClientDisconnect
OpenFile	ITQSstatRemoteControl.OpenFile
TransmitFile	ITQSstatRemoteControl.TransmitFileExt (reset the loaded data)
TransmitAppend	ITQSstatRemoteControl. TransmitFileExt (append values to previously loaded data)
EvaluateAll	ITQSstatRemoteControl.EvaluateAllChar
Evaluate 1	ITQSstatRemoteControl.EvaluateChar (evaluate char. defined in the “Part” and “Char” box)
stat. result	ITQSstatRemoteControl.GetStatResultExt (retrieve a numerical result defined in the “Part” , “Char”, Key, Subkey box)
GetGraphic	ITQSstatRemoteControl.GetStatResultExt (retrieve a graphic defined in the “Part” , “Char”, Key, Subkey box)
ChangeStrat.	ITQSstatRemoteControl.SetEvaluationStrategy (set the strategy to the that, which is choosen in the combo box below)
ChangeModule	ITQSstatRemoteControl.SetModule (set the module to the that, which is choosen in the combo box below)
ChangeLang.	ITQSstatRemoteControl.SetLanguage (set the language to the that, which is choosen in the combo box below)

Combobox "evaluation strategy"	IEnumEvaluationStrategy shows how to retrieve a list of available evaluation strategies
Combobox "Module"	IEnumModule shows how to retrieve a list of available qs-STAT modules
Combobox "Lang."	IEnumLanguage shows how to retrieve a list of available languages
Database	IEnumDatabase, IEnumCatalogues shows, how to create user defined queries to the qs-STAT database and how to retrieve catalogue informations from the qs-STAT server

1.4 Source Code

The sourcecode for the client application consist of the following files:

.source\comclient1\comclient1.dpr	project source
.source\comclient1\f_comclient.pas	source for the application window & interactions with the server
.source\comclient1\f_comclient.dfm	definition of the window face
.source\qsstat\qs_stat_tlb.pas	converted IDL file
.source\qsstat\qs_stat_tlb_safecall.pas	converted IDL file for connection points
.source\qsstat\qs_statevents.pas.pas	implementation fo connection points

2 C++ Client

2.1 General remarks

This sample shows, how to integrate the qs-STAT server into a C++ source project. It was built for MSVC 6.0 and uses the MSFC library. It shows the functionalities of both, the Offline server and the Online server. The methods for drawing the bitmaps onto the view are implemented in a very basic way and could be improved, but we didn't want to set the focus on drawing methods.

Due to the fact, that the BitBlt method seems to have some problems, when using specific settings for the color resolution (16-bit and 24-bit color depth), we recommend to use it either at 32 bit colors or 8 bit colors (256 color shows the graphics in a coarse resolution).

2.2 Usage

2.2.1 Offline Server

1. Press the „Start qs-STAT Server“ Menu item (“Server\ Start qs-STAT Server”)
 - The Client creates an instance of the COM object.
 - Wait until a messagebox „Server is initialized“ appears.
2. Press the „Connect to qs-STAT“ Menu item (“Server\Connect to qs-STAT”)
 - The Client connects itself to the server (a new Client instance is created inside the server).
3. Press the „Open File“ Menu item (File\Open file)
 - The client forces the server to open a data set (in this case in form of a file based on the qs-STAT dataformat). The data set will evaluated automatically after the file is opened by using the necessary COM server methods.
4. Get any results „Get stat. result“, „Get Graphic“ (Menu item “Commands\..”)
5. „Get stat. Result“ (Menu item “Commands\Get stat. Result”):
 - The Client asks for a statistical result according to the value. The part number, characteristic number and the key for the result are hard coded inside the source (part: 1, characteristic: 1 key: 1000 (average)). these may be changed (see Delphi client)
6. „Get Graphic“(Menu item “Commands\Get Graphic”):
 - The Client asks for a graphic according to specified values (part number, characteristic number, graphic number). The part number, characteristic number and the key for the result are hard coded inside the source (part: 1, characteristic: 1 key: 3300 (histogramm)). these may be changed (see Delphi client)

2.2.2 Online server

1. Start the server and establish a connection (see 2.2.1 Offline Server)
2. Open a base file to be visualized (Menu item “Online Transmission\Open base file”)
3. Start to monitor changes at the base file (Menu item “Online Transmission\Start file scan”). A value chart graphic for the data in the file will be displayed
4. Change the file manually (e.g. by using notepad) or let the client application to simulate values (Menu item “Online Transmission\Start DFQ Simulate” -> this adds cyclically values to the base file or Menu item “Online Transmission\Start value” ->

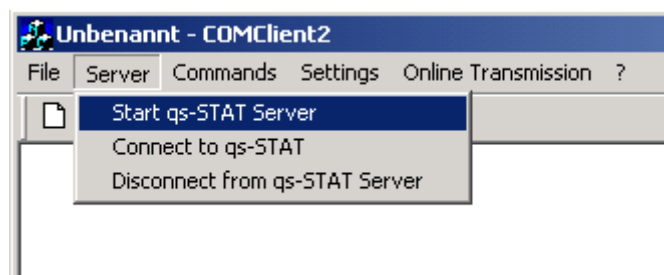
this adds one line to the base file). The selected graphic will be updated automatically, if the file was changed.

5. You can setup the client application to watch for occurred statistical alarm (Menu item "Online Transmission\Alarm settings"). Please note that you have to configure the alarm behaviour inside the qs-STAT user interface (process capability module, configuration of evaluation).
6. You can also display the previously occurred alarms in a list (Menu item "Online Transmission\Show Alarm").
7. You can do a process capability by pressing the Menu item "Online Transmission\process capability". This command will display a C-value plot after the data is evaluated.

2.3 Functionalities / used server methods

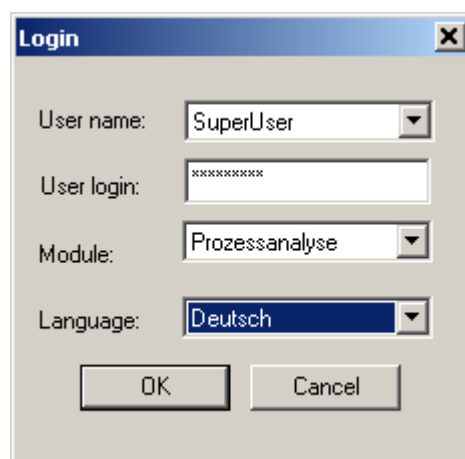
2.3.1 Connection to the server (“Server”)

This sample shows how to create an instance of the qs-STAT COM object inside a client application and how to establish a connection to the qs-STAT server.



Start qs-STAT server	Creates an instance of the qs-STAT server object in a separate thread
Connect to qs-STAT	usage of the CLIENTCONNECT method <ul style="list-style-type: none"> - connect to the qs-STAT server for using the process capability module and sets the default language - registering inside the server as “Superuser” - creates a data instance inside the server for the client - the client retrieves the need handle from the server
Disconnect from qs-STAT	usage of the CLIENTDISCONNECT method <ul style="list-style-type: none"> - disconnects form the server for the previously retrieved handle - releases the data instance for the retrieved handle

Entering a username and login according into qs-STAT user administration.
 Selecting if any shows one of Module and Language or type a Module number like “20” for the “process capability analysis” and Language number like “49” for “German”.

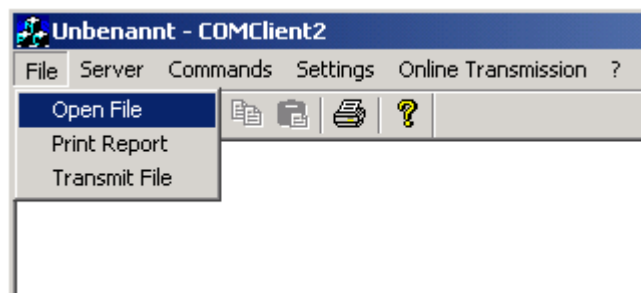


2.3.2 Loading data into the server and print reports (“File”)

This lets you load data into the server by 2 different ways:

- Open a file directly: the server loads the selected file conventionally (local COM server)
- Transmit file in a stream (local or remote COM Server)

In addition you can select one of the predefined reports and print them to a destination printer.



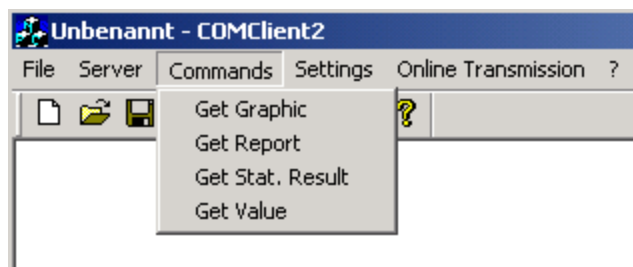
Menu OpenFile	usage of the OpenFile Method <ul style="list-style-type: none"> - forces the server to load the selected file - evaluates the loaded data
TransmitFile	usage of the TransmitFileExt – Method <ul style="list-style-type: none"> - Read a selected file to a buffer and transmits it to the server by using the TRANSMITFILEEXT method - evaluates the loaded data
PrintReport	usage of the PrintReportExt . Method <ul style="list-style-type: none"> - allows you to select a predefined report or the selected module and prints it on a selected printer

2.3.3 General commands

This sample shows you, how to retrieve results from the qs-STAT server. Please note, that you have to load data first, before you can retrieve any statistical result.

It allows you to :

- Get a desired graphic from the qs-STAT server,
- Get a predefined report,
- retrieve specific numerical results (->"output points"),
- Get information about the loaded data (-> "K-fields").



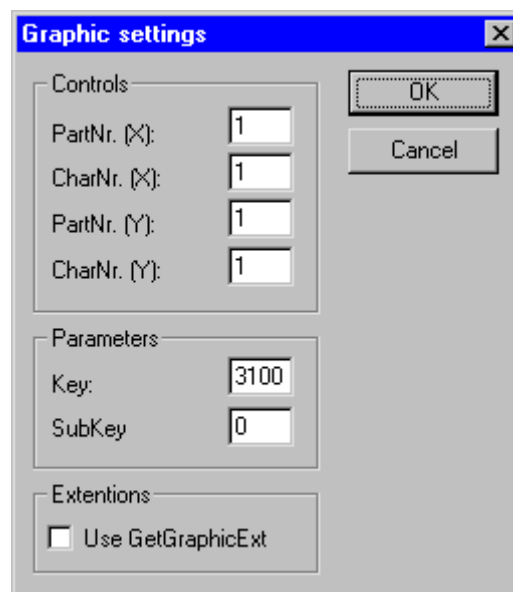
GetGraphic	usage of the GETGRAPHIC and GETGRAPHICEXT method
Get Report	usage of the GETREPORT method usage of the IENUMREPORT interface
Get stat. result	usage of the GETSTATRESULT and GETSTATRESULTTEXT method
Get value	usage of the GETVALUEINFO method

2.3.3.1 GetGraphic

After a click to this menu item, a dialog will be displayed, which allows you to enter informations about the desired graphic:

You have to decide:

- which graphic you want to show on the client's screen ("Key", "SubKey")
- which characteristics / values should be addressed ("PartNr (X)", "CharNr (X)",...) Note that specific graphics need a different parameter set, e.g. at X-Y plots you have to address both related Parts/characteristics X and Y
- you can decide, whether to use the GetGraphic- or The GetGraphicExt – method ("Use Ext.Function")

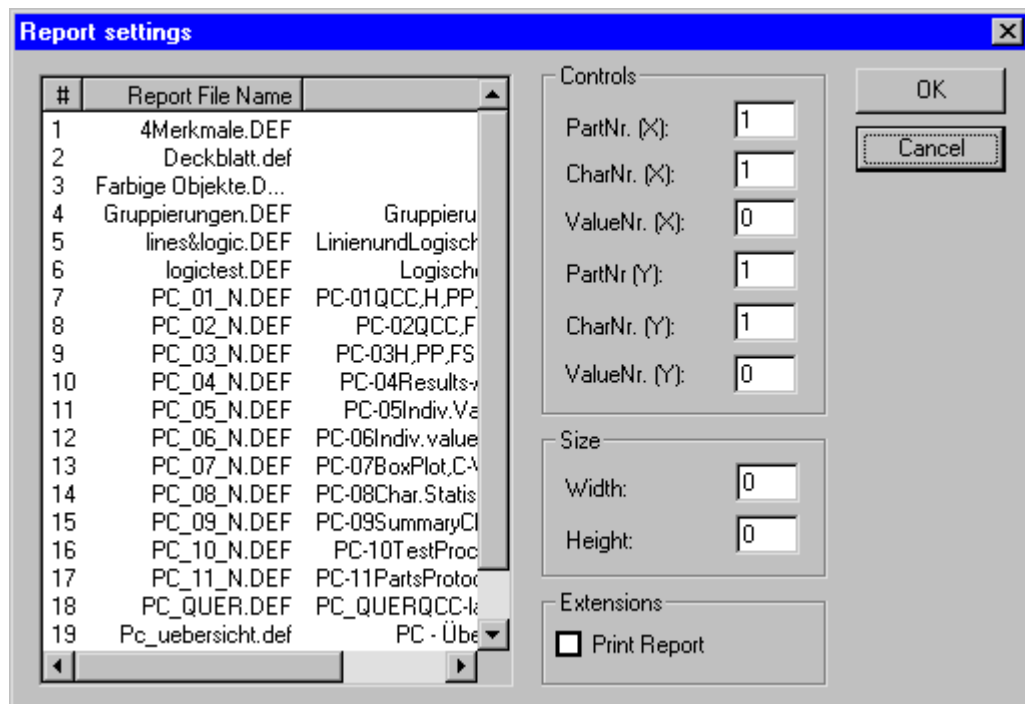


2.3.3.2 GetReport

After a click to this menu item, a dialog will be displayed, which allows you to enter informations about the desired report:

You have to decide:

- which report should be displayed on the clients screen (-> "Report file name")
- which part/characteristic shall be addressed (-> "Controls")
- the dimension of the report (-> "Size")



The dialog box titled "Report settings" contains the following elements:

- Report File Name:** A list box with 19 items, each with a number and a file name. The list is scrollable.

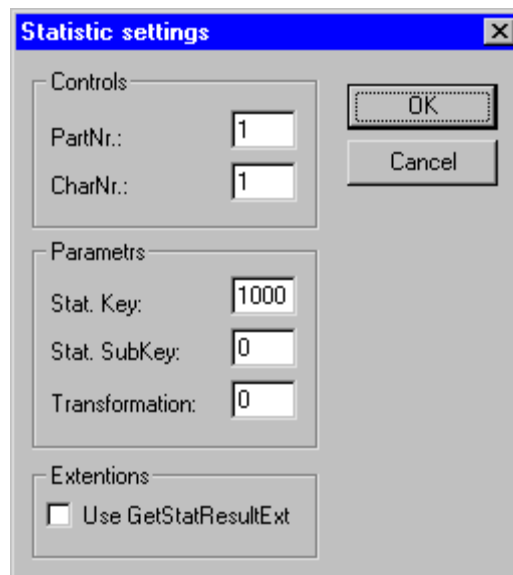
#	Report File Name
1	4Merkmale.DEF
2	Deckblatt.def
3	Farbige Objekte.D...
4	Gruppierungen.DEF
5	lines&logic.DEF
6	logictest.DEF
7	PC_01_N.DEF
8	PC_02_N.DEF
9	PC_03_N.DEF
10	PC_04_N.DEF
11	PC_05_N.DEF
12	PC_06_N.DEF
13	PC_07_N.DEF
14	PC_08_N.DEF
15	PC_09_N.DEF
16	PC_10_N.DEF
17	PC_11_N.DEF
18	PC_QUER.DEF
19	Pc_uebersicht.def
- Controls:** A group box containing six input fields:
 - PartNr. (X): 1
 - CharNr. (X): 1
 - ValueNr. (X): 0
 - PartNr. (Y): 1
 - CharNr. (Y): 1
 - ValueNr. (Y): 0
- Size:** A group box containing two input fields:
 - Width: 0
 - Height: 0
- Extensions:** A group box containing a checkbox labeled "Print Report", which is currently unchecked.
- Buttons:** "OK" and "Cancel" buttons are located on the right side of the dialog.

2.3.3.3 Get stat. result

After a click to this menu item, a dialog will be displayed, which allows you to enter informations about the desired numerical result:

You have to decide:

- which numerical result should be displayed on the clients screen (-> "Star Key", "Stat Sub Key")
- which part/characteristic should be addressed (-> "Controls")
- whether to use the GETSTATRESULT or the GETSTATRESULTTEXT method ("Use GetStatResultExt")



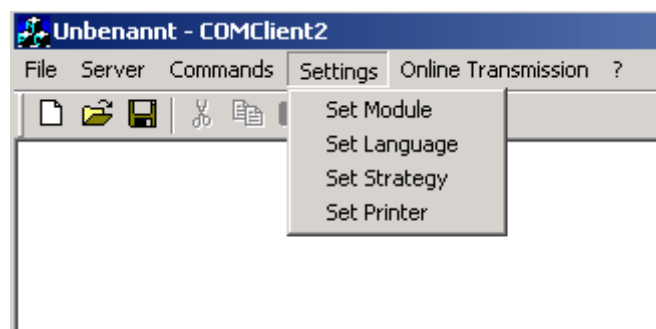
The dialog box titled "Statistic settings" contains three sections: "Controls", "Params", and "Extentions".

- Controls:** Contains two input fields: "PartNr." with value "1" and "CharNr." with value "1".
- Params:** Contains three input fields: "Stat. Key" with value "1000", "Stat. SubKey" with value "0", and "Transformation" with value "0".
- Extentions:** Contains a checkbox labeled "Use GetStatResultExt" which is currently unchecked.

On the right side of the dialog, there are "OK" and "Cancel" buttons.

2.3.4 Settings

This sample shows you, how to change some of the basic settings of the qs-STAT Server. After pressing one of the menu items, a list of the currently available settings will be displayed. These were retrieved by using the related IENUMxxxx interfaces and the selected list entry will be set by using the related Setxxxx method



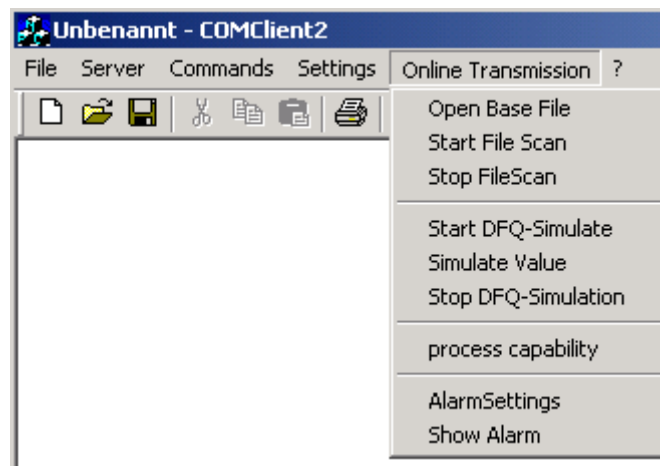
Set Module	usage of the SETMODULE method usage of the IENUMMODULE interface
Set Language	usage of the SETLANGUAGE method usage of the IENUMLANGUAGE interface
Set Strategy	usage of the SETSTRATEGY method usage of the IENUMEVALUATIONSTRATEGY interface
Set Printer	usage of the SETPRINTER method usage of the IENUMOPRINTERS interface

2.3.5 Online transmission

This sample shows you the usage of the qs-STAT Online and Online Analysis server.

It uses 3 different threads:

- Thread for scanning the basefile (created when “Start File Scan” is clicked)
- Thread for simulation of Data (created when Start DFQ-Simulate” is clicked)
- Thread for alarm detection (created when “AlarmSettings” is clicked)



A call to the “Process capability” menu item stops the running threads and does a call to the

```
ITQsstatRemoteControl.EvaluateAllChar - method
```

The data is evaluated according to the settings for the process capability module. After the statistical evaluation is finished (notified by the Event `IQSEvent.OnEvaluationDone` , a request for the C-Value plot is done, to show the calculated Cp- and Cpk values on the screen. This is done by a call to the

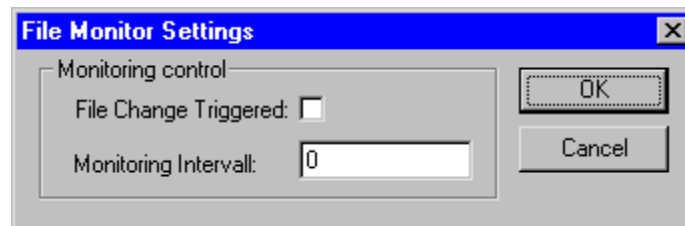
```
ITQsstatRemoteControl.GetGraphicExt - method
```

Finally, all previously occurred statistical alarm might be displayed, when clicking the “Show Alarm” menu item.

According to the names of the menu items, the threads have the following functionality:

2.3.5.1 Thread for scanning base file

This thread scans the previously opened file for file changes. File changes are event triggered or cyclic according to the settings done inside the dialog:



Each time, a file change was detected, the added data is loaded from the file into a buffer and is transmitted to the server by using the

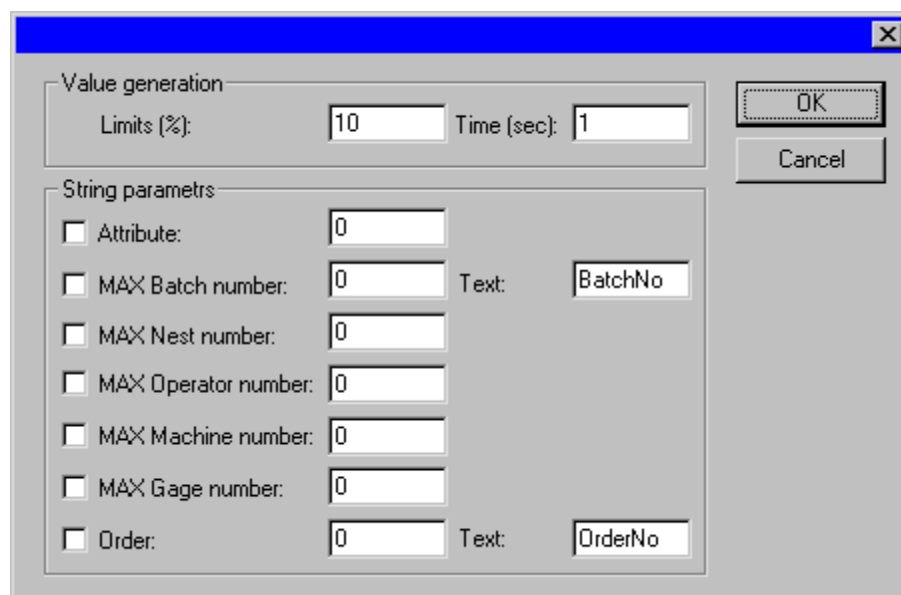
`ITQsstatRemoteControl.TransmitFileExt` - method

After data is transmitted (which is notified by the `IQSEvent.OnDataAvail` Event), the presetted graphic is recalled to obtain a “Online” visualization. This is done by using either the

`ITQsstatRemoteControl.GetGraphic` – method or
`ITQsstatRemoteControl.GetGraphicExt` - method

2.3.5.2 Thread for Simulation of data

This thread simulates values and appends the simulated data to the previously opened base file. The simulation is done accoring to the settings in the following dialog:



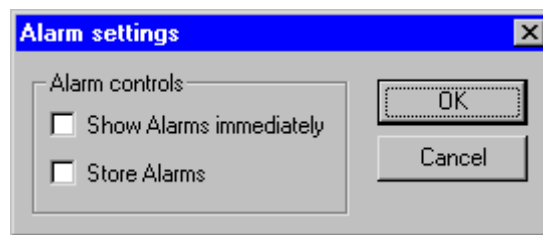
This dialog allows you to select, which how the data is simulated (range of the values in % of violation of the specification limits (-> “Value generation”, “Limits (%)”). This work is done cyclically. The cycle time is defined in the “Time (sec)” box in seconds. Additional data might be simulated as well, if the checkboxes are marked. The range for the random values for additional data can be defined inside the related boxes.

2.3.5.3 Thread for alarm detection

This thread waits for occuring alarms, which are notified by the

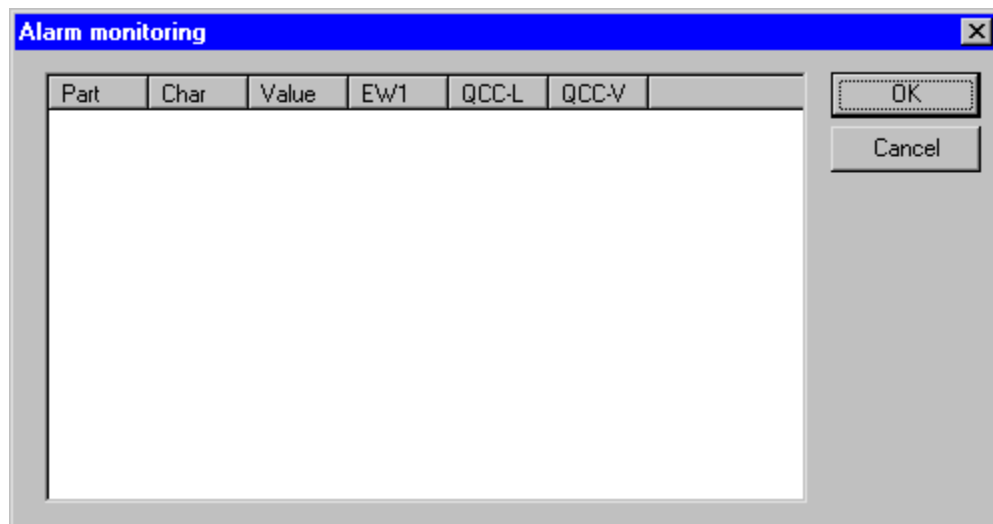
`IQSEvent.OnStatisticalAlarm – Event.`

As a result, the alarms maybe shown immediately or stored in a list according to the settings in the dialog:



2.3.5.4 Show occurred alarms


This dialog shows all occurred alarms, since the alarm dialog was shown the last time. It displays directly the bitcoded alarms, which are sent from the qs-STAT server. To get the meaning of the columns, see the documentation of the `IQSEvent.OnStatistical` interface.



2.4 Source Code

The sourcecode for the client application consists of the following files:

<code>.\source\comclient2\comclient2view.cpp</code>	source for the view & interactions with the server
<code>.\source\comclient2\qsEventSink.cpp</code>	Implementation of the connection points
<code>.\source\comclient2\Dib.cpp</code>	drawing of a bitmap (MSVC sample)
<code>.\source\comclient2\idl\qs_stat.idl</code>	IDL file of the server

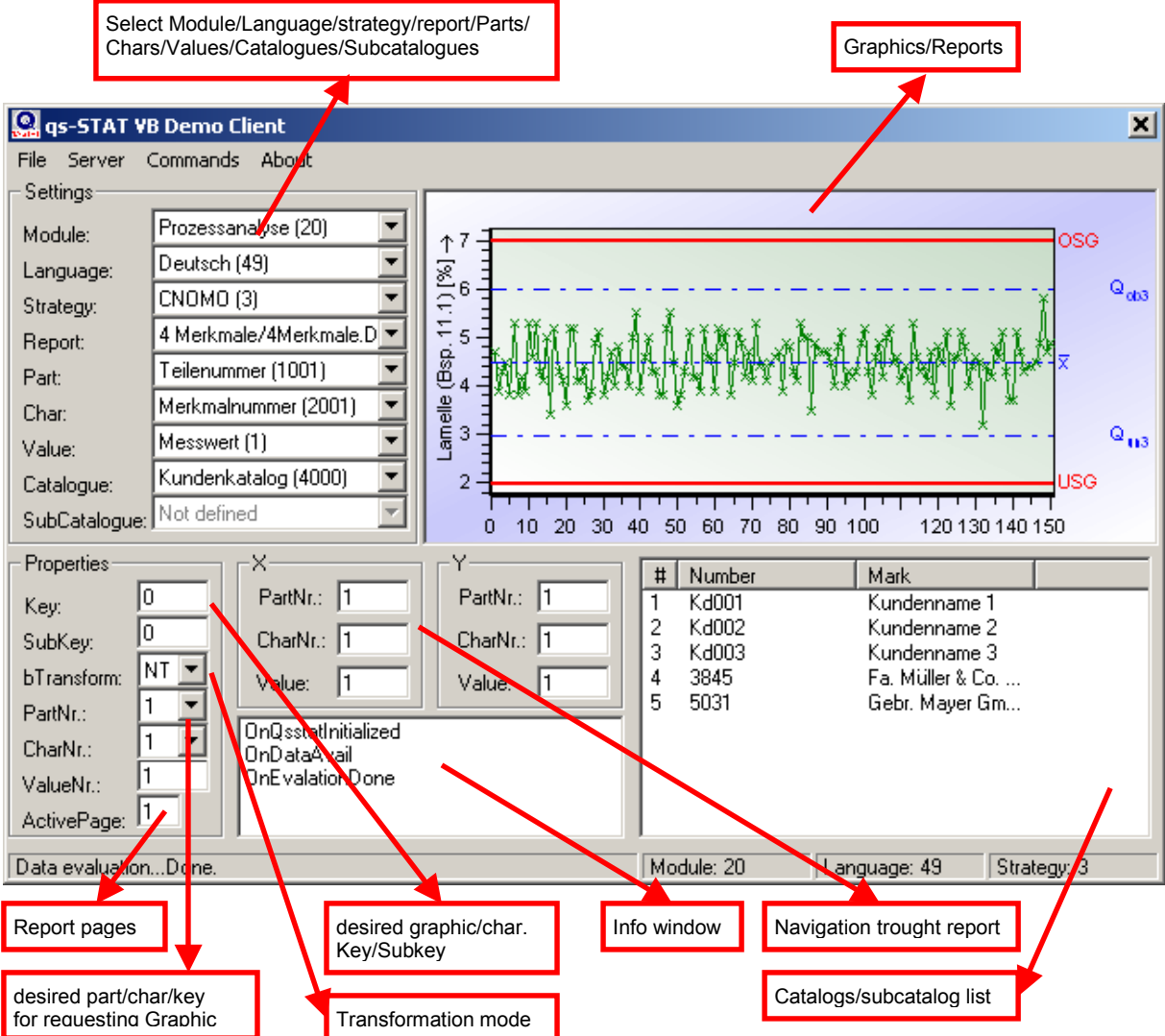
	<h1>DCOM Clients Samples</h1>	<p>Seite 16 / 24</p>
---	-----------------------------------	----------------------

Important note:

The concurrently processed commands of the server (open file, evaluate, etc.) are synchronised by events, which are setted, when the corresponding connection point is called by the server.

3 Visual basic client

This sample shows, how to integrate the qs-STAT server into a Visual Basic source project. It was built for Microsoft Visual Basic 6.0. It shows the functionalities of the Offline server.

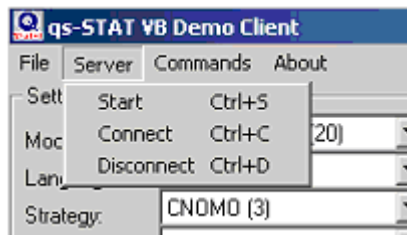


The screenshot shows the 'qs-STAT VB Demo Client' window. It features a menu bar (File, Server, Commands, About), a 'Settings' panel on the left with dropdowns for Module, Language, Strategy, Report, Part, Char, Value, Catalogue, and SubCatalogue, and a 'Properties' panel below it with input fields for Key, SubKey, bTransform, PartNr., CharNr., ValueNr., and ActivePage. A central graph displays 'Lamelle (Bsp. 11.1) [%]' on the y-axis (ranging from 2 to 7) against an x-axis from 0 to 150. The graph shows a fluctuating green line with 'x' markers, bounded by a red box labeled 'OSG' at the top and 'USG' at the bottom. A blue dashed line represents the mean, and blue dots represent standard deviation. To the right of the graph is a table with columns '#', 'Number', and 'Mark'. Below the table are status indicators for 'OnQsstatInitialized', 'OnDataAvail', and 'OnEvaluationDone'. At the bottom, a status bar shows 'Data evaluation...Done.' and 'Module: 20 Language: 49 Strategy: 3'. Red arrows point from various parts of the interface to labels: 'Select Module/Language/strategy/report/Parts/Chars/Values/Catalogues/Subcatalogues' points to the Settings panel; 'Graphics/Reports' points to the graph; 'Report pages' points to the 'bTransform' dropdown; 'desired part/char/key for requesting Graphic' points to the 'PartNr.', 'CharNr.', and 'ValueNr.' fields; 'desired graphic/char. Key/Subkey' points to the 'Key' and 'SubKey' fields; 'Transformation mode' points to the 'bTransform' dropdown; 'Info window' points to the status indicators; 'Navigation through report' points to the table; and 'Catalogs/subcatalog list' points to the 'Catalogue' and 'SubCatalogue' dropdowns.

The Info window on the bottom side displays the incoming events which are sent by the IQSEvent interface.

3.1 Usage

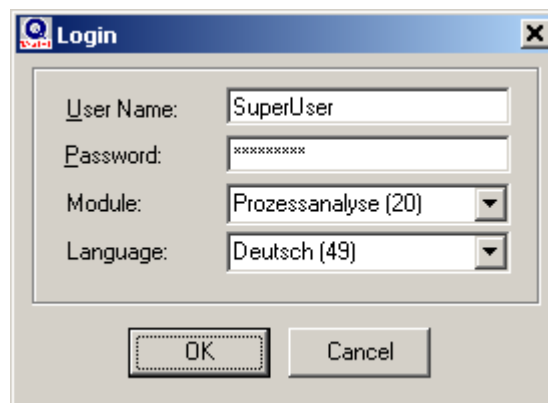
1. Select the „Start“ menu from Menu „Server“.
It shows a message „Starting qs-STAT Server...“
Wait until a message „OnQsstatIntialized“ appears at Info window



The Client creates an instance of the COM object.

Wait until shows the Message „Starting qs-STAT Server...Done. Login to Server...“

2. Enter your Username and login according into qs-STAT user administration.
Select if any shows one of Module and Language or type a Module number like „20“ for the „process capability analysis“ and Language number like „49“ for „German“.

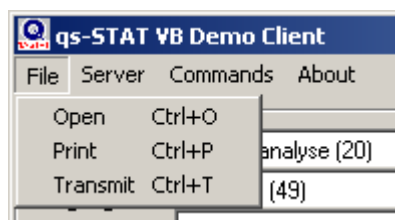


Press the „OK“ button

The Client connects itself to the qs-STAT server (a new Client instance is created inside the server used the selected user/login and module/language).

If the login to the qs-STAT server was successful, then shows message „Starting qs-STAT Server...Done. Login to Server...Done. Version XXX“

3. Select the „Open“ menu from Menu „File“ and browse a file based on the qs-STAT data format
It shows message „Loading in progress...“



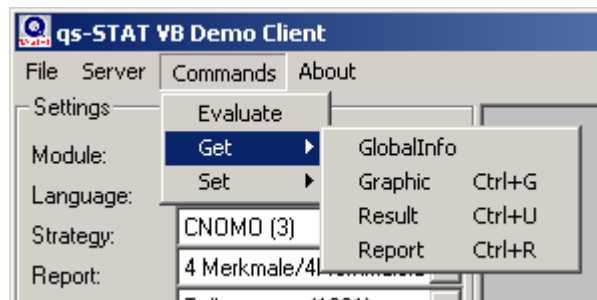
The client forces the server to open a data set

Wait until shows the Message „Loading in progress...Done. Evaluation in progress...”

The client forces the qs-STAT server to evaluate the loaded data set completely, according to the current qs-STAT module and predefined evaluation strategy.

Wait until shows the Message „Loading in progress...Done. Evaluation in progress...Done”
 After that the Client requests the global info about the loaded and evaluated data set in the boxes „Part“ and „Char“.

4. Select the „Result” menu from Menu “Commands->Get” to show the Statistical results



The Client asks for a statistical result according to the value in the „Graphic/Stat. results“-Box for part x characteristic x, which are preset in the boxes „Part“ and „Char“.

1001 average
 1101 median
 2001 variance
 etc.

5. „GetGrafik“

The Client asks for a graphic according to the value in the „Graphic/Stat. results“-Box for part x characteristic x, which are preset in the boxes „Part“ and „Char“.

Graphics supported:

3101 value chart
 3201 value plot
 3301 Histogram
 etc.

6. „GetPrinter“

The Client asks for all available Printer on your System and shows you a printer dialog

7. „GetReport“

Select the „Report“ menu from the menu “Commands->Get”

The Client asks for a report file from Combo: “Report” and shows a selected report.

3.2 Functionalities / used server methods

The following table illustrates, which server methods are called at the corresponding control on the client screen:

Button	Server methods
Start server	Creating an instance of the COM object
Connect	ITQSstatRemoteControl.ClientConnect
Disconnect	ITQSstatRemoteControl.ClientDisconnect
OpenFile	ITQSstatRemoteControl.OpenFileExtTM
TransmitFile	ITQSstatRemoteControl.TransmitFileExtTM
EvaluateAll	ITQSstatRemoteControl.EvaluateAllCharsTM
Stat. results	ITQSstatRemoteControl.GetStatResultExt
GetReport	ITQSstatRemoteControl.GetReportExt
GetReportPages	ITQSstatRemoteControl.GetReportPagesVA
GetGraphic	ITQSstatRemoteControl.GetStatResultExt
ChangeModule	ITQSstatRemoteControl.SetModule
ChangeLanguage	ITQSstatRemoteControl.SetLanguage
Combobox "Module"	IEnumModule shows how to retrieve a list of available qs-STAT modules
Combobox "Lang."	IEnumLanguage shows how to retrieve a list of available qs-STAT languages
Combobox "Strategy"	IEnumStrategy shows how to retrieve a list of available qs-STAT strategies
Combobox "Report"	IEnumReport shows how to retrieve a list of available qs-STAT reports
Combobox "Part"	IEnumGlobalInfo shows how to retrieve a list of available qs-STAT parts (after data loaded)
Combobox "Chars"	IEnumGlobalInfo shows how to retrieve a list of available qs-STAT chars (after data loaded)
Combobox "Catalogue"	IEnumCatalogue shows how to retrieve a list of available qs-STAT catalogues
Combobox "Subcatalogue"	IEnumSubcatalogue shows how to retrieve a list of available qs-STAT subcatalogues (for catalog certain)

3.3 Source Code

The source code for the client application consist of the following files:

\qsSTATDemoclient.vbp	project source
\ClientForm.frx	Main form
\frmAbout.frx	About form
\frmLogin.frx	Login form
\ClientForm.frm	Main form properties
\frmAbout.frm	About form properties
\frmLogin.frm	
\qsSTATlogo.emf, qsstat.ico	additional files

4 HTML (VBScript based) Client

4.1 General remarks

This HTML COM Client shows, how to integrate the qs-STAT server into a HTML-File based on the Visual Basic Script engine.

Features:

- Start qs-STAT Servers object
- connect and disconnect to the qs-STAT Server
- Authorization on the qs-STAT Server via User name and login
- Select Module and Language for login
- Open and Transmit Files (qs-STAT based format) on the qs-STAT Server
- Show Graphics, Statistics and predefined Reports



The screenshot shows the VBScript COM Client interface within a Microsoft Internet Explorer window. The interface is divided into several sections, each with specific controls and a corresponding annotation box:

- Server:** Contains a "START qs-STAT Server" button, a "user:" field (set to "superuser"), a "login:" field (masked with asterisks), and "Connect" and "Disconnect" buttons. Annotations point to "Start qs-STAT Server", "desired user/login", and "Connect/disconnect qs-STAT Server".
- Settings:** Includes a "module:" dropdown (set to "Process Capability Analysis (20)") and a "language:" dropdown (set to "German (49)"). An annotation points to "Module/Language".
- File:** Features a "File Open" button and a "File transmit" button. An annotation points to "File open/transmit".
- Graphic/Statistical results:** Contains fields for "key nr.", "subkey nr.", "btransform nr.", and "transmit mode". It also has "Graphic" and "as File" buttons. Annotations point to "desired part/char/key", "desired graphic/char. Key/Subkey", "Transformation mode", "Show Graphic", and "Transmit mode".
- Report:** Includes fields for "part X nr.", "part Y nr.", "char X nr.", "char Y nr.", "value X nr.", and "value Y nr.". It also has a "Report" button and a "as File" button. Annotations point to "Navigation trough report", "Requesting report", and "Report pages".
- Image:** Contains fields for "width:" (set to 640) and "height:" (set to 480), a "Format:" dropdown (set to BMP), and a "Stat results" button. Annotations point to "Graphic dimensions", "Statistical results", and "Graphic Output format".

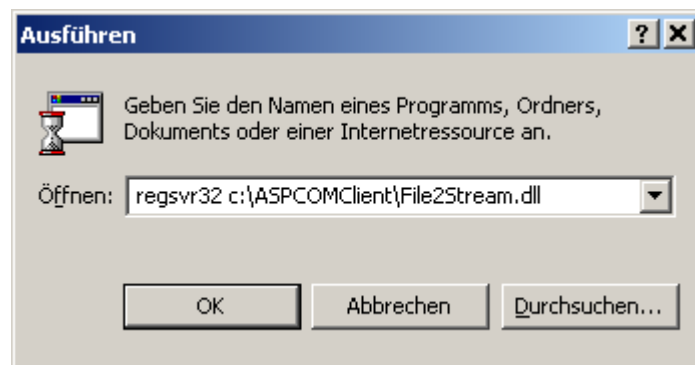
The main display area shows a scatter plot with green data points and a blue trend line. The y-axis is labeled "Durchmesser ob. Durch. (Bsp. 12) [mm]" and the x-axis is labeled "Wert Nr. →". The plot is bounded by a green area and a blue area, with labels "OSG" and "USG" on the right side.

4.2 Requirements

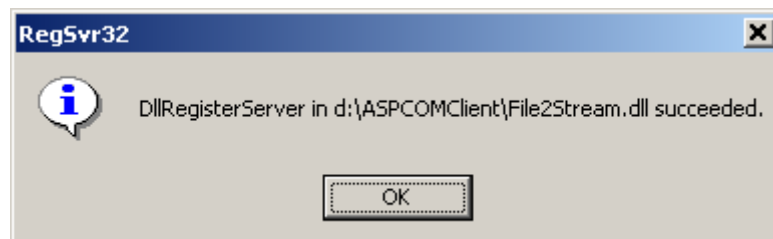
This example was tested at the Microsoft Internet Explorer Version 5.

To Start the HTML COM Client one additional component - **<File2Stream.dll>**. are needed. This component is necessary for the conversion of streams to files forward and backwards and must be registered on the system by using the tool REGSVR32.EXE. You can find this component in the qs-STAT installation directory, where the HML file is stored.

For example (C:\ASPCOMCLIENT\File2Stream.dll is the path where the component is located):

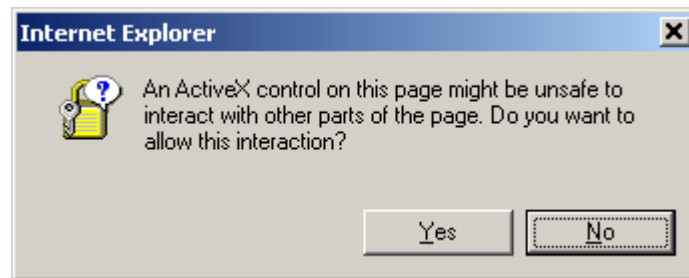


If the registration was successful, the following message is shown:



4.3 Usage

1. Press the „Start qs-STAT Server“ Button.
Wait until a messagebox „ActiveX interaction control“ appears. If this message is shown, depends on the security settings of your browser



Press the „Yes“ button.

The Client creates an instance of the COM object.

Wait until shows the Message „*qs-STAT Server started*“ at status line

2. Enter your Username and login according into qs-STAT user administration

3. Select a Module and Language

4. Press the „Connect“ Button.

It shows message „*Establishing connection to qs-STAT Server...*“ at status line

The Client connects itself to the qs-STAT server (a new Client instance is created inside the server).

Wait until displays the message „*Establishing connection to qs-STAT Server...Done.*“

5. Press the „Browse“ Button and select a file based on the qs-STAT data format

6. Press the „Open“ Button

It shows message „*Load file: <FileName> ...*“ at status line

The client forces the server to open a data set

Wait until shows the Message „*Load file: <FileName> ...Done. Evaluation in progress...*“

The client forces the qs-STAT server to evaluate the loaded data set completely, according to the current qs-STAT module.

Wait until shows the Message „*Load file: <FileName> ...Done. Evaluation in progress...Done*“

After that the Client requests global info about the loaded data and set the boxes „Part“ and „Char“.

7. „Stat Result“:

The Client asks for a statistical result according to the value in the „Graphic/Stat. results“-Box for part x characteristic x, which are selected in the boxes „Part“ and „Char“.

1002 average

1102 median

2002 variance

etc.

8. „GetGrafik“

The Client asks for a graphic according to the value in the „Graphic/Stat. results“-Box for part x characteristic x, which are selected in the boxes „Part“ and „Char“.

Graphics supported:

3102 value chart

3202 value plot

3302 Histogram

etc.

4.4 Functionalities / used server methods

The following table illustrates, which server methods are called at the corresponding control on the client screen:

Button	Server methods
Start server	Creating an instance of the COM object
Connect	ITQSstatRemoteControl.ClientConnect
Disconnect	ITQSstatRemoteControl.ClientDisconnect
OpenFile	ITQSstatRemoteControl.OpenFileExtTM
TransmitFile	ITQSstatRemoteControl.TransmitFileExtTM (reset the loaded data – mode 2)
TransmitAppend	ITQSstatRemoteControl.TransmitFileExtTM (append values to previously loaded data – mode 6)
EvaluateAll	ITQSstatRemoteControl.EvaluateAllCharsTM
Stat. results	ITQSstatRemoteControl.GetStatResultExt
GetReport	ITQSstatRemoteControl.GetREportExt
GetReport as file	ITQSstatRemoteControl.GetReportExtAsFile
GetReportPages	ITQSstatRemoteControl.GetReportPagesVA
GetGraphic	ITQSstatRemoteControl.GetStatResultExt
ChangeModule	ITQSstatRemoteControl.SetModule
ChangeLanguage	ITQSstatRemoteControl.SetLanguage
Combobox “Module”	IEnumModule shows how to retrieve a list of available qs-STAT modules
Combobox “Lang.”	IEnumLanguage shows how to retrieve a list of available qs-STAT languages

Remarks:

- This Demo doesn't use the concurrent processing of commands. All methods are serialized by using the ITQSstatRemoteControl.XXXXXTM –methods, setting the parameter ThreadMode to 1, that indicates to the server to process commands sequentially. In this case a client application needn't wait for the server's synchronisation events. (-> openFilexxxTM, TransmitFileExtTM, EvaluateAllCharsxxxTM)
- The Demo also uses an additional helper component, which generates files from OleVariant arrays forwards and backwards to load the graphics / reports and to transmit files to the server in form of a stream.

4.5 Source Code

The source code for the client application consist of the following files:

.\source\htmlcomclient.htm project source