FastReport® Server

Users' manual

Contents

Introduction

This users' manual contains information about FastReport Server application. This Server allows building reports on client-server technology with the help of standard FastReport 3 engine and additional components.

FastReport Server is a highly productive report generator with unique capabilities.

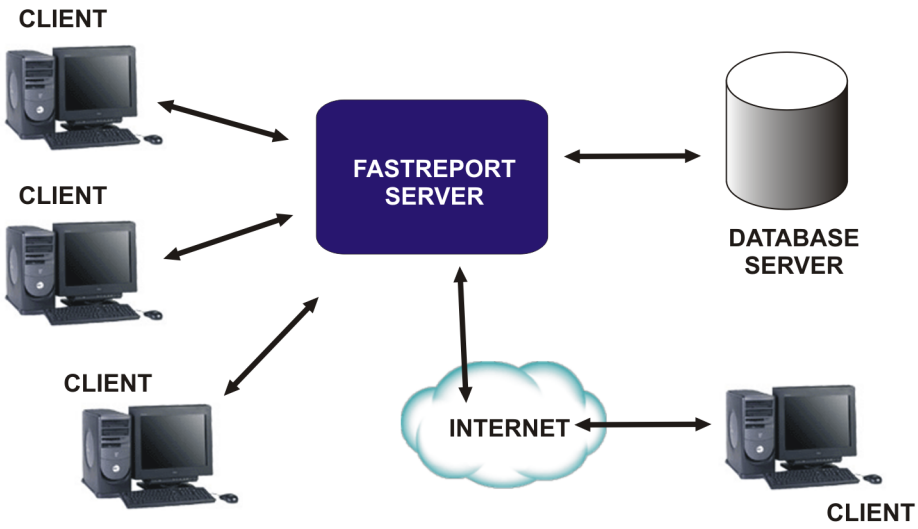
This guide describes the architecture of a report server and the principles of its functioning. Furthermore, it gives recommendations concerning optimization and usage of new capabilities in already existing applications and those developed anew.

The experienced FastReport users will be interested in recommendations about increasing server components' speed, optimization of reports for their correct export to various tabular formats, application of rules of information safety for application protection from non-authorized access.

We have been constantly improving the FastReport Server. That is why there is a probability that some capabilities are not mentioned in this manual. Descriptions of all changes will be necessarily included to the next version of this manual.

FastReport Server

The "client-server" technology is based on interaction between a client application (which inquires, analyzes, and displays requested information) and a server application that performs basic work related to various complex calculations.



There are several serious advantages of using client-server technology in your applications:

- low hardware requirements for client PCs;
- reducing of network traffic due to reducing the amount of information transferred between a client's application and a database server;
- simplicity of system management of the existing client-server;
- higher level of information protection.

However, the client-server technology has some considerable disadvantages:

- high hardware requirements for a PC used as a server;

- certain difficulties in development of client-server applications.

When developing FastReport Server, we take into account all major requirements for client-server applications. FastReport Server allows you to:

- run any reports on the server side on client request, without necessity to directly connect the client to the database server;

- manage several client requests simultaneously in separate threads; it minimizes response time of the server;

- since we use Hyper Text transfer protocol (HTTP, RFC 2068 [2]), you can use different existing applications, such as web-browsers (Internet Explorer, Netscape Navigator, Mozilla, Opera etc), proxy-servers, web-servers (Internet Information Server, Apache etc), together with FastReport Server without any additional requirements;

- use data compression algorithms (GZip, RFC 1952 [6]). This reduces network traffic and increases client-server processing power;

- use of MD5 algorithm for the MIC (Message Integrity Checksum, RFC 1321 [4], RFC 1864 [5]) increases data integrity;

- compatibility with FastReport 3 report files (with some restrictions) allows you to easily redesign your application to use client-server technology;

- standalone server application (without necessity to apply IIS, Apache or other web-server technologies) has a high processing power, short response time, and economical use of system resources (in comparison with solutions based on CGI technology);

- you can use the server as a simple HTTP server for storing and displaying any HTML documents;

- application of the Server Side Include (SSI) technology allows you to use the server as an engine for your web-site;

- managing the connection logs, error logs, and/or any additional system information allows you to keep record of the work, quickly track down the bugs and unauthorized access attempts;

- usage of authentications and “allow/deny” IP lists allows you to restrict access to the server;

- you can use several database connections in one report simultaneously;

- you can use FastReport client components for interaction between a client application and the server. You can use any web-browser as well;

- your reports may have a dialogue forms that will be used for entering some values before running a report;

- supported formats of the prepared reports are: HTML, PDF, RTF, XML, XLS, JPEG, Text etc;

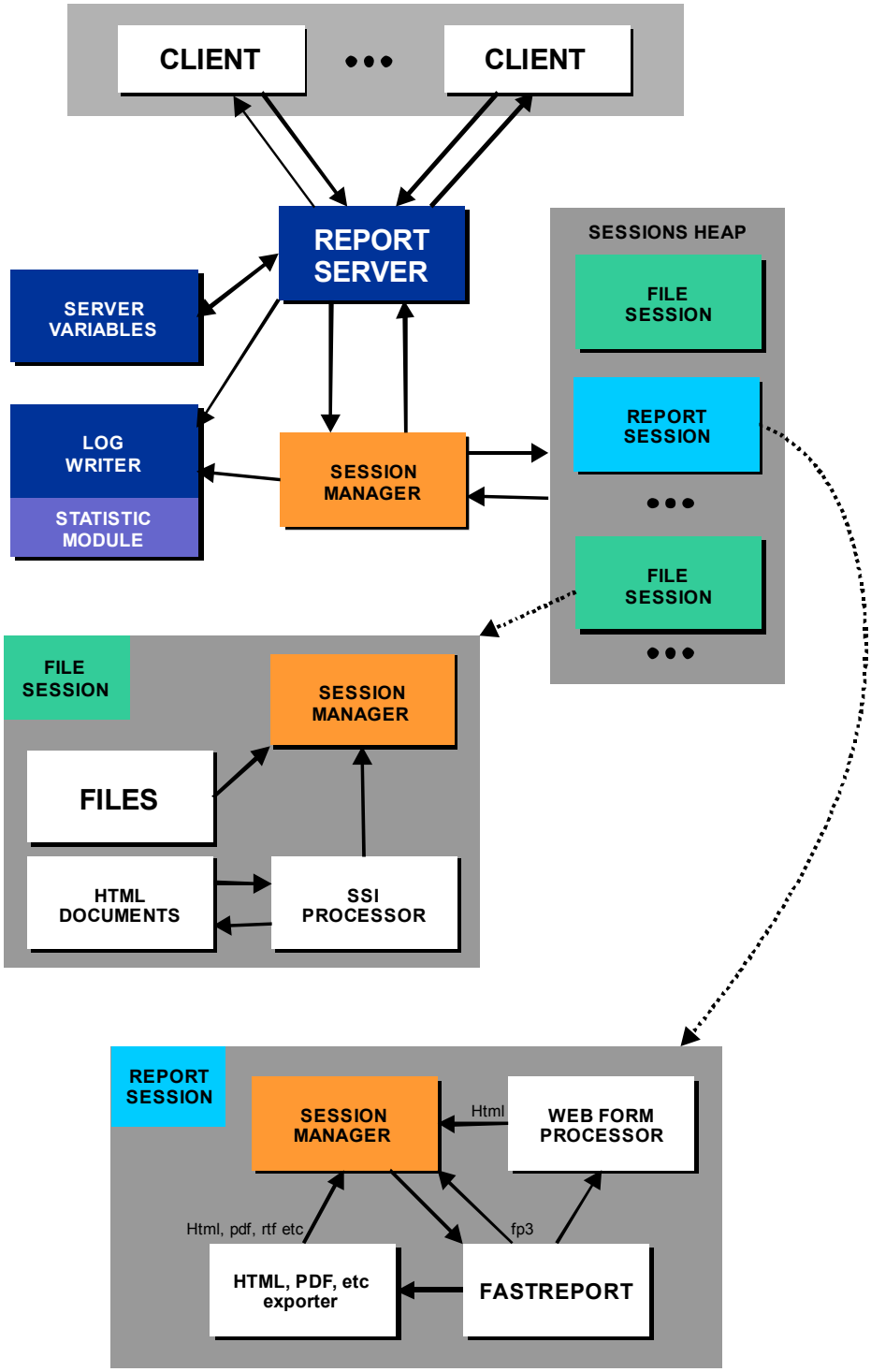
- you can use several modes of displaying the prepared report in your web-browser: single-page document, page-separated with page navigator;

- you can use FastReport Server ISAPI extension together with Microsoft IIS, Apache Web Server, and etc.

Internal architecture

Server is represented as an autonomous HTTP server with a capability of report generating. The Report server is able to transact several reports simultaneously, logging any events, and collecting the statistics.

The scheme displays the server's internal structure:



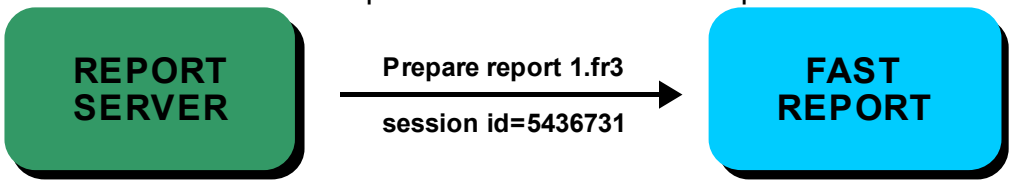
The sessions with unique identifier are created when a request from client comes. The line of the request is analyzed. If the requested file exists, then the server sends a positive response with the file to the client. Logs are updated with new record about this event. If the request contains the report query, then a special report session is created.

After the report is built, the result is saved to folder with session number as a name. The server responses to the client, and reports a new file location. The client sends a new request to the new file location, and receives the file with the result. Session with the resulting file is stored by server until session time expires.

Below is a step-by-step graphical overview of the report query transaction with the web browser:

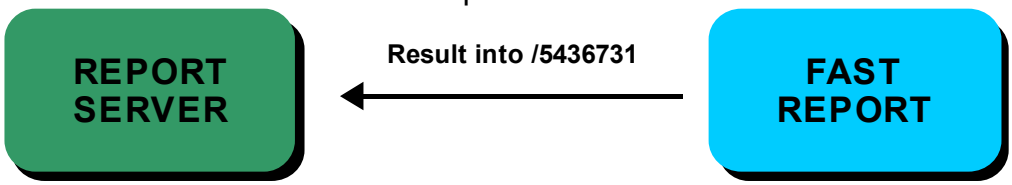
- client sends query; the report's title is "1.fr3"

- the server creates a new FastReport instance and delivers parameters of the request

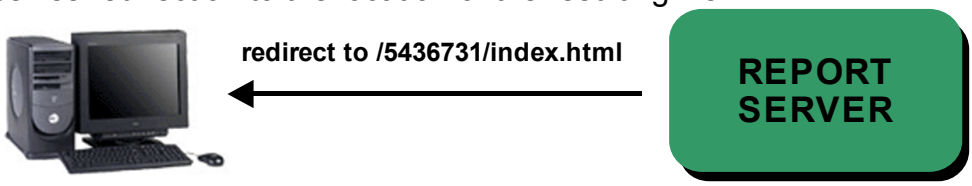


- FastReport prepares the report and exports results to a html file into the folder, the name of the folder is the same as the session's number

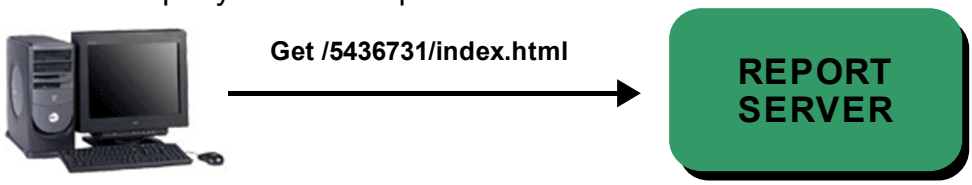
- server waits for the results from FastReport



- client receives redirection to the location of the resulting file



- client sends a new query with the request of results file

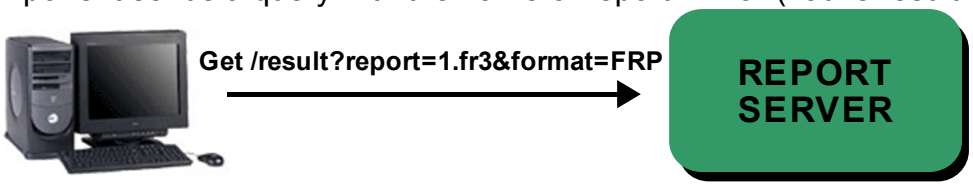


- server delivers the resulting file to the client

Step-by-step graphical overview of the report query transaction with the FastReport (TfrxReportClient):

- a client wants to show report "1.fr3":

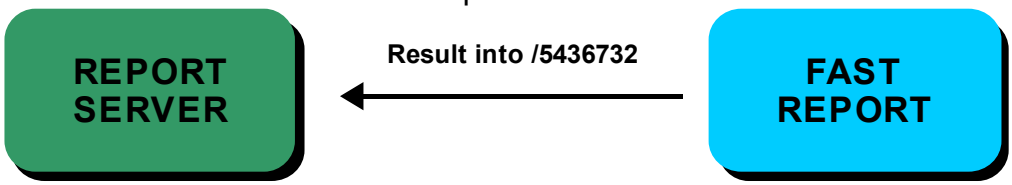
- client component sends a query with the name of report "1.fr3" (native result format)



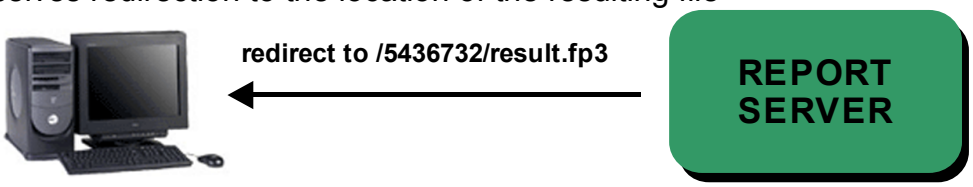
...

- FastReport prepares the report and saves the results to a native fp3; the name of the folder is the same as the session's number

- server waits for the results from FastReport

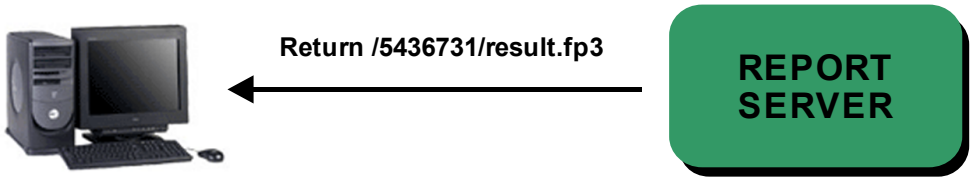


- client receives redirection to the location of the resulting file

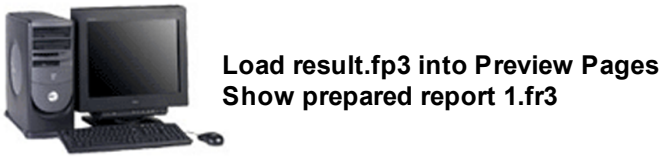


...

- server sends the result file to the client

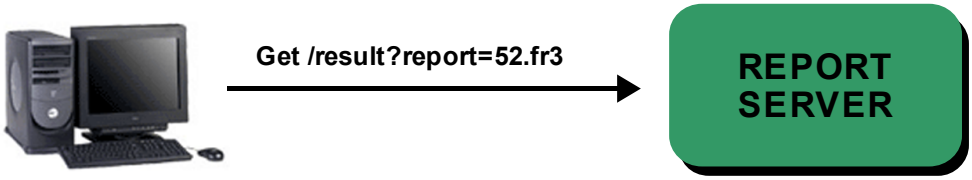


- client displays the report

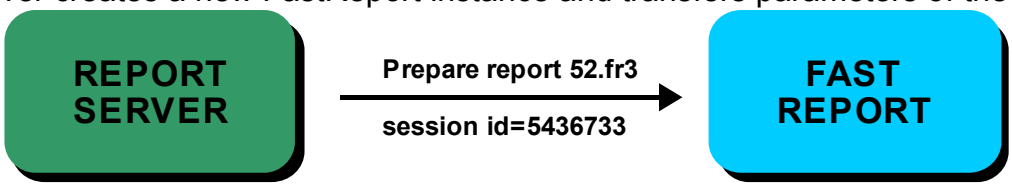


If the inquired report contains any forms, the process becomes more complicated:

- client client component sends a query with the name of report "1.fr3"

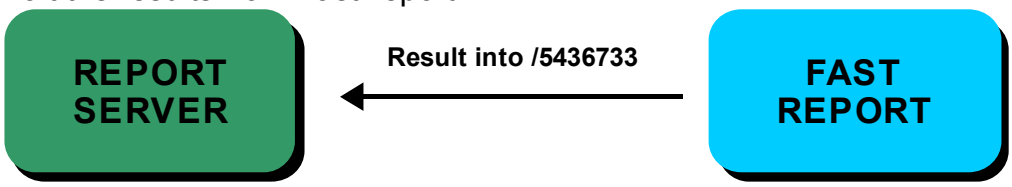


- the server creates a new FastReport instance and transfers parameters of the request



- FastReport prepares report and saves the web-form into the folder named according to the session number

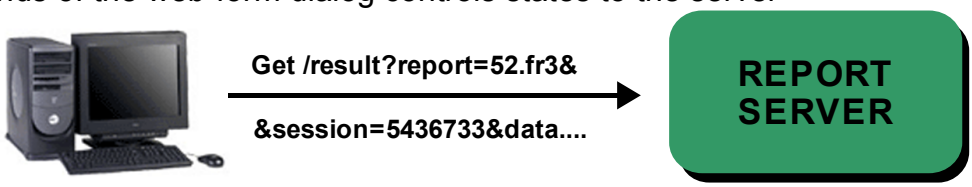
- server wait the results from FastReport



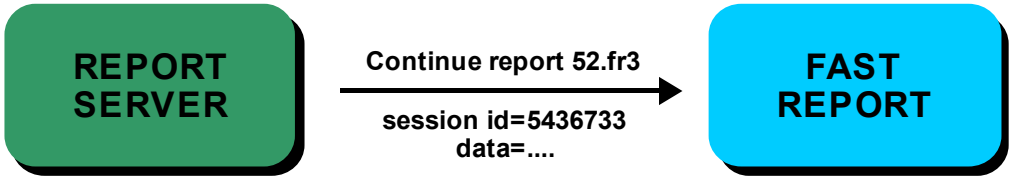
- server redirects the client to the web-form file

- client receives the web-form, while FastReport waits

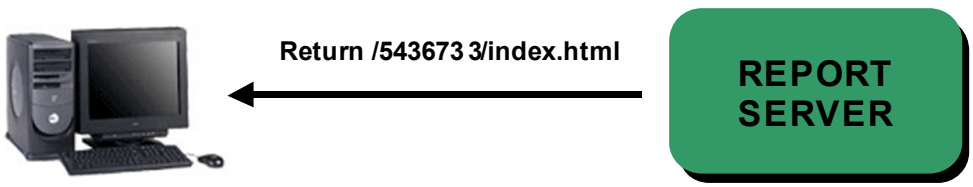
- client sends of the web-form dialog controls states to the server



- the server transfers the values of the control elements to the server



- server delivers the received information to FastReport

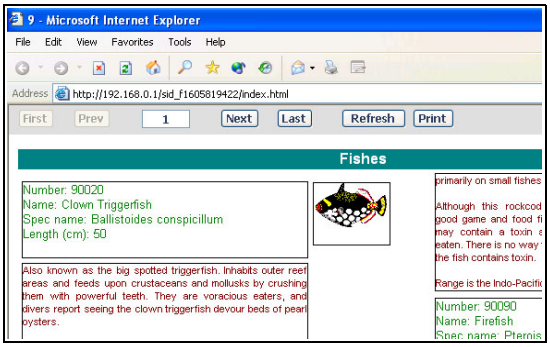


Format of the server request line, logging, authentication and other issues concerning server's functioning, are described below.

Supported formats of the report results

FP3 is a native FastReport 3 format. It is represented as a XML document. FP3 is used during transaction between Server and FastReport client. This format is the most appropriate one for document printing. In most cases, use of this format reduces both transaction time and size of the transferred files (except reports containing high quality images).

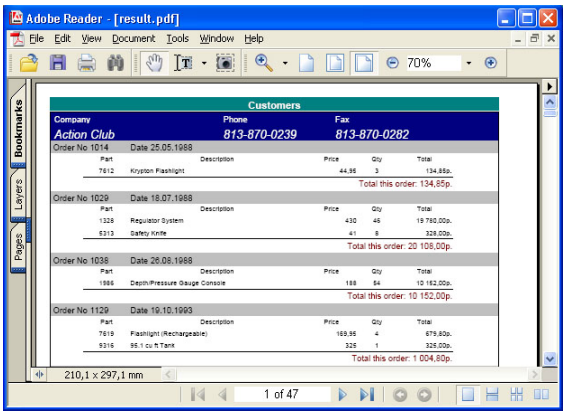
HTML format used by most web-sites in the Internet network is intended for previewing document in low resolution. It is quite difficult to perform high-quality printing of document using this format. HTML format is convenient for most web-browsers. If you use a web-browser as a client, then this format is appropriate for you. FastReport server creates web pages with a report navigator, with the help of which you can scroll the pages.



PDF format by Adobe is designed specially for documents intended for printing.

FastReport makes high-quality export to this format. For viewing and printing PDF documents, you should install Adobe Acrobat Reader program on your computer.

During previewing HTML pages with report results, a file in this format is generated (by pressing the "Print" button on report navigator panel).



The Server also supports the following formats:

- RTF format. A RichText document can be opened in most text processors;
- XLS and XML. These are the Excel spreadsheets formats;

- text and CSV files;
- graphic files jpeg, bmp, gif, tiff.

If type of a returned format is not specified during request, then the server generates the result in HTML format.

Query syntax

When using an ordinary web-browser as a client, you can use parameters of the query line:

report=name

name - name of the report available on server

Example: result?report=1.fr3 (query report 1.fr3, resulting format - HTML).

format=name

name - format of the required file, available formats: HTM (HTML), XML (xml table), XLS (Excel table), RTF (rich-text document), TXT (text file), PDF(Adobe Acrobat file), JPG (jpeg image), FP3 (internal FastReport prepared report format) and CSV, BMP, GIF, TIFF.

By default format is HTM (HTML).

Example: result?report=1.fr3&format=TXT (query report 1.fr3, resulting format - text file).

pagerange=value

value - result page range (for FP3 this option is inaccessible).

Example of the page range: 1,3,5-12.

Example of the query line: result?report=3.fr3&pagerange=20-25 (query report 3.fr3, pages from 20 to 25, resulting format - HTML).

multipage=param

Only for HTM format. If param value is “1”, then the resulting report will be presented as several pages (one file on each page). If param set as “0”, then a single resulting page will be generated that will contain all report pages. Default parameter value setting is “1”.

Example: result?report=3.fr3&multipage=0 (query report 3.fr3, resulting format - HTML, all pages on one HTML page).

pagenav=param

Only for HTM format. To enable page navigator, set param value as “1”. If param value is “0”, then page navigator is off. For correct page displaying, use web-browser with javascript and frames support. Default setting of this parameter is “1”.

Example: result?report=9.fr3&multipage=0&pagenav=0 (query report 9.fr3, resulting format - HTML, all result pages on one HTML page, page navigator is off).

Transferring parameters to the report

If other parameters are presented in the request line (not listed above), then server interprets them as parameters for building a report as internal FastReport variable.

Example:

result?report=myreport.fr3&param1=Hello%20World! (query report "myreport.fr3", FastReport variable "param1" setting is "Hello World!")

Below are some restrictions concerning parameters transferred:

- all strings can be converted in Unicode UTF-8 format and can be compatible with

HTTP query standard;

- all parameters are transferred to report as strings. Please keep in mind this when you use these parameters in the report script;

- all variables contained in FastReport Client are automatically sent to the server.

Internal server variables

During server's working, the Server contains the following automatically created and updated variables:

SERVER_NAME – server name;

SERVER_COPYRIGHT – copyright;

SERVER_SOFTWARE – server version;

SERVER_LAST_UPDATE –last update date;

SERVER_UPTIME – up time of the server;

SERVER_TOTAL_SESSIONS – sum total of sessions;

SERVER_TOTAL_REPORTS – sum total of reports;

SERVER_TOTAL_ERRORS – sum total of errors;

SERVER_MAX_SESSIONS – maximal number of simultaneous sessions;

SERVER_MAX_REPORTS - maximal number of simultaneous report generations.

Example of getting a variable SERVER_TOTAL_REPORTS:

http://127.0.0.1/result?getvariable=SERVER_TOTAL_REPORTS

Using HTML documents

The Server can be used as a simple HTTP server for viewing any HTML documents or any other files.

Correspondingly, a document with this name must exist in the root folder.

SSI (Server Side Include) commands description.

Include any file in document.

<!--#include virtual="filename.html" -->

Include the file with name filename.html in current document position. Path to file is specified from Root Path.

Example:

```
<!--#include virtual="header.html" --> || Command line help
<!--#include virtual="top.html" -->
<font face="Tahoma" size="3"><a href="index.html"><b>Back to main
page</b></a><b><br>
</b></font>
<hr>
...
```

Insert value of server variable.

<!--#echo var="VARIABLE"-->

Insert the value of variable with the "VARIABLE" name in current document position.

Example:

```
...
<tr>   <td align="right" width="200"><b>Uptime:</b></td>
       <td width="300"><!--#echo var="SERVER_UPTIME"--></td></tr>
<tr>   <td align="right"><b>Total sessions:</b></td>
       <td><!--#echo var="SERVER_TOTAL_SESSIONS"--></td></tr>
```

```
<tr>   <td align="right"><b>Total reports:</b></td>
       <td><!--#echo var="SERVER_TOTAL_REPORTS"--></td></tr>
<tr>   <td align="right"><b>Max sessions:</b></td>
       <td><!--#echo var="SERVER_MAX_SESSIONS"--></td></tr>
<tr>   <td align="right"><b>Max reports:</b></td>
       <td><!--#echo var="SERVER_MAX_REPORTS"--></td></tr>
...
```

Use of SSI commands optimizes website development.

Example of the site with SSI you can see in the “\htdocs” folder.

Logs

The server supports 5 logs:

- log of the accessed clients “access.log” - contains information about date, time, session id, IP and query line. Log fragment:

```
10/26/2004 23:56:19 sid_f1672494035    192.168.0.2  result?report=3.fr3
10/26/2004 23:56:23 sid_f1340767011    192.168.0.2  sid_f1672494035/index.html
10/26/2004 23:56:23 sid_f1949776310    192.168.0.2  sid_f1672494035/index.nav.html
10/26/2004 23:56:23 sid_f1150188690    192.168.0.2  sid_f1672494035/index.1.html
```

- log of the connected program type “agent.log”, contains information about date, time, IP, and program name. Log fragment:

```
10/26/2004 23:56:19 192.168.0.2 Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
10/26/2004 23:56:23 192.168.0.2 Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
10/26/2004 23:56:23 192.168.0.2 Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
```

- log of the referencing URLs “referer.log”, contains information about date, time, IP and referencing URL. Log fragment:

```
10/26/2004 23:56:19 192.168.0.2  http://192.168.0.1/
10/26/2004 23:56:23 192.168.0.2  http://192.168.0.1/
10/26/2004 23:56:23 192.168.0.2  http://192.168.0.1/sid_f1672494035/index.html
```

- errors log “error.log”, contains information about errors:

```
10/25/2004 13:30:52 192.168.0.2  588864044016/index.1.html document not found
10/26/2004 0:03:11  192.168.0.2  Software caused connection abort.(10053)
10/26/2004 0:43:42  192.168.0.2  Connection reset by peer.(10054)
```

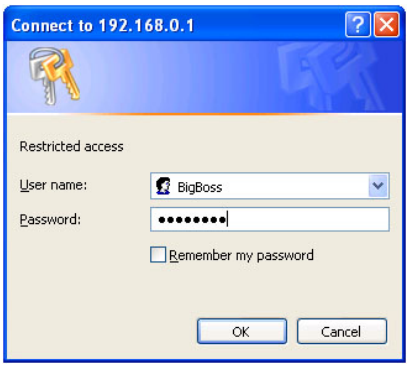
- server log “server.log”, contains summary server information:

```
10/25/2004 19:38:15 Started
10/25/2004 19:38:15 HTTP server created
10/25/2004 19:58:57 HTTP server closed
10/25/2004 19:58:57 Stopped
Uptime: 0 days 0 hours 20 minutes 42 seconds
Total sessions: 654
Total reports: 327
Total errors: 0
Max sessions: 84
Max reports: 42
```

Do not forget to archive the log files.

Authentification

The server supports basic HTTP authentication. If you set the “Login” properties, then request header must contain authentication info (RFC 2068 [2]). If client receives answer from server with 401 “Unauthorized” error coded, then the client must retry sending the query with correct authentication data. At that, web browser simply shows dialog window with login and password request:



Access restriction by IP address

The Server supports the restriction by client IP address.

Property TfrxReportServer.DenyIP can contain list of the restricted client IPs'.

Property TfrxReportServer.AllowIP can contain list of the allowed client IPs'.

Each list must contain one IP address in one line.

Here is an example of such list:

```
192.168.0.10
192.168.0.12
192.168.0.54
```

If the "DenyIP" and "AllowIP" lists are empty, then all clients are allowed to connect to the server.

If the "DenyIP" list is empty, while the "AllowIP" list contains an IP address, then only one client with this IP address can connect to the server.

If the IP address of a connected client is not included in the "DenyIP" list, then the server checks if this address is included in the "AllowIP" list.

IP addresses' masks are not supported.

Examples:

1. Only local host can connect to the server:

AllowIP:

```
127.0.0.1
```

DenyIP is empty.

2. IP addresses 192.168.0.2 – 192.168.0.6 can connect to the server.

AllowIP:

```
192.168.0.2
192.168.0.3
192.168.0.4
192.168.0.5
192.168.0.6
```

DenyIP is empty.

3. IP addresses from range 192.168.0.8 – 192.168.0.10 cannot be connected to the server.

AllowP is empty.

DenyIP:

```
192.168.0.8
192.168.0.9
192.168.0.10
```

Database connections

Most of reports use data from databases. To connect to a database, you should

use internal data access components (such as TfrxADOTable, TfrxADOQuery) in your report. To connect to the database, these components should use the default connection or internal report connection TfrxADODatabase.

You will be able to connect to different databases at one time.

Read the "FastReport 3 - User manual" to learn more about creating reports with internal data access components.

Using the reports cache

Caching of reports allows achieving high efficiency because of saving prepared reports in temporary files of the server. Depending on server configuration, after preparation the result can be placed in cache.

After specified time, the result of the report will be removed from the cache.

If during this time a query with the *same name of the report and the same values of parameters* is received from a client, the response will be immediately returned to it. The reply will be based on the result saved in cache, and will be represented in the format requested by the client.

In that case, the server will waste time only on conversion of the prepared report in the requested format without building a report. It considerably increases the productivity.

Depending on tasks performed by a server, it is possible to assign an individual storage time in cache for each particular report.

Time value is set by the administrator of a server, according to actuality of a report, after certain period of time.

For example, the annual report about activity of an enterprise can be stored in cache long enough, since the information will be relevant for a long period of time, and it would not become outdated very soon. On the contrary, a report about a large commercial organization warehouse would be relevant during a small period, and therefore it consequently should be stored in cache not too long.

Correction of parameters configuration will minimize time of working clients and will reduce total traffic on the server.

Increasing server's processing power

Use the following recommendations to increase the report server performance:

- optimize your SQL queries. In some cases running the SQL query may take a longer time than the report execution;

- do not use high-resolution bitmaps in your reports - it will increase the report execution time and network traffic;

- do not use complex scripts in your reports;

- when developing a report, keep in mind recommendations from 4.2;

- increase the memory size, use the faster CPU on PC used as a report server.

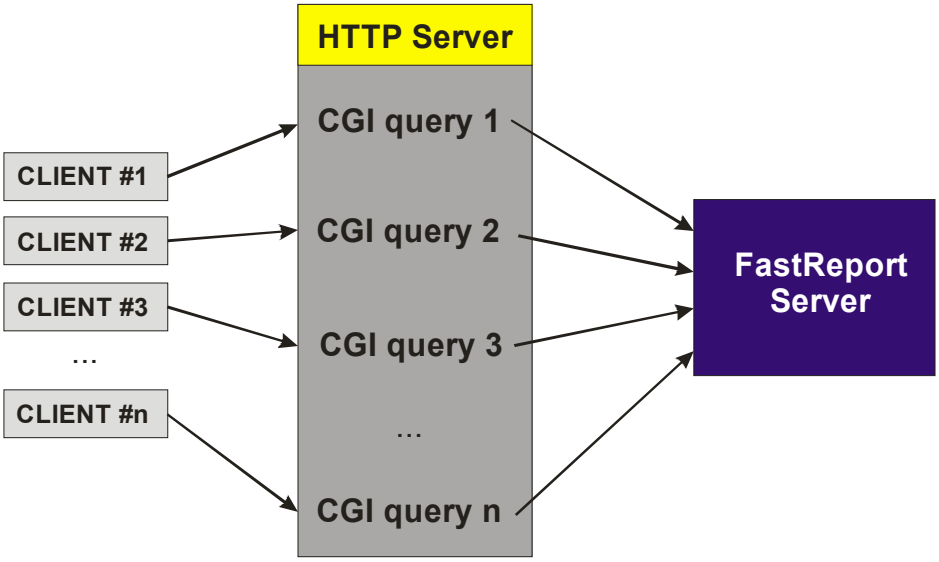
Using the FastReport Server together with other HTTP servers (Apache, IIS, etc)

CGI wrapper

To use already existing solutions based on other HTTP servers, their integration with the FastReport server is possible by means of the “CGI” mechanism. It gives an advantage in comparison with using a built-in HTTP server FastReport. Reports can be

built in an already-working system (site). HTTP server and a server of reports can work on different computers. Usage “SSL” encoding for operation with HTTP a server is possible (this possibility is unavailable in HTTP server FastReport yet).

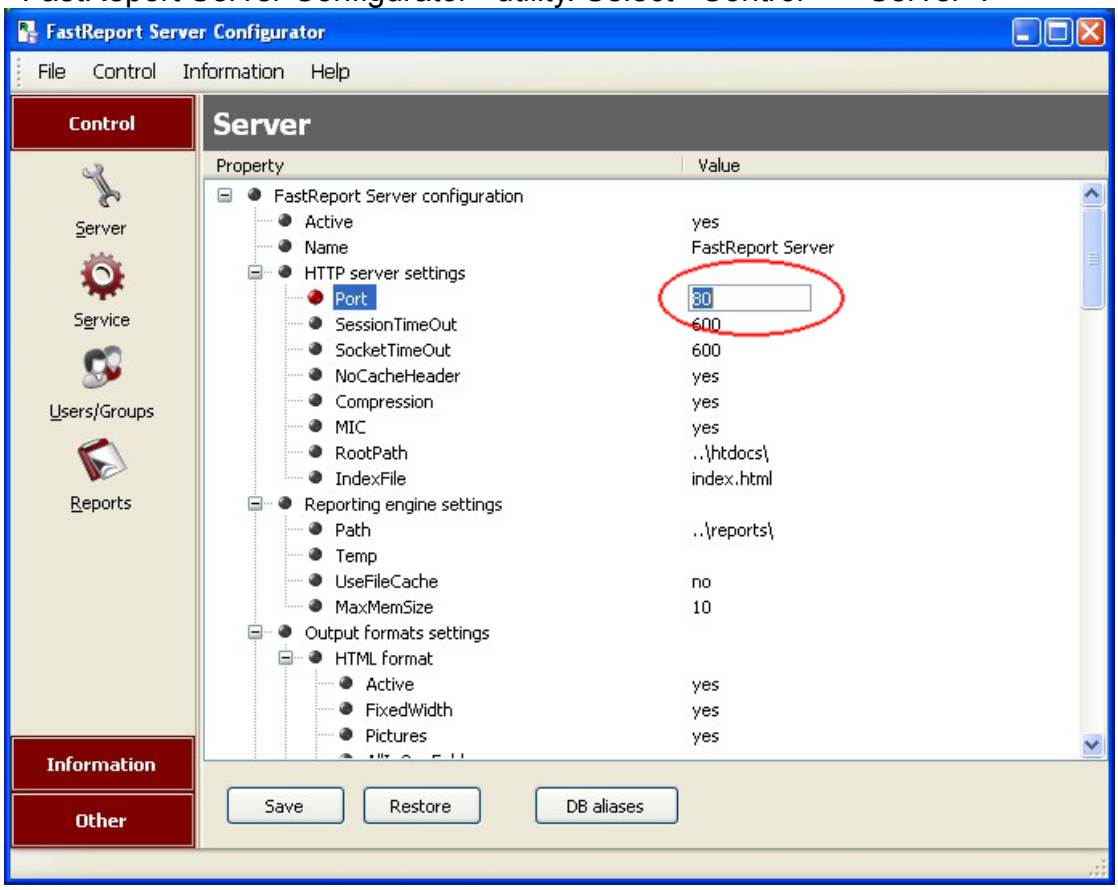
Applying such method, CGI becomes an intermediate for tranfering a query to the “FastReport” server, obtaining results from a server of reports, and return of the results to the client.



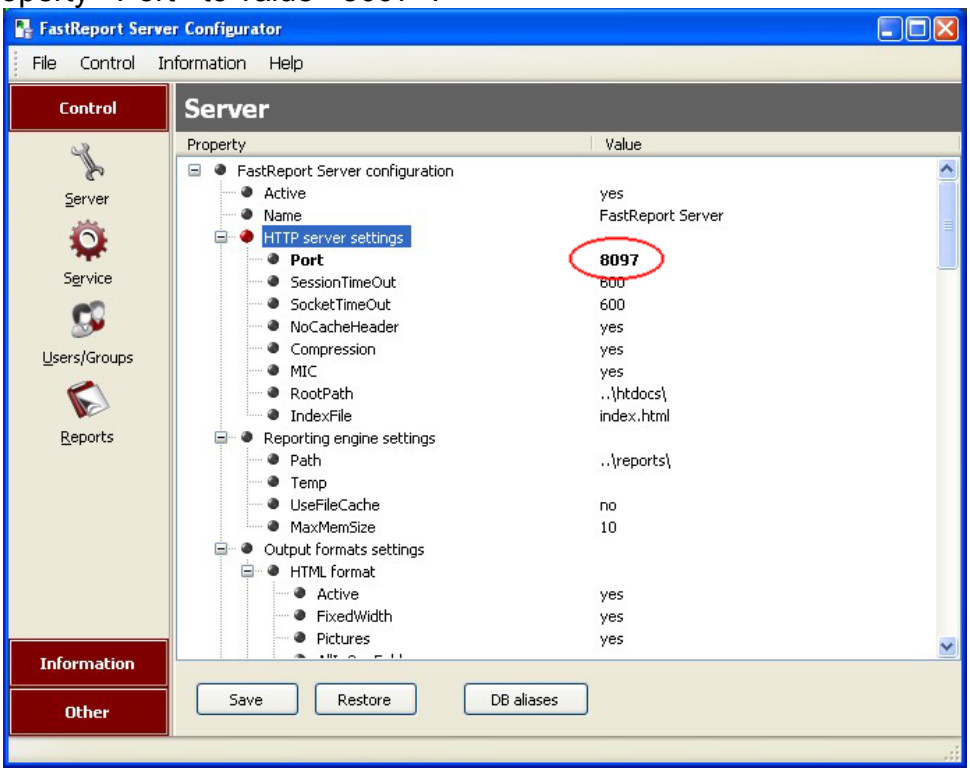
You can found the CGI wrapper in the “\bin” folder.

To us the CGI wrapper:

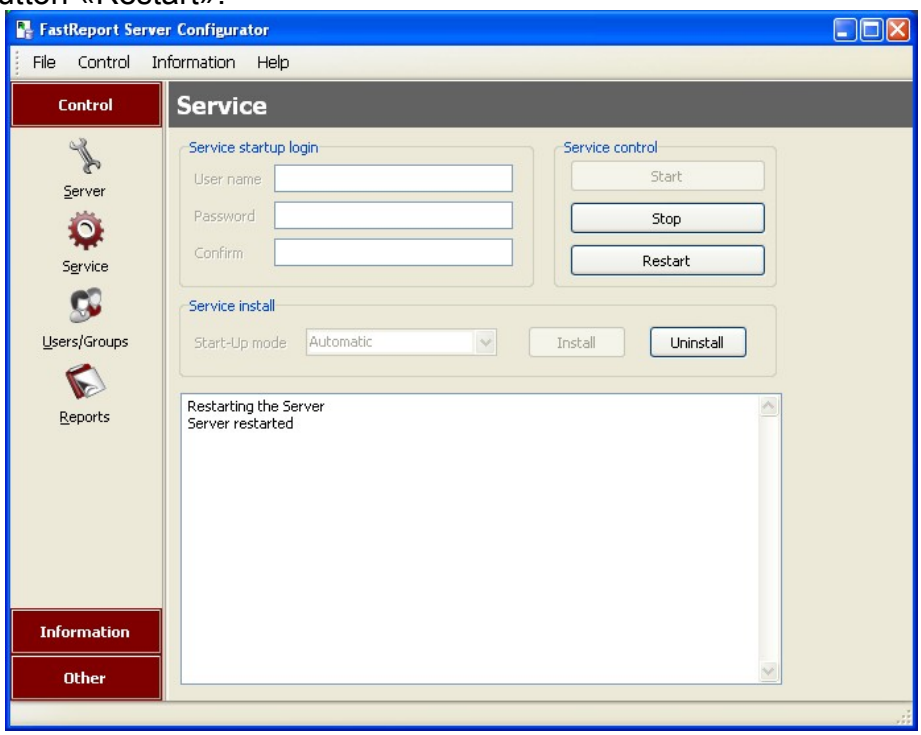
Run «FastReport Server Configurator» utility. Select «Control» - «Server».



Set the property «Port» to value «8097».



Press the button «Save» and click «Yes». Open the tab «Control» - «Service» and press the button «Restart».



You can set other values in the file «cgi-bin\fastreport.ini» to for correct FastReport Server connection (default: Host=127.0.0.1, Port=8097).
You can restrict access to FastReport Server from other computers if you write in «FastReport Server\Bin\allow.conf» local IP «127.0.0.1».

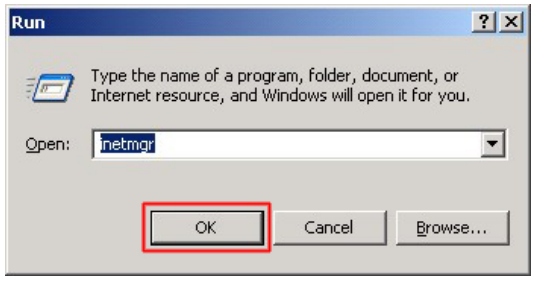
Report query example with using of CGI application:
http://127.0.0.1/cgi-bin/fastreport.exe?report=67.fr3&multipage=0&pagenav=0

Replace the “result” keyword in this point at “cgi-bin/fastreport.exe” construction.

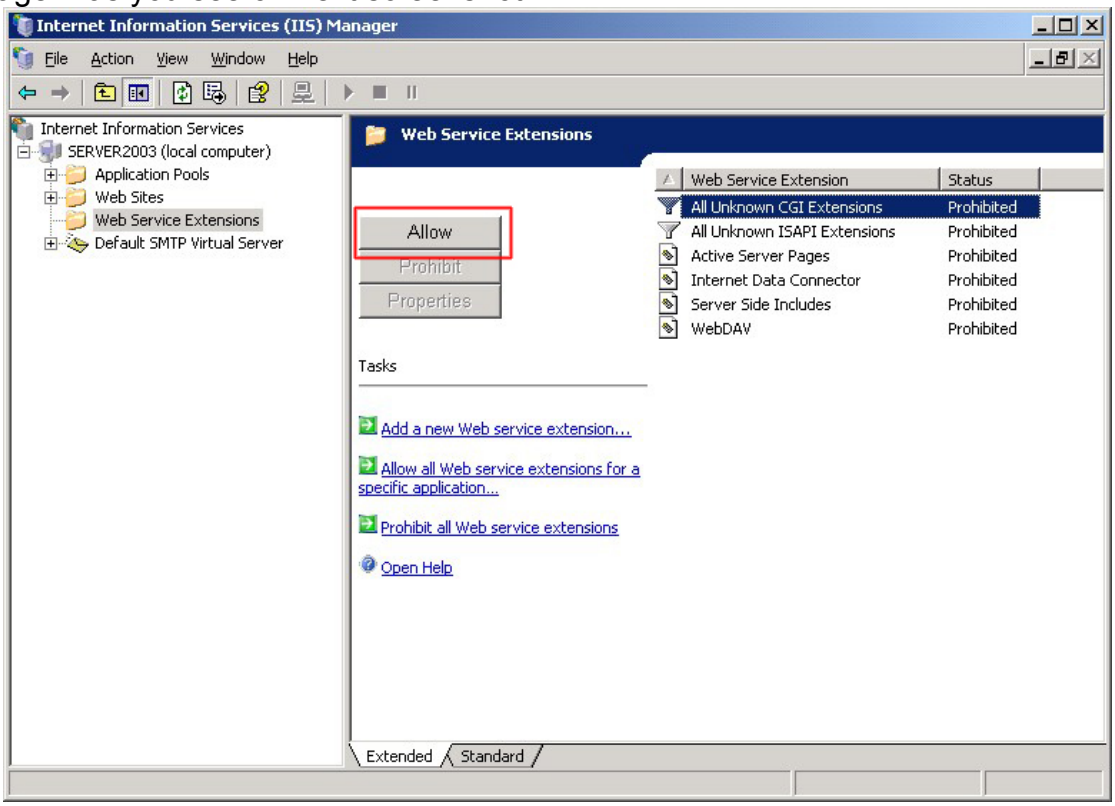
Using the FastReport Server CGI together with Microsoft IIS v 6.0

Note: solution tested with the IIS 6.0 and Miscrosoft Windows 2003 Web Edition (build 3790).

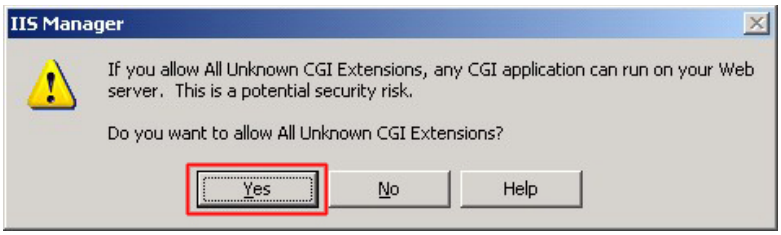
1. You should create the folder named «cgi-bin» in site location directory (by default c:\Inetpub\wwwroot). Copy all files from «\FastReport Server\Bin\cgi\» to the «cgi-bin» folder.
2. Press Start and select Run. Enter the text «inetmgr» and press OK.



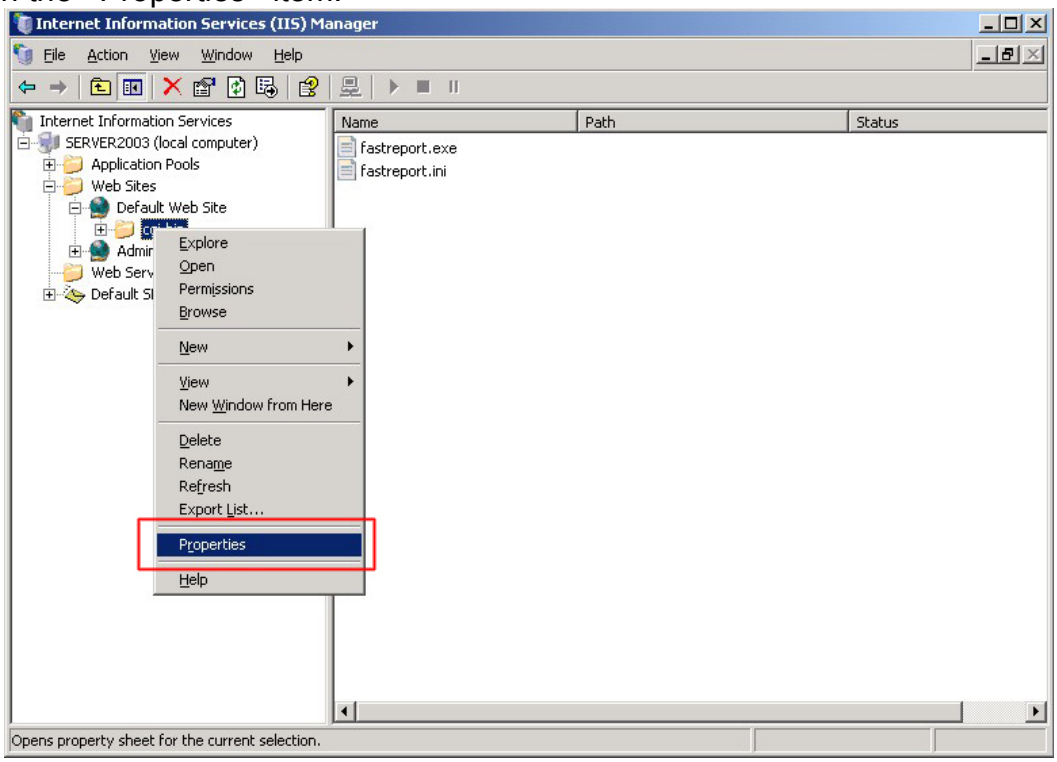
Open the «Web Service Extensions» after launch the «Internet Information Service Manager» as you see on next screenshot.



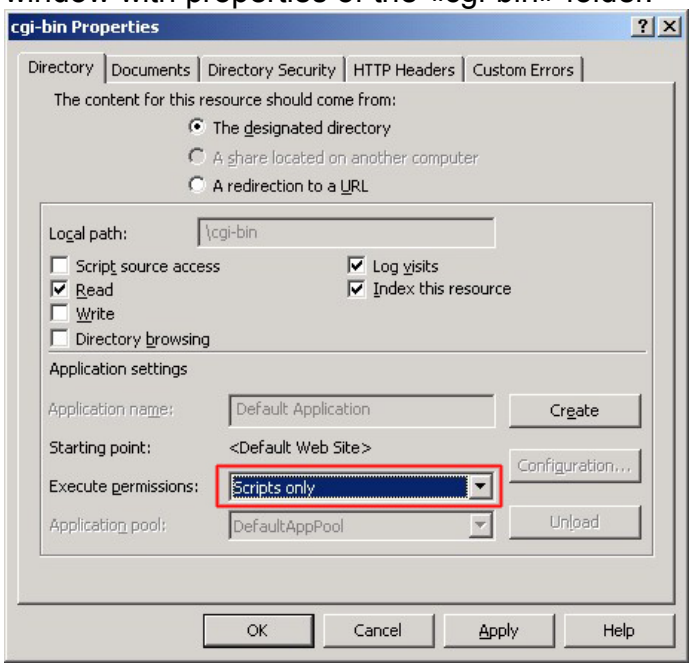
Select «All Unknown CGI Extensions» and press the button «Allow».
Select OK in next window.



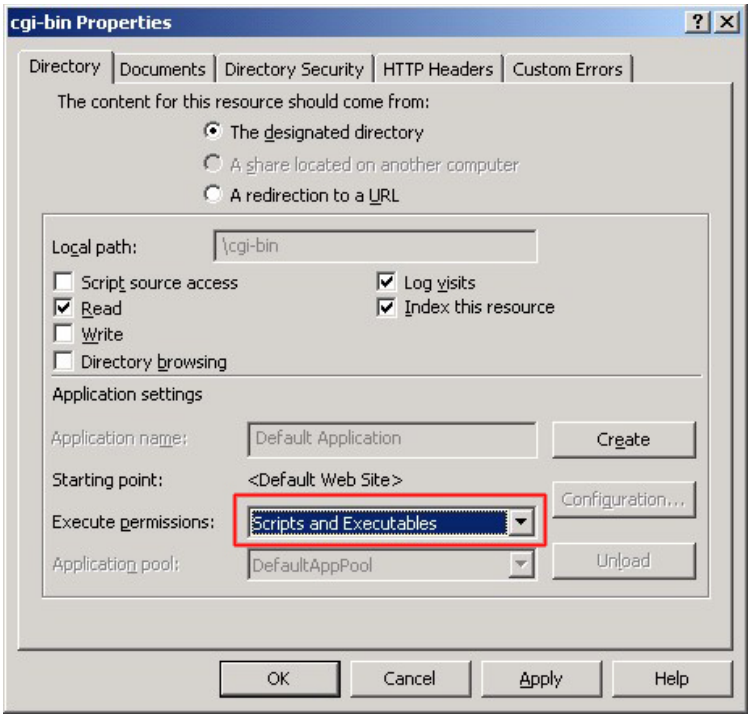
Then go to «Web Sites» section and click right mouse button on the «cgi-bin» folder. Click on the «Properties» item.



You should see the window with properties of the «cgi-bin» folder.

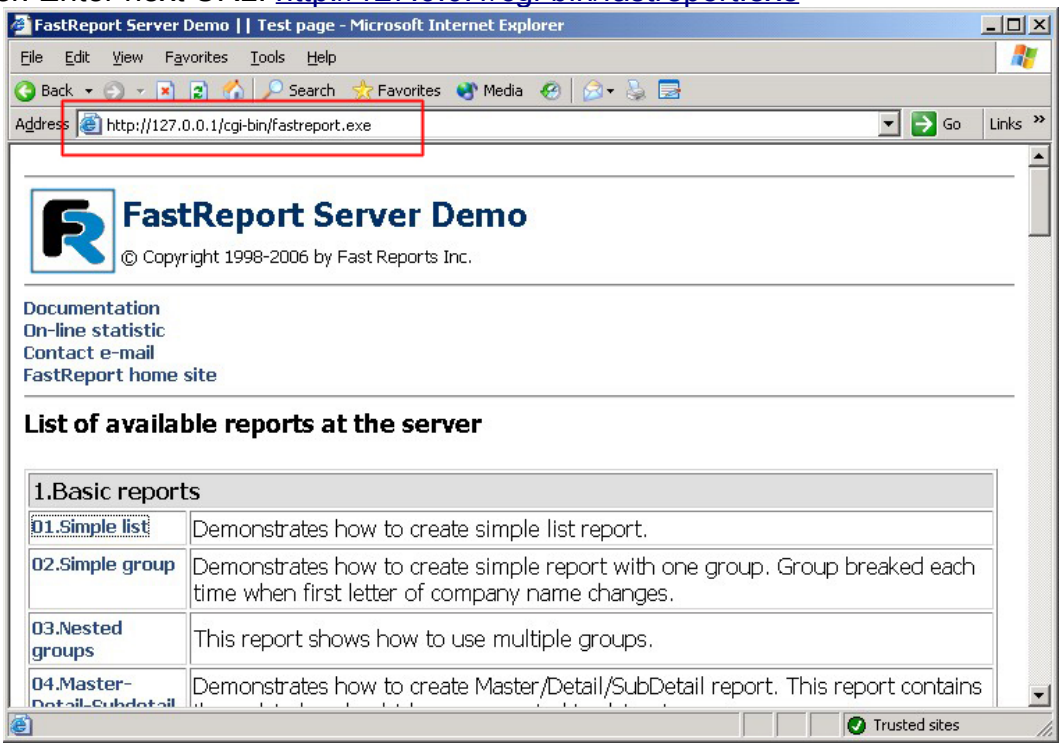


You should set the «Execute permissions» into «Scripts and executables» on the «Directory» tab.



Press «OK». Configuration of the IIS is done.

3. Open the Internet Explorer for check working the FastReport Server through CGI wrapper. Enter next URL: http://127.0.0.1/cgi-bin/fastreport.exe



You can see this picture if all works fine.

Using the FastReport Server CGI together with Apache Web Server v 2

Open the httpd.conf file (default location: “\Program Files\Apache Group\Apache2\conf\httpd.conf”) in text editor. Find directory description of “/Program Files/Apache Group/Apache2/cgi-bin”. Add “ExecCGI” directive to the “Options” line.

Example:

```
<Directory "C:/Program Files/Apache Group/Apache2/cgi-bin">
```

```
    AllowOverride None
    Options None ExecCGI
    Order allow,deny
    Allow from all
</Directory>
```

Copy all files from «\FastReport Server\Bin\cgi\» to the «cgi-bin» folder. Restart the Apache Web server.

Using the FastReport Server CGI together with Lotus Domino Web Server v 7.0

1. Create the “cgi-bin” folder in “\Lotus\Domino\data\domino” location (default “C:\Program Files\Lotus\Domino\data\domino”).

2. Copy all files from «\FastReport Server\Bin\cgi\» to the «cgi-bin» folder.

3. Open the Internet Explorer for check working the FastReport Server through CGI wrapper and enter the URL
http://127.0.0.1/cgi-bin/fastreport.exe

FastReport Server ISAPI extension

Internet Server Application Programming Interface (ISAPI), is an API developed to provide the application developers with a powerful way to extend the functionality of Internet Information Server (IIS).

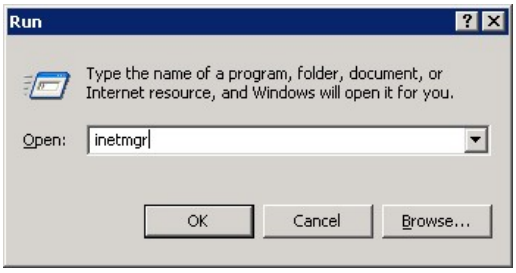
The FastReport Server ISAPI extension encapsulates all multithread features, logging etc. You can configure the FastReport Server ISAPI extension from FastReport Server Configuration utility. FastReport ISAPI extension can work with Internet Information Server 5.0 (Windows Server 2000), 6.0 (Windows Server 2003), and with Apache 2.0.55 (Win32) and later. Important! FastReport Server ISAPI extension can not work with Apache 1.3x.

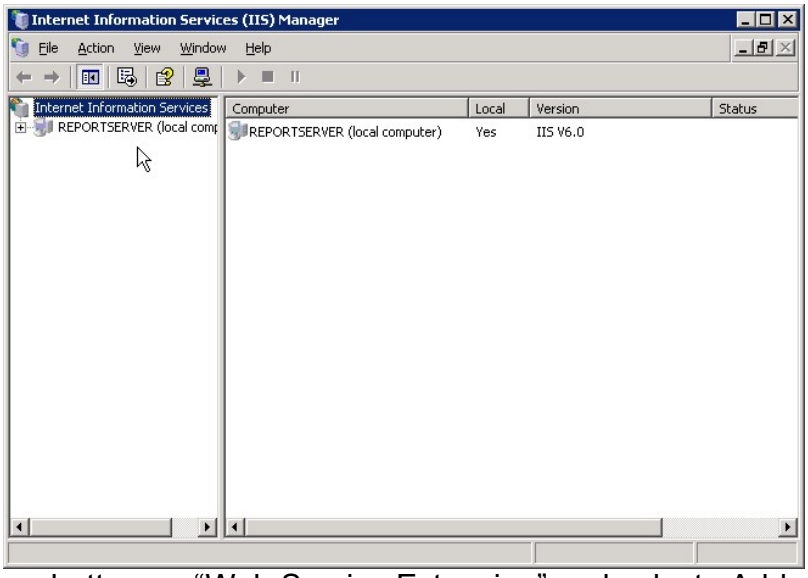
Prepare for use of FastReport Server ISAPI extension.

You should stop the FastReport Server service before using the FastReport Server ISAPI extension. Load the FastReport Server Configuration Utility for performing of this requirement and press the button «Uninstall» in «Service» section.

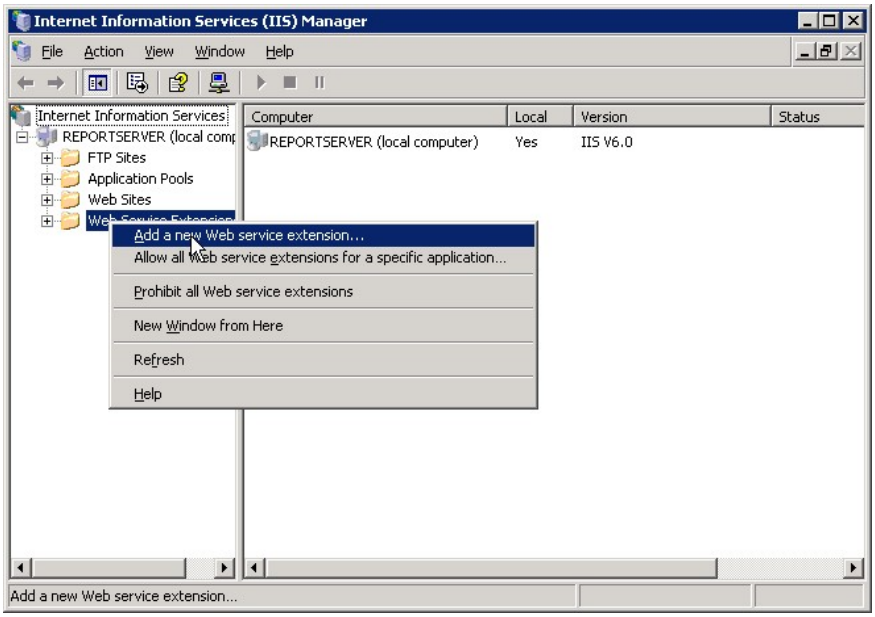
Install the FastReport Server ISAPI extension together with IIS 6.0

1. Press “Start” button and select Run. Enter «inetmgr» and press OK for launch the Internet Information Services Manager.

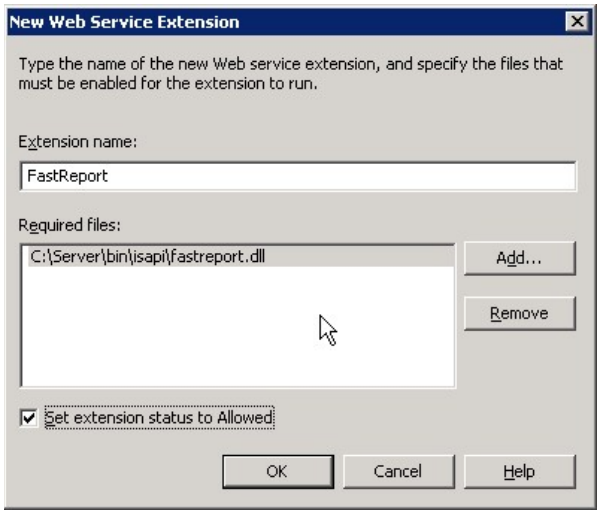




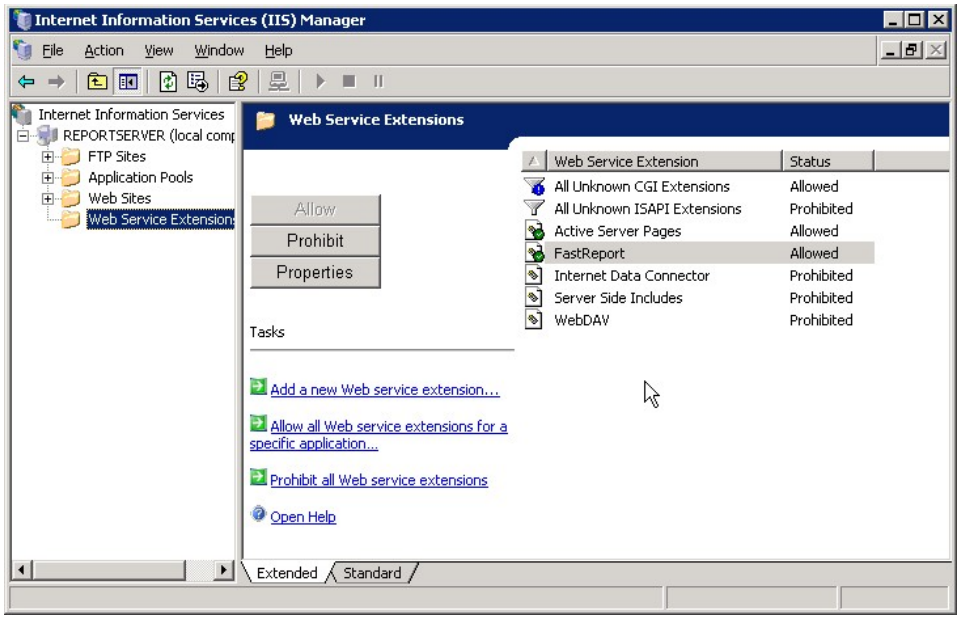
Click right mouse button on “Web Service Extension” and select «Add new Web service extension...»



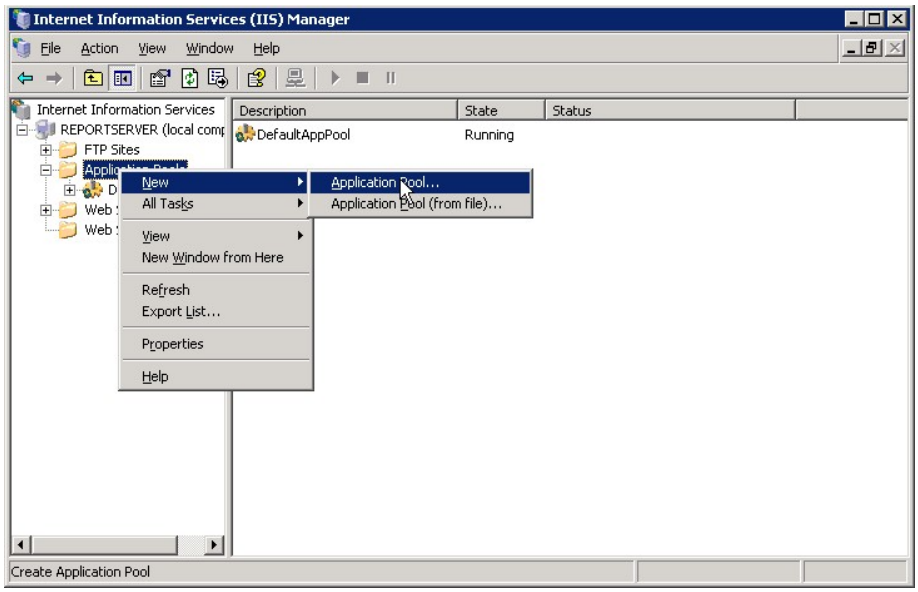
Enter extension name «FastReport» and add FastReport Server ISAPI dll file in «Required files» list (default: «C:\Program Files\FastReports\FastReport Server\bin\isapi\fastreport.dll»). Check «Set extension status to Allowed».



Press «OK». See the FastReport Server extension in list.

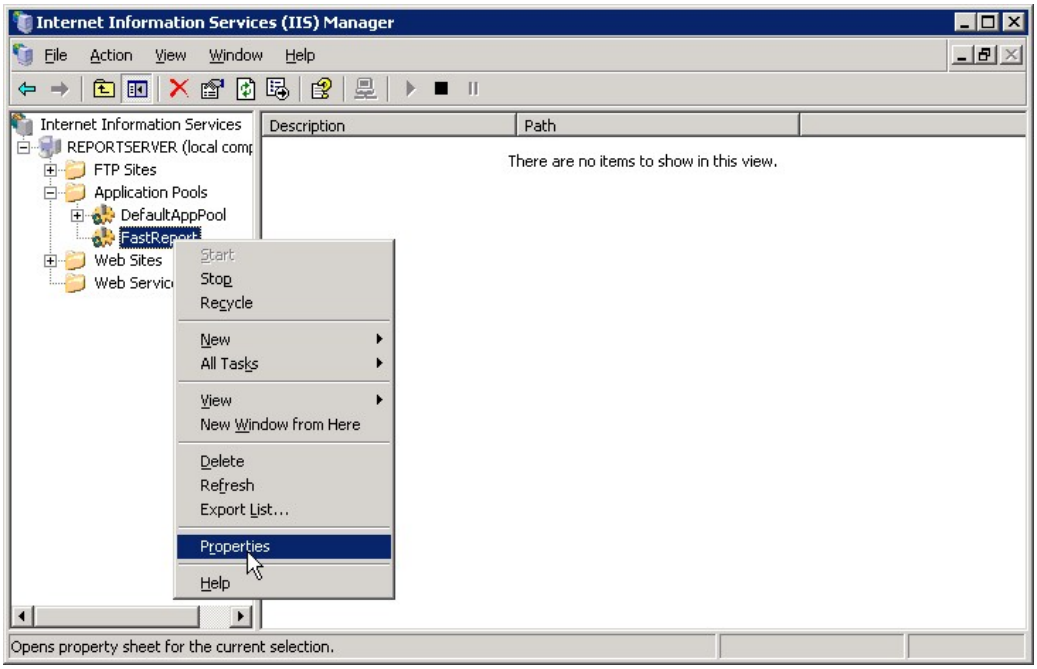


Press right mouse button on «Application Pools» and select «New – Application Pool...».

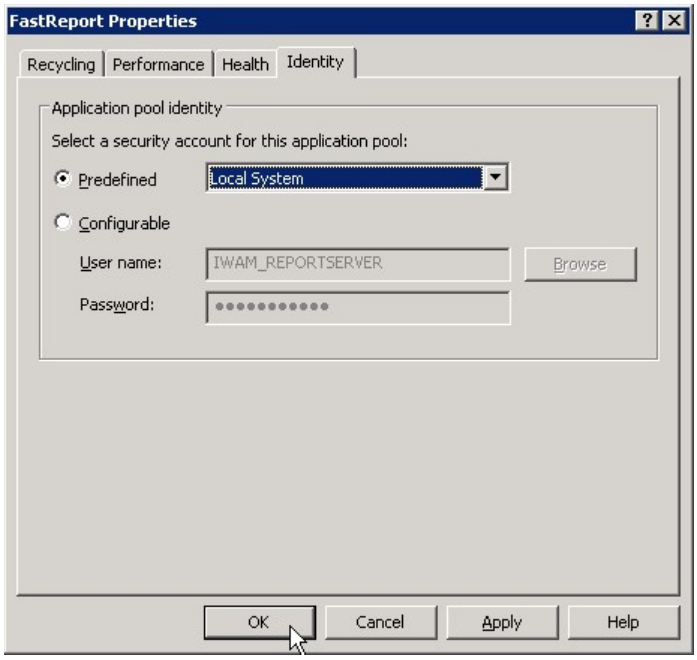


Enter application pool ID «FastReport» and press «OK».

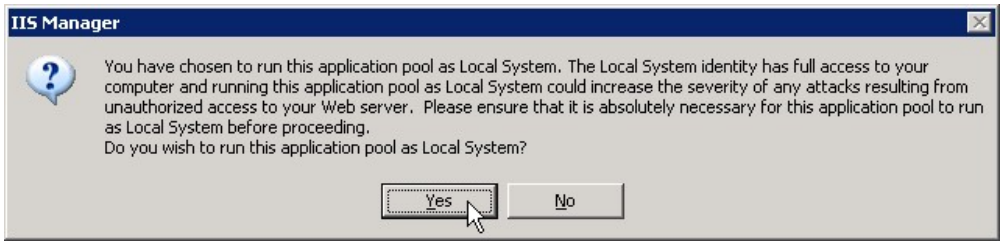
Later change the properties of new application pool. Click right mouse button on “FastReport” pool in в section “Application Pools” and select “Properties”.



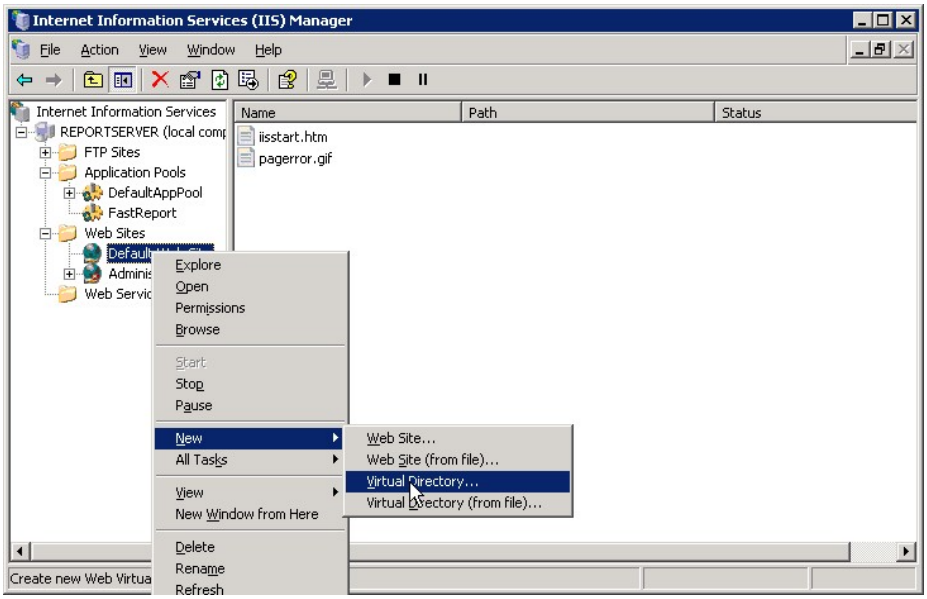
Go to page «Identity» and change the Security account from “Network Service” to “Local System”.



Press «OK» and click on «Yes».

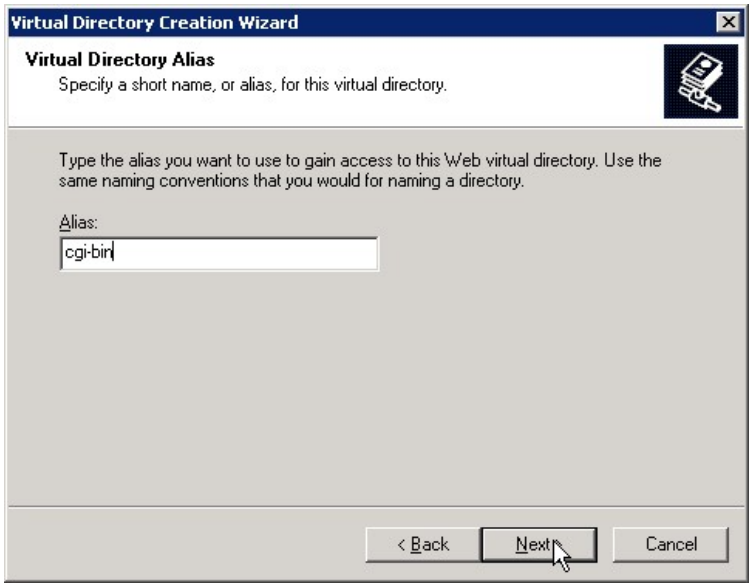


Open the section «Web Sites». Click right mouse button on necessary site (default «Default Web Site»). Select «New – Virtual Directory...».

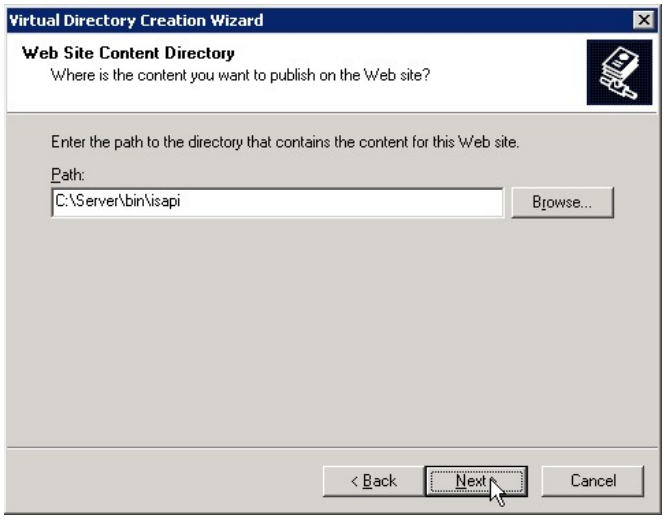


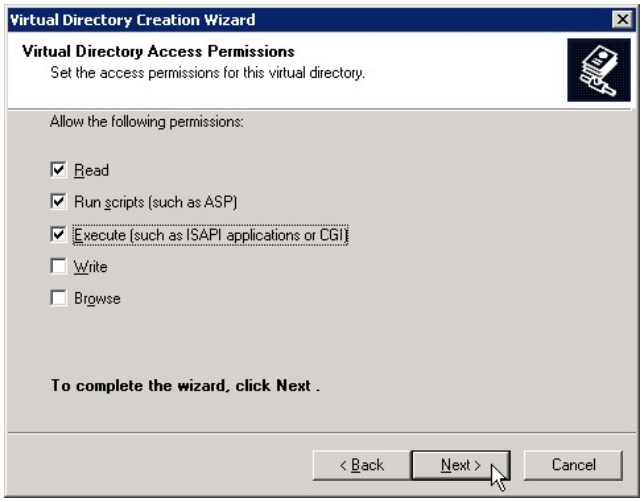
Press «Next» button.

Enter alias name (default «cgi-bin»). Press «Next».

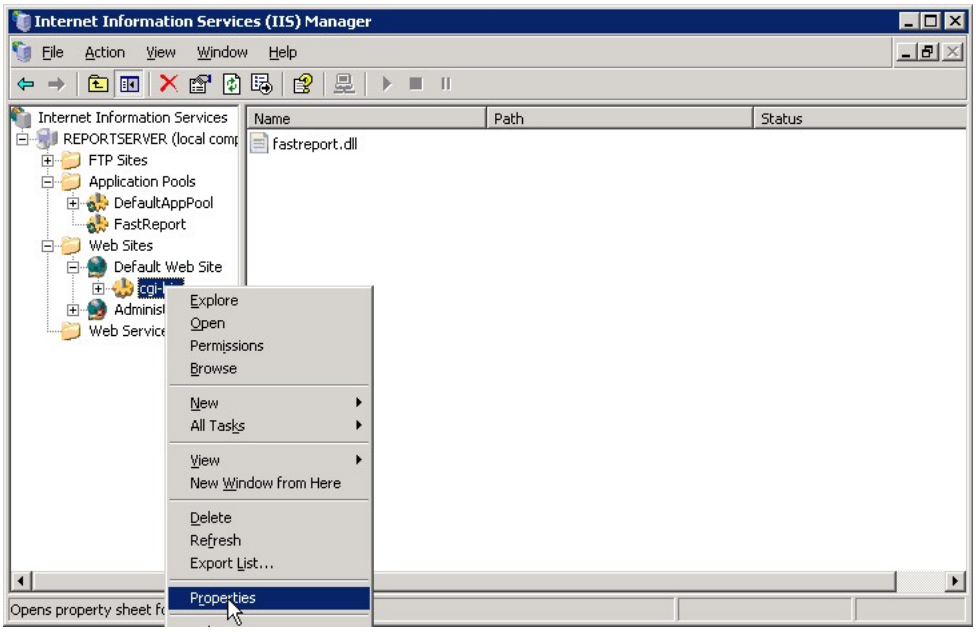


Select folder with FastReport Server ISAPI extension in next window (default: «C:\Program Files\FastReports\FastReport Server\bin\isapi»). Press «Next». Check «Execute (such as ISAPI applications or CGI)» in next window and press «Next».

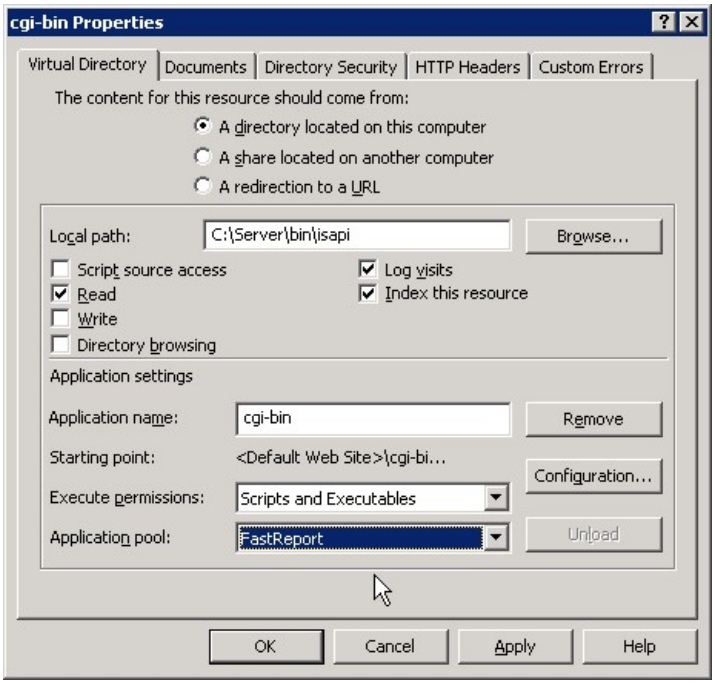




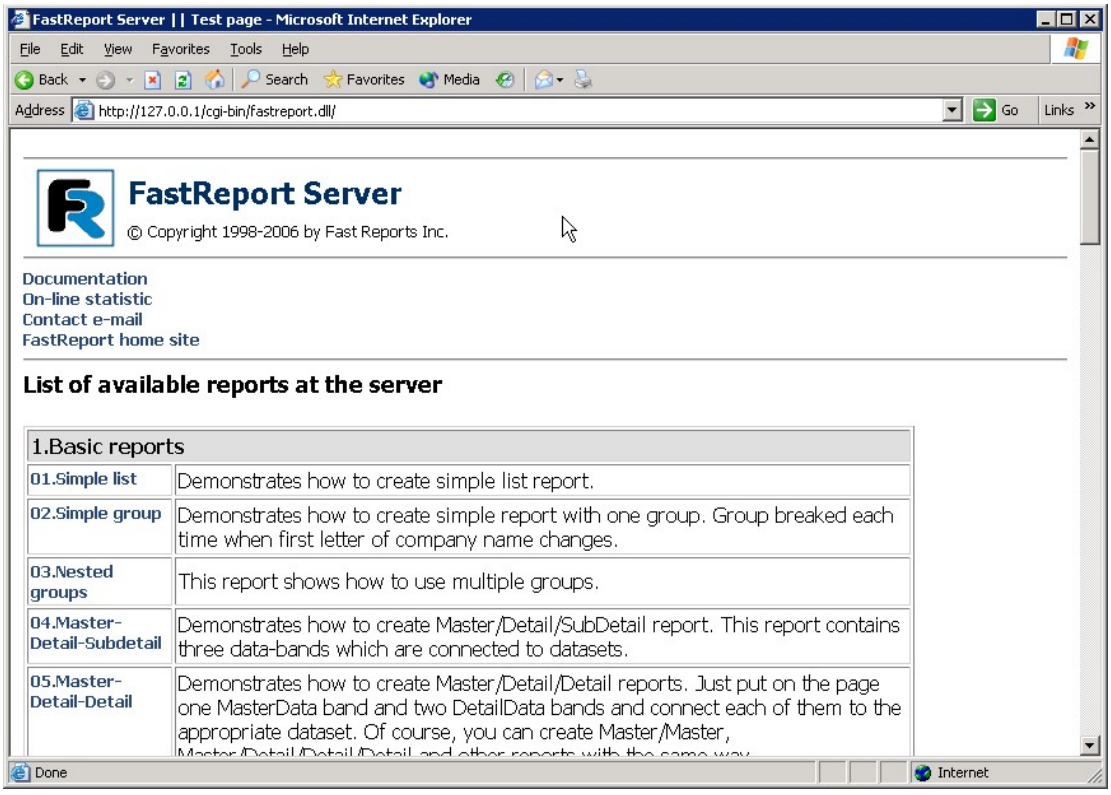
Click on «cgi-bin» virtual folder and select «Properties».



Go to page «Virtual Directory» and set property «Application pool» to «FastReport». Click «OK».



Enter in web browser http://localhost/cgi-bin/fastreport.dll/. If all works fine, you will see FastReport Server Welcome page.



Install FastReport Server ISAPI extension together with Apache 2.x

1. Open the configuration file httpd.conf (default C:\Program Files\Apache Group\Apache2\conf\httpd.conf).

2. Check and uncomment next line:

```
LoadModule isapi_module modules/mod_isapi.so
```

3. Add the «ExecCGI» value in Options property for the ISAPI directory.
Example:

```
<Directory "C:/Program Files/Apache Group/Apache2/cgi-bin">
    AllowOverride None
    Options None ExecCGI
    Order allow,deny
    Allow from all
</Directory>
```

4. Add two lines below on end of the configuration file:

```
AddHandler isapi-isa .dll
ISAPIFakeAsync on
```

5. Copy the file fastreport.dll (default C:\Program Files\FastReports\ FastReport Server\bin\ISAPI\fastreport.dll) to the /cgi-bin folder.

6. Restart the Apache server.

Examples of using the FastReport Server ISAPI extension

Get index page from FastReport Server:
http://localhost/cgi-bin/fastreport.dll/
Run the report named “01.Simple list.fr3”:
http://localhost/cgi-bin/fastreport.dll/result?report=1.Basic%20reports\01.Simple%20list.fr3

Developing the reports

Developing of the FastReport reports was described in the "FastReport 3 - user manual."

Some advices concerning the design of a report

Many of the document formats use table-style data representation. For representing of resulting reports, the server uses such formats as HTML, XLS, and RTF.

Table-style documents cannot have intersected cells, while FastReport document can. FastReport uses free-form data layout - there is no "lines", "table cells" like in Word, Excel or other such formats. FastReport export filters for table-style formats (RTF, HTML, and XLS) uses special algorithm to convert intersected cells into table cells and optimally arranges them. In places where FastReport objects intersect with each other, export filter may generate additional table rows and columns. It is necessary for better WYSIWYG, but may result in increased number of rows and columns in a resulting layout, which makes the table layout unusable for further analysis and slows down the export process.

Keep in mind these export limitations when developing a report, if you intend to export your report into such table-style formats. To avoid the objects' intersection, use alignment tools of the FastReport designer. Turn on the "grid align" option.

When creating tables in a report, put the table cells side-by-side, if possible, and avoid cells’ intersection. If cells are intersected, the export algorithm would make clipping, and the export result may differ from the original report.

If possible, place objects along the horizontal and vertical guide lines. Use designer's guide lines to do this.

Following these instructions would help your reports to look perfect during exporting to any of the supported formats.

Clients

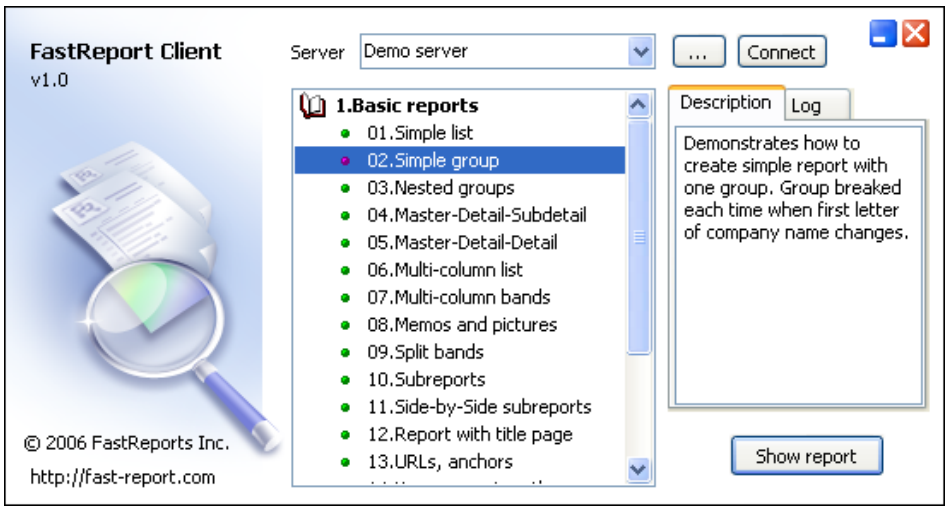
There are two kinds of clients of the FastReport Server:

- applications that use TfrxReportClient component;
- any stand-alone HTTP-clients, such as web-browsers.

Native FastReport Client

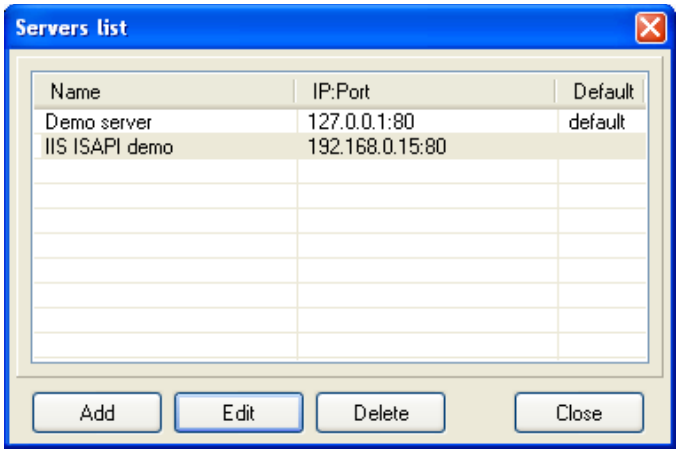
FastReport Client is designed specially for client applications. This component allows querying a report from the server, passing some report parameters (variables) to the server. It receives prepared report in the FP3 format (native FastReport format). The prepared report can be displayed and printed on the client side. You can also export the prepared report to any of the supported formats, using export filter components. In most cases, this solution is optimal for client applications. Clients that use the TfrxReportClient component make low network traffic and use less server system resources. You can use this Client in custom application together FastReport Studio SDK. You can see example in folder «examples».

As example the FastReport Server package contain the FastReport Server Client (bin\frxClient.exe).

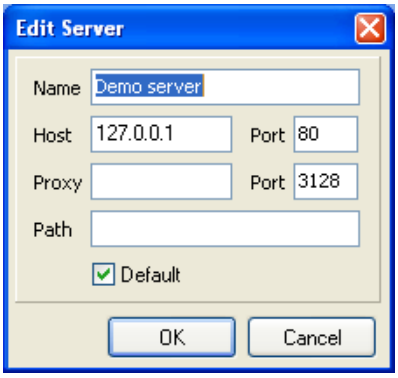


FastReport Client can connect to many FastReport Servers, can get reports list and show selected report.

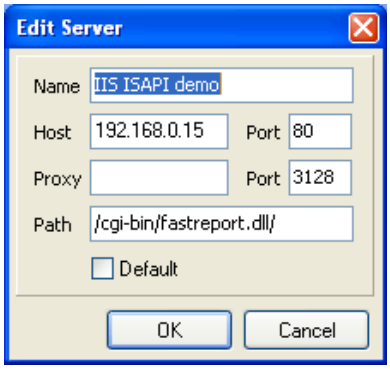
List of servers window:



Server edit window:

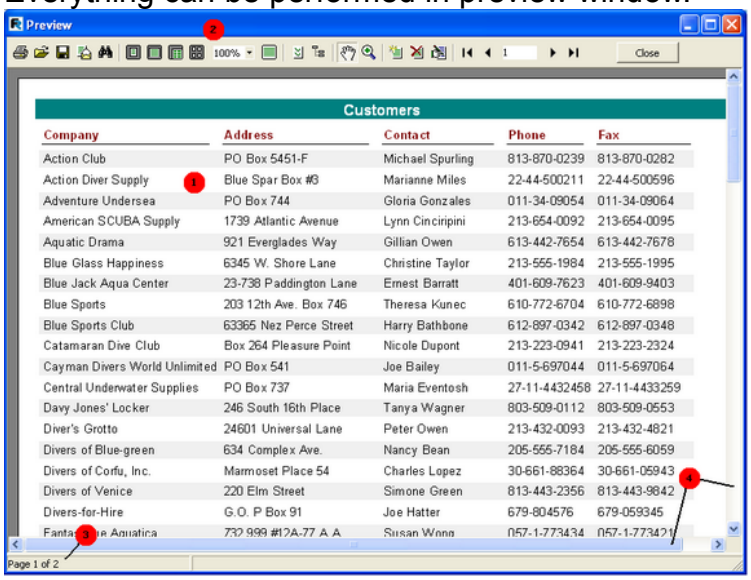


Name – FastReport Server name,
Host, Port – IP address and port of FastReport Server
Proxy, Port – IP address and port of HTTP Proxy Server
Path – path to ISAPI extension on server, (blank by default,
for example «/cgi-bin/fastreport.dll/»)
Default – connect to this server on Client run.
Example of this window for work with ISAPI extension:



Report viewing, printing and export

The built report can be displayed, printed or exported into one of the supported formats. Everything can be performed in preview window.



On the picture with figures the following is displayed:
1 – ready report;
2 – toolbar;

3 – status line;
4 – scroll bars.

The following buttons are on the toolbar:

Icon	Name	Description
	Print report	Prints report. Hotkey analogue – Ctrl+P.
	Open report	Opens file with ready report (*.fp3).
	Save report	Saves report to file (*.fp3).
	Report export	Exports report to one of the supported formats.
	Export to PDF	Exports report to Adobe Acrobat file (*.pdf).
	Send via e-mail	Exports report to one of the supported formats and sends it via e-mail as enclosure.
	Text search	Text search in report. Hotkey analogue – Ctrl+F.
	Scale	Entire page.
		Page edgewise.
		Initial size (100%).
		Two pages side by side.
100%		Selects arbitrary scale.
	Full screen	Displays report at full screen. For returning to normal conditions perform double-click on report.
	Page properties	Calls dialogue with page properties.
	Report tree	Shows or hides report tree.
	Hand	Hand tool. Move mouse with pressed right button for report scrolling.
	Magnifier	Magnifier tool. Left button of mouse zooms in, the right one – zooms out.
	New page	Inserts empty page to report.
	Delete page	Deletes current page from report.
	Edit page	Edits current page.
	To beginning	Transfer to the first report page.
	Previous page	Transfer to previous report page.
1	Page number	Transfer to report page with pointed number. Enter number and click Enter.

	Next page	Transfer to next report page.
	To end	Transfer to the last report page.
Close	Close window	Close preview window.

Control keys

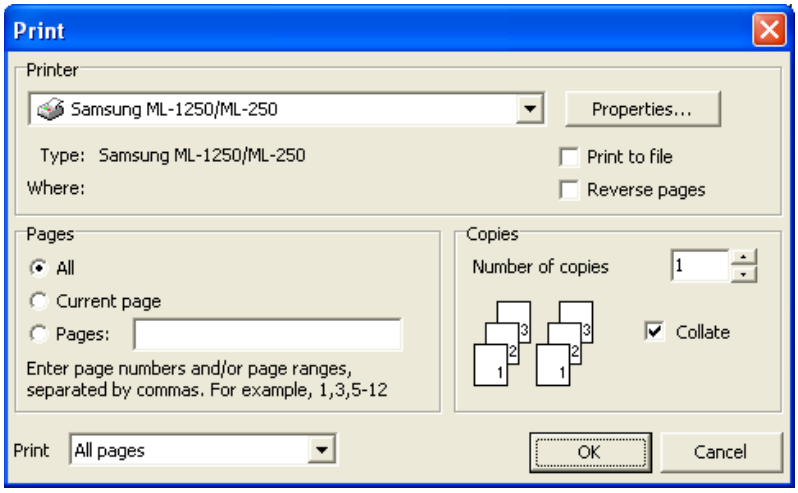
Keys	Description
Ctrl+S	Save report to *.fp3 file.
Ctrl+P	Print report.
Ctrl+F	Text search.
F3	Continue search.
Стрелки	Smooth document scrolling.
PageUp, PageDown	Up/down scrolling.
Ctrl+PageUp, PageDown	Next/previous page scrolling.
Home	Document beginning.
End	Document end.

Mouse control

Action	Description
Left button	Click on selected object (in interactive report); report scrolling in “hand” mode (move mouse with pressed button); zoom in is performed in “magnifier” mode.
Right button	Context menu; in “magnifier” mode zoom out is performed.
Double-click	It full screen mode it performs returning to normal conditions.
Mouse scroll	Report list scrolling.

Report printing

To print a report click on button (or Ctrl+P hotkey). The window appears – it is printing dialogue.



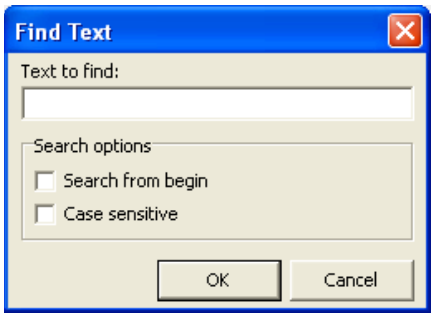
You can select a printer via which you want to print a report. Also you can specify the following settings:

- Properties – set printer properties, for example, printing quality;
- Print to file – printing is not performed via printer, report is saved to file instead;
- Reverse pages – pages are printed in reverse order, beginning with the last one;
- All – print all report pages;
- Current page – print only current page;
- Pages – print pages with specified numbers;
- Copies – how many report copies are to be printed;
- Collate – if there is a task to print several copies, at first one report copy is printed, then – the next etc. If flag is disabled, several copies of the first page are printed, then – several copies of the second one etc.;
- Print all pages – which pages are to be printed. Variants: All pages, Even pages, Odd pages.

After clicking on OK report printing begins. If the “Print to file” flag is selected, file name is called. And report is saved to this file (file with *.prn extension. It contains a copy of information sent to printer).

Text search in report

FastReport allows to search a set text line in a text in preview window. To perform that there is button on toolbar (or its hotkey analogue - Ctrl+F). After that search dialogue appears:



Here you can set search line and options as well:
- Search from beginning – to search text from the beginning of document. Otherwise search will be performed from current page;
- Distinguish letter case – to distinguish letter cases (lower-case and capital types) on

searching.

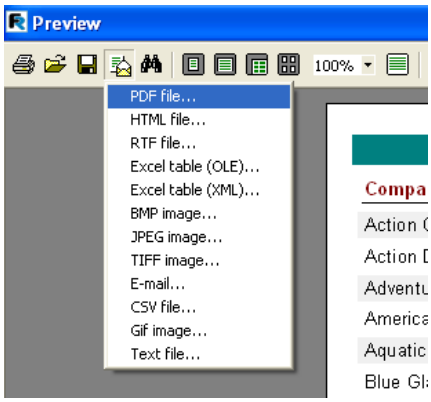
On clicking OK text search is performed and the first found element is highlighted:

To continue search click F3. The following element will be highlighted

Report Export

FastReport allows exporting a built (output) report to different formats for further editing, archiving, sending via e-mail, etc... To export you must add the desired FR export components to the Delphi form.

Export to 11 formats is supported. They are: PDF, Excel, XML, RTF, HTML, text, CSV, BMP, Jpeg, Tiff, and Gif. There is the ability to send report via e-mail in any above-listed formats with FastReport means.



Exports in FastReport use one of the following three methods:

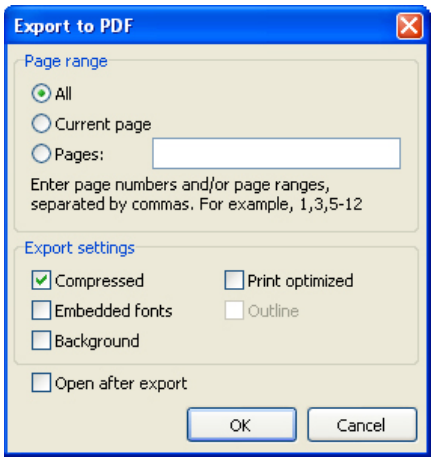
- Layer-by-layer – object transferring to resulting file is performed alternately. Expert accuracy is approximated to original;
- Table – on object transferring transitional matrix of object allocation is used. There is high accuracy to original based on the assumption that rules of creating correct report sample were followed (“Report Design References” chapter);
- Enveloping – all report objects enveloping is performed on page image. There is full original accuracy. It is used on export to graphic formats.

Export to PDF Format

PDF (Portable Document Format): a platform-stand alone format of electronic documents created by Adobe Systems. The free Acrobat Reader package is used for viewing. This format is rather flexible – it allows inclusion of necessary fonts, vector and bitmapped images; it allows transferring and storage of documents intended for viewing and further printing.

Export method is a layered one.

On exporting to PDF format the dialogue box for output file parameter settings appears.



Export parameters:

- Compressed – output file compressing. It reduces file size but increases export time;
- Embedded fonts – all fonts used in report will be contained in the PDF output file for correct file displaying on computers where these fonts may be absent. Output file size increases considerably;
- Background – export of graphic image assigned to a page into PDF file. It considerably increases output file size;
- Print optimized – output of graphic images in high resolution for further correct printing. This option enabling is necessary only when the document contains graphics and its printing is necessary. It considerably increases output file size;
- Outline – option is enabled when report outline is used. It enables export of the outline to the PDF document;
- Open after export – resulting file is opened right after export via PDF files viewing program which must be installed in OS by default (for example, Adobe Acrobat Reader).

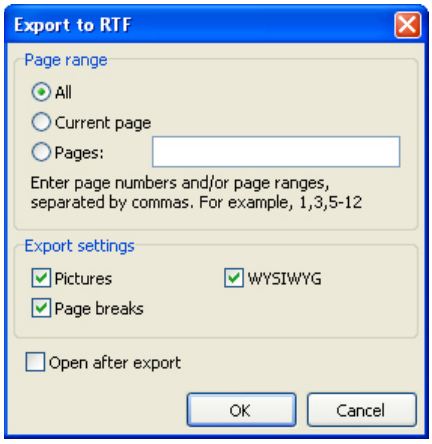
Export peculiarities: RichText objects are exported as a graphic.

Export to RTF Format

RTF (Rich Text Format) was developed by Microsoft as a standard for text documents interchange. Now RTF documents are supported by many modern text editors and operating systems.

Export method is a table one.

On exporting to RTF format the dialogue box for output file parameter settings appears.



Export parameters:

- Pictures – enables graphic images export to output file;
- Page breaks – enables page breaks in RTF file;
- WYSIWYG – full compliance to report appearance. Disabling the option allows optimization, reducing the number of lines and columns in the output file;
- Open after export – output file will be opened right after export via RTF files viewing program which must be installed in OS (for example, Microsoft WordPad).

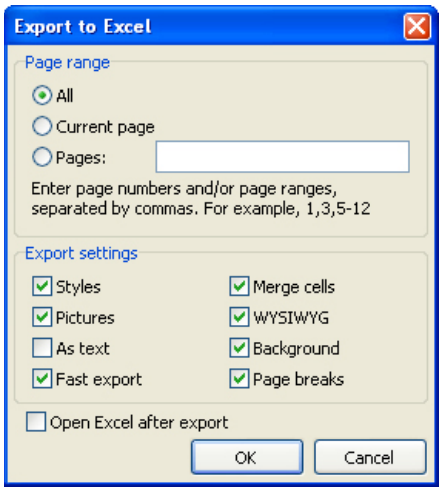
Export peculiarities: RichText objects are fully integrated into RTF format. File appearance and size depend on report sample accuracy (“Report Design References” chapter);

Export to Excel

Excel – application for working with electronic worksheets. It is included into Microsoft Office System.

Export method is a table/diagram one.

On exporting to Excel format the dialogue box for output file parameter settings appears.



Export parameters:

- Styles – transferring of text objects design styles into the table. Disabling increases

exporting speed but worsens document appearance;
- Pictures – includes graphic images export into output table;
- As text – all objects are transferred into table/diagram as text ones. This option may be useful when transferring numeric fields with complicated formatting;
- Fast export – usage of optimized fast data transferring to Excel. This option disabling slows down data transferring but increases export compatibility on any errors during data transferring;
- Merge cells – cells integration in resulting table/diagram for achieving maximum correspondence to the original. Disabling increases exporting but reduces document appearance;
- WYSIWYG – full compliance to report appearance. On this option disabling the optimization for reducing the number of lines and columns in resulting table is performed;
- Background – export of filling color assigned to report page;
- Page breaks – includes page breaks in Excel;
- Open Excel after export – resulting file will be opened right after exporting into Excel.

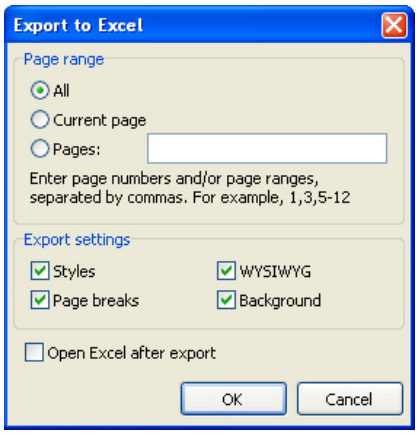
Export peculiarities: Excel program must be installed on your PC. RichText objects are transferred as simple text, graphic images transference is supported.

Export to XML Format

XML (Extensible Markup Language) is an expansible marking language. XML is intended for structured data storage and also for information interchange between different programs. FastReport uses XML format for data transferring into Excel table/diagram editor ver. 2003 and later.

Export method is a table/diagram one.

On exporting to XML format the dialogue box for output file parameter settings appears.



Export parameters:

- Styles – transferring of text objects design styles. Disabling speeds up exporting but worsens document appearance;
- WYSIWYG – full compliance to report appearance. Disabling allows reducing the number of lines and columns in resulting table;
- Background – export of filling color assigned to report page;
- Page breaks – includes page breaks in resulting document;
- Open Excel after export – resulting file will be opened right after exporting into Excel.

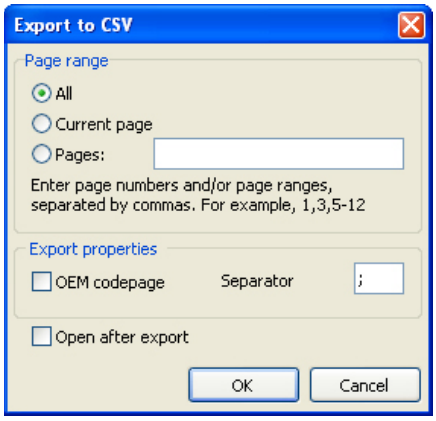
Export peculiarities: Excel program must be installed on your PC. RichText objects are transferred as a simple text; graphic images transference is not supported.

Export to CSV Format

CSV-file contains values formatted in the form of a table/diagram and adjusted in such a way that every value in column is divided from value in the next column be means of separator, and every new row begins with new line. This format may be imported into different table/diagram editors.

Export method is a table/diagram one.

On exporting to CSV format the dialogue box for output file parameter settings appears.



Export parameters:

- OEM codepage – resulting file OEM coding selecting;
- Separator – values separator in files;
- Open after export – resulting file will be opened right after exporting via CSV files viewing program which must be installed in OS.

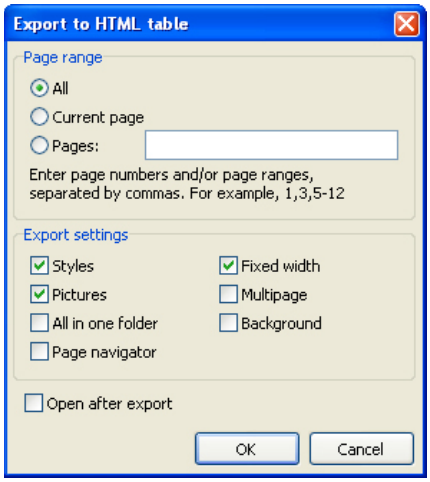
Export peculiarities: on transferring into this format report design is not saved. Graphic images are not supported.

Export into HTML Format

HTML (Hypertext Markup Language) is regarded as standard language for document marking in the Internet. HTML was created as a language for scientific and technical documentation interchange suitable for usage by people who are not specialists in nesting. It is used for creating relatively simple but nicely designed documents. Besides document structure simplification hypertext support is included into HTML.

Export method is a table/diagram one.

On exporting to HTML format the dialogue box for output file parameter settings appears.



Export parameters:

- Styles – transferring of text objects design styles. Disabling increases exporting but worsens document appearance;
- Pictures – includes graphic images exporting possibility;
- All in one folder – all additional files are saved in the same folder with main file;
- Page navigator – special navigator for fast shift between pages is created;
- Fixed width – blocking of automatic table/diagram width modifying on changing preview window size;
- Multipage – every page will be written to separate file;
- Background – export of graphic attributes assigned to report page;
- Open after export – resulting file will be opened right after exporting via HTML files viewing program which is allocated in OS by default.

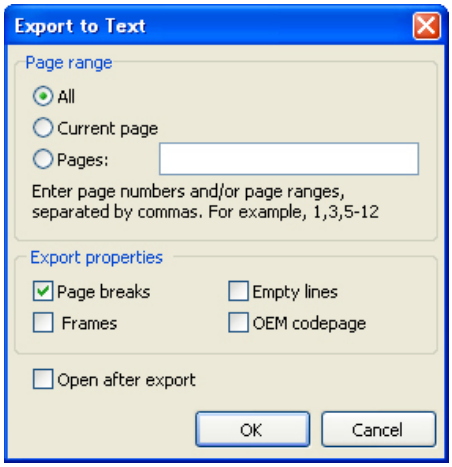
Export peculiarities: export may consist of several files. Each graphic image is supported and saved in their own separate file, RichText objects are transferred as simple text. Appearance and file volume depend greatly on report sample (“Report Design References” chapter)

Export to Text Format

Usual text file. It contains information from report. This information is optimized to the limit and converted in accordance with the given format peculiarity.

Export method is a table/diagram one.

On exporting to text format the dialogue box for output file parameter settings appears.



Export parameters:

- Page breaks – export of page breaks to resulting file;
- Empty lines – export of empty lines;
- Frames – export of text objects frames;
- OEM codepage – resulting file OEM coding selecting;
- Open after export – resulting file will be opened right after exporting via default text files viewing program which is installed in OS.

Export peculiarities: report design is not saved on transferring to this format, graphic images are not supported, exported page width is figured automatically depending on type of text objects on report page.

Export to Jpeg, BMP, Gif, Tiff Graphic Formats

FastReport allows exporting information to graphic formats.

JPEG (Joint Photographic Experts Group) – is a format based on shrinking algorithm which is based not on the same elements search but on difference between pixels. It is characterized by high compression level at the expense of partial graphic information loss.

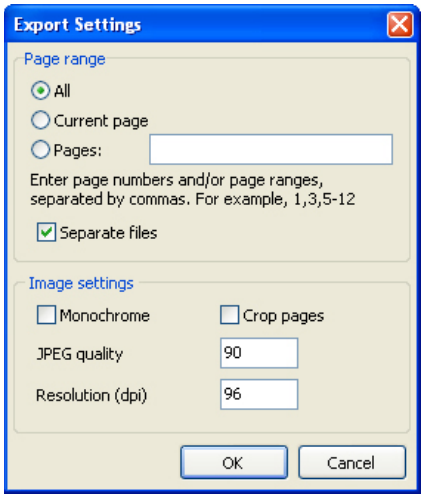
BMP (Windows Device Independent Bitmap) – is used for storage of bitmap images assigned for usage in Windows. A standard file format for computers under Windows control.

GIF (Graphics Interchange Format) – independent from hardware support the GIF format was developed for bitmap images transferring through networks. It allows compression of files containing many homogeneous fillings (logos, inscriptions, schemes) rather well.

TIFF, TIF (Target Image File Format) – hardwarily stand-alone format. Today it is one of the most widespread and reliable in polygraphy and facsimile information transferring.

Export principle is enveloping.

On exporting to one of above-named graphic formats the dialogue box for image parameters setting appears.



Export parameters:

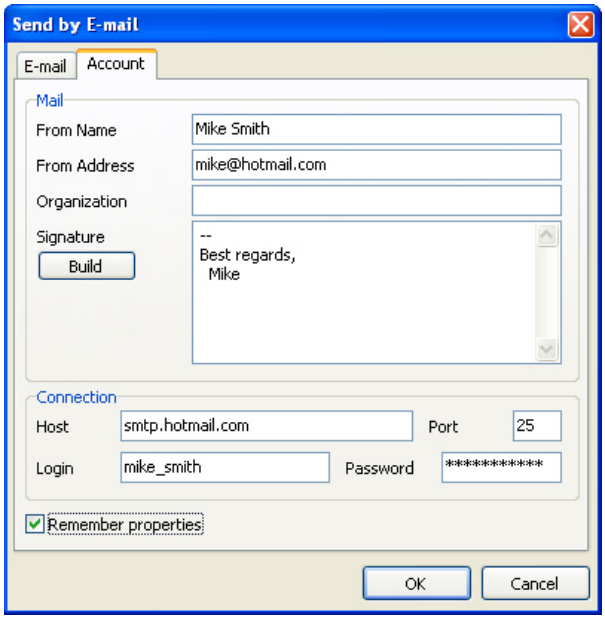
- Separate files – if option is enabled, every report page is exported to separate file. File name is given according to the selected one with addition of underlining and page number;
- Monochrome – monochrome picture creating;
- Crop pages – after exporting blank area cropping will be performed along edges;
- JPEG quality – JPEG file compression ratio. Option is enabled only on exporting to JPEG format;
- Resolution – output graphic presentation resolution.

Export peculiarities: on exporting several pages to one file (on disabled Separate files option) it is necessary to remember large resources capacity of export.

Sending a Report via E-mail

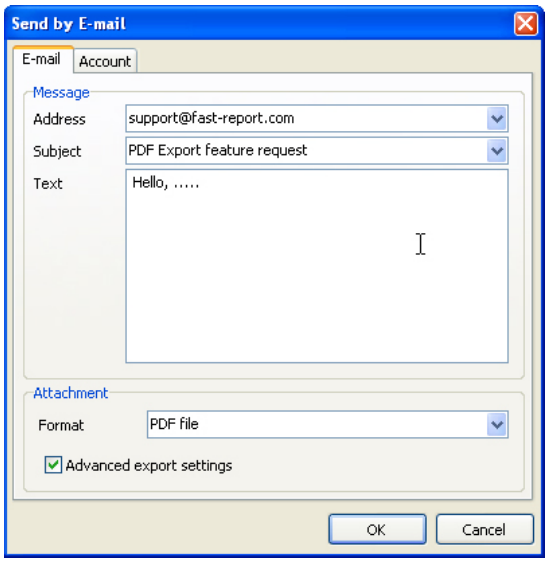
FastReport allows sending a ready report by e-mail in any format you need. You do not need any additional programs to send mail.

On selecting export by e-mail the dialogue box for setup of message and exporting format parameters appears. Before export forming and sending via e-mail, it is necessary to set parameters of mail box owner. All these parameters are on the “Account” page tab:



- From Name – sender's name;
- From Address – sender's e-mail;
- Organization – sender's organization;
- Signature – signature for mail. It may be automatically formed on clicking on "Build" button on condition that the earlier examined fields are filled;
- Host – SMTP server port;
- Port – SMTP server port;
- Login – access name for authorization on SMTP server, if its usage is necessary for mail sending via specified SMTP server;
- Password – authorization password;
- Remember properties – remember all parameters for further usage.

After filling in the necessary parameters for mail sending, you must fill in message parameters in "E-mail" page tab:



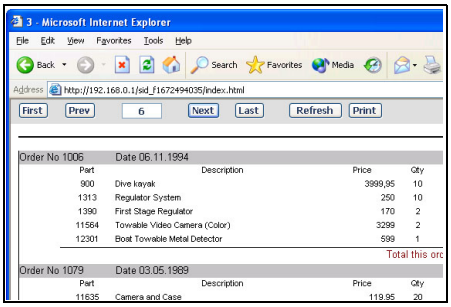
- Address – e-mail address of receiver. Earlier used addresses can be selected in drop-down menu;
- Subject – message subject. Earlier used topics can be selected in drop-down menu;
- Text – message text;

- Format – format of report attached to mail. One of the available export formats and also own format of FastReport (FR3) ready report may be selected;
- Advanced export settings – on this option enabling after clicking on “OK” the dialogue box for selected export format setting appears. Otherwise default export parameters will be used.

Export via e-mail peculiarities: only plain authentication on SMTP servers is supported. If authentication is not required, it is not necessary to fill “Login” and “Password” fields in settings.

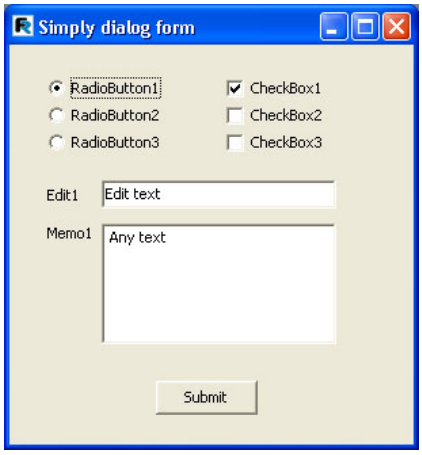
Other clients

The FastReport server gives you wide opportunities of choosing the client program due to using standard HTTP protocol. You can use any HTTP-compatible client such as web-browser that supports JavaScript, tables, and frames.

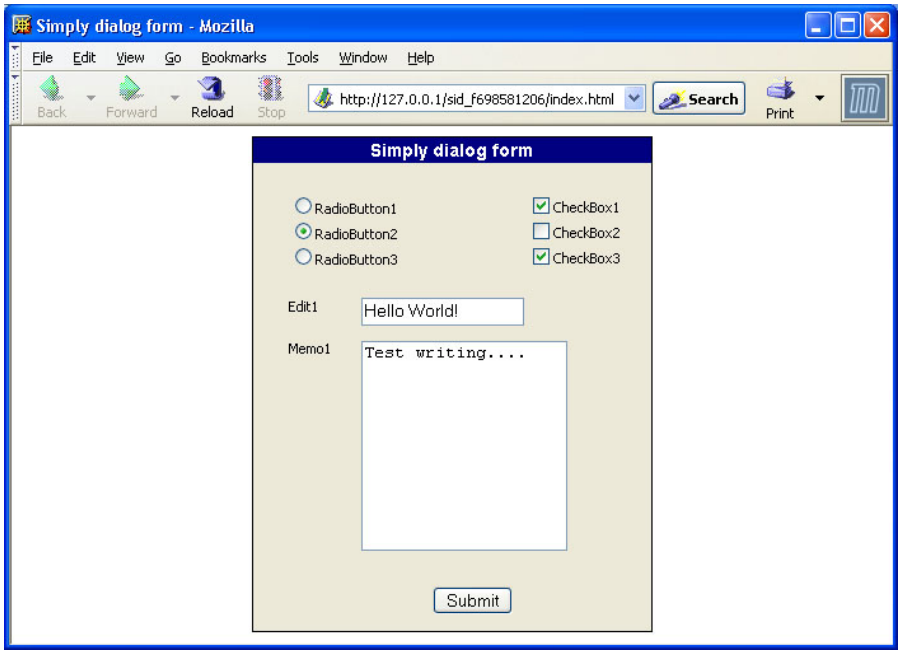


When using dialogue forms in your reports, the server will convert them to web-forms and pass them to a client. Client should fill in the form and return it to the server.

This is how the dialogue form looks when running a report in a simple (non-client-server) application:



The same form appears in the Mozilla web-browser, when running a report in the client-server application.



Important security issues

1. When using a report server on Microsoft Windows platform over the Internet, it is recommended to use a firewall between server and internet network.

2. It is obligatory to use the authentication of the client program.

3. Use the “allow/deny IP” function in local network.

4. If you have any gateways to Internet in local network, then include IP addresses of these gateways to the “deny” list of the report server (section 3.9).

5. Do not pass parameters to database connection from client if you use reports with internal database components.

6. In reports folder, store only those reports, which you use in your application.

7. Do not store any private documents in the HTTP root folder.

1. If you find any bugs in security system of the FastReport Server, send a note to the developers of the product.

Developers' contact information

If you have any suggestions concerning improvement and development of FastReport Enterprise, please contact us:

e-mail:	support@fast-report.com
news:	http://fast-report.com/en/support/newsgroups.php
web site:	http://www.fast-report.com