www.asix.com.pl

AsRaport –

asix system reporting functionality
based on the Microsoft Reporting Services

Doc. No. ENP6100
Version: 22-10-2010

ASKOM

ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,
tel. +48 32 3018100, fax +48 32 3018101,

http://www.askom.com.pl, e-mail: office@askom.com.pl

1 Introduction

The 6th version of **asix** system has been enriched in the reporting system based on Microsoft Reporting Services.

Microsoft® SQL Server™ 2008 Reporting Services is a comprehensive server platform designed to meet a broad range of enterprise-wide reporting-related needs. The Reporting Services (in fact a component of the SQL Server 2008 database) enable report creation from various data sources; report environment management (planning the moment of report generation/subscribing/access control), as well as delivering reports to the users in the format and the way most convenient for them. The delivery methods include e-mail subscription and embedding the reports in business applications and/or Web portals.

In order **to be able to cooperate with the Reporting Services reporting platform**, the **asix** has been extended by the following components:

- Askom.Data.Host program
- AsRaport query editor
- AsixConnect database (run on a Microsoft SQL server)
- independent alarm archiver to store alarms in Microsoft SQL database.

The **AsixConnect database** is a key element. Its stored procedures with the use of Askom.Data.Host share process data from **asix** applications to the reporting environment. The **AsRaport** query editor plays a vital role in the report designing phase as the tool used to work out necessary SQL queries. Data from **asix** applications retrieved by the queries are used by the **Report Builder** or the **Report Designer** client applications of the Reporting Services environment to generate final reports. The AsRaport program is a visual editor so that SQL queries may be constructed even by users not familiar with syntax of the Structured Query Language.

Report Builder is a tool that enables the workers without programming skills to construct report definitions for individual report generation ad hoc (on demand) by end users. The Reporting Services environment is similar to the Microsoft Office package and may be intuitively operated especially by persons acquainted with the Office Excel or Access program.

Report Designer is a very flexible and productive reporting environment for programmers and advanced report designers. It is a component of SQL Server Business Intelligence Development Studio of Business Intelligence programming environment based on the well known Microsoft Visual Studio programming interface. Simple reports may be quickly and easily developed with the help of the Report Wizard, more sophisticated ones may be worked out in visual development environment of the Report Designer platform.

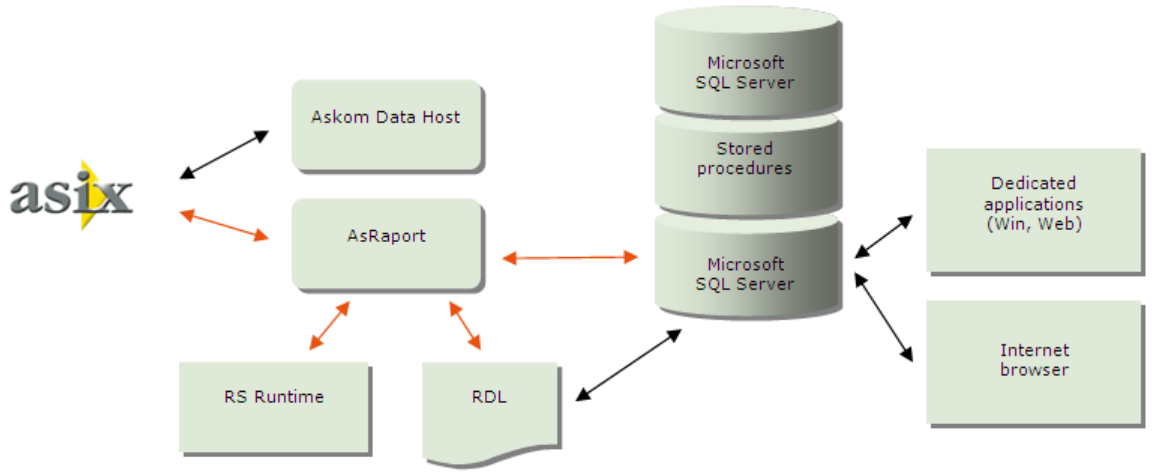

*Fig. Architecture of the interface between the **asix** system and the Reporting Services environment*

Users who develop reports that are going to retrieve some data from **asix** system applications simultaneously works in AsRaport interface and Report Builder/Designer interface.

The AsRaport and Askom.Data.Host.exe programs must be first configured and the AsixConnect database must be created to get the **asix** system ready for cooperation with the Reporting Services environment.

Design works on each new report form starts from defining syntax of SQL queries posted to database of stored procedures in order to retrieve source data, on which the reports are to be based; the AsRaport query editor is used to that end. Next, report layout must be specified in the Report Builder/Designer program - it demands a layout template with pre-defined data sections (tables, arrays, charts) to be selected, selected data elements to be placed in the project view, and suitable filtering criteria to retrieve source data for the reports to be defined. Report Builder/Designer may also be used to add new computational fields to the report form, almost arbitrarily format appearance of the report, and to view/print/Web-publish the report. A completed report form is stored on the Reporting Services central report server. The server shares the stored report forms throughout the entire enterprise using various methods, including making them available to Internet browsers.

More information on generation of reports in SQL Server 2008 Reporting Services may be found on the Microsoft Webpage.

1.1 Software requirements

SQL Server 2008 Service Pack 1 must be installed for cooperation of the **asix** 6 system with the Microsoft Reporting Services environment.

2 Access to asix system data

To use Microsoft Reporting Services, first of all install MS SQL Server 2008 on the computer, on which the report forms are to be developed. The procedure to configure access to application process data that are to be used as report source data includes the following steps (discussed in subsequent chapters below):

1. AsixConnect database creation.
2. AsRaport environment configuration.
3. Running Askom.Data.Host.exe.

2.1 The AsixConnect database

AsixConnect database stored procedures with use of the Askom.Data.Host program retrieve process data from **asix** applications and make them available to the reporting environment.

Use the Architect program to create AsixConnect database. It is the first step of the procedure to get the **asix** system ready for reporting based on Microsoft Reporting Services.

ATTENTION: AsixConnect database is indispensable if **asix** application process data are to be used by the reporting system.

To create AsixConnect database with access process data of an asix application:

1. Run the Architect program and open in that program the **asix** application configuration file.
2. Run the AsRaport set-up program:

   Architect > *Application* menu> *Configure Reporting System AsRaport* > *AsxConnect database* tab

3. Select MS SQL server name for the AsixConnect database, select the to-be-used user authentication method, and click the ***Create Database*** button.

Three options used to select method to authenticate AsixConnect database users include:

1. Log-in as the current Windows user – use operating system user credentials

2. Log-in as a user of MS SQL Server named '_asix_internal' – such user was automatically created by the **asix** package installer if only it was running on a computer with previously installed MS SQL server

3. Log-in as the specified user of MS SQL Server – enter credentials of an user defined via the Architect program in the AsRaport set-up program prior to creating the AsixConnect database.

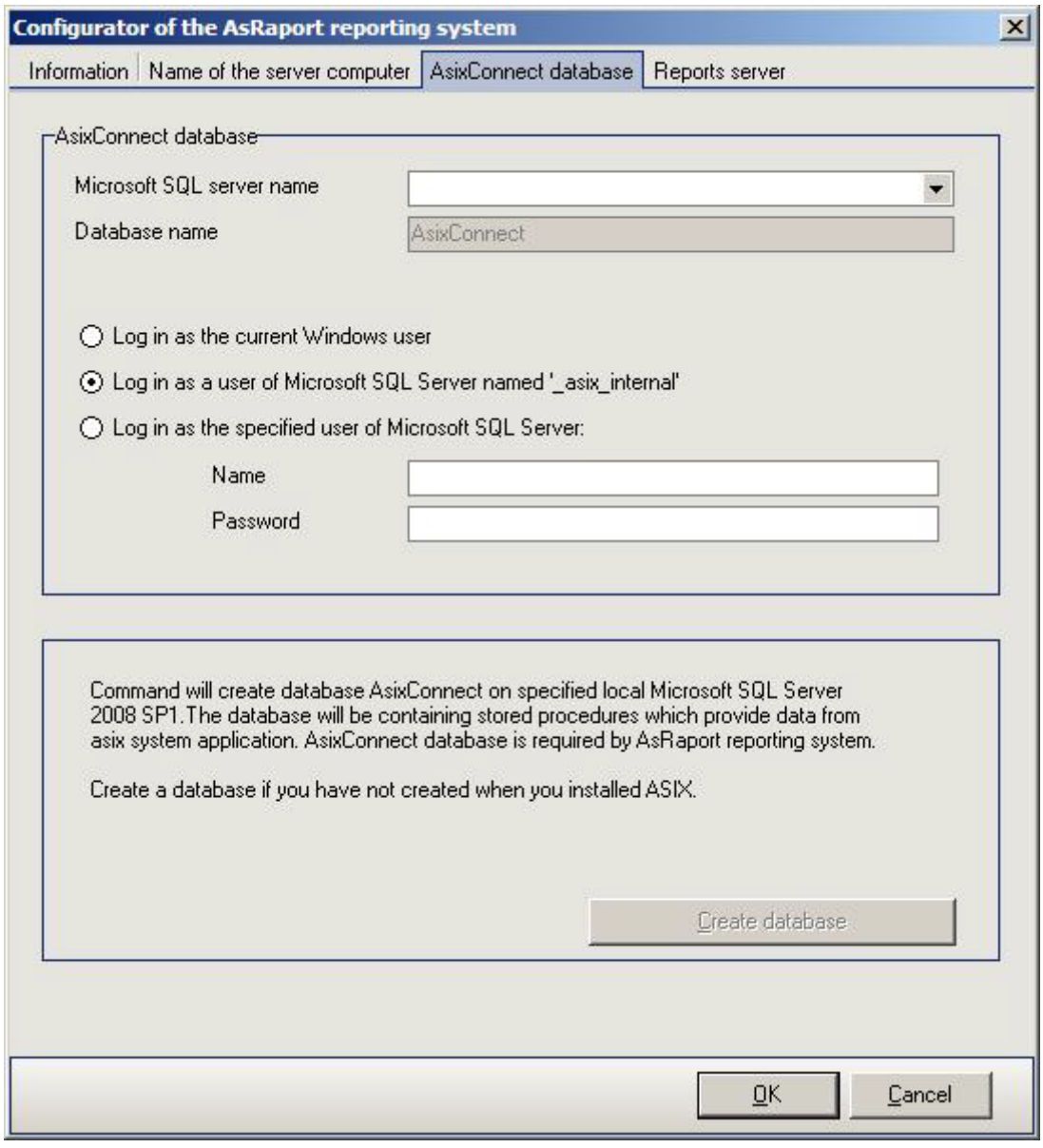

Fig.: 'Configurator of the AsRaport reporting system' window – AsixConnect database.

2.2 The AsRaport query editor

Configuration parameters of the AsRaport query editor include:

- **asix** application server computer;
- name of the report server.

The following application parameters are retrieved from the indicated **asix application server computer**:

- location of the variable definition database,
- network name of the alarm server,
- name of the MS SQL server and alarm database name.

Give attention to...

Lcation of the variable definition database is declared in:
Architect > *Databases* > *Variable Definition Base* > *Format* tab

Network name of the alarm server is declared in:
Architect > *Fields and Computers* > *Alarms system* > *Alarms/Network Name* tab

Name of the SQL server and alarm database name are declared in:
Architect > *Fields and Computers* > *Alarms system* > *Archive /Microsoft SQL Server* tab

To declare asix application server computer:

1. Run the Architect program and open in that program the **asix** application configuration file.
2. Run the AsRaport set-up program:

   Architect > *Application* menu > *Configure reporting system AsRaport* > *Name of the server computer* tab

3. Drop down list of server names, select the name of computer and click ***OK***.

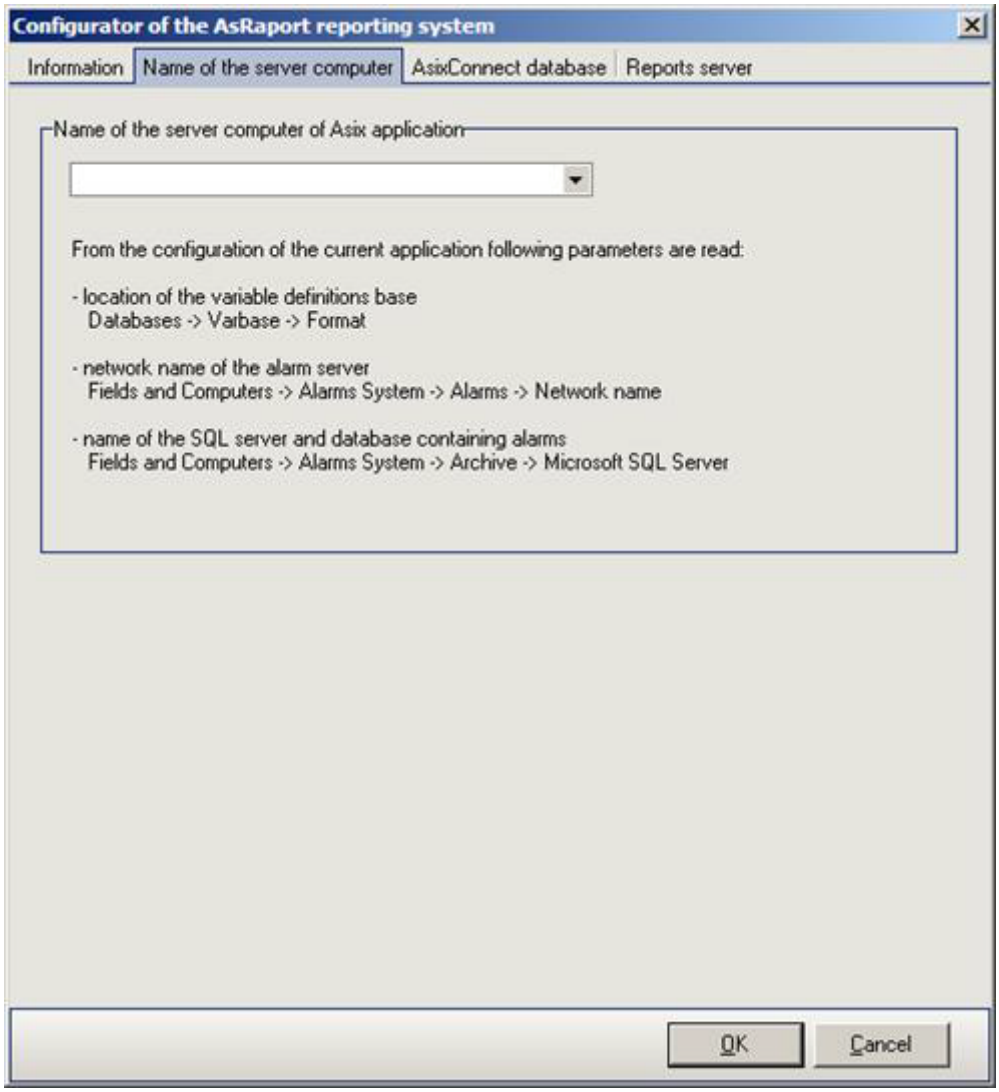

Fig.: 'Configurator of the AsRaport reporting system' window – Name of the server computer.

Declaration of MS Reporting Services report server is necessary to be able to open reports generated within the MS Reporting Services environment directly from the **asix** system application level (using the AsRapView module of the **asix** system).

To declare report server:

1. Run the Architect program and open in that program the **asix** application configuration file.
2. Run the AsRaport set-up program:

   Architect > *Application* menu> *Configure reporting system AsRaport* > *Reports server* tab

3. Declare report server address and name of the folder within the server, as well as select the method to be used to authenticate users.

4. Click ***OK***.

Three methods to authenticate report server users include:

1. Log-in as the current Windows user – use operating system user credentials.

2. Log-in as the 'AskomInternal' Windows user automatically created by the **asix** package installer.

3. Log-in as the specified Windows user – enter credentials of an user defined via the Architect program in the AsRaport set-up program

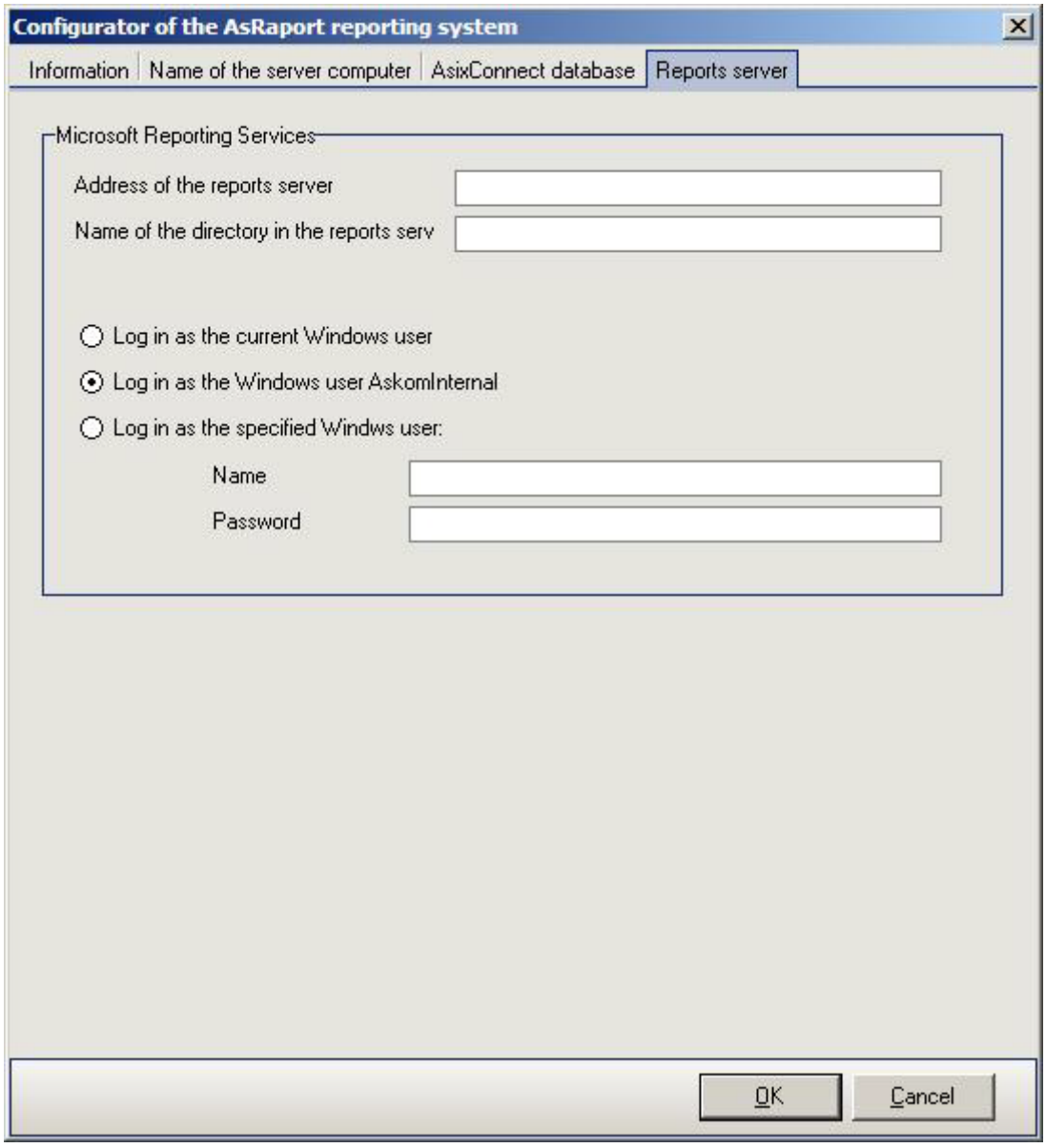

Fig.: 'Configurator of the AsRaport reporting system' window – Report server.

2.3 The Askom.Data.Host.exe program

The Askom.Data.Host program must be running if retrieval of process data from **asix** application is to be possible. By default the **asix** package installer puts the program within the package C:\Program Files\Askom**Asix** main folder.

3 AsixConnect SQL queries built in AsRaport

The AsRaport query editor is run by the AsRaport.exe executable put by default by the **asix** package installer into the package C:\Program Files\Askom**Asix** main folder.

The ***Tools*** and **View** tabs at the top part of the main window of the editor display collections of tools. The remaining part of the window is consumed by ***Query1***, ***Query2***,... tabs used to define various queries to the database of process value archive and to the database of variable definitions.

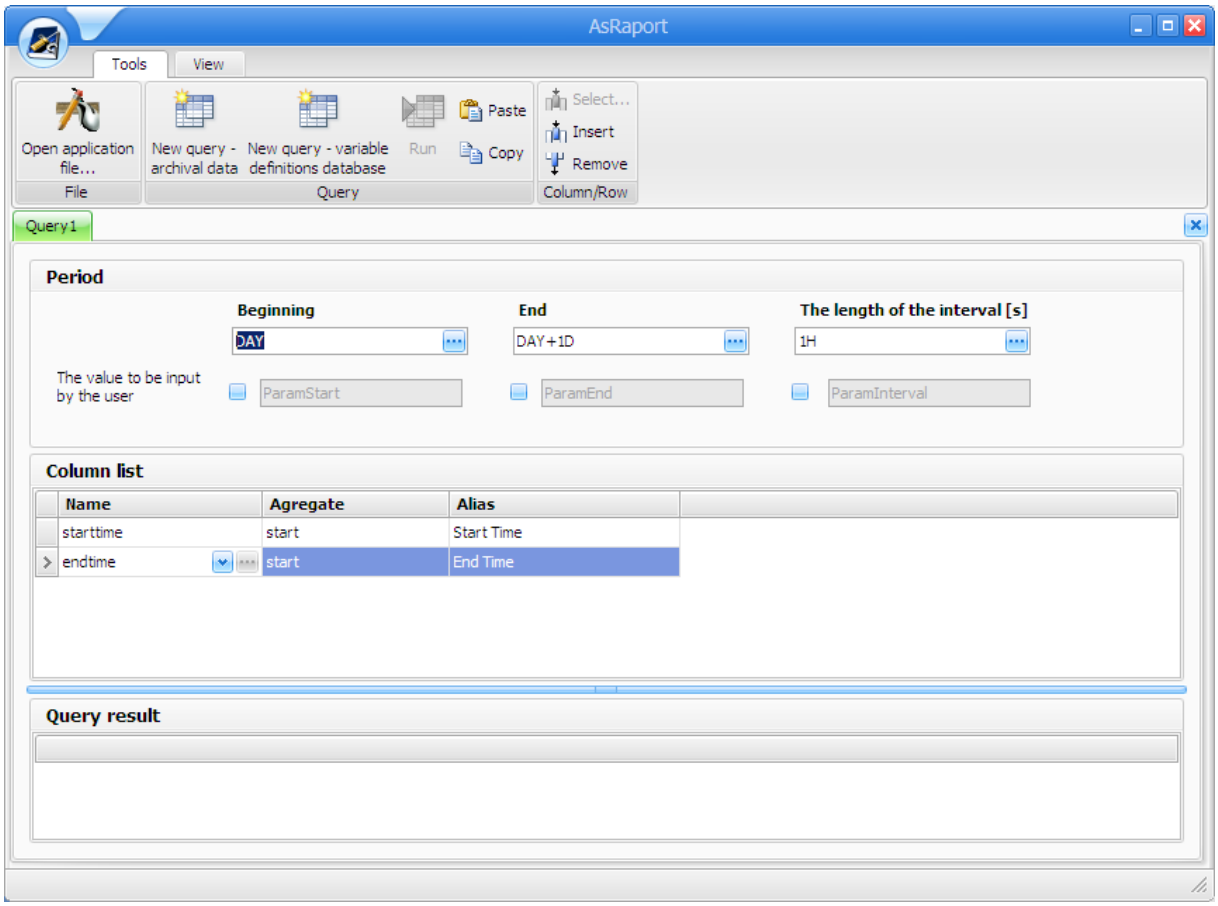

Fig: AsRaport – main window – the tab for a query to the database of process value archive.

The ***Tools*** tab includes:

Open application file – open the **asix** application configuration file (XML file).

New query – archival data – add a tab for a new query to the database of process value archive.

New query – variable definitions database – add a tab for a new query to the database of variable definitions.

Run – display results of the so-far defined query.

Copy – copy query defined in the AsRaport window to the Windows Clipboard (to paste it later at the report development stage to the Report Builder/Designer query definition window).

Paste – paste a query from the Windows Clipboard to the AsRaport window.

Select – select process variable.

Insert – insert a new row to list of columns (in case of a query to the database of process value archive) or a new row to list of attributes/variables (in case of a query to the database of variable definitions).

Remove – remove the highlighted row from list of columns (in case of a query to the database of process value archive) or list of attributes/variables (in case of a query to the database of variable definitions).

Tools of the ***View*** tab are used to select styles and colors used in the AsRaport window to control its appearance.

Each ***Query*** tab opened by the ***New query – archival data*** command displays the following elements:

Specifications of the period to be covered by the query:

Beginning – beginning of the period

End – end of the period

The length of the interval [s] – interval between subsequently retrieved process variable values (the entire query period is divided into constant-length time intervals).

Column list – list of fields (report columns) to be presented in the report. Click in the new row in the ***Name*** column, then click the  button to drop down selection list of all pre-defined field names:

starttime – interval start time

endtime – interval end time

shiftnumber – operator shift number. Three shifts have been pre-defined: 6 am–2 pm (the 1st shift); 2 pm–10 pm (the 2nd shift); and 10 pm–6 am (the 3rd shift).

shiftdate – date when the given shift commenced.

For each query field an **Agregat** (aggregate: average0, average, delta, end, max, min, range, start, Total, quality_bad, quality_good, quality_uncertain) and an **Alias** may be defined. The aggregate is calculated for each interval using all archived data belonging to that interval.

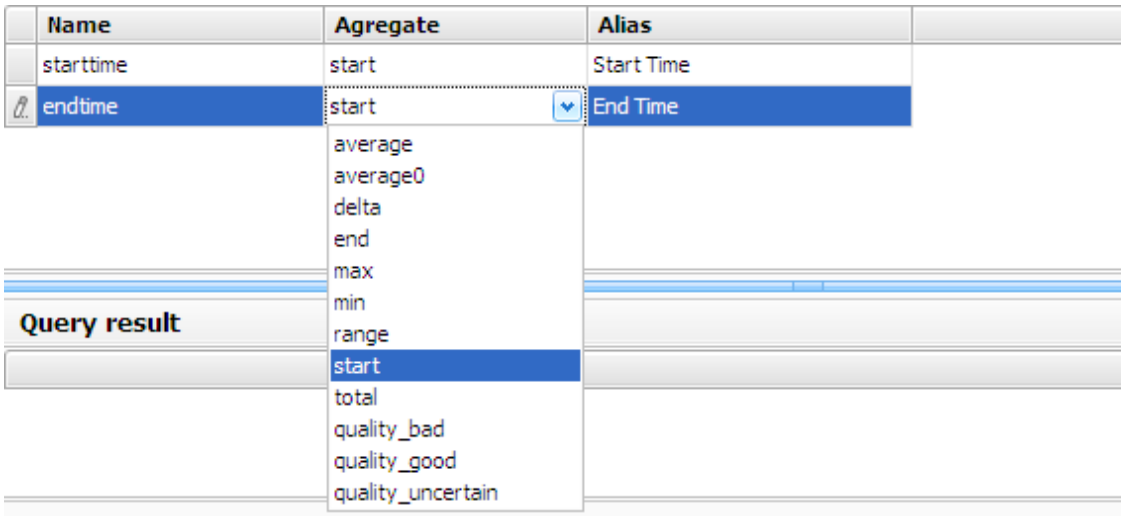

Fig. The query field declaration for a query to the database of process value archive.

Query result – result of a query posted to the database of process value archive.

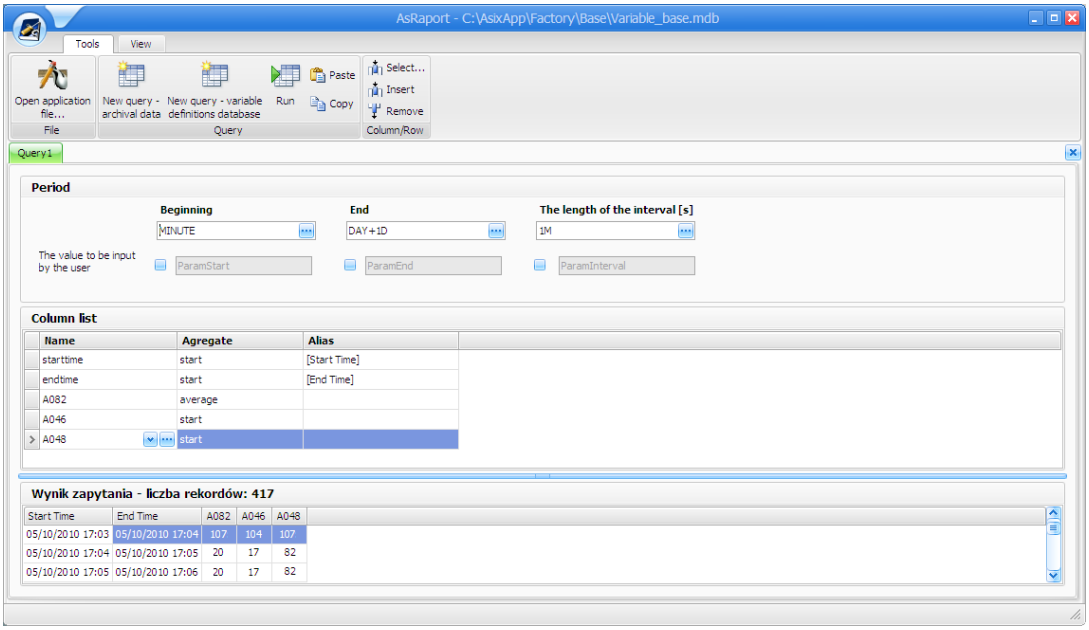

Fig. The tab for a query to the database of process value archive.

Each ***Query*** opened by the ***New query - variable definitions database*** command displays the following elements:

Attribute list – list of attributes of variables to be presented in the report.

Variable list – list of names of variables to be presented in the report.

Query result – results of the query posted to the database of variable definitions.

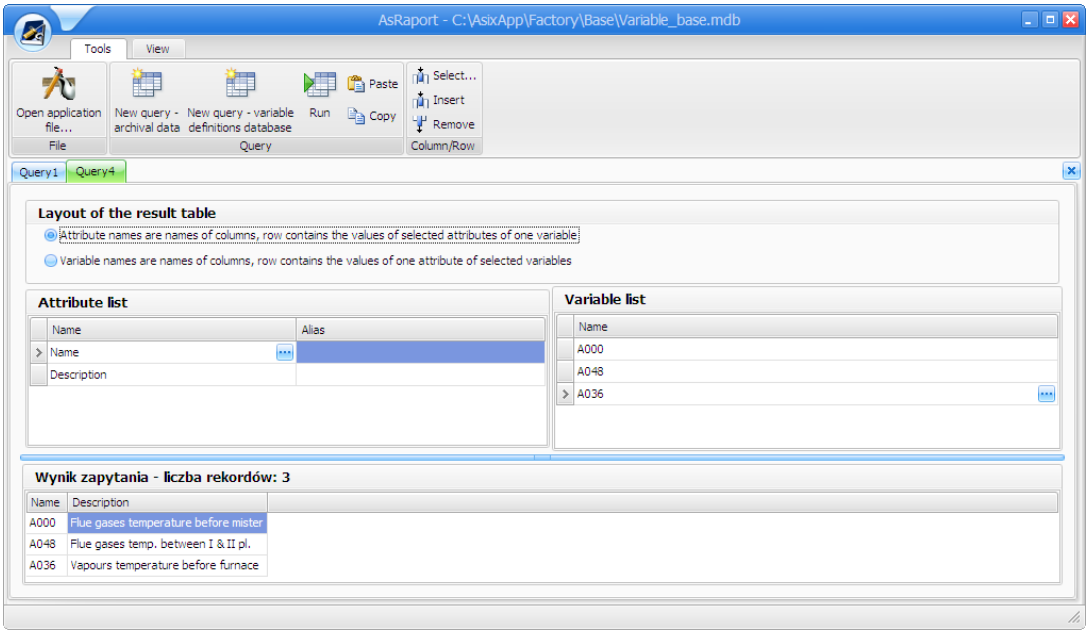

Fig: AsRaport – main window – the tab for a query to the database of variable definitions.

3.1 Data retrieval – AsixConnect database stored procedures

The AsRaport query visual editor may assist users who are not familiar with syntax of the SQL language. However, some users may prefer to construct the queries – including queries to alarm archives – on their own. Time-related parameters of the query (periodStart, periodEnd, resampleInterval, periodLen) may be specified in local time, or in the OPC time format (see the *Date/time/period OPC format* section below). Retrieval of archived process data, data from variable definition database, alarms from the SQL archive, and alarms from file archives is discussed in subsequent sections below.

3.1.1 Date/time/period OPC format

The OPC time format

The syntax:
keyword +/- offset number +/- offset number …
'keyword': NOW, MINUTE, HOUR, DAY, WEEK, MONTH, YEAR
'offset': S, M, H, D, W, MO, Y

Examples: DAY-1D, YEAR+1MO

The OPC duration format

The syntax:
+/- offset number +/- offset number …
It doesn't contain 'keyword'
'offset': S, M, H, D, W, MO, Y
Examples: 5M, 1H, 12H, 1MO, 1D+12h

3.1.2 Retrieval of archived data

ReadProcessed periodStart, periodEnd, resampleInterval, column1, column2, column3, …

Time period specification:
periodStart, periodEnd: date, for example: '2009-1-1 0:0:0'

resampleInterval: number of seconds

Columns with variable values:

- Variable name: 'A000'
- As before + alias: 'A000 as [Flue gases temp.]'
- Aggregate and variable name: 'avg(A000)'
- As before + alias: 'avg(A000) as [Averaged flue gases temp.]'

Time columns:

- stamp time of interval start – ‘starttime’
- stamp time of interval end – ‘endtime’
- date of shift start – ‘shiftdate’
- number of shift – ‘shiftnumber’

3.1.3 Retrieval of data from variable definition database

Retrieve data from the database of variable definitions:

- ReadAttributes variableNames, column1, column2, column3, …
- variableNames: the list of variable names, for example.
  ‘A000, A004, A008’

Retrieve and transpose data from the database of variable definitions:

- ReadAttributesTranspose attributeNames, column1, column2, column3, …
- attributeNames: the list of variable attributes, for example.
  ‘Name, Description, Unit’

3.1.4 Retrieval of historical alarms (SQL archive)

Retrieve historical alarms from the SQL archive:

- ReadProcessedHistoricalAlarms resourceName, periodStart, periodEnd, alarmIdPattern, alarmTextPattern, alarmTextLang
- only alarms that were initiated within the specified period will be retrieved
- alarm start/end/confirmation date, ID, text and duration will be retrieved
- periodStart, periodEnd – period in local time/OPC time format
- alarmIdPattern – empty, alarm ID, alarm template or list of IDs, e.g.

  1,2,3,
  1-3
- alarmTextPattern – alarm text or its template e.g.

  <text>
- alarmTextLang – alarm language ID (‘pl’ for Polish)

Alarm activity periods

- ReadProcessedHistoricalAlarms_ActivePeriods resourceName, periodStart, periodEnd, alarmIdPattern, alarmTextPattern, alarmTextLang
- only alarms that were active within the specified period will be retrieved
- each alarm start/end date earlier/later than the query period start/end date will be replaced with the query period start/end date, respectively.

Sum of activity period lenghts

- ReadProcessedHistoricalAlarms_ActivePeriodTotal resourceName, periodStart, periodEnd, alarmId, alarmTextPattern, alarmTextLang

- length of the period during which the alarm was active within the specified query period will be returned.

3.1.5 Retrieval of historical alarms (file archive)

- ReadHistoricalAlarmsOpc periodStart, periodLen, alarmIdRange, alarmTextMask, alarmType, alarmStatus, maxNumberOfAlarms
- Alarm start/end dates will be returned as separate rows.

4 Designing reports in Report Builder

Basic steps of the procedure that must be performed to develop a report utilizing some values of process variables/variable definitions retrieved from **asix** system applications are described below using an example of the "Factory" demo application delivered with the **asix** package. Reporting Services from the SQL Server 2008 Service Pack 1 version Express have been used. For detailed instructions see documentation of the Report Builder program.

Main window of the Report Builder program in version 2 is shown in the figure below.

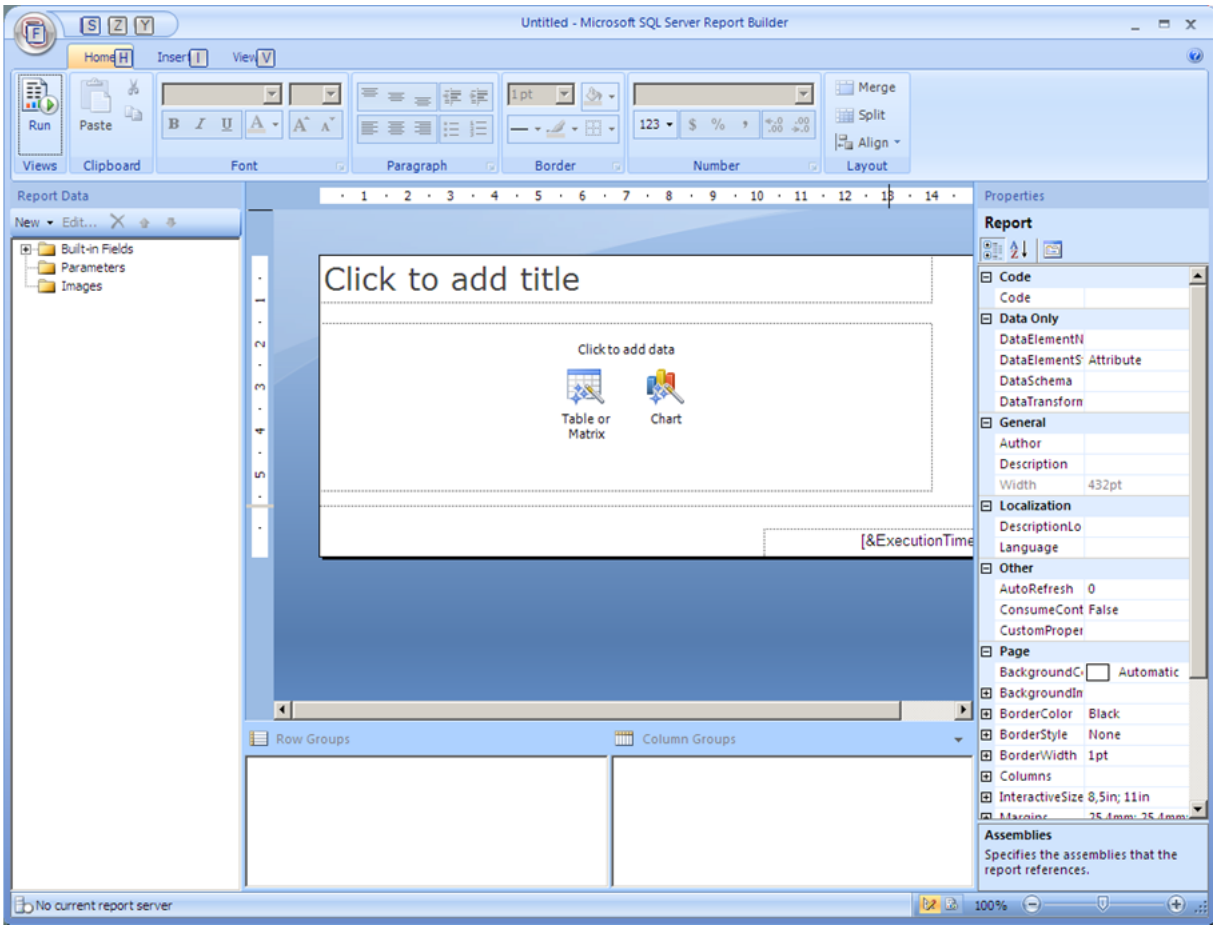

Fig. Report Builder - main window.

Form used to define new reports contains by default report contents, page footnote, and the following elements:

- text box with the "click to add title" prompt
- link to the table/matrix wizard and link to the chart wizard
- the embedded [&ExecutionTime] field in the footnote (to add report execution date/time to each page of the report at page bottom).

4.1 Create a new report

Click the **Start** button and select the **Programy/Microsoft SQL Server 2008 Report Builder/Report Builder 2.0** option to create a new report. The displayed Report Builder window will show new report in the project view.

4.2 Declare data source/layout/format (wizard)

The first step is to select source of the to-be-reported data and to indicate the format, in which the reported data are to be arranged. Data presented in examples below are arranged in tables.

Table may be added to the report using the table wizard. The wizard helps to define links to data sources, to create query specifying which data are to be reported, to organize data in some groups, and to add summary rows (in which data contained within the group in question are aggregated).

1. Click the ***Design*** button to switch to the project view.

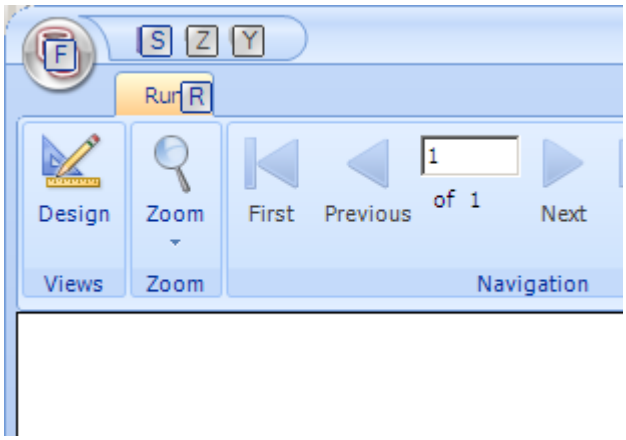

Fig. Design button to switch to the project view.

2. Click the ***Table or Matrix*** icon within the ***Click to add title*** area.

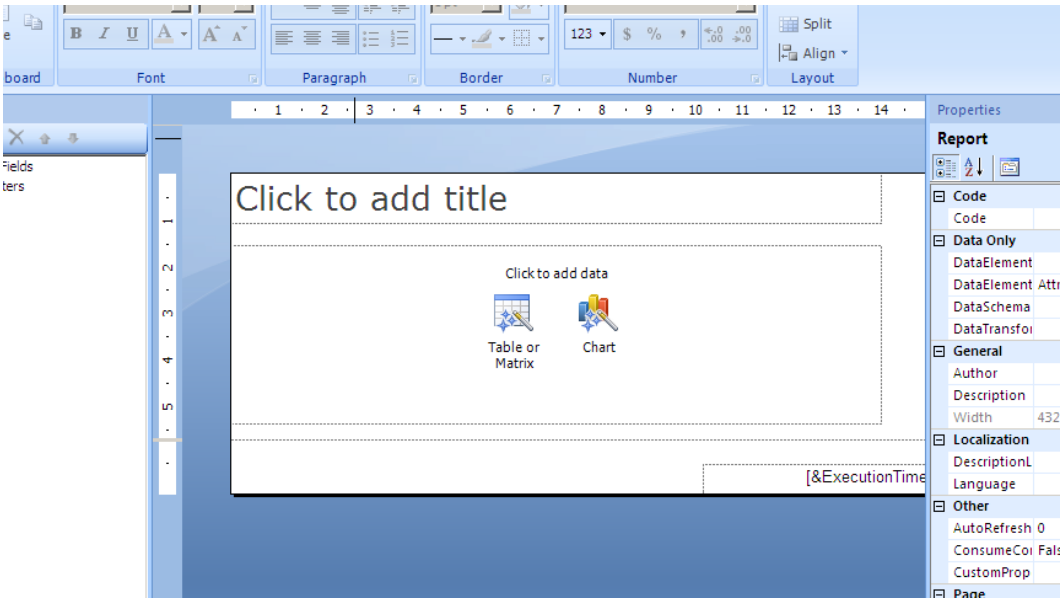

Fig. 'Click to add title' area.

The ***'Choose a connection to a data source'*** window appears.

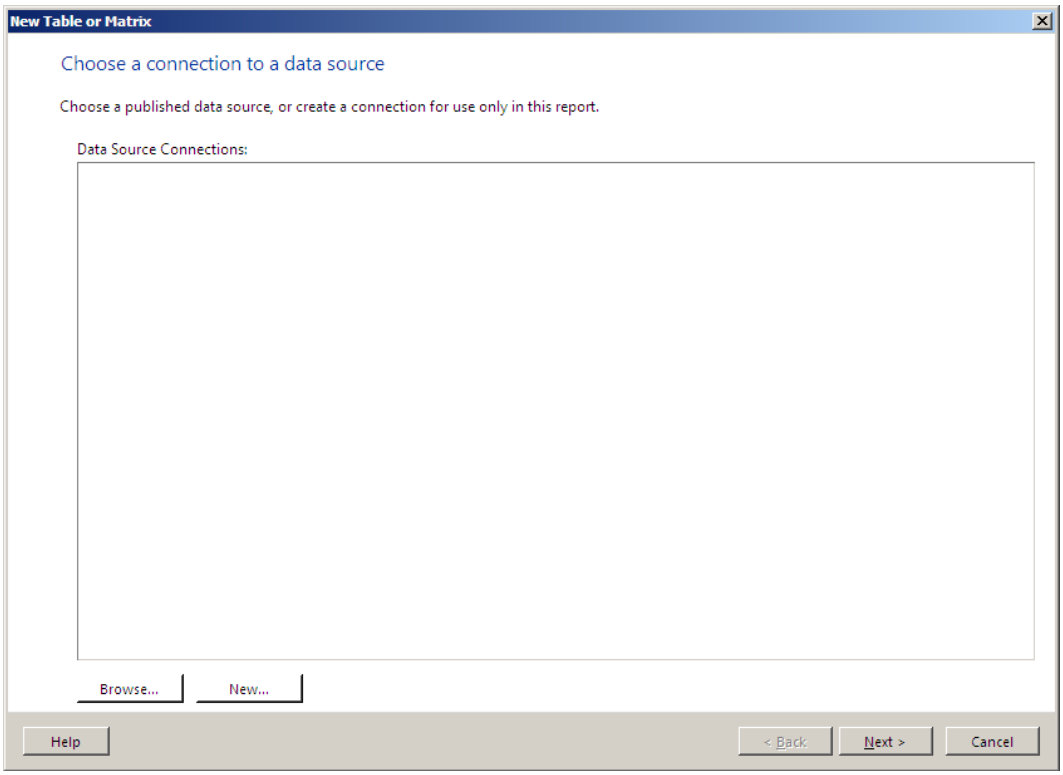

Fig. 'Choose a connection to a data source' window.

List of data sources appears, if some folder with shared source data has been defined on the report server. Otherwise such folder (data source) must be defined.

3. Click the ***New...*** button to add a data source from the report level.

4. Select the '**Microsoft SQL Server**' connection type, then click the ***Build*** button to define connection properties.

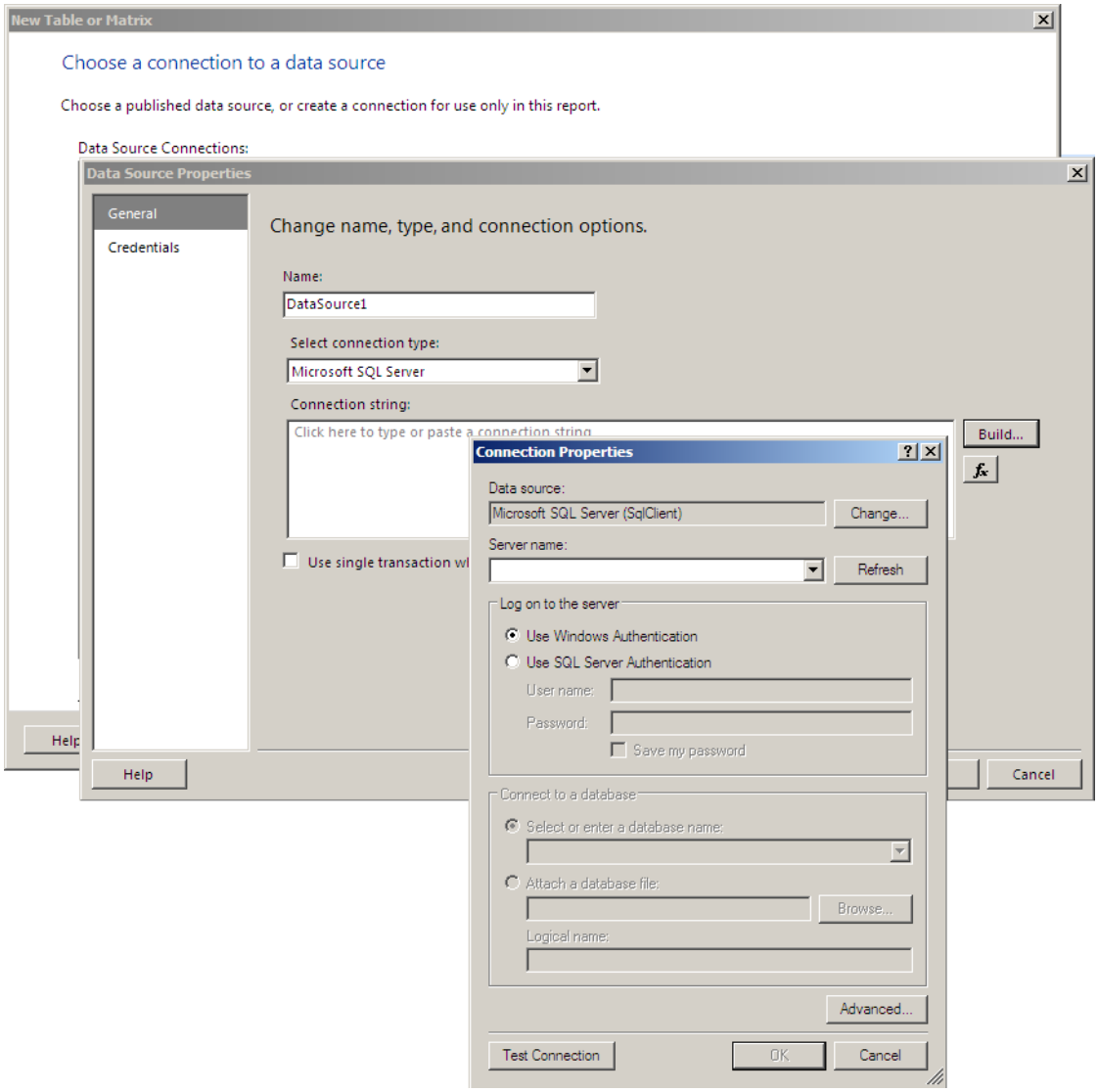

Fig. 'Connection properties' window.

5. Enter path to the AsixConnect database.

6. Click ***OK*** to close the *'Connection Properties'* window, another ***OK*** to close the *'Data Source Properties'* window, and ***Next*** to proceed with the wizard.

ATTENTION: The above procedure may be used to define data sources from the report level. However, in order to be able to freely modify reports stored on a report server, the data source should be declared on that server, too. Such sources are automatically displayed on the *'Choose a connection to a data source'* wizard screen. This may be done (for example) using the Report Manager tool, which may be invoked as follows:

Menu *Start* > *Microsoft SQL Server 2008* > *Configuration Tools* > *Reporting Services Configuration Manager* > *Report Manager URL* > *URLs*

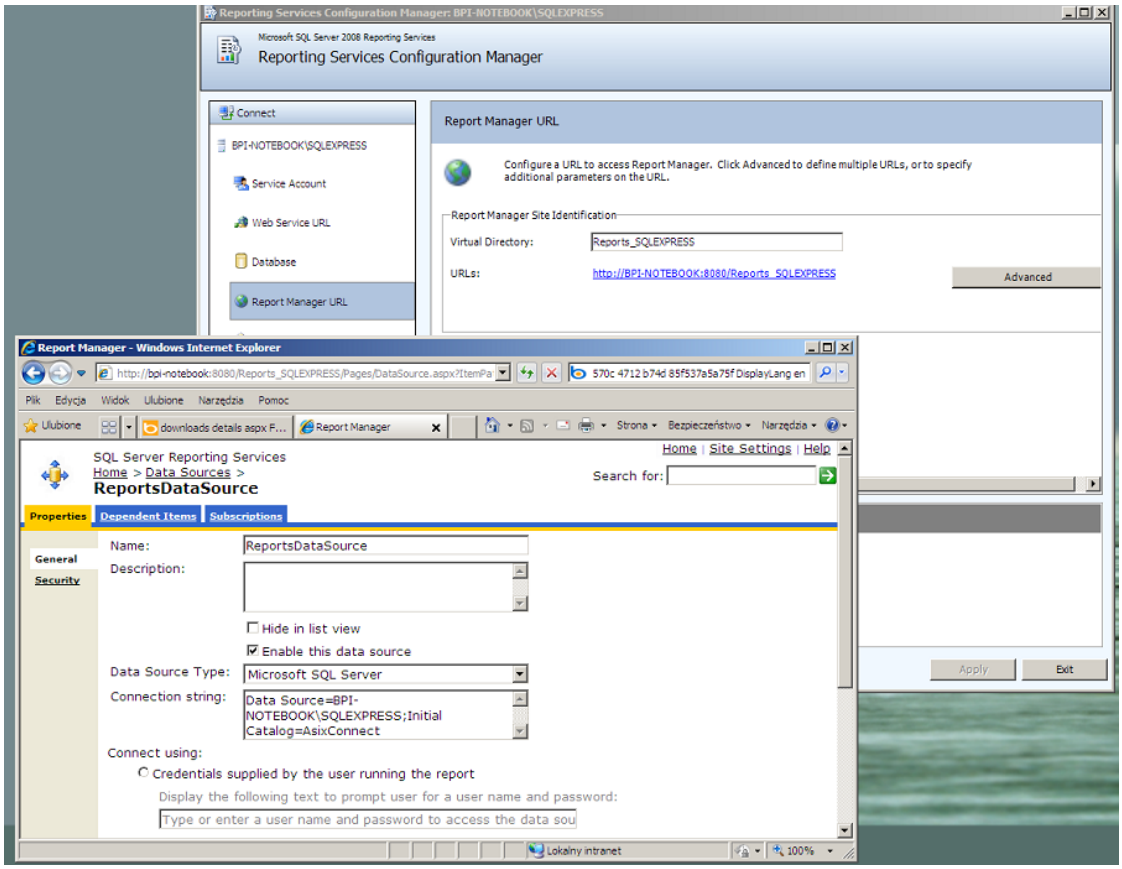

Fig. Report Manager - report data source declaration.

Once data source is declared, declare a query to retrieve the to-be-reported data. Proceed with the table wizard:

7. Once path to the AsixConnect database is declared, the table wizard will display the ***'Design a query'*** window.
8. Provided that the query has already been created in the AsRaport editor, copy it from the editor to the Windows Clipboard, navigate to the ***'Design a query'*** screen of the Report Builder wizard, select the ***Edit As Text*** option and paste the Clipboard contents. Click the button

! to display query results.

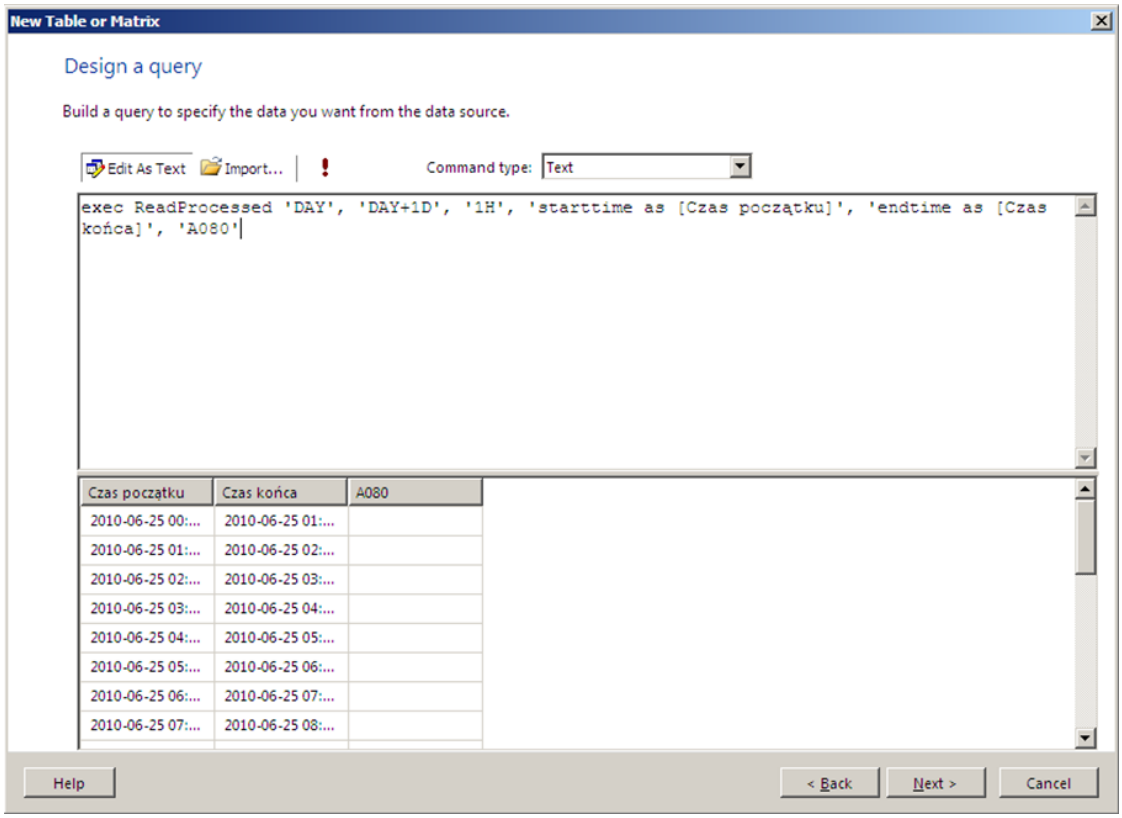

Fig. Report Bulider – table wizard – 'Design a query' window.

9. Click ***Next***.

10. Arrange the fields grouping them into rows, columns or rows/columns and click ***Next***.

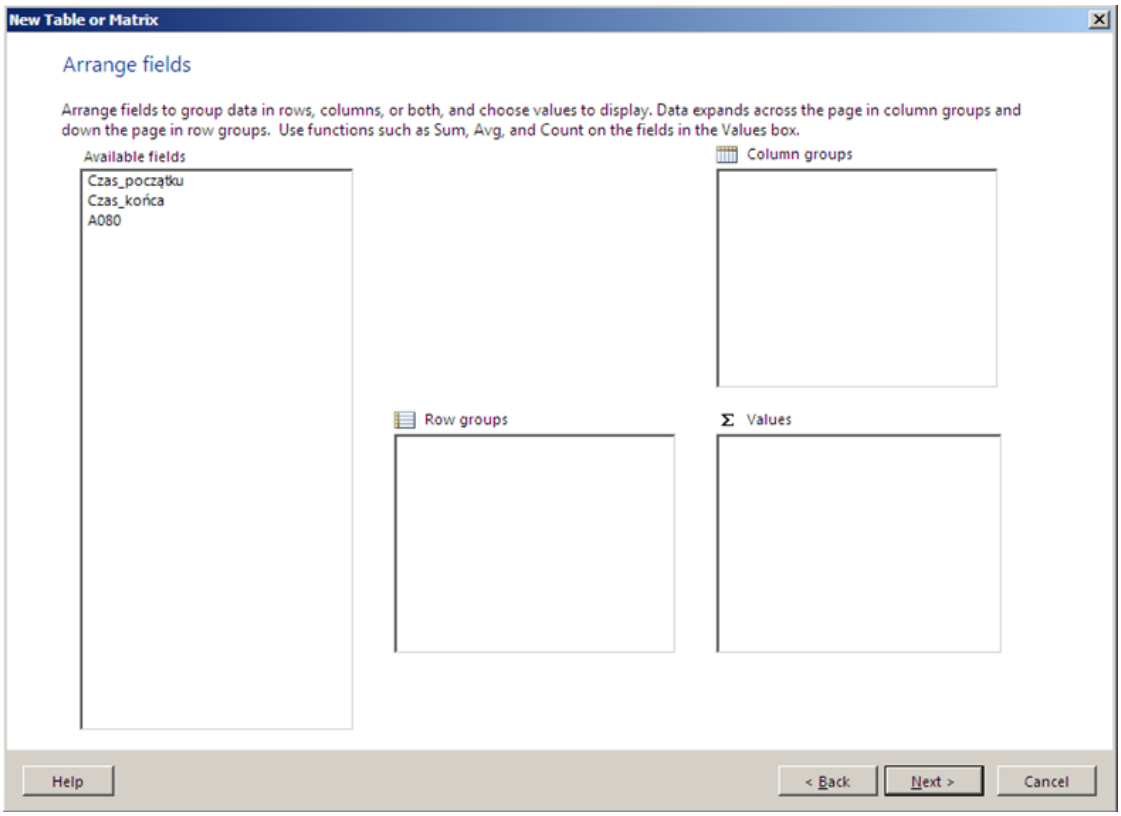

Fig. Report Bulider – table wizard – 'Arrange fields'.

11. Select a layout for the table and click ***Next***.

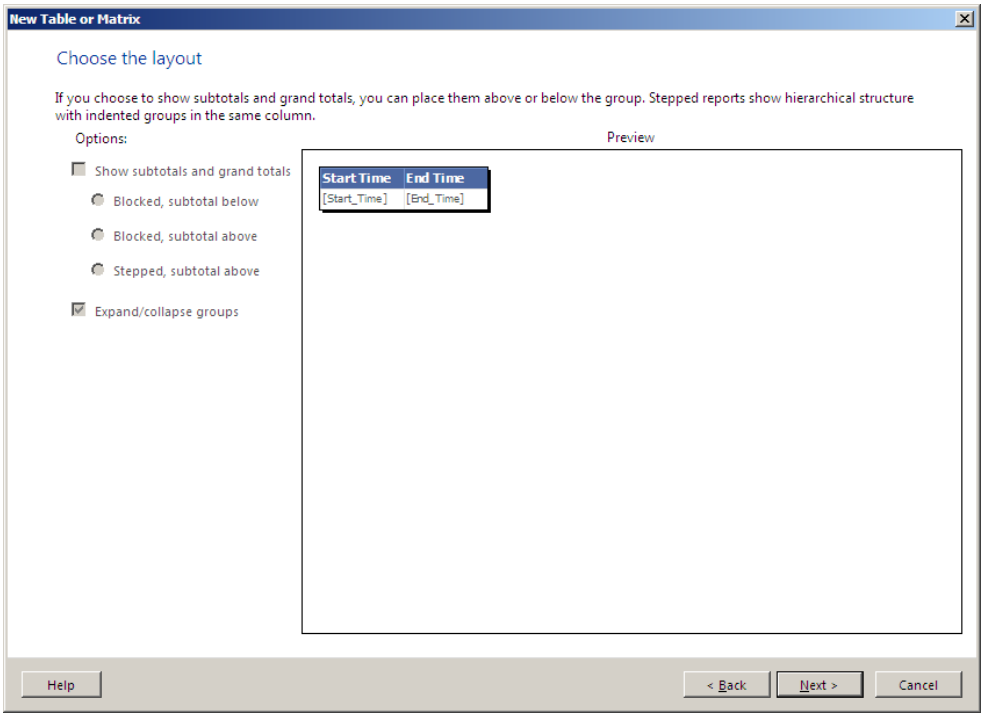

Fig. Report Bulider – table wizard – 'Choose the layout' window.

12. Select a style for the table and click ***Finish***.

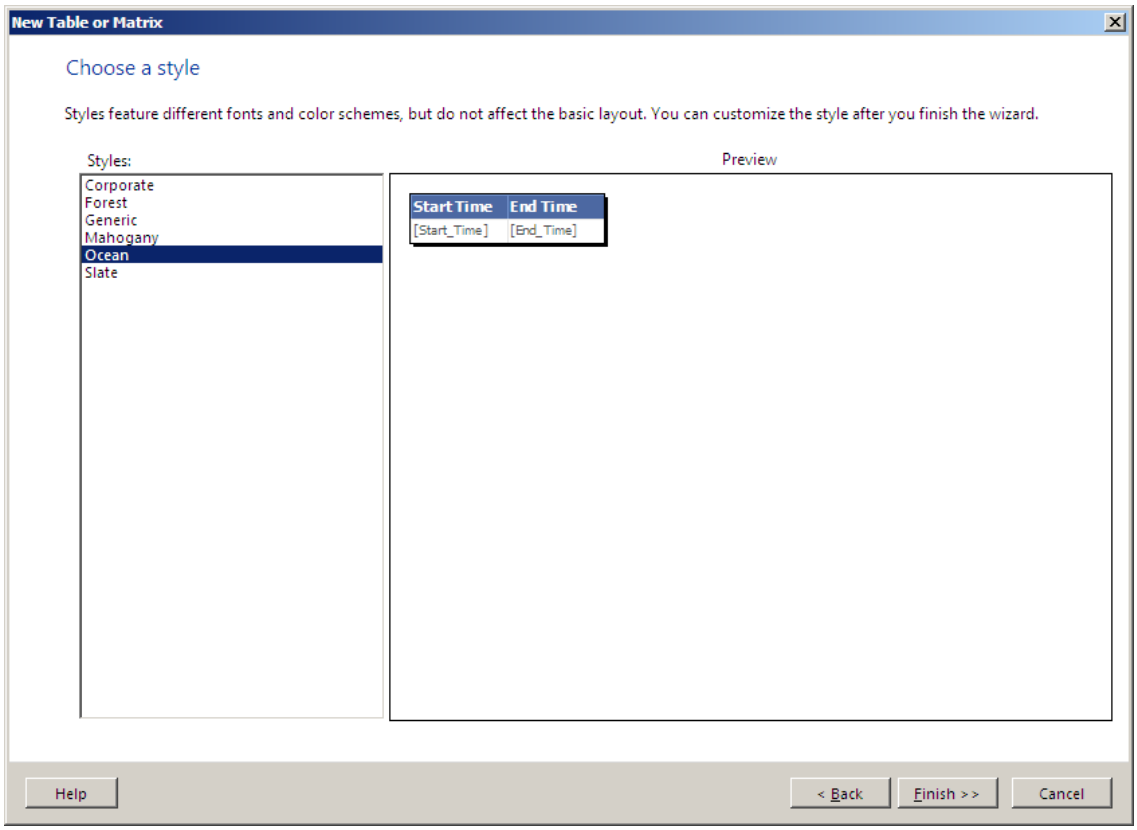

Fig. Report Bulider – table wizard – 'Choose a style' window.

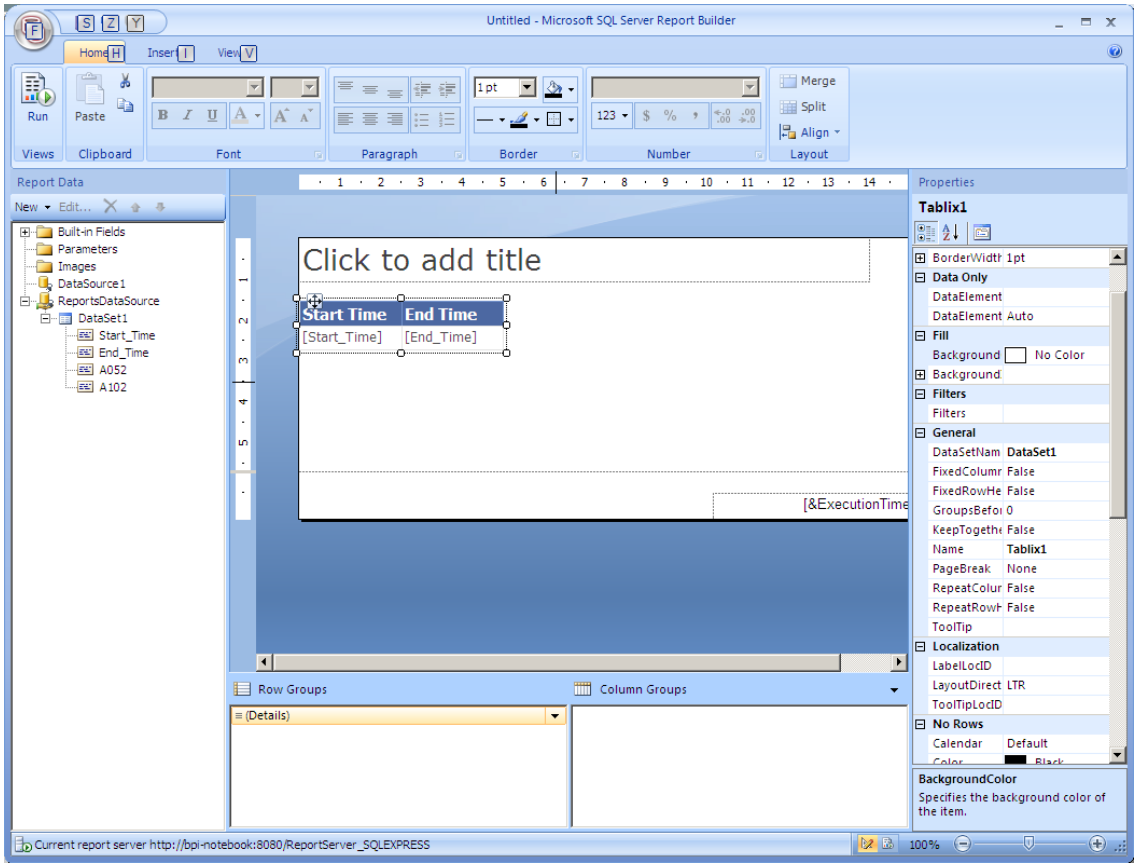

Fig. Report in 'Design' view.

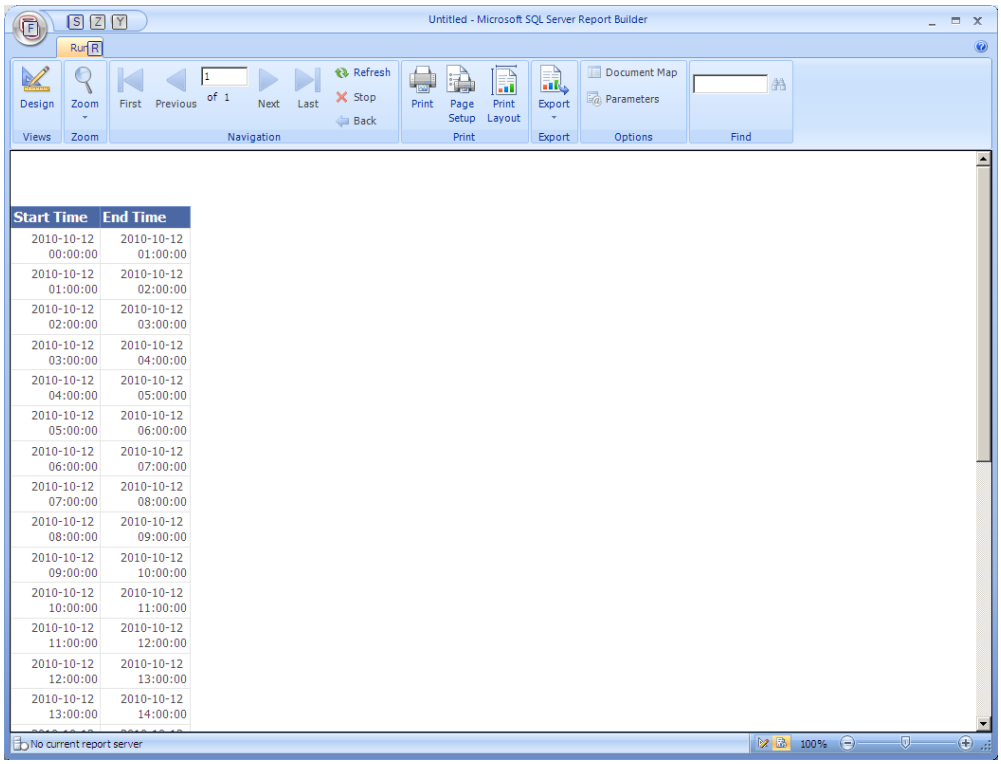

Fig. Report in 'Run' view.

4.3 View the report

Use the ***Run*** command on the Report Builder *Home* tab to view the report.

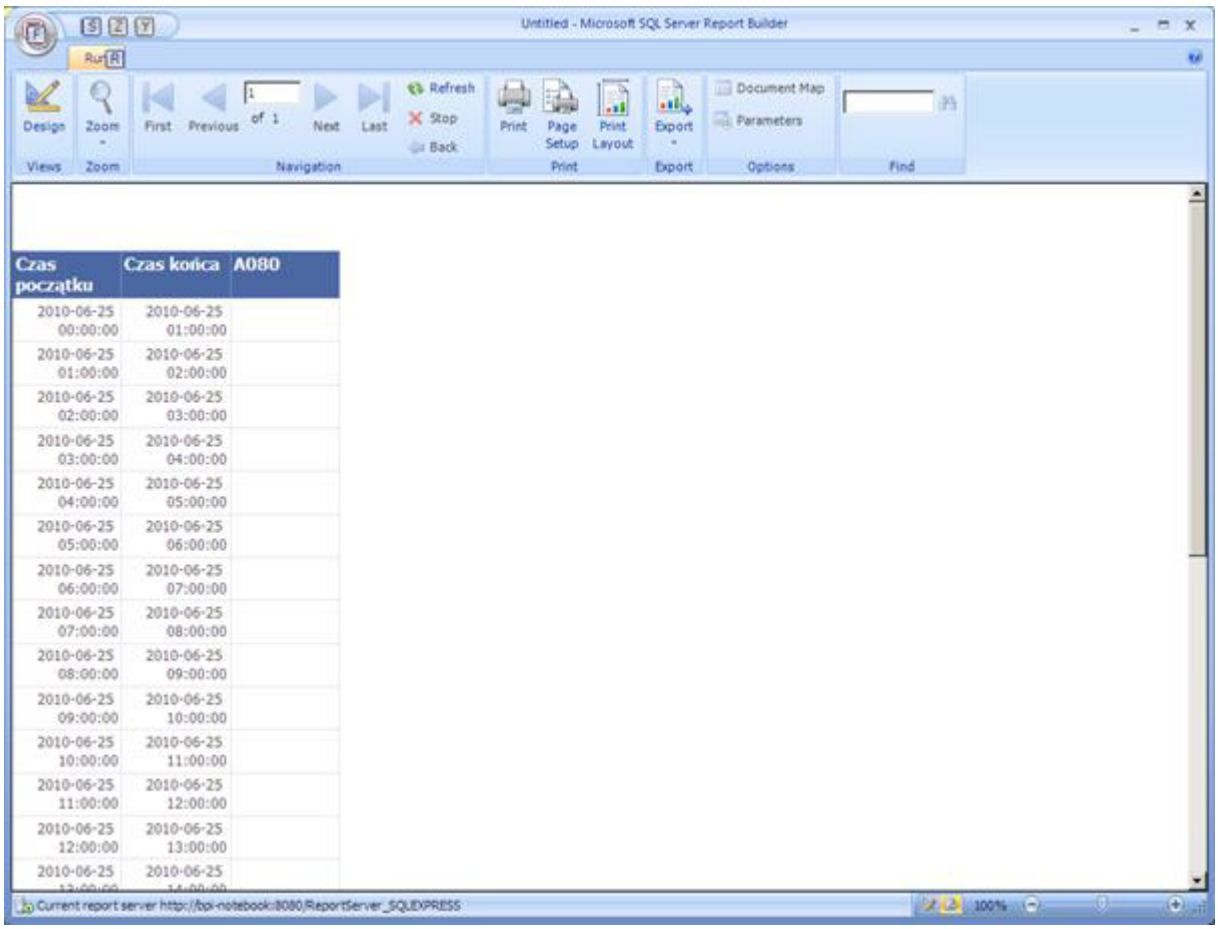

Fig. Report in 'Run' view.

4.4 Add/edit data set

Data sets may be edited from the '***Report Data'*** window pane (the part of Report Builder main window).

ATTENTION: Data source must be specified.

To create a data set:

1. Right-click data source name displayed in the '***Report Data'*** window pane, then click the ***Add Dataset…*** command button to open the ***'Dataset Properties'*** dialog window.

If there is no data source item in the '*Report Data'* window pane select ***New*** menu and ***Data Source...*** command.

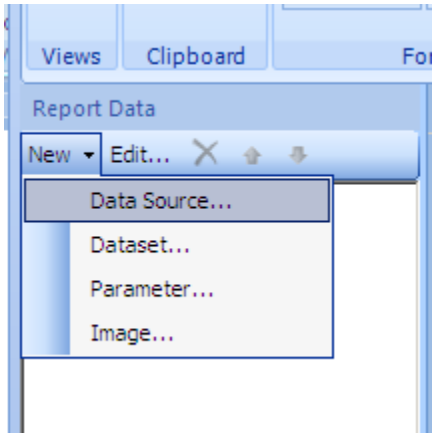

Fig. 'New' menu in the 'Report Data' window.

2. Accept the default name for the edited data set or enter another name into the ***Name*** field.
3. Select name of one of the existing and shared data sources displayed in the ***Data source*** field, or click the ***New…*** button to create a new embedded data source.
4. Depending on the data source type select the appropriate ***Query type*** option:
   - select the ***Text*** option for queries written in the data source query language
   - select the ***Table*** option to retrieve all fields of a relational database table
   - select the ***Stored Procedure*** option to run stored procedure specified by its name.
5. Enter name of the query/stored procedure/table to the ***Query*** field. Alternatively you may click the ***Query Designer…*** button to open a graphical/text query editor, or the ***Import…*** button to import the query from some existing report.
6. Enter maximum number of seconds the report server is to wait for a replay from the database into the ***Time out*** field. Default value 0 means that there will be no limit at all.
7. Click ***OK***.

The data set and its collection of fields is displayed in the *'Report Data'* window pane in the data source node of the Report Builder main window.

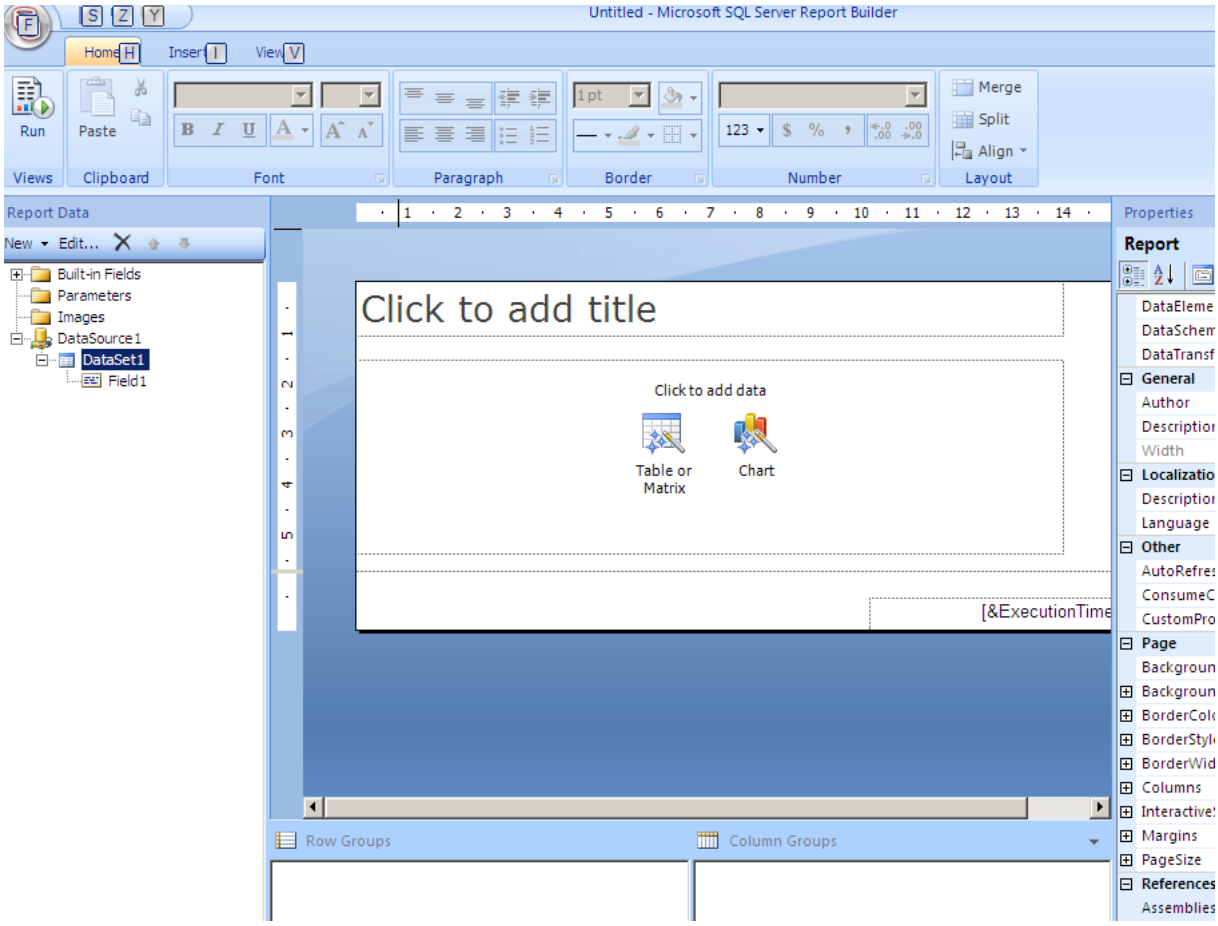

Fig. The data set and its collection of fields in the 'Report Data' window pane.

To edit a dataset:

Right-click the data source name displayed in the *'Report Data'* window pane, then right-click the data set name and click the ***Dataset Properties*** button to open dataset properties window in the query properties view.

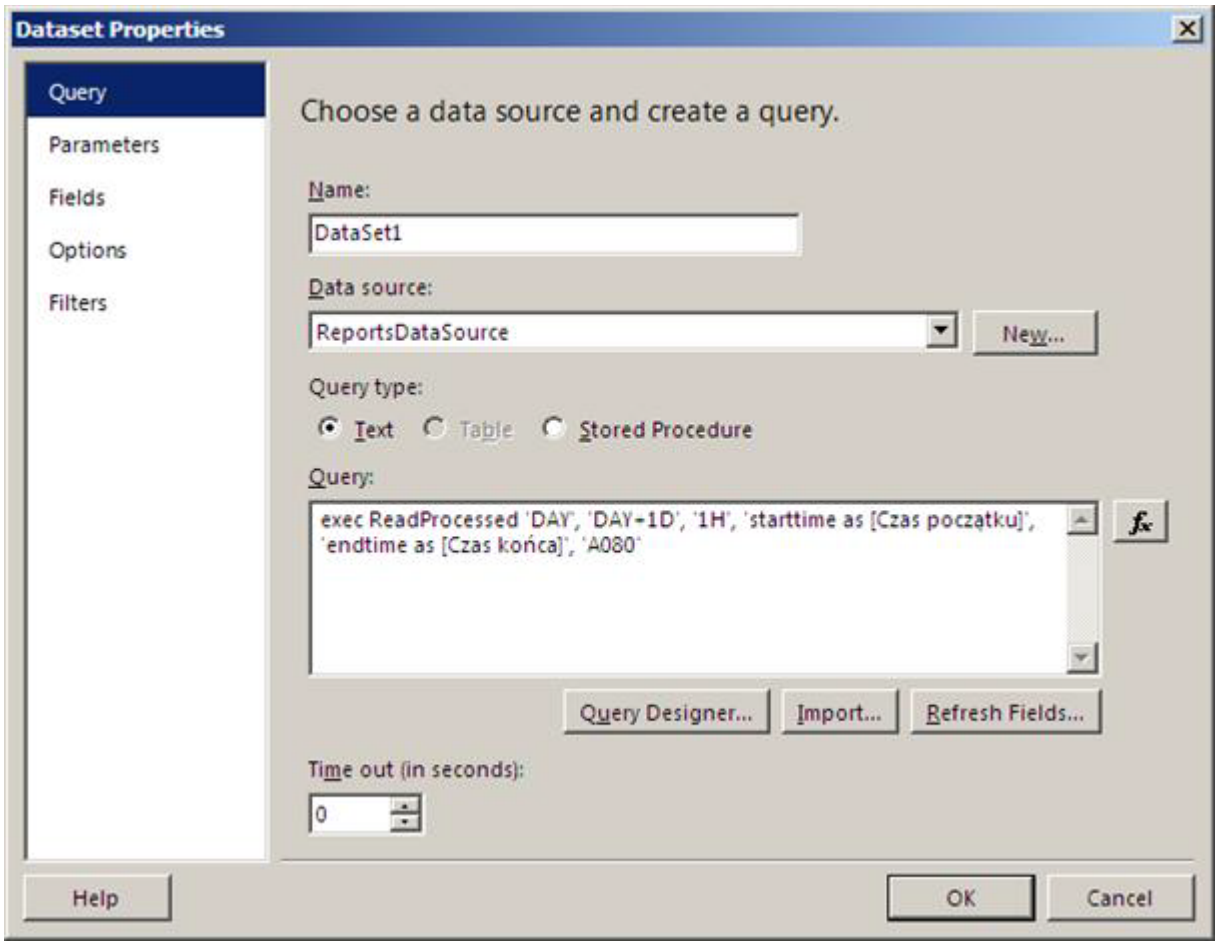

Fig. 'Dataset Properties' window.

Select another category in the left window pane to switch to another dataset property.

4.5 Embedded vs. shared sources of data

Report Builder 2.0 can link a report to a data source shared on a report server or embed a data source into a report; in the latter case the data will be used exclusively by the report.

To be able to link a report to a data source shared on a report server one has to have access to the server and to know data source location within the server. To embed data source one has to specify connection and to know what privileges to specify so that the data needed within the report might be retrieved.

Once a shared data source is selected on a report server, the server becomes the current report server.

To create a link to a shared data source (shared connection):

1. Click the ***New…*** menu bar option in the '***Report Data'*** (left) window pane of the Report Builder main window, then click the ***Data Source…*** option to open the ***'Data Source Properties'*** dialog window.

2. Accept the default name for the edited data source or enter another name into the ***Name*** field.

3. Set the ***Use a shared connection or report model*** option to display list of shared data sources and report models used in the report. If the list is empty, click the ***Browse…*** button and navigate to the report server folder, in which the shared data sources reside.

4. Highlight the shared data source and click ***OK***.

Data source name will appear in the ***'Report Data'*** window pane.

To embed a data source:

1. Click the ***New…*** menu bar option in the ***'Report Data'*** (left) window pane of the Report Builder main window, then click the ***Data Source…*** option to open the ***'Data Source Properties'*** dialog window.

2. Accept the default name for the edited data source or enter another name into the ***Name*** field.

3. Make sure the ***Use a connection embedded In my report*** option is selected.

4. Drop down the ***Select connection type*** list and select data source type e.g. 'Microsoft SQL Server' or 'OLE DB'.

5. Using one of the following methods specify the connection string:

- enter the string directly into the ***Connection string*** text box
- click the ***fx*** formula button to display the ***'Expression'*** dialog window, enter expression into the ***Set expression for:*** box, click ***OK***.

6. Click the ***Build...*** button to open the ***'Connection Properties'*** dialog window with properties of the above specified connection.

7. Depending on the selected data source type fill up appropriate fields of the ***'Connection Properties'*** dialog window (data source type/name, credentials used to connect etc.). Click the ***Test Connection*** button to make sure that the data source is available and that the supplied credentials are correct.

8. Click the ***Advanced*** butoon - the ***'Connection Properties'*** window will appear.

9. Specify credentials to be used to connect to the data source. Data owner is responsible for selecting acceptable credentials. Sometimes the owner decides to share data source on a report server and sets (in the server properties) up credentials available for other users. Ask data owner for the credentials.

10. Click ***OK***.

Data source name will be displayed in the ***'Report Data'*** window pane.

4.6 Save the report

To save the report:

1. Click the icon in the top left corner of the Report Builder main window and select the ***Save As*** command.

2. Specify URL address of the target server report or a local computer target folder. The report will be saved in a *.RDL file.

3. Click the ***Save*** button.

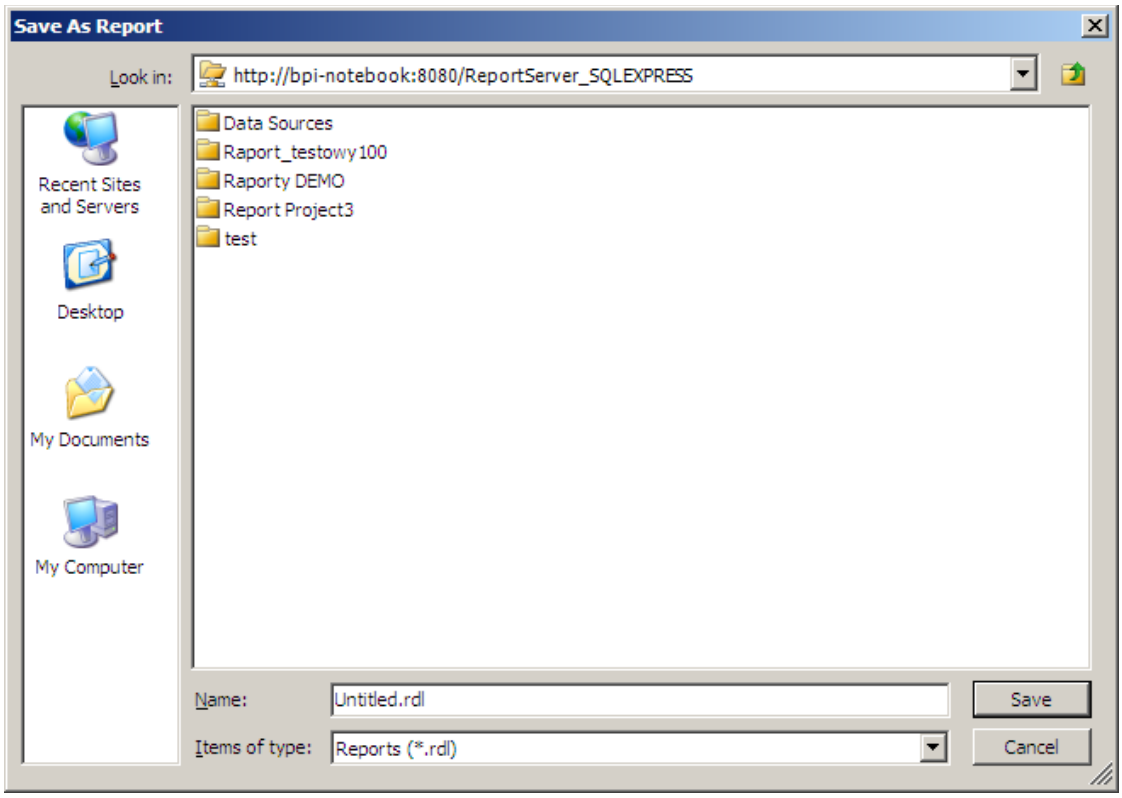

Fig. 'Save As Report' window.

5 Designing reports in Report Designer

Only basic steps of the procedure to define simple reports utilizing process data sources from **asix** system applications have been described in this section. The “Factory” demo application delivered with the **asix** package has been used as a sample application. Services from the SQL Server 2008 Service Pack 1 version Express have been used. For detailed instructions see documentation of the Report Designer program.

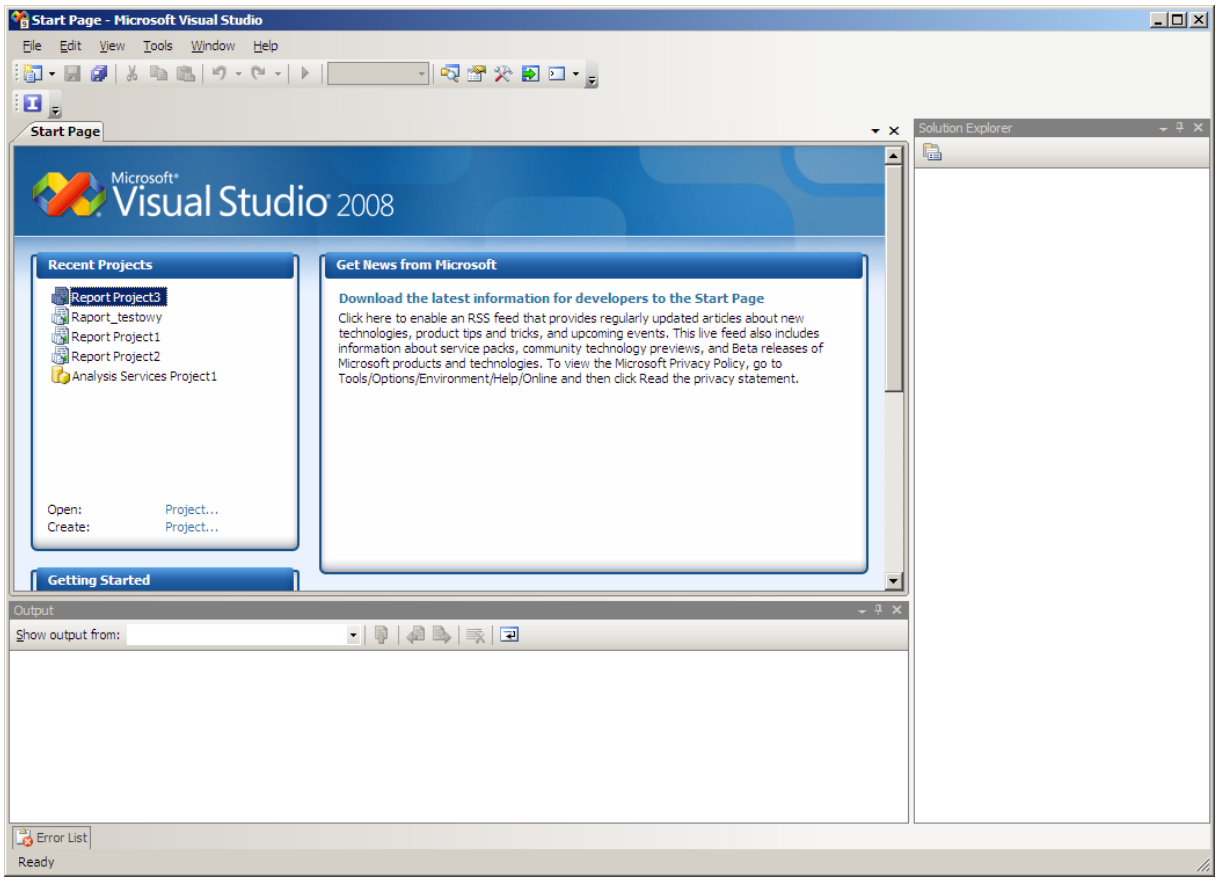

Fig. The main Report Designer window.

1. Run *Start* > *Programs* > *Microsoft SQL Server 2008* > ***SQL Server Business Intelligence Development Studio***
2. Select the ***Create: Project...*** option in the ***'Recent Projects'*** window pane.
3. Enter basic report parameters: its name and location. Click ***OK***.

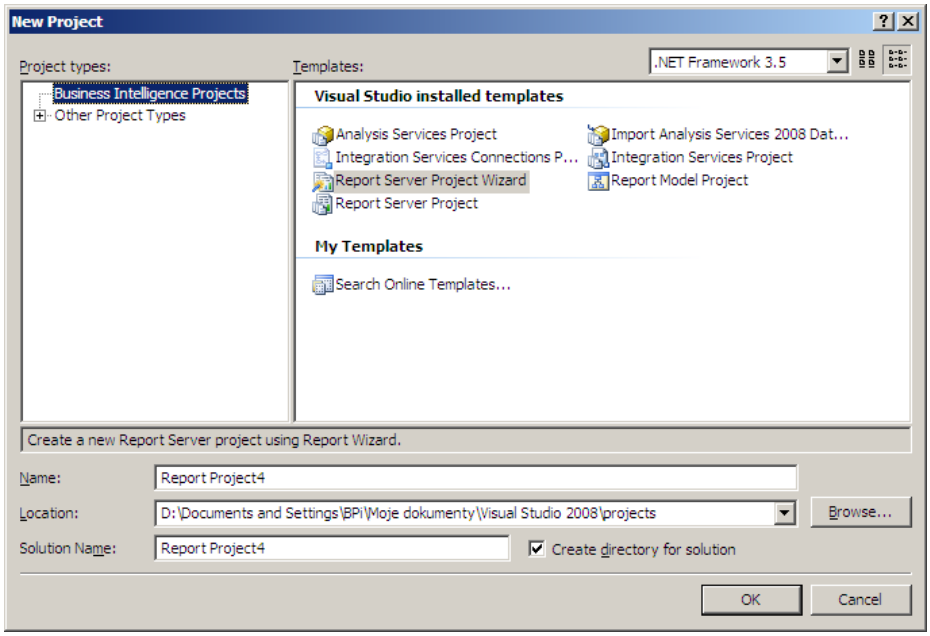

Fig. 'New Project' window.

4. Report Wizard welcome screen appears. Click ***Next***.

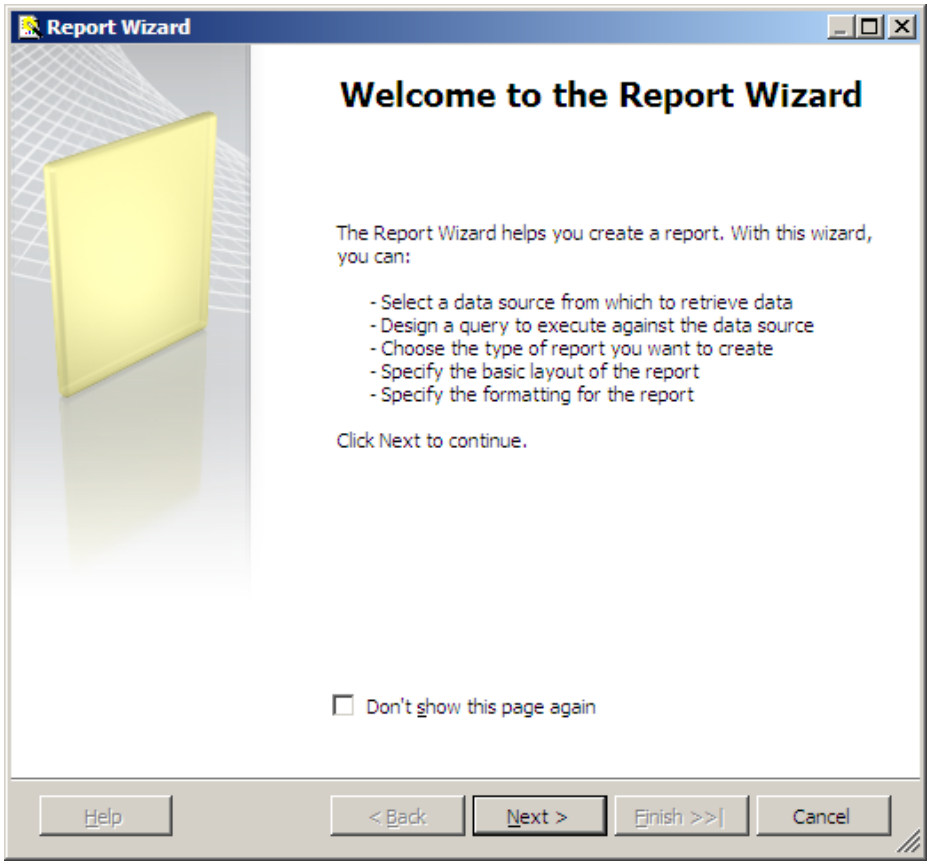

Fig. Report Wizard - greeting window.

5. On the ***'Select the Data Source'*** screen specify data source: name of the SQL server serving the AsixConnect database and the AsixConnect database - use ***Edit...*** button to open ***'Connection Properties'*** window. Click ***Next***.

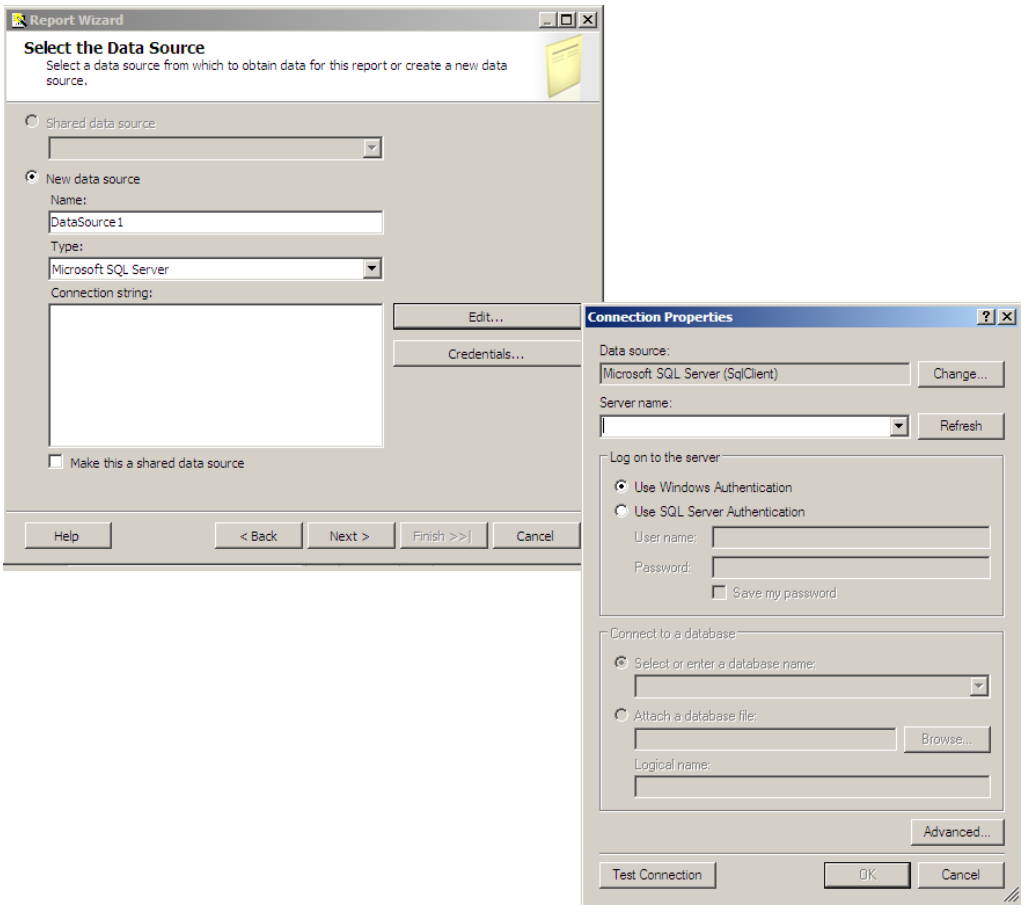

Fig. Report wizard - 'Connection Properties' window.

6. The ***'Design the Query'*** screen appears. Provided that the query has already been created in the AsRaport editor, copy it from the editor to the Windows Clipboard, navigate to the ***'Design the query'*** screen, and paste the Clipboard contents into the ***Query string:*** field.

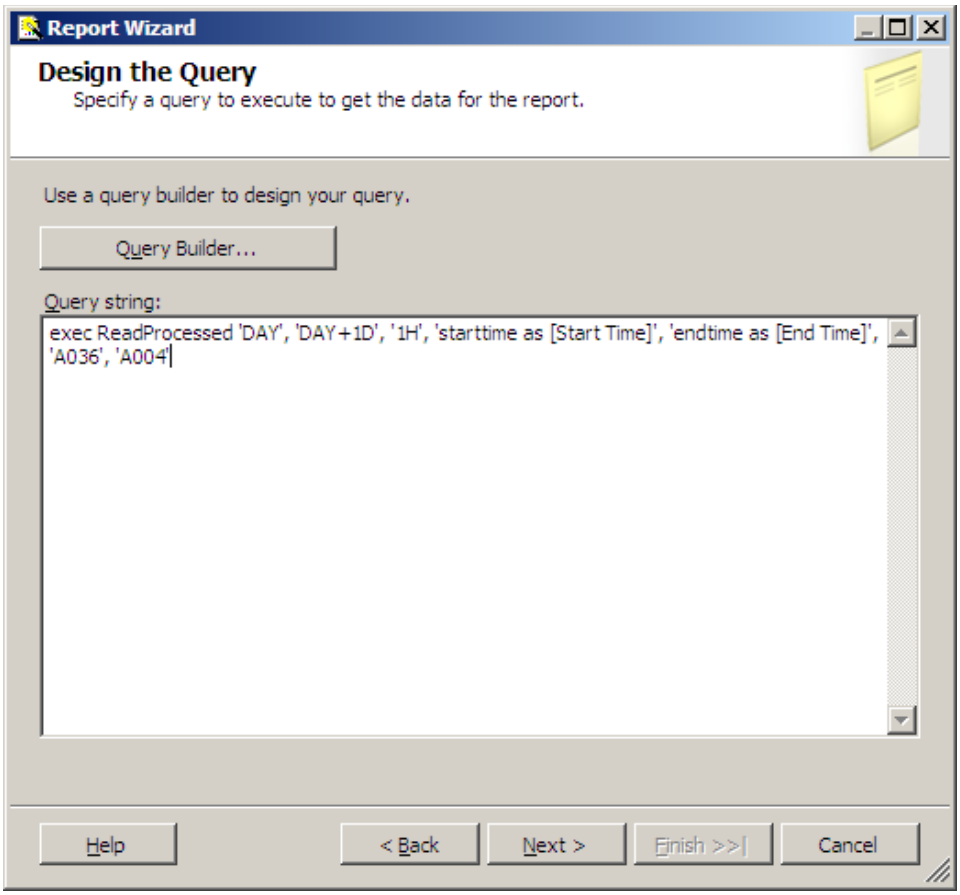

Fig. Report wizard - 'Design the Query'.

7. Select report type: 'Tabular' or 'Matrix'. Click ***Next***.

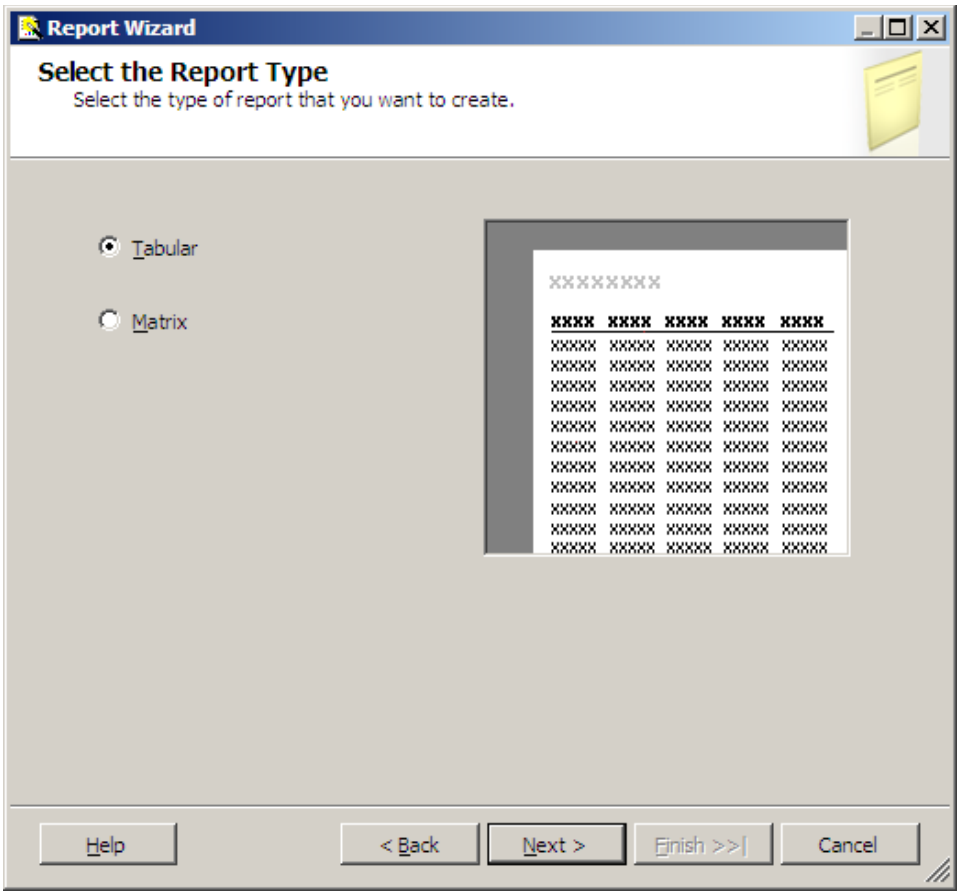

Fig. Report wizard – 'Select the Report Type' window.

8. Design a layout of the data within the report (a table was selected in this example). Click ***Next***.

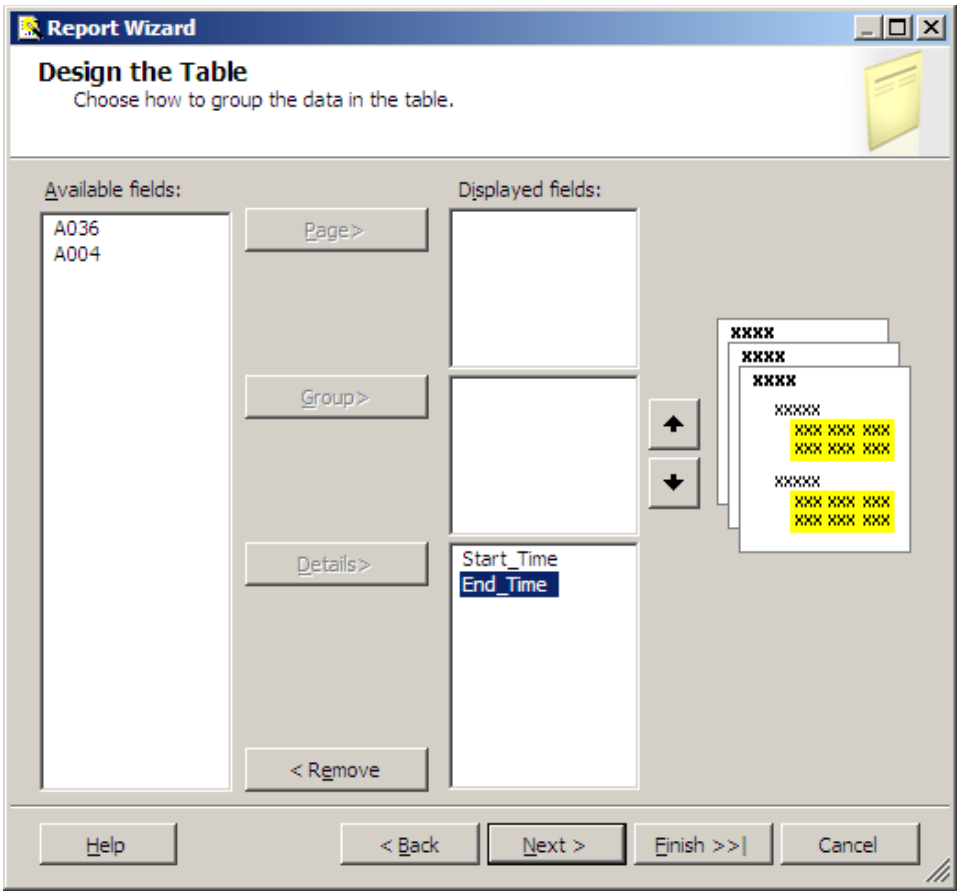

Fig. Report wizard – 'Design the Table'.

9. Select table style. Click ***Next***.

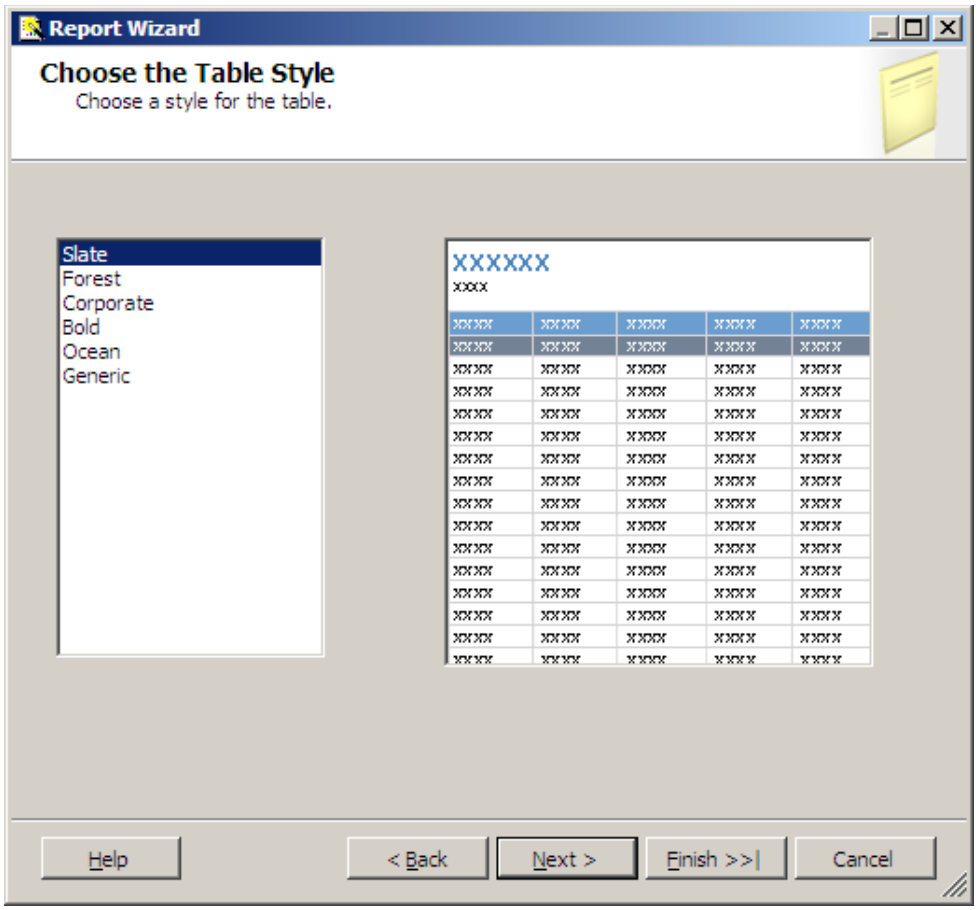

Fig. Report wizard – 'Choose the Table Style'.

10. Specify report server and its folder, to which the reports will be transferred (deployed). Click ***Next***.

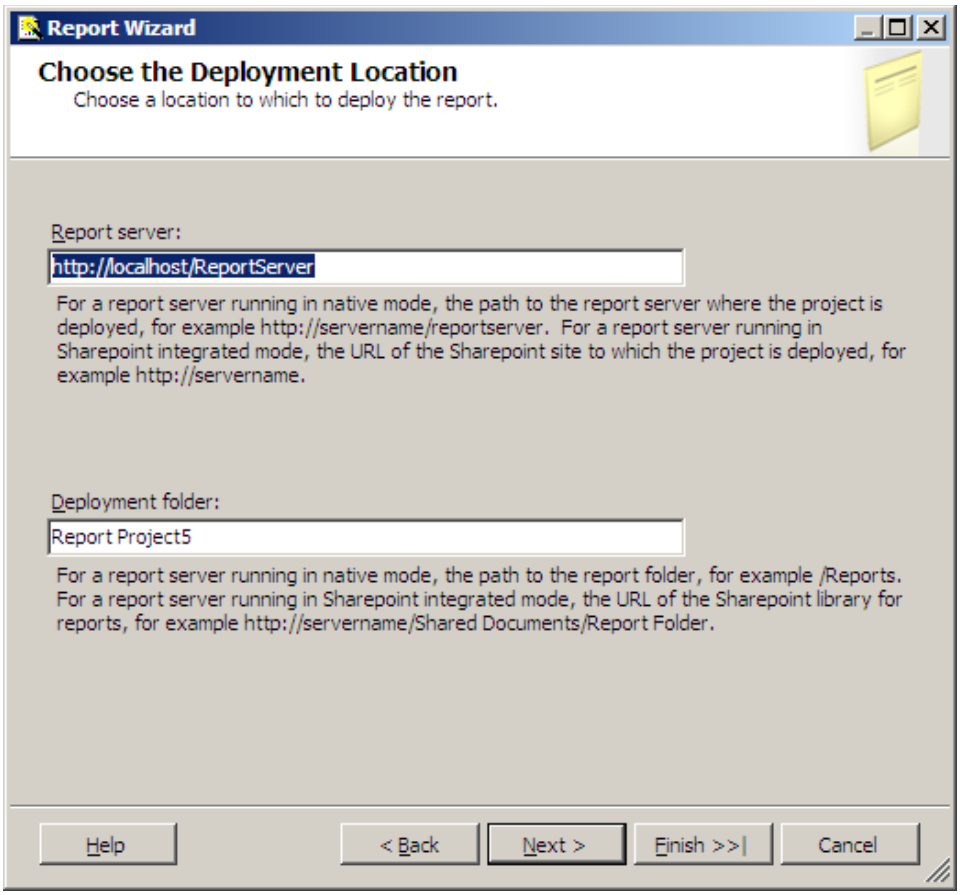

Fig. Report wizard – 'Choose the Deployment Location' window.

11. Specify report name. Click ***Finish***. The report will be displayed in the project view.

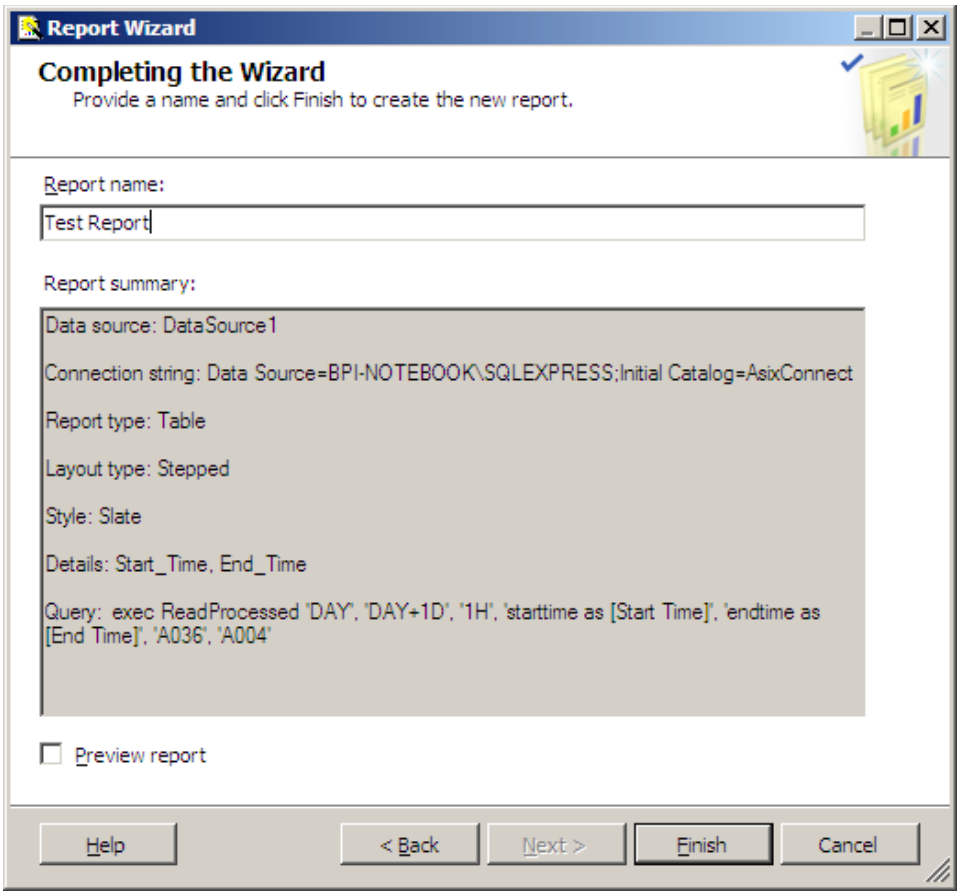

Fig. Report wizard – 'Completing the Wizard' window.

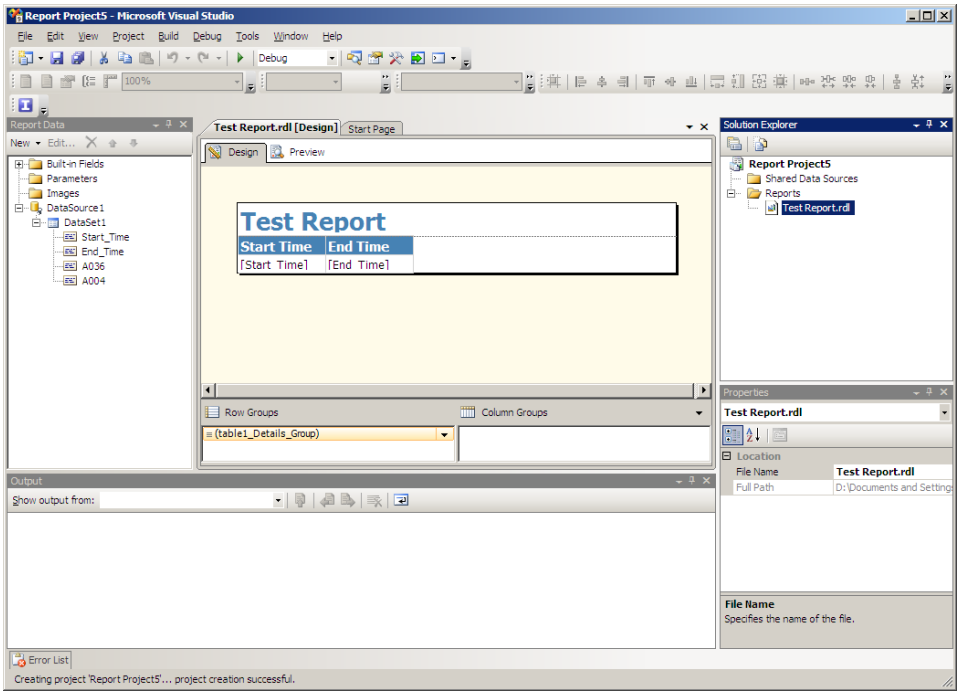

Fig. Final report in design view.

12. Click the Preview button to view the report.

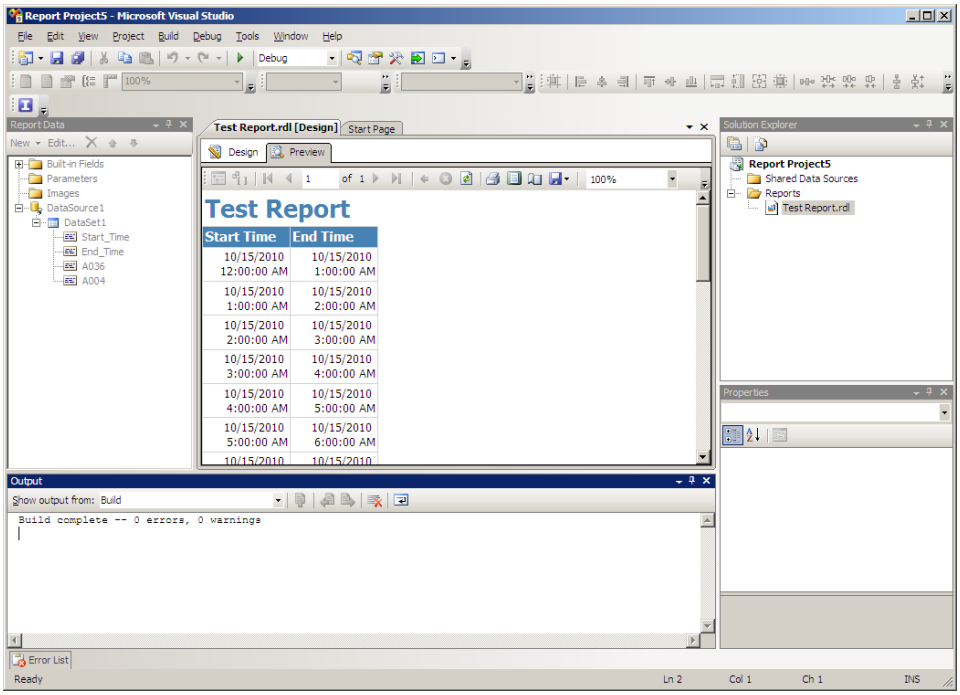

Fig. *Final report in preview*.

13. Select the ***Deploy <report_name>*** option of the ***Build*** menu to store the report on the report server.

6 Access to reports stored on report server

Reports shared on a report server may be viewed in any Internet browser. To view a report, specify URL address of the server. In case of the MS SQL Server 2008 Reporting Services environment the address for report server is:

http://<*computer_name*>/ReportServer_<*SQL_server_instance_name*>

Fig. Access to reports stored on report server via Internet browser.

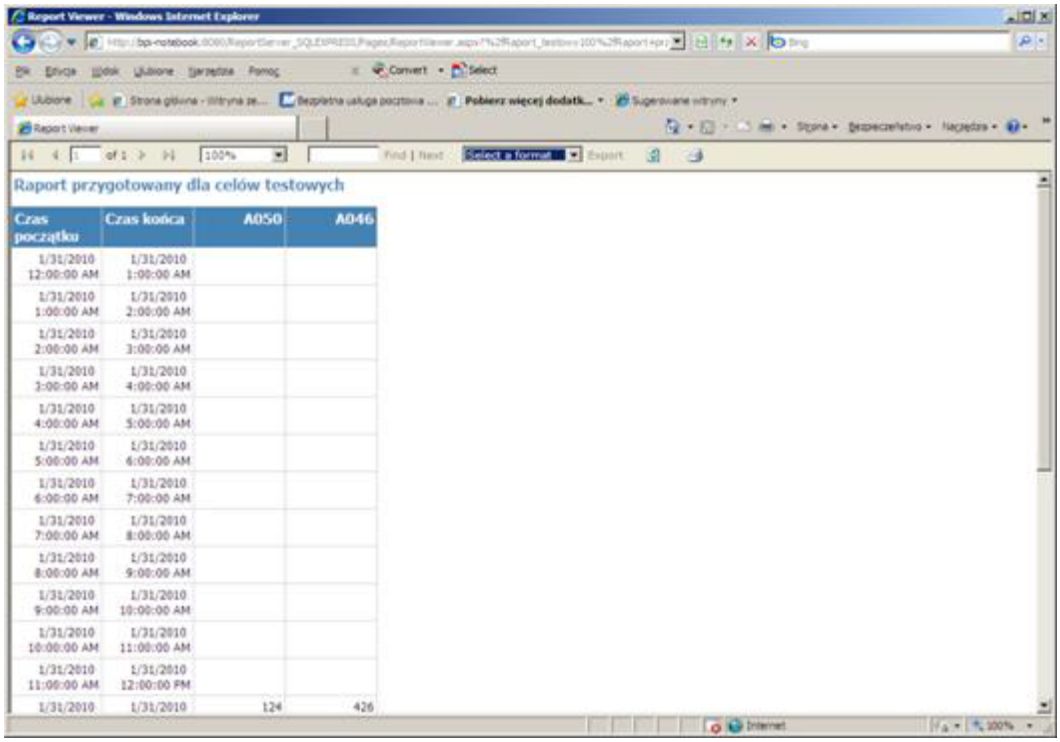

Fig. *Final report in Internet browser.*

Reports accessed via Internet browsers may be printed and/or exported to PDF, Excel and/or Word formats.

7 Viewing reports directly from asix applications

Reports created within the MS SQL Reporting Services environment may be directly viewed from the **asix** system application level using the **AsRapView** report viewer. The viewer may be started by the ASRAPORT operator action according to the syntax:

ASRAPORT /RN=<*report_name*>[<*report_parameters*>]

where:

report_name - name of the report created within the MS SQL Reporting Services environment (report definitions are stored in *.RDL files)

report_parameters - additional parameters of the report.

Location of reports (within the MS SQL Reporting Services report server) to be displayed in the **AsRapView** viewer must be defined in the AsRapViewReportServerConfig.xml configuration file - it is declared with the use of ***Configurator of the AsRaport reporting system*** run from Architect > *Application* menu > *Configure Reporting System AsRaport*.