

Cosmos Data

Cosmos Data is a Cosmos utility that allows to query and export data from a MultiBase/CTSQL database in a simple and intuitive way, and store queries to access them later.

BASE100

BASE 100, S.A.
www.base100.com

Index

1. ARCHITECTURE.....	3
2. CONFIGURATION.....	4
2.2 CONFIGURATION.INI	4
2.2.1 Section [General].....	4
2.2.2 Section [Export].....	4
2.2.3 Section [Include Tables]	5
2.2.4 Section [Exclude Tables].....	5
2.2.5 Section [Include Columns].....	5
2.2.6 Section [Exclude Columns]	5
2.2.7 Section [Rename Tables Labels].....	5
2.2.8 Section [Rename Columns Labels]	5
2.2.9 Section [Expressions].....	6
3. RUNNING COSMOS DATA.....	8
3.1 COMMAND LINE PARAMETERS	8
3.1.1 Parameter <i>-report</i> <SRW file>	8
3.1.2 Parameter <i>-outfile</i> <output file>	8
3.1.3 Parameter <i>-config</i> <configuration file>	8
3.2 FROM ANOTHER COSMOS APPLICATION	8
4. CREATION OF A REPORT.....	10
4.1 "SELECTION" TAB.....	10
4.1.1 List of available tables.....	10
4.1.2 Dependencies of the main table	11
4.1.3 Columns of the selected table.....	11
4.2 "CONDITIONS" TAB	11
4.2.1 Conditon parameters	11
4.3 "GROUPS" TAB.....	13
4.4 "SORT" TAB	13
4.5 "AGGREGATES" TAB	14
4.6 "SUMMARY" LIST.....	15
5. RUNNING A REPORT FROM COSMOS DATA	16
5.1 SCREEN OUTPUT	16
5.2 FILE OUTPUT	16
5.3 EXTERNAL EXPORT	16
6. REPORT FILES SRW	19
6.1 ".CONFIGURATION" SECTION	19
6.2 ".VTABLE" SECTION	19
6.3 ".CONDITIONS", ".PARAMETERS" AND ".WINDOWS" SECTIONS.....	19
6.4 ".JOINS" SECTION.....	19
6.5 ".VCOLS" SECTION.....	20
6.6 ".GROUPS" SECTION.....	20
6.7 ".SORT" SECTION.....	20
6.8 ".AGGREGATES" SECTION	20

1. Architecture

Cosmos Data allows you to easily extract information from a database by graphically selecting the tables and columns to display and defining the selection conditions.

Cosmos Data is a graphical tool that shows the structure of tables and columns of a database defined in the CRF repository of the Cosmos project. The tables are shown following a tree structure according to the JOINS defined in the repository.

The end user begins by selecting columns from a table, and from there, can be linked to any of the related tables.

2. Configuration

Cosmos Data uses two configuration files: cosmosdata.ini and configuration.ini.

2.1 Cosmosdata.ini

The cosmosdata.ini file is the Cosmos Project configuration file used by Cosmos Data. It stores the connection data to the database, the environment parameters such as the DBTEMP variable, and so on.

2.2 Configuration.ini

The configuration.ini file allows you to customize the information displayed to the end user of Cosmos Data.

Here you define the CRF repository you want to use, the name of the connection and the database that contains the data to display, the default directory where the query files will be stored, etc.

The name "configuration.ini" is the default value for this file, but Cosmos Data allows the use of a different configuration file and pass it as the execution parameter of the application "-config <configuration_file.ini>".

The configuration.ini file is divided into the following sections:

2.2.1 Section [General]

In this section you define the variables that indicate the name of the connection and the database that you want to use, the name of the repository and other parameters of internal operation of the application.

CONNECTION. This variable indicates the name of the database connection to use when executing the data query process. The connection name must match a connection to a database defined in the cosmosdata.ini file.

DATABASE. Indicates the name of the database to be accessed when the data query process is executed.

REPOSITORY. Indicates the name of the tables repository CRF file to be used in the Cosmos Data application.

REPORTSPATH. Option variable that will indicate the default directory where the report files will be stored. These files are ASCII files with SRW extension where the name of the report, the name of the master table, the list of columns and joins, grouping columns, ordering, aggregate calculation and report conditions are defined.

BACKUPREPORT. Optional usage variable whose value is a Boolean (TRUE / FALSE, YES / NO) that indicates whether an automatic copy of the report file will be performed each time the [Save] button is pressed.

DEBUGMODE. Optional use variable whose value is a Boolean (TRUE / FALSE, YES / NO) that indicates whether all column information, joins, etc., of the report tables in the application type list controls will be displayed.

2.2.2 Section [Export]

This section will define the project modules that can be added externally to perform data processing tasks. See "External Export" below.

2.2.3 Section [Include Tables]

This section will specify the list of repository tables that will be visible from the application's visual environment. If no tables are included, all tables are displayed. One table per line should be indicated.

2.2.4 Section [Exclude Tables]

This section will list the repository tables that will not be visible from the application's visual environment. In the case that no excluded tables are indicated, all tables are displayed. One table per line should be indicated.

If values are indicated in both "Include Tables" and "Exclude Tables" the behavior is as follows: Only the tables defined in "Include Tables" other than "Exclude Tables" will be displayed.

2.2.5 Section [Include Columns]

In this section you will see the list of columns (table.column) that will be visible from the visual environment of the application. If no columns are indicated, all the columns in the visible tables are displayed. You should indicate one column per line.

2.2.6 Section [Exclude Columns]

This section will list the name of of columns (table.column) that will not be visible from the visual environment of the application. If no columns are indicated, all the columns in the visible tables are displayed. You should indicate one column per line.

If values are indicated in both "Include Columns" and "Exclude Columns" the behavior is as follows: Only columns defined in "Include Columns" other than "Exclude Columns" will be displayed.

2.2.7 Section [Rename Tables Labels]

This section will list the tables in the repository that you want displayed with a description different from the one in the repository.

In other words, if there is a "invoices" table in the repository with a description of "invoices header", in this section we can indicate that it is shown in another description, with the format "<table> = <new description>". For example:

```
invoices=Sales invoices header
```

One table per line should be indicated.

It will also be possible to modify the label of a table from the application environment by right-clicking on the table name and choosing the "Edit Table Name" option.

2.2.8 Section [Rename Columns Labels]

In this section you will see the list of columns in the repository that you want displayed with a description different from the one in the repository.

In other words, if in the repository the description of the "invoice" column of the invoices table is "Delivery Invoice Code" and we want it to be shown as "Delivery Invoice Number", we must indicate the description with the format "<table.column> = <New description> ". For example:

```
invoices.invoice=Delivery Invoice Number
```

You should indicate one column per line.

It will also be possible to modify the label of a column from the application environment by right-clicking the column name and choosing the "Edit Column".

2.2.9 Section [Expressions]

This section should include non-existent columns in the repository tables and which are the result of expressions or formulas between the columns in the table.

The format will be "<table.column.descriptor>=<expression>"

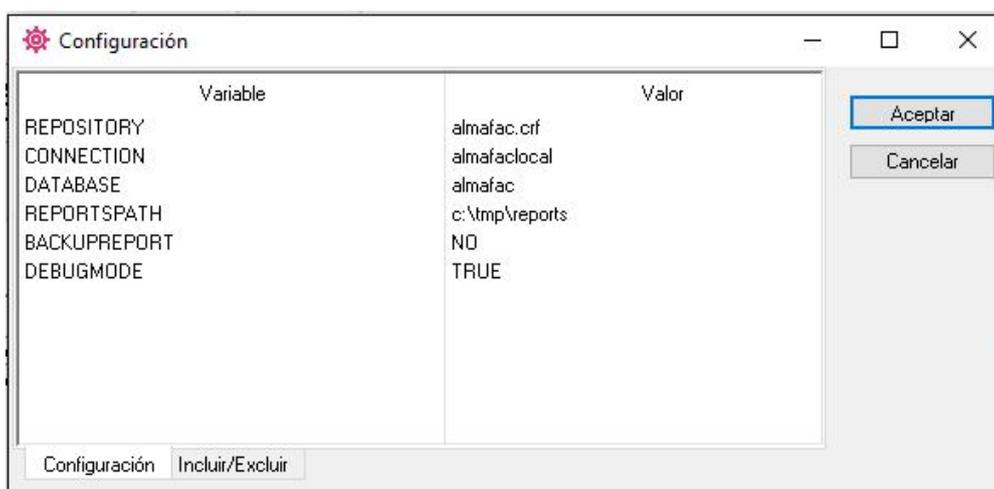
Example:

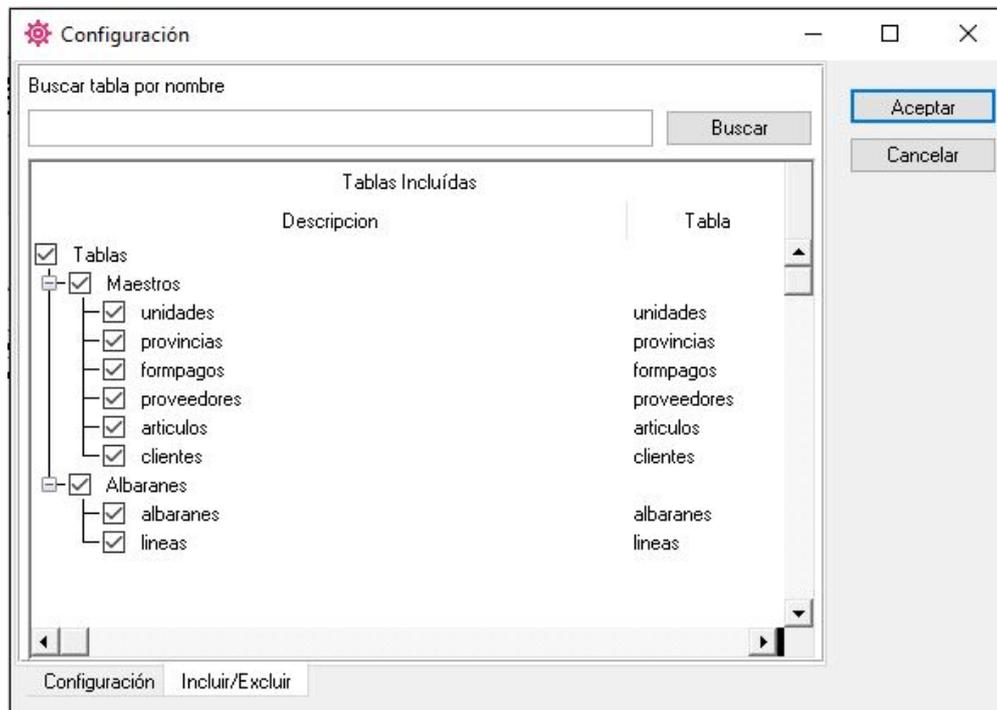
```
Invoices.invoice_year.Invoice Year=to_char($T.invoice_year,'YYYY')
Invoices.today.Today=today
```

If in the expression you want to use the name of the table in front of the name of a column, it should be indicated with the string "\$T".

It will also be possible to add an expression column from the application environment by right clicking on the table and choosing the option "Add Expression".

The Cosmos Data main screen displays a button that allows access to the configuration screen. Clicking this button displays a window where you can change the parameters indicated in the [General] section and the list of included and excluded tables indicated in the sections [Include Tables] and [Exclude Tables]:





3. Running Cosmos Data

Cosmos Data is a Cosmos application, so you need a Cosmos runtime to run it.

The command line for execution would be:

```
C:\cosmos\bin\cosrun.exe -prj c:\cosmos\cosmosdata\cosmosdata.prj
```

Cosmos Data allows its execution without parameters or using the parameters "-report" and "-outfile".

3.1 Command line parameters

Cosmos Data supports two optional parameters: "-report" and "-outfile".

3.1.1 Parameter `-report` <SRW file>

If you want to run Cosmos Data and show when you start a previously created report file, the command line would be:

```
C:\cosmos\bin\cosrun.exe -prj c:\cosmos\cosmosdata\cosmosdata.prj  
-arg "-report c:\myreports\customers_by_state.srw"
```

3.1.2 Parameter `-outfile` <output file>

The "-outfile" parameter is used to tell Cosmos Data to open a report passed as a parameter with "-report", to execute it and export the data to an UNL, CSV or Excel file. At the end of the execution of the report the application will finish automatically.

The "-outfile" parameter must be accompanied by the "-report" parameter.

Example: Si se desea ejecutar Cosmos Data y que al arrancar ejecute un report creado previamente, vuelque la salida del report a fichero y finalice la aplicación, su línea de comando sería:

```
C:\cosmos\bin\cosrun.exe -prj c:\cosmos\cosmosdata\cosmosdata.prj  
-arg "-report c:\myreports\customers_by_state.srw -outfile  
c:\myreports\customers_by_state.xls"
```

3.1.3 Parameter `-config` <configuration file>

The "-config" parameter is used to tell Cosmos Data to use a configuration file other than the default configuration file "configuration.ini". In this way, you can run Cosmos Data with more than one database configuration, repository, etc.

```
C:\cosmos\bin\cosrun.exe -prj c:\cosmos\cosmosdata\cosmosdata.prj  
-arg "-config c:\cosmos\cosmosdata\configuracion_alternative.ini"
```

3.2 From another Cosmos application

The RunCmdExt command is an internal Cosmos command of the Form class that allows you to invoke from a form with a FormTable a command specified in the RUNCMDTEXT variable.

This execution can be performed using a key combination defined in the Cosmos project.

For example, you can invoke the Cosmos Data application by passing the FormTable table name (\$MASTERTABLE) as a parameter, without having to modify all the forms in the application, but only by defining the accelerator and the command line.

Example:

In the application's accelerators, indicate a key combination, for example "Control-Alt-D", and assign the RunCmdExt command to it...

Define the RUNCMDEXT variable in the [Environment] section of the application's INI file.

```
[Environment]
RUNCMDEXT=c:\cosmos\bin\cosrun.exe -prj c:\cosmos\CosmosData\cosmosdata.PRJ
-arg "-mastertable $MASTERTABLE"
```

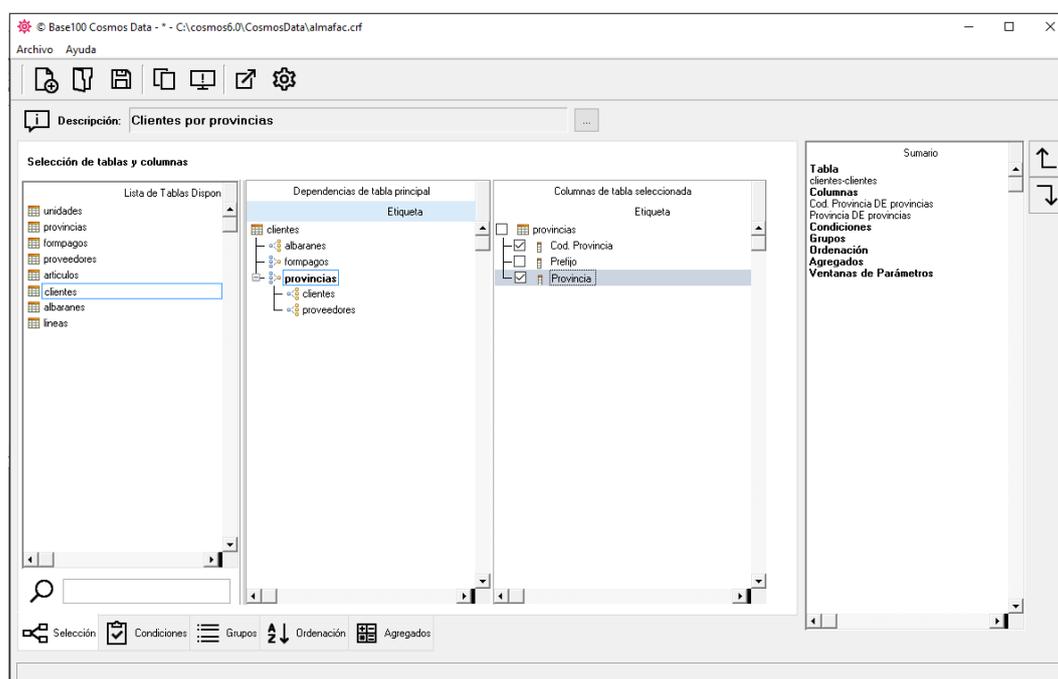
4. Creation of a report

The steps for creating a report can be summarized in:

1. From the "Selection" tab, selection of master table and selection of columns of the master table and its dependencies.
2. From the tab "Conditions", define conditions that will be taken into account when the report is executed.
3. From the "Groups" tab, select the columns by which you want to group.
4. From the "Sorting" tab, select the columns for which you want to sort.
5. From the "Aggregates" tab, select columns for which you want to perform a calculation on the set of values in the group..
6. Save the report with the [Save] button or with the "Save" menu option in the "File" menu.

Cosmos Data main screen consists of five tabs: "Selection", "Conditions", "Groups", "Sorting" and "Aggregates", and a list control showing the summary of the list: Master table, selected columns, conditions, groups, sorting and aggregates.

4.1 "Selection" tab



In this tab are the controls that show the list of available tables, the master table of the report and its dependencies, and the columns of the selected tables.

4.1.1 List of available tables

To the left of the "Selection" tab of the main screen of Cosmos Data, the list of tables defined in the repository is shown and that can be used in the report. In that list, and as the first step to create a query, the master table will be selected by double clicking on it.

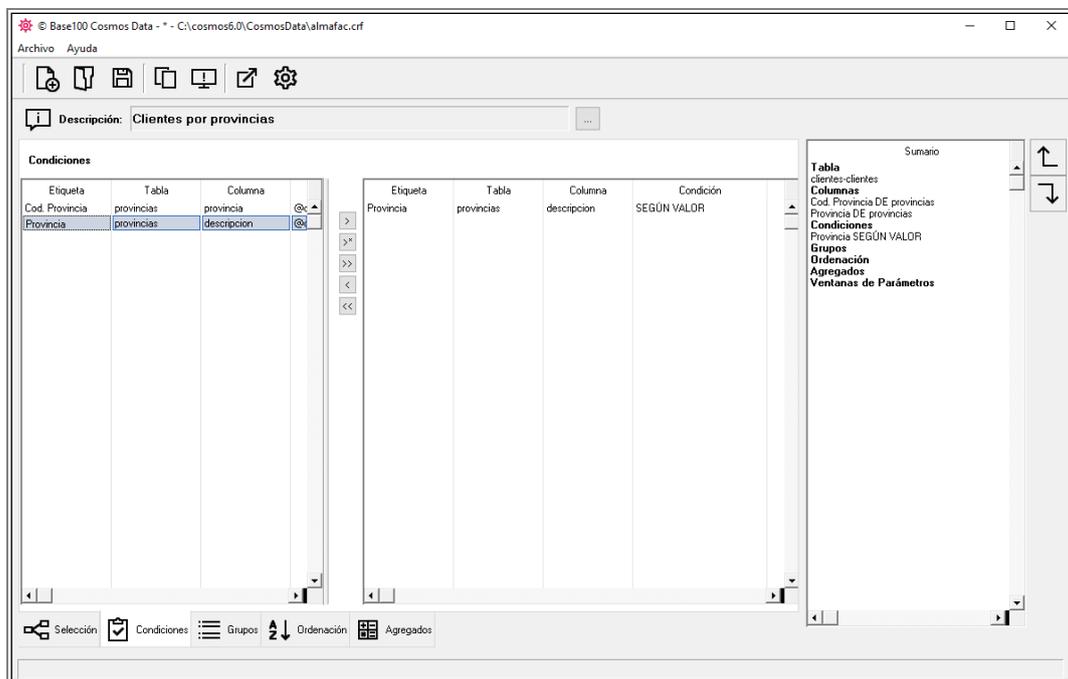
4.1.2 Dependencies of the main table

When selecting the master table, the second list will display the master table and all its dependencies (joins and lines) in the form of a tree. When you select any of the tables in this list, the dependency tree of the table is displayed and the columns in the table available for selection are displayed in the third list (Columns of the selected table).

4.1.3 Columns of the selected table

To select a column of the selected table, click on the column selection box. At that time the selected column will appear in the summary list of the list to the right of the screen (Summary).

4.2 “Conditions” tab



In this tab you can define the conditions to be met by the database records in each of the columns selected to form part of the query result.

In the list on the left you will see all the columns selected in the list, while in the list on the right there will be the columns for which you want to apply the condition.

If more than one condition is selected, these will be evaluated as a Boolean AND operation.

You can define more than one condition for a column by selecting it as many times as necessary.

For a column to be part of a condition it will be necessary to double-click the column in the list on the left. At that time the column will be copied in the list on the right and the parameters of the condition can be defined.

4.2.1 Condition parameters

The list of conditions shows the description of the column, the description of the table to which it belongs, the name of the column and the join to which it belongs.

In addition, the type of condition, the value of the condition and a check will be displayed, indicating whether the filter value will be asked when the list is executed, or whether it will be defined at design time.

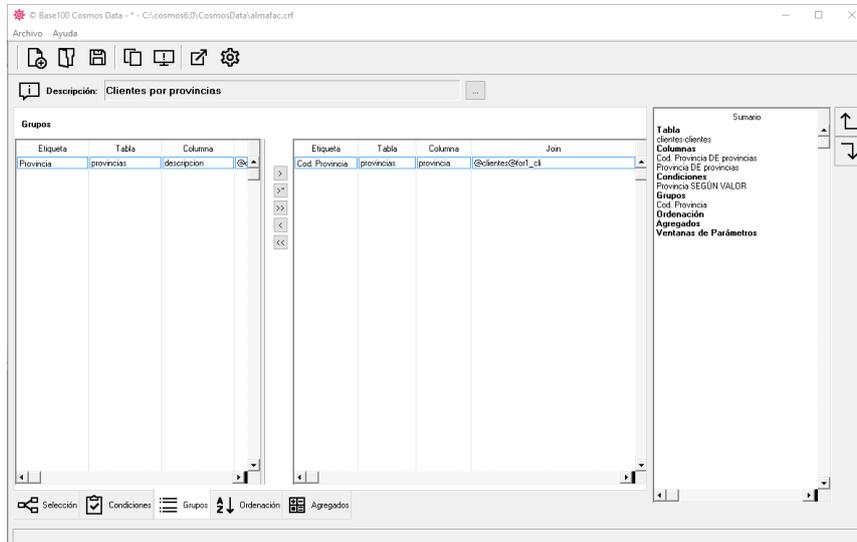
In the "Condition" column of the list the condition type will be indicated with respect to the value indicated in the "Value" column. Clicking on the "Condition" field in the selected column will display a drop-down list where you can choose the type of condition that will be applied to the column.

The possible values in the "Condition" column are:

LESS THAN	The value of the column in the query must be less than the value indicated in the "Value" field.
GREATHER THAN	The value of the column in the query must be greather than the value indicated in the "Value" field.
LESS THAN OR EQUAL TO	The value of the column in the query must be less than or equal to the value indicated in the "Value" field.
GREATHER THAN OR EQUAL TO	The value of the column in the query must be greater than or equal to the value indicated in the "Value" field.
DISTINT OF	The value of the column in the query must be different from the value indicated in the "Value" field.
IS NULL	The value of the column in the query must be null.
IS NOT NULL	The value of the column in the query should not be null.
IN THESE VALUES	The value of the column in the query must be in the list of values, separated by commas, indicated in the "Value" field.
NOT IN THESE VALUES	The value of the column in the query should not be in the list of values, separated by commas, indicated in the "Value" field.
MATCHES	As in EQUAL, allowing the use of metacharacters (? *)
LIKE	As in EQUAL, allowing the use of metacharacters (_%)
ACCORDING TO VALUE	The value of the column in the query must match the expression that is defined in the "Value" column. In this case, in the column "Value" are defined, in addition to values, comparison operators.

If a column is deselected from the column list in the "Selection" tab, it will also be deselected from the list of conditions.

4.3 “Groups” tab



In this tab the list of columns to be grouped will be defined.

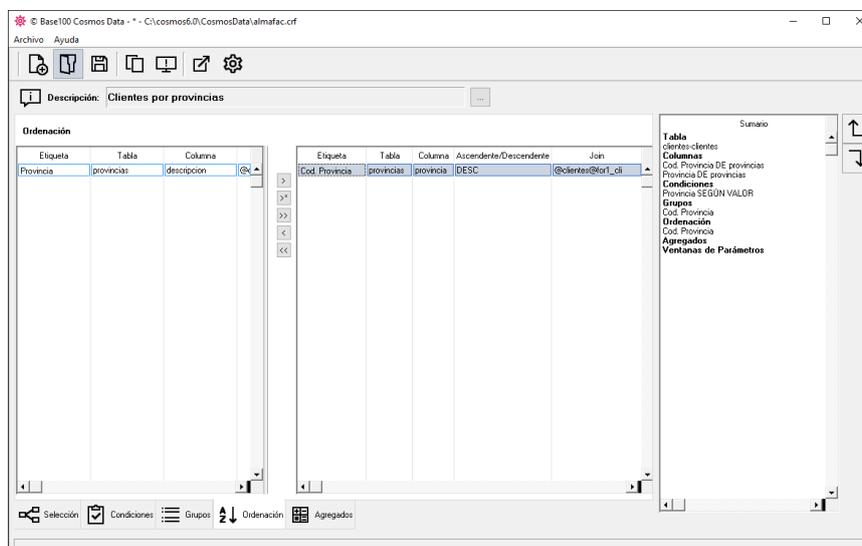
In the list on the left you will see all the columns selected in the list, while in the list on the right are displayed the columns by which you want to group.

For a column to be part of a group it will be necessary to double-click on the column in the list on the left. At that time the column will be copied in the list on the right and will disappear from the list on the left, since it is possible to group only once by the same column.

The order of grouping in the list is defined by the order in which the columns are found in the list on the right.

If a column is deselected from the column list in the "Selection" tab, it will also be deselected from the list of grouping columns.

4.4 “Sort” tab



This tab will define the list of columns by which you want to sort the list, ascending or descending.

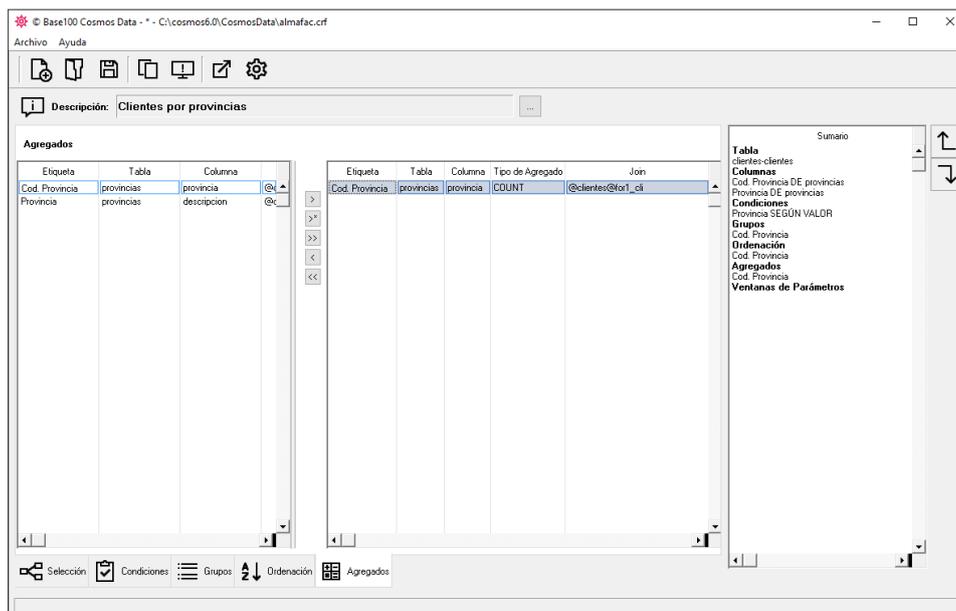
In the list on the left will appear all the columns selected in the list, and in the list on the right will appear the columns for which you want to sort.

For a column to be part of the sorting columns it will be necessary to double-click on the column in the list on the left. At that time the column will be copied in the list on the right and will disappear from the list on the left, since it is only possible to sort once by the same column. By default, the sort is ascending. If you want to change the sorting type, you must click on the column on the value UP / DOWN, and select from the drop-down list.

The order of the sort columns in the list is defined by the order in which the columns are found in the list on the right.

If a column is deselected from the list of columns in the "Selection" tab, it is also deselected from the list of sorting columns.

4.5 "Aggregates" tab



In this tab the list of columns by which you want to calculate aggregate values is defined. In the list on the left you will see all the columns selected in the list, and in the one on the right the columns for which you want to perform aggregate calculation.

To perform a calculation of aggregates of a column will need to double-click on the column in the list on the left. At that time, the column will be copied to the list on the right. It is possible to perform more than one calculation of aggregates by the same column.

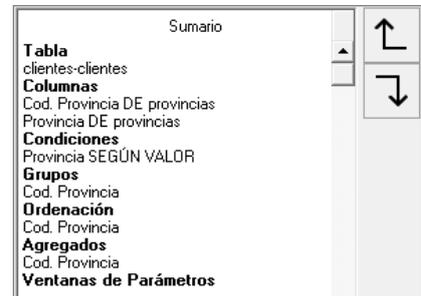
To indicate the type of aggregate calculation (MIN, MAX, SUM, COUNT, AVG), click on the "Aggregate Type" column and select the desired aggregate type from the drop-down list.

If a column is deselected from the column list in the "Selection" tab, it will also be deselected from the aggregates calculation list.

4.6 “Summary” List

To the right of the main screen of the application a summary of the characteristics of the listing is shown. In this summary will appear the name of the table and its description, the name of the columns selected for the list, the selected filter conditions, the columns by which it is grouped, the columns by which to sort and the columns of aggregates.

The buttons to the right of the list allow you to change the position of the columns in the report.



5. Running a report from Cosmos Data

The result of the execution of a report can be displayed on screen or dump to a text file in UNL, CSV or Excel format.

If the report has not been saved or changes have been made that have not been archived, the file must be saved from the "Save" option in the "File" menu or the [Save] button before execution. If you want to change the name of the report file, you must execute the "Save As" option in the "File" menu.

For the execution of the report to be correct, Cosmos Data must have defined the name of the connection and the name of the database to be connected to extract the data. See the CONNECTION and DATABASE variables in the "General section" of the "configuration.ini" configuration file discussed earlier in this document.

The structure and name of the tables and the name and type of columns of the CRF repository used in the report must match those of the database from which the data is to be extracted.

5.1 Screen output

To execute and display the result of the execution of a report created in Cosmos Data, you must select the "Execute" option from the "File" menu or click on the [Execute] button.

The result of the report will appear on a separate screen, in list format.

If grouping columns are selected, the list will appear in tree format. It is possible to change the grouping columns in the result window by pressing the [Group] button, and dragging the columns from the list header to the grouping bar. From this grouping bar it is also possible to indicate aggregate columns. Pressing the [OK] button will recalculate the list and will be displayed in the display list.

5.2 File output

Cosmos Data allows storing the output of a report execution in a text file in UNL, CSV or Excel format.

The output in UNL format will create a text file in which you will write one line per record, and each field will be delimited by a pipe character "|".

The output in CSV format will create a text file in which one line will be written per record, and each field will be delimited by a semicolon character ";".

The output in Excel format will create an Excel XML file.

To select the output to file, you must execute the "Export To ..." option in the "File" menu and then choose the desired option (UNL, CSV or Excel). A selection window will appear where the directory and the name of the output file will be indicated.

5.3 External export

Cosmos Data allows the execution of Cosmos modules created by the programmer in order to customize the data output of an SRW report.

These modules must be programmed in Cosmos language, added to "cosmosdata.prj" and associated with the external export from the "configuration.ini" file in the [Export] section.

The external export modules will be defined in the [Export] section of the "configuration.ini" file as follows:

```
<description of external export>=<Cosmos module name>
```

Example:

```
[Export]
Dev Extreme Pivot Table=pivotdevextreme
Graphic Charts 2=gc2
```

The above example defines two external export modules: "pivotdevextreme.omd" and "gc2.omd". These two modules should be added to the "cosmosdata.prj" project from the Cosmos development environment.

When the user selects the "External Export" option in the "File" menu, Cosmos Data will display a screen with the external export modules defined in the [Export] section. When choosing an external export module and pressing the [OK] key, Cosmos Data will run the report and dump the output to a file. It will then execute the Main method of the module associated with external export.

The external export modules must implement a Main method that will receive as parameter the path of the output file in the execution of an SRW report, and the description and SQL type of the columns of the same. In this way, these external modules programmed in Cosmos will be able to treat the information generated from Cosmos Data and display graphs, export data to other formats, etc.

Main method example:

```
main (dataFile as char, columnsDescribe as char, columnsType
as char)
```

The "dataFile" parameter is the path of a text file in UNL format with the output of the report. Each row will be a line and the columns will be separated by pipes ("|").

The "columnsDescribe" parameter is a character string with the descriptions of the report columns separated by pipes ("|").

The "columnsType" parameter is a character string with the SQL types of the report columns separated by pipes ("|"). The strings of data types will be:

Data type	Parameter
CHAR	0
SMALLINT	1
INTEGER	2
TIME	3
DECIMAL	5
SERIAL	6
DATE	7
MONEY	8

Data type	Parameter
NULL	9
DATETIME	10
BINARY	11
NOT NULL flag	Data type + 256

The Cosmos external export module will be responsible for deleting the data file passed as the first parameter at the end of its execution.

Until the module is executed, the control will not be returned to the main screen of Cosmos Data.

6. Report files SRW

The report description files generated by Cosmos Data are stored in text format and with the SRW extension.

To use an existing report you must open it from the "Open" option in the "File" menu.

Cosmos Data manages the report description files from the application environment. It is recommended not to manually edit these files. If it is necessary to modify them manually from a text editor, then a description of the sections of the text is made.

6.1 “.CONFIGURATION” section

In this section the description of the listing will be indicated.

```
.CONFIGURATION
DESCRIPTION "Report of customers by state "
```

6.2 “.VTABLE” section

This section describes the description and name of the report master table.

```
.VTABLE
Customers=customers
```

6.3 “.CONDITIONS”, “.PARAMETERS” and “.WINDOWS” sections

When you create a report from Cosmos Data and define a condition, it is stored in the ".CONDITIONS" section.

```
.CONDITIONS
@clientes@for1_cli.provincia|complex|?param_1
```

If a condition has been defined in the report condition screen and the "Ask to Run" box has been checked, the condition is said to be parameterized. The parameter definition is stored in the ".PARAMETERS" section.

```
.PARAMETERS
State Code = param_1,A100, WINDOW window_param_1
```

In the ".WINDOWS" section, database query data will be stored when the "Ask to Run" checkbox is selected in the definition of a condition. A parameter will always be associated with a WINDOW.

```
window_param_1 "select distinct(provincia) from provincias" 10 LINES 3
COLUMNS LABEL "Seleccione registro por el que filtrar" GET 1
```

6.4 “.JOINS” section

In this section you define the relationships between the tables to which the columns selected in the report belong.

```
.JOINS
@customers@for1_cus = MASTER <state> states <state>
```

6.5 “.VCOLS” section

In this section you define the columns selected in the report (name of table or join to which they belong, description of column, type of column, etc.).

```
.VCOLS  
State = description,A-20, JOIN @customers@for1_cus  
State Code = state,A+12, JOIN @customers@for1_cus
```

6.6 “.GROUPS” section

In this section the grouping columns are described.

```
.GROUPS  
@customers@for1_cus.state
```

6.7 “.SORT” section

In this section the ordering columns are indicated, and also if the sort is ascending or descending.

```
.SORT  
@customers@for1_cus.state=DESC
```

6.8 “.AGGREGATES” section

This section defines columns from aggregate calculations.

```
.AGGREGATES  
@customers@for1_cus.description=COUNT
```