

DOCUMENTATION of the component

4D_Info_Report_v4

(Also known as "The information Component")

4D_Info_Report Reference version 23, set April 4, 2017, by Thomas Schlumberger, 4D France.

This documentation is matching the version 4.9rZ of the component 4D_Info_Report.

SUMMARY:

Summary.....	Page 1
Presentation.....	Page 1
List of the shared methods.....	Page 3
Details on the shared methods	Page 4
Description of the main parameter for the creation of reports.....	Page 13
Discover the Component run in stand-Alone	Page 14
The Format and content of the generated reports:	Page 31
Deployment of the Component	Page 38
Conclusion	Page 42

PRESENTATION:

This information component is mainly designed to provide in readable text documents a description of:

- The computer (Hardware / Operating system)
- The version of the 4D Application
- The Host database settings (and table summary)

Supported version of 4D:

4D v16 version: v16.x, v16 Rx
4D v15 version: v15.x, v15 Rx

4D v14 version: Oldest: 4D v14.1 (including 4D Server v14 64-bit)
4D v13 version: Oldest: 4D v13.1 (including 4D Server v13 64-bit)
4D v12 version: Oldest: 4D v12.1 (including 4D Server v12.1 64-bit)

All versions of the component can handle reports created using 4D v16, 4D v15, 4D v14, 4D v13, 4D v12, with English or French description.

Some information about this component:

- It does not require or use plug-in or other component (including 4D SVG).
So it can be used with any Host database (4D v12 to 4D v16 Rx), or as a stand-alone database (will ask for an initial creation of a data file when used stand-alone).
- This component does not create or use table / record in the Host database (or in an external database).

- Confidentiality:

The component does not access or share any record content, resource content, etc... It will only describe some basic elements of the structure (number and name of the tables, number of fields or indexes (but not their name or their type), and the number of records for each table. (optional parameter values let you hide some details). It will get some details on the computer and operating system, and the path names used for the database and the application.

- Compatibility:

4D_Info_Report_v4 can handle all reports created by aa4D_Report (since v2.0), with French or English content (the two possible ones), or reports created years ago.

Supported language for the interface and the content of the reports):

(based on the release of 4D used: in French with a French release of 4D, otherwise in English. Now the menu bar of the dialogs match other versions (except Japanese))

Supported platform, O.S, processor:

All Operating systems compatible with 4D, Intel and AMD processors.

Naming convention:

- Name of the component:

The name of the component is "4D_Info_Report_v4" since March 2014.
("_v4" is for all v4.x versions, and will be replaced by "4D_Info_Report_v5" next).

The name of the provided zip archives is completed with more details:
(the exact version (like "_9" for v4.9), and the built version of 4D, like "_v12" for 4D v12, "_v13" for 4D v13, "_v14" for 4D v14.x, or v14 Rx, , "_v15" for v15 or v15 Rx, , "_v16" for v16 or v16 Rx)

- Name of the shared methods (not changed for historical reasons):

The name of the shared methods always begins with "aa4D_".

List of the shared methods of the component: (by category)

Creation of reports:

[aa4D_M_CreateReport_faceless](#)
[aa4D_NP_Util_CreateReport](#)
[aa4D_NP_Util_CreateReport_Serv](#)
[aa4D_M_Report_CreateOnServerSee](#)
[aa4D_NP_Schedule_Reports_Server](#)

Display of existing reports:

[aa4D_NP_Get_Last_Server_Report](#)

Display of dialogs:

[aa4D_NP_Report_Analyse_Display](#)
[aa4D_NP_Report_Compare_Display](#)
[aa4D_NP_Report_Manage_Display](#)
[aa4D_NP_Report_Export_Display](#)

Utility: (* = invisible shared methods)

[aa4D_M_Get_Build_4D_Text_Call](#)
[aa4D_M_Folder_Rep_SetPath_local](#) *
[aa4D_M_Folder_Rep_GetPath_local](#) *
[aa4D_M_Get_Component_Version](#) (New in v4.2)
[aa4D_M_Get_Expiration_Date](#) (New in v4.2)
[aa4D_M_Get_Information_Text](#) (New in v4)
[aa4D_NP_Reports_Max_Set_Limit](#) *
[aa4D_NP_Reports_Max_Get_Limit](#) *
[aa4D_NP_Agent_Schedule_Call](#) * (New in v4.7)
[aa4D_M_Agent_Get_Records_Call](#) * (New in v4.7)

Details on the shared methods of the component:

aa4D_M_CreateReport_faceless

aa4D_M_CreateReport_faceless ({ContentMode}; {CommentPointer}; {Nb processes list})

Parameter	Type		Description
ContentMode	Longint	→	Define the mode of the description of report (for tables) Default is to describe all if this parameter is omitted
CommentPointer	Pointer	→	Pointer to a text from the Host DB to be included in the report
Nb processes list	Longint	→	Define the number of processes to list in the report

Description

The **aa4D_M_CreateReport_faceless** command allows you to create a report within the current process of your Host database. It was mainly designed as a tool to debug your code, when you need for example to get details on the memory and cache usage at a particular condition, or when repeating in a loop a part of your code to detect memory leaks. (it is usual to hide the Table list via the ContentMode parameter in this case).

aa4D_M_Get_Build_4D_Text_call

aa4D_M_Get_Build_4D_Text_call ({ContentPointer}; { PathPointer})

Parameter	Type		Description
ContentPointer	Pointer	→	Pointer to a text variable to complete with result
PathPointer	Pointer	→	Pointer to a text variable containing the full path of the file.

Description

The **aa4D_M_Get_Build_4D_Text_call** command allows you to display (or retrieve) the detail of the version and build number of an executable or a plug-in.

It works cross-platform, meaning that you can get the build number of an executable file on Mac OS, or parse the "Info.plist" of a Mac Application on Windows.

(When you parse a file that contains a "/contents/info.plist", this is the file to select in the dialog to retrieve the information).

If no parameter is passed, a dialog to select the file is displayed, and the result is displayed in an Alert. If you pass a pointer to a text variable as a first parameter, the content of the Alert will be redirected to the pointed text variable.

If you pass a pointer to a text variable as a second parameter, the pointed text variable must contains the full pathname of the file to be examined.

aa4D_NP_Get_Last_Server_Report

aa4D_NP_Get_Last_Server_Report ({Report_Name_p}; {Report_Content_p})

Parameter	Type		Description
Report_Name_p	Pointer	→	Pointer to a text to retrieve the name of the report
Report_Content_p	Pointer	→	Pointer to a text to retrieve the content of the report

Description

The **aa4D_NP_Get_Last_Server_Report** command allows you to retrieve the last report created in the Folder_reports of the database, even from 4D Remote.

Without parameter, it will display it's content in a dialog. You can then save it in your computer.

(if run in Local mode, it will retrieve the last report of your application).

aa4D_M_Report_CreateOnServerSee

aa4D_M_Report_CreateOnServerSee ({ContentMode}; {CommentPointer}; {Nb processes list})

Parameter	Type		Description
ContentMode	Longint	→	Define the mode of the description of report (for tables) Default is to describe all if this parameter is omitted
CommentPointer	Pointer	→	Pointer to a text from the Host DB to be included in the report
Nb processes list	Longint	→	Define the number of processes to list in the report

Description

The **aa4D_M_Report_CreateOnServerSee** command allows you to create a report on the Server, and after a small wait, retrieve the last created report from 4D Server, and display it in the same dialog as when creating a report in Local mode: But from there, you will be able to save it locally.

(If no previous report was created on the Server, this command might miss the last created report and retrieve the previous one. This is because the first report created after the Startup of the application is doing more processing than for later reports, completing a small document named "Array_profiler.txt". In this case, just execute "aa4D_NP_Get_Last_Server_Report" to be sure to retrieve the last created report.)

aa4D_NP_Report_Analyse_Display

aa4D_NP_Report_Analyse_Display ({Report name or Full path}; {ReportContent})

Parameter	Type		Description
Report name or Full path	Text	→	Name of the report or full path name of the report
ReportContent	Text	→	Text content of the corresponding report

Description

The **aa4D aa4D_NP_Report_Analyse_Display** command allows you to display a dialog that let you compare the (last created) report with an earlier one (or a later one), and see for example the variation of records in each table (by value and percentage), to get an idea of what is moving most.

If you pass a Full path in \$1, the Report will be directly parsed to retrieve it's content.

If you pass the Report Content in \$2, the content will be parsed without opening the report file.

aa4D_NP_Report_Compare_Display

aa4D_NP_Report_Compare_Display

Description

The **aa4D_NP_Report_Compare_Display** command allows you to display a dialog with a List box that let you open a local folder of reports to compare the main usage (Number of users, Process, etc..) and main memory and cache evolutions.

From this dialog, you can export the results in an external document, including Excel.

You can also display a new dialog with a SVG graph.

This is the main dialog to analyse external reports. (see some details later in the documentation).

aa4D_NP_Report_Manage_Display (secured)

aa4D_NP_Report_Manage_Display

Description

The **aa4D_NP_Report_Manage_Display** command allows you to display a dialog to upload reports (or a selection of reports) from 4D Server, and copy them to a local folder.

You can also set (start and stop) via the dialog the stored procedure that creates reports on 4D Server. (same as if executing aa4D_NP_Schedule_Reports_Server).

aa4D_NP_Schedule_Reports_Server (secured)

aa4D_NP_Schedule_Reports_Server ({DelayBetween}; {ContentMode}; {CommentPointer}; {Nb processes list})

Parameter	Type		Description
DelayBetween	Real	→	Define in minute(s) the delay between two reports
ContentMode	Longint	→	Define the mode of the description of report (for tables) Default is to describe all if this parameter is omitted
CommentPointer	Pointer	→	Pointer to a text from the Host DB to be included in the report
Nb processes list	Longint	→	Define the number of processes to list in the report

Description

The **aa4D_NP_Schedule_Reports_Server** command allows you to manage a Stored procedure on 4D Server, that will create a new report every N minutes (or tenth of minutes if <1), so you keep a trace of the usage of your 4D Server.

Later on, you can check the evolution of this usage by parsing the reports via the Compare dialog (called by the shared method: aa4D_NP_Report_Compare_Display)

If you pass no parameter, the default value of 5 minutes will be used.

To stop the stored procedure, just pass 0 as the first parameter when calling this method.

aa4D_NP_Util_CreateReport

aa4D_NP_Util_CreateReport ({FacelessMode}; {ContentMode}; {CommentPointer}; {Nb processes list})

Parameter	Type		Description
FacelessMode	Longint	→	If >0, hide the dialog after creating the report.
ContentMode	Longint	→	Define the mode of the description of report (for tables) Default is to describe all if this parameter is omitted
CommentPointer	Pointer	→	Pointer to a text from the Host DB to be included in the report
Nb processes list*	Longint	→	Define the number of processes to list in the report

Description

The **aa4D_NP_Util_CreateReport** command is the main command of this component:

This is the one you will ask your customers to use when you need to get a picture of the configuration, settings and volume of the database.

If you ask to execute it on 4D Server, there will be no dialog indicating when it is done, but waiting for a few minutes will let you be sure that the report was created on the Server.

If executed in Local Mode or running Stand-Alone, a dialog will be displayed with the content of the report: from this dialog, you will be able to access the created text file(s).

The first time a report is created:

- A folder named "Folder_Reports" will be created next to the Data file (if it does not exist)
- Some Launch External process will parse some information about the computer
- A document "Array_Profiler.txt" will be created in the "Folder_Reports", with some constant values retrieved (only the first time after starting the 4D Application).

(See a large report content example later in the documentation).

Please pay some attention at the Attention section of the report: we try to remind here what should be noticed (and in some case changed).

Note: Nb processes list*: pass -1 for all tasks (up to the last 4000 current active ones).

aa4D_NP_Util_CreateReport_Serv

aa4D_NP_Util_CreateReport_Serv ({FacelessMode}; {ContentMode}; {CommentPointer}; {Nb processes list})

Parameter	Type		Description
FacelessMode	Longint	→	If >0, hide the dialog after creating the report.
ContentMode	Longint	→	Define the mode of the description of report (for tables) Default is to describe all if this parameter is omitted
CommentPointer	Pointer	→	Pointer to a text from the Host DB to be included in the report
Nb processes list	Longint	→	Define the number of processes to list in the report

Description

The **aa4D_NP_Util_CreateReport_Serv** command is very similar to the **aa4D_NP_Util_CreateReport** command.

The only difference is that it will create the report on 4D Server when executed in Remote application.

Note: In a Server / Remote configuration, no dialog will display on 4D Server when executed.

Shared methods since version 3.2:

aa4D_M_Folder_Rep_SetPath_local (invisible / secured)

aa4D_M_Folder_Rep_SetPath_local (vp_FolderNewPath)

Parameter	Type		Description
vp_FolderNewPath	Pointer	→	Point to the text variable containing the Full pathname of the new "Folder_reports" to be used. After modifying the "Folder_Reports" location, \$1-> will contain the new Full pathname
Vb_OK	Boolean	←	No error was detected for this operation: \$0=True.

Description

The **aa4D_M_Folder_Rep_SetPath_local** function let you set a new location for the "Folder_Reports". (instead of it's default location next to the Data file).

This method must be called after each restart of the Host database to not use the default location.

Example Host method code to use this shared method:

```
// Method: aa4D_M_Host_Set_folder_Path
// Use the shared (but invisible) method "aa4D_M_Folder_Rep_SetPath_local"
// of the component 4D_Info_Report_v4

ARRAY TEXT($at_Components;0)
COMPONENT LIST($at_Components)
If (Find in array($at_Components;"4D_Info_Report@")>0) // the component is loaded

    C_BOOLEAN($vb_FolderPath_OK)
    C_TEXT($vt_Folder_Pathname)

    // set a path for the folder to create or select (on Windows in this example).
    $vt_Folder_Pathname:="C:\\Component Testing\\Subfolder\\Another folder\\"

    EXECUTE METHOD("aa4D_M_Folder_Rep_SetPath_local";$vb_FolderPath_OK;->$vt_Folder_Pathname)
    // Create or select a local folder for next reports
    // ($vt_Folder_Pathname now contains the path in the 4D remote
    // that will be used for next report creation.

    If (Test path name($vt_Folder_Pathname)=0) // this folder now exists
        CONFIRM("Create a report in this Local folder?\n"+$vt_Folder_Pathname)
        // to be commented, just to check the name of the folder
        // Keep this path if you need to check it's content or reuse it later.

        // ---- to Test the creation of a report in this Folder Pathname ----
        If (OK=1)
            EXECUTE METHOD("aa4D_NP_Util_CreateReport";*;1)
            // Create a local report in the selected folder(1=without interface display)
        End if
    End if
End if
```


aa4D_M_Folder_Rep_GetPath_local (invisible / secured)

aa4D_M_Folder_Rep_GetPath_local (vp_FolderPath)

Parameter	Type		Description
vp_FolderPath	Pointer	→	Point to the text variable that will get the Full pathname of the current "Folder_reports". \$1-> will contain the current Full pathname
Vb_OK	Boolean	←	No error was detected for this operation: \$0=True.

Description

The [aa4D_M_Folder_Rep_GetPath_local](#) function let you get the current location of "Folder_Reports". (it's default location is next to the Data file. if the Folder_Reports did not exist, it will be created).

aa4D_M_Get_Component_Version (New in v4.2)

aa4D_M_Get_Component_Version (vp_TextResult)

Parameter	Type		Description
vp_TextResult	Pointer	→	Pointer to the text variable (or field) to get the result.

Description

The [aa4D_M_Get_Component_Version](#) function let you get the current version of the component 4D_Info_Report_v4 more explicit name.

Example of content in the pointed variable after executing this shared method: "4.2 (2014-03-20)".

aa4D_M_Get_Expiration_Date (New in v4.2) (invisible)

aa4D_M_Get_Expiration_Date (vp_Pointer_Result)

Parameter	Type		Description
vp_Pointer_Result	Pointer	→	Pointer to the Date variable (or field) to get the result. or
vp_Pointer_Result	Pointer	→	Pointer to the text variable (or field) to get the result.

Description

The [aa4D_M_Get_Expiration_Date](#) function let you get the current expiration date of the 4D_Info_Report_v4 current version for your current 4D version.

Note: There is no Time-bomb for the versions 4D v12.x

If the passed pointer is pointing a Date, it will get the Date of the Expiration (or a nil date)

If the passed pointer is pointing a Text or a string, it will get the following result:

"Expiration date of 4D_Info_Report (YYYY_MM_DD):\r\n"+"2018_01_01"

or

"(No time-bomb for this version of 4D)" // for v12 currently

or an ALERT (not on 4D Server) if there is a wrong pointer:

"Error: Invalid pointer to a text or a string:\r\naa4D_M_Get_Expiration_Date(->xxx)"

aa4D_M_Get_Information_Text

aa4D_M_Get_Information_Text (vt_Selector;vp_TextResult)

Parameter	Type		Description
vt_Selector	Text	→	Selector for the kind of expected result (explicit or number)
vp_TextResult	Pointer	→	Pointer to the text variable (or field) to get the result.

Description

The **aa4D_M_Get_Information_Text** function let you get the current text value depending of the selector. vt_Selector is a text parameter: il can contains either the string of a numeric integer, or a more explicit name. (if no Array_profiler.txt & report were created, these results remains available). (these text results are provided from 4D Server if called from 4D Remote).

"Array_Profiler"	or «0»	→	Get the content of the Array_profiler.txt
"Manufacturer"	or «1»	→	Get the Manufacturer in the Array_profiler.txt
"Model Identifier"	or «2»	→	Get the Model Identifier in the Array_profiler.txt
"Memory"	or «3»	→	Get the RAM value (MB) in the Array_profiler.txt
"System"	or «4»	→	Get the System description in the Array_profiler.txt
"Nb Processors"	or «5»	→	Get the Number of Processors in the Array_profiler.txt
"Processor Name"	or «6»	→	Get the Processor Name in the Array_profiler.txt
"CPU Speed"	or «7»	→	Get the CPU Speed in the Array_profiler.txt
"Total Nb Cores"	or «8»	→	Get the Total number of Cores in the Array_profiler.txt
"Nb CPU threads"	or «9»	→	Get the Total number of CPU threads in the Array_profiler.txt
"System Web Browser"	or «10»	→	Get the Default System Web Browser in the Array_profiler.txt
"4D Internal Build"	or «11»	→	Get the version and build of 4D in the Array_profiler.txt
"More infos 4D release"	or «12»	→	Get More info on 4D (if available) in the Array_profiler.txt
"Memory modules"	or «13»	→	Get the description of the Memory modules
"Disk"	or «14»	→	Get the description of the Disk
"Volume"	or «15»	→	Get the description of the Volume
"Disk & Volume"	or «16»	→	Get the description of Disk and Volume
"Plugin_List"	or «17»	→	Get the list of Plugin versions
"Component_List"	or «18»	→	Get the list of Component versions
"MainUUID"	or «19»	→	Get the Data file Main UUID
"Cache Settings"	or «20»	→	Get the Cache settings (new in v4.5)
"Scheduler_SP_Minutes"	or «21»	→	Get the number of minutes set for the stored procedure (= 0 when run in stand-alone. new in v4.7)
"Agent_SP_Seconds"	or «22»	→	Get the number of minutes set for the stored procedure That returns JSON results (=0 when run in stand-alone)
"Attention_file"	or «23»	→	Get the text content of the Attention_report.txt file.
"Cache_Memory_values"	or «24»	→	Get More info on 4D (if available) in the Array_profiler.txt
"IPv4"	or «25»	→	Get More IPv4 and Web Services of the 4D Server

aa4D_NP_Report_Export_Display (secured)

aa4D_NP_Report_Export_Display

Description

The **aa4D_NP_Report_Export_Display** command allows you to display a dialog to select what you want to display in a compare dialog or what you want to export in your remote computer.

aa4D_NP_Reports_Max_Set_Limit (invisible / secured)

aa4D_NP_Reports_Max_Set_Limit ({Max_Nb_of_reports})

Parameter	Type	Description
Max_Nb_of_reports	Longint	→ Maximum number of reports to keep in the "Folder_reports". This value must be set again before each creation of a first report after starting 4D or 4D Server

Description

The [aa4D_NP_Reports_Max_Set_Limit](#) command fix the maximum number of last reports to be kept in the folder "Folder_Reports" of the database.

If no value is passed as parameter, the interprocess variable handling this value on 4D Server will be set to 0. This value is taken care of after each creation of a report.

If the limit of number of reports is reached, the oldest reports will be deleted.

Be aware that the reduction of existing reports to the new limit is not done at once: only when creating new reports, this new limit is taken care of, and the number of reports will be reduced by 10 (maximum) after each creation of reports. You can manage the original reduction of reports yourself.

If aa4D_NP_Reports_Max_Set_Limit was not used before, the default limit is 0, and thus ignored by the component. When used, this limit must be set after every restart of the application.

aa4D_NP_Reports_Max_Get_Limit (invisible)

aa4D_NP_Reports_Max_Set_Limit

Result	Type	Description
Max_Nb_of_reports	Longint	← Maximum number of reports in the interprocess variable handling this limitation of reports in "Folder_reports".

Description

The [aa4D_NP_Reports_Max_Get_Limit](#) command retrieve the maximum number of last reports to be kept in the folder "Folder_Reports" of the database.

If the returned default limit is 0, then there is no limitation set (via aa4D_NP_Reports_Max_Set_Limit)

Shared methods since version 4.7:

aa4D_NP_Agent_Schedule_Call (invisible)

aa4D_NP_Agent_Schedule_Call

Parameter	Type	Description
Nb_of_seconds	Longint	→ Number of seconds to retrieve values in memory in the stored procedure (answer in JSON, pass 0 to stop it)

Description

The [aa4D_NP_Agent_Schedule_Call](#) command set the number of seconds for the stored procedure that calculate values without creating reports, and return a JSON answer.

aa4D_M_Agent_Get_Records_Call (invisible)

aa4D_M_Agent_Get_Records_Call

Parameter	Type	Description
Vt_format_answer	Text	→ "JSON" or "XML" (optional, default is JSON)
SQL_time_min	Longint	→ Minimal SQL timestamp to retrieve results (optional)
SQL_time_max	Longint	→ Maximal SQL timestamp to retrieve results (optional)

Result	Type	Description
Content result	Text	← Answer in JSON (or XML) containing the results for the SQL time limits.

Description

The [aa4D_NP_Agent_Get_Records_Call](#) command can retrieve calculated results (using a buffer of the last 100 values) depending of the criteria.

Note: The stored procedure must be activated first ([aa4D_NP_Agent_Schedule_Call](#))

Note for the shared methods description:

"(secured)" next to the name of the shared method of this component means that if a Host database method named "aa4D_M_Host-Allow_Report_access" exists (and is shared), it will be called to allow the execution of the shared method (see page 40 of the documentation for the code to be used in this Host method).

This method "aa4D_M_Host-Allow_Report_access" is part of the Host template database that is available with this component.

Description of the main parameter for the creation of reports:

(For all shared methods that create a report, most often the second (optional) parameter).

- ContentMode Value (Type: **Longint**) Define the mode of the description of report (for tables)

Main parameter value for the creation of reports:	-2	-1	0	1	2	3	4	5	6	7	8	9
Count total number of records...	X	X	X	X	X	X	X	X	X	X	X	X
except for invisible tables	X				X	X	X	X	X			
Hide Table List	X	X										
(Default if parameter omitted) Show all tables list, with all structure details			X									
Show Table List:			X	X	X	X	X	X	X	X	X	X
Show Table number			X	X	X	X	X	X	X	X	X	X
Show Table name			X	X	X	X	X	X	X	X	X	X
Show number of records of the table			X	X	X	X	X	X	X	X	X	X
Show number of indexed fields			X			X	X	X	X	X	X	X
Show number of subtable (if any)			X			X		X			X	
Show triggers			X			X	X			X		

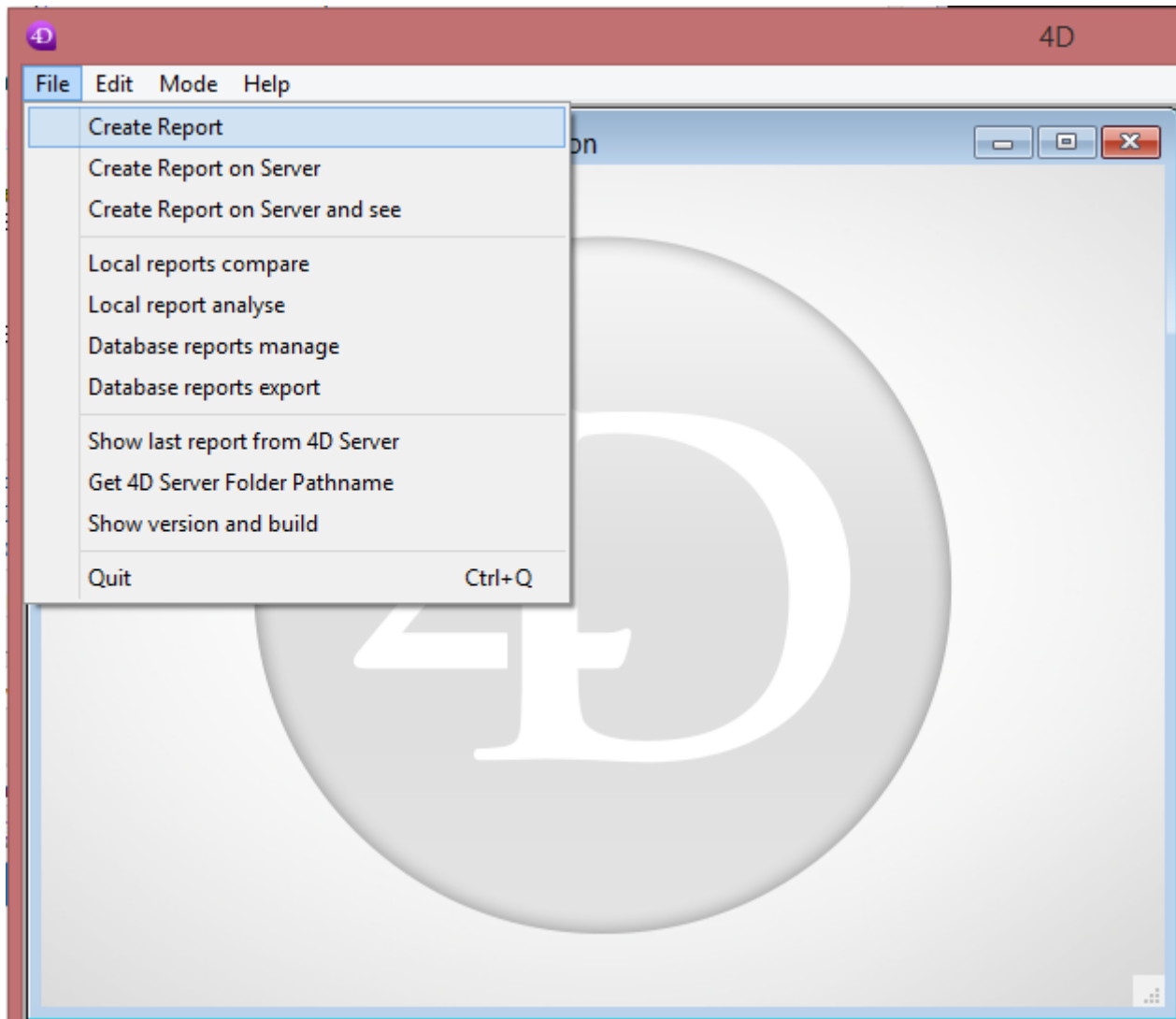
(For all shared methods that create a report, most often the last (optional) parameter).

- Nb processes list (Type: **Longint**) Define the number of processes to list in the report

(Pass the value -1 to list all the (alive) processes running on 4D / 4D Server.

Discover the Component run in stand-Alone:

(Opening directly the component database, and creating once a requested data file).



Using the custom menu when run in Stand-Alone, you can check how it works:

First section of the File menu:

Creation of one report

- **Create Report** will execute the shared method : "aa4D NP Util CreateReport" (without parameter)
generating after a few seconds a new report displayed inside a dialog:
- **Create Report on Server** (When executed on 4D in Remote mode), will execute the shared method : "aa4D NP Util CreateReport Serv", that will generate a new report on 4D Server (no warning on 4D Remote when the report is created).
(When executed in local mode, will behave like **Create Report**)
- **Create Report on Server and See** (When executed on 4D in Remote mode), will execute the shared method : "aa4D M Report CreateOnServerSee" (without parameter)
(generating after a few seconds a new report on the Server and displayed inside a dialog)

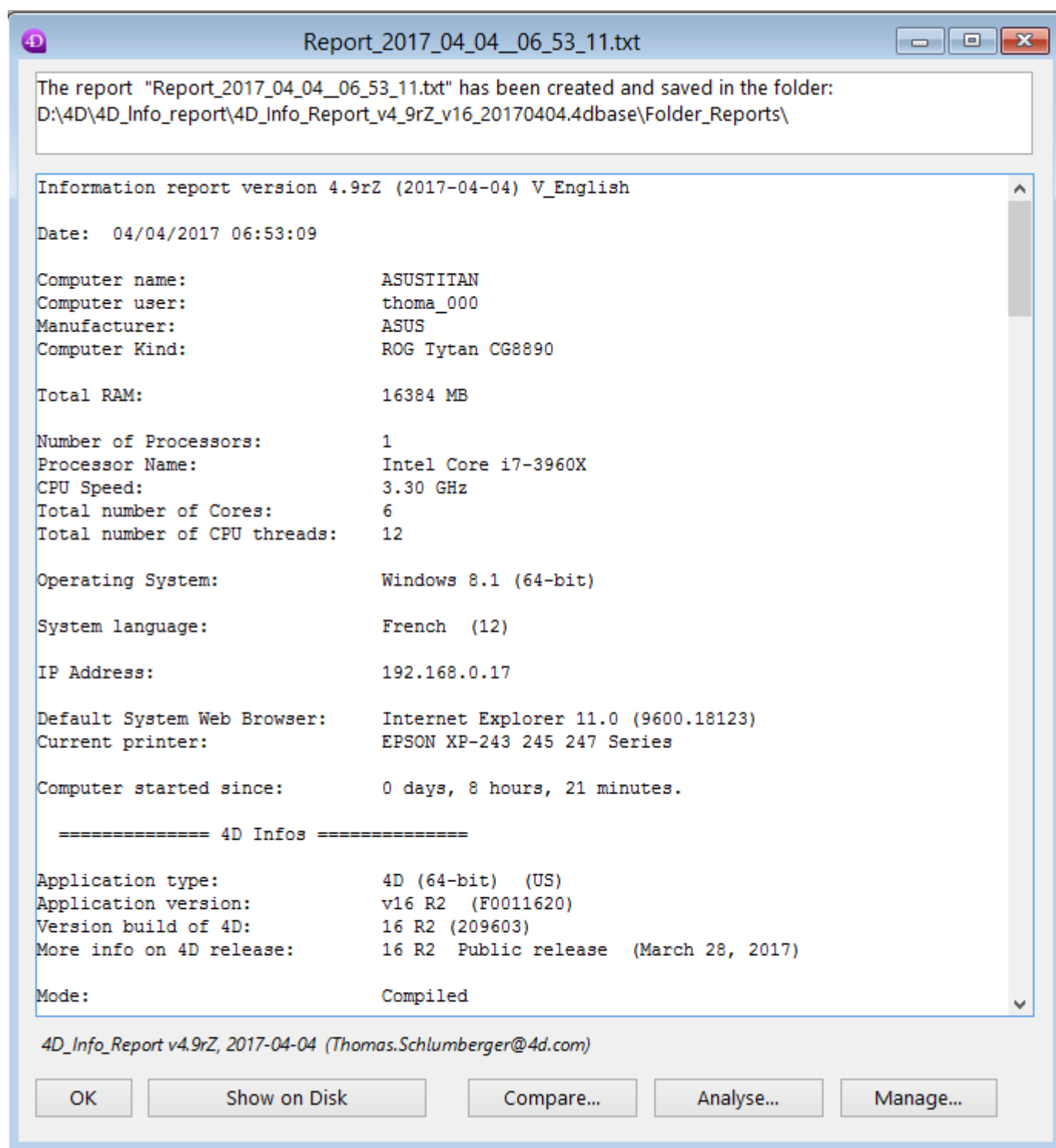
Second section of the File menu:

- **Local reports compare** will execute the shared method :
"aa4D NP Report Compare Display" that display a dialog (see captures in next page)
- **Local reports analyse** will execute the shared method :
"aa4D NP Report Analyse Display" that display another dialog to compare 2 or 3 reports.
- **Database reports manage** will execute the shared method :
"aa4D NP Report Manage Display" that display new dialog to set the stored procedure, or import reports from 4D Server.
- **Database reports export** will execute the shared method :
"aa4D NP Report Export Display" that display a new dialog to set criteria for reports to be compared in the reports main values, and also to import exported files (later at work).

Third section of the File menu:

-
- **Show last report from 4D Server** will execute the shared method:
"aa4D NP Get Last Server Report" that will display in an alert the last generated report on 4D Server (When executed on 4D in Remote mode), or on 4D in Stand-Alone mode.
- **Get 4D Server Folder Pathname** will display in an alert the full path of the folder
"Folder_Reports" (located next to the (Host) data file).
(As there is currently no way to change this location or the name of the current reports folder, no need to use a shared method to retrieve it: in you Host database code, just complete the parent Folder of the data file with "Folder_Reports".)
- **Show version and build** will execute the shared method : "aa4D M Get Build 4D Text call" that will let you select in a dialog a 4D application or a plugin bundle, and display in an alert:
 - the version of the file,
 - the build number of the file (if it exists)
 - More info (public versions) corresponding to the build number and version,
 - the full path of the selected file.

Executing the shared method "aa4D NP Util CreateReport" or "aa4D M Report CreateOnServerSee", after a few seconds (the first time), a new dialog is displayed including the generated report content:



When in Local mode, the "Show on Disk" button will open the content of the "Folder_Reports", with the last report selected.

The upper part of the dialog will indicate where the created report is saved.

When in Remote mode, if the report was not generated locally, then you can select a local folder to save the imported reports (via the "Save in Local Reports Folder" button). If this folder is not already set on 4D remote, a dialog will let you set one.

The "Analyse" button will display a new dialog, allowing a simple comparison with one or two other reports. (use the shared method "aa4D_NP_Report_Analyse_Display").

Compare records P_5							
Report file name		Total Nb records	Appli Type	4D Version			
Open...	Close	Report_2015_04_18_20_12_02.txt	6 744 114	4D Server	14.3 (183401)		
Open...	Close	Report_2017_02_15_23_47_09.txt	10 250 080	4D Server	15.4 (208269)		
Open...	Close						
Num	Table Name	Nb Fields	Nb Index	Nb records 1	Nb records 2	Diff_rec_1_2	Diff_%_1_2
12	Konstanten	147	1	0	1	1	9 999.9 %
71	ArtikelKunde	16	6	14	81	67	478.6 %
5	Hilfe	7	5	108	468	360	333.3 %
2	Rapport	42	5	1 222	3 652	2 430	198.8 %
18	Einheit	14	1	44	96	52	118.2 %
15	Artikelgruppe	13	3	17	37	20	117.6 %
3	Projekte	43	3	5	10	5	100.0 %
16	Sprache	119	1	2	4	2	100.0 %
80	AdressenInfo	10	4	1 249	2 449	1 200	96.1 %
70	ArtikelLief	19	5	5 221	9 821	4 600	88.1 %
11	Anrede	7	1	5	9	4	80.0 %
81	AdressenSachb	25	6	22 304	35 335	13 031	58.4 %

Note: if a Diff_% value displays as "9999.9 %", this mean infinite diff. (there was no record before).

The "Manage" button will display a new dialog, allowing (in Remote mode) to import reports stored on 4D Server, and set the stored procedure that create a new report every N minutes:

Start

aa4D_NP_Schedule_Reports_Server
(Local stored procedure)

Create a report every:

5

minute(s)

of type:

0

Processes listed:

0

(-1 = All (alive) processes)

Local folder:

E:\4D\4D_Info_report\Local_Folders\Last_Report\

Show list of the database reports

Nb. reports in the 184 / 184

Show list of stored reports

Report_2016_12_18_10_08_37.txt

Report_2016_12_18_18_10_39.txt

Report_2016_12_18_18_11_17.txt

Report_2017_02_01_19_08_50.txt

Report_2017_03_02_06_47_52.txt

Report_2017_03_31_11_24_43.txt

Report_2017_04_04_02_39_46.txt

Report_2017_04_04_02_39_59.txt

Report_2017_04_04_02_40_10.txt

Report_2017_04_04_04_18_04.txt

Report_2017_04_04_04_18_20.txt

Report_2017_04_04_06_51_56.txt

Report_2017_04_04_06_53_11.txt

Import Selection

Import Missings

Delete Selection

Open Local Reports Folder

Select local folder

Nb. reports in local folder: 184

Local Reports

Report_2017_04_04_06_53_11.txt

Report_2017_04_04_06_51_56.txt

Report_2017_04_04_04_18_20.txt

Report_2017_04_04_04_18_04.txt

Report_2017_04_04_02_40_10.txt

Report_2017_04_04_02_39_59.txt

Report_2017_04_04_02_39_46.txt

Report_2017_04_04_02_39_59.txt

Report_2017_03_31_11_24_43.txt

Report_2017_03_02_06_47_52.txt

Report_2017_02_01_19_08_50.txt

Report_2016_12_18_18_11_17.txt

Report_2016_12_18_18_10_39.txt

Report_2016_12_18_10_08_37.txt

4D_Info_Report v4.9rZ, 2017-04-04 (Thomas.Schlumberger@4d.com)

When this dialog is displayed: if the stored procedure is not already started, A "Start" button is shown: Click the "Start" button to start the stored procedure on 4D Server (or 4D Stand-Alone now), with the displayed settings. If the stored procedure is already started, the button displayed will be "Stop": Click on it to stop the stored procedure, or apply new settings clicking "Start" again.

4D_Info_Report_v4_9_ Reference

April 4, 2017

page 17 (v23)

The "Compare" button will display a new dialog, allowing to compare the main values of saved reports in this folder. (same as when executing the shared method "aa4D NP Report Compare Display")

When opening the component as a database, you can display this dialog via the menu of the Application window: File/Local reports compare.

As the dialog is opened in a new process, you can handle multiple folders at the same time.

Same dialog display when executing the shared method "aa4D NP Report Compare Display")

4D_Info_Report v4.9rZ, 2017-04-04 [Delete...] [Graph...] [Prefs Graph...]

Select folder: D:\4D\4D_Info_report\4D_Info_Report_v4_9rZ_v16\Folder_Reports\

Nb. Reports		MIN:	DATE-TIME REPORT:	NB USERS	NB TASKS	USER PROCESS	USED CACHE
63		MIN:	April 3, 2017 01:34:09	18	106	63	1 531
		MAX:	April 4, 2017 06:57:53	34	117	112	1 541

Ignore	ID	At	Report Name	DateReport	TimeReport	Users	Tasks	UsersProc
<input checked="" type="checkbox"/>	52	4	Report_2017_04_04_01_29_03.txt	04/04/2017	01:29:03	22	117	73
<input checked="" type="checkbox"/>	53	4	Report_2017_04_04_01_58_56.txt	04/04/2017	01:58:56	22	117	74
<input checked="" type="checkbox"/>	54	4	Report_2017_04_04_02_28_50.txt	04/04/2017	02:28:50	22	117	73
<input checked="" type="checkbox"/>	55	4	Report_2017_04_04_02_58_44.txt	04/04/2017	02:58:44	22	117	73
<input type="checkbox"/>	56	4	Report_2017_04_04_03_28_37.txt	04/04/2017	03:28:37	22	117	75
<input type="checkbox"/>	57	4	Report_2017_04_04_03_58_31.txt	04/04/2017	03:58:31	22	117	73
<input type="checkbox"/>	58	4	Report_2017_04_04_04_28_24.txt	04/04/2017	04:28:24	22	117	74
<input type="checkbox"/>	59	4	Report_2017_04_04_04_58_18.txt	04/04/2017	04:58:18	22	117	74
<input type="checkbox"/>	60	4	Report_2017_04_04_05_28_12.txt	04/04/2017	05:28:12	22	117	74
<input type="checkbox"/>	61	4	Report_2017_04_04_05_58_05.txt	04/04/2017	05:58:05	22	117	73
<input type="checkbox"/>	62	4	Report_2017_04_04_06_27_59.txt	04/04/2017	06:27:59	22	117	77
<input type="checkbox"/>	63	4	Report_2017_04_04_06_57_53.txt	04/04/2017	06:57:53	23	117	74

- If you click one or more "Ignore" checkbox, the corresponding report will be ignored in the MINIMUM and MAXIMUM values calculation, and also when choosing the "Graph..." button.

(To modify many "Ignore" checkboxes at once, just set a selection of Row in the List Box, and click one of the corresponding "Ignore" checkbox to invert the state in the selection).

You can also double-click one row of the List Box to open directly the report.

(If you have parsed a folder, or if the Import of the Blob was done without changing the location)

Next to the "Select folder" button, there is a text variable where you can input or paste the path of the folder to parse.

You can also drop a folder (containing reports) over it to input the path name of this folder.

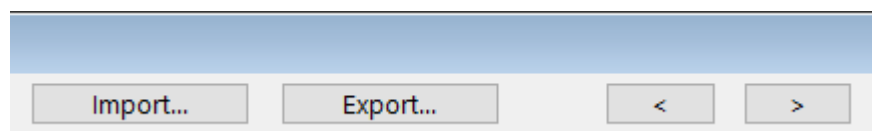
The "Select folder" button is used to select a folder or another folder containing reports created by the component.

If there is a valid folder path already entered next to the button, you will get this confirm dialog:

"Do you want to select another report folder?"

Clicking "Cancel" will force the parsing of the current folder, clicking "OK" will let you navigate to select another folder, and after confirmation of the selected folder, will parse this new folder.

When a folder is selected, a parsing of the reports inside this folder will populate arrays in the List box about the usage of the database. If some reports contains "Attention :" items, the column "At" will show how many are found. If any is found in each report (in the Attention section), and a red line is shown above the column.



The "Import" button allows to import a Blob generated via the Export button, that only contains the values used in the List Box.

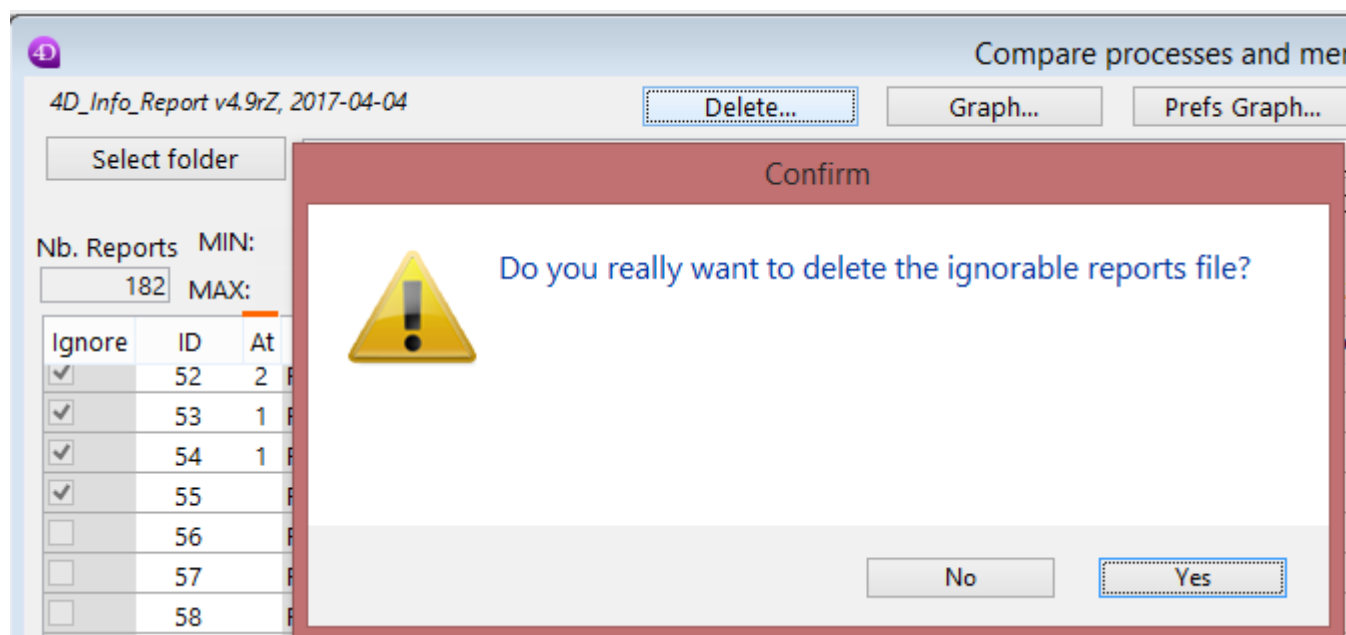
It is much faster than processing all the reports again when there are many reports in the folder, and if the original folder is still in it's original location, you will be able to open the reports via a double-click in the List box as if you have parsed the folder again.

The "Export" button allows to select a format and a name to export the content of these arrays. You can for example:

- Generate a Blob (Compare_Blob) that can be later imported to populate the List Box,
- Generate an Excel document (.xml format compatible with Excel 2003 or later),
- Generate an XML document (.xml),
- Generate a text file document (.txt).

The " < " and " > " buttons allow to resize the width and height of the dialog.

The "Delete..." button is visible if at least one checkbox "Ignore" has been set:

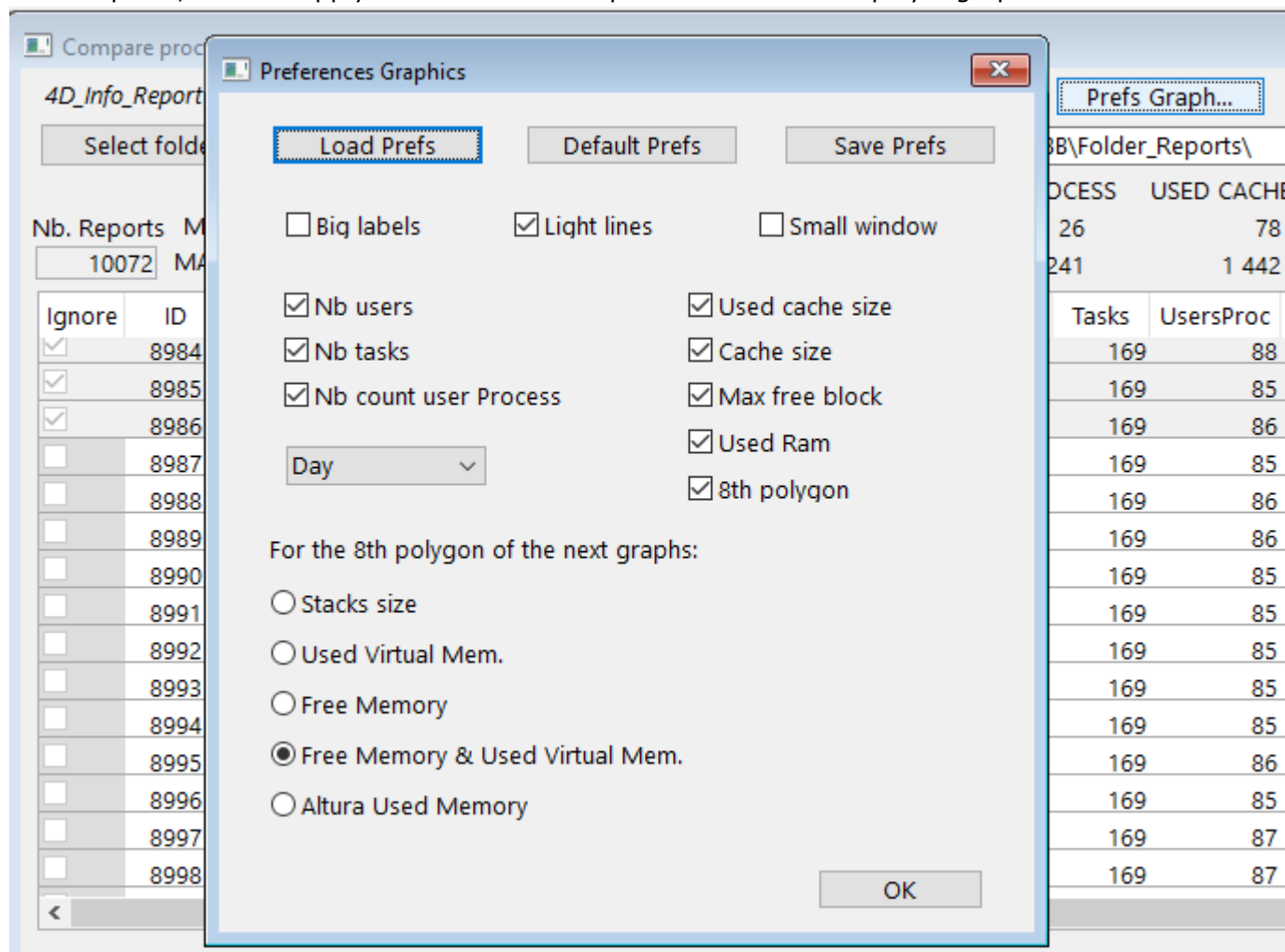


Attention: If you click "Yes", the reports with "Ignore" checkbox checked will be deleted from the parsed folder.

Note: this feature also works on 4D remote, as long as the location of the folder of reports is local (not on 4D Server).

There are two lines (MIN and MAX) above the List box: These 2 lines shows the Minimum and Maximum values found in non ignored reports for each type of information.

The "Prefs Graph..." allows to set the default display settings of a graph: these settings are stored on the computer, and will apply each time the component is used to display a graphic:



It can be: the Stack size (sum of the Stack size, default value) or you can choose to display in the graph (as the seventh polygon) for example the variation of the Used Virtual Mem (Memory), Or Free Memory (in RAM).

Note: when the Graph window is displayed, via the button "Update SVG", if you use Shift down or Right click, the seventh polygon will alternate among three of these kind of values.

If you create reports with 4D_Info_Report v4 under Windows, Some values will be retrieved and added in the reports: Altura Allocated & Used memory. These recent values are recorded at the end of this section of the reports:

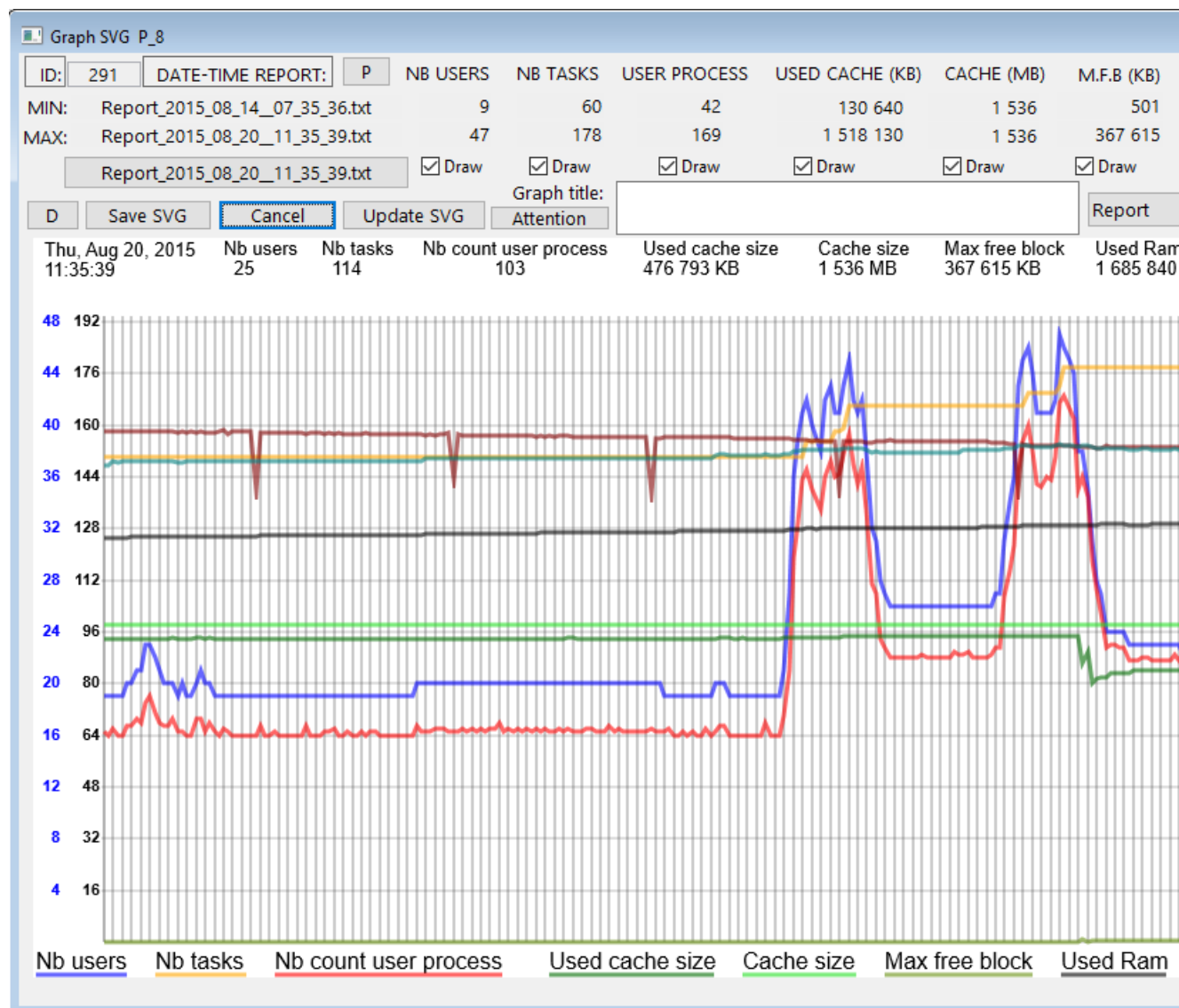
----- other infos via GET MEMORY STATISTICS: -----

```
Used cache Size :           3 247 KB
Free Memory :             5 343 852 KB
Used physical memory :     127 928 KB
Used virtual memory :     557 980 KB
Size of stacks :          3 692 KB
Altura Allocated Memory :   2 797 KB (*)
Altura Used Memory :       1 166 KB (*)
Biggest free block in cache : 25 599 KB
```

(*): On Windows only, since latest 4D v12 release, no more on v16 64-bit

The "Graph..." button will display a new dialog with an SVG graph displaying the values of the arrays. If you click this button with Shift down or Right click, you force the display of a new Graph dialog. (If Alt (Option on OS X) is down, this graph dialog will be condensed vertically).

Otherwise, if a related Graph dialog already exists for the compare dialog, it will be brought to front:

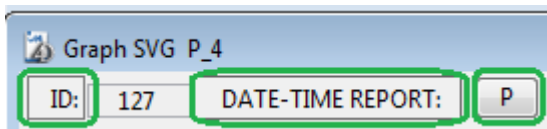


Use the "Draw" checkboxes to display or hide the polygon of the corresponding kind of values.

If you click the button "Update SVG", you will force the redraw of the SVG area. (if a Graph title is entered (two lines allowed), it will be added in the top of the Graph).

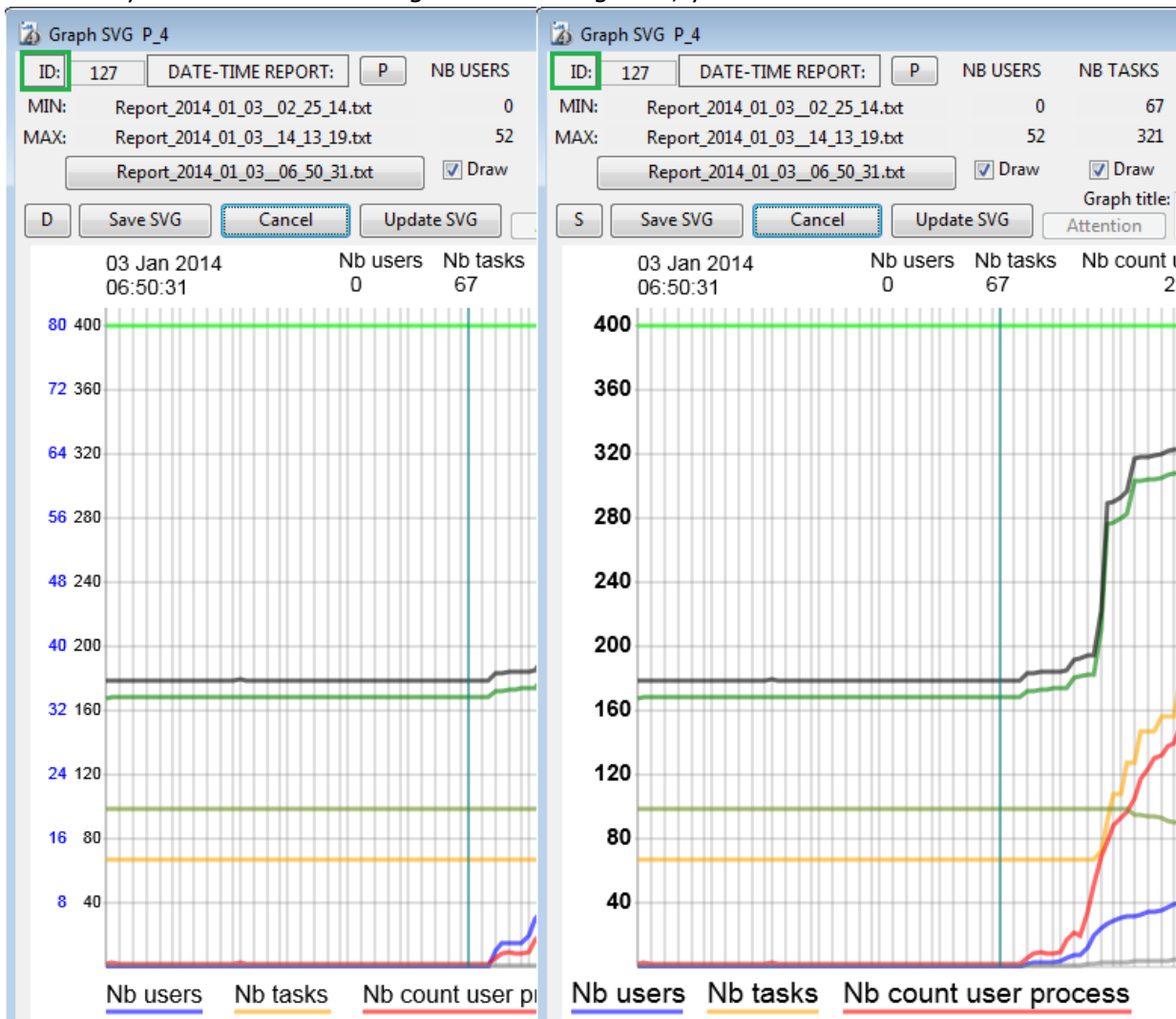
If Shift down is pressed (or Right click) while clicking, you will rotate possible values for the 8 polygon:

Default: Stacks, or Used Virtual Memory, or Free Memory (see the Prefs Graph... settings, page 20).

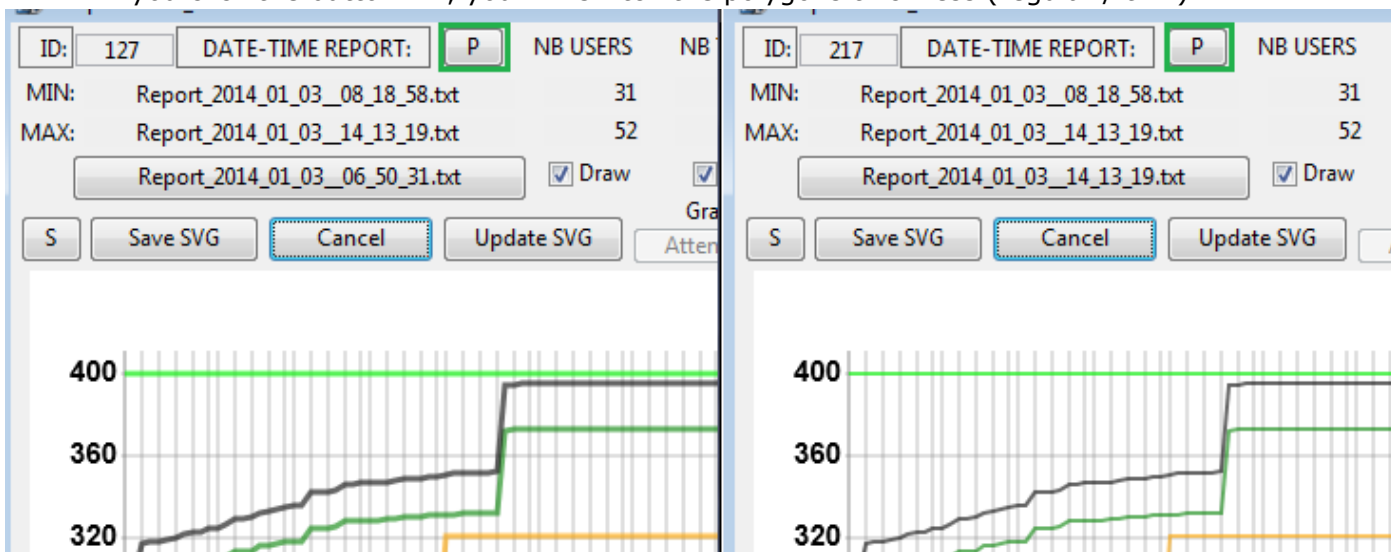


NEW IN VERSION 4: 3 new buttons in the Graph dialog

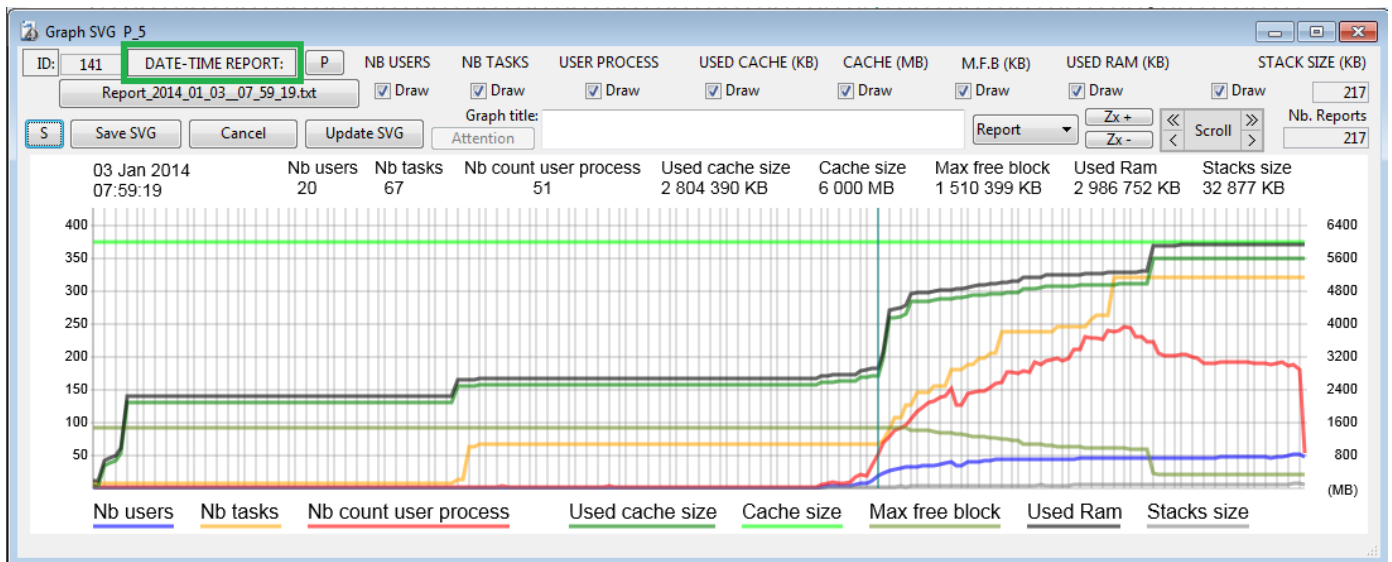
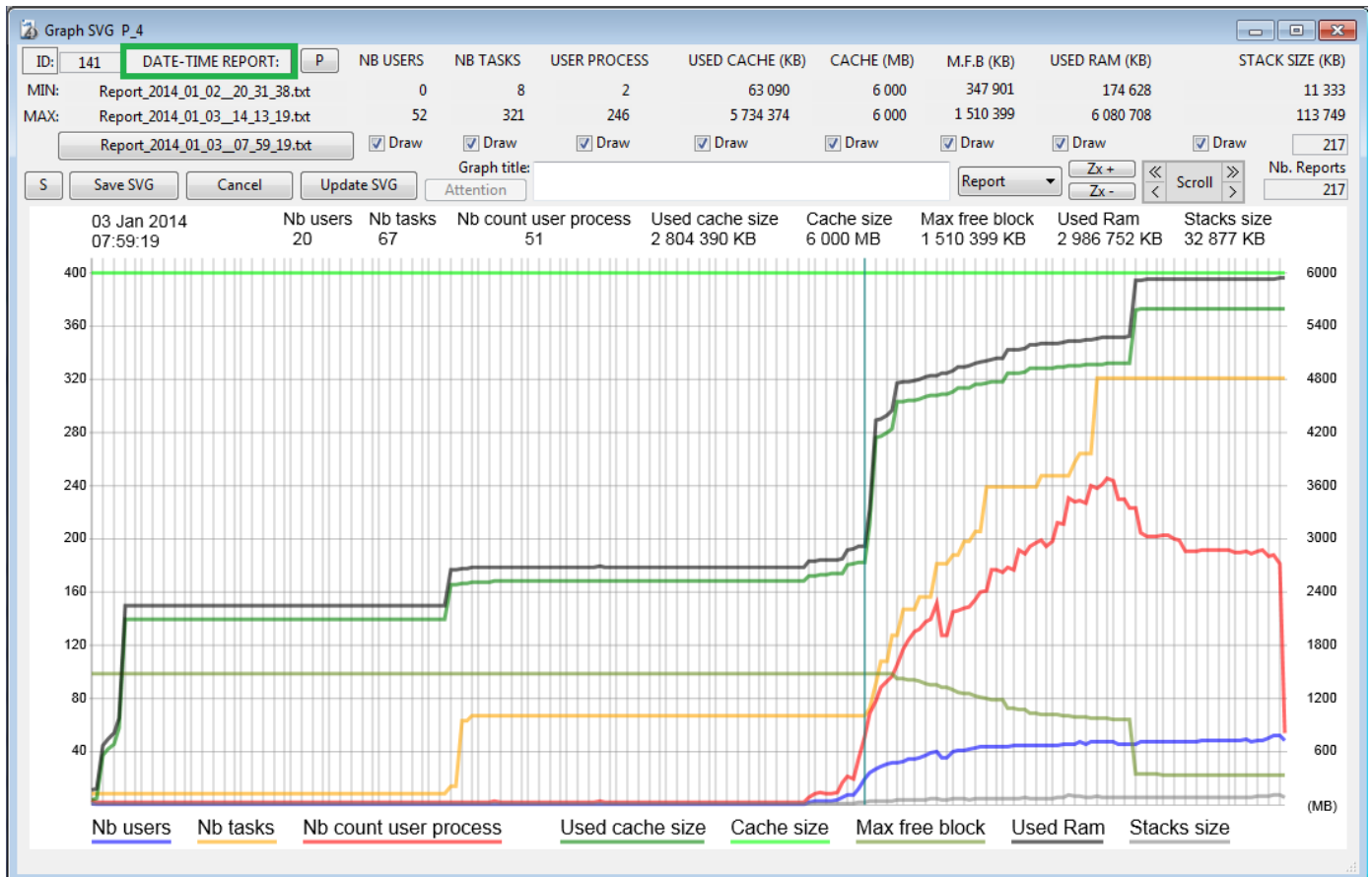
- If you click in the rectangle surrounding "ID", you will switch the size of labels and marks:



- If you click the button "P", you will switch the polygons thickness (regular / thin):



- If you click in the rectangle surrounding "DATE-TIME REPORT:", you will switch the vertical size of the Graph dialog, from standard (with MIN and MAX values visible), to condensed:



Note: this size switch via this button is convenient to force a proportional resizing of this window.

To zoom on a selection of the Graph: click twice to set the left limit, and right click (or Shift click) to set the right limit: the new selection will be highlighted, the Min and Max recalculated, and the zoom applied.

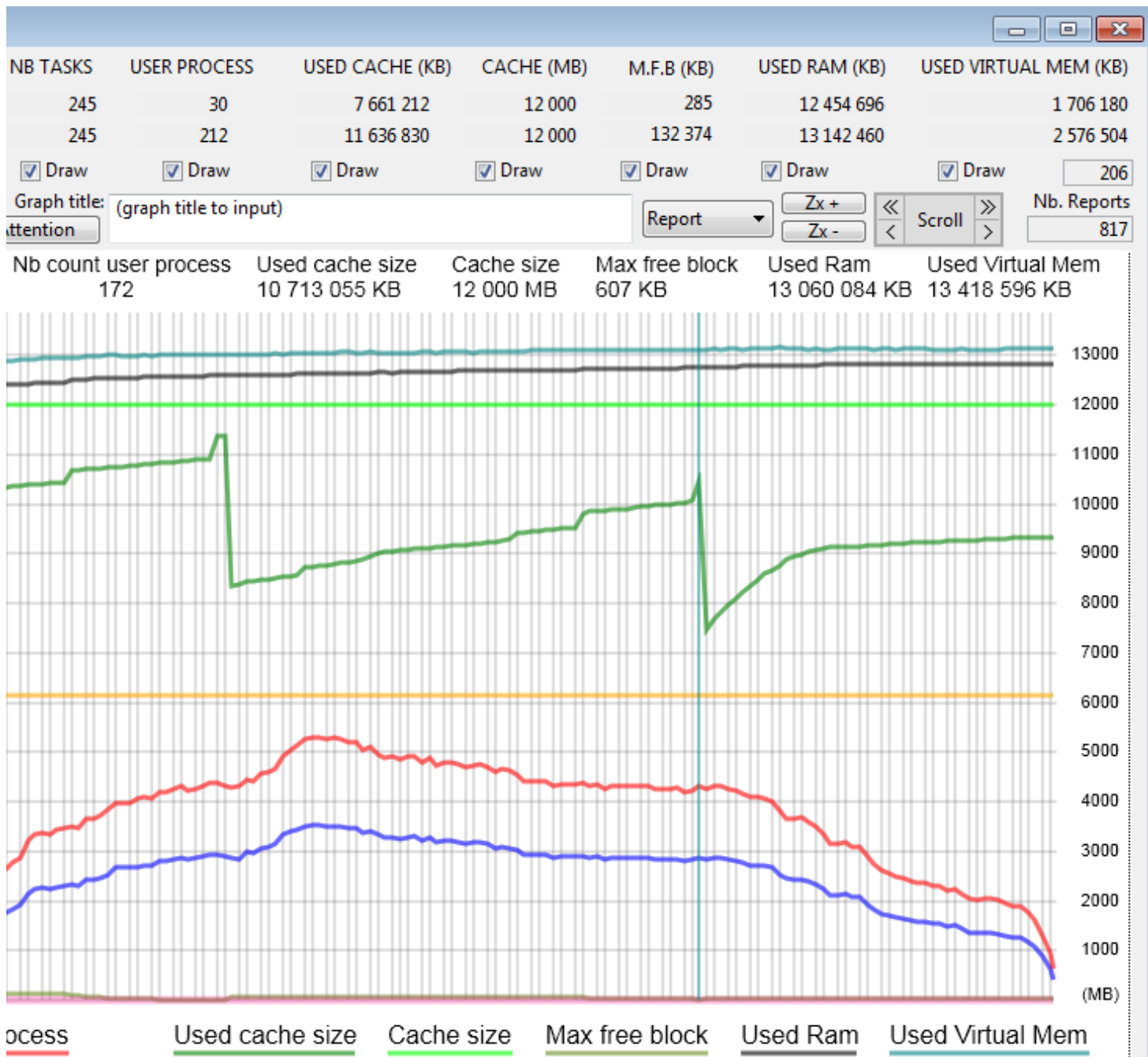
To un-zoom, hit the Delete key. (you can cumulate zoom action).

If you click the "Save SVG" button, you will save the graph in an SVG document.

There are two kinds of values:

- the first one (on the left) is relating to : the number of users, tasks, and user process.
- the second one (on the right) correspond to the memory usage, and is relating to: the used cache size, the cache size, the Used ram, the total stack size. (scaled in the right of the graph).

GRAPH INTERACTION:



If you click in the graph, the nearest report will be highlighted, and the values of this report will be displayed in the top of the graph: then just move the mouse X on the left or the right to get these values updated. if the vertical line is red, there is Attention item(s) in the corresponding report.

If the vertical line is blue, there is no Attention counted in the report.

If the vertical line is red, there is at least one attention in the report.

(to escape this mode, move the X position left or right of the graph part displaying the polygons).

If you click again, the selected report will be frozen, until you click again in another position.

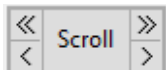
If you double-click, the corresponding report content will be displayed in a reduced window.
(when a report name is displayed under the MIN(IMUM) and MAX(IMUM) report name, you can click the button containing it's name to open this report).

You can also open the selected report via Command/Ctrl Shift O, or just click the button containing the name of the current selected report.

Precise selection of a report in the graph:

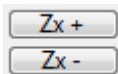
To be sure to move to each report next to the blue (or red) vertical line, you can use the Keyboard Left Arrow Key or the Right Arrow key.

Scrolling in the graph:



(Double arrow sign will speed-up the scroll compared to single arrow sign).

Zooming in the graph:



With these two buttons, you can increase or decrease the scale of the X values.

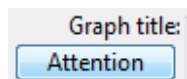
- with Zx +, you will scale the X values up to ratio 256

- with Zx -, you will scale down the X values to a quarter of the default scale:

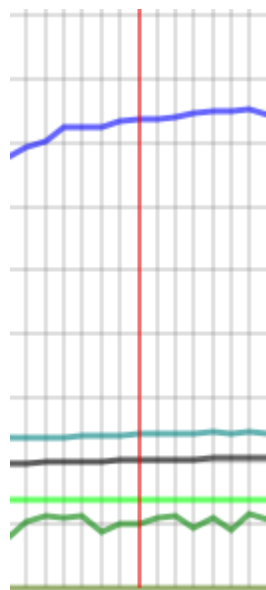
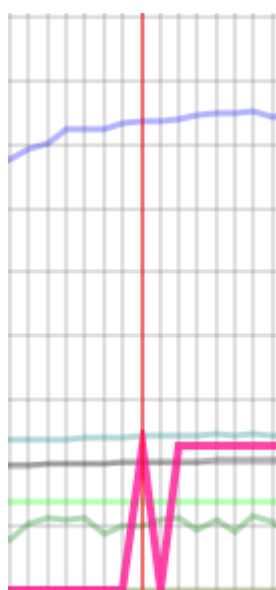
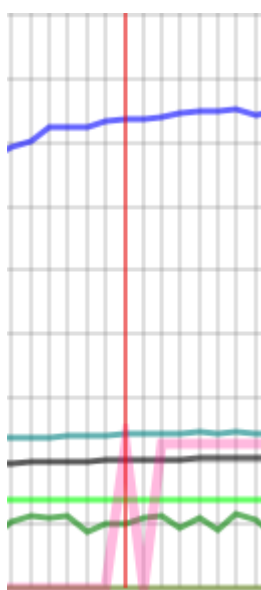
(clicking again this button will switch the scale to half, quarter and scale 1).

You can also scale up or down via the keyboard Up Arrow Key and Down Arrow Key.

===== Attention section highlight =====



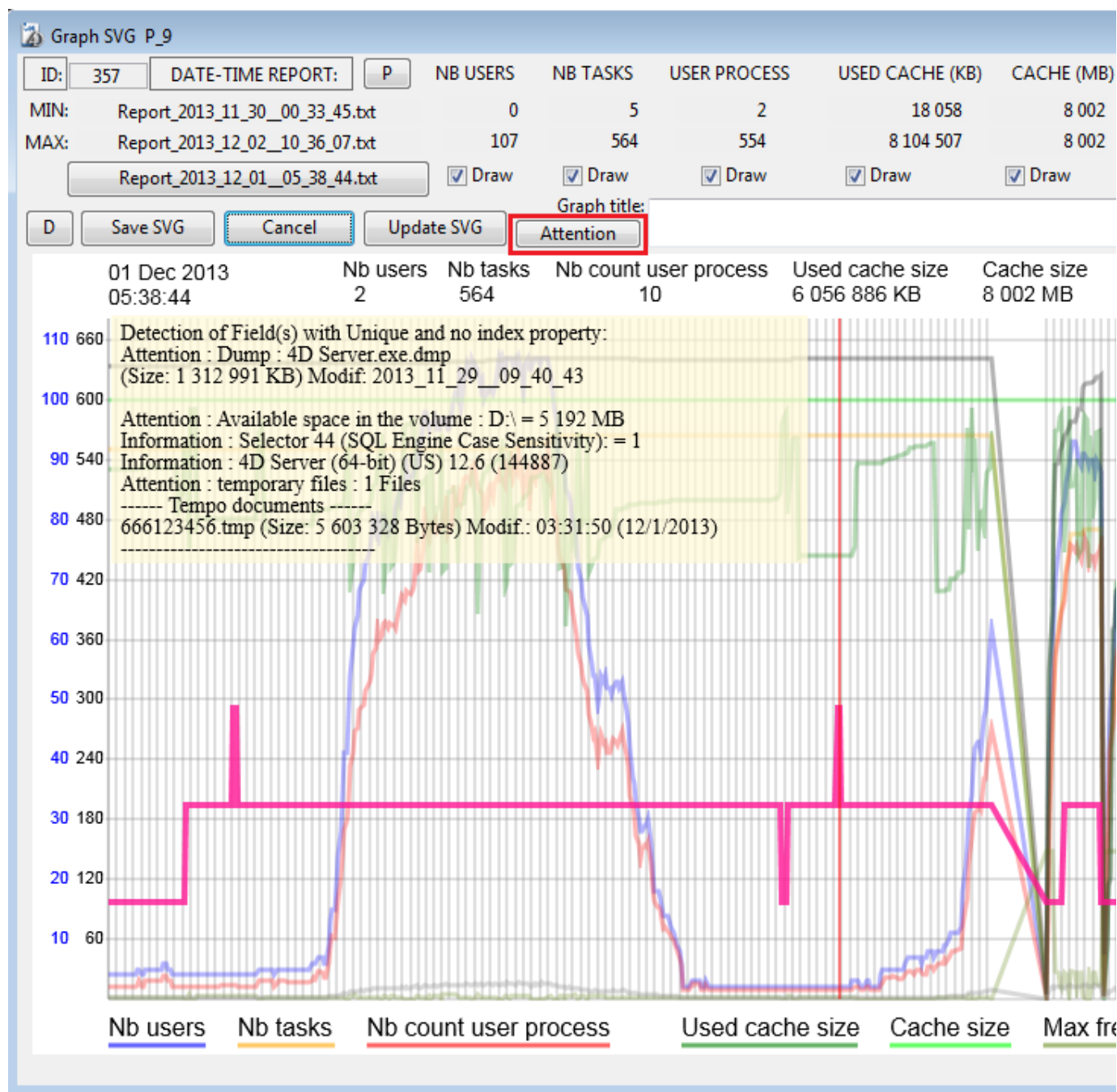
If the button "Attention" is enabled, you can click once to get a pale pink polygon, that will mark by step the number of Attention item found in the Attention section of each report. If you click another time, this new polygon will appear darker, and other polygons will be dimmed. (If you click again another time, this Attention polygon will be hidden.).



After clicking (twice) to get the Attention polygon (this pink polygon has its own vertical scale)

The content of the Attention section is also available in the Graph:

When moving the mouse over reports in this mode, if there is at least one Attention, you will get the detail of the Attention section displayed on top of the SVG graph (at the opposite side)
This text is drawn over a background rectangle (with some transparency)

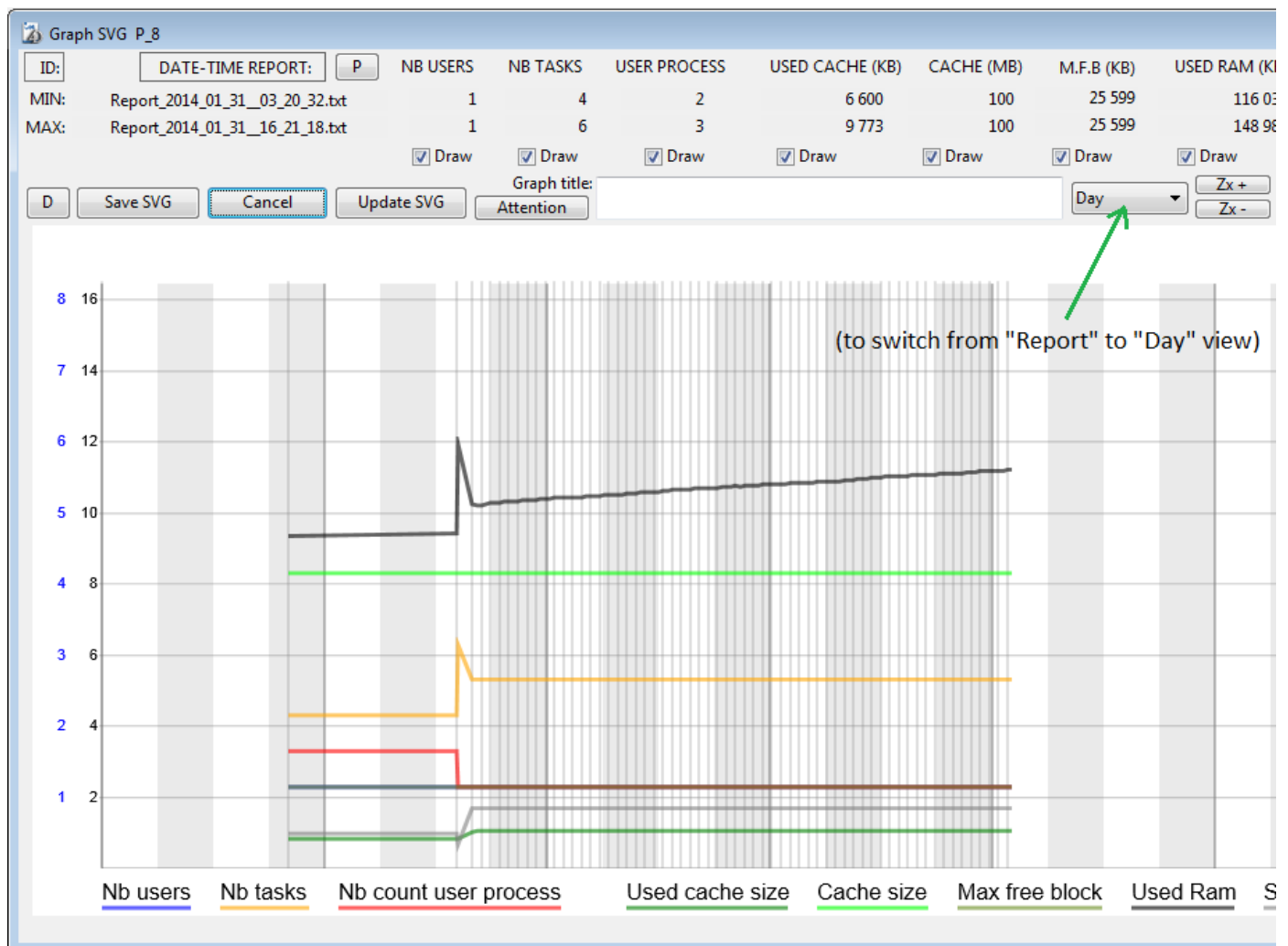


If the background color is a pale yellow, it means that the content of the Attention section changed compared to the Attention section of the previous existing report, otherwise it is white.

The dynamic display of the Attention section avoid the need to open the related report to check it's content. Once this content is displayed, you can click again the Attention button to change the display of the other polygons.

Vue by 'Day' instead of the default vue by 'Report'

If in the pop-up (at the left of the zoom X buttons) you choose 'Day', a day view will be displayed.




This 'Day' view will be scaled for X by default on 24 hours of the current day.
Alternate vertical background will mark each hour of the day.

If a current report was selected in the 'Report' view before switching to the 'Day' view, then the daily reports will correspond to the current day of this selected report.
(if no report was selected, the day of the last report will be displayed).

New in v4.9rA:

- When "Day" is selected, you can now navigate to previous or next day (if it exists) :

 (the "Z+" and "Z-" buttons labels are replaced with "Day+" and "Day-")
(if when clicking, you shift down, you will move to the next or previous week same calendar day).

You can switch back to the 'Report' view (where you will be able to scroll).
The highlighted report should remain the same when switching the view.

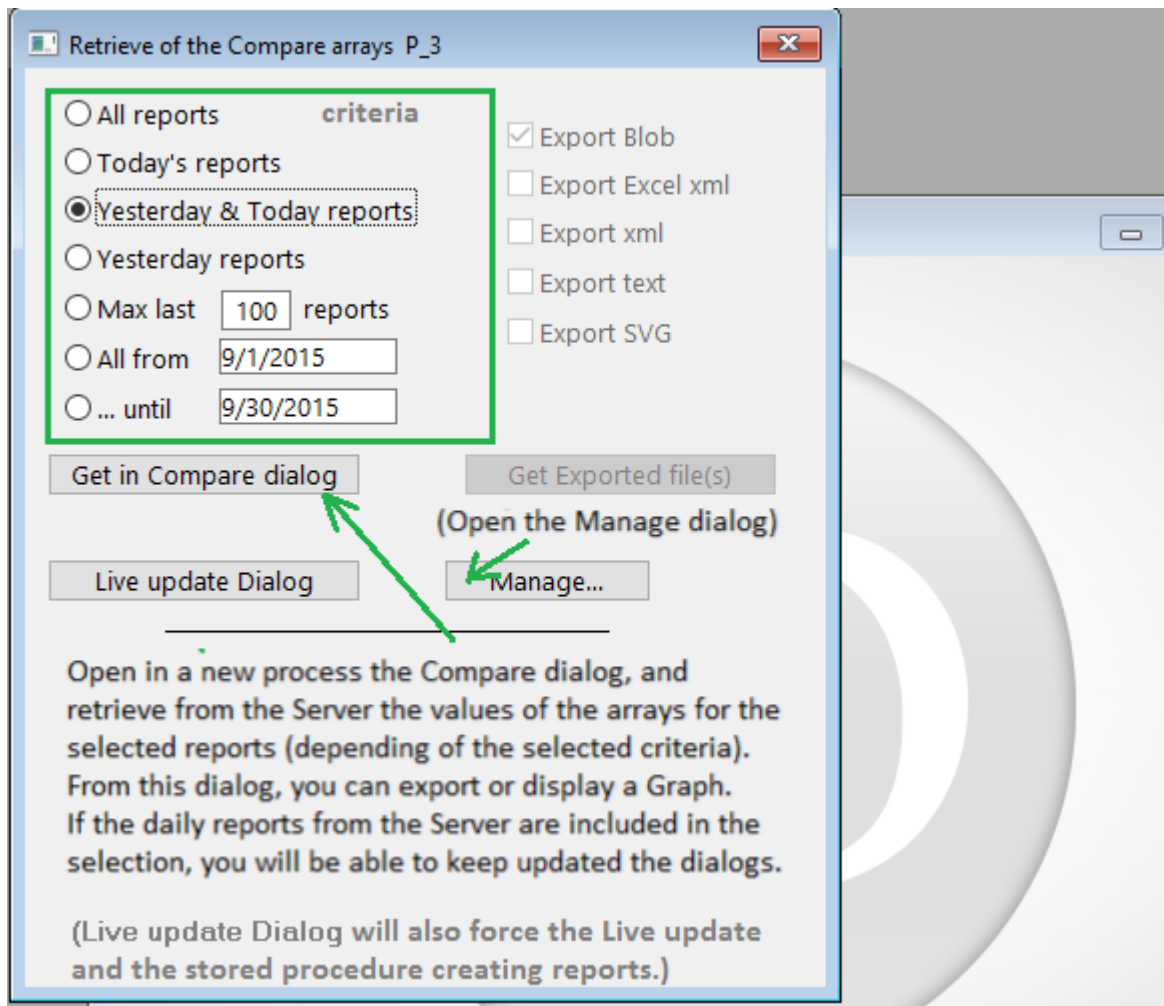
A dialog is available (also in Remote mode), via the method "aa4D_NP_Report_Export_Display" * (or the menu "File / Database reports export" when the component is opened directly)

- If there is one menu item (for example "4D Info Report") that you implement for administration in your Host database, we recommend that it executes a simple method like this:

```

ARRAY TEXT ($at_Components;0)
COMPONENT LIST ($at_Components)
If (Find in array ($at_Components;"4D_Info_Report@")>0)
  // to display this dialog of the component:
  EXECUTE METHOD ("aa4D_NP_Report_Export_Display")
End if

```



Below "Get in Compare dialog", there is this button:

Live update Dialog

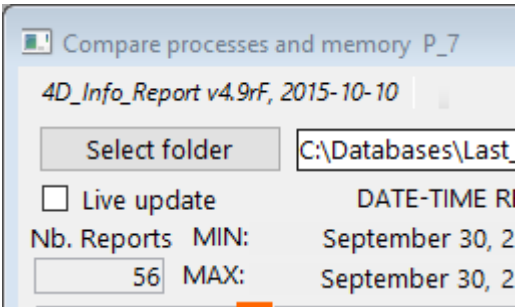
If you click this button, you will directly get the Compare dialog with the next reports from the stored procedure. If this stored procedure is not activated, it will be by clicking this button, creating regular reports every 12 seconds.

In the compare dialog, the "Live update" checkbox will be checked directly.

(Remind to stop the stored procedure when you no more need it via the "Manage" button).

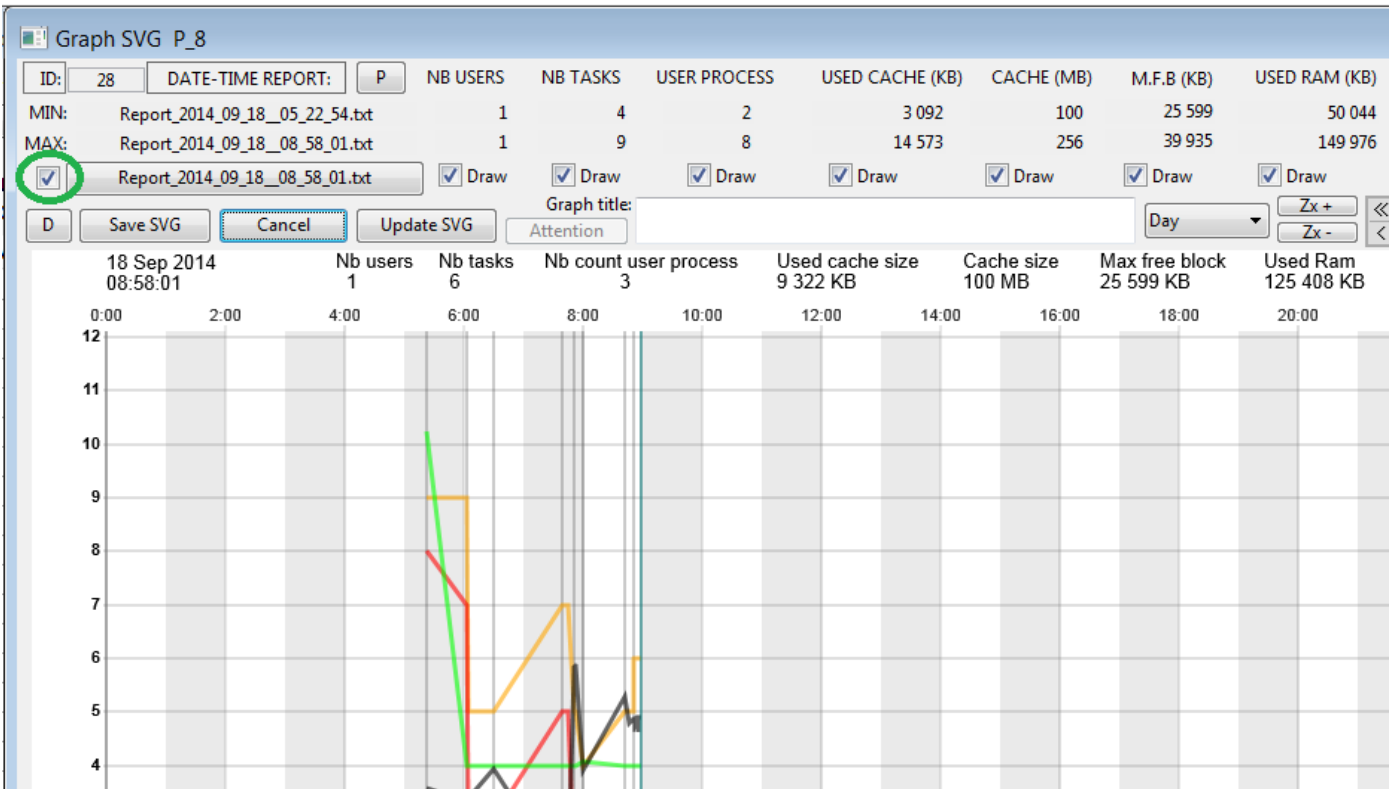
The first time a selection of report (from 4D Server too) will be selected to be displayed in a dialog, these array values will fill interprocess arrays on the Server to avoid another parsing of the values for another time (until the Server is restarted), thus the dialog will display faster another time.

If the requested reports include the daily reports from the Server, a new checkbox 'Live update' will appear on the Compare dialog:



If the stored procedure creating reports is started on the Server, if you click the checkbox, the new created reports values will be added to the current list of values.

If you open a Graph dialog from this Compare dialog, it will also complete the polygons with the created reports. (you can also auto-complete the graph when in the 'Day' view).

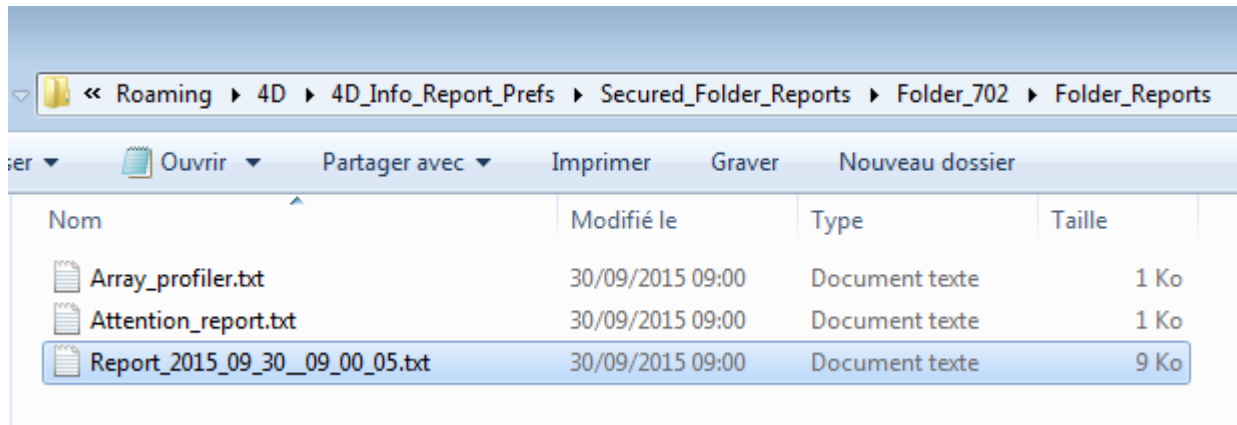


Since v3.8, the creation of reports can handle some errors:

If (for example) you have started the stored procedure to create reports every N minutes on the Server, and the current folder where the new reports are saved is no more available:

- The component will automatically create a new folder in the "4D_Info_Report_Prefs" folder of your computer (this is where is stored some preferences and scripts of the component).

(It will also recreate the Array_profiler.txt, like during the creation of the first report after the restart of 4D or 4D Server):



So with this version, if for any reason the "Folder_Reports" (by default next to the Data file) is no more available, Check if there is a folder named "Secured_Folder_Reports" in the Temp folder.

If this folder exists, check what is the latest created folder inside:

(These created sub folders are named "Folder_" followed by a random number between 100 and 900)

Inside this sub folder, there is a unique "Folder_Reports" that contains the last created reports.

The Attention_report.txt and the Attention section of the created report will contain this kind of content:

Attention : Your folder 'Folder_Reports' was no more available !
({your current location}\Folder_Reports\)

A 'Folder_Reports' was created and is now used in the 4D_Info_Report_Prefs folder :

(C:\Users\{currentuser}\AppData\Roaming\4D\4D_Info_Report_Prefs\Secured_Folder_Reports\Folder_702\Folder_Reports\)

If you have set another volume for the folder to contain the report

(via the shared method: aa4D_M_Folder_Rep_SetPath_local)

and this volume is by accident disconnected, the reports will still be created (in this new folder).

Note: Location of the folder "4D_Info_Report_Prefs" (in **Get 4D folder**(Active 4D Folder) :

On Windows: C:\{Users}\{User name}\AppData\Roaming\4D\4D_Info_Report_Prefs

On OS X: {System volume}:\{Users}:\{User name}:Library:Application Support:
4D:4D_Info_Report_Prefs

Also tested, the access of the Data file:

if there is an error accessing the volume or the Data file itself, a new Attention will be created in the current report. The component will try again to access it when creating the next report.

The Format and content of the generated reports:

1/ All generated files have a structured name :

For example, if a report is created in August 21, 2015 : "Report_2015_08_21__17_16_49.txt"

All reports name begin with "Report_", followed by a Time stamp formatted this way:

"YYYY_MM_DD__HH_MM_SS" (Y=year, M=Month, D=day, H=hour, M=minute, S=second)

If you need to later process a report via the component, you cannot change the name of the document ("Report_", length, etc).

2/ Each document is generated with a UTF-8 format, with all lines ending with CR+LF.

They are Cross platform (OS X and Windows).

(There is a 3 byte BOM at the beginning of the document).

3/ The content of the report is structured, allowing the component to find searched items.

(older reports created by the component "aa4D_Report" might have a different structure, but the compatibility of their structure is handled).

Also, Reports created with French releases of 4D will have french labels, with parsing compatibility.

A/ The Header: (general description of the Computer):

Information report version 4.9rF (2015-10-10*) V_English /** (Date of the version)

Date: 10/10/2015 07:16:49 // (Date format depends of the O.S. language setting)

Computer name:	AMAZONA-NECV76L
Computer user:	Administrator
Manufacturer:	Xen
Computer Kind:	HVM domU

Total RAM:	34944 MB
------------	----------

Number of Processors:	1
Processor Name:	Intel Xeon E5-2665 0
CPU Speed:	2.40 GHz
Total number of Cores:	8
Total number of CPU threads:	16

Operating System:	Windows Server 2008 R2 Datacenter SP1 (64-bit)
-------------------	--

System language:	French (12)
------------------	-------------

IP Address:	10.62.43.60
-------------	-------------

Default System Web Browser:	Internet Explorer 9.00 (8112.16464)
-----------------------------	-------------------------------------

Current printer:	Microsoft XPS Document Writer
------------------	-------------------------------

Computer started since:	0 days, 1 hours, 27 minutes.
-------------------------	------------------------------

B/ Description of the 4D Application : (version, build, 64-bit?):

===== 4D Infos =====

Application type: 4D Server (64-bit) (US)
Application version: v15.0 (F0021500)
Version build of 4D: 15.0 (187516)
More info on 4D release: 15.0 Public release (July 16, 2015)

Mode: Compiled

Application:
"D:\test\15.0\4D Server_64\4D Server.exe"

PICTURE CODEC LIST: (11)
.4pct .jpg .png .bmp .gif .tif .emf .pict .pdf .svg .wdp

C/ Description of the Database (size, location, plugins, components, settings):

===== 4D Database =====

Main UUID: 9F463893A87C478F9F3127EB3AA30DE0 (Structure)
Main UUID: 9F463893A87C478F9F3127EB3AA30DE0 (Data)

Structure File:
"D:\test\15.0\4D Server_64\Bench-64-bit-v15.dbase\Bench-64-bit-v15.4DB"
Size: 1 152 KB Modif: 2015_10_10__07_13_49

Structure File: (index)
"D:\test\15.0\4D Server_64\Bench-64-bit-v15.dbase\Bench-64-bit-v15.4DIndy"
Size: 192 KB Modif: 2015_10_10__07_13_57

Data File:
"D:\ : \test\15.0\4D Server_64\Bench-64-bit-v15.dbase\Bench-64-bit-v15.4DD"
Size: 14 589 248 KB Modif: 2015_10_10__07_13_42

Data File: (index)
"D:\test\15.0\4D Server_64\Bench-64-bit-v15.dbase\Bench-64-bit-v15.4DIndx"
Size: 3 365 312 KB Modif: 2015_10_10__07_13_10

Match File:
"D:\test\15.0\4D Server_64\Bench-64-bit-v15.dbase\Bench-64-bit-v15.Match"
Size: 17 KB Modif: 2015_10_10__07_13_57

<----- Plugins -----

4D Internet Commands

----- Plugins ----->

```

<----- Plugins Bundle -----

--- Database ---
4D InternetCommands.bundle Executable Key: 4D InternetCommands Short Version
Key: 14 R5 (181353)

----- Plugins Bundle ----->

<----- Components -----

4D SVG
4D Widgets

----- Components ----->

<----- Components info -----

--- Database ---
4D_Info_Report_v4.4dbase   Get info String:   4D_Info_Report v4.9rF (22) for
4D v15
--- Application ---
4D SVG.4dbase Get info String: 4D SVG version 15.0 (47), ©4D SAS 2009-2015
4D Widgets.4dbase

----- Components info ----->

<----- Backup infos -----

Backup.XML location:
"\Preferences\Backup\Backup.XML"

Backup.XML Scheduler:
Daily, every 1 day

Log File:
(No Log File found.)

Last Backup:
"E:\_Backup_Files\Bench-64-bit-v15[0001].4BK"
Size:      14 589 248 KB      Modif:  2015_10_09__07_13_20

Last Backup Infos:
( Last Backup date & time (in Backup.XML) : 2015_10_09__07_13_20 )

( GET BACKUP INFORMATION;2 : Err.: 0 : No detected error.)

( GET BACKUP INFORMATION;0 : 2015_10_10__07_13_20 )

Next Backup:
( GET BACKUP INFORMATION;4 : 2015_10_11__07_13 )

```

Preferences Backup Advanced:

<BackupFailure>

TryBackupAtTheNextScheduledDate: False
TryToBackupAfter: 0000_00_00__00_01_00.000
AbortIfBackupFail: False
RetryCountBeforeAbort: 5

(Automatic Restore)

Restore last backup if database is damaged: False
Integrate last log if database is incomplete: True

AutomaticRestart: True

----- Backup infos ----->

Involved disk description:

Disk	NbPart.	Type	Status	Media Type	Total Size	Description
0	2	SCSI	OK	Fixed hard disk media	102 398 MB	RHEL DISK SCSI Disk Device

Involved volume description:

Letter	Disk	NumPart.	Type	Total Size	FileSystem
C:	0	0	2 (Fixed)	35 837 MB	NTFS
D:	0	1	2 (Fixed)	66 558 MB	NTFS

Involved volume (free space):

Volume	C:\ (Operating System)	4 217 MB
Volume	D:\	47 959 MB

Description of the memory modules:

Locator	Capacity	Speed	Type	Manufacturer
ChannelA_Dimm1	4 GB	2133 MHz	DDR3	Undefined
ChannelB_Dimm1	4 GB	2133 MHz	DDR3	Undefined
ChannelC_Dimm1	4 GB	2133 MHz	DDR3	Undefined
ChannelD_Dimm1	4 GB	2133 MHz	DDR3	Undefined

=====

<----- Comment -----

(content of a pointed text variable (\$3) if passed).

----- Comment ----->

Maximum number of users (license): 1001

User list size: 2
Group list size: 0

Web: (No checking of the Web Server)
Count tasks: 970
Count user processes: 967
Number of users: 483

----- Database parameters -----

Database Cache: 1 536 MB
4D Server Timeout: ??? Minute(s) // if Host DB w/o spec. method*
4D Remote Timeout: ??? Minute(s) // if Host DB w/o spec. method*

Selector 28 (4D Server Log Recording): 0
Selector 34 (Debug Log Recording): 0
Selector 44 (SQL Engine Case Sensitivity): 1
Selector 53 (stack size of preemptive thread): 1024 KB
Selector 54 (Idle Connections Timeout): 20 seconds.
Selector 61 (Maximum Temporary Memory Size): 500 MB
Selector 65 (CacheLog Recording): 1
Selector 66 (Cache unload minimum size): 157 286 KB
Selector 67 (Time-out forced disconnections): 3 seconds.
Selector 68 (Log disconnections errors): 1
Selector 69 (Direct2D Status): 5
Selector 74 (Direct2D Get active status): 3

----- other info via GET MEMORY STATISTICS: -----

Used cache Size : 1 476 470 KB
Free Memory : 31 838 820 KB
Used physical memory : 1 804 908 KB
Used virtual memory : 2 821 048 KB
Size of stacks : 733 197 KB
Altura Allocated Memory : 14 061 KB
Altura Used Memory : 11 555 KB
Biggest free block in cache : 180 KB

Scheduler (CPU settings):
4D Server: 0 - 8 - 0 Medium (Standard setting), recommended.

Processing duration in the report: 640 Milliseconds

***** Attention section *****

Attention : (any Attention content, that will create an "Attention_Report.txt" file).

Attention : Backup activated without Logfile!

Information : 4D Server (64-bit) (US) 15.0 (187516)

Information : Administration window opened on Server

Note: It is important to read this **Attention section**, as it contains a summary of possible causes of problems, and also some reminders (Information).

Typical elements in the Attention section:

- = Attention when a log is activated
- = Attention or Information about the Operating system if:
 - It is a beta version, or an Insider Preview (Windows).
 - The Operating System was not certified for the version, or not recommended.
 - The Operating System is no more updated with security fixes
- = Some low limits are reached (available space in a volume, memory, cache usage)
- = Some settings are not the recommended ones.
- = Some Hardware (real or emulated) numbers are low (RAM, CPU speed, number of CPU-threads).
- = The current IP of the 4D Application has changed.
- = Information if a Virtual machine is detected
- = Attention or Information if the Host Database is not in Unicode mode*

When this section Attention is populated in a report, a new "**Attention_Report.txt**" file is generated, with the content of this section, so it is quite easy to detect and read, when checking the content of the Folder_reports.

When this Attention section is updated, the new content is sent to the Host database if the Host method "**aa4D_M_Host_Attention_Reported**" is copied and shared with components.

If you are asked to provide a report, better compress the "Folder_reports" folder, and provide this archive, as other specific files might be added by the component while creating a report.

*To detect this setting (and others) about the Host database, be sure to add and share this method (included in the Host template):

```
// Method name: aa4D_Host_GetDBParam (must be shared with components)
C_LONGINT($1) // selector
C_REAL($0)
$0:=-1 // Error, no parameter
If (Count parameters>0)
$0:=Get database parameter($1)
End if
```

D/ Infos on the Tables (num, name, nb records, fields, indexed fields, subtables, triggers):

===== TABLES =====

(Triggers description: N = On saving New record, E = On saving Existing record,
D = On Deleting record)

Total number of records: 58 095 555

Number of tables: 167 Number of valid tables: 167

Total number of valid fields: 1079

Total number of indexed fields: 412

	Num	Name		Nb of Records	Fields	(Index		SubT.)	Trig
T.:	3	Cities	-----	43 607	4	(1	-)	N---
T.:	2	Companies	-----	68 850	5	(2	-)	----
T.:	110	Companies0005	-----	40 000	5	(0	-)	NE--
T.:	247	Companies0006	-----	70 000	5	(0	-)	----
T.:	245	Companies0007	-----	30 000	5	(0	-)	----
T.:	243	Companies0008	-----	40 000	5	(0	-)	--D-
T.:	241	Companies0009	-----	60 000	5	(0	-)	----
T.:	239	Companies0010	-----	50 000	5	(0	-)	----
T.:	237	Companies0011	-----	80 000	5	(0	-)	----
[...]									

The table description is optional, you can avoid listing them, hide structure information (only showing the number of records), hide "Invisible" tables, etc...

The total number of records will always be reported. (with option on Invisible tables).

Deployment of the component:

To use the component, add it to the "Components" folder next to the Structure file of your application.

(Create this folder if it does not exist).

This component does not create Table(s) in the Host database. The Host database remains unchanged. Only the folder "Folder_reports" is added next to the data file, and filled with created reports.

Accessing the code of the component from the Host database.

While in development, or if you allow access to the execution of methods, the shared methods of the components can be executed directly.

(you can also see them in the Explorer (project methods), after deploying the component list, and selecting this component)

If you plan to deploy your application, we recommend to avoid tokenizing the shared methods of the components in your Host database, in case you want to remove it.

You can use this example of code to avoid the "hard link" of the component with your Host database:

```
//----- to execute a shared method and be able to compile after removing the component ---  
ARRAY TEXT($at_Components;0)  
COMPONENT LIST($at_Components)  
If (Find in array ($at_Components;"4D_Info_Report@")>0)  
  If (Shift down) // Only on demand  
    EXECUTE METHOD("aa4D_M_Util_CreateReport") // Create one report  
  End if  
End if
```

You can also set a Boolean interprocess to avoid parsing the Component list each time, like <>vb_Is_4DIR_ready, and only test at startup if the component is available, and then use this alternative code:

```
If (<>vb_Is_4DIR_ready) // if the component 4D_Info_Report is loaded  
  If (Is compiled mode) // if the Host database is compiled  
    If (Application type=5) // 4D Server  
      EXECUTE METHOD("aa4D_NP_Schedule_Reports_Server";*;30;-2) // or another shared method.  
      // EXECUTE METHOD("aa4D_NP_Schedule_Reports_Server";*;0) // to stop the stored procedure.  
    End if  
  End if  
End if
```

- Remind that all shared methods beginning with "aa4D_NP" create a new process, so you don't need to create one in the Host database to execute these methods. Also, via some dialogs of the component, you can execute directly some other shared methods (if allowed).

You can limit the access of reports stored in the Server, or the right to set or change the setting of the stored procedure ("aa4D_NP_Schedule_Reports_Server") from a remote user, via this code to implement, as the component execute this shared Host method if it exists, and expect a True Boolean result:

```
// Host method: "aa4D_M_Host_Allow_Report_access" (example code)
C_BLOB($1)
C_BOOLEAN($0)
C_BLOB($vxBlob)
C_BOOLEAN($Allow)
C_LONGINT($Offset)
Allow:=True // default answer
If (Count parameters>=1)
  If (Type($1)=Is BLOB )
    vxBlob:=$1 //--- in Unicode only, as the component is built in Unicode ---"
    ARRAY TEXT($at_infos;0)
  Allow:=True // default answer
  If (Count parameters>=1)
    If (Type($1)=Is BLOB )
      vxBlob:=$1 //--- in Unicode only, as the component is built in Unicode ---"
      ARRAY TEXT($at_infos;0)
      BLOB TO VARIABLE($vxBlob;$at_infos;$Offset)
      //- $at_infos{1}:=Current user
      //- $at_infos{2}:= (local) I.P. address
      //- $at_infos{3}:=Current machine
      //- $at_infos{4}:=Current machine owner
      //- $at_infos{5}:=Calling component method (optional)
      //
      // --- set your own conditions, depending of the 5 array elements ---
      $Allow:=False // default answer
      If (Application type#4D Server )
        // ALERT(aa4D_M_Host_Allow_Report_access' OK") // to control that it is called
      End if
    End if
  End if
End if
End if
$0:=$ Allow
```


It is up to you to design the conditions test (for example limiting this access to local network users, such Current user, etc...)

To ease a better usage of the component, there is a small Host template database, that contains few methods to be added in your Host database(s).

In the "4D_Info_Report_Host_Template_v6_v12.4dbase", there is this HDI:
(content of the archive "4D_Info_Report_Host_T_v6_v12.zip")

```
// Method: aa4D_M_Host__How_Do_I
// (Thomas.Schlumberger@4d.com, August 21, 2015)
//
// If you are using the Information component (4D_Info_Report_v4.9rC or later),
// Some specific Host methods will be tested by the component,
// If they exist and are shared, they will be used to communicate with the Host database.
//
// Note: if the component is removed, these methods will not disturb the behavior of the Host
database, so you can be integrated in your Host database without drawback.
//
// at least copy (by Drag and Drop) these seven methods in your Host database:
// - "aa4D_H__Startup" // to be called in the On Startup method
// - "aa4D_H_On_Server_Startup" // to be called in the On Server Startup method
// - "aa4D_Host_GetDBParam" (new shared method now usable with all versions of 4D).
// - "aa4D_M_Host_Manage_Info_Report"
// - "aa4D_M_Host_Attention_Reported"
// - "aa4D_M_Host_Allow_Report_access"
// - "aa4D_M_Host_SP_Scheduler" // Mandatory if using "aa4D_H_On_Server_Startup"

// - "aa4D_M_Host_Is_in_Unicode" (deprecated now, replaced by aa4D_Host_GetDBParam)
// - "aa4D_M_Host_Get_both_Timeout" (can use aa4D_Host_GetDBParam instead)

// You must keep their original name, and set "shared by component and host database"

// Reminder: to use the component "4D_Info_Report_v4",
// just add it to the "Components" folder of the Host database or the 4D Application.
// (if this "Components" folder does not exist, just create it).
//
// As this database is compatible with 4D v12 database:
// To use these methods with a Host database that is running with 4D v13, v14 or later versions,
// just convert a copy of this Template database with your current 4D version
// before drag & dropping these methods into the Host database.
//
// If you use Compiler_methods, change the settings of the compiler
// to generate the Compiler expected methods.
```

After implementing these provided methods in your Host database:

- More information will be available in the created reports, such as the Unicode mode, the timeout values (4D Server, 4D Remote)

- When a new Attention is raised in a new report, the Attention content is received by the Host database, that can take appropriate action depending of the content changes.
- For some kind of actions performed via the component, you can disallow such actions depending of the profile of the remote user.

Note: The first report generated by the component after the startup of 4D or 4D Server will take a longer time to be created.
Until the database is opened again, following reports will be created much faster, because some deep parsing is already done and memorized.

Also, during the creation of the first report, a checking of the SQL syntax compliance for tables and fields will be performed:
A file "Info_SQL_Naming.txt" can be created with the summary for tables and fields names that does not comply.

Updates of the component:

There is on average a monthly update of the component, that include new features, and recognition of new versions of 4D and 4D Server.

It is then important to check from time to time if an update of the component is available via the forums or source where it was downloaded.

The current version 4.9rZ (for 4D v16 and later) has a Time-bomb set to the end of December 2018, preventing the creation of new reports beyond this limit:

If the Host database is not compiled, a new file will be created in the Folder_Report:
"Attention_component_expired.txt", with a dialog (not on 4D Server) asking the user to contact the technical support to get an updated version.

If you create reports in 2018 (after the end of September) with v4.9rZ, at the beginning of the Attention section, you will get this information:

***** Attention section *****

Information : Update the component 4D_Info_Report before January 2019

You have plenty of time to get new versions of the component before it usually expires.
(Next one should be a v5 update).

Conclusion:

This component has already been widely used since more than five years, even with heavily loaded 4D Server.

We advise installing it in your 4D Server application, so you can monitor the Server or create a report for the Tech Support upon request without restarting your application.