1   Cover Page

PLAN EXPLORER USER GUIDE

2   Table of Contents

WELCOME TO THE PLAN EXPLORER USER GUIDE!

3.1   INTRODUCTION

Plan Explorer is a lightweight .NET-based tool that builds upon and supplements the capabilities of SSMS for execution plan analysis.

There are several high value features, including:

- ability to open multiple tabs
- plan history tracking
- index analysis
- wait stats analysis
- exposure of the full query call stack
- open and view SQL Server deadlocks
- and much more

In addition to this guide, there are several blog posts and videos available which explain and demonstrate how and why you should use **Plan Explorer**.

3.2   CONTACT US

If you have any technical questions, please don't hesitate to contact us:

Email: support@sentryone.com

Phone: 704-895-6241

Toll Free: 855-775-7733

Support Forum: http://answers.sqlsentry.com

Query Plan Q&A: https://answers.sqlperformance.com

INSTALLATION

SentryOne Plan Explorer is a single installation file. Plan Explorer can be [downloaded here](#).

5	Using Plan Explorer

5.1	Plan Explorer Sessions

Plan Explorer introduces the concept of a **Plan Explorer Session** and its associated **Plan Explorer Session file** (.pesession).

Plan Explorer Sessions are designed to help you manage an historical record as you fine tune queries. By default, an historical entry will be generated as part of the **Plan Explorer Session** during estimated and actual plan retrieval

> 📑 **Note:** You can change the default behavior of how Plan Explorer saves history for session files through the Preferences (**Edit menu → Preferences**). See also: **Preferences (Section 6.1)**

5.1.1 VERSION HISTORY

Each historical entry retains all captured plan details and metrics within the various Plan Explorer tabs. Each version is associated with a unique version number. **Plan Explorer** includes a **History pane** (**View menu → History**) which allows you to quickly navigate through the different versions within the active Plan Explorer Session. You can also delete a version with the **Delete command** accessed in the **History pane** context menu.

> 💡 The **Status bar**, located at the bottom of the Plan Explorer window, will update as actions are taken, including each time a new historical version is loaded.

5.1.2 ADDING COMMENTS

Plan Explorer also allows you to add comments to each historical version (**View menu → Comments**). This allows you to easily keep track of the reasoning behind any changes you make, such as changes to the command text or any indexing optimization.

5.1.3 MULTIPLE SESSIONS AND TABS

As each session is managed within its own tab, multiple **Plan Explorer Sessions** can be open at the same time. **Plan Explorer** was designed with a multiple document interface allowing you a number of options when managing tabbed sessions, including the ability to arrange tabs both horizontally and vertically, and the ability to tear off tabs. With any tab selected, right-click the tab header to access these options. Tab windowing options can also be found in the **Window menu**.

5.1.4 SHARING SESSION FILES

Plan Explorer Session files can be easily shared with others.

This is possible because each historical entry in a **Plan Explorer Session** is actually a .queryanalysis file. Plan Explorer Session files are archive files, containing both the individual .queryanalysis files and the metadata about the session.

> 💡 You can also save any single historical entry of a Plan Explorer Session as a stand-alone **.queryanalysis** or **.sqlplan** file. With the entry you wish to save active, use the **Save As** command (**File menu → Save as**) and choose the desired file type.

5.1.5 STARTING A NEW SESSION

You can start a new **Plan Explorer Session** a number of ways, including:

- **File menu → New**
- From the **Toolbar** use the **New button**
- Dragging and dropping a plan file onto the application

5.1.6   SAVING A SESSION

You can save a **Plan Explorer Session** as follows:

- **File menu → Save / Save As / Save All**
- From the Toolbar use the **Save** or **Save All button**

5.2   Full Query Call Stack

Whenever an **Actual Plan (Section 5.5)** is retrieved in **Plan Explorer** the full query call stack is also assembled. This ability has been a part of Performance Analysis in SentryOne for some time, and was also made available as part of the **Plan Explorer** feature set.

Plan Explorer offers the complete call stack, including nested procedure calls and statements called with dynamic SQL. The complete query call stack can be invaluable, as it lets you see exactly how all of the captured statements are related and where they fit into the query plan. Most importantly, it shows you which statements did the bulk of the work, and which are candidates for optimization.

5.3   Deadlock Files

Plan Explorer allows you to open deadlock files. Deadlock (.xdl) files can be captured or collected from a number of places, including:

- **SentryOne Performance Analysis** captures deadlocks happening in your monitored environment. These .xdl files can be exported and then shared.
- **SQL Server Profiler** allows you to create a trace that will capture deadlock events. For more information see the following MSDN article: Save Deadlock Graphs

5.3.1   OPENING A DEADLOCK FILE

After you have captured a deadlock file (.xdl) you can open it with **Plan Explorer** just as you would a plan file. A deadlock is displayed in two main sections. The top section contains a **Deadlock Gridview** which breaks down the deadlock by SPID and lock resource. The bottom section contains the **Deadlock Graph** with representations for each process and resource involved in the deadlock.

5.3.2   DEADLOCK GRIDVIEW

In the **Deadlock Gridview** each record is expandable to view the details regarding the processes that were involved in the deadlock. For general information about the metrics displayed see the **Deadlock Metrics** section at the end of this topic.

- The deadlock victim is highlighted with a red background.

5.3.3   DEADLOCK GRAPH

The **Deadlock Graph** is formed according to the deadlock XML. Process and Resource nodes are independently represented, along with any relationships which exist between them.

- The deadlock victim is highlighted with a red background.
- The numbers (1) and associated arrows, between each node, indicate the actual sequence of events which took place, leading to the deadlock.
- The letters indicate the requested lock mode.
- Selecting object nodes in the graphical representation will set focus to that object in the gridview.
- You may zoom in and out on the Deadlock Graph using **ctrl + mouse wheel** or through the right-click context menu.
- You may drag and rearrange nodes as desired.

DEADLOCK METRICS

Deadlock Overview

Name	Description
Victim SPID	The session process ID of the victim involved in the deadlock.
Victim Host	The workstation belonging to the victim thread.
Victim Application	The application name belonging to the victim thread.
Victim Database	The name of the database on which the process took place
Victim Text Data	The associated text data of the victim.
Deadlock XML	This is the actual captured Deadlock XML.

Lock Details

The Lock Details area breaks down the deadlock by specific lock types, including the owners and waiters involved in each lock.

Name	Description
SPID [ecid]	The session process ID of the associated owner/waiter.
Host	The workstation name.
Application	The associated application
Database	The associated database.
Log Used	The amount of Log Space used by the process.
Deadlock Priority	Specifies the Deadlock Priority. Zero (0) or NORMAL is the default priority. In cases where each session has the same deadlock priority, SQL Server will choose the victim based on the least expensive session to roll back. For general information about the DEADLOCK_PRIORITY option see this MSDN article.
Wait Time	Time in (ms) milliseconds spent waiting on the resource.
Transaction Start Time	Time that the transaction began.
Last Batch Start Time	The last time a client process started batch execution.
Last Batch Completion Time	The last time a client process completed batch execution.
Mode/Type	The Mode/Type designates the resource lock mode. For general information on Lock Modes see this MSDN article.
Status	State of the task.
Isolation Level	The current transaction isolation level. For general information on isolation levels see this MSDN article.

Name	Description
Login Name	The Login Name associated with the session.

Owner/Waiter Details

Name	Description
Object	Indicates the associated object name.
Line Number	The line number which was being executed when the lock occurred.
Text Data	The associated text data.

5.4   Viewing Query Plans

There are multiple ways to view a query plan in Plan Explorer:

- Using the Management Studio Add-In (**a separate install**), right-click a plan in SSMS and choose, **View with SentryOne Plan Explorer**. This will open the plan(s) in Plan Explorer, providing the correct server/database context for further operations.

  💡 Be sure you have the latest version of the SSMS Add-in installed.

- In SSMS, right-click a graphical plan and select **Show Execution Plan XML**, then copy and paste the plan XML into Plan Explorer. It doesn't matter which pane is active, the clipboard handler is global and will auto-detect plan XML. This makes it quick and easy to move plans from SSMS into Plan Explorer if you're not using the SSMS add-in.

- Save an execution plan from SSMS to a .sqlplan file, then open the file using the **Open** toolbar button or **File->Open**. Raw plan XML files and .QueryAnalysis files are also supported.

- Right-click an existing .sqlplan file and select **Open with** -> **SentryOne Plan Explorer**.

- Drag-and-drop a plan file onto the application.

- Retrieve the estimated or actual plan from TSQL in the Command Text pane using the **Get Estimated Plan (CTRL-L)** or **Get Actual Plan (CTRL-E / F5)** button on the toolbar. The connection dialog box will open to connect to a valid SQL Server instance (unless Plan Explorer was launched using the add-in, in which case the connection information will be pre-populated).

5.4.1   RETRIEVING THE ESTIMATED PLAN

Copy and paste, or type raw TSQL into the Command Text tab, click the **Get Estimated Plan** toolbar button or use the **CTRL-L** keyboard shortcut to connect to a server and get the estimated plan.

5.4.2   PLAN DIAGRAM CONTEXT MENUS

Once you have a plan loaded, right-click anywhere in the diagram area to access context menus, which control how the graphical plan is displayed. By default, color-scaling and per node costs are enabled.

5.4.3   ACTUAL PLANS

If an actual plan is loaded, actual metrics are shown by default. You can quickly toggle the plan diagram back and forth between estimated and actual metrics using the **Show Estimated Plan** toolbar button. This button is only enabled when an actual plan is loaded.

5.5   Generating an Actual Plan

To generate an actual plan use the **Get Actual Plan button** on the **toolbar (Section 6.1)** or use the **CTRL-E** / **F5** keyboard shortcut. When you generate an actual plan the query is executed against the server. All the statements in the Text Data will be executed. This allows you to collect actual runtime metrics, such as duration, CPU and reads, which you cannot get from an estimated plan. By design, the actual query results are not returned to the Plan Explorer interface.

Whenever you generate an actual plan an execution warning message will also appear. It can be disabled through the Preferences (**Edit menu → Preferences → Warn on actual execution)**.

5.6   SQLPerformance.com Integration

Plan Explorer includes support for uploading several different plan file types directly to SQLPerformance.com from the client. This feature allows you to post a question, with an included plan, to the SQLPerformance.com Q&A site answers.sqlperformance.com.

> 📑 **Note**: For further information about Plan Explorer integration with SQLPerformance.com please see this announcement by SQL Server MVP Aaron Bertrand.

You may post the following file types directly to SQLPerformance.com from the Plan Explorer Client:

> **.pesession**
> **.xdl deadlock**
> **.queryanalysis**
> **.sqlplan** *
> **.xml** *

> 📑 **Note:** When you post either a **.sqlplan** file or a **.xml** file through the Plan Explorer Client to SQLPerformance.com it will automatically be converted to a **.queryanalysis** file.

POSTING A PLAN TO SQLPERFORMANCE.COM

The first time you post a plan to SQLPerformance.com you will be prompted to register. Your login credentials for SQLPerformance.com will be saved locally in a secure manner. There are two different options for posting plans to SQLPerformance.com from the Plan Explorer Client.
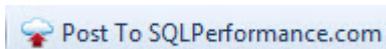
☁ Post To SQLPerformance.com   From the context of any open plan within Plan Explorer, you may use the **Post To SQLPerformance.com** button found on the toolbar. This will open a question form, automatically attaching the current plan, allowing you to add your desired comment or question.
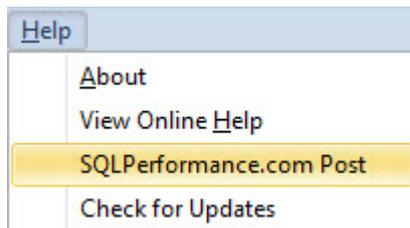
Alternatively, you may upload a plan from the Help Menu (**Help → SQLPerformance.com Post**). Using this method you are not restricted to the currently open plan. You may upload any plan of your choice.

> 📑 **Note:** When you upload a plan from the Help Menu a link back to the post will not be embedded in the local file, but the embedded link will be available for those who download the plan from SQLPerformance.com.

5.6.1   ANONYMIZE PLANS

Plan Explorer has the ability to obfuscate the sensitive data in your plans. There are two ways to use this feature.

- During plan upload you will be presented with the option to anonymize the plan. On the Post to SQLPerformance.com dialog form select the **Anonymize the query plan before upload** checkbox option.
- Alternatively, you may anonymize the currently open plan using the **Anonymize Button** found on the toolbar.
  - In Plan Explorer the anonymized plan will open in a new tab.

> 📑 **Note:** You can read more about this feature in this blog post by Aaron Bertrand.

5.6.2   EMBEDDED LINK

Once you have uploaded a plan to SQLPerformance.com a web link is embedded in the file. This link will allow you to jump directly to your post. This link will also be available for anyone who downloads the plan.

5.6.3   POSTING .SQLPLAN AND .XML FILES

In addition to Plan Explorer's file types (**.queryanalysis**, **.pesession**) you may also post **.sqlplan** and **.xml** file types directly to SQLPerformance.com through the Plan Explorer Client. During the upload process these files will be automatically converted to the **.queryanalysis** file type.

5.6.4   SECURITY OBFUSCATION

Regardless of how you upload a plan, connection parameters stored by Plan Explorer are removed from the plan during the upload process.

The following identifying connection parameters are automatically removed from plans during the upload process:

- **Instance**
- **Database**
- **Login**

The following identifying information is removed from deadlock .xdl files during the upload process:

- **Login**
- **Host**

5.7   Live Query Profile

Live Query Profiling allows for the capture and replay of live query execution data. To enable the collection of live query data, click the arrow next to the Get Actual Plan button and click the option With Live Query Profile. While this setting is enabled, live query execution data will be captured along with the actual plan. Plan Explorer adds the benefit of being able to replay a live query. This replay data is saved as part of the Plan Explorer session file.

5.7.1   RELATED SETTINGS

Default Actual Plan Collection Mode - This setting allows you to chose between getting an actual plan only, or getting the actual plan and live profile by default.

Live Profile Data Collection Interval - This setting determines how often Live Query data is collected. The minimum value is 100 ms.

For additional information about this feature, see this blog post by Jason Hall.

> **Note:** A minimum of SQL Server 2014 SP1 is required to collect Live Query Profile data.

6	Interface and Features Overview

Below are links to the primary features and information about the different tabs found within Plan Explorer. Since the graphical plan diagram in SSMS was used as a starting point for the design of Plan Explorer, some features are compared with the current capabilities of SSMS in order to illustrate the reasoning behind the feature.

- **Menus and Toolbar (Section 6.1)**
- **Command Text (Section 6.2)**
- **Results (Section 6.3)**
  - **Statement Section (Section 6.3.1)**
  - **General Query and Plan Info Section (Section 6.3.2)**
  - **Plan Details Section (Section 6.3.3)**
- **Wait Stats (Section 6.4)**
- **Index Analysis (Section 6.5)**

6.1	Menus and Toolbar

MENUS

6.1.1   FILE MENU

Command	Description
New	Create a new **Plan Explorer Session (Section 5.1)** in a new tab.
Open	Open an existing Execution Plan file (.sqlplan | .xml | .queryanalysis | .pesession), deadlock file, or SQL file.
Save	Save the current file.
Save As	Save the current file with a new name/new location.
Save All	Save all files opened in any tabbed sessions.
Recent Files	Display a list of recently opened files.
Exit	Exit the program.

6.1.2   EDIT MENU

Command	Description

Check for updates on startup

If checked Plan Explorer will automatically check for the newest build when starting.

> 💡 You can also check for updates at anytime through the Help menu.
>
> **Help menu →Check for Updates**

Warn on actual execution

If checked a warning will be displayed each time an actual plan is retrieved.

Preferences

Only save history when command text or connection settings change

If checked, a versioned entry will be generated as part of the **PE session (Section 5.1)** for estimated and actual plan retrieval, only if the command text or connection settings have changed.

If unchecked, a versioned entry will be generated as part of the **PE session (Section 5.1)** each time an estimated or actual plan is retrieved, regardless of any changes made to the command text or connection settings.

> 💡 Use the History pane to quickly move between the different versions of a plan contained within the Plan Explorer session (**View menu → History**).

Connection

Specify a Server name, Database name, and authentication method to use when retrieving the estimated or actual plan.

Copy

Copy the selection.

Paste

Paste from the clipboard.

Anonymize

Removes sensitive information from your plans. This anonymized plan will open in a new tab.

Read more about this feature in this blog post by Aaron Bertrand.

6.1.3   VIEW MENU

Command	Description
History	Opens the History pane.
Comments	Opens the Comments pane.
Status	Opens the Status bar located at the bottom of the Plan Explorer window.
Layout Editor	Opens the Layout Editor. Use the Layout Editor to customize the position of the various panes.

6.1.4   WINDOW MENU

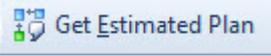| Command	Description
Reset Window Layout	Resets the windows to the default state.
New Horizontal Tab Group	Creates a new Horizontal tab group with the selected tab.
New Vertical Tab Group	Creates a new Vertical tab group with the selected tab.
Move to Next Tab Group	Moves the selected tab group to the next group.
Move to Previous Tab Group	Moves the selected tab group to the previous group.
Windows...	Use the Windows.. command to manage all your open windows. Use the Activate command to set focus to the selected window.

6.1.5   HELP MENU

Command	Description
About	View information about the Plan Explorer, including build number and number of evaluation days remaining.
Check for Updates	Check for the newest version of Plan Explorer

TOOLBAR

Button	Description
	New - Creates a new Plan Explorer Session in a tab
	Open - Opens an existing Session file, Execution Plan, Deadlock, or SQL File
	Save - Saves the current file
	Save All - Saves all of the currently open files
	Copy

Button	Description
Paste	**Paste**
	Horizontal Layout - Switches the current tab to a horizontal layout
	Vertical Layout - Switches the current tab to a vertical layout

Button	Description
Get Estimated Plan	Generates an Estimated Plan. (Shortcut key: **CTRL-L**)
Get Actual Plan	Generates an **Actual Plan (Section 5.5)**. (Shortcut key: **F5** or **CTRL-E**)
	The Query Execution warning message may be disabled with the preferences. **Edit menu → Preferences → Warn on actual execution**
Show Estimated Plan	Toggles the **Plan Diagram tab** between the Estimated and Actual Plan.
Post To SQLPerformance.com	Share the currently open plan on SQLPerformance.com. See also: **SQLPerformance.com integration (Section 5.6)**
Anonymize	Obfuscate the sensitive data in your plans. See Also: **Anonymize Plans (Section 5.6)**. Read more about this feature in this blog post by Aaron Bertrand.
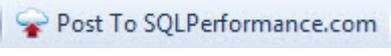
6.2 Command Text

The **Command Text** section shows an editable version of the query text for the statement or batch being analyzed. You can make changes to the query, then see how this impacts the execution plan by clicking on the **Get Estimated Plan** or **Get Actual Plan** button on the toolbar.

When retrieving a plan, you will be asked for the server, database, and credentials in order to connect (unless you launched Plan Explorer from within SSMS using our Management Studio Add-In).

To generate the actual plan, the query is run against the target server. By design, the actual query results are not returned to the Plan Explorer interface. By default, there is a message box explaining this behavior whenever an actual plan is captured.

Note: The Query Execution warning message may be disabled from the Preferences (**Edit menu → Preferences → Warn on actual execution)**.

If query execution fails a message will be displayed in the Status bar, located at the bottom of the Plan Explorer window.

6.3 Results

- **Statement Section (Section 6.3.1)**
- **General Query and Plan Info Section (Section 6.3.2)**
- **Plan Details Section (Section 6.3.3)**

6.3.1 Statement Section

The top section of the Results tab contains the Statement section or Statements tree. The Statement section represents one of the biggest differences between **Plan Explorer** and SSMS. Its function is critical when navigating multi-statement plans. What it shows is a logical breakdown of the entire plan tree, including all control structures, nested procedure calls, and estimated and/or actual metrics for each statement, including operation counts.

6.3.1.1 COMMON OPERATIONS

- Click on any statement in the grid to be shown the associated plan for that statement only.

- Click on any control structure (IF or WHILE) or EXEC higher up the tree to see a filtered view of the plan starting from that point.

- Sort the statements list and the highest cost trees and statements will immediately bubble up to the top.

6.3.1.2 CONTEXT MENU COMMANDS

Command	Description
Show Captured Statements Only	📄 **Note**: Applicable to only .queryanalysis files which originated in the **SentryOne Performance Analysis** integrated version of Plan Explorer.
	In the integrated version of Plan Explorer, by default only statements which exceed the trace filters are displayed. If a statement is part of the query call stack, but does not exceed the filters it is hidden. This helps you in quickly identifying the costliest parts of the query. If you uncheck this option, the entire query call stack will be shown.
Show Hidden Captured Statements	📄 **Note**: Applicable to only .queryanalysis files which originated in the **SentryOne Performance Analysis** integrated version of Plan Explorer.
	In the integrated version of Plan Explorer, you are able to hide statements in the Statements tree. If checked those statements will no longer be hidden.

Command	Description
Expand Subtree	Expands all the child nodes of the selected statement.
Expand All	Expands the entire Statement tree.
Collapse All	Collapses the entire Statement tree.
Copy Statement to Command Text	Copies the selected statement to the Command Text tab overwriting any text currently in the tab.
Copy All Statements to Command Text	Copies all statements belonging to the query call stack over to the Command Text tab.

> **NOTE:** When using a column filter that uses a percent value, use a decimal value for the filter. For example, if you'd like to the filter to be 50%, use 0.5.

6.3.2   General Query and Plan Info Section

The General Query and Plan Information section is located in the center panel by default. It contains the **Text Data tab** and the **Plan XML tab**.

6.3.2.1   COMMON OPERATIONS

- If you modify the query in the Command Text tab, the query text and XML will be synchronized when you generate a new Estimated or Actual plan from the toolbar.
- Like plans, the T-SQL statements are synchronized with the Statements Tree, meaning that if you select a row on the tree view it will auto-select the associated T-SQL statement and its execution plan, and vice versa.

6.3.2.2   CONTEXT MENU COMMANDS

Command	Description
Copy	Copies the selected text to the clipboard
Select All	Selects the text in the active tab
Save SQL	Save the Text Data as a **.sql file**.
Save Execution Plan	Save the Plan XML as a **.sqlplan file**.
Word Wrap	Wraps text to the current width of the panel.
Open	Open a .sql file in the Text tab.

6.3.3   Plan Details Section

PLAN DIAGRAM TAB

The **Plan Diagram** tab has an optimized layout algorithm which renders plans in a much more condensed view than SSMS, so more of the plan fits on the screen without having to zoom out. If needed, you can easily zoom in and out by holding the ctrl key and using your mouse scroll wheel. Optimized plan node labels prevent truncation of object names in most cases. To disable truncation completely so full object names are always visible use the Full Object Names context menu command.

The estimated cost of the operation is displayed above each node for maximum readability. These cost labels use color scaling by CPU, IO, or CPU+IO so highest cost operations are instantly obvious, even on larger plans. CPU+IO is used by default, this can be changed through the context menu Costs By option. All costs in the Plan diagram are shown to the first decimal place.

Through the context menu of the Plan Diagram you may also choose to show cumulative costs in lieu of per node costs; when combined with color scaling this feature makes it easy to see which subtrees are contributing most to the plan cost.

The connector line width can be scaled by either row or data size with the Line Widths By context menu command. The metric you choose to scale by, will be displayed above all connector lines within the Plan Diagram.

Plan Diagram Context Menu

Command	Description
Actual Costs	Show metrics captured during Actual Plan retrieval.
Estimated Costs	Show metrics associated with the Estimated Plan.
Per Node Costs	The default view, costs are per node, representing each individual nodes contribution to the total cost.
Cumulative Costs	Cost are cumulative, representing each individual nodes contribution plus the contribution of any of its children.
Costs By	**I/O** -Show Costs by I/O **CPU** - Show Costs by CPU **CPU + I/O** - Show Costs by CPU +I/O
Line Widths By	Rows Data Size (MB)
Show Color Scale	Node Costs are highlighted with an appropriate color scale, ranging from a light yellow to red.
Nested Procedures	Show nested procedures
Full Object Names	Objects are represented with full object names.
Expand Expressions	Expands all expressions.
Zoom	Various options for Zooming in and out. Alternatively use, **CTRL + Mouse Wheel**.

PLAN EXPLORER ADVANCED LAYOUT OPTIONS

Plan Explorer offers several advanced styles and options which can be used for structuring and shaping the layout of plans. These options allow you to create a very specific view of plans for both sharing and troubleshooting purposes.

> 💡 For a rundown of all the advanced layout options please see this blog post by Greg Gonzalez.

Option	Description
Scale	Scale the Plan Diagram between 6% - 400% of its original size. You may also use CTRL + mouse scroll wheel to perform this operation.
Filter	Filter nodes from the Plan Diagram based on Cost.
Stretch	Stretch the links between Plan Diagram nodes.
Flatten	Flattens the space between node levels in the Plan Diagram.

Option	Description
Mode	There are several Modes available which will change the layout of the entire Plan Diagram tree.
Link Style	The Link Style used between nodes in the Plan Diagram.
Rotation	Rotates the entire Plan Diagram 90°.
Defaults	All plan diagram options are reset.

PLAN TREE

The Plan Tree tab contains a tree representation of the plan's operations and associated metrics. Use the arrows to expand and collapse sections of the Plan Tree. Right click any column header and use the Column Chooser command to access additional plan metrics. Any significant differences between estimates and actuals are highlighted, as are possible problematic operations like scans and bookmark lookups.

TOP OPERATIONS

The **Top Operations** tab contains a sortable list view of all plan operations. The list view is sorted by total cost descending by default so you can immediately see which operations are the highest cost.

QUERY COLUMNS

The **Query Columns** tab shows exactly how data is accessed for each column used by the query, including the associated operation and estimated rows. This view makes it easy to identify bookmark lookups or index scans occurring because indexes are non-covering. Columns for related operations are grouped together (indicated by a thick separator bar), with the indexes used and bookmark (key / rid) lookup columns highlighted in orange. This makes it easy to see how indexes can be modified to eliminate lookups.

JOIN DIAGRAM

If there are joins present in the query, the Join Diagram tab will be present, showing a visual representation of the joins involved. Similar to the database diagram or Query Designer features within Management Studio - but only showing the columns that actively participate in joins. Plan Explorer looks past any views and shows the base tables and base columns, making it quicker to analyze queries that involve, for example, nested views.

PARAMETERS

The Parameters tab shows compiled and runtime parameters referenced in the plan, if any exist (otherwise the tab is not visible). For an estimated plan, it will only show the compiled parameter value; for an actual plan, you can compare compiled values with runtime, which can in turn help point out potential parameter sniffing issues.

EXPRESSIONS

If there are any expressions in the plan the Expressions tab is present. You can see the operator type, the shorthand expression name and the expanded version of the expression, as well as any other expressions that reference it or are referenced by it, allowing you to quickly see if you are performing the same operations over and over again.

TABLE I/O

The Table I/O tab is present for actual plans. This tab breaks down all the reads from a query by object, making it very easy to see where the majority of your I/O is coming from.

6.4   Wait Stats

When an **Actual Plan (Section 5.5)** is retrieved with related wait stats, the Wait Stats tab will be displayed next to the Results tab. The Wait Type is shown along with any associated Wait Time or Signal Time.

The Wait Stats tab can help you in identifying potential resource bottlenecks, including those related to memory pressure, CPU pressure, disk I/O, and networking. For general information about wait types see this technet article.

> **Note:** Waits Stats Analysis is only supported for queries executed against servers **running SQL Server 2008 or newer.**

6.5   Index Analysis

The **Plan Explorer Index Analysis** tab provides a sandbox environment for planning indexing strategy relative to a specific operation in a query.
For Index Analysis to be populated, the query must be executed from within the Plan explorer session using the "Get Actual Plan" menu option.

Index analysis involves 5 major areas.

THE STATEMENT TREE AND PLAN DIAGRAM

See the **Statement Section (Section 6.3.1)** and **Plan Details Section (Section 6.3.3)** for more more information.

THE TEXT DATA AREA

See the **General Query and Plan Info Section (Section 6.3.2)** for more more information.

THE INDEXES DATA GRID

The top row is the index score. This is based on several factors including covering, value density, sorting effectiveness, and seekability. 100% is the best possible score, however there are some situations that will prevent the possibility of a 100% score. One known cause is when columns are missing statistics. This will be indicated by the presence of a "?" in several of the column statistics values for a table column.

Column statistics are provided for each table column used by the query, with table columns used by the selected operation displayed in bold.

DATA GRID COLUMNS:

Table Column	Name of the column in the relevant database table
Density	How selective the row values are for the column (lower values are more selective)
Last Update	The date the column statistics were last updated
Avg. Length	Average length of data values
Estimated Size (MB)	The estimated size of the column values (in MB)
Output	Checked if the column is an output for the selected operation
Sort Type	Sort order (ASC, DESC)
Sort Order	Ordinal position of the column in the sort
Predicate	The search or join predicate(s) applied to the table column by the operation

Columns after the "Predicate" column are all index columns. These include both some special case indexes, as well as the indexes defined on the table.
They are listed from left to right in the following order:

1. Clustered index (if existing)
2. Index used by the operation (if not the clustered index)
3. Missing indexes as listed in the SQL Server missing indexes DMO
4. Recommended indexes generated by Plan explorer. On occasion, Plan Explorer can recommend a better index
5. The other non-clustered and/or column store indexes that exist for the table, ordered descending by score from left to right

Indexes can be modified, as well as scripted for create, drop, or drop and create. You may also manually update statistics using the button at the bottom of the index column.

INDEX COLORING:

Shades of green indicate that the ordinal position of the column is such that a favorable operation may occur, such as a seek over a scan. Darker shades of green indicate that the column is filtered.

Yellow indicates that the column is used in the query, and covered, but is not part of the left subset of the index key. Included columns are also listed in yellow.

Red indicates that the column involved is not covered by the index.

THE PARAMETERS DATA GRID

The parameters grid lists parameters used by the query. This differs from the Plan explorer Parameters tab in that it uses tipping point logic to help determine the type of operation (SEEK or SCAN) that the optimizer might choose for that parameter value. This is not an exact science, and should be used as a rough guideline.

The parameters grid provides a "Test Value" column which allows you to enter a new parameter value, and ask for a new estimated plan. This allows you to test different parameter values against the plan to see if the plan shape will change given that parameter value.

The small arrows to the right of parameter values allow you to quickly place the value of that parameter into the Test Values column.

THE STATISTICS HISTOGRAM CHART

The histogram chart displays the histogram steps for the index statistics visually. This allows you to quickly spot potential parameter sniffing issues.

This chart also works in conjunction with the parameters grid, and selected parameters that are present in the histogram will have their histogram buckets highlighted on the chart. This also works for parameter ranges.

> **Note:** See this blog post by Aaron Bertrand for detailed examples of Index Analysis

7	Appendix

7.1	File Types

Plan Explorer can open several different file types including raw plan XML files, SQL plan files, and .QueryAnalysis files, as well as deadlock files and Plan Explorer Sessions files.

7.1.1   PLAN EXPLORER SESSIONS

Plan Explorer uses Plan Explorer Session files (.pesession) to help you manage changes and version history as you fine tune queries. For an overview see the **PE Sessions (Section 5.1)** topic.

7.1.2   DEADLOCKS

Plan Explorer supports opening deadlock files (.xdl). For an overview see the **Deadlock Files (Section 5.3)** topic.

7.1.3   QUERY ANALYSIS FILE (.QUERYANALYSIS)

All information shown in Plan Explorer can be saved to the .QueryAnalysis file format by clicking the Save toolbar button. This proprietary, portable format contains the plan XML along with additional information, including actual statements and actual metrics captured via SQL Sentry's Top SQL trace.

This means that DBAs can easily save all estimated and actual data for a query to file for later analysis, or share it with other team members. For example, a DBA using the full version of SQL Sentry can send a .QueryAnalysis file to a developer for tuning, and all the developer needs is the Plan Explorer utility, versus the entire SentryOne Client application.

7.1.4   SQL PLANS AND EXECUTION PLAN XML

Plan Explorer also includes support for opening **Execution Plan files (.sqlplan)** and **Execution Plan XML files (.xml).**

7.2	Exception Reporter

If you are working in **SentryOne Plan Explorer** and see a small form pop-up unexpectedly, it means we've captured an unhandled exception. If you would, please enter your name and email address along with brief steps to reproduce into the textbox, then submit it – it comes to us via secure web service. If the exception seems to pertain to a particular plan, please also save it to a .QueryAnalysis file and email to support@sentryone.com.

7.3	Checking for Updates

Plan Explorer can check our servers to ensure you are using the most recent version, and offer to update your software if there is a newer version available. You can do this manually by using **Help → Check for Updates**. By default, it will also check each time you launch the application. You can disable this automatic check through the Preferences (**Edit menu → Preferences**).

8   Index