

# Examining the relationship between Precise for SQL Server database objects

This section includes the following topics:

- [About the Objects tab](#)
- [How the Objects tab is structured](#)
- [About Objects tab entities](#)
- [How the Objects tab can help you identify performance problems](#)

## About the Objects tab

It is important to understand the relationship between database objects and statements to make well-founded tuning decisions.

The Objects tab is a comprehensive browser that allows you to understand the relationships and associations between database schema objects. It also enables further understanding of the relationships between data dictionary components and the statements that access them. This information allows you to identify the statements that will be affected by changes to a database object, and the schema relationship impacts that are associated with an object change.

The Objects tab is also very useful for determining scaling strategies for your database. Are you considering scaling out or scaling up? If you thoroughly understand the relationship between a table, the SQL executed on the table and the resources consumed by these SQL statements, you can identify the tables that may be candidates for either a horizontal or vertical scale-out. To engage a successful scale-out, you need to ensure that the tables you are horizontally partitioning to different databases engage in enough activity and resource consumption to spread the load across multiple systems. The same is true for vertical scaling. If you know the overall resource consumption of the instance and the consumption per table, you can make educated decisions as to how to best vertically partition your tables into multiple databases.

The Objects tab is also very useful for scale-ups. Too frequently companies add CPUs to an existing server in their attempt to improve performance. Although occasionally successful, step one of any scale-up activity should be to identify the major latencies in the system and resolve them. Only once the major wait states are resolved, will a scale-up actually be able to improve performance. Otherwise, you may be simply wasting money and valuable resources.

The Objects tab also allows you to observe storage utilization, over time, at varying levels, such as, database, file group, database files.

While the Activity tab focuses on SQL tuning, the Object tab focuses on database object tuning. In many cases, you may want to tune a database object rather than a specific query or program. This is especially true if you did not personally write the application in question, as for example in ERP and CRM applications, and cannot therefore change the text of any statement.

Object tuning is a powerful tool. Taking into consideration important aspects regarding objects can ensure maximum application performance. Precise for SQL Server allows you to examine your index structure by displaying the usage of each index. Storage layout can be optimized by analyzing the object storage configuration and metric information.

Because Precise for SQL Server tracks changes in the database objects, the effect of schema changes on performance can be easily observed.

## How the Objects tab is structured

The Objects tab displays information on a selected entity and its associated entities. When you open the Objects tab the selected entity is by default Instance, meaning that information is displayed on the instance level. When you launch the Objects tab from the Activity tab with one of the following entities—Database, Table, Index, DB file or Locked Object—the tab is launched in context with the entity you selected in the Activity tab.

If you open the Objects tab from another tab, the historical settings (those settings which were selected when you left the tab, such as the last entity you drilled down to) are taken into account and the information displayed the last time you viewed the tab is displayed (similar to clicking the History button and returning to a previous tab).

The selected entity is always reflected in the Tab heading, which serves as a point of orientation. The highest level entity you can view information for in the Objects tab is the Instance level. You can select an instance from the Instance list.

The entities displayed in the Association area are associated with the selected entity displayed in the Main area. At times, the relationship between the entity displayed in the Main area and those displayed in the Association area is that of parent to child and sometimes it merely represents that there is a relationship between the selected entity and the entities displayed in the Association area.

## About the Main area in the Explore Objects tab

The Main area shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. You can, for example, focus exclusively on SQL Server Log data, or Instance Parameters, or Instance Parameters changes.

## About the Association area in the Explore Objects tab

The Association area provides corresponding information on the child entities. You can view information on one type of child entity at a time, such as instances only or databases only, by selecting a control from the View Controls list. The selection you make is reflected in the Association area only; the Main area remains unchanged. For some entities, tabs above the Association area enable you to view additional information: Clicking a tab changes the table columns for the same type of child entities.

From the Association area, you can also drill down to another entity by clicking a table row. A drill-down affects the whole tab: When you drill down to another entity, the Tab heading reflects the new selection, the Main area displays information on the newly selected entity, and the Association area shows data on its associated entities.

For example, when you want to view information on a specific logical file, choose Logical Files from the Association controls. The Association area changes to display all the files groups associated with the selected Logical File. Note that the Tab heading and the Main area remain unchanged. In the Association area, click the row of the logical file you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays an overview of the logical file you want to investigate, and the Association area shows general details on the file groups that are associated with the Logical File you selected.

See [About Precise for SQL Server tabs](#).

## About the Recommend tab

Precise for SQL Server uses the Microsoft® Index Tuning Wizard to achieve recommended indexes or statistics for the selected table. This helps the optimizer choose a better access plan and enhances the performance of the statements or batches that access the selected table. The recommendations are based on a workload. The workload contains all the statements stored in the PMDB that were executed during the selected time period that have an average duration time exceeding the value defined in the registry (where 0 is the default). The recommend indexes process only performs the recommend process on indexes or statistics additions.

The following information is displayed in the Recommend tab:

- Recommended Index/Statistics
- Recommendation details

## About viewing Recommended Indexes/Statistics

Displays the recommended indexes or statistics, as follows:

- **Icon.** Indicates whether the item is a recommended index or statistic.
- **Table.** Displays the name of the table to which the recommendation refers.
- **Unique.** Indicates whether the index is unique. This value is not relevant for recommended statistics.
- **Clustered.** Indicates whether the index is clustered. This value is not relevant for recommended statistics.
- **Columns List.** Displays a list of the index's or statistic's columns separated by commas. The sort order (ASC, DESC) is also displayed for recommended indexes.

## About viewing Recommendation details

Displays details on the recommended index or statistic. The following tabs are available:

- **Columns.** Displays a list of all the columns in the recommended index's or statistic's table as follows:
  - **Icon.** Indicates whether the data in the column will be stored in the index in ascending (ASC) or descending (DESC) order. No icon is displayed for columns that are not part of the recommended index.
  - **Column.** Name of the column.
  - **Key Number.** Key number of the column in the index.
  - **Column Type.** The physical storage type of the column.
  - **Part of Index.** Indicates whether the column participates in an existing index.
  - **First Key of Index.** Indicates whether the column participates as the first key in an existing index.
  - **Used.** Number of times the column is used in the breakdown of the workload statements according to the following types: Range Scan, Table Scan, Sort and Join.
- **DDL.** Displays the DDL text of the recommended index or statistic.
- **Table's Indexes/Statistics.** Displays a list of the existing indexes or statistics of the table selected in the recommendation area.

The following information is displayed:

  - **Icon.** Indicates whether the item is an index or statistic.
  - **Name.** Displays the name of the index or statistic.
  - **Unique.** Indicates whether the index is unique. This value is not relevant for statistics.
  - **Clustered.** Indicates whether the index is clustered. This value is not relevant for statistics.
  - **Columns List.** Displays a list of the columns of the index or statistic separated by commas. The sort order (ascending, descending) is also displayed for indexes.
- **Analyzed Statements.** Displays a list of all the statements in the workload.

The following information is displayed:

  - **Statement.** Indicates the statement ID.
  - **In MS-SQL.** Displays the resource consumption (as a stacked bar) of the specified statement within the selected time period.
  - **Text.** Displays the statement's text.

## About Objects tab entities

The following entities can be examined in the Objects tab:

- Instance
- Database
- Active Objects
- Table and Table Contain Same Column

- Index and Unused Index
- Partition
- Column
- View and View Contain Same Column
- User
- Logical File
- File Group
- Statistics
- Stored Procedure
- Function
- Trigger
- Statement
- Batch
- Physical Disk
- Logical Volume
- Disk Partition

## About the Instance entity

The Instance entity displays configuration information, configuration changes, as collected by the Collect Schema Changes process, and performance information, as collected by the Precise for SQL Server Collector for the entire instance.

The following views are available:

- Overview
- SQL Server Log
- Instance Parameters
- Instance/Database Changes
- Databases Created/Dropped

## About getting an overview of instance configuration and performance information

The Overview displays configuration and performance information for the entire instance. The Overview is divided into the following areas:

- **Server Details.** Displays server configuration details such as version and number of CPUs.
- **Instance Details.** Displays SQL Server instance configuration details such as version, service pack and server details.
- **In MS-SQL.** Displays resource consumption, over the selected time period, of the selected instance, as collected by the Precise for SQL Server Collector.
- **Space Allocation.** Displays the total free space and total used space (in MB) over the selected time period, of all the files in the selected instance, as collected by the Collect Space Utilization process.

## About viewing SQL server log messages

The SQL Server Log view displays the SQL Server log messages from the recent error log file, for the selected time period.

## About viewing instance configuration parameters

The Instance Parameter view displays the selected instance's configuration parameters. This includes the possible range values and configuration values, for each parameter. A change icon is displayed if the Collect Schema Change Process detects a change in any of the parameters.

Precise for SQL Server reports changes that occurred within the selected time period. If you observe that a parameter was changed, select Instance Parameter Changes from the View controls, to view additional information.

## About viewing changes to instance parameters and database options

The Instance/Database view displays the changes to instance parameters and database options vs. resource consumption, over the selected time period, to enable you to identify configuration performance changes caused by these actions, as follows:

- **Instance/Database Changes.**
  - **Sample Date.** The date when the changes were captured by the Collect Schema Changes process.
  - **Database.** The name of the database where the change occurred. For Instance Configuration changes, the value is N/A.
  - **Data Changed.** Displays one of the following values:
    - **Schema.** Any change in the database schema.
    - **Database Option.** A description of the database option that was changed
    - **Configuration Parameter.** A description of the instance configuration parameter that was changed.
  - **New Value.** Displays the new value that resulted following the change.
  - **Old Value.** Displays the previous value that preceded the change.
- **In MS-SQL.** Displays resource consumption of the selected instance, over the selected time period, as collected by the Precise for SQL Server Collector.
- **Schema Changes.** Displays an overtime graph that indicates whether changes occurred in the database, or a breakdown of the instances into the following groups:
  - **Schema Changed.** Any change in one of the database's Schema.
  - **Database Option Changed.** Any change in one of the database's options.
  - **Instance Property Changed.** Any change in one of the instance configuration parameters.

The following configuration parameters that can be displayed:

- Maximum recovery interval in minutes
- Number of locks for all users
- Number of open database objects
- Maximum worker threads.
- Network packet size
- Memory for index create sorts (KB)
- Priority boost
- Minimum memory per query (KB)
- Query wait(s)
- Set working set size
- Affinity mask
- Cost threshold for parallelism
- Maximum degree of parallelism
- Minimum Server Memory size (MB)
- Maximum Server Memory size (MB)
- Maximum estimated cost of query allowed to run by query governor
- User mode scheduler uses lightweight pooling
- Recovery Model Changed
- Auto Update Statistics Changed
- Ansi Null Changed
- Auto Shrink File Changed
- Close Cursors On Commit Changed
- Status Changed
- Auto Create Statistics Changed
- Truncate Log Changed
- Compatibility Level Changed

### About viewing databases that were created or dropped

The Databases Created/Dropped view displays the databases that were dropped or created vs. resource consumption over the selected time period to enable you to identify performance changes caused by these actions.

- **In MS-SQL.** Displays resource consumption of the selected instance, over the selected time period, as collected by the Precise for SQL Server Collector.
- **Schema Changes.** Displays resource consumption of the selected instance, over the selected time period, as collected by the Precise for SQL Server Collector.

### About the Database entity

The Database entity displays configuration information, configuration changes collected by the Collect Schema Changes process and performance information collected by the Precise for SQL Server collector of the selected database.

The following views are available:

- Overview
- Database Options
- Database Option Changes
- Object Changes

### About getting an overview of configuration, performance and storage information for a selected database

The Overview displays configuration, performance and storage information for the selected database, as follows:

- **Properties.** Displays general database details, such as, status, collation, and compatibility level.
- **In MS-SQL.** Displays resource consumption of the selected database, over the selected time period, as collected by the Precise for SQL Server Collector.
- **Space Allocation.** Displays total free space and total used space (in MB), over the selected time period, of all the files in the selected database, as collected by the Collect Space Utilization process.
- **Schema Changes.** Displays an overtime graph that indicates whether changes were made to the database objects and the database configuration options.
- **Data vs. Log.** Displays total free space and total used space (in MB), over the selected time period, of all the files in the selected database, as collected by the Collect Space Utilization process, and displayed according to file types as datafiles or log files.

### About viewing database options

The Database Options view displays the database options, such as Auto Create Statistics and Auto Shrink File, etc. If the Collect Schema Changes process detects a change in any of the options, an icon is displayed in the first column. Select the Database Option Changes view to observe additional information regarding the detected change. The following information is displayed:

- **Database Options.** Displays the current value of the database options.
- **In MS-SQL.** Displays resource consumption information of the selected database, over the selected time period.

- **Schema Changes.** Displays option changes of the selected database, over the selected time period, as detected by the Schema Change Process.

## About viewing database option changes

The Database Option Changes view displays the database option changes vs. resource consumption, over the selected time period, so as to enable you to identify configuration changes that led to performance problems.

The following database options changes are displayed:

- Recovery Model Changed
- Ansi Null Changed
- Close Cursors On Commit Changed
- Auto Create Statistics Changed
- Compatibility Level Changed
- Auto Update Statistics Changed
- Auto Shrink File Changed
- Status Changed
- Truncate Log Changed

The following information is displayed:

- **Database Option Changes.** Displays the changes made to the database options of the selected database, over the selected time period.
- **In MS-SQL.** Displays resource consumption information of the selected database, over the selected time period.
- **Schema Changes.** Displays option changes of the selected database, over the selected time period, as detected by the Schema Change Process.

## About viewing object changes

The Object Changes view displays the schema changes vs. resource consumption over the selected time period to enable you to identify performance changes caused by these actions, as follows:

- **Object Changes.**
  - **Sample Date.** The estimated last change date of the table (and /or its sub-objects). The change is reported by the Precise for SQL Server Schema Change Process, which runs, by default, once a day. This means that the Sample Date actually represents the date when the Schema Change Process was executed.
  - **Object Type.** The Object type can be one of the following: Table, View, Stored procedure, Trigger, Foreign key, Scalar function, Inline Table function, Table function, and Extended storage procedure.
  - **Object.** Displays the object name.
  - **Object Owner.** Displays the owner of the object.
  - **Change Type.** Can be one of the following:
    - **Created.** Indicates object was created.
    - **Dropped.** Indicates object was dropped.
    - **Index.** Indicates any change that was made in the object's index(es).
    - **Change in structure.** Indicates any change that was made in the object's column(s) and the creation or deletion of the object's trigger(s).
    - **Change in properties.** Indicates any change that was made in one of the following object's properties: Name, Owner, File Group and Exec is ANSI null on.
    - **Partition function.**
    - **Partition schema.**
    - **Partition function values.**
- **In MS-SQL.** Displays resource consumption of the selected database, over the selected time period, as collected by the Precise for SQL Server Collector.
- **Schema Changes.** Displays an overtime graph that indicates whether changes were made to the database objects, broken down into the following groups:
  - **Index.** Indicates any change, addition, or deletion of an index.
  - **Object Change in Structure.** Indicates any change in the object's structure. For example, whether a new column was added to a table.
  - **Object Change in Property.** Indicates any change in the object's property. For example, whether the name of the object was changed.
  - **Object Created/Dropped.** Indicates whether any of the objects sampled (table, view, stored procedure, etc.) was dropped or created.

## About the Database entity in the Association area

You can view the entities associated with the selected Database entity in the Association area. It is possible to associate to the Database entity from the Instance entity.

## About the Active Objects entity

The Active Objects entity pinpoints objects (tables including their indexes statistics) that may be a good candidate for tuning. If the object is an SQL Server 2005, additional information regarding I/O operations and waits, contentions and logical operations is also displayed. Clicking on the object's icon launches to the Objects tab with the selected object in context.

The Active Objects entity indicates the following:

- the name of the user, object, and database it is associated with
- a summary of the time it spent in SQL Server, during the time frame you selected

- its size (in pages)

See [About the Table and Table Contain Same Column entities](#) and [About the Index and Unused Index entities](#).

## About the Active Objects Association area

The following additional information is available for SQL Server 2005 objects:

- **I/O Operations.** Indicates the amount of I/O operations performed on the table or index, over the selected time period.
- **I/O Waits.** Displays the I/O wait for the table or index, over the selected time period.
- **Contentions.** Displays the number of contentions experienced by the table or index, over the selected time period.
- **Contention Waits.** Displays the contention wait data for the selected table or index, over the selected time period.
- **Logical Operations.** Displays how many range scan and Singleton lookups were performed on the table or index, over the selected time period.
- **Size (pages).** Displays the size of the table or index (in pages).

## About the Table and Table Contain Same Column entities

The Table entity displays general details regarding resource consumption, space utilization and schema changes for the selected table.

The Table Contain same Column Entity displays tables that have the same column as the column displayed in the Main area. The information displayed is the same as displayed in the Table entity and can only be associated to from the Column entity.

The following views are available:

- Overview
- Show Contig
- Change Log Summary
- Change Log Detailed
- Usage Report
- Common Access Patterns (SQL Server 2005 only)
- Operational Statistics (SQL Server 2005 only)

## About getting an overview of table entities

The Overview displays general table properties, along with the resource consumption, space utilization, schema changes and number of rows in the selected table, over the selected time period.


- **Properties.** Displays general table details, such as, number of columns, file group, and whether or not the table uses a clustered index.
- **In MS-SQL.** Displays the Using CPU and I/O Wait counters of the selected table, over the selected time period, based on its contribution to overall performance, when accessed by the execution plan.
- **Space Allocation.** Displays the amount of free space and used space (in MB) in the selected table, over the selected time period, and specifies whether the space is populated by an index or data.  
The values may be incorrect due to out-of-date space usage information. To update the values, use the `DBCC UPDATEUSAGE` command to recalculate the space usage fields. For more information, see *SQL Server Books Online*.
- **Schema Changes.** Displays an overtime graph that indicates whether changes were made to the selected table and a breakdown of its indexes or columns into the following groups:
  - **Table Change in Property.** Indicates any change made to one of the table's properties, such as if the table was renamed.
  - **Table Change in Structure.** Indicates any change made to the table's structure, such as whether a new column was added to the table.
  - **Index Change in Property.** Indicates any change made to one of the index's properties, such as whether the index was changed from clustered to non-clustered.
  - **Index Change in Structure.** Indicates any change made to the index's structure, such as whether a new key was added to the index.
  - **Index Created/Dropped.** Indicates any addition or deletion of indexes.
- **Rows.** Displays the number of rows in the selected table, over the selected time period, as collected by the Collect Space Utilization process.

## About viewing data storage and fragmentation information in the Show Contig view

The Show Contig view displays data storage and fragmentation information for the selected table. The following information is displayed:

- **Minimum Record Size.** Indicates the minimum record size in the data pages.
- **Maximum Record Size.** Indicates the maximum record size in the data pages.
- **Average Record Size.** Indicates the average record size in the data pages.
- **Number of Forwarded Records.** When there is not enough free space to write an updated record in its page, SQL Server allocates the updated record to another page and instructs the old record to point to the new location.  
The Number of forwarded records is the number of records allocated to another page, since there was no free space in the original page. Accessing forwarded records adversely affects performance because additional I/O is performed. Check the number of forwarded records in your table or index in relation to the total number of records. If the ratio is high, consider reorganizing the table or index to achieve a better performance. To reorganize an index, use `DBCC DBREINDEX`. If the index is a clustered index, a Reindex action will reorganize the data pages.
- **Pages.** Displays the number of data pages.
- **Extents.** Displays the number of extents used to store the data pages.
- **Extent Switches.** The number of times SQL Server moved from one extent to another while scanning the data pages (not including the first extent access).
- **Scan Density.** Indicates the ratio between the optimum number of extent switches and the actual number of extent switches. This value determines the general level of fragmentation in a table. The higher the value, the better. The scan density is not valid, if the table spans multiple files.

- **Best Count.** The optimum number of extent switches. This number can be achieved, if the extents and pages are perfectly linked, and there is no fragmentation.
- **Actual Count.** Number of times the SQL Server moved from one extent to another while scanning the data pages (including the first extent access).
- **Average Free Bytes per Page.** Average free bytes per page in the data pages.
- **Average Page Density (%).** The average page density is a percentage indicating how full the data pages are. Space that cannot be utilized due to row size constraints is counted as used. The higher the percentage, the better.  
When space is available within the index pages, that is, when the indexes are not making the most efficient use of space, scanning the table involves more read operations than if no free space were available on the pages. However, keeping free space on the pages is sometimes desirable to accommodate future expansion in the storage of the table's data and to reduce the potential for page splits. You can request to keep free space available by specifying a low fill factor value, when you create an index. For example, a fill factor of "10" means that the pages will be 10% full while creating the index. Having room on a page means that there is space to insert more rows without having to split a page. Splitting is a relatively expensive operation and can lead to a break of contiguity between pages, because the new page is not usually contiguous to the page being split.
- **Logical Fragmentation (%).** The logical fragmentation indicates how well ordered the data pages are inside an extent and between extents. "Well ordered" means that the number of the current page scanned is higher than the previous one. The logical fragmentation is computed by dividing the number of out-of-order pages by the number of the data pages. The lower the value, the less fragmentation there is. This is relevant only when there is a clustered index on the table, because ordered pages are important.
- **Extent Fragmentation (%).** The extent fragmentation indicates how well the extents, which contain the data pages, are physically contiguous on the disk.  
The page number of its first page identifies an extent. If the extents 8, 16, 24, or 32 make up an index, then we can see that the extents are contiguous on the disks, which means there are no gaps. On the other hand, if the extents 8, 24, 32, and 64 make up an index, then we can see that we have a gap. The extent fragmentation is computed by dividing the number of gaps by the number of extents, so in this example, the extent fragmentation is 1/4, or 25 percent.  
The lower the value, the less fragmentation there is.  
This number is not relevant to heaps, because physical order has no importance in the data pages.

 By default, not all dbcc fields are displayed in the Show Contig view.

## About viewing changes made to the table and index levels in the Change Log Summary

The Change Log Summary view displays changes made to the table and index levels vs. resource consumption, over a selected time period, to enable you to locate schema changes that caused performance problems. The following information is displayed:

- **Change Log Summary.**
  - **Sample Date.** Indicates the estimated last change date of the table (and/or its sub-objects). This value reflects the date that the change was first identified by the Precise for SQL Server Schema Changes Process, not the date the object was changed.
  - **Object Type.** Indicates whether the object is a table or index.
  - **Object.** Displays the table or index name.
  - **Change Type.** Indicates the type of change detected. The following change types are reported:
    - **Created.** Indicates that the index was created
    - **Dropped.** Indicates that the index was dropped
    - **Change in Structure Table.** Indicates any change that was made to the table's column(s) and the creation or deletion of the table's trigger(s).
    - **Index.** Indicates any change that was made to the index keys of the index or to the index depth.
    - **Change in Properties Table.** Indicates any change that was made to one of the following table's properties: Name, Owner or File Group.
    - **Index.** Indicates any change that was made to one of the following index's properties: File Group, Fill Factor, Is Auto Statistics, Is Clustered Index, Is Pad Index, Is Page Lock Disallowed and Is Row Lock Disallowed.
- **In MS-SQL.** Displays resource consumption of the selected table, over a selected time period, based on its contribution to overall performance, when accessed by the execution plan.
- **Schema Changes.** Displays an overtime graph that indicates whether changes were made to the selected table and a breakdown of its indexes or columns into the following groups:
  - **Table Change in Property.** Indicates any change made to one of the table's properties, such as if the table was renamed.
  - **Table Change in Structure.** Indicates any change made to the table's structure, such as whether a new column was added to the table.
  - **Index Change in Property.** Indicates any change made to one of the index's properties, such as whether the index was changed from clustered to non-clustered.
  - **Index Change in Structure.** Indicates any change made to the table's index properties, such as whether a new key was added to the index.
  - **Index Created/Dropped.** Indicates any addition or deletion of indexes.

## About viewing schema changes made on a selected table in the Change Log Detailed view

The Change Log Detailed view details the schema changes made on the selected table and its sub-objects (column, index and index key).

The following table information is monitored by the schema change process:

- Name
- File Group
- Exec is ANSI null on
- Table has Cluster Index
- Insert Trigger Count
- Owner
- No. of Columns

- Table has Index
- Delete Trigger Count
- Update Trigger Count
- Partition Schema
- Partition Function

## About viewing usage information and access methods in the Usage Report

The Usage Report view displays usage information and access methods based on statements stored in the PMDB. The following information is displayed:

- **Report was run with the following parameters.** Displays usage information of the selected index, as follows:
  - **Start Date.** Indicates the minimum execution date for a statement, within the selected time period.
  - **End Date.** Indicates the maximum execution date for a statement, within the selected time period.
- **Statement Explain.** Displays the following items:
  - **Statement Access the Object.** Indicates the total number of statements that were explained successfully in the PMDB that refer to the selected table, within the selected time period.
  - **Statement Not Access the Object.** Indicates the total number of statements that were explained successfully in the PMDB that do not refer to the selected table, within the selected time period.
  - **Statements Not Explained Yet.** Indicates the total number of statements in the PMDB that were not yet explained, within the selected time period.
  - **Statements Explained Unsuccessfully.** Indicates the total number of statements in the PMDB that were explained unsuccessfully, within the selected time period.
- **Table Profile.** Displays a breakdown, by statement type, of the number of statements that referred to the specified table (SELECT, INSERT, UPDATE, DELETE); for example, 100 statements performed SELECT on the specified table. The In MS-SQL counters (Using CPU and I/O Wait) are also displayed.

## About viewing common access patterns for tables or index (SQL Server 2005 only)

The Common Access Patterns view displays information that indicates to what extent a table or index was used. The following information is displayed:

- **User Operations.** Displays the number of user seeks, scans, lookups or updates.
- **System Operations.** Displays the number of system seeks, scans, lookups or updates.
- **User vs. System Operations.** Graph comparing user operations to system operations, over the selected time period.
- **Last User or System Seek, Lookups, Scans or Updates.** Displays when the last user or system seek, lookup, scan or update operation occurred.

## About viewing operational statistics

The Operational Statistics view displays information that indicates whether or not an index or table is being used. The following graphs are displayed:

- Range scan vs. Singleton lookups
- Physical I/O operations
- Locking and Latching contentions
- Locking and Latching contention waits

## About the Table and Table Contain Same Column entities Association area

You can view general details of the entities associated with the selected Table entity in the Association area. It is possible to associate to the Table entity from the following entities: Database, Column, Index, Physical Disk, Logical Volume, HP Storage Device, HDS Storage Device, Statement, Batch, Stored Procedure, Function and Trigger.

The following additional information is available for SQL Server 2005 objects:

- **Partitions.** Indicates the number of partitions defined for the selected table.
- **Table Partitioned.** Indicates whether or not the table is partitioned.
- **Partition Schema.** Indicates under which file groups to map the partition.
- **Partition Function.** Displays the name of the function that defines how the table should be partitioned.

## About the Index and Unused Index entities

The Index entity displays general details regarding resource consumption, space utilization and schema changes for the selected index.

The Unused Index entity displays general details regarding space utilization and schema changes for the selected index. An index that is not in use is identified through the Execution plan of the statements stored in the PMDB (SQL Server 200) or through the relative dynamic views (DMV) supplied by SQL Server 2005.

The following views are available:

- Overview
- Show Contig
- Statistics
- Schema Changes
- Usage Report
- Common Access Patterns (SQL Server 2005 only)
- Operational Statistics (SQL Server 2005 only)



## About getting an overview of space utilization and schema changes for a selected index

The Overview displays space utilization and schema changes for a selected index, as follows:

- **Properties.** Displays general details on the index such as number of keys, fill factor, and whether the index is clustered or unique.
- **In MS-SQL.** Displays Using CPU and I/O Wait of the selected index over a selected time period, based on its contribution to overall performance when accessed by the execution plan.
- **Space Allocation.** Displays the amount of free space and used space (in MB) in the selected index, over the selected time period.



The values may be incorrect due to out-of-date space usage information. To update the values, use the `DBCC UPDATEUSAGE` command to recalculate the space usage fields. For more information, see *SQL Server Books Online*.

- **Schema Changes.** Displays an overtime graph that indicates whether there were index changes, broken down into the following groups:
  - **Change in Property.** Indicates any change made to the index properties, such as if the index was changed from clustered to non-clustered.
  - **Change in Structure.** Indicates any change made to the index structure, such as if a new key was added to the index.
- **Rows.** Displays the number of rows in the selected index, over the selected time period, as collected by the Collect Space Utilization process.

## About viewing data storage and fragmentation information of a selected index in the Show Contig view

The Show Contig view displays data storage and fragmentation information for the selected index. The following information is displayed:

- **Minimum Record Size.** Displays the minimum record size of the pages in the leaf level of the index.
- **Maximum Record Size.** Displays the maximum record size of the pages in the leaf level of the index.
- **Average Record Size.** Displays the average record size of the pages in the leaf level of the index.
- **Number of Forwarded Records.** When there is not enough free space to write an updated record in its page, SQL Server allocates the updated record to another page and instructs the old record to point to the new location.

The Number of forwarded records is the number of records allocated to another page, because there was no free space available in the original page.

Every access to a forwarded record requires an additional step than access to a regular record, which affects performance. Check the number of forwarded records in your table or index in relation to the total number of records. If the ratio is high, consider reorganizing the table or index to achieve better performance. Use `DBCC DBREINDEX` to reorganize an index. If the index is a clustered index, reindexing will reorganize the data pages.
- **Pages.** Displays the number of pages in the leaf level of the index.
- **Number of Extents.** Number of extents used to store the pages in the leaf level of the index.
- **Extent Switches.** Displays the number of times SQL server moved from one extent to another while scanning the leaf level pages (not including the first extent access).
- **Scan Density.** Indicates the ratio between the optimum number of extent switches and the actual number of extent switches. This value determines the general level of fragmentation in an index. The higher the value, the less fragmentation there is in the index. The scan density is not valid if the table spans multiple files.
- **Best Count.** Displays the optimum number of extent switches. This number can be achieved, if the extents and pages are perfectly linked, and there is no fragmentation.
- **Actual Count.** Displays the number of times SQL Server moved from one extent to another while scanning the index pages (including the first extent access).
- **Average Free Bytes per Page.** Displays the average free bytes per page in the leaf level pages.
- **Average Page Density (%).** The average page density is a percentage indicating how full the index pages are.

Space that cannot be utilized due to row size constraints is counted as used. The higher the percentage, the better.

When space is available within the index pages, that is, when the indexes are not making the most efficient use of space, this means that scanning the table involves more read operations than if no free space were available on the pages. However, maintaining free space on the pages is sometimes desirable to accommodate future expansion in the storage of the table's data and to reduce the potential for page splits. You can request to keep free space available by specifying a low fill factor value, when you create an index. For example, a fill factor of "10" means that the pages will be 10% full while creating the index. Having room on a page means that there is space to insert more rows without having to split a page. Splitting is a relatively expensive operation and can lead to a break of contiguity between pages, because usually the new page is not contiguous to the page being split.
- **Logical Fragmentation.** Indicates how well ordered the leaf pages are inside an extent and between extents. Well ordered means that the number of the current page scanned is higher than the previous one. Logical fragmentation is computed by dividing the number of out-of-order pages by the number of pages in the table. The lower the value, the less fragmentation there is in the index. This is relevant only when there is a clustered index on the table because then there is significance to ordered pages.
- **Extent Fragmentation (%).** The extent fragmentation indicates how well the extents, which contain the leaf pages, are physically contiguous on the disk.

The page number of its first page identifies an extent. If the extents 8, 16, 24, or 32 make up an index, then we can see that the extents are contiguous on the disks, which means there are no gaps. On the other hand, if the extents 8, 24, 32, and 64 make up an index, then we can see that we have a gap. The extent fragmentation is computed by dividing the number of gaps by the number of extents, so in this example, the extent fragmentation is 1/4, or 25 percent.

The lower the number, the less fragmentation there is.

This number is not relevant to heaps, because physical order has no importance in the data pages.



By default, not all `dbcc` fields are displayed on the Show Contig view.

## About viewing distribution statistics for a selected index

The Statistics view displays distribution statistics for the selected index.

## General

Displays general information on the distribution statistics, as follows:

- **Statistics Updated.** Indicates the last time the statistics were updated.
- **Row Sampled.** Indicates the number of rows sampled for the statistics calculation.
- **Average Key Length.** Indicates the average length of all the keys in the index.
- **Steps.** The number of steps that were sampled and used in the statistics calculation. A "step" is a range of key values (the first key in the index) that were sampled.  
The following values are recorded for each step:
  - High value in the range
  - Density =  $1 / (1 + \text{No. of distinct values in the range})$
  - No. of rows in the range (not including the High value)
  - No. of rows equal to the "High value"
  - No. of distinct values in the range (not including the "High value")
  - Avg. No. of rows per distinct value ("No. Of rows in range" / "No. of distinct values in the range")
- **Rows Sampled.** The percentage of rows sampled for the statistics calculation.

## About Density

Indicates how selective an index is, as follows:

- **Density.** This indicator provides a measure of how selective the index is. The more selective an index is, the more useful it is, since this means that more rows can be eliminated from consideration. For example, a unique index is the most selective index since each index entry can point to only one row.  
Density =  $1 / \text{No. of unique values}$ . Density values can be between "0" and "1". The lower the Density value, the more selective the index.
- **Average Length.** Indicates the average length of the specified set of keys.
- **Columns.** The left-based subset of key columns in the index (including the first column alone). If there is a clustered index on the table, its keys are considered part of the key in every non-clustered index row, because non-clustered index rows point to the clustered index rows. For example, if there is a non-clustered index with the columns, "CustomerID" and "OrderDate". The table also contains a clustered index, "OrderID". The columns will be:
  - CustomerID
  - CustomerID, OrderDate
  - CustomerID, OrderDate, OrderID

## About Distribution

Displays additional distribution statistics for the selected index, as follows:

- **High Value.** Each row in the Distribution table represents information about a range of key values (the first key in the index). This range is called a "Step". "High value" is the upper value in the range.
- **Rows in Range.** Indicates the number of rows in the sample that have the specified key range values, not including the high value in the range.
- **Rows Equal to High Value.** Indicates the number of rows from the sample that have the same value as the specified High value
- **Distinct Values in Range.** Indicates the number of distinct values inside the range not including the High value.
- **Rows per Distinct Values in Range (Avg).** The average number of rows, per distinct value, in the specified range, not including the High value in the range. This value is computed by dividing the number of rows in the range by the number of distinct values in the range (the number of distinct values in the range always being greater than zero).
- **Density.** Step Density means the uniqueness of the data values. This information provides a measure of how selective the index is. The more selective an index is, the more useful it is, because this means more rows can be eliminated from consideration. A unique index, for example, is the most selective index, where each index entry can point to only one row.  
Step Density =  $1 / (1 + \text{No. of distinct values in the range})$  In this case, we calculate the density of the current step.  
Step Density values can be between "0" and "1". The lower the Step Density value, the more unique the data values.

## About viewing schema changes for a selected index and index key

The Schema Changes view shows a table that summarizes the schema changes made on the selected index and index keys. For example, this would include whether the first key of an index was dropped, making the index unusable in statements.

The following index information is monitored by the schema change process:

- File Group
- Fill Factor
- Is Pad Index
- Index Depth
- Is Page Lock Disallowed
- No of Keys
- Is Clustered Index
- Is Unique
- Is Auto Statistics
- Is Row Lock Disallowed
- Position
- Order by
- Partition Schema
- Partition Function

## About viewing a usage report

The Usage Report view displays usage information and access methods based on statements stored in the PMDB. The following information is displayed:

- **Report was run with the following parameters.** Displays usage information of the selected index, as follows:
  - **Start Date.** Indicates the minimum execution date for a statement, within the selected time period.
  - **End Date.** Indicates the maximum execution date for a statement, within the selected time period.
- **Statement Explain.** Displays statement explain, as follows:
  - **Statement Access the Object.** Indicates the total number of statements that were explained successfully in the PMDB that refer to the selected index, within the selected time period.
  - **Statement Not Access the Object.** Indicates the total number of statements that were explained successfully in the PMDB that do not refer to the selected index, within the selected time period.
  - **Statements Not Explained Yet.** Indicates the total number of statements in the PMDB that were not yet explained, within the selected time period.
  - **Statements Explained Unsuccessfully.** Indicates the total number of statements in the PMDB that were explained unsuccessfully, within the selected time period.
- **Table Profile.** Displays a breakdown, by statement type, of the number of statements that referred to the specified table (SELECT, INSERT, UPDATE, DELETE); for example, 100 statements performed SELECT on the specified table. The In MS-SQL counters (Using CPU and I/O Wait) are also displayed.

## About viewing common access patterns for indexes (SQL Server 2005 only)

The Common Access Patterns view displays information that indicates to what extent an index was used. The following information is displayed:

- **User Operations.** Displays the number of user seeks, scans, lookups or updates.
- **System Operations.** Displays the number of system seeks, scans, lookups or updates.
- **User vs. System Operations.** Graph comparing user operations to system operations, over the selected time period.
- **Last User or System Seek, Lookups, Scans or Updates.** Displays when the last user or system seek, lookup, scan or update operation occurred.

## About viewing operational statistics

The Operational Statistics view displays information that indicates whether or not the index on a table is being used. The following graphs are displayed:

- Range scan vs. Singleton lookups
- Physical I/O operations
- Locking and Latching contentions
- Locking and Latching contention waits

## About the Index and Unused Index entities Association area

You can view general details on the entities associated with the selected Index entity in the Association area. It is possible to associate to the Index entity from the following entities: Database, Table, Column, View, User, File Group, Physical Disk, Logical Volume, HP Storage Device, HDS Storage Device, Statement and Batch.

The following additional information is available for SQL Server 2005 and later objects:

- **Partitions.** Indicates the number of partitions defined for the selected index.
- **Index Partitioned.** Indicates whether or not the index is partitioned.
- **Partition Schema.** Indicates under which file groups to map the partition.
- **Partition Function.** Displays the name of the function that defines how the index should be partitioned.
- **Index Aligned.** Indicates whether or not the index is aligned with its table.
- **Number of Non-key Columns.** Displays the number of non-key columns used within an index.
- **Index Non-Key Columns.** Displays the non-key column names.

## About the Partition entity (SQL Server 2005 only)

The Partition entity displays information on how an index or table is partitioned. All tables and indexes in a database are considered to be partitioned even if they are made up of only one partition. The data of partitioned tables and indexes is divided into units that can be spread across more than one file group in a database. The data is partitioned horizontally, so that groups of rows are mapped into individual partitions.

The following view is available:

- Overview

## About getting an overview of an index or table that is partitioned

The Overview displays general details on the selected index or table, as follows:

- **Properties.** The following information is displayed for each partition:
  - **Object.** Name of the object being partitioned.
  - **Partition.** Displays the partitions id.
  - **Filigreed.** Indicates where the data related to a partition is stored.
  - **Rows.** Displays the number of rows per partition.
  - **Max Value.** Indicates the upper value of a partition.
  - **Min Value.** Indicates the lower value of a partition.

- **Upper Boundary.** Indicates whether data falling on the maximum value will be assigned to the next partition or to the same partition—yes will assign it to the next partition; no to the same partition it falls in.
- **Space Allocation.** Displays the amount of space utilized by the partitions.
- **Rows.** Displays the number of rows in a partition, over the selected time period.

## About the Partitions entity Association area

You can view general details on the entities that are associated with the selected Partition entity in the Association area. It is possible to associate to the Partition entity from an Index or Table entity.

## About the Column entity

The Column entity displays general details and schema changes for the selected column. The following views are available:

- Overview
- Schema Changes

## About getting an overview of a selected column

The Overview displays general details on the selected column such as type, length, and default value.

## About viewing schema changes made on a selected column

The Schema Changes view shows a table that summarizes the schema changes made on the selected column, such as, if the column length has changed.

The following column information is monitored by the schema change process:

- Name
- Length
- Precision
- Type
- Scale
- Is Nullable

## About the Column entity Association area

You can view general details on the entities that are associated with the selected Column entity in the Association area. It is possible to associate to the Column entity from the Table, Index, and View entities.

SQL Server 2005 allows you to extend the functionality of non-clustered indexes by adding non-key columns to the leaf level of the non-clustered index. This allows you to create non-clustered indexes that cover more queries, thereby considerably improving query performance. The Non-Key Column field (displayed for SQL Server 2005 non-clustered indexes only), indicates whether or not the selected column is a non-key column.

## About the View and View Contain Same Column entities

The View entity displays general details and schema changes for the selected view.

The View Contain Same Column entity displays views that have the same column as the column displayed in the Main area. The information displayed is the same as displayed in the View entity and can only be associated to from the Column entity.

The following views are available:

- Overview
- Change Log Summary
- Change Log Detailed

## About getting an overview of the View and View Contain Same Column entities

The Overview displays general details on the view such as creation date, number of columns and displays the view text.

## About viewing the Change Log Summary of the selected view and its indexes

The Change Log Summary displays schema changes made in the selected view and its indexes, as follows:

- **Change Log Summary Table.** The following information is displayed:
  - **Sample Date.** The estimated last change date of the view (or/and its sub-objects). This value reflects the date that the change was first identified by the Precise for SQL Server Schema Changes Process, not the date that the object was changed.
  - **Object Type.** Indicates whether the object type is a View or Index. Object—displays the name of the View Index.
  - **Change Type.** Indicates the type of change detected. The following change types are reported:
    - **Created.** Indicates that the index was created
    - **Dropped.** Indicates that the index was dropped
    - **Change in Structure.**

- **View.** Indicates any change that was made to the view's column(s).
- **Index.** Indicates any change that was made to the index keys of the index or to the index depth.
- **Change in Properties.**
  - **View.** Indicates any change that was made to one of the following view's properties: Name and Owner.
  - **Index.** Indicates any change that was made to one of the following index's properties: File Group, Fill Factor, Is Auto Statistics, Is Clustered Index, Is Pad Index, Is Page Lock Disallowed and Is Row Lock Disallowed.
- **Schema Changes.** Displays an overtime graph that indicates whether changes were made to the selected view and its indexes or columns, broken down into the following groups:
  - **View Change in Property.** Indicates any change made to the view properties. For example, whether the view was renamed.
  - **View Change in Structure.** Indicates any change made to the view structure. For example, whether a new column was added to the view.
  - **Index Change in Property.** Indicates any change made to one of the view's index properties, such as, the index was changed from clustered to non-clustered.
  - **Index Change in Structure.** Indicates any change made to the index structure, for example, whether a new key was added to the index.
  - **Index Created/Dropped.** Indicates any addition or deletion of indexes.

## About viewing Change Log Details

The Change Log Detailed view displays the schema changes made on the selected view and/or its sub-objects.

## About the View and View Contain Same Column entities Association area

You can display general details on the entities that can be associated with the selected View entity in the Association area. It is possible to associate to the View entity from the Database, Table, Column and Index entities.

## About the User entity

The User entity displays general details of the selected user, such as, Roles and Database Access.

## About the User entity Association area

You can view general details on the entities that can be associated to the selected User entity in the Association area. It is possible to associate to the User entity from the Database entity.

## About the Logical File entity

The Logical File entity displays general details, I/O activity and storage information on the selected database file.

The following views are available:

- Overview
- Statistics

## About getting an overview of the Logical File entity

The Overview displays general details on the Logical File, as follows:

- **Properties.** Displays general details on the Logical File such as full filename (including path), file group, and file growth.
- **I/O Wait.** Displays the I/O wait for the selected Logical File, over the selected time period, as collected by the Collect DB File Statistics process.
- **I/O Read Wait.** Displays the I/O read wait for the selected Logical File, over the selected time period, as collected by the Collect DB File Statistics process.
- **I/O Write Wait.** Displays the I/O write wait for the selected Logical File, over the selected time period, as collected by the Collect DB File Statistics process.
- **Bytes Requested.** Displays the I/O activity (bytes read and written) over the selected time period, for the selected Logical File, as collected by the Collect DB File Statistics process.
- **Requests.** Displays the I/O activity (number of read/writes) over the selected time period, for the selected Logical File, as collected by the Collect DB File Statistics process.
- **Space Allocation.** Displays free space and used space (in MB) over the selected time period, for the selected logical file, as collected by the Collect Space Utilization process.

## About viewing I/O statistics on a Logical File

The Statistics view displays I/O statistics on the Logical File over the selected time period, as follows:

- **I/O Wait.** Displays the I/O wait for the selected Logical File, over the selected time period, as collected by the Collect DB File Statistics process.
- **Bytes Requested.** Displays the I/O activity (bytes read and written) over the selected time period, for the selected Logical File, as collected by the Collect DB File Statistics process.
- **Requests.** Displays the I/O activity (number of read/writes) over the selected time period, for the selected Logical File, as collected by the Collect DB File Statistics process.

## About the Logical File entity Association area

You can view general details on the entities associated to the Logical File entity in the Overview tab in the Association area. The Statistics tab shows information on I/O wait, bytes requested, number of requests, and size of logical files. It is possible to associate to the Logical File entity from the following entities: Instance, Database, File Group, Physical Disk, Logical Volume, HP Storage Device, HDS Storage Device, Symmetrix Storage Device and Clariion Storage Device.

## About the File Group entity

The File Group entity displays general details and space utilization, over the selected time period, for the selected file group.

### About the Overview

The Overview displays general File Group details such as whether the file group is read-only and information regarding the file group's space utilization, over the selected time period, as follows:

- **Properties.** Displays general File Group details such as whether the file group is read only, or whether this is the default file group.
- **Storage.** Displays the amount of free space and used space of the selected file group, over the selected time period, based on the data collected by the Collect Space Utilization process.

### About the File Group entity Association area

You can view general details on the entities that are associated to the selected File Group in the Association area. It is possible to associate to the File Group from the following entities: Database, Table, Index, Logical File, Physical Disk, Logical Volume, HP Storage Device and HDS Storage Device.

## About the Statistics entity

The statistic entity displays distribution statistics on the selected collection (set of columns). The following views are available:

- Overview
- Statistics

### About getting an overview of the selected statistic

The Overview displays general details of the selected Statistic such as keys and an indication when the statistics were last calculated.

### About viewing distribution statistics

The Statistics view displays distribution statistics for the selected statistic. See [About viewing distribution statistics for a selected index](#).

### About the Statistics entity Association area

You can view general details on the selected Statistics entity in the Association area. It is possible to associate to the Statistics entity from the Table and Column entities.

## About the Stored Procedure entity

The stored procedure entity displays general details and resource consumption of the selected stored procedure. The following views are available:

- Overview
- Text

### About getting an overview of a stored procedure

The Overview displays general properties of the stored procedure, provides resource consumption information, in addition to the number of times the selected stored procedure's statements were executed, based on the data collected by the Precise for SQL Server Collector for the selected time period.

The following information is displayed:

- **Properties.** Displays general details of the stored procedure such as whether the stored procedure is an extended procedure, and whether the stored procedure was created during installation.
- **In MS-SQL.** Displays the resource consumption of the selected stored procedure, based on the data collected by the Precise for SQL Server Collector for the selected time period.
- **Attributes.** Displays selected attributes that exist for each stored procedure, such as if the stored procedure has the flag `Anis Null On` set to *On* or *Off*.
- **Statement Execution.** Displays the number of times the selected stored procedure's statements were executed, based on the data collected by the Precise for SQL Server Collector for the selected time period.

### About viewing the text of a stored procedure

The Text view displays the stored procedure's text.

## About the Stored Procedure entity Association area

You can view general details on the entities associated with the selected Stored Procedure entity in the Association area. It is possible to associate to the Stored Procedure entity from the Database, Table, View, User, Stored Procedure and Function entities.

## About the Function entity

The function entity displays general details and resource consumption of the selected function. The following views are available:

- Overview
- Text

## About getting an overview of the Function entity

The Overview displays general details of Function entity, as follows:

- **Properties.** Displays general properties of the function, such as type of function (inline, scalar, etc.), and whether or not the function was created during installation.
- **In MS-SQL.** Displays the resource consumption of the selected function, based on the data collected by the Precise for SQL Server Collector for the selected time period.
- **Attributes.** Displays selected attributes that exist for each stored procedure, such as if the stored procedure has the flag `Anis Null On set to Or` or `Off`.
- **Statement Executions.** Displays the number of executions of the selected function's statements, based on the data collected by the Precise for SQL Server Collector for the selected time period.

## About viewing the text of a function

The Text view displays the text of the function.

## About the Function entity Association area

You can view general details on the entities associated with the selected Function entity in the Association area. It is possible to associate to the Function entity from the Database, Table, View, User, Stored Procedure and Function entities.

## About the Trigger entity

The trigger entity displays general details and resource consumption of the selected trigger. The following views are available:

- Overview
- Text

## About getting an overview of general trigger properties

The following information is displayed in the Overview:

- **Trigger.** Displays general trigger properties such as trigger name, and the name of the table on which the trigger is defined, and whether this is the first trigger fired when data is deleted from a table.
- **In MS-SQL.** Displays resource consumption of the selected trigger, over a selected time period, based on the data on trigger statements collected by the Precise for SQL Server Collector.
- **Attributes.** Displays selected attributes that exist for each stored procedure, such as if the stored procedure has the flag `Anis Null On set to Or` or `Off`.
- **Statement Executions.** Displays the number of times the selected trigger's statements were executed, based on the data collected by the Precise for SQL Server Collector for the selected time period.

## About viewing the text of a trigger

The Text view displays the text of the trigger.

## About the Trigger entity Association area

You can view general details on the entities associated with the selected Trigger entity in the Association area. It is possible to associate to the Trigger entity from the Database, Table, View and User entities.

## About the Statement entity

The Statement entity displays access plan details and resource consumption for the selected statement. The following views are available:

- Overview
- Performance
- SQL Text

## About getting an overview of a statement

The Overview displays general details about the statement, along with details regarding its last access plan, as follows:

- **Statement.** Displays general statement details such as statement ID, database and parsing user.
- **Access Plan.** Provides details regarding to the last access plan of the statement, such as whether a table scan, index seek and index scan was performed.

## About viewing the performance of a selected statement

The Performance view displays resource consumption information on the selected statement, as follows:

- **In MS-SQL.** Displays resource consumption information for the selected statement, over the selected time period.
- **Statement Executions.** Displays the number of statement executions carried out over the selected time period.

## About viewing the SQL text of a selected SQL statement

The SQL Text view displays the text of the selected SQL statement.

## About the Statement entity Association area

You can view general details on the entities associated with the selected Statement entity in the Association area. It is possible to associate to the Statement entity from the Table and Index entities.

The following tabs are available provide additional information in the Association area:

- **Performance.** Provides general details and resource consumption of the statement.
- **Plan.** Displays details on the last access plan of the statement such as whether a Table Scan was used, and whether an Index Seek was.

## About the Batch entity

The Batch entity displays access plan details and resource consumption for the selected batch. The following views are available:

- Overview
- Performance
- SQL Text

## About getting a general overview of batch details

The Overview displays general batch details such as last show plan and most recent plan, in addition to details regarding the selected batch's last access plan, as follows:

- **Batch.** Displays general details of the batch such as database and parsing user.
- **Access Plan.** Displays details on the last access plan of the batch such as, whether a table scan or index scan was performed.

## About viewing the performance of a batch

The Performance view displays resource consumption and number of statement executions for a statement, as follows:

- **In MS-SQL.** Displays resource consumption information for the selected statement, over the selected time period.
- **Statement Executions.** Displays the number of statement executions carried out over the selected time period.

## About viewing the SQL text of a batch

The SQL Text view displays the batch's text.

## About the Batch entity Association area

You can view general details on the entities associated with the selected Batch entity in the Association area. It is possible to associate to the Batch entity from the Table and Index entities.

The following tabs are available in the Association area:

- **Performance.** Displays general details and resource consumption for the selected batch.
- **Plan.** Displays details on the last access plan of the selected batch such as whether a table scan or index seek was used.

## About the Physical Disk entity

The Physical Disk entity displays general details and I/O wait data for the selected physical disk.



## About getting an overview of general physical disk details

The Overview displays general physical disk details along with I/O wait data for the selected physical disk, as follows:

- **Properties.** Displays general physical disk details such as type, and number of logical volumes.
- **I/O Wait.** Displays the I/O wait on the selected physical disk as collected by Precise for SQL Server Collector over the specified time period.

## About the Physical Disk entity Association area

You can view general details on the entities associated with the selected Physical Disk entity in the Association area. It is possible to associate to the Physical Disk entity from the following entities: Database, Table, Index, Logical File, File Group, Logical Volume, Disk Partition, HP Storage Device and HDS Storage Device.

## About the Logical Volume entity

The Logical Volume entity displays general details and storage utilization for the selected logical volume.

## About getting an overview of logical volume details

The Overview displays general logical volume details, in addition to the free space and used space of the selected logical volume, as collected by the Collect Space Utilization process over the specified time period, as follows:

- **Properties.** Displays general details of the logical volume such as type, and number of Logical files stored in the logical volume.
- **Storage.** Displays the free space and used space of the selected logical volume as collected by the Collect Space Utilization process over the specified time period.

## About the Logical Volume entity Association area

You can view general details on the entities associated to the selected Logical Volume entity in the Association area. It is possible to associate to the Logical Volume entity from the following entities: database, table, index, logical file, file group, physical disk, disk partition, HP storage device and HDS storage device.

## About the Disk Partition entity

The Disk Partition entity displays general details for the selected disk partition, such as disk partition number in physical disk and disk partition number in logical volume, disk partition offset and length.

## About the Disk Partition entity Association area

You can view general details on the entities associated to the selected Disk Partition entity in the Association area. It is possible to associate to the Disk Partition entity from the Physical Disk, Logical Volume, HP Storage Device and HDS Storage Device entities.

## About the Storage Unit entity

The Storage Unit entity displays a list of all storage entities connected to instances in the Tier. The Storage Unit entity can only be associated to from the Instance level.

For more information about the Storage Device entity, see [About Storage Unit entities](#).

## About the Storage Unit entity Association area

You can view general details on the entities associated to the selected Storage Units in the Association area. This information is based on the mapping carried out by the Precise for SQL Server Storage agents that are installed.

## How the Objects tab can help you identify performance problems

The following section describes a few examples on how you can use the Objects tab to identify performance problems.

- [Analyzing how configuration changes affect resource consumption](#)
- [Analyzing how schema changes affect the resource consumption of various entities](#)
- [Examining an object's space usage over time](#)
- [Examining fragmentation of table data and indexes](#)
- [Examining access methods to tables and indexes](#)
- [Examining the distribution statistics of indexes](#)
- [Determining whether to recommend an index for a table](#)

## Analyzing how configuration changes affect resource consumption

Changing an instance configuration parameter (or database parameter) may affect the performance of your system. Precise for SQL Server allows you to track the configuration changes and compare them with the resource consumption of your application, over time. This can be done by examining the Instance/Database Changes view in the Instance entity. The Instance/Database Changes view allows you to compare the changes made to the instance configuration parameters and the database options, with the resource consumption of the entire instance, over a selected time period. In this way you can pinpoint the changes that affect the resource consumption of the entire database.

To analyze how configuration changes affect resource consumption

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the View controls in the Main area, click **Instance/Database Changes**. Compare the changes made to the instance configuration parameters and the database options, with the resource consumption of the entire instance, over the selected time period. This allows you to pinpoint the changes that affect the resource consumption of the entire database.
4. On the Association controls in the Association area, click **Databases**.
5. Compare the changes made to the instance configuration parameters and the database options, with the resource consumption of the entire instance, over the selected time period. This allows you to pinpoint the changes that affect the resource consumption of the entire database.
6. On the View controls in the Main area, click **Database Created/Dropped**. Analyze how changes to database options affect In MS-SQL resource consumption.

## Analyzing how schema changes affect the resource consumption of various entities

Changing the database schema may affect the performance of your system. These changes are done to improve the performance of your application. You need to monitor them to ensure that you have achieved your goals. Precise for

SQL Server allows you track the schema changes and compare them with the resource consumption of the entire database or a table. This allows you to detect changes that affect the resource consumption of the database. The Object Changes view of the Database entity allows you to check the database objects that where changed, dropped or created in reference to the resource consumption of the entire database over time. A summary of the schema changes is displayed. Drilling down to the specific object provides detailed information on the schema changes.

See [About the Database entity](#).

To analyze how schema changes affect the resource consumption of various entities

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. On the View controls in the Main area, click **Object Changes**. Check the database objects that where changed, dropped or created in reference to the resource consumption of the entire database over time and determine if they affect In MS-SQL resource consumption.
5. Drill down to a specific table entity. On the View controls in the Main area, click **Change Log Detailed** and view detailed information on the changes made to the table's schema.

## Examining an object's space usage over time

Space usage information, collected by the Collect Space Utilization process, allows you to examine the space allocated versus the space used, in a table and index. This enables you to easily keep track of object growth over time and decide how to spread database files over various disks and how to connect objects to file groups in the specified database.

To examine an object's space usage over time

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. Drill down to a specific table entity. On the View controls in the Main area, click **Overview**.
5. In the Main area, analyze the Space over Time graph to determine how tablespace has been utilized over time.



The values might be incorrect due to out-of-date space usage information. To update the values, use the `DBCC UPDATEUSAGE` command to recalculate the space usage fields. For more information, see *SQL Server Books Online for DBCC UPDATE USAGE*.

## Examining fragmentation of table data and indexes

Fragmentation of table data and indexes may affect the performance of your system. The fragmentation occurs as a result of data modification (INSERT, UPDATE, and DELETE statements) performed on the table. Fragmentation increases the amount of page reads performed by the query, increasing their resource consumption.

To examine fragmentation of table data and indexes

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. On the Association controls in the Association area, click **Tables** or **Indexes**. Drill down to a specific table or index entity (depending upon your selection).

5. On the View controls in the Main area, click **Show Contig** and analyze the fragmentation information displayed for the selected table or index. Determine whether the table or index is heavily fragmented and needs to be reorganized.

See [About viewing data storage and fragmentation information in the Show Contig view](#) and [About viewing data storage and fragmentation information of a selected index in the Show Contig view](#).

## Examining access methods to tables and indexes

Examining the access methods used on tables and indexes can help you make decisions regarding the types of indexes you should employ. For example if most of the statements accessing the specified table are of type update (INSERT, UPDATE, DELETE) and only a few are of type SELECT you should add as few indexes as possible to efficiently access data, on the one hand, and avoid overhead when performing updates, on the other hand. Also if you observe an increase in I/O wait you should consider rewriting the statements using the specified table or index, making sure you implement appropriate indexes and reorganize any table data and indexes that may suffer from fragmentation.

To examine access methods to tables and indexes

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. On the Association controls in the Association area, click **Tables** or **Indexes**. Drill down to a specific table or index entity (depending upon your selection).
5. On the View controls in the Main area, click **Usage Report** to observe information on the access methods used on the table or index.

See [About viewing usage information and access methods in the Usage Report](#) and [About viewing a usage report](#).

## Examining the distribution statistics of indexes

Distribution statistics of an index can help you determine whether or not the index is useful to the query optimizer. The information displayed in the Statistics view of the Index entity indicates the selectivity of an index; the more selective an index is, the more useful it is, since this means more rows can be eliminated from consideration.

See [About viewing distribution statistics for a selected index](#).

To examine the distribution statistics of indexes

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. On the Association controls in the Association area, click **Indexes**. Drill down to a specific index.
5. On the View controls in the Main area, click **Statistics** and observe the distribution statistics for the selected index in the Distribution table.

## Determining whether to recommend an index for a table

The decision to add a specific index to a table is complex. The index can improve one statement and may negatively affect another. Therefore, before adding an index, you should examine every statement accessing the table to see how the new index will impact it.

Precise for SQL Server uses the Microsoft® Index Tuning Wizard to obtain recommended indexes and statistics for a selected table. The recommendations are based on the statements that were executed during the selected time period that have an average duration time greater than the predefined threshold, that refer to the PMDB table. "0" seconds is the default.

By associating to statements from the selected table you can view all the statements using the specified table and their resource consumption. This information can help you estimate which statements will be affected by the new index and how they will be affected.

To determine whether to recommend an index for a table

1. In the Time Frame list, choose the period of time you want to analyze.
2. In the Instance list, choose the item you want to analyze.
3. On the Association controls in the Association area, click **Databases**. Drill down to the database you want to analyze.
4. On the Association controls in the Association area, click **Tables**. Drill down to the table you want to analyze.
5. On the Recommend tab, analyze the statements that are associated with the selected table and observe their resource consumption.

The Recommend Indexes output shows a list of recommended indexes and statistics. For each recommended item, detailed information is displayed in the Details area, such as a list of key columns and the DDL of the Create Index or Statistics text.