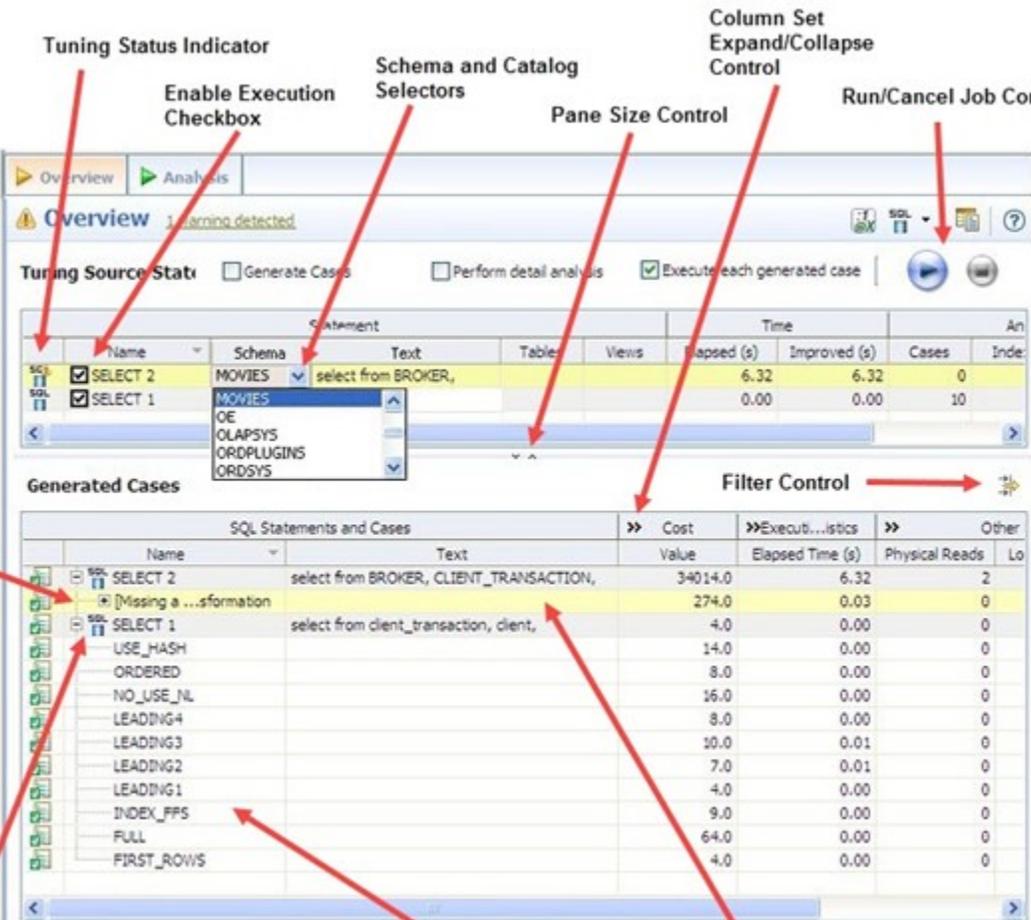


# Running a Tuning Job

After you add SQL statements to the job, click the Overview tab. Once you choose your tuning options and click the **Run Job** icon, the DML is parsed from the statements and added to the **Generated Cases** area. The Generated Cases are alternative execution paths or explain paths that could be better or worse than the default path the database uses. When these cases are executed, you can use the execution statistics to determine which case would optimize performance.

Each extracted statement is listed by **Name** and **Text**. Additionally, each statement has a **Cost**, **Elapsed Time**, and **Other Execution Statistics** value that provide information on how effectively each case executes on the specified data source. These parameters let you compare the efficiency of the original statements to the cases generated by the tuning process when it is executed.

 You can double-click a generated case to view or edit the SQL source of the statement.



**Tuning Status Indicator**

**Enable Execution Checkbox**

**Schema and Catalog Selectors**

**Pane Size Control**

**Column Set Expand/Collapse Control**

**Run/Cancel Job Controls**

**Generated Cases**

Name	Text	Cost	Elapsed Time (s)	Physical Reads	Logical Reads
SELECT 2	select from BROKER, CLIENT_TRANSACTION,	34014.0	6.32	2	0
[Missing a ...sformation		274.0	0.03	0	0
SELECT 1	select from client_transaction, client,	4.0	0.00	0	0
USE_HASH		14.0	0.00	0	0
ORDERED		8.0	0.00	0	0
NO_USE_NL		16.0	0.00	0	0
LEADING4		8.0	0.00	0	0
LEADING3		10.0	0.01	0	0
LEADING2		7.0	0.01	0	0
LEADING1		4.0	0.00	0	0
INDEX_FFS		9.0	0.00	0	0
FULL		64.0	0.00	0	0
FIRST_ROWS		4.0	0.00	0	0

**Transformation Case**

**Generated Case Expand/Collapse Control**

**Hint-Based Cases**

**Extracted SQL Statements**

The **Tuning Status Indicator** provides the status of each statement or case, and indicates if they are ready for execution. In some cases, SQL code may need to be corrected or bind variables may need to be set prior to executing statements. When you try to tune a statement containing a bind variable, you are now warned that either the type is not set or the value is not set.

Use the check boxes to select which statements and cases you want to run, and then click the **Run** icon in the upper right-hand corner of the screen. The **Execute each generated case** field enables you to execute each selected statement or case.

Use the **Schema and Catalog Selectors** to select a schema and catalog for the tuning job. The catalog selector is available only for SQL Server and Sybase data sources. By specifying the schema and catalog, the tuner can use the paths of the schema and catalog selected to find the tables queried in the job rather than use the paths of the schema and catalog used to connect to the data source. If you change the schema or catalog used in a tuning statement, you will need to refresh the tuning statements in order for new cases to be generated, which take into consideration the schema used. Right-click a tuning statement, and then select **Refresh Tuning Statements**.

Once you execute a tuning job, the **Generated Cases** tab reflects SQL Tuner's analysis of the specified statements. Once analyzed, you can proceed to modifying the Tuner results and applying specified cases on the data source to optimize its performance.

# Run a Tuning Job

## To run a Tuning Job

1. Once you have a SQL statement that is a tuning candidate, navigate to the Overview tab.
2. In the **Tuning Statements** area, select the checkbox next to the Statement name that you want to analyze and then:
  - *To analyze the SQL statements*, click **Generate cases**.
  - *To perform the analysis that populates the Analysis tab now*, click **Perform detail analysis**. Otherwise, this analysis is performed when you click the Analysis tab.
  - *To have the system generate execution statistics*, click **Execute each generated case**, and then select the number of times the system should execute each generated case. Multiple executions can verify that the case results are not skewed by caching. For example, the first time a query is run, data might be read off of disk, which is slow, and the second time the data might be in cache and run faster. Thus, one case might seem faster than another but it could be just benefiting from the effects of caching. Generally, you only need to execute the cases once, but it may be beneficial to execute the cases multiple times to see if the response times and statistics stay the same.
3. Click the **Run Job** icon at the top right-hand side of the window. The tuning job runs, analyzing each statement and case, and providing values in the appropriate columns.