

IBM Database Services Expansion Pack
SQL Tuning Services 2.1

Installation and Getting Started Guide
Last updated: 2024-04-02



Notes

Before using this information and the product it supports, be sure to read the general information under "Notices" at the end of this information.

Subsequent editions of this PDF will not be delivered in IBM Publications Center. Always download the latest edition from [IBM Documentation](#).

2024-04-02 edition

This edition applies to IBM® SQL Tuning Services, Version 2.1, which is a component of the Database Services Expansion Pack feature delivered with Db2 Accessories Suite for z/OS, V4.2.1, product number 5697-Q05, and to any subsequent releases to IBM SQL Tuning Services until otherwise indicated in new editions. Make sure that you are using the correct edition for the level of the IBM SQL Tuning Services offering that you are using.

© **Copyright International Business Machines Corporation 2022, 2024.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

- About this information..... V**
 - Accessibility features for SQL Tuning Services.....v
- Chapter 1. Overview of IBM SQL Tuning Services..... 1**
- Chapter 2. What's new in SQL Tuning Services.....7**
- Chapter 3. Installation roadmap..... 9**
- Chapter 4. Installation prerequisites for SQL Tuning Services..... 11**
- Chapter 5. Setting up required user IDs and permissions..... 13**
- Chapter 6. Configuring network ports.....17**
- Chapter 7. Configuring the SQL Tuning Services z/OS environment for the setup ID.....19**
- Chapter 8. Installing and configuring SQL Tuning Services..... 21**
- Chapter 9. Enabling extended tuning features..... 25**
- Chapter 10. Configuring AT-TLS for SQL Tuning Services..... 27**
- Chapter 11. Getting started.....29**
- Chapter 12. Reconfiguring SQL Tuning Services..... 37**
- Chapter 13. Upgrading SQL Tuning Services..... 41**
- Chapter 14. Invoking SQL Tuning Services APIs.....43**
- Chapter 15. Changing the repository database user ID and password.....45**
- Chapter 16. API reference for SQL Tuning Services..... 47**
- Chapter 17. Troubleshooting..... 55**
- Notices.....57**
 - Trademarks..... 58
 - Terms and conditions for product documentation..... 58
 - Privacy policy considerations..... 59
- Index..... 61**

About this information

This guide provides installation, configuration, administration, usage, and reference information for SQL Tuning Services.

Accessibility features for SQL Tuning Services

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in z/OS® products, including SQL Tuning Services. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers and screen magnifiers.
- Customization of display attributes such as color, contrast, and font size

Keyboard navigation

For information about navigating ISPF panels using TSO/E or ISPF, refer to the *z/OS TSO/E Primer*, the *z/OS TSO/E User's Guide*, and the *z/OS ISPF User's Guide*. These guides describe how to navigate each interface, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

IBM and accessibility

See the *IBM Accessibility Center* at <http://www.ibm.com/able> for more information about the commitment that IBM has to accessibility.

Chapter 1. Overview of IBM SQL Tuning Services

SQL Tuning Services is a set of tools, delivered as RESTful APIs, that you use to analyze and tune SQL applications that work with Db2® for z/OS. You can call these APIs directly from an application and you can use them from within IBM Db2 Administration Foundation for z/OS and IBM Db2 for z/OS Developer Extension.

SQL Tuning Services features

SQL Tuning Services features are provided in the following two products:

- **IBM Database Services Expansion Pack**, which is available at no additional cost to licensed Db2 Accessories Suite for z/OS users, includes a subset of all the available SQL Tuning Services features.
- **IBM Db2 Query Workload Tuner for z/OS 6.1**, which can be purchased separately or is included in the Db2 Performance Solution Pack for z/OS, includes all of the features that are included with IBM Database Services Expansion Pack plus a more robust set SQL analysis and tuning features.

The following table describes all of the SQL Tuning Services features and indicates which features are included with the related products:

Feature	IBM Database Services Expansion Pack (no charge)	Db2 Query Workload Tuner 6.1 (priced feature)
Access Path Advisor Generates access path recommendations that can improve the performance of a single query.		✓
Access Path Comparison Compares two access paths and displays the results of the comparison. A graphical representation of the access path flow is provided for each access path along with any meaningful differences in the plan table for each access path.		✓
Capture Query Environment Captures details about the environment in which you are running an SQL statement. These details are saved to a file that you can provide to IBM Support when you are trying to resolve a performance problem with an SQL statement or that you can use to re-create an environment on another subsystem.	✓	✓

Table 1. SQL Tuning Services features (continued)

Feature	IBM Database Services Expansion Pack (no charge)	Db2 Query Workload Tuner 6.1 (priced feature)
<p>Index Advisor Generates recommendations for creating, altering, or dropping indexes that can improve the performance of a single query. This feature provides the DDL scripts that you can run and information about the existing indexes from the query that you're tuning.</p>		✓
<p>Index Impact Analyzer Generates a report that shows the impact of applying the index changes that are recommended by Index Advisor and Workload Index Advisor.</p>		✓
<p>Job Management Enables you to query information about SQL Tuning Services jobs and cancel, delete, and view the results of SQL Tuning Services jobs.</p>	✓	✓
<p>Query Rewrite Advisor Evaluates how efficiently a query is written and generates best-practice recommendations for rewriting it to improve its performance.</p>		✓
<p>SQL Annotator Gathers statistical information about database objects that are used for access path selection and provides a cost estimate for running a particular query. This information can help you understand how a query is processed by the Db2 optimizer so that you can determine which other SQL tuning tools might be able to improve that query's performance.</p>		✓
<p>SQL Capture Retrieves statements from the following sources:</p> <ul style="list-style-type: none"> • A user-defined local repository • A local file • For dynamic SQL, from the statement cache • For stabilized dynamic SQL and static SQL, from Db2 catalog tables • A sequential data set or a member of a partitioned data set (PDS or PDSE) 	✓	✓

Table 1. SQL Tuning Services features (continued)

Feature	IBM Database Services Expansion Pack (no charge)	Db2 Query Workload Tuner 6.1 (priced feature)
SQL Formatter Formats an SQL statement for a single query so that you can more easily read and understand the structure of the statement.	✓	✓
Statistics Advisor Generates a recommended set of <u>RUNSTATS</u> commands to improve query performance.	✓	✓
Visual Explain Generates a graphical representation and a grid <u>plan table</u> view of the access paths for an SQL statement. This information is useful for tuning queries for better performance.	✓	✓
Virtual Index Analyzer Virtually tests indexes to determine if the performance of a single query can be improved by creating or dropping the indexes.		✓
Workload Access Path Advisor Generates recommendations for improving the access paths that are used by an SQL workload.		✓
Workload Access Path Comparison Compares, annotates, and generates reports on the access plans and estimated cost changes for SQL statements in different packages or workloads.		✓
Workload Analytics Acceleration Advisor Generates a report that shows the eligible and ineligible statements for offloading to IBM Db2 Analytics Accelerator and quantifies the CPU savings that can be made by offloading.		✓
Workload Candidate Acceleration Analyzer Generates a report that shows the eligible and ineligible statements for offloading to IBM Db2 Analytics Accelerator and quantifies the CPU savings that can be made by offloading with user input candidate tables.		✓
Workload Explain Gathers explain information for all statements in a workload and stores this information in the repository database. This information is used by other workload advisors to analyze these statements.		✓

<i>Table 1. SQL Tuning Services features (continued)</i>		
Feature	IBM Database Services Expansion Pack (no charge)	Db2 Query Workload Tuner 6.1 (priced feature)
Workload Index Advisor Generates recommendations for creating, altering, or dropping indexes that can improve the performance an SQL workload. This feature provides the DDL scripts that you can run and information about the existing indexes from the workload that you're tuning.		√
Workload Management Provides a set of APIs for common workload-related tasks such as creating, updating, And deleting workloads, listing the statements in a workload, refining workload results by applying filter criteria, and so on.		√
Workload Query Environment Collector Captures details about the environment in which you are running an SQL workload. These details are saved to a file that you can provide to IBM Support when you are trying to resolve a performance problem with an SQL workload or that you can use to re-create an environment on another subsystem.		√
Workload Query Rewrite Advisor Generates recommendations for rewriting statements in an SQL workload to improve its overall performance.		√
Workload Statistics Advisor Generates recommendations and RUNSTATS DDL for creating or modifying statistical views.		√
Workload Virtual Index Analyzer Virtually tests indexes to determine if the performance of a workload can be improved by creating or dropping the indexes.		√

How to use SQL Tuning Services

Because SQL Tuning Services functionality is delivered as RESTful APIs, you can call them directly from an application or development environment (for example, from within a CI/CD DevOps pipeline).

Additionally, SQL Tuning Services functionality is integrated into the user interfaces of the following products:

- IBM Db2 Administration Foundation for z/OS leverages the SQL Tuning Services APIs to improve the performance of SQL applications that are running on your system. A database administrator can identify, evaluate, and improve SQL queries that affect system performance.

- [IBM Db2 for z/OS Developer Extension](#) integrates the SQL Tuning Services APIs into a Microsoft Visual Studio Code development environment so that application developers can analyze and tune their SQL applications as they are writing them.

Restriction: Workload analysis and tuning features are available only as stand-alone APIs and with Db2 Administration Foundation. Currently, they are not integrated into Db2 Developer Extension.

Chapter 2. What's new in SQL Tuning Services

This topic describes new and changed SQL Tuning Services functions and identifies the APARs that deliver these functions.

APAR	Date	Description
PH59959	March 2024	<ul style="list-style-type: none">• Support for using a subset of the SQL Tuning Services APIs without creating or specifying a tuning connection profile.• Support for using the following security mechanisms for tuning connection profiles:<ul style="list-style-type: none">– 7 - User ID and encrypted password– 9 - Encrypted user ID and encrypted password– 18 - Certificate authentication• Support for sorting workloads by their creation time in descending order.• Support for sorting jobs by their start time in descending order.• An enhancement to the installation script in which, if you are using key ring for Liberty SSL, you are no longer prompted for a key ring password.• An enhancement to the installation script that adds the <code>sslTrustStoreType</code> and <code>sslTrustStorePassword</code> parameters to the <code>tmsservice.config</code> configuration file.• Miscellaneous fixes. See the APAR for details.
PH58058	January 2024	<ul style="list-style-type: none">• A new <code>repoDBCredReset.sh</code> script, which enables an administrator to reset the repository database connection logon credentials.• An enhancement to the <code>GET/DELETE jobs</code> API that adds more filters.• An enhancement to the <code>GET connection profile</code> API that adds more properties.• The removal of the dependency on <code>db2jcc_license_cisuz.jar</code>.• Miscellaneous fixes. See the APAR for details.
PH56970	October 2023	<ul style="list-style-type: none">• Support for the following new SQL tuning features:<ul style="list-style-type: none">– Workload Environment Capture– Workload Analytics Acceleration Advisor– Workload Candidate Acceleration Analyzer• Support for TLS 1.3.• To support Db2 13 for z/OS, an enhancement to the SQL capture API that supports the new value of S in the VALID column of the SYSPACKAGE and SYSPACKCOPY catalog tables.• An enhancement to the SQL Capture and Index Impact Advisor APIs for customizing the versions of bound packages.• An enhancement to the Index Impact Analyzer API for adding a job description to specify the source job.• Miscellaneous fixes. See the APAR for details.

APAR	Date	Description
PH55158	July 2023	<ul style="list-style-type: none"> • Support for the following new SQL tuning features: <ul style="list-style-type: none"> – Single query and Workload Virtual Index Analysis API – Single query and Workload Impact Analysis API – Workload Refine API – Workload Explain failure reason API • An enhancement to the installation process that enables you to specify the repository database after the server is installed and started. • Sample JCL to upgrade the repository database. • A new <code>set_repo</code> API • An enhancement to the <code>List workloads</code> API to include more information in a summary report. • A change to the <code>CreateRepo</code> API to support the creation of only a new repository database. • Miscellaneous fixes. See the APAR for details.
PH53536	May 2023	<ul style="list-style-type: none"> • The new <code>Capture SQL from z/OS data set</code> API. • An enhancement that updates the value of <code>Explain_Status</code> for each statement in a workload. • Enhancements to simplify the configuration script. • An enhancement that reduces idle CPU consumption. • Enhancements to sample JCL to remove dependency on <code>SYSPROC.DSNAHVPM</code> and eliminate <code>SYSADM</code> authority. • Miscellaneous fixes. See the APAR for details.

Chapter 3. Installation roadmap

This topic provides a comprehensive list of the installation and configuration tasks that are presented in the prescribed sequence and identifies the various roles that are associated with each task.

Installing and configuring SQL Tuning Services requires z/OS system programmer and UNIX System Service (UNIX System Services) administrator skills. The skills of a database administrator, security administrator, network administrator, or UNIX shell programmer might also be needed to complete specific installation or configuration tasks.

Use the following list of tasks as a roadmap to install and configure SQL Tuning Services:

Step	Task and instructions	Role
1	Allocate an appropriate amount of system capacity (system, processor, memory, and disk space), and verify that your system meets the <u>minimum hardware and software requirements</u> for installing SQL Tuning Services.	z/OS system programmer
2	Assign the user IDs that are required to install, <u>configure, and administer</u> your SQL Tuning Services installation, and grant them the required privileges and permissions.	Security administrator
3	<u>Configure networks ports.</u>	Network administrator
4	<u>Configure the SQL Tuning Services environment and setup ID.</u>	z/OS system programmer
5	<u>Install and configure SQL Tuning Services.</u>	z/OS system programmer
6	To use the extended set of tuning features that are available with Db2 Query Workload Tuner for z/OS 6.1, <u>enable the Db2 Query Workload Tuner license.</u>	z/OS system programmer

Chapter 4. Installation prerequisites for SQL Tuning Services

Before you start to install SQL Tuning Services, make sure that your system meets the minimum requirements and that you have the SQL Tuning Services program materials.

System-level requirements

SQL Tuning Services uses both IBM proprietary and open source technologies and requires various hardware and software products in the z/OS environment. Make sure that your system meets all the following requirements before you install SQL Tuning Services.

System capacity

To maximize the performance of your SQL Tuning Services implementation, allocate the appropriate level of system capacity based on your actual workload.

The following table shows the minimum system capacity that is required to install and use SQL Tuning Services.

Important: Although you can use the minimum level of system capacity to tune a reasonably sized workload, heavier workloads and a high number of concurrent consumers require more system capacity. You might need to increase the minimum values based on your environment.

Hardware	Number of LPARs or servers	CPU per LPAR or server	Memory per LPAR or server	DASD/disk space per LPAR or server
Z system	1 LPAR	1 zIIP 1 GCP	4 GB	5 GB

Operating system

z/OS 2.3 with APAR OA56570 applied or any later version of z/OS.

WebSphere® Liberty server

IBM Liberty Embedded 21.0.0.9 (included with z/OS) is the minimum supported level; however, using the most current release is recommended.

For more information about the WebSphere Liberty server and its current supported releases, see the [WebSphere Support Lifecycle Policy](#) and the [list of fixes for WebSphere Liberty server](#).

Database

Db2 12 for z/OS or later

Java™ SDK

IBM 64-bit SDK for z/OS Java Technology Edition Version 8, Service Refresh 4 FixPack 6 (Java8SR4FP6) or later, with JAVA_HOME set to your installation directory.

Cryptographic software

z/OS Integrated Cryptographic Service Facility (ICSF). Make sure that the ICSF component is properly configured and started. See [ICSF Version and FMID Cross Reference](#) for details.

SQL Tuning Services program materials

Verify that the order that you received from IBM contains the following program materials for SQL Tuning Services, which is a component of IBM Database Services Expansion Pack:

- The SMP/E product image
- The [IBM Database Services Expansion Pack Program Directory](#)
- The license information for the IBM Database Services Expansion Pack CD image

- Available maintenance packages

Extended tuning features

Use of the extended set of SQL analysis and tuning features requires Db2 Query Workload Tuner for z/OS 6.1. For more information, see the [Db2 Query Workload Tuner for z/OS Program Directory](#).

Requirements for specific APIs

To use the Workload Analytics Acceleration Advisor API and the Workload Candidate Acceleration Analyzer API, Db2 has the following requirements:

Workload Analytics Acceleration Advisor

- The SYSPROC.ADMIN_INFO_SYSPARM stored procedure is required to collect necessary system information. For more information, see [ADMIN_INFO_SYSPARM stored procedure \(Db2 SQL\)](#).
- The following Db2 subsystem parameters must be set as shown:

```
DSN6SPRM.QUERY_ACCELERATION = NONE  
DSN6SPRM.ACCELMODEL = YES
```

Workload Candidate Acceleration Analyzer

- The SYSPROC.ADMIN_INFO_SYSPARM stored procedure is required to collect necessary system information. For more information, see [ADMIN_INFO_SYSPARM stored procedure \(Db2 SQL\)](#).
- The SYSPROC.ADMIN_COMMAND_DB2 stored procedure is required to run system commands. For more information, see [ADMIN_COMMAND_DB2 stored procedure \(Db2 SQL\)](#).
- One of the following authorities is required to start, stop, and display information about virtual accelerators:

```
SYSOPR  
SYSCTRL  
SYSADM
```

- The following Db2 subsystem parameters must be set as shown:

```
DSN6SPRM.QUERY_ACCELERATION = NONE  
DSN6SPRM.ACCEL = AUTO oI = COMMAND
```

- The following tables must exist on Db2:
 - SYSACCEL.SYSACCELERATORS
 - SYSACCEL.SYSACCELERATEDPACKAGES
 - SYSACCEL.SYSACCELERATEDTABLES
 - SYSACCEL.SYSACCELERATEDTABLESAUTH

For more information, see [Enabling Db2 for IBM Db2 Analytics Accelerator for z/OS \(Db2 Performance\)](#).

Chapter 5. Setting up required user IDs and permissions

Specific user IDs with sufficient permissions are required to install, configure, administer, and use SQL Tuning Services, Db2 for z/OS, and other related components. You must allocate or create these user IDs before you start to install SQL Tuning Services.

Installing and configuring SQL Tuning Services and its related products requires several different user IDs that have specific privileges and permissions. These IDs are listed in the following table. The names in the user ID column are used throughout the installation and configuration documentation, but you can assign any names that you want to these IDs.

User ID	Description	Required privilege or permission
<i>tms_setup_userid</i>	This ID is typically used by a system programmer to install, configure, and start SQL Tuning Services in UNIX System Services.	<ul style="list-style-type: none"> • \$JAVA_HOME/bin defined in the \$PATH environment variable in the user's profile • The \$_BPXK_AUTOCVT environment variable set to ON in the user's profile • Permission to read and execute to the <i>install_dir_zos</i> directory or a directory similar to \$TMS_HOME that's used by the SMP/E installation process. • The \$IBM_JAVA_OPTIONS environment variable set to the following value in the user's profile: <pre>-Dfile.encoding=UTF-8</pre>
<i>db2_authid_R</i>	This user ID is used to access the Db2 for z/OS subsystem where the SQL Tuning Services <i>repository</i> database resides.	<p>The CREATEDBA privilege and the CREATEIN privilege on SCHEMA IBMTMS are required on the Db2 for z/OS subsystem where the SQL Tuning Services repository database will reside:</p> <pre>CREATEDBA ON SYSTEM TO <i>db2_authid_R</i> CREATEIN ON SCHEMA IBMTMS TO <i>db2_authid_R</i></pre> <p>Additionally, to use the extended tuning features, the <i>db2_authid_R</i> ID requires EXECUTE privilege on all of the packages that are listed in the DSN5RTRP sample job. Use the DSN5RTRP sample job to grant these privileges.</p>
<i>tms_userid</i>	This user ID is used to log on to SQL Tuning Services.	<p>All SQL Tuning Services users need the following permissions:</p> <ul style="list-style-type: none"> • Permission to connect to the SQL Tuning Services repository database through JDBC • Permission to execute the CANVIEW authentication UDF (by default, IBMTMS.CANVIEW) as a valid SQL Tuning Services user <pre>EXECUTE ON FUNCTION IBMTMS.CANVIEW TO <i>tms_userid</i></pre> <p>Additionally, you need a <i>tms_userid</i> that has permission to execute the CANADMINISTER authentication UDF (by default, IBMTMS.CANADMINISTER).</p> <pre>EXECUTE ON FUNCTION IBMTMS.CANADMINISTER TO <i>tms_userid</i></pre>

User ID	Description	Required privilege or permission
		<p>This user ID is considered to be an SQL Tuning Services administrator and is responsible for the following tasks:</p> <ul style="list-style-type: none"> • Setting up the repository database • Managing all tuning jobs • Managing all connection profiles <p>A user who does not have permission to execute the CANADMINISTER UDF can see only their jobs.</p>
db2_authid_T	<p>This user ID is a Db2 for z/OS authorization ID that's used to connect to the <i>target</i> Db2 for z/OS subsystem to run various tuning APIs.</p> <p>To run certain tuning APIs, the <i>db2_authid_T</i> requires EXPLAIN privileges. For more information about these privileges, see the Authorization section of EXPLAIN (Db2 SQL).</p>	<p>Use the DSN5RTTG sample job to grant all of the following privileges.</p> <p>General privileges</p> <p>All <i>db2_authid_T</i> IDs require EXECUTE privilege on the packages that are listed in the DSN5RTTG sample job.</p> <ul style="list-style-type: none"> • DSN5OADM • DSN5OEPN • DSN5OFMM • DSN5ONPT • DSN5OPKN • DSN5OSCM <p>Privileges for SQL Capture, Statistics Advisor, and Query Environment Collector</p> <p>To use these three APIs, the <i>db2_authid_T</i> ID requires SQLADM authority.</p> <p>To use the Query Environment Collector, the <i>db2_authid_T</i> requires the privileges that are documented in the Authorization section of ADMIN_INFO_SQL stored procedure (Db2 SQL).</p> <p>Privileges to capture data from a user-defined SQL repository</p> <p>Applications that can gather runtime metrics about the performance of SQL statements within specified intervals of time can offload the runtime metrics and the SQL statements into Db2 for z/OS tables. These applications also gather and offload information about the database objects that were referenced by the SQL statements that ran during an interval.</p> <p>To access this data, ensure that the <i>db2_authid_T</i> ID has SELECT privileges on the tables that contain the text of SQL statements, runtime metrics, and information about objects that the SQL statements reference. If you use a view to join tables that contain the runtime metrics and the SQL text, ensure that the <i>db2_authid_T</i> ID has SELECT privilege on that view.</p> <p>Privileges to manage Explain tables</p> <p>To manage Explain tables, the <i>db2_authid_T</i> ID requires the privileges that are documented in the Authorization</p>

User ID	Description	Required privilege or permission
		<p>section of ADMIN_EXPLAIN_MAINT stored procedure (Db2 SQL).</p> <p>Privileges for extended tuning features</p> <p>To use the extended set of tuning features, the <i>db2_authid_T</i> ID requires SQLADM authority and EXECUTE privilege on the following packages:</p> <ul style="list-style-type: none"> • DSN50IA2 • DSN50IA7 • DSN50IA8 • DSN50IAL • DSN50QRA • DSN5OSCM • DSN5OWCN

Example

Carol is an SQL Tuning Services administrator who is who is responsible for creating the repository database, creating repository tables, granting privileges to SQL Tuning Services users, managing all tuning connections, and so on. To perform these administrative tasks, Carol needs a *tms_userid* ID with CANVIEW and CANADMINISTER privilege on the SQL Tuning Services repository database.

The following figure illustrates the function of Carol's *tms_userid* ID.

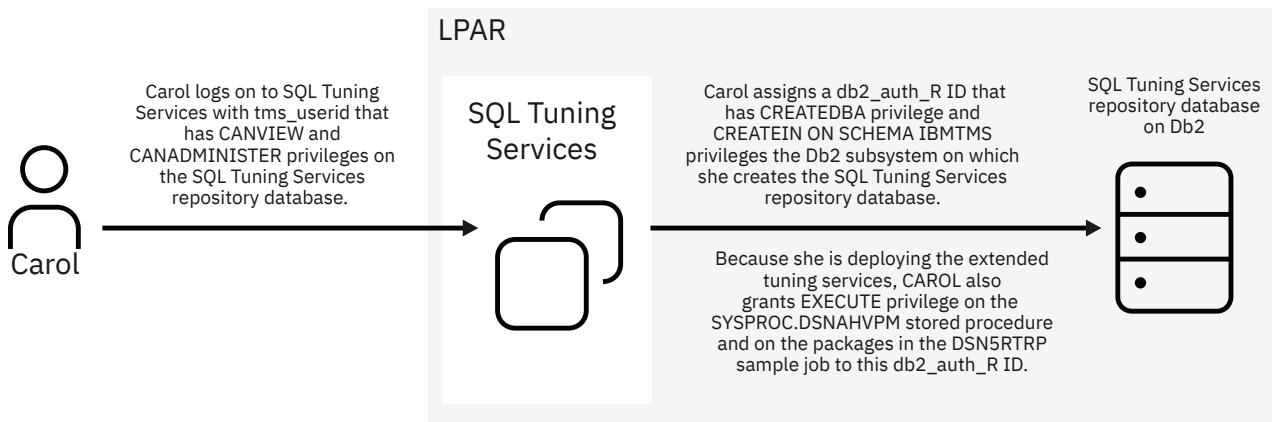


Figure 1. IDs for setting up and administering SQL Tuning Services

Zhou is an SQL Tuning Services user who needs to tune SQL on Db2 for z/OS. As an SQL Tuning Services user, Zhou needs the following IDs:

- A *tms_userid* ID to log on to SQL Tuning Services. Because he is a user with no administrative responsibilities, Zhou's *tms_userid* ID must have UDF CANVIEW privilege on the repository database.
- A *db2_authid_T* ID to connect to the target Db2 subsystem.
- Access to the functional *db2_authid_R* ID.

The following figure illustrates the function of the IDs that Zhou needs to use. In this figure, the repository database resides on its own subsystem, but it can also reside on the same subsystem as a target Db2 database.

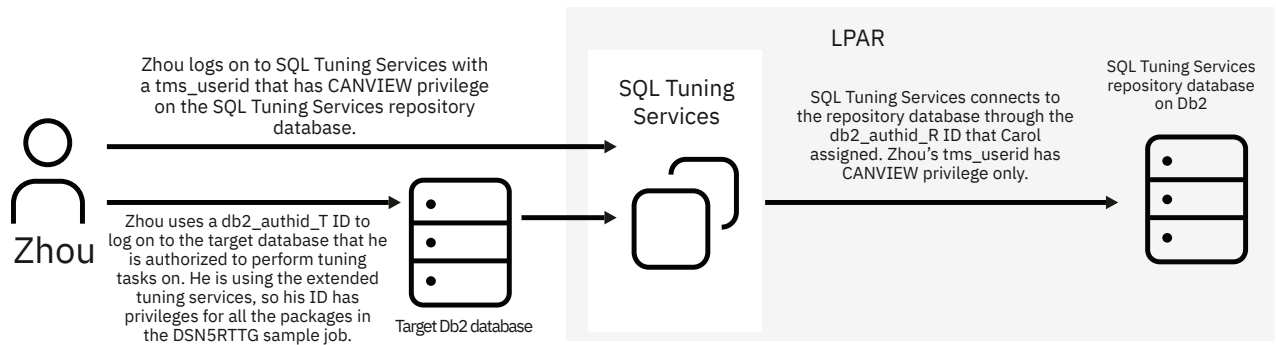


Figure 2. IDs for using SQL Tuning Services

Chapter 6. Configuring network ports

SQL Tuning Services requires dedicated networks and ports to communicate across component systems and services. SQL Tuning Services supports only HTTPS connections with clients and optionally AT-TLS connections with Db2 for z/OS.

Procedure

Assign the following ports before you install SQL Tuning Services:

System or service	Port number	Outbound system	Inbound system
Db2 for z/OS	User defined ¹	z/OS system	Db2 for z/OS subsystem
SQL Tuning Services	9444 or user defined ²	Your network	z/OS system

Notes:

1. The assignment of this port depends on your Db2 configuration.
2. The assignment of this port depends on the configuration of the WebSphere Liberty server.

What to do next

After you assign and configure the ports, make sure that all component systems in your installation can communicate with each other.

Chapter 7. Configuring the SQL Tuning Services z/OS environment for the setup ID

Before you install and configure SQL Tuning Services, customize your z/OS environment for the *tms_setup_userid*, which is the user ID that's used to install, configure, and run SQL Tuning Services.

Procedure

1. Allocate a minimum of 500 MB disk space to the home directory for *tms_setup_id*.
2. Set the CPU_TIMEMAX, ASSIZEMAX, and MEMLIMIT parameters to appropriate values in the OMVS segment of the RACF® profile for *tms_setup_id*.
 - a) Because SQL Tuning Services is considered a server process and requires sufficient system CPU to run without interruption, set CPU_TIMEMAX to unlimited; for example:

```
/bin/ulimit -t unlimited
```

- b) Set ASSIZEMAX to a minimum value of 1 GB; for example:

```
/bin/ulimit -A 1048576
```

- c) Set MEMLIMIT to a minimum value of 2.5 GB; for example:

```
/bin/ulimit -M 2560
```

To verify these settings for *tms_setup_userid*, issue the following command in a z/OS UNIX System Services shell session. The command returns output similar to the following example:

```
/bin/ulimit -a
core file          8192b
cpu time           unlimited
data size         unlimited
file size         unlimited
stack size        unlimited
file descriptors  520000
address space     1048576k
memory above bar  2560m
```

The values for `cpu time`, `address space`, and `memory above bar` correspond to CPU_TIMEMAX, ASSIZEMAX, and MEMLIMIT respectively.

Chapter 8. Installing and configuring SQL Tuning Services

This topic provides instructions for installing, configuring, and starting SQL Tuning Services.

Before you begin

Make sure that you complete all the preceding steps in the [installation roadmap](#).

Procedure

1. Install SQL Tuning Services on your z/OS system by following the instructions in the [IBM Database Services Expansion Pack Program Directory](#).

The SMP/E program installs some SQL Tuning Services components in a UNIX System Services directory and some components in partitioned data sets.

- By default, the UNIX System Services components are installed in the `/usr/lpp/IBM/db2tms/v2r1` directory, which is the parent folder of the SQL Tuning Services file system paths that are defined for DDNAME SDSN5TZF in the DSN5TDEF job. This directory is referred to as the `install_dir_zos` directory. For example, if the PATH of SDSN5TZF is defined to be `/usr/lpp/IBM/db2tms/v2r1/IBM`, the structure of `install_dir_zos` looks similar to the following example:

```
_Dir      755  OMVSKERN      8192  tmsinstall
_Dir      755  OMVSKERN      8192  tmsservice
_Dir      755  OMVSKERN      8192  IBM
```

The `install_dir_zos/IBM` directory contains two files: the DSN5DTMS pax file and the DSN5INSH script for unpacking the DSN5DTMS pax file.

- Two partitioned data sets are created to contain the following components:

hlq. SDSN5TSA

This data set contains the sample JCL.

hlq. SDSN5TDB

This data set contains the DBRMs.

2. Configure the required SSL certificate.

SQL Tuning Services uses the HTTPS protocol to secure network communications. You can use either a file-based keystore (PKCS12) or a key ring-based keystore (JCERACFKS).

You can use an existing key ring if one is available on your system, or you can generate a file-based certificate on the z/OS system that you are installing SQL Tuning Services on. The certificate must be a SAN (Subject Alternative Name) or wildcard SSL certificate that allows the specification of multiple domains or hosts. Make sure that your SSL certificate contains the IP addresses of your SQL Tuning Services system. The certificate must be CA- or self-signed.

If you decide to use a new CA-signed certificate, self-signed certificate, or an existing certificate pair (consisting of a certificate and a private key), complete the following steps to configure the certificate:

- a. Use an FTP program to copy the certificate in binary mode into the `install_dir_zos/tmsservice/Config/security` directory.
- b. Convert the certificate to the PKCS12 format.

3. Copy the `install_dir_zos/tmsinstall/tmsservice.config` file into a writable directory (`new_dir/tmsservice.config`) and edit the following installation options:

liberty_path

The system-level WebSphere Liberty server PATH directory. If your environment does not use the system-level Liberty server, modify this option with the correct path.

wlp_user_dir

The user-defined WebSphere Liberty server directory. If the directory that you specify does not exist, the installation process creates it for you. If the directory that you specify does exist, the *tms_setup_userid* must have write access to it because the installation process copies files into this directory.

server_name

The new WebSphere Liberty server for SQL Tuning Services. When this server name is confirmed, \$TMS_HOME will be *wlp_user_dir/servers/server_name*.

Important: This name is used as BPX_JOBNAME when SQL Tuning Services is started.

author_udf_admin

The UDF name for the SQL Tuning Services administrator; for example, IBMTMS.CANADMINISTER.

author_udf_user

The UDF name for the SQL Tuning Services user; for example, IBMTMS.CANVIEW.

host_name

The hostname of the server on which SQL Tuning Services is installed.

httpsport

The security port of the server on which SQL Tuning Services is installed.

log_level (optional)

The level of details (ALL, DEBUG, ERROR, unrecoverable, information, OFF, TRACE, WARN) to be recorded in the authentication and training service logs. The default is information (for example, `log_level=INFO`).

token_absolute_timeout (optional)

The absolute timeout in milliseconds for a user.

visual_explain_cache_timeout (optional)

The amount of time (in seconds) that the cache for Visual Explainer is kept.

visual_explain_resource_pool_size (optional)

Specify a number that indicates the number of Visual Explain, Access Path Comparison, and Workload Access Path Comparison request results that can be cached for each user.

Important: Specifying a large number causes SQL Tuning Services to use more memory and can negatively affect system performance. The recommended setting is a number less than 20.

capture_query_environment_cache_timeout (optional)

The amount of time (in days) that the result files of the Capture Query Environment function are kept in the server.

query_job_list_batch_size (optional)

The number of rows that are returned per batch by the Job Management function when jobs are queried.

4. Configure SQL Tuning Services by running the `tmservice.sh` script in the *install_dir_zos/tmsinstall* directory:

```
./tmservice.sh new_dir/tmservice.config
```

The script checks for the required certs files and other variables in your environment. When prompted, provide the following information:

temporary username and password

Specify a temporary username and password that are used before a repository database connection is set successfully in SQL Tuning Services. After the connection to the repository database is set successfully, this username and its corresponding password will be revoked immediately. You will need to log on to SQL Tuning Services with a *tms_userid* that is authorized to access the repository database.

TLSCertificatePassword

Specify the password of the TLS certificate file that is used to establish the https service through the WebSphere Liberty server. This value is case-sensitive.

5. Create the required UDFs in Db2 for z/OS, which is the repository database for SQL Tuning Services. The SQL Tuning Services repository service uses Db2 for z/OS for storing metadata tables. You must create an authentication UDF and grant access to users before you start the repository service:
 - a) Follow the instructions in the DSN5RUDF file to customize this sample job.
 - b) Submit your copy of the customized DSN5RUDF job to create the required UDFs and to grant required privileges to all *tms_userid* IDs.
 - c) Verify that the job runs successfully (with a return code of 0).
6. Bind the required packages in Db2 for z/OS.

To tune SQL on the database, you must bind the packages that SQL Tuning Services needs in advance.

 - a) Follow the instructions in the DSN5NDRP and DSN5NDTG sample jobs to customize them for your environment.
 - b) Submit your customized copy of the DSN5NDRP job to bind all the packages to the repository database. You need to submit this job only once because there is only one repository database.
 - c) Submit your customized copy of the DSN5NDTG job to bind all packages to tuning target databases. You need to submit this job for each tuning target database.
 - d) Optional: Submit your copy of the customized DSN5NBND job to bind all the packages to the same database where the repository database and target database reside together.
 - e) Verify that the jobs run successfully.
7. Grant required privileges to the *db2_authid_R* and *db2_authid_T* IDs.
 - a) Follow the instructions in the DSN5RTRP and DSN5RTTG sample jobs to customize them for your environment.
 - b) Submit your customized copy of the DSN5RTRP job to grant required privileges to the *db2_auth_R* ID.
 - c) Submit your customized copy of the DSN5RTTG job to grant required privileges to the *db2_auth_T* ID.
 - d) Optional: If the repository database and target database are on the same system, submit your copy of the customized DSN5RGRT job to grant all the required privileges to the *db2_authid_R* and *db2_authid_T* IDs.
 - e) Verify that the job runs successfully and ends with a return code of 0.
8. Optional: Set the access control on `replib_override.properties` to 640 so that only authorized UNIX System Services users can read or modify it.
9. Customize the DSN5STRT sample JCL procedure to start the SQL Tuning Services server.

Results

After all services are successfully deployed and the SQL Tuning Services server is started, the process of installation, configuration, and service deployment is complete. The base set of SQL Tuning Services features is ready for use by authorized users.

To enable the extended set of SQL Tuning Services that is provided by Db2 Query Workload Tuner, complete the steps in [Chapter 9, “Enabling extended tuning features,”](#) on page 25.

Chapter 9. Enabling extended tuning features

Before you can use the extended set of SQL Tuning Services features, you must install the license that is provided with IBM Db2 Query Workload Tuner for z/OS 6.1.

About this task

This task provides instructions for enabling the following extended set of tuning features, which are provided by Db2 Query Workload Tuner 6.1:

- Access Path Advisor
- Access Path Comparison
- Index Advisor
- Index Impact Analyzer
- Query Rewrite Advisor
- SQL Annotator
- Virtual Index Analyzer
- Workload Access Path Advisor*
- Workload Access Path Comparison*
- Workload Analytics Acceleration Advisor*
- Workload Candidate Acceleration Analyzer*
- Workload Explain*
- Workload Index Advisor*
- Workload Management*
- Workload Query Environment Collector*
- Workload Query Rewrite Advisor*
- Workload Statistics Advisor*
- Workload Virtual Index Analyzer

*: Workload analysis and tuning features are available for use as stand-alone APIs and with Db2 Administration Foundation. Their functionality is not integrated into Db2 Developer Extension.

Before you start the following procedure, make sure that Db2 Query Workload Tuner has been installed. For more information, see the [Db2 Query Workload Tuner for z/OS Program Directory](#).

Procedure

1. Locate the Db2 Query Workload Tuner `license.jar` file. After the SMP/E installation, the default location of this file is `/usr/lpp/IBM/qwtz`.
2. Create a license folder at the following location:

```
wlp_user_dir/servers/server_name/license
```

where `wlp_user_dir` and `server_name` match the values that you specified for these parameters in `tmservice.config` when you installed and configured SQL Tuning Services.

3. Copy the `license.jar` file to the license folder that you created.

Results

After you install the Db2 Query Workload Tuner license, you can start using the extended tuning features. You do not need to restart the SQL Tuning Services server.

Chapter 10. Configuring AT-TLS for SQL Tuning Services

SQL Tuning Services supports the use of AT-TLS to provide secure communication. Complete this step only if your environment uses AT-TLS.

Before you begin

The following instructions assume that the z/OS Communications Server policy agent (PAGENT) and AT-TLS are already configured on your system. For more information, see the following sections in [IBM z/OS V2R2 Communications Server TCP/IP Implementation: Volume 4 Security and Policy-Based Networking \(IBM Redbooks\)](#):

- Policy agent
- Application Transparent Transport Layer Security

Procedure

1. Define the SQL Tuning Services port in the AT-TLS policy.
2. Modify the WebSphere Liberty server to use the http protocol. By default, SQL Tuning Services uses the https protocol.

Edit the `wlp_user_dir/servers/server_name/server.xml` file and swap the `httpPort` and `httpsPort` values; for example:

Before

```
<httpEndpoint id="defaultHttpEndpoint"
  host="abc.com"
  httpPort="-1"
  httpsPort="9444"/>
```

After

```
<httpEndpoint id="defaultHttpEndpoint"
  host="abc.com"
  httpPort="9444"
  httpsPort="-1"/>
```

3. Edit the `wlp_user_dir/servers/server_name/config/tmsserver_override.properties` file and add the following entry:

```
host4url=https://service_ip
```

where `service_ip` is the IP address or hostname that SQL Tuning Services starts with. This variable is defined in the `tmsservice.conf` file.

4. Restart the SQL Tuning Services server.

Chapter 11. Getting started with SQL Tuning Services

This topic provides instructions for using SQL Tuning Services after it has been installed and configured to tune an SQL statement.

About this task

This task consists of the following parts:

- **Part 1** shows you how to generate a token for the temporary SQL Tuning Services administrator and how to specify the repository database that SQL Tuning Services uses to manage its own data, both of which are one-time-only steps.
- **Part 2** shows you how to create the repository database and tables by using the Repository Database Setup API (repopdb). As an alternative, you can also use DSN5REPO sample job in SDSN5TSA to create the repository database and tables. Creating the repository database is a one-time-only step.
- **Part 3** shows you how to connect to the target Db2 subsystem that contains SQL that you want to tune.
- **Part 4** shows you how to create the EXPLAIN tables that are used by SQL Tuning Services.
- **Part 5** shows you how to use the Visual Explainer SQL Tuning Services API to tune an SQL statement. You can apply these instructions to any of the other SQL Tuning Services APIs.

Notes:

- The example `curl` commands in this topic use syntax that is compatible with Linux® and UNIX environments. If you are using a Windows environment, the syntax will be different.
- When you issue any of the example `curl` commands in this topic, you might receive an error that indicates a problem with the server certificate, as shown in the following example:

```
curl: (60) Peer's certificate issuer has been marked as not trusted by the user.  
More details here: http://curl.haxx.se/docs/sslcerts.html
```

If you are sure that the server certificate is valid, you can disable it by issuing the `-k/--insecure` command.

- You can invoke all the SQL Tuning Services APIs from the Swagger interface. See [Invoking SQL Tuning Services APIs](#) for instructions.

Procedure

Part 1

1. Generate a token for the temporary SQL Tuning Services administrator by calling the Authentication Service API.

You'll use this token as a Bearer token in the next step to specify the SQL Tuning Services repository database. To call the Repository Database Setup API, you need the temporary username and password that you defined in installation and configuration step “4” on page 22.

```
curl -X 'POST' \  
  'https://service_ip:httpsport/tuningservice/v1/auth/tokens' \  
  -H 'Accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "userid": "ADMIN1",  
    "password": "password"  
  }'
```

2. Specify the SQL Tuning Services repository database by using the token that you created in the previous step.

Specifying the SQL Tuning Services repository database is a one-time step.

```
curl -X POST 'https://service_ip:httpsport/tuningservice/v1/set_repo' \
--header 'Authorization: Bearer Bearer_token_generated_in_step_1' \
--header 'Content-Type: application/json' \
--data-raw '{
  "credential": {
    "password": "password",
    "security_mechanism": "3",
    "user": "db2_authid_R"
  },
  "location": "location",
  "port": "port_number",
  "host": "Db2_host"
}'
```

Part 2

3. Generate a token for an SQL Tuning Services administrator by calling the Authentication Service API again.

After you specify the repository database successfully, the temporary username is immediately revoked. You need to generate the token again with an ID that is authorized to access the repository database. This ID must also have permission to execute the `author.udf.user` UDF and the `author.udf.admin` UDF.

```
curl -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/auth/tokens' \
-H 'Accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "userid": "ADMIN1",
  "password": "password"
}'
```

4. Create the SQL Tuning Services repository database by calling the Repository Database Setup API (`repopdb`) with the token that you created in the previous step. Creating the SQL Tuning Services repository database is a one-time step. You don't need to repeat this step to use other tuning APIs.

Note: This step creates the initial SQL Tuning Services repository database. If you already created the SQL Tuning Services repository database and need to upgrade it, follow the instructions in [Upgrading SQL Tuning Services](#).

Make sure that the ID that you use has been granted the privileges for the `db2_authid_R` user ID.

Two modes are available for running this API:

- Use Run DDL mode to create the repository database directly by running the API with `"runddl": true`.
- Use Download DDL mode to download the customized DDL files from the `installation_dir` by running the API with `"runddl": false`.

```
curl -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/repopdb' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_1' \
-H 'Content-Type: application/json' \
-d '{
  "default_sqlid": "ADMIN1",
  "ix16kbufferpool": "BP16K2",
  "ix4kbufferpool": "BP2",
  "ix8kbufferpool": "BP8K2",
  "runddl": true,
  "storagegroup": "SYSDEFLT",
  "ts16kbufferpool": "BP16K1",
  "ts4kbufferpool": "BP1",
  "ts8kbufferpool": "BP8K1"
}'
```

Part 3

5. Create a tuning connection profile to connect to the Db2 for z/OS subsystem that contains the SQL that you want to tune.

Note: A tuning connection profile is not required to use the following tuning features:

Access Path Advisor	SQL Annotator
Capture Query Environment	SQL Formatter
EXPLAIN Tables Management	Statistics Advisor
Index Analyzer	Visual Explain
Index Impact Analysis	What if
Query Rewrite Advisor	

If you intend on using only the features that are listed in the previous table, you do not need to create a tuning connection profile.

A tuning connection profile is a set of JDBC connection information that you define to make a Db2 for z/OS database available to SQL Tuning Services. You can add multiple tuning target connection profiles to SQL Tuning Services. When you create a tuning connection profile, you provide a Db2 for z/OS user ID and password, which persists on the repository Db2 subsystem. An advantage of using tuning connection profiles is that multiple users can use the same profile to run tuning actions. However, because the password persists on the Db2 subsystem, whenever that password changes, the tuning connection profile will need to be modified with the new password.

You can create, update, delete, inquire, and share a database connection profile by calling Connection Profile Management APIs. Any SQL Tuning Services user can manage their own database connection profile. The SQL Tuning Services administrator can manage all the existing database connection profiles.

To create a tuning connection profile, modify and run the following example POST request. Specify the Bearer token that you created in step 3. In this example, the tuning connection profile name is *tuning_connection_profile_1*.

```
curl -X 'POST' \
  'https://service_ip:httpsport/tuningservice/v1/connections' \
  -H 'Accept: application/json' \
  -H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
  -H 'Content-Type: application/json' \
  -d '{
    "collection_cred": {
      "password": "password",
      "security_mechanism": "3",
      "user": "db2_authid_T"
    },
    "data_server_type": "DB2Z",
    "host": "Db2_host",
    "location": "location",
    "name": "tuning_connection_profile_1",
    "port": "port_number"
  }'
```

Tips:

- SQL Tuning Services also supports multi-factor authentication (MFA) on every account password field. To implement logging with an MFA token, append a factor separator and a valid MFA token after the password for the account:
 - a. Use the IBM Verify application to generate a QR code, and then scan the QR code to obtain an MFA token (in the following example, the token is *123456*).
 - b. Append a colon (:) as the factor separator and the MFA token (*123456*) after the original account password in the Connection Profile Management API.

For example, in the previous example POST request, you would replace:

```
"password": "password",
```

with:

```
"password": "original_password:123456",
```

- To configure an IPv6 connection, when you specify the host parameter, specify the IPv6 address and enclose it in brackets. For example:

```
"host": "[::ffff:9.30.84.95]",
```

Alternatively, you can use SSL to connect to Db2. By default, the version of the TLS protocol is determined by the JRE that is used by the application. To specify a TLS version that is different than the one that the JRE uses, modify and run the following example POST request as needed:

```
curl -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/connections' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
-H 'Content-Type: application/json' \
-d '{
  "collection_cred": {
    "password": "your_password",
    "user": "db2_authid_T"
  },
  "data_server_type": "DB2Z",
  "host": "Db2_host",
  "location": "location",
  "sslConnection": "true",
  "sslTrustStoreLocation": "safkeyring://sysadm/tlsKeyring",
  "sslTrustStorePassword": "password",
  "additionalProperties": {
    "sslTrustStoreType": "JCERACFKS",
    "sslVersion": "TLSv1.3"
  },
  "name": "profile_1",
  "port": "port_number"
}'
```

Part 4

6. Create the EXPLAIN tables to store the EXPLAIN information that SQL Tuning Services relies on to tune SQL. Creating the EXPLAIN tables is a one-time step. You don't need to repeat this step to use other tuning APIs.

You can create EXPLAIN tables by referring to the Db2 for z/OS [EXPLAIN tables documentation](#), or you can use the EXPLAIN Tables Management API.

The syntax for using the EXPLAIN Tables Management API depends on whether you are using a tuning connection profile, which is documented in the previous step, and whether you are using SSL. Select the appropriate method for your environment:

If you are using a tuning connection profile:

Modify and run the following example POST request. Use the token that you generated in step 3. The EXPLAIN tables that you create are qualified by a user ID for you or your group. Only the SQL Tuning Services administrator or the user who owns the database connection can call this API.

To call the EXPLAIN Tables Management API, the user in profile_1 must be authorized to call the ADMIN_EXPLAIN_MAINT store procedure. Make sure that the account you use has the proper privileges to create the EXPLAIN tables as documented in [Chapter 5, "Setting up required user IDs and permissions,"](#) on page 13.

```
curl -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/explaintb' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
-H 'Content-Type: application/json' \
-d '{
  "action": "CREATE",
  "auth_id": "db2_authid_T",
  "bp16kblob": "BP16K1",
  "bp32kblob": "BP32K1",
}
```

```

"bp4kblob": "BP1",
"bp8kblob": "BP8K1",
"connection": "tuning_connection_profile_1",
"database_name": "TMSCTRDB",
"ixbufferpool": "BPIDX1",
"managealias": "NO",
"mode": "RUN",
"schema_alias": "IBMTMS_alias",
"schema_name": "IBMTMS",
"storagegroup": "SYSDEFLT",
"storagegroup_idx": "SYSDEFLT",
"tableset": "ALL",
"ts16kbufferpool": "BP16K2",
"ts32kbufferpool": "BP32K2",
"ts4kbufferpool": "BP4K2",
"ts8kbufferpool": "BP8K2"
}'

```

If you are not using a tuning connection profile:

Modify and run the following example POST request to use the database connection attributes. Make sure that the account information you use in the connection attributes has the proper privileges to create the EXPLAIN tables.

```

url -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/explaintb' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
-H 'Content-Type: application/json' \
-d '{
  "action": "CREATE",
  "auth_id": "db2_authid_T",
  "bp16kblob": "BP16K1",
  "bp32kblob": "BP32K1",
  "bp4kblob": "BP1",
  "bp8kblob": "BP8K1",
  "connection_attributes": {
    "collection_cred": {
      "password": "your_password",
      "security_mechanism": "3",
      "user": "db2_authid_T"
    },
    "data_server_type": "DB2Z",
    "host": "Db2_host",
    "location": "location",
    "port": "port_number"
  },
  "database_name": "TMSCTRDB",
  "ixbufferpool": "BPIDX1",
  "managealias": "NO",
  "mode": "RUN",
  "schema_alias": "IBMTMS_alias",
  "schema_name": "IBMTMS",
  "storagegroup": "SYSDEFLT",
  "storagegroup_idx": "SYSDEFLT",
  "tableset": "ALL",
  "ts16kbufferpool": "BP16K2",
  "ts32kbufferpool": "BP32K2",
  "ts4kbufferpool": "BP4K2",
  "ts8kbufferpool": "BP8K2"
}'

```

If you are using SSL:

```

curl -X 'POST' \
'https://service_ip:httpsport/tuningservice/v1/explaintb' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
-H 'Content-Type: application/json' \
-d '{
  "action": "CREATE",
  "auth_id": "db2_authid_T",
  "bp16kblob": "BP16K1",
  "bp32kblob": "BP32K1",
  "bp4kblob": "BP1",
  "bp8kblob": "BP8K1",
  "connection_attributes": {
    "collection_cred": {
      "password": "your_password",

```

```

    },
    "user": "db2_authid_T"
  },
  {
    "data_server_type": "DB2Z",
    "host": "Db2_host",
    "location": "location",
    "sslConnection": "true",
    "port": "port_number",
    "sslTrustStoreLocation": "safkeyring://sysadm/tlsKeyring",
    "sslTrustStorePassword": "password",
    "additionalProperties": {
      "sslTrustStoreType": "JCERACFKS"
    }
  },
  {
    "database_name": "TMSCTRDB",
    "ixbufferpool": "BPIDX1",
    "managealias": "NO",
    "mode": "RUN",
    "schema_alias": "IBMTMS_alias",
    "schema_name": "IBMTMS",
    "storagegroup": "SYSDEFLT",
    "storagegroup_idx": "SYSDEFLT",
    "tableset": "ALL",
    "ts16kbufferpool": "BP16K2",
    "ts32kbufferpool": "BP32K2",
    "ts4kbufferpool": "BP4K2",
    "ts8kbufferpool": "BP8K2"
  }
}'

```

7. Tune a query by calling the Visual Explainer API.

After you have completed the previous steps, you can use SQL Tuning Services APIs to run the tuning features that are available in your SQL Tuning Services environment. This step uses the Visual Explainer API to demonstrate how to use SQL Tuning Services with a simple SELECT statement, but can be applied to any other SQL Tuning Services API.

Note: The tuning features that are available depend on whether you are using SQL Tuning Services that is provided with IBM Database Services Expansion Pack or SQL Tuning Services that is provided with Db2 Query Workload Tuner for z/OS. See [Overview of SQL Tuning Services](#) for more information.

Make sure that the account that you use has the proper privileges to call the Visual Explainer API as documented in [Chapter 5, "Setting up required user IDs and permissions,"](#) on page 13.

When you submit a tuning job by calling a SQL Tuning Services API, the job runs using the database connection credential ID.

The syntax for submitting a Visual Explainer job depends on whether you are using a tuning connection profile. Select the appropriate method for your environment:

If you are using a tuning connection profile:

Modify and submit the following Visual Explainer POST request:

```

curl -X 'POST' \
  'https://service_ip:httpsport/tuningservice/v1/ve' \
  -H 'Accept: application/json' \
  -H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
  -H 'Content-Type: application/json' \
  -d '{
    "sql_text": "select * from sysibm.systables",
    "connection": "tuning_connection_profile_1",
    "explain_schema": "schema_name_in_step_6"
  }'

```

If you are not using a tuning connection profile:

Modify and submit the following Visual Explainer POST request:

```

curl -X 'POST' \
  'https://service_ip:httpsport/tuningservice/v1/ve' \
  -H 'Accept: application/json' \
  -H 'Authorization: Bearer Bearer_token_generated_in_step_3' \
  -H 'Content-Type: application/json' \
  -d '{
    "sql_text": "select * from sysibm.systables",
    "connection_attributes": {
      "collection_cred": {

```



```
"password": "your_password:",
"security_mechanism": "3",
"user": "db2_authid_T"
},
"data_server_type": "DB2Z",
"host": "Db2_host",
"location": "location",
"port": "port_number"
},
"explain_schema": "schema_name_in_step_6"
}'
```

After you submit the job, you can use the job ID to view the results to query the job information, or to cancel the job. For example, the following command returns a URL that you can copy into a web browser to view the results of running the Visual Explainer job.

```
curl -X 'GET' \
'https://service_ip:httpsport/tuningservice/v1/jobs/job_ID/result' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer Bearer_token_generated_in_step_3'
```

Chapter 12. Reconfiguring SQL Tuning Services

To change your SQL Tuning Services configuration after the initial installation and configuration, you need to modify one or more configuration files depending on the types of changes that you want to make.

Procedure

1. Edit the following configuration files, which are located in the *TMS_HOME/Config* directory, and update the parameters as needed.

Configuration properties are defined in three separate configuration files:

- [application.properties](#)
- [tmserver_override.properties](#)
- [reopdb_override.properties](#)

application.properties

This file contains the following system-level global settings:

log4j2-related settings:

logger.ibm.type=AsyncLogger



Attention: Delete the **logger.ibm.type=AsyncLogger** parameter from the `application.properties` file. This parameter can cause an out of memory condition that requires the system to be restarted.

logger.ibm.appenderRef.rollingFileInfo.ref=rollingFileInfo

The logging options set name.

property.LOG_FILE_PATH=logs

The file path for logging files.

property.APP_NAME=T.S

The prefix for the names of logging files.

property.CHARSET=utf-8

The encoding set for logging files.

property.ROLLING_INTERVAL=1

How often a log rollover occurs based on the most specific time unit in the date pattern.

property.MAX_LOG_SIZE=10MB

The maximum size for logging files.

property.MAX_HISTORY=30

The maximum number of compressed logging files that are retained.

property.MAX_AGE=15d

The amount of time, in days, that logging files are retained.

property.LOG_PATTERN=%date{HH:mm:ss.SSS}\${APP_NAME}[%logger{64}]%-5level-%msg%n

The naming convention that's used to convert the logging file name from `console.log` to the name defined by LOG PATTERN.

Circuit breaker settings

A circuit breaker pattern is a lightweight fault tolerance capability that you can use to protect SQL Tuning Services from saturated access and continuing failure (for example, a high volume of visits during business peak time). Circuit breakers can proactively isolate and then recover the protected APIs quickly based on statistical trends in service layer monitoring. Currently, the protection is available on the token generation and connection profile synchronous APIs.

The following circuit breaker properties can be customized for your runtime environment and specific business requirements:

failureRateThreshold

The failure rate threshold as a percentage. When the failure rate is equal to or greater than the threshold, the circuit breaker transitions to open and starts short-circuiting calls. The default value is 50.

slowCallRateThreshold

The slow call rate threshold as a percentage. The circuit breaker considers a call to be slow when the call duration is greater than the `slowCallDurationThreshold`. When the percentage of slow calls is equal to or greater than the threshold, the circuit breaker transitions to open and starts short-circuiting calls. The default value is 100.

slowCallDurationThreshold

The duration threshold above which calls are considered to be slow and increase the rate of slow calls. The default value is 60000 ms.

permittedNumberOfCallsInHalfOpenState

The number of permitted calls when the circuit breaker is half open. The default value is 10.

slidingWindowType

The type of the sliding window that is used to record the outcome of calls when the circuit breaker is closed. The sliding window can either be `COUNT-BASED` or `TIME-BASED`:

- If the sliding window is `COUNT_BASED`, the last `slidingWindowSize` calls are recorded and aggregated.
- If the sliding window is `TIME_BASED`, the calls of the last `slidingWindowSize` seconds are recorded and aggregated.

The default setting is `COUNT_BASED`.

slidingWindowSize

The size of the sliding window that is used to record the outcome of calls when the circuit breaker is closed. The default value is 100.

minimumNumberOfCalls

The minimum number of calls that are required (per sliding window period) before the circuit breaker can calculate the error rate or slow call rate. For example, if `minimumNumberOfCalls` is 10, then at least 10 calls must be recorded before the failure rate can be calculated. If only nine calls have been recorded, the circuit breaker will not transition to open even if all nine calls have failed. The default value is 100.

waitDurationInOpenState

The time that the circuit breaker should wait before transitioning from open to half-open. The default value is 60000 ms.

automaticTransitionFromOpenToHalfOpenEnabled

If this property is set to true, the circuit breaker will automatically transition from open to half-open, and no call is needed to trigger the transition. A thread is created to monitor all the instances of circuit breakers to transition them to `HALF_OPEN` when `waitDurationInOpenState` passes. Whereas, if this property is set to false, the transition to `HALF_OPEN` happens only if a call is made, even after `waitDurationInOpenState` is passed. The advantage of using this property is that no thread monitors the state of all circuit breakers. The default value is true.

You can set the following bulkhead pattern properties:

maxConcurrentCalls

The maximum number of parallel executions allowed by the bulkhead. The default value is 25.

maxWaitDuration

The maximum amount of time that a thread should be blocked when it attempts to enter a saturated bulkhead. The default value is 0.

tmserver_override.properties

This file contains the following SQL Tuning Services-related settings:

appl_id=APPLDB1C

The application ID that was set in RACF when the specified credential type is passticket.

token_absolute_timeout=28800000

The absolute timeout time for a user, in milliseconds.

author.udf.user=IBMTMS.CANVIEW

The UDF name for a SQL Tuning Services user (for example, IBMTMS.CANVIEW).

author.udf.admin=IBMTMS.CANADMINISTER

The UDF name for a SQL Tuning Services administrator (for example, IBMTMS.CANADMINISTER).

ve.cache.timeout=1200

The amount of time, in seconds, that the cache for the Visual Explainer function is valid.

ve.cache.limitation=10

The number of Visual Explainer result sets that are kept in memory per user.

Important: Keeping a large number of Visual Explainer result sets in memory can negatively affect system performance.

cqe.file.retention.period=7

The length of time, in days, that the result files of the Capture Query Environment function are retained.

query.joblist.batchsize=50

The number of rows that are returned per batch by the Job Management function when jobs are queried.

repopdb_override.properties

This file contains the following repository database-related settings:

port=8010

The port number of the Db2 for z/OS subsystem that's used as the repository database.

user=TSADM1

The username of the *db2_authid_R* ID that's used to access the Db2 for z/OS subsystem that's used as the repository database.

sslConnection=false

To enable SSL connections to the repository database, set this parameter to true.

credential_type=

If you are using PassTickets as the authentication method, specify passticket.

locationName=DB1

The location name of the Db2 for z/OS subsystem that's used as the repository database. This value is case-sensitive.

password=wtiv2_252b0ce7696f03b959dfa17264ac267b

The password of the *db2_authid_R* ID that's used to access the Db2 for z/OS subsystem that's used as the repository database. Typically, this password needs to be changed only when it expires.

For instructions for changing this password, see [Chapter 15, “Changing the repository database user ID and password,”](#) on page 45.

host=9.30.137.24

The IP address of the Db2 for z/OS subsystem that's used as the repository database.

sslTrustStoreLocation=

The location of the SSL truststore file.

sslTrustStorePassword=

The password of the SSL truststore file. Typically, this password needs to be changed only when it expires. This value is case-sensitive.

You can enter this password in regular text or you can run the following command to encrypt your new password and then paste the result in this field:

```
'java -classpath path_to_the_jar/com.ibm.aps.tools.util.crypt.jar  
-Ddshome=TMS_HOME/bin com.ibm.aps.tools.util.crypt.CryptUtils new_password
```

sslKeyStoreLocation=

The location on your z/OS system where the SSL keystore (keystore.jks) file is stored.

sslKeyStorePassword=

The password where you store the SSL keystore (keystore.jks) file on your z/OS system.

Typically, this password needs to be changed only when it expires. This value is case-sensitive.

You can enter this password in regular text or you can run the following command to encrypt your new password and then paste the result in this field:

```
'java -classpath path_to_the_jar/com.ibm.aps.tools.util.crypt.jar  
-Ddshome=TMS_HOME/bin com.ibm.aps.tools.util.crypt.CryptUtils new_password
```

sslCertLocation=

The location of the AT-TLS certificate file that's used to establish https service through the WebSphere Liberty server.

2. When you're finished changing configuration parameters, use the *tms_setup_userid* ID to restart SQL Tuning Services by running the */tms-stop.sh* script followed by the *tms-start.sh* script.

These scripts are located in the *install_dir_zos* directory.

Chapter 13. Upgrading SQL Tuning Services

When you upgrade to a new release of SQL Tuning Services, you need to modify several configuration parameters to use the new release. You might also need to upgrade the repository database as part of the upgrade process.

Procedure

1. After you complete the SMP/E installation of a new SQL Tuning Services server, perform installation steps “3” on page 21 and “4” on page 22 to specify a unique WebSphere Liberty server directory (**wlp_user_dir**) for the new SQL Tuning Services runtime.

This approach enables you to retain the existing SQL Tuning Services runtime in the original WebSphere Liberty server directory.

2. Modify the DSN5STRT sample JCL start procedure to point to the new WebSphere Liberty server by following the instructions at the top of the job.
3. After you start the SQL Tuning Services server, specify the existing repository database to the SQL Tuning Services server.

Existing tuning activities will be loaded by the new SQL Tuning Services server.



Attention: An SQL Tuning Services repository database cannot be shared by more than one SQL Tuning Services server. Ensure that each SQL Tuning Services server that is started uses a different repository database.

4. If necessary, upgrade the SQL Tuning Services repository database.

When you upgrade to a new SQL Tuning Services release or apply maintenance, you might also need to upgrade the repository database. Refer to the information that accompanies the release or maintenance to determine if the repository database must be upgraded.

Note: Before the availability of APAR PH55158, the /tuning-service/v1/repodb API was used to both create a new repository database and to upgrade an existing repository database. APAR PH55158 and subsequent APARs provide JCL that simplifies the process of upgrading the SQL Tuning Services repository database.

After the SMP/E installation of APAR PH55158 is complete, the following two JCL jobs are available in the SDSN5TSA data set:

- DSN5UPR1 upgrades the repository database from SQL Tuning Services 1.0 to the most current version.
- DSN5UPR2 upgrades the repository database from SQL Tuning Services 2.1 to the most current version.

After the SMP/E installation of APAR PH56970 is complete, the following additional JCL job is available in the SDSN5TSA data set:

- DSN5UPR3 upgrades the repository database from SQL Tuning Services 2.1 (PH56970) to the most current version.

Select the job that is appropriate for your version of SQL Tuning Services, modify it for your environment, and run it to upgrade an existing repository database.

5. Customize and run the DSN5NBND, DSN5NDRP, and DSN5NDTG sample jobs to bind the required packages.

Chapter 14. Invoking SQL Tuning Services APIs

This topic provides instructions for invoking SQL Tuning Services APIs from the Swagger interface.

About this task

SQL Tuning Services offers the Swagger interface to make it easy to try out Restful APIs. The address of the Swagger interface is:

```
https://{host}:{port}/tuning-service/v1/swagger-ui/index.html
```

You can invoke all of the SQL Tuning Services APIs directly from the Swagger interface after you generate and authorize a Bearer token for your user ID.

Procedure

1. Generate a token.
 - a) Expand `/tuning-service/v1/auth/tokens`
 - b) Click **Try it out**.
 - c) Specify your user ID and password.
 - d) Click **Execute** to generate the token.
 - e) Copy the generated token (without the quotation marks).
2. Authorize the token that you generated.
 - a) Click **Authorize** in the upper right corner of the window.
 - b) In the **Value** field, specify the token in the following format:

```
Bearer <token_value>
```

Make sure that the string `Bearer` and the token value are separated by a single blank space.

- c) Click **Authorize**.

Results

You can now invoke the SQL Tuning Services APIs from the Swagger interface.

Chapter 15. Changing the repository database user ID and password

You will likely need to change the password for the `db2_authid_R` ID at regular intervals based on your password-expiration policies or if you encounter authentication issues with the SQL Tuning Services repository database. Similarly, you might need to assign a different user to the `db2_authid_R` ID at some point.

Before you begin

If you are changing the password only, you can proceed directly to the procedure.

If you are assigning a new user to the `db2_authid_R` ID, you must first set up the new ID by following the instructions in the `db2_authid_R` section of [Chapter 5, “Setting up required user IDs and permissions,”](#) on page 13.

If the repository database has already been created by the previous `db2_authid_R` ID, you need to grant the required DBADM privilege to the new `db2_authid_R` ID by running the following command:

```
GRANT DBADM ON DATABASE IBMTMSDB TO db2_authid_R_new WITH GRANT OPTION
```

About this task

To change the `db2_authid_R` password or to assign a different user to this ID, run the `repoDBCredReset.sh` script.

Procedure

1. Stop the SQL Tuning Services server.
2. Run the `wlp_user_dir/repoDBCredReset.sh` script to specify a new password, a new user ID, or both.
3. Restart the SQL Tuning Services server.

Chapter 16. API reference for SQL Tuning Services

SQL Tuning Services provides the following APIs.

Important: For detailed information about these APIs, including the parameters that are required to use them, consult the swagger doc that is available after you install the product at https://service_ip:httpsport/tuningservice/v1/swagger-ui/index.html. To display the parameters for an API, expand the entry for that API and click the **Schema** tab.

Access Path Advisor ^{“1” on page 54}

API

POST /tuningservice/v1/apa

Description

Generate access path recommendations for improving the performance of a single query and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET `{job_id}/result` API.

Access Path Comparison ^{“1” on page 54}

API

POST /tuningservice/v1/apc

Description

Compare two existing access paths by specifying their job IDs and return a job ID as an asynchronous request. Both jobs must be explained before the comparison request can be submitted. The requester can retrieve the results of the comparison by using the job management APIs.

Authentication Service

API

POST /tuningservice/v1/auth/tokens

Description

Retrieve a reusable token that is used to securely call the other SQL Tuning Services APIs. The requester must be a valid SQL Tuning Services user. When specifying this token in other APIs, it must be prefixed by the label "Bearer" (for example, `Bearer token`).

API

GET /tuningservice/v1/license

Description

Retrieve license information from the tuning services server for a permanent license or from the repository database for a trial license.

Connection Profile Management

API

GET /tuningservice/v1/connections

Description

Retrieve information about all database connection profiles. The requester must have administrator privileges.

API

POST /tuningservice/v1/connections

Description

Create a connection profile in a database that is used for SQL Tuning Services.

API

DELETE /tuningservice/v1/connections/{dbprofile_name_list}

Description

Delete one or more database connection profiles. The requester must have administrator privileges or be the owner of the connection profile.

API

GET /tuning/v1/connections/{dbprofile_name}

Description

Retrieve information about a specific database connection profile. The requester must have administrator privileges or be the owner of the connection profile.

API

PUT /tuning/v1/connections/{dbprofile_name}

Description

Modify a database connection profile. The requester must have administrator privileges or be the owner of the connection profile.

EXPLAIN Tables Management**API**

POST /tuning/v1/explaintb

Description

Create EXPLAIN tables, upgrade the tables to the format for the current Db2 version, and complete other administrative tasks. The requester must have administrator privileges or be the owner of the connection profile.

Index Advisor [“1” on page 54](#)**API**

POST /tuning/v1/ia

Description

Generate recommendations for creating, altering, or dropping indexes that can improve the performance of a single query and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET {job_id}/result API.

Index Impact Analyzer [“1” on page 54](#)**API**

POST /tuning/v1/iaa

Description

Generates a report that shows the impact of applying the index changes that are recommended by Index Advisor and Workload Index Advisor.

Job Management**API**

GET /tuning/v1/jobs

Description

Query information about multiple jobs (job name, connection, job status, and other characteristics). Jobs are returned in batches. You control the batch size by using the start_row and end_row parameters. The maximum batch size is 1000 rows per batch. If the stmt_text parameter is set to true, the maximum batch size is 50 rows per batch. Because the job list can be part of qualified jobs, check the total_job_no parameter to determine the total number of jobs, and use pagination to retrieve the remaining jobs. The returned job information list is retrieved from all job records in the repository database and ordered by job ID. If the requester has administrator privileges, multiple jobs are returned regardless of who created them. Otherwise, only jobs that were created by the requester are returned.

API

GET /tuning/v1/jobs/{job_id}

Description

Query information about a job (job name, connection, job status, and so on). The requester must be the creator of the job or have administrator privileges.

API

POST /tuning-service/v1/jobs/{job_id}/cancel

Description

Cancel a job that is in RUNNING status. The requester must be the creator of the job or have administrator privileges.

API

GET /tuning-service/v1/jobs/{job_id}/result

Description

Get the results of a job that is in COMPLETED status. The results content varies depending on the job component (CQE, FM, SA, VE, and so on). The requester must be the creator of the job or have administrator privileges.

API

DELETE /tuning-service/v1/jobs/{job_id_list}

Description

The requester must be the creator of the job or have administrator privileges.

Query Environment Collector**API**

POST /tuning-service/v1/cqe

Description

Generate diagnostic information for a single query, which is needed by IBM Support for troubleshooting purposes, and return a job ID as an asynchronous request. The requester can get the download link of the diagnostic information by using the GET *job_result* API.

Query Rewrite Advisor [“1” on page 54](#)**API**

POST /tuning-service/v1/qra

Description

Generate recommendations for rewriting a single query to improve its performance and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET *{job_id}/result* API.

Repository Database Setup**API**

POST /tuning-service/v1/repodb

Description

The SQL Tuning Services repository database is a Db2 for z/OS database that SQL Tuning Services uses to manage its own data. To create the repository database, you must be an administrator and you must first generate a token by calling the Authentication Service API.

SQL Annotator [“1” on page 54](#)**API**

POST /tuning-service/v1/annotator

Description

Annotate an SQL statement for a single query and return a job ID as an asynchronous request. The requester can retrieve the annotated statement structure for a single SQL query by using the job management APIs.

SQL Capture**API**

POST /tuning-service/v1/sqlcapture/catalog

Description

Retrieve all queries from packages, plans, and stabilized dynamic queries that are stored in Db2 catalog tables.

API

POST /tuning-service/v1/sqlcapture/file

Description

Retrieve all queries from an uploaded file. The following file encoding schemes are supported:

- UTF8 ("UTF-8")
- UTF16 ("UTF-16")
- ISO88591 ("ISO8859-1")
- ISO885915 ("ISO8859-15")
- CP1251 ("CP1251")
- CP1252 ("CP1252")
- GB18030 ("GB18030")
- SJIS ("SJIS")

API

POST /tuning-service/v1/sqlcapture/stmtcache

Description

Retrieve all queries that are saved in the dynamic statement cache. Some filters are based on execution statistics (for example, STAT_CPU). To enable execution statistics collection, start a Db2 statistics trace for IFCIDs 316, 317, and 318.

API

POST /tuning-service/v1/sqlcapture/userdefinedrepo

Description

Retrieve all queries from tables and views that are stored in a user-defined repository. The text of the SQL statements and the runtime metrics might be in more than one table. If so, you need to create a view that joins the tables on an ID that is unique to each set of runtime metrics, and you must have the SELECT privilege on that view.

For example, the following statement creates a view that joins columns from the Db2 SYSIBM.SYSPACKSTMT and SYSIBM.SYSPACKAGE tables. The TEXT and SCHEMA columns are required. Other columns are included to facilitate filtering and ordering.

```
CREATE VIEW MYVIEW
(SUBSYSTEM, PLANNAME, COLLID, PACKNAME, VERSION, CONSISTOKEN, OWNER, TEXT, ID,
SEQNO, SCHEMA, STARTINTERVAL, ENDINTERVAL, SECTNO, STMTNO, EXECCOUNT, CPUTIME,
ELAPTIME, NGETPAGE, METRICID )
AS ( SELECT 'db2subsys', 'plannam', A.COLLID, A.NAME, C.VERSION,
C.CONTOKEN, C.OWNER, A.STATEMENT, A.STMT_ID, A.STMT_ID, C.OWNER,
TIMESTAMP('2015-06-15 13:05:12'), TIMESTAMP('2015-06-15 14:05:12'),
A.SECTNO, A.STMTNO, 1, -1, -1, -1, -1
FROM SYSIBM.SYSPACKSTMT AS A, SYSIBM.SYSPACKAGE AS C
WHERE A.COLLID = C.COLLID AND A.NAME = C.NAME AND C.NAME = 'AOC50ADM');;
```

API

POST /tuning-service/v1/sqlcapture/dataset

Description

Retrieve the contents of a sequential data set or a member of a partitioned data set (PDS or PDSE). To retrieve the contents of an uncataloged data set, include the volume serial on the request. This API invokes the z/OSMF API to read the data set.

SQL Formatter**API**

POST /tuning-service/v1/formatter

Description

Format an SQL statement for a single query and return a job ID as an asynchronous request. The requester can retrieve the formatted SQL by using the job management APIs.

Statistics Advisor

API

POST /tuning-service/v1/sa

Description

Generate statistics data collection advice for database objects or for a single query and return a job ID as an asynchronous request. To invoke this API, the Db2 STATFDBK_SCOPE subsystem parameter must be set to ALL, DYNAMIC, or STATID. The requester can get the recommendations by using the GET *job_result* API.

User Privilege Management on Connection Profile

API

POST tuning-service/v1/userprofileprivileges

Description

Set the owner and user for connection profiles. The requester must have administrator privileges or be the owner of the connection profile. Sharing a tuning profile with another user grants that user all of the privileges associated with that profile. Ensure that the user or users who you are sharing the profile with have these privileges.

API

GET /tuning-service/v1/userprofileprivileges

Description

Get the owner and user for target connection profiles. The requester must have administrator privileges or be the owner of the connection profile.

API

GET /tuning-service/v1/userprofileprivileges/{dbprofile_name}

Description

Get the owner and user of a specific target connection profile. The requester must have administrator privileges or be the owner of the connection profile.

Virtual Index Analyzer^{“1” on page 54}

API

POST /tuning-service/v1/whatif

Description

Virtually tests indexes to determine if the performance of a single query can be improved by creating or dropping the indexes.

Visual Explainer

API

POST /tuning-service/v1/ve

Description

Generate a visual representation of access plans for a single query and return a job ID as an asynchronous request. The requester can retrieve access plan graphs (a URL that can be opened in browser) by using the GET *job_result* API.

Workload Access Path Advisor^{“1” on page 54}

API

POST /tuning-service/v1/wapa

Description

Generate access path recommendations for improving the performance of a workload and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET {*job_id*} *result* API.

Workload Access Path Comparison^{“1” on page 54}

API

POST /tuning-service/v1/wapc

Description

Explain the SQL statements for two workloads, compare their access plans, and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET `{job_id}` result API.

Workload Analytics Acceleration Advisor [“1” on page 54](#)**API**

POST /tuning-service/v1/waaa

Description

Generate a report that shows the eligible and ineligible statements for offloading to IBM Db2 Analytics Accelerator and quantifies the CPU savings that can be made by offloading.

Workload Candidate Acceleration Analyzer [“1” on page 54](#)**API**

POST /tuning-service/v1/wcaa

Description

Generate a report that shows the eligible and ineligible statements for offloading to IBM Db2 Analytics Accelerator and quantifies the CPU savings that can be made by offloading with user input candidate tables.

Workload Explain [“1” on page 54](#)**API**

POST /tuning-service/v1/wldexplain

Description

Explain the SQL statements in a workload and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET `{job_id}` result API.

Workload Index Advisor [“1” on page 54](#)**API**

POST /tuning-service/v1/wia

Description

Generate recommendations for creating, altering, or dropping indexes to improve the performance of a workload and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET `{job_id}` result API.

Workload Management**API**

POST /tuning-service/v1/workloads

Description

Create a workload.

API

GET /tuning-service/v1/workloads/{dbprofile_name_list}

Description

List the workloads of the connections that are specified in the `{dbprofile_name_list}`.

API

DELETE /tuning-service/v1/workloads/{workload_name_list}

Description

Delete specified workloads. Workloads that are specified in the `{workload_name_list}` will be deleted.

API

GET /tuning-service/v1/workloads/{workload_name}/statements

Description

Get the statement list of a workload.

API

DELETE /tuning-service/v1/workloads/{workload_name}/statements/
 {inst_id_list}

Description

Delete specified statements from the workload. Statements that are specified in the statement ID list will be deleted.

API

PUT /tuning-service/v1/workloads/{workload_name}

Description

Add statements to an existing workload.

API

GET /tuning-service/v1/workloads/{workload_name}/jobs/{job_component}

Description

Get a list of all the jobs that are associated with the specified workload.

API

POST /tuning-service/v1/workloads/{workload_name}/refine

Description

Refine an existing workload by defining one or more additional filter criteria to return a smaller, more focused result set.

API

GET /tuning-service/v1/workloads/{workload_name}/wldunexplainreason

Description

Return unexplained statements for a workload and the reasons why the statements were not explained.

Workload Query Environment Collector [“1” on page 54](#)**API**

POST /tuning-service/v1/wcqe

Description

Generate diagnostic information for a workload, which is needed by IBM Support, and return a job ID as an asynchronous request. The requester can retrieve this information by using the GET {job_id} result API.

Workload Query Rewrite Advisor [“1” on page 54](#)**API**

POST /tuning-service/v1/wqra

Description

Generate recommendations for rewriting the queries in a workload to improve its performance and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET {job_id} result API.

Workload Statistics Advisor [“1” on page 54](#)**API**

POST /tuning-service/v1/wsa

Description

Generate statistics data collection recommendations for a workload and return a job ID as an asynchronous request. The requester can retrieve the recommendations by using the GET {job_id} result API.

Workload Virtual Index Analyzer [“1” on page 54](#)**API**

POST /tuning-service/v1/wldwhatif

Description

Virtually test indexes to determine if the performance of a workload can be improved by creating or dropping the indexes.

Note:

1. This feature requires IBM Db2 Query Workload Tuner for z/OS 6.1. For more information, see the [Db2 Query Workload Tuner for z/OS Program Directory](#).

Chapter 17. Troubleshooting

If you encounter problems while using SQL Tuning Services, use the information in this topic to gather diagnostic information before contacting IBM Support.

1. Locate the log information in the *TMS_HOME/logs* directory for the period of time when problem occurred.

If you are able to re-create the problem, enable detailed logging by setting `logger.ibm.appenderRef.rollingFileInfo.ref=rollingFileDebug`. Then restart SQL Tuning Services and perform the actions that cause the problem to occur.

2. Download the log from server.
3. Download the `application.properties`, `tmserver_override.properties`, and `repodb_override.properties` files from the *TMS_HOME/Config* directory.
4. Open a support case at <https://www.ibm.com/mysupport/s/>.
5. Attach the log and properties files to the support case.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785 US*

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785 US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as shown below:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. (enter the year or years).

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered marks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at: <http://www.ibm.com/legal/copytrade.shtml>.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions:

Applicability: These terms and conditions are in addition to any terms of use for the IBM website.

Personal use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights: Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Privacy policy considerations

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at <http://www.ibm.com/privacy> and IBM’s Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at <http://www.ibm.com/software/info/product-privacy>.

Index

A

accessibility
 keyboard [v](#)
 shortcut keys [v](#)
API reference [47](#)
AT-TLS [27](#)

C

configuring
 AT-TLS [27](#)
 ports [17](#), [25](#)
 setup ID [19](#)
 SQL Tuning Services [21](#)
 z/OS environment [19](#)
cryptographic software, required [11](#)

D

Db2 for z/OS
 required version [11](#)
db2_authid_R
 changing password [45](#)
disability [v](#)

G

getting started [29](#)

I

installation prerequisites [11](#)
installation roadmap [9](#)
installation user IDs [13](#)
installing SQL Tuning Services [21](#)
Integrated Cryptographic Service Facility [11](#)
invoking SQL Tuning Services APIs [43](#)

J

Java SDK
 required version [11](#)

L

links
 non-IBM Web sites
 [58](#)

P

passwords
 changing for the repository database [45](#)
permissions

permissions (*continued*)
 for installing [13](#)
port settings [17](#), [25](#)
product overview [1](#)

R

reconfiguring SQL Tuning Services [37](#)
repository database
 upgrading [41](#)

S

shortcut keys
 keyboard [v](#)
system capacity [11](#)

T

troubleshooting [55](#)

U

upgrading) [41](#)
user IDs
 for installing and configuring [13](#)

W

What's new [7](#)

Z

z/OS
 required version [11](#)



Product Number: 5697-Q05

GC27-9831-00

