

Pregrado

Programa de
Ingeniería de
Sistemas

GESTIÓN DE DATOS E INFORMACIÓN II

Sesión 15

Tema:

Uso de SQL Server Profiler
y
el Optimizador del motor de
Base de Datos





Resultado de aprendizaje

Maneja herramientas administrativas e implementa seguridad de una base de datos SQL Server para una organización

Evidencia de aprendizaje

Los estudiantes demostrarán su capacidad para para configurar y utilizar SQL Server Profiler de manera efectiva, interpretar y analizar planes de ejecución, identificar y resolver problemas de rendimiento, y aplicar técnicas de optimización en consultas.

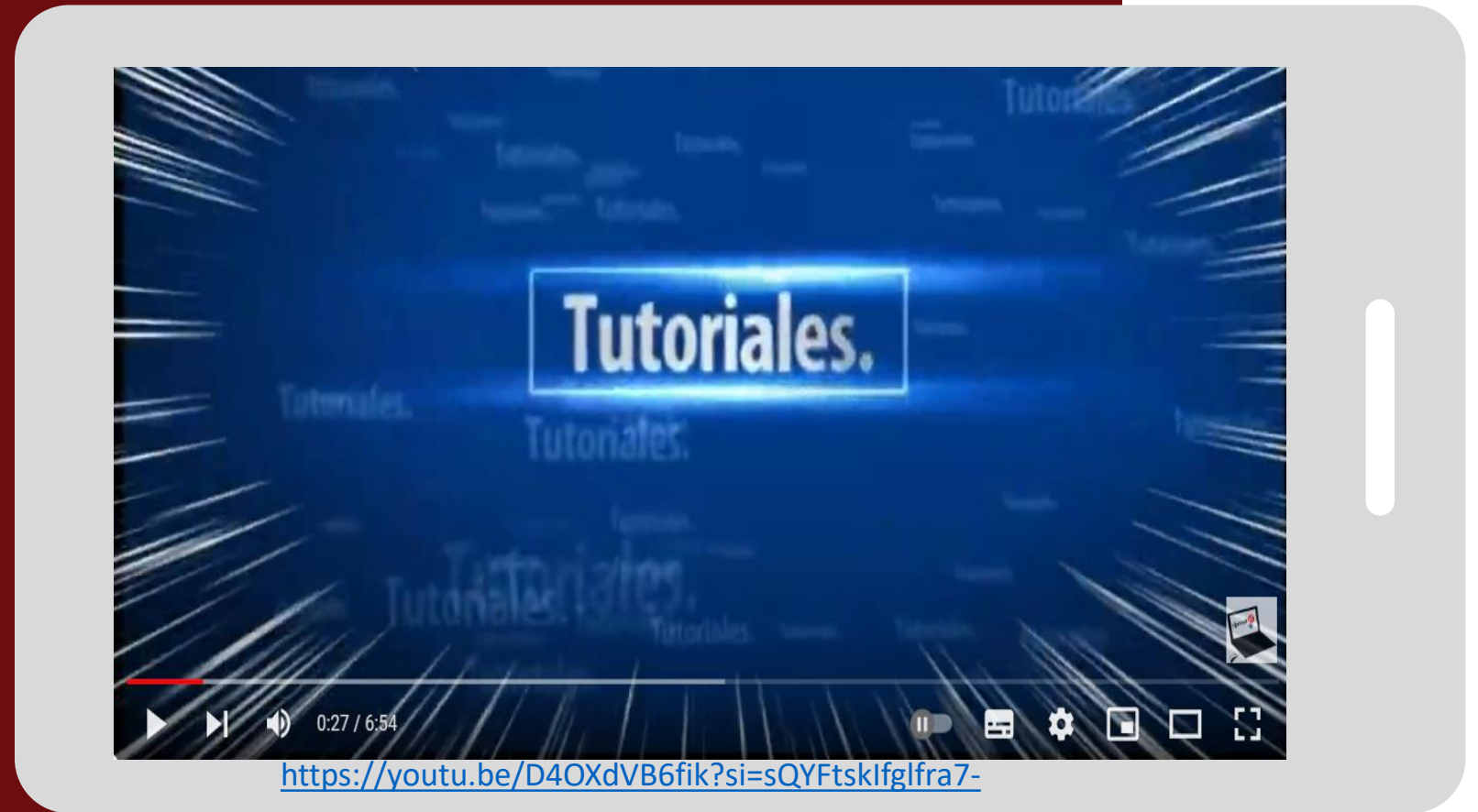


Contenido

Nombre del tema

- SQL Server Profiler
- Optimizador del motor de BD

**Revisa el
siguiente
video:**



Después de haber visualizado el video en la slide anterior, reflexionamos y respondemos las siguientes interrogantes:

01

¿Qué es el SQL Server Profiler?

02

¿Para qué sirve el SQL Server Profiler?

03

¿En qué momento recomendarías su uso del SQL Server Profiler?





Tema

SQL Server Profiler

SQL SERVER PROFILER

Es una herramienta de monitoreo y análisis que forma parte de Microsoft SQL Server. Su propósito principal es permitir a los administradores de bases de datos y desarrolladores rastrear, capturar y analizar eventos que ocurren en una instancia de SQL Server.

Algunos de los usos y funciones principales de SQL Server Profiler incluyen:

SQL SERVER PROFILER

Captura de eventos: Permite capturar eventos específicos que ocurren en una instancia de SQL Server. Estos eventos pueden incluir consultas SQL ejecutadas, conexiones a la base de datos, errores, entre otros.

Análisis de rendimiento: Facilita la identificación de cuellos de botella y problemas de rendimiento en consultas y transacciones. Puedes analizar el tiempo que tarda cada consulta en ejecutarse, identificar consultas lentas o costosas, y optimizar el rendimiento del sistema.

SQL SERVER PROFILER

Depuración y análisis de consultas: Permite analizar el comportamiento de las consultas SQL ejecutadas en el servidor, lo que resulta útil para depurar problemas y optimizar el código SQL.

Auditoría y seguimiento: Ofrece la capacidad de auditar actividades en la base de datos, lo que puede ser esencial para cumplir con requisitos de seguridad y cumplimiento normativo.

SQL SERVER PROFILER

Identificación de problemas de bloqueo: Ayuda a identificar problemas de bloqueo y detectar las consultas que podrían estar causando bloqueos en el sistema.

Monitoreo en tiempo real: Proporciona una vista en tiempo real de las actividades en la base de datos, lo que permite a los administradores supervisar y responder rápidamente a eventos críticos.

SQL SERVER PROFILER

EventClass	TextData	ApplicationName	NTUserName	LoginName	CPU	Reads	Writes	Duration	ClientProcessID	SPID	StartTime	EndTime	Binary	
ExistingConnection	-- network protocol: LPC set quoted...	Microsoft SQ...	Garridox	DESKTO...					17752	60	2020-09-08 12:27:09...			
ExistingConnection	-- network protocol: LPC set quoted...	Report Server	Garridox	DESKTO...					16000	63	2020-09-08 12:54:25...			
Audit Logout		RSManagement	Garridox	DESKTO...	0	53793	0	10014	6444	58	2020-09-08 12:54:23...	2020-09-08 12:54:33...		
RPC:Completed	exec sp_reset_connection	RSManagement	Garridox	DESKTO...	0	0	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
Audit Login	-- network protocol: LPC set quoted...	RSManagement	Garridox	DESKTO...					6444	58	2020-09-08 12:54:33...			
RPC:Completed	exec GetCurrentProductInfo	RSManagement	Garridox	DESKTO...	0	3	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	exec GetAllConfigurationInfo	RSManagement	Garridox	DESKTO...	0	3	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	exec GetAllConfigurationInfo	RSManagement	Garridox	DESKTO...	0	3	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	exec GetAllConfigurationInfo	RSManagement	Garridox	DESKTO...	0	3	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	exec GetAllConfigurationInfo	RSManagement	Garridox	DESKTO...	0	3	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	exec GetAnnouncedKey @InstallationID...	RSManagement	Garridox	DESKTO...	0	5	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
RPC:Completed	declare @p5 int set @p5=1 exec Ann...	RSManagement	Garridox	DESKTO...	0	14	0	0	6444	58	2020-09-08 12:54:33...	2020-09-08 12:54:33...	0X000	
Audit Logout		Report Server	Garridox	DESKTO...	46	20705	5	255997	16000	56	2020-09-08 12:50:19...	2020-09-08 12:54:35...		
RPC:Completed	exec sp_reset_connection	Report Server	Garridox	DESKTO...	0	0	0	0	16000	56	2020-09-08 12:54:35...	2020-09-08 12:54:35...	0X000	
Audit Login	-- network protocol: LPC set quoted...	Report Server	Garridox	DESKTO...					16000	56	2020-09-08 12:54:35...			
RPC:Completed	exec PollEventsForRSProcess @NumberD...	Report Server	Garridox	DESKTO...	0	0	0	0	16000	56	2020-09-08 12:54:35...	2020-09-08 12:54:35...	0X000	
SQL:BatchStarting		...	Report Server	Garridox	DESKTO...				16000	56	2020-09-08 12:54:35...			
SQL:BatchCompleted		...	Report Server	Garridox	DESKTO...	0	0	0	0	16000	56	2020-09-08 12:54:35...	2020-09-08 12:54:35...	

```
declare @BatchID uniqueidentifier
set @BatchID = newid()
UPDATE [Notifications] WITH (TABLOCKX)
SET [BatchID] = @BatchID,
    [ProcessStart] = GETUTCDATE(),
    [ProcessHeartbeat] = GETUTCDATE()
FROM (
    SELECT TOP 4 [NotificationID] FROM [Notifications] WITH (TABLOCKX) WHERE ProcessStart is NULL and
        (ProcessAfter is NULL or ProcessAfter < GETUTCDATE()) ORDER BY [NotificationEntered]
) AS t1
WHERE [Notifications].[NotificationID] = t1.[NotificationID]

select top 4
```

Trace is running. Ln 23, Col 1 Rows: 23

SQL SERVER PROFILER

Es importante tener en cuenta que el uso de **SQL Server Profiler** debe hacerse con precaución, ya que su uso excesivo o incorrecto puede afectar el rendimiento del servidor. Además, en versiones más recientes de SQL Server, se recomienda considerar el uso extendido de **Extended Events** como alternativa, ya que proporciona funcionalidades similares con una menor carga en el rendimiento del sistema.



Tema

Optimizador del motor de
Base de Datos

OPTIMIZADOR DEL MOTOR DE LA BASE DE DATOS

Se refiere generalmente al componente encargado de la optimización de consultas y planes de ejecución en un sistema de gestión de bases de datos (DBMS). Este optimizador es una parte esencial del motor de la base de datos y su función principal es determinar la mejor manera de ejecutar las consultas para obtener resultados eficientes.

A continuación, se detallan los elementos clave relacionados con el optimizador del motor de la base de datos:

OPTIMIZADOR DEL MOTOR DE LA BASE DE DATOS

Análisis de consultas: El optimizador analiza las consultas SQL enviadas al DBMS para entender la operación solicitada y las relaciones entre las tablas involucradas.

Generación de planes de ejecución: Considera diversas estrategias y operaciones para ejecutar la consulta, como métodos de acceso a los datos, operaciones de filtrado, unión y ordenación.

OPTIMIZADOR DEL MOTOR DE LA BASE DE DATOS

Estimación de costos: Evalúa los costos asociados con cada plan de ejecución propuesto, tomando en cuenta factores como la complejidad de las operaciones y la distribución de datos en la base de datos.

Selección del mejor plan: Opta por el plan de ejecución con el menor costo estimado, lo que se espera que resulte en una ejecución más eficiente de la consulta.

OPTIMIZADOR DEL MOTOR DE LA BASE DE DATOS

El optimizador utiliza información estadística, índices y otros metadatos disponibles para tomar decisiones informadas sobre la mejor manera de acceder y manipular los datos. Este proceso es transparente para los usuarios y se lleva a cabo internamente en el motor de la base de datos.

Un buen optimizador es crucial para el rendimiento general de un sistema de gestión de bases de datos, ya que puede afectar significativamente la velocidad de ejecución de las consultas.

OPTIMIZADOR DEL MOTOR DE LA BASE DE DATOS

Los **desarrolladores y administradores de bases de datos** pueden influir en el rendimiento del optimizador mediante la escritura de consultas eficientes, la actualización de estadísticas y la creación adecuada de índices. Es importante señalar que diferentes sistemas de gestión de bases de datos pueden tener enfoques ligeramente diferentes en sus optimizadores, pero todos comparten el objetivo común de mejorar el rendimiento de las consultas.



Autoevaluación

Sesión 15



Pregunta 1

¿Cuál es el propósito principal de SQL Server Profiler?

- ☐ Crear bases de datos en SQL Server.
- ☐ Rastrear y analizar eventos en una instancia de SQL Server.
- ☐ Optimizar consultas mediante el uso de índices.
- ☐ Ninguna.

Pregunta 2

¿Qué eventos pueden ser rastreados utilizando SQL Server Profiler?

- ☐ Solo consultas SELECT.
- ☐ Eventos de inicio de sesión y consultas DML
- ☐ Solamente eventos de bloqueo.
- ☐ Ninguna

Pregunta 3

¿Cuál es el propósito principal del Optimizador del Motor de Base de Datos en SQL Server?

- ☐ Realizar copias de seguridad de la base de datos.
- ☐ Interpretar y generar planes de ejecución eficientes para consultas.
- ☐ Configurar políticas de contraseña para usuarios.
- ☐ Ninguna

Pregunta 4

¿Cuál es un indicador común de problemas de rendimiento que se puede identificar con SQL Server Profiler?

- ☐ Índices no utilizados.
- ☐ Largas duraciones de consultas y alto uso de recursos.
- ☐ Eventos de inicio de sesión frecuentes.
- ☐ Ninguna

Autoevaluación
¡Vamos por más logros!

¡Felicitaciones!
Ha concluido la autoevaluación



Conclusiones

SQL Server Profiler se presenta como una herramienta esencial para rastrear y analizar eventos en una instancia de SQL Server. Permite a los administradores y desarrolladores obtener información detallada sobre las actividades en tiempo real, identificando patrones de comportamiento, problemas de rendimiento y eventos críticos para la administración y mejora del sistema.

La capacidad de configurar trazas específicas mediante **SQL Server Profiler** ofrece flexibilidad para seleccionar y monitorear eventos relevantes. Esto incluye desde eventos de inicio de sesión y consultas DML hasta bloqueos y otros eventos de sistema, permitiendo un análisis detallado de la actividad en la base de datos.

El **Optimizador del Motor de Base de Datos** desempeña un papel crucial en la eficiencia del sistema al interpretar y generar planes de ejecución eficientes para consultas. Al entender cómo el optimizador toma decisiones, los desarrolladores pueden ajustar sus consultas para mejorar el rendimiento y optimizar la utilización de recursos.

SQL Server Profiler se destaca en la identificación de problemas de rendimiento mediante la captura de eventos como largas duraciones de consultas y alto uso de recursos. Esta capacidad facilita la detección temprana de problemas potenciales, permitiendo a los administradores tomar medidas correctivas antes de que afecten significativamente el rendimiento del sistema.



Aplicando lo aprendido:

Desarrollar la Guía de Laboratorio N°15

Referencias

CAPACHO, José y Wilson NIETO. Diseño de Bases de Datos [en línea]. Barranquilla: Universidad del Norte, 2017. ISBN 9789587418255. Disponible en: <https://search.ebscohost.com/login.aspx?direct=true&db=e000xww&AN=1690049&lang=es&site=ehost-live>

WANUMEN Luis, RIVAS Edwin, Mosquera Darín. Bases de datos en SQL Server [en línea]. Bogotá: Ecoe Ediciones, 2017. ISBN 9789587715705. Disponible en: <https://www.digitaliapublishing.com/a/66605>

HUESO Luis. Bases de datos [en línea]. Madrid: Rama Editorial, 2014. ISBN 9788499641577. Disponible en: <https://www.digitaliapublishing.com/a/109943>

PRIETO, Rafael. SGBD e instalación: administración de bases de datos (UF1469) [en línea]. Antequera : IC Editorial. ISBN 9788416433360. Disponible en: <https://www.digitaliapublishing.com/a/86830>





Pregrado