

# Oracle Performance I

Introducción al análisis de rendimiento de Oracle

# ¡BIENVENIDOS!

## Soy Enrique Gallardo

Soy Administrador de Bases de Datos Oracle



<https://www.linkedin.com/in/enriquegallardo/>

# Sobre mi

- Más de 12 años como DBA
- DBA líder en proyectos internacionales
- OCP en Oracle 11g y 12c
- OCE en RAC & Grid
- Especializado en Alta Disponibilidad
- Trabajando con Oracle desde la versión 8i hasta 20c



# Objetivos

Durante este taller veremos algunas nociones sobre el análisis de rendimiento en Oracle, como veremos es un tema bastante amplio y que requiere amplios conocimientos para resolución de problemas complejos, intentaremos realizar una aproximación a las herramientas más usadas y problemas más comunes, se proporcionará información y herramientas para todo aquel que quiera ahondar en la materia.

# ÍNDICE

- Introducción al análisis de rendimiento
- Revisión reactiva/proactiva
- Revisión de sesiones – V\$SESSION y otras vistas útiles
- Fases del procesamiento de una sentencia SQL

# Introducción al análisis de rendimiento

Una de las ramas en la administración de bases de datos es el análisis del rendimiento, en este taller daremos una introducción al análisis de rendimiento y los aspectos a tener en cuenta, dentro del análisis de rendimiento se crearán más talleres sobre características concretas, por lo que este taller servirá como introducción para el análisis del rendimiento.



# Introducción al análisis de rendimiento

Como veréis, el análisis de rendimiento es un campo complejo donde entran en juego muchos conceptos avanzados, por lo que se requiere amplios conocimientos tanto de los diferentes componentes de la instancia de base de datos como de los diferentes componentes de cualquier sistema informático (almacenamiento, red, memoria, etc.).

# Formación recomendada

Se recomienda previamente haber realizado las siguientes formaciones disponibles en la plataforma:

- [Arquitectura básica de una BBDD](#)
- [Creación de una instancia de BBDD Oracle 12c desde cero](#)

De esta manera creemos que se puede sacar más provecho a la formación propuesta y entender mejor los conceptos.

# Gestión reactiva y proactiva

Dependiendo del entorno de trabajo en el que estemos realizaremos una gestión reactiva o proactiva del rendimiento, aunque es habitual realizar una mezcla de ambas:

**Gestión reactiva**, es la que se produce a consecuencia de un problema, es el método más tradicional, se actúa cuando se han producido los problemas y normalmente viene acompañada de plazos muy cortos, problemas graves asociados (caída del servicio, degradación en el rendimiento, eternización de consultas, etc.) y con poco margen de actuación.

# Gestión reactiva y proactiva

**Gestión proactiva**, es cuando se analizan y se ofrecen soluciones o recomendaciones de manera continua y antes de que se produzcan problemas importantes que afecten al rendimiento o estabilidad del sistema.

Lo ideal es dedicar el mayor tiempo posible a la gestión proactiva, de manera que podamos anticipar posibles problemas de rendimiento y ofrecer modificaciones y mejoras que optimicen el uso de recursos.

# Tipos de análisis de rendimiento

Podemos separar el análisis de rendimiento en dos grandes ramas muy diferenciadas:

- **Mejora rendimiento SQL (SQL performance tuning)**

En este caso nos fijamos en una transacción unitaria en la que detectamos que existen problemas de rendimiento, la analizamos e intentamos detectar problemas y solucionarlos en la medida que sea posible.

- **Mejora rendimiento instancia BD (Oracle performance tuning)**

En este caso nos fijamos en el rendimiento global de la instancia de base de datos, donde intentamos analizar todos los componentes y ver si existe algún cuello de botella (bottleneck).

# SQL performance tuning

Cuando analizamos el rendimiento de consultas tenemos determinadas herramientas para el análisis de las mismas, la principal es el plan de ejecución:

- Plan de ejecución (EXPLAIN PLAN), el plan de ejecución nos indica qué camino ha elegido el optimizador de la base de datos para resolver la consulta que hemos realizado, dependiendo de muchos factores, el optimizador describirá el plan que considera óptimo para acceder a los datos que nos tiene que mostrar

# SQL performance tuning

Existen herramientas adicionales que nos dan más información:

- Active Sesión History – Información sobre la ejecución histórica de una consulta
- Tabla DBA\_TAB\_STATS\_HISTORY – Información sobre las estadísticas de una tabla
- Asesor de ajustes SQL

# Oracle performance tuning

En este caso enfocamos la mejora hacia el funcionamiento global de la instancia de base de datos, por lo que las soluciones no tienen que ir enfocadas a mejorar consultas u objetos concretos:

- Análisis de rendimiento de instancia e BD.
- Análisis de rendimiento de sistema operativo y almacenamiento.
- Análisis de rendimiento de procesos batch.
- Control y rendimiento del backup (RMAN).
- Parametrización de la BD.
- Identificación y resolución de eventos de espera.
- Identificación de procesos pesados.
- Recomendaciones e implantaciones de nuevas configuraciones para la mejora del rendimiento.

# Revisión de sesiones – V\$SESSION

Una de las primeras vistas que usamos en el análisis del rendimiento es la vista V\$SESSION (GV\$SESSION para entornos RAC).

En esta vista encontramos información sobre las sesiones actuales en la instancia de base de datos y nos puede ser muy útil para identificar problemas sencillos.

Vamos a revisar los campos que encontramos en esta vista del sistema:

<https://docs.oracle.com/en/database/oracle/oracle-database/19/refrn/V-SESSION.html#GUID-28E2DC75-E157-4C0A-94AB-117C205789B9>

# Revisión de sesiones – SQL Developer

Veremos ahora la vista de sesiones desde el SQL Developer, desde donde además podemos realizar muchas acciones sobre las sesiones que seleccionemos.

Cómo podemos ver, en esta vista obtenemos una visión clara de lo que se está rodando en el sistema, del número de sesiones activas, del programa usado para la conexión e incluso información sobre el host desde el que se ejecuta, hora de conexión, etc.

# Revisión de sesiones – Detectar bloqueos

Realizan uso de esta vista podemos por ejemplo revisar posible bloqueos existentes y matar aquellas sesiones que estén haciendo bloqueos no esperados.

Veremos en una práctica como provocar diferentes bloqueos y la información que nos ofrecerá la vista V\$SESSION de los mismos.

Existe multitud de vistas en las que nos podemos apoyar para obtener información del estado actual de la instancia de base de datos.

# Revisión de sesiones – Tipos de sesiones

Como podemos ver, la vista V\$SESSION es bastante útil y potente en la detección y resolución de problemas y saber realizar uso de la misma requiere práctica y conocimientos de base de datos, pero con unas pocas nociones podemos ser capaces de consultar las sesiones existentes e identificar algunos de los problemas más comunes en la misma.

Podemos clasificar las sesiones de base de datos en dos grandes grupos divididas por el campo TYPE:

- Sesiones en background, son aquellos procesos internos de la base de datos ejecutados por el propio sistema (arch, ckpt, dbwr, etc).
- Sesiones de usuario, son aquellas sesiones realizadas por usuarios de la base de datos.

# Revisión de sesiones – Vistas adicionales

Adicionalmente existen otras vistas dinámicas útiles a la hora de revisar problemas existentes, vamos a listar algunas de ellas:

- **V\$SGASTAT** - Tamaño de todas las estructuras de memoria
- **V\$STATNAME** - Estadísticas del V\$SESSTAT
- **V\$SYSSTAT** - Estadísticas del uso de cpu para todas las sesiones activas
- **V\$SESSTAT** - Estadísticas de las sesiones activas
- **V\$WAITSTAT** - Refleja la contención en términos del número de esperas en cuatro tipos de bloques de rollback

# Revisión de sesiones – Identificación de bloqueos

A la hora de identificar posibles bloqueos podemos usar las siguientes vistas

- **DBA\_LOCKS** - información sobre el tipo de bloqueo y las sesiones
- **DBA\_BLOCKERS** - ID del bloqueo
- **DBA\_WAITERS** - Información adicional sobre la sesión en espera

Realizaremos una práctica en la que provocaremos un bloqueo y veremos la información que tenemos en las diferentes vistas.

# Fases del procesamiento de una sentencia SQL

Cuando ejecutamos una sentencia SQL sobre un motor de base de datos, pasa por cuatro fases principales (parsing, binding, ejecución y fetching):

**1. Parsing**, durante la fase de parseo el servidor Oracle realiza las siguientes acciones:

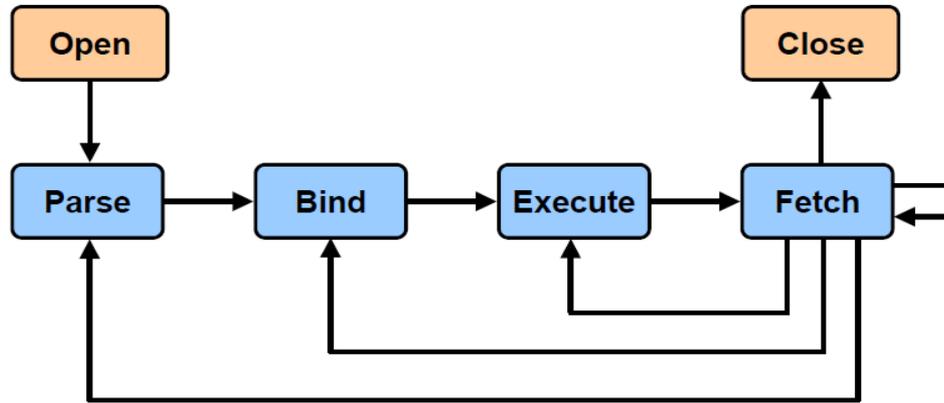
- Busca la sentencia en la memoria compartida (Shared pool).
- Chequea la sintaxis y gramática de la sentencia.
- Chequea la semántica, asegurándose que los objetos a los que se intenta acceder existen y que el usuario que lo ejecuta tiene los permisos necesarios para acceder.
- Transforma Vistas a lenguaje SQL simplificado
- **Determina y almacena el plan de ejecución o utiliza uno existente.**

# Fases del procesamiento de una sentencia SQL

2. Fase de **binding**, en esta fase el servidor identifica las variables de la sentencia SQL y les asigna valores a cada variable, habitualmente se recomienda el uso de Bind Variables para facilitar la re-ejecución de sentencias similares, ya que en estos casos nos ahorraríamos la fase de parseo, que a veces puede ser superior a la fase de ejecución.
3. Fase de **ejecución**, en esta fase se ejecuta la sentencia siguiendo el plan de ejecución y se ordenan los valores en caso de ser necesario
4. Fase de **fetching**, en esta fase se devuelven los registros de una sentencia SELECT, en cada iteración (o fetch) se devuelven múltiples registros

# Fases del procesamiento de una sentencia SQL

## SQL Statement Processing Phases



# Enlaces útiles

Documentación oficial sobre rendimiento

<https://docs.oracle.com/en/database/oracle/oracle-database/19/performance.html>

Hands-on labs

<https://www.oracle.com/virtualization/technologies/vm/downloads/hol-oraclevm-downloads.html>

White Paper sobre rendimiento

<https://www.oracle.com/technetwork/oem/db-mgmt/wp-self-managing-database-3406478.pdf>