



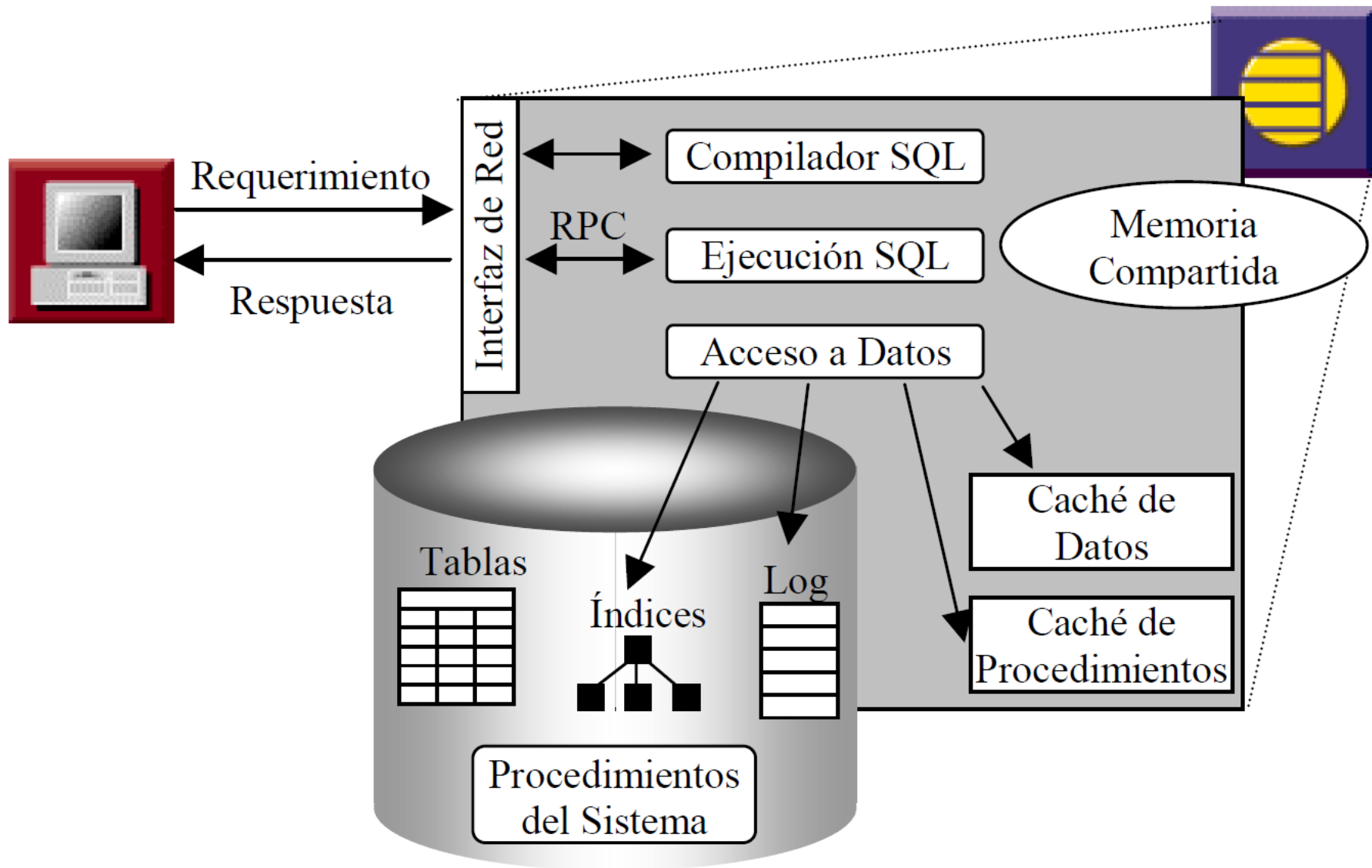
Técnicas y herramientas de tuning para servidores de bases de datos SQL server SYBASE

M. en A. Armando Vega.



Objetivo

Dar a conocer algunas herramientas y técnicas que permitan maximizar el desempeño de los servidores de base de datos SQL server Sybase desde la versión 15 incluyendo la versión 16.x





Niveles da Afinamiento

- **Capa da aplicación** – La mayoría de ganancias en rendimiento vienen del afinamiento de consultas, basado en el buen diseño de base de datos.
- **Capa de base de datos** – Las aplicaciones comparten recursos a nivel de base de datos, incluyendo discos, el log de transacciones y el caché de datos (memoria).
- **Capa de servidor** – A nivel de servidor hay muchos recursos compartidos, incluyendo los cachés de datos y de procedimientos almacenados, candados, y CPUs.



- **Capa de dispositivos** – El(los) disco(s) que almacenan sus datos.
- **Capa de red** – Los servicios de red que permiten a los usuarios conectarse con ASE.
- **Capa de hardware** – Las CPUs disponibles.
- **Capa de sistema operativo** – Idealmente, ASE es la aplicación principal en la máquina y debe tan solo compartir CPU, memoria y otros recursos con el sistema operativo y otro software de Sybase como el Backup Server o el XP Server.



Nivel de aplicación

- Usar procesamiento remoto o replicado para mover el soporte a decisiones fuera de la máquina OLTP.
- Usar procedimientos almacenados para reducir el tiempo de compilación y el uso de red.
- Usar el nivel mínimo de bloqueo que se ajuste a las necesidades de su aplicación.



Capa de base de datos

- Usar umbrales de log de transacciones para automatizar el vaciado de los logs y evitar quedarse sin espacio.
- Usar umbrales para el monitoreo de espacio en los segmentos de datos.
- Usar particiones para agilizar el acceso a los datos.
- Ubicar objetos sobre diferentes dispositivos físicos para evitar la contención sobre los discos y tomar ventaja del paralelismo de las operaciones de lectura/escritura.
- Definir cachés para proporcionar alta disponibilidad a las tablas e índices críticos.



Capa de servidor

- Afinar el uso de memoria.
- Decidir entre procesamiento en el cliente vs. procesamiento en el servidor – ¿puede alguna parte del procesamiento llevarse a cabo en el cliente?
- Configurar el tamaño de los cachés y el tamaño de los bloques de lectura/escritura.
- Agregar más CPU.
- Programar trabajos en lote y generación de reportes fuera de horas pico.
- Determinar si es posible mover aplicaciones DSS a otro ASE.



Capa de hardware

- Agregar más CPU para responder a la carga.
- Seguir los lineamientos de diseño de aplicaciones en entornos SMP para reducir la contención.
- Configurar múltiples cachés de datos (cachés con nombre).



CASOS PRÁCTICOS



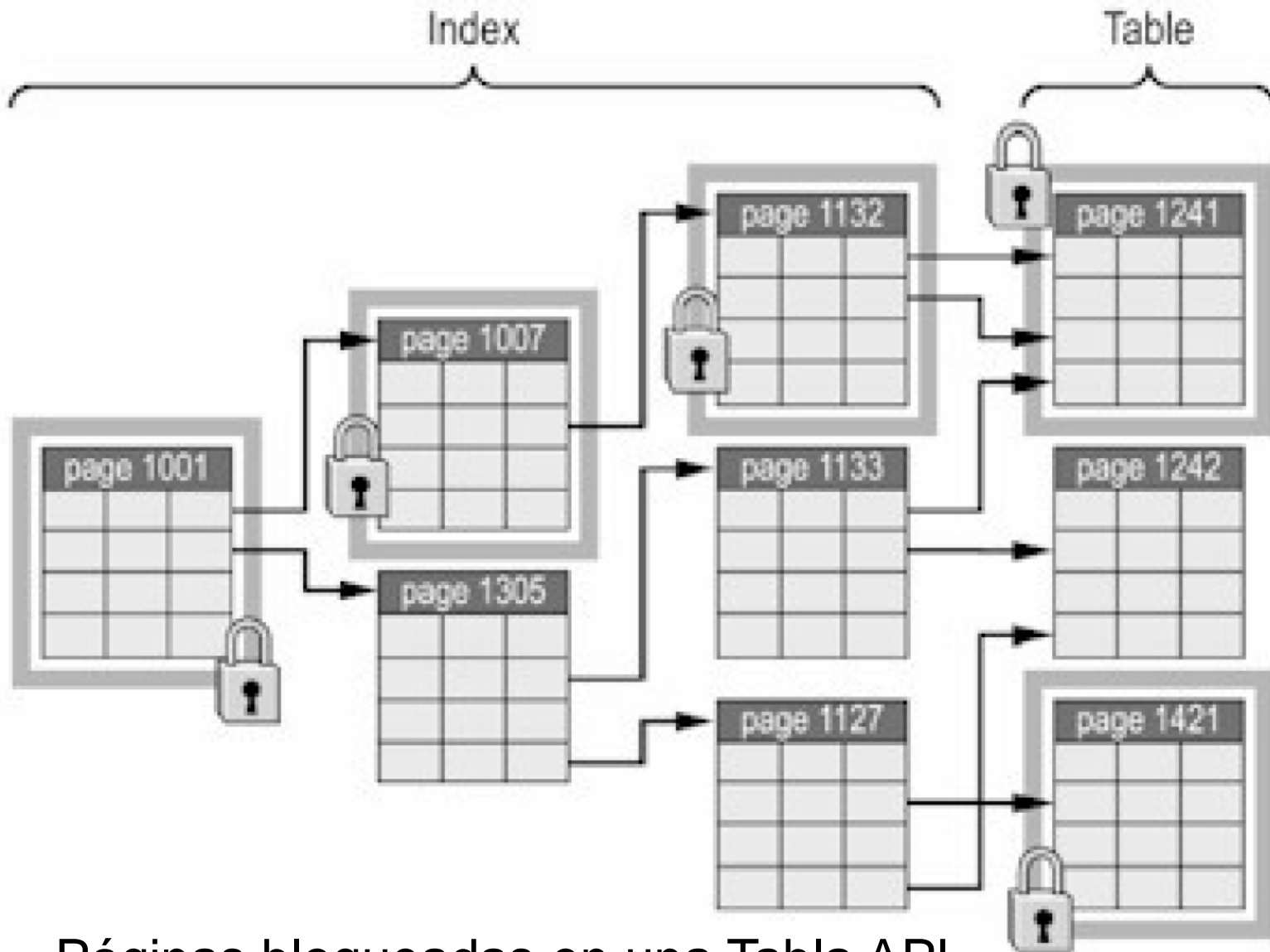
La fragmentación de datos

- La fragmentación de datos (también conocida como un mal agrupamiento o *clustering* de las páginas de datos) se presenta debido a la actividad de operaciones de modificación (**insert**, **update**, **delete**) en las tablas. La fragmentación representa un uso ineficaz del espacio, y un alto nivel de fragmentación significa también que la eficacia de las operaciones de lectura/escritura será pobre, ya que se requerirán operaciones adicionales de lectura/escritura para tener acceso a los datos.

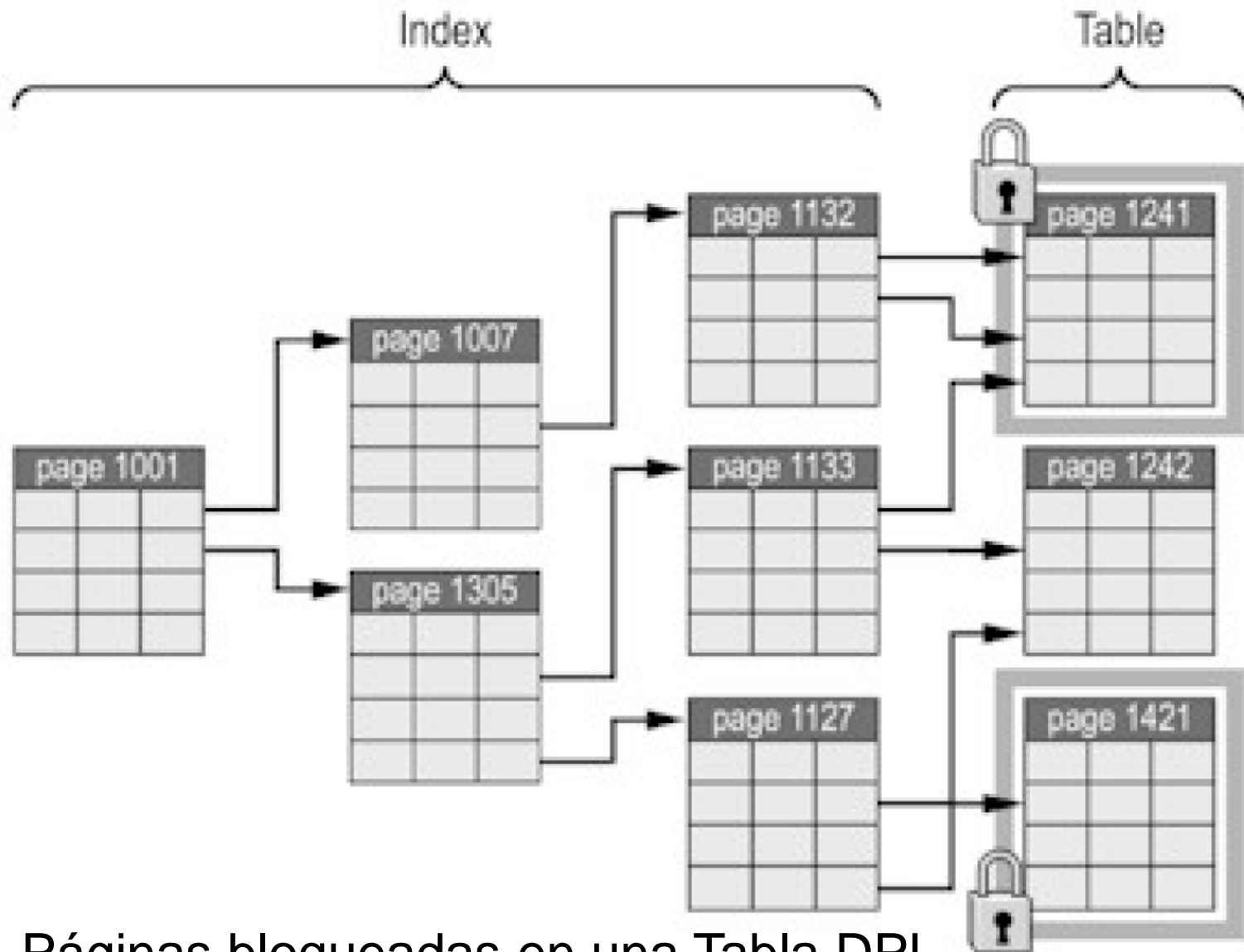


Existen 3 tipos de bloqueo para ASE

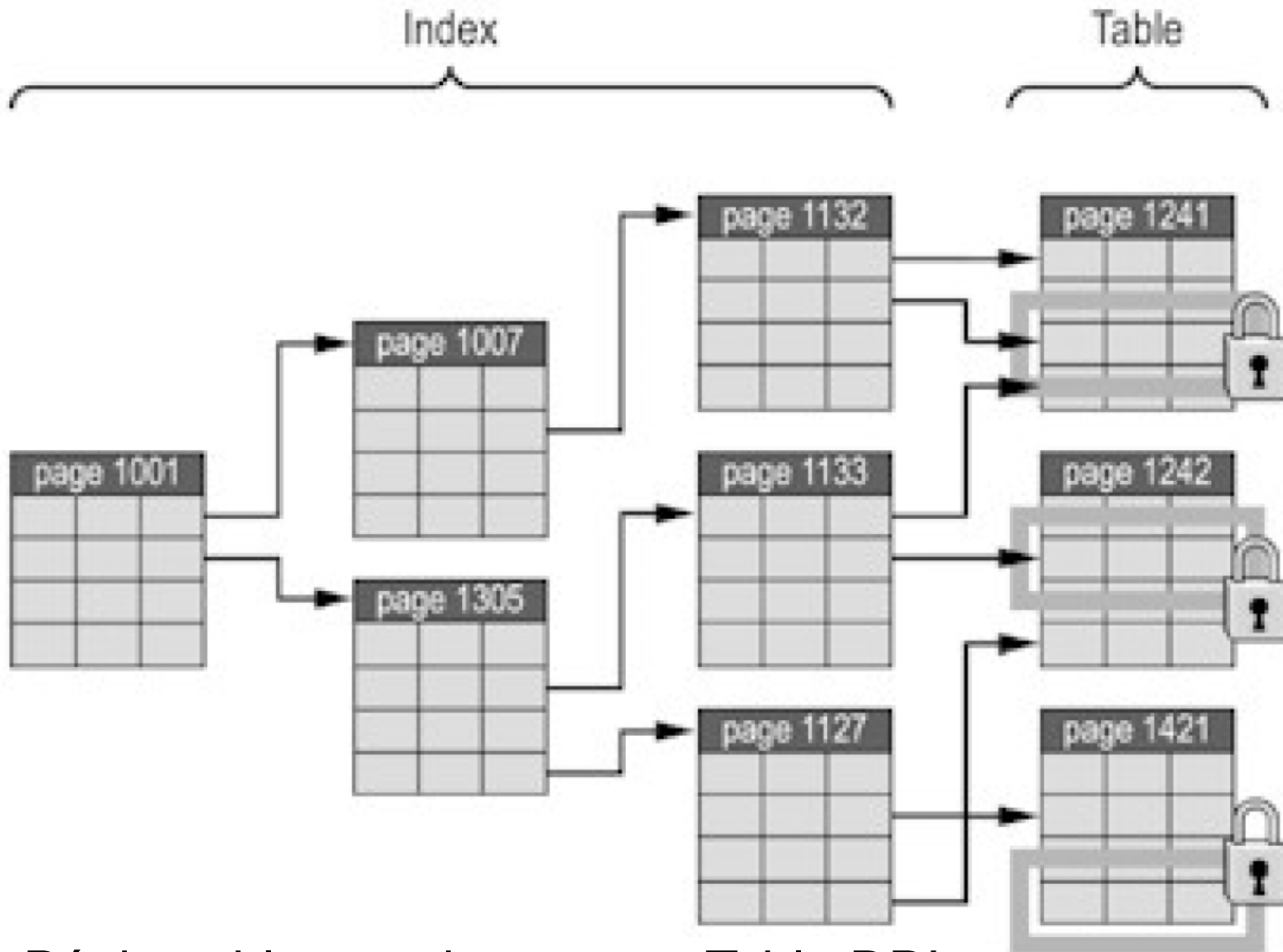
- Allpages (APL)
- Datapages (DPL)
- Datarows (DRL)



Páginas bloqueadas en una Tabla APL



Páginas bloqueadas en una Tabla DPL



Páginas bloqueadas en una Tabla DRL



optdiag & reorg rebuild

- El programa **optdiag** permite visualizar las estadísticas para cada una de las tablas e índices de ASE.
- El comando **reorg** puede ser usado para recolección de basura y defragmentación de tablas



optdiag statistics pubs2..titles -Usa -Ppasswd -o titles.opt

Statistics for table: "titles"

Data page count: 662

Empty data page count: 10

Data row count: 4986.0000000000000000

Forwarded row count: 18.0000000000000000

Deleted row count: 87.0000000000000000

Data page CR count: 86.0000000000000000

OAM + allocation page count:5

First extent data pages: 3

Data row size: 238.8634175691937287

Derived statistics:

Data page cluster ratio: 1.0000000000000000

Space utilization: 0.9035689867593

Large I/O efficiency: 1.0000000000000000



Este es un ejemplo del uso del comando **reorg:**

```
$ isql -Usa -P -Sase125_prd -Dpubs3
```

```
1> reorg rebuild titles
```

```
2> go
```

```
Beginning REORG REBUILD of 'titles'.
```

```
There are approximately 4670 pages to be  
processed.
```

```
REORG REBUILD of 'titles' completed.
```



Aumento del Tráfico de Red Durante la Ejecución de Procedimientos Almacenados

- Un procedimiento almacenado batch (no retorna resultados al cliente, solo un valor de estado al finalizar) se ejecuta en aproximadamente 2 a 4 minutos si es disparado desde la consola del servidor donde está corriendo Adaptive Server Enterprise. Al dispararlo desde una PC, conectado al servidor a través de una WAN de bajo desempeño, su ejecución se demora al rededor de 2 horas o más



- Se identificó que la causa del incremento en el tráfico en la red corresponde al hecho de que Adaptive Server Enterprise envía al cliente unos paquetes llamados paquetes DONEINPROC; estos son enviados, por defecto, después de cada uno de los comandos que hacen parte del procedimiento almacenado.
- Soluciones
 - Una posibilidad es incluir un comando **SET NOCOUNT ON** al comienzo del procedimiento almacenado

Para apagar los mensajes, ejecute el comando:

 - **dbcc tune (doneinproc, 0)**
 - Para activar los mensajes:
 - **dbcc tune (doneinproc, 1)**



- **Mantenga copias de las tablas de datos y scripts de creación de objetos**
 - Ddlgen: Una -herramienta basada en Java que crea definiciones para los objetos de nivel de servidor - y base de datos-en Adaptive Server.
 - Ejemplo: Genera DDL para una base de datos llamada pubs2 en un Máquina llamada HARBOR usando puerto 1955

```
ddlgen -Uroy -Proy123 -SHARBOR:1955 -TDB -Npubs2  
select "bcp pubs2.."+name+" -Usa -SSYBASE -c " from  
sysobjects where type="U"
```



Con bcp para respaldar información de una tabla

```
select "bcp pubs2.."+name+" -Usa -SSYBASE -c " from sysobjects where type="U"
```

```
bcp pubs2..authors -Usa -SSYBASE -c  
bcp pubs2..publishers -Usa -SSYBASE -c  
bcp pubs2..roysched -Usa -SSYBASE -c  
bcp pubs2..sales -Usa -SSYBASE -c  
bcp pubs2..salesdetail -Usa -SSYBASE -c  
bcp pubs2..titleauthor -Usa -SSYBASE -c  
bcp pubs2..titles -Usa -SSYBASE -c  
bcp pubs2..stores -Usa -SSYBASE -c
```



Monitoreo del SQL en Ejecución

- Usted debe configurar el parámetro '*max SQL text monitored*' para permitir que Adaptive Server Enterprise guarde en memoria compartida las sentencias SQL en ejecución. Para esto use el procedimiento **sp_configure** así:

```
sp_configure 'max SQL text monitored', bytes_por_conexión
```

```
sp_configure 'max SQL text monitored', 2048
```

```
go
```



sp_who

go

fid spid status loginame origname hostname

dbname cmd

0 8 send sleep pgomez pgomez solaris2

pubs3 SELECT

0

exec sp_showplan 8, @batch output, @context output, @statement output

go

QUERY PLAN FOR STATEMENT 5 (at line 9).

STEP 1

The type of query is SELECT.

FROM TABLE

authors

Nested iteration.

Index : aunmind

Forward scan.

Positioning at index start.

Index contains all needed columns. Base table will not be read.

Using I/O Size 2 Kbytes for index leaf pages.

With LRU Buffer Replacement Strategy for index leaf pages.

(return status = 0)

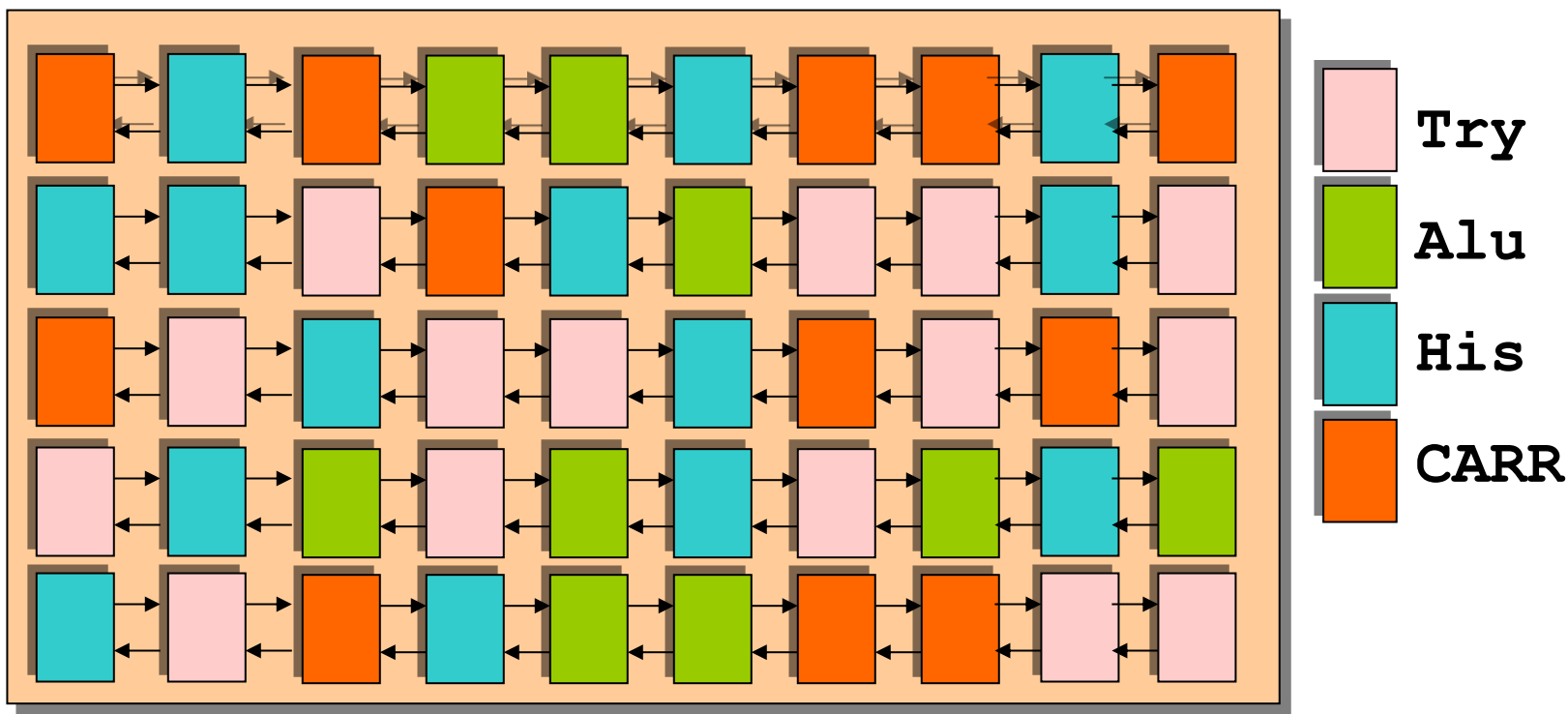
Return parameters:

1948 0 5



Configuración de memoria

RAM del servidor de base de datos

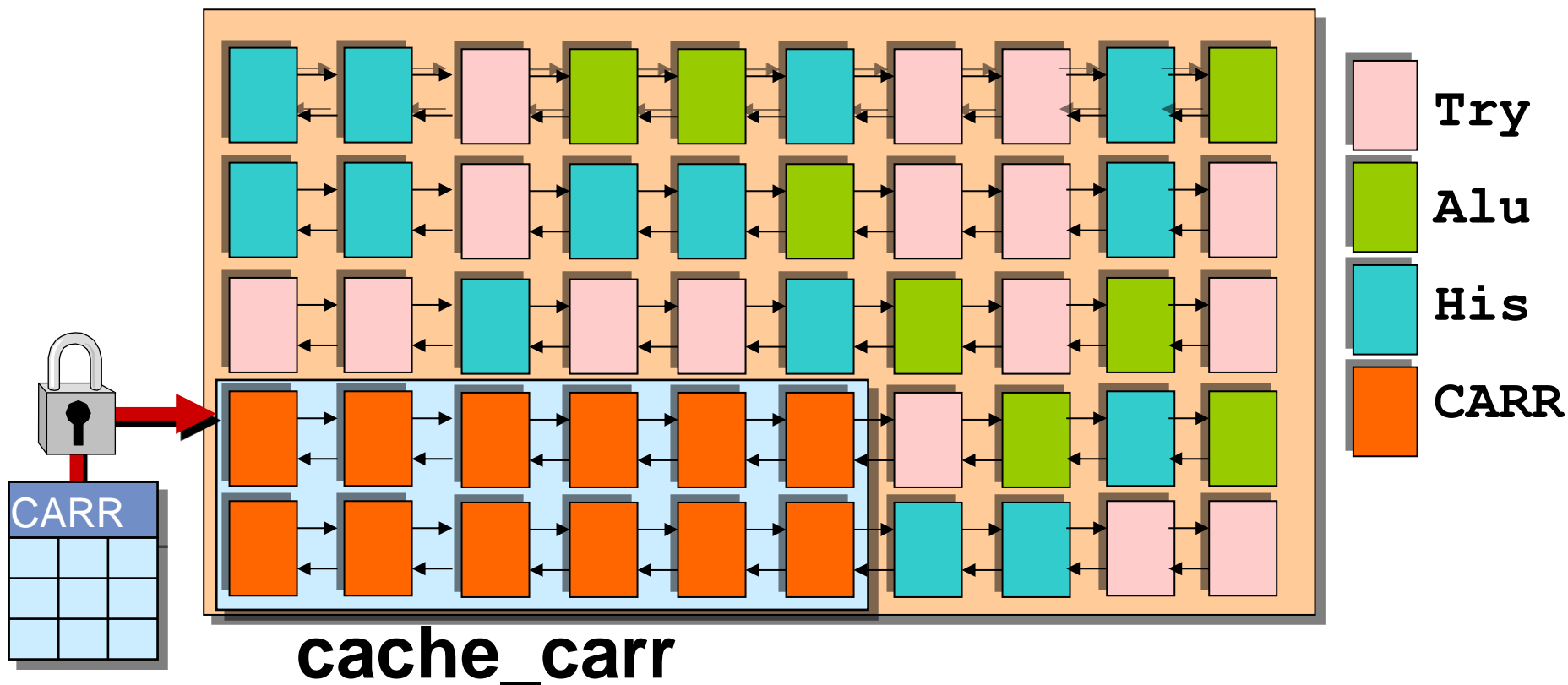


De forma natural TODOS los querys se disputan la memoria del servidor



Configuración de memoria

RAM del servidor de base de datos





Creación de cache_carr

```
1> sp_cacheconfig "cache_carr", "8M"
```

```
2> go
```

```
1> sp_helpcache
```

```
2> go
```

Cache Name	Config Size	Run Size	Overhead
cache_carr	8.00 Mb	8.00 Mb	0.44 Mb
default data cache	0.00 Mb	237.20 Mb	24.03 Mb

```
1> sp_bindcache "cache_carr", "pubs2", "carr"
```

```
2> go
```



**MUCHAS
GRACIAS!!!**

avega@unam.mx