

ANEXO II

Aplicación Lectores – Escritores

INDICE

Objetivo	2
Tecnología Utilizada.....	2
Diagrama de Clases	2
Configuración de la Aplicación.....	6
SQL Server 2000	7
SQL Server 2005	7
DB2.....	7

Objetivo

El objetivo es crear una aplicación para mostrar y comparar como se resuelven en los distintos productos comerciales los problemas de concurrencia entre lectores y escritores en una base de datos.

La idea principal de esta aplicación es visualizar la concurrencia que permiten los distintos modelos que utilizan las bases de datos en el mercado actual.

De ésta manera, se pueden comparar los modelos de concurrencia manejados por **Bloqueos** como el que implementa SQL Server y DB2 y el modelo de **Múltiples Versiones** como el que implementa Oracle, PostgreSQL y MySQL (en este último si se usan tablas tipo InnoDB).

Tecnología Utilizada

La aplicación se llevó a cabo utilizando lenguaje Java, y conectándose con los diferentes DBMS vía jdbc (de tipo 4, es decir, driver codificado completamente en Java). Las pruebas se llevaron a cabo con: ORACLE 9i, SQL Server 2000 y 2005, DB2 v7 y MySQL v5.

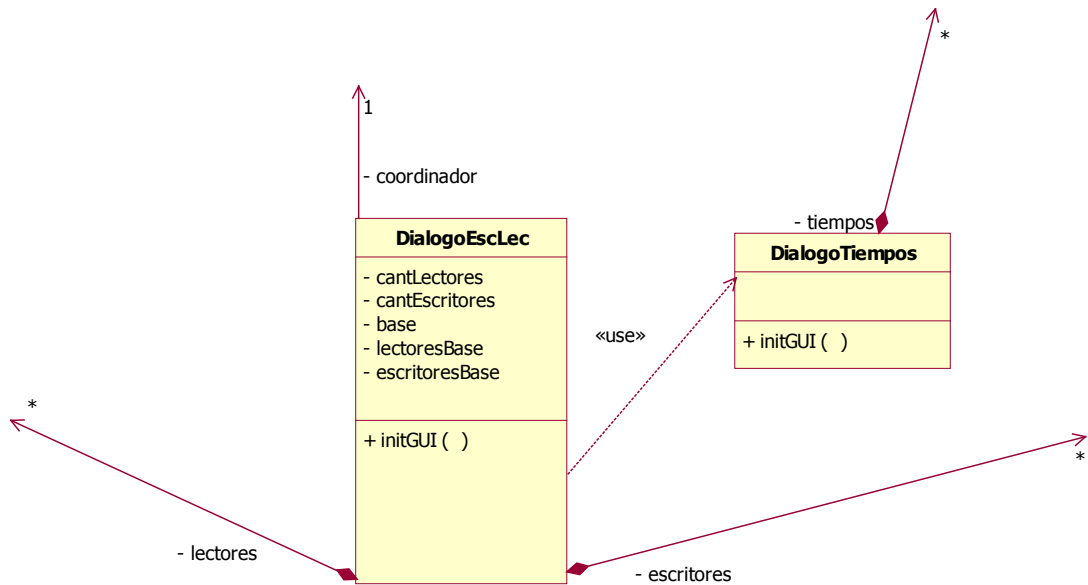
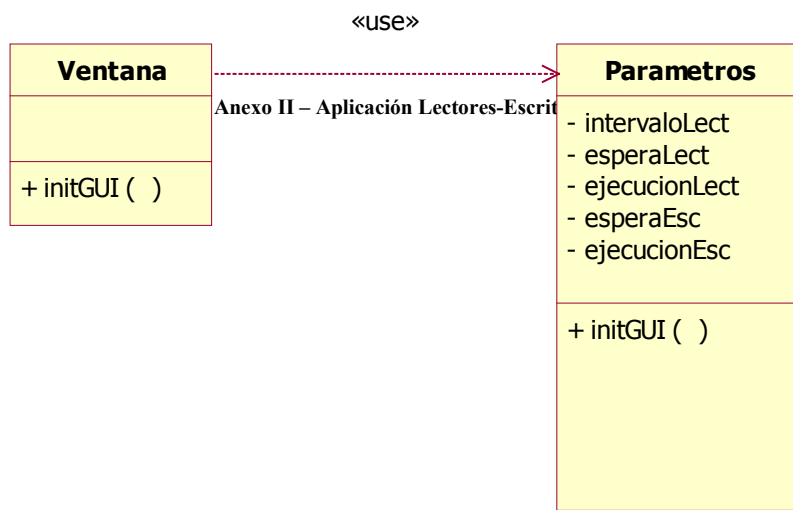
Diagrama de Clases

Las clases Ventana, Parámetros, DialogoEscLec y DialogoTiempos se utilizan para la parte visual de la aplicación. La clase Ventana es la clase principal, que muestra la base de datos con la que se quiere trabajar, y permite elegir la cantidad de lectores/escritores con la que se quiere realizar la prueba sobre la base de datos. La clase Parámetros nos permite seleccionar:

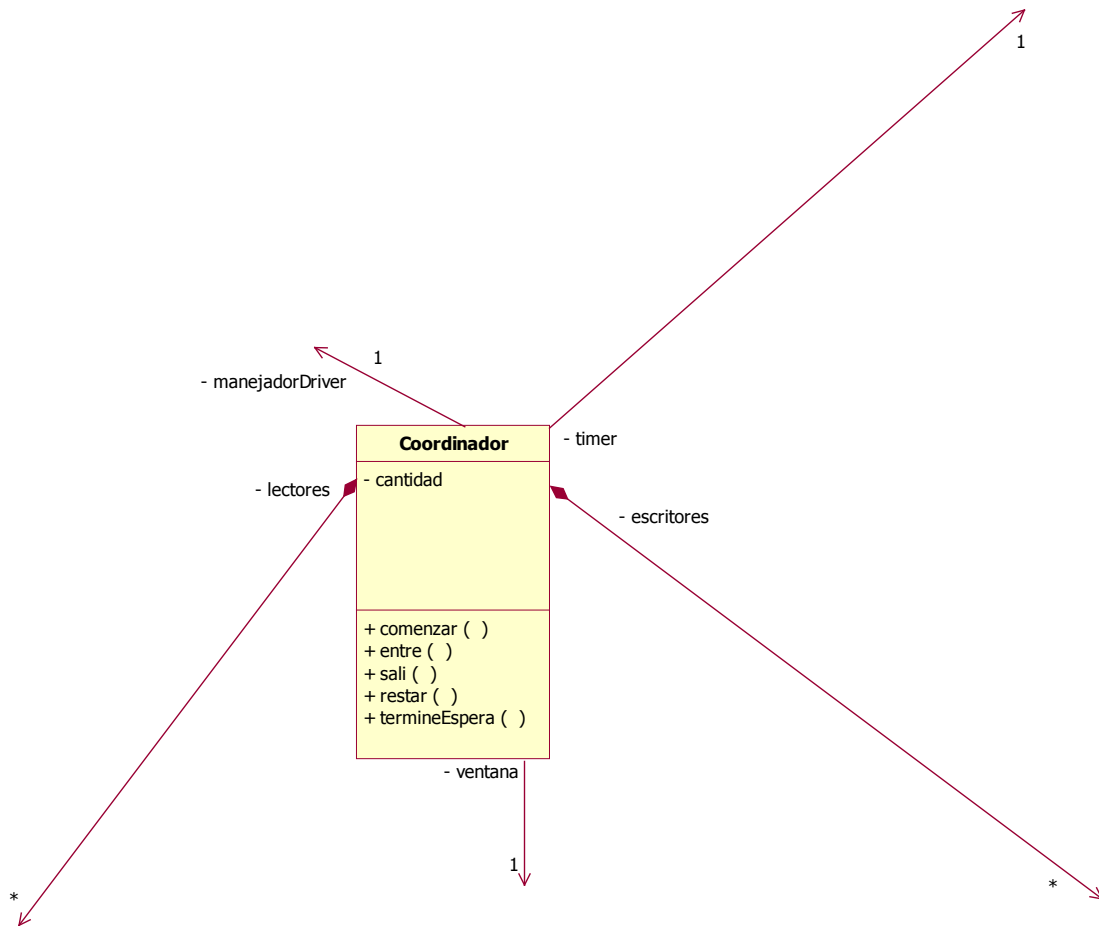
- Intervalo largada Lectores
- Tiempo espera Lectores
- Tiempo ejecución Lectores
- Tiempo espera Escritores
- Tiempo ejecución Escritores

La clase DialogoEscLec sirve para poder visualizar la ejecución de la prueba de los lectores/escritores sobre la base de datos preseleccionada.

La clase DialogoTiempos sirve para representar la ejecución de la prueba en forma gráfica mediante un gráfico de barras horizontal.

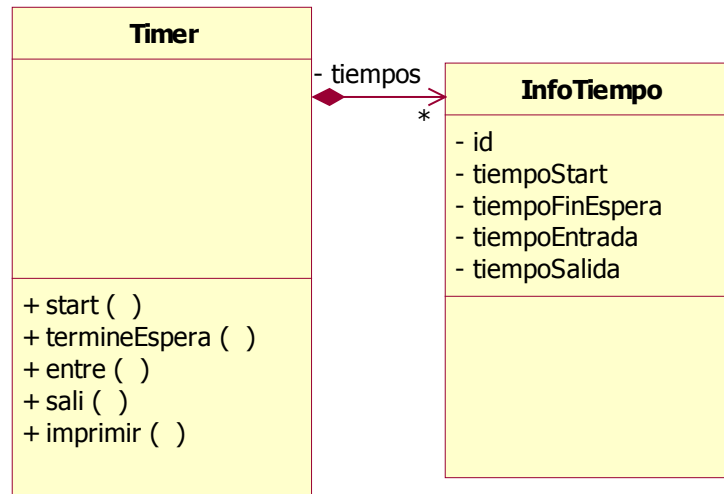


La clase Coordinador es la encargada de crear los Lectores y los Escritores (Threads) que van a acceder a la base de datos. Además es a quien le avisan los Lectores y los Escritores cuando terminan su espera (tiempo de espera establecido como parámetro), cuando “entran” a la base de datos y cuando “salen” de la base de datos.

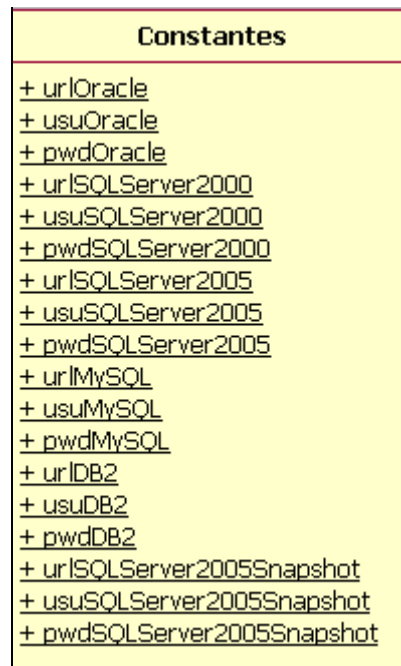


La clase Lector y Escritor implementan la interfaz Runnable, lo que les da la capacidad de ejecutarse como Threads a través del método run(). Además son los encargados de pedirle a la clase ManejadorDriver que registre el Driver que utiliza para conectarse con la base de datos y cree la conexión con la base de datos.

Las clases Timer e InfoTiempo ayudan al Coordinador y al DialogoTiemplos para poder calcular los tiempos de los Lectores/Escritores.

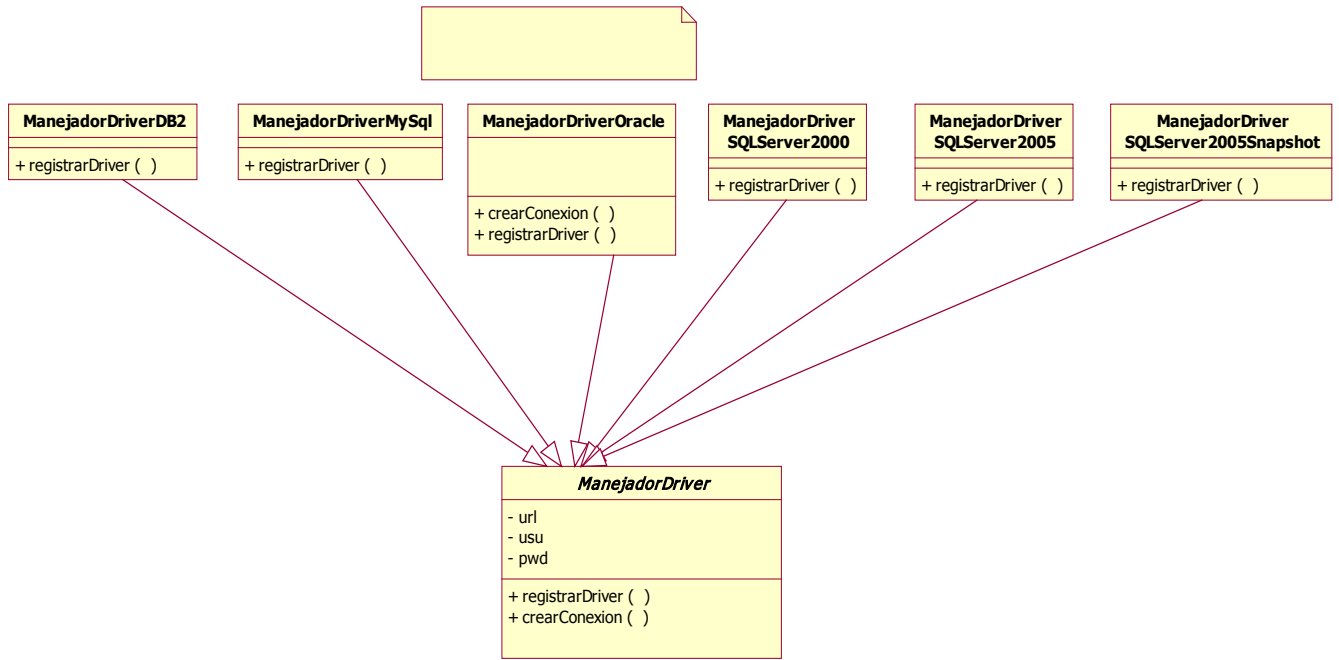


La clase Constantes se utiliza para almacenar la URL, nombre de usuario y password de cada base de datos.



La jerarquía de ManejadorDriver se utiliza para registrar el Driver y crear la conexión de cada uno de los tipos de bases de datos.

Anexo II – Aplicación Lectores-Escritores



Ejemplo del archivo config.properties:

#

#

#

SQL Server 2000

Para el caso de SQL Server 2000 se debe reemplazar el tipo de datos

SQL Server 2005

Al igual que en SQL Server 2000, en SQL Server 2005 se debe reemplazar el tipo de datos

Para el caso de SQL Server 2005 cuando utiliza MVCC (SNAPSHOT), se debe crear otro usuario (USUARIO_TESIS2) con la misma clave (UT), otra base de datos (TESIS2), “correr” el script para crear la tabla “TABLA1” y ejecutar la siguiente sentencia, para que el manejador funcione en modo SNAPSHOT:
ALTER DATABASE TESIS2 SET READ_COMMITTED_SNAPSHOT ON;

DB2

Para el caso de DB2 se necesita agrandar el logfile. Para ello ejecutar las siguientes sentencias:

Conectarse a la base de datos tesis:

```
C:\DB2\SQLLIB\BIN>
```

Verificar el valor que tiene actualmente para el logfile:

```
C:\DB2\SQLLIB\BIN>
```

Ampliar el logfile:

```
C:\DB2\SQLLIB\BIN>
```


Detener el motor de base de datos:

```
C:\DB2\SQLLIB\BIN>
```

Arrancar el motor nuevamente:

```
C:\DB2\SQLLIB\BIN>
```

Verificar el valor que tiene actualmente para el logfile(debería ser de 10000):

```
C:\DB2\SQLLIB\BIN>
```

Además en DB2, el script de creación no debe tener caracteres “Enter”, solo debe estar separado por “espacios en blanco”, y las claves primarias de las tablas deben tener a continuación de la definición la palabra “ ”.