

Ejercicio 2 – Sistema Cliente-Servidor: Micro Base de Datos Transaccional

Este módulo extiende el sistema concurrente del TP1 con una capa de red, permitiendo a múltiples clientes concurrentes consultar y modificar el inventario de forma transaccional mediante TCP.

1. Arquitectura y Concurrencia del Servidor

El servidor está implementado con el modelo **fork** por cliente y admite control de capacidad y cola de espera.

Parámetro	Descripción	Valor Predeterminado
Clientes Activos (-c)	Límite de procesos hijos concurrentes.	5
Cola de Espera (-w)	Tamaño de la cola para nuevas conexiones.	10

Gestión de Conexiones

1. **Conexión Admitida:** Si hay un *slot* libre ($< -c$), el cliente se conecta inmediatamente y comienza a operar.
2. **Lista de Espera:** Si el límite de clientes activos ($-c$) está lleno, el cliente es **encolado** (hasta $-w$) y recibe: **INFO;WAIT;pos=X cap=Y**.
 - Cuando un cliente activo se desconecta, el servidor promueve automáticamente al primer cliente en espera, enviándole: **INFO;ADMITTED**.
3. **Servidor Lleno:** Si la cola de espera ($-w$) también está llena, el servidor responde: **ERROR;Servidor ocupado. Lista de espera llena.** y cierra la conexión inmediatamente para evitar la sobrecarga.

2. Protocolo de Comunicación y Transacciones

El protocolo es textual, línea por línea, y se basa en respuestas **OK;...** (éxito) o **ERROR;...** (fallo).

Control Transaccional (Bloqueo Exclusivo)

Todas las operaciones de modificación de datos se ejecutan dentro de una transacción que utiliza el bloqueo de archivo exclusivo (`flock(LOCK_EX | LOCK_NB)`).

Comando	Tipo de Operación	Descripción y Requisitos
<code>BEGIN_TRANSACTION</code>	Bloqueo (DCL)	Intenta obtener un lock exclusivo (<code>LOCK_EX</code>). Si el <i>lock</i> está en uso, responde: <code>ERROR;La base de datos está bloqueada....</code>
<code>COMMIT_TRANSACTION</code>	Bloqueo (DCL)	Aplica los cambios permanentes y libera el <i>lock</i> exclusivo.
<code>INSERT;...</code>	Modificación (DML)	Requiere transacción activa. Asigna el ID máximo actual + 1 (protegido por el lock).
<code>UPDATE;...</code>	Modificación (DML)	Requiere transacción activa. Permite modificar <code>SKU</code> , <code>Cantidad</code> o <code>Generator</code> .
<code>DELETE;...</code>	Modificación (DML)	Requiere transacción activa. Elimina el registro por ID.

Consultas (Bloqueo Compartido)

Las operaciones de lectura (`QUERY`) usan un lock compartido no bloqueante (`flock(LOCK_SH | LOCK_NB)`):

Comando	Uso del Lock	Comportamiento
<code>QUERY;</code>	Lock Compartido (<code>LOCK_SH</code>)	Lista todos los registros. Múltiples lecturas son concurrentes si no hay una transacción activa.
<code>QUERY;Field=Value</code>	Lock Compartido (<code>LOCK_SH</code>)	Permite filtrar por <code>ID</code> , <code>SKU</code> , <code>CANTIDAD</code> o <code>GENERATOR</code> .

Si un cliente intenta una `QUERY` mientras otro tiene una `BEGIN_TRANSACTION` activa, la consulta **falla inmediatamente** (porque no se puede obtener el `LOCK_SH` sobre un `LOCK_EX` activo) y recibe el error.

Comando	Descripción
<code>EXIT</code>	Cierra la conexión.

HELP	Muestra los comandos disponibles en el cliente.
MENU	Abre un modo interactivo asistido en el cliente.

3. Robustez y Manejo de Fallas

El sistema está diseñado para manejar desconexiones inesperadas y apagados ordenados o abruptos.

Escenario de Falla	Comportamiento del Servidor
Caída de Cliente en Transacción	El hijo detecta el corte (<code>recv <= 0</code>), cierra el archivo y el socket. El flock se libera automáticamente al cerrar el descriptor de archivo, permitiendo la operación inmediata a otros clientes.
Apagado Ordenado del Servidor (Ctrl+C / <code>kill -TERM</code>)	El proceso padre cierra el <code>socket</code> de escucha. Los hijos monitorean una "lifeline" (tubería) y, al detectar la terminación del padre, salen ordenadamente y liberan sus <i>locks</i> .
Apagado Abrupto del Servidor (<code>kill -KILL</code>)	Los hijos detectan el cierre de la <i>lifeline</i> y terminan rápidamente. Los <i>locks</i> y <i>sockets</i> se liberan al finalizar los procesos hijos.

4. Opciones de Línea de Comandos (CLI)

La configuración se puede establecer mediante un archivo opcional (`server.conf`) o mediante CLI, siendo los **parámetros CLI** prioritarios.

Servidor (`./server`)

Opción	Descripción
<code>-f <db_file></code>	Ruta al CSV (obligatorio).
<code>-a <address></code>	Dirección de escucha (default: <code>127.0.0.1</code>).
<code>-p <port></code>	Puerto de escucha (default: <code>8080</code>).
<code>-c <max_clients></code>	Límite de clientes concurrentes (default: <code>5</code>).

`-w <wait_queue>` Tamaño de cola de espera (default: `10`).

Cliente (`./client`)

Opción	Descripción
<code>-a</code> <code><address></code>	Dirección del servidor (default: <code>127.0.0.1</code>).
<code>-p <port></code>	Puerto del servidor (default: <code>8080</code>).

Autores

Grupo 04 Materia: Sistemas Operativos - 3649

Universidad Nacional de La Matanza

Segundo Cuatrimestre 2025 - Comisión 01-1900