**1. ¿Qué es SQLite y cómo funciona?**  
SQLite es un sistema de gestión de bases de datos relacional (DBMS) pequeño, rápido y autónomo, implementado como una biblioteca en C que se integra directamente dentro de aplicaciones, sin requerir un servidor independiente [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[Wikipedia](https://es.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com). La base de datos completa reside en un único archivo que contiene definiciones, tablas, índices y datos [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[Wikipedia](https://es.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com). SQLite ofrece compatibilidad con la mayoría del estándar SQL-92, transacciones ACID, y se orienta al uso ligero y embebido [sqlite.org](https://www.sqlite.org/features.html?utm_source=chatgpt.com)[Wikipedia](https://es.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com).

**2. Ventajas y desventajas en relación con los criterios propuestos:**

| **Criterio** | **Ventaja en SQLite** | **Desventaja o limitación** |
| --- | --- | --- |
| Control de redundancia | Soporta esquemas relacionales que reducen redundancia | – |
| Coherencia de los datos | ACID asegura consistencia incluso ante fallos [sqlite.org](https://www.sqlite.org/features.html?utm_source=chatgpt.com) | – |
| Más información con los mismos datos | Lenguaje SQL permite consultas flexibles | – |
| Compartición de los datos | Multihilo y multiproceso pueden leer; escritura secuencial con WAL [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com) | Concurrencia limitada, bloqueo de archivo [DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com)[Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com) |
| Mayor integridad de los datos | Transacciones ACID refuerzan integridad | – |
| Mayor seguridad | Seguridad básica mediante permisos de sistema de archivos [DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com) | No hay gestión de usuarios ni privilegios de DBMS [Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com) |
| Imposición de estándares | Sigue SQL-92 parcialmente | No implementa procedimientos almacenados ni ALTER TABLE completo [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com) |
| Economía de escala | Muy eficiente y ligero, ideal para entornos embebidos [sqlite.org+1](https://www.sqlite.org/whentouse.html?utm_source=chatgpt.com) | – |
| Equilibrio entre requerimientos conflictivos | Ofrece robustez sin servidor ni administración | – |
| Accesibilidad y capacidad de respuesta | API sencilla, baja latencia al evitar IPC [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[sqlite.org](https://www.sqlite.org/features.html?utm_source=chatgpt.com) | – |
| Mayor productividad | Zero-configuration; “funciona tal cual” [clarionmag.jira.com](https://clarionmag.jira.com/wiki/spaces/clarion/pages/404750/SQLite%2BAdvantages%2Band%2BDisadvantages?utm_source=chatgpt.com)[DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com) | – |
| Mantenimiento más sencillo | No requiere DBA; archivo único portable [sqlite.org](https://www.sqlite.org/whentouse.html?utm_source=chatgpt.com)[clarionmag.jira.com](https://clarionmag.jira.com/wiki/spaces/clarion/pages/404750/SQLite%2BAdvantages%2Band%2BDisadvantages?utm_source=chatgpt.com) | – |
| Mayor concurrencia | Lecturas concurrentes; WAL mejora algo | Escrituras secuenciales; concurrencia limitada [Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[news.ycombinator.com](https://news.ycombinator.com/item?id=26816954&utm_source=chatgpt.com)[Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com) |
| Mejor respaldo y recuperación | ACID y WAL apoyan recuperación; respaldos simples mediante copia cuando no hay actividad [Reddit](https://www.reddit.com/r/programming/comments/1fdntj3/sqlite_is_not_a_toy_database/?utm_source=chatgpt.com)[sqlite.org](https://www.sqlite.org/whentouse.html?utm_source=chatgpt.com) | Copia directa no segura si la DB está activa [Reddit](https://www.reddit.com/r/programming/comments/1fdntj3/sqlite_is_not_a_toy_database/?utm_source=chatgpt.com) |

**3. Funcionalidades de un SGBD que ofrece SQLite y las que no:**

* **Ofrecidas**:
  + Almacenamiento, recuperación y actualización de datos mediante SQL.
  + Soporte de transacciones ACID [sqlite.org](https://www.sqlite.org/features.html?utm_source=chatgpt.com).
  + Servicios de recuperación integrados vía ACID/WAL.
  + Integridad soportada por transacciones y restricciones SQL.
* **Ausentes o limitadas**:
  + Un catálogo accesible para usuario como un esquema formal.
  + Servicios completos de control de concurrencia: solo bloqueo de archivos y escritura secuencial [DigitalOcean](https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems?utm_source=chatgpt.com)[Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com).
  + Servicios de autorización (usuarios y permisos).
  + Catálogo formal y robusto de metadatos accesibles.
  + Procedimientos almacenados y soporte limitado para ALTER TABLE y triggers [Wikipedia](https://it.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com)[Wikipedia](https://en.wikipedia.org/wiki/SQLite?utm_source=chatgpt.com).

**4. ¿Se adhiere SQLite a la arquitectura de tres niveles ANSI-SPARC?**

SQLite no sigue de forma estricta el modelo ANSI-SPARC (externo, conceptual, interno). Este es un estándar abstracto que separa vistas de usuario, modelo conceptual y almacenamiento físico [Wikipedia](https://en.wikipedia.org/wiki/ANSI-SPARC_Architecture?utm_source=chatgpt.com)[GeeksforGeeks](https://www.geeksforgeeks.org/the-three-level-ansi-sparc-architecture/?utm_source=chatgpt.com). SQLite sí implementa cierto grado de separación (por ejemplo, su API oculta detalles físicos y el usuario ve datos relacionales), pero no provee esquemas externos personalizados ni independencia conceptual física completa. En esencia, la separación es limitada y no cumple claramente los tres niveles.