OmniCortex Storage Architecture

Technical Reference

OmniCortex uses SQLite for all data storage - a deliberate architectural choice that prioritizes simplicity, portability, and reliability over distributed complexity.

Why SQLite?

Technical Benefits

- Zero configuration required
- Single-file databases
- ACID compliant
- Built-in FTS5 for search

Practical Benefits

- Works offline
- Easy backup (copy file)
- No server process
- Cross-platform portable

Storage Locations

Location	Path	Purpose
Project DB	.omni-cortex/cortex.db	Memories, activities, sessions for project
Global DB	~/.omni-cortex/global.db	Cross-project search index
Project Config	~/.omni-cortex/projects.json	Dashboard settings, favorites, scan dirs

Database Schema

memories Table

Column	Type	Description
id	TEXT PK	ULID-based unique identifier

content	TEXT	Main memory content
context	TEXT	Additional context
memory_type	TEXT	fact, decision, solution, etc.
status	TEXT	fresh, needs_review, outdated, archived
importance_score	INTEGER	1-100 importance rating
tags	TEXT	JSON array of tags
created_at	TIMESTAMP	Creation time
last_accessed	TIMESTAMP	Last access time
access_count	INTEGER	Number of accesses

activities Table

Column	Type	Description
id	TEXT PK	ULID-based unique identifier
event_type	TEXT	pre_tool_use, post_tool_use, etc.
tool_name	TEXT	Name of tool called
tool_input	TEXT	JSON input parameters
tool_output	TEXT	JSON output result
success	BOOLEAN	Whether operation succeeded
duration_ms	INTEGER	Execution time in ms
session_id	TEXT	FK to sessions table
timestamp	TIMESTAMP	Event time

memory_relationships Table

Column	Type	Description
source_id	TEXT FK	Source memory ID
target_id	TEXT FK	Target memory ID
relationship_type	TEXT	related_to, supersedes, derived_from, contradicts
strength	REAL	Relationship strength 0.0-1.0
created_at	TIMESTAMP	When relationship was created

sessions Table

Column	Type	Description
id	TEXT PK	Session identifier
started_at	TIMESTAMP	Session start time
ended_at	TIMESTAMP	Session end time (nullable)
summary	TEXT	Auto-generated or manual summary
key_learnings	TEXT	JSON array of learnings

Full-Text Search (FTS5)

OmniCortex uses SQLite's FTS5 extension for fast full-text search across memory content and context.

```
FTS5 Index: memories_fts (content, context)
Supports: phrase matching, prefix search, boolean operators
```

Global Index

The global database at ~/.omni-cortex/global.db maintains a cross-project search index for finding memories across all your projects.

Column	Type	Description
memory_id	TEXT	Original memory ID
project_path	TEXT	Source project path
content	TEXT	Memory content (synced)
memory_type	TEXT	Memory type
tags	TEXT	JSON tags array
synced_at	TIMESTAMP	Last sync time

Project Configuration

Dashboard project preferences are stored in ~/.omni-cortex/projects.json:

Field	Type	Description
version	integer	Config schema version
scan_directories	string[]	Directories to scan for projects
registered_projects	object[]	Manually added projects with path, display_name, added_at
favorites	string[]	Paths of favorite projects
recent	object[]	Recently accessed projects (last 10) with path, last_accessed

Backup & Migration

Because OmniCortex uses file-based storage, backup and migration are straightforward:

- **Project Backup:** Copy the .omni-cortex/ directory
- **Global Backup:** Copy ~/.omni-cortex/ directory
- **Migration:** Move directories to new machine - no reconfiguration needed
- **Export:** Use cortex_export tool for JSON/Markdown/SQLite exports