cxcodex Manual (Field Guide)

Last updated: 2026-02-22

How to Read This

This is a practical field guide: short rules, common failure modes, and “do this next” recipes. It’s written for the moment you’re stuck, skeptical, or trying to keep costs under control.

1 First Checks (30 Seconds)

```
./bin/cx version
./bin/cx diag
./bin/cx where version diffsum-staged task optimize
```

If `execution_path` is not Rust, you are on fallback and should expect reduced guarantees.

2 Capture and Context (The Budget Is the Product)

Symptoms

- Runs become slow and expensive.
- Responses get vague (model is buried in noise).
- Cache hit rate degrades (prompts drift).

Fixes

- Reduce system output: prefer targeted `git diff -- <paths>` over whole-repo dumps.
- Enforce budgets: set `CX_CONTEXT_BUDGET_CHARS` / `LINES` temporarily smaller to test clipping.
- Use chunking for fan-out: chunk output and turn it into parallelizable tasks.

```
CX_CONTEXT_BUDGET_CHARS=2000 CX_CONTEXT_BUDGET_LINES=40 ./bin/cx cxo git status
./bin/cx chunk < some_big_output.txt
```

3 Schemas (Determinism You Can Audit)

What “Schema-Enforced” Means

- Prompt requests JSON-only output.
- Output parses as JSON.
- Output validates against `.codex/schemas/<name>.schema.json`.
- Failure triggers quarantine + non-zero exit.

```
./bin/cx schema list
./bin/cx commitjson | jq .
./bin/cx diffsum-staged | jq .
```

If It Fails

```
./bin/cx quarantine list
./bin/cx replay <quarantine_id>
```

4 Safety (You Are Not Your Shell History)

Block List (Minimum)

- `sudo`, `rm -rf`, `curl | bash`
- chmod/chown on system paths
- writes outside repo root

```
./bin/cx policy show
./bin/cx policy check "sudo rm -rf /"
```

5 Tasks (Parallel-Ready, Single-Worker Today)

Use tasks when the objective is bigger than one bounded prompt.

```
./bin/cx task add "Tighten schema errors" --role implementer
./bin/cx task fanout "Ship release note improvements" --from staged-diff
./bin/cx task run-all --status pending
```

6 Telemetry and Logs (Trust, but Verify)

```
./bin/cx logs validate --fix=false
./bin/cx metrics 50
./bin/cx trace
```

Interpretation Cheatsheet

- `effective_input_tokens` high: prompts too large or too variable.
- `clipped=true` often: reduce capture scope or raise budgets intentionally.
- cache trend down: stabilize prompt templates and system capture.

7 Optimize (Make It Cheaper, Faster, Safer)

```
./bin/cx optimize 200
./bin/cx optimize 200 --json | jq .
```