Discussion catalogs for any project

STATE
committed

TYPE
Standards Track

AUTHORS
Vikrant Rathore

CREATED
2026-10-12

UPDATED
2026-10-15

DISCUSSION
https://github.com/insanai/guidedog/issues/1

LABELS
library, cli, process

Record status

The catalog defines the review process and the meaning of each state. This numbered record forms part of the catalog.

Contents

1. Abstract 1
2. Motivation and scope 1
3. Specification 2
4. Rationale and alternatives 3
5. Security and operational considerations 3
6. Backwards compatibility 3
7. Acceptance criteria and implementation status 3
8. Open questions 3
9. Review history 3
10. References 4

1. Abstract

GDS 0001 gave Guidedog a way to propose, review, and record its own decisions. This record makes that process a library and a set of commands that any project can use. A *discussion catalog* is a folder of reStructuredText records under a prefix: `gds` for Guidedog, `sds` for Sanin, `eds` for Enso. `gd-discussion` keeps catalogs, and its plugin adds the `guidedog dsc` commands, linked into `guidedog`. Guidedog's own catalog in `docs/gds` is one catalog among others. A reader of any catalog sees its metadata as Number, Title, Authors, Updated, and Labels, in HTML and PDF.

2. Motivation and scope

A prototype in Sanin, `discussion-cli`, copied Guidedog's catalog code to keep its own catalog. Copies drift: a fix to recovery in one does not reach the others, and each must be qualified again. Issue 1 asks for four things. One reusable package. One command line, with a build that needs no separate installation. Readable metadata for every prefix. Explicit failures where the prototype could report success with an incomplete plan or an empty record.

This record covers catalog management, the commands, the metadata presentation, and each catalog's configuration of its lifecycle, record types, labels, and numbering, and a bound on the memory a catalog needs. A catalog without configuration follows GDS 0001.

3. Specification

3.1. Library

`gd-discussion` takes a validated `Prefix` (2 to 16 lowercase letters), or the default `gds`, and names everything after it: fields `:eds-number:`, the lock `.eds.lock`, the journal `.eds-journal.json`, generated markers `.. eds-generated-start`, problem codes `eds.date`. A GDS catalog therefore reads and writes exactly what GDS 0001 specifies.

A catalog has a configuration, `discussion.toml`: the prefix, a display name, a default author, and its process. `init` writes it with `conf.toml`, `index.rst`, `workflow.rst`, a record template, the generated index, and a process draft. The older `.discussion.json` definitions remain readable. A catalog without either file is named by the number field of its template, so existing catalogs need no change.

Records keep the GDS 0001 fields. `last-updated` may stand for `updated` (never both), and a record may carry `category`, `status`, and `summary`; `status` follows the state. A metadata edit replaces only its field's line, so every other byte, line endings included, stays as it was.

Types make misuse impossible where they can. `apply` and `recover` require the `Lock` (released on drop), so an unlocked write does not compile. Each operation has one request type (`NewDraft`, `Decision`, `BuildOptions`, `Init`), and `replace_field` takes a typed `FieldValue`. Problems carry a line, an excerpt with a caret and a "Did you mean" suggestion, rendered by `gd-report`.

3.2. Commands

`gd_discussion::plugin()` contributes the commands through the `gd-plugins` command interface, and `cmd/guidedog` links it beside the Pages plugin. So `guidedog dsc` offers `init`, `new`, `list`, `show`, `promote`, `state`, `index`, `check`, `build`, `recover`, and `config` on any catalog. `--dir` names the catalog; without it, the command uses the catalog the current folder is in.

A build checks the catalog, then asks the host to build it as a Guidedog project, warnings as errors, with the books as `pdf_documents`. The `Builder` trait is that request. Older Typst catalogs compile with `gd-typst` inside `gd-discussion`. Guidedog's own catalog has no command of its own: its prefix, workflow, labels and numbering are in `docs/gds/discussion.toml`, and we run `guidedog dsc` in `docs/gds`.

3.3. Configuration

A catalog's `discussion.toml` holds its prefix, name, default author, labels, record types, numbering, and workflow. `init` writes every setting with a comment, and `dsc config` prints the settings in effect. The defaults are GDS 0001's process, so a catalog without the file, such as a Sanin catalog, behaves as GDS 0001 says.

The workflow names the state of a new draft and the state promotion opens it in. Each state lists the states the `state` command may move a record to (`next`), whether a move into it needs evidence, and the text of a record's optional `status` there. The evidence requirement also applies when promotion enters the open state. Promotion is the only move from the draft state to the open state, because it gives the number. Otherwise a draft may move only to states its own `next` lists, and a record without a number may be only in those states. Every state change needs a reason.

Numbers have at least `width` digits (default 4), and grow past it: 10000 follows 9999, up to nine digits, and records sort by value, not by text. Promotion numbers from `start`, up to `limit` when one is set. Both govern new numbers only, so changing them never invalidates a record. Changing `width` would, and the check names the record and the setting. A mistake in the file is reported with its line and the fix, and unknown settings are refused rather than ignored.

The process draft `init` writes describes the configured lifecycle, with a table of the states and a diagram of the moves, so it is never out of step with the catalog.

3.4. Memory

`load_catalog` reads each record a chunk at a time. It keeps only the metadata, at most 64 KiB in lines of at most 4 KiB, and which of the required sections the record has. It stops reading once it knows both. The

record a command changes is read whole, up to 16 MiB. A catalog costs about a kilobyte a record, may hold 100,000 records, and its metadata may total 256 MiB. Duplicate numbers and slugs are found in linear time.

3.5. Metadata presentation

The leading field list of a page is a discussion's metadata when its first field has a prefix of 2 to 16 lowercase letters and the list carries that prefix's `number`, `title`, or `reserved-through`. Its known fields show as Number, Title, State, Type, Authors, Created, Updated, Discussion, Labels, Reserved through, Category, Status, and Summary; authors and labels show as plain lists. Unknown fields, fields of another prefix, and later field lists keep their names. The source is not changed.

4. Rationale and alternatives

A compatibility layer for each project's own copy. Rejected: it would be a second copy of the API to keep in step, which is the problem this record removes.
Separate programs, one that runs an installed Guidedog and one with Guidedog built in. Rejected: two programs mean two archives and two ways a build can differ. A plugin linked into `guidedog` builds with the same engine as every other project, and needs nothing else installed.

5. Security and operational considerations

Commands that change records hold the catalog's kernel lock. `apply` checks that each file still has the contents the plan was made from, and that each change is whole: a record that parses and has its sections, an index with one generated region. Only then does it write a journal of the originals and replace each file atomically. `recover --rollback` restores the originals and refuses to overwrite a later edit. Each staging file is removed when its guard is dropped, so an interrupted change leaves no staging file behind.

6. Backwards compatibility

GDS records, numbers, locks, journals, generated markers, and problem codes are those of GDS 0001; the repository's catalog passes `guidedog dsc check` and `index --check`. `replace_field` accepts only literal values, not Typst expressions. Catalogs made by the Sanin prototype need no change.

7. Acceptance criteria and implementation status

Implemented, with tests:

- `gd-discussion` and its tests: every prefix, memo fields, field edits that keep every other byte, init, discovery, transitions, generated regions;
- interruption sweeps: `apply` is stopped after every file, and `recover` restores the catalog byte for byte;
- `discussion.toml`: reading with line-numbered problems, rendering, and a round trip of the defaults; a configured workflow with its own types, evidence, start and limit; numbers past their width and their order; `dsc config` and `init --config`;
- `load_catalog` under a measured bound;
- `guidedog dsc` integration tests (`tests/integration/tests/discussions`);
- metadata presentation tests in HTML, text, and the PDF book's Typst source.

8. Open questions

Should promotion rewrite prose links to a renamed record? Should `dsc` offer `serve`? Should a reason be optional for some moves?

9. Review history

2026-10-12: drafted with the implementation of issue 1.
2026-10-12: revised for 0.2.2: each catalog configures its lifecycle, types, labels, and numbering in `discussion.toml`; numbers grow past their width; loading keeps metadata only.
2026-10-12: independent review found gaps in promotion evidence, empty type lists, metadata limits, and TOML nesting. The working implementation has regression fixes. Semantic configuration errors now retain source lines, and initialization refuses an empty vocabulary safely. These changes are not in the published

0.2.2 archives. This record remains in discussion pending qualification and release of the corrections. See Discussion catalog review, 12 October 2026.
2026-10-12: the corrections shipped in 0.2.3 after candidate and tagged-release qualification on every release platform. Independent published-archive smoke tests also passed. The design is adopted and implemented; the open questions remain outside this implementation's acceptance criteria.
2026-10-15: the discussion commands became a plugin linked into `guidedog` (`guidedog dsc`).

10. References

- GDS 0001, the discussion process.
- Issue 1, "Add a reusable discussion CLI with CLI-only and complete builds".
- `crates/gd-discussion/README.md`, the library's cheat sheet, and `docs/manual/internals/discussion.rst`.
- Work on a discussion, the commands of this catalog.

Lifecycle event

```
2026-10-12: prediscussion -> discussion. Assigned a permanent number and
opened for discussion.
```

Lifecycle event

```
2026-10-12: discussion -> published. The maintainer-requested review adopted
the reusable catalog design after qualification.
```

Lifecycle event

```
2026-10-12: published -> committed. The implementation and review
corrections shipped in Guidedog 0.2.3.
docs/gds/evidence/discussion-review-20261012.rst;
https://github.com/insanai/guidedog/actions/runs/38225096535;
https://github.com/insanai/guidedog/actions/runs/38234710527; published dsc
and dscc HTML/PDF smoke tests.
```