

Artifact Sync System — Convergence Contract v2.5.2

Version 2.5.2

1. Purpose

This contract defines a deterministic, safe convergence process that synchronizes Source Documents into version-controlled Markdown artifacts under `artifacts/`.

The system is a reconciler: it converges derived repository state to match the current set of eligible Source Documents while enforcing representation and safety invariants.

2. Definitions

Source Document

Externally managed document identified by a stable `source_id`.

Folder Spec

Configuration tuple:

- `folder_id`
- `project`
- `recursive` (boolean)

`folder_ids` MUST be unique.

Derived Unit

Repository directory:

`artifacts/<project_norm>/<source_id>/`

Managed Unit

A Derived Unit that passes Managed Unit Validation (§6).

SupportedSchemaVersions

Set of acceptable `schema_version` values.

If not configured, default MUST be `["2.5.2"]`.

If configured, it MUST include `"2.5.2"`.

3. Source Model

3.1 Canonical Identity

Canonical identity is `source_id`.

`source_id` MUST be globally unique and stable within the active Profile.

3.2 Eligibility

Only documents with MIME type `application/vnd.google-apps.document` are eligible.

All other MIME types MUST be ignored and logged.

3.3 Discovery Pipeline (Normative Order)

Profile MUST execute discovery in this order:

1. Traverse configured Folder Specs.
2. Verify listing completeness (§7.1).
3. Evaluate multi-folder membership policy (§9.3).
4. Deduplicate by `source_id`.

Policy evaluation MUST occur before deduplication.

3.4 Project Assignment

Each eligible document MUST map to exactly one project via Folder Spec configuration.

Folder name MUST NOT be used for project derivation.

If project normalization fails, outcome MUST be `SKIPPED_METADATA`.

4. Derived State Model

4.1 Project Normalization

`project` → `project_norm`:

- lowercase
- trim
- whitespace → -
- allow `[a-z0-9-]` only
- collapse repeated -
- trim leading/trailing -

If empty → `SKIPPED_METADATA`.

4.2 Output Layout

For each successfully exported document:

- `artifacts/<project_norm>/<source_id>/doc.md`
- `artifacts/<project_norm>/<source_id>/meta.json`

4.3 meta.json Schema

`meta.json` MUST contain:

- `schema_version`
- `source_id`
- `project_norm`
- `title`
- `exported_at` (ISO-8601 timestamp)

Additional fields are prohibited unless introduced in a future schema version.

`schema_version` MUST be a member of `SupportedSchemaVersions`.

5. Representation Invariants

5.1 Markdown Dialect

Canonical dialect is GitHub-Flavored Markdown (GFM).

Structural HTML is prohibited.

5.2 Structural Fidelity

Exporter MUST derive representation from authoritative structural metadata in the Google Docs `documents.get` response.

Headings, paragraphs, lists (including nesting), tables, inline formatting, hyperlinks, and inline code MUST be preserved as derivable from that metadata.

If structural metadata is missing or insufficient to reliably determine structure, outcome MUST be `FAILED_REPRESENTATION`.

Tables MUST render as GFM pipe tables.

Merged cells that cannot be normalized with explicit loss markers MUST result in `FAILED_REPRESENTATION`.

5.3 Idempotence

Idempotence applies to identical `source_snapshot`.

`source_snapshot` MUST be the normalized structural representation derived from the Google Docs API `documents.get` response.

Given identical `source_snapshot`:

- `doc.md` MUST be byte-identical.
- `meta.json` MAY differ only in `exported_at`.
- Block ordering MUST be stable.

6. Managed Scope and Validation

6.1 Managed Scope

Exporter manages only:

`artifacts/<project_norm>/<source_id>/**`

Unmanaged project-level content MUST NOT be modified or deleted.

Project directories MUST NOT be deleted.

6.2 Managed Unit Validation

A directory is a Managed Unit if:

- Path matches `artifacts/<project_norm>/<source_id>/`
- `meta.json` exists
- `meta.schema_version` \in `SupportedSchemaVersions`
- `meta.source_id` == `<source_id>`
- `meta.project_norm` == `<project_norm>`

Exporter MUST NOT delete directories failing validation.

6.3 Atomic Write

Exporter MUST:

1. Build full Derived Unit in temporary directory.
2. Validate Representation Invariants.
3. Write complete `meta.json`.
4. Atomically rename into place.

Incomplete units MUST NOT be committed.

7. Convergence Model

7.1 Folder Listing Atomicity

All configured `folder_ids` MUST be fully listed.

If pagination fails, listing is incomplete, or configuration is invalid, outcome MUST be run-level `FAILED_OPERATIONAL`.

No deletions occur.

7.2 Convergence Target

DesiredSet = documents with outcome `EXPORTED`.

RetainedSet = documents with outcome `SKIPPED_METADATA` or `FAILED_POLICY`.

CurrentUnits = all directories passing Managed Unit Validation.

Delete (if deletion gates allow):

CurrentUnits - (DesiredSet \cup RetainedSet)

7.3 Eligibility Drop

If a previously eligible document is not discovered in the current run and deletion gates allow, its Managed Unit MUST be deleted.

Deletion preserves Git history.

8. Failure Model

Document outcomes:

- EXPORTED
- SKIPPED_METADATA
- FAILED_OPERATIONAL
- FAILED_REPRESENTATION
- FAILED_POLICY

Run-level FAILED_OPERATIONAL MUST result in:

- Exit code = 1
- No deletions

If any document outcome is FAILED_OPERATIONAL or FAILED_REPRESENTATION:

- Exit code = 1
- No deletions

FAILED_POLICY MUST NOT abort run.

9. Folder-Scoped Profile

9.1 Folder Spec

Each Folder Spec defines:

- folder_id
- project
- recursive

Multiple Folder Specs MAY map to the same project.

9.2 Recursive Traversal

Default traversal is non-recursive.

If `recursive=true`, traversal MUST be full-depth.

Traversal order MUST NOT affect output determinism.

9.3 Multi-Folder Membership Policy

If a document appears under more than one distinct `folder_id`:

- Outcome = `FAILED_POLICY`
 - Conflicting `folder_ids` MUST be logged
 - Document MUST be included in `RetainedSet`
 - Export MUST NOT occur
-

10. Fixture Suite

Fixtures MUST exist under `fixtures/`.

Each fixture MUST include:

- `source_snapshot.json`
- Expected `doc.md`
- Expected `meta.json`

CI MUST execute fixture validation.

Failure to match fixture output constitutes `FAILED_REPRESENTATION`.

11. Non-Goals

- Using Drive Labels API.
- Garbage collection of invalid Managed Units.
- Pixel-perfect source layout preservation.
- Partial merges of manual edits.

- Supporting HTML as canonical structural output.