

Artifact Sync System — Convergence Contract v2.6.3

Version 2.6.3

1. Purpose

This contract defines a deterministic, safe convergence process that synchronizes Google Docs “artifact” documents into version-controlled Markdown artifacts under `artifacts/`. The system converges derived repository state to match the current set of eligible Source Documents while enforcing representation and safety invariants.

2. Definitions

2.1 Source Document

Externally managed Google Docs document identified by a stable `source_id` (Google Docs document ID).

2.2 Document Tabs

A Google Docs document may contain one or more “tabs” (sub-pages). Tabs may be nested via child tabs.

For this contract, “Source Document” means the full tab set of the document, not only the first/default tab.

2.3 Tab Body Scope

Only **tab body** content is in-scope for derived representation.

Out-of-scope (ignored) unless a future contract version adds explicit handling:

- footnotes
- headers/footers
- comments/suggestions
- revision history

2.4 Folder Spec

Configuration tuple:

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

`folder_ids` MUST be unique.

2.5 Derived Unit

Repository directory:

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

2.6 Managed Unit

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

2.7 SupportedSchemaVersions

Set of acceptable `schema_version` values.

- If not configured, default MUST be ["2.6.3", "2.6.2", "2.5.2"].
- If configured, it MUST include "2.6.3".
- "2.6.2" and "2.5.2" MAY be included for backward compatibility.

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`.

3.5 Tabs Retrieval (MUST)

Exporter MUST retrieve document content in a mode that includes Document Tabs content.

Exporter MUST request tab content (`includeTabsContent=true` or equivalent)

When using the Google Docs API, exporter MUST set the `documents.get` request to include tab content (e.g., `includeTabsContent=true` or an equivalent mechanism).

3.6 Tabs Retrieval Observability (MUST)

Exporter MUST validate response shape to confirm tab-capable retrieval occurred. The response MUST satisfy exactly one of:

- **Tabs-capable response:** `Document.tabs` is present (may be length 1), OR
- **Explicitly unsupported:** the API explicitly indicates tab content is unavailable/unsupported for this document type/account.

If neither condition is true, outcome MUST be `FAILED_OPERATIONAL`.

If the response is “explicitly unsupported,” outcome MUST be `FAILED_OPERATIONAL` unless the Profile explicitly permits “tabs-unavailable mode.”

3.7 Multi-Tab Determination (MUST)

A document is considered “multi-tab” if:

- `Document.tabs` contains more than one root tab, OR
- any tab has `childTabs` non-empty.

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`.

`schema_version` MUST be `"2.6.3"`.

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.

Document Tabs MUST be preserved within Tab Body Scope (§2.3):

- All tab bodies MUST be included in the derived representation.
- Tab ordering MUST be deterministic and stable.
- Tab boundaries MUST be represented deterministically in `doc.md`.

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` (the normalized structural representation derived from the Source Model).

Given identical `source_snapshot`:

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

5.4 Tab Ordering (MUST)

Exporter MUST traverse tabs deterministically:

- Parent tab content MUST appear before any of its child tabs.
- Root tabs MUST be ordered by `tabProperties.index` ascending **only if** `index` is an integer ≥ 0 ; otherwise treat as missing.
- Child tabs MUST be ordered by `tabProperties.index` ascending **only if** `index` is an integer ≥ 0 ; otherwise treat as missing.
- If `index` is missing/invalid, preserve API array order.
- If a tie remains after `index` and array order, tie-break by `tabProperties.tabId` lexical ascending.

5.5 Tab Boundary Markers (MUST)

If the document is multi-tab (§3.7), exporter MUST emit a boundary marker before each tab's body content.

If the document is not multi-tab, exporter MAY omit boundary markers.

Boundary marker format (single line) MUST be:

- # [TAB] <title> (<tab_id>)
- If <title> is empty or unavailable: # [TAB] (<tab_id>)

<tab_id> MUST be `tabProperties.tabId`.

If <tab_id> is missing for any tab, outcome MUST be `FAILED_OPERATIONAL`.

Title normalization MUST be:

- replace CRLF/CR with LF
- replace LF with a single space
- trim leading/trailing whitespace

Title rendering MUST be safe and deterministic:

- The <title> segment MUST be wrapped in backticks.
- Any backtick characters inside the title MUST be escaped by doubling them.

The marker format is normative; fixtures MUST assert exact bytes of this marker when multi-tab.

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 a temporary directory.
2. Validate Representation Invariants.
3. Write complete `meta.json`.
4. Atomically rename into place.

Incomplete units MUST NOT be committed.

6.4 Schema Migration Policy

On successful export of a Source Document, exporter MAY rewrite an existing Managed Unit for the same `source_id` to the current `schema_version` ("2.6.3") by overwriting `doc.md` and `meta.json` per this contract.

No separate migration step is required.

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 U 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.

7.4 Deletion Gates (MUST)

Deletions are controlled by a single explicit control surface:

- `ARTIFACT_SYNC_ALLOW_DELETIONS`

Allowed values:

- `"true"` enables deletions.
- Any other value (including unset) disables deletions.

7.5 CI Deletion Safety (MUST)

CI runs MUST execute with deletions disabled.

CI MUST NOT set `ARTIFACT_SYNC_ALLOW_DELETIONS=true` except in a deletion-specific test job explicitly designed to validate deletion behavior.

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`.

Required fixtures:

- At least one multi-tab document fixture:
 - expected `doc.md` includes at least two tab boundary markers
 - expected `doc.md` includes content from more than one tab
- At least one nested child-tabs fixture (depth ≥ 2).
- At least one tab-title escaping fixture (title includes at least: `#`, `(`, `)`, and a backtick).

11. Non-Goals

- Pixel-perfect source layout preservation.

- Supporting out-of-scope structures (footnotes, headers, comments, suggestions) without explicit contract handling.
- Partial merges of manual edits.
- HTML as canonical structural output.