

Release notes

The reviewed source repository contains one tagged release: AIP Core 1.0.0. The documentation also describes a later five-commit source range that has not been tagged as a release.

SECTION	Project and Releases
TYPE	Reference
PROTOCOL	AIP 1.0
SOURCE	docs/reference/release-notes.md

Status	Identity	Use it for
Released	v1.0.0 → 034a520608eea44e6b14e61c34734bd402d989c0	Reproducing or evaluating the recorded private source release
Reviewed, unreleased	97be86e9efedf07ecf1783b03800f683f107fb04	Reading this v2 documentation and evaluating the post-release connector-fleet implementation

Both source states declare workspace version 1.0.0. The version string alone therefore cannot identify which behavior or deployment model a reader has. Use the tag or full commit, then bind built artifacts by digest.

AIP Core 1.0.0 — 21 July 2026

Release identity

Field	Value
Version	1.0.0
Annotated tag	v1.0.0
Tag object	270ffb55d6f239c42d3487a6842a011d4a7f8d2a
Release commit	034a520608eea44e6b14e61c34734bd402d989c0
Release tree	0784fbb30a9795e3098eb2498126d9ddccde243d
Commit time	2026-07-21T14:49:24+04:00
Visibility recorded by the source manifest	Private
Workspace version	1.0.0

The repository records the tag and release commit locally. Its source manifest records a private publication target, but this documentation review did not verify the remote repository or a release page.

What the release contains

The initial source release includes:

- the native AIP 1.0 model, validation, generated schemas, and Rust facade;
- authentication, discovery, runtime, PostgreSQL storage, gateway, and native transport implementations;
- MCP, A2A, and webhook compatibility components;
- the connector SDK and the five then-maintained public connectors: Cal.diy,

Hermes Agent, Chatwoot, Dify, and CrewAI;

- `aipd`, `aipctl`, conformance tooling, examples, legal notices, and release

automation.

The release was a clean-root source snapshot. `SOURCE_SNAPSHOT.json` records the source and transformation:

Snapshot field	Recorded value
Source revision	15b89cba1b42bdb460d5abef5b1345eee3ce345d
Source tree	583fa47a6ce043b56b9f4ec327a494933c11c18d
Tracked source files	293
Export method	git archive
History strategy	One clean root; no imported commits or tags
Version normalization	Workspace 1.0.2 → release 1.0.0
License metadata normalization	Apache-2.0 → BUSL-1.1
Recorded behavior change from normalization	false

The source release deliberately excludes the documentation frontend, top-level benchmarks, top-level tests, and qualification-evidence tree. `README.md` is the only Markdown document inside that publication boundary. These exclusions describe packaging, not AIP wire behavior.

See [release artifacts](#) for the complete artifact and verification boundary, and [licensing](#) for the controlling terms.

Unreleased changes after 1.0.0

Exactly five commits separate `v1.0.0` from the reviewed source revision. The `CHANGELOG` and workspace version were not advanced for this range, so the range must not be presented as another `1.0.0` release.

Date	Commit	Reader-visible change
24 July	5bf85a42e87a30f216123a7d5a500dd4af3f3e0d	Decoupled the product connector fleet from the core daemon; added the product-neutral registry, host runtime, remote dispatch, least-privilege control plane, and transitional bundled daemon
24 July	29415ac93315a54ae2a8afa536874c706adf7cb0	Completed the connector-fleet separation across hosts, storage, routing, control, CLI, deployment, and checks
25 July	79367097b1ff8b6830d3860e45d8b94405649dbb	Added the complete fleet deployment, standalone product hosts, tenant propagation, qualification components, and release documentation
25 July	417e17672510f81975c035d437e5a3f58ee53574	Added the Twenty connector and host, remote transaction routing, image coverage, and product-boundary enforcement
26 July	97be86e9efedf07ecf1783b03800f683f107fb04	Hardened durable connector delivery, replay handling, CrewAI persistence, immutable qualification images, and recovery

These commits produce the six-connector, separately hosted fleet documented by this site. They also add the normalized registry and PostgreSQL implementation, admission, orchestration, remote connector protocol, host bootstrap, control plane, scale gate, fleet qualification, and expanded release automation.

Separate protocol compatibility from deployment compatibility

The native AIP model and generated native schemas are unchanged across this specific range:

- the `crates/aip-core` tree at `v1.0.0` and at `97be86e9` is identical;
- the complete `schemas/aip` tree has no diff between those revisions;

- both source states retain native protocol version `1.0` and workspace

version `1.0.0`.

This supports a narrow conclusion: the five post-tag commits do not introduce a native AIP schema migration. It does not make the deployment, connector SDK, runtime, gateway, MCP server, CLI, persistence, or operational behavior interchangeable.

Surface	<code>1.0.0</code> release	Reviewed unreleased source	Consequence
Native AIP model and schemas	AIP 1.0	Same core and native schema trees	No native wire-schema migration in this range
Primary daemon	Bundles selected connectors	Product-neutral <code>aipd</code>	Existing bundled deployments need an explicit migration choice
Compatibility path	Original daemon composition	<code>aipd-legacy-bundled</code> retains the earlier topology	Use only as a bounded transition, not the target architecture
Connector placement	Mixed in-process and separate crates	One separately built and admitted host per public product	Provision host identities, credentials, health, and lifecycle independently
Routing	Local composition	Tenant-scoped registry, leases, replicas, routing, and remote dispatch	Deploy registry storage and reconcile admission before traffic
Operations	Existing daemon and CLI surface	Control plane, expanded <code>aipctl</code> , signed packages, and lifecycle workflows	Update operator roles, commands, and evidence procedures
Maintained public products	Five	Six, adding Twenty	Treat Twenty support as unreleased at this revision
Qualification	Release-era checks and historical reports	Fleet failure, scale, image, and product-oriented gates in source	A gate's presence is not a retained PASS for the reviewed revision

Follow the **bundled connector migration** before replacing a `v1.0.0` deployment. The migration must preserve tenant, credential, idempotency, approval, transaction, provider-effect, and recovery state; a successful process start is not sufficient.

Choose the correct source identity

Use `v1.0.0` when you need the released private source snapshot and its exact publication boundary. Use the full `97be86e9...` commit only when you intend to evaluate the documented but unreleased connector-fleet implementation.

For either choice:

1. Resolve the tag to its commit or select the complete commit directly.
2. Require a clean source tree and record its Git tree.
3. Record every built image or binary digest and generated manifest digest.
4. Match configuration, migration, connector upstream, and qualification

identities to that source.

5. Do not combine release notes, binaries, or qualification results from the two source states.

Known release-note gaps

- There is no tag after `v1.0.0` in the reviewed repository.
- The `CHANGELOG` contains no entry for the five post-tag commits.
- Workspace version `1.0.0` was not advanced after the tagged release.
- The current private-SDK publication manifest remains disabled pending its

registry bootstrap and verification prerequisites.

- No retained result in this review proves that the tagged release or pinned source passed every current release and live-product gate.
- Remote release publication, signatures, attestations, and registry contents were not inspected during this code-only documentation review.

Related documentation

- [Release artifacts and verification](#)
- [Bundled connector migration](#)
- [Implementation status](#)
- [Licensing](#)
- [Connector documentation](#)