

Hermes Agent capabilities

Use this index to select the Hermes operation that owns your task and read its AIP policy contract before invocation. Every configured endpoint publishes the same 35 operations under `cap:hermesagent::`.

SECTION	Connectors
TYPE	Connector
PROTOCOL	AIP 1.0
SOURCE	docs/connectors/hermes-agent/capabilities/README.md

The endpoint segment is an admitted routing identity. It is not an input field, display label, or caller-selected URL.

Catalogue identity

Property	Reviewed value
AIP source	97be86e9efedf07ecf1783b03800f683f107fb04
Pinned Hermes source	7426c09beee73bdf94d916015bac71384f6bc92
Connector profile	aip.connector.hermes_agent.v1
Capability prefix	cap:hermes_agent:
Operations per endpoint	35
Stable operations	31
Experimental operations	4
Read operations	14
Non-read operations	21
Approval required	11
Required idempotency key	15
Streaming or governed streaming	5

A different connector artifact, Hermes revision, or endpoint set requires a new manifest review. A matching operation suffix alone does not prove that a tenant can discover or invoke the capability.

Choose a family

Family	Focused page	Total	Read	Non-read	Low	Medium	High	Approval	Key	Stream
Health, models, and discovery	health-models-and-discovery.md	6	6	0	6	0	0	0	0	0
Chat and Responses API	chat-and-responses.md	6	1	5	1	4	1	1	1	2
Durable runs and approvals	durable-runs-and-approvals.md	5	2	3	2	3	0	2	3	1
Operator and delegation	operator-and-delegation.md	1	0	1	0	0	1	1	1	1
Sessions	sessions.md	9	3	6	3	5	1	1	4	1
Scheduled jobs	scheduled-jobs.md	8	2	6	2	4	2	6	6	0
Total		35	14	21	14	16	5	11	15	5

Key counts operations that require an AIP idempotency key. **Stream** counts four direct SSE operations plus the governed operator lifecycle.

Read a capability ID

For endpoint `primary`, a model-list capability has this shape:

```
TEXT
cap:hermes_agent:primary:models
■          ■      ■■ fixed operation
■          ■■■■■■■■ normalized endpoint ID
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■ connector namespace
```

Endpoint IDs are trimmed, restricted to ASCII letters, digits, hyphens, and underscores, and normalized to lowercase before manifest construction.

Every capability has one `aip.connector.hermes_agent.v1` binding. Its metadata contains the system, endpoint ID, operation, configured base URL, provider method, fixed path template, and streaming marker. Treat that metadata as deployment information, not authority to call the provider directly.

Operation catalogue

Legend: **Req** means required, **Opt** means optional, and **—** means not applicable. **Stream** describes the advertised AIP execution contract.

Operation suffix	Provider method and path	Kind	Risk	Approval	Key	Stream	Stability
<code>health</code>	<code>GET /health</code>	Tool	Low	—	Opt	—	Stable
<code>health_detailed</code>	<code>GET /health/detailed</code>	Tool	Low	—	Opt	—	Stable
<code>models</code>	<code>GET /v1/models</code>	Tool	Low	—	Opt	—	Stable
<code>capabilities</code>	<code>GET /v1/capabilities</code>	Tool	Low	—	Opt	—	Stable

Operation suffix	Provider method and path	Kind	Risk	Approval	Key	Stream	Stability
skills	GET /v1/skills	Tool	Low	—	Opt	—	Experimental
toolsets	GET /v1/toolsets	Tool	Low	—	Opt	—	Experimental
chat	POST /v1/chat/completions	Agent	Medium	—	Opt	—	Stable
chat_stream	POST /v1/chat/completions	Agent	Medium	—	Opt	Yes	Stable
responses	POST /v1/responses	Agent	Medium	—	Opt	—	Stable
responses_stream	POST /v1/responses	Agent	Medium	—	Opt	Yes	Stable
response_get	GET /v1/responses/{response_id}	Agent	Low	—	Opt	—	Experimental
response_delete	DELETE /v1/responses/{response_id}	Agent	High	Req	Req	—	Experimental
run_start	POST /v1/runs	Workflow	Medium	—	Req	—	Stable
run_status	GET /v1/runs/{run_id}	Workflow	Low	—	Opt	—	Stable
run_events	GET /v1/runs/{run_id}/events	Workflow	Low	—	Opt	Yes	Stable
run_approval	POST /v1/runs/{run_id}/approval	Workflow	Medium	Req	Req	—	Stable
run_stop	POST /v1/runs/{run_id}/stop	Workflow	Medium	Req	Req	—	Stable
operator	POST /v1/runs	Agent	High	Req	Req	Yes	Stable
sessions_list	GET /api/sessions	Tool	Low	—	Opt	—	Stable
session_create	POST /api/sessions	Tool	Medium	—	Req	—	Stable
session_get	GET /api/sessions/{session_id}	Tool	Low	—	Opt	—	Stable
session_patch	PATCH /api/sessions/{session_id}	Tool	Medium	—	Req	—	Stable
session_delete	DELETE /api/sessions/{session_id}	Tool	High	Req	Req	—	Stable
session_messages	GET /api/sessions/{session_id}/messages	Tool	Low	—	Opt	—	Stable
session_fork	POST /api/sessions/{session_id}/fork	Tool	Medium	—	Req	—	Stable
session_chat	POST /api/sessions/{session_id}/chat	Agent	Medium	—	Opt	—	Stable
session_chat_stream	POST /api/sessions/{session_id}/chat/stream	Agent	Medium	—	Opt	Yes	Stable
jobs_list	GET /api/jobs	Tool	Low	—	Opt	—	Stable

Operation suffix	Provider method and path	Kind	Risk	Approval	Key	Stream	Stability
job_create	POST /api/jobs	Tool	Medium	Req	Req	—	Stable
job_get	GET /api/jobs/{job_id}	Tool	Low	—	Opt	—	Stable
job_update	PATCH /api/jobs/{job_id}	Tool	Medium	Req	Req	—	Stable
job_delete	DELETE /api/jobs/{job_id}	Tool	High	Req	Req	—	Stable
job_pause	POST /api/jobs/{job_id}/pause	Tool	Medium	Req	Req	—	Stable
job_resume	POST /api/jobs/{job_id}/resume	Tool	Medium	Req	Req	—	Stable
job_run	POST /api/jobs/{job_id}/run	Agent	High	Req	Req	—	Stable

The catalogue contains 14 GET, 16 POST, 2 PATCH, and 3 DELETE operations. Sessions and jobs are executable Tool capabilities; they are not entries in the native AIP resource-read API.

Authentication legend

health is the only operation without an endpoint-auth requirement. The other 34 capability records declare Bearer auth sourced from endpoint.api_key.

The standalone host adds that key after endpoint and tenant selection. The AIP caller supplies its own gateway credential and never receives the endpoint key. The capability contract's credentials field is null because the provider token is startup-loaded rather than selected by each action.

Host route credential revision checks remain a separate authorization layer.

Kind and risk legend

AIP kind	Count	Meaning in this catalogue
Tool	20	Discovery, session management, or job management action
Workflow	5	Structured Hermes run lifecycle
Agent	10	Model work, stored response action, operator work, session chat, or immediate job run

Risk	Count	Required interpretation
Low	14	Read-only provider operation; still authenticate, authorize, and classify output
Medium	16	Provider or conversation state may change; preserve action identity and review impact
High	5	Operator execution, destructive deletion, or immediate job execution; apply the strictest policy

Risk is policy input, not authorization. A medium operation can require an idempotency key without requiring human approval, as run_start and several session operations do.

Approval legend

Approval is required for exactly 11 operations:

- `response_delete`;
- `run_approval` and `run_stop`;
- `operator`;
- `session_delete`;
- `job_create`, `job_update`, `job_delete`, `job_pause`, `job_resume`, and `job_run`.

Each published approval policy uses the tenant-policy selector, a 900000 ms validity window, and requires a reason, input snapshot, and policy decision. It grants no delegated authority by itself.

The direct `operator` capability always requires approval. First-class delegation can add its own connector-level approval gate; an exemption from that gate does not remove the capability's native approval contract.

Idempotency and retry legend

Fifteen operations require a capability-scoped idempotency key:

- `response_delete`;
- `run_start`, `run_approval`, `run_stop`, and `operator`;
- `session_create`, `session_patch`, `session_delete`, and `session_fork`;
- every job mutation: create, update, delete, pause, resume, and run.

The published key TTL is 86400000 ms. A collision returns the original result. Reuse a key only for the same canonical action contract; a returned original result is not proof that different input was accepted.

The 14 read-only operations advertise retry support and safe retry semantics. All 21 non-reads advertise no connector retry support and unsafe retry semantics, including chat operations whose idempotency key is optional. Preserve the original action ID and reconcile an uncertain provider outcome before any repeat.

Execution and SLA legend

Every operation supports synchronous invocation. Five also advertise async support:

- `chat_stream` and `responses_stream`;
- `run_start` and `run_events`;
- `operator`.

The four direct streams, plus `operator`, advertise streaming and cancellation. `session_chat_stream` is streaming and cancellable but does not advertise async support. Only `run_start`, `run_events`, and `operator` set the separate SLA `async_expected` marker.

Execution class	Expected latency	Timeout	Expected completion
Non-streaming operation	5000 ms	60000 ms	Synchronous
Direct SSE operation	30000 ms	300000 ms	Streaming
Governed operator	30000 ms	900000 ms	Streaming

All operations publish a 10000 ms maximum queue-delay hint and a 99.9% availability target. These are declared contract values, not observed qualification or a deployment guarantee.

Transaction and compensation legend

No Hermes capability publishes an AIP transaction contract. Required idempotency on state-changing operations does not add `dry_run`, `plan`, `commit`, or `reconcile` modes.

Every operation declares `rollback_not_supported` and names no compensating capability or window. Approval may be required for the original action, but no automatic inverse is promised.

Side-effect and data legend

Every operation declares external network access and read access. Additional markers are:

Marker	Operations
Write	Every non-read except the three destructive deletes
Delete	<code>response_delete</code> , <code>session_delete</code> , and <code>job_delete</code>
Send message	Chat, Responses API, session-chat, and their streaming forms
Code execution	<code>operator</code> only

The marker set describes possible effect classes. It does not prove that a model replied, a scheduled job delivered output, or an operator tool executed.

Data sensitivity	Count	Operations
Internal	15	Discovery plus other read and job-control metadata not classified higher
Confidential	15	Model work, run start/events/operator, session writes and chat, job create/update/run
Restricted	5	Response/session/job deletion and run approval/stop

All 21 non-read operations declare PII. Redaction is required for 29 operations: every capability except `health`, `health_detailed`, `models`, `capabilities`, `skills`, and `toolsets`.

These are minimum handling labels. Provider input or output may require a stricter tenant classification.

Stability legend

`skills`, `toolsets`, `response_get`, and `response_delete` are experimental. The other 31 operations are stable at the pinned connector revision.

Stable does not mean qualified against every Hermes build or model provider. Experimental operations require an explicit compatibility and rollout review.

Choose the next page

- Use `health-models-and-discovery.md` for the six non-mutating discovery contracts.
- Use `chat-and-responses.md` for OpenAI-compatible chat, Responses API, and stored-response operations.
- Use `durable-runs-and-approvals.md` for structured run start, observation, approval, and stop.
- Use `operator-and-delegation.md` for governed operator and first-class AIP

delegation.

- Use `sessions.md` for persisted conversations.
- Use `scheduled-jobs.md` for durable job definitions and controls.

Related documentation

- [Hermes Agent connector](#)
- [Hermes configuration](#)
- [Capabilities and contracts](#)
- [Approvals and policy](#)
- [Transactions and compensation](#)