

Dify knowledge pipelines and uploads

Use this family to discover datasource nodes, upload a local source, run one datasource node, or run a full Dify knowledge pipeline. The four operations use one workspace-scoped knowledge credential.

SECTION	Connectors
TYPE	Connector
PROTOCOL	AIP 1.0
SOURCE	docs/connectors/dify/capabilities/knowledge-pipelines-and-uploads.md

Pipeline inputs can fetch external content, invoke plugins, and create provider state. Approval must bind the dataset, pipeline version, source identities, credentials, inputs, and expected result handling.

Choose an operation

Operation suffix	Method and route	Provider output	Policy
<code>pipeline.datasource_plugin.list</code>	<code>GET /v1/datasets/{dataset_id}/pipeline/datasource-plugins</code>	JSON list	Low-risk read; retry supported
<code>pipeline.datasource_node.run</code>	<code>POST /v1/datasets/{dataset_id}/pipeline/datasource/nodes/{node_id}/run</code>	Provider SSE, buffered by connector	Medium mutation; approval and key required
<code>pipeline.run</code>	<code>POST /v1/datasets/{dataset_id}/pipeline/run</code>	Blocking JSON or provider SSE	Medium mutation; approval and key required
<code>pipeline.file.upload</code>	<code>POST /v1/datasets/pipeline/file-upload</code>	JSON upload record	Medium mutation; approval and key required

These Tool operations have knowledge scope and no app-mode projection. The capability template is:

```
TEXT
cap:dify:knowledge:<credential_id>:<operation>
```

The first three routes require a provider UUID dataset owned by the selected workspace. Pipeline file upload is workspace-scoped and has no dataset route value.

Discover datasource nodes

`pipeline.datasource_plugin.list` accepts optional query `is_published`. It defaults to true; false selects the current draft pipeline.

```
JSON
{
  "dataset_id": "0190c42f-2d5a-7000-8000-000000000801",
  "query": {
    "is_published": true
  }
}
```

Each returned entry can include `node_id`, plugin and provider identity, `datasource_type`, title, user input variables, and available datasource credentials. A credential entry can mark the provider default.

Discovery and execution must use the same published or draft selection. Persist the selected node, variable contract, datasource type, and credential identity before a run.

Upload one local pipeline file

`pipeline.file.upload` accepts one Action `file` and no `body`:

```
JSON
{
  "file": {
    "filename": "approved-source.pdf",
    "content_type": "application/pdf",
    "content_base64": "JVBERi0="
  }
}
```

The connector requires a safe basename, decodes the base64, and sends one multipart file. Its decoded upload limit is 32 MiB. Provider file type, tenant quota, and configured upload size can be stricter.

Diffy returns status 201 with `id`, `name`, `size`, `extension`, `mime_type`, `creator`, and creation time. Use that upload ID as the local-file reference in a later pipeline run.

Uploading does not attach the file to a dataset or prove that a pipeline has processed it. Retain the approved upload ID and apply the workspace's orphaned file policy after uncertain outcomes.

Run one datasource node

`pipeline.datasource_node.run` requires `node_id` and these body fields:

- `inputs`, containing the node's discovered variables
- `datasource_type`, selecting the source family
- `is_published`, selecting published or draft state
- optional `credential_id`, otherwise the provider default is used

Datasource type accepts `local_file`, `online_document`, `website_crawl`, or `online_drive`.

```
JSON
{
  "dataset_id": "0190c42f-2d5a-7000-8000-0000000000801",
  "node_id": "datasource-node-1",
  "body": {
    "inputs": {
      "source_id": "upload-0190"
    },
    "datasource_type": "local_file",
    "credential_id": "credential-0190",
    "is_published": true
  }
}
```

The provider always returns an event stream for this route. The connector declares the operation as synchronous JSON, reads the complete response, and places unparsed SSE text in the generic result `body`.

This operation does not emit AIP stream chunks and has no cancellation contract. The response-size limit applies to the entire buffered stream.

Run the full pipeline

`pipeline.run` requires `inputs`, `datasource_type`, `datasource_info_list`, `start_node_id`, `is_published`, and `response_mode`.

```
JSON
{
  "dataset_id": "0190c42f-2d5a-7000-8000-0000000000801",
  "body": {
    "inputs": {},
    "datasource_type": "local_file",
    "datasource_info_list": [
```

```

    {
      "reference": "upload-0190",
      "name": "approved-source.pdf"
    }
  ],
  "start_node_id": "datasource-node-1",
  "is_published": true,
  "response_mode": "blocking"
}
}

```

Use `blocking` for a structured JSON result through this connector version. If `streaming` is selected, the provider returns SSE, but the connector buffers it and exposes the complete event text as one generic result body.

Datasource items follow the selected family:

Type	Required item identity
<code>local_file</code>	<code>reference</code> ; <code>related_id</code> is a provider alias
<code>online_document</code>	<code>workspace_id</code> and <code>page.page_id</code> plus <code>page.type</code>
<code>website_crawl</code>	<code>url</code>
<code>online_drive</code>	<code>id</code> and <code>type</code> , where <code>type</code> is <code>file</code> or <code>folder</code>

Local file can add `name`. Online document can add page name and credential. Website crawl can add title. Online drive can add bucket and name.

Treat every remote URL, workspace, page, file, folder, bucket, plugin, and credential as untrusted until policy verifies its source and scope. The connector does not add datasource-specific allowlists.

Interpret the connector result

Blocking JSON returns through the generic envelope:

```

JSON
{
  "http_status": 200,
  "body": {
    "event": "workflow_finished",
    "data": {}
  },
  "content_type": "application/json",
  "provider_request_id": null
}

```

For provider SSE, `body` is a string containing the buffered event-stream text. Parse it only with a bounded SSE parser after verifying content type. Do not reinterpret that string as an AIP streaming execution.

The connector response limit defaults to 8 MiB and allows 1 to 64 MiB. An oversized or non-terminating provider stream fails the operation without a remote cancellation guarantee.

Apply mutation governance

All three mutations require a non-blank Action idempotency key of at most 512 bytes. AIP retains its tenant-scoped reservation for 24 hours and revalidates the input hash on collision.

The key does not prove provider deduplication. These operations publish no automatic retry, transaction, reconciliation, cancellation, or compensation support.

Treat uploads, external source references, credentials, node inputs, provider events, generated documents, and errors as confidential PII-bearing data. Redact them from general logs.

Recover safely

Observation	Decision
Dataset, node, input, or file is rejected before dispatch	Correct the bounded request without claiming a provider effect
Upload response is lost	Reconcile provider upload records before another upload
Node or credential differs from discovery	Refresh the same published or draft node list before execution
Buffered SSE exceeds the response bound	Treat the outcome as unknown; do not replay automatically
Pipeline streaming connection does not terminate	Preserve timing and provider identity, then inspect dataset state
Blocking run loses its response	Inspect provider pipeline and generated document state before another run
Remote source fails policy	Reject the run before dispatch and preserve the authorization decision
Draft execution reaches a production workflow	Stop traffic and treat it as a release-boundary incident
Returned content crosses a workspace or dataset boundary	Stop traffic and treat it as an isolation incident

Preserve the Action ID, digest, approval, idempotency key, credential, dataset, pipeline version, node, source identities, upload ID, and provider evidence needed to settle an ambiguous outcome.

Related documentation

- [Dify capability index](#)
- [Dify knowledge documents and indexing](#)
- [Dify connector configuration](#)
- [Authentication and credential scopes](#)
- [Errors and retry decisions](#)