

# Twenty workspace metadata

Use these five operations to read or change one of twelve fixed Twenty metadata resource types. The connector selects the route from a closed resource enum and applies resource-specific list-query guards.

|          |                                                           |
|----------|-----------------------------------------------------------|
| SECTION  | Connectors                                                |
| TYPE     | Connector                                                 |
| PROTOCOL | AIP 1.0                                                   |
| SOURCE   | docs/connectors/twenty/capabilities/workspace-metadata.md |

Metadata changes can alter schema, views, layouts, and webhook delivery policy. All three mutations are critical, approval-required, idempotency-required, unsafe to retry, and unsupported by rollback or compensation.

## Supported resources

| Resource          | Purpose                                         | List query                           |
|-------------------|-------------------------------------------------|--------------------------------------|
| objects           | Object definitions                              | limit, starting_after, ending_before |
| fields            | Field definitions                               | limit, starting_after, ending_before |
| views             | Saved views                                     | Optional objectMetadataId            |
| viewFields        | View field membership                           | Optional viewId                      |
| viewFilters       | View filter definitions                         | Optional viewId                      |
| viewGroups        | View grouping definitions                       | Optional viewId                      |
| viewSorts         | View sort definitions                           | Optional viewId                      |
| viewFilterGroups  | Nested view filter groups                       | Optional viewId                      |
| pageLayouts       | Record, index, dashboard, or standalone layouts | Optional object and type pair        |
| pageLayoutTabs    | Tabs under one layout                           | Required pageLayoutId                |
| pageLayoutWidgets | Widgets under one tab                           | Required pageLayoutTabId             |
| webhooks          | Provider webhook registrations                  | No query                             |

Any other resource value fails before provider dispatch. The connector does not expose a caller-selected metadata path.

## Compare the operations

| Operation                    | Provider request                      | Risk     | Required input            |
|------------------------------|---------------------------------------|----------|---------------------------|
| <code>metadata.list</code>   | GET<br>/rest/metadata/{resource}      | Low      | resource                  |
| <code>metadata.get</code>    | GET /rest/metadata/{resource}/{id}    | Low      | resource, id              |
| <code>metadata.create</code> | POST<br>/rest/metadata/{resource}     | Critical | resource, object body     |
| <code>metadata.update</code> | PATCH /rest/metadata/{resource}/{id}  | Critical | resource, id, object body |
| <code>metadata.delete</code> | DELETE /rest/metadata/{resource}/{id} | Critical | resource, id              |

The two reads need no approval, accept an optional key, and advertise safe retry. Every mutation requires approval and a key and advertises unsafe retry.

All five support synchronous execution only. None supports asynchronous execution, streaming, cancellation, transactions, reconciliation, or compensation.

## List metadata

`cap:twenty:metadata.list` accepts the selected resource and only its allowed query keys.

List object definitions with bounded pagination:

```
JSON
{
  "resource": "objects",
  "query": {
    "limit": 50,
    "starting_after": "0190c42f-2d5a-7000-8000-000000000050"
  }
}
```

`limit` accepts 0 through 200. Both cursors must be RFC 4122 UUIDs with version 1 through 5 when present.

List view fields for one view:

```
JSON
{
  "resource": "viewFields",
  "query": {
    "viewId": "0190c42f-2d5a-7000-8000-000000000051"
  }
}
```

`viewId` is optional for the five view-child resource types. An omitted value can yield a wider provider result; deployment policy should require the narrowest useful scope.

List page layouts for an object and type:

```
JSON
{
  "resource": "pageLayouts",
  "query": {
    "objectMetadataId": "0190c42f-2d5a-7000-8000-000000000052",
    "pageLayoutType": "RECORD_PAGE"
  }
}
```

Allowed layout types are `RECORD_INDEX`, `RECORD_PAGE`, `DASHBOARD`, and `STANDALONE_PAGE`. A supplied type requires `objectMetadataId` because the pinned provider otherwise ignores it.

`pageLayoutTabs` requires `pageLayoutId`. `pageLayoutWidgets` requires `pageLayoutTabId`. Both values must pass the UUID validator.

The `webhooks` resource accepts no query. Unknown or resource-inapplicable keys fail before provider dispatch.

## Get one metadata resource

`cap:twenty:metadata.get` accepts a supported resource and UUID:

```
JSON
{
  "resource": "fields",
  "id": "0190c42f-2d5a-7000-8000-000000000053"
}
```

No query object is accepted. The connector sends the fixed workspace token and returns provider JSON without resource-specific normalization.

## Create metadata

`cap:twenty:metadata.create` accepts a supported resource and object body:

```
JSON
{
  "resource": "views",
  "body": {
    "name": "AIP Review View",
    "objectMetadataId": "0190c42f-2d5a-7000-8000-000000000052"
  }
}
```

The outer schema permits at most 2,048 top-level body properties, and the serialized request cannot exceed 16 MiB. Twenty validates resource-specific fields and relationships.

Create requires critical-action approval and a stable idempotency key. The connector does not synthesize metadata IDs or force provider upsert.

## Update metadata

`cap:twenty:metadata.update` requires resource, target UUID, and object body:

```
JSON
{
  "resource": "views",
  "id": "0190c42f-2d5a-7000-8000-000000000054",
  "body": {
    "name": "AIP Approved View"
  }
}
```

No query object is accepted. The connector does not reject an empty body at its outer layer; policy should require a reviewed, meaningful change.

## Delete metadata

`cap:twenty:metadata.delete` accepts only resource and target UUID:

```
JSON
{
  "resource": "views",
  "id": "0190c42f-2d5a-7000-8000-000000000054"
}
```

Deletion is critical and has no connector rollback. Preflight dependent fields, views, layout children, automations, and webhook ownership through current provider reads before approval.

## Handle webhook metadata secrets

The `webhooks` metadata resource has an additional secret boundary. An Action body cannot contain a field named `secret` for create or update.

Create one webhook registration without a secret field:

```
JSON
{
  "resource": "webhooks",
  "body": {
    "targetUrl": "https://connector.example.test/webhooks/twenty"
  }
}
```

Create rejects `rotateConfiguredSecret` and injects the current host-owned webhook secret. The operation fails when webhook security is not configured on the host.

Request controlled secret rotation on an existing provider registration:

```
JSON
{
  "resource": "webhooks",
  "id": "0190c42f-2d5a-7000-8000-000000000055",
  "body": {
    "rotateConfiguredSecret": true
  }
}
```

Update removes the control field before provider dispatch and injects the configured secret only when its value is Boolean `true`. A value of `false`, another type, or a caller-supplied `secret` is invalid.

When the resource is `webhooks`, the connector recursively removes every response field named `secret`. This redaction applies to list, get, create, update, and delete results.

The redaction protects the provider response path. It does not replace log, trace, database, secret-manager, or evidence redaction outside the connector.

## Apply approval and idempotency

Create, update, and delete require a tenant-policy decision with reason, input snapshot, and policy decision. The approval validity window is 900,000 ms.

Mutation keys accept 1 through 1,024 ASCII graphic bytes. The scope is the external account, collision behavior revalidates the input hash, and the published TTL is 604,800,000 ms.

Bind approval and key to the exact resource, ID, body, fixed workspace, credential revision, and expected dependency impact. Secret bytes remain outside approval input and evidence.

## Interpret results and failures

A completed result contains provider `http_status`, optional `provider_request_id`, and decoded provider body. The connector records its provider-effect-committed checkpoint after a successful mutation response.

| Condition                                                           | Read decision               | Mutation decision                         |
|---------------------------------------------------------------------|-----------------------------|-------------------------------------------|
| Invalid resource, UUID, query, body, secret field, or rotation flag | Correct input               | Correct input under a new governed intent |
| Provider 401 or 403                                                 | Repair fixed credential     | Repair binding; do not replay             |
| Provider 429 or 5xx                                                 | Retryable                   | Unsafe and uncertain                      |
| Transport failure                                                   | Retryable                   | Unsafe; may be uncertain after connection |
| Oversized or invalid provider response                              | Not automatically retryable | Unsafe and uncertain                      |

All three metadata mutations advertise unsafe retry. Preserve original Action, key, approval, and provider request identity and inspect current metadata before any recovery mutation.

## Security and data boundaries

All five contracts classify data as confidential, possibly containing personal information, with redaction required. Metadata can disclose tenant schema, identifiers, saved views, layouts, and event destinations.

The connector does not implement field-level metadata authorization, dependency analysis, migration ordering, provider rollback, or workspace backup. Apply those controls in deployment policy and the application workflow.

## Related documentation

- [Twenty capability index](#)
- [Twenty connector configuration](#)
- [Authentication and workspace isolation](#)
- [Approvals and policy](#)
- [Errors and retry decisions](#)