Core API

Registry v4.0.0 introduces the *Core API*, a collection of higher level APIs that provide transaction-oriented operations, rather than the lower level, model-oriented REST API v1. The Core API is implemented as a standard Plugin that is enabled by default.

Enabling Core APIs

Core APIs must be enabled, and each instantiation is associated with an (unprivileged) API User. Before continuing, create any desired API User(s). To instantiate an API, go to *Configuration* > *Core APIs* > *Add a New Core API*. Select the desired API and API User.

Core CO Person APIs are accessed using an Identifier on the CO Person record, not the internal CO Person ID that shows up (for example) in the CoPerson API. This implies an Identifier of the specified type is available on all relevant CO Person records, see Extended Types and Configuring Registry Identifier Assignment for possibly helpful resources. Select the desired Identifier Type in the Core API configuration.

It is possible to create multiple instantiations, for example to

- Grant access to a Core API to multiple API Users
- Grant access to multiple Core APIs to the same API User
- Expose the same records using multiple Identifier Types

When a request is processed, if there is more than one valid configuration for the authenticated API User, it is non-deterministic as to which configuration will be used.

Granting an API User access to the Write API will also grant access to the Read API. Do not create two entries for the same API User for each API, as Write access may be denied depending on which configuration is returned first by the database.

Authentication

Core APIs use Basic Auth, using the API Username and API Key as created for the API User.

CO Person Read API

The CO Person Read API is a collection of two APIs that create a consolidated view of most attributes associated with a CO Person, including

- CoGroupMember
- CoPersonRole
  - Address
  - AdHocAttribute
  - TelephoneNumber
- EmailAddress
- Identifier
- Name
- OrgIdentity
  - Address
  - EmailAddress
  - Identifier
  - Name
  - TelephoneNumber
  - Url
- Url
- Authenticators
- Clusters

The data returned by the Index API can be configured via the *Response Type* configuration:

- **Full Profile**: All available attributes (defined above) are returned.
- **Identifier Only**: Only the Identifier of the configured type is returned. The full record may be obtained using the Read API.

The Index API is available as of Registry v4.1.0.

The Core APIs do not support Extended Attributes, which are deprecated. AdHoc Attributes should be used instead. Alternately, the REST API v1 can be used to manage Extended Attributes.

Read API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/people/`***`identifier`*** where • **coid**: CO ID • **identifier**: CO Person Identifier, of the configured type

Index API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`**`coid`**`/core/v1/people[?&limit=`**`limit`**`&page=`**`page`**`&direction=`**`dir`**`]` where • **coid**: CO ID • **direction**: *asc* (return older records first) or *desc* (return newer records first) • **limit**: The maximum number of records to return in the response (see note below) • **page**: Return this page of the result set

Message Format

The response is a JSON document as described by this JSON Schema.

Identifier via Query Parameter

As of Registry v4.1.0, identifiers may be provided to the read API via query parameters. Effectively, all read requests will be treated as *Index* operations, though when an identifier is provided at most one record will be returned.

```
GET https://$SERVER/registry/api/co/<coid>/core/v1/people?identifier=<foo>
```

CO Person Write API

The CO Person Write API is a collection of three APIs allow creation, updating, and deletion of CO Person records.

> Create and Delete support are available as of Registry v4.1.0.

The Create and Update APIs accept the same attributes as returned by the Read API. Note that most metadata returned by the Read API is not permitted (and will be ignored) in the Write API. The exception is the sub-element id, which when present is used to indicated the update of an existing sub-element (such as an EmailAddress) rather than the creation of a new one.

The behavior of the Delete API can be configured via the Core API configuration. If *Expunge on Delete* is enabled, a Delete action will be treated as if the record were Expunged via the UI. Otherwise, all CoPersonRoles and the CoPerson record itself will be set to Deleted status.

> The Write API cannot be used to update attributes that are not normally modifiable via other mechanisms, including
>
> - Automatic Group Memberships
> - Attributes attached to Org Identities created from Organizational Identity Sources

Create API Endpoint

Request Method	`POST`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/people` where • **coid**: CO ID

Delete API Endpoint

Request Method	`DELETE`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/people/`***`identifier`*** where • **coid**: CO ID • **identifier**: CO Person Identifier, of the configured type

Update API Endpoint

Request Method	`PUT`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/people/`***`identifier`*** where • **coid**: CO ID • **identifier**: CO Person Identifier, of the configured type

Message Format

For the Create and Update APIs, the request is sent with `Content-Type` of `application/json` with a body containing a JSON document as described by this JSON Schema. An empty body or document is not valid for Create or Update. There is no request body for the Delete API.

The Create request will include the newly assigned Identifier of the configured type in the response body.

Identifier via Query Parameter

As of Registry v4.1.0, identifiers may be provided to the write API via query parameters.

```
PUT https://$SERVER/registry/api/co/<coid>/core/v1/people?identifier=<foo>
DELETE https://$SERVER/registry/api/co/<coid>/core/v1/people?identifier=<foo>
```

Department Read API

The Department Read API is a collection of two APIs that create a consolidated view of most attributes associated with a Department. The Department Read API is available as of Registry v4.3.0.

Supported attributes include:

- Address
- AdHocAttribute
- EmailAddress
- Identifier
- TelephoneNumber
- Url

The data returned by the Index API can be configured via the *Response Type* configuration:

- **Full Profile**: All available attributes (defined above) are returned.
- **Identifier Only**: Only the Identifier of the configured type is returned. The full record may be obtained using the Read API.

Read API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/departments/`***`identifier`*** where • **coid**: CO ID • **identifier**: Department Identifier, of the configured type
Alternate URL	`GET https://$SERVER/registry/api/co/`***`coid`***`/core/v1/departments?identifier=`***`identifier`***

Index API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`**`coid`**`/core/v1/departments[?&limit=`**`limit`**`&page=`**`page`**`&direction=`**`dir`**`]` where • **coid**: CO ID • **direction**: *asc* (return older records first) or *desc* (return newer records first) • **limit**: The maximum number of records to return in the response (see note below) • **page**: Return this page of the result set

Message Format

The response is a JSON document as described by this JSON Schema.

Department Write API

The Department Write API is a collection of three APIs allow creation, updating, and deletion of Department records. The Department Write API is available as of Registry v4.3.0.

The Create and Update APIs accept the same attributes as returned by the Read API. Note that most metadata returned by the Read API is not permitted (and will be ignored) in the Write API. The exception is the sub-element id, which when present is used to indicated the update of an existing sub-element (such as an EmailAddress) rather than the creation of a new one.

Create API Endpoint

Request Method	`POST`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/departments` where • **coid**: CO ID

Delete API Endpoint

Request Method	`DELETE`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/departments/`***`identifier`*** where • **coid**: CO ID • **identifier**: Department Identifier, of the configured type

Update API Endpoint

Request Method	`PUT`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/departments/`***`identifier`*** where • **coid**: CO ID • **identifier**: Department Identifier, of the configured type

Message Format

For the Create and Update APIs, the request is sent with `Content-Type` of `application/json` with a body containing a JSON document as described by this JSON Schema. An empty body or document is not valid for Create or Update. There is no request body for the Delete API.

The Create request will include the newly assigned Identifier of the configured type in the response body.

Match Callback API

The Match Callback API implements the COmanage Match Endpoint Notification Protocol in order to automate the reprocessing of records following a Match resolution. For more information, see Integrating With ID Match.

Organization Read API

The Organization Read API is a collection of two APIs that create a consolidated view of most attributes associated with an Organization. The Organization Read API is available as of Registry v4.3.0.

Supported attributes include:

- Address
- AdHocAttribute
- EmailAddress
- Identifier
- TelephoneNumber
- Url

The data returned by the Index API can be configured via the *Response Type* configuration:

- **Full Profile**: All available attributes (defined above) are returned.
- **Identifier Only**: Only the Identifier of the configured type is returned. The full record may be obtained using the Read API.

Read API Endpoint

Request Method	`GET`

URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/organizations/`***`identifier`*** where • **coid**: CO ID • **identifier**: Organization Identifier, of the configured type
Alternate URL	`GET https://$SERVER/registry/api/co/`***`coid`***`/core/v1/organizations?identifier=`***`identifier`***

Index API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`**`coid`**`/core/v1/organizations[?&limit=`**`limit`**`&page=`**`page`**`&direction=`**`dir`**`]` where • **coid**: CO ID • **direction**: *asc* (return older records first) or *desc* (return newer records first) • **limit**: The maximum number of records to return in the response (see note below) • **page**: Return this page of the result set

Message Format

The response is a JSON document as described by this JSON Schema.

Organization Write API

The Organization Write API is a collection of three APIs allow creation, updating, and deletion of Organization records. The Organization Write API is available as of Registry v4.3.0.

The Create and Update APIs accept the same attributes as returned by the Read API. Note that most metadata returned by the Read API is not permitted (and will be ignored) in the Write API. The exception is the sub-element id, which when present is used to indicated the update of an existing sub-element (such as an EmailAddress) rather than the creation of a new one.

Create API Endpoint

Request Method	`POST`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/organizations` where • **coid**: CO ID

Delete API Endpoint

Request Method	`DELETE`
URL	`https://$SERVER/registry/api/co/`***`coid`***`/core/v1/organizations/`***`identifier`*** where • **coid**: CO ID • **identifier**: Organization Identifier, of the configured type

Update API Endpoint

Request Method	`PUT`

URL	`https://$SERVER/registry/api/co/`***coid***`/core/v1/organizations/`***identifier*** where • **coid**: CO ID • **identifier**: Organization Identifier, of the configured type

Message Format

For the Create and Update APIs, the request is sent with `Content-Type` of `application/json` with a body containing a JSON document as described by this JSON Schema. An empty body or document is not valid for Create or Update. There is no request body for the Delete API.

The Create request will include the newly assigned Identifier of the configured type in the response body.

Petition Read API

The Petition Read API creates a consolidated view of most attributes associated with a CO Petition. The Petition Read API is available as of Registry v4.3.0.

Supported attributes include:

- ApproverCoPerson
- CoEnrollmentFlow
- Cou
- EnrolleeCoPerson
- SponsorCoPerson
- CoInvite
- VettingRequest

Read API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`***coid***`/core/v1/petitions/:`***id*** where • **coid**: CO ID • **id**: Petition ID

Index API Endpoint

Request Method	`GET`
URL	`https://$SERVER/registry/api/co/`**coid**`/core/v1/petitions[?status=`**status**`&couid=`**COU ID**`&limit=`**limit**`&page=`**page**`&direction=`**dir**`]` where • **coid**: CO ID • **couid**: COU ID • **status**: [A | Y | C | CR | X | N | D2 | F | PA | PC | PV] • **direction**: *asc* (return older records first) or *desc* (return newer records first) • **limit**: The maximum number of records to return in the response (see note below) • **page**: Return this page of the result set

Message Format

The response is a JSON document as described by this JSON Schema.

Implementation Notes

Pagination Limits

As of Registry v4.2.0, the maximum value for the `limit` parameter for `index` requests is 1000. Prior to Registry v4.2.0, the maximum value was 100.