

master document for generating PDF

= Genomic Data Management (GDM)

Table of Contents

Overview.....	1
Version information	1
License information	1
URI scheme	1
Tags	1
Overview.....	2
Language and Framework.....	3
The DTO Payload	3
Request/Response Structure	3
Authentication	5
Error Handling	5
Paths	6
POST /analyses	6
GET /analyses	6
GET /analyses/{analysisId}.....	7
PUT /analyses/{analysisId}.....	7
POST /auth	8
GET /configsettings.....	9
GET /contacts-search	9
POST /contacts.....	10
GET /contacts.....	10
GET /contacts/{contactId}.....	11
PUT /contacts/{contactId}.....	11
GET /contacts/{email}	12
GET /cvgroups/{cvGroupId}/cvs	13
GET /cvgroups/{cvGroupTypeId}	13
Get /cvgroups/{groupName}	14
POST /cvs	14
GET /cvs	15
GET /cvs/{cvId}	15
PUT /cvs/{cvId}	16
DELETE /cvs/{cvId}	17
GET /cvs/{groupName}	17
POST /datasets.....	18
GET /datasets.....	18
GET /datasets/types	19
GET /datasets/types/{id}	19
GET /datasets/{dataSetId}.....	20

PUT /datasets/{dataSetId}	21
GET /datasets/{dataSetId}/analyses	21
GET /datasets/{datasetId}/jobs	22
POST /displays	22
GET /displays	23
GET /displays/{displayId}	23
PUT /displays/{displayId}	24
GET /entities	25
Gets /entities/{entityNameParent}/{parentId}/{entityNameChild}/count	25
GET /entities/{entityName}/count	26
GET /entities/{entityName}/lastmodified	26
POST /experiments	27
GET /experiments	27
GET /experiments/{experimentId}	28
PUT /experiments/{experimentId}	29
GET /experiments/{experimentId}/protocols	29
GET /files/loader/{directoryName}	30
PUT /files/loader/{directoryName}	30
POST /files/{destinationType}	31
DELETE /files/{destinationType}	32
POST /files/{gobiiJobId}/{destinationType}	32
GET /files/{gobiiJobId}/{destinationType}	33
POST /instructions/extractor	34
GET /instructions/extractor/jobs/{jobName}	34
GET /instructions/extractor/{instructionFileName}	35
POST /instructions/loader	36
GET /instructions/loader/jobs/{jobName}	36
GET /instructions/loader/{instructionFileName}	37
POST /jobs	37
GET /jobs	38
POST /jobs/dnasamples/{jobName}	38
GET /jobs/{jobName}	39
PUT /jobs/{jobName}	40
POST /manifests	40
GET /manifests	41
GET /manifests/{manifestId}	41
PUT /manifests/{manifestId}	42
GET /maps	42
POST /mapsets	43
GET /mapsets	43
GET /mapsets/{mapsetId}	44

PUT /mapsets/{mapsetId}	44
GET /marker-search	45
POST /markergroups	46
GET /markergroups	46
GET /markergroups/{markerGroupId}	47
PUT /markergroups/{markerGroupId}	47
POST /markers	48
GET /markers	48
GET /markers/{markerId}	49
PUT /markers/{markerId}	49
POST /names/{entity}	50
GET /names/{entity}	51
POST /organizations	52
GET /organizations	52
GET /organizations/{organizationId}	53
PUT /organizations/{organizationId}	53
POST /platforms	54
GET /platforms	55
GET /platforms/protocols/{vendorProtocolId}	55
GET /platforms/{platformId}	56
PUT /platforms/{platformId}	56
GET /projects	57
GET /projects	57
GET /projects/{projectId}	58
PUT /projects/{projectId}	58
POST /protocols	59
GET /protocols	60
GET /protocols/{protocolId}	60
PUT /protocols/{protocolId}	61
POST /protocols/{protocolId}/vendors	61
GET /protocols/{protocolId}/vendors	62
PUT /protocols/{protocolId}/vendors	62
POST /references	63
GET /references	64
GET /references/{referenceId}	64
PUT /references/{referenceId}	65
/restprofiles/	65
Definitions	66
AnalysisDTO	66
ConfigSettingsDTO	67
ContactDTO	67

CvDTO	68
CvGroupDTO	69
DataSetDTO	69
DisplayDTO	71
DnaSampleDTO	72
EntityPropertyDTO	73
EntityStatsDTO	74
ExperimentDTO	74
ExtractorInstructionFilesDTO	75
File	75
GobiiDataSetExtract	76
GobiiExtractorInstruction	78
GobiiFile	78
GobiiFileColumn	79
GobiiLoaderInstruction	80
Header	82
HeaderAuth	82
HeaderStatusMessage	83
InputStream	83
InputStreamResource	83
JobDTO	83
Link	84
LinkCollection	84
LoaderFilePreviewDTO	84
LoaderInstructionFilesDTO	85
ManifestDTO	85
MapsetDTO	86
MarkerDTO	87
MarkerGroupDTO	88
MarkerGroupMarkerDTO	88
NameIdDTO	89
OrganizationDTO	90
Pagination	90
PayloadAnalysisDTO	91
PayloadConfigSettingsDTO	91
PayloadContactDTO	91
PayloadCvDTO	91
PayloadCvGroupDTO	91
PayloadDataSetDTO	92
PayloadDisplayDTO	92
PayloadDnaSampleDTO	92

PayloadEntityStatsDTO	92
PayloadEnvelopeAnalysisDTO	92
PayloadEnvelopeConfigSettingsDTO	93
PayloadEnvelopeContactDTO	93
PayloadEnvelopeCvDTO	93
PayloadEnvelopeCvGroupDTO	93
PayloadEnvelopeDataSetDTO	93
PayloadEnvelopeDisplayDTO	94
PayloadEnvelopeDnaSampleDTO	94
PayloadEnvelopeEntityStatsDTO	94
PayloadEnvelopeExperimentDTO	94
PayloadEnvelopeExtractorInstructionFilesDTO	94
PayloadEnvelopeJobDTO	95
PayloadEnvelopeLoaderFilePreviewDTO	95
PayloadEnvelopeLoaderInstructionFilesDTO	95
PayloadEnvelopeManifestDTO	95
PayloadEnvelopeMapsetDTO	95
PayloadEnvelopeMarkerDTO	96
PayloadEnvelopeMarkerGroupDTO	96
PayloadEnvelopeNameIdDTO	96
PayloadEnvelopeOrganizationDTO	96
PayloadEnvelopePlatformDTO	96
PayloadEnvelopeProjectDTO	97
PayloadEnvelopeProtocolDTO	97
PayloadEnvelopeReferenceDTO	97
PayloadEnvelopeRestProfileDTO	97
PayloadExperimentDTO	97
PayloadExtractorInstructionFilesDTO	98
PayloadJobDTO	98
PayloadLoaderFilePreviewDTO	98
PayloadLoaderInstructionFilesDTO	98
PayloadManifestDTO	98
PayloadMapsetDTO	99
PayloadMarkerDTO	99
PayloadMarkerGroupDTO	99
PayloadNameIdDTO	99
PayloadOrganizationDTO	99
PayloadPlatformDTO	100
PayloadProjectDTO	100
PayloadProtocolDTO	100
PayloadReferenceDTO	100

PayloadRestProfileDTO	100
PlatformDTO	101
ProjectDTO	102
PropNameId	102
ProtocolDTO	103
ReferenceDTO	104
RestProfileDTO	104
ServerConfigItem	105
Status	106
TableColDisplay	106
VcfParameters	106
VendorProtocolDTO	106

Overview

The GDM web service API provides a comprehensive set of RESTful methods for accomplishing the following core tasks in the GOBii Genomic Data Management (GDM) system. This manual explains the GDM web architecture and its web service operations.

Version information

Version : 1.0

License information

License : Apache 2.0

License URL : <http://www.apache.org/licenses/LICENSE-2.0.html>

Terms of service : null

URI scheme

Host : localhost:8081

BasePath : /api

Schemes : HTTP

Tags

- Analyses
- Authentication
- ConfigSettings
- Contacts
- ContactsSearch
- ControlledVocabularies
- CvGroups
- CvGroups.cvs
- Cvs
- Datasets
- Datasets.analyses
- Datasets.jobs
- Datasets.types
- Displays
- Entities
- Entities.count

- Entities.lastModified
- Entities.parent.child.count
- Experiments
- Experiments.protocols
- Files
- Files.jobs
- Files.loader
- Instructions.extractor
- Instructions.extractor.jobs
- Instructions.loader
- Instructions.loader.jobs
- Jobs
- Jobs.dnasamples
- Manifests
- Maps
- Mapsets
- MarkerGroups
- MarkerSearch
- Markers
- Names
- Organizations
- Platform.protocols
- Platforms
- Projects
- Protocols
- Protocols.vendor
- References
- RestProfiles

Overview

GOBii's internal documentation explains in detail how the web service infrastructure is architected. It is not necessary for a consumer of the GOBii web API to understand this in great detail. However, knowing a few things about it will make it easier to understand the web method documentation.

Language and Framework

GOBii Web Services are implemented in Java with the Spring framework. Virtually all calls conform to the same basic structure. For example, the /analyses POST method is as follows:

```
@RequestMapping(value = "/analyses", method = RequestMethod.POST)
@ResponseBody
public PayloadEnvelope<AnalysisDTO> createAnalysis(@ApiParam(required = true)
@RequestBody PayloadEnvelope<AnalysisDTO> analysisPostEnvelope,
                                                    HttpServletRequest request,
                                                    HttpServletResponse response) {

    PayloadEnvelope<AnalysisDTO> returnVal = new PayloadEnvelope<>();

    // populate returnVal . . .

    return returnVal;
}
```

The request and response are almost always wrapped in a type-parameterized PayloadEnvelope. In this case, the PayloadEnvelope is type-parameterized for the AnalysisDTO. Other methods are type parameterized according to the entity that they touch (e.g., the implementation of the /datasets POST method is type-parameterized for the DataSetDTO).

The DTO Payload

The DTO objects are lightweight classes that contain simple data properties. They correspond roughly but not exactly to entities in the database (for example, in some cases a DTO defines a child collection of foreign-keyed entities).

For a client of this application, however, the strongly-typed nature of the implementation is irrelevant: all request responses and bodies are conveyed in plain JSON, and it is the application's job to serialize and deserialize accordingly. What a client implementation must take into account, however, is the structure of the DTO that pertains to a specific call. From the reference section for each web method in this document, you can navigate to the DTO definition.

Request/Response Structure

The following illustrates the http body structure encompassed by the PayloadEnvelope. As mentioned above, it is the same for almost all responses as well as for the body of POST and PUT requests.

```

{
  "payload":{
    "linkCollection":{
      "linksPerDataItem":[

      ],
      "exploreLinksPerDataItem":[

      ]
    },
    "data":[

    ]
  },
  "header":{
    "gobiiProcessType":"",
    "dtoHeaderAuth":{
      "userName":null,
      "password":null,
      "token":"",
      "gobiiCropType":""
    },
    "status":{
      "succeeded":true,
      "statusMessages":[

      ],
      "statusMessagesByCode":{

      }
    },
    "pagination":null,
    "gobiiVersion":"2.0_2018-07-31",
    "cropType":""
  }
}

```

For requests, the following keys can be omitted:

- payload.linkCollection
- header

The content of the "data" array will vary depending on how the call is type-parameterized. For example, for a POST to the the /project resource, it will contain a project object, as defined by ProjectDTO.

Authentication

All calls except for /configsettings GET require a valid authentication token. The token can be acquired via a POST to the /auth resource with an empty body and the following request headers set:

- X-Username
- X-Password

The specified header values must, of course, be a valid GOBii username and password. Both values are expected to be unencrypted (https will provide encryption for the overall request stream).

GOBii users must be defined in the system's LDAP and in the GOBii contact table. Accordingly, there can be two types of authentication failures:

- 401 Unauthorized: the user does not authenticate to LDAP;
- 403 Forbidden: the contact record does not exist. In this case, the error will indicate "Missing contact info for user <user-name> in crop database <crop-db>; a contact record must have username = <user-name>";

When authentication succeeds, the response headers will contain an X-Auth-Token key whose value is a valid token. For subsequent requests, it is sufficient to supply X-Auth-Token in the request headers with the token that was received from the /auth POST. Some browsers prevent the client code from reading custom response headers. For such cases, the token value can also be retrieved from the "token" key in the "dtoHeaderAuth" object.

Error Handling

Errors internal to GOBii are reported in the "header" key accompanying the response. An example of how to retrieve error data is provided in the following code fragment.

```
if (!header.getStatus().isSucceeded() ||
    header
        .getStatus()
        .getStatusMessages()
        .stream()
        .filter(headerStatusMessage ->
headerStatusMessage.getGobiiStatusLevel().equals(GobiiStatusLevel.VALIDATION))
        .count() > 0) {
    returnVal = true;
    System.out.println();
    System.out.println("*** Header errors: ");
    for (HeaderStatusMessage currentStatusMesage :
header.getStatus().getStatusMessages()) {
        System.out.println(currentStatusMesage.getMessage());
    }
}
```

Paths

POST /analyses

POST /gobii/v1/analyses

Description

Create analysis entity for GDM system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeAnalysisDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Tags

- Analyses

GET /analyses

GET /gobii/v1/analyses

Description

List of all Analysis entities.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Tags

- Analyses

GET /analyses/{analysisId}

```
GET /gobii/v1/analyses/{analysisId}
```

Description

Get analyses by analyses Id.

Parameters

Type	Name	Description	Schema
Path	analysisId —	ID of Analysis to be extracted.	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Tags

- Analyses

PUT /analyses/{analysisId}

```
PUT /gobii/v1/analyses/{analysisId}
```

Description

Updates the Analysis entity having the specified analysisId.

Parameters

Type	Name	Description	Schema
Path	analysisId —	ID of Analysis to be updated	integer (int32)
Body	body —		PayloadEnvelopeAnalysisDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Tags

- Analyses

POST /auth

```
POST /gobii/v1/auth
```

Description

The user credentials are specified in the request headers X-Username and X-Password; the response and the response headers include the token in the X-Auth-Token header. X-Auth-Token header and value obtained from /auth call will be used as an API-key for the rest of the GDM calls.

Parameters

Type	Name	Description	Schema
Header	X-Password —	User password	string
Header	X-Username —	User Identifier	string
Body	body —		string

Responses

HTTP Code	Description	Schema
200	OK Headers : `X-Auth-Token` (string) : API key to authenticate GDM api calls.	string

Tags

- Authentication

GET /configsettings

```
GET /gobii/v1/configsettings
```

Description

List all configuration settings. +Provides generic configuration information about the GOBii instances in a given deployment. This call does not require authentication

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeConfigSettingsDTO

Tags

- ConfigSettings

GET /contacts-search

```
GET /gobii/v1/contact-search
```

Description

List contacts for emailid, lastname, firstname, username.

Parameters

Type	Name	Description	Schema
Query	email —	contact's email	string
Query	firstName —	contact's first name	string
Query	lastName —	contact's last name	string
Query	userName —	contact's user name	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- [ContactsSearch](#)

POST /contacts

POST /gobii/v1/contacts

Description

Create new contact.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeContactDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- [Contacts](#)

GET /contacts

GET /gobii/v1/contacts

Description

List all contacts.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- Contacts

GET /contacts/{contactId}

```
GET /gobii/v1/contacts/{contactId}
```

Description

Get contacts by Contact Id.

Parameters

Type	Name	Description	Schema
Path	contactId —	ID of contacts to be extracted.	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- Contacts

PUT /contacts/{contactId}

```
PUT /gobii/v1/contacts/{contactId}
```

Description

Update contact by contact id.

Parameters

Type	Name	Description	Schema
Path	contactId —	ID of contacts to be updated.	integer (int32)
Body	body —		PayloadEnvelopeContactDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- Contacts

GET /contacts/{email}

```
GET /gobii/v1/contacts/{email}
```

Description

Get contact by email id.

Parameters

Type	Name	Description	Schema
Path	email —	email id of contacts to be extracted.	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Tags

- Contacts

GET /cvgroups/{cvGroupId}/cvs

```
GET /gobii/v1/cvgroups/{cvGroupId}/cvs
```

Description

List CV terms in given group

Parameters

Type	Name	Description	Schema
Path	cvGroupId —	ID of the CV group.	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- CvGroups.cvs

GET /cvgroups/{cvGroupTypeId}

```
GET /gobii/v1/cvgroups/{cvGroupTypeId}
```

Description

Get CV group by cv group type ID

Parameters

Type	Name	Description	Schema
Path	cvGroupTypeId —	ID of the cv group type	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvGroupDTO

Tags

- CvGroups

Get /cvgroups/{groupName}

```
GET /gobii/v1/cvgroups/{groupName}
```

Description

Get CV group by name

Parameters

Type	Name	Description	Schema
Path	groupName —	name of the cv group to be extracted	string
Query	cvGroupTypeId —		integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvGroupDTO

Tags

- CvGroups

POST /cvs

```
POST /gobii/v1/cvs
```

Description

Creates new cv's.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeCvDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- ControlledVocabularies

GET /cvs

```
GET /gobii/v1/cvs
```

Description

List all cvs in the GDM.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- ControlledVocabularies

GET /cvs/{cvId}

```
GET /gobii/v1/cvs/{cvId}
```

Description

Get cv by the id.

Parameters

Type	Name	Description	Schema
Path	cvId —	ID of the CV to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- ControlledVocabularies

PUT /cvs/{cvId}

```
PUT /gobii/v1/cvs/{cvId}
```

Description

Update cv by cvId.

Parameters

Type	Name	Description	Schema
Path	cvId —	ID of the CV to be updated	integer (int32)
Body	body —		PayloadEnvelopeCvDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- ControlledVocabularies

DELETE /cvs/{cvId}

```
DELETE /gobii/v1/cvs/{cvId}
```

Description

Deletes cv by id.

Parameters

Type	Name	Description	Schema
Path	cvId —	ID of cv to be deleted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- ControlledVocabularies

GET /cvs/{groupName}

```
GET /gobii/v1/cvs/{groupName}
```

Description

List cvs by the group name.

Parameters

Type	Name	Description	Schema
Path	groupName —	name of cv group to be extracted	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Tags

- Cvs

POST /datasets

POST /gobii/v1/datasets

Description

Creates a new dataset in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeDataSetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDataSetDTO

Tags

- Datasets

GET /datasets

GET /gobii/v1/datasets

Description

List all the existing datasets in the system. The list can be retrieved by page and specific page size.

Parameters

Type	Name	Description	Schema
Query	pageNo —	Retrieve the specified page by number	integer (int32)
Query	pageSize —	Specify the custom page size	integer (int32)
Query	queryId —		string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDataSetDTO

Tags

- Datasets

GET /datasets/types

```
GET /gobii/v1/datasets/types
```

Description

Lists dataset types in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeNameIdDTO

Tags

- Datasets.types

GET /datasets/types/{id}

```
GET /gobii/v1/datasets/types/{id}
```

Description

Gets the Dataset type by type ID.

Parameters

Type	Name	Description	Schema
Path	id —	ID of the dataset type	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDatasetDTO

Tags

- Datasets.types

GET /datasets/{dataSetId}

```
GET /gobii/v1/datasets/{dataSetId}
```

Description

Gets the Dataset entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	dataSetId —	ID of the Dataset to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDatasetDTO

Tags

- Datasets

PUT /datasets/{datasetId}

```
PUT /gobii/v1/datasets/{datasetId}
```

Description

Updates the Dataset entity having the specified datasetId.

Parameters

Type	Name	Description	Schema
Path	datasetId —	ID of the Dataset to be updated	integer (int32)
Body	body —		PayloadEnvelopeDatasetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDatasetDTO

Tags

- Datasets

GET /datasets/{datasetId}/analyses

```
GET /gobii/v1/datasets/{datasetId}/analyses
```

Description

Lists all the analysis in a dataset identified by dataset id.

Parameters

Type	Name	Description	Schema
Path	datasetId —	ID of the dataset	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Tags

- Datasets.analyses

GET /datasets/{datasetId}/jobs

```
GET /gobii/v1/datasets/{datasetId}/jobs
```

Description

List information for active job for a given datasetId

Parameters

Type	Name	Description	Schema
Path	datasetId —	ID of the dataset	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Datasets.jobs

POST /displays

```
POST /gobii/v1/displays
```

Description

Creates displays in GDM.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeDisplayDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Tags

- Displays

GET /displays

```
GET /gobii/v1/displays
```

Description

Lists all Displays in GDM

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Tags

- Displays

GET /displays/{displayId}

```
GET /gobii/v1/displays/{displayId}
```

Description

Get the Display by display Id

Parameters

Type	Name	Description	Schema
Path	displayId —	ID of the Display to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Tags

- Displays

PUT /displays/{displayId}

```
PUT /gobii/v1/displays/{displayId}
```

Description

Updates the Display entity having the specified displayId.

Parameters

Type	Name	Description	Schema
Path	displayId —	ID of the Display to be updated	integer (int32)
Body	body —		PayloadEnvelopeDisplayDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Tags

- Displays

GET /entities

```
GET /gobii/v1/entities
```

Description

Lists all the existing entities in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeEntityStatsDTO

Tags

- Entities

Gets

/entities/{entityNameParent}/{parentId}/{entityNameChild}/count

```
GET /gobii/v1/entities/{entityNameParent}/{parentId}/{entityNameChild}/count
```

Description

Retrieves the total count of the children for the given entity

Parameters

Type	Name	Description	Schema
Path	entityNameChild —	Name of the child entity	string
Path	entityNameParent —	Name of the parent entity	string
Path	parentId —	ID of the parent entity	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeEntityStatsDTO

Tags

- Entities.parent.child.count

GET /entities/{entityName}/count

```
GET /gobii/v1/entities/{entityName}/count
```

Description

Gets the total Entity count for the given entityName

Parameters

Type	Name	Description	Schema
Path	entityName —	Name of the Entity	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeEntityStatsDTO

Tags

- Entities.count

GET /entities/{entityName}/lastmodified

```
GET /gobii/v1/entities/{entityName}/lastmodified
```

Description

Gets last modified Entity for the given entityName

Parameters

Type	Name	Description	Schema
Path	entityName —	Name of the Entity	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeEntityStatsDTO

Tags

- Entities.lastModified

POST /experiments

POST /gobii/v1/experiments

Description

Creates a new experiment in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeExperimentDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Tags

- Experiments

GET /experiments

GET /gobii/v1/experiments

Description

List all the existing experiments in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Tags

- Experiments

GET /experiments/{experimentId}

GET /gobii/v1/experiments/{experimentId}

Description

Gets the Experiment entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	experimentId —	ID of the Experiment to be updated	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Tags

- Experiments

PUT /experiments/{experimentId}

```
PUT /gobii/v1/experiments/{experimentId}
```

Description

Updates the Experiment entity having the specified experimentId.

Parameters

Type	Name	Description	Schema
Path	experimentId —	ID of the Experiment to be updated	integer (int32)
Body	body —		PayloadEnvelopeExperimentDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Tags

- Experiments

GET /experiments/{experimentId}/protocols

```
GET /gobii/v1/experiments/{experimentId}/protocols
```

Description

Retrieves all the protocols having the specified experimentId in the system.

Parameters

Type	Name	Description	Schema
Path	experimentId —	ID of the Experiment	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Tags

- Experiments.protocols

GET /files/loader/{directoryName}

```
GET /gobii/v1/files/loader/{directoryName}
```

Description

Gets file preview for the specified directory name

Parameters

Type	Name	Description	Schema
Path	directoryName	Name of the directory	string
Query	fileFormat	Format/Extension of the file	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeLoaderFilePreviewDTO

Tags

- Files.loader

PUT /files/loader/{directoryName}

```
PUT /gobii/v1/files/loader/{directoryName}
```

Description

Updates a directory in the system that will be used for storing the data files for loading

Parameters

Type	Name	Description	Schema
Path	directoryName	Name of the directory/folder	string
Body	body		PayloadEnvelopeLoaderFilePreviewDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeLoaderFilePreviewDTO

Tags

- Files.loader

POST /files/{destinationType}

```
POST /gobii/v1/files/{destinationType}
```

Description

Uploads an arbitrary file to the specified destination

Parameters

Type	Name	Description	Schema
Path	destinationType	Destination type where the file will be uploaded to	string
Query	file	The file to be uploaded	string
Query	fileName	Name of the file to be uploaded	string

Responses

HTTP Code	Description	Schema
200	successful operation	string

Tags

- Files

DELETE /files/{destinationType}

```
DELETE /gobii/v1/files/{destinationType}
```

Description

Deletes an arbitrary file from the specified destination

Parameters

Type	Name	Description	Schema
Path	destinationType —	Destination type where the file will be delete from	string
Query	fileName —	Name of the file to be deleted	string

Responses

HTTP Code	Description	Schema
200	successful operation	string

Produces

- text/plain

Tags

- Files

POST /files/{gobiiJobId}/{destinationType}

```
POST /gobii/v1/files/{gobiiJobId}/{destinationType}
```

Description

Uploads the specified file for a specific job to the specified directory

Parameters

Type	Name	Description	Schema
Path	destinationType —	Destination type where the file will be uploaded to	string
Path	gobiiJobId —	ID of the Job that the file will be associated to	string
Query	file —	The file to be uploaded	string
Query	fileName —	Name of the file	string

Responses

HTTP Code	Description	Schema
200	successful operation	string

Tags

- Files.jobs

GET /files/{gobiiJobId}/{destinationType}

```
GET /gobii/v1/files/{gobiiJobId}/{destinationType}
```

Description

Downloads the specified file for a specific job from the specified directory

Parameters

Type	Name	Description	Schema
Path	destinationType —	Destination type where the file will be downloaded from	string
Path	gobiiJobId —	ID of the Job	string
Query	fileName —	Name of the file to be downloaded	string

Responses

HTTP Code	Description	Schema
200	successful operation	InputStreamResource

Tags

- Files.jobs

POST /instructions/extractor

```
POST /gobii/v1/instructions/extractor
```

Description

Creates extractor instruction file and then submits a new Job.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeExtractorInstructionFilesDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExtractorInstructionFilesDTO

Tags

- Instructions.extractor

GET /instructions/extractor/jobs/{jobName}

```
GET /gobii/v1/instructions/extractor/jobs/{jobName}
```

Description

Retrieves the extract job status along with other job details having the specified Job Name.

Parameters

Type	Name	Description	Schema
Path	jobName —	Name of the job	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Instructions.extractor.jobs

GET /instructions/extractor/{instructionFileName}

```
GET /gobii/v1/instructions/extractor/{instructionFileName}
```

Description

Retrieves the extractor instruction file entity having the specified instruction file name.

Parameters

Type	Name	Description	Schema
Path	instructionFileName —	Name of the instruction file to be retrieved	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExtractorInstructionFilesDTO

Tags

- Instructions.extractor

POST /instructions/loader

POST /gobii/v1/instructions/loader

Description

Creates loader instruction file and then submits a new Job.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeLoaderInstructionFilesDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeLoaderInstructionFilesDTO

Tags

- Instructions.loader

GET /instructions/loader/jobs/{jobName}

GET /gobii/v1/instructions/loader/jobs/{jobName}

Description

Gets the loading job status along with other job details having the specified Job Name.

Parameters

Type	Name	Description	Schema
Path	jobName —	Name of the job	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Instructions.loader.jobs

GET /instructions/loader/{instructionFileName}

```
GET /gobii/v1/instructions/loader/{instructionFileName}
```

Description

Gets the loader instruction file entity having the specified instruction file name.

Parameters

Type	Name	Description	Schema
Path	instructionFileName	Name of the instruction file to be retrieved.	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeLoaderInstructionFilesDTO

Tags

- Instructions.loader

POST /jobs

```
POST /gobii/v1/jobs
```

Description

Creates a new job in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeJobDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Jobs

GET /jobs

```
GET /gobii/v1/jobs
```

Description

Lists all the existing jobs in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Jobs

POST /jobs/dnasamples/{jobName}

```
POST /gobii/v1/jobs/dnasamples/{jobName}
```

Description

Creates DNA samples for a given Job having the specified name in the system.

Parameters

Type	Name	Description	Schema
Path	jobName —	Name of the Job that the DNA samples will be added to	string
Body	body —		PayloadEnvelopeDnaSampleDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Jobs.dnasamples

GET /jobs/{jobName}

```
GET /gobii/v1/jobs/{jobName}
```

Description

Gets the Job entity having the specified name.

Parameters

Type	Name	Description	Schema
Path	jobName —	Name of the Job to be extracted	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Jobs

PUT /jobs/{jobName}

```
PUT /gobii/v1/jobs/{jobName}
```

Description

Updates the Job entity having the specified jobName.

Parameters

Type	Name	Description	Schema
Path	jobName —	Name of the Job to be updated	string
Body	body —		PayloadEnvelopeJobDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Tags

- Jobs

POST /manifests

```
POST /gobii/v1/manifests
```

Description

Creates a Manifest entity for GOBii system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeManifestDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope ManifestDTO

Tags

- Manifests

GET /manifests

```
GET /gobii/v1/manifests
```

Description

Lists an unfiltered list of all Manifest entities.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope ManifestDTO

Tags

- Manifests

GET /manifests/{manifestId}

```
GET /gobii/v1/manifests/{manifestId}
```

Description

Gets the Manifest entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	manifestId —	ID of the Manifest to be retrieved	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeManifestDTO

Tags

- Manifests

PUT /manifests/{manifestId}

```
PUT /gobii/v1/manifests/{manifestId}
```

Description

Updates the Manifest entity having the specified manifestId.

Parameters

Type	Name	Description	Schema
Path	manifestId —	ID of the Manifest to be updated	integer (int32)
Body	body —		PayloadEnvelopeManifestDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeManifestDTO

Tags

- Manifests

GET /maps

```
GET /gobii/v1/maps
```

Description

Lists all the existing Mapsets in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMapsetDTO

Tags

- Maps

POST /mapsets

POST /gobii/v1/mapsets

Description

Creates a new mapset in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMapsetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMapsetDTO

Tags

- Mapsets

GET /mapsets

GET /gobii/v1/mapsets

Description

List all the existing mapsets in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MapsetDTO

Tags

- Mapsets

GET /mapsets/{mapsetId}

```
GET /gobii/v1/mapsets/{mapsetId}
```

Description

Gets the Mapset entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	mapsetId —	ID of the Mapset to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MapsetDTO

Tags

- Mapsets

PUT /mapsets/{mapsetId}

```
PUT /gobii/v1/mapsets/{mapsetId}
```

Description

Updates the Mapset entity having the specified mapsetId.

Parameters

Type	Name	Description	Schema
Path	mapsetId —	ID of the Mapset to be updated	integer (int32)
Body	body —		PayloadEnvelopeMapsetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMapsetDTO

Tags

- Mapsets

GET /marker-search

```
GET /gobii/v1/marker-search
```

Description

List Marker search results.

Parameters

Type	Name	Description	Schema
Query	name —	Name of the marker	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Tags

- MarkerSearch

POST /markergroups

POST /gobii/v1/markergroups

Description

Creates a new marker group in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMarkerGroupDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Tags

- MarkerGroups

GET /markergroups

GET /gobii/v1/markergroups

Description

Lists all the existing marker groups in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Tags

- MarkerGroups

GET /markergroups/{markerGroupId}

```
GET /gobii/v1/markergroups/{markerGroupId}
```

Description

Retrieves the Marker Group entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	markerGroup Id —	ID of the marker group to be updated	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MarkerGroupDTO

Tags

- MarkerGroups

PUT /markergroups/{markerGroupId}

```
PUT /gobii/v1/markergroups/{markerGroupId}
```

Description

Updates the Marker Group entity having the specified markerGroupId.

Parameters

Type	Name	Description	Schema
Path	markerGroup Id —	ID of the Marker Group to be updated	integer (int32)

Type	Name	Description	Schema
Body	body —		PayloadEnvelopeMarkerGroupDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Tags

- MarkerGroups

POST /markers

POST /gobii/v1/markers

Description

Creates a new marker in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMarkerDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Tags

- Markers

GET /markers

GET /gobii/v1/markers

Description

Lists all the existing markers in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MarkerDTO

Tags

- Markers

GET /markers/{markerId}

```
GET /gobii/v1/markers/{markerId}
```

Description

Retrieves the Marker entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	markerId —	ID of the Marker to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MarkerDTO

Tags

- Markers

PUT /markers/{markerId}

```
PUT /gobii/v1/markers/{markerId}
```


Description

Updates the Marker entity having the specified markerId.

Parameters

Type	Name	Description	Schema
Path	markerId —	ID of the Marker to be updated	integer (int32)
Body	body —		PayloadEnvelopeMarkerDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Tags

- Markers

POST /names/{entity}

```
POST /gobii/v1/names/{entity}
```

Description

Retrieves a list of name/ID combination for a given entity and name list. For the list of entities supported see class GobiiEntityNameType. This service is specifically implemented for these filter types: NAMES_BY_NAME_LIST - given a list of names, return the same list with the corresponding ID in the database. If name doesn't exist, ID will be 0. NAMES_BY_NAME_LIST_RETURN_EXISTS - given a list of names, return the list of names with ID that exists in the database. NAMES_BY_NAME_LIST_RETURN_ABSENT - given a list of names, return the list of names that doesn't exist in the database with 0 as the ID. Filter value varies per entity. This can be cv group name, project ID, platform ID, etc. Example use case: entity = CV; filterType = NAMES_BY_NAME_LIST; filterValue = germplasm_typeResult will be a list of CV terms with ID having germplasm_type as the cv group

Parameters

Type	Name	Description	Schema
Path	entity —	The entity to be retrieved	string

Type	Name	Description	Schema
Query	filterType —	The filter type for the name list	string
Query	filterValue —	The value for the filter type	string
Body	body —		PayloadEnvelopeNameIdDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeNameIdDTO

Tags

- Names

GET /names/{entity}

```
GET /gobii/v1/names/{entity}
```

Description

List of name/ID combination for a given entity. For the list of entities supported see class GobiiEntityType. List can further be filtered out by specifying the filter type and value. For the list of filter types supported see class GobiiFilterType. Example use case: entity = CV; filterType = NAMES_BY_TYPE_NAME; filterValue = statusResult will be a list of CV terms having status as the cv group

Parameters

Type	Name	Description	Schema
Path	entity —	The entity to be retrieved	string
Query	filterType —	The filter type for the name list	string
Query	filterValue —	The value for the filter type	string

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeNameIdDTO

Tags

- Names

POST /organizations

POST /gobii/v1/organizations

Description

Creates a new organization in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeOrganizationDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Organizations

GET /organizations

GET /gobii/v1/organizations

Description

List all the existing organizations in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Organizations

GET /organizations/{organizationId}

```
GET /gobii/v1/organizations/{organizationId}
```

Description

Retrieves the Organization entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	organizationId —	ID of the Organization to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Organizations

PUT /organizations/{organizationId}

```
PUT /gobii/v1/organizations/{organizationId}
```

Description

Updates the Organization entity having the specified organizationId.

Parameters

Type	Name	Description	Schema
Path	organizationId —	ID of the Organization to be updated	integer (int32)
Body	body —		PayloadEnvelopeOrganizationDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Organizations

POST /platforms

POST /gobii/v1/platforms

Description

Creates a new Platform in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopePlatformDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Tags

- Platforms

GET /platforms

```
GET /gobii/v1/platforms
```

Description

List all the existing platforms in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Tags

- Platforms

GET /platforms/protocols/{vendorProtocolId}

```
GET /gobii/v1/platforms/protocols/{vendorProtocolId}
```

Description

Gets the Platform entity having the specified Vendor Protocol ID.

Parameters

Type	Name	Description	Schema
Path	vendorProtocolId	ID of the vendor protocol	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Tags

- Platform.protocols

GET /platforms/{platformId}

```
GET /gobii/v1/platforms/{platformId}
```

Description

Retrieves the Platform entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	platformId —	ID of the Platform to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Tags

- Platforms

PUT /platforms/{platformId}

```
PUT /gobii/v1/platforms/{platformId}
```

Description

Updates the Platform entity having the specified platformId.

Parameters

Type	Name	Description	Schema
Path	platformId —	ID of the Platform to be updated	integer (int32)
Body	body —		PayloadEnvelopePlatformDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Tags

- Platforms

GET /projects

POST /gobii/v1/projects

Description

Creates a new project in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeProjectDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Tags

- Projects

GET /projects

GET /gobii/v1/projects

Description

List all the existing projects in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Tags

- Projects

GET /projects/{projectId}

```
GET /gobii/v1/projects/{projectId}
```

Description

Retrieves the Project entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	projectId —	ID of the Project to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Tags

- Projects

PUT /projects/{projectId}

```
PUT /gobii/v1/projects/{projectId}
```

Description

Updates the Project entity having the specified projectId.

Parameters

Type	Name	Description	Schema
Path	projectId —	ID of the Project to be updated	integer (int32)
Body	body —		PayloadEnvelopeProjectDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Tags

- Projects

POST /protocols

POST /gobii/v1/protocols

Description

Creates a new Protocol in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeProtocolDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Tags

- Protocols

GET /protocols

```
GET /gobii/v1/protocols
```

Description

List all the existing protocols in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Tags

- Protocols

GET /protocols/{protocolId}

```
GET /gobii/v1/protocols/{protocolId}
```

Description

Retrieves the Protocol entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	protocolId —	ID of the Protocol to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Tags

- Protocols

PUT /protocols/{protocolId}

```
PUT /gobii/v1/protocols/{protocolId}
```

Description

Updates the Protocol entity having the specified protocolId.

Parameters

Type	Name	Description	Schema
Path	protocolId —	ID of the Protocol to be updated	integer (int32)
Body	body —		PayloadEnvelopeProtocolDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Tags

- Protocols

POST /protocols/{protocolId}/vendors

```
POST /gobii/v1/protocols/{protocolId}/vendors
```

Description

Creates a new Vendor Protocol in the system for specified Protocol ID.

Parameters

Type	Name	Description	Schema
Path	protocolId —	ID of the Protocol	integer (int32)
Body	body —		PayloadEnvelopeOrganizationDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Protocols.vendor

GET /protocols/{protocolId}/vendors

```
GET /gobii/v1/protocols/{protocolId}/vendors
```

Description

List all the vendor protocols given protocolId in the system.

Parameters

Type	Name	Description	Schema
Path	protocolId —	ID of the Protocol ID	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Protocols.vendor

PUT /protocols/{protocolId}/vendors

```
PUT /gobii/v1/protocols/{protocolId}/vendors
```

Description

Updates the Vendor Protocol entity having the specified protocolId.

Parameters

Type	Name	Description	Schema
Path	protocolId —	ID of the Protocol	integer (int32)
Body	body —		PayloadEnvelopeOrganizationDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Tags

- Protocols.vendor

POST /references

POST /gobii/v1/references

Description

Creates a new reference in the system.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeReferenceDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Tags

- References

GET /references

```
GET /gobii/v1/references
```

Description

Lists all the existing references in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Tags

- References

GET /references/{referenceId}

```
GET /gobii/v1/references/{referenceId}
```

Description

Gets the Reference entity having the specified ID.

Parameters

Type	Name	Description	Schema
Path	referenceId —	ID of the Reference to be extracted	integer (int32)

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Tags

- References

PUT /references/{referenceId}

```
PUT /gobii/v1/references/{referenceId}
```

Description

Updates the Reference entity having the specified referenceId.

Parameters

Type	Name	Description	Schema
Path	referenceId —	ID of the Reference to be updated	integer (int32)
Body	body —		PayloadEnvelopeReferenceDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Tags

- References

/restprofiles/

```
PUT /gobii/v1/restprofiles
```

Description

Update REST profiles When the Header of the payload envelope for a resource contains maxGet, maxPost, and maxPut values, this resource provides a means to update the max for a given rest resource ID and for a given HTTP verb. The values are transient in the sense that they will be confined only to a specific web service deployment. They are stored in the web service configurationdocument

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeRestProfileDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeRestProfileDTO

Tags

- RestProfiles

Definitions

AnalysisDTO

Name	Schema
algorithm —	string
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
analysisDescription —	string
analysisId —	integer (int32)
analysisName —	string
anlaysisTypeId —	integer (int32)
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)

Name	Schema
parameters —	< EntityPropertyDTO > array
program —	string
programVersion —	string
referenceId —	integer (int32)
sourceName —	string
sourceUri —	string
sourceVersion —	string
statusId —	integer (int32)
timeExecuted —	string (date-time)

ConfigSettingsDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
id —	integer (int32)
maxUploadSizeMbytes —	integer (int32)
serverCapabilities —	< string, boolean > map
serverConfigs —	< string, ServerConfigItem > map

ContactDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
code —	string
contactId —	integer (int32)

Name	Schema
createdBy —	integer (int32)
createdDate —	string (date-time)
email —	string
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
firstName —	string
id —	integer (int32)
lastName —	string
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
organizationId —	integer (int32)
roles —	< integer (int32) > array
userName —	string

CvDTO

Name	Schema
abbreviation —	string
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
cvId —	integer (int32)
definition —	string
entityStatus —	integer (int32)

Name	Schema
group —	integer (int32)
groupId —	integer (int32)
groupType —	integer (int32)
id —	integer (int32)
rank —	integer (int32)
term —	string
xrefId —	integer (int32)

CvGroupDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
cvGroupId —	integer (int32)
definition —	string
groupType —	integer (int32)
id —	integer (int32)
name —	string

DataSetDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
analysesIds —	< integer (int32) > array
callingAnalysisId —	integer (int32)
callingAnalysisName —	string

Name	Schema
createdBy —	integer (int32)
createdDate —	string (date-time)
dataFile —	string
dataSetId —	integer (int32)
dataTable —	string
datasetName —	string
datatypeId —	integer (int32)
datatypeName —	string
experimentId —	integer (int32)
experimentName —	string
id —	integer (int32)
jobId —	integer (int32)
jobStatusId —	integer (int32)
jobStatusName —	string
jobSubmittedDate —	string (date-time)
jobTypeId —	integer (int32)
jobTypeName —	string
loaderFirstName —	string
loaderLastName —	string
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)

Name	Schema
piContactId —	integer (int32)
piEmail —	string
piFirstName —	string
piLastName —	string
platformId —	integer (int32)
platformName —	string
projectId —	integer (int32)
projectName —	string
protocolId —	integer (int32)
protocolName —	string
qualityFile —	string
qualityTable —	string
scores —	< integer (int32) > array
statusId —	integer (int32)
totalMarkers —	integer (int32)
totalSamples —	integer (int32)

DisplayDTO

Name	Description	Schema
allowedProcessTypes —		< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
columnName —		string

Name	Description	Schema
createdBy —		integer (int32)
createdDate —		string (date-time)
displayId —		integer (int32)
displayName —		string
displayRank —		integer (int32)
entityNameType —		enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —		integer (int32)
includeDetailsList —	Default : false	boolean
modifiedBy —		integer (int32)
modifiedDate —		string (date-time)
tableName —		string
tableNamesWithColDisplay —		< string, < TableColDisplay > array > map

DnaSampleDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
analysisMethodName —	string
dnaRunId —	integer (int32)
dnaSampleId —	integer (int32)
dnaSampleName —	string
dnaSampleNum —	integer (int32)
dnarunName —	string
experimentId —	integer (int32)
externalCode —	string
germplasmExternalCode —	string
germplasmId —	integer (int32)
germplasmName —	string
id —	integer (int32)
markerCount —	integer (int32)
projectId —	integer (int32)

EntityPropertyDTO

Name	Schema
entityIdId —	integer (int32)
propertyId —	integer (int32)
propertyName —	string
propertyValue —	string

EntityStatsDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
count —	integer (int32)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
entityStateDateType —	enum (INSERT_UPDATE, INSERT_ONLY)
id —	integer (int32)
lastModified —	string (date-time)

ExperimentDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
datasets —	< DataSetDTO > array
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
experimentCode —	string
experimentDataFile —	string
experimentId —	integer (int32)

Name	Schema
experimentName —	string
id —	integer (int32)
manifestId —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
projectId —	integer (int32)
statusId —	integer (int32)
vendorProtocolId —	integer (int32)

ExtractorInstructionFilesDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
gobiiExtractorInstructions —	< GobiiExtractorInstruction > array
id —	integer (int32)
instructionFileName —	string
jobId —	string

File

Name	Description	Schema
absolute —	Default : false	boolean
absoluteFile —		File
absolutePath —		string
canonicalFile —		File

Name	Description	Schema
canonicalPath —		string
directory —	Default : false	boolean
file —	Default : false	boolean
freeSpace —		integer (int64)
hidden —	Default : false	boolean
name —		string
parent —		string
parentFile —		File
path —		string
totalSpace —		integer (int64)
usableSpace —		integer (int64)

GobiiDataSetExtract

Name	Description	Schema
accolate —	Default : false	boolean
dataSet —		PropNameId
extractDestinationDirectory —		string
extractedFiles —		< File > array
gobiiDatasetType —		PropNameId
gobiiExtractFilterType —		enum (UNKNOWN, WHOLE_DATASET, BY_MARKER, BY_SAMPLE)

Name	Description	Schema
gobiiFileType —		enum (HAPMAP, FLAPJACK, GENERIC, VCF, META_DATA)
gobiiJobStatus —		enum (CV_PROGRESSTAT US_PENDING, CV_PROGRESSTATU S_INPROGRESS, CV_PROGRESSTATU S_COMPLETED, CV_PROGRESSTATU S_FAILED, CV_PROGRESSTATU S_VALIDATION, CV_PROGRESSTATU S_DIGEST, CV_PROGRESSTATU S_TRANSFORMATIO N, CV_PROGRESSTATU S_METADATALOAD, CV_PROGRESSTATU S_MATRIXLOAD, CV_PROGRESSTATU S_ABORTED, CV_PROGRESSTATU S_METADATAEXTRA CT, CV_PROGRESSTATU S_FINALASSEMBLY, CV_PROGRESSTATU S_QCPROCESSING, CV_PROGRESSTATU S_NOSTATUS)
gobiiSampleLi stType —		enum (GERMPLASM_NAM E, EXTERNAL_CODE, DNA_SAMPLE)
listFileName —		string
logMessage —		string
markerGroup s —		< PropNameId > array
markerList —		< string > array

Name	Description	Schema
platforms —		< PropNameId > array
principleInvestigator —		PropNameId
project —		PropNameId
sampleList —		< string > array

GobiiExtractorInstruction

Name	Description	Schema
contactEmail —		string
contactId —		integer (int32)
dataSetExtracts —		< GobiiDataSetExtract > array
gobiiCropType —		string
mapsetIds —		< integer (int32) > array
qcCheck —	Default : false	boolean

GobiiFile

Name	Description	Schema
createSource —	Default : false	boolean
delimiter —		string
destination —		string
gobiiFileType —		enum (HAPMAP, FLAPJACK, GENERIC, VCF, META_DATA)
requireDirectoriesToExist —	Default : false	boolean

Name	Description	Schema
source —		string

GobiiFileColumn

Name	Description	Schema
constantValue —		string
dataSetOrientationType —		enum (SAMPLE_FAST, MARKER_FAST)
dataSetType —		enum (NUCLEOTIDE_2_LETTER, IUPAC, SSR_ALLELE_SIZE, DOMINANT_NON_NUCLEOTIDE, CO_DOMINANT_NON_NUCLEOTIDE, VCF)
filterFrom —		string
filterTo —		string
findText —		string
getcCoord —		integer (int32)
getrCoord —		integer (int32)
gobiiColumnType —		enum (CSV_COLUMN, CSV_ROW, CSV_BOTH, VCF_SAMPLE, VCF_MARKER, VCF_VARIANT, VCF_METADATA, VCF_INFO, CONSTANT, AUTOINCREMENT, VCF_MARKER_POS)
index —		string

Name	Description	Schema
metaDataId —		string
name —		string
replaceText —		string
subcolumn —	Default : false	boolean
subcolumnDelimiter —		string

GobiiLoaderInstruction

Name	Description	Schema
columnsByTableName —		< string, < GobiiFileColumn > array > map
contactEmail —		string
contactId —		integer (int32)
dataSet —		PropNameId
dataSetId —		integer (int32)
datasetType —		PropNameId
experiment —		PropNameId
gobiiCropType —		string
gobiiFile —		GobiiFile
gobiiFileColumns —		< GobiiFileColumn > array

Name	Description	Schema
gobiiJobStatus —		enum (CV_PROGRESSTAT US_PENDING, CV_PROGRESSTATU S_INPROGRESS, CV_PROGRESSTATU S_COMPLETED, CV_PROGRESSTATU S_FAILED, CV_PROGRESSTATU S_VALIDATION, CV_PROGRESSTATU S_DIGEST, CV_PROGRESSTATU S_TRANSFORMATIO N, CV_PROGRESSTATU S_METADATALOAD, CV_PROGRESSTATU S_MATRIXLOAD, CV_PROGRESSTATU S_ABORTED, CV_PROGRESSTATU S_METADATAEXTRA CT, CV_PROGRESSTATU S_FINALASSEMBLY, CV_PROGRESSTATU S_QCPROCESSING, CV_PROGRESSTATU S_NOSTATUS)
jobPayloadType —		enum (CV_PAYLOADTYPE_ SAMPLES, CV_PAYLOADTYPE_ MARKERS, CV_PAYLOADTYPE_ MATRIX, CV_PAYLOADTYPE_ MARKERSAMPLES, CV_PAYLOADTYPE_ ALLMETA)
logMessage —		string
mapset —		PropNameId
platform —		PropNameId

Name	Description	Schema
project —		PropNameId
qcCheck —	Default : false	boolean
table —		string
vcfParameters —		VcfParameters

Header

Name	Schema
cropType —	string
dtoHeaderAuth —	HeaderAuth
gobiiProcessType —	enum (CREATE, READ, UPDATE, DELETE, NONE)
gobiiVersion —	string
maxGet —	integer (int32)
maxPost —	integer (int32)
maxPut —	integer (int32)
pagination —	Pagination
status —	Status

HeaderAuth

Name	Schema
gobiiCropType —	string
password —	string
token —	string
userName —	string

HeaderStatusMessage

Name	Schema
gobiiStatusLevel —	enum (ERROR, VALIDATION, WARNING, INFO, OK)
gobiiValidationStatusType —	enum (NONE, UNKNOWN, VALIDATION_COMPOUND_UNIQUE, VALIDATION_NOT_UNIQUE, NONEXISTENT_FK_ENTITY, BAD_REQUEST, MISSING_REQUIRED_VALUE, ENTITY_DOES_NOT_EXIST, ENTITY_ALREADY_EXISTS, UNKNOWN_ENUM_VALUE, RESOURCE_LIMIT)
message —	string

InputStream

Type : object

InputStreamResource

Name	Description	Schema
description —		string
file —		File
filename —		string
inputStream —		InputStream
open —	Default : false	boolean
readable —	Default : false	boolean
uri —		string (uri)
url —		string (url)

JobDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array

Name	Schema
datasetIds —	< integer (int32) > array
id —	integer (int32)
jobId —	integer (int32)
jobName —	string
message —	string
payloadType —	string
status —	string
submittedBy —	integer (int32)
submittedDate —	string (date-time)
type —	string

Link

Name	Schema
description —	string
href —	string
methods —	< enum (GET, POST, PUT, PATCH, DELETE, OPTIONS) > array

LinkCollection

Name	Schema
exploreLinksPerDataItem —	< < Link > array > array
linksPerDataItem —	< Link > array

LoaderFilePreviewDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
directoryName —	string
fileList —	< string > array
filePreview —	< < string > array > array
id —	integer (int32)
previewFileName —	string

LoaderInstructionFilesDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
gobiiLoaderInstructions —	< GobiiLoaderInstruction > array
id —	integer (int32)
instructionFileName —	string

ManifestDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
code —	string
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)

Name	Schema
filePath —	string
id —	integer (int32)
manifestId —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
name —	string

MapsetDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
code —	string
createdBy —	integer (int32)
createdDate —	string (date-time)
description —	string
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPPLASM, JOB)
id —	integer (int32)
mapType —	integer (int32)
mapsetId —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)

Name	Schema
name —	string
properties —	< EntityPropertyDTO > array
referenceId —	integer (int32)
statusId —	integer (int32)

MarkerDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
alts —	< integer (int32) > array
code —	string
id —	integer (int32)
markerId —	integer (int32)
markerName —	string
platformId —	integer (int32)
platformName —	string
ref —	string
referenceId —	integer (int32)
sequence —	string
status —	integer (int32)
strand —	integer (int32)
strandId —	integer (int32)
variantId —	integer (int32)

MarkerGroupDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
code —	string
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
germplasmGroup —	string
id —	integer (int32)
markerGroupId —	integer (int32)
markerMap —	< string, < string > array > map
markers —	< MarkerGroupMarkerDTO > array
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
name —	string
statusId —	integer (int32)

MarkerGroupMarkerDTO

Name	Description	Schema
favorableAllele —		string

Name	Description	Schema
gobiiProcessType —		enum (CREATE, READ, UPDATE, DELETE, NONE)
markerExists —	Default : false	boolean
markerId —		integer (int32)
markerName —		string
platformId —		integer (int32)
platformName —		string

NameIdDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
entityLasetModified —	string (date-time)
fkId —	integer (int32)
gobiiEntityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
gobiiFkEntityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
name —	string
queryObject —	object

OrganizationDTO

Name	Schema
address —	string
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
name —	string
organizationId —	integer (int32)
statusId —	integer (int32)
vendorProtocols —	< VendorProtocolDTO > array
website —	string

Pagination

Name	Schema
currentPage —	integer (int32)
pageSize —	integer (int32)
pagedQueryId —	string

Name	Schema
queryTime —	string (date-time)
totalPages —	integer (int32)

PayloadAnalysisDTO

Name	Schema
data —	< AnalysisDTO > array
linkCollection —	LinkCollection

PayloadConfigSettingsDTO

Name	Schema
data —	< ConfigSettingsDTO > array
linkCollection —	LinkCollection

PayloadContactDTO

Name	Schema
data —	< ContactDTO > array
linkCollection —	LinkCollection

PayloadCvDTO

Name	Schema
data —	< CvDTO > array
linkCollection —	LinkCollection

PayloadCvGroupDTO

Name	Schema
data —	< CvGroupDTO > array
linkCollection —	LinkCollection

PayloadDataSetDTO

Name	Schema
data —	< DataSetDTO > array
linkCollection —	LinkCollection

PayloadDisplayDTO

Name	Schema
data —	< DisplayDTO > array
linkCollection —	LinkCollection

PayloadDnaSampleDTO

Name	Schema
data —	< DnaSampleDTO > array
linkCollection —	LinkCollection

PayloadEntityStatsDTO

Name	Schema
data —	< EntityStatsDTO > array
linkCollection —	LinkCollection

PayloadEnvelopeAnalysisDTO

Name	Schema
header —	Header
payload —	PayloadAnalysisDTO

PayloadEnvelopeConfigSettingsDTO

Name	Schema
header —	Header
payload —	PayloadConfigSettingsDTO

PayloadEnvelopeContactDTO

Name	Schema
header —	Header
payload —	PayloadContactDTO

PayloadEnvelopeCvDTO

Name	Schema
header —	Header
payload —	PayloadCvDTO

PayloadEnvelopeCvGroupDTO

Name	Schema
header —	Header
payload —	PayloadCvGroupDTO

PayloadEnvelopeDataSetDTO

Name	Schema
header —	Header
payload —	PayloadDataSetDTO

PayloadEnvelopeDisplayDTO

Name	Schema
header —	Header
payload —	PayloadDisplayDTO

PayloadEnvelopeDnaSampleDTO

Name	Schema
header —	Header
payload —	PayloadDnaSampleDTO

PayloadEnvelopeEntityStatsDTO

Name	Schema
header —	Header
payload —	PayloadEntityStatsDTO

PayloadEnvelopeExperimentDTO

Name	Schema
header —	Header
payload —	PayloadExperimentDTO

PayloadEnvelopeExtractorInstructionFilesDTO

Name	Schema
header —	Header
payload —	PayloadExtractorInstructionFilesDTO

PayloadEnvelopeJobDTO

Name	Schema
header —	Header
payload —	PayloadJobDTO

PayloadEnvelopeLoaderFilePreviewDTO

Name	Schema
header —	Header
payload —	PayloadLoaderFilePreviewDTO

PayloadEnvelopeLoaderInstructionFilesDTO

Name	Schema
header —	Header
payload —	PayloadLoaderInstructionFilesDTO

PayloadEnvelopeManifestDTO

Name	Schema
header —	Header
payload —	PayloadManifestDTO

PayloadEnvelopeMapsetDTO

Name	Schema
header —	Header
payload —	PayloadMapsetDTO

PayloadEnvelopeMarkerDTO

Name	Schema
header —	Header
payload —	PayloadMarkerDTO

PayloadEnvelopeMarkerGroupDTO

Name	Schema
header —	Header
payload —	PayloadMarkerGroupDTO

PayloadEnvelopeNameIdDTO

Name	Schema
header —	Header
payload —	PayloadNameIdDTO

PayloadEnvelopeOrganizationDTO

Name	Schema
header —	Header
payload —	PayloadOrganizationDTO

PayloadEnvelopePlatformDTO

Name	Schema
header —	Header
payload —	PayloadPlatformDTO

PayloadEnvelopeProjectDTO

Name	Schema
header —	Header
payload —	PayloadProjectDTO

PayloadEnvelopeProtocolDTO

Name	Schema
header —	Header
payload —	PayloadProtocolDTO

PayloadEnvelopeReferenceDTO

Name	Schema
header —	Header
payload —	PayloadReferenceDTO

PayloadEnvelopeRestProfileDTO

Name	Schema
header —	Header
payload —	PayloadRestProfileDTO

PayloadExperimentDTO

Name	Schema
data —	< ExperimentDTO > array
linkCollection —	LinkCollection

PayloadExtractorInstructionFilesDTO

Name	Schema
data —	< ExtractorInstructionFilesDTO > array
linkCollection —	LinkCollection

PayloadJobDTO

Name	Schema
data —	< JobDTO > array
linkCollection —	LinkCollection

PayloadLoaderFilePreviewDTO

Name	Schema
data —	< LoaderFilePreviewDTO > array
linkCollection —	LinkCollection

PayloadLoaderInstructionFilesDTO

Name	Schema
data —	< LoaderInstructionFilesDTO > array
linkCollection —	LinkCollection

PayloadManifestDTO

Name	Schema
data —	< ManifestDTO > array
linkCollection —	LinkCollection

PayloadMapsetDTO

Name	Schema
data —	< MapsetDTO > array
linkCollection —	LinkCollection

PayloadMarkerDTO

Name	Schema
data —	< MarkerDTO > array
linkCollection —	LinkCollection

PayloadMarkerGroupDTO

Name	Schema
data —	< MarkerGroupDTO > array
linkCollection —	LinkCollection

PayloadNameIdDTO

Name	Schema
data —	< NameIdDTO > array
linkCollection —	LinkCollection

PayloadOrganizationDTO

Name	Schema
data —	< OrganizationDTO > array
linkCollection —	LinkCollection

PayloadPlatformDTO

Name	Schema
data —	< PlatformDTO > array
linkCollection —	LinkCollection

PayloadProjectDTO

Name	Schema
data —	< ProjectDTO > array
linkCollection —	LinkCollection

PayloadProtocolDTO

Name	Schema
data —	< ProtocolDTO > array
linkCollection —	LinkCollection

PayloadReferenceDTO

Name	Schema
data —	< ReferenceDTO > array
linkCollection —	LinkCollection

PayloadRestProfileDTO

Name	Schema
data —	< RestProfileDTO > array
linkCollection —	LinkCollection

PlatformDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
platformCode —	string
platformDescription —	string
platformId —	integer (int32)
platformName —	string
properties —	< EntityPropertyDTO > array
statusId —	integer (int32)
typeId —	integer (int32)

ProjectDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
piContact —	integer (int32)
principleInvestigators —	< string, string > map
projectCode —	string
projectDescription —	string
projectId —	integer (int32)
projectName —	string
projectStatus —	integer (int32)
properties —	< EntityPropertyDTO > array

PropNameId

Name	Schema
id —	integer (int32)

Name	Schema
name —	string

ProtocolDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
description —	string
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
name —	string
platformId —	integer (int32)
props —	< EntityPropertyDTO > array
protocolId —	integer (int32)
status —	integer (int32)
typeId —	integer (int32)
vendorProtocols —	< VendorProtocolDTO > array

ReferenceDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
createdBy —	integer (int32)
createdDate —	string (date-time)
entityNameType —	enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
filePath —	string
id —	integer (int32)
link —	string
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
name —	string
referenceId —	integer (int32)
version —	string

RestProfileDTO

Name	Schema
allowedProcessTypes —	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
id —	integer (int32)
max —	integer (int32)
restMethodType —	enum (GET, POST, PUT, PATCH, DELETE, OPTIONS)

Name	Schema
restResourceId —	enum (GOBII_AUTH, GOBII_PING, GOBII_PROJECTS, GOBII_NAME_ID_LIST_DEPRECATED, GOBII_NAMES, GOBII_FILE_LOAD, GOBII_FILE_LOAD_INSTRUCTIONS, GOBII_FILE_EXTRACTOR_INSTRUCTIONS, GOBII_FILE_EXTRACTOR_JOBS, GOBII_FILE_LOADER_JOBS, GOBII_DISPLAY, GOBII_CV, GOBII_CVGROUP, GOBII_CONTACTS, GOBII_CONTACT_SEARCH, GOBII_ORGANIZATION_SEARCH, GOBII_REFERENCE, GOBII_EXPERIMENTS, GOBII_DATASETS, GOBII_DATASETTYPES, GOBII_ANALYSIS, GOBII_MARKERGROUP, GOBII_MANIFEST, GOBII_ORGANIZATION, GOBII_PLATFORM, GOBII_MAPSET, GOBII_CONFIGSETTINGS, GOBII_ROLES, GOBII_MARKERS, GOBII_MARKER_SEARCH, GOBII_CALLS, GOBII_GERMPLASM, GOBII_OBSERVATION_VARIABLES, GOBII_PROTOCOL, GOBII_VENDORS, GOBII_FILES, GOBII_JOB, GOBII_ENTITIES, GOBII_COUNT, GOBII_LAST_MODIFIED, GOBII_VERTICES, GOBII_VALUES, GOBII_REST_PROFILES, GOBII_DNASAMPLES, GOBII_DNARUN, GOBII_LINKAGEGROUP, GOBII_JOB_DNASAMPLE, BRAPI_ALLELE_MATRIX_SEARCH, BRAPI_ALLELE_MATRIX_SEARCH_STATUS, BRAPI_ALLELE_MATRICES, KDC_START, KDC_STATUS, KDC_DOWNLOAD, KDC_PURGE, BRAPI_LOGIN, BRAPI_STUDIES_SEARCH, BRAPI_STUDIES, BRAPI_CALLS)
templateParameter —	string

ServerConfigItem

Name	Schema
confidentialityNotice —	string
contextRoot —	string
domain —	string
fileLocations —	< string, string > map
gobiiCropType —	string
port —	integer (int32)

Status

Name	Description	Schema
statusMessages		< HeaderStatusMessage > array
statusMessagesByCode		< string, string > map
succeeded	Default : false	boolean

TableColDisplay

Name	Schema
columnName	string
displayId	integer (int32)
displayName	string
rank	integer (int32)

VcfParameters

Name	Description	Schema
maf		number (float)
minDp		number (float)
minQ		number (float)
removeIndels	Default : false	boolean
toIupac	Default : false	boolean

VendorProtocolDTO

Name	Schema
allowedProcessTypes	< enum (CREATE, READ, UPDATE, DELETE, NONE) > array

Name	Schema
id —	integer (int32)
name —	string
organizationId —	integer (int32)
protocolId —	integer (int32)
status —	integer (int32)
vendorProtocolId —	integer (int32)