

master document for generating PDF

= GOBii API

Table of Contents

Overview.....	1
Version information	1
License information	1
URI scheme	1
Overview.....	1
Language and Framework.....	1
The DTO Payload	2
Request/Response Structure	2
Authentication	4
Error Handling	4
Paths	5
Creates analysis entity	5
Gets all the analyses	5
Gets analysis by Analysis ID	6
Updates analysis by Analysis ID	6
Authentication service for accessing GOBii System.....	7
Gets the configuration settings for GOBii system.....	7
/contacts-search	8
Adds new contact to GOBii system.....	8
Gets all the contacts in the GOBii system	9
Gets contacts by contact ID	9
Updates contacts by contact ID.....	10
Gets contacts by email id	10
Gets CV terms by CV group ID.....	11
Gets CV groups by CV group type ID	11
Gets CV Group details given CV group name and type ID	12
/cvs.....	12
/cvs.....	13
/cvs.....	13
/cvs.....	14
/cvs.....	14
/cvs/{groupName:[a-zA-Z_]+}.....	15
Creates a dataset	15
Gets all the datasets.....	16
Gets the defined Dataset Types in the system.....	16
Gets all the datasets with the given dataset type ID	17
Gets the dataset given Dataset ID.....	17
Updates the Dataset by Dataset ID	18

Gets all the analysis for the given Dataset ID	18
Gets the job details for the given Dataset ID	19
Creates a display entity	19
Gets all the displays	20
Gets display by Display ID	20
Updates display by Display ID	21
Gets all the entities	21
Gets the total count of the children for the given Entity	21
Gets the total entity count for the given Entity Name	22
Gets the last modified entity for the given Entity Name	23
Creates an experiment	23
Gets all the experiments	24
Gets the experiment given Experiment ID	24
Updates the Experiment by Experiment ID	24
Gets all the Protocols by Experiment ID	25
Gets file preview for the specified directory name	25
Creates a Directory to be used for loading	26
Upload file	27
Delete file	27
Upload file for Job	28
Download file for specified Job	29
Creates instruction file for extracting data	29
Gets the extract job status by Job name	30
Gets the extract instruction information given file name	30
Creates instruction file for loading data	31
Gets the loading job status by Job name	31
Gets the loading instruction information given file name	32
Creates a job	32
Gets all the jobs	33
Created DNA samples for a given Job	33
Gets the job given Job Name	34
Updates the Job by Job Name	34
Creates manifest entity	35
Gets all the Manifests	35
Gets the manifest given Manifest ID	36
Updates manifest by Manifest ID	36
Gets all the Mapsets	37
Creates a mapset	37
Gets all the mapsets	38
Gets the mapset given Mapset ID	38
Updates the Mapset by Mapset ID	39

Marker search	39
Creates a Marker Group	40
Gets all the marker groups	40
Gets the marker group given Marker Group ID	41
Updates the Marker Group by Marker Group ID	41
Creates a marker	42
Gets all the markers	42
Gets the marker given Marker ID	43
Updates the Marker by Marker ID	43
Gets a list of name/ID combination for given Entity and a name list	44
Gets a list of name/ID combination for given Entity	45
Creates a organization	45
Gets all the organizations	46
Gets the organization given Organization ID	46
Updates the Organization by Organization ID	47
Creates a Platform	47
Gets all the platforms	48
Gets the platform details given Vendor Protocol ID	48
Gets the platform given Platform ID	49
Updates the Platform by Platform ID	49
Creates a project	50
Gets all the projects	50
Gets the projects given Project ID	51
Updates the Project by Project ID	51
Creates a Protocol	52
Gets all the protocols	52
Gets the protocol given Protocol ID	53
Updates the Protocol by Protocol ID	53
Creates a Vendor Protocol record for given Protocol ID	54
Gets all the Vendor Protocols given Protocol ID	54
Updates the Vendor Protocol by Protocol ID	55
Creates a reference	55
Gets all the references	56
Gets the reference given Reference ID	56
Updates the Reference by Reference ID	57
Gets REST profiles of GOBii system	57
Definitions	58
AnalysisDTO	58
ConfigSettingsDTO	59
ContactDTO	59
CvDTO	60

CvGroupDTO	61
DataSetDTO	61
DisplayDTO	63
DnaSampleDTO	65
EntityPropertyDTO	65
EntityStatsDTO	66
ExperimentDTO	66
ExtractorInstructionFilesDTO	67
File	67
GobiiDataSetExtract	68
GobiiExtractorInstruction	70
GobiiFile	70
GobiiFileColumn	71
GobiiLoaderInstruction	72
Header	74
HeaderAuth	74
HeaderStatusMessage	75
InputStream	75
InputStreamResource	75
JobDTO	76
Link	76
LinkCollection	76
LoaderFilePreviewDTO	77
LoaderInstructionFilesDTO	77
ManifestDTO	77
MapsetDTO	78
MarkerDTO	79
MarkerGroupDTO	80
MarkerGroupMarkerDTO	81
NameIdDTO	81
OrganizationDTO	82
Pagination	83
PayloadAnalysisDTO	83
PayloadConfigSettingsDTO	83
PayloadContactDTO	83
PayloadCvDTO	84
PayloadCvGroupDTO	84
PayloadDataSetDTO	84
PayloadDisplayDTO	84
PayloadDnaSampleDTO	84
PayloadEntityStatsDTO	85

PayloadEnvelopeAnalysisDTO	85
PayloadEnvelopeConfigSettingsDTO	85
PayloadEnvelopeContactDTO	85
PayloadEnvelopeCvDTO	85
PayloadEnvelopeCvGroupDTO	86
PayloadEnvelopeDataSetDTO	86
PayloadEnvelopeDisplayDTO	86
PayloadEnvelopeDnaSampleDTO	86
PayloadEnvelopeEntityStatsDTO	86
PayloadEnvelopeExperimentDTO	87
PayloadEnvelopeExtractorInstructionFilesDTO	87
PayloadEnvelopeJobDTO	87
PayloadEnvelopeLoaderFilePreviewDTO	87
PayloadEnvelopeLoaderInstructionFilesDTO	87
PayloadEnvelopeManifestDTO	88
PayloadEnvelopeMapsetDTO	88
PayloadEnvelopeMarkerDTO	88
PayloadEnvelopeMarkerGroupDTO	88
PayloadEnvelopeNameIdDTO	88
PayloadEnvelopeOrganizationDTO	89
PayloadEnvelopePlatformDTO	89
PayloadEnvelopeProjectDTO	89
PayloadEnvelopeProtocolDTO	89
PayloadEnvelopeReferenceDTO	89
PayloadEnvelopeRestProfileDTO	90
PayloadExperimentDTO	90
PayloadExtractorInstructionFilesDTO	90
PayloadJobDTO	90
PayloadLoaderFilePreviewDTO	90
PayloadLoaderInstructionFilesDTO	91
PayloadManifestDTO	91
PayloadMapsetDTO	91
PayloadMarkerDTO	91
PayloadMarkerGroupDTO	91
PayloadNameIdDTO	92
PayloadOrganizationDTO	92
PayloadPlatformDTO	92
PayloadProjectDTO	92
PayloadProtocolDTO	92
PayloadReferenceDTO	93
PayloadRestProfileDTO	93

PlatformDTO	93
ProjectDTO	94
PropNameId	95
ProtocolDTO	95
ReferenceDTO	96
RestProfileDTO	97
ServerConfigItem	98
Status	98
TableColDisplay	98
VcfParameters	98
VendorProtocolDTO	99

Overview

The GOBii 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 GOBii 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

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;
    }

```

Ther request and response are almost always wrapped in a type-parameterized PayloadEvenlope. 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 entitites).

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 all. From the reference section for each web method in this document, youcan 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

Creates analysis entity

POST /gobii/v1/analyses

Description

Creates an analysis entity for GOBii system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeAnalysisDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Gets all the analyses

GET /gobii/v1/analyses

Description

Retrieves an unfiltered list of all Analysis entities. Consult the [Request/Response Section](#) before attempting to use this call.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Gets analysis by Analysis ID

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

Description

Retrieves the Analysis entity having the specified ID. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

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

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Updates analysis by Analysis ID

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

Description

Updates the Analysis entity having the specified analysisId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Authentication service for accessing GOBii System

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. this header and value be included in the request headers for subsequent requests. The token value is also supplied in the dtoHeaderAuth object.

Parameters

Type	Name	Schema
Body	body —	string

Responses

HTTP Code	Description	Schema
200	successful operation	string

Gets the configuration settings for GOBii system

GET /gobii/v1/configsettings

Description

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

/contacts-search

GET /gobii/v1/contact-search

Description

Gets contacts by searching 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

Adds new contact to GOBii system

POST /gobii/v1/contacts

Description

Adds new contact.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeContactDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Gets all the contacts in the GOBii system

```
GET /gobii/v1/contacts
```

Description

Gets all contacts.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeContactDTO

Gets contacts by contact ID

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

Description

Gets 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

Updates contacts by contact ID

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

Description

Updates contacts for a given 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

Gets contacts by email id

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

Description

Gets contacts 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

Gets CV terms by CV group ID

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

Description

Retrieves all the CV terms having the specified CV group ID

Parameters

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

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

Gets CV groups by CV group type ID

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

Description

Retrieves all CV groups having the specified 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

Gets CV Group details given CV group name and type ID

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

Description

Retrieves the details for a given CV Group name and type ID

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

/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

/cvs

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

Description

Gets all cvs in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeCvDTO

/cvs

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

Description

Gets 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

/cvs

PUT /gobii/v1/cvs/{cvId}

Description

Updates cv's 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

/cvs

DELETE /gobii/v1/cvs/{cvId}

Description

Deletes cv by the 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

/cvs/{groupName:[a-zA-Z_]+}

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

Description

Gets 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

Creates a dataset

```
POST /gobii/v1/datasets
```

Description

Creates a new dataset in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeDataSetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDataSetDTO

Gets all the datasets

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

Description

Retrieves 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

Gets the defined Dataset Types in the system

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

Description

Retrieves all the defined dataset types in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeNameIdDTO

Gets all the datasets with the given dataset type ID

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

Description

Retrieves all the Datasets having the specified 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

Gets the dataset given Dataset ID

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

Description

Retrieves 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

Updates the Dataset by Dataset ID

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

Description

Updates the Dataset entity having the specified datasetId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets all the analysis for the given Dataset ID

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

Description

Retrieves all the analysis for the given datasetId

Parameters

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

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

Gets the job details for the given Dataset ID

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

Description

Retrieves the 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

Creates a display entity

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

Description

Creates a display entity with the given information. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeDisplayDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Gets all the displays

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

Description

Retrieves a list of all the Display entities

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeDisplayDTO

Gets display by Display ID

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

Description

Retrieves the Display entity having the specified 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

Updates display by Display ID

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

Description

Updates the Display entity having the specified displayId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets all the entities

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

Description

Retrieves all the existing entities in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeEntityStatsDTO

Gets the total count of the children for the given Entity

```
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

Gets the total entity count for the given Entity Name

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

Description

Retrieves 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

Gets the last modified entity for the given Entity Name

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

Description

Retrieves 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

Creates an experiment

POST /gobii/v1/experiments

Description

Creates a new experiment in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeExperimentDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Gets all the experiments

```
GET /gobii/v1/experiments
```

Description

Retrieves all the existing experiments in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeExperimentDTO

Gets the experiment given Experiment ID

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

Description

Retrieves 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

Updates the Experiment by Experiment ID

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

Description

Updates the Experiment entity having the specified experimentId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets all the Protocols by Experiment ID

```
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

Gets file preview for the specified directory name


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

Description

Retrieves 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

Creates a Directory to be used for loading

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

Description

Creates 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

Upload file

```
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

Delete file

```
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`

Upload file for Job

```
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

Download file for specified Job

```
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

Creates instruction file for extracting data

```
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

Gets the extract job status by Job name

```
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

Gets the extract instruction information given file name

```
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

Creates instruction file for loading data

```
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

Gets the loading job status by Job name

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

Description

Retrieves 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

Gets the loading instruction information given file name

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

Description

Retrieves 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

Creates a job

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

Description

Creates a new job in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeJobDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Gets all the jobs

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

Description

Retrieves all the existing jobs in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Created DNA samples for a given Job

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

Description

Creates DNA samples for a given Job having the specified name in the system. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets the job given Job Name

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

Description

Retrieves 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

Updates the Job by Job Name

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

Description

Updates the Job entity having the specified jobName. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

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

Type	Name	Description	Schema
Body	body —		PayloadEnvelopeJobDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeJobDTO

Creates manifest entity

POST /gobii/v1/manifests

Description

Creates a Manifest entity for GOBii system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeManifestDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeManifestDTO

Gets all the Manifests

GET /gobii/v1/manifests

Description

Retrieves an unfiltered list of all Manifest entities.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeManifestDTO

Gets the manifest given Manifest ID

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

Description

Retrieves 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

Updates manifest by Manifest ID

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

Description

Updates the Manifest entity having the specified manifestId. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

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

Type	Name	Description	Schema
Body	body —		PayloadEnvelopeManifestDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeManifestDTO

Gets all the Mapsets

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

Description

Retrieves all the existing Mapsets in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMapsetDTO

Creates a mapset

```
POST /gobii/v1/mapsets
```

Description

Creates a new mapset in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMapsetDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MapsetDTO

Gets all the mapsets

```
GET /gobii/v1/mapsets
```

Description

Retrieves all the existing mapsets in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelope MapsetDTO

Gets the mapset given Mapset ID

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

Description

Retrieves 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

Updates the Mapset by Mapset ID

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

Description

Updates the Mapset entity having the specified mapsetId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Marker search

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

Description

Gets Marker by searching by name

Parameters

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

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Creates a Marker Group

POST /gobii/v1/markergroups

Description

Creates a new marker group in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMarkerGroupDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Gets all the marker groups

GET /gobii/v1/markergroups

Description

Retrieves all the existing marker groups in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Gets the marker group given Marker Group ID

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

Description

Retrieves the Marker Group entity having the specified ID.

Parameters

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

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Updates the Marker Group by Marker Group ID

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

Description

Updates the Marker Group entity having the specified markerGroupId. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Description	Schema
Path	markerGroupId —	ID of the Marker Group to be updated	integer (int32)
Body	body —		PayloadEnvelopeMarkerGroupDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerGroupDTO

Creates a marker

POST /gobii/v1/markers

Description

Creates a new marker in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeMarkerDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Gets all the markers

GET /gobii/v1/markers

Description

Retrieves all the existing markers in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeMarkerDTO

Gets the marker given Marker ID

```
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	PayloadEnvelopeMarkerDTO

Updates the Marker by Marker ID

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

Description

Updates the Marker entity having the specified markerId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets a list of name/ID combination for given Entity and a name list

```
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
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

Gets a list of name/ID combination for given Entity

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

Description

Retrieves a 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

Creates a organization

```
POST /gobii/v1/organizations
```

Description

Creates a new organization in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeOrganizationDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Gets all the organizations

```
GET /gobii/v1/organizations
```

Description

Retrieves all the existing organizations in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeOrganizationDTO

Gets the organization given Organization ID

```
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

Updates the Organization by Organization ID

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

Description

Updates the Organization entity having the specified organizationId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Creates a Platform

```
POST /gobii/v1/platforms
```

Description

Creates a new Platform in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopePlatformDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Gets all the platforms

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

Description

Retrieves all the existing platforms in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopePlatformDTO

Gets the platform details given Vendor Protocol ID

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

Description

Retrieves 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

Gets the platform given Platform ID

```
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

Updates the Platform by Platform ID

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

Description

Updates the Platform entity having the specified platformId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Creates a project

POST /gobii/v1/projects

Description

Creates a new project in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeProjectDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Gets all the projects

GET /gobii/v1/projects

Description

Retrieves all the existing projects in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProjectDTO

Gets the projects given Project ID

```
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

Updates the Project by Project ID

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

Description

Updates the Project entity having the specified projectId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Creates a Protocol

POST /gobii/v1/protocols

Description

Creates a new Protocol in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeProtocolDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Gets all the protocols

GET /gobii/v1/protocols

Description

Retrieves all the existing protocols in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeProtocolDTO

Gets the protocol given Protocol ID

```
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

Updates the Protocol by Protocol ID

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

Description

Updates the Protocol entity having the specified protocolId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Creates a Vendor Protocol record for given Protocol ID

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

Description

Creates a new Vendor Protocol in the system for specified Protocol ID. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets all the Vendor Protocols given Protocol ID

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

Description

Retrieves 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

Updates the Vendor Protocol by Protocol ID

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

Description

Updates the Vendor Protocol entity having the specified protocolId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Creates a reference

```
POST /gobii/v1/references
```

Description

Creates a new reference in the system. Consult the [Request/Response Section](#) before attempting to use this call.

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeReferenceDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Gets all the references

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

Description

Retrieves all the existing references in the system.

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeReferenceDTO

Gets the reference given Reference ID

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

Description

Retrieves 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

Updates the Reference by Reference ID

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

Description

Updates the Reference entity having the specified referenceId. Consult the [Request/Response Section](#) before attempting to use this call.

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

Gets REST profiles of GOBii system

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

Description

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 configuration document

Parameters

Type	Name	Schema
Body	body —	PayloadEnvelopeRestProfileDTO

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeRestProfileDTO

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)

Name	Schema
modifiedDate —	string (date-time)
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

Name	Schema
contactId —	integer (int32)
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

Name	Schema
entityStatus —	integer (int32)
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)

Name	Schema
callingAnalysisName —	string
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)

Name	Schema
modifiedDate —	string (date-time)
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
allowedProcessesTypes —		< enum (CREATE, READ, UPDATE, DELETE, NONE) > array

Name	Description	Schema
columnName —		string
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

Name	Schema
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

Name	Schema
experimentDataFile —	string
experimentId —	integer (int32)
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

Name	Description	Schema
absolutePath —		string
canonicalFile —		File
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

Name	Description	Schema
gobiiExtractFilterType —		enum (UNKNOWN, WHOLE_DATASET, BY_MARKER, BY_SAMPLE)
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)
gobiiSampleListType —		enum (GERMPLASM_NAM E, EXTERNAL_CODE, DNA_SAMPLE)
listFileName —		string
logMessage —		string

Name	Description	Schema
markerGroups —		< PropNameId > array
markerList —		< string > array
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

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

Name	Description	Schema
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
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

Name	Description	Schema
gobiiCropType —		string
gobiiFile —		GobiiFile
gobiiFileColumns —		< GobiiFileColumn > array
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)

Name	Description	Schema
logMessage —		string
mapset —		PropNameId
platform —		PropNameId
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

Name	Schema
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)

Name	Description	Schema
url —		string (url)

JobDTO

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

Name	Schema
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)
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, GERMPLASM, JOB)
id —	integer (int32)

Name	Schema
mapType —	integer (int32)
mapsetId —	integer (int32)
modifiedBy —	integer (int32)
modifiedDate —	string (date-time)
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

Name	Schema
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

Name	Schema
statusId —	integer (int32)

MarkerGroupMarkerDTO

Name	Description	Schema
favorableAllele —		string
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)

Name	Schema
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)

Name	Schema
vendorProtocols —	< VendorProtocolDTO > array
website —	string

Pagination

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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
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

Name	Schema
projectId —	integer (int32)
projectName —	string
projectStatus —	integer (int32)
properties —	< EntityPropertyDTO > array

PropNameId

Name	Schema
id —	integer (int32)
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

Name	Schema
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)

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