RNAget Compliance Documentation

Jeremy Adams, Sean Upchurch, Emilio Palumbo

Apr 11, 2023

CONTENTS

Welcome to the RNAget Server Compliance documentation.

This documentation (and associated compliance testing application) determines a server's compliance with the RNAget API specification for serving RNAseq datasets. The specification, developed by the Global Alliance for Genomics and Health, serves to provide a standardized API framework and data structure to allow for interoperability of datasets hosted at different institutions.

RNAGET SERVERS

RNAget Servers enable access to RNAseq datasets and their metadata using unique identifiers for each object in a hierarchical data model. Servers can be queried for expression matrices, as well as project and study information associated with the raw RNAseq data. Servers support the slicing of expression and continuous signal intensity matrices by various parameters, so that only subsets of large matrix files need to be transferred. Expression matrices can be subsetted by gene, sample, and/or expression value thresholds. Continuous signal intensity matrices can be subsetted by chromosome and start/end base positions.

COMPLIANCE DOCUMENT

This documentation is for implementers of RNAget servers. Implementers **MUST** adhere to this documentation during development of RNAget-compliant servers, as the compliance tests outlined herein conform with the RNAget specification. The testing suite performs API testing on all routes discussed in the specification. Reference server responses **MUST** comply with the correct responses in this document for requests made to each route/endpoint.

RNAget maintains an up-to-date report of the compliance status of its reference implementers. Click here to view the latest report.

Click here to learn more about the API Specification

TABLE OF CONTENTS

3.1 API Specification

The RNAget specification describes, at maximum, 18 API routes. These routes are subdivided into 4 groups according to the data type they serve. These routes, broken down by data type, are as follows:

1. Projects: Serves project-related information. 3 Routes

    - `/projects/:id`

    - `/projects`

    - `/projects/filters`

2. Studies: Serves study-related information. 3 Routes

    - `/studies/:id`

    - `/studies`

    - `/studies/filters`

3. Expressions: Handles requests for querying, slicing, and downloading expression matrix data by gene name/id. 6 Routes

    - `/expressions/formats`

    - `/expressions/filters`

    - `/expressions/:id/ticket`

    - `/expressions/:id/bytes`

    - `/expressions/ticket`

    - `/expressions/bytes`

4. Continuous: Handles requests for querying, slicing, and downloading expression matrix data by genomic coordinate. 6 Routes

    - `/continuous/formats`

    - `/continuous/filters`

    - `/continuous/:id/ticket`

    - `/continuous/:id/bytes`

    - `/continuous/ticket`

    - `/continuous/bytes`

Detailed descriptions of each route are available in the specification document.

Important points:

- A server **MAY** implement all 18 API routes

- A server **MAY** have unimplemented routes from the Expressions and/or Continuous data type groups. If a data type group is unimplemented, then **ALL** of the associated routes **MUST** be unplemented. In other words, a server **MUST NOT** implement only some routes for a data type group.

3.2 Test Dataset

This page contains links to test dataset files. The compliance suite expects the test dataset project, study, expression, and continuous resources to be available on the server and accessible by all relevant routes.

- Project json

  - `Download project.json`

  - id: 9c0eba51095d3939437e220db196e27b

  - Format: json

  - **How to Prepare for Compliance Testing**

    * upload `project.json` to RNAget server

    * `project.json` **MUST** be accessible by its id, ie. `/projects/9c0eba51095d3939437e220db196e27b`

    * `project.json` **MUST** appear in `/projects/search` results array

    * `project.json` **MUST** be filterable by implemented search filters

- Study json

  - `Download`

  - id: f3ba0b59bed0fa2f1030e7cb508324d1

  - Format: json

  - **How to Prepare for Compliance Testing**

    * upload `study.json` to RNAget server

    * `study.json` **MUST** be accessible by its id, ie. `/studies/f3ba0b59bed0fa2f1030e7cb508324d1`

    * `study.json` **MUST** appear in `/studies/search` results array

    * `study.json` **MUST** be filterable by implemented search filters

- Expression ticket json

  - `Download`

  - id: ac3e9279efd02f1c98de4ed3d335b98e

  - Format: json

  - **How to Prepare for Compliance Testing**

    * modify `expression_ticket.json url` property, so that it points to the test expression matrix file on the server

  * modify `expression_ticket.json` `fileType` property to indicate test expression matrix file format (ie. 'loom' if loom, 'tsv' if tsv)

  * upload modified `expression_ticket.json` to RNAget server

  * `expression_ticket.json` **MUST** be accessible by its `id`, ie. `/expressions/ac3e9279efd02f1c98de4ed3d335b98e/ticket`

  * a valid ticket must be returned for requests for an expression ticket pertaining to the compliance dataset (e.g. `/expressions/ticket?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1`

- Expression loom

  - `Download`

  - id: ac3e9279efd02f1c98de4ed3d335b98e

  - Format: loom

  - **How to Prepare for Compliance Testing**

    * upload `expression.loom` **ONLY** if `expression_ticket.json` `fileType` property was set to 'loom'

    * if uploaded, make `expression.loom` accessible by the url in the `url` property of `expression_ticket.json`

    * if uploaded, `expression.loom` must be sliceable by all subsetting parameters (featureIDList, featureNameList, sampleIDList)

- Expression tsv

  - `Download`

  - id: ac3e9279efd02f1c98de4ed3d335b98e

  - Format: tsv

  - **How to Prepare for Compliance Testing**

    * upload `expression.tsv` **ONLY** if `expression_ticket.json` `fileType` property was set to 'tsv'

    * if uploaded, make `expression.tsv` accessible by the url in the `url` property of `expression_ticket.json`

    * if uploaded, `expression.tsv` must be sliceable by all subsetting parameters (featureIDList, featureNameList, sampleIDList)

- Continuous ticket json

  - `Download`

  - id: 5e22e009f41fc53cbea094a41de8798f

  - Format: json

  - **How to Prepare for Compliance Testing**

    * modify `continuous_ticket.json` `url` property, so that it points to the test continuous matrix file on the server

    * modify `continuous_ticket.json` `fileType` property to indicate test continuous matrix file format (ie. 'loom' if loom, 'tsv' if tsv)

    * upload modified `continuous_ticket.json` to RNAget server

---

* `continuous_ticket.json` **MUST** be accessible by its `id`, ie. `/continuous/5e22e009f41fc53cbea094a41de8798f/ticket`

* a valid ticket must be returned for requests for a continuous ticket pertaining to the compliance dataset (e.g. `/continuous/ticket?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1`

- Continuous loom

  - `Download`

  - id: 5e22e009f41fc53cbea094a41de8798f

  - Format: loom

  - **How to Prepare for Compliance Testing**

    * upload `continuous.loom` **ONLY** if `continuous_ticket.json` `fileType` property was set to 'loom'

    * if uploaded, make `continuous.loom` accessible by the url in the `url` property of `continuous_ticket.json`

    * if uploaded, `continuous.loom` must be sliceable by all subsetting parameters (chr, start, end)

- Continuous tsv

  - `Download`

  - id: 5e22e009f41fc53cbea094a41de8798f

  - Format: tsv

  - **How to Prepare for Compliance Testing**

    * upload `continuous.tsv` **ONLY** if `continuous_ticket.json` `fileType` property was set to 'tsv'

    * if uploaded, make `continuous.tsv` accessible by the url in the `url` property of `continuous_ticket.json`

    * if uploaded, `continuous.tsv` must be sliceable by all subsetting parameters (chr, start, end)

- Filter json

  - `Download`

  - Format: json

  - **How to Prepare for Compliance Testing**

    * `filter.json` is an example, this file does not need to be uploaded, but requests for supported filters must respond with an array of valid `filter` objects

- Error json

  - `Download`

  - Format: json

  - **How to Prepare for Compliance Testing**

    * `error.json` is an example, this file does not need to be uploaded, but requests that produce an error must respond with a valid `Error` json object

    * valid `Error` json objects **MUST** at least contain a "message" property

3.3 Project Tests

This page outlines the success and failure criteria for all tests of Project resources.

3.3.1 Project API Tests

Project API tests assert the correct configuration of project-related API routes. In order for a test case to succeed, the following conditions must be met when evaluating the response:

- For all cases, `Content-Type` checking is enforced. The response **MUST** have a `Content-Type` header of `application/vnd.ga4gh.rnaget.v1.0.0+json` OR `application/json`

- For all cases, `Status Code` checking is enforced. The response **MUST** have the expected status code

- For all cases, schema checking is enforced. The json object in the response body **MUST** conform to a pre-defined schema of required fields and data types, which is specific to each API route

3.3.2 Project API Test Cases

- *Get Test Project*
- *Project Not Found*
- *Project Filters*
- *Search Projects Without Filters*
- *Search Projects With All Filters*
- *Search Projects With Single Filter, 1*
- *Search Projects With Single Filter, 2*
- *Project Search Filters Non-Matching Resources*

Get Test Project

- **Route:** `/projects/<id>`

- **Description:** Requests test project by its `id`. Expects the returned project to match the `Project` json schema.

- **Rationale:** Asserts that the `/projects/<id>` endpoint returns **one** valid project object, with id matching the request.

- **Request:**

```
GET /projects/9c0eba51095d3939437e220db196e27b
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
   "id": "9c0eba51095d3939437e220db196e27b",
   "version": "1.0",
   "name": "RNAgetTestProject0",
```

(continues on next page)

```
    "description": "Test project object used by RNAget compliance testing suite."
}
```

- **Success Criteria:** `Status Code == 200` AND response body is valid `Project` json
- **Failure Criteria:** `Status Code != 200` OR response body is NOT valid `Project` json

Project Not Found

- **Route:** `/projects/<id>`
- **Description:** Requests a project with an invalid `id`, that is, an `id` that does not correspond to any `Project` on the server. Expects a `404 Not Found` status code in the response, and a response body with a message explaining that the specified resource could not be found.
- **Rationale:** Asserts that the `/projects/<id>` endpoint does not return arbitrary `Project` objects, and only returns a `Project` when the `id` matches.
- **Request:**

```
GET /projects/nonexistentid999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Project Filters

- **Route:** `/projects/filters`
- **Description:** Requests the filters that can be used to narrow search results for a list of `Projects`
- **Rationale:** Asserts that the endpoint returns an array of `Filter` objects
- **Request:**

```
GET /projects/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "fieldType": "string",
```

```
    "values": [
      "1.0"
    ],
    "filter": "version",
    "description": "version to search for"
  },
  {
    "fieldType": "string",
    "values": [
      "PCAWG",
      "RNAgetTestProject0"
    ],
    "filter": "name",
    "description": "name of project"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Filters`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Filters`

Search Projects Without Filters

- **Route:** `/projects`

- **Description:** Searches for all projects, without specifying any filtering parameters. Expects an array of `Projects` in the response body.

- **Rationale:** Asserts that `/projects` returns an array, and that each element in the array is a `Project`.

- **Request:**

```
GET /projects
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "9c0eba51095d3939437e220db196e27b",
    "version": "1.0",
    "name": "RNAgetTestProject0",
    "description": "Test project object used by RNAget compliance testing suite."
  },
  {
    "description": "Pan Cancer Analysis of Whole Genomes test data from Expression␣
→Atlas E-MTAB-5423",
    "id": "43378a5d48364f9d8cf3c3d5104df560",
    "name": "PCAWG",
    "version": "1.0"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Project` json AND `Array Length >= 1`

3.3. Project Tests 13

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Project` json OR `Array Length < 1`

Search Projects With All Filters

- **Route:** `/projects`

- **Description:** Searches projects, using all filtering parameters associated with test project. Expects an array of `Projects` to be returned in the response body. Array must contain at least 1 object.

- **Rationale:** Asserts that `/projects` returns an array of `Projects` even when specifying filters. The returned array MUST have at least 1 object, as the parameter filters must match the attributes of the test project.

- **Request:**

```
GET /projects?version=1.0&name=RNAgetTestProject0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "9c0eba51095d3939437e220db196e27b",
    "version": "1.0",
    "name": "RNAgetTestProject0",
    "description": "Test project object used by RNAget compliance testing suite."
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Project` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Project` json OR `Array Length < 1`

Search Projects With Single Filter, 1

- **Route:** `/projects`

- **Description:** Searches projects using only 1 filtering parameter associated with test project. Expects an array of `Projects`, with length of 1 or greater.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields the test `Project` in the search results.

- **Requests:**

```
GET /projects?version=1.0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "9c0eba51095d3939437e220db196e27b",
    "version": "1.0",
    "name": "RNAgetTestProject0",
    "description": "Test project object used by RNAget compliance testing suite."
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Project` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Project` json OR `Array Length < 1`

Search Projects With Single Filter, 2

- **Route:** `/projects`

- **Description:** Searches projects using only 1 filtering parameter (a different filter than above) associated with test project. Expects an array of `Projects`, with length of 1 or greater.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields the test `Project` in the search results.

- **Requests:**

```
GET /projects?name=RNAgetTestProject0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "9c0eba51095d3939437e220db196e27b",
    "version": "1.0",
    "name": "RNAgetTestProject0",
    "description": "Test project object used by RNAget compliance testing suite."
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Project` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Project` json OR `Array Length < 1`

Project Search Filters Non-Matching Resources

- **Route:** `/projects`

- **Description:** Tests that the project search endpoint correctly filters out non-matching `Projects` based on url parameters. Makes a request to the `/projects` endpoint with invalid filters (not matching any `Project`), and expects an empty array as a response.

- **Rationale:** Asserts that the endpoint correctly filters out non-matching entities, that the endpoint does not return an arbitrary list of `Projects` that differ from filters.

- **Request:**

```
GET /projects?version=nonexistentid999999999999999999&
↪name=nonexistentid999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[]
```

- **Success Criteria:** `Status Code == 200` AND response body is an empty array
- **Failure Criteria:** `Status Code != 200` OR response body is NOT an empty array

3.3.3 Project API Non-Implemented Test Cases

Project Get Not Implemented

- **Route:** `/projects/<id>`

- **Description:** If the `Projects` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/projects/<id>` endpoint, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Project` related endpoints are correctly non-implemented according to the spec

- **Request:**

```
GET /projects/nonexistentid999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Project Search Not Implemented

- **Route:** `/projects`
- **Description:** If the `Projects` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/projects` endpoint, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts that `Project` related endpoints are correctly non-implemented according to the spec
- **Request:**

```
GET /projects
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Project Filters Not Implemented

- **Route:** `/projects/filters`
- **Description:** If the `Projects` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/projects/filters` endpoint, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts that `Project` related endpoints are correctly non-implemented according to the spec
- **Request:**

```
GET /projects/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

3.4 Study Tests

This page outlines the success and failure criteria for all tests of Study resources.

3.4.1 Study API Tests

Study API tests assert the correct configuration of study-related API routes. In order for a test case to succeed, the following conditions must be met when evaluating the response:

- For all cases, `Content-Type` checking is enforced. The response **MUST** have a `Content-Type` header of `application/vnd.ga4gh.rnaget.v1.0.0+json` OR `application/json`

- For all cases, `Status Code` checking is enforced. The response **MUST** have the expected status code

- For all cases, schema checking is enforced. The json object in the response body **MUST** conform to a pre-defined schema of required fields and data types, which is specific to each API route

3.4.2 Study API Test Cases

Get Test Study

- **Route:** `/studies/<id>`

- **Description:** Requests test study by its `id`. Expects the returned study to match the `Study` json schema.

- **Rationale:** Asserts that the `/studies/<id>` endpoint returns **one** valid study object, with id matching the request.

- **Request:**

```
GET /studies/f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "id": "f3ba0b59bed0fa2f1030e7cb508324d1",
  "version": "1.0",
  "name": "RNAgetTestStudy0",
  "description": "Test study object used by RNAget compliance testing suite.",
  "parentProjectID": "9c0eba51095d3939437e220db196e27b"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is valid `Study` json

- **Failure Criteria:** `Status Code != 200` OR response body is NOT valid `Study` json

Study Not Found

- **Route:** `/studies/<id>`

- **Description:** Requests a study with an invalid `id`, that is, an `id` that does not correspond to any `Study` on the server. Expects a `404 Not Found` status code in the response, and a response body with a message explaining that the specified resource could not be found.

- **Rationale:** Asserts that the `/studies/<id>` endpoint does not return arbitrary `Study` objects, and only returns a `Study` when the `id` matches.

- **Request:**

```
GET /studies/nonexistentid9999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Study Filters

- **Route:** `/studies/filters`

- **Description:** Requests the filters that can be used to narrow search results for a list of `Studies`

- **Rationale:** Asserts that the endpoint returns an array of `filter` objects

- **Request:**

```
GET /studies/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "fieldType": "string",
    "values": [
      "1.0"
    ],
    "filter": "version",
    "description": "version to search for"
  },
  {
    "fieldType": "string",
    "values": [
```

```
      "PCAWG",
      "RNAGetTestStudy0"
    ],
    "filter": "name",
    "description": "name of study"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `filters`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `filters`

Search Studies Without Filters

- **Route:** `/studies`

- **Description:** Searches for all studies, without specifying any filtering parameters. Expects an array of `Studies` in the response body.

- **Rationale:** Asserts that the `/studies` returns an array, and that each element in the array is a `Study`.

- **Request:**

```
GET /studies
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "6cccbbd76b9c4837bd7342dd616d0fec",
    "parentProjectID": "43378a5d48364f9d8cf3c3d5104df560",
    "name": "PCAWG",
    "description": "PCAWG study",
    "version": "1.0"
  },
  {
    "id": "f3ba0b59bed0fa2f1030e7cb508324d1",
    "parentProjectID": "9c0eba51095d3939437e220db196e27b",
    "name": "RNAgetTestStudy0",
    "description": "Test study object used by RNAget compliance testing suite.",
    "version": "1.0"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Study` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Study` json OR `Array Length < 1`

Search Studies With All Filters

- **Route:** `/studies`

- **Description:** Searches studies, using all filtering parameters associated with test study. Expects an array of `Studies` to be returned in the response body. Array must contain at least 1 object.

- **Rationale:** Asserts that `/studies` returns an array of `Studies` even when specifying filters. The returned array MUST have at least 1 object, as the parameter filters must match the attributes of the test study.

- **Request:**

```
GET /studies?version=1.0&name=RNAgetTestStudy0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "f3ba0b59bed0fa2f1030e7cb508324d1",
    "parentProjectID": "9c0eba51095d3939437e220db196e27b",
    "name": "RNAgetTestStudy0",
    "description": "Test study object used by RNAget compliance testing suite.",
    "version": "1.0"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Study` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Study` json OR `Array Length < 1`

Search Studies With Single Filter, 1

- **Route:** `/studies`

- **Description:** Searches studies using only 1 filtering parameter associated with test study. Expects an array of `Studies`, with length of 1 or greater.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields the test `Study` in the search results.

- **Requests:**

```
GET /studies?version=1.0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "f3ba0b59bed0fa2f1030e7cb508324d1",
    "parentProjectID": "9c0eba51095d3939437e220db196e27b",
    "name": "RNAgetTestStudy0",
    "description": "Test study object used by RNAget compliance testing suite.",
```

```
    "version": "1.0"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Study` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Study` json OR `Array Length < 1`

Search Studies With Single Filter, 2

- **Route:** `/studies`

- **Description:** Searches studies using only 1 filtering parameter (a different filter than above) associated with test study. Expects an array of `Studies`, with length of 1 or greater.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields the test `Study` in the search results.

- **Requests:**

```
GET /studies?name=RNAgetTestStudy0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "id": "f3ba0b59bed0fa2f1030e7cb508324d1",
    "parentProjectID": "9c0eba51095d3939437e220db196e27b",
    "name": "RNAgetTestStudy0",
    "description": "Test study object used by RNAget compliance testing suite.",
    "version": "1.0"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `Study` json AND `Array Length >= 1`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `Study` json OR `Array Length < 1`

Study Search Filters Non-Matching Resources

- **Route:** `/studies`

- **Description:** Tests that the study search endpoint correctly filters out non-matching `Studies` based on url parameters. Makes a request to the `/studies` endpoint with invalid filters (not matching any `Study`), and expects an empty array as a response.

- **Rationale:** Asserts that the endpoint correctly filters out non-matching entities, that the endpoint does not return an arbitrary list of `Studies` that differ from filters.

• **Request:**

```
GET /studies?version=nonexistentid999999999999999999&
↪name=nonexistentid999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

• **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[]
```

• **Success Criteria:** `Status Code == 200` AND response body is an empty array

• **Failure Criteria:** `Status Code != 200` OR response body is NOT an empty array

3.4.3 Study API Non-Implemented Test Cases

• *Study Get Not Implemented*

• *Study Search Not Implemented*

• *Study Search Filters Not Implemented*

Study Get Not Implemented

• **Route:** `/studies/<id>`

• **Description:** If the `Studies` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/studies/<id>` endpoint, expecting a `501 Not Implemented` status code response

• **Rationale:** Asserts that `Study` related endpoints are correctly non-implemented according to the spec

• **Request:**

```
GET /studies/nonexistentid999999999999999999
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

• **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

• **Success Criteria:** `Status Code == 501`

• **Failure Criteria:** `Status Code != 501`

Study Search Not Implemented

• **Route:** `/studies`

• **Description:** If the `Studies` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/studies` endpoint, expecting a `501 Not Implemented` status code response

• **Rationale:** Asserts that `Study` related endpoints are correctly non-implemented according to the spec

• **Request:**

```
GET /studies
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

• **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

• **Success Criteria:** `Status Code == 501`
• **Failure Criteria:** `Status Code != 501`

Study Search Filters Not Implemented

• **Route:** `/studies/filters`

• **Description:** If the `Studies` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/studies/filters` endpoint, expecting a `501 Not Implemented` status code response

• **Rationale:** Asserts that `Study` related endpoints are correctly non-implemented according to the spec

• **Request:**

```
GET /studies/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

• **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

• **Success Criteria:** `Status Code == 501`
• **Failure Criteria:** `Status Code != 501`

3.5 Expression Tests

This page outlines the success and failure criteria for all tests of Expression resources.

3.5.1 Expression API Tests

Expression API tests assert the correct configuration of expression-related API routes. In order for a test case to succeed, the following conditions must be met when evaluating the response:

• For all cases, `Content-Type` checking is enforced. The response **MUST** have a `Content-Type` header of `application/vnd.ga4gh.rnaget.v1.0.0+json` OR `application/json`

• For all cases, `Status Code` checking is enforced. The response **MUST** have the expected status code

• For all cases, schema checking is enforced. The json object in the response body **MUST** conform to a predefined schema of required fields and data types, which is specific to each API route

3.5.2 Expression API Test Cases

- *Get Supported Expression Formats*
- *Expression Filters*
- *Get Test Expression Ticket*
- *Single Expression Ticket - Not Found*
- *Get Test Expression Bytes*
- *Single Expression Bytes - Not Found*
- *Expression Ticket by Format*
- *Expression Ticket - All Filters*
- *Expression Ticket - Single Filter, 1*
- *Expression Ticket - Single Filter, 2*
- *Expression Ticket - Format Not Specified*
- *Expression Ticket - Filetype Matches*
- *Expression Bytes by Format*
- *Expression Bytes - All Filters*
- *Expression Bytes - Single Filter, 1*
- *Expression Bytes - Single Filter, 2*
- *Expression Bytes - Format Not Specified*

Get Supported Expression Formats

- **Route:** `/expressions/formats`
- **Description:** Requests the available expression data file formats on the server. Expects an array of strings to be returned in the response body.
- **Rationale:** Asserts that `/expressions/formats` returns an array of strings, indicating which expression file formats the server supports
- **Request:**

```
GET /expressions/formats
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  "loom",
  "tsv"
]
```

- **Success Criteria:** `Status Code == 200` AND response body is an array of strings in json format
- **Failure Criteria:** `Status Code != 200` OR response body is NOT an array of strings in json format

Expression Filters

- **Route:** `/expressions/filters`

- **Description:** Requests the filters that can be used to narrow search results for a list of `Expressions`

- **Rationale:** Asserts that the endpoint returns an array of `filter` objects

- **Request:**

```
GET /expressions/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "fieldType": "string",
    "values": [
      "1.0"
    ],
    "filter": "version",
    "description": "version to search for"
  },
  {
    "fieldType": "string",
    "filter": "studyID",
    "description": "parent studyID"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `filters`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `filters`

Get Test Expression Ticket

- **Route:** `/expressions/<id>/ticket`

- **Description:** Requests test expression `ticket` by its `id`. Expects the returned expression to match the `ticket` json schema.

- **Rationale:** Asserts that the `/expressions/<id>/ticket` endpoint returns a valid `ticket`.

- **Request:**

```
GET /expressions/ac3e9279efd02f1c98de4ed3d335b98e/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "id": "ac3e9279efd02f1c98de4ed3d335b98e",
```

```
  "version": "1.0",
  "studyID": "f3ba0b59bed0fa2f1030e7cb508324d1",
  "url": "https://url/to/expression/file",
  "units": "TPM",
  "fileType": "loom/tsv"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is valid `ticket`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT valid `ticket`

Single Expression Ticket - Not Found

- **Route:** `/expressions/<id>/ticket`

- **Description:** Requests an expression ticket with an invalid `id`, that is, an `id` that does not correspond to any `Expression` on the server. Expects a `404 Not Found` status code in the response, and a response body with an error message.

- **Rationale:** Asserts that the `/expressions/<id>/ticket` endpoint does not return arbitrary `ticket`.

- **Request:**

```
GET /expressions/nonexistentid999999999999999999/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json

- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Get Test Expression Bytes

- **Route:** `/expressions/<id>/bytes`

- **Description:** Requests test expression matrix as `bytes` by its `id`.

- **Rationale:** Asserts that the `/expressions/<id>/bytes` endpoint returns matrix bytes.

- **Request:**

```
GET /expressions/ac3e9279efd02f1c98de4ed3d335b98e/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`

- **Failure Criteria:** `Status Code != 200`

Single Expression Bytes - Not Found

- **Route:** `/expressions/<id>/bytes`

- **Description:** Requests expression matrix `bytes` with an invalid `id` that doesn't correspond to any `Expression` on the server. Expects a `404 Not Found` status code and an error message in response body.

- **Rationale:** Asserts that the `/expressions/<id>/bytes` endpoint does not return arbitrary matrix.

- **Request:**

```
GET /expressions/nonexistentid999999999999999999/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json


{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Expression Ticket by Format

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix, specifying only the required 'format' parameter. Expects `ticket` response.

- **Rationale:** Asserts that the `/expressions/ticket` returns a valid `ticket`.

- **Request:**

```
GET /expressions/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

 {
   "studyID": "6cccbbd76b9c4837bd7342dd616d0fec",
   "url": "/path/to/E-MTAB-5423-query-results.tpms.loom",
   "units": "TPM",
   "version": "1.0",
   "fileType": "loom"
 }
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`
- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`

Expression Ticket - All Filters

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix, using all filtering parameters associated with test expression. Expects `ticket`.

- **Rationale:** Asserts that `/expressions/ticket` returns `ticket` when specifying filters.

- **Request:**

```
GET /expressions/ticket?format=loom&version=1.0&
→studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

 {
   "url": "https://path/to/expression.loom",
   "units": "TPM",
   "fileType": "loom"
 }
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`
- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`

Expression Ticket - Single Filter, 1

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix using only 1 filtering parameter associated with test expression (in addition to format). Expects `ticket`.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields a valid `ticket`.

- **Requests:**

```
GET /expressions/ticket?format=loom&version=1.0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

 {
   "url": "https://path/to/expression.loom",
   "units": "TPM",
   "version": "1.0",
   "fileType": "loom"
 }
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`
- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`

Expression Ticket - Single Filter, 2

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix using only 1 filtering parameter (a different filter than above) associated with test expression (in addition to format). Expects `ticket`.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields a valid `ticket`.

- **Requests:**

```
GET /expressions/ticket?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

 {
   "studyID": "f3ba0b59bed0fa2f1030e7cb508324d1",
   "url": "/path/to/expression.loom",
   "units": "TPM",
   "fileType": "loom"
 }
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`.

- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`.

Expression Ticket - Format Not Specified

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix WITHOUT specifying the required `format` parameter. Expects a `4xx` response with error message.

- **Rationale:** As the `format` parameter is required to specify file format for the `/expressions/ticket` endpoint, this test asserts malformed requests raise an error.

- **Request:**

```
GET /expressions/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Input payload validation failed"
}
```

- **Success Criteria:** `Status Code == 4xx` AND response body is valid `Error` json

- **Failure Criteria:** `Status Code != 4xx` AND response body is NOT valid `Error` json

Expression Ticket - Filetype Matches

- **Route:** `/expressions/ticket`

- **Description:** Requests joined matrix, only specifying the required `format` parameter. Checks that `ticket` has a `fileType` matching requested `format`.

- **Rationale:** Asserts that the `/expressions/ticket` endpoint returns `ticket` with fileType matching the request.

- **Request:**

```
GET /expressions/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

 {
   "url": "/path/to/E-MTAB-5423-query-results.tpms.loom",
   "units": "TPM",
   "fileType": "loom"
 }
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket` AND `fileType` matches requested `format`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket` OR `fileType` DOES NOT match requested `format`

Expression Bytes by Format

- **Route:** `/expressions/bytes`

- **Description:** Requests joined matrix bytes, only specifying the required `format` parameter.

- **Rationale:** Asserts that the `/expressions/bytes` endpoint returns matrix bytes.

- **Request:**

```
GET /expressions/bytes?format=loom
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`

- **Failure Criteria:** `Status Code != 200`

Expression Bytes - All Filters

- **Route:** `/expressions/bytes`

- **Description:** Requests joined matrix bytes using all expression filters.

- **Rationale:** Asserts that the `/expressions/bytes` endpoint returns matrix bytes when all filters are provided.

- **Request:**

```
GET /expressions/bytes?format=loom&version=1.0&
↪studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`
- **Failure Criteria:** `Status Code != 200`

Expression Bytes - Single Filter, 1

- **Route:** `/expressions/bytes`
- **Description:** Requests joined matrix bytes using one expression filter (in addition to format).
- **Rationale:** Asserts filter parameters can be used independently of one another.
- **Request:**

```
GET /expressions/bytes?format=loom&version=1.0
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`
- **Failure Criteria:** `Status Code != 200`

Expression Bytes - Single Filter, 2

- **Route:** `/expressions/bytes`
- **Description:** Requests joined matrix bytes using one expression filter (different than above), in addition to format.
- **Rationale:** Asserts filter parameters can be used independently of one another.
- **Request:**

```
GET /expressions/bytes?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`

- **Failure Criteria:** `Status Code != 200`

Expression Bytes - Format Not Specified

- **Route:** `/expressions/bytes`

- **Description:** Requests joined matrix bytes WITHOUT specifying required `format` parameter. Expects a `4xx` response with error message.

- **Rationale:** As the `format` parameter is required to specify file format for the `/expressions/bytes` endpoint, this test asserts malformed requests raise an error.

- **Request:**

```
GET /expressions/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Input payload validation failed"
}
```

- **Success Criteria:** `Status Code == 4xx` AND response body is valid `Error` json

- **Failure Criteria:** `Status Code != 4xx` AND response body is NOT valid `Error` json

3.5.3 Expression API Non-Implemented Test Cases

- *Expression Formats Not Implemented*
- *Expression Ticket by Id Not Implemented*
- *Expression Bytes by Id Not Implemented*
- *Expression Filters Not Implemented*
- *Expression Ticket Not Implemented*
- *Expression Bytes Not Implemented*

Expression Formats Not Implemented

- **Route:** `/expressions/formats`

- **Description:** If the `Expressions` endpoint is specified as `Not Implemented` in the config file, then this test will be run. Requests the `/expressions/formats` endpoint, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Expression` related endpoints are correctly non-implemented according to the spec

- **Request:**

```
GET /expressions/formats
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Expression Ticket by Id Not Implemented

- **Route:** `/expressions/<id>/ticket`
- **Description:** If expressions endpoints are `Not Implemented`, this test will request `/expressions/<id>/ticket`, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts `Expression` related endpoints are correctly non-implemented
- **Request:**

```
GET /expressions/nonexistentid999999999999999999/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Expression Bytes by Id Not Implemented

- **Route:** `/expressions/<id>/bytes`
- **Description:** If expressions endpoints are `Not Implemented`, this test will request `/expressions/<id>/bytes`, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts `Expression` related endpoints are correctly non-implemented
- **Request:**

```
GET /expressions/nonexistentid999999999999999999/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Expression Filters Not Implemented

- **Route:** `/expressions/filters`

- **Description:** If expressions endpoints are `Not Implemented`, this test will request `/expressions/filters`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Expression` related endpoints are correctly non-implemented

- **Request:**

```
GET /expressions/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

Expression Ticket Not Implemented

- **Route:** `/expressions/ticket`

- **Description:** If expressions endpoints are `Not Implemented`, this test will request `/expressions/ticket`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Expression` related endpoints are correctly non-implemented

- **Request:**

```
GET /expressions/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

Expression Bytes Not Implemented

- **Route:** `/expressions/bytes`

- **Description:** If expressions endpoints are `Not Implemented`, this test will request `/expressions/bytes`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Expression` related endpoints are correctly non-implemented

- **Request:**

```
GET /expressions/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

3.5.4 Expression Content Tests

Expression content tests assert that expression matrices downloaded from the RNAget server contain the expected content/results based on the request. Matrix rows, columns, and values are cross-referenced against the request to ensure the correct data has been returned.

3.5.5 Expression Content Test Cases

- *Expression Ticket by Id Content Test Cases*
- *Expression Bytes by Id Content Test Cases*
- *Expression Ticket Content Test Cases*
- *Expression Bytes Content Test Cases*

Expression Ticket by Id Content Test Cases

- **Route:** `/expressions/<id>/ticket`

- **Description:** Download test expression by ticket multiple times (sometimes slicing by featureIDList, featureNameList, sampleIDList).

- **Rationale:** Asserts correct matrix file is associated with the test expression `id`. Validates expression values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /expressions/ac3e9279efd02f1c98de4ed3d335b98e/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Test expression matrix columns, rows, values match expected

- **Failure Criteria:** Test expression matrix columns, rows, values DO NOT match expected

Expression Bytes by Id Content Test Cases

- **Route:** `/expressions/<id>/bytes`

- **Description:** Download test expression by bytes multiple times (sometimes slicing by featureIDList, featureNameList, sampleIDList).

- **Rationale:** Asserts correct matrix file is associated with the test expression `id`. Validates expression values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /expressions/ac3e9279efd02f1c98de4ed3d335b98e/bytes
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Test expression matrix columns, rows, values match expected

- **Failure Criteria:** Test expression matrix columns, rows, values DO NOT match expected

Expression Ticket Content Test Cases

- **Route:** `/expressions/ticket`

- **Description:** Download joined expression matrix by ticket multiple times (sometimes slicing by featureIDList, featureNameList, sampleIDList).

- **Rationale:** Asserts joined matrix. Validates expression values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /expressions/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Joined expression matrix columns, rows, values match expected

- **Failure Criteria:** Joined expression matrix columns, rows, values DO NOT match expected

Expression Bytes Content Test Cases

- **Route:** `/expressions/bytes`

- **Description:** Download joined expression matrix by bytes multiple times (sometimes slicing by featureIDList, featureNameList, sampleIDList).

- **Rationale:** Asserts joined matrix. Validates expression values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /expressions/bytes?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Joined expression matrix columns, rows, values match expected

- **Failure Criteria:** Joined expression matrix columns, rows, values DO NOT match expected

3.6 Continuous Tests

This page outlines the success and failure criteria for all tests of Continuous resources.

3.6.1 Continuous API Tests

Continuous API tests assert the correct configuration of continuous-related API routes. In order for a test case to succeed, the following conditions must be met when evaluating the response:

- For all cases, `Content-Type` checking is enforced. For most routes, a value of `application/vnd.ga4gh.rnaget.v1.0.0+json` OR `application/json` is expected. For the `/continuous/<id>` endpoint, a content type matching the format of the file attachment is expected.

- For all cases, `Status Code` checking is enforced. The response **MUST** have the expected status code

- For all cases, schema checking is enforced. The json object in the response body **MUST** conform to a pre-defined schema of required fields and data types, which is specific to each API route

3.6.2 Continuous API Test Cases

Get Supported Continuous Formats

- **Route:** `/continuous/formats`

- **Description:** Requests the available continuous file formats on the server. Expects an array of strings to be returned in the response body.

- **Rationale:** Asserts that `/continuous/formats` returns an array of strings, indicating which continuous file formats the server supports

- **Request:**

```
GET /continuous/formats
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  "loom",
  "tsv"
]
```

- **Success Criteria:** `Status Code == 200` AND response body is an array of strings in json format

- **Failure Criteria:** `Status Code != 200` OR response body is NOT an array of strings in json format

Continuous Filters

- **Route:** `/continuous/filters`

- **Description:** Requests the filters that can be used to narrow search results for a list of `Continuous`

- **Rationale:** Asserts that the endpoint returns an array of `filters`

- **Request:**

```
GET /continuous/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

[
  {
    "fieldType": "string",
    "values": [
      "1.0"
    ],
    "filter": "version",
    "description": "version to search for"
  },
  {
    "fieldType": "string",
    "filter": "studyID",
```

(continues on next page)

```
    "description": "parent studyID"
  }
]
```

- **Success Criteria:** `Status Code == 200` AND response body is array of `filters`
- **Failure Criteria:** `Status Code != 200` OR response body is NOT array of `filters`

Get Test Continuous Ticket

- **Route:** `/continuous/<id>/ticket`
- **Description:** Requests continuous ticket by its `id`. Expects valid `ticket`.
- **Rationale:** Asserts that `/continuous/<id>/ticket` returns valid `ticket`.
- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "id": "5e22e009f41fc53cbea094a41de8798f",
  "version": "1.0",
  "studyID": "f3ba0b59bed0fa2f1030e7cb508324d1",
  "url": "https://url/to/continuous/file",
  "units": "TPM",
  "fileType": "loom/tsv"
}
```

- **Success Criteria:** `Status Code == 200` AND response is `ticket`
- **Failure Criteria:** `Status Code != 200` OR response is NOT `ticket`

Single Continuous Ticket - Not Found

- **Route:** `/continuous/<id>/ticket`
- **Description:** Requests continuous ticket with an invalid `id`. Expects a `404 Not Found` status code in the response, and error message.
- **Rationale:** Asserts that the `/continuous/<id>/ticket` endpoint does not return arbitrary `ticket`.
- **Request:**

```
GET /continuous/nonexistentid999999999999999999/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json

- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Single Continuous Ticket - Start Specified Without Chr

- **Route:** `/continuous/<id>/ticket`

- **Description:** Requests continuous ticket by its id, specifying a start position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/<id>/ticket` endpoint raises an error when `start` is specified without `chr`

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/ticket?start=5
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json

- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Single Continuous Ticket - End Specified Without Chr

- **Route:** `/continuous/<id>/ticket`

- **Description:** Requests test continuous by its id, specifying an end position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/<id>/ticket` endpoint raises an error when `end` is specified without `chr`

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/ticket?end=1000
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Single Continuous Ticket - Start Greater Than End

- **Route:** `/continuous/<id>/ticket`
- **Description:** Requests test continuous ticket by its id, specifying `chr`, `start`, and `end`, however, `start` position is greater than `end`. Expects a `400 Bad Request` status code in the response, and an error message
- **Rationale:** Asserts that the `/continuous/<id>/ticket` endpoint raises an error when `start` is greater than `end`
- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/ticket?chr=1&start=200&end=100
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "start cannot be greater than end"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Get Test Continuous Bytes

- **Route:** `/continuous/<id>/bytes`
- **Description:** Requests continuous matrix bytes by its id.
- **Rationale:** Asserts that `/continuous/<id>/bytes` returns matrix bytes.
- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`

• **Failure Criteria:** `Status Code != 200`

Single Continuous Bytes - Not Found

- **Route:** `/continuous/<id>/bytes`

- **Description:** Requests continuous bytes with an invalid `id`. Expects a `404 Not Found` status code in the response, and error message.

- **Rationale:** Asserts that the `/continuous/<id>/bytes` endpoint does not return arbitrary matrix.

- **Request:**

```
GET /continuous/nonexistentid999999999999999999/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 404 Not Found
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json


{
  "message": "Entry not found in database."
}
```

- **Success Criteria:** `Status Code == 404` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 404` OR response body is NOT valid `Error` json

Single Continuous Bytes - Start Specified Without Chr

- **Route:** `/continuous/<id>/bytes`

- **Description:** Requests continuous matrix bytes by its id, specifying a start position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/<id>/bytes` endpoint raises an error when `start` is specified without `chr`

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/bytes?start=5
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json


{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Single Continuous Bytes - End Specified Without Chr

- **Route:** `/continuous/<id>/bytes`

- **Description:** Requests test continuous matrix bytes by its id, specifying an end position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/<id>/bytes` endpoint raises an error when `end` is specified without `chr`

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/bytes?end=1000
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json


{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Single Continuous Bytes - Start Greater Than End

- **Route:** `/continuous/<id>/bytes`

- **Description:** Requests test continuous matrix bytes by its id, specifying `chr`, `start`, and `end`, however, `start` position is greater than `end`. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/<id>/bytes` endpoint raises an error when `start` is greater than `end`

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/bytes?chr=1&start=200&end=100
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json


{
  "message": "start cannot be greater than end"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Ticket by Format

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix, specifying only the required 'format' parameter. Expects `ticket`.

- **Rationale:** Asserts that `/continuous/ticket` returns a `ticket` with which the matrix can be downloaded.

- **Request:**

```
GET /continuous/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "studyID": "6cccbbd76b9c4837bd7342dd616d0fec",
  "url": "/path/to/signal-query-results.loom",
  "units": "TPM",
  "version": "1.0",
  "fileType": "loom"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`

Continuous Ticket - All Filters

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix, using all filtering parameters. Expects `ticket`.

- **Rationale:** Asserts that `/continuous/ticket` returns `ticket` when specifying filters.

- **Request:**

```
GET /continuous/ticket?format=loom&version=1.0&
↪studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "studyID": "f3ba0b59bed0fa2f1030e7cb508324d1",
  "url": "/path/to/continuous.loom",
  "units": "TPM",
  "version": "1.0",
  "fileType": "loom"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`

Continuous Ticket - Single Filter, 1

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix, using only 1 filtering parameter associated with test continuous (in addition to format). Expects `ticket`.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields a valid `ticket`.

- **Requests:**

```
GET /continuous/ticket?format=loom&version=1.0
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "url": "/path/to/signal-query-results.loom",
  "units": "TPM",
  "version": "1.0",
  "fileType": "loom"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`.
- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`.

Continuous Ticket - Single Filter, 2

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix, using only 1 filtering parameter (different than above) associated with test continuous (in addition to format). Expects `ticket`.

- **Rationale:** Asserts filtering parameters can be used independently of one another, and that each filter yields a valid `ticket`.

- **Requests:**

```
GET /continuous/ticket?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "studyID": "6cccbbd76b9c4837bd7342dd616d0fec",
  "url": "/path/to/signal-query-results.loom",
  "units": "TPM",
```

```
  "fileType": "loom"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket`.
- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket`.

Continuous Ticket - Format Not Specified

- **Route:** `/continuous/ticket`
- **Description:** Requests joined continuous matrix WITHOUT specifying the required `format` parameter. Expects a `4xx` response with error message.
- **Rationale:** As the `format` parameter is required to specify file format for the `/continuous/ticket` endpoint, this test asserts malformed requests raise an error.
- **Request:**

```
GET /continuous/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Input payload validation failed"
}
```

- **Success Criteria:** `Status Code == 4xx` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 4xx` AND response body is NOT valid `Error` json

Continuous Ticket - Filetype Matches

- **Route:** `/continuous/ticket`
- **Description:** Request joined continuous matrix, only specifying the required `format` parameter. Checks that `ticket` has a `fileType` matching requested `format`.
- **Rationale:** Asserts that the `/continuous/ticket` endpoint returns `ticket` with fileType matching the request.
- **Request:**

```
GET /continuous/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "studyID": "6cccbbd76b9c4837bd7342dd616d0fec",
```

```
  "url": "/woldlab/castor/home/sau/public_html/rnaget/signal-query-results.loom",
  "units": "TPM",
  "version": "1.0",
  "fileType": "loom"
}
```

- **Success Criteria:** `Status Code == 200` AND response body is `ticket` AND `fileType` matches requested `format`

- **Failure Criteria:** `Status Code != 200` OR response body is NOT `ticket` OR `fileType` DOES NOT match requested `format`

Continuous Ticket - Start Specified Without Chr

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix ticket, specifying a start position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/ticket` endpoint raises an error when `start` is specified without `chr`

- **Request:**

```
GET /continuous/ticket?format=loom&start=5
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Ticket - End Specified Without Chr

- **Route:** `/continuous/ticket`

- **Description:** Requests joined continuous matrix ticket, specifying an end position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message

- **Rationale:** Asserts that the `/continuous/ticket` endpoint raises an error when `end` is specified without `chr`

- **Request:**

```
GET /continuous/ticket?format=loom&end=1000
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Ticket - Start Greater Than End

- **Route:** `/continuous/ticket`
- **Description:** Requests joined continuous matrix ticket, specifying `chr`, `start`, and `end`, however, `start` position is greater than `end`. Expects a `400 Bad Request` status code in the response, and an error message
- **Rationale:** Asserts that the `/continuous/ticket` endpoint raises an error when `start` is greater than `end`
- **Request:**

```
GET /continuous/ticket?format=loom&chr=1&start=200&end=100
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "start cannot be greater than end"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Bytes by Format

- **Route:** `/continuous/bytes`
- **Description:** Requests joined continuous matrix as bytes, specifying only the required 'format' parameter.
- **Rationale:** Asserts that `/continuous/bytes` returns matrix bytes.
- **Request:**

```
GET /continuous/bytes?format=loom
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`

- **Failure Criteria:** `Status Code != 200`

Continuous Bytes - All Filters

- **Route:** `/continuous/bytes`

- **Description:** Requests joined continuous matrix as bytes, using all filtering parameters.

- **Rationale:** Asserts that `/continuous/ticket` returns matrix bytes when specifying filters.

- **Request:**

```
GET /continuous/bytes?format=loom&version=1.0&studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`
- **Failure Criteria:** `Status Code != 200`

Continuous Bytes - Single Filter, 1

- **Route:** `/continuous/bytes`

- **Description:** Requests joined continuous matrix as bytes, using only 1 filtering parameter associated with test continuous (in addition to format).

- **Rationale:** Asserts filtering parameters can be used independently of one another.

- **Requests:**

```
GET /continuous/bytes?format=loom&version=1.0
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`
- **Failure Criteria:** `Status Code != 200`

Continuous Bytes - Single Filter, 2

- **Route:** `/continuous/bytes`

- **Description:** Requests joined continuous matrix as bytes, using only 1 filtering parameter (different than above) associated with test continuous (in addition to format).

- **Rationale:** Asserts filtering parameters can be used independently of one another.

- **Requests:**

```
GET /continuous/bytes?format=loom&studyID=f3ba0b59bed0fa2f1030e7cb508324d1
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

- **Success Criteria:** `Status Code == 200`
- **Failure Criteria:** `Status Code != 200`

Continuous Bytes - Format Not Specified

- **Route:** `/continuous/bytes`
- **Description:** Requests joined continuous matrix as bytes WITHOUT specifying the required `format` parameter. Expects a `4xx` response with error message.
- **Rationale:** As the `format` parameter is required to specify file format for the `/continuous/bytes` endpoint, this test asserts malformed requests raise an error.
- **Request:**

```
GET /continuous/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "Input payload validation failed"
}
```

- **Success Criteria:** `Status Code == 4xx` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 4xx` AND response body is NOT valid `Error` json

Continuous Bytes - Start Specified Without Chr

- **Route:** `/continuous/bytes`
- **Description:** Requests joined continuous matrix as bytes, specifying a start position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message
- **Rationale:** Asserts that the `/continuous/bytes` endpoint raises an error when `start` is specified without `chr`
- **Request:**

```
GET /continuous/bytes?format=loom&start=5
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Bytes - End Specified Without Chr

- **Route:** `/continuous/bytes`
- **Description:** Requests joined continuous matrix as bytes, specifying an end position without a chromosome. Expects a `400 Bad Request` status code in the response, and an error message
- **Rationale:** Asserts that the `/continuous/bytes` endpoint raises an error when `end` is specified without `chr`
- **Request:**

```
GET /continuous/bytes?format=loom&end=1000
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "chr required if either start or end is specified"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

Continuous Bytes - Start Greater Than End

- **Route:** `/continuous/bytes`
- **Description:** Requests joined continuous matrix as bytes, specifying `chr`, `start`, and `end`, however, `start` position is greater than `end`. Expects a `400 Bad Request` status code in the response, and an error message
- **Rationale:** Asserts that the `/continuous/bytes` endpoint raises an error when `start` is greater than `end`
- **Request:**

```
GET /continuous/bytes?format=loom&chr=1&start=200&end=100
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 400 Bad Request
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json

{
  "message": "start cannot be greater than end"
}
```

- **Success Criteria:** `Status Code == 400` AND response body is valid `Error` json
- **Failure Criteria:** `Status Code != 400` OR response body is NOT valid `Error` json

3.6.3 Continuous API Non-Implemented Test Cases

- *Continuous Formats Not Implemented*
- *Continuous Ticket by Id Not Implemented*
- *Continuous Bytes by Id Not Implemented*
- *Continuous Filters Not Implemented*
- *Continuous Ticket Not Implemented*
- *Continuous Bytes Not Implemented*

Continuous Formats Not Implemented

- **Route:** `/continuous/formats`
- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/formats`, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented
- **Request:**

```
GET /continuous/formats
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

  - **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`
- **Failure Criteria:** `Status Code != 501`

Continuous Ticket By Id Not Implemented

- **Route:** `/continuous/<id>/ticket`
- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/<id>/ticket`, expecting a `501 Not Implemented` status code response
- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented
- **Request:**

```
GET /continuous/nonexistentid999999999999999999/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

Continuous Bytes By Id Not Implemented

- **Route:** `/continuous/<id>/bytes`

- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/<id>/bytes`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented

- **Request:**

```
GET /continuous/nonexistentid999999999999999999/bytes
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

Continuous Filters Not Implemented

- **Route:** `/continuous/filters`

- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/filters`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented

- **Request:**

```
GET /continuous/filters
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

- **Success Criteria:** `Status Code == 501`

- **Failure Criteria:** `Status Code != 501`

Continuous Ticket Not Implemented

- **Route:** `/continuous/ticket`

- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/ticket`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented

- **Request:**

```
GET /continuous/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

  - **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

  - **Success Criteria:** `Status Code == 501`
  - **Failure Criteria:** `Status Code != 501`

Continuous Bytes Not Implemented

- **Route:** `/continuous/bytes`

- **Description:** If `Continuous` endpoints are `Not Implemented`, then request `/continuous/bytes`, expecting a `501 Not Implemented` status code response

- **Rationale:** Asserts that `Continuous` related endpoints are correctly non-implemented

- **Request:**

```
GET /continuous/bytes?format=loom
Accept: application/octet-stream, application/vnd.loom, text/tab-separated-values
```

  - **Successful Response:**

```
HTTP/1.1 501 Not Implemented
Content-Type: application/vnd.ga4gh.rnaget.v1.0.0+json
```

  - **Success Criteria:** `Status Code == 501`
  - **Failure Criteria:** `Status Code != 501`

3.6.4 Continuous Content Tests

Continuous content tests assert that continuous matrices downloaded from the RNAget server contain the expected content based on the request. Continuous file tracks, positions, and intensity values are cross-referenced against the request to ensure the expected data has been returned.

3.6.5 Continuous Content Test Cases

- *Continuous Ticket by Id Content Test Cases*
- *Continuous Bytes by Id Content Test Cases*

Continuous Ticket by Id Content Test Cases

- **Route:** `/continuous/<id>/ticket`

- **Description:** Download test continuous matrix by ticket multiple times (sometimes slicing by chr, start, end).

- **Rationale:** Asserts correct matrix file is associated with the test continuous `id`. Validates signal intensity values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/ticket
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Test continuous matrix columns, rows, values match expected

- **Failure Criteria:** Test continuous matrix columns, rows, values DO NOT match expected

Continuous Bytes by Id Content Test Cases

- **Route:** `/continuous/<id>/bytes`

- **Description:** Download test continuous matrix bytes multiple times (sometimes slicing by chr, start, end).

- **Rationale:** Asserts correct matrix file is associated with the test continuous `id`. Validates signal intensity values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /continuous/5e22e009f41fc53cbea094a41de8798f/bytes
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Test continuous matrix columns, rows, values match expected

- **Failure Criteria:** Test continuous matrix columns, rows, values DO NOT match expected

Continuous Ticket Content Test Cases

- **Route:** `/continuous/ticket`

- **Description:** Download joined continuous matrix by ticket multiple times (sometimes slicing by chr, start, end).

- **Rationale:** Asserts joined matrix. Validates signal intensity values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /continuous/ticket?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Joined continuous matrix columns, rows, values match expected

- **Failure Criteria:** Joined continuous matrix columns, rows, values DO NOT match expected

Continuous Bytes Content Test Cases

- **Route:** `/continuous/bytes`

- **Description:** Download joined continuous matrix as bytes multiple times (sometimes slicing by chr, start, end).

- **Rationale:** Asserts joined matrix. Validates signal intensity values match expected. Validates returned columns/rows match expected based on slice parameters.

- **Request:**

```
GET /continuous/bytes?format=loom
Accept: application/vnd.ga4gh.rnaget.v1.0.0+json, application/json
```

- **Success Criteria:** Joined continuous matrix columns, rows, values match expected

- **Failure Criteria:** Joined continuous matrix columns, rows, values DO NOT match expected

3.7 Getting Started

The RNAget compliance tool provides insight into how compliant a server is with the specification. The tool can be installed and executed from the command line, and can be configured to run compliance tests against any RNAget server of interest.

When run, the compliance tool generates a report, indicating which test cases succeeded and failed, as well as messages explaining why a test may have failed.

The next article provides instructions on how to install the RNAget compliance tool.

3.8 Installation

This section provides instructions on how to install the RNAget compliance suite application.

As a prerequisite, python 3 and pip must be installed on your system. The application can be installed by running the following from the command line.

1. Clone the latest build from https://github.com/ga4gh-rnaseq/rnaget-compliance-suite

```
git clone https://github.com/ga4gh-rnaseq/rnaget-compliance-suite
```

2. Enter rnaget-compliance-suite directory and install

```
cd rnaget-compliance-suite
python setup.py install
```

3. Confirm installation by executing the rnaget-compliance command

```
rnaget-compliance report
```

The next article explains how to run the compliance application.

3.9 Usage

The application can be run from the command line with the following command:

```
rnaget-compliance report
```

Command line parameters are as follows:

Table 1: rnaget-compliance report command line parameters

Parameter	Short Name	Required?	Description
–user-config	-c	Yes	Path to yaml file, indicating which server(s), project(s), expression(s), etc. to query
–output_dir	-o	No	Output directory to write report results and web files. Directory must not already exist. Defaults to ./rnaget-compliance-results if not specified
–serve	N/A	No	Flag. If set, the program will spin up a local server serving the HTML report after all compliance tests are completed
–uptime	-u	No	Integer. The local report server will shut down automatically after this time in seconds
–no-tar	N/A	No	Flag. If set, the program will not write a .tar.gz archive of report results
–force	-f	No	Flag. If set, the program will overwrite the output directory if it exists

3.9.1 User Config YAML File

The compliance application requires configurations to be written to a YAML file, which is specified on the command line. The following is a template of the config file, which can be modified to query different servers, projects, studies, etc.

```
servers:
  - server_name: Caltech
    base_url: https://felcat.caltech.edu/rnaget/
    implemented:
      projects: true
      studies: true
      expressions: true
      continuous: true
```

The .yml config file contains the following important features, which must be taken into consideration when modifying it for a different server:

1. *servers* must be the root property. The value of *servers* is a list of server definitions (ie the application can query multiple servers in a single run).

2. Each server definition requires the following parameters:

    - server_name: a string uniquely identifying the server

    - base_url: the base url at which the RNAget API can be reached

3. If a server requires client authentication, then a server definition can include the OAuth 2.0 bearer access token under the "token" property:

```
servers:
  - server_name: Caltech
    base_url: https://felcat.caltech.edu/rnaget/
    token: abcdefghijklmnop
```

```
    implemented:
      projects: true
      studies: true
      expressions: true
      continuous: true
```

The token will be used for all API tests executed by the compliance suite.

4. A server's *implemented* property indicates which routes (projects, studies, expressions, continuous) have been implemented. This property can be removed if all routes are implemented, as routes are expected to be implemented by default. Use `${ROUTENAME}:   false` to indicate non-implemented routes.

3.9.2 Basic Usage

The program requires, at minimum, a YAML config file to be specified in order to execute:

```
rnaget-compliance report -c user_config_template.yaml
```

The user can also specify the location at which to create the output report directory. The program will only run if the output directory does not already exist (will not overwrite existing files):

```
rnaget-compliance report -c user_config_template.yaml -o ./results/output
```

The program creates a series of web files at the output directory, which can be used to view the compliance results as an HTML report. A web server can be started manually at the output directory to serve the report. By specifying the `--serve` option, the report server will be started automatically upon completion of the compliance tests:

```
rnaget-compliance report -c user_config_template.yaml -o ./results/output --serve
```

The next article explains how to view the compliance report, including how to diagnose errors (compliance failures)

3.10 Report

Once the compliance report has been generated, the HTML report can be accessed by first starting a web server at the output directory, then navigating to the page via web browser.

3.10.1 Viewing the Report

Report: Index

The report index page displays a tabular overview of test successes/failures by tested servers and API routes. An example report index is displayed below:

Clicking on any of the *Pass*/*Fail* buttons will take you to an in-depth report of the status of all tests for that server.

Report: Test Status

This page shows a more detailed report of all tests run on a particular server, organized by API route and test case. For any given test, clicking the *Info* button will display information for that test, including the request url, parameters, and the response body. The *Info* window is also helpful in diagnosing failed compliance tests, as it reports the type of error encountered and related debug messages.

Below is an example of *Projects* test results for one server in the test status page:

Clicking the *Info* button shows test information, displayed below:

Test results can also be viewed as a matrix:

 Back

RNAGet Implementation Compliance Report

Server: Caltech

Base URL: https://felcat.caltech.edu/rnaget/

Total tests: 36

Total tests passed: 36

Total tests failed: 0

Total tests skipped: 0

Total warnings generated: 0

Text   Matrix

Server name	Object and Id	Test case	Status
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Get	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Get Not Found	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Search	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Search Url Params All	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Search Url Params Cases	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Search Filters Out	PASSED
Caltech	Projects: 43378a5d48364f9d8cf3c3d5104df560	Project Search Filters	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Get	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Get Not Found	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Search	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Search Url Params All	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Search Url Params Cases	PASSED
Caltech	Studies: 6cccbbd76b9c4837bd7342dd616d0fec	Study Search Filters Out	PASSED