GitHub Gist

meatcar / avro+ga4gh.md
Last active 4 months ago

These notes should help you undestand Avro, and give you an entrypoint to start understanding the ga4gh API.

`avro+ga4gh.md`

What is Avro?

Parsing the ga4gh API requires you first to understand what is Avro.

Avro is a project that allows you to

- formaly declare and define data structures (schemas in Avro) to be later used in different languages.
- store data in a file
- fetch data from a file

Avro uses a couple of different file extensions that you'll encounter:

- `.avsc` - A JSON representation of an Avro schema (for a single object). This file is parsed by Avro libraries.
- `.avpr` - A JSON representation of an Avro protocol (a collection of schemas)
- `.avdl` - A code-like language that gets translated to `.avsc` or `.avpr` using the `avro-tools.jar`. This file is not used by Avro libraries, as far as I can tell...

Avro allows you to generate Java object from the schemas, and use them! See Java docs

Python is a bit less friendly, you need to pass in dictionaries which are then validated by Avro. It would be nicer if Avro generated native python objects for you, but it doesn't. See Python docs

Using Avro in python

1. Translate Avro `.avdl` into `.avpr` OR translate `.avdl` to `.avsc` as shown by ga4gh
2. How to use Avro in Python. There is also a guide from the Avro docs
   - `.avpr` is a protocol file. Use `avro.protocol.parse`, then extract the individual schemas.
   - `.avps` is a schema file. Use `avro.schema.parse`
   - I'm using python3, and some function + variable names in the `avro-python3` library differ from tutorial.
   - When opening the file to pass to the `DataFileWriter`, use `"wb"` instead of just `"w"`.

Useful Resources

Reading straight Avro files is a bit difficult. There's a nice introruction to the ga4gh API on their website that gives you an overview of their datastructures, as well as a link to HTML-formatted schemas at the bottom of the page.

The ga4gh reference server implementation has some useful examples of how to deal with Avro. Specifically:

- processing from Avro `.avdl` to `.avsc`
- using the `.avsc` files in python