Setting Up the JGI Integration Workflow

Introduction

This document provides step-by-step instructions for setting up and running the JGI Integration Workflow. The workflow is distributed in two parts:

1. A Docker image (pulled from an online repository to your system)
2. Project data (distributed as an archived folder via secure link from JGI)

Note: The tutorial examples provided are for setup via a Unix-style command line, but the workflow should be portable to any system by using analogous commands or a graphical interface to perform each setup step. Also, the examples use specific directory paths and locations for illustration purposes only and should be substituted for paths on your own system.

Prerequisites

Docker Desktop installed on your system. If you do not have the application, download it from the official Docker website. Make sure to choose the correct version for your system. This program will be used to set up the software environment for running the integration workflow.

Step 1: Download and Unzip the Project Data Folder

A. Create a dedicated directory on your system that will run and store results from the workflow, then navigate into it. For example (use your own path here):

```
> INTEGRATION_DIR=/home/user/JGI_Integration

> mkdir $INTEGRATION_DIR && cd $INTEGRATION_DIR
```

B. Download the `project_data` folder that was provided via a secure link. It will download as a zipped archive file.

Note: Your system may not download the archive with the exact file name of `project_data.zip`, but the name of this archive and its directory after unzipping is not critical for the workflow functionality.

C. Unzip the project data archive into the new directory you created in (A). For example:

```
> unzip /home/user/Downloads/project_data.zip -d $INTEGRATION_DIR
```

Note: Make sure your main directory has the following structure (do not alter the file or folder locations):

```
── $INTEGRATION_DIR
   └── project_data
       ├── input_data
       │   ├── config
       │   ├── docker-compose.yml
       │   ├── link_script
       │   └── raw_data
       └── output_data
```

Step 2: Launch the Docker Container

A. Make sure that the Docker Desktop app is open and running on your system. *Hint:* this command should print app details without any error messages (warnings are fine):

```
> docker info
```

B. Navigate into the project data directory (the archive you unzipped in Step 1) and into the subdirectory that holds your pre-processed input data:

```
> cd $INTEGRATION_DIR/project_data/input_data
```

C. Run the docker compose file to pull and run the container. In the command below, substitute `tag=arch` for the following architecture depending on your operating system: Windows, `tag=windows-amd64`; MacOS Apple, `tag=mac-arm64`; MacOS Intel, `tag=mac-amd64`; Linux, `tag=linux-amd64`.

```
> tag=arch docker compose -p jgi-integration up
```

D. This will print details about the docker container boot to standard output, and most can be ignored. About halfway down, after the line “`Jupyter Server 2.8.0 is running at:`”, there will be two web addresses. Copy the one beginning with `http://127.0.0.1:8888/lab...` into a web browser search bar (or cmd/ctrl-click).

Step 3: Run the workflow in JupyterLab

1. JupyterLab will render in your web browser. Details about how to navigate the JupyterLab interface can be found on the Jupyter documentation page.
2. Switch to the *run.pdf* instructions to run the workflow!