**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 three parts:

1. GitHub repository (pulled from the repository to your system)
2. Docker Desktop (installed from online to your system)
3. User data (distributed via an archive folder to you from JGI)

**Prerequisites**

* **A blue sign with white text

  AI-generated content may be incorrect.Docker Desktop installed on your system.** You can download it from the [official Docker website](https://www.docker.com/products/docker-desktop). Make sure to choose the correct version for your operating system. This software will be used to access the correct software environment for running the integration workflow.
* **A close up of a logo

  AI-generated content may be incorrect.The Git tool suite installed on your system.** You can find instructions for installation on the [official git website.](https://git-scm.com/downloads) This will be used to access the repository of code that runs the workflow.
* All steps in the workflow can run on any system, but examples below will be given only for Unix-style (e.g., MacOS, Linux) command line interface. To use a graphical interface or Microsoft Command Prompt/PowerShell, follow the instructions but not the examples.

**Step 1: Download and Unzip the Project Data Folder**

1. Create a dedicated **integration directory** on your system that will run and store results from the workflow, then navigate into it. For example:

> mkdir /home/user/JGI\_integration

> cd /home/user/JGI\_integration

1. Download the project\_data.zip archive file that was provided via a secure link.
2. Unzip the project data archive into the new directory to create the **data directory**:

> unzip /home/user/Downloads/project\_data.zip -d ./

**Step 2: Clone the JGI Integration Repository**

1. While still in the **integration directory**, clone the workflow code repository using the git command line tool to create the **code directory**:

> git clone <https://github.com/bkieft-usa/jgi_integration.git>

1. For more details on cloning a repository, visit the [GitHub Docs page](https://docs.github.com/en/repositories/creating-and-managing-repositories/cloning-a-repository).

**Step 3: Check Directory Structure**

1. Check directory structure to ensure the workspace is set up correctly. The **data directory** and **code directory** should be inside the **integration directory**:

> tree -L 3

├── jgi\_integration

│ ├── docker

│ │ ├── Dockerfile

│ │ └── requirements.txt

│ ├── docs

│ │ ├── analysis\_parameters\_explained.md

│ │ ├── normalization\_parameters\_explained.md

│ │ └── setup.docx

│ ├── integration\_workflow.ipynb

│ └── tools

│ ├── \_\_init\_\_.py

│ ├── helpers.py

│ └── objects.py

└── project\_data

├── input\_data

│ ├── config

│ ├── docker-compose.yml

│ ├── link\_script

│ └── raw\_data

└── results

**Step 3: Run the Docker Image**

1. Make sure that the Docker Desktop app is open and running. **Hint**: you can ensure Docker Desktop is running by checking its status; running this command should print a lot of info without any ERROR messages (Warnings are fine):

> docker info

1. Pull the Docker image from the repository to your system. The image that you want will depend on your system (Windows <windows-amd64>, MacOS with an Apple <mac-arm64> or Intel chip <mac-amd64>, or Linux <linux-amd64>). If you need a different architecture, please contact JGI:

> docker pull bkieft/jgi-integration:<tag>

**Step 4: Launch the Docker Container**

1. Navigate into the new **project directory** (the folder you unzipped in Step 1) and into the subdirectory input\_data:

> cd /home/user/JGI\_integration/project\_data/input\_data

1. Inside this directory you should see a few folders and the docker-compose.yml file.
2. Run the docker composition file to build a local container:

> docker compose up

1. This will print lots of information lines to the standard output, and most can be ignored. After the line “Jupyter Server 2.8.0 is running at:”, there will be two URLs – ctrl/cmd-click or copy+paste into your web browser (e.g., Firefox, Chrome, Safari) the one beginning with the location <http://127.0.0.1:8888/lab...>

**Step 6: Run the workflow in JupyterLab**

1. You will see the JupyterLab interface on your web browser. Double-click the workflow notebook integration\_workflow.ipynb in the left menu navigator to bring it into the workspace.
2. You may also want to bring the configuration file into your workspace. Editing this configuration file will allow you to produce different outputs (/

**Directory Structure**

The final directory structure should look like this:

project/

├── input\_data/

│ ├── config/

│ ├── docker-compose.yml

│ ├── link\_script/

│ └── raw\_data/

└── results/

└── jgi\_integration/

├── docker/

│ ├── Dockerfile

│ └── requirements.txt

├── docs/

│ └── configuration\_doc.md

├── integration\_workflow.ipynb

└── tools/

├── init.py

├── helpers.py

└── objects.py