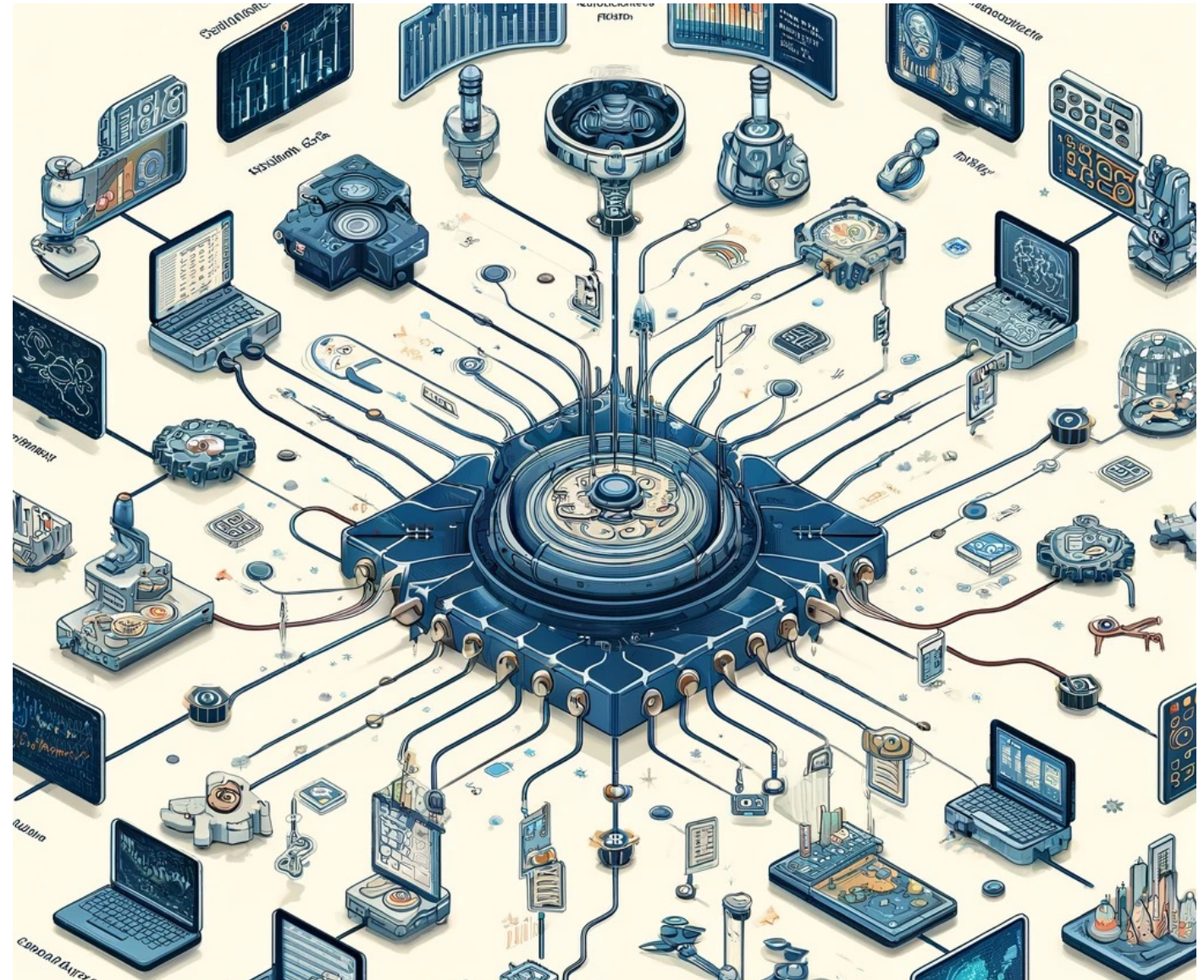


Exploring Scientific Workflows with CWL and dispel4py

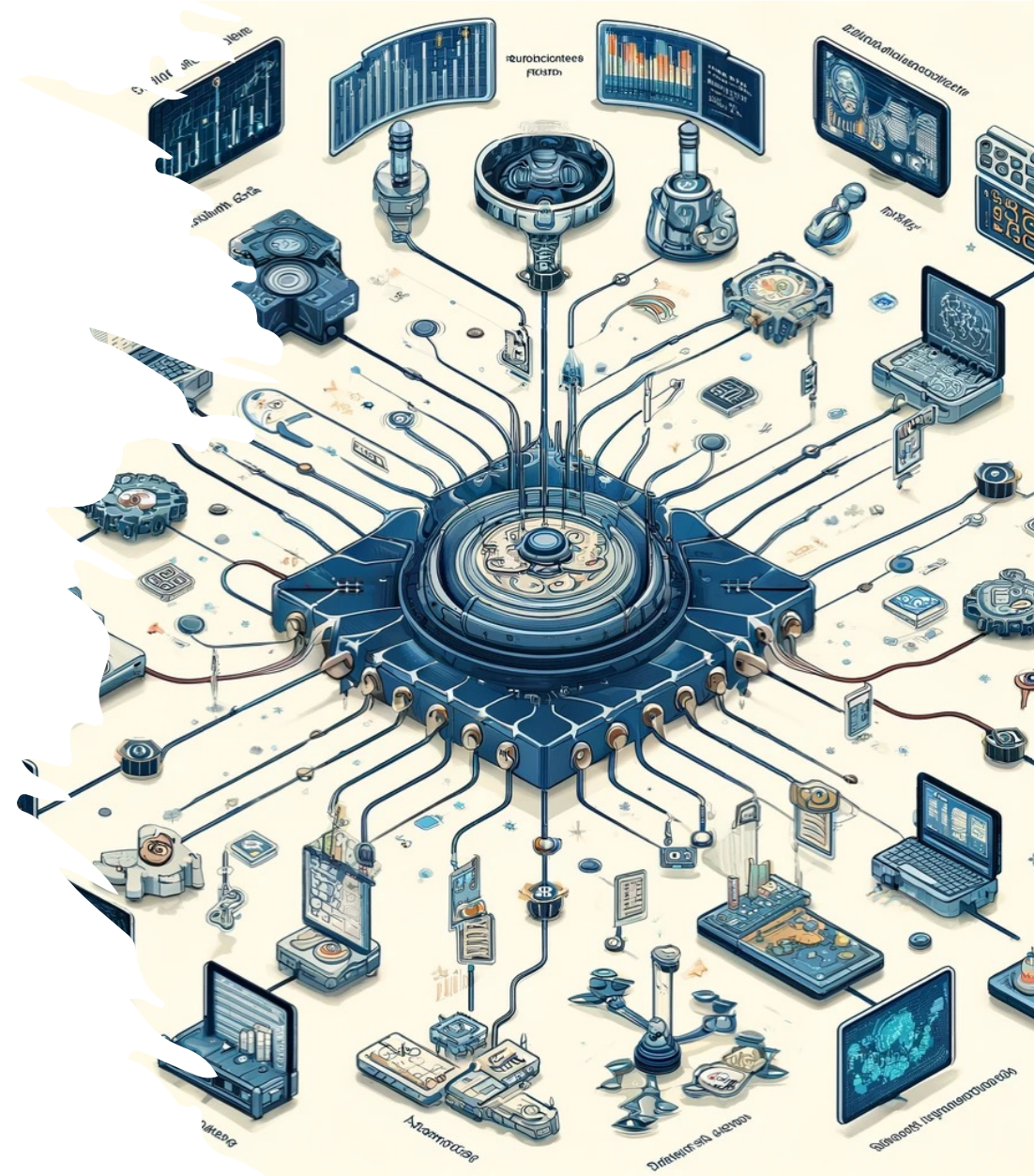
Module 3

- Dr. Rosa Filgueira
- Lecturer at the School of Computer Science
- University of St Andrews
- rf208@st-andrews.ac.uk
- rosa.filgueira.vicente@gmail.com



Seminar Overview

- Day 1: Understanding Scientific Workflows (4 Hours)
 - Module 1: Introduction to Scientific Workflows
 - Module 2: Creating Workflows with CWL
- Day 2: Exploring dispel4py and its applications (4 Hours)
 - Module 3: Introduction to dispel4py
 - dispel4py – Basic Concepts
 - Dispel4py – Advanced Concepts
 - Hands-On exercises (Part I)
 - Module 4: Hands-on Exercises & latest research works
 - Hands-On Exercises (Part II)
 - dispel4py latest research updates



Module 3: Introduction to dispel4py

We have several options to install dispel4py! It uses **Python 3.10** .

1) Follow the instructions specified in our [StreamingFlow/d4py](#) repository

Via pip

1. `conda create --name stream-d4py_env python=3.10`
2. `conda activate stream-d4py_env`
3. `conda install -c conda-forge mpi4py mpich` OR `pip install mpi4py` (Linux)
4. `pip install stream-d4py`

Via cloning this repo

1. `conda create --name stream-d4py_env python=3.10`
2. `conda activate stream-d4py_env`
3. `https://github.com/StreamingFlow/stream-d4py.git`
4. `cd dispel4py`
5. `conda install -c conda-forge mpi4py mpich` OR `pip install mpi4py` (Linux)
6. `python setup.py install`

- Read the notes about “[Known Issues](#)”

Module 3: Introduction to dispel4py

2) Installation of dispel4py with Docker:

- Clone our [StreamingFlow/d4py](https://github.com/StreamingFlow/d4py) repo

```
$ git clone https://github.com/StreamingFlow/d4py.git
```

The Dockerfile in the dispel4py root directory installs dispel4py and mpi4py.

```
docker build . -t mydispel4py
```



Note: If you want to re-build an image without cache, use this flag: `--no-cache`

Start a Docker container with the dispel4py image in interactive mode with a bash shell:

```
docker run -it mydispel4py /bin/bash
```



Module 3: Introduction to dispel4py

3) Use directly our [Google Notebook Colab](#), which installs dispel4py automatically

Testing_dispel4py_2_0.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text Copy to Drive

Table of contents

dispel4py

- Python Version
- Getting Started with dispel4py
- Part I - Simple Examples
 - Word_Count dispel4py Workflow
- Part II - Real Use Cases
 - Covid Workflow
 - Sensor Workflow
 - Postprocess
 - Sensor_output.txt

+ Section

dispel4py

dispel4py is a free and open-source Python library for describing abstract stream-based workflows for distributed data-intensive applications. It enables users to focus on their scientific methods, avoiding distracting details and retaining flexibility over the computing infrastructure they use. It delivers mappings to diverse computing infrastructures, including cloud technologies, HPC architectures and specialised data-intensive machines, to move seamlessly into production with large-scale data loads. The dispel4py system maps workflows dynamically onto multiple enactment systems, and supports parallel processing on distributed memory systems with MPI and shared memory systems with multiprocessing, without users having to modify their workflows.

Python Version

This version of dispel4py has been tested with Python 3.10

For earlier versions of dispel4py compatible with Python <3.10 (2.7.5, 2.7.2, 2.6.6 and Python 3.4.3, 3.6, 3.7) we recommend to go [here](#).

Getting Started with dispel4py

Our journey begins with the installation of dispel4py, ensuring you have the necessary tools to dive into data processing. We present two distinct approaches to install dispel4py:

- Version 1: Direct installation using pip, suitable for quickly adding dispel4py and its stream-d4py companion to your Python environment.
- Version 2: For those preferring the latest versions or contributing to development, cloning from the GitHub repository and manual installation offers the cutting edge of dispel4py functionalities.

```
[ ]
#Version 1:
!pip install mpi4py
!pip install stream-d4py

# Version 2:
!git clone https://github.com/StreamingFlow/d4py.git
%cd d4py
!pip install mpi4py
!python setup.py install

Collecting mpi4py
  Downloading mpi4py-3.1.5.tar.gz (2.5 MB)
    2.5/2.5 MB 28.2 MB/s eta 0:00:00
Installing build dependencies ... done
Getting requirements to build wheel ... done
Preparing metadata (pyproject.toml) ... done
Building wheels for collected packages: mpi4py
  Building wheel for mpi4py (pyproject.toml) ... done
  Created wheel for mpi4py: filename=mpi4py-3.1.5-cp310-cp310-linux_x86_64.whl size=2746507 sha256=b4020eb5e2e9b75c7729b9ae881f0e33596fe7c90e6495b17bad8b015f107064
  Stored in directory: /root/.cache/pip/wheels/18/2b/7f/c852523089e9182b45fca50ff56f49a51eeb6284fd25a66713
Successfully built mpi4py
Installing collected packages: mpi4py
Successfully installed mpi4py-3.1.5
Collecting stream-d4py
  Downloading stream_d4py-2.9.1-pv3-none-any.whl (121 kB)
```