International Centre for Radio Astronomy Research

The DALiuGE Ecosystem

A workflow editor and execution framework for transforming existing code into scalable, distributed workflows

Ryan Bunney, Professor Andreas Wicenec, Rodrigo Tobar, Nicholas Pritchard, James Strauss, Moritz Wicenec

ICRAR is a joint venture between Curtin University and The University of Western Australia and receives support from the Western Australian and Australian Governments.

Outline

- Science Workflows
  - What
  - Why
  - How
- The software
  - EAGLE
  - Code 'palette generator'
  - DALiuGE
- Current Work
  - DINGO
  - WALLABY
  - CHILES
- Is DALiuGE right for you?
- Questions

Science Workflows

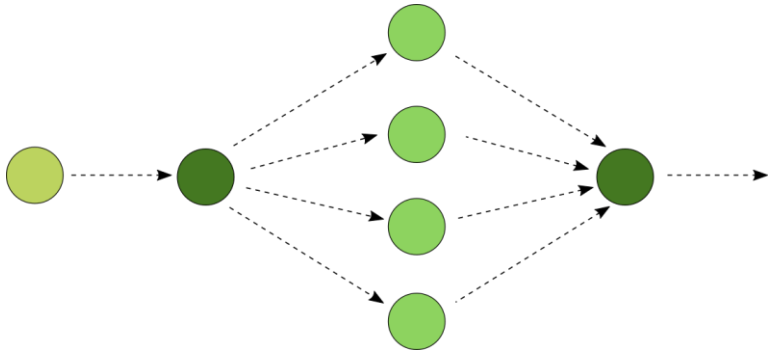
Science workflows

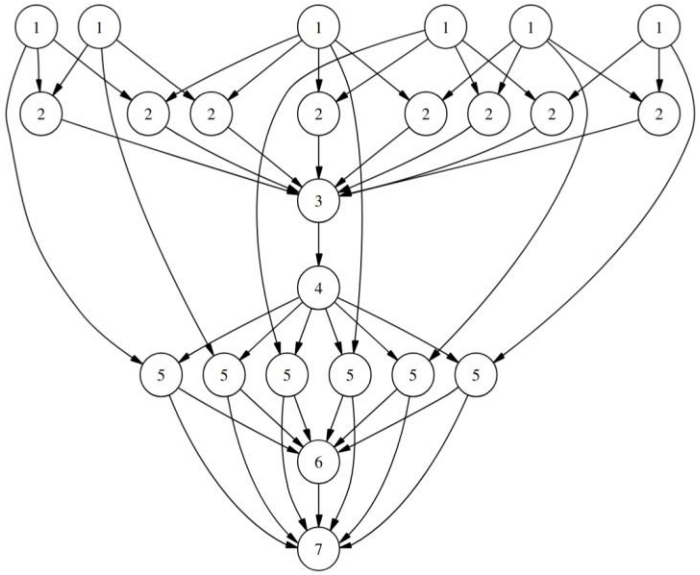
Workflow management systems

Existing systems

Science Workflows: What?

- Workflows are a common tool in HPC for grouping together interrelated computing jobs
- Jobs often have precedence constraints
- Typically use Directed Acyclic Graph (DAG) representation
  - Compute tasks are nodes
  - Data dependencies are edges





Montage workflow (Deelman *et al.*, 2005)

Science Workflows: Why?

Computational

- Scheduling & Resource Allocation are NP-Hard jobs
- DAG/stricter definitions can improve heuristic performance

Practical

- Separate researchers from optimisation and deployment
- Improve robustness
  - N people on N similar codes vs. N people on 1 shared workflow
- Improve reproducibility
  - Federated workflows allow for re-computation across multiple scientists/groups

Science Workflows Management Systems (SWFMS)

Apache Taverna

(Retired 2020)

nextflow

Apache Airflow

ASKALON

流 DALiuGE

snakemake

A framework for reproducible data analysis

Galaxy PROJECT

reana

Pegasus

DALiuGE

- Data-Activated Flow (Liu) Graph Engine
  - Deployment and execution of radio astronomy workflows
- Prototype workflow engine for the SKA Phase 1
- SC 2020 Gordon Bell Prize Finalist
- Demonstrated at scale:
  - run on 4,560 compute nodes
  - used 27,360 GPUs to generate 2.6 PB of data

What DALiuGE does differently

Feature	SFWMs	DALiuGE
Separation of concerns (Scientist vs. Developer)	• Requires learning a DSLs • Time spent transitioning existing work to new tool	• UI for drawing graphs • Re-use of existing Python/scripting code
Control flow	• Fixed shell-like management of workflow execution	• Support for conditional execution of the graph
Data management	• Data is implicit (edge in DAG)	• Data as first-class citizen • Controls its own runtime execution
Data types	• Reliance on files for intermediary data storage	• In-memory support between different types of applications (Python -> Bash -> Docker)

EAGLE

Logical Graph Editor

Repository and Palette Management

Team

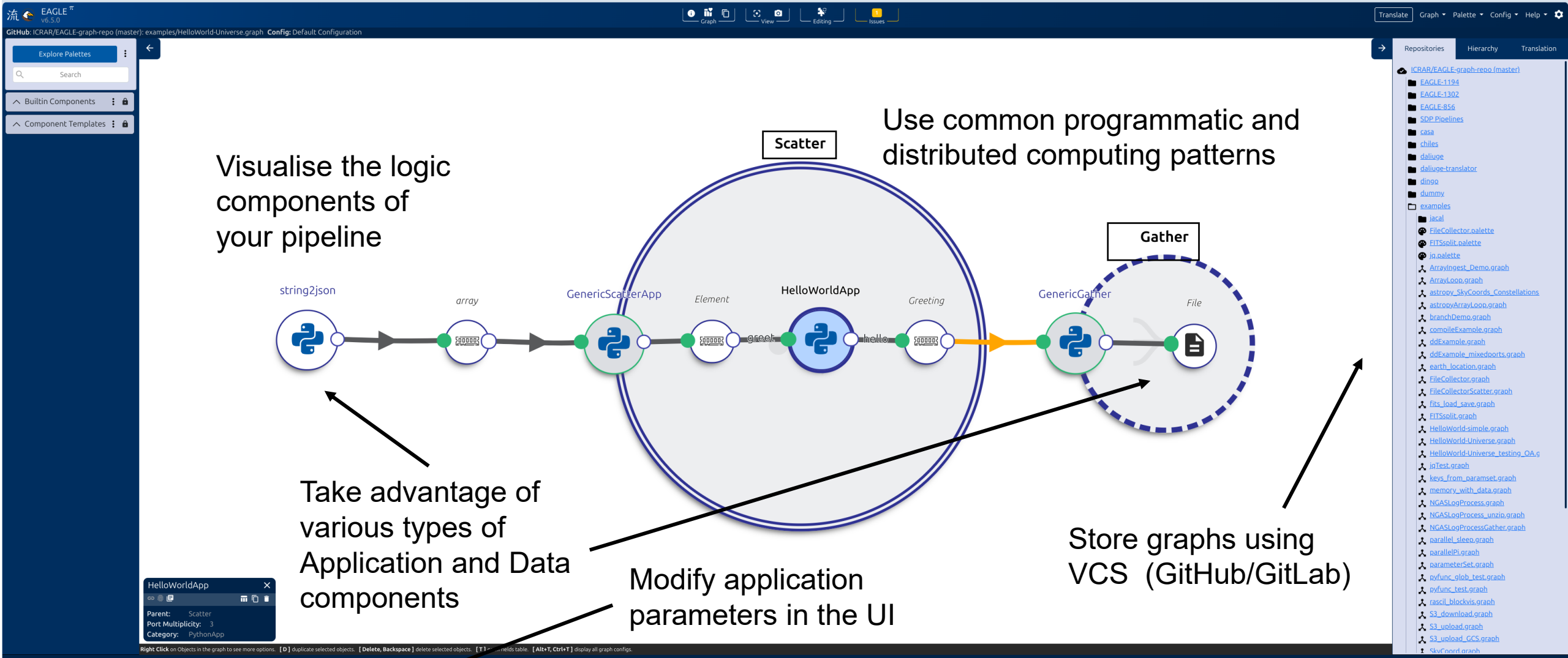
Andreas Wicenec
Project Lead

James Strauss
Software Engineer

Moritz Wicenec
Software Engineer

EAGLE$^{\pi}$

https://eagle.icrar.org



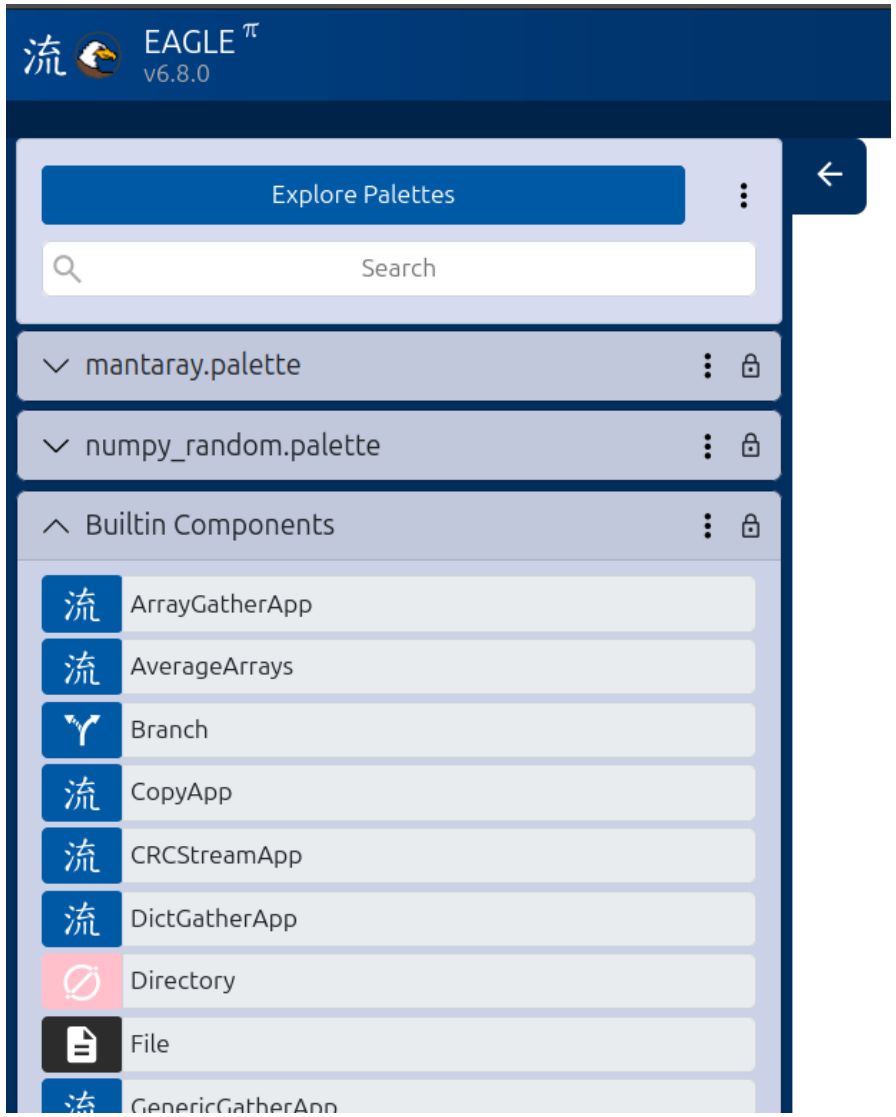
Palette Generator

Point to existing code

Use palette in EAGLE

Run palette with DALiUGE

dlg_paletteGen

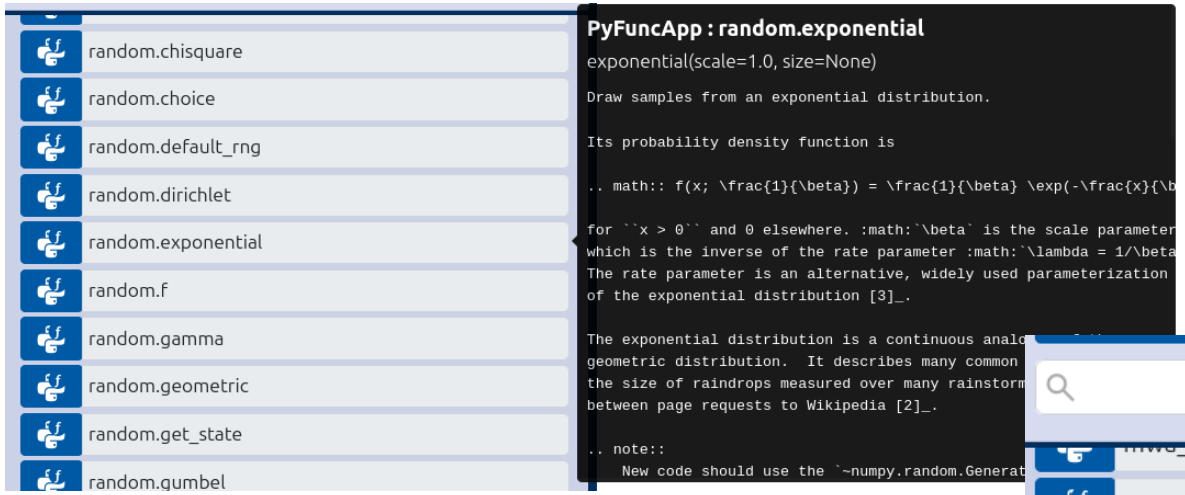


- EAGLE graphs draw components from their palettes
- There are built-in palettes that support generic applications
- dlg_paletteGen tool generate palettes from arbitrary Python modules in a fully automatic way
- The tool can extract component information from plain Python modules, but also C extensions and PyBind11 generated modules

dlg_paletteGen

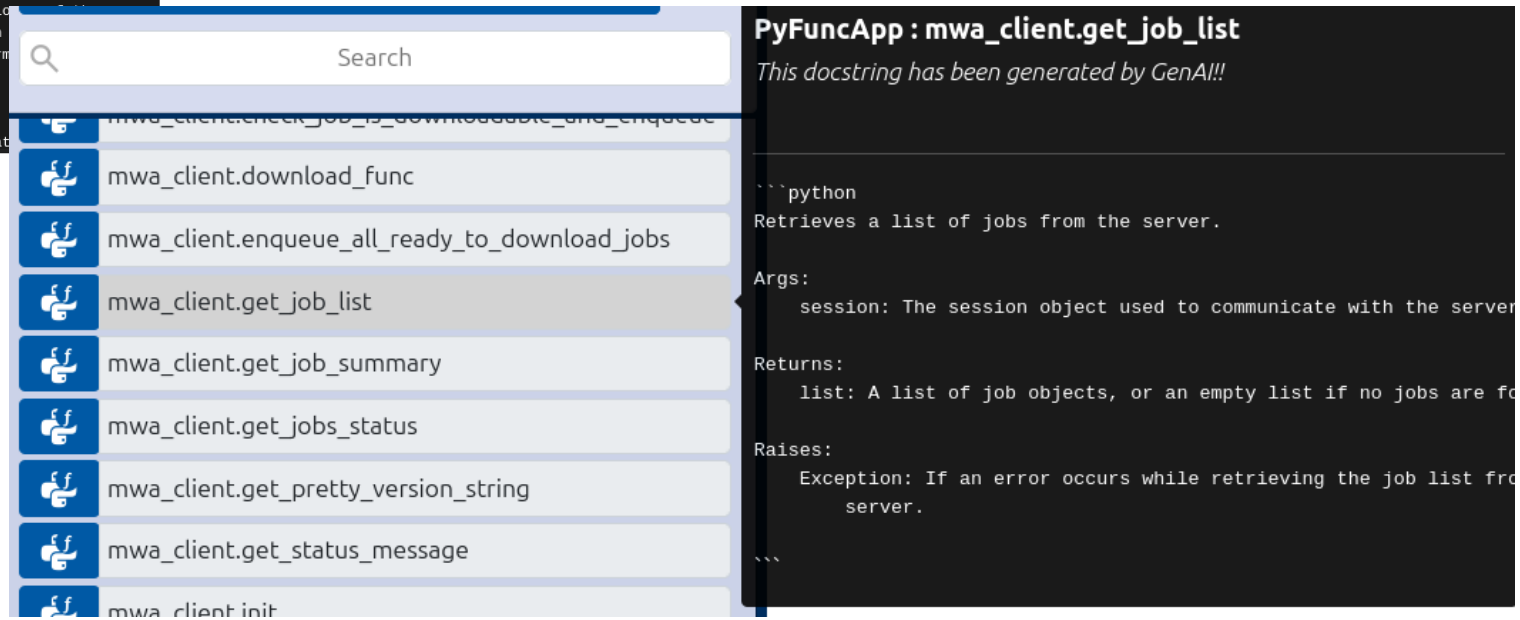
Selection | Content

Attribute Name	Value	Default Value	Description	Type
scale	1	1	The scale parameter, :math:`\beta = 1/\lambda`. Must be	float
size			Output shape. If the given shape is, e.g., ``(m, n, k)``, then	int or tuple
func_name	numpy.random.exponential	numpy.random.exponential	Complete import path of function or just a function name which is also used in func_code below.	String



- All in-line documentation (docstrings) and parameter descriptions are extracted

- (New feature!) No docstrings? Gemini will figure it out for you!



DALiuGE

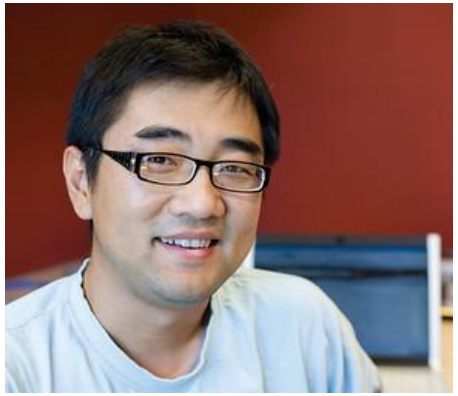
Scheduling and Partitioning

Runtime Management of Applications

Remote Deployment and Execution

Team (over the years)

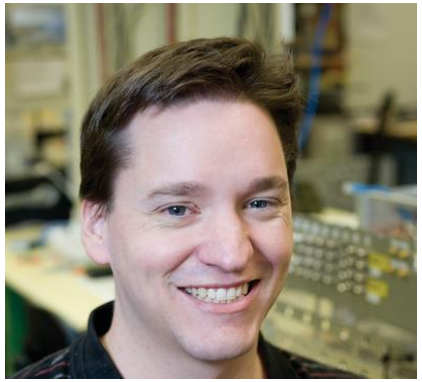
Andreas Wicenec



Chen Wu

Rodrigo Tobar

Kevin Vinsen



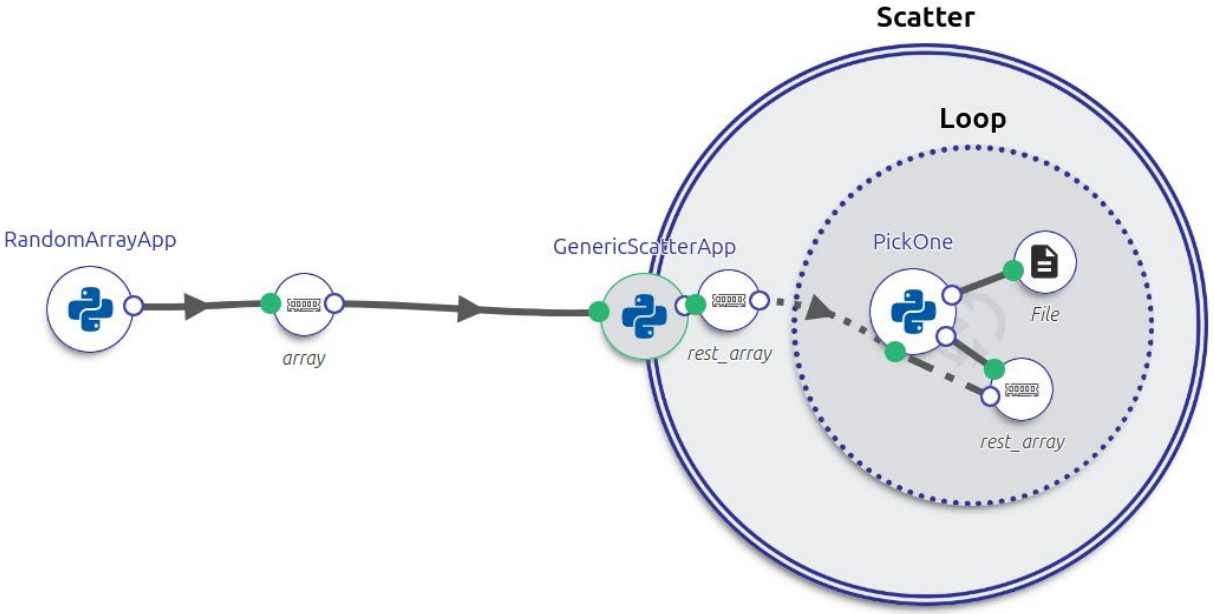
Dave Pallot

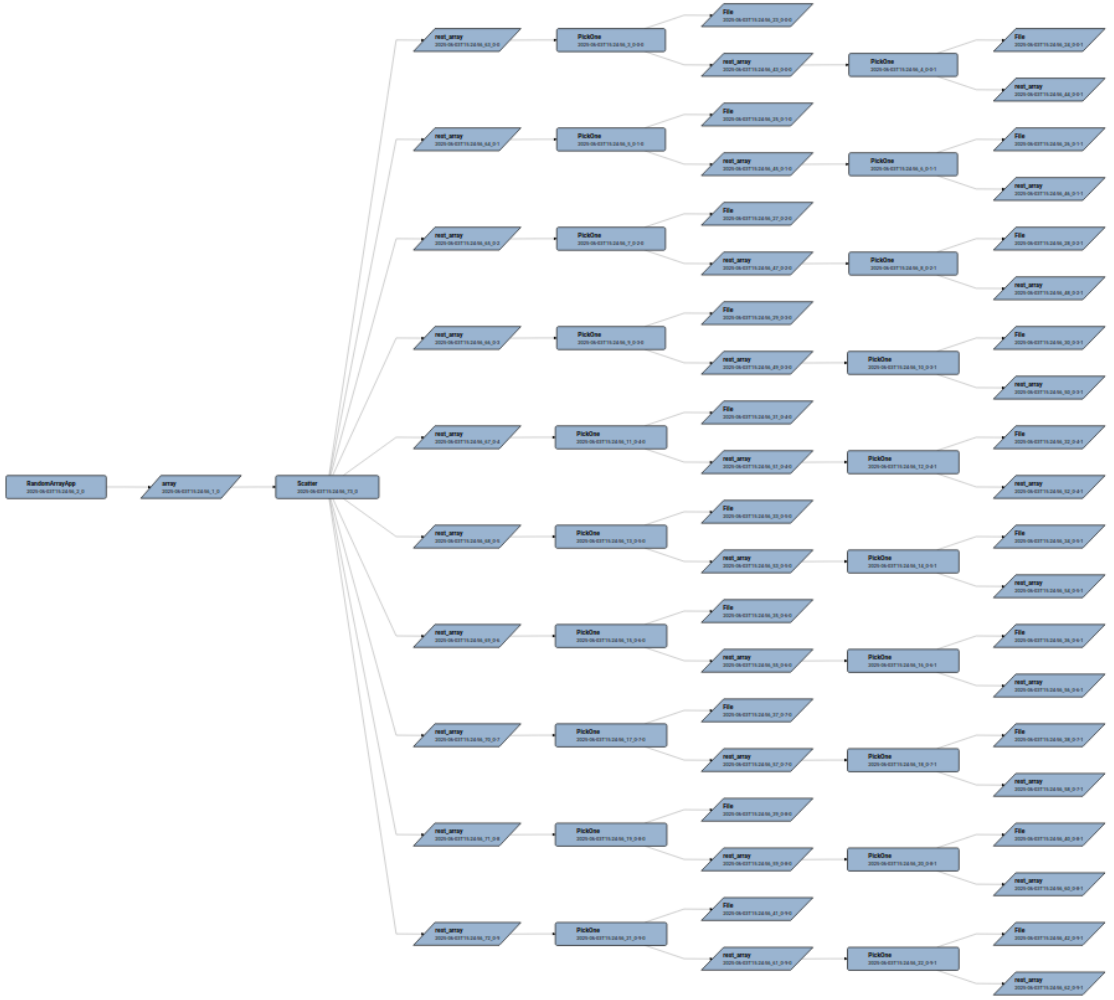
Nicholas Pritchard

Callan Gray

Julian Carravick

Translating, Partitioning, Scheduling

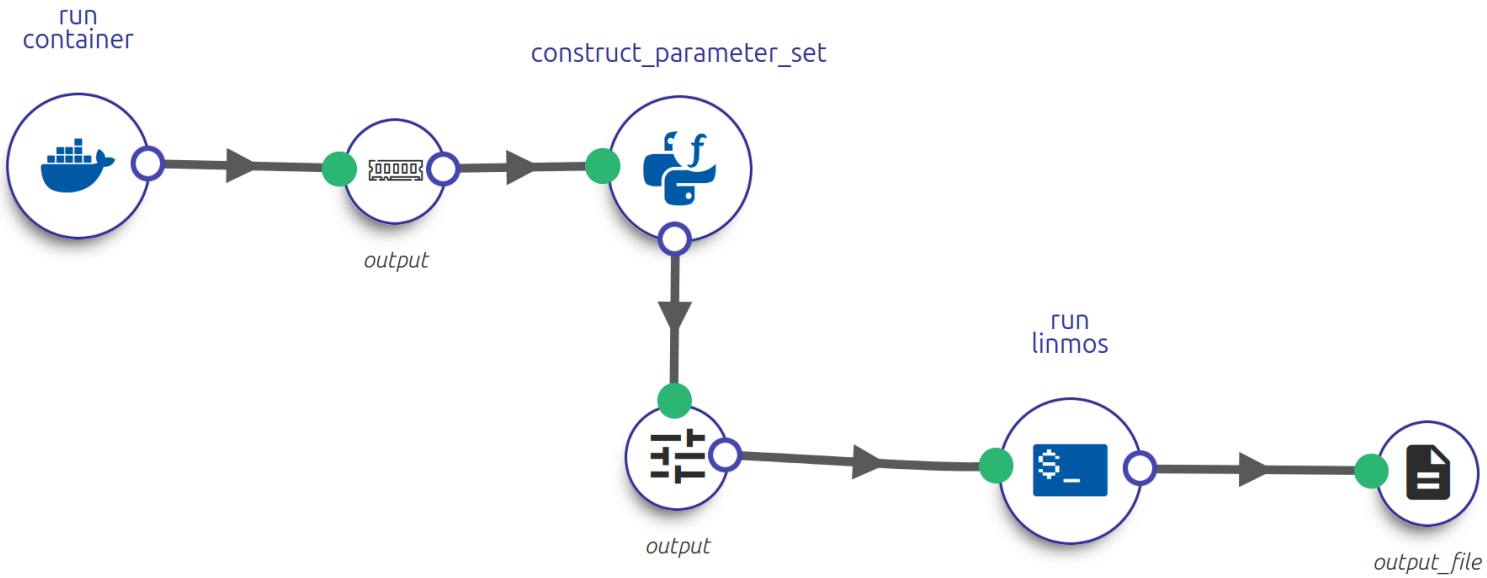




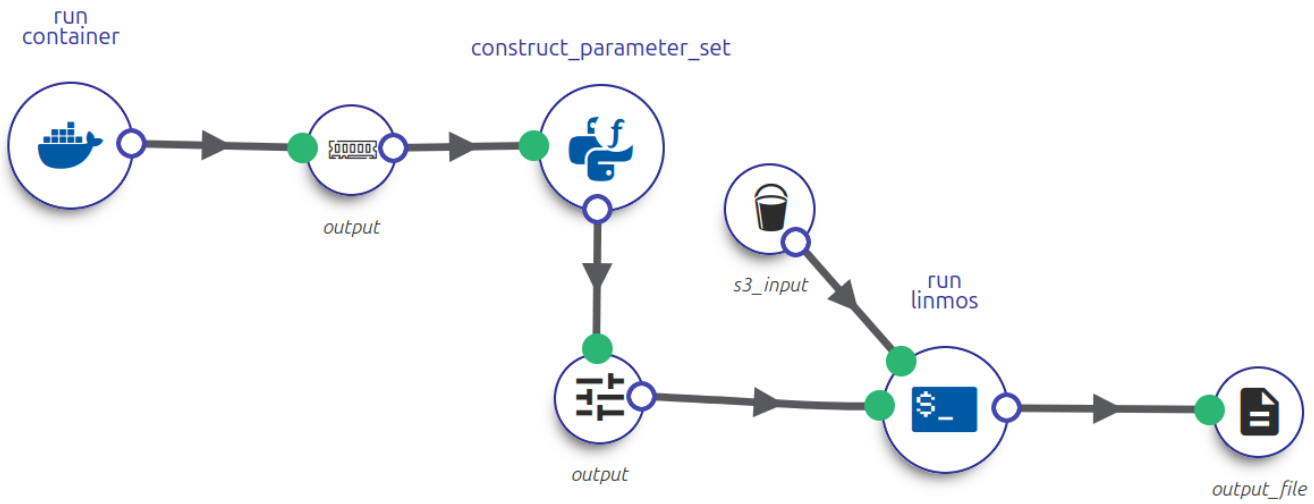
Applications

Built-in support for:

- BashApp
- DockerApp
- **PyFuncApp**
- DALiuGEApps, e.g.
  - CopyApp
  - ScpApp



Data



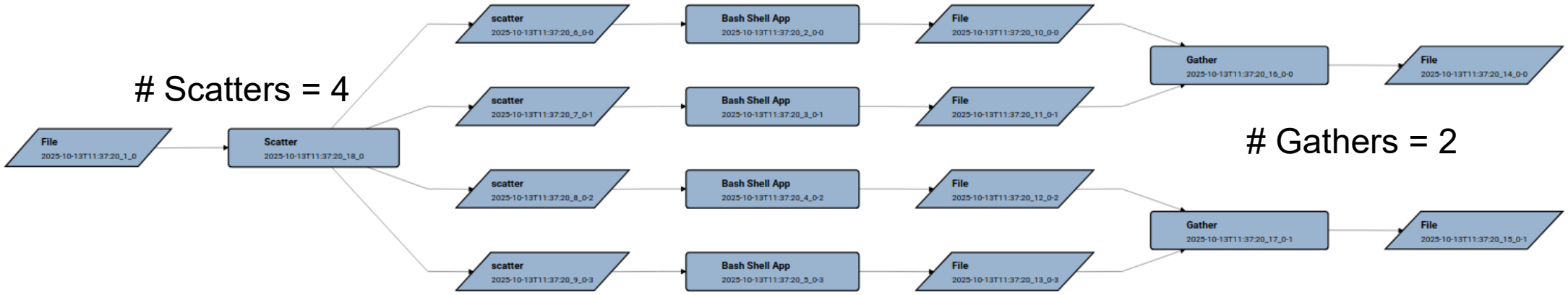
Supported types include

- Memory
- File
- ParameterSet
- NGAS
- S3

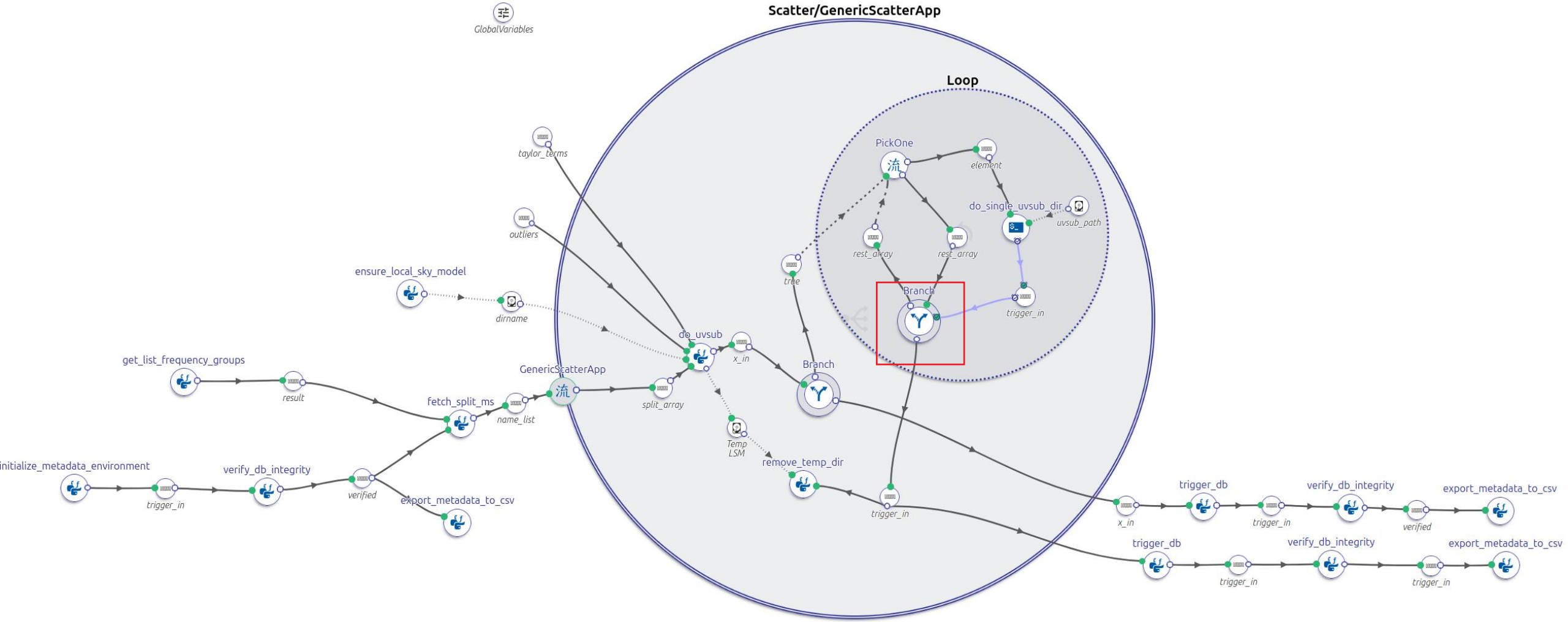
- Data manages itself at runtime and triggers consumers
- In-memory data support across apps

Constructs: Loop example

Constructs: Scatter & Gather



Constructs: Branches example

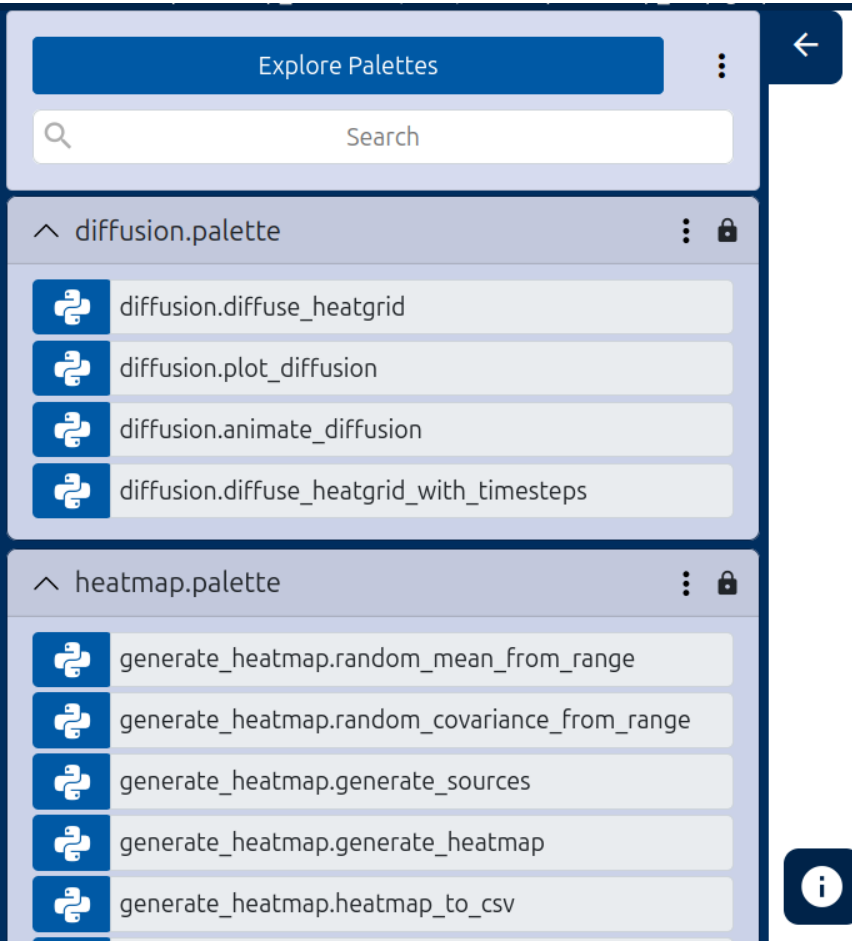


- Iterate through dynamic number of files/data
- Apply convergence criteria to workflow execution

Reproducibility

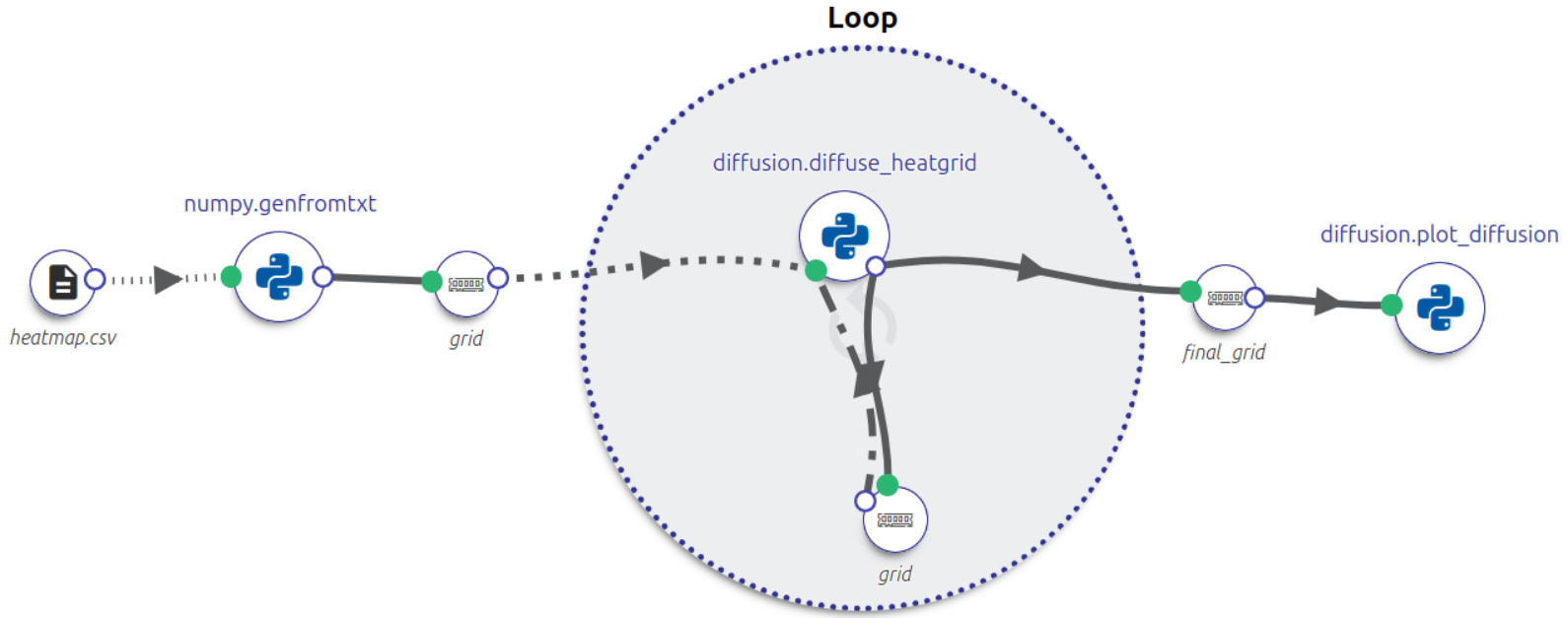
- DALiuGE can grade a workflow on 7 Reproducibility 'tests'
  - Rerun
  - Repeat
  - Recompute
  - Reproduce
  - Scientific replication
  - Computational replication
  - Total Replication
- Implemented using BlockDAG and MerkleTrees to implement run hashes
- Further reading: *Pritchard, N. J., & Wicenec, A. (2024). Formal Definition and Implementation of Reproducibility Tenets for Computational Workflows*

Putting it all together



Import palette into EAGLE

Run dlg_paletteGen on local code & create a palette



Build & run code from palette through DALiuGE

Recent & current work

DINGO
Wallaby Hires
CHILES
Yours?

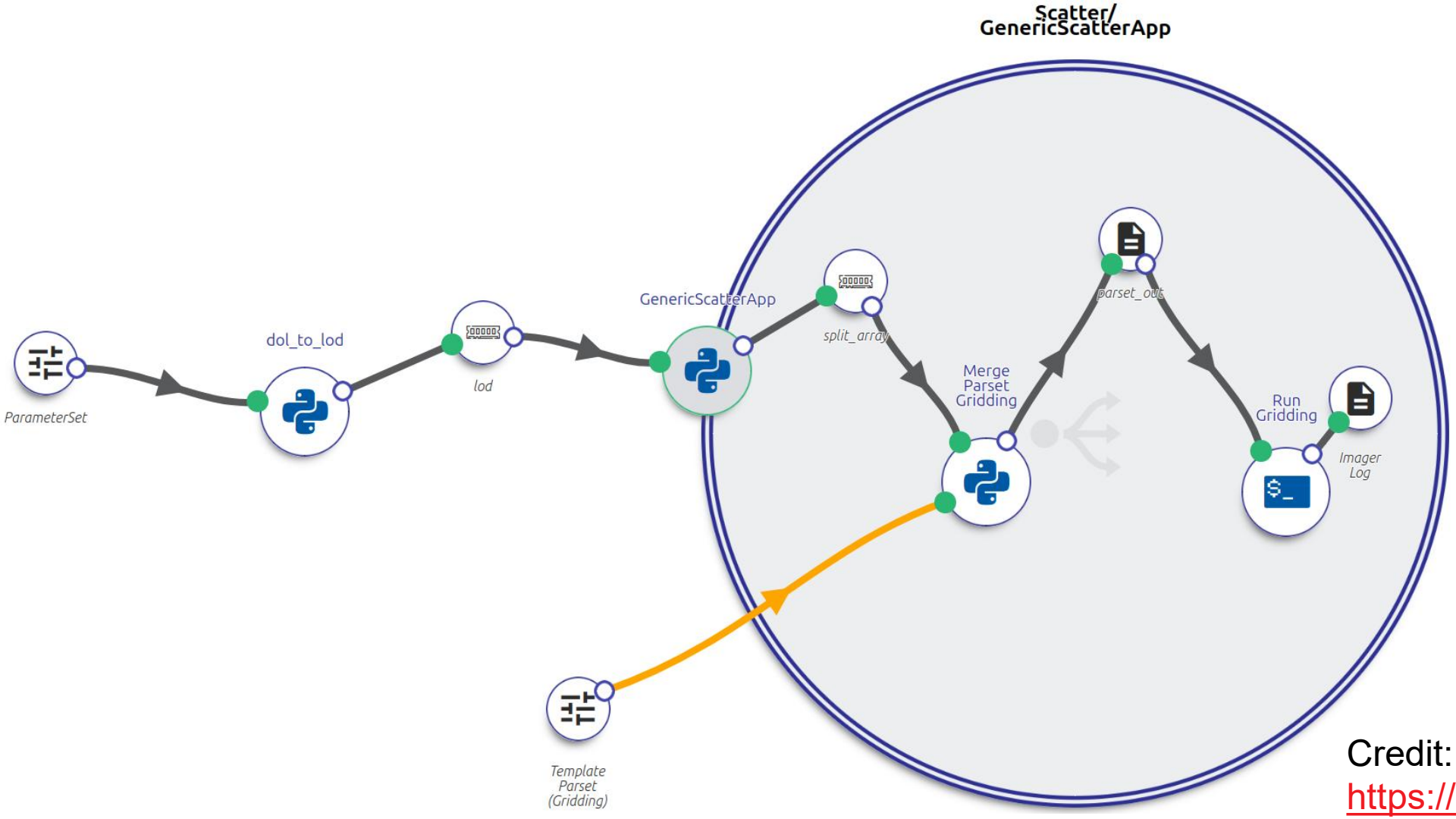
Project Scientists& Engineers

Omar Anwar

Alex Williamson

Gayatri Aniruddha

Deployment: Setonix

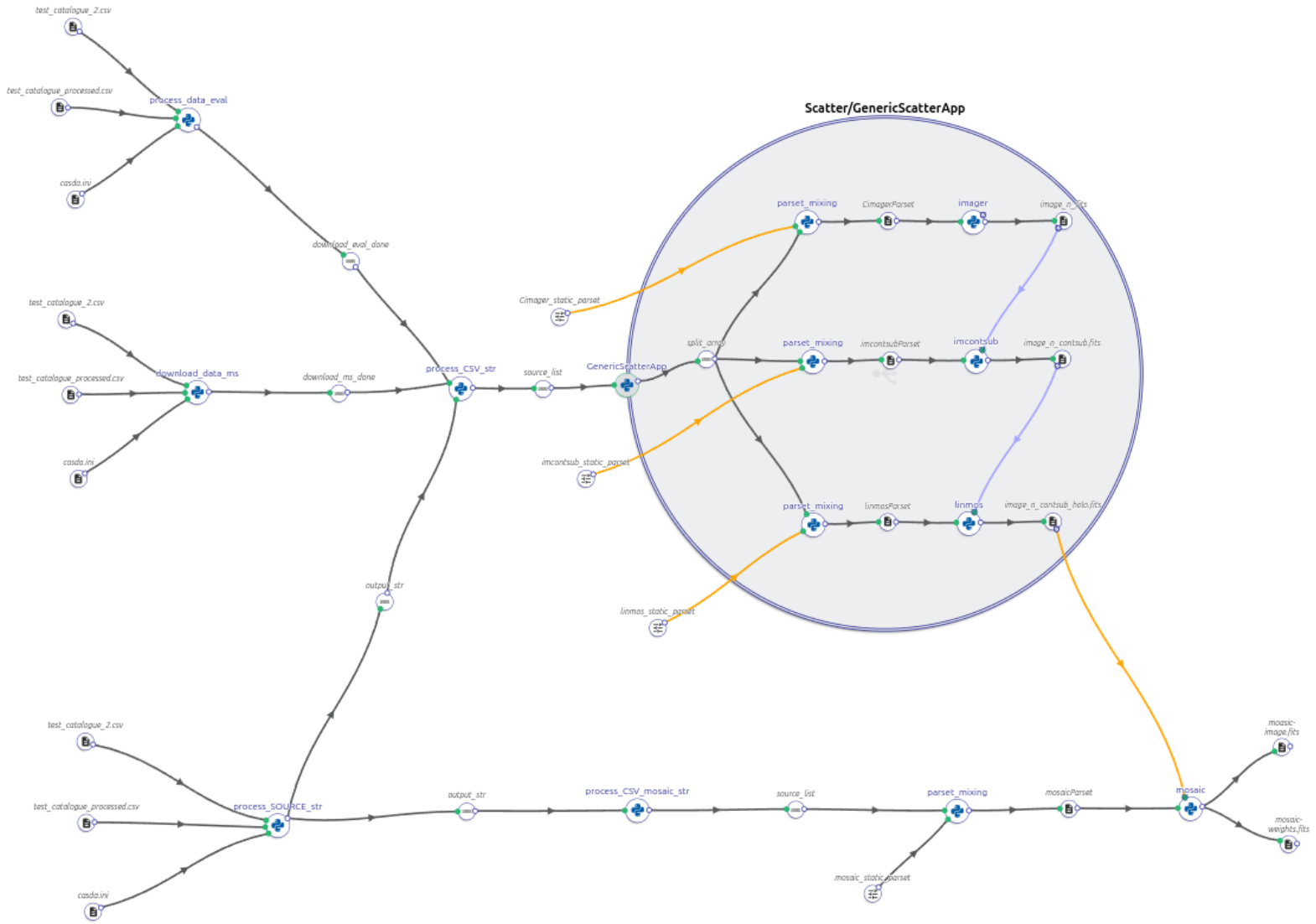


Credit: Dr. Alex Williamson
https://github.com/ICRAR/dingo

Dodson et al, 2025

Wallaby HiRes

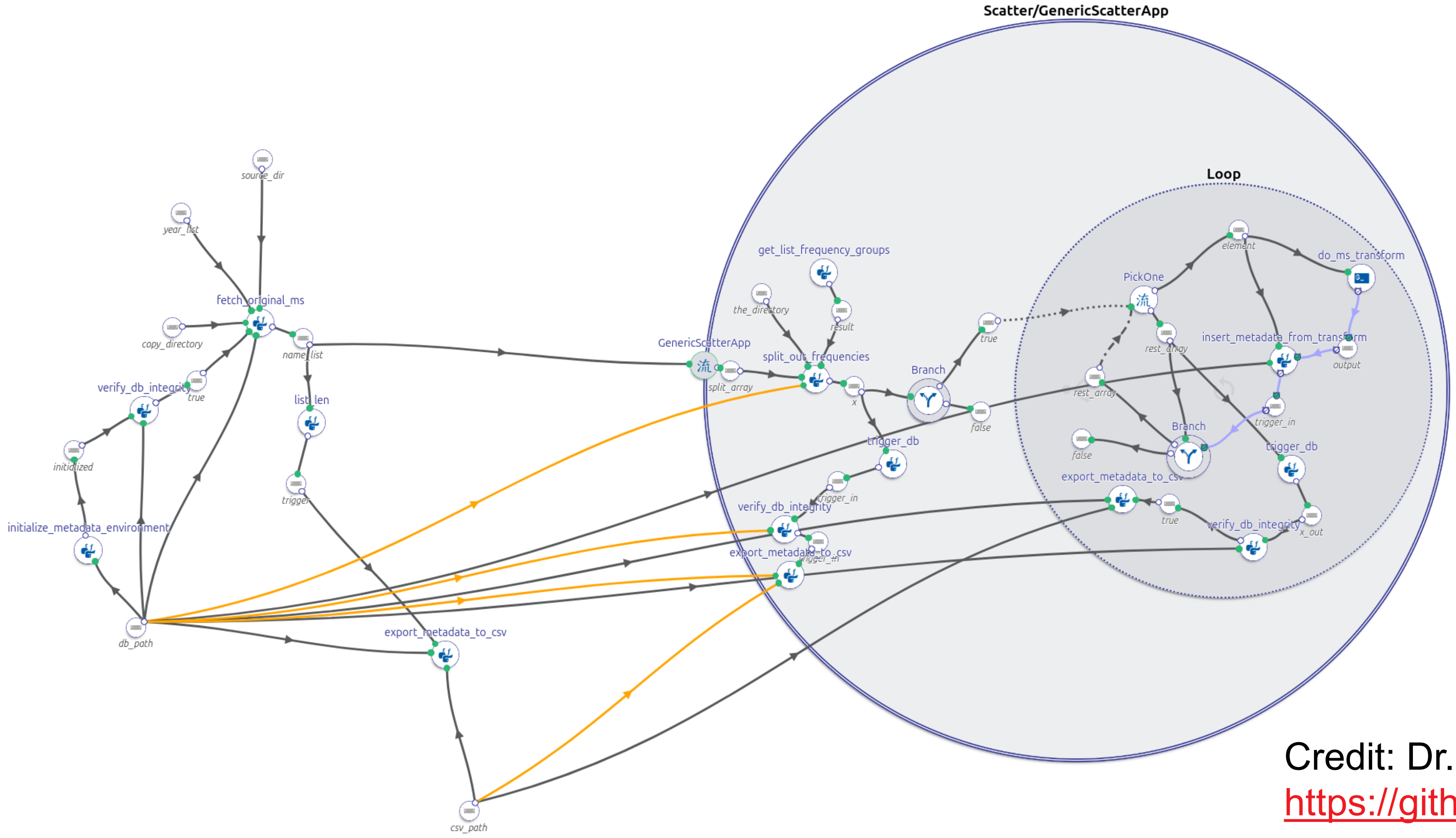
Deployment: Setonix



Credit: Gayatri Aniruddha https://github.com/ICRAR/wallaby-hires

CHILES
COSMOS HI Large Extragalactic Survey

Deployment: Setonix (In-progress!)



Credit: Dr. Omar Anwar
https://github.com/ICRAR/chiles-daliuge

The Pitch

- We would love to engage with non-Astronomy researchers and practitioners
- Access to in-house developers
- Quick(?) turn-around on bugs and features
- Let you focus on research rather than code

流 DALiuGE

EAGLE π

Questions?

ICRAR is a joint venture between Curtin University and The University of Western Australia and receives support from the Western Australian and Australian Governments.