Gladier: Automation with FuncX and Flows

Nickolaus Saint

globus®

Problem: How do we fetch remote data?

- **Data is collected on remote system**
- **How do we move it into an HPC environment to run our FuncX functions?**
- **Automation with Globus Flows**

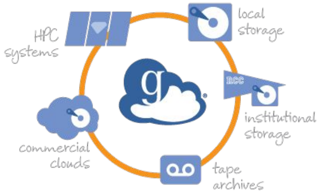
Gladier: The Globus Architecture for Data-Intensive Experimental Research

- Accelerate and simplify flow development and deployment
- Compose reliable services into flexible, secure, distributed flows
- Simplify the connection between instruments and computing facilities
- Automate data collection and publication adhering to FAIR principles

Auth
Flows
Transfer
Share
Search

https://globus.org & https://funcx.org

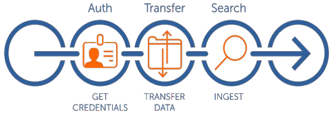
Unified Data Access



Data Transfer

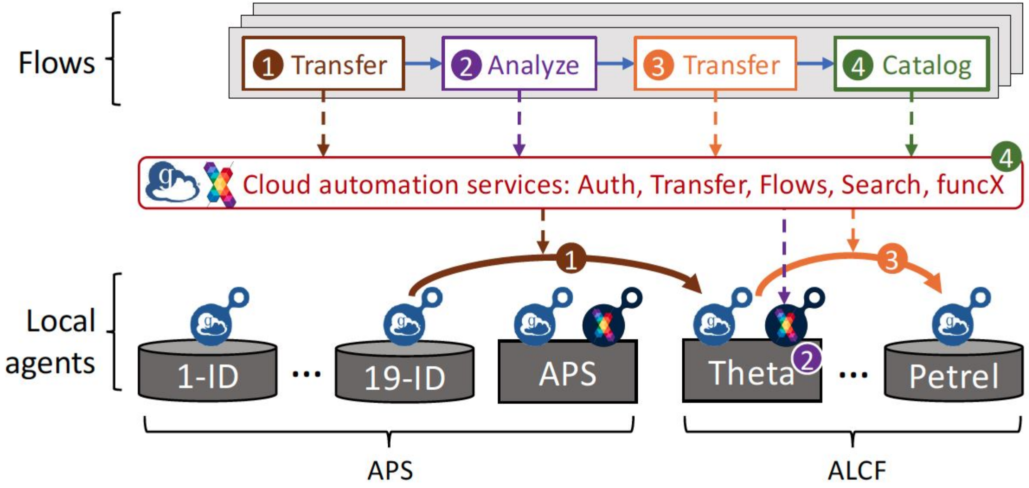
A
fast
secure transfer
reliable
B

Distributed Automation

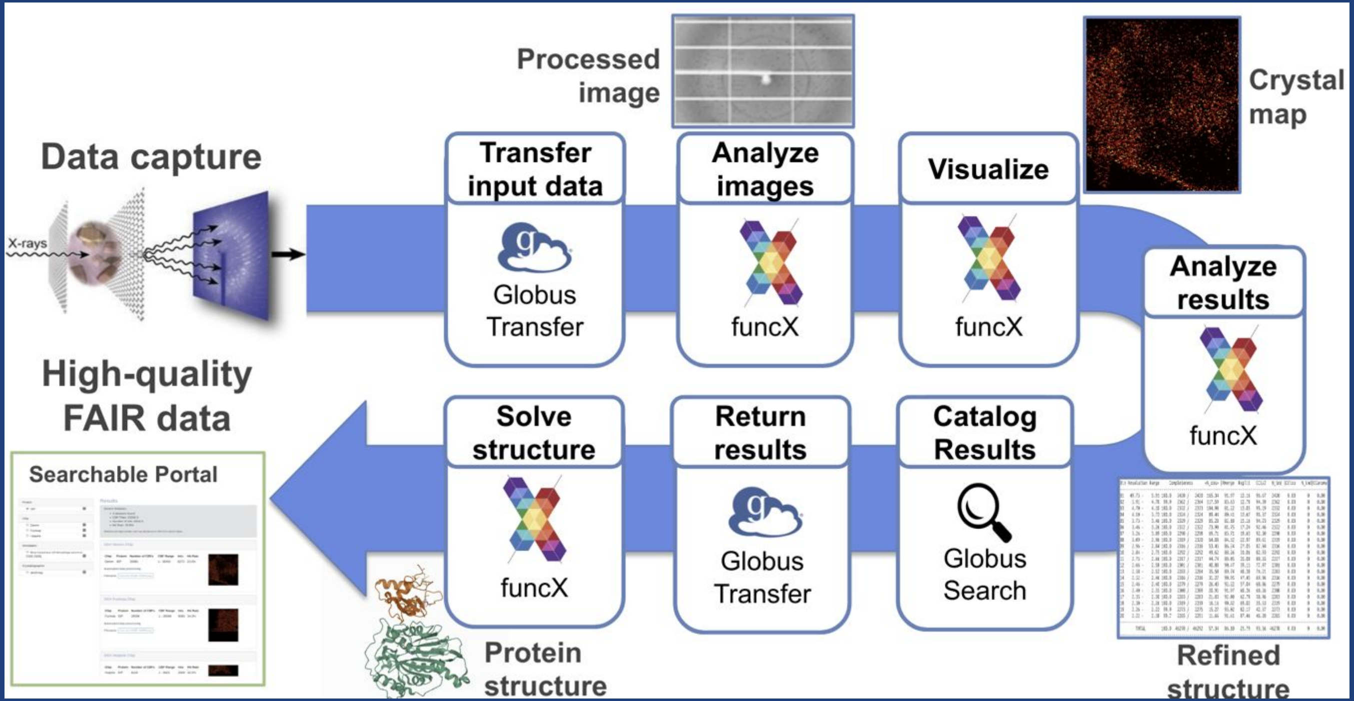


Remote Execution

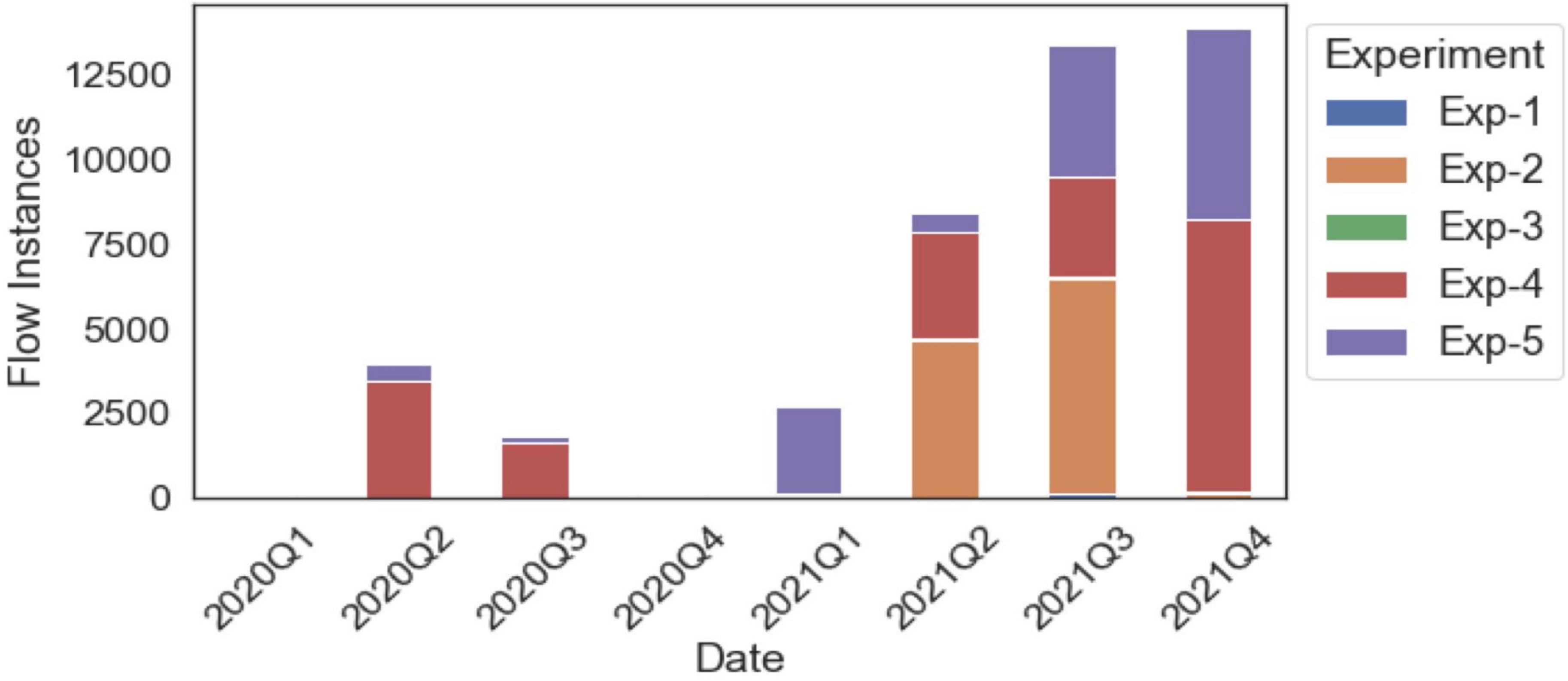
funcX



Automation Pipeline

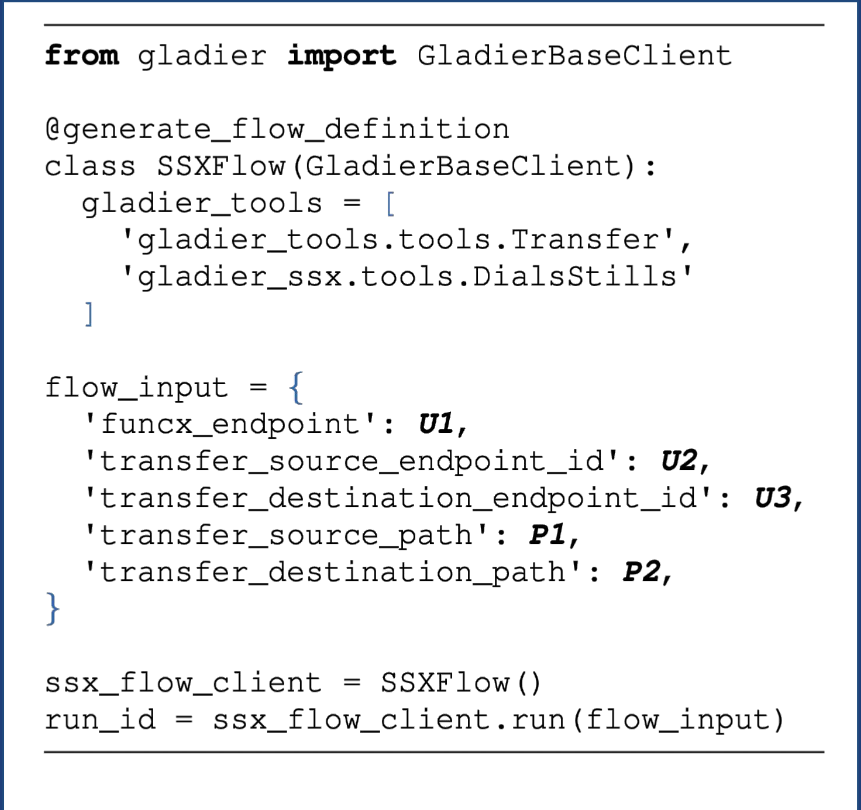


Flows over time for APS experiments 2020-21



Gladier: Automating Around FuncX

- Gladier:
  - Define Tools
  - Specify Input
  - Run!
- Tools:
  - Contain any number of flow states
  - May contain FuncX Functions
  - Define required flow inputs
- Where are the FuncX Function IDs?



```
from gladier import GladierBaseClient

@generate_flow_definition
class SSXFlow(GladierBaseClient):
  gladier_tools = [
    'gladier_tools.tools.Transfer',
    'gladier_ssx.tools.DialsStills'
  ]

flow_input = {
  'funcx_endpoint': U1,
  'transfer_source_endpoint_id': U2,
  'transfer_destination_endpoint_id': U3,
  'transfer_source_path': P1,
  'transfer_destination_path': P2,
}

ssx_flow_client = SSXFlow()
run_id = ssx_flow_client.run(flow_input)
```

Questions

https://gladier.readthedocs.io/