Bundled for Success: A WfBench-Driven Assessment of Texera

Meng Wang

joint work with Shiva Jahangiri, Ketan Maheshwari, Chen Li

Workflow Management Systems (WMS) Are Rapidly Evolving

texera

Apache Airflow

dask

swift-t

Parsl

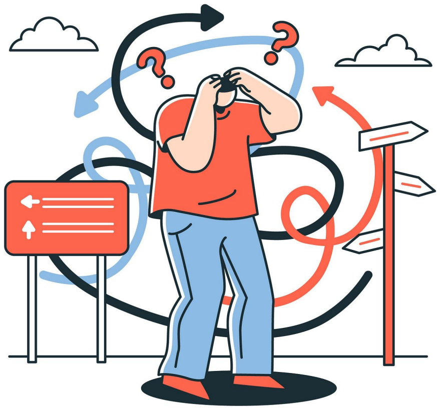
Pegasus

Galaxy PROJECT

Apache Taverna

nextflow workflow management

The Importance of Benchmarking in Workflow Systems



Benchmark

Texera + WfBench: Integration and Evaluation

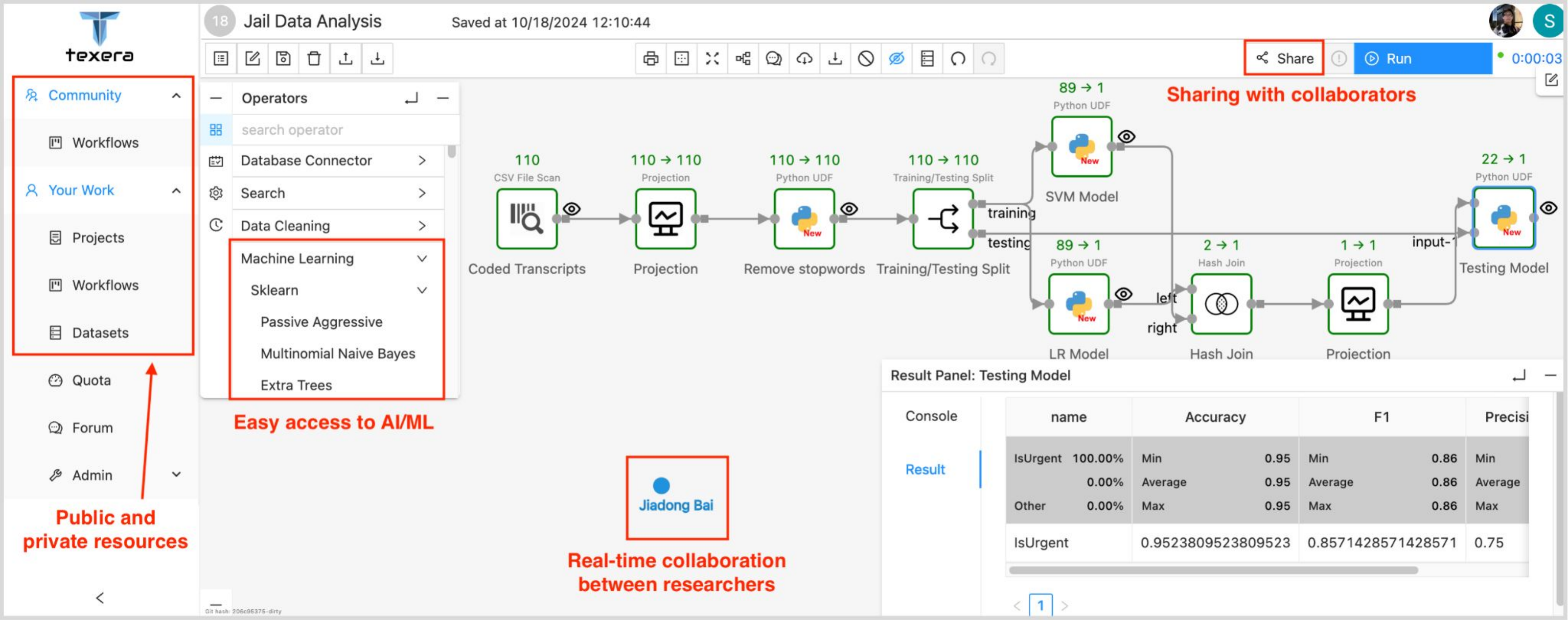
- Introducing Texera to the broader workflow community
- Contributing back by sharing some experience how to integrate a WMS with a benchmark framework

texera

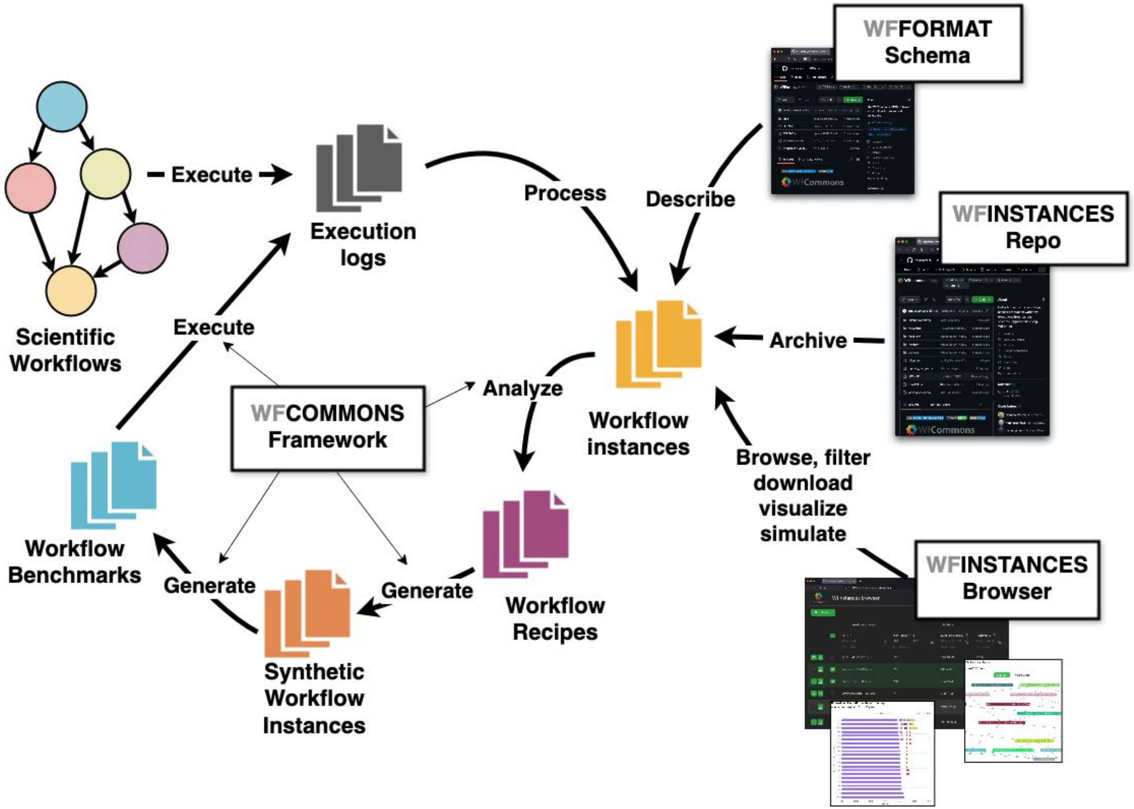
WfCommons

(WfBench: WfCommons' benchmark-generation component)

Texera at a Glance

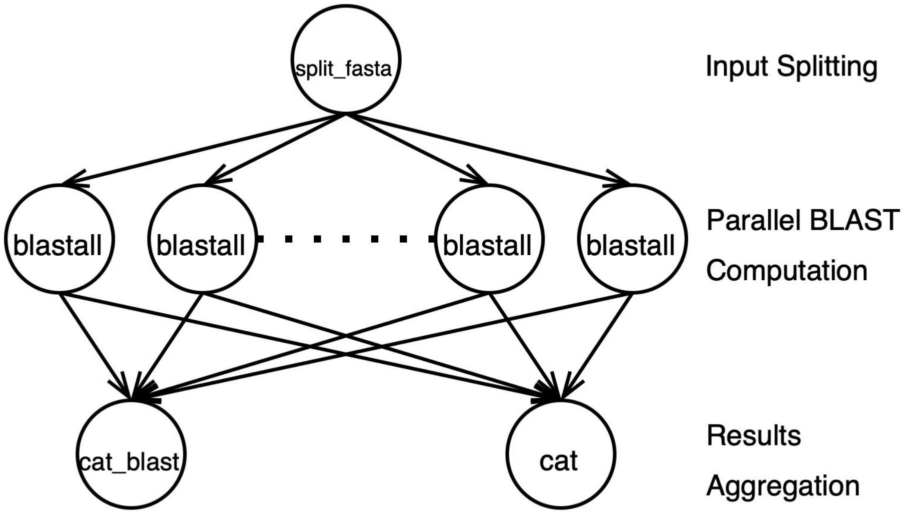


WfCommons at a Glance



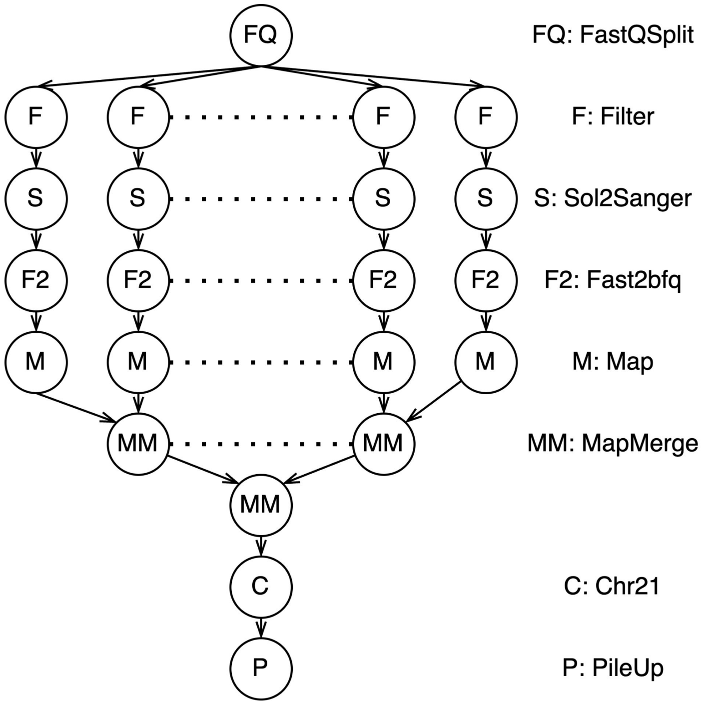
Workflow Selection — BLAST

BLAST Workflow



Workflow Selection — Epigenomics

Epigenomics Workflow



Texera Translator — Motivation and Role

- Conversion: it converts WfBench generated format into Texera-compatible workflows.
- Alignment: it ensures each workflow task is represented in a way that fits Texera's execution model.


```json
{
  "name": "Blast-Benchmark",
  "description": "Instance generated with WfCommons - https://wfcommons.org",
  "createdAt": "2025-06-30T18:00:45.511357-07:00",
  "schemaVersion": "1.5",
  "author": {
    "name": "wangmeng",
    "email": "support@wfcommons.org"
  },
  "workflow": {
    "specification": {
      "tasks": [
        {
          "name": "split_fasta",
          "id": "split_fasta_00000001",
          "parents": [],
          "children": [
            "blastall_00000002",
            "blastall_00000003",
            "blastall_00000004",
            "blastall_00000005",
            "blastall_00000006",
            "blastall_00000007",
            "blastall_00000008",
            "blastall_00000009",
            "blastall_00000010",
            "blastall_00000011",
            "blastall_00000012",
            "blastall_00000013",
            "blastall_00000014",
            "blastall_00000015",
            "blastall_00000016",
```


WfBench benchmark format
(Json)

Translating

→

(Texera Translator)

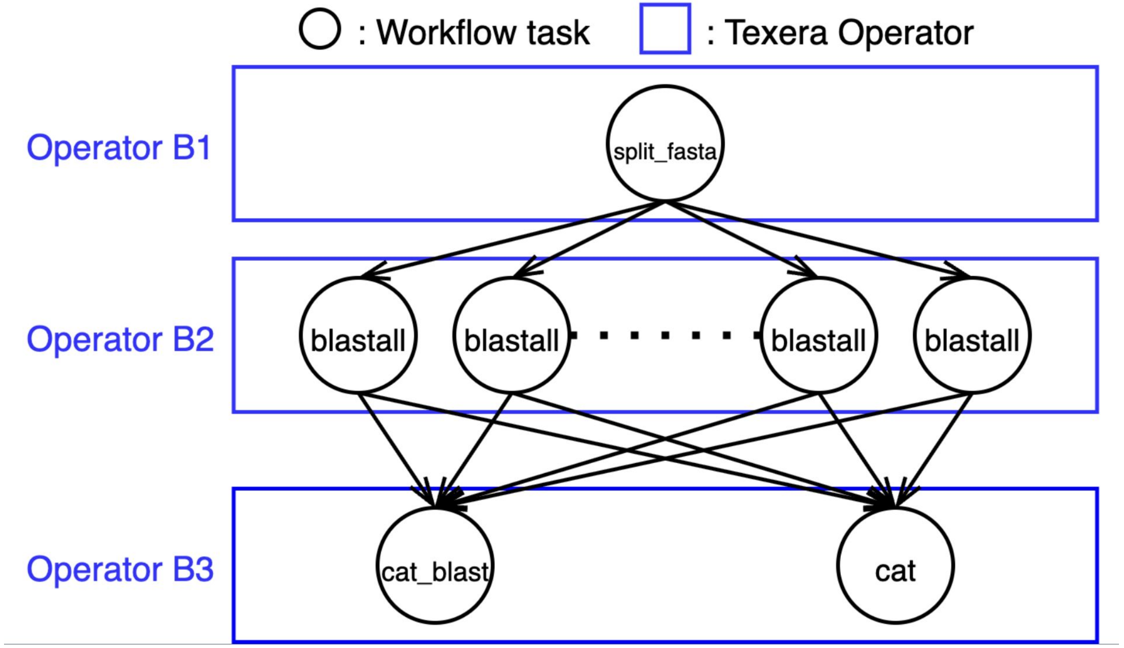

```json
{
  "operators": [
    {
      "operatorID": "Bash-operator-7af9a78d-d9ef-4c25-a338-10147cc4a125-chr21_00000001",
      "operatorType": "Bash",
      "cmd": "./bin/wfbench --name chr21_00000001 --percent-cpu 0.6 --cpu-work 10 --output-files \"
      "inputPorts": [
        {
          "portID": "input-0",
          "displayName": "",
          "allowMultiInputs": true,
          "isDynamicPort": false
        }
      ],
      "outputPorts": [
        {
          "portID": "output-0",
          "displayName": "",
          "allowMultiInputs": false,
          "isDynamicPort": false
        }
      ]
    },
    {
      "operatorID": "Bash-operator-17975654-e55b-4c0f-bcd5-7d97f2d0d778-mapMerge_00000002",
      "operatorType": "Bash",
      "cmd": "./bin/wfbench --name mapMerge_00000002 --percent-cpu 0.6 --cpu-work 10 --output-files
      "inputPorts": [
        {
          "portID": "input-0",
          "displayName": "",
          "allowMultiInputs": true,
          "isDynamicPort": false
        }
```


Texera format
(Json)

Translator Design — Approach 1

Advantages:
- Fewer operators → simpler workflow structure
- Fewer edges → less inter-operator data transmission

Limitations:
- Limited visibility into individual tasks
- Harder to pause, debug, or monitor tasks inside a grouped operator

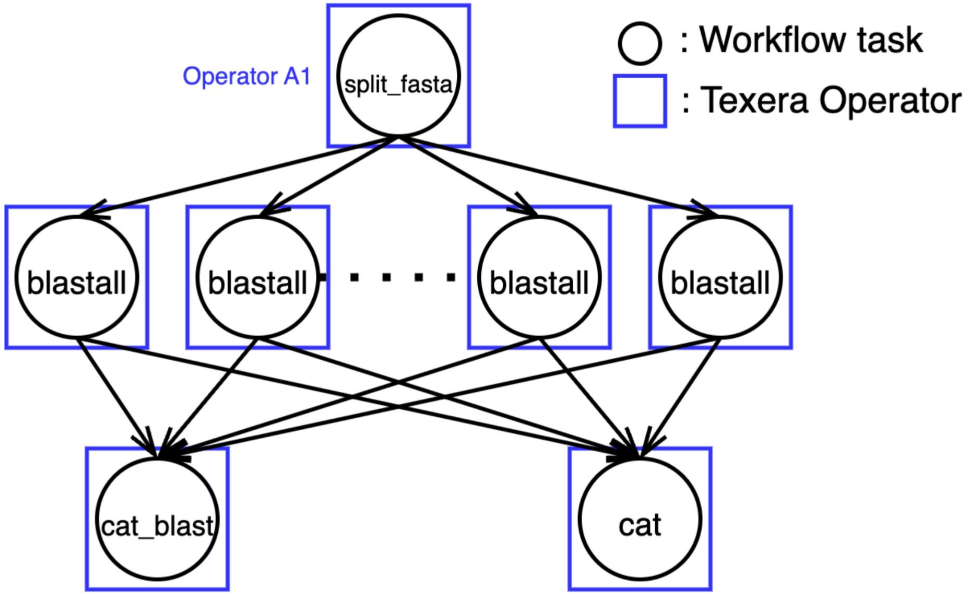


Encapsulating each workflow level as a Texera operator

Translator Design — Approach 2

Advantages:
- Fine-grained monitoring, pausing, and debugging
- Better support for interactive workflow execution
- Better aligned with Texera's operator model

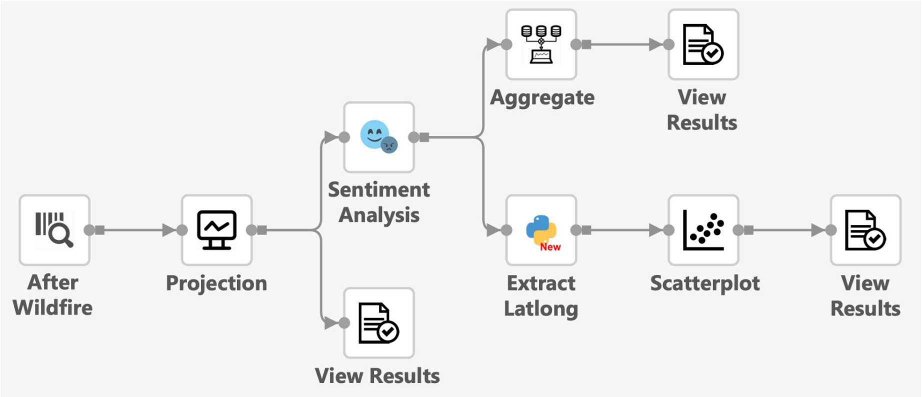
Limitations:
- Larger number of operators
- More complex workflows to schedule



Encapsulating each task as a Texera operator

Integration Consideration — Texera Interfaces

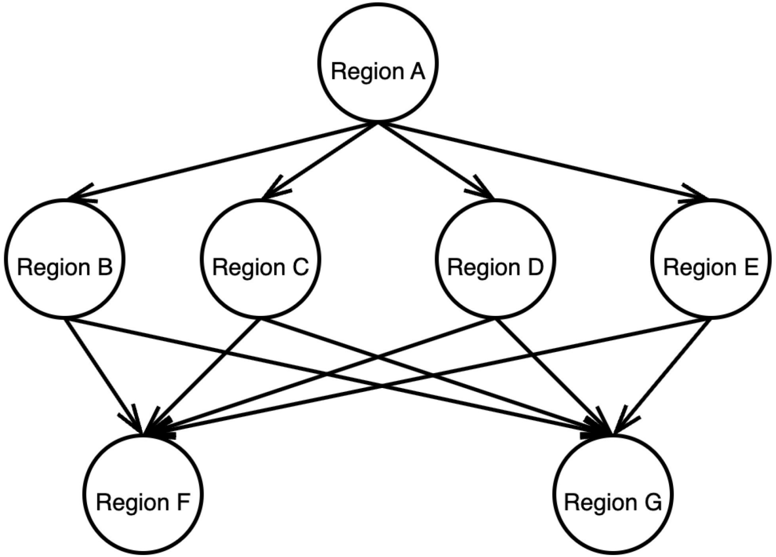
Gui



Python Client (command-line based)

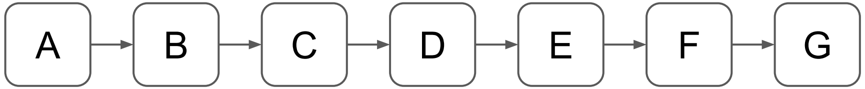
```
Available commands:
  exec <plan.json> [<name>]
  page <operatorId> <size> <pageIdx> [--export <dir>]
  kill

> 2025-09-03 03:57:37 | INFO     | WorkflowState → Uninitialized
exec ../example.json
> 2025-09-03 03:57:45 | INFO     | Sent execute request: name=example plan=../example.json
2025-09-03 03:57:45 | INFO     | WorkflowState → Initializing
2025-09-03 03:57:46 | INFO     | WorkflowState → Initializing
2025-09-03 03:57:48 | INFO     | WorkflowState → Running
2025-09-03 03:57:50 | INFO     | Result Operator Update:
 PythonUDFV2-operator-648785ec-a5cb-47af-a2f3-f110dcdf58b9 → totalNumTuples=0
2025-09-03 03:58:09 | INFO     | WorkflowState → Completed
2025-09-03 03:58:09 | INFO     | Final execution time: 23.50s
2025-09-03 03:58:09 | INFO     | Result Operator Update:
 PythonUDFV2-operator-648785ec-a5cb-47af-a2f3-f110dcdf58b9 → totalNumTuples=800
page PythonUDFV2-operator-648785ec-a5cb-47af-a2f3-f110dcdf58b9 50 1
> 2025-09-03 03:58:22 | INFO     | Sent pagination request:
 operator=PythonUDFV2-operator-648785ec-a5cb-47af-a2f3-f110dcdf58b9 size=50 page=1
2025-09-03 03:58:22 | INFO     | Result:
 operatorID=PythonUDFV2-operator-648785ec-a5cb-47af-a2f3-f110dcdf58b9 size=50 page=1
2025-09-03 03:58:22 | INFO     | Schema: [{'attributeName': 'test', 'attributeType': 'integer'},
 {'attributeName': 'message', 'attributeType': 'string'}]
2025-09-03 03:58:22 | INFO     | Row: {'test': 1, 'message': 'hello'}
2025-09-03 03:58:22 | INFO     | Row: {'test': 1, 'message': 'hello'}
```

Integration Consideration — Region Scheduler Constraints

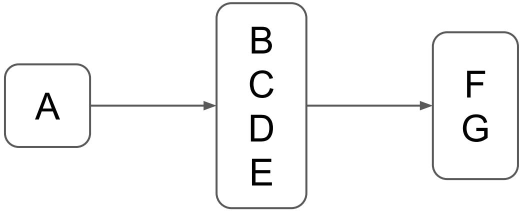


- A region is a subset of operators connected only by non-blocking edges
- In our WfBench integration case, each operator is its own region

Region execution order (Sequential scheduler):

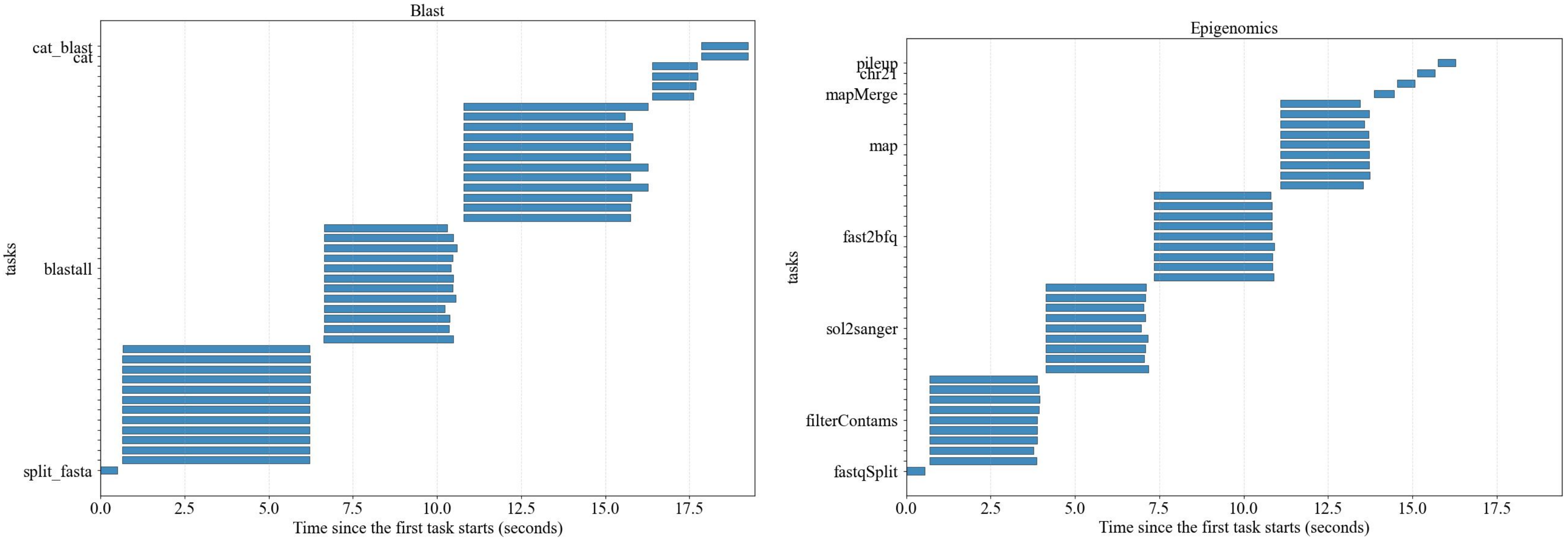


Region execution order (Parallel scheduler):

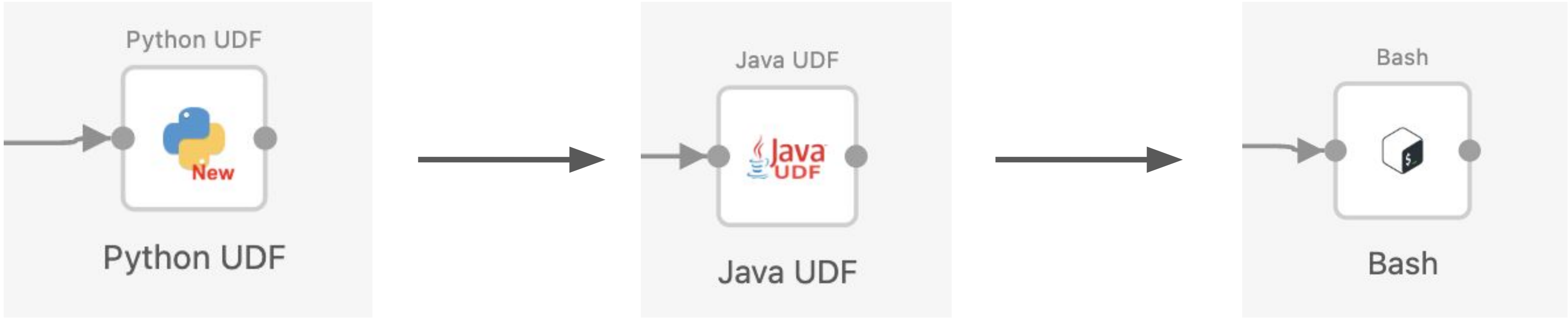


Integration Consideration — Region Scheduler Constraints

Max-Concurrent-Region: 12



Integration Consideration — Optimization of UDF(User Defined Function) Operators



UDF Operator

UDF Operator

Native Operator

Experiment Setup

Workflows:

- Blast
- Epigenomics

Workload Scenario:

- CPU-intensive (cpu_work = 600, data = 10)
- Balanced (cpu_work = 200, data = 200)
- Data-intensive (cpu_work = 10, data = 600)

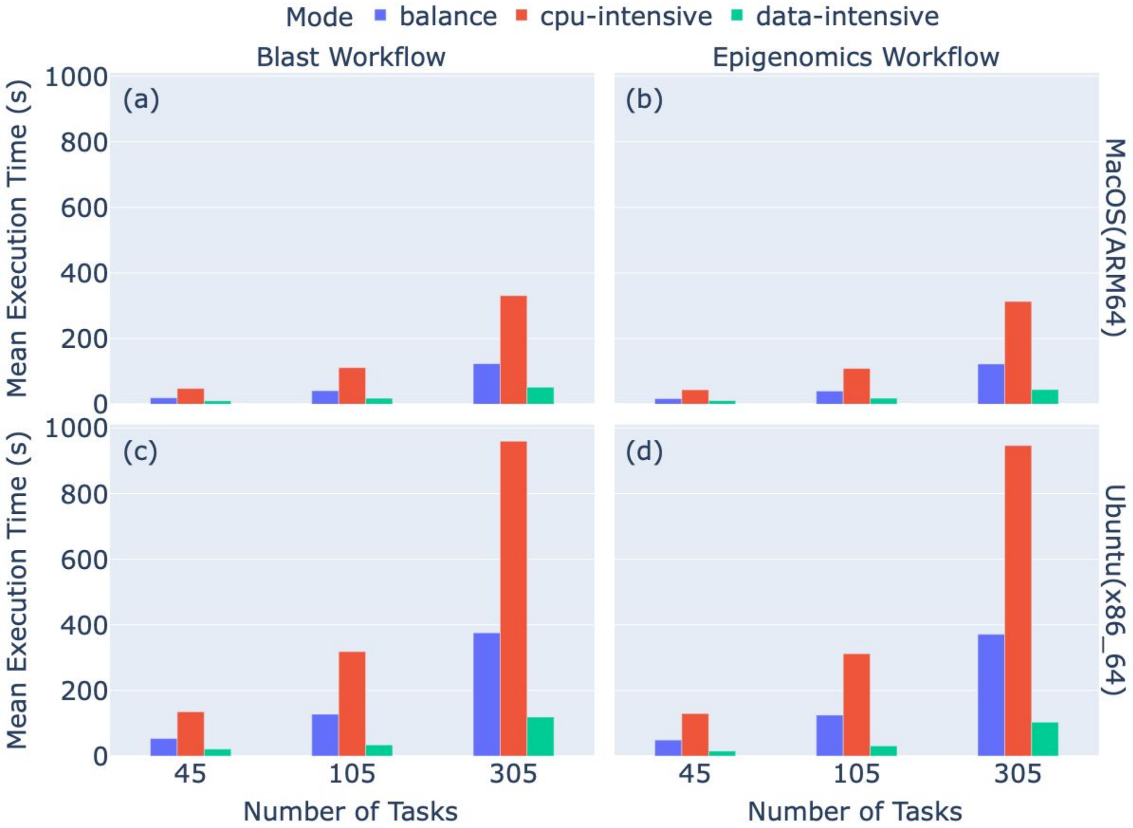
Workflows sizes:

- 45 tasks
- 105 tasks
- 305 tasks

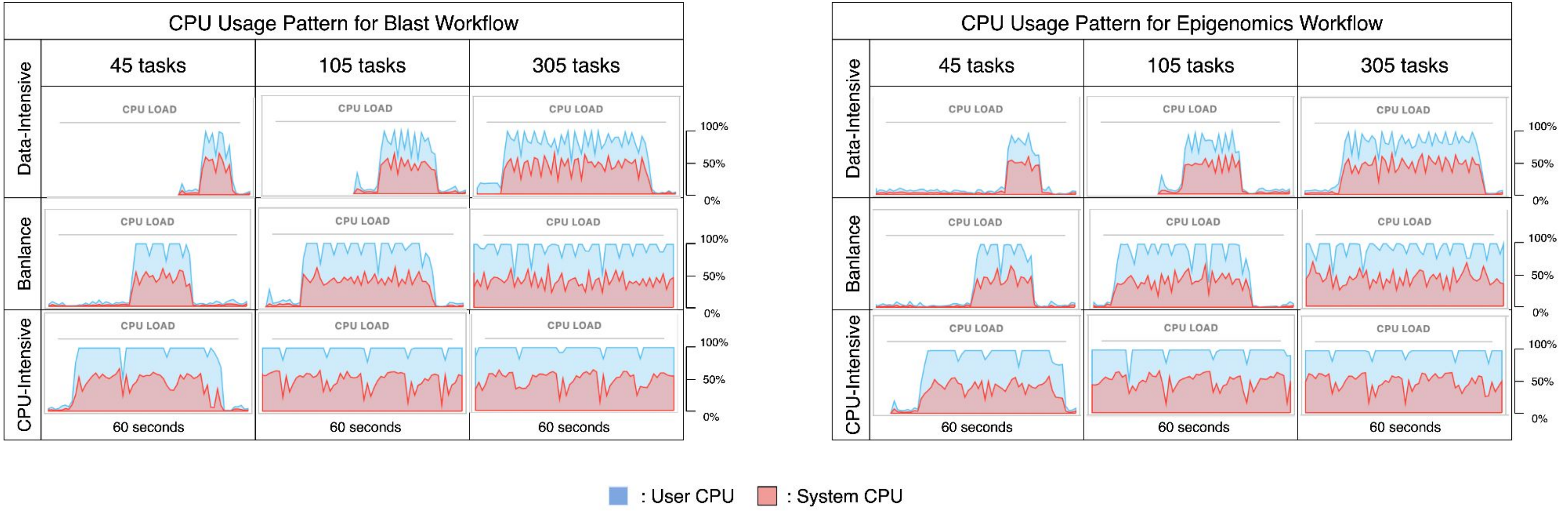
Hardware:

- MacBook Pro (12 cores, MacOS)
- Dell Server (4 cores, Linux)

Execution Time Results



CPU Usage Patterns



Conclusion and Future Directions

- Integrated Texera with WfBench for standardized benchmarking
- Demonstrated Texera's adaptability across workflows, profiles, and platforms
- Validated stable and efficient performance of Texera
- Plan to conduct comparative evaluations against other workflow systems
- Aim to optimize max-concurrent-region for dynamic scheduling performance

Thank You!

Meng Wang

Shiva Jahangiri

Ketan Maheshwari

Chen Li

Texera

Texera at https://texera.io

UCI University of California, Irvine

1851 Santa Clara UNIVERSITY

OAK RIDGE National Laboratory