Improving FAIRness of AI/ML in Earth Science via Reproducible Big Data Analytics in the Cloud

Session: "Improving 'FAIRness' and 'Fairness' of AI/ML in Geoscience
ESIP January 2022 Meeting

Jianwu Wang (jianwu@umbc.edu)[1], Xin Wang (xinwang11@umbc.edu)[1], Jinbo Wang (jinbo.wang@jpl.nasa.gov)[2]

[1] Big Data Analytics Lab, University of Maryland, Baltimore County

[2] PO.DAAC, Jet Propulsion Laboratory, California Institute of Technology

Introduction

- **One main FAIRness challenge for AI/ML model is reproducibility.** By reproducing an existing computational experiment and obtaining consistent results, we can have more confidence in the research. Further, besides reproducing the exact process, we could easily explore how the experiment behaviors with different input datasets, execution arguments and environments.
- **Cloud-based reproducibility** has been a major approach for reproducible computing services because the full stack of the computation environment, including data, software and hardware, could all be provisioned and shared via various cloud services.
- **Cloud will become a major environment for Earth Science community** because more and more Earth data from NASA and other data providers will be stored on the Cloud.

Our Work

- Our proposed approach and toolkit, Reproducible and Portable big data Analytics in the Cloud (RPAC), integrate serverless computing techniques to automate end-to-end big data analytics pipeline.
- To deal with the vendor lock-in challenge, we propose a Cloud Agnostic Application Model (CAAM) to support execution and reproduction with different cloud providers. Our RPAC toolkit supports both AWS and Azure cloud environments.
- We benchmark both CPU-based and GPU-based big data analytics applications using our RPAC toolkit.

Background

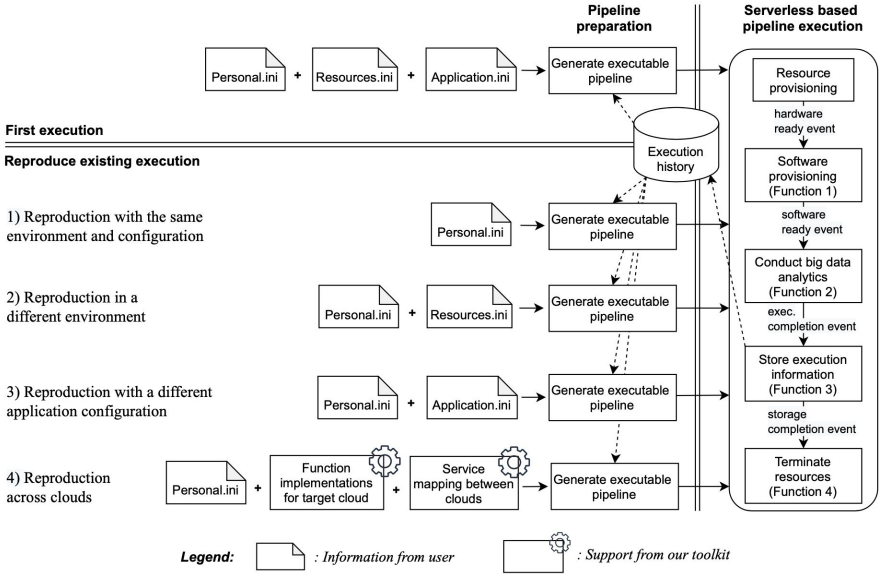
- Besides re-running exactly the same application, three aspects could vary during reproduction of an existing application for specific reasons:
  - Different **application configuration** (dataset, application argument, etc.) to know how the application performs with different datasets or arguments.
  - Different cloud provider **hardware environment** (virtual machine type and number, etc.) within the same cloud provider to test scale-up and scale-out.
  - Different **cloud provider** to avoid vendor lock-in problem.

Background (2)

- *Serverless computing* asks the cloud provider to allocate machine resources on demand, taking care of the servers on behalf of their customers.
- Advantages of serverless computing in cloud:
  - It responds to user service requests **without maintaining back-end servers** in the cloud.
  - It employs **Function as a Service (FaaS)** architecture that allows customers to develop separate functions directly rather than standalone cloud applications. Each application logic/pipeline is split into functions and application execution is based on internal or external events.

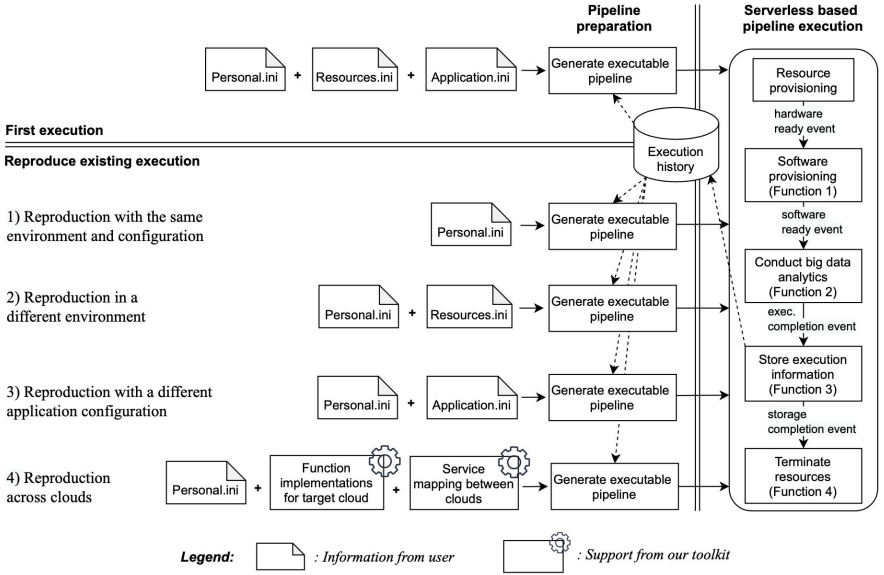
Overview of Reproducible and Portable big data Analytics in the Cloud (RPAC)

- RPAC enables users easily re-run previous experiments with the same or different setups.
  - First execution of an application
  - Reproduction of the existing execution by querying historical configurations
- The execution and reproduction is fully automated by the RPAC toolkit and can be done via a single command.
- Minimal information requirements from users: application program URI, cloud instance type, etc.



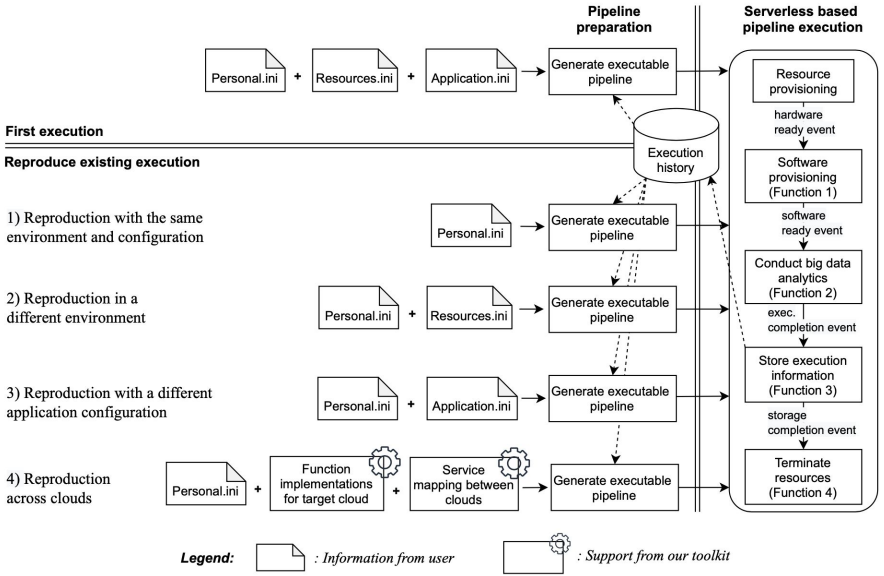
RPAC: First Execution of an Application

- Prepare **configurations** for the whole execution.
  - Personal, resources, and application config
- Create the executable **pipeline** for the target cloud.
- Execute analytics in the cloud and output results to the storage automatically.



RPAC: Reproduction of an Existing Execution

- Reproduce an existing execution with the exact environment and configuration.
  - Directly use **pipeline** file within the storage
- Reproduce in different environment or application configuration.
  - Combine **changed configurations** with the historical execution information to generate a new **pipeline**
- Reproducing on a different cloud.
  - Provides **cloud service mapping** and implementations of **serverless functions** to generate a new **pipeline**



Scalable Execution for Big Data Analytics

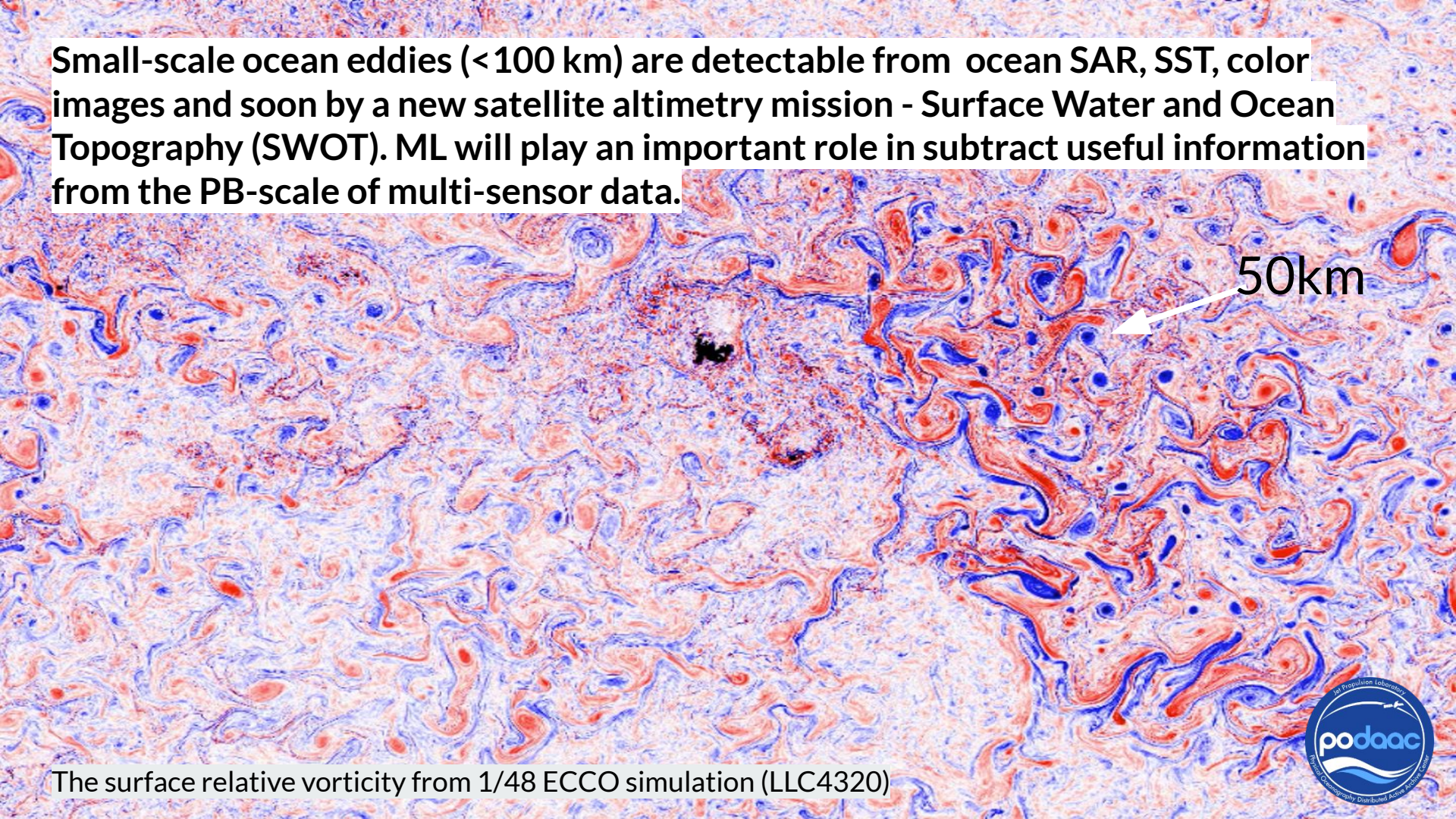
We also provide three parallel frameworks of scalable execution in the cloud:

- **Spark-based** big data analytics on virtual CPU nodes.
    - The resource manager like YARN NodeManager initiates the environment from a pulled docker image, and allocates one virtual instance in cluster as the master while others as workers. The master runs Spark command.
- **Dask-based** big data analytics on virtual CPU nodes.
    - Each virtual instance in cluster initiates one docker container and our pipeline assigns one of the containers to be the Dask scheduler and others to be workers. The scheduler runs Dask command.
- **Horovod-based** big data analytics on virtual GPU nodes.
    - RPAC executes multi-instance GPU-based data analytics within our pre-built Docker containers, setting one of them as the primary worker and others as secondary workers. The primary worker runs the MPI parallel command while secondary workers listen to a specific port.

Earth Science Applications

- **Cloud property retrieval:** It trains a Random Forest machine learning model for cloud mask and cloud thermodynamic-phase retrieval from VIIRS and CALIOP satellite observations.
- **Ocean eddy identification:** It trains a CNN based deep learning model for binary classification and YOLO deep learning model for ocean eddy bounding boxes (object detection) from SAR datasets.

Small-scale ocean eddies (<100 km) are detectable from ocean SAR, SST, color images and soon by a new satellite altimetry mission - Surface Water and Ocean Topography (SWOT). ML will play an important role in subtract useful information from the PB-scale of multi-sensor data.



The surface relative vorticity from 1/48 ECCO simulation (LLC4320)

FAIRness Requirements for Ocean Eddy Detection

We think cloud services can help make the whole pipeline more FAIR and reproducible

- How to easily prepare training data and make them FAIR?
  - We are exploring AWS SageMaker to access data from AWS and do collaborative labeling
  - One difficulty is how to seamlessly load NASA (PO.DAAC cloud) data into SageMaker
- How to record machine learning model training process for easy reproducibility
  - Our toolkit can help automated model training and reproducibility by leveraging AWS cloud services
- How to iteratively improve ocean eddy detection capability with collaborative efforts?
  - The services/tools above could help others easily add additional data for better model accuracy and/or build better machine learning models for fair comparison

Conclusions

- By leveraging serverless, containerization and adapter design pattern techniques, our RPAC toolkit can achieve reproducibility, portability and scalability for cloud based big data analytics.
- We are applying the toolkit with Earth science applications.

Detailed Information

- Paper and GitHub repository of our toolkit: https://bdal.umbc.edu/tools/#reproducible-data-analytics
- **Toolkit demo** at today's Research Showcase Poster & Demo Live Event (4:30-6:00 ET, 1/19/2022)

Acknowledgement

- Grants from NSF, NASA, ARL and ESIP