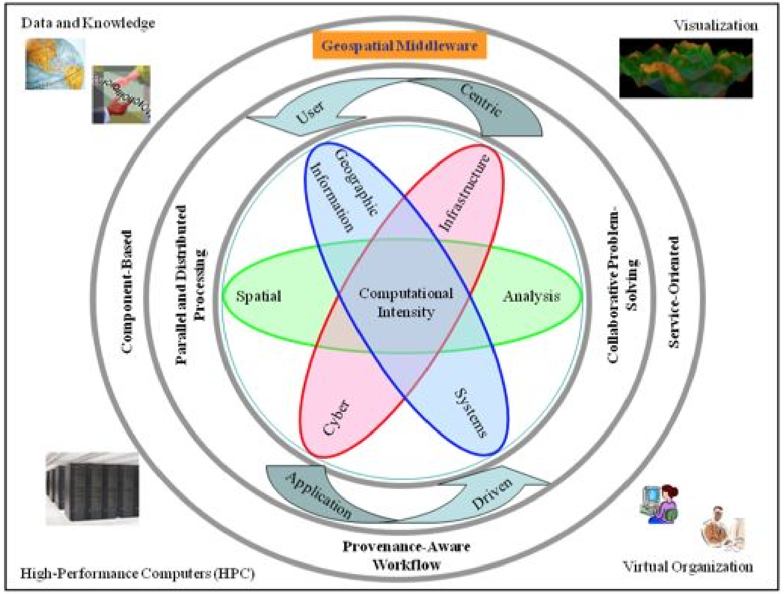
Container Usage within CyberGIS

Jeff Terstriep[1,3] , Yan Liu[1,2,3] and Dandong Yin[1,2,3]
[1]CyberGIS Center for Advanced Digital and Spatial Studies
[2]Department of Geography and Geographic Information Science
[3]National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign

Container Analysis Environments Workshop
August 14, 2017, Urbana, IL

CyberGIS

- CyberGIS -- geographic information science and systems (GIS) based on advanced CI
  - Innovate new-generation GIS
  - Focus on computational and data-intensive geospatial problem-solving within various research and education domains
  - Bridge gaps between geospatial big data, software and applications through innovative cyberGIS supercomputer -- ROGER



Wang, S. (2010) A CyberGIS Framework for the Synthesis of Cyberinfrastructure, GIS, and Spatial Analysis. *Annals of the Association of American Geographers*, 100(3): 535-557

CyberGIS Needs for Containers

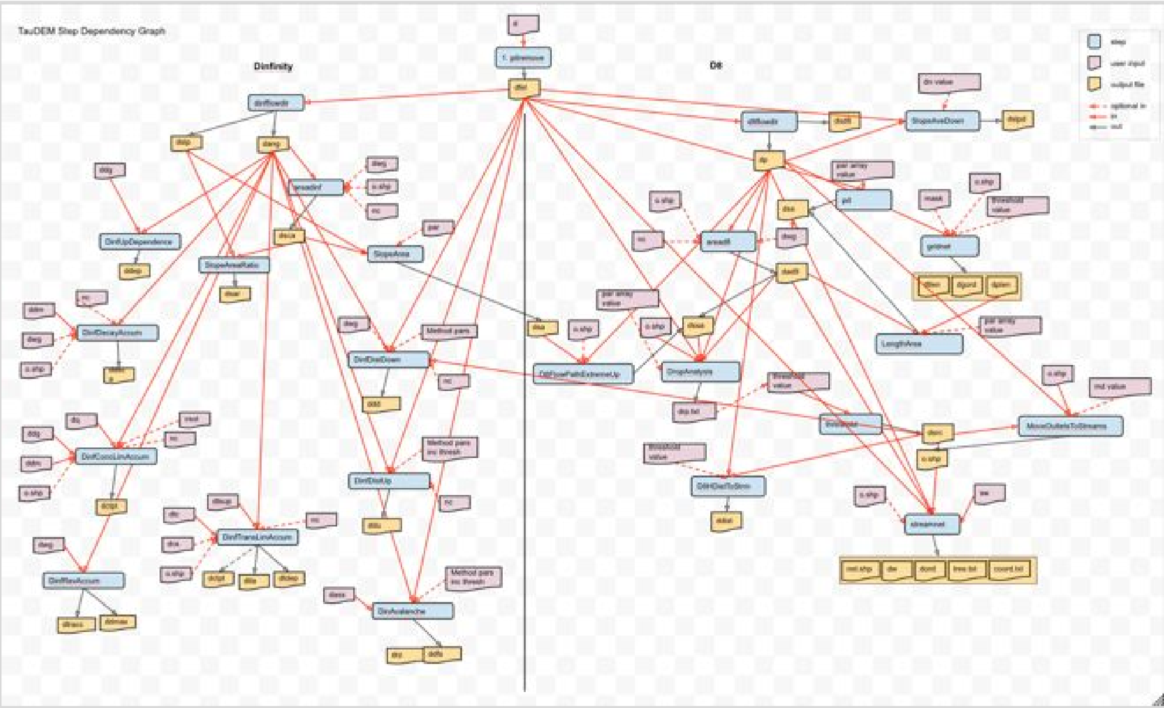
- Gateway app development
  - Lower development cost
  - Help production deployment
- Cyberinfrastructure integration in containers
  - Big data access
  - Complex workflow execution
  - Transparent integration as a software as service solution
- Interactive analysis environment for geospatial studies
  - Geospatial computing environment in containers
  - Geo-visualization in containers
- Collaborative research and development
  - Methodology co-development, testing, validation, and result sharing
  - Faster turnaround time, more productive

Case Studies

- Gateway: web app and Jupyter
  - Analytical apps: web app *vs.* Jupyter-based app
  - Data apps: production data delivery environment *vs.* community engagement for further requirements
  - Containerization for both application development modes
- Collaborative research and development: National Flood Interoperability Experiment (NFIE)

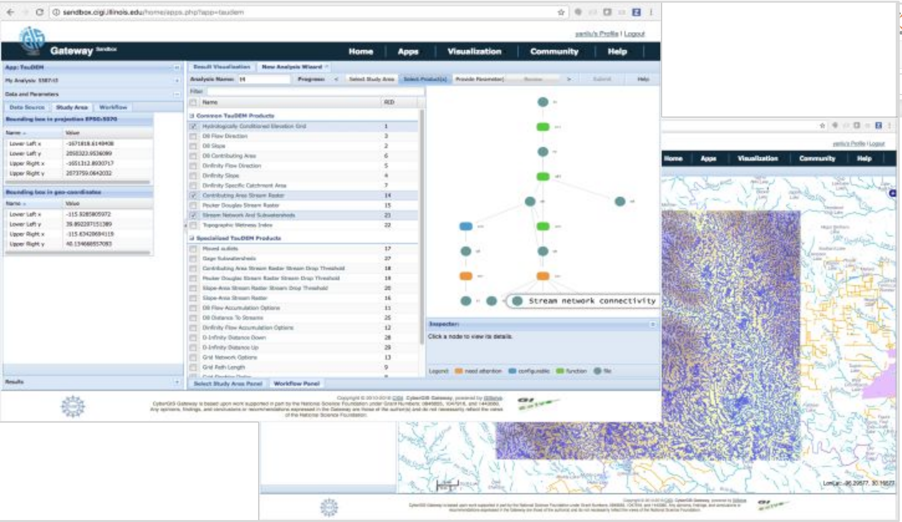
Analysis Example: TauDEM

- High-performance hydrologic information analysis
- 30+ functions with dependencies
- Web app built
  - Computed on HPC
  - Input data: 700GB national elevation dataset
- Jupyter notebook built
  - Computed in container or on HPC
- Which one is better?
  - Who are the audience?

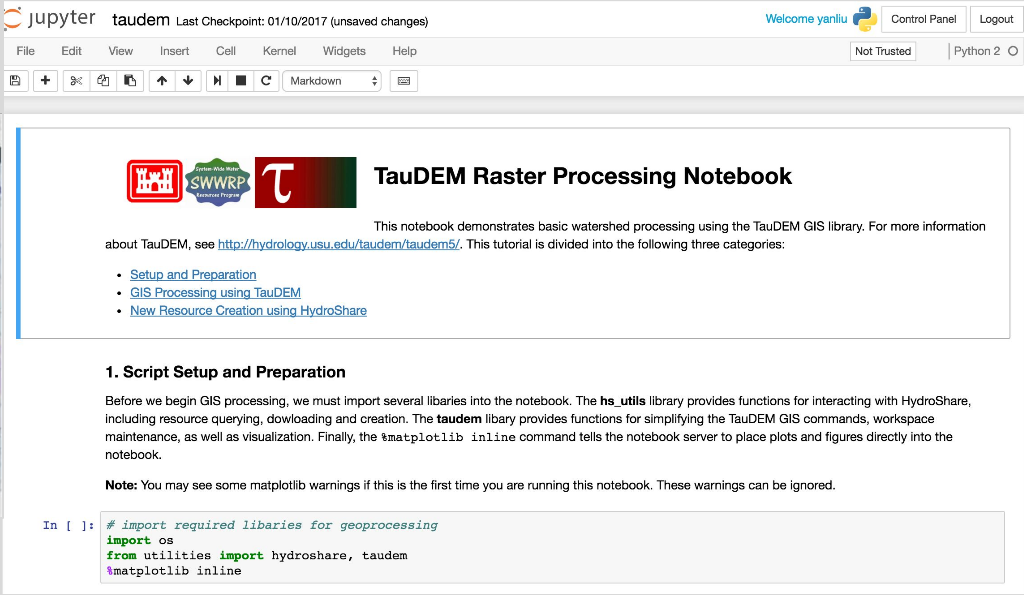


Web app *vs.* Jupyter

TauDEM web app



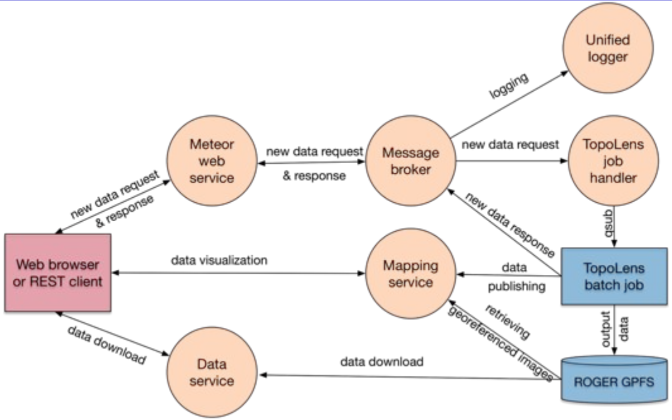
TauDEM Jupyter notebook

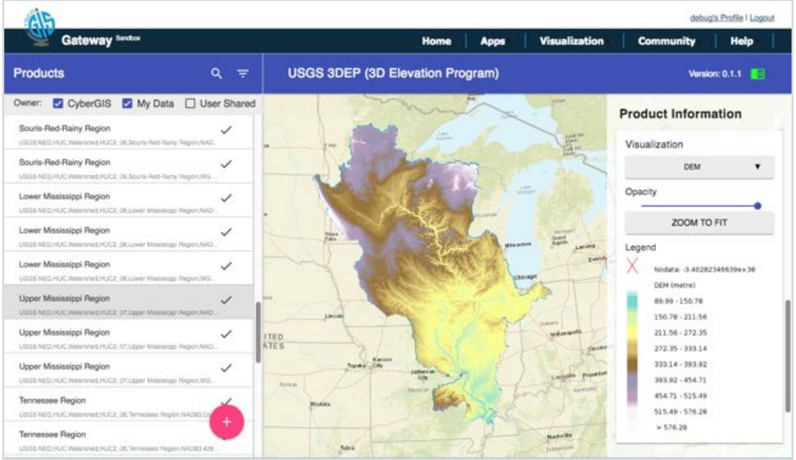


Fan, Y., Liu, Y.Y., Wang, S., Tarboton, D., Yildirim, A., and Wilkins-Diehr, N. 2014. “Accelerating TauDEM as a Scalable Hydrological Terrain Analysis Service on XSEDE”. Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment (XSEDE'14), pages: 5:1–5:2. ACM Press. July 13-18, Atlanta, GA. http://doi.acm.org/10.1145/2616498.2616510

Data App: TopoLens

- End user-oriented national topographic data delivery service
- Web app built on microservices in containers
- Computation on HPC
- Value added services
  - Jupyter can help to gather community requirements

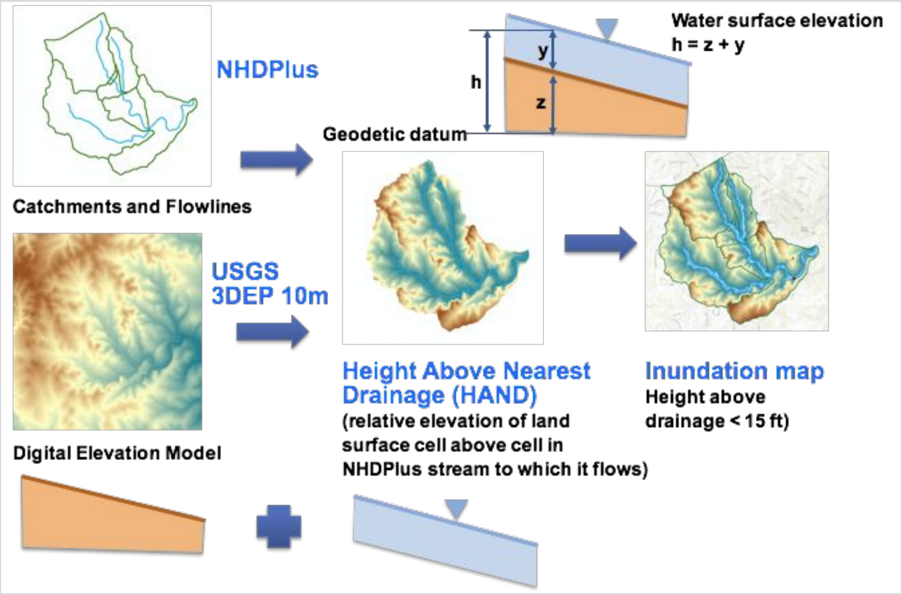


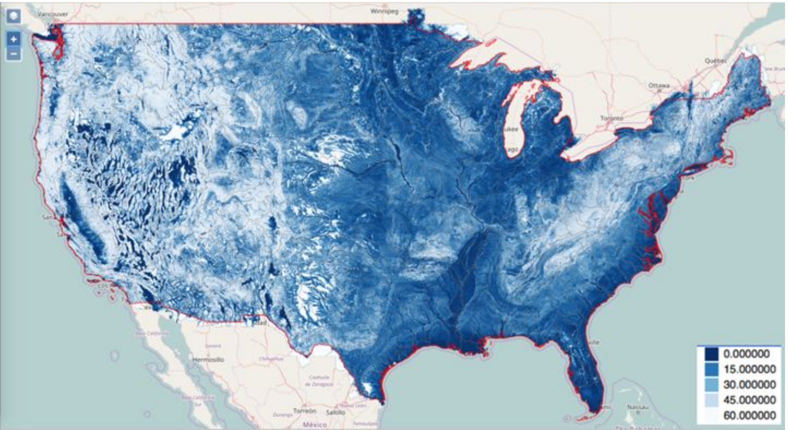


Hu, H., Hong, X., Terstriep, J., Liu, Y., Finn, M., Rush, J., Wendel, J., Wang, S. 2016. "TopoLens: Building A CyberGIS Community Data Service for Enhancing the Usability of High-resolution National Topographic Datasets". *Proceedings of the 2016 Annual Conference on Extreme Science and Engineering Discovery Environment (XSEDE'16)*. July 17-21. Miami, Florida.

Collaborative Research: NFIE

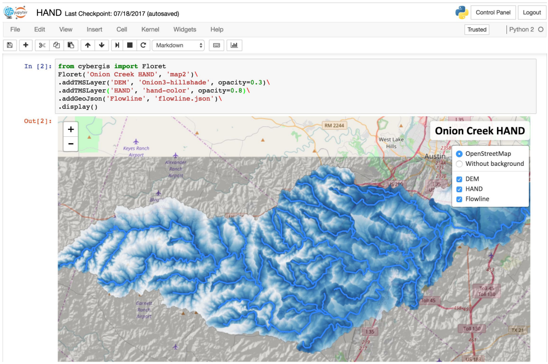
- Build national flood inundation mapping framework for CONUS
  - 10m and higher resolution
  - Near real-time inundation forecast
- Co-development of methodologies
  - Hydrology community, NOAA, USGS, EPA





Containerized Collaboration

Collaboration Requirements	*Current*	*Enhanced*
Methodology development	• Writeups as shared document • Communicated via emails, teleconf	• Methodology notebook – Math formula – Code snippets – Sample results
Software development	• Source codes only; computing environment needs to be maintained and synchronized manually	• Function notebooks with both source codes and computing environments effectively synchronized between researchers
Computation	• Conducted by a dedicated person, a bottleneck	• Notebook interface to workflow computation on ROGER – Everyone can launch
Result validation	• Data: direct download or via iRODS • Validation results: shared document in Google Drive	• Integrated validation notebooks with reproducible input, statistics, and output
Visualization	• Local: download and use desktop GIS. Almost impossible for large outputs • Online: Tile Map Service (TMS); webGIS. Only available for major output data	• Integrated data, code and visualization notebooks – Jupyter visualization libraries – Jupyter map cells – Jupyter IFrame cells

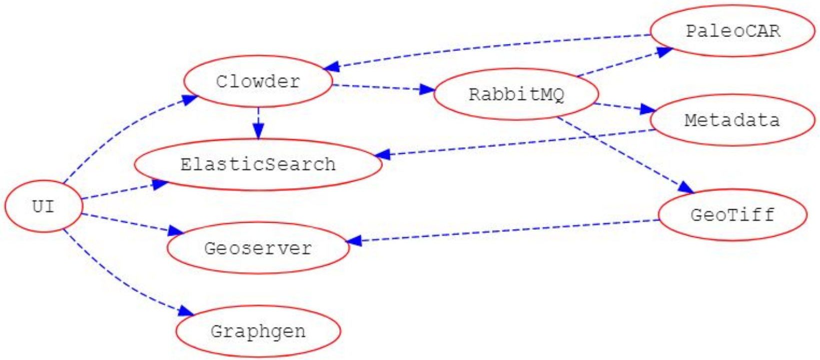


NFIE HAND notebook in CyberGIS-Jupyter

Yin, D., Liu, Y., Padmanabhan, A., Terstriep, J., Rush, J., and Wang, S. 2017. “A CyberGIS-Jupyter Framework for Geospatial Analytics at Scale”. In: *Proceedings of the 2017 Practice & Experience in Advanced Research Computing (PEARC'17)*. July 9–13. New Orleans, LA.

Containerized Applications

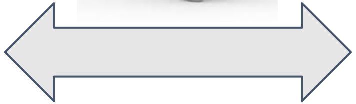
- Simple Services
  - Web Servers
  - RESTful APIs
  - S3 Servers
  - Geoserver Farms
- Microservice Architectures
  - TopoLens
  - TERRA-REF
  - SKOPE
- JupyterHub
  - CyberGIS-Jupyter

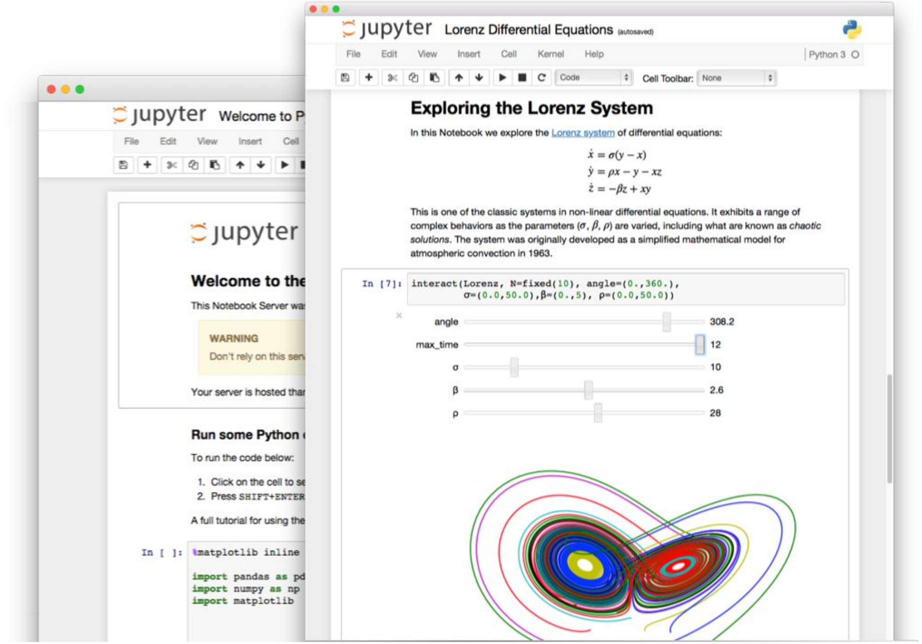


SKOPE Architecture

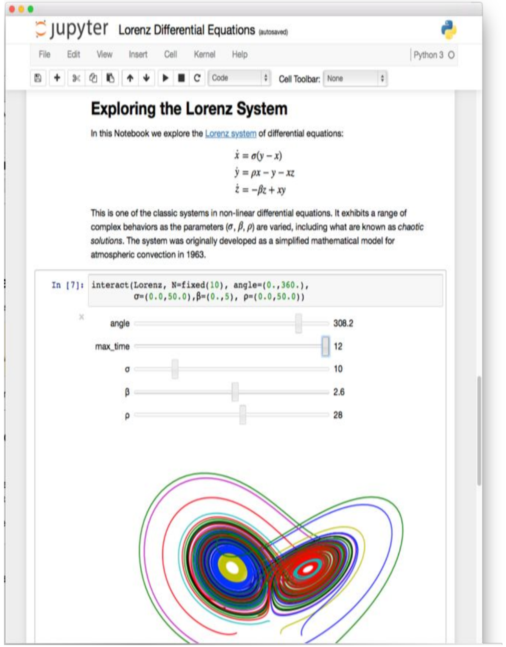
CyberGIS-Jupyter

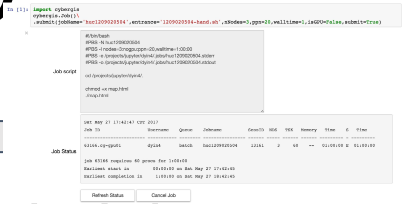
- An interactive scripting interface
- Declarative UI widgets
- Supports general computation and visualization
- Increasingly popular in data- and computational sciences





CyberGIS-Jupyter: Technologies





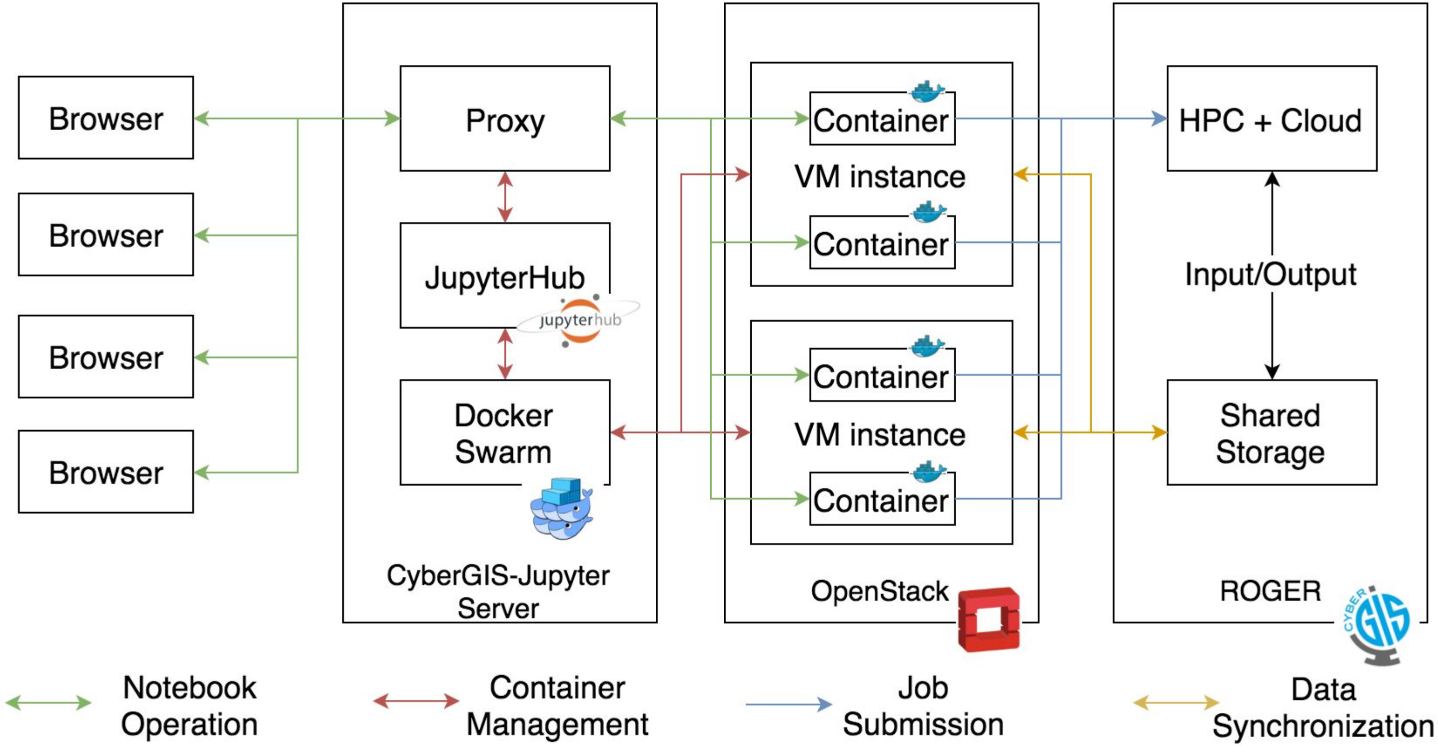
Batch Job management

openstack CLOUD SOFTWARE docker

Cloud-based architecture

Data/storage synchronization

Architecture



On-going Issues

- Deploying VM Infrastructure
  - Ansible
- User Authentication and Authorization
  - PAM with LDAP + SSSD
- Container Orchestration
  - Dynamically scaling and load balancing
  - Recovering idle resources
- Remote Computation Management
  - Using qsub and qstat to control PBS jobs
- Logging and Monitoring
  - ELK stack
- Software and computing environment - containerized
  - Can we use one Dockerfile to populate HPC nodes, OpenStack VMs, and docker instances?

Container Storage

- OpenStack Storage
  - Root disk
  - Cinder volumes
  - GPFS using NFS mounts
- Docker Volume Service
  - Docker-volume-netshare
- NFS auto-exporter
  - Listen to OpenStack service for changes to floating IP and automatically export appropriate directories based on groups

Acknowledgements

- This work is supported in part by the National Science Foundation (NSF) under grant numbers 1047916 and 1443080. The computational work used the NSF-supported ROGER supercomputer (1429699).

- This work is also supported in part by the ECSS program of XSEDE, which is supported by NSF grant number 1053575.