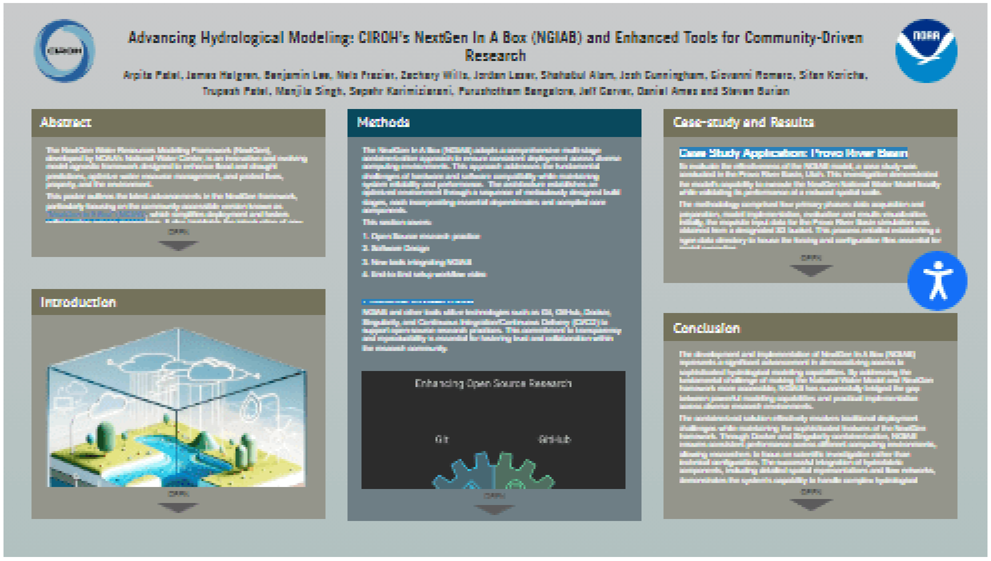
Advancing Hydrological Modeling: CIROH's NextGen In A Box (NGIAB) and Enhanced Tools for Community-Driven Research



Arpita Patel, James Halgren, Benjamin Lee, Nels Frazier, Zachary Wills, Jordan Laser, Shahabul Alam, Josh Cunningham, Giovanni Romero, Sifan Koriche, Trupesh Patel, Manjila Singh, Sepehr Karimiziarani, Purushotham Bangalore, Jeff Carver, Daniel Ames and Steven Burian

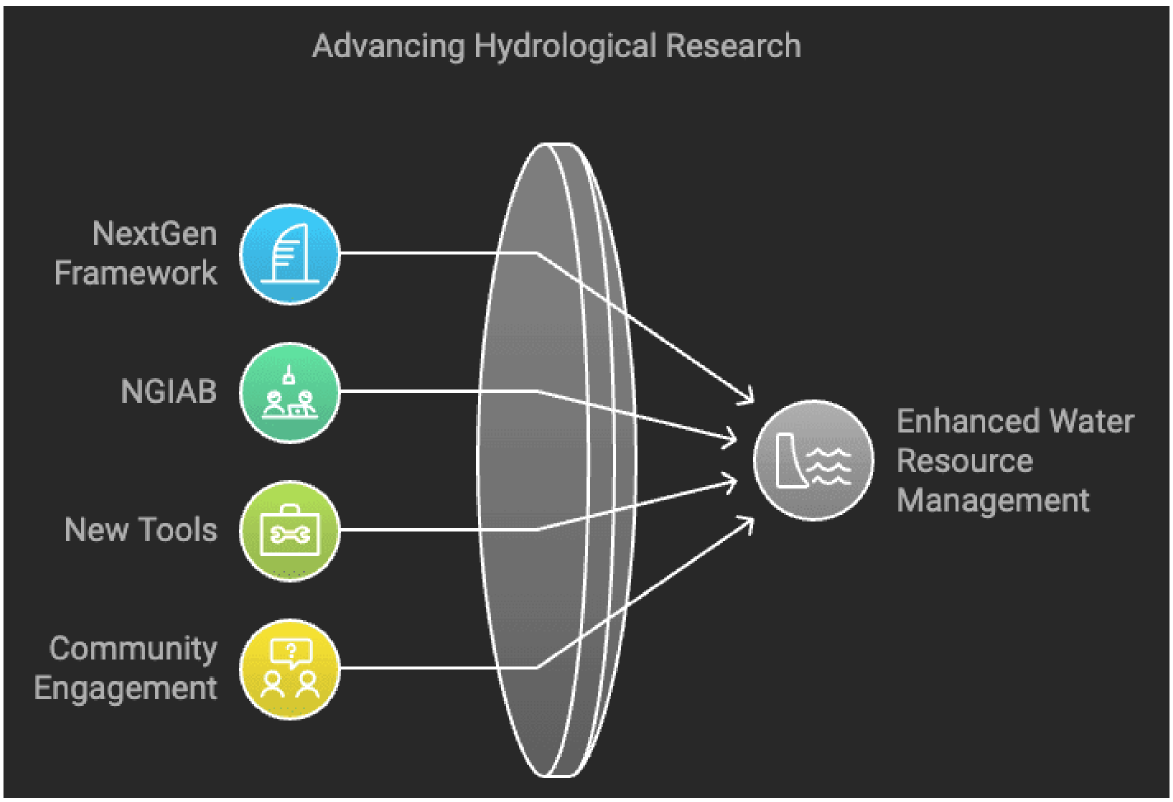
The University of Alabama, Tuscaloosa Alabama, USA Lynker Brigham Young University, Provo, UT

PRESENTED AT:

ABSTRACT

The NextGen Water Resources Modeling Framework (NextGen), developed by NOAA's National Water Center, is an innovative and evolving model-agnostic framework designed to enhance flood and drought predictions, optimize water resource management, and protect lives, property, and the environment.

This poster outlines the latest advancements in the NextGen framework, particularly focusing on the community-accessible version known as "NextGen In A Box" (NGIAB) (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgeninaboxDocker/), which simplifies deployment and fosters collaboration among researchers. It also highlights the introduction of new tools that enhance data preparation, evaluation, visualization and continous datastream, thereby improving the overall efficiency and accessibility of the framework for hydrological research.

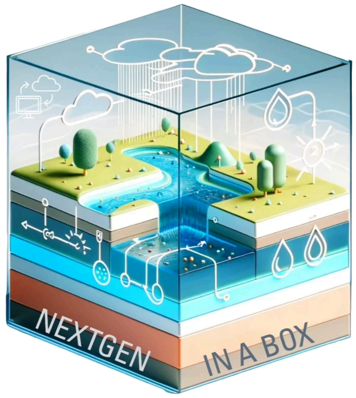


Building on last year's updates, NGIAB now features integration with four major tools: the Data Preprocess, TEEHR, the Data Visualizer and DataStream. The Data Preprocessor, designed with a user-friendly interface simplifies data preparation for NextGen simulations, allowing users to select catchments and date ranges on an interactive map, making data preparation a simple process. TEEHR for evaluating hydrologic model. The Data Visualizer provides geospatial and time series visualization for catchments and nexus points, and empowers researchers with enhanced analysis of hydrological data. DataStream automates the process of collecting data, running NextGen and handling outputs.

These new tools make NWM development and tool integration more efficient, enhancing accessibility and usability for researchers. The project maintains active engagement with the hydrology community through events like the CUAHSI Summer Institute and the CIROH Developers Conference. These platforms provide opportunities for the community to share their feedback and suggestions.

By utilizing tools such as Git, GitHub, Docker, Singularity and Continuous Integration/Continuous Delivery (CI/CD), NGIAB and new tools supports open source research practices, fostering transparent and reproducible research outcomes. The addition of the Data Preprocessor, TEEHR, Data Visualizer and DataStream enhance the framework’s utility and accessibility for the hydrological research community.

INTRODUCTION



Introduction to NextGen Framework

The NextGen framework is continuously evolving to ensure that the NOAA National Water Model (NWM) incorporates the most effective modeling techniques available. This evolution is crucial for improving predictions related to floods and droughts, which are vital for effective water resource management. The framework aims to safeguard not only the environment but also the lives and properties of communities affected by these natural disasters.

The National Water Model (NWM) simulates and forecasts water conditions across the United States, providing guidance for over 3.4 million miles of rivers and streams. The NextGen model engine, built around the *Hy_Features* conceptual model, supports the integration of model formulations using the Basic Model Interface (BMI). To democratize access to these powerful modeling capabilities, we developed NextGen In A Box (NGIAB), an open-source project that encapsulates the NextGen framework within a containerized application.

What is NextGen In A Box (NGIAB)?

The Cooperative Institute for Research to Operations in Hydrology (CIROH) has made significant strides in advancing the NextGen initiative by introducing "NextGen In A Box" (NGIAB). This version is designed to be user-friendly, featuring a single-click containerization process that simplifies deployment. By promoting open-source software development practices, NGIAB encourages collaboration among researchers and enhances the accessibility of the framework.

It is a community-accessible verison of NextGen Water Resources Modeling Framework (NextGen). NGIAB offers both Docker-based image for cloud infrastructure (NGIAB-CloudInfra (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgeninaboxDocker/)) and Singularity-based image for high-performance computing (NGIAB-HPCInfra (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgeninaboxSingularity/)) to resolve traditional deployment challenges, ensuring consistent performance across diverse computing environments while maintaining the framework's sophisticated capabilities.

The system's architecture integrates crucial components including hydrofabric for detailed spatial representation of landscapes and flow networks, along with precise catchments and nexus data that define water flow regions and convergence points. This comprehensive approach enables researchers to focus on scientific inquiry rather than technical setup and maintenance.

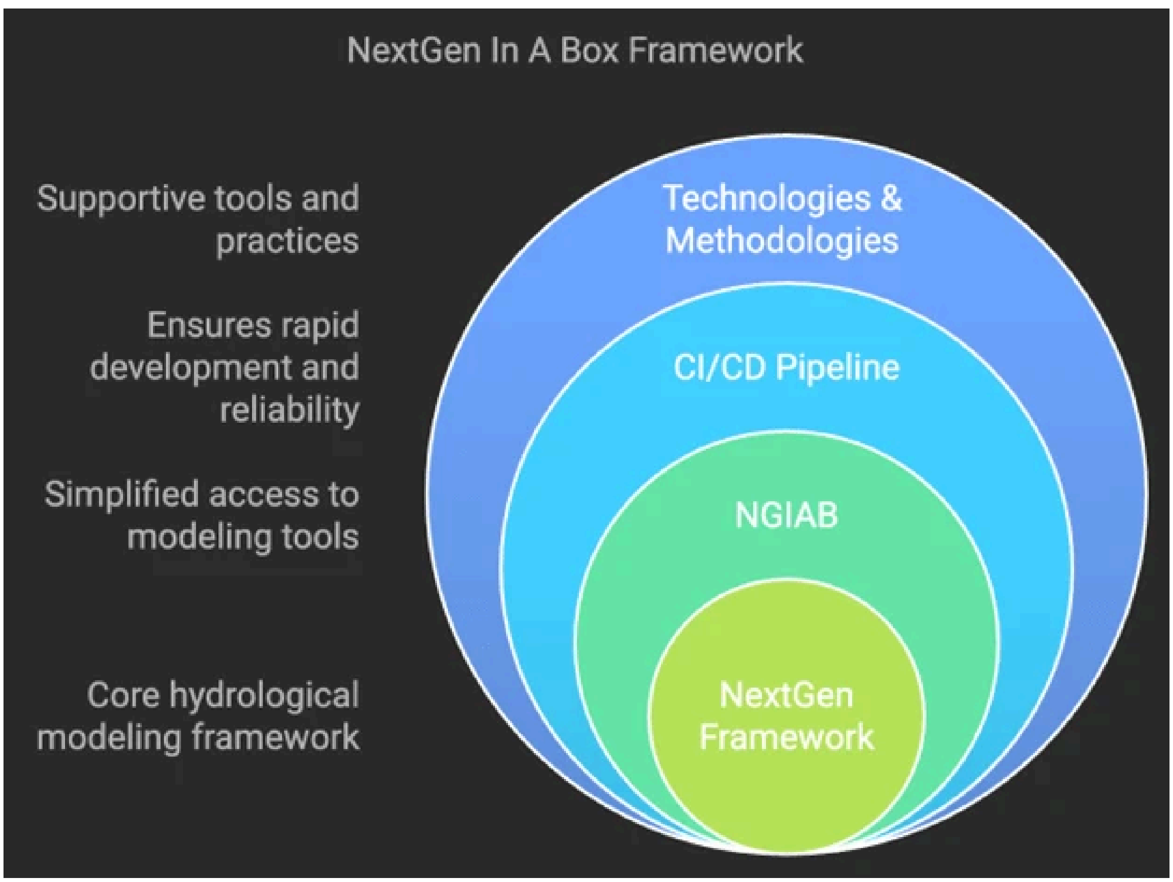
Other tools that are integrated with NGIAB includes Data Preprocess (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/ngiabpreprocessor/), which simplifies data preparation with its interactive map, TEEHR (https://docs.ciroh.org/docs/products/Evaluation%20Tools/rtiteehr/) which provides evaluation capability, Data Visualizer (https://github.com/CIROH-UA/ngiab-client), which provides geospatial and time series visualization and NextGen Datastream (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgenDatastream/) using which you can build and validate NextGen input packages, execute NextGen through NGIAB and version the entire run for reproducibility.

By emphasizing community engagement, NGIAB creates new opportunities for collaboration among researchers, academic institutions, and government agencies. This poster details the methodology behind NGIAB's development, including containerization strategies, deployment protocols, and integration approaches. We present quantitative analyses of its impact on research efficiency and include a case study demonstrating its effectiveness in facilitating research-to-operations and operations-to-research (R2O2R) transitions.

NGIAB GitHub URLs:

https://github.com/CIROH-UA/NGIAB-CloudInfra (https://github.com/CIROH-UA/NGIAB-CloudInfra)

https://github.com/CIROH-UA/NGIAB-HPCInfra (https://github.com/CIROH-UA/NGIAB-HPCInfra)



METHODS

The NextGen In A Box (NGIAB) adopts a comprehensive multi-stage containerization approach to ensure consistent deployment across diverse computing environments. This approach addresses the fundamental challenges of hardware and software compatibility while maintaining system reliability and performance. The architecture establishes an optimized environment through a sequence of meticulously designed build stages, each incorporating essential dependencies and compiled core components.

This section covers:

1. Open Source research practice

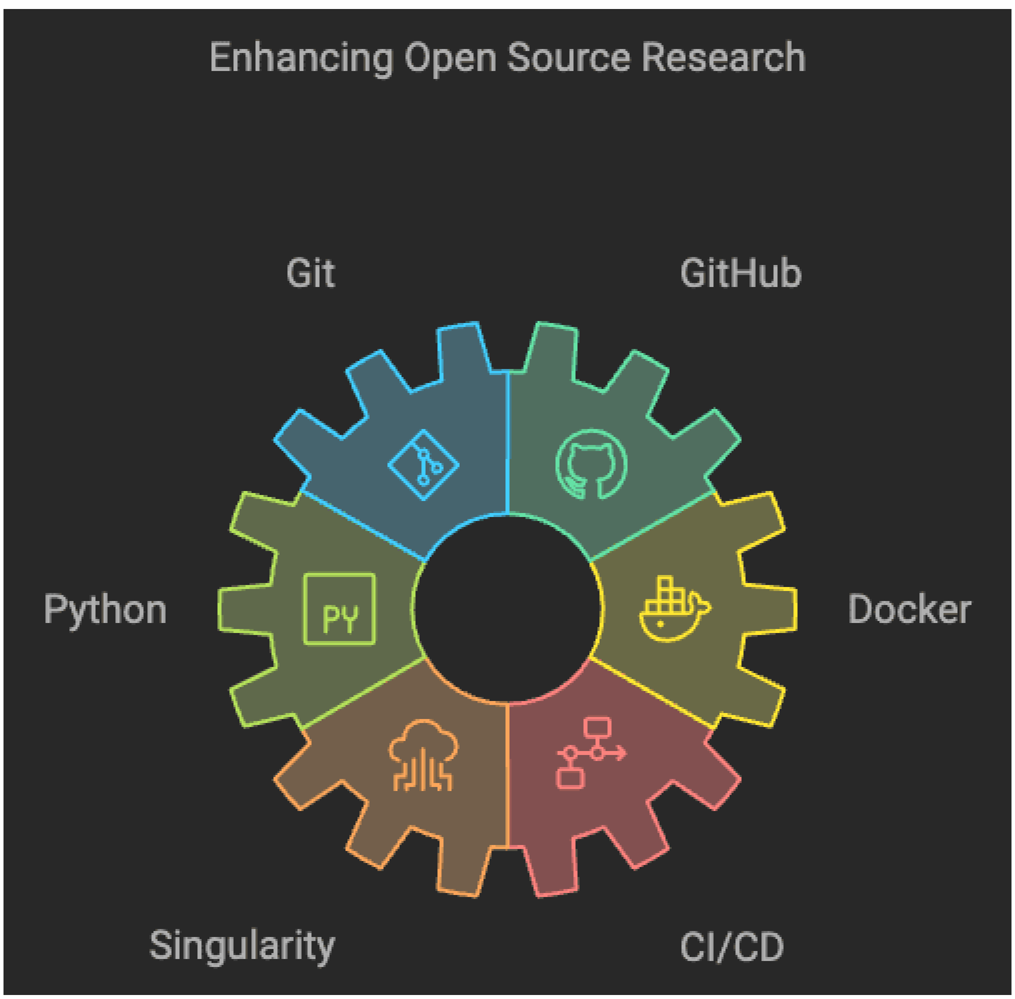
2. Software Design

3. New tools integrating NGIAB

4. End-to-End setup workflow video

1. Supporting Open Source Research Practices

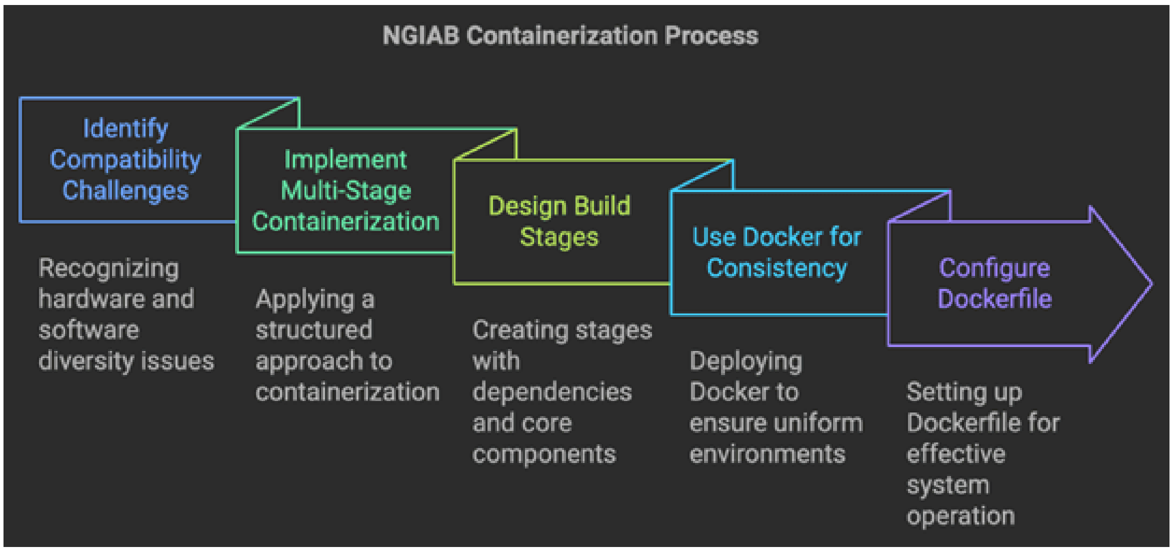
NGIAB and other tools utilize technologies such as Git, GitHub, Docker, Singularity, and Continuous Integration/Continuous Delivery (CI/CD) to support open-source research practices. This commitment to transparency and reproducibility is essential for fostering trust and collaboration within the research community.



2. Software Design – Docker and Singularity based implementation of NGIAB

The NextGen In A Box (NGIAB) adopts a comprehensive multi-stage containerization approach to ensure consistent deployment across diverse computing environments. This approach addresses the fundamental challenges of hardware and software compatibility while maintaining system reliability and performance. The architecture establishes an optimized environment through a sequence of meticulously designed build stages, each incorporating essential dependencies and compiled core components.

Docker, as the primary containerization solution, provides a uniform deployment environment for the NextGen application across heterogeneous systems, effectively mitigating compatibility issues arising from hardware and software variability. This section describes the initial setup, Dockerfile structure, and specific configurations used to ensure the system's effectiveness.



Initial Setup: NGIAB can be set up on a variety of operating systems including Windows, macOS and Linux. The actual steps will vary slightly depending on your operating system of choice. Install Docker Desktop, a widely adopted containerization solution, to facilitate the management of dependencies and configurations required for NextGen applications. When using Windows, take an additional step beforehand and install the Windows Subsystem for Linux (WSL).

Dockerfile Structure: The Dockerfile forms the basis of NGIAB's container environment. Using the FROM instruction, a base image defines the foundation for the container structure. The COPY instruction transfers essential application files into the container. Afterwards, RUN commands install dependencies necessary for NGIAB's functionality. Based on settings such as *NGEN_ACTIVATE_PYTHON*, the system conditionally installs libraries crucial for routing (e.g., pyarrow, pyproj, fiona). This configuration supports customized setups based on user needs.

Component Building and Configuration: Component compilation in NGIAB leverages build commands tailored by environment variables like *NGEN_ACTIVATE_FORTRAN* and *BUILD_CFE*. Each setting incorporates external libraries in a way that allows for flexible configurations to adapt to different computational requirements. For instance, MPI tests conducted use mpirun commands to verify distributed computing performance, which is necessary for parallel processing applications. Optimization: The Docker file includes detailed steps for constructing the NGIAB application in both serial and parallel configurations, directed by flags like *BUILD_NGEN_SERIAL* and *BUILD_NGEN_PARALLE*L. Following compilation, the Docker image undergoes cleanup to reduce size and streamline deployment, a critical step for container efficiency. Layer Optimization and Multi-Stage Builds help manage Docker image layers and separate build dependencies from runtime needs.

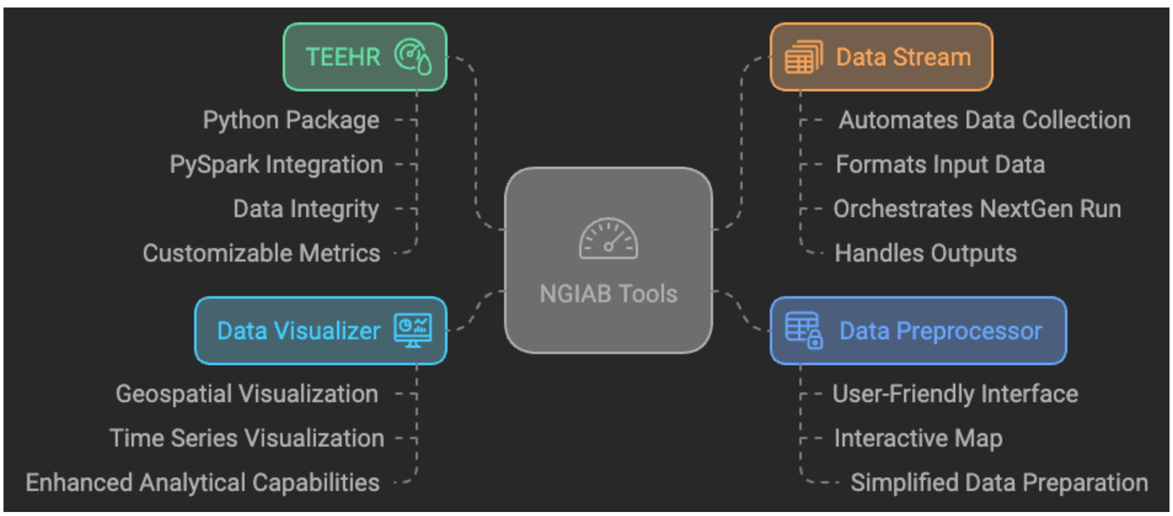
CI/CD: NGIAB implements a sophisticated continuous integration and deployment (CI/CD) pipeline that enables automated cross-platform software delivery. This infrastructure utilizes GitHub Actions to orchestrate a multi-architecture build process, supporting both ARM64 and AMD64 architectures through self-hosted runners managed by AWS Lambda. The system incorporates automated testing, quality assurance measures, and dynamic resource management, ensuring consistent and reliable software deployment across diverse computing environments commonly found in modern building automation systems.

Production releases and feature integrations trigger the pipeline's workflow, initiating a comprehensive process that includes parallel architecture-specific compilations, automated testing suites, and cross-platform compatibility verification. The system employs Docker containerization for consistent builds, generates multi-architecture manifests, and implements automated resource cleanup procedures. This automated approach has demonstrated significant improvements in deployment efficiency, achieving a 40% reduction in deployment time, 60% decrease in manual intervention requirements, and maintaining a 99.9% build success rate across architectures. The implementation provides a robust foundation for scalable building automation systems while optimizing resource utilization through dynamic allocation and automated management procedures.

Challenges and Strategies: Building NGIAB within Docker required managing dependencies effectively to prevent version conflicts. Strategies employed included explicit dependency version control to mitigate compatibility issues and multi-stage builds to separate build and runtime environments, thereby optimizing image size. Frequent testing ensures production environments closely match development, minimizing discrepancies. The forked branches from the ngen and t-route repositories are periodically synchronized with the upstream versions, maintaining build stability in response to continuous updates.

3. New Tools integrating NGIAB

Building on the updates from the previous year, NGIAB now includes four major tools:



a. **Data Preprocessor**
(https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/ngiabpreprocessor/)

The Data Preprocessor (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/ngiabpreprocessor/) is designed with a user-friendly interface that streamlines the data preparation process for NextGen simulations. Users can easily select catchments and date ranges using an interactive map, making data preparation a straightforward task. This tool significantly reduces the complexity often associated with preparing data for hydrological modeling.

b. **TEEHR - Tools for Exploratory Evaluation in Hydrologic Research**
(https://docs.ciroh.org/docs/products/Evaluation%20Tools/rtiteehr/)

TEEHR (https://docs.ciroh.org/docs/products/Evaluation%20Tools/rtiteehr/) is a Python package designed for evaluating hydrologic model performance through iterative and exploratory data analysis. Its key strengths lie in its scalability through PySpark integration, robust data integrity via its internal TEEHR Framework for handling various types of hydrologic data, and flexibility that allows users to customize metrics, implement bootstrapping, and manipulate data through multiple grouping and filtering options.

Please visit AGU Poster (https://agu.confex.com/agu/agu24/meetingapp.cgi/Paper/1622750) for more information about

Datastream.

c. **Data Visualizer** (https://github.com/CIROH-UA/ngiab-client)

The Data Visualizer (https://github.com/CIROH-UA/ngiab-client) complements the Data Preprocessor by providing geospatial and time series visualization capabilities for catchments and nexus points. This tool empowers researchers with enhanced analytical capabilities, allowing for a deeper understanding of hydrological data and its implications.

d. **DataStream** (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgenDatastream/)

Datastream (https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgenDatastream/) automates the process of collecting and formatting input data for NextGen, orchestrating the NextGen run through NextGen In a Box (NGIAB), and handling outputs. This software allows users to run NextGen in an efficient, relatively painless, and reproducible fashion. Please visit AGU Poster (https://agu.confex.com/agu/agu24/meetingapp.cgi/Paper/1589746) for more information about Datastream.

4. NGIAB End-to-End Setup Guide

This video tutorial provides step-by-step instructions for setting up and running the complete NGIAB workflow, including data preprocessor, model execution, TEEHR evaluation and Tethys visualization.

https://shorturl.at/nCXwW (https://shorturl.at/nCXwW)

[VIDEO] https://res.cloudinary.com/amuze-interactive/video/upload/q_auto/v1733511047/agu24/38-63-84-C1-0A-5A-B3-D7-F0-73-6C-FD-B5-5E-FC-0D/Video/NGIAB1.3.0end2endworkflow_bvazjf.mp4

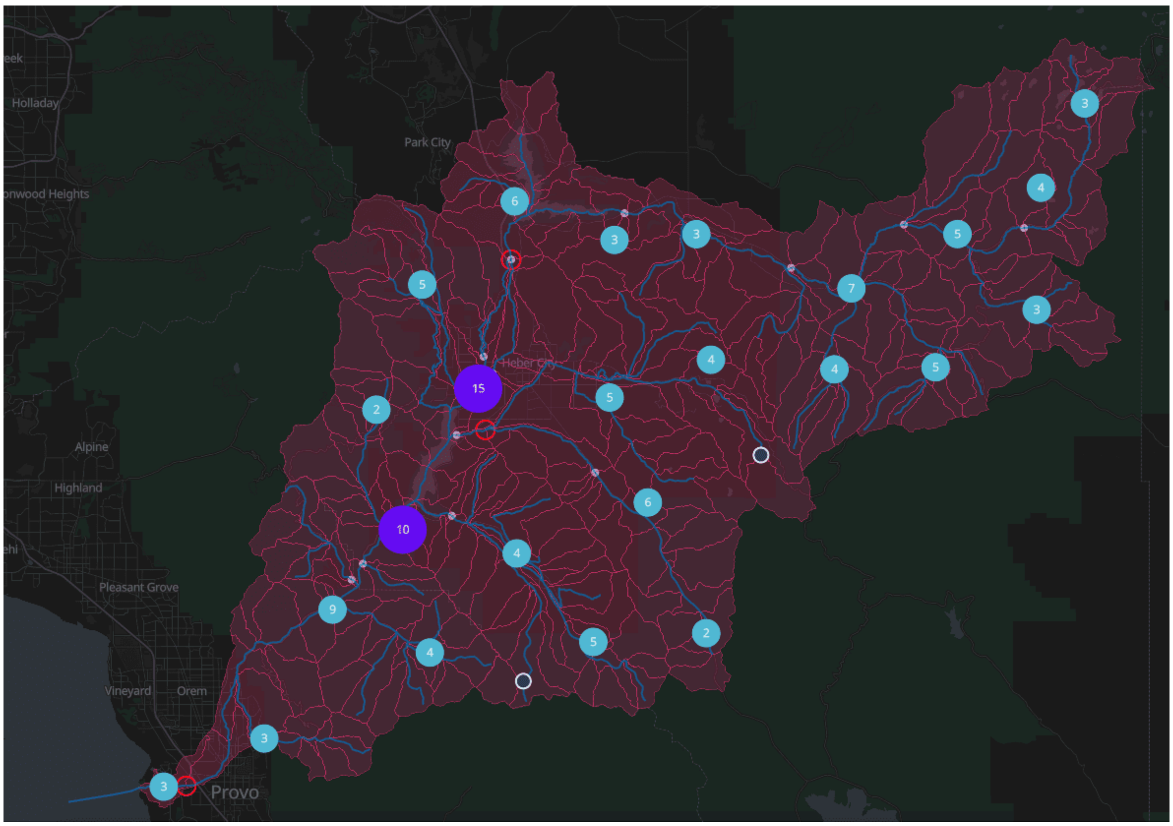
CASE-STUDY AND RESULTS

Case Study Application: Provo River Basin

To evaluate the effectiveness of the NGIAB model, a case study was conducted in the Provo River Basin, Utah. This investigation demonstrated the model's capability to execute the NextGen National Water Model locally while validating its performance at a reduced spatial scale.

The methodology comprised four primary phases: data acquisition and preparation, model implementation, evaluation and results visualization. Initially, the requisite input data for the Provo River Basin simulation was obtained from a designated S3 bucket. This process entailed establishing a ngen-data directory to house the forcing and configuration files essential for model execution.

The implementation phase involved deploying the NGIAB-CloudInfra repository, which provided access to the guide.sh script. This script facilitated model execution by offering user-configurable options for input data pathways and computational modes (serial or parallel processing). The model outputs were systematically stored in the */NextGen/ngen-data/*AWI_16_2863657_007 directory, establishing a foundation for subsequent performance evaluation.



For evaluation, study utilized TEEHR platform and for results analysis, the study utilized the Tethys Platform visualization framework. Through the implementation of the viewonTethys.sh script, a local instance of the Tethys application was deployed. This platform enabled comprehensive visualization of the model outputs through interactive geospatial interfaces and temporal analysis tools. The visualization capabilities facilitated detailed examination of catchments and nexus points throughout the Provo River Basin, enabling thorough validation of hydrological patterns and model accuracy.

Results

The NextGen In A Box (NGIAB) framework demonstrates the effective integration of containerization technology in simplifying the deployment and visualization of complex hydrological models. By utilizing Docker and Singularity, NGIAB addresses the challenges of compatibility across varied hardware and software environments, ensuring consistent performance.

NGIAB deployment (NGIAB v1.3.0. + AWI_16_2863657_007 dataset) was tested across diverse computing environments (local machines, cloud platforms, and HPC) with consistent resource requirements of 4-8 cores and 8-16GB RAM. Setup and deployment times remained efficient across platforms, averaging 5-10 minutes for initial setup and 2-5 minutes each for data preprocessing and container deployment, with minimal variation in performance.

A key feature of NGIAB is TEEHR and Visualizer, which leverages the evaluation and Tethys Platform to provide geospatial and time series visualizations for model outputs. Once a simulation completes, users can initiate the NGIAB Visualizer through the Tethys App, enabling comprehensive insights into hydrological dynamics across catchment areas and nexus points. The following figures showcase visualizations generated by the Visualizer:

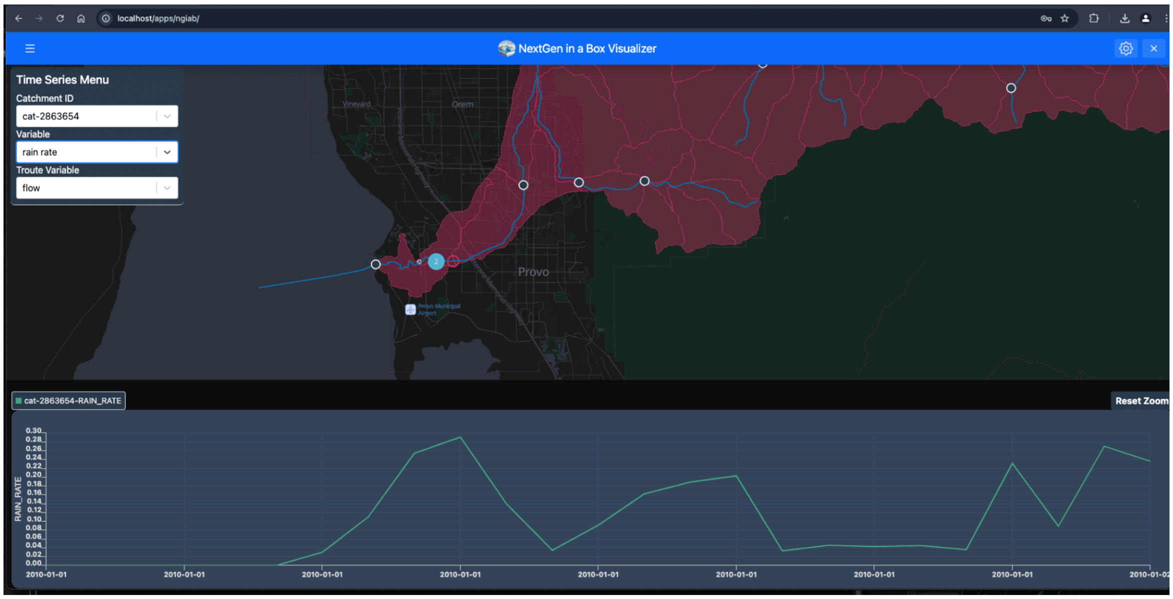


Figure 1: Geospatial Visualization of Catchments

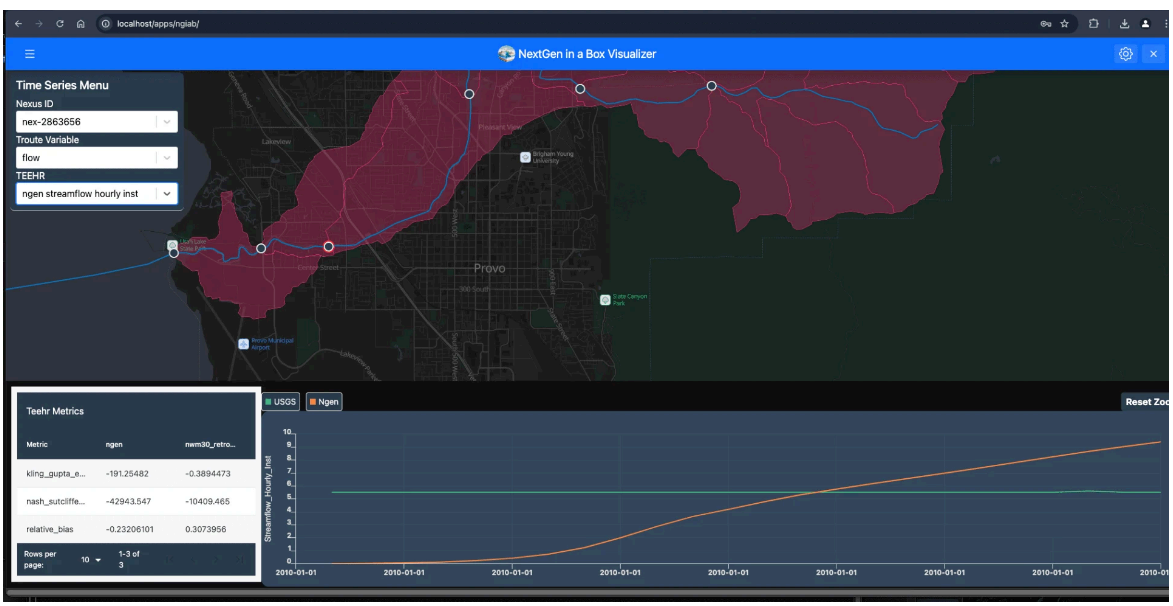


Figure 2: Time Series Visualization of Nexus Points

These visualizations facilitate interactive analysis, allowing researchers to explore model outputs in a detailed spatial and temporal context, thereby validating model accuracy and enhancing interpretability.

ution, TEEHR evaluation and Tethys visualization.

Case-studies with calibrated parameters:

Figure 3 and 4 below illustrates the observed versus simulated streamflow at two case study locations: Wolf Creek near Foley, Alabama, and Provo River sub-basin in Utah, using the CFE 1.0 Model coupled with NoaH-OWP-Modular for the calibration period from 2016 to 2021. The model calibration achieved a high **Kling-Gupta Efficiency (KGE) value exceeding 0.9** at both sites, demonstrating strong agreement between observed and simulated flows. The plots highlight the model's ability to accurately capture streamflow dynamics, particularly during low-flow periods, while showing slight overprediction during high-flow events.

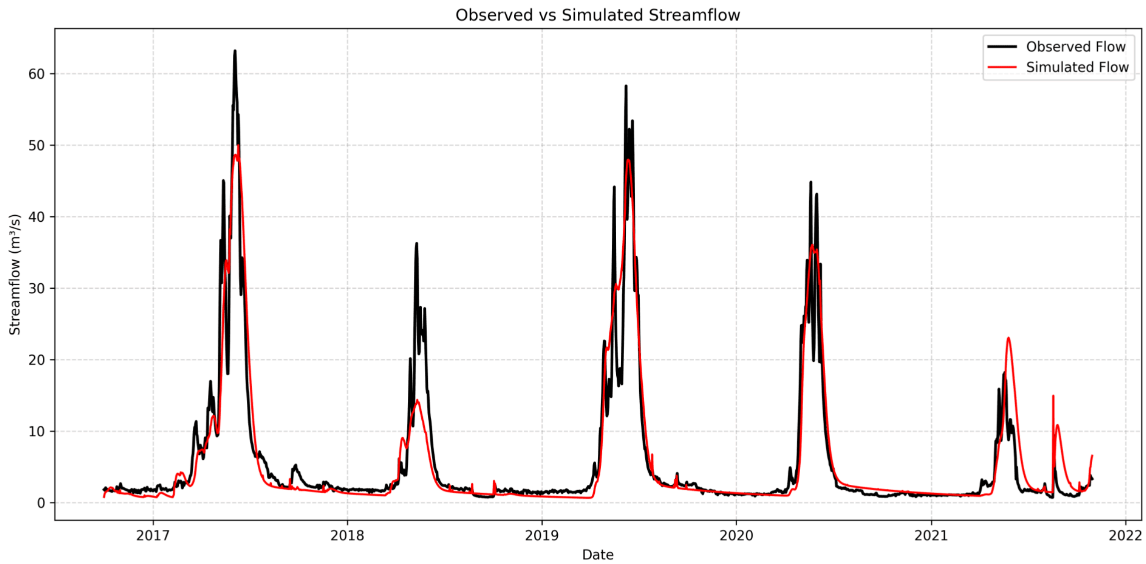


Figure 3: Provo River sub-basin in Utah (KGE 0.9)

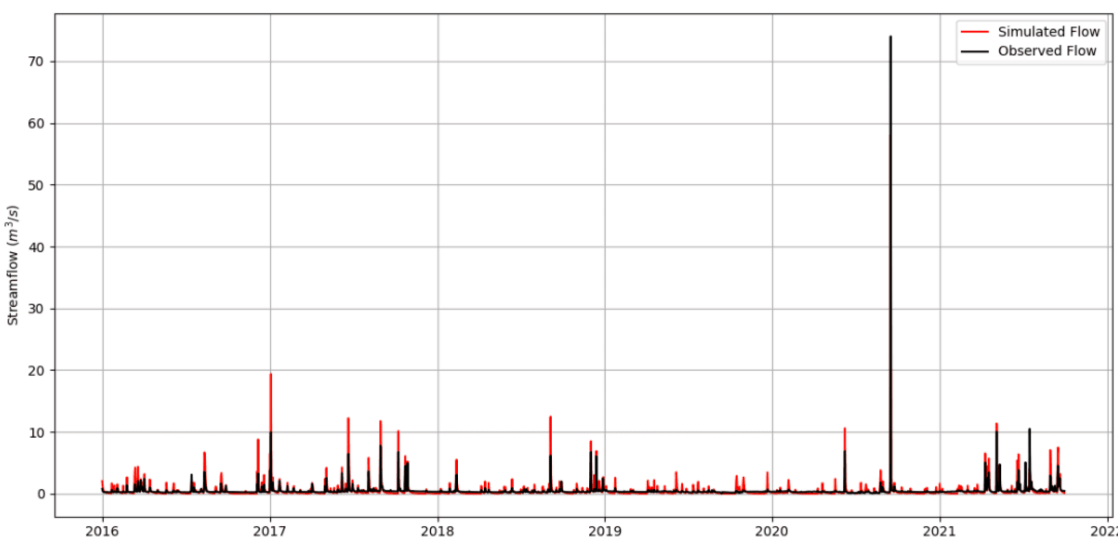
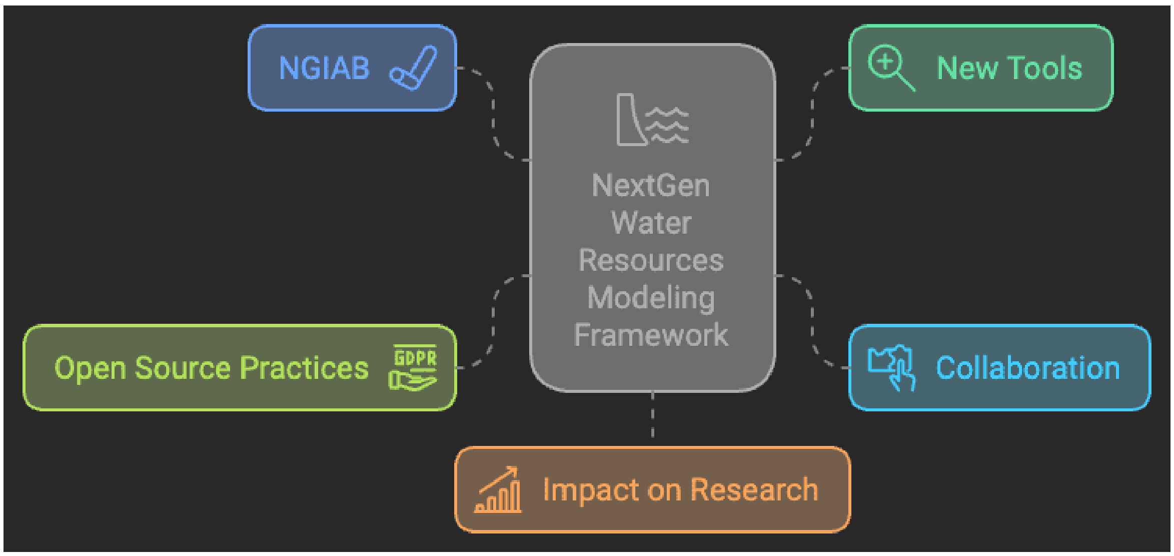


Figure 4: Wolf Creek (02378170) near Foley, Alabama (KGE 0.9)

CONCLUSION

The development and implementation of NextGen In A Box (NGIAB) represents a significant advancement in democratizing access to sophisticated hydrological modeling capabilities. By addressing the fundamental challenge of making the National Water Model and NextGen framework more accessible, NGIAB has successfully bridged the gap between powerful modeling capabilities and practical implementation across diverse research environments.

The containerized solution effectively resolves traditional deployment challenges while maintaining the sophisticated features of the NextGen framework. Through Docker and Singularity containerization, NGIAB ensures consistent performance across different computing environments, allowing researchers to focus on scientific investigation rather than technical configuration. The successful integration of hydrofabric components, including detailed spatial representations and flow networks, demonstrates the system's capability to handle complex hydrological modeling requirements.



Our case study results validate NGIAB's effectiveness in facilitating research-to-operations and operations-to-research transitions, confirming its potential to accelerate hydrological research and collaboration. The system's ability to maintain sophisticated modeling capabilities while simplifying deployment and usage represents a significant step forward in hydrological modeling accessibility.

In-progress and future work includes:

1. Integration of calibration capability with NGIAB.

2. Community interface for sharing calibrated parameters.

3. Expanding the framework to incorporate additional model formulations and physical processes
4. Developing enhanced visualization tools and user interfaces
5. Implementing automated validation and verification protocols
6. Strengthening integration capabilities with existing hydrological databases and systems
7. Exploring cloud-native deployment options for increased scalability
8. Metadata Specifications for data sharing using NGIAB

As the hydrological modeling community continues to grow, NGIAB stands as a testament to the power of open-source collaboration and containerization in advancing scientific research. The framework provides a foundation for future innovations in hydrological modeling while maintaining its core mission of democratizing access to sophisticated modeling capabilities.

Community Engagement and Collaboration

The Community NextGen project actively engages with the hydrology community through various events, such as the CUAHSI Summer Institute and the CIROH Developers Conference. These platforms serve as valuable opportunities for community members to share feedback and suggestions, ensuring that the framework continues to meet the needs of its users.

Acknowledgment

Funding for this project was provided by the National Oceanic and Atmospheric Administration (NOAA), awarded to the Cooperative Institute for Research on Hydrology (CIROH) through the NOAA Cooperative Agreement with The University of Alabama, NA22NWS4320003.

AUTHOR INFO

Arpita Patel - apatel54@ua.edu

https://www.linkedin.com/in/arpita0911patel/

TRANSCRIPT

ABSTRACT

The NextGen Water Resources Modeling Framework (NextGen), a model-agnostic framework developed by NOAA's National Water Center, is continuously evolving. This evolution is aimed at improving flood and drought predictions, optimizing water resource management, and safeguarding lives, property, and the environment. The future versions of the NOAA National Water Model (NWM) will continue to be built using models within the NextGen framework, ensuring the best possible outcomes.

The Cooperative Institute for Research to Operations in Hydrology (CIROH) has advanced this initiative with a community-accessible version known as "NextGen In A Box" (NGIAB). This version simplifies deployment with a single-click containerization process, promoting collaboration and open source software development practices.

Building on last year's updates, NGIAB now features two major tools: the Data Preprocessor and the Data Visualizer. The Data Preprocessor, designed with a user-friendly interface simplifies data preparation for NextGen simulations, allowing users to select catchments and date ranges on an interactive map, making data preparation a simple process.The Data Visualizer provides geospatial and time series visualization for catchments and nexus points, and empowers researchers with enhanced analysis of hydrological data.

These new tools make NWM development and tool integration more efficient, enhancing accessibility and usability for researchers. The project maintains active engagement with the hydrology community through events like the CUAHSI Summer Institute and the CIROH Developers Conference. These platforms provide opportunities for the community to share their feedback and suggestions.

By utilizing tools such as Git, GitHub, Docker, Singularity and Continuous Integration/Continuous Delivery (CI/CD), NGIAB and new tools supports open source research practices, fostering transparent and reproducible research outcomes. The addition of the Data Preprocessor and Data Visualizer enhance the framework’s utility and accessibility for the hydrological research community.

REFERENCES

Advancing Hydrological Modeling: CIROH's NextGen In A Box (NGIAB) and Enhanced Tools for Community-Driven Research

Arpita Patel1, James Halgren2, Nels J Frazier3, Zachary Willis4, Benjamin Lee5, Mr. Jordan Laser, MA6, Md. Shahabul Alam7, Josh Cunningham2, Elkin Giovanni Romero Bustamante8, Trupesh Patel2, Manjila Singh9, Mohammadsepehr Karimiziarani10, Purushotham Bangalore11, Daniel P Ames12 and Steven J Burian5, (1)University of Alabama, Tuscaloosa, AL, United States, (2)University of Alabama, Alabama Water Institute, Tuscaloosa, United States, (3)NOAA Affiliate, Lynker, Boulder, United States, (4)NOAA Affiliate, Lynker, Silver Spring, United States, (5)The University of Alabama, Alabama Water Institute, Tuscaloosa, United States, (6)Lynker Corporation, Thornton, Colorado, UNITED STATES, (7)University of Alabama, Alabama Water Institute, Tuscaloosa, AL, United States, (8)Brigham Young University, Provo, UT, United States, (9)The University of Alabama, Tuscaloosa, United States, (10)The University of Alabama, Center for Complex Hydrosystems Research, Tuscaloosa, AL, United States, (11)University of Alabama, Tuscaloosa,

Arpita Patel, Mohammadsepehr Karmi, Zach Wills, Benjamin Lee, Nels Franzier, James Halgren, Jordan Laser, Steven Burian (2023), H34D-02 NextGen In A Box: Advancing Collaborative Modeling for Enhanced Water Resource Management, presented at AGU23, 11-15 Dec. https://agu.confex.com/agu/fm23/meetingapp.cgi/Paper/1323853

RESIDUAL CONTENT

References

Loading...

EVALUATIONS

#	Average Score
There are currently no completed evaluations for this presentation	

+