

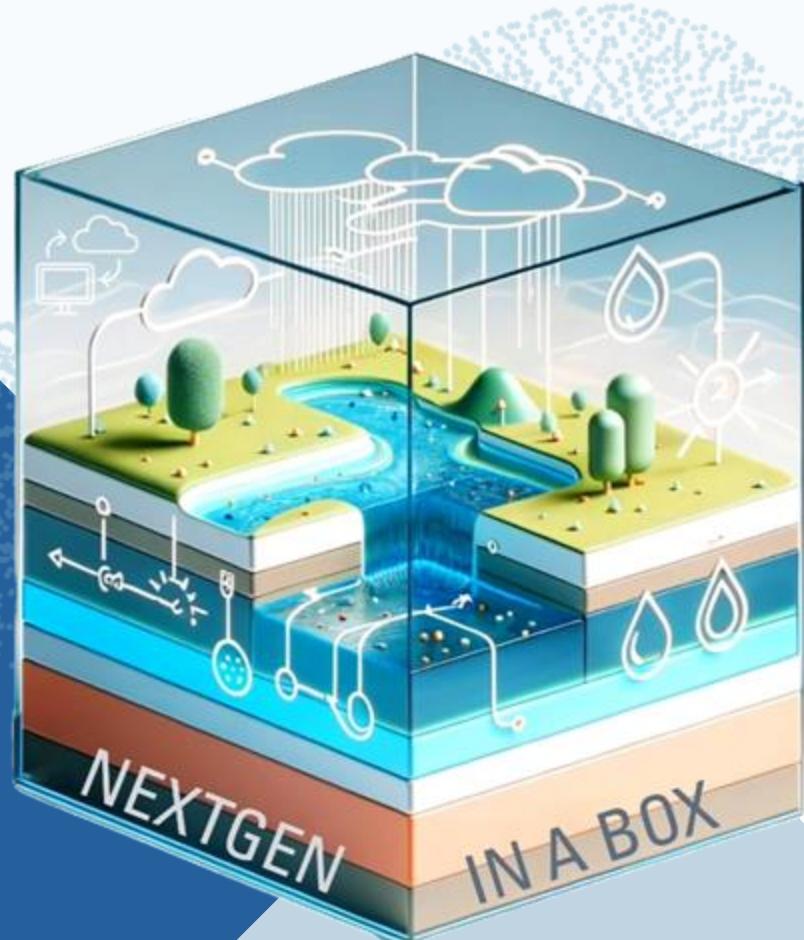


Navigating the NextGen Ecosystem and NextGen In A Box (NGIAB):

Advancing Hydrologic Modeling through
Community-Driven Development

A. Patel, J. Cunningham
The University of Alabama

Z. Wills
Lynker



Agenda

- Introduction
- NextGen EcoSystem
- NGIAB EcoSystem
- Hands-on demo
- Q&A – (10 mins)

DevCon25 NextGen Track Workshop Schedule

Day 1 Session 1 - Navigating the NextGen Ecosystem and NextGen In A Box (**NGIAB**)
(Arpita Patel, Zach Wills, Josh Cunningham)

Day 1 Session 2 - **BMI** Basics for NextGen (Keith Jennings, Chad Perry)

Day 2 Session 1 - Output Visualization through Tethys and evaluation customization using TEEHR (Sam Lamont, Gio)

Day 2 Session 2 - NextGen Calibration Workshop (Sifan Koriche, Shahab Alam, Josh C, Xia Feng)

Day 3 Session 1 - NextGen Research **DataStream**: How to Contribute to Improving NextGen Forecasts (Jordan Laser)

Day 3 Session 2 - Community **HydroFabric** (Mike Johnson)

Principal Investigators



Steve Burian



Jeffrey Carver



Purushotham Bangalore



Arpita Patel

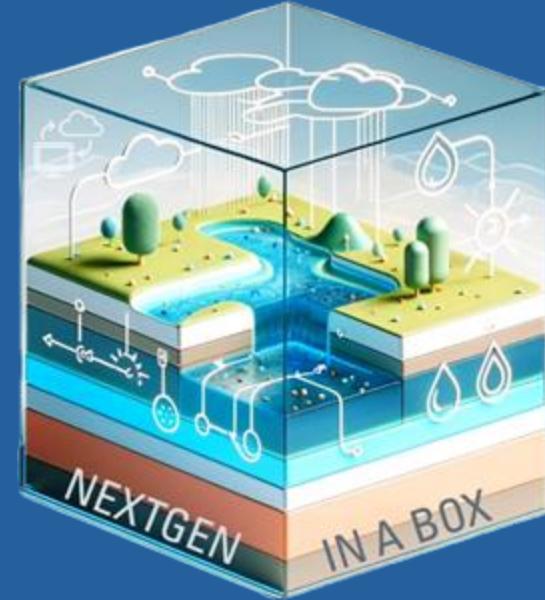


James Halgren

PROJECT	TITLE	PRINCIPAL INVESTIGATORS
P3	Community Water Model Infrastructure, Stewardship, and Integration	Steve Burian , Jeff Carver, Purushotham Bangalore, James Halgren
P29	Community Accessible Development: NextGen Water Resources Modeling Framework in the CIROH Research to Operations Hybrid Cloud	Jeff Carver , Purushotham Bangalore , James Halgren, Arpita Patel
P43	Community NextGen Tools and Frameworks Training and Documentation	Jeff Carver , Purushotham Bangalore, James Halgren, Arpita Patel
P44	Advancing Community NextGen and NextGen In A Box (NGIAB) - Paving the Pathway to Operations	Arpita Patel , Jeff Carver, Purushotham Bangalore, James Halgren

Partners

UA OIT		
2i2c		
Google Cloud		Sponsor
AWS		Sponsor
NSF		
Lynker		



Community NextGen UA Team



Arpita Patel



James Halgren



Benjamin Lee



Josh Cunningham



Trupesh Patel



Shahab Alam



Sifan Koriche



Jonathan Frame



Chad Perry



Xia Feng



Manjila Singh



Nia Minor



Savalan Neisary



Quinn Lee



Hari Jajula

Community NextGen Partner Team



Nels Frazier(Lynker) Zach Wills(Lynker) Keith Jennings (UVM) Jordan Laser(Lynker) Mike Johnson(Lynker)



Josh Sturtevant
(Lynker)

Matthew Denno
(RTI)

Sam Lamont
(RTI)

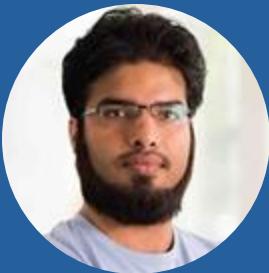
Giovanni Romero
(Aquaveo)

Dan Ames
(BYU)

Community NextGen Partner Team



David Tarboton (USU)



Furqan Baig (USU)



Pabitra Dash (USU)



Ayman Nassar (USU)



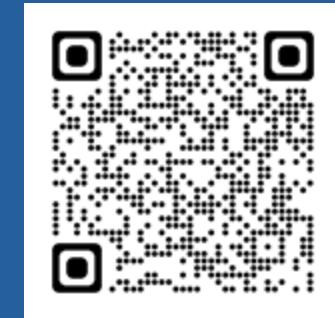
Anthony Castranova
(CUAHSI)



Irene Garousi-Nejad
(CUAHSI)

Next Generation Water Resources Modeling Framework (NextGen)

- NextGen developed by the NOAA OWP at the National Water Center
- A model-agnostic, standards-based framework for water resources modeling that enables seamless coupling and interoperability between different hydrological models and modules.
- Designed to provide multi-model hydrologic prediction advances for the National Water Model (NWM)
- NWM provides hydrologic predictions for **over 2.7 million river reaches across the US**.



NextGen GitHub Repository:
[GitHub: NOAA-OWP/ngen](https://github.com/NOAA-OWP/ngen)

NextGen Wiki:
[GitHub: NOAA-OWP/ngen/wiki](https://github.com/NOAA-OWP/ngen/wiki)

Why NextGen?

Core Features:

- Model Interoperability - Sequential sharing of states/fluxes between models
- Standards-Based - Ensures consistency and compatibility
- Model-Agnostic - Integrates any model regardless of origin
- Multi-Language Support - C++, C, Fortran, Python

NextGen GitHub Repository:

[GitHub: NOAA-OWP/ngen](https://github.com/NOAA-OWP/ngen)

NextGen Wiki:

[GitHub: NOAA-OWP/ngen/wiki](https://github.com/NOAA-OWP/ngen/wiki)

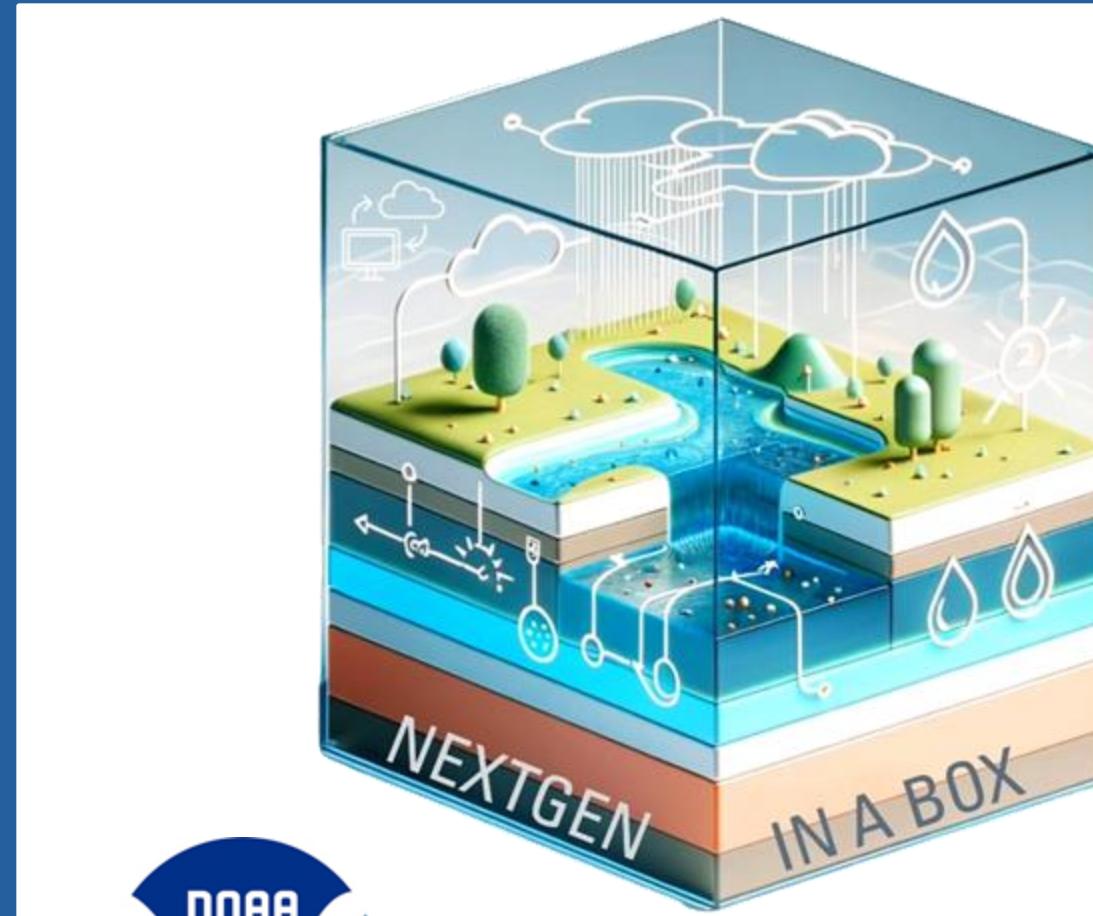
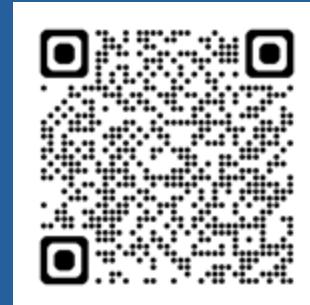


NGIAB (NextGen In A Box)

ready-to-run, containerized, and cloud-friendly version of the NextGen Water Resources Modeling Framework.

Developed by CIROH & Lynker

1. Open-source
2. Community tool
3. Accelerate Research
4. Pathway towards R2O



NGIAB Journal Research Paper (under review)!

The screenshot shows a web-based manuscript tracking system. At the top, there's a header with the logo 'em Environmental Modelling and Software' and a user profile 'Arpita Patel | Logout'. Below the header is a navigation bar with links for 'Home', 'Main Menu', 'Submit a Manuscript', 'About', and 'Help'. The main content area has a title '← Submissions Being Processed for Author ⓘ'. It displays a table with one row, showing the following information:

Action	Manuscript Number	Title	Initial Date Submitted	Status Date	Current Status
Action Links	(Hidden)	NextGen In A Box (NGIAB): Advancing Community Modeling with the U.S. National Water Model	Mar 28, 2025	May 06, 2025	Required Reviews Completed

Below the table, it says 'Page: 1 of 1 (1 total submissions)' and 'Results per page: 10'. There are also 'Action' and 'View' buttons next to the table.

Page: 1 of 1 (1 total submissions)

NextGen In A Box (NGIAB): Advancing Community Modeling with the U.S. National Water Model

Arpita Patel^{a*}, James Halgren^a, Zach Wills^d, Nels Frazier^d, Benjamin Lee^a, Joshua Cunningham^a, Jordan Laser^d, Mohammadsepehr Karimiziaran^{a,b,c}, Trupesh Patel^{a,c}, Giovanni Romero^b, Matthew Denno^f, Sam Lamont^f, Iman Maghami^e, Hari Teja Jajula^c, M. Shahabul Alam^a, Sifan A. Koriche^a, Manjila Singh^c, Savalan Naser Neisary^b, Quinn Lee^a, Steven Burian^{a,b}, Fred L. Ogden^e, Purushotham Bangalore^c, Jeffrey C. Carver^c, Daniel P. Ames^e

NextGen Ecosystem

Introduction to NextGen

The U.S. National Water Model (NWM) provides hydrologic predictions for over 2.7 million river reaches across the United States (Cosgrove et al., 2024). **The Next Generation Water Resources Modeling Framework (NextGen) is an advancement of the NWM**, setting the stage for a more flexible modeling approach. NextGen promotes model interoperability and standardizes data workflows, allowing the integration of various hydrologic models tailored to specific regional processes, providing key flexibility needed for future success with continental-scale modeling. The NextGen framework continues to undergo testing, improvements, and updates through research efforts at the NOAA Cooperative Institute for Research to Operations in Hydrology (CIROH).

NOAA's National Water Model: Advancing operational hydrology through continental-scale modeling

Brian Cosgrove ✉ David Gochis, Trey Flowers, Aubrey Dugger, Fred Ogden, Tom Graziano, Ed Clark, Ryan Cabell, Nick Casiday, Zhengtao Cui, Kelley Eicher, Greg Fall, Xia Feng, Katelyn Fitzgerald, Nels Frazier, Camaron George, Rich Gibbs, Liliana Hernandez, Donald Johnson, Ryan Jones, Logan Karsten, Henok Kefelegn, David Kitzmiller, Haksu Lee, Yuqiong Liu, Hassan Mashriqui, David Mattern, Alyssa McCluskey, James L. McCreight, Rachel McDaniel, Alemayehu Midekisa, Andy Newman, Linlin Pan, Cham Pham, Arezoo RafieeiNasab, Roy Rasmussen, Laura Read, Mehdi Rezaeianzadeh, Fernando Salas, Dina Sang, Kevin Sampson, Tim Schneider, Qi Shi, Gautam Sood, Andy Wood, Wanru Wu, David Yates, Wei Yu, Yongxin Zhang ... See fewer authors ^

First published: 12 January 2024 | <https://doi.org/10.1111/1752-1688.13184> | Citations: 19

Paper No. JAWR-23-0005-P of the *Journal of the American Water Resources Association* (JAWR). Discussions are open until six months from publication: .

NextGen Framework

- The NextGen Water Resources Modeling Framework (NextGen) is developed by **NOAA OWP at the National Water Center**
 - Enhance the forecasting of **flooding and drought**
 - Improve water resource management
 - Protect lives, property, and the environment
 - Multimodel computational framework for **NWM (National Water Model)**
- NextGen GitHub Repository: [GitHub: NOAA-OWP/ngen](https://github.com/NOAA-OWP/ngen)
- NextGen Wiki: [GitHub: NOAA-OWP/ngen/wiki](https://github.com/NOAA-OWP/ngen/wiki)

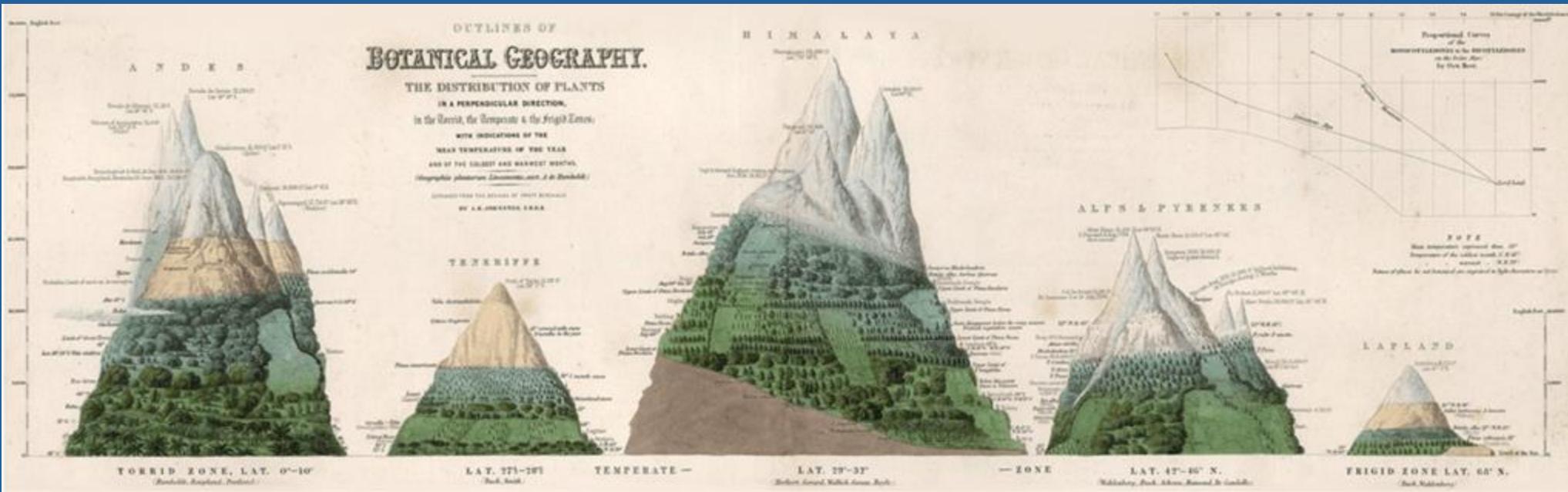


Community NextGen Ecosystem - NGIAB and other extensions!

Modeling with NextGen

- Modeling is a layered concept!
 - Landscapes
 - Relationships
 - Interfaces

Landscape



```
~/Documents/projects/AWI/NGIAB_data_preprocess> open ~/Downloads/conus_nextgen.gpkg | query db "SELECT * FROM gpkg_contents"
```

#	table_name	data_type	identifier	description	last_change	min_x	min_y	max_x	max_y	srs_id
0	flowpaths	features	flowpaths		2024-10-02T21:39:27.356Z	-2355820.00	310953.68	2255727.43	3163261.04	5070
1	divides	features	divides		2024-10-02T21:40:02.814Z	-2356125.00	209715.00	2258235.00	3506235.00	5070
2	lakes	features	lakes		2024-10-02T21:40:03.033Z	-2306232.85	329124.79	2240264.28	3149850.05	5070
3	pois	attributes	pois		2024-10-02T21:40:34.220Z					0
4	hydrolocations	attributes	hydrolocations		2024-10-02T21:40:35.451Z					0
5	flowpath-attributes	attributes	flowpath-attributes		2024-10-02T21:40:43.663Z					0
6	flowpath-attributes-ml	attributes	flowpath-attributes-ml		2024-10-02T21:40:53.358Z					0
7	network	attributes	network		2024-10-02T21:42:24.445Z					0
8	nexus	features	nexus		2024-10-02T21:42:40.611Z	-2355820.00	299145.00	2255727.43	3163920.00	5070
9	divide-attributes	attributes	divide-attributes		2024-10-02T21:43:31.277Z					0

Hydrofabric: Relates the landscape into data

.gpkg file that quantifies the relationship between segments of the landscape

<https://www.lynker-spatial.com>

~/Documents/projects/AWI/NGIAB_data_preprocess> open ~/Downloads/conus_nextgen.gpkg query db "SELECT * FROM gpkg_contents"										
#	table_name	data_type	identifier	description	last_change	min_x	min_y	max_x	max_y	srs_id
0	flowpaths	features	flowpaths		2024-10-02T21:39:27.356Z	-2355828.00	310955.68	2255727.43	3163261.04	5078
1	divides	features	divides		2024-10-02T21:40:02.814Z	-2356125.00	209715.00	2258235.00	3506235.00	5078
2	lakes	features	lakes		2024-10-02T21:40:03.033Z	-2306232.85	329124.79	2240264.20	3149850.05	5078
3	pois	attributes	pois		2024-10-02T21:40:34.228Z					0
4	hydrolocations	attributes	hydrolocations		2024-10-02T21:40:35.451Z					0
5	flowpath-attributes	attributes	flowpath-attributes		2024-10-02T21:40:43.663Z					0
6	flowpath-attributes-ml	attributes	flowpath-attributes-ml		2024-10-02T21:40:53.358Z					0
7	network	attributes	network		2024-10-02T21:42:24.445Z					0
8	nexus	features	nexus		2024-10-02T21:42:40.611Z	-2355828.00	299145.00	2255727.43	3163928.00	5078
9	divide-attributes	attributes	divide-attributes		2024-10-02T21:43:31.277Z					0

Realization file maps the model(s) onto the Hydrofabric

.json file that defines what mathematics to run in each segment, or all segments

```
        "main_output_variable": "z",
        "init_config": "/dev/null",
        "allow_exceed_end_time": true,
        "fixed_time_step": false,
        "model_params": {
            "sloth_ice_fraction_schaake(1,double,m,node)": "0",
            "sloth_ice_fraction_xinanjiang(1,double,1,node)": "0",
            "sloth_soil_moisture_profile(1,double,1,node)": "0"
        },
        "library_file": "/dmod/shared_libs/liblothmodel.so",
        "registration_function": "none"
    }
},
{
    "name": "bmi_fortran",
    "params": {
        "name": "bmi_fortran",
        "model_type_name": "NoahOWP",
        "main_output_variable": "QINSUR",
        "init_config": "config/cat_config/NOAH-OWP-M/noah-owp-modular-init-{{id}}.namelist.input",
        "allow_exceed_end_time": true,
        "fixed_time_step": true,
        "variables_names_map": {
            "PRCPNONC": "precip_rate",
            "Q2": "SPFH_2maboveground",
            "SFCTMP": "TMP_2maboveground",
            "UU": "UGRD_10maboveground",
            "VV": "VGRD_10maboveground",
            "LWDN": "DLWRF_surface",
            "SOLDN": "DSWRF_surface",
            "SFCPRS": "PRES_surface"
        },
        "library_file": "/dmod/shared_libs/libsurfacebmi.so"
    }
},
{
    "name": "bmi_c",
    "params": {
        "name": "bmi_c",
        "model_type_name": "CFE",
        "main_output_variable": "Q_OUT",
        "init_config": "config/cat_config/CFE/CFE_{{id}}.ini",
        "allow_exceed_end_time": true,
        "fixed_time_step": false,
        "variables_names_map": {
            "atmosphere_water_liquid_equivalent_precipitation_rate": "QINSUR",
            "water_potential_evaporation_flux": "EVAPOTRANS",
            "ice_fraction_schaake": "sloth_ice_fraction_schaake",
            "ice_fraction_xinanjiang": "sloth_ice_fraction_xinanjiang",
            "soil_moisture_profile": "sloth_soil_moisture_profile"
        },
        "library_file": "/dmod/shared_libs/libcfbmi.so.1.0.0",
        "registration_function": "register_bmi_cfe"
    }
}
```

Relationships

Each cat-# can get
a “stack” of models

Add models easily
following the same
format

```
        "main_output_variable": "z",
        "init_config": "/dev/null",
        "allow_exceed_end_time": true,
        "fixed_time_step": false,
        "model_params": {
            "sloth_ice_fraction_schaake(1,double,m,node)": "0",
            "sloth_ice_fraction_xinanjiang(1,double,1,node)": "0",
            "sloth_soil_moisture_profile(1,double,1,node)": "0"
        },
        "library_file": "/dmod/shared_libs/libslothmodel.so",
        "registration_function": "none"
    },
    {
        "name": "bmi_fortran",
        "params": {
            "name": "bmi_fortran",
            "model_type_name": "NoahOWP",
            "main_output_variable": "QINSUR",
            "init_config": "config/cat_config/NOAH-OWP-M/noah-owp-modular-init-{{id}}.namelist.input",
            "allow_exceed_end_time": true,
            "fixed_time_step": true,
            "variables_names_map": {
                "PRCPNONC": "precip_rate",
                "Q2": "SPFH_2maboveground",
                "SFCTMP": "TMP_2maboveground",
                "UU": "UGRD_10maboveground",
                "VV": "VGRD_10maboveground",
                "LWDN": "DLWRF_surface",
                "SOLDN": "DSWRF_surface",
                "SFCPRS": "PRES_surface"
            },
            "library_file": "/dmod/shared_libs/libsurfacebmi.so"
        },
        {
            "name": "bmi_c",
            "params": {
                "name": "bmi_c",
                "model_type_name": "CFE",
                "main_output_variable": "Q_OUT",
                "init_config": "config/cat_config/CFE/CFE_{{id}}.ini",
                "allow_exceed_end_time": true,
                "fixed_time_step": false,
                "variables_names_map": {
                    "atmosphere_water_liquid_equivalent_precipitation_rate": "QINSUR",
                    "water_potential_evaporation_flux": "EVAPOTRANS",
                    "ice_fraction_schaake": "sloth_ice_fraction_schaake",
                    "ice_fraction_xinanjiang": "sloth_ice_fraction_xinanjiang",
                    "soil_moisture_profile": "sloth_soil_moisture_profile"
                },
                "library_file": "/dmod/shared_libs/libcfbmi.so.1.0.0",
                "registration_function": "register_bmi_cfe"
            }
        }
    }
}
```

Relationships

Contains variable name maps from one model to the next, and/or Parameter values inline or a file

Runs top → bottom

```
"main_output_variable": "z",
"init_config": "/dev/null",
"allow_exceed_end_time": true,
"fixed_time_step": false,
"model_params": {
    "sloth_ice_fraction_schaake(1,double,m,node)": "0",
    "sloth_ice_fraction_xinanjiang(1,double,1,node)": "0",
    "sloth_soil_moisture_profile(1,double,1,node)": "0"
},
"library_file": "/dmod/shared_libs/libslothmodel.so",
"registration_function": "none"
},
{
    "name": "bmi_fortran",
    "params": {
        "name": "bmi_fortran",
        "model_type_name": "NoahOWP",
        "main_output_variable": "QINSUR",
        "init_config": "config/cat_config/NOAH-OWP-M/noah-owp-modular-init-{{id}}.namelist.input",
        "allow_exceed_end_time": true,
        "fixed_time_step": true,
        "variables_names_map": {
            "PRCPNONC": "precip_rate",
            "Q2": "SPFH_2maboveground",
            "SFCTMP": "TMP_2maboveground",
            "UU": "UGRD_10maboveground",
            "VV": "VGRD_10maboveground",
            "LWDN": "DLWRF_surface",
            "SOLDN": "DSWRF_surface",
            "SFCPRS": "PRES_surface"
        },
        "library_file": "/dmod/shared_libs/libsurfacebmi.so"
    }
},
{
    "name": "bmi_c",
    "params": {
        "name": "bmi_c",
        "model_type_name": "CFE",
        "main_output_variable": "Q_OUT",
        "init_config": "config/cat_config/CFE/CFE_{{id}}.ini",
        "allow_exceed_end_time": true,
        "fixed_time_step": false,
        "variables_names_map": {
            "atmosphere_water_liquid_equivalent_precipitation_rate": "QINSUR",
            "water_potential_evaporation_flux": "EVAPOTRANS",
            "ice_fraction_schaake": "sloth_ice_fraction_schaake",
            "ice_fraction_xinanjiang": "sloth_ice_fraction_xinanjiang",
            "soil_moisture_profile": "sloth_soil_moisture_profile"
        },
        "library_file": "/dmod/shared_libs/libcfbmi.so.1.0.0",
        "registration_function": "register_bmi_cfe"
    }
},
```

Hierarchy of Effect

HydroFabric	Denotes the physical landscape Always changing	Amalgamated Standardized
Model Selection	Denotes physical processes Code, not Data	Easily Testable Region Specific (e.g. snow models)
Parameter Coupling	Ensuring that the correct output of the upper model is coupled to the lower model in a way that makes sense	
Calibration	Parameter values for specific areas are heterogeneous based on what data has been collected and when, and the availability of that data	Not a responsibility of the hydrofabric, which may or may not contain placeholder values for your catchments

Basic Sequence

1

Select
Hydrofabric

2

Select
Models

3

Run and
Evaluate

4

Iterate
through
each layer

Progressive Sequence

1

Select
HydroFabric
from a known
event

2

Run new basic
sequence for
the event
period

3

Evaluate
performance
against
existing
forecast

4

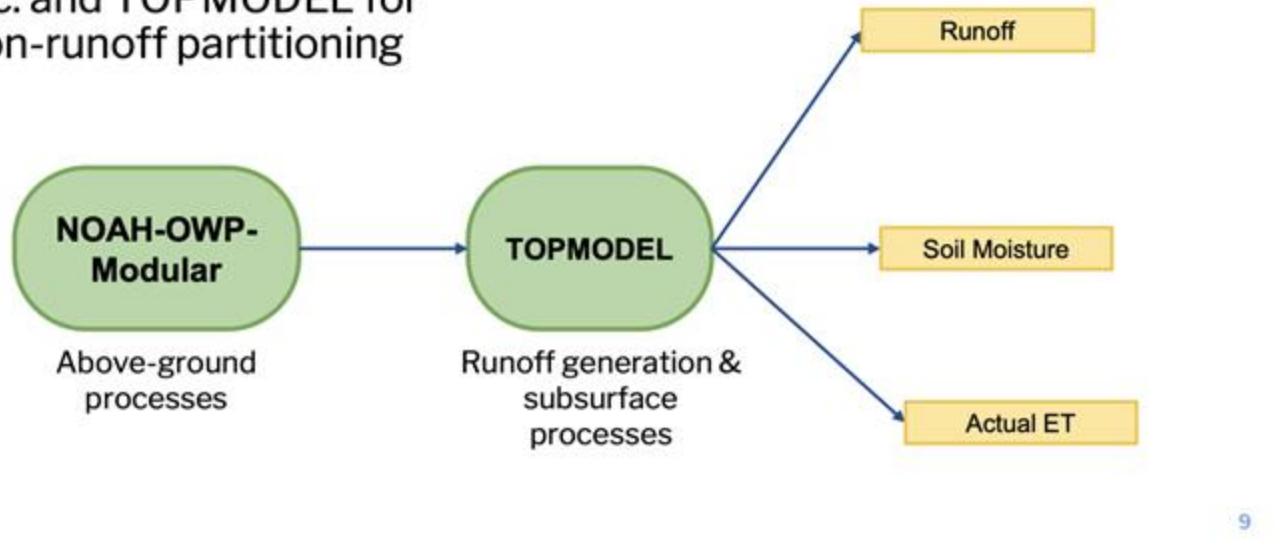
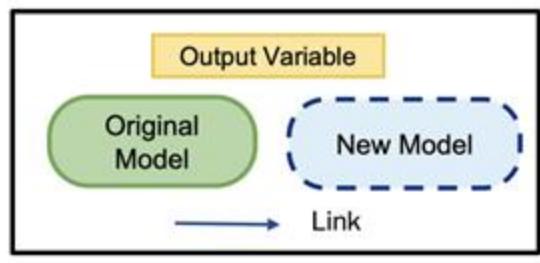
Experiment
with timing,
volume,
runtime (lead
time) etc.

Example Progressive Sequence

Keith Jennings AMS
2024

Example Setup

Mountain basin with Noah-OWP-Modular for interception, snow, etc. and TOPMODEL for infiltration-runoff partitioning



Improving Operational Hydrologic Prediction using Mosaiced Model Formulations with the Next Generation Water Resources Modeling Framework.

Starting point for mountain basin
(more at: <https://github.com/NOAA-OWP/OWP-Presentations>)

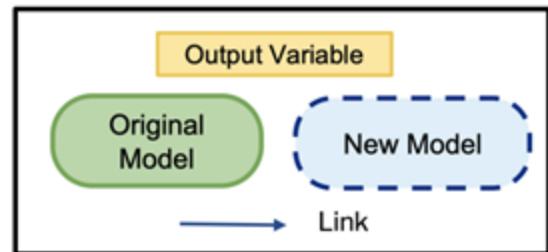
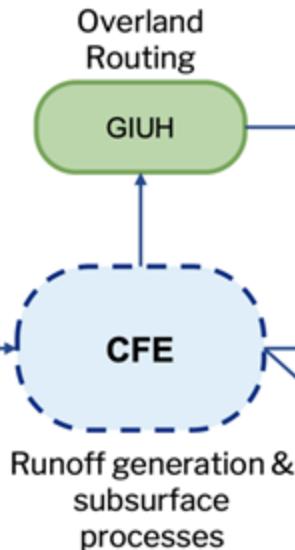
Example Progressive Sequence

Example Setup

Quickly swap out TOPMODEL for CFE to see if performance improves (point NextGen to a new library)

**NOAH-OWP-
Modular**

Above-ground
processes



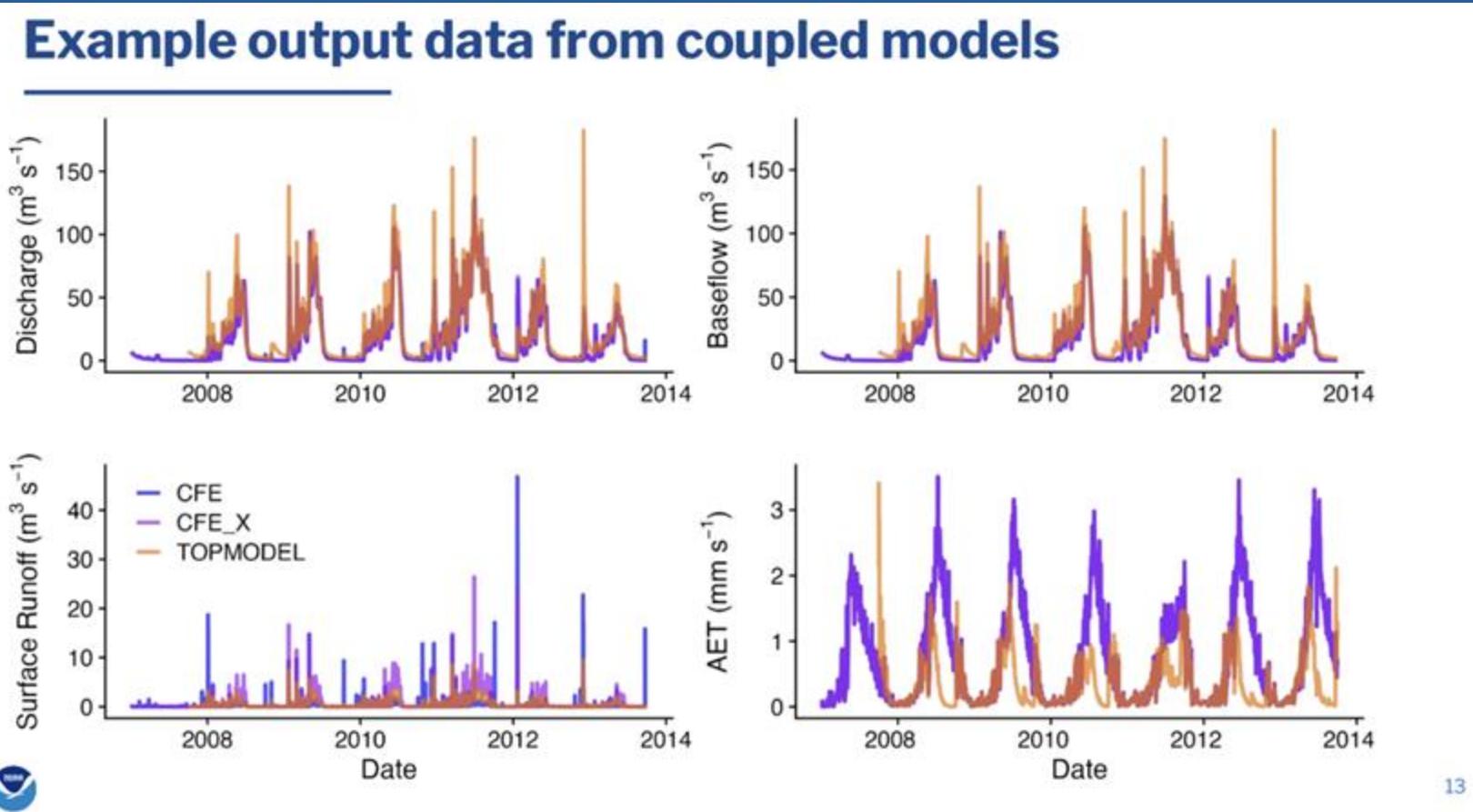
Keith Jennings AMS
2024

Improving Operational Hydrologic Prediction using Mosaiced Model Formulations with the Next Generation Water Resources Modeling Framework.

11

Improved point for mountain basin
(more at: <https://github.com/NOAA-OWP/OWP-Presentations>)

Example Progressive Sequence



Keith Jennings AMS
2024

Improving Operational
Hydrologic Prediction
using Mosaiced Model
Formulations with the
Next Generation Water
Resources Modeling
Framework.

Combinatorial Connections

Combinations with repetition:

If we have N elements out of which we want to choose K elements and it is allowed to choose one element more than once, then number of ways are given by:

$${}^{N+K-1}C_K = \frac{(N + K - 1)!}{(K)!(N - 1)!}$$

In this case we have **N models** out of which we are searching for the correct combination of models aligned to the correct catchment (and later perhaps the best timestep).

So each correct "stack of models" is a **K** elements selected from **N models**.

Iterating and Selecting for the correct models through the realization file should take the sequence of "model, order, coupling, parameters".

Creativity in the sequence

At present, SLoTH does not read any configuration file (an empty string should be provided to `Initialize()`, and it will be ignored--though this feature is expected to be added!). Instead, configuration of tautologies is done by setting values--any variable and value set on the model will become a new output variable of SLoTH.

```
auto s = new Sloth();
double v = 0.0;
s.SetValue("adouble", &v);
// ^ Creates a new, single, type double output variable named 'adouble' that will always be 0.
```

This is especially well suited to take advantage of `ngen's model_param mechanism`, as you can create SLoTH variables directly in the realization config; variables set using `model_params` as if they were configuration settings will become output variables with the set value.

Updating parameters for models from assimilated ground station data would need:

Add config file input to `Initialize()`,

Pipe station data from SLoTH to any parameter in the model stack in the same BMI-coupled pattern

Still runs top->bottom and couples the same way as model -> model

DA example

Realization SLoTH relationship

```
{
  "name": "bmi_c++",
  "params": {
    "model_type_name": "bmi_c++_sloth",
    "library_file": "./extern/sloth/cmake_build/libslothmodel",
    "init_config": "/dev/null",
    "allow_exceed_end_time": true,
    "main_output_variable": "x",
    "uses_forcing_file": false,
    "model_params": {
      "ice_fraction_schaeke(1,double,a,node)": 0.0,
      "ice_fraction_xinanjiang(1,double,none,node)": 0.0,
      "soil_moisture_profile(1,double,none,node)": 0.0
    }
  },
  "name": "bmi_fortran",
  "params": {
    "model_type_name": "bmi_fortran_noahmp",
    "library_file": "./extern/noah-omp-modular/cmake_build/libsurfacedbmi.so",
    "forcing_file": "",
    "init_config": "./data/bmi/fortran/noah-omp-modular-init-{{id}}.manelist.input",
    "allow_exceed_end_time": true,
    "main_output_variable": "QNSUR",
    "variables_names_map": {
      "PAPCPHNO": "atmosphere_water_liquid_equivalent_precipitation_rate",
      "Q2": "atmosphere_air_water-vapor__relative_saturation",
      "SFC TMP": "land_surface_air_temperature",
      "UW": "land_surface_wind_x_component_of_velocity",
      "VW": "land_surface_wind_y_component_of_velocity",
      "LWDN": "land_surface_radiation-incoming-longwave__energy_flux",
      "SLDN": "land_surface_radiation-incoming-shortwave__energy_flux",
      "SFCPRS": "land_surface_air_pressure"
    },
    "uses_forcing_file": false
  }
}
```

Why NextGen In a Box (NGIAB)?

Containerized Solution

- Run anywhere
- **Pre-compiled images** available in Docker Hub

Cloud Friendly Nature

- Reduces the **research time**
- Easily configure **multiple models**
- **Multi-cloud** compatible

Simplifies NextGen Access

- Reduces **learning curve** for NextGen framework
- Compare **model performance**
- Facilitates accessibility and **accelerates modeling**

Collaborative Modeling Tool

- **Easy to use**
- **Reproducible** research outcomes
- Increases **collaboration** among researchers

NGIAB Data Preprocess

This tool prepares data to run a NextGen simulation by creating a run package that can be used with NGIAB.

- Uses Hydrofabric 2.2
- Repo : https://github.com/CIROH-UA/NGIAB_data_preprocess
- Easy Map based tool to create CFE-based NextGen realizations with all necessary inputs.

Commands:

1. uvx --from ngiab_data_preprocess cli -i gage-10154200 -sfr --start 2017-09-01 --end 2018-09-01 --source aorc
2. uvx --from ngiab_data_preprocess map_app



NGIAB's Core Purpose & Benefits

Simplified Access to NextGen

- Lowers the barrier to entry for using the powerful NextGen framework

Accelerated Research & Flexible

- Helps researchers reduce time spent on setup and configuration
- Runs anywhere from local laptop to HPC to Cloud

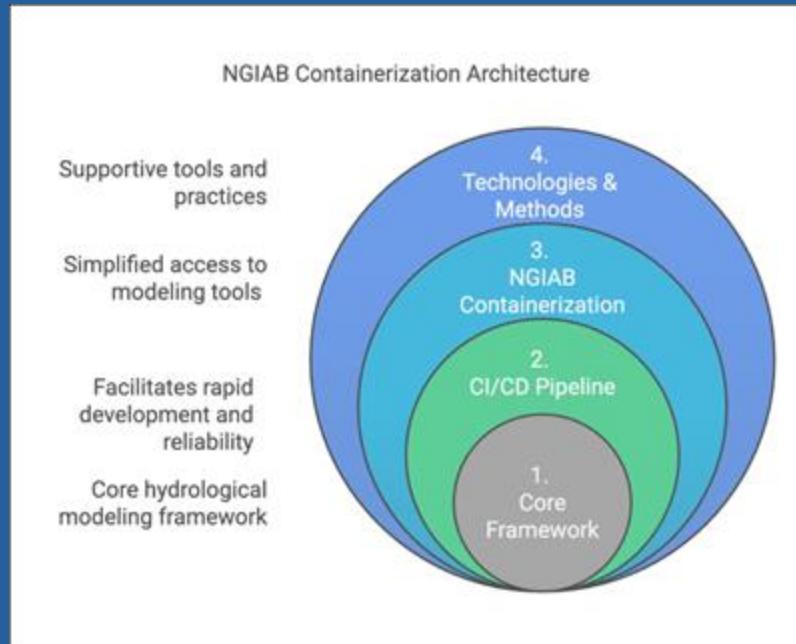
Reproducible Research

- NGIAB promotes reproducible research outcomes through its standardized environment

Collaboration & Community Contribution

- Collaborative modeling tool designed to increase collaboration
- Enables contribution to the National Water Model.

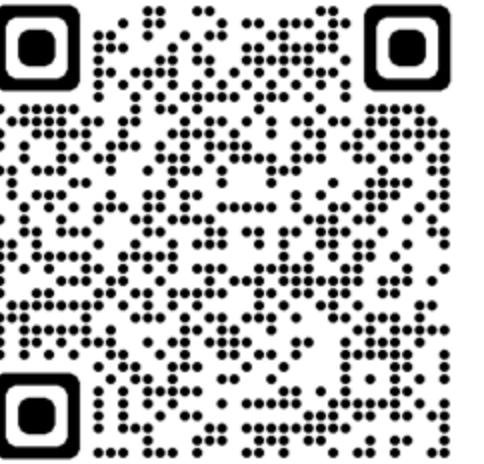
NextGen In a Box (NGIAB)



Provides seamless model coupling and interoperability via the BMI standard within NextGen.

Open-Source Technologies

- GitHub
- GitHub Actions - CI/CD yaml files
- Docker - Dockerfile
- Singularity
- Shell Scripting

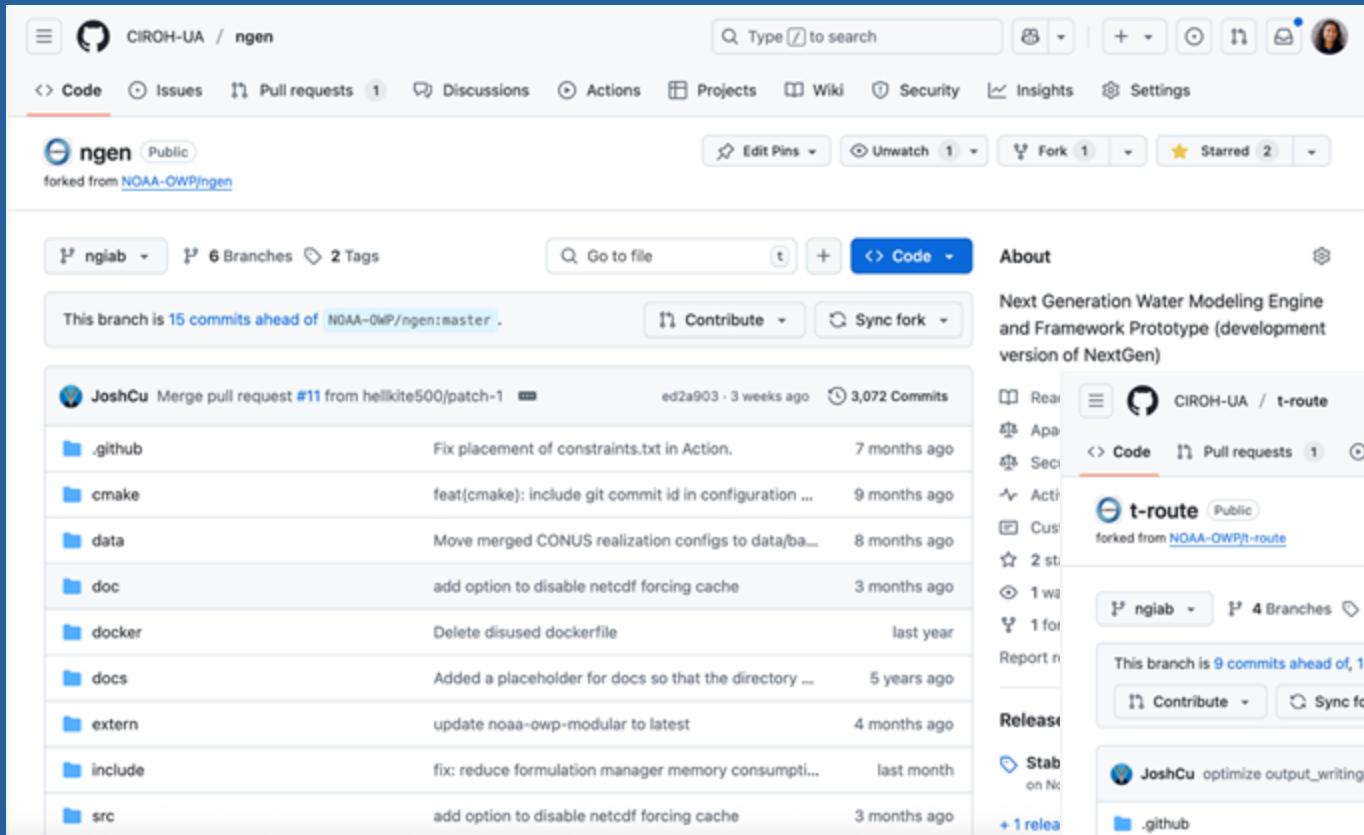


COMMUNITY IMPACT

Join the growing community of researchers using NextGen In A Box



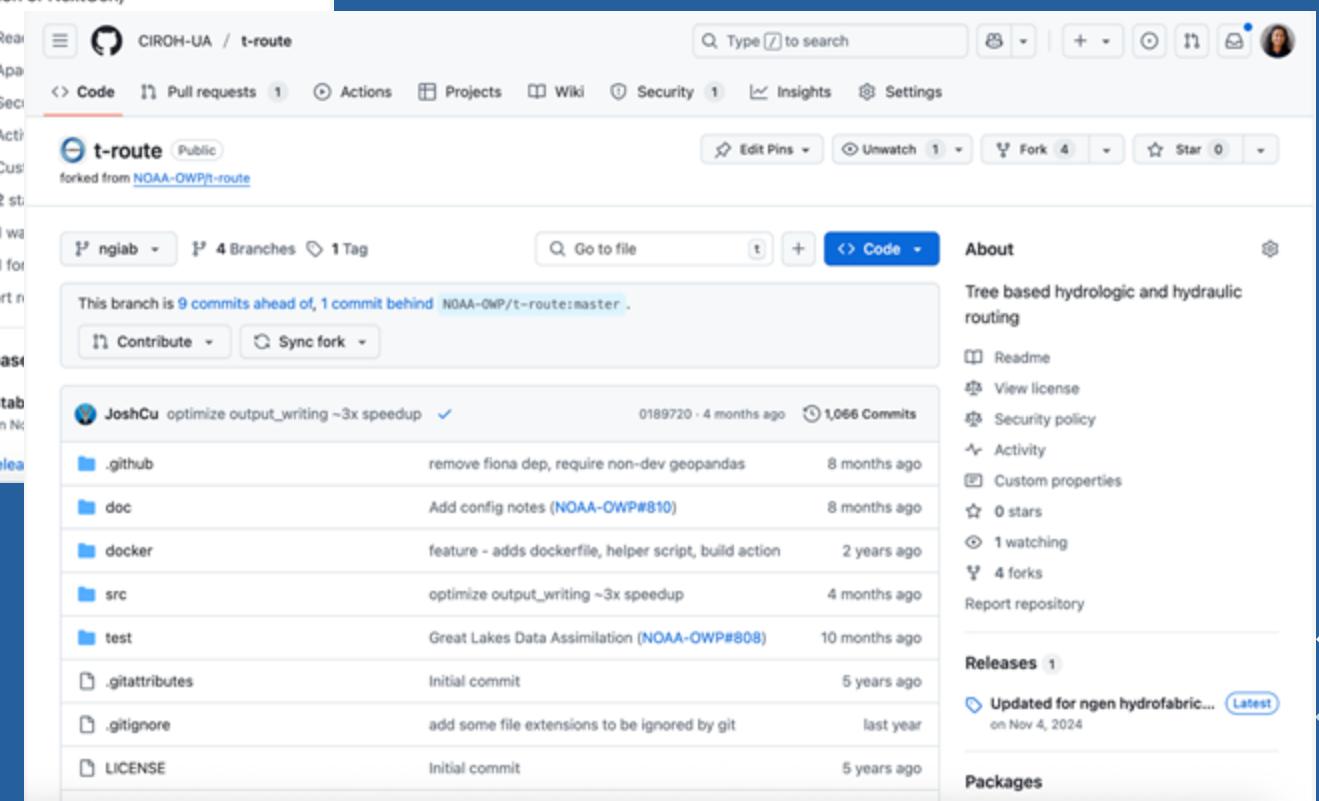
NextGen Community ngen and t-route Forks



This screenshot shows the GitHub repository page for CIROH-UA / ngen. The repository is public and forked from NOAA-OWP/ngen. It has 6 branches and 2 tags. The 'Code' tab is selected. A message at the top indicates that the 'nglab' branch is 15 commits ahead of the 'NOAA-OWP/ngen:master' branch. The repository's purpose is described as the Next Generation Water Modeling Engine and Framework Prototype (development version of NextGen). The commit history shows contributions from JoshCu, including merges from hellkite500/patch-1 and various fixes and improvements across different subdirectories like .github, cmake, data, doc, docker, docs, extern, include, and src.

<https://github.com/CIROH-UA/ngen>

<https://github.com/CIROH-UA/t-route>



This screenshot shows the GitHub repository page for CIROH-UA / t-route. The repository is public and forked from NOAA-OWP/t-route. It has 4 branches and 1 tag. The 'Code' tab is selected. A message at the top indicates that the 'nglab' branch is 9 commits ahead of the 'NOAA-OWP/t-route:master' branch. The repository's purpose is described as Tree based hydrologic and hydraulic routing. The commit history shows contributions from JoshCu, including optimizations for output_writing and updates to Dockerfiles and documentation. The repository also includes releases and packages sections.

Community NextGen related Repos

Community Hydrologic Modeling > GitHub Repository Dashboard

GitHub Repository Dashboard

Introduction Community NextGen related repositories CIROH Workflow Statuses

Community NextGen related repositories

This section contains the list of NGIAB-related repositories that CIROH is maintaining.

- NGIAB-CloudInfra**
NextGen In A Box: NextGen Generation Water Modeling Framework for Community Release (Docker version)
[View Repository](#)
- ngen-datastream**
Infrastructure and configuration for the Research DataStream along with the back-end NextGen workflow/tooling DataStreamCLI.
[View Repository](#)
- NGIAB-HPCInfra**
NextGen In A Box: NextGen Generation Water Modeling Framework for Community Release (Singularity version)
[View Repository](#)
- ngen**
Next Generation Water Modeling Engine and Framework Prototype (development version of NextGen)
[View Repository](#)
- NGIAB_data_preprocess**
Tools to subset hydrofabrics, generate forcings, create default realizations for NGIAB
[View Repository](#)

ngiab-client
NGIAB Data Visualizer
[View Repository](#)

t-route
Tree based hydrologic and hydraulic routing
[View Repository](#)

lstm
[View Repository](#)

ngiab-teehr
A repository for coupling TEEHR with Nextgen In A Box (NGIAB) simulation output
[View Repository](#)

training-NGIAB-101
This repository will be used for preparing training module for NGIAB
[View Repository](#)

ngiab-cal
a tool for calibrating NGIAB compliant data packages
[View Repository](#)

ngiab-website
[View Repository](#)



CIROH
Cooperative Institute for Research
to Operations in Hydrology

NGIAB AWI UA News!

The screenshot shows a news article from the Alabama Water Institute on the University of Alabama's website. The header includes the university's name, a slogan "WHERE LEGENDS ARE MADE*", and navigation links for myBama, search, and a menu. The main content features a large image of a newspaper printing press and the title of the article.

NextGen In A Box (NGIAB): Revolutionizing Hydrological Modeling with a 30-Minute Setup

Written by: Kayla Roberson Published: February 14, 2025

TUSCALOOSA, Ala. — The Cooperative Institute for Research to Operations in Hydrology (CIROH) proudly announces NextGen In A Box (NGIAB), a transformative solution for hydrological modeling. Until now, cutting-



NextGen Standards and Guidelines

The screenshot shows a website navigation bar with links to DocuHub, Products, Services, Community Impact, Learn, Blog, News, Release Notes, and Portal. A sidebar on the left is titled 'Policies and Best Practices' and includes sections for Data and Code Sharing, Policy and Guidance, Recommendations, and NextGen Framework. The main content area features a title 'Data, Code Sharing and Infrastructure Policies' with a subtext about providing practical guidance for CIROH researchers. Below the title is an illustration of a person pointing at a clipboard labeled 'POLICIES' next to a shield icon.



Technical guidance for the inclusion of models/modules in the NextGen Water Resources Modeling Framework

Fred Ogden, Nels Frazier, Keith Jennings, Jonathan Frame, Wouter Knoben, Tadd Bindas, Yalan Song, Irene Garousi-Nejad, Jeffrey Carver, Andy Wood, Anthony Castranova, Arpita Patel, Shahabul Alam, Sifan A. Koriche, Junwei Guo, Cyril Thébault, Raymond J. Spiteri, Ahmad J. Khattak, James Halgren, Patrick J. Clemins, Mukesh Kumar, and Martyn Clark

1. Introduction

NOAA/OWP started development of the "Next Generation Water Resources Modeling Framework" (nicknamed "NextGen") to provide flexible options to experiment with representations of the hydrologic cycle in a model-agnostic framework. The NextGen Framework is a model-agnostic, standards-based, model interoperability software tool that allows explicit coupling of models through the sequential sharing of computed states and/or fluxes between different domain science models or modules. The intent is that NextGen empowers the broader science and applications community to collaborate on water resources modeling problems.

Identification of general design requirements for the NextGen Framework occurred during Interagency meetings held 26-28 October, 2020, involving NOAA, USGS, USACE, USBR, and DOE. These design requirements resulted from discussions around the question "What features would a useful modeling framework possess?" Resulting design requirements included the following:

- Maximum flexibility - as models, data sources, and needs evolve, the framework supports changes and additions
- Model agnostic
- Common architecture to avoid duplication and promote interoperability
- Open source development
 - Promote code reuse and development efficiency
 - Authoritative repository for federal water models
 - Ease/encourage participation by partners and community
- Apply standards where applicable and necessary
 - Coding, coupling
 - Data and metadata
 - Model verification/validation and test data
- Friendly to domain scientists and engineers to facilitate community development
 - Avoid wholesale rewriting of domain science code
 - Encourage and ease adoption
- Sharing models, data, and results
 - Library of model codes and data sets
 - Evaluation tools
- Establish and maintain a glossary and define terms to communicate clearly across disciplinary boundaries
- Use mature open source libraries where appropriate
- Multi-language support (C++, C, Fortran, Python)

NGIAB Product Portfolio website

NextGen In A Box

Revolutionizing Water Modeling

NGIAB

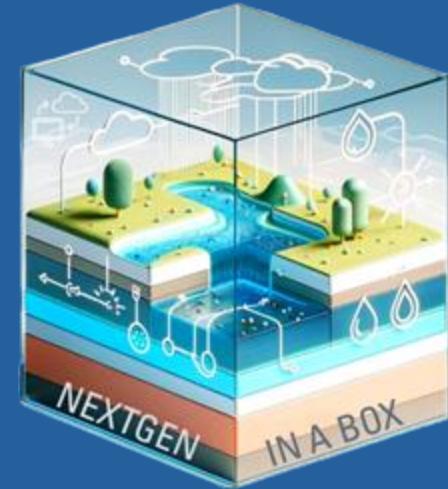
Transforming complex hydrological modeling into accessible solutions

30-Minute Setup

Open Source



ngiab.ciroh.org



The screenshot shows the homepage of the ngiab.ciroh.org website. The header features the CIROH logo and the text "NEXTGEN IN A BOX". Below the header, the tagline "Transforming Hydrology Modeling using open-source technologies" is displayed. At the bottom, there are two buttons: "OUR TOOLS" and "GETTING STARTED".

NGIAB Prerequisites to run locally!

The screenshot shows a GitHub repository page for 'NGIAB-CloudInfra'. The 'Prerequisites' section is highlighted. It contains instructions for Windows, Mac, and Linux. The Windows section includes a command-line code block for installing WSL. The Mac and Linux sections include lists of steps with links to Docker's official website and Mac installer page.

Prerequisites

Windows

1. Install WSL

```
wsl --install
# If the above doesn't work, try:
sudo apt install wsl
```
2. Install Docker Desktop
 - Download from [Docker's official website](#)
 - Launch Docker Desktop
 - Open WSL as Administrator
 - Verify installation: `docker ps -a`

Mac

1. Install Docker Desktop
 - Download from [Docker's Mac installer page](#)
 - Launch Docker Desktop
 - Verify installation: `docker ps -a`

Linux

1. Install Docker
 - Follow [Linux installation guide](#)
 - Start Docker service
 - Verify installation: `docker ps -a`

NGIAB Sample Data - CFE, Noah-OWP, LSTM

The screenshot shows a GitHub repository page for 'CIROH-UA/NGIAB-CloudInfra'. The main content is the 'README' file, which contains instructions for setting up a project directory and downloading sample data for three different hydrological models: SLOTH, Noah-OWP, and CFE.

1. Set Up Project Directory

```
mkdir -p NextGen/ngen-data  
cd NextGen/ngen-data
```

2. Download Sample Data

Option 1: AWI-009 input data (realization file includes - SLOTH, NoahOWP, CFE) - calibrated realization file for Provo River near Woodland, UT

```
wget https://ciroh-ua-ngen-data.s3.us-east-2.amazonaws.com/AWI-009/AWI_16_10154200_009.tar.gz  
tar -xf AWI_16_10154200_009.tar.gz
```

Option 2: AWI-007 input data (realization file includes - SLOTH, NoahOWP, CFE)

```
wget https://ciroh-ua-ngen-data.s3.us-east-2.amazonaws.com/AWI-007/AWI_16_2863657_007.tar.gz  
tar -xf AWI_16_2863657_007.tar.gz
```

Option 3: AWI-008 input data (realization file includes - SLOTH, Demonstration LSTM)

```
wget --no-parent https://ciroh-ua-ngen-data.s3.us-east-2.amazonaws.com/AWI-008/AWI_16_2863806_  
tar -xf AWI_16_2863806_008.tar.gz
```

NGIAB Run

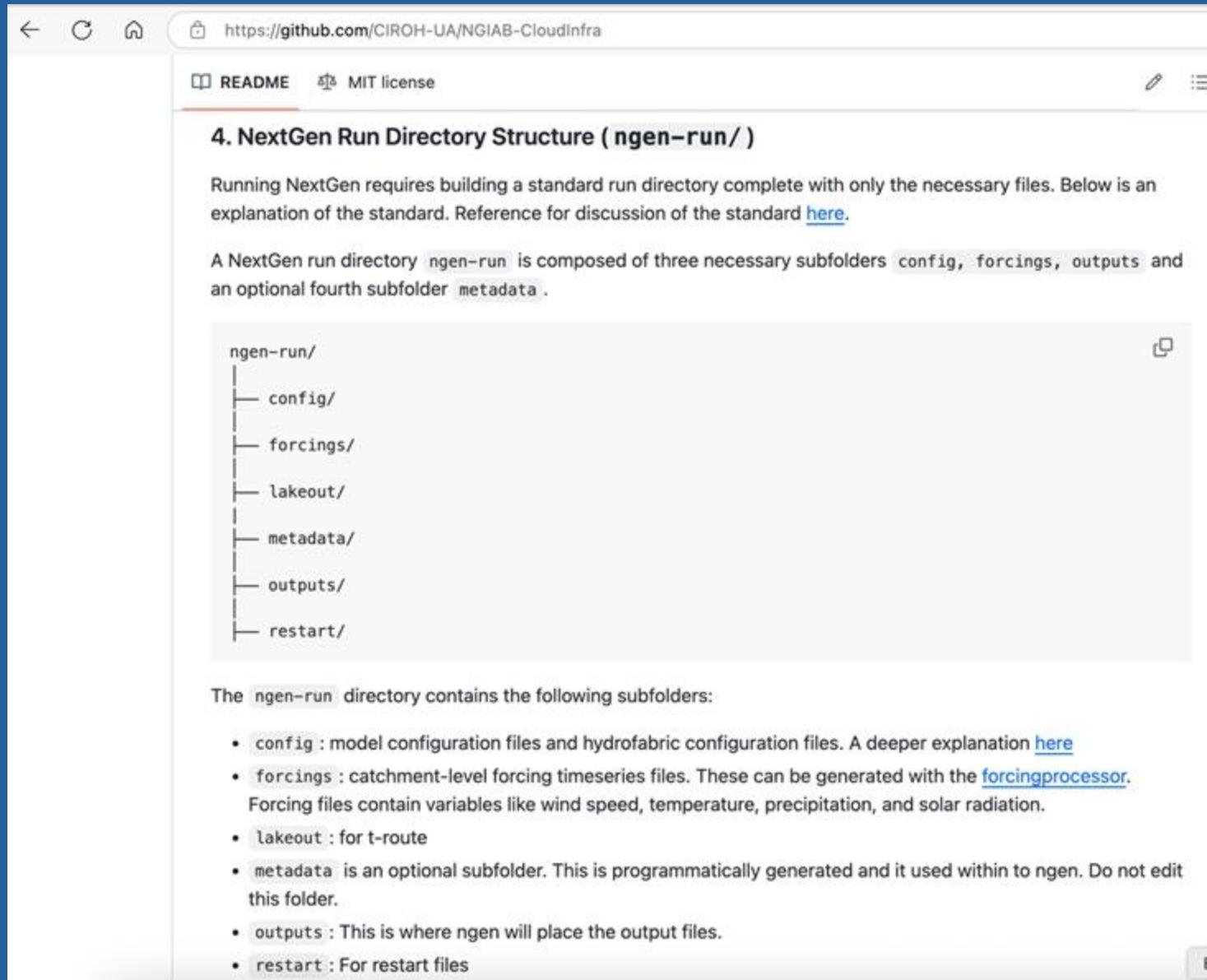
← ⌍ ⌂ https://github.com/CIROH-UA/NGIAB-CloudInfra

📄 README MIT license

3. Clone and Run

```
cd NextGen  
git clone https://github.com/CIROH-UA/NGIAB-CloudInfra.git  
cd NGIAB-CloudInfra  
../guide.sh
```

ngen-run directory structure



The screenshot shows a GitHub README page for the repository "CIROH-UA/NGIAB-CloudInfra". The URL in the address bar is <https://github.com/CIROH-UA/NGIAB-CloudInfra>. The page displays the "README" and "MIT license" sections. Below these, a section titled "4. NextGen Run Directory Structure (ngen-run/)" is shown. It contains text explaining the standard run directory structure and a diagram of the directory tree.

Running NextGen requires building a standard run directory complete with only the necessary files. Below is an explanation of the standard. Reference for discussion of the standard [here](#).

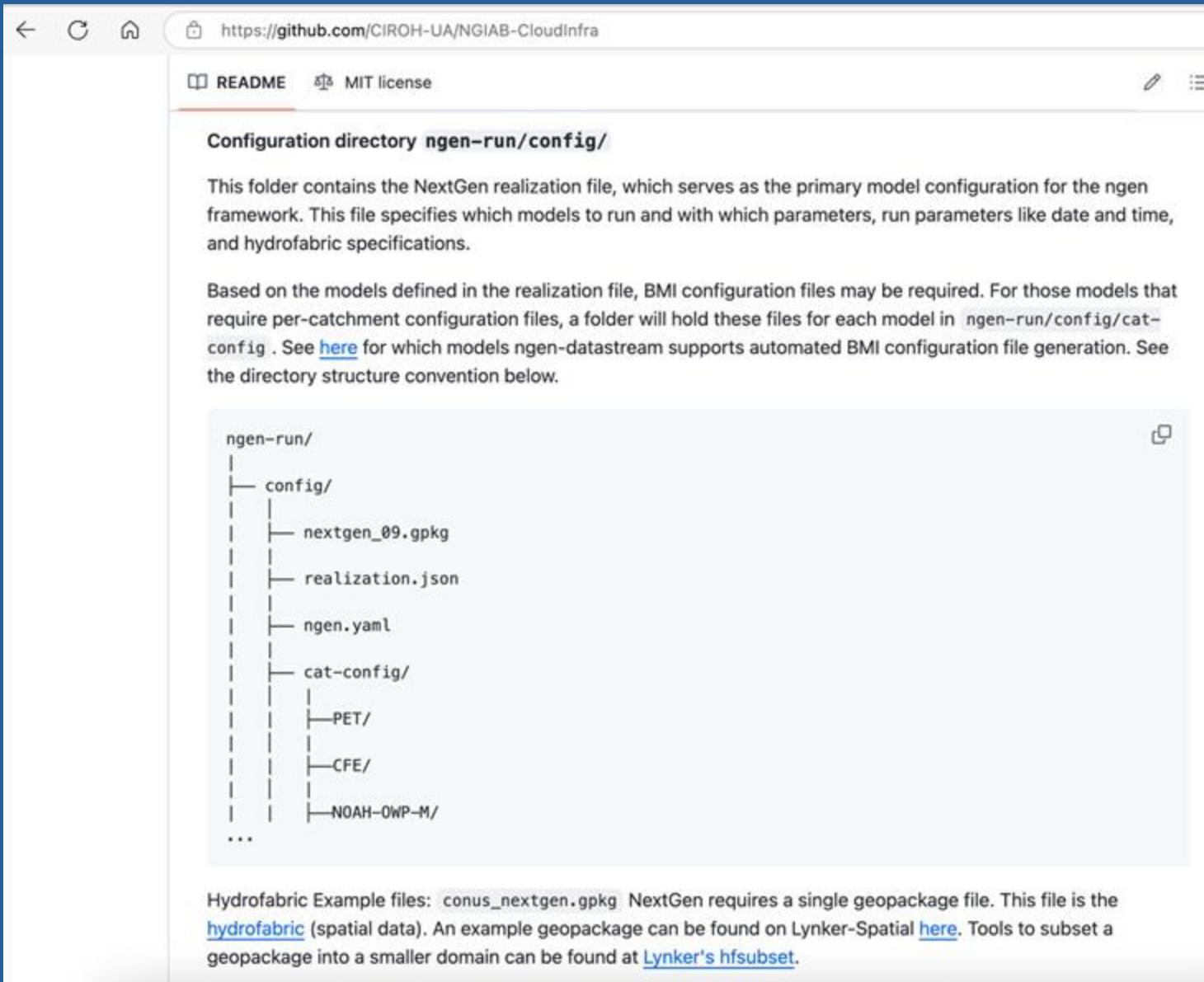
A NextGen run directory `ngen-run` is composed of three necessary subfolders `config`, `forcings`, `outputs` and an optional fourth subfolder `metadata`.

```
ngen-run/
├── config/
├── forcings/
├── lakeout/
└── metadata/
└── outputs/
└── restart/
```

The `ngen-run` directory contains the following subfolders:

- `config` : model configuration files and hydrofabric configuration files. A deeper explanation [here](#)
- `forcings` : catchment-level forcing timeseries files. These can be generated with the [forcingprocessor](#). Forcing files contain variables like wind speed, temperature, precipitation, and solar radiation.
- `lakeout` : for t-route
- `metadata` is an optional subfolder. This is programmatically generated and it used within to ngen. Do not edit this folder.
- `outputs` : This is where ngen will place the output files.
- `restart` : For restart files

ngen-run/config directory structure



The screenshot shows a GitHub README page for the repository "CIROH-UA/NGIAB-CloudInfra". The page title is "Configuration directory ngen-run/config/". The content describes the folder structure and its purpose. It states that the folder contains the NextGen realization file, which serves as the primary model configuration for the ngen framework. This file specifies which models to run and with which parameters, run parameters like date and time, and hydrofabric specifications. It also mentions that based on the models defined in the realization file, BMI configuration files may be required. A folder will hold these files for each model in `ngen-run/config/cat-config`. A link is provided for more information on automated BMI configuration file generation.

Configuration directory `ngen-run/config/`

This folder contains the NextGen realization file, which serves as the primary model configuration for the ngen framework. This file specifies which models to run and with which parameters, run parameters like date and time, and hydrofabric specifications.

Based on the models defined in the realization file, BMI configuration files may be required. For those models that require per-catchment configuration files, a folder will hold these files for each model in `ngen-run/config/cat-config`. See [here](#) for which models ngen-datastream supports automated BMI configuration file generation. See the directory structure convention below.

```
ngen-run/
  config/
    nextgen_09.gpkg
    realization.json
    ngen.yaml
    cat-config/
      PET/
      CFE/
      NOAH-OWP-M/
    ...
  
```

Hydrofabric Example files: `conus_nextgen.gpkg`. NextGen requires a single geopackage file. This file is the [hydrofabric](#) (spatial data). An example geopackage can be found on Lynker-Spatial [here](#). Tools to subset a geopackage into a smaller domain can be found at [Lynker's hfsubset](#).

NGIAB Dockerfile

The screenshot shows a GitHub file viewer for the Dockerfile located at <https://github.com/CIROH-UA/NGIAB-CloudInfra/blob/main/docker/Dockerfile>. The interface includes a sidebar with file navigation and a main area for viewing the code.

Code View: The Dockerfile contains 209 lines of code. The code defines a base image, sets environment variables, installs dependencies, and performs multiple builds for different stages of the Docker image.

```
FROM rockylinux:9.1 AS base
ENV TROUTE_REPO=CIROH-UA/t-route
ENV TROUTE_BRANCH=ngiab
ENV NGEN_REPO=CIROH-UA/ngen
ENV NGEN_BRANCH=ngiab

# Install final dependencies to make sure ngen is build and deployed with matching versions
# Needed here for build caching
RUN echo "max_parallel_downloads=10" >> /etc/dnf/dnf.conf
RUN dnf update -y && \
    dnf install -y epel-release && \
    dnf config-manager --set-enabled crb && \
    dnf install -y \
        vim libgfortran sqlite \
        bzip2 expat udevunits2 zlib \
        mpich hdf5 netcdf netcdf-fortran netcdf-cxx netcdf-cxx4-mpich

FROM base AS build_base
# no dnf update to keep devel packages consistent with versions installed in base
RUN echo "max_parallel_downloads=10" >> /etc/dnf/dnf.conf
RUN dnf install -y epel-release && \
    dnf config-manager --set-enabled crb && \
    dnf install -y \
        sudo gcc gcc-c++ cmake ninja-build tar git gcc-gfortran libgfortran sqlite sqlite-devel \
        python3 python3-devel python3-pip \
        expat-devel flex bison udevunits2-devel zlib-devel \
        wget mpich-devel hdf5-devel netcdf-devel \
        netcdf-fortran-devel netcdf-cxx-devel lld

FROM build_base AS boost_build
```

NGIAB Continuous Integration (CI)

The screenshot shows a GitHub Actions pipeline for the repository `NGIAB-CloudInfra`. The pipeline is triggered via push and has a total duration of 16m 37s. It consists of several jobs:

- start-runner**: 5m 6s
- Matrix: init**:
 - init (AMD64)**: 0s
 - init (ARM64)**: 0s
- Matrix: build**:
 - build (amd64)**: 10m 30s
 - build (arm64)**: 9m 17s
- Matrix: check_result**:
 - check_result (AMD64)**: 0s
 - check_result (ARM64)**: 0s
- create-manifest**

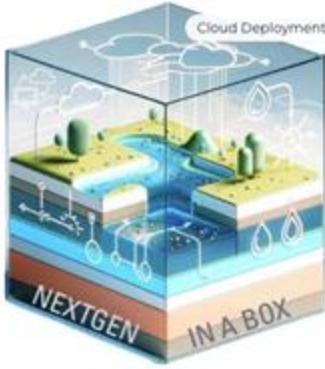
The pipeline also includes a `slack` job and a `stop-runner` job at the end.

NGIAB Product Site ngiab.ciroh.org



NGIAB-DataPreprocess

Simplify and accelerate data preparation for NextGen simulations with our intuitive preprocessing tool.



NGIAB-CloudInfra

Deploy NextGen Framework seamlessly in cloud environments with our containerized solution.



NGIAB-HPCInfra

Scale your hydrological modeling with our high-performance computing infrastructure.



NCIAB-TEEHR

Advanced tools for iterative and exploratory analysis of hydrologic model performance.



NCIAB-Visualizer

Interactive visualization platform for exploring hydrological data and model outputs.



NCEN-DataStream

Streamline your hydrologic simulations with our automated workflow orchestration tool.

NEW NGIAB 101 Training Module!



https://docs.ciroh.org/training-NGIAB-101/instructor/index.html

Pre-Alpha

NGIAB 101 Key Points Instructor Notes Extract All Images More

Search the All in One page

EPISODES

- Summary and Schedule
- 1. Introduction
- 2. Installation and Setup
- 3. Data Preparation
- 4. Model Execution
- 5. Evaluation
- 6. Visualization
- 7. Advanced Topics

Next: Introduction... +

Summary and Schedule

This is a new lesson built with [The Carpentries Workbench](#).

Setup Instructions Download files required for the lesson

00h 00m 1. Introduction What is the NextGen Framework? What is NextGen in a Box (NGIAB)? What is containerization? Why should I use NGIAB?



How NGIAB accelerates research?

The screenshot shows a web browser displaying a blog post from the CIROH DocuHub. The title of the post is "δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling". The post is dated May 16, 2025, and is described as a 2-minute read. It features five authors: Yalan Song, Leo Lonzarich, Arpita Patel, James Halgren, and a photo of a woman. The post includes two figures: (a) a line graph comparing model efficiency (NSE) and (b) a map showing hydrological modeling results. The sidebar on the left lists recent news items from 2025 and 2024.

δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling

May 16, 2025 - 2 min read

Yalan Song, Research Assistant Professor; Leo Lonzarich, Graduate Researcher; Arpita Patel, DevOps Manager and Enterprise Architect; James Halgren, Assistant Director of Science

2025

- δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling
- Google Cloud Next 2025: Innovation at Scale
- UA's Alabama Water Institute Showcases 30-Minute Hydrological Modeling Revolution
- Pennsylvania State University Researchers Leverage CIROH Cyberinfrastructure for Advanced Hydrological Modeling

2024

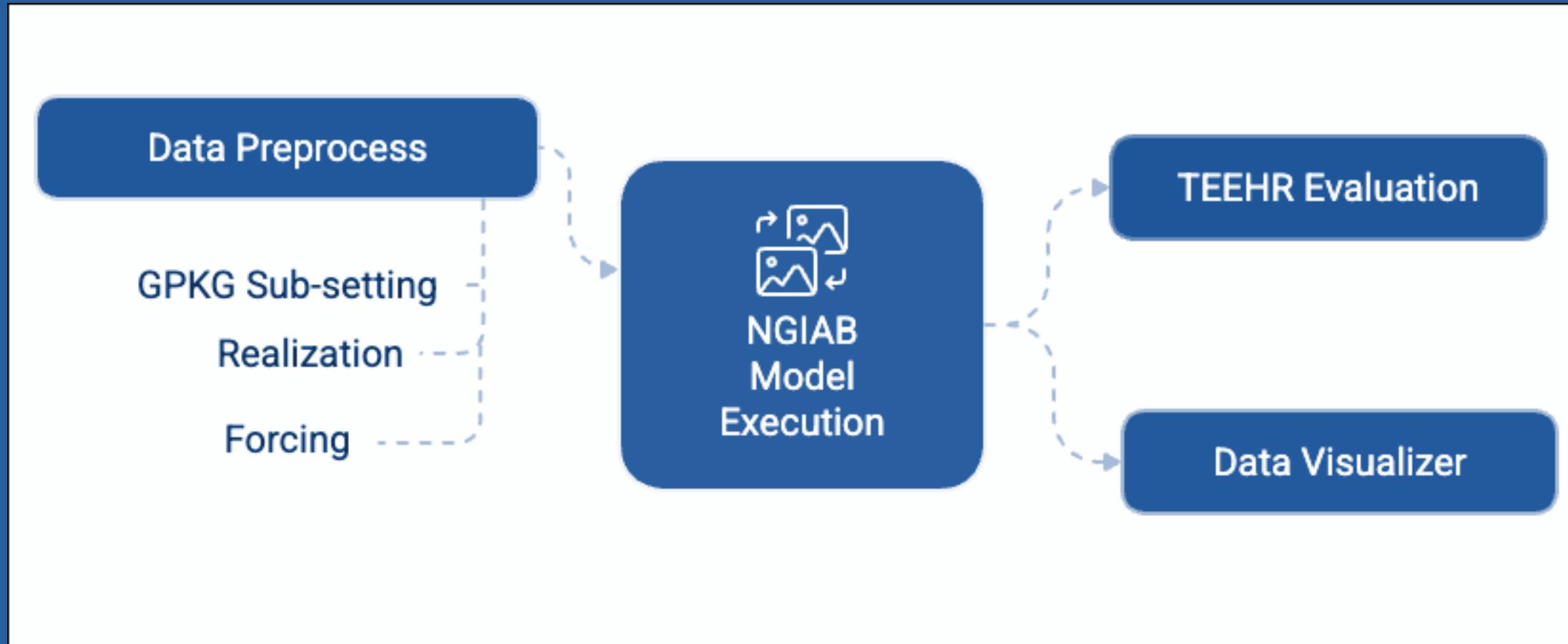
- CIROH at AGU 2024
- Community NextGen Updates
- CIROH Science Meeting 2024
- Accessing National Water Model (NWM)

(a) Cumulative Distribution Function vs Nash-Sutcliffe Model Efficiency (NSE)

(b) Map showing hydrological modeling results

Thanks to NGIAB, users don't have to worry about tricky setups or whether the model will run correctly. NGIAB ensures that our models are compatible everywhere and, most importantly, that they run exactly as designed, consistently and faithfully, every single time, no babysitting required. This means users get the full power of advanced modeling, without the headaches.

NGIAB Extensions



NGIAB and Extensions - Features

The screenshot shows a navigation bar with links to DocuHub, Products, Services, Community Impact, Learn, Blog, News, and Release Notes. The main content area has a header "Key Features" with tabs for Key Features, Capabilities, and Access Methods. A sidebar lists various products under "Community Hydrologic Modeling". The main table details the features of different extensions:

NGIAB and Extensions	Key features	NOAA-OWP Tools/Libraries Utilized
Data Preprocess	<ul style="list-style-type: none">Specializes in initial data preparationHandles subsetting and forcing processingSupports basic data processing tasksHelps with running NGIAB	<ul style="list-style-type: none">t-routehydrotoolshydrofabric tools
NGIAB Implementation (Cloud, HPC)	<ul style="list-style-type: none">Focused specifically on model executionCore engine for running simulationsDoes not handle pre/post-processing tasks	
TEEHR Evaluation	<ul style="list-style-type: none">Handles both input and output processingSupports full workflow, from data preparation to cloud deployment	Built to evaluate OWP model outputs
Data Visualizer	<ul style="list-style-type: none">Focused on analysis and validationSupports data processing and output analysis	Designed for OWP hydrofabric visualization
DataStreamCLI	<ul style="list-style-type: none">Complete workflow for creating inputs for and executing NGIAB and managing outputsBackend of the NextGen Research DataStreamDiscrete tooling for tasks like forcing processing and BMI file generation	<ul style="list-style-type: none">ngen-calt-routehydrofabric tools

NGIAB and Extensions - Capabilities

Capability	Data Preprocess	TEEHR Evaluation	Data Visualizer	DataStreamCLI	NGIAB-Cal
GUI	✓	-	✓	-	-
Hydrofabric Subsetting	✓	-	✓ (view only)	✓ ↗	-
NetCDF Forcing Processing	-	✓	-	✓ ↗	-
Zarr Forcing Processing	✓	✓	-	-	-
Forcing Metadata Generation	✓	-	-	✓ ↗	-
NextGen BMI Configuration File Generation	✓	-	-	✓ ↗	-
Directory and File Format Validation	↗	-	-	✓ ↗	-
NextGen Execution via NGIAB	✓	-	-	✓	-
Execution Metadata Generation	↗	-	-	✓ ↗	-
Calibration	-	-	-	-	✓
Evaluation	-	✓ ↗	↗ (displays TEEHR results)	✓	-
Visualization	-	↗ (metrics visualization)	✓	-	-

NGIAB and Extensions - Access Methods

Key Features	Capabilities	Access Methods					
Access method	Data Preprocess	NGIAB Implementation (Cloud, HPC)	TEEHR Evaluation	Data Visualizer	DataStreamCLI	NGIAB-Cal	
Docker	-	✓	✓	✓	✓	✓	
Python Package (pip/uv)	✓	✓	✓	-	-	✓	
Web Interface	✓	-	-	✓	-	-	
Notebook (ipynb)	-	-	✓	-	-	-	
Singularity (HPC)	-	✓	-	-	-	-	

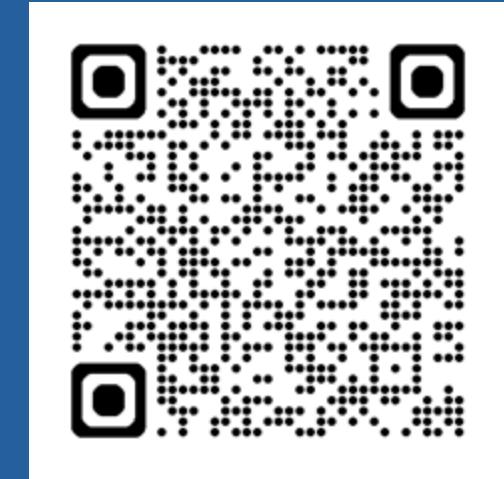
NGIAB Empowers Hydrologic Forecasting Researchers

Local Machine

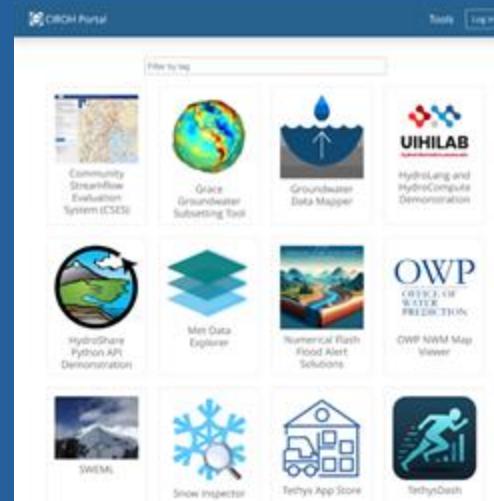
Cloud

HPC

AnyWhere!
(JetStream VM,
CIROH
JupyterHub e.g.)



docs.ciroh.org



portal.ciroh.org

Running NextGen In A Box (NGIAB) on HPC!

Using Singularity Image

The screenshot shows a web browser displaying the CIROH website at <https://docs.ciroh.org/docs/products/Community%20Hydrologic%20Modeling%20Framework/nextgeninaboxSingular...>. The page title is "NGIAB-HPCInfra". The left sidebar under "Products" has a section for "Community Hydrologic Modeling" with several options, including "NGIAB-HPCInfra" which is highlighted. The main content area starts with a "NOTE" section stating "Below content is rendered from <https://github.com/CIROH-UA/NGIAB-HPCInfra/blob/main/README.md>". Below this is a large heading "NextGen In A Box (NGIAB)". A sub-section below it says "Run the NextGen National Water Resources Modeling Framework locally with ease." and describes NGIAB as a containerized solution for running the framework locally. To the right of the text is a 3D illustration of a cube labeled "NEXTGEN IN A BOX" containing a landscape with clouds, water, and terrain.

DocuHub Products Services Community Impact Learn ▾

Blog News Release Note

Products

Community Hydrologic Modeling

GitHub Repository Dashboard

NGIAB Data Preprocess

Community Hydrofabric Patcher

NGIAB-CloudInfra >

NGIAB-HPCInfra

NGIAB TEEHR Integration

NGIAB Tethys Visualization Integration

NGIAB Calibration

NextGen Datastream >

Cyberinfrastructure and Community NextGen Office Hours

Evaluation Tools >

Data Management and Access Tools >

Snow Sensing and Modeling Tools >

Machine Learning and AI Tools >

Visualization and Analysis Tools >

CIROH Research Portal

Community Flood Inundation Mapping >

Community Hydrologic Modeling > NGIAB-HPCInfra

NGIAB-HPCInfra

NOTE

Below content is rendered from <https://github.com/CIROH-UA/NGIAB-HPCInfra/blob/main/README.md>

NextGen In A Box (NGIAB)

Run the NextGen National Water Resources Modeling Framework locally with ease.

NGIAB provides a containerized and user-friendly solution for running the NextGen framework, allowing you to control inputs, configurations, and execution on your local machine.

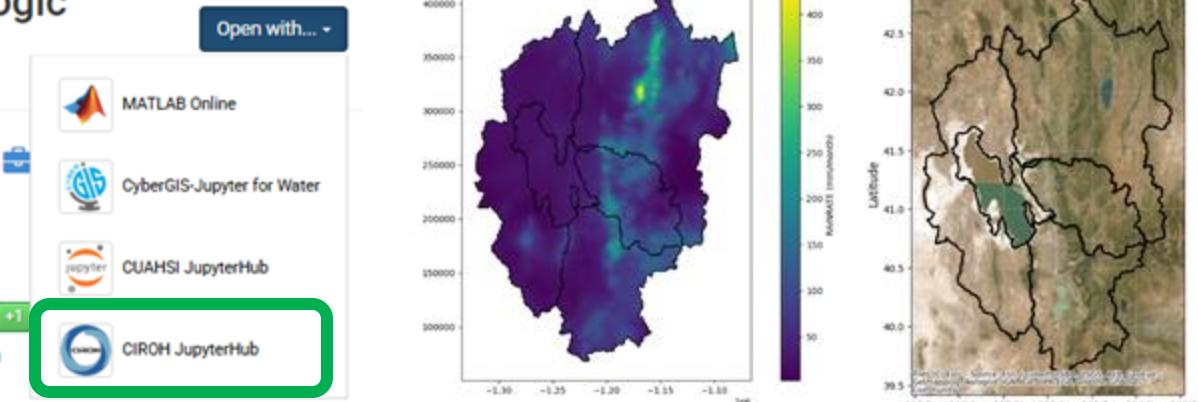
NEXTGEN IN A BOX

Enhancing and simplifying access to NOAA Analysis of Record for Calibration (AORC) dataset on CIROH JupyterHub

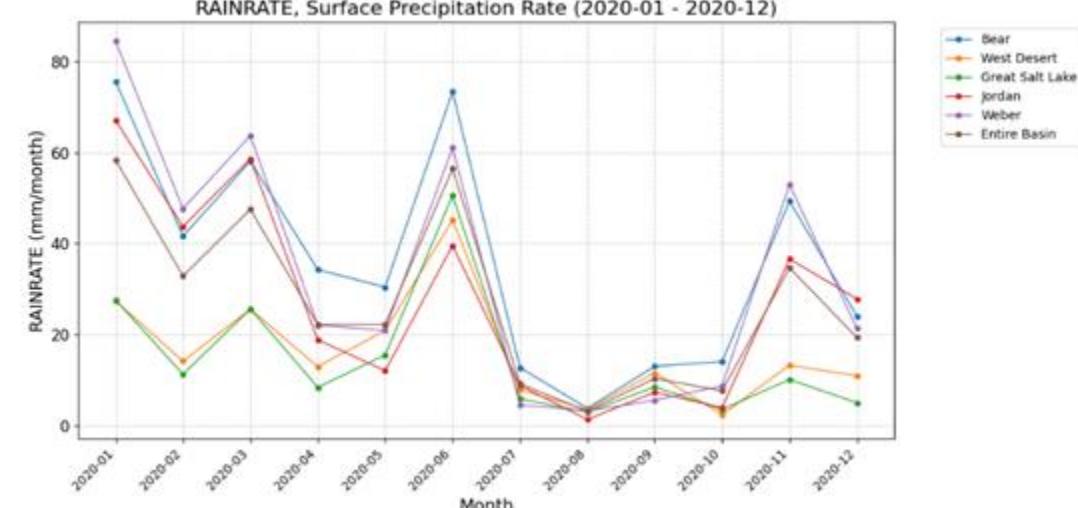
Salehabadi, H., D. Tarboton, A. Nassar, A. M. Castranova, P. Dash (2025). Jupyter Notebooks for the Retrieval of AORC Data for Hydrologic Analysis, HydroShare, <http://www.hydroshare.org/resource/72ea9726187e43d7b50a624f2acf591f>

Jupyter Notebooks for the Retrieval of AORC Data for Hydrologic Analysis

Authors:	Homa Salehabadi David Tarboton Ayman Nassar Anthony M. Castranova Pabitra Dash	Sharing Status:	Public
Owners:	Ayman Nassar David Tarboton Homa Salehabadi Anthony M. Castranova	Views:	2750
Type:	Resource	Downloads:	830
Storage:	The size of this resource is 6.8 MB	+1 Votes:	Be the first one to +1
Created:	Mar 18, 2024 at 7:24 p.m.	Comments:	No comments (yet)
Last updated:	Apr 17, 2025 at 4:36 p.m.		



1. Retrieve and aggregate data from the latitude-longitude gridded dataset for a specific point using geographic coordinates.
2. Retrieve and aggregate data from the latitude-longitude gridded dataset for an area defined by a polygon shapefile.
3. Retrieve and aggregate data from the NWM projected dataset for a specific point.
4. Retrieve and aggregate data from the NWM projected dataset for an area defined by a polygon shapefile.



NGIAB data preprocess and NextGen on CIROH JupyterHub

Tutorial in HydroShare search for cirohdevcon25

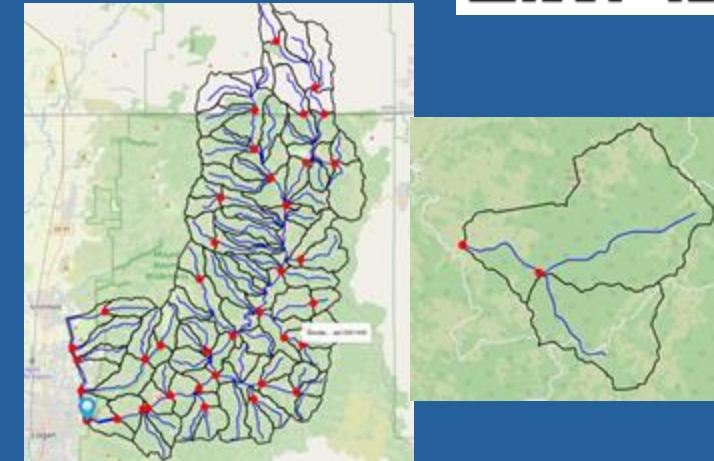


Working with HydroShare, AORC data, HydroFabric and NextGen on CIROH JupyterHub Tutorial

Authors: David Tarboton | Horma Salehabadi | Ayman Nassar | Furqan Baig | Anthony M. Castranova | Irene Garousi-Nejad | Arpita Patel
Owners: David Tarboton
Type: Resource
Storage: The size of this resource is 8.4 MB
Created: May 26, 2025 at 3:03 a.m.
Last updated: May 26, 2025 at 7:01 p.m.
Citation: See how to cite this resource

Sharing Status: Public
Views: 15
Downloads: 0
+1 Votes: Be the first one to +1 this.
Comments: No comments (yet)

Open with... MATLAB Online CUAHSI JupyterHub CIROH JupyterHub CIROH JupyterHub



source /ngen/.venv/bin/activate

Virtual environment

python -m ngiab_data_cli -i "gage-10109001" -s
python -m ngiab_data_cli -i "cat-2861446" -s

Hydrofabric

python -m ngiab_data_cli -i "cat-2861446" -f --start "2021-10-01" --end "2022-09-30"

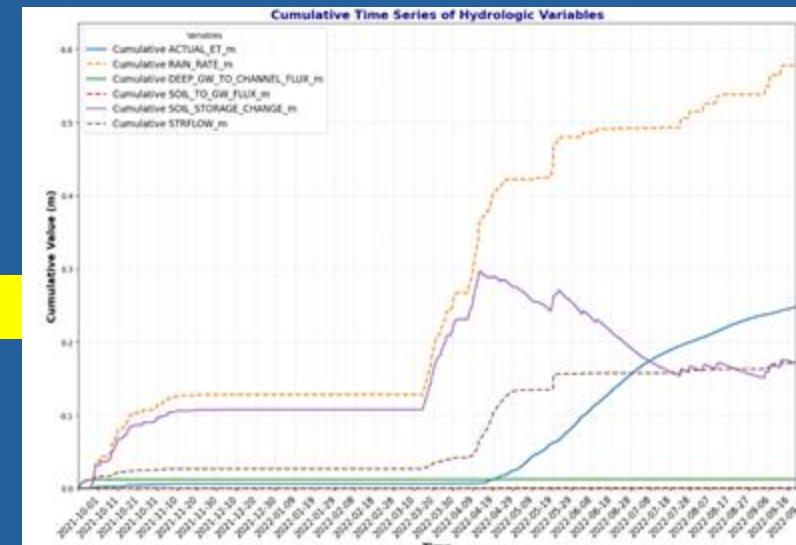
Forcing

python -m ngiab_data_cli -i "cat-2861446" -r --start "2021-10-01" --end "2022-09-30"

Configuration

/dm0d/bin/ngen-serial config/cat-2861446_subset.gpkg all config/cat-2861446_subset.apkg all

config/realization.json



NGIAB data preprocess and NextGen on CIROH JupyterHub

Hydrofabric

```
(ingen) (notebook) jovyan@jupyter-arpita0911patel:~$ python -m ngiab_data_cli -i "gage-10109001" -s
Hydrofabric is missing. Would you like to download it now? [y/n] (y): y
Downloading hydrofabrics/community/conus_nextgen.tar.gz to /home/jovyan/.ngiab/hydrofabric/v2.2/conus_nextgen.tar.gz...
The file downloads faster with no progress indicator, this should take around 30s
Please use network monitoring on your computer if you wish to track the download
2025-05-27 01:24:34,533 - WARNING - Connection pool is full, discarding connection: communityhydrofabric.s3.amazonaws.com. Connection pool size: 10
2025-05-27 01:24:34,546 - WARNING - Connection pool is full, discarding connection: communityhydrofabric.s3.amazonaws.com. Connection pool size: 10
```

```
Decompressing Hydrofabric...
: Decompressing 0:00:20
: Decompressing 0:01:16
Output directory is not set. Would you like to use the default? ~/ngiab_preprocess_output/ [y/n] (y): 2025-05-27 01:26:04,765 - INFO - Getting catid for 10109001, in /home/jovyan/.ngiab/hydrofabric/v2.2/conus_nextgen.gpk
2025-05-27 01:26:05,092 - INFO - Found cat-2861391 from gage-10109001
2025-05-27 01:26:05,094 - INFO - Processing cat-2861391 in /home/jovyan/ngiab_preprocess_output/gage-10109001
2025-05-27 01:26:05,096 - INFO - Building network graph
2025-05-27 01:27:23,265 - INFO - Upstream catchments: 88
2025-05-27 01:27:23,265 - INFO - Subsetting hydrofabric
cat-2861391
2025-05-27 01:27:23,599 - INFO - Subsetting tables: ['divides', 'divide-attributes', 'flowpath-attributes', 'flowpath-attributes-ml', 'flowpaths', 'hydrolocations', 'nexus', 'pois', 'lakes', 'network']
2025-05-27 01:27:34,098 - INFO - Subset complete for 213 features (catchments + nexuses)
2025-05-27 01:27:34,099 - INFO - Subsetting complete.
2025-05-27 01:27:34,099 - INFO - All operations completed successfully.
2025-05-27 01:27:34,099 - INFO - Output folder: file:///home/jovyan/ngiab_preprocess_output/gage-10109001
(ingen) (notebook) jovyan@jupyter-arpita0911patel:~$
```

NGIAB data preprocess and NextGen on CIROH JupyterHub

Forcing

```
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~$ python -m ngiab_data_cli -i "cat-2861446" -f --start "2021-10-01" --end "2022-09-30"
2025-05-27 02:19:55,259 - INFO - Processing cat-2861446 in /home/jovyan/ngiab_preprocess_output/cat-2861446
2025-05-27 02:19:56,161 - INFO - Upstream catchments: 2
2025-05-27 02:19:56,161 - INFO - Generating forcings from 2021-10-01 00:00:00 to 2022-09-30 00:00:00...

2025-05-27 02:20:33,587 - INFO - No cache found
2025-05-27 02:20:33,606 - INFO - Selected time range and clipped to bounds
2025-05-27 02:20:33,606 - INFO - Downloading and caching forcing data, this may take a while
[#####] | 27% Completed | 22.8s
forcing path cat-2861446 /home/jovyan/ngiab_preprocess_output/cat-2861446/forcings
2025-05-27 02:21:49,999 - INFO - Computing zonal stats in parallel for all timesteps
Forcings processed in 3.569252 seconds 100% 8/8 • Elapsed Time: 0:00:... Remaining Time: 0:00:...
2025-05-27 02:21:54,149 - INFO - Forcing generation complete! Zonal stats computed in 4.150324 seconds
2025-05-27 02:21:54,265 - INFO - Saving to disk
2025-05-27 02:21:54,455 - INFO - Forcings generation complete.
2025-05-27 02:21:55,456 - INFO - All operations completed successfully.
2025-05-27 02:21:55,456 - INFO - Output folder: file:///home/jovyan/ngiab_preprocess_output/cat-2861446
```

Configuration

```
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~$ python -m ngiab_data_cli -i "cat-2861446" -r --start "2021-10-01" --end "2022-09-30"
2025-05-27 02:22:17,098 - INFO - Processing cat-2861446 in /home/jovyan/ngiab_preprocess_output/cat-2861446
2025-05-27 02:22:17,992 - INFO - Upstream catchments: 2
2025-05-27 02:22:17,992 - INFO - Creating realization from 2021-10-01 00:00:00 to 2022-09-30 00:00:00...
2025-05-27 02:22:26,722 - INFO - Realization creation complete.
2025-05-27 02:22:26,723 - INFO - All operations completed successfully.
2025-05-27 02:22:26,723 - INFO - Output folder: file:///home/jovyan/ngiab_preprocess_output/cat-2861446
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~$ cd /home/jovyan/ngiab_preprocess_output/cat-2861446
```

NextGen Run

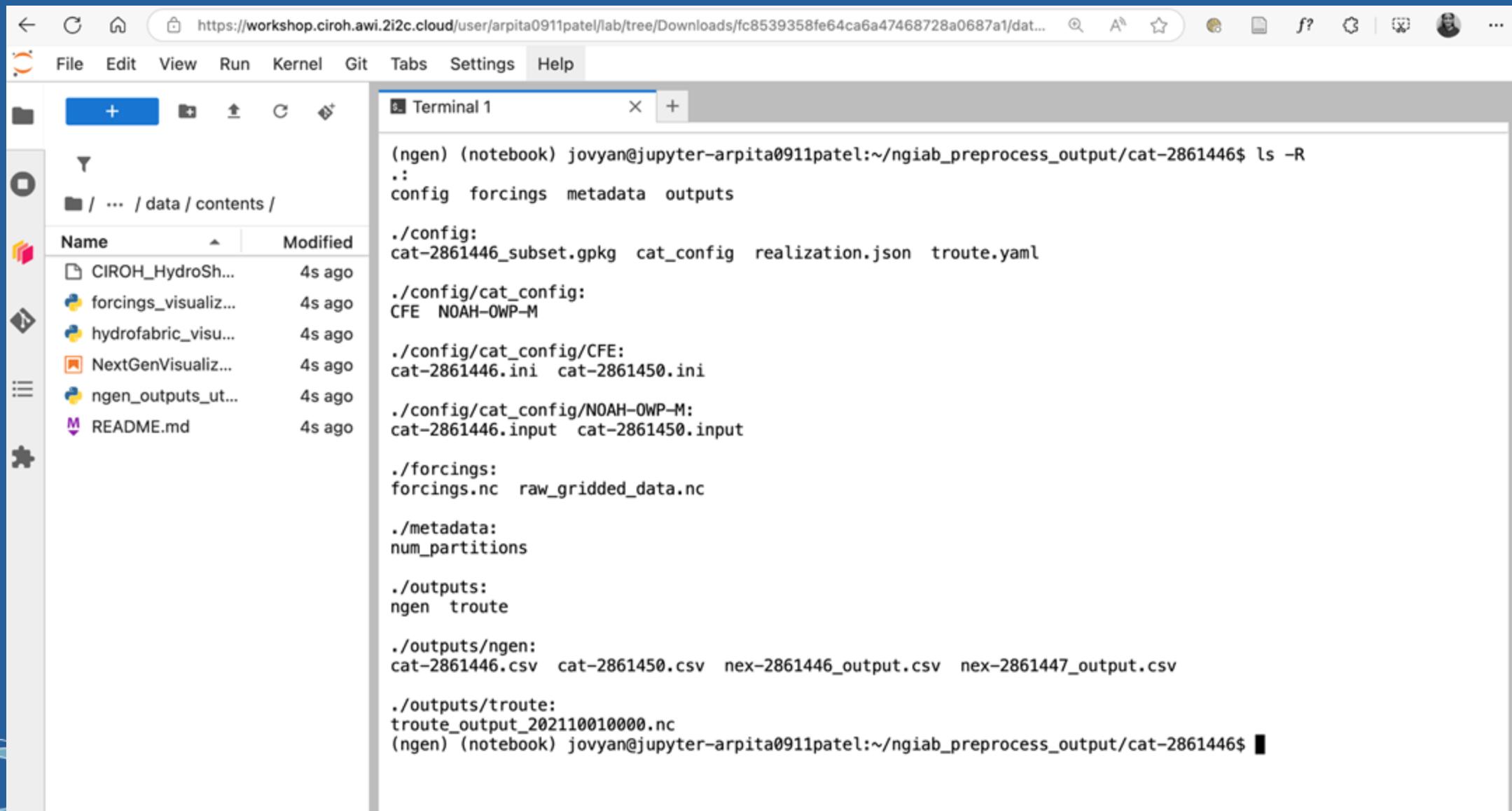
on CIROH JupyterHub

Run

```
.json
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~/ngiab_preprocess_output/cat-2861446$ ./dmod/bin/ngen-serial
config/cat-2861446_subset.gpkg all config/cat-2861446_subset.gpkg all config/realization.json
NGen Framework 0.3.0
Building Nexus collection
Reading 2 features from layer nexus using ID column `id`
Building Catchment collection
Reading 2 features from layer divides using ID column `divide_id`
Initializing formulations
[ {
    name :    bmi_c++,
    params :   {
        allow_exceed_end_time :           true,
        fixed_time_step :               false,
        init_config :                  /dev/null,
        library_file :                 /dmod/shared_libs/libslothmodel.so,
        main_output_variable :          z,
        model_params : {
            sloth_ice_fraction_schaake(1,double,m,node) :      0,
            sloth_ice_fraction_xinanjiang(1,double,1,node) :  0,
            sloth_soil_moisture_profile(1,double,1,node) :     0,
        },
    },
}
```

```
ulation
2025-05-27 02:29:54,989 - root - INFO - [compute.py:907 - compute_nhd_routing_v02]: PARALLEL TIME 4.4919953346
25244 seconds.
2025-05-27 02:29:54,991 - root - INFO - [output.py:180 - nwm_output_generator]: Handling output ...
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:340 - main_v04]: ****TIMING SUMMARY ****
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:341 - main_v04]: -----
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:342 - main_v04]: Network graph construction: 2.11 secs, 2
9.24 %
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:349 - main_v04]: Forcing array construction: 0.09 secs, 1
.22 %
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:356 - main_v04]: Routing computations: 4.49 secs, 62.36 %
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:363 - main_v04]: Output writing: 0.51 secs, 7.15 %
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:370 - main_v04]: -----
2025-05-27 02:29:55,506 - root - INFO - [__main__.py:371 - main_v04]: Total execution time: 7.199999999999999
secs
Finished routing
NGen top-level timings:
    NGen::init: 2.13784
    NGen::simulation: 3.3627
    NGen::routing: 7.21504
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~/ngiab_preprocess_output/cat-2861446$
```

NextGen o/p folder structure



The screenshot shows a Jupyter Notebook interface with a file browser on the left and a terminal window on the right.

File Browser:

- Path: / ... / data / contents /
- Items:

 - CIROH_HydroSh... (modified 4s ago)
 - forcings_visualiz... (modified 4s ago)
 - hydrofabric_visu... (modified 4s ago)
 - NextGenVisualiz... (modified 4s ago)
 - ngen_outputs_ut... (modified 4s ago)
 - README.md (modified 4s ago)

Terminal Window:

```
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~/ngiab_preprocess_output/cat-2861446$ ls -R
.:
config forcings metadata outputs

./config:
cat-2861446_subset.gpkg cat_config realization.json troute.yaml

./config/cat_config:
CFE NOAH-OWP-M

./config/cat_config/CFE:
cat-2861446.ini cat-2861450.ini

./config/cat_config/NOAH-OWP-M:
cat-2861446.input cat-2861450.input

./forcings:
forcings.nc raw_gridded_data.nc

./metadata:
num_partitions

./outputs:
ngen troute

./outputs/ngen:
cat-2861446.csv cat-2861450.csv nex-2861446_output.csv nex-2861447_output.csv

./outputs/troute:
troute_output_202110010000.nc
(ngen) (notebook) jovyan@jupyter-arpita0911patel:~/ngiab_preprocess_output/cat-2861446$
```

NGIAB Case Study HandsOn Demo

Provo River Basin Case Study!

Please visit

<https://docs.ciroh.org/training-NGIAB-101/devcon25js.html>



Workshop credentials email

[EXTERNAL] DevCon 2025 - NextGen Workshop Credentials

AP Arpita Patel via Smartsheet <automation@app.smartsheet.com>
To: CIROH IT Admin

CIROH
Cooperative Institute for Research
to Operations in Hydrology

DevCon 2025 - NextGen Workshop Credentials

Dear CIROH IT Administration,

Hello! Here are your credentials for your NextGen track workshops during DevCon 2025:

Username: excuser
Password: N/A
IP Address: N/A

To connect to your instance, use the following command in your Unix terminal:
ssh -L 5906:localhost:5906 excuser@N/A

Instances are currently shutdown to avoid using the unnecessary credits. Our schedule to start the instances are as below:
May 27 2025 (Tue): 9am EST to 11am EST
May 28 2025 (Wed): 8am EST to 6pm EST
May 29 2025 (Thur): 8am EST to 5pm EST
May 30 2025 (Fri): 8am EST to 4:30pm EST

These credentials will be used during the following workshops:

- "Navigating the NextGen Ecosystem and NextGen In A Box (NGIAB)"
- "Output visualization through Tethys and evaluation customization using TEEHR"
- "NextGen Calibration Workshop"
- "NextGen Research DataStream: How to Contribute to Improving NextGen Forecasts"

Please be sure to install a VNC client prior to your workshop. While any VNC client should work, these are some common choices:

- TigerVNC (<https://tigervnc.org/>)
- RealVNC (<https://www.realvnc.com/en/connect/download/viewer/>)

We look forward to seeing you at DevCon 2025! For any questions please reach out to us at ciroh-it-support@ua.edu

— CIROH Research Cyberinfrastructure Team

Task 1

When logging in for the first time, you may be asked whether you'd like to trust the host. Type 'yes' to do so.

After that, simply type in your password to gain access to your instance's terminal.

Task 1: Running NGIAB With Prepared Data

- Run the following command to preprocess data:

```
BASH < >
uvx --from ngiab_data_preprocess cli -i gage-10154200 -sfr --st 17-
```

[Copy](#)
- Run the following commands to clone the NGIAB-CloudInfra repo and run guide script:

```
BASH < >
mkdir NGIAB_demo
cd NGIAB_demo
git clone https://github.com/CIROH-UA/NGIAB-CloudInfra.git
cd NGIAB-CloudInfra
./guide.sh
```

[Copy](#)
- When prompted, enter the following input data directory path:

```
/home/exouser/ngiab_preprocess_output/gage-10154200
```
- Follow the prompts. Choose between serial or parallel mode. Serial will run NextGen on one process, whereas parallel will run NextGen on multiple processes at once.
- When prompted to redirect command output to `/dev/null`, select yes. This keeps your output logs clean.
- After the run is completed, run the TFFHR evaluation when prompted. Choose Option 1 (use

NGIAB Data Preprocess

```
exouser@devcon25-ngiab-demo ~  
uvx --from ngiab_data_preprocess cli -i gage-10154200 -sfr --start 2017-09-01 --end 2018-09-01 --source aorc  
Installed 77 packages in 473ms  
Output directory is not set. Would you like to use the default? ~/ngiab_preprocess_output/ [y/n] (y): █
```

When prompted type “y” Use the default output folder given.

```
exouser@devcon25-ngiab-demo ~ (im 41.65s)
uvx --from ngiab_data_preprocess cli -i gage-10154200 -sfr --start 2017-09-01 --end 2018-09-01 --source aorc
2025-05-24 15:50:18,444 - INFO - Getting catid for 10154200, in /home/exouser/.ngiab/hydrofabric/v2.2/conus_n
2025-05-24 15:50:18,451 - INFO - Found cat-2863631 from gage-10154200
2025-05-24 15:50:18,452 - INFO - Processing cat-2863631 in /home/exouser/ngiab_preprocess_output/gage-1015420
2025-05-24 15:50:18,452 - INFO - Building network graph
2025-05-24 15:50:54,672 - INFO - Upstream catchments: 53
2025-05-24 15:50:54,672 - INFO - Subsetting hydrofabric
2025-05-24 15:50:54,895 - INFO - Subsetting tables: ['divides', 'divide-attributes', 'flowpath-attributes', 'butes-ml', 'flowpaths', 'hydrolocations', 'nexus', 'pois', 'lakes', 'network']
2025-05-24 15:51:03,512 - INFO - Subset complete for 135 features (catchments + nexuses)
2025-05-24 15:51:03,513 - INFO - Subsetting complete.
2025-05-24 15:51:03,513 - INFO - Generating forcings from 2017-09-01 00:00:00 to 2018-09-01 00:00:00...
2025-05-24 15:51:05,583 - INFO - Loading AORC zarr datasets from 2017 to 2018
2025-05-24 15:51:05,583 - INFO - This should take roughly 6.0 seconds
2025-05-24 15:51:08,335 - INFO - No cache found
2025-05-24 15:51:08,346 - INFO - Selected time range and clipped to bounds
2025-05-24 15:51:08,346 - INFO - Downloading and caching forcing data, this may take a while
2025-05-24 15:51:23,452 - INFO - Computing zonal stats in parallel for all timesteps
Forcings processed in 4.347495 seconds ██████████ 100% 8/8 • Elapsed Time: 0:00:03 Remaining
2025-05-24 15:51:28,291 - INFO - Forcing generation complete! Zonal stats computed in 4.838826 seconds
2025-05-24 15:51:28,351 - INFO - Saving to disk
2025-05-24 15:51:28,521 - INFO - Forcings generation complete.
2025-05-24 15:51:28,521 - INFO - Creating realization from 2017-09-01 00:00:00 to 2018-09-01 00:00:00...
2025-05-24 15:51:28,830 - INFO - downloaded calibrated parameters for gage-10154200
2025-05-24 15:51:28,867 - INFO - Realization creation complete.
2025-05-24 15:51:29,248 - INFO - All operations completed successfully.
2025-05-24 15:51:29,248 - INFO - Output folder: file:///home/exouser/ngiab_preprocess_output/gage-10154200
```

NGIAB Data Preprocess

Output Folder Structure from Data Preprocess

```
exouser@devcon25-ngiab-demo:~/ngiab_preprocess_output/gage-10154200 (0.057s)
ls -R /home/exouser/ngiab_preprocess_output/gage-10154200
/home/exouser/ngiab_preprocess_output/gage-10154200:
config forcings metadata outputs

/home/exouser/ngiab_preprocess_output/gage-10154200/config:
cat_config downloaded_params.json gage-10154200_subset.gpkg realization.json troute.yaml

/home/exouser/ngiab_preprocess_output/gage-10154200/config/cat_config:
CFE NOAH-OWP-M

/home/exouser/ngiab_preprocess_output/gage-10154200/config/cat_config/CFE:
cat-2863621.ini cat-2863627.ini cat-2863809.ini cat-2863815.ini cat-2863821.ini cat-2863827.ini cat-2863833.ini cat-2863839.ini cat-2863845.ini
cat-2863622.ini cat-2863628.ini cat-2863810.ini cat-2863816.ini cat-2863822.ini cat-2863828.ini cat-2863834.ini cat-2863840.ini cat-2863846.ini
cat-2863623.ini cat-2863629.ini cat-2863811.ini cat-2863817.ini cat-2863823.ini cat-2863829.ini cat-2863835.ini cat-2863841.ini cat-2863847.ini
cat-2863624.ini cat-2863630.ini cat-2863812.ini cat-2863818.ini cat-2863824.ini cat-2863830.ini cat-2863836.ini cat-2863842.ini cat-2863848.ini
cat-2863625.ini cat-2863631.ini cat-2863813.ini cat-2863819.ini cat-2863825.ini cat-2863831.ini cat-2863837.ini cat-2863843.ini cat-2863849.ini
cat-2863626.ini cat-2863808.ini cat-2863814.ini cat-2863820.ini cat-2863826.ini cat-2863832.ini cat-2863838.ini cat-2863844.ini

/home/exouser/ngiab_preprocess_output/gage-10154200/config/cat_config/NOAH-OWP-M:
cat-2863621.input cat-2863627.input cat-2863809.input cat-2863815.input cat-2863821.input cat-2863827.input cat-2863833.input cat-2863839.input cat-2863845.
cat-2863622.input cat-2863628.input cat-2863810.input cat-2863816.input cat-2863822.input cat-2863828.input cat-2863834.input cat-2863840.input cat-2863846.
cat-2863623.input cat-2863629.input cat-2863811.input cat-2863817.input cat-2863823.input cat-2863829.input cat-2863835.input cat-2863841.input cat-2863847.
cat-2863624.input cat-2863630.input cat-2863812.input cat-2863818.input cat-2863824.input cat-2863830.input cat-2863836.input cat-2863842.input cat-2863848.
cat-2863625.input cat-2863631.input cat-2863813.input cat-2863819.input cat-2863825.input cat-2863831.input cat-2863837.input cat-2863843.input cat-2863849.
cat-2863626.input cat-2863808.input cat-2863814.input cat-2863820.input cat-2863826.input cat-2863832.input cat-2863838.input cat-2863844.input

/home/exouser/ngiab_preprocess_output/gage-10154200/forcings:
forcings.nc raw_gridded_data.nc

/home/exouser/ngiab_preprocess_output/gage-10154200/metadata:
num_partitions

/home/exouser/ngiab_preprocess_output/gage-10154200/outputs:
ngen troute

/home/exouser/ngiab_preprocess_output/gage-10154200/outputs/ngen:

/home/exouser/ngiab_preprocess_output/gage-10154200/outputs/troute:
```

NGIAB Run

```
exouser@devcon25-ngiab-demo ~  
mkdir NGIAB_demo  
cd NGIAB_demo  
git clone https://github.com/CIROH-UA/NGIAB-CloudInfra.git  
cd NGIAB-CloudInfra  
../guide.sh
```

CIROH: NextGen In A Box (NGIAB)
Advanced Hydrologic Modeling Tool

→ Visit our website: <https://ngiab.ciroh.org>
Developed by CIROH & Lynker

NGIAB Execution

MODEL INPUT REQUIREMENTS

This application requires the following directory structure for proper operation:

- `forcings/` - Contains hydrofabric input data for model simulations
 - └ Meteorological and terrain data for hydrologic calculations
- `config/` - Contains all configuration settings for the model
 - └ Model parameters, simulation period, and execution settings
- `outputs/` - Target directory for simulation results
 - └ Flow estimates, water levels, and diagnostic information

i Please specify a directory containing these components below.

- Last used data directory: `/home/exouser/ngiab_preprocess_output/gage-10154200`
 - Use the same path? [Y/n]:

**Copy path from
previous
`dataprocess`
o/p and paste
here!
Type “Y” if path
is displayed**

NGIAB Execution

VALIDATING INPUT DIRECTORY

Checking directory structure at: /home/exouser/ngiab_preprocess_output/gage-10154200

- ✓ Analyzing directory structure - Complete!
 - Checking for **forcings** directory... ✓ Found with 2 files
 - Checking for **config** directory... ✓ Found with 110 files
 - Checking for **outputs** directory... ✓ Found but empty (0 files)
 - Checking for **restarts** directory... △ Optional directory not found. Continuing...

SUCCESS: Directory structure validated successfully!

FILE MANAGEMENT

- Checking **Outputs** directory for existing files...
✓ **Outputs** is clean and ready for new simulations!
- Checking **Restarts** directory for existing files...
✓ **Restarts** is clean and ready for new simulations!

NGIAB Execution

HYDROFABRIC ANALYSIS

```
✓ Scanning for model files - Complete!
→ Searching for hydrofabric files...
  ✓ Found 1 files:
    /home/exouser/ngiab_preprocess_output/gage-10154200/config/gage-10154200_subset.gpkg
→ Searching for realization files...
  ✓ Found 1 files:
    /home/exouser/ngiab_preprocess_output/gage-10154200/config/realization.json
```

SYSTEM CONFIGURATION

```
→ Hardware Detection:
  i Operating System: Linux
  i Architecture: x86_64

→ Checking for Docker:
  ✓ Docker detected (version: 26.1.3)
```

Select Option 1
(use existing
Docker image)

MODEL EXECUTION OPTIONS

```
→ Please select an option to proceed:
1) Run NextGen using existing local docker image 3) Exit
2) Update to latest docker image and run
#?
```

NGIAB Execution

Selected files:

Catchment: ./config/gage-10154200_subset.gpkg

Nexus: ./config/gage-10154200_subset.gpkg

Realization: config/realization.json

Generated 8 partitions

Your NGEN run command is mpirun -n 8 /dmod/bin/ngen-parallel ./config/gage-10154200_subset.gpkg all ./config/gage-10154200_subset.gpkg all config/realization.json /ngen/ngen/data/partitions_8.json

If your model didn't run, or encountered an error, try checking the Forcings paths in the Realizations file you selected.

Do you want to redirect command output to /dev/null? (y/N, default: n):

When prompted to redirect command output to /dev/null, select no to see details about the run

```
Updating layer: surface layer
Running timestep 3000
Updating layer: surface layer
Running timestep 3000
Updating layer: surface layer
Running timestep 3100
Updating layer: surface layer
Running timestep 4300
Updating layer: surface layer
Running timestep 3100
Updating layer: surface layer
Running timestep 3600
Updating layer: surface layer
Running timestep 3100
Updating layer: surface layer
Running timestep 1000
Updating layer: surface layer
Running timestep 4400
Updating layer: surface layer
Running timestep 3100
Updating layer: surface layer
Running timestep 3100
Updating layer: surface layer
Running timestep 3200
Updating layer: surface layer
Running timestep 3700
```

```
Finished 8761 timesteps.
creating supernetwork connections set
2025-05-24 16:24:52,318 - root - INFO - [AbstractNetwork.py:525 - create_independent_networks]: organizing connections into reaches ...
2025-05-24 16:24:52,319 - root - INFO - [AbstractNetwork.py:682 - initial_warmstate_preprocess]: setting channel initial states ...
2025-05-24 16:24:52,329 - root - INFO - [AbstractNetwork.py:128 - assemble_forcings]: Creating a DataFrame of lateral inflow forcings ...
supernetwork connections set complete
... in 0.9965684413909912 seconds.
2025-05-24 16:24:53,210 - root - INFO - [DataAssimilation.py:77 - __init__]: NudgingDA class is Started.
2025-05-24 16:24:53,212 - root - INFO - [DataAssimilation.py:286 - __init__]: PersistenceDA class is started.
2025-05-24 16:24:53,213 - root - INFO - [DataAssimilation.py:840 - __init__]: RFCDA class is started.
2025-05-24 16:24:53,214 - root - INFO - [DataAssimilation.py:719 - __init__]: great_lake class is started.
2025-05-24 16:24:53,215 - root - INFO - [__main__.py:1178 - nwm_route]: executing routing computation ...
2025-05-24 16:24:53,217 - root - INFO - [compute.py:659 - compute_nhd_routing_v02]: JIT Preprocessing time 0.00056314468383
78906 seconds.
2025-05-24 16:24:53,217 - root - INFO - [compute.py:660 - compute_nhd_routing_v02]: starting Parallel JIT calculation
2025-05-24 16:24:56,195 - root - INFO - [compute.py:907 - compute_nhd_routing_v02]: PARALLEL TIME 2.9787158966064453 seconds.
2025-05-24 16:24:56,199 - root - INFO - [output.py:180 - nwm_output_generator]: Handling output ...
```

NGIAB Execution

```
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:340 - main_v04]: ***** TIMING SUMMARY *****
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:341 - main_v04]: -----
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:342 - main_v04]: Network graph construction: 1.0 secs, 18.51 %
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:349 - main_v04]: Forcing array construction: 0.9 secs, 16.61 %
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:356 - main_v04]: Routing computations: 2.98 secs, 55.29 %
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:363 - main_v04]: Output writing: 0.51 secs, 9.53 %
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:370 - main_v04]: -----
2025-05-24 16:24:56,713 - root - INFO - [__main__.py:371 - main_v04]: Total execution time: 5.39 secs
Finished routing
NGen top-level timings:
    NGen::init: 4.31709
    NGen::simulation: 14.8263
    NGen::routing: 5.41092

real    0m25.071s
user    1m34.448s
sys     0m33.774s
Finished executing command successfully.
Would you like to continue?
Select an option (type a number):
1) Interactive-Shell
2) Exit
#?
```

NextGen run is complete!

**Select Option 2
(exit)**

NextGen Output

```
exouser@devcon25-ngaib-demo ~/ngaib_preprocess_output/gage-10154200/outputs (0.058s)
ls -R
.:
ngen  troute

./ngen:
cat-2863621.csv  cat-2863811.csv  cat-2863825.csv  cat-2863839.csv      nex-2863624_output.csv  nex-2863826_output.csv
cat-2863622.csv  cat-2863812.csv  cat-2863826.csv  cat-2863840.csv      nex-2863625_output.csv  nex-2863828_output.csv
cat-2863623.csv  cat-2863813.csv  cat-2863827.csv  cat-2863841.csv      nex-2863626_output.csv  nex-2863829_output.csv
cat-2863624.csv  cat-2863814.csv  cat-2863828.csv  cat-2863842.csv      nex-2863627_output.csv  nex-2863830_output.csv
cat-2863625.csv  cat-2863815.csv  cat-2863829.csv  cat-2863843.csv      nex-2863628_output.csv  nex-2863831_output.csv
cat-2863626.csv  cat-2863816.csv  cat-2863830.csv  cat-2863844.csv      nex-2863629_output.csv  nex-2863832_output.csv
cat-2863627.csv  cat-2863817.csv  cat-2863831.csv  cat-2863845.csv      nex-2863630_output.csv  nex-2863836_output.csv
cat-2863628.csv  cat-2863818.csv  cat-2863832.csv  cat-2863846.csv      nex-2863631_output.csv  nex-2863837_output.csv
cat-2863629.csv  cat-2863819.csv  cat-2863833.csv  cat-2863847.csv      nex-2863632_output.csv  nex-2863838_output.csv
cat-2863630.csv  cat-2863820.csv  cat-2863834.csv  cat-2863848.csv      nex-2863809_output.csv  nex-2863842_output.csv
cat-2863631.csv  cat-2863821.csv  cat-2863835.csv  cat-2863849.csv      nex-2863810_output.csv  nex-2863843_output.csv
cat-2863808.csv  cat-2863822.csv  cat-2863836.csv  nex-2863621_output.csv  nex-2863821_output.csv
cat-2863809.csv  cat-2863823.csv  cat-2863837.csv  nex-2863622_output.csv  nex-2863822_output.csv
cat-2863810.csv  cat-2863824.csv  cat-2863838.csv  nex-2863623_output.csv  nex-2863823_output.csv

./troute:
troute_output_201709010000.nc
```

TEEHR Evaluation

EVALUATION OPTIONS

→ Would you like to run a TEEHR evaluation on the output?

- ↳ This will analyze your simulation results using the TEEHR toolkit
- ↳ Learn more: <https://rtiinternational.github.io/ngiab-teehr/>

Run TEEHR evaluation? [Y/n]:

For TEEHR evaluation

Type “Y”

CIROH: NextGen In A Box (NGIAB) - TEEHR Evaluation
Model Performance Assessment Tool

↳ Developed by CIROH

TEEHR EVALUATION SETUP

↳ TEEHR will evaluate model outputs against observations

→ Learn more: <https://rtiinternational.github.io/ngiab-teehr/>

→ Would you like to run a TEEHR evaluation on your model outputs?

Run evaluation? [Y/n]:

Type “Y” to
proceed

TEEHR Evaluation

CIROH: NextGen In A Box (NGIAB) - TEEHR Evaluation
Model Performance Assessment Tool

- Developed by CIROH

TEEHR EVALUATION SETUP

- TEEHR will evaluate model outputs against observations
 - Learn more: <https://rtiinternational.github.io/ngiab-teehr/>
- Would you like to run a TEEHR evaluation on your model outputs?
Run evaluation? [Y/n]: Y
- System architecture detected: x86_64
 - Recommended image tag: x86

For JetStream VM,
choose x86

Choose Option 1

TEEHR Evaluation

- System architecture detected: x86_64
 - i Recommended image tag: x86
 - ✓ Using default tag: x86

CONTAINER MANAGEMENT

- Select an option:

- 1) Run TEEHR using existing local image
- 2) Update to latest TEEHR image
- 3) Exit

#? █

For JetStream VM,
choose x86

Choose Option 1

TEEHR Evaluation

✓ Using existing local TEEHR image

RUNNING TEEHR EVALUATION

```
| Evaluating model outputs in: /home/exouser/ngiab_preprocess_output/gage-10154200
|   → This analysis may take several minutes depending on your dataset size
✓ Initializing TEEHR evaluation - Complete!

→ Cleaning up resources...
✓ Cleanup completed
/usr/local/lib/python3.11/site-packages/pyspark/bin/load-spark-env.sh: line 68: ps: command not found
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/05/24 16:33:27 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

```
INFO:teeehr.visualization.dataframe_accessor:Generating timeseries plot.
INFO:teeehr.visualization.dataframe_accessor:Processing combination: ('ngen', 'ngen-2863631', NaT)
INFO:teeehr.visualization.dataframe_accessor:
    reference_time == NaT, ignoring reference_time for
    combo: ('ngen', 'ngen-2863631', NaT)

INFO:teeehr.visualization.dataframe_accessor:Plotting data for combination: ('ngen', 'ngen-2863631', NaT)
INFO:teeehr.visualization.dataframe_accessor:Processing combination: ('nwm30_retrospective', 'nwm30-10373692', NaT)
INFO:teeehr.visualization.dataframe_accessor:
    reference_time == NaT, ignoring reference_time for
    combo: ('nwm30_retrospective', 'nwm30-10373692', NaT)

INFO:teeehr.visualization.dataframe_accessor:Plotting data for combination: ('nwm30_retrospective', 'nwm30-10373692', NaT)
INFO:teeehr.visualization.dataframe_accessor:Saving timeseries plot at data/teeehr
```

EVALUATION COMPLETE

TEEHR evaluation completed successfully!

TEEHR Evaluation

EVALUATION COMPLETE

TEEHR evaluation completed successfully!

- ℹ Results have been saved to your outputs directory:
→ `/home/exouser/ngiab_preprocess_output/gage-10154200/outputs/teehr/`
- ℹ You can visualize these results using the Tethys platform
→ Run `./viewOnTethys.sh /home/exouser/ngiab_preprocess_output/gage-10154200` to start visualization

Thank you for using NGIAB!

- ℹ For support, please email: ciroh-it-support@ua.edu

- Cleaning up resources...
✓ Cleanup completed
- Cleaning up resources...
✓ Cleanup completed

TEEHR Evaluation

```
exouser@devcon25-ngiab-demo ~/ngiab_preprocess_output/gage-10154200 (0.053s)
ls tee

---


__init__.py dataset      ngen_usgs_crosswalk.parquet  readme.md  tee

---

.log
cache          metrics.csv  nwm_usgs_crosswalk.parquet  scripts    timeseries_plot_streamflow_hourly_inst.html
```

```
exouser@devcon25-nglab-demo ~/ngiab_preprocess_output/gage-10154200 (0.053s)
ls -R tee

---


tee

---

:
__init__.py dataset      ngen_usgs_crosswalk.parquet  readme.md  tee

---

.log
cache          metrics.csv  nwm_usgs_crosswalk.parquet  scripts    timeseries_plot_streamflow_hourly_inst.html

tee

---

/cache:
fetching loading locations.parquet ngen_output.parquet readme.md

tee

---


```

```
tee

---

/cache/loading:
location_crosswalks locations secondary_timeseries

tee

---

/cache/loading/location_crosswalks:
ngen_usgs_crosswalk.parquet nwm_usgs_crosswalk.parquet

tee

---

/cache/loading/locations:
locations.parquet

tee

---

/cache/loading/secondary_timeseries:
ngen_output.parquet

tee

---

/dataset:
attributes joined_timeseries location_crosswalks primary_timeseries units
configurations location_attributes locations secondary_timeseries variables

tee

---

/dataset/attributes:
_readme.md attributes.csv

tee

---

/dataset/configurations:
_readme.md configurations.csv

tee

---

/dataset/joined_timeseries:
_readme.md 'configuration_name=ngen' 'configuration_name=nwm30_retrospective'

'tee

---

/dataset/joined_timeseries/configuration_name=ngen':
'veariable_name=streamflow_hourly_inst'

'tee

---

/dataset/joined_timeseries/configuration_name=ngen/variable_name=streamflow_hourly_inst':
part-00000-c4736269-4023-449e-967d-902a68650f65.c000.snappy.parquet
```

NGIAB Visualizer

VISUALIZATION OPTIONS

→ Would you like to visualize results using Tethys?

i This will provide an interactive web interface to explore your model results
Visualize results? [Y/n]:

Type "Y"

```
=====
| CIROH: NextGen In A Box (NGIAB) - Tethys |
| Interactive Model Output Visualization |
=====
```

i Developed by CIROH

PREPARING VISUALIZATION ENVIRONMENT

```
✓ Copied → /home/exouser/ngiab_visualizer/gage-10154200
Checking for /home/exouser/ngiab_visualizer/ngiab_visualizer.json...
✓ Model run "gage-10154200" registered (f9f9b308-eabe-4fd1-9f4d-501f6fcba474)
i Directory /home/exouser/.datastream_ngiab doesn't exist – creating it...
i No existing Datastream cache found - a fresh download will be used.
```

LAUNCHING TETHYS VISUALIZATION

Specify the Tethys image tag to use:

→ Tag (e.g. v0.2.1, default: latest):

Choose the default latest image but hitting enter

NGIAB Visualizer

Specify the Tethys image tag to use:

- Tag (e.g. v0.2.1, default: latest):
- i Found local image `awiciroh/tethys-ngiab:latest`
- Use local copy (L) or Pull latest from registry (P)? [L/P]:

Type “L” to choose local copy

✓ Using local image

Select a port to run Tethys on. [Default: 80]

- Port:

Hit enter so select default port 80, if in use then select 81

- Using default port 80 for Tethys.
 - ✓ Port 80 selected
 - Launching Tethys container...
 - i Tethys container is already running. Stopping it first...
 - ⚠️ Forcibly removing container...
 - i Setting up Docker network for Tethys...
 - ✓ Network `tethys-network` created successfully.
 - i Starting Tethys container...
 - i Running docker command...
- `8c90ff9a54a6e294630b0f1b72915cc028cadc7e5af5cba303cf5c80580cb074`
- ✓ Tethys container started successfully.
 - i Waiting for container: `tethys-ngen-portal` to become healthy. This can take a couple of minutes...

This might take a minute or so!

NGIAB Tethys Visualizer

```
i Starting Tethys container...
i Running docker command...
8c90ff9a54a6e294630b0f1b72915cc028cadc7e5af5cba303cf5c80580cb074
✓ Tethys container started successfully.
i Waiting for container: tethys-ngen-portal to become healthy. This can take a couple of minutes...

✓ Container tethys-ngen-portal is now healthy!
```

VISUALIZATION READY

Your model outputs are now available for visualization!

- i Access the visualization at: <http://localhost:80/apps/ngiab>
- i Login credentials:
 - Username: admin
 - Password: pass
- i Source code: <https://github.com/CIROH-UA/ngiab-client>

**Tethys Visualizer is
now running!
Open VNC Client to
browse this URL**

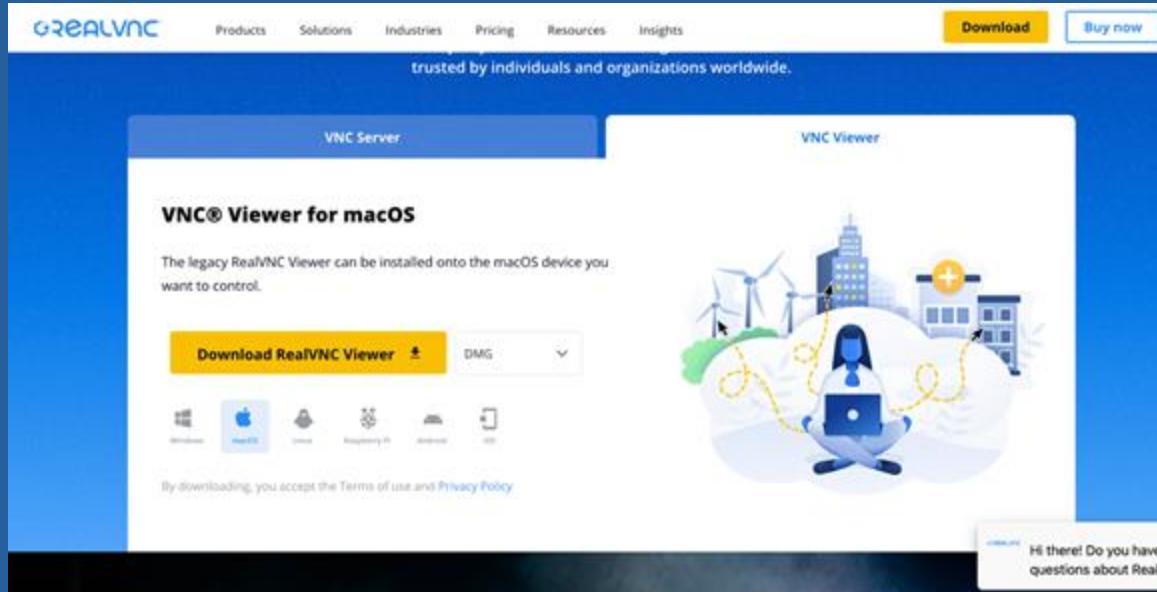
Tethys is now running

- i Access the visualization at: <http://localhost:80/apps/ngiab>
- i Press **Ctrl+C** to stop Tethys when you're done.

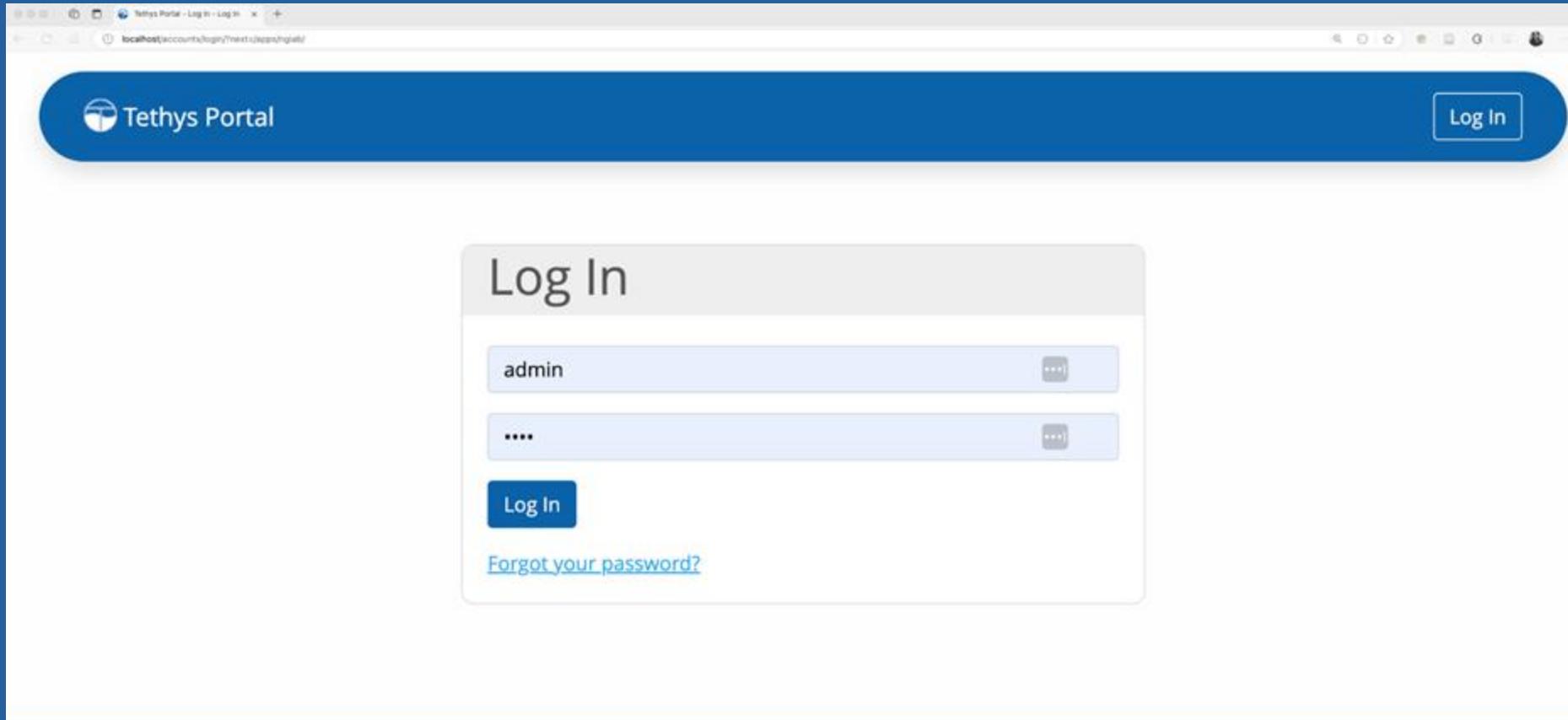
VNC Client Setup

While any VNC client should work, these are common choices:

- TigerVNC (<https://tigervnc.org/>)
- RealVNC
(<https://www.realvnc.com/en/connect/download/viewer/>)



NGIAB Tethys Visualizer

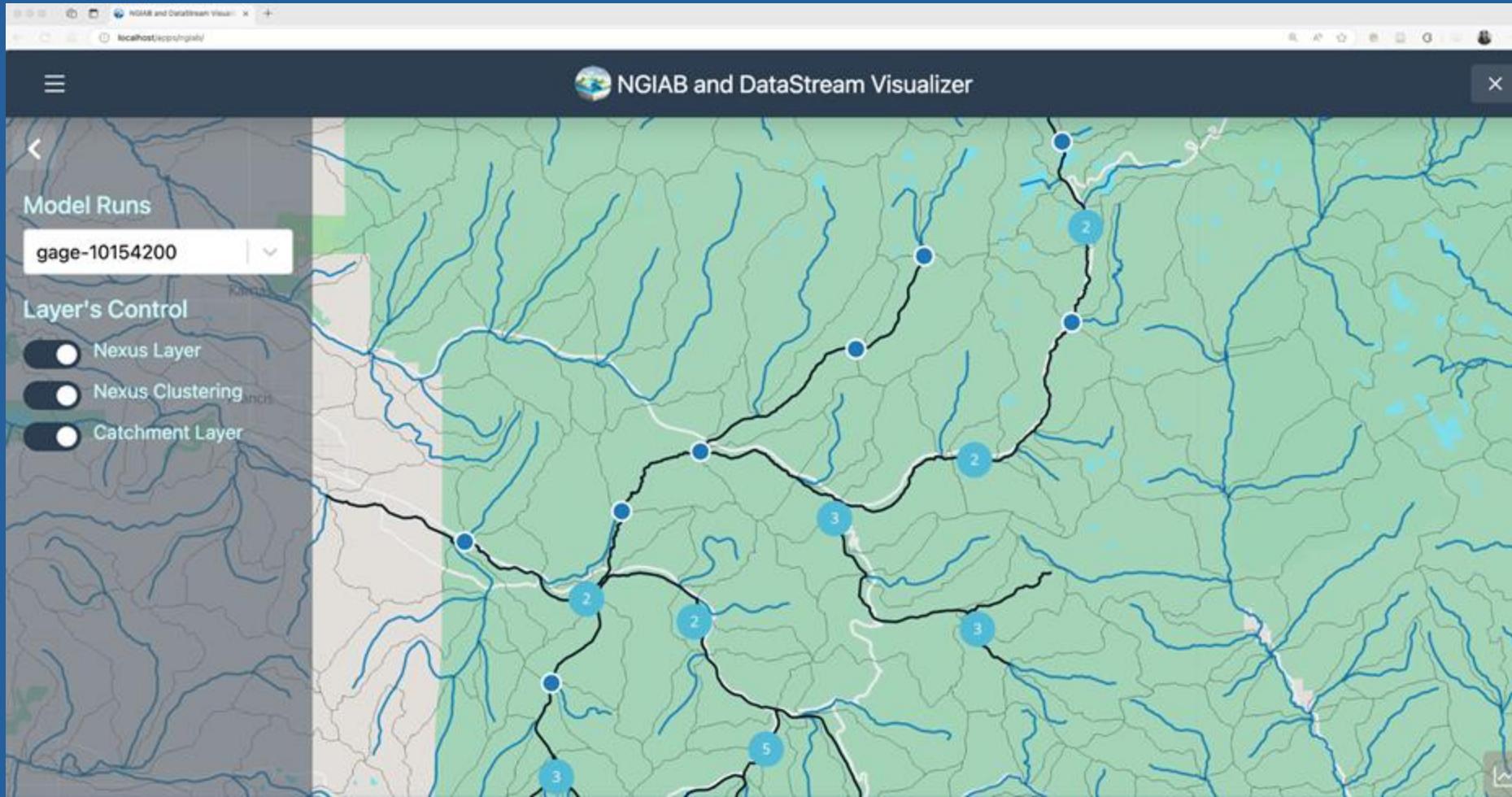


Enter id/pwd:
admin/pass

Copyright © 2025 Your Organization

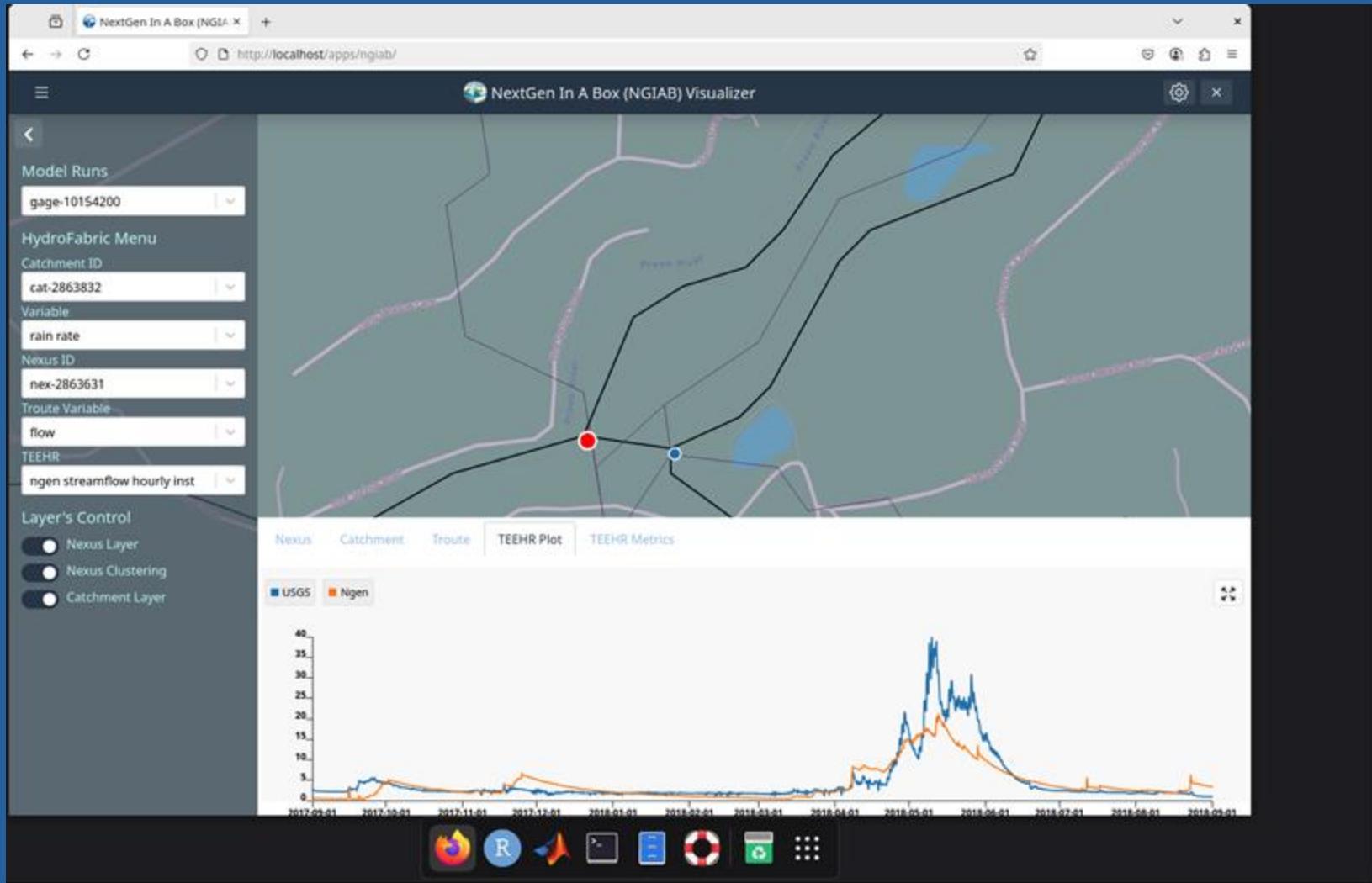
Powered by Tethys Platform

NGIAB Tethys Visualizer



Select
gage-
10154200 from
the dropdown

NGIAB Visualizer



Select catchment ID:
cat-2863832

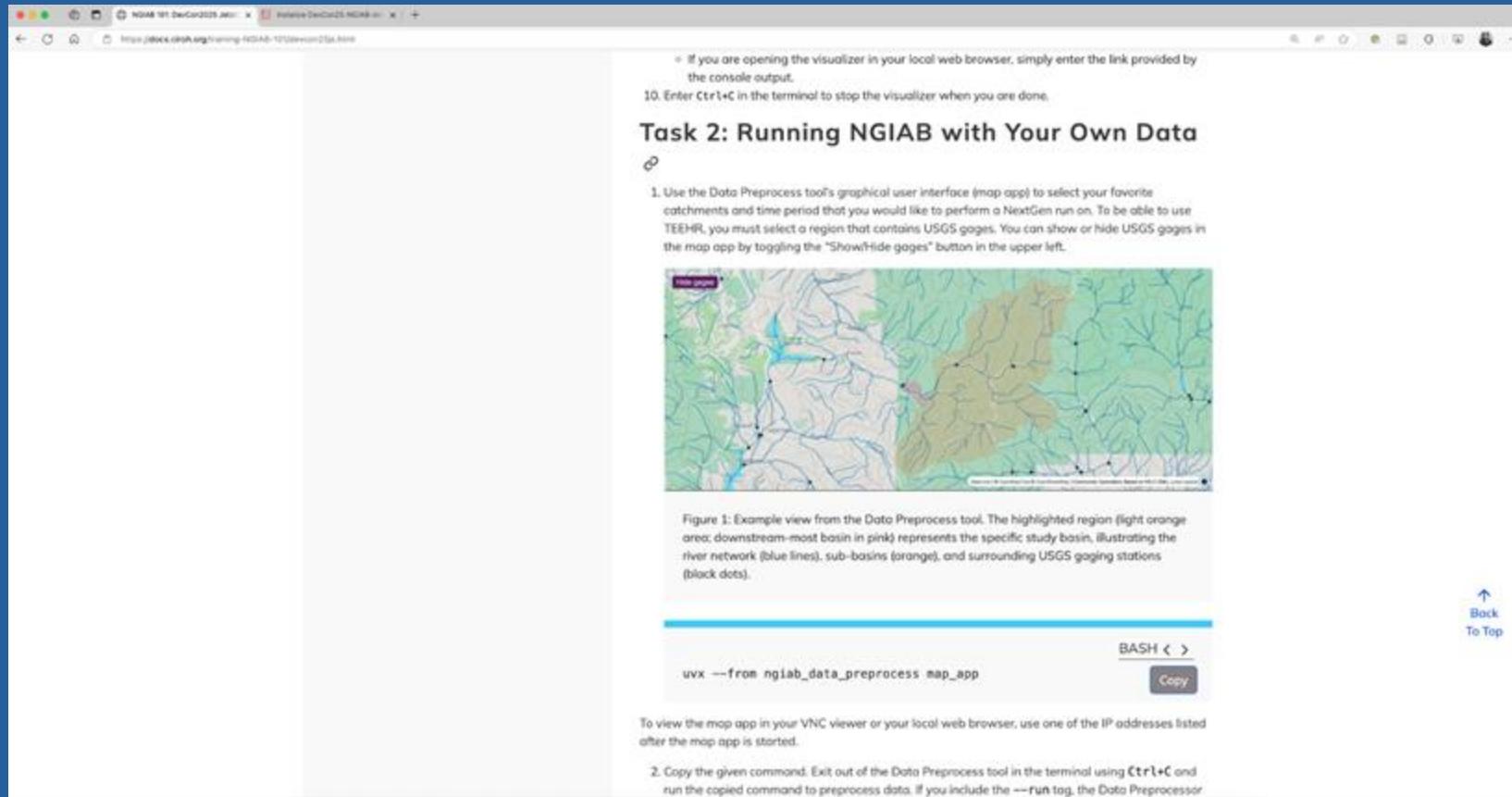
Variable: rain rate

Nexus ID: **nex-2863631**

Troute Variable:
flow

TEEHR:
**ngen streamflow
hourly inst**

Task 2



The screenshot shows a web browser window with the URL <https://devc.ciroh.org/training-NGIAB-101/Version25a.html>. The page title is "Task 2: Running NGIAB with Your Own Data". It includes instructions for opening the visualizer in a browser and stopping it with Ctrl+C. Below this, a section titled "Task 2: Running NGIAB with Your Own Data" contains a map of a river network with highlighted regions and gaging stations. A caption describes the map. At the bottom, there is a terminal command in a BASH session and a note about viewing the map app.

If you are opening the visualizer in your local web browser, simply enter the link provided by the console output.
10. Enter **Ctrl+C** in the terminal to stop the visualizer when you are done.

Task 2: Running NGIAB with Your Own Data

1. Use the Data Preprocess tool's graphical user interface (map app) to select your favorite catchments and time period that you would like to perform a NextGen run on. To be able to use TEHR, you must select a region that contains USGS gages. You can show or hide USGS gages in the map app by toggling the "Show/Hide gages" button in the upper left.



Figure 1: Example view from the Data Preprocess tool. The highlighted region (light orange area; downstream-most basin in pink) represents the specific study basin, illustrating the river network (blue lines), sub-basins (orange), and surrounding USGS gaging stations (black dots).

BASH < >
`uvx --from ngiab_data_preprocess map_app` Copy

To view the map app in your VNC viewer or your local web browser, use one of the IP addresses listed after the map app is started.

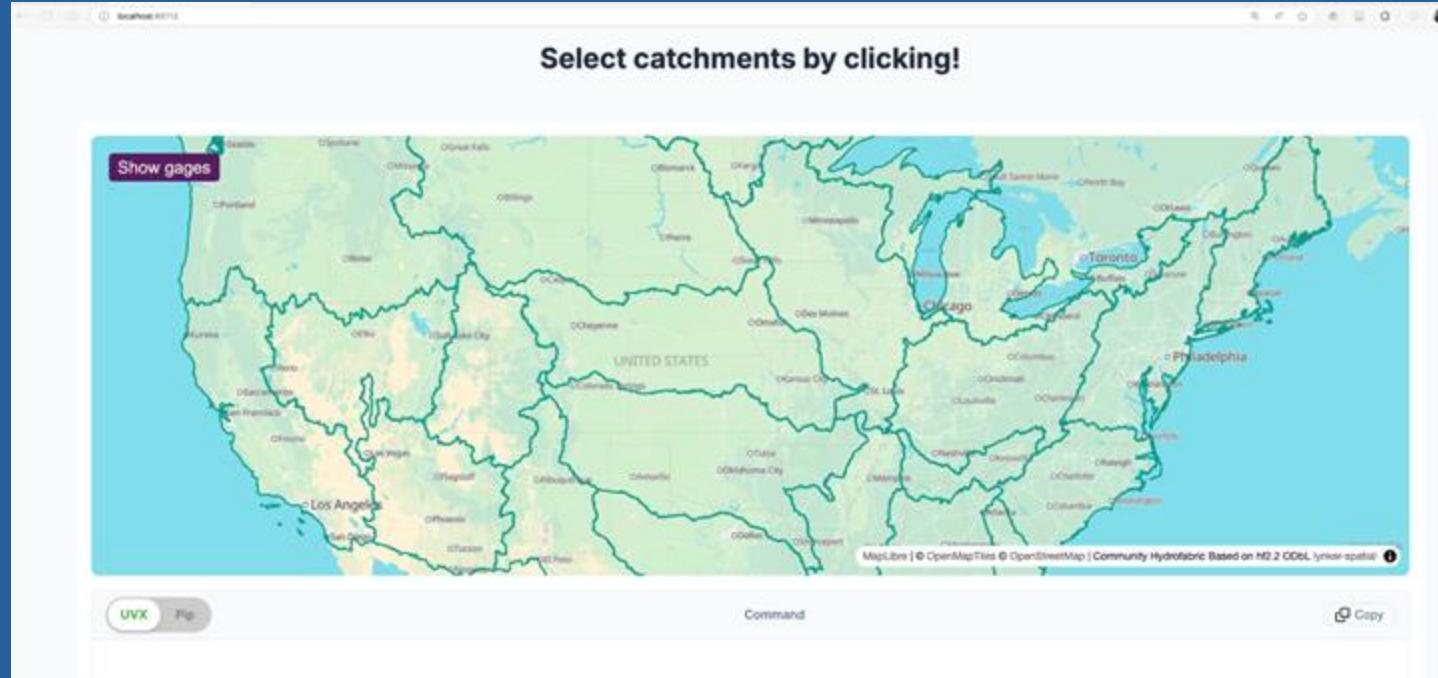
2. Copy the given command. Exit out of the Data Preprocess tool in the terminal using **Ctrl+C** and run the copied command to preprocess data. If you include the **--run** tag, the Data Preprocessor

For Task 2:

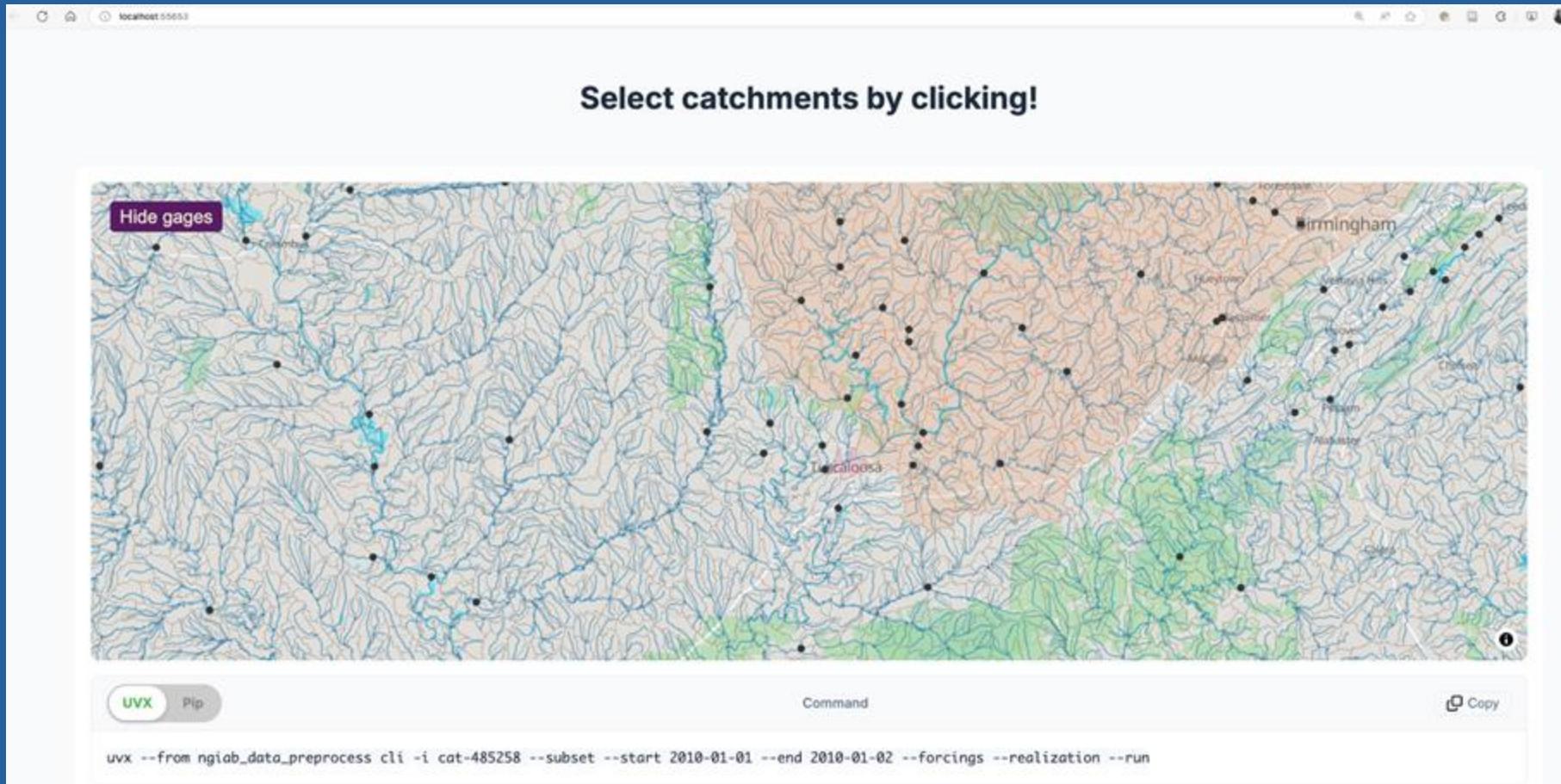
Ctrl+C - to stop the existing visualizer

Data Preprocess Tool Map

```
~/NGIAB_demo/NGIAB-CloudInfra git:(main)
uvx --from ngiab_data_preprocess map_app
 * Serving Flask app 'map_app'
 * Debug mode: off
werkzeug : INFO     WARNING: This is a development server. Do not use it in a production deployment. Use a production W
SGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:65112
 * Running on http://192.168.1.61:65112
werkzeug : INFO     Press CTRL+C to quit
```



NGIAB Data Preprocess GUI (map app)

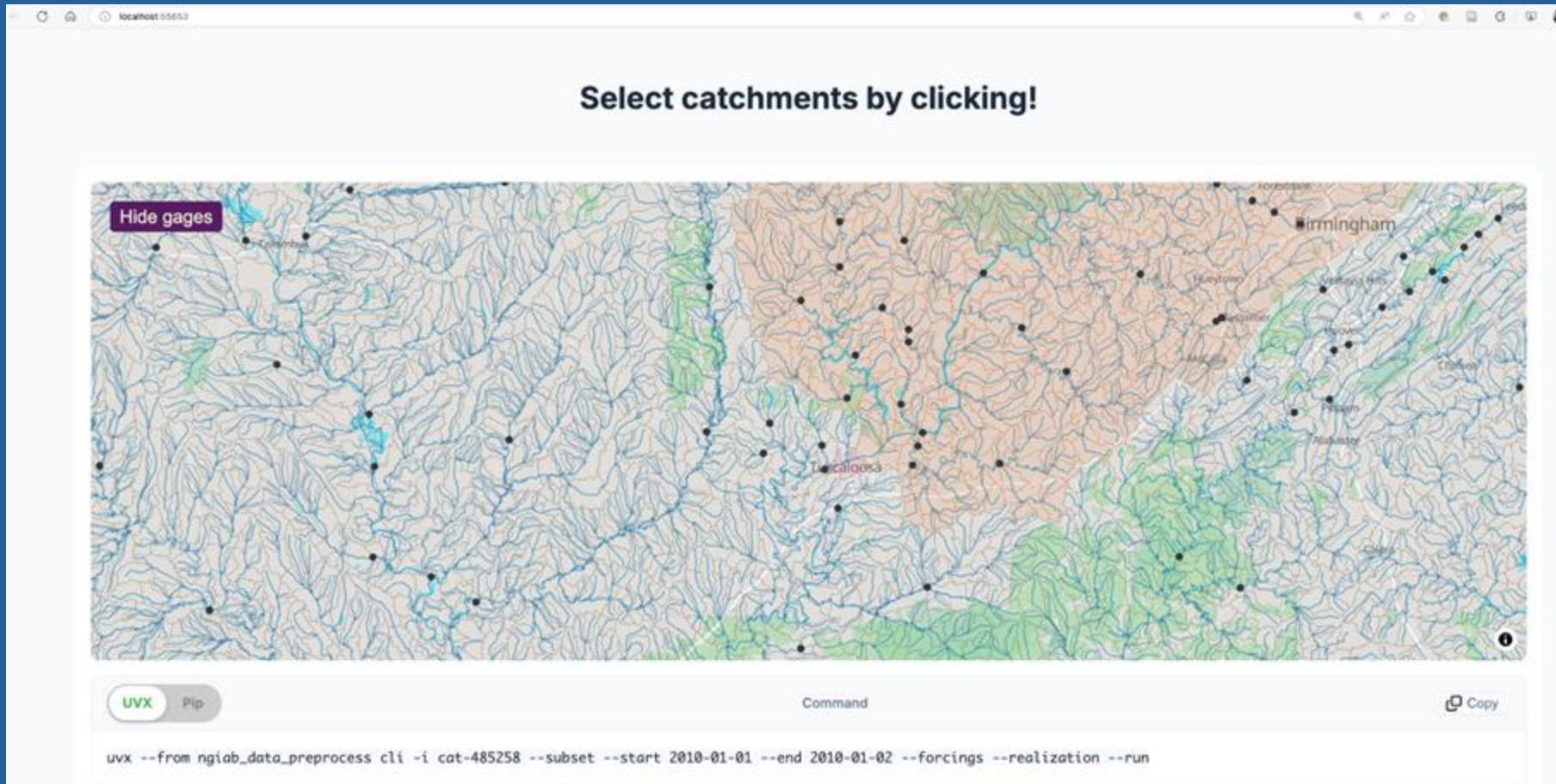


Click on Show gages
button to see the
gages!

The highlighted region (light orange area);
downstream-most basin in pink) represents the specific
study basin illustrating the river network (blue lines)
sub-basins (orange)
surrounding USGS gaging stations (black dots)

Go to select
basin near
Tuscaloosa, AL
to perform a
NextGen run
on.

NGIAB Data Preprocess GUI (map app)



If you include
--run tag, the
Data
Preprocess will
automatically
execute
NextGen using
NGIAB

Copy the
command
and using
Ctrl+C run on
terminal

NGIAB Data Preprocess using uvx and --run

```
~/NGIAB_demo/NGIAB-CloudInfra git:(main)±1
uvx --from ngiab_data_preprocess cli -i cat-485258 --subset --start 2010-01-01 --end 2010-01-02 --forcings --realization --run

2025-05-24 13:01:17,372 - INFO - Running all missing steps required to run ngiab.
2025-05-24 13:01:17,373 - INFO - Processing cat-485258 in /Users/apatel54/ngiab_preprocess_output/cat-485258
2025-05-24 13:01:17,759 - INFO - Upstream catchments: 1625
2025-05-24 13:01:17,759 - INFO - Subsetting hydrofabric
2025-05-24 13:01:17,877 - INFO - Subsetting tables: ['divides', 'divide-attributes', 'flowpath-attributes', 'flowpath-attributes-ml', 'flowpaths', 'hydrolocations', 'nexus', 'pois', 'lakes', 'network']
2025-05-24 13:01:20,567 - INFO - Subset complete for 4038 features (catchments + nexuses)
2025-05-24 13:01:20,568 - INFO - Subsetting complete.
2025-05-24 13:01:20,568 - INFO - Generating forcings from 2010-01-01 00:00:00 to 2010-01-02 00:00:00...
```

```
2025-05-24 18:02:17,783 - root - INFO - [output.py:180 - nwm_output_generator]: Handling output ...
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:340 - main_v04]: **** ***** TIMING SUMMARY **** ****
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:341 - main_v04]: -----
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:342 - main_v04]: Network graph construction: 0.39 secs, 18.89 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:349 - main_v04]: Forcing array construction: 1.18 secs, 56.3 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:356 - main_v04]: Routing computations: 0.5 secs, 24.08 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:363 - main_v04]: Output writing: 0.01 secs, 0.62 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:370 - main_v04]: -----
2025-05-24 18:02:17,797 - root - INFO - [__main__.py:371 - main_v04]: Total execution time: 2.0799999999999996 secs
Finished routing
NGen top-level timings:
    NGen::init: 7.13841
    NGen::simulation: 1.17989
    NGen::routing: 2.10485
Run completed successfully, exiting, have a nice day!
2025-05-24 13:02:18,319 - INFO - Next Gen run complete.
2025-05-24 13:02:18,320 - INFO - All operations completed successfully.
2025-05-24 13:02:18,320 - INFO - Output folder: file:///Users/apatel54/ngiab_preprocess_output/cat-485258
```

One command
to RUN
NextGen using
NGIAB!!!

If you use
--run tag, you
will need to
run the
[runTeehr.sh](#)
and
[viewonTethys.s](#)
h scripts
separately.

NGIAB Data Preprocess using uvx

```
2025-05-24 18:02:17,783 - root - INFO - [output.py:180 - nwm_output_generator]: Handling output ...
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:340 - main_v04]: ***** TIMING SUMMARY *****
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:341 - main_v04]: -----
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:342 - main_v04]: Network graph construction: 0.39 secs, 18.89 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:349 - main_v04]: Forcing array construction: 1.18 secs, 56.3 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:356 - main_v04]: Routing computations: 0.5 secs, 24.08 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:363 - main_v04]: Output writing: 0.01 secs, 0.62 %
2025-05-24 18:02:17,796 - root - INFO - [__main__.py:370 - main_v04]: -----
2025-05-24 18:02:17,797 - root - INFO - [__main__.py:371 - main_v04]: Total execution time: 2.0799999999999996 secs
Finished routing
NGen top-level timings:
    NGen::init: 7.13841
    NGen::simulation: 1.17989
    NGen::routing: 2.10485
Run completed successfully, exiting, have a nice day!
2025-05-24 13:02:18,319 - INFO - Next Gen run complete.
2025-05-24 13:02:18,320 - INFO - All operations completed successfully.
2025-05-24 13:02:18,320 - INFO - Output folder: file:///Users/apatel54/ngiab_preprocess_output/cat-485258
```

**Copy the
output folder
for the next
run**

NGIAB TEEHR evaluation using runTeehr.sh

CIROH: NextGen In A Box (NGIAB) - TEEHR Evaluation
Model Performance Assessment Tool

- ℹ Developed by CIROH
- ℹ Last used data directory: /Users/apatel54/ngiab_preprocess_output/cat-484072
 - Use this path? [Y/n]: n
 - Enter your input data directory path: /Users/apatel54/ngiab_preprocess_output/cat-485258
 - ✓ Path saved for future use

TEEHR EVALUATION SETUP

- ℹ TEEHR will evaluate model outputs against observations
 - Learn more: <https://rtiinternational.github.io/ngiab-teehr/>
 - Would you like to run a TEEHR evaluation on your model outputs?
- Run evaluation? [Y/n]: Y

- Specify TEEHR image tag [default: latest]:
- ✓ Using default tag: latest

CONTAINER MANAGEMENT

- Select an option:
- 1) Run TEEHR using existing local image 3) Exit
- 2) Update to latest TEEHR image

#? 1

Run
[./runTeehr.sh](#)

Enter “n”
and enter the
correct path to
use from
previous run
Enter Y to run
evaluation

Enter 1 to run
evaluation

NGIAB TEEHR evaluation using runTeehr.sh

```
EVALUATION COMPLETE

TEEHR evaluation completed successfully!

ℹ Results have been saved to your outputs directory:
→ /Users/apatel54/ngiab_preprocess_output/cat-485258/outputs/teehr/

ℹ You can visualize these results using the Tethys platform
→ Run ./viewOnTethys.sh /Users/apatel54/ngiab\_preprocess\_output/cat-485258 to start visualization

Thank you for using NGIAB!
ℹ For support, please email: ciroh-it-support@ua.edu

→ Cleaning up resources...
✓ Cleanup completed

→ Cleaning up resources...
✓ Cleanup completed
```

NGIAB Tethys Visualizer using viewonTethys.sh

```
~/NGIAB_demo/NGIAB-CloudInfra git:(main)±1  
./viewOnTethys.sh
```

```
=====  
| CIROH: NextGen In A Box (NGIAB) - Tethys |  
| Interactive Model Output Visualization |  
=====  
: Developed by CIROH  
:  
: Last used data directory: /Users/apatel54/ngiab_preprocess_output/cat-485258  
→ Use this path? [Y/n]: Y
```

NGIAB Tethys Visualizer using viewonTethys.sh

```
i Last used data directory: /Users/apatel54/ngiab_preprocess_output/cat-485258
→ Use this path? [Y/n]: Y
✓ Using previously configured path
✓ Path saved for future use.
```

Enter K

PREPARING VISUALIZATION ENVIRONMENT

```
⚠ /Users/apatel54/ngiab_visualizer is not empty.
```

```
→ Keep (K) or Fresh start (F)? [K/F]: K
```

```
⚠ /Users/apatel54/ngiab_visualizer is not empty.
```

```
→ Keep (K) or Fresh start (F)? [K/F]: K
```

```
✓ Copied → /Users/apatel54/ngiab_visualizer/cat-485258
```

```
Checking for /Users/apatel54/ngiab_visualizer/ngiab_visualizer.json...
```

Enter K

```
✓ Model run " registered (3AB164B5-83DB-4E70-8E7D-A0A8227E348D)
```

```
i Existing Datastream cache detected: /Users/apatel54/.datastream_ngiab
```

```
Keeping it avoids re-downloading archives, but a large cache
```

```
can slow the first container start-up depending on your system.
```

```
→ Keep cache (K) or Fresh start (F)? [K/F]: K
```

NGIAB Tethys Visualizer using viewonTethys.sh

LAUNCHING TETHYS VISUALIZATION

Specify the Tethys image tag to use:

- Tag (e.g. v0.2.1, default: latest):
- i Found local image **awiciroh/tethys-ngiab:latest**
- Use local copy (L) or Pull latest from registry (P)? [L/P]: L

Enter L

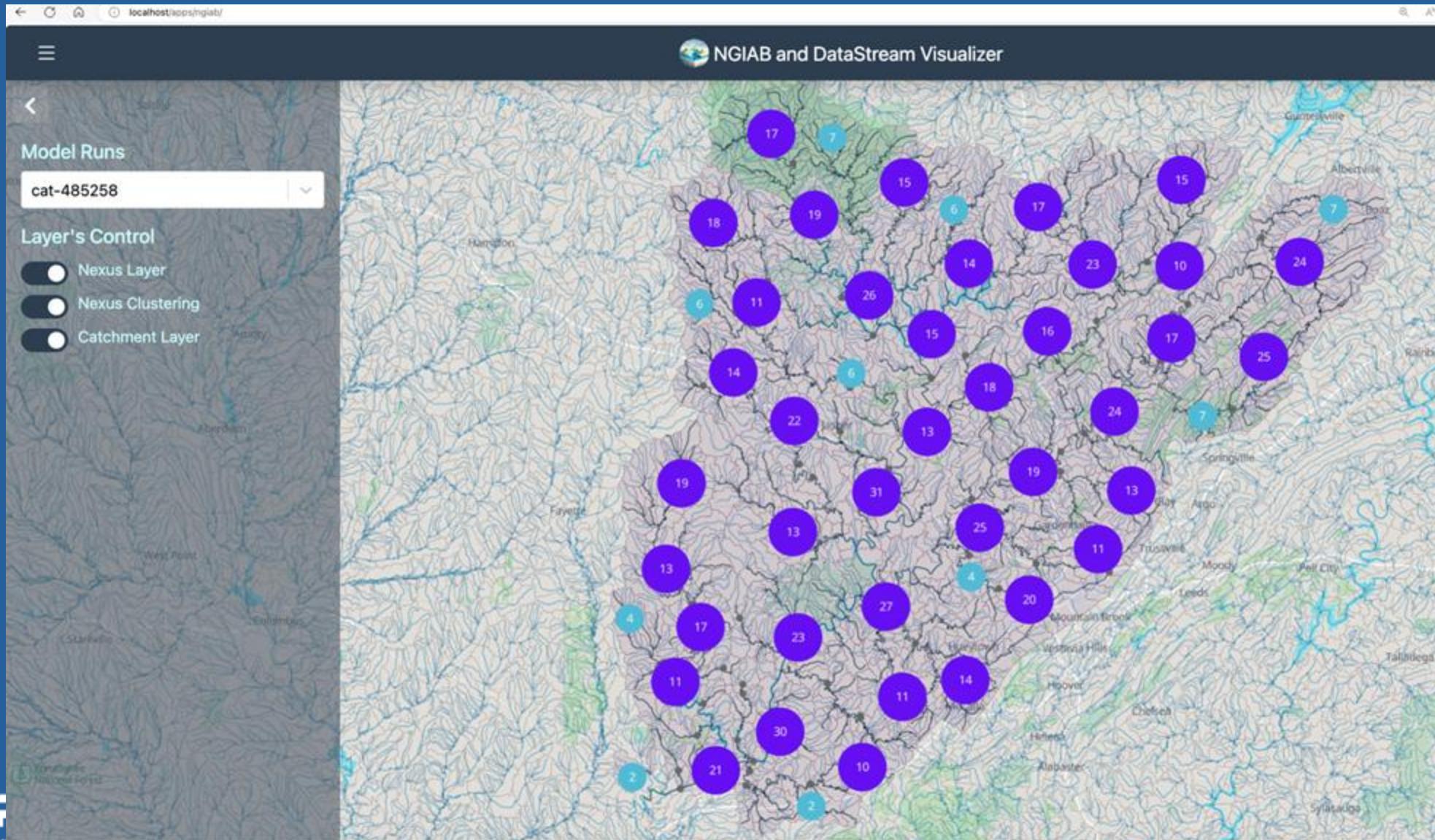
```
✓ Using local image
Select a port to run Tethys on. [Default: 80]
→ Port:
→ Using default port 80 for Tethys.
✓ Port 80 selected
→ Launching Tethys container...
  i Tethys container is already running. Stopping it first...
  △ Forcibly removing container...
i Setting up Docker network for Tethys...
  ✓ Network tethys-network created successfully.
  i Starting Tethys container...
  i Running docker command...
c8441d4fc31ae8866ecc51ce462fd068c00ac3bf9a805b55fd68d6e5dad80769
  ✓ Tethys container started successfully.
i Waiting for container: tethys-ngen-portal to become healthy. This can take a couple of minutes...
```

NGIAB Tethys Visualizer using viewonTethys.sh

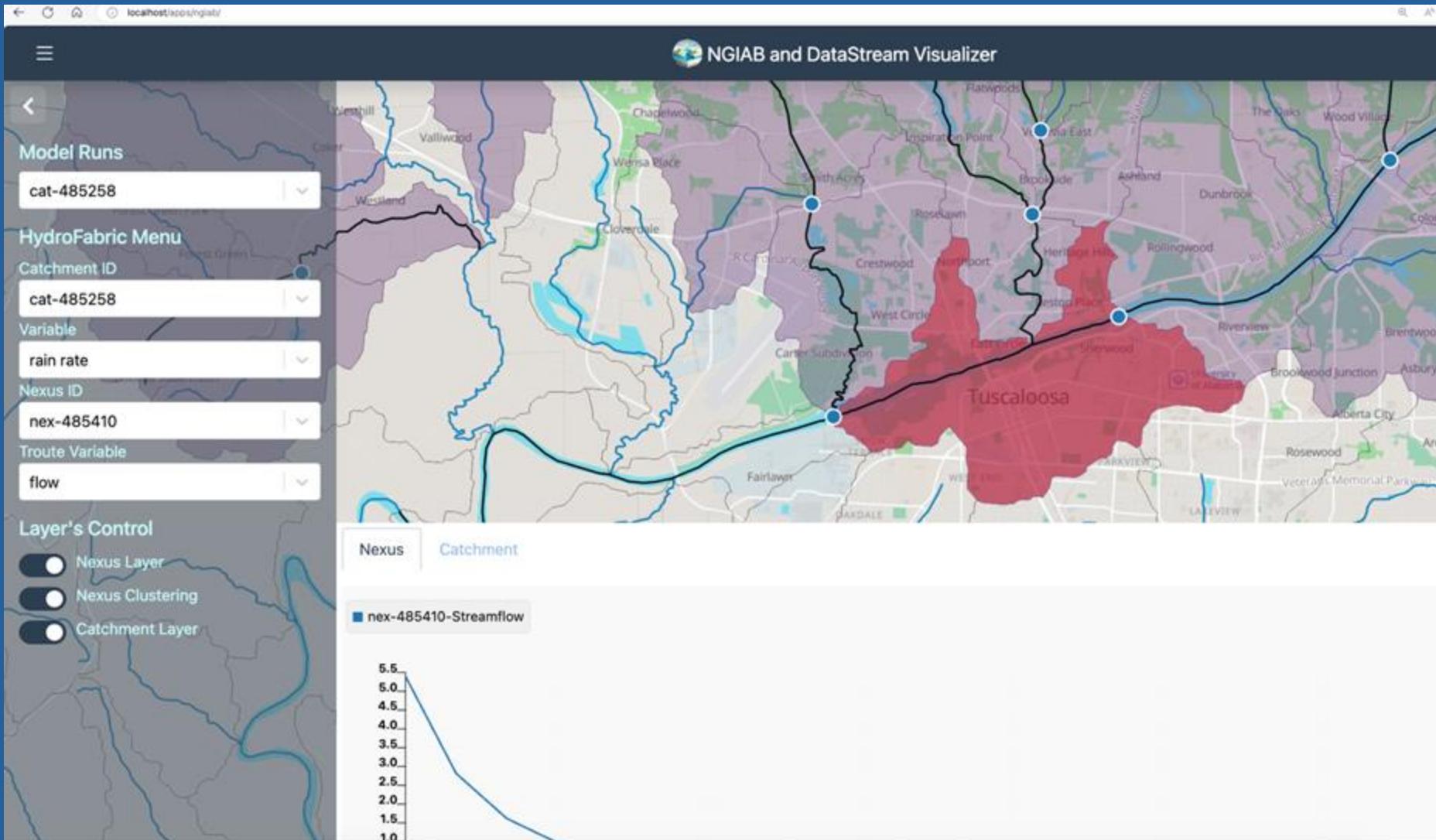
```
✓ Container tethys-ngen-portal is now healthy!  
  
VISUALIZATION READY  
  
Your model outputs are now available for visualization!  
  
ℹ Access the visualization at: http://localhost:80/apps/ngiab  
ℹ Login credentials:  
→ Username: admin  
→ Password: pass  
  
ℹ Source code: https://github.com/CIROH-UA/ngiab-client  
  
Tethys is now running  
ℹ Access the visualization at: http://localhost:80/apps/ngiab  
ℹ Press Ctrl+C to stop Tethys when you're done.
```

Open URL in browser and enter admin and pass to login!

NGIAB Tethys Visualizer using viewonTethys.sh



NGIAB Tethys Visualizer using viewonTethys.sh

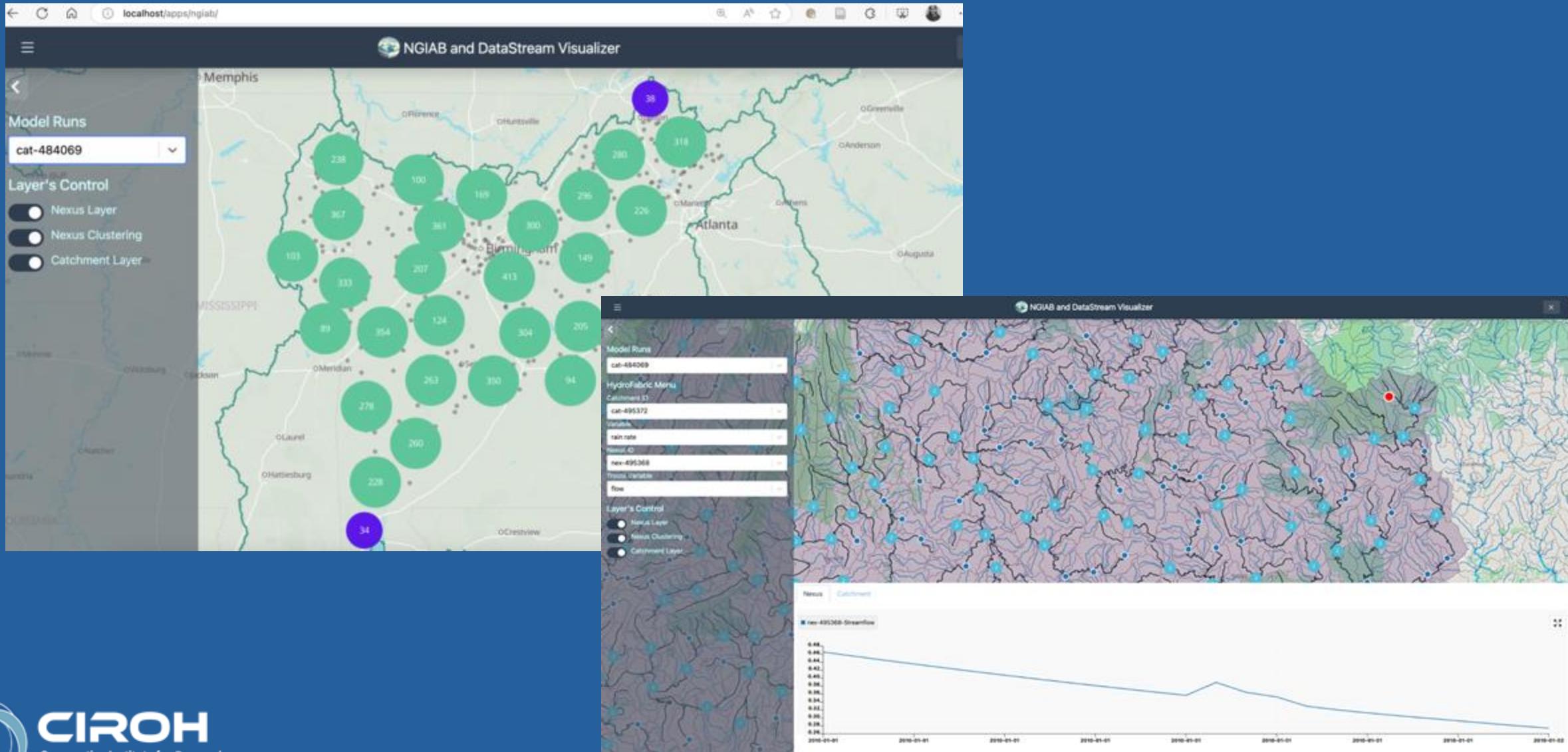


Another example: cat-484069

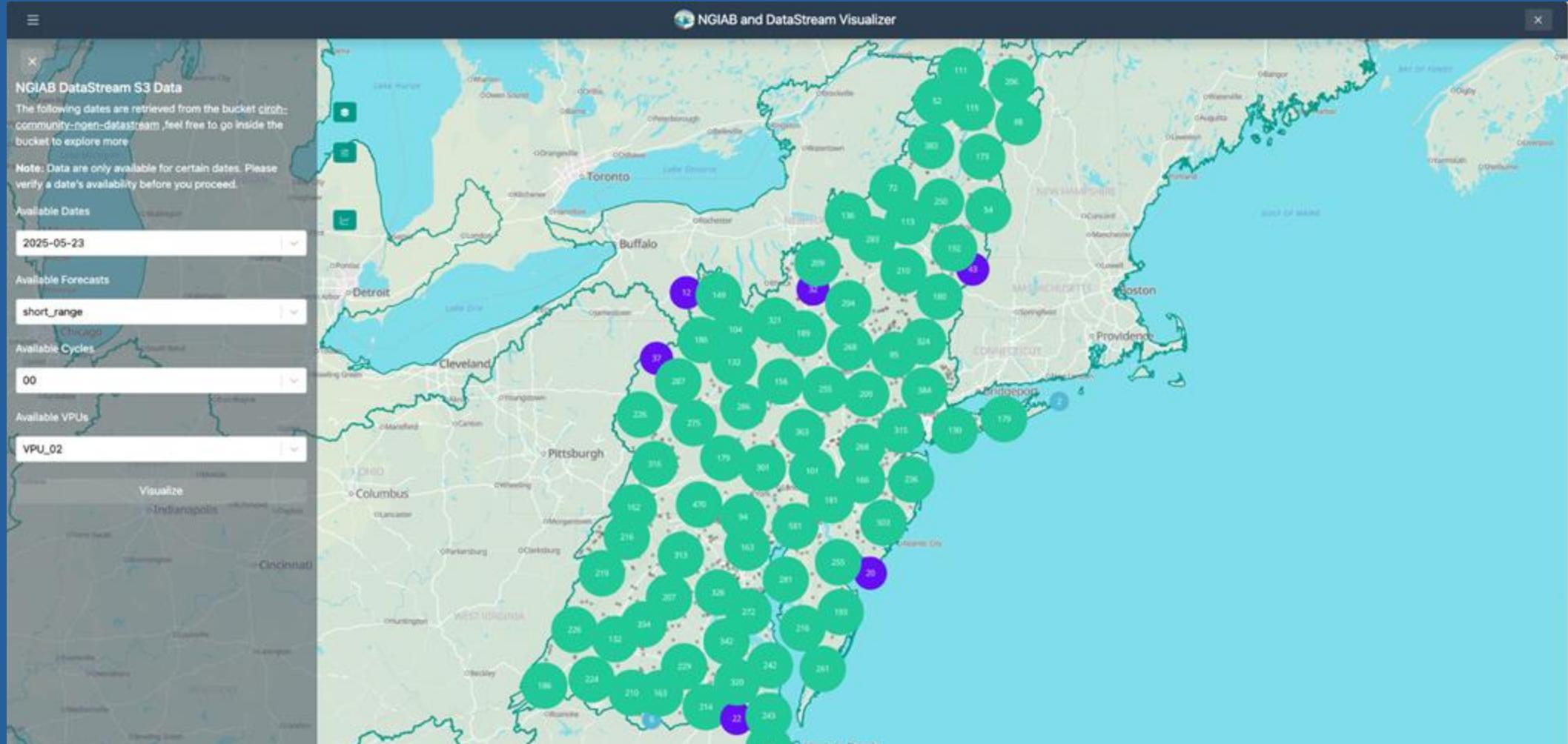


Run cmd: uvx --from nglab_data_preprocess cli -i cat-484069 --subset --start 2010-01-01 --end 2010-01-02 --forcings --realization --run

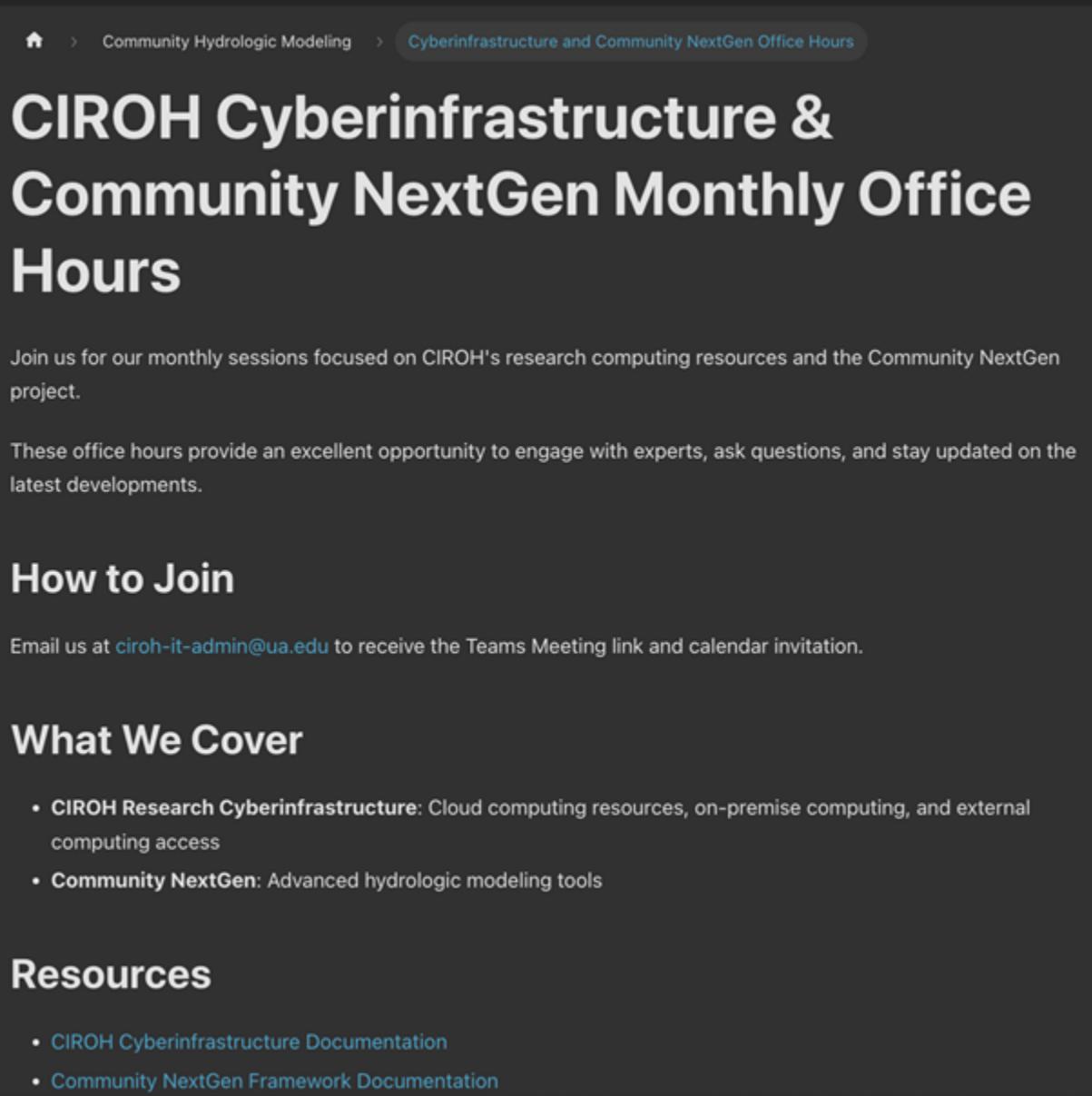
NGIAB Tethys Visualizer - cat-484069



DataStream Visualizer



CIROH Community Office Hours



The screenshot shows a dark-themed webpage for 'Community Hydrologic Modeling' under 'Cyberinfrastructure and Community NextGen Office Hours'. The main title is 'CIROH Cyberinfrastructure & Community NextGen Monthly Office Hours'. Below the title, a paragraph invites users to join monthly sessions focused on CIROH's research computing resources and the Community NextGen project. It highlights the opportunity to engage with experts, ask questions, and stay updated on the latest developments. A 'How to Join' section provides instructions to email 'ciroh-it-admin@ua.edu' for meeting links and calendar invitations. A 'What We Cover' section lists topics including CIROH Research Cyberinfrastructure and Community NextGen modeling tools. A 'Resources' section links to documentation for both.

Community Hydrologic Modeling > Cyberinfrastructure and Community NextGen Office Hours

CIROH Cyberinfrastructure & Community NextGen Monthly Office Hours

Join us for our monthly sessions focused on CIROH's research computing resources and the Community NextGen project.

These office hours provide an excellent opportunity to engage with experts, ask questions, and stay updated on the latest developments.

How to Join

Email us at ciroh-it-admin@ua.edu to receive the Teams Meeting link and calendar invitation.

What We Cover

- CIROH Research Cyberinfrastructure: Cloud computing resources, on-premise computing, and external computing access
- Community NextGen: Advanced hydrologic modeling tools

Resources

- CIROH Cyberinfrastructure Documentation
- Community NextGen Framework Documentation

Calibration News!

DocuHub Products Services Community Impact Learn ▾

Blog News Release Notes Portal 🔍 ⌂

Community NextGen Updates

Stay connected with the latest developments in NextGen water modeling! This news hub brings you updates, breakthroughs, and opportunities from across our community of practice.

Discover how researchers and practitioners are applying NextGen frameworks to solve pressing water challenges, learn about upcoming training events, and explore new resources to enhance your modeling workflow. Our community-driven approach ensures you'll always be informed about the innovations that matter most.

Click to collapse



News

April 2025 Updates

feature New NGIAB-Calibration Feature 🎉

Major update to NextGen In A Box! It now supports extended calibration for CFE and NOAA OWP modules. The new calibration framework provides more flexible parameter tuning and improved model performance.

Uses ngen-cal branch: https://github.com/CIROH-UA/ngen-cal/tree/ngiab_cal

For detailed instructions on how to use the new calibration capabilities, please check out: https://github.com/CIROH-UA/ngen-cal/tree/ngiab_cal#how-to-use-this

CIROH Research Cyberinfrastructure

supporting Community NextGen

Google Cloud & AWS Sponsored \$5000 for
DevCon2025

The screenshot shows the homepage of the CIROH Research Cyberinfrastructure. At the top left is the University of Alabama logo and the Alabama Water Institute logo. To the right is the CIROH logo with the full name "Cooperative Institute for Research to Operations in Hydrology". Below these is a large circular icon containing a 3D rendering of a city skyline with water droplets and various icons. The main title "RESEARCH CYBERINFRASTRUCTURE" is centered above a detailed description of the service. The description highlights how the CIROH CyberInfrastructure and DevOps team empowers the consortium by providing a scalable, efficient, and user-friendly computing platform. It mentions the challenges researchers face in managing computational resources and how the CIROH CyberInfrastructure alleviates these burdens by offering a suite of pre-configured environments and resources for hydrology research. The text also notes that the team optimizes both cloud-based (AWS, GCP and CIROH-2i2c JupyterHub) and on-premise infrastructure (Pantarhei, Wukong and OpenStack Sandbox) to ensure unparalleled flexibility and scalability. Below this is a "CONTACT US" section with links to the CIROH website, DocuHub, Admin Email, Slack support, and Office Hours. It also features QR codes and the NOAA logo. On the left side, there is a list of the CyberInfrastructure & DevOps Team members and their roles.

CyberInfrastructure & DevOps Team

Arpita Patel
DevOps Manager & Enterprise Architect

James Halgren
Assistant Director of Science

Benjamin Lee
DevOps Engineer

Trupesh Patel
Software Developer II

Manjila Singh
Graduate Student

Office of Information Technology
UA

RESEARCH CYBERINFRASTRUCTURE

The CIROH CyberInfrastructure and DevOps team empowers CIROH consortium by providing a scalable, efficient, and user-friendly computing platform. We understand the challenges researchers face in managing computational resources, and the CIROH CyberInfrastructure alleviates these burdens by offering a suite of pre-configured environments and resources for hydrology research. Our team of engineers, researchers and students meticulously optimizes both cloud-based (AWS, GCP and CIROH-2i2c JupyterHub) and on-premise infrastructure (Pantarhei, Wukong and OpenStack Sandbox) to ensure unparalleled flexibility and scalability.

CONTACT US

CIROH Website: ciroh.ua.edu
CIROH DocuHub: docs.ciroh.org
Admin Email: ciroh-it-admin@ua.edu
Slack support: [cirohworkspace](#)
Office Hours: Monthly

The graphic is a promotional slide for the CIROH Research Cyberinfrastructure. It features a large blue circular icon with a water drop and various scientific symbols. To the right of this icon is a white circle containing the AWS logo. The text "POWERING WATER SCIENCE WITH COMPUTING INNOVATION" is written in a bold, sans-serif font. Below this is another white circle containing the Google Cloud and 2i2c logos. The text "ACCELERATING SCIENTIFIC RESEARCH EXPONENTIALLY" is written in a smaller font. At the bottom right, it says "Supports 400+ Active Researchers 130 Research Projects". The text "ENHANCED COLLABORATION CAPABILITIES" is also present. The background of the slide features a dark blue circuit board pattern.

List of Services

- Scalable and efficient computing platform
- Easy access to cloud-based and on-premise VM and HPC resources
- Unparalleled flexibility for diverse research needs

Partners

- AWS
- Google Cloud
- 2i2c

POWERING WATER SCIENCE WITH COMPUTING INNOVATION

ACCELERATING SCIENTIFIC RESEARCH EXPONENTIALLY

Supports
400+ Active Researchers
130 Research Projects

ENHANCED COLLABORATION CAPABILITIES

Community Cyberinfrastructure supporting Community NextGen

More than 74
Infrastructure
request
supported!

CIROH-UA / NGIAB-CloudInfra

Type to search

Issues 13

Code Pull requests Discussions Actions Projects Wiki Security Insights Settings

is:issue state:open label:"R2OHC Resource request"

Open 8 Closed 74

Author Labels Projects Milestones Assignees Types Newest

Workshop Resource Request: [USGS Python Data Tools] 2i2c JupyterHub DevCon2025 hydroinformatics R2OHC Resource request workshop #292 · shamshaw opened yesterday

Workshop Resource Request: [SHAP] 2i2c JupyterHub DevCon2025 Machine Learning R2OHC Resource request workshop #291 · klunderw-uvm opened 5 days ago

Workshop Resource Request: [TEVA] 2i2c JupyterHub DevCon2025 Machine Learning R2OHC Resource request workshop #290 · klunderw-uvm opened 5 days ago

Workshop Resource Request: [CIROH DevCon 2025 Water Quality Forecasts] 2i2c JupyterHub DevCon2025 Machine Learning R2OHC Resource request workshop #284 · jtkemper opened 2 weeks ago

Workshop Resource Request: [Community NextGen Track Workshop] DevCon2025 NextGen NSF Jetstream R2OHC Resource request workshop #281 · arpita0911patel opened 3 weeks ago

Workshop Resource Request: [Machine Learning for Post-Processing Hydrological Model Outputs] 2i2c JupyterHub DevCon2025 Machine Learning R2OHC Resource request workshop #279 · savalann opened last month

Workshop Resource Request: flows2fim (CIROH Dev Con) aws DevCon2025 FIM R2OHC Resource request workshop #277 · arpitadavidson opened on Apr 9

First paper using NGIAB!!!

δ HBV2.0 + NGIAB

Differentiable Modeling in Operations

Leo Lonzarich

Tadd Bindas, Farshid Rahmani, Yalan Song, Haoyu Ji, Jiangtao Liu, Doaa Aboelyazeed, Jonathan Frame, *Chaopeng Shen (Advisor)*

The screenshot shows a blog post on the DocuHub platform. The title is "δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling". The post is dated May 16, 2025, and has a reading time of 2 minutes. It features four authors: Yalan Song (Research Assistant Professor), Leo Lonzarich (Graduate Researcher), Arpita Patel (DevOps Manager and Enterprise Architect), and James Halgren (Assistant Director of Science). The post includes a graph comparing model efficiency (NSE) and a map of the United States showing hydrological data. Below the main content, there are sections for 2025 and 2024, each listing various research and development activities.

DocuHub Blog

Exclusive content for researchers utilizing CIROH cyberinfrastructure resources. Share your insights, discoveries, and experiences with the hydrologic science community.

This blog platform is dedicated to highlighting the innovative work of researchers who have leveraged CIROH's computational tools and resources to advance water science. Your stories help demonstrate the value of our shared infrastructure and inspire new applications across the field.

2025

- δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling
- Google Cloud Next 2025: Innovation at Scale

2024

- UA's Alabama Water Institute Showcases 30-Minute Hydrological Modeling Revolution
- Pennsylvania State University Researchers Leverage CIROH Cyberinfrastructure for Advanced Hydrological Modeling
- CIROH at AGU 2024
- Community NextGen Updates
- CIROH Science Meeting 2024
- Accessing National Water Model (NWM) Data via Google Cloud BigQuery API
- CIROH Cloud User Success Story
- CIROH Research Cyberinfrastructure Update
- CIROH Developers Conference 2024

δHBV2.0: How NGIAB and Wukong HPC Streamlined Advanced Hydrologic Modeling

May 16, 2025 · 2 min read

Yalan Song
Research Assistant Professor

Leo Lonzarich
Graduate Researcher

Arpita Patel
DevOps Manager and Enterprise Architect

James Halgren
Assistant Director of Science

(a) Cumulative Distribution Function vs Nash-Sutcliffe Model Efficiency (NSE)

(b) Map of the United States showing hydrological data

(c) Map of the United States showing hydrological data

(d) Map of the United States showing hydrological data

Predicting water flow with precision across the vast U.S. landscape is a complex challenge. That's why Song et al. 2024 developed δHBV2.0, a cutting-edge hydrologic model. It's built with high-

Community NextGen Repos

Source code at:

- <https://github.com/CIROH-UA/NGIAB-CloudInfra>
- <https://github.com/CIROH-UA/ngen>
- <https://github.com/CIROH-UA/t-route>
- <https://github.com/CIROH-UA/lstm>
- <https://github.com/CIROH-UA/NGIAB-HPCInfra>
- https://github.com/CIROH-UA/NGIAB_data_preprocess
- <https://github.com/CIROH-UA/ngen-datastream>
- <https://github.com/CIROH-UA/ngiab-teehr>
- <https://github.com/CIROH-UA/ngiab-client>
- <https://github.com/CIROH-UA/ngiab-cal>



Q&A

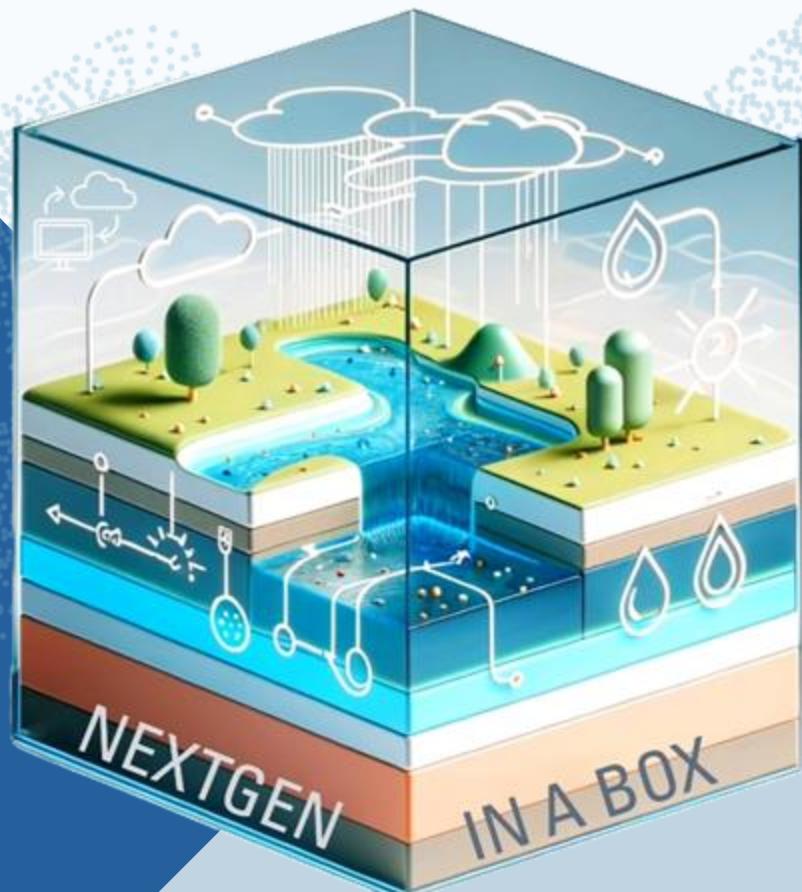


CIROH
Cooperative Institute for Research
to Operations in Hydrology

THE UNIVERSITY OF
ALABAMA®

Alabama Water
Institute

Thank You!



Acknowledgement

This research was supported by the Cooperative Institute for Research to Operations in Hydrology (CIROH) with funding under award NA22NWS4320003 from NOAA Cooperative Institute Program. The statements, findings, conclusions, and recommendations are those of the author(s) and do not necessarily reflect the opinions of NOAA.