

This work is supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics, under contract DE-AC02-06CH11357, and Argonne's LDRD program.

SOFTWARE R&D FUNDED BY ARGONNE LDRD (ULAB)



AUTOMATED WORKFLOWS FOR DETECTOR DESIGN



SYLVESTER JOOSTEN

sjoosten@anl.gov

On behalf of the Argonne EIC LDRD team

WHITNEY ARMSTRONG

warmstrong@anl.gov

A NEW DETECTOR FROM SCRATCH?

From the EIC Yellow Report to an optimized EIC detector

η	Nomenclature	Tracking						Electrons and Photons			$\pi/K/p$		HCAL		Muons	
		Resolution	Relative Momentum	Allowed X/X_0	Minimum- p_T (MeV/c)	Transverse Pointing Res.	Longitudinal Pointing Res.	Resolution σ_E/E	PID	Min E Photon	p-Range	Separation	Resolution σ_E/E	Energy		
< -4.6	Low-Q2 tagger	Not Accessible														
-4.6 to -4.0		Reduced Performance														
-4.0 to -3.5		Reduced Performance														
-3.5 to -3.0	Backward Detector		$\sigma_{\eta}/p \sim 0.1\% \times p @ 2\%$	-5% or less	150-300				$1\%/E @ 2.5\%/E @ 1\%$	π suppression up to $1:10^4$	20 MeV	≤ 10 GeV/c		$50\%/E @ 10\%$	-500 MeV	Muons useful for background suppression and improved resolution
-3.0 to -2.5			$\sigma_{\eta}/p \sim 0.02\% \times p @ 1\%$					$2\%/E @ (4-8)\%/E @ 2\%$	π suppression up to $1:(10^3-10^2)$	50 MeV						
-2.5 to -2.0						$dca(xy) \sim 40 p_T \mu m @ 10 \mu m$	$dca(z) \sim 100 p_T \mu m @ 20 \mu m$									
-2.0 to -1.5																
-1.5 to -1.0	Barrel		$\sigma_{\eta}/p \sim 0.02\% \times p @ 5\%$		400	$dca(xy) \sim 30 p_T \mu m @ 5 \mu m$	$dca(z) \sim 30 p_T \mu m @ 5 \mu m$	$2\%/E @ (12-14)\%/E @ (2-3)\%$	π suppression up to $1:10^2$	100 MeV	≤ 6 GeV/c	$\geq 3\sigma$	$100\%/E @ 10\%$			
-1.0 to -0.5																
-0.5 to 0.0																
0.0 to 0.5	Forward Detectors		$\sigma_{\eta}/p \sim 0.02\% \times p @ 1\%$		150-300	$dca(xy) \sim 40 p_T \mu m @ 10 \mu m$	$dca(z) \sim 100 p_T \mu m @ 20 \mu m$	$2\%/E @ (4-12)\%/E @ 2\%$	$3\sigma e/\pi$ up to 15 GeV/c	50 MeV	≤ 50 GeV/c		$50\%/E @ 10\%$			
0.5 to 1.0																
1.0 to 1.5																
1.5 to 2.0																
2.0 to 2.5																
2.5 to 3.0			$\sigma_{\eta}/p \sim 0.1\% \times p @ 2\%$													
3.0 to 3.5																
3.5 to 4.0	Instrumentation to separate charged particles from photons	Reduced Performance														
4.0 to 4.5		Not Accessible														
> 4.6	Proton Spectrometer															
	Zero Degree Neutral Detection															

Detector & reconstruction requirements

Extensive list of key performance parameters inform detector choices. This table of requirements could be interpreted as a series of automated tests that a detector implementation needs to pass.

Physics requirements

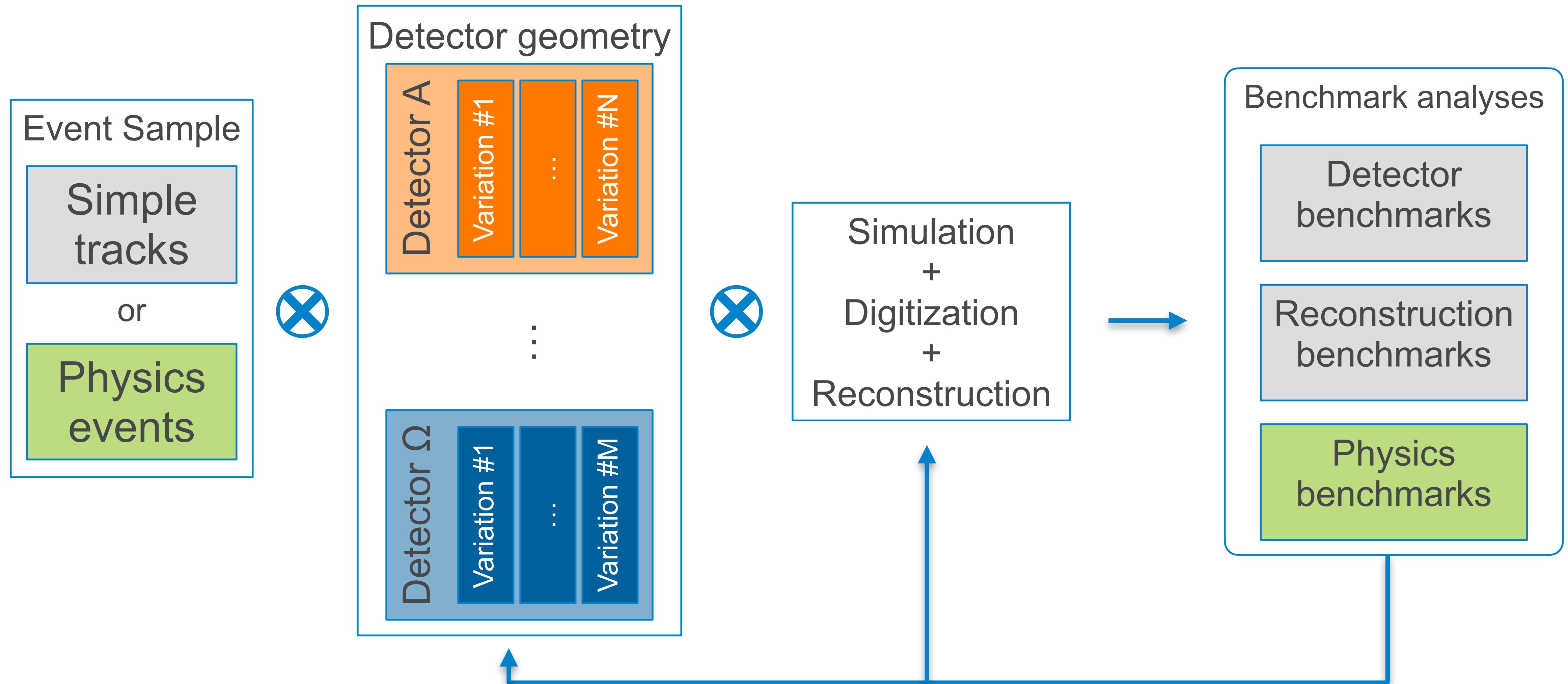
Detector design has to enable many key physics measurements, while being flexible enough to accommodate new developments through the next 2 decades

2	Physics Measurements and Requirements	6
2.1	Introduction	6
2.2	Origin of Nucleon Spin	9
2.3	Origin of Nucleon Mass	10
2.4	Multi-Dimensional Imaging of the Nucleon	11
2.5	Imaging the Transverse Spatial Distributions of Partons	12
2.6	Physics with High-Energy Nuclear Beams at the EIC	13
2.7	Nuclear Modifications of Parton Distribution Functions	14
2.8	Passage of Color Charge Through Cold QCD Matter	15

+ new developments

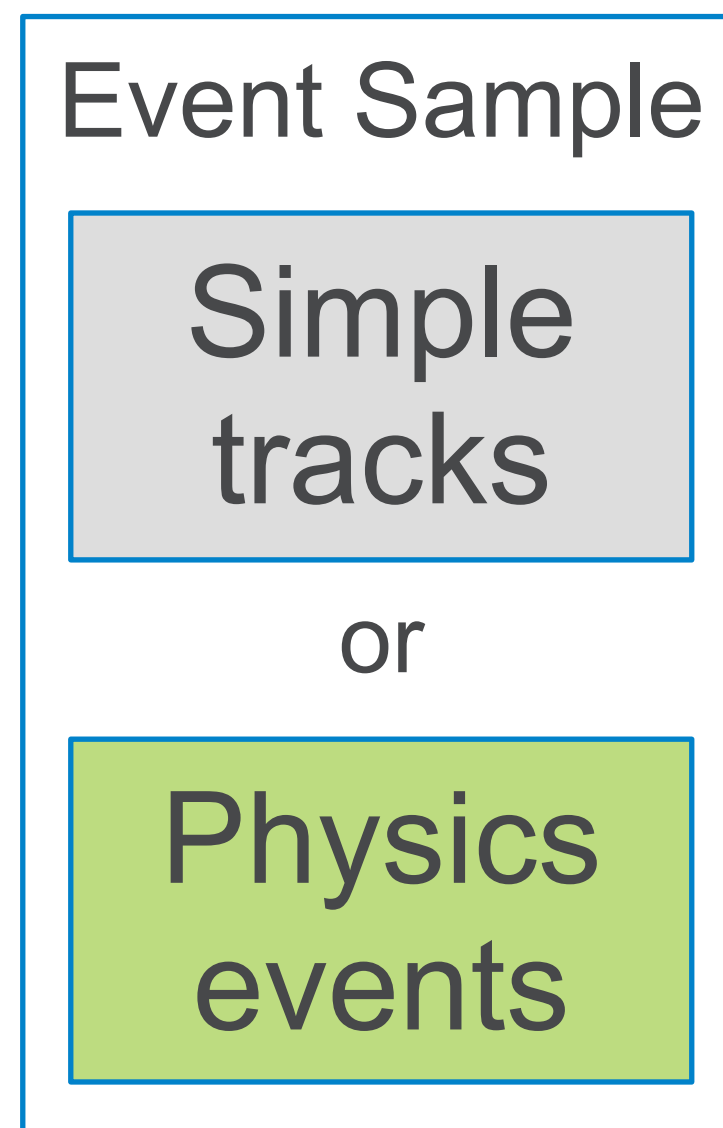
Enabled by modular, interoperable toolkit

DETECTOR OPTIMIZATION AS AN AUTOMIZED WORKFLOW



EVENT GENERATION

Standardized around HepMC3 data format



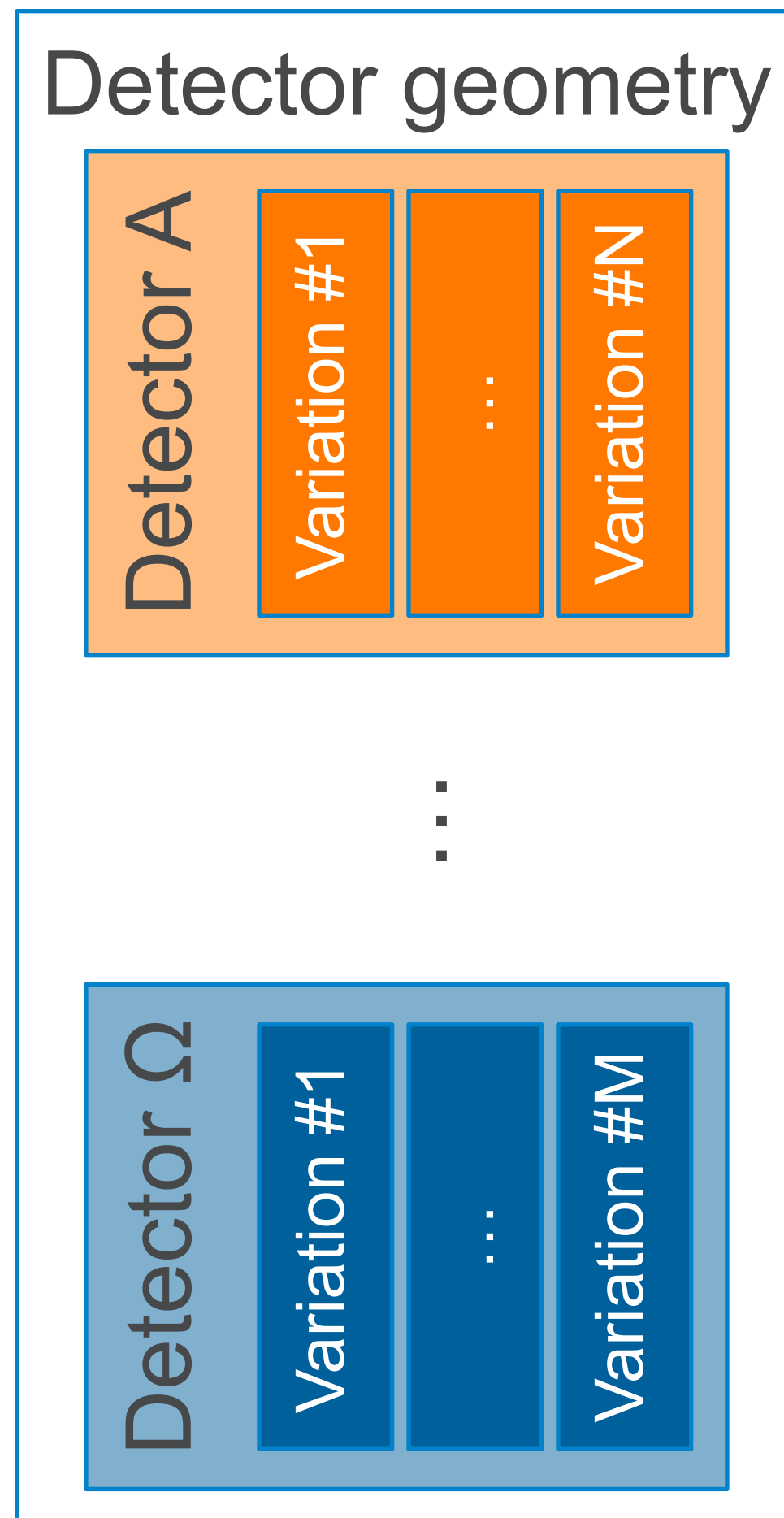
▶ Validate detector and benchmark reconstruction chain against YR requirements (using particle guns)

▶ Characterize and benchmark detector setup for desired physics observables (using physics event generators)

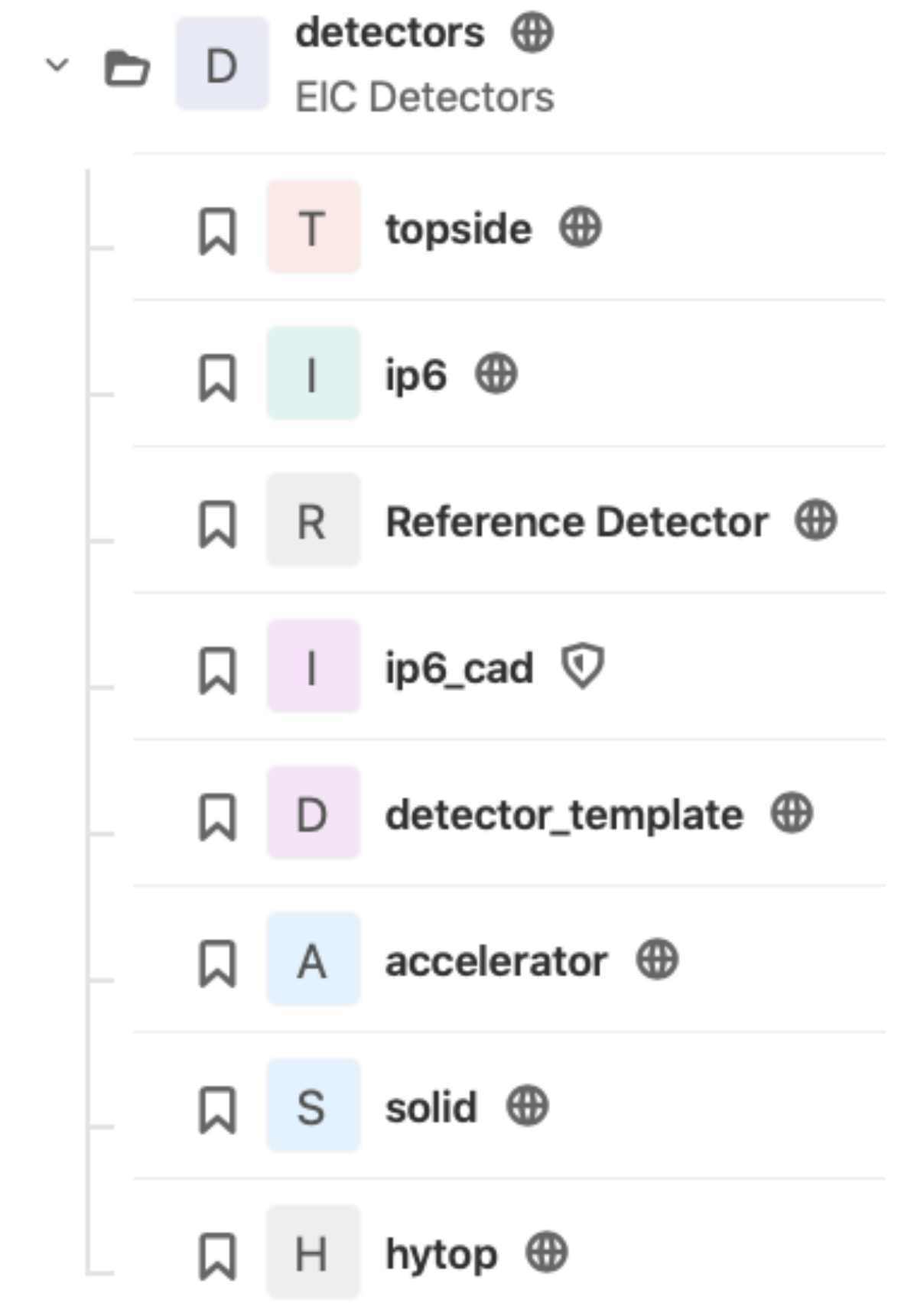
- Towards a fully integrated setup for on-demand event generation to cover all NAS/YR requirements for EIC
- Software chain standardized around the HepMC3 format

DETECTOR GEOMETRY

Parametrized detector implementations with DD4hep + NPDet



- Parametrized detector description defined using DD4hep
- NPDet: an extension for DD4hep that adds parametrized detectors for NP experiments (e.g. Cherenkov counters) and various DD4hep based tools (visualization, conversion, inspection, ...)
- **Single, unique source for geometry:** dispatches full GEANT4 simulation and provides geometry for reconstruction
- Aim towards library of configurable detector options to feed into benchmark & optimization process.



MAIN TOOLKIT

Simulation/digitization/reconstruction

Simulation
+
Digitization
+
Reconstruction

DD4hep: Geant4 geometry fully defined through detector plugin library, provides wrappers to run Geant4

ACTS: Experiment-independent tracking toolkit developed for HEP in modern C++, ACTS' geometry constructed from DD4hep via plugin

GAUDI: Generic open project for building event processing frameworks. Enables modern task-based concurrent execution in a heterogeneous computing environment.

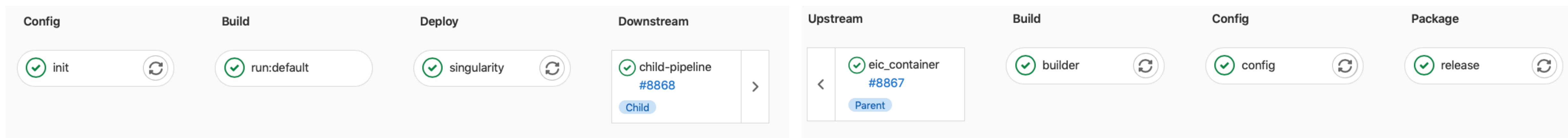
Podio: Robust data model definition to cross the boundaries between the tools

Project Juggler: Prototype event processing framework for EIC, tying together GAUDIO, ACTS, DD4hep and Podio. Used for the digitization and reconstruction step.

Note similarity with key4hep approach!

BUILDING THE SOFTWARE

Containerization and deployment



Container Registry

5 Image repositories Expiration policy will run in 6 days

With the GitLab Container Registry, every project can have its own space to store images. [More information](#)

Filter results

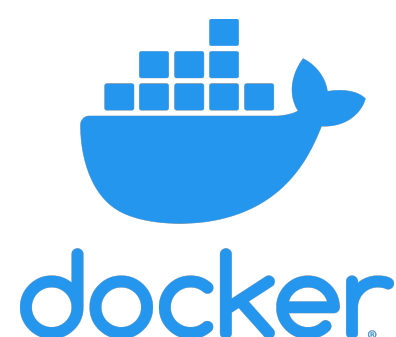
- containers/eic_container/eic 8 Tags
- containers/eic_container/eic_builder 8 Tags
- containers/eic_container/node 1 Tag
- containers/eic_container/rtd_builder 1 Tag
- containers/eic_container/alpine 1 Tag



Leverage own GitLab server instance ([eicweb](#)), with continuous integration and dedicated build cluster, to automatically build container



Reproducible container build with Spack



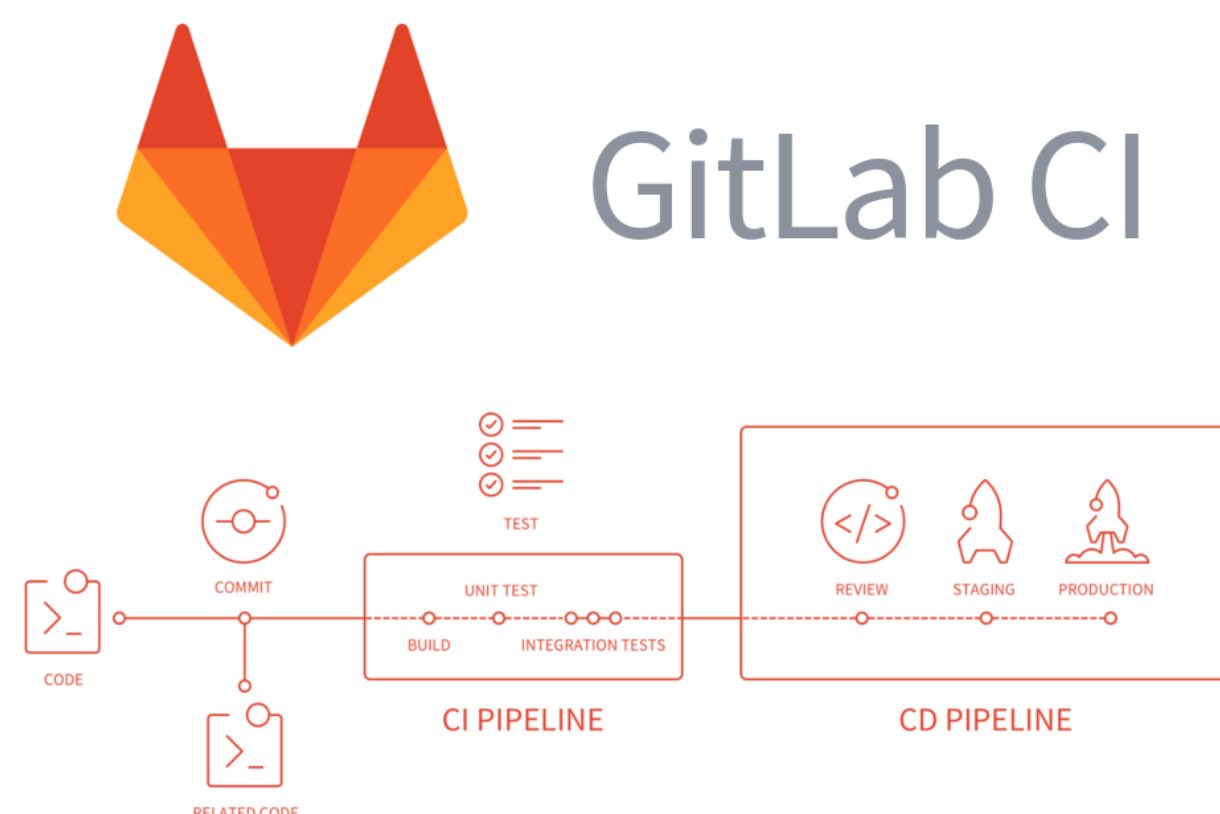
Docker container used in CI toolchain (and local development on MacOS)



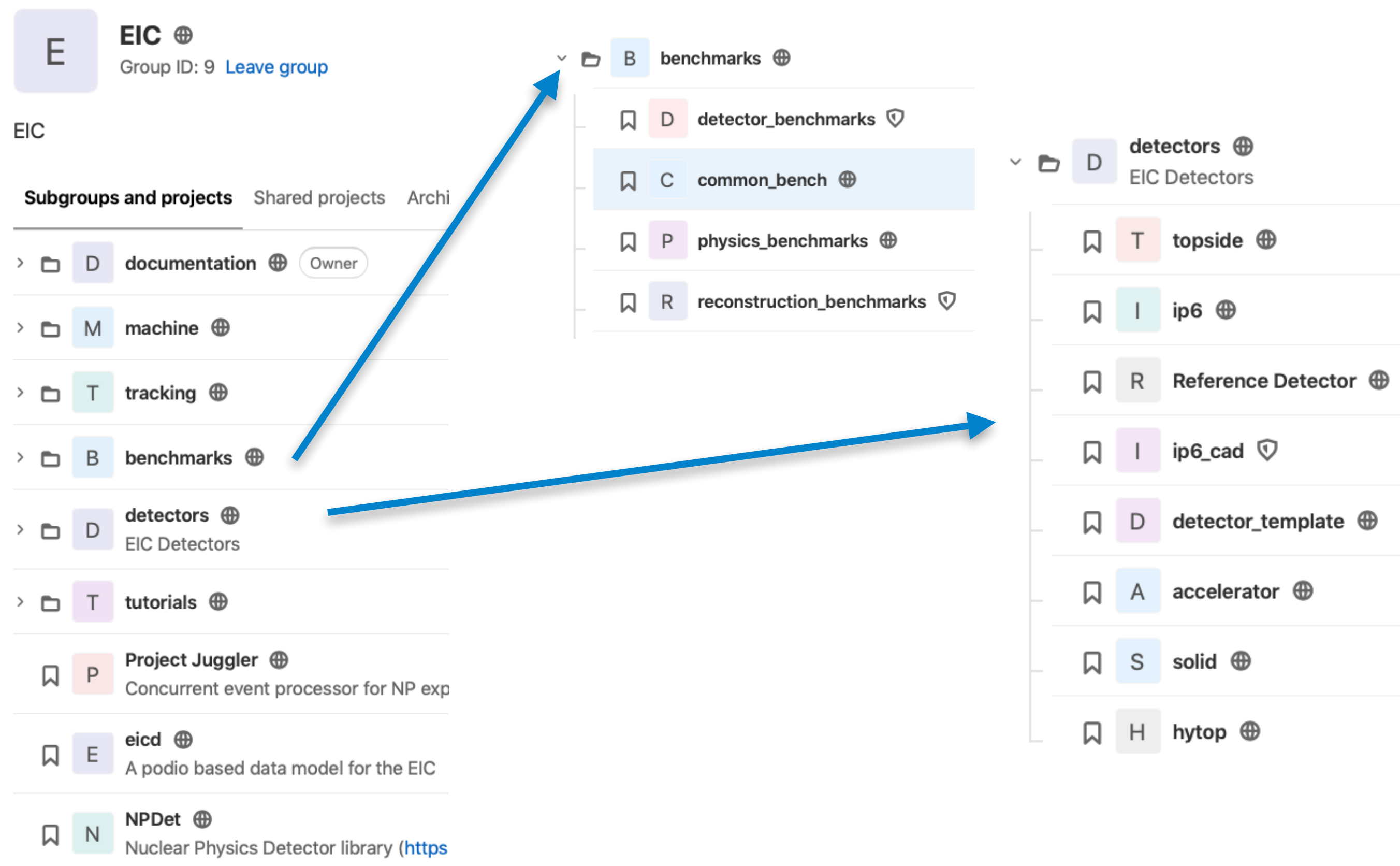
Singularity container used for deployment to HPC environments (and local development)

NEW PARADIGM IN SIMULATION WORKFLOW

From classic CI/CD to detector and physics benchmarks



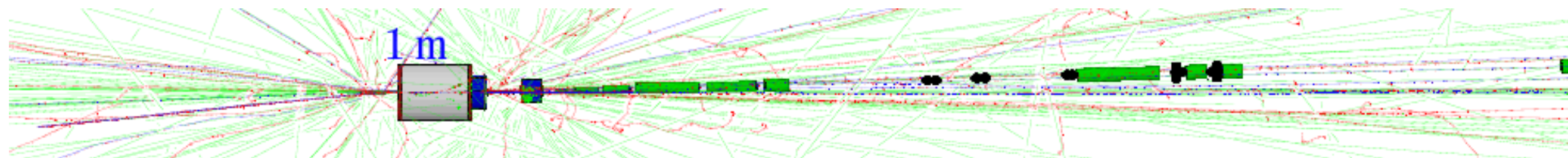
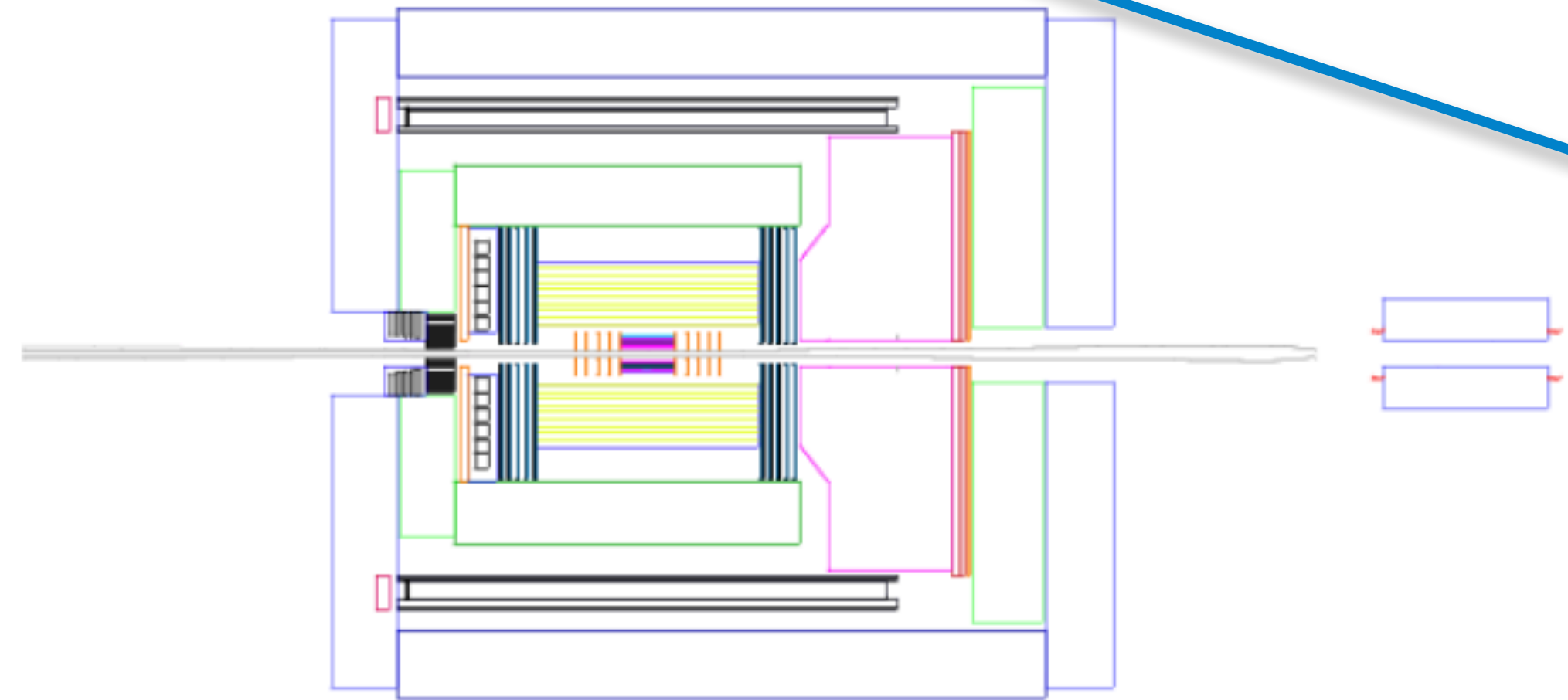
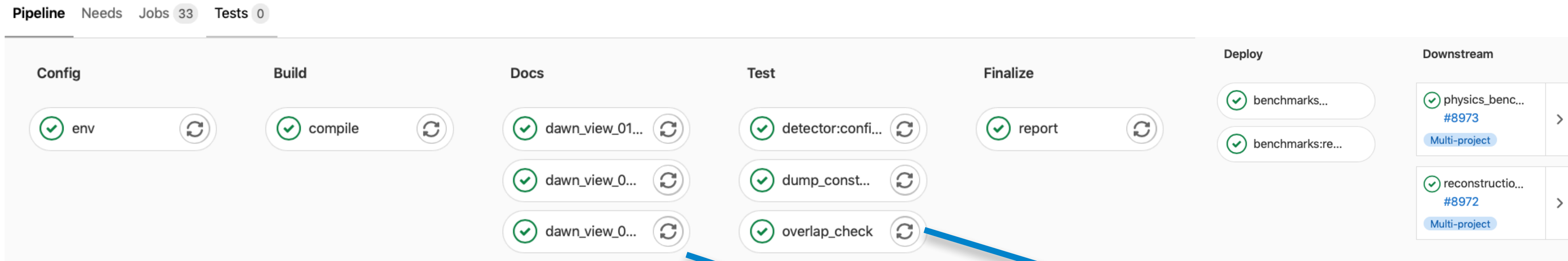
- **CI/CD:** continuous integration & Deployment
- **Pipeline:** Collection of interdependent jobs that run in a certain sequence
- **Job:** A task (script) that runs in a CI pipeline
- **Artifact:** File(s) produced by a job uploaded to GitLab
- **Pipeline trigger:** Connecting pipelines



Many repos chained together, organized with GitLab groups

PIPELINE EXAMPLES

Automatic detector tests and documentation



Geometry overlap checks running as part of every merge request

Automatic visualizations for detector geometries, saved as job artifacts (browsable!)

EIC > benchmarks > reconstruction_benchmarks > Jobs > #43398 > Artifacts

passed Job #43398 in pipeline #7165 for a989e4bc from master by Jihee Kim 1 week ago

Artifacts / results

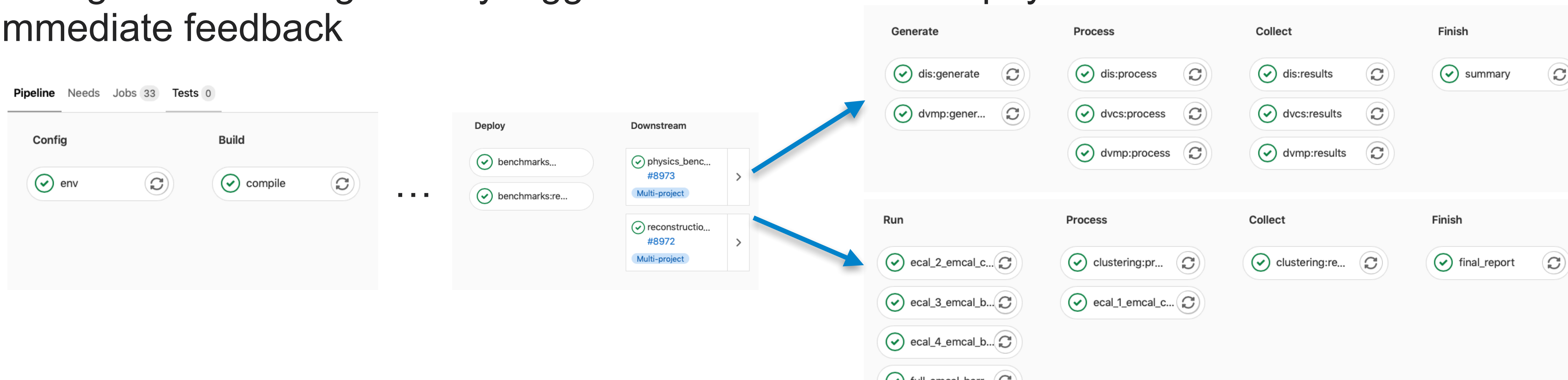
Name	Size
emcal_pi0s_Eres_nc2.pdf	15.9 KB
emcal_pi0s_Eres_nc2.png	13.1 KB
emcal_pi0s_Eres_nc2_cut.pdf	16.1 KB
emcal_pi0s_Eres_nc2_cut.png	13.9 KB
emcal_pi0s_angle_two_photons_nc2.pdf	14.8 KB
emcal_pi0s_angle_two_photons_nc2.png	12.6 KB

PIPELINE TRIGGERS

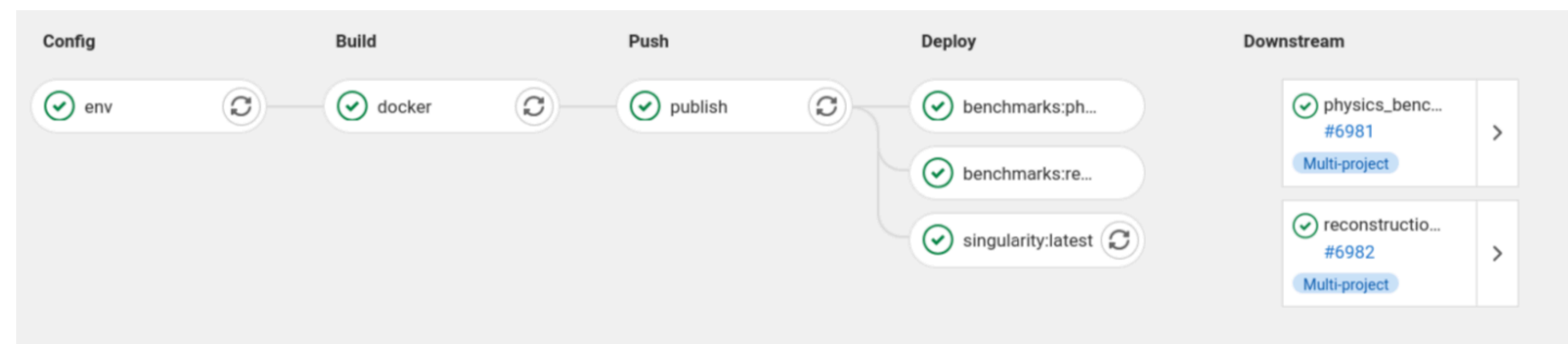
Connecting pipelines for immediate feedback



Change in detector geometry triggers reconstruction and physics benchmarks for immediate feedback



Also dispatch from Jugger CI: Due to merge request trigger, we can ensure proper operation of new/updated reconstruction algorithms






OPEN CHALLENGES

- Need full library of Monte Carlo generators that cover entire YR program
- Automatic publication/documentation of performance metrics to webpage for discoverability
- Compiling/combining metrics from benchmarks to guide optimization
- Distributed data persistency required, as GitLab artifact approach does not scale for large simulations
- Dispatch to HPC integrated in workflow (custom singularity gitlab runner? K8s? Manual pre-computations?)

EXAMPLE: SIMPLE BROWSER-BASED WORKFLOW

BROWSER-BASED WORKFLOW EXAMPLE

Let's change VertexTrackerRadius1 from 30mm to 29mm

```
vertex_tracker.xml 8.8 KB Edit Web IDE Replace Delete     
1 <?xml version="1.0" encoding="UTF-8"?>  
2 <lccdd>  
3 <define>  
4  
5 <constant name="VertexTrackerRadius1" value="30*mm"/>  
6 <constant name="VertexTrackerRadius2" value="38*mm"/>  
7 <constant name="VertexTrackerRadius3" value="46*mm"/>  
8 <constant name="VertexTrackerRadius4" value="54*mm"/>  
9 <constant name="VertexTrackerRadius5" value="62*mm"/>  
10 <constant name="VertexTrackerRadius6" value="70*mm"/>  
11 <constant name="VertexTrackerRadius7" value="78*mm"/>  
12 <constant name="VertexTrackerRadius8" value="86*mm"/>  
13 <constant name="VertexTrackerNModules1" value="14"/>  
14 <constant name="VertexTrackerNModules2" value="18"/>  
15 <constant name="VertexTrackerNModules3" value="22"/>  
16 <constant name="VertexTrackerNModules4" value="24"/>  
17 <constant name="VertexTrackerNModules5" value="28"/>  
18 <constant name="VertexTrackerNModules6" value="32"/>
```

BROWSER-BASED WORKFLOW EXAMPLE

Simplest workflow uses the browser-based GitLab IDE

The image displays two views of the GitLab IDE interface. The top view shows a browser-based editor for a file named 'vertex_tracker.xml' (8.8 KB). The code is XML, starting with a root element 'lccdd' and a 'define' block containing two constants: 'VertexTrackerRadius1' with a value of '30*mm' and 'VertexTrackerRadius2' with a value of '38*mm'. A blue arrow points from the 'Edit' button in this view to the IDE interface below.

The bottom view shows the full IDE interface. The left sidebar contains a file explorer with a list of files: 'solenoid.xml', 'topside_defs.xml', 'topside_rich.xml', 'vertex_tracker.xml' (selected), 'views', '.clang-format', '.gitignore', '.gitlab-ci.yml', and 'CMakeLists.txt'. A 'Commit...' button is visible at the bottom of the sidebar, and a status bar indicates '1 changed file'. The main editor area shows the XML code being edited, with the 'VertexTrackerRadius1' constant value changed from '30*mm' to '29*mm'. The code continues with constants for radii 2 through 8 and NModules 1 through 8, ending with 'VertexTrackerBarrelLength' set to '200.0*mm'.

BROWSER-BASED WORKFLOW EXAMPLE

Commit changes to a new branch, and...

The screenshot displays a web-based code editor interface. On the left, a 'Changes' panel shows a commit message dialog with the text 'Tweaked the radius'. Below the dialog are options: 'Commit to master branch' (selected), 'Create a new branch' (with a text input field containing 'vertex_tracker_radius'), and 'Start a new merge request' (checked). A green 'Commit' button and a 'Discard draft' button are at the bottom of the panel. The main editor area shows a diff of 'topside/vertex_tracker.xml'. The left side of the diff (original) shows line 5 as `5-ertexTrackerRadius1" value="30*mm"/>`. The right side (modified) shows line 5 as `5+ertexTrackerRadius1" value="29*mm"/>`. Other lines in the diff include `ertexTrackerRadius2` through `ertexTrackerRadius8`, `ertexTrackerNModules1` through `ertexTrackerNModules8`, and `ertexTrackerBarrelLength` with a value of `200.0*pi`. A 'Discard changes' button is visible in the top right corner of the editor area.

BROWSER-BASED WORKFLOW EXAMPLE

...create a new Merge Request

New Merge Request

From `vertex_tracker_radius` into `master` [Change branches](#)

Title

Start the title with `Draft:` or `WIP:` to prevent a merge request that is a work in progress from being merged before it's ready.

Add [description templates](#) to help your contributors communicate effectively!

Description

Write Preview

B *I* ” </>      ↗

Describe the goal of the changes and what reviewers should be aware of.

[Markdown](#) and [quick actions](#) are supported

 [Attach a file](#)

BROWSER-BASED WORKFLOW EXAMPLE

...the Merge Request then triggers the CI.

EIC > detectors > topside > Pipelines > #7395

running Pipeline #7395 triggered 41 seconds ago by Whitney Armstrong Cancel running Delete

Tweaked the radius

🕒 45 jobs for !91 with `vertex_tracker_radius` (queued for 3 seconds)

📄 latest detached

🔗 30bc4360

🔗 1 related merge request: !91 Tweaked the radius

Pipeline Needs Jobs 45 Tests 0

```
graph LR; subgraph Config; C1[env]; end; subgraph Build; B1[compile]; end; subgraph Docs; D1[dawn_view_01:...]; D2[dawn_view_01:...]; D3[dawn_view_01:...]; D4[dawn_view_02:...]; end; subgraph Test; T1[overlap_check]; T2[topside:config...]; T3[view_01]; T4[view_02]; end; subgraph Finalize; F1[report]; end; subgraph Deploy; DE1[benchmarks:ph]; DE2[benchmarks:re..]; end; C1 --> B1; B1 --> D1; B1 --> D2; B1 --> D3; B1 --> D4; D1 --> T1; D1 --> T2; D2 --> T1; D2 --> T2; D3 --> T1; D3 --> T2; D4 --> T1; D4 --> T2; T1 --> F1; T2 --> F1; T3 --> F1; T4 --> F1; F1 --> DE1; F1 --> DE2;
```

**ALL WITHOUT LEAVING THE BROWSER OR
INSTALLING ANY SOFTWARE LOCALLY!**



**U.S. DEPARTMENT OF
ENERGY**

Argonne National Laboratory is a
U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC.

18

Argonne 
NATIONAL LABORATORY

75
1946-2021

Our software R&D in the context of the EICUG software EoI

A WORKFLOW-CENTRIC SOFTWARE APPROACH FOR EIC

► **Workflows** - *leveraging the power of GitLab CI*

- ✓ Automized simulation-reconstruction-analysis pipelines
- ✓ Low bar of entry, were able to onboard external collaborators (students!) with minimal training, already doing valuable work.
- ✓ **next:** build out analysis portfolio, better dispatch to HPC, persistent distributed data storage, “publish” benchmark results through GitLab Pages



► **Detector simulations**

- ✓ Geometry and simulation tooling around DD4hep
- ✓ **next:** develop library of parametrized detector concepts, accelerate simulation with GANs

► **Reconstruction**

- ✓ Functioning generic tracking with ACTS (and Gaudi)
- ✓ **next:** reconstruction for other subsystems, improve task-based concurrency, accelerate part of workflow on GPUs, or with AI.

► **Data model**

- ✓ Data model definition with podio
- ✓ **next:** optimize and freeze exact data model, together with community

► **Explore user-centered design**

- ✓ Modular workflow with low bar of entry for analysis
- ✓ **next:** automatic publication of pipeline results to website

► **Data analysis preservation:**

- ✓ Analysis as part of benchmark portfolio pipeline
- ✓ **next:** website to aggregate analysis results and monitor quality of results (in the vein of benchmarks/tests).

► **Discoverable software**

- ✓ Software pipeline fully containerized, software stack built and managed with Spack
- ✓ **next:** integration of newer software tools with Spack

THE END



U.S. DEPARTMENT OF
ENERGY

Argonne National Laboratory is a
U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC.

20

Argonne 
NATIONAL LABORATORY

75
1946-2021