

Infrastructure 2023

Paul Gessinger

CERN

2023-11-08 - ACTS Workshop Orsay 2023



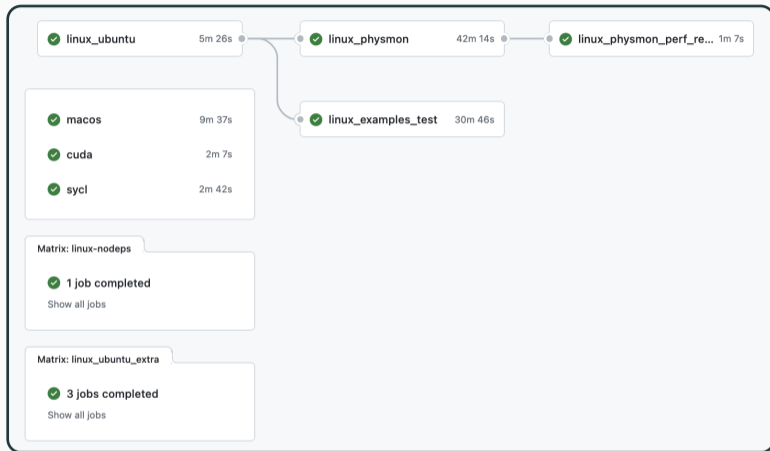
Build & Infrastructure

- Continuous integration
- Developer tools
- Release strategy

Continuous Integration

Continuous Integration jobs overview

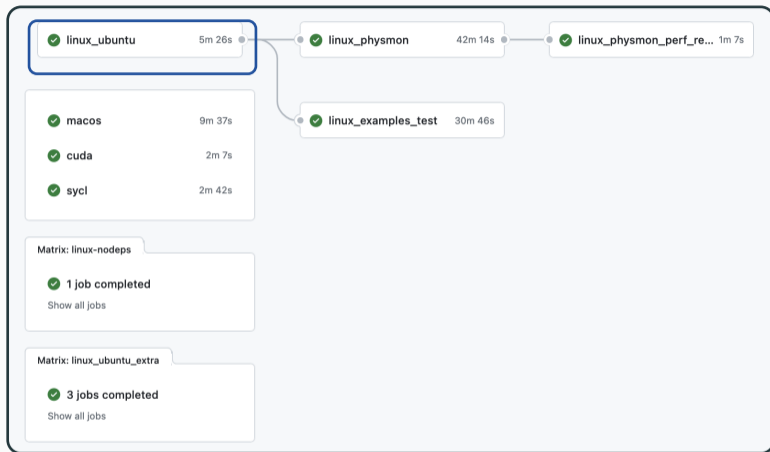
- Continuous integration tests our code automatically
- Is required to pass for PRs to be merged
- Note: **GitHub CI runs on merge result of PR into `main`**
 - ▶ That means jobs do not get triggered in case of merge conflicts



CI: Build workflow

linux_ubuntu

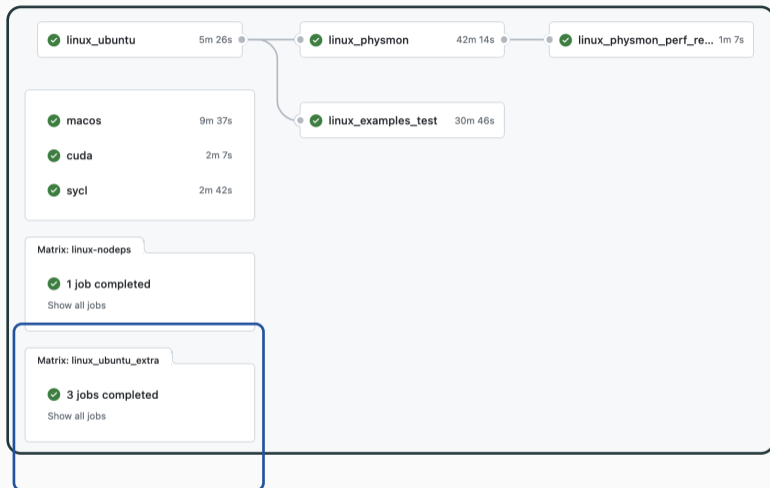
- The main CI build: this job builds everything!
- Runs unit tests, integration tests, downstream project usage
- Uses Ubuntu 22.04 image that bundles all dependencies



CI: Build workflow

linux_ubuntu_extra

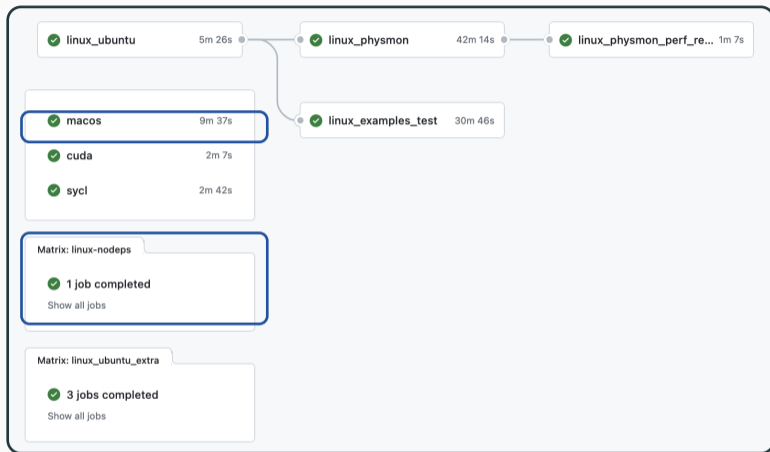
- Build using Ubuntu 22.04 with updated GCC, based on central Ubuntu test
- Ubuntu 22.04 test using clang (+libstdc++) instead of GCC



CI: Build workflow

lcg + macos

- LCG tests removed from GitHub CI
- macOS: All dependencies, tests + downstream test
- nodeps: build core without external dependencies



Unit tests

- Comprehensive unit test suite
- Quality of tests varies:
 - ▶ Some tests really test expectations
 - ▶ Some tests simply test something runs without crashing
- Tests are run on all CI jobs
- Special job with debug symbols to measure coverage
- Have let this slide a bit:
quick-turnaround unit tests are invaluable for development!

```
...
290/293 Test #291: ConvertTrackPodio ..... Passed 0.01 sec
      Start 292: PodioTrackStateContainer
291/293 Test #292: PodioTrackStateContainer . Passed 0.02 sec
      Start 293: Alignment
292/293 Test #293: Alignment ..... Passed 0.01 sec
293/293 Test #287: TGeoParser ..... Passed 0.28 sec
```

100% tests passed, 0 tests failed out of 293

Total Test time (real) = 60.51 sec

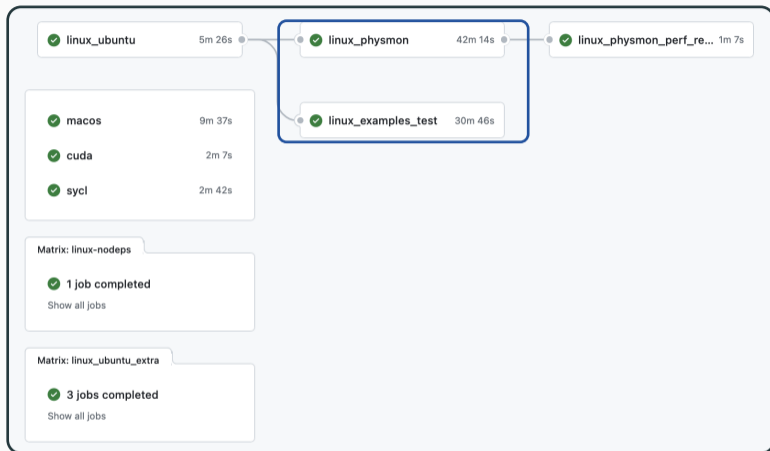
```
@@          Coverage Diff          @@
##          main    #2621    +/-    ##
=====
Coverage    49.59%    49.60%
=====
Files        473        473
Lines       26824       26828    +4
Branches    12355       12355
=====
+ Hits        13304       13308    +4
Misses       4753        4753
Partials     8767        8767
```


CI: Build workflow

Performance monitoring

linux_examples_test

- Uses Python bindings for the examples
- Python-level tests



Python level examples tests

- Comprehensive suite of `pytest` tests
- Directly tests the examples code, but indirectly also the core code
- Implements [hash-based ROOT file-level reproducibility checks](#)
 - ▶ CI notifies on PRs if outputs change
 - ▶ Manual follow up and reference update required
- Allows us to catch basic regressions early
- Ordering independent: reproducibility checks on MT code

CI: Build workflow

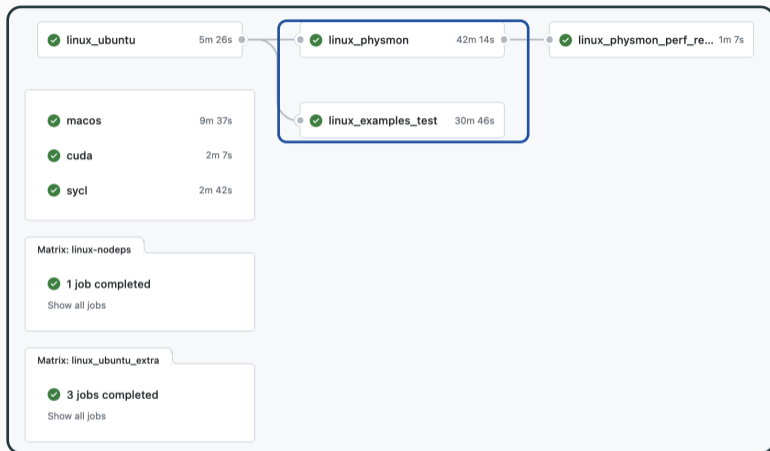
Performance monitoring

linux_examples_test

- Uses [Python bindings for the examples](#)
- Python-level tests

linux_physmon

- Runs tracking workflows and records performance metrics
- Compares to references



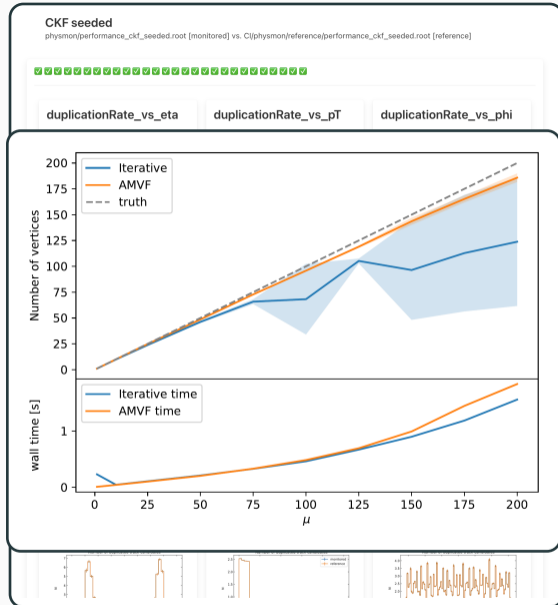
Physics performance monitoring



- Higher level automated performance checks for physics workflows
- Added only 2 years ago: but quickly expanded
- Has become one of the main ways for us to check impact of developments
- **Caveat:** this only runs workflows on OpenDataDetector, and only in limited combinations (resource constraint)
- We are not sensitive to experiment-specific impacts with this (see next talk by Carlo)

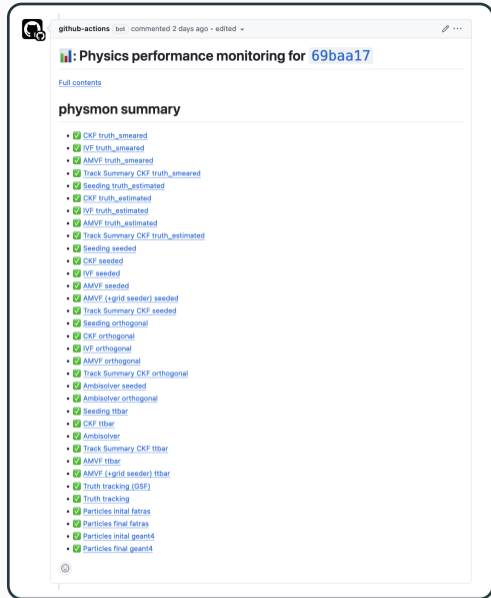
Physics performance monitoring

- Implemented workflows:
 - ▶ Truth tracking with KF and GSF
 - ▶ CKF + seeding, truth estimation, truth smearing + vertexing
 - ▶ CKF on $t\bar{t}$ event at pile-up 200
 - ▶ Fatras and G4 simulation
 - ▶ Dedicated vertexing workflow
- Runs histogram comparisons using [histcmp](#) and produces a neat single-HTML-file report with embedded plots
- Implements basic compatibility checks: KS, χ^2 , ratio, residual, integral
 - ▶ Looking into replacing this with `dcube`



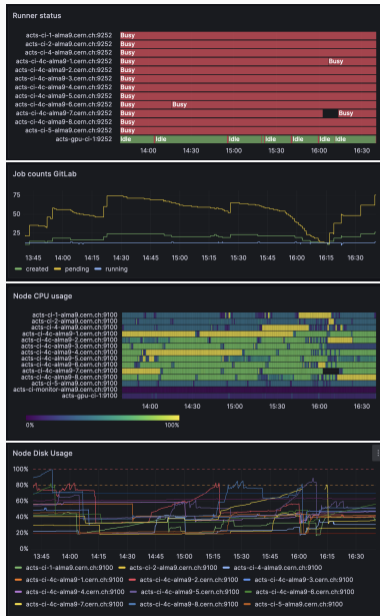
Herald and report comment

- Report job posts a summary of the physmon run to open Pull Requests
 - ▶ Contains links to [Herald](#), which allows browsing artifact contents without downloading
- Comment is updated when physmon reruns



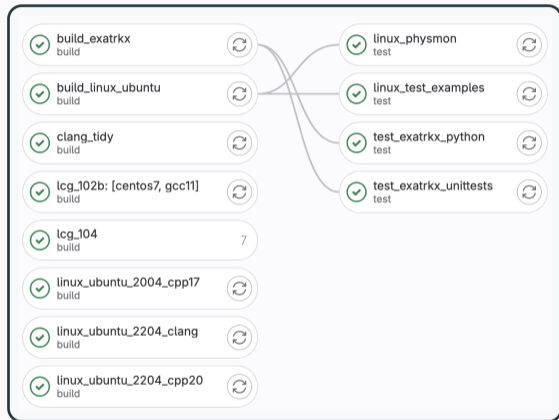
Bridged CI jobs

- Have custom setup to run CI job on CERN resources
 - ▶ Currently $\mathcal{O}(10)$ slots running in VMs + 1 physical machine with a GPU attached
 - ▶ Machines managed using ansible: trivial to add / recreate machines from scratch
 - ▶ Monitoring implemented using Prometheus and Grafana
 - ▶ Allows diagnosing problems in \sim real time
- Access steered by a custom allow-list: if you are a new contributor and are getting "Pipeline refused" on these jobs, **get in touch!**



Bridged CI jobs

- LCG test jobs: LCG 102b , 104 × CentOS7, Alma9 × GCC 11, 12, 13, clang 16 (with some combinations skipped)
- clang-tidy job (static code analysis, details [here](#))
- GPU CI job: build jobs runs on CPU runner, test job runs on GPU runner under my desk
- Ubuntu 20.04/22.04 build on C++17 compiler (to ensure we're compatible) + 1 C++20 build
- Ubuntu 22.04 build on Clang
- Also runs physmon and python tests **with FPE monitoring** (not enough resources on GitHub CI)



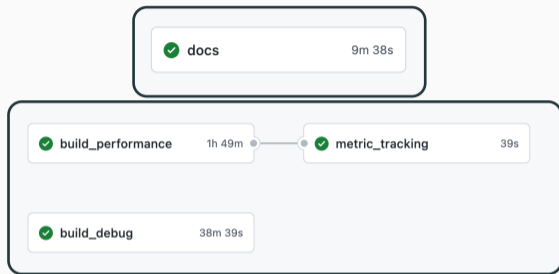
CI: Checks workflow

- Formatting: C++ (clang-format) and python (black)
 - ▶ Can run locally, or apply patch from CI artifact. Details [here](#)
- License check: validate that all source files start with the correct [license block](#)
- Include guards: check for `#ifndef MY_HEADER`: use `#pragma once` instead
- Check consistent end of lines
- Check against usage of `BOOST_TEST()`
- Check smearing config
- Ensure CMake options are documented
- Spell-check in the code
- Check for missing includes
- Check FPE masks are documented (more on that later)

| | |
|--------------------|---------|
| ✓ format | 25s |
| ✓ format-py | 15s |
| ✓ license | 7s |
| ✓ include_guards | 5s |
| ✓ pragma_once | 8s |
| ✓ end_of_line | 5s |
| ✓ boost_test_macro | 8s |
| ✓ smearing_config | 7s |
| ✓ cmake_options | 5s |
| ✓ spelling | 11s |
| ✓ missing_includes | 15m 45s |
| ✓ fpe_masks | 16s |

CI: Code analysis and docs workflow

- Docs job builds the documentation, checks for missing comments etc.
- Build performance runs `cmakeperf` and measures compilation unit memory consumption
 - ▶ Report on the main branch [here](#)
- Build debug: similar to `linux_ubuntu`, but without examples
 - ▶ Uses debug build to run unit tests and generate code coverage report
 - ▶ Coverage report is posted to each PR



Additional bots and workflows

- **merge-sentinel**: manages which jobs / checks are required (custom bot)
- **review-required**: checks approval from one of the reviewers (self-hosted policybot instance)
- **kodiakhq**: enables auto-merging (central instance)
- **WIP**: blocks merging PRs with label "WIP" (central instance)
 - ▶ Discouraged in favor of using GitHub's draft PR feature
- **codecov**: Coverage reporting (central instance)
- **readthedocs.org**: Preview build of the documentation, deployed to RTD
- **milestone-set**: Blocks merging without a milestone

| | | | |
|---------------------------|--|----------|---------|
| merge-sentinel | In progress — Waiting for 1 job: review-required: main | Required | Details |
| review-required: main | Pending — 0/1 rules approved | Required | Details |
| kodiakhq: status | Ignored (no automerge label: 'automerge') | | Details |
| WIP | Successful in 2s — Ready for review | Required | Details |
| codecov/project | 48.57% (-0.01%) compared to eb94af6 | | Details |
| docs/readthedocs.org:acts | Read the Docs build succeeded! | | Details |
| milestone-set | Milestone is set! | Required | Details |

Developer tools

Developer tools

- CMake option: `ACTS_FORCE_ASSERTIONS` force-enables `assert`
 - ▶ Allows having checks in ACTS code + 3rd party code even in Release builds
 - ▶ Affects performance, be aware when benchmarking
- CMake options `ACTS_LOG_FAILURE_THRESHOLD=<LEVEL>` and `ACTS_ENABLE_LOG_FAILURE_THRESHOLD` steer log level failure threshold ([details](#))
 - ▶ If enabled, logging at or above `<LEVEL>` will cause a job to fail
 - ▶ Can be set at compile-time or at runtime via env variable or set from Python
 - ▶ Allows us to check workflows covered by CI do not produce WARNINGS and ERRORS

FPE monitoring

- FPE monitoring introduced as a plugin
- Uses custom infrastructure to enable FPE trap signal and handle
- Signal handler collects stack trace in async-safe way and records it
 - ▶ FPE locations are deduplicated based on the top-most stack frame **source code location!**
- Sequencer can be configured to ignore (**mask**) FPEs
- If configured: Sequencer terminates job when FPE is encountered (and not masked)
- At end of job: FPE are accumulated per algorithm and reported
 - ▶ Job is still failed if unmasked FPE are encountered

FPE masking

- Mask is a combination of source file and line range
 - ▶ Matching is performed from bottom to top in stack frame: if any frame matches mask, FPE is considered masked
 - ▶ Limit number of FPEs per event (but keep in mind that FPE state has to be reset manually)
- Typically want to **fix** the FPE, masking does not mask them outside of Examples
- Statistical process: if you get them depends on the workflow / inputs

```
FPE masks
-----
86  const auto &[weight_l, pars_l, cov_l] = projector(cmp);
87
88  cov += weight_l * cov_l; // MARK: fpeMask(FLTUND, 1, #2347)
89
90  ActsVector<D> diff = pars_l - mean;
91
/Users/pagessin/dev/acts/Core/include/Acts/Utilities/GaussianMixtureReduction.hpp
-----
196  std::apply([&](auto... dsc) { wrap(dsc, ...); }, angleDesc);
197
198  // MARK: fpeMaskBegin(FLTUND, 1, #2347)
199  const auto cov =
200  gaussianMixtureCov(components, mean, sumOfWeights, projector, angleDesc);
201  // MARK: fpeMaskEnd(FLTUND)
202
203  return RetType(mean, cov);
204 }
/Users/pagessin/dev/acts/Core/include/Acts/Utilities/GaussianMixtureReduction.hpp
-----
117  }
118  }
119  // MARK: fpeMaskBegin(FLTUND, 1, #2590)
120  bool keepVertex = isGoodVertex &&
121  keepNewVertex(vtxCandidate, allVerticesPtr, fitterState);
122  // MARK: fpeMaskEnd(FLTUND)
123  ACTS_DEBUG("New vertex will be saved: " << keepVertex);
124
125  // Delete vertex from allVertices list again if it's not kept
/Users/pagessin/dev/acts/Core/include/Acts/Vertexing/AdaptiveMultiVertexFinder.hpp
-----
517
518  // track parameters error
519  // MARK: fpeMaskBegin(FLTINV, 1, #2348)
520  m_err_eLOC0[ipar].push_back(
521  std::sqrt(covariance(Acts::eBoundLoc0, Acts::eBoundLoc0)));
522  m_err_eLOC1[ipar].push_back(
523  std::sqrt(covariance(Acts::eBoundLoc1, Acts::eBoundLoc1)));
524  m_err_ePHI[ipar].push_back(
525  std::sqrt(covariance(Acts::eBoundPhi, Acts::eBoundPhi)));
526  m_err_eTHETA[ipar].push_back(
527  std::sqrt(covariance(Acts::eBoundTheta, Acts::eBoundTheta)));
528  m_err_eQOP[ipar].push_back(
529  std::sqrt(covariance(Acts::eBoundQOverP, Acts::eBoundQOverP)));
530  m_err_eT[ipar].push_back(
531  std::sqrt(covariance(Acts::eBoundTime, Acts::eBoundTime)));
532  // MARK: fpeMaskEnd(FLTINV)
533
534  // track parameters pull
535  m_pull_eLOC0[ipar].push_back(
```


Release strategy

Release strategy

- We follow semantic versioning: vMAJOR.MINOR.PATCH
- We currently limit MAJOR bumps to once a month
 - ▶ Target MINOR/PATCH release about once a week
- General development targets the `main` via PRs
 - ▶ Requires one approval from one of our reviewers (get in touch if you want to review!)
- Mainline tagged releases go onto the `releases` branch
- Before making a release, `main` gets merged into `releases`

- We follow semantic versioning
- We currently implement a target minor version
 - ▶ Target MINOR version
- General development
 - ▶ Requires one commit to review!
- Mainline tagged
- Before making a new release

```

tig
13
8000000 2022-09-23 14:51 +0100 Unknown o Untracked changes
7750c1d 2022-09-23 12:28 +0000 github-actions[bot] o [releases] {origin/releases} <v20.2.0> Bump to version v2
48f9dbb 2022-09-23 13:55 +0200 Paul Gessinger M Merge remote-tracking branch 'origin/main' into release
f3b20f7 2022-09-23 11:41 +0200 Benjamin Huth o [main] {origin/main} fix: maybe-uninitialized warning i
bf86784 2022-09-23 10:20 +0200 Paul Gessinger o chore: Update policybot configuration [skip ci]
e5d1970 2022-09-22 16:07 +0200 Andreas Stefl o refactor: improve full_chain_odd.py example (#1538)
bfc3676 2022-09-22 09:20 +0100 Tim Adye o docs: remove --recursive on git clone in quick start do
b99368a 2022-09-19 23:35 +0100 Tim Adye o refactor: improve full_chain_itk.py example (#1513)
896b47e 2022-09-19 23:04 +0200 Paul Gessinger o fix: ParticleSmearing options not setup in ANMF example
d9816f2 2022-09-19 22:00 +0200 Paul Gessinger o docs: Contribution guidelines (#1525)
3da30d6 2022-09-19 11:05 +0200 Paul Gessinger o chore: Add priority merge label to kodiak config (#1532)
9e22341 2022-09-19 10:12 +0200 Benjamin Huth o feat: Allow configurable particle selection and reprodu
4a41d30 2022-09-16 15:49 +0200 Corentin-Allaire o refactor: Improve material mapping speed (#1458)
4046059 2022-09-16 15:27 +0200 Guilherme Almeida o fix: Added missing return to seedfinder::CreateSeedsFor
2a17808 2022-09-16 12:34 +0200 Benjamin Huth o docs: Gaussian Sum Filter (#1403)
b409627 2022-09-16 12:21 +0200 Benjamin Huth o feat: Exa.TrkX with torchscript backend (#1473)
ff772f1 2022-09-16 09:59 +0200 Benjamin Huth o docs: update markdown cheatsheet (#1524)
33ce28f 2022-09-16 09:46 +0200 Paul Gessinger o docs: Update logging doc, add info on thresholds (#1520)
6b97a09 2022-09-15 15:31 +0200 Paul Gessinger o docs: Tracking in a nutshell (#1399)
4ad5016 2022-09-15 13:54 +0200 Paul Gessinger o docs: Update magnetic field documentation (#1518)
71f5c74 2022-09-15 12:44 +0200 Paul Gessinger o docs: Enable sphinx linkcheck build in CI (#1523)
9d4a4fd 2022-09-15 11:12 +0200 robertlangenberg o docs: Fix seeding example documentation (#1340)
4e270e7 2022-09-14 13:14 +0200 Paul Gessinger o docs: Remove obsolete config, and invalid doc markup (#
91ae8e7 2022-09-13 19:56 +0200 Benjamin Huth o feat: Add operator → to Acts::Result (#1509)
487a6cc 2022-09-13 18:19 +0200 Paul Gessinger o fix: Sequencer correctly calls algorithm finalize (#151
eb94af6 2022-09-13 11:45 +0200 Paul Gessinger o refactor: Reuse MultiTrajectory in (C)KF (#1507)
087ed98 2022-09-07 14:52 +0200 Paul Gessinger o fix: Make JsonMaterialDecorator IO error more explicit
433a379 2022-09-06 19:53 +0200 Andreas Stefl o refactor: python examples log level overwrite mechanism
f3ea581 2022-09-06 17:58 +0200 Paul Gessinger o ci: Add ccache stats output (#1489)
f5d312f 2022-09-06 14:19 +0200 Paul Gessinger o refactor: Vertex performance monitoring (#1417)
5f4052a 2022-09-06 12:07 +0200 Paul Gessinger o docs: Switch to doxygen FAIL_ON_WARNINGS (#1497)
3b91080 2022-09-05 23:38 +0200 Benjamin Huth o docs: fix image paths (#1495)
1fd479b 2022-09-05 19:04 +0200 Andreas Stefl o refactor: loop protection as a function (#1490)
df05834 2022-09-05 17:08 +0200 Benjamin Huth o docs: Add as small cheatsheet with some Myst-Markdown s
abf9f38 2022-09-05 16:53 +0200 Paul Gessinger o ci: RTD fix for API listing (#1492)
d54fa84 2022-09-05 16:40 +0200 Luis Falda Coelho o feat: split deltaR in seedFinderOrthogonal (#1471)
2b34aa1 2022-09-05 04:17 -0700 Tomohiro Yamazaki o refactor: SP builder update (#1218)
a1569de 2022-09-02 16:08 +0200 Luis Falda Coelho o feat: check if yOuter takes the square root of a negati
be36226 2022-09-02 08:17 +0000 github-actions[bot] o <v20.1.0> Bump to version v20.1.0
1a6bc29 2022-09-02 09:57 +0200 Paul Gessinger M Merge remote-tracking branch 'origin/main' into release
b172918 2022-09-02 00:46 -0700 Carlo Varni o fix: Set s_binningValues as const (#1485)
42b16ef 2022-09-01 19:12 +0200 Paul Gessinger o test: Use tempdir to produce BB vis output (#1483)
eb8bd04 2022-09-01 16:59 +0200 Andreas Stefl o test: big bounding box vs frustum intersection (#1478)
[main] Untracked changes 0%
Unknown key, press h for help

```

Release strategy

- We follow semantic versioning: vMAJOR.MINOR.PATCH
- We currently limit MAJOR bumps to once a month
 - ▶ Target MINOR/PATCH release about once a week
- General development targets the `main` via PRs
 - ▶ Requires one approval from one of our reviewers (get in touch if you want to review!)
- Mainline tagged releases go onto the `releases` branch
- Before making a release, `main` gets merged into `releases`
- When required: create `develop/vX.Y.Z` branch as needed
 - ▶ Should only receive backports, no approval needed since no new developments
- Example:
 - ▶ Created `develop/v19.x` branch after `v20.0.0` was tagged
 - ▶ `develop/v19.x` can receive MINOR and PATCH updates, but no MAJOR updates
 - ▶ Additional matching `release/vX.Y.Z` created which receives tags

Release strategy

- We follow semantic versioning
- We currently limit releases to MINOR updates
 - ▶ Target MINOR updates
- General development
 - ▶ Requires one additional branch
- Mainline tagged
- Before making a release
 - ▶ Should only require MINOR updates
- When required:
 - ▶ Should only require MINOR updates
- Example:
 - ▶ Created `develop` branch
 - ▶ `develop/v19.x` for development
 - ▶ Additional matching

```
tig 13
0000000 2022-09-23 15:02 +0100 Unknown
0000000 2022-09-23 15:02 +0100 Unknown
b655e18 2022-09-23 12:35 +0000 github-actions[bot]
a6b785f 2022-09-23 13:55 +0200 Paul Gessinger
e3cfd68 2022-09-23 13:52 +0200 acts-project-service
617a244 2022-09-23 13:26 +0200 Paul Gessinger
98aed00 2022-09-23 10:20 +0200 Paul Gessinger
c55f4c5 2022-09-23 09:48 +0200 Paul Gessinger
42c6af8 2022-09-22 19:49 +0200 Paul Gessinger
a4dc506 2022-09-22 17:39 +0200 Paul Gessinger
92fb4d9 2022-09-22 17:36 +0200 acts-project-service
b60219c 2022-09-22 16:46 +0200 Paul Gessinger
919e4fc 2022-09-22 15:38 +0200 acts-project-service
8c6aa54 2022-09-22 12:58 +0200 Paul Gessinger
b3fb5c3 2022-09-22 09:00 +0200 Paul Gessinger
78e04c8 2022-09-22 09:00 +0200 Paul Gessinger
d31e7cf 2022-09-20 09:59 +0200 acts-project-service
70d5869 2022-09-20 05:32 +0200 acts-project-service
fb1afdc 2022-09-20 04:41 +0200 acts-project-service
af87624 2022-09-20 03:47 +0200 acts-project-service
6bdf4bc 2022-09-20 01:39 +0200 acts-project-service
2c8f937 2022-09-19 23:12 +0200 Paul Gessinger
9431612 2022-09-19 22:30 +0200 acts-project-service
8e0ec65 2022-09-19 11:10 +0200 Paul Gessinger
bb82e7b 2022-09-14 14:50 +0200 acts-project-service
7582072 2022-09-07 09:10 +0000 github-actions[bot]
a919bba 2022-09-07 11:09 +0200 Paul Gessinger
0584907 2022-09-06 09:19 +0200 acts-project-service
d05e8d9 2022-09-02 13:15 +0200 acts-project-service
f3fcdcd8 2022-09-01 16:59 +0200 acts-project-service
2f2851b 2022-08-31 23:53 +0200 acts-project-service
4612409 2022-08-31 21:37 +0200 acts-project-service
6b7415b 2022-08-31 17:20 +0200 acts-project-service
b8c99eb 2022-08-31 14:20 +0200 Paul Gessinger
e1611b9 2022-08-23 20:51 +0200 acts-project-service
2477aed 2022-08-22 15:28 +0200 acts-project-service
2d6a24e 2022-08-22 14:35 +0200 acts-project-service
20fc5e2 2022-08-22 10:23 +0200 acts-project-service
93a7be6 2022-08-19 18:12 +0200 Paul Gessinger
867f126 2022-08-18 14:16 +0200 acts-project-service
91c9f65 2022-08-17 09:33 +0200 acts-project-service
9433998 2022-08-16 15:30 +0200 acts-project-service
a098828 2022-08-16 12:41 +0200 acts-project-service

o Untracked changes
o Staged changes
o [release/v19.x] {origin/release/v19.x} <v19.9.0> Bump to
M Merge remote-tracking branch 'origin/develop/v19.x' into
o [origin/develop/v19.x] fix: maybe-uninitialized warning
o refactor: SP builder update [backport #1218 to develop/
o [develop/v19.x] chore: Update policybot configuration [
o feat: Allow configurable particle selection and reprodu
o perf: Remove x87 elliptic integral in solenoid B-field
o refactor: improve full_chain_odd.py example [backport #
o refactor: improve full_chain_itk.py example [backport #
o fix: Set s_binningValues as const [backport of #1485 to
o feat: use RangeID<> class for extent [backport #1424 to
o refactor: Vertex performance monitoring [backport #1417
o feat: Add operator -> to Acts::Result [backport #1509 t
o refactor: python examples log level overwrite mechanism
o fix: ParticleSmearing options not setup in AMVF example
o fix: Added missing return to seedfinder::CreateSeedsFor
o feat: check if yOuter takes the square root of a negati
o fix: Make JsonMaterialDecorator IO error more explicit
o refactor: Improve material mapping speed [backport #145
o feat: addCKFTracks option to disable some output [backp
o feat: split deltaR in seedFinderOrthogonal [backport #1
o chore: Add priority label to kodiak config (#1533)
o fix: Sequencer correctly calls algorithm finalize [back
o <v19.8.0> Bump to version v19.8.0
M Merge remote-tracking branch 'origin/develop/v19.x' into
o refactor: Loop protection as a function [backport #1490
o feat: Checks for configuration of delta R values in see
o feat: Update RootVertexPerformanceWriter to work with t
o feat: core add particle stopped aborter [backport #1453
o feat: Identifier can be customized for sensitive surfac
o refactor: Remove dependency on deprecated std::binary_f
o refactor: no return for addx python helpers [backport o
o feat: Algorithm lifecycle methods: initialize & finaliz
o refactor: no return for addx python helpers [backport #
o [backport/1447-addCKFTracks] refactor: consistent namin
o feat: don't use TBB in Sequence(numThreads=1) [backport
o ci: Update macOS CI dependencies [backport of #1445 to
o refactor: debug print particle id value [backport #1435
o fix: navigator priority with negative boundary distance
o fix: full chain vertexing [backport #1299 to develop/v1
o fix: RootBFieldWriter incorrect output [backport #1381

[main] Untracked changes 0%
```

(want to review!)

developments

MINOR updates

Questions?

Backup

License block

```
// This file is part of the Acts project.  
//  
// Copyright (C) 2016-2020 CERN for the benefit of the Acts project  
//  
// This Source Code Form is subject to the terms of the Mozilla Public  
// License, v. 2.0. If a copy of the MPL was not distributed with this  
// file, You can obtain one at http://mozilla.org/MPL/2.0/.
```