

Carlo VARNI^{1,2}

on behalf of the Acts-ITk group and the ACTS developers

ATLAS ACTS infrastructure

ACTS Developers Workshop

IJCLab (Orsay, France)

07-10 November 2023

¹ UC Berkeley [US]

² Lawrence Berkeley National Laboratory [US]



Berkeley
UNIVERSITY OF CALIFORNIA

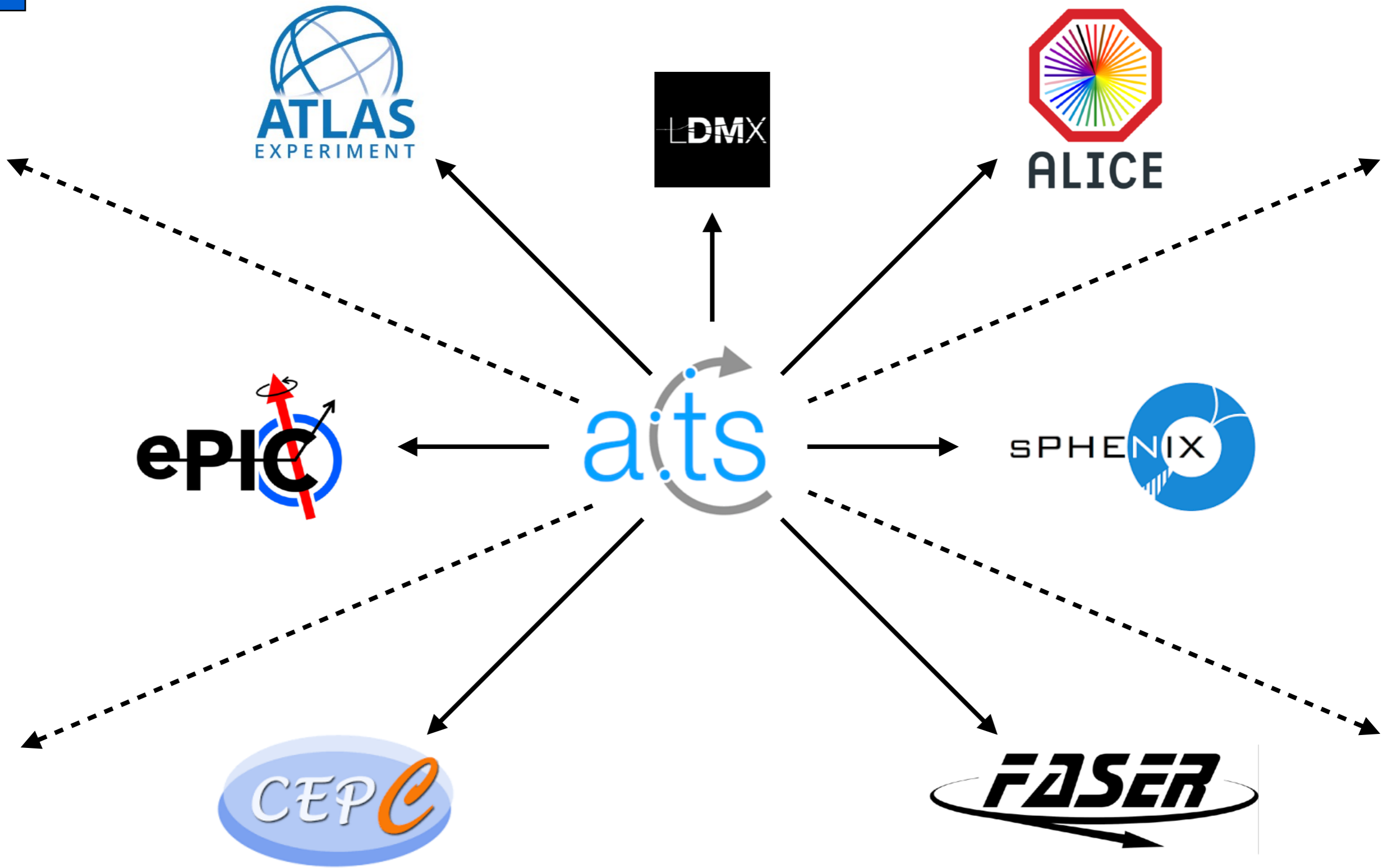
Overview



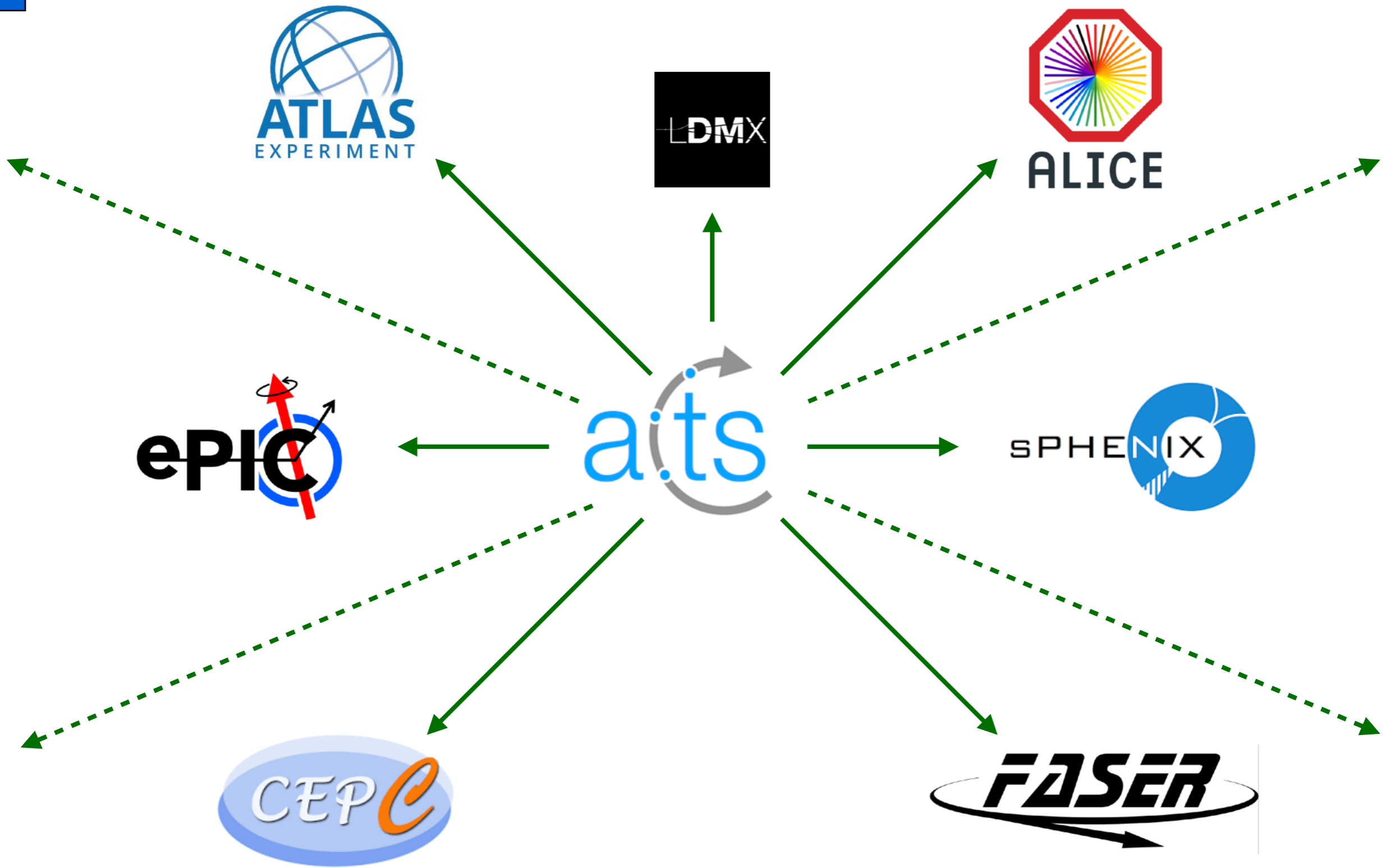
Latest ACTS Tag: v30.3.2

- **ACTS developments and current release policy**
 - ACTS cuts a new release ~once a week
 - Breaking changes in ACTS or changes causing experiment output changes:
MAJOR version (one **MAJOR** version per month at most)
 - **MINOR / PATCH** versions *should* keep compatibility
- **The integration problem**
 - What the challenges are when deploying ACTS to multiple experiments
 - Our strategy to solve the problem (from our experience with Athena)
- **The Athena experience**
 - Athena infrastructure
 - ACTS-Athena infrastructure

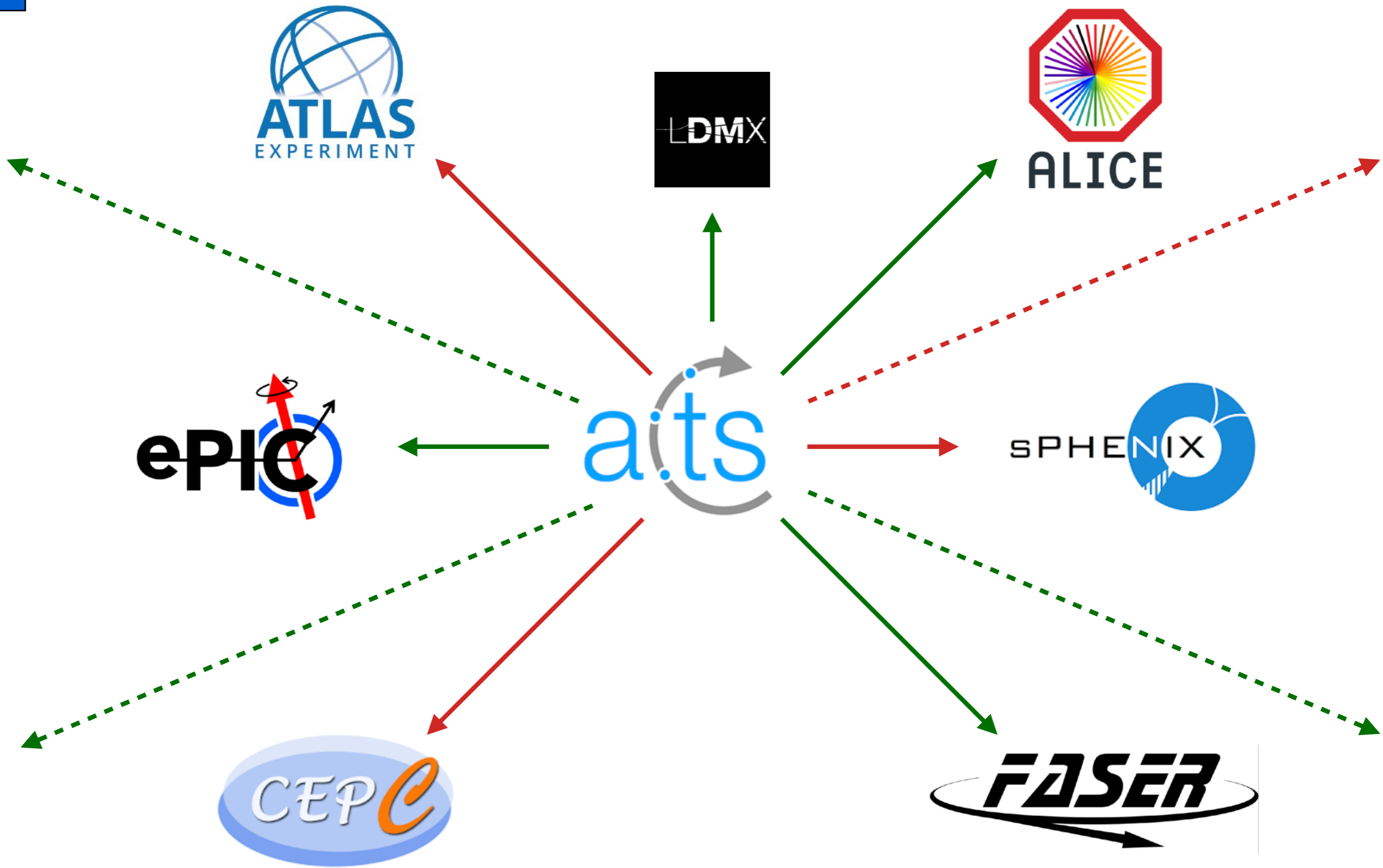
The Integration Problem



A Common Tracking Software (ACTS) is currently (and will be) used by several experiments but that poses several developments/integration challenges



In an ideal case, updates within ACTS will directly be deployed to experiments without issues



But our experience says otherwise and a minor change can make the bump to a new ACTS version not possible (for several reasons) without a patch

The integration problem

- The ACTS testing suite, albeit extensive, cannot cover the same level of details of all tests the experiments conduct on their software
 - The more experiments deploy ACTS in their workflows, the more extensive and robust the ACTS testing infrastructure must be
 - Hard (or even impossible) to cover experiment all specific corner cases
 - Issues may arise from interference of experiment-specific design and the ACTS software
- Seemingly harmless changes in ACTS core can have dire consequences for experiment
 - Changes of distributions downstream that trigger failures when comparing against references [**the good**]
 - Compilation/Run time issues that require an ACTS patch [**the bad**]
 - Mysterious seg fault that require a painfully-long amount of passionate debugging time [**the awful**]
- Athena has quite an extensive testing infrastructure, which some times made the ACTS integration definitely not smooth.
 - We have developed a plan on how to tackle the integration problem!

The integration problem

- The plan:

The ACTS testing infrastructure that currently checks the developments in the ACTS Core component



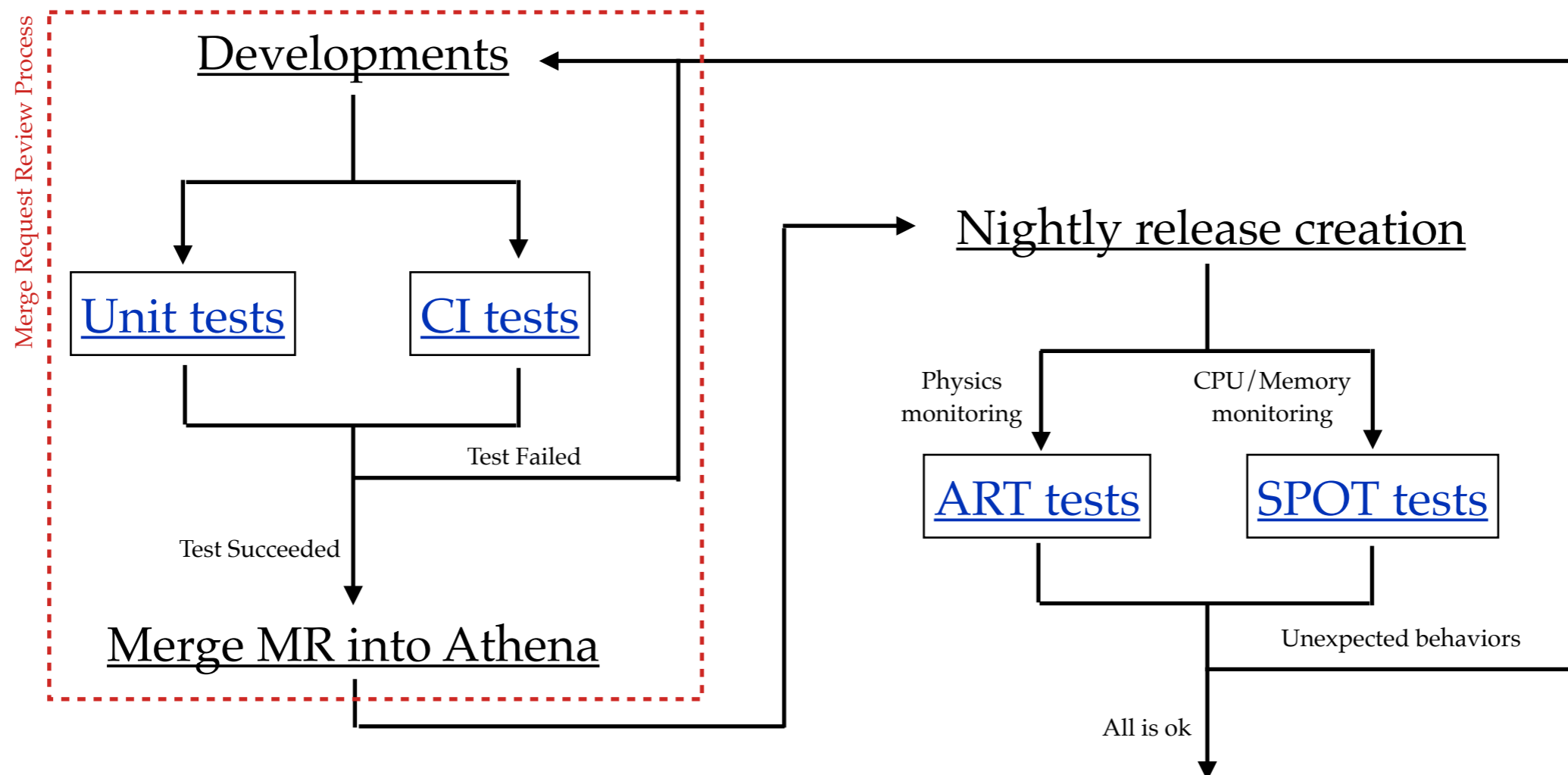
A testing infrastructure that, for each development in ACTS Core Component, checks the experiment reaction

The Athena (or any experiment) testing infrastructure that checks the developments in the experiment software

The Athena Infrastructure

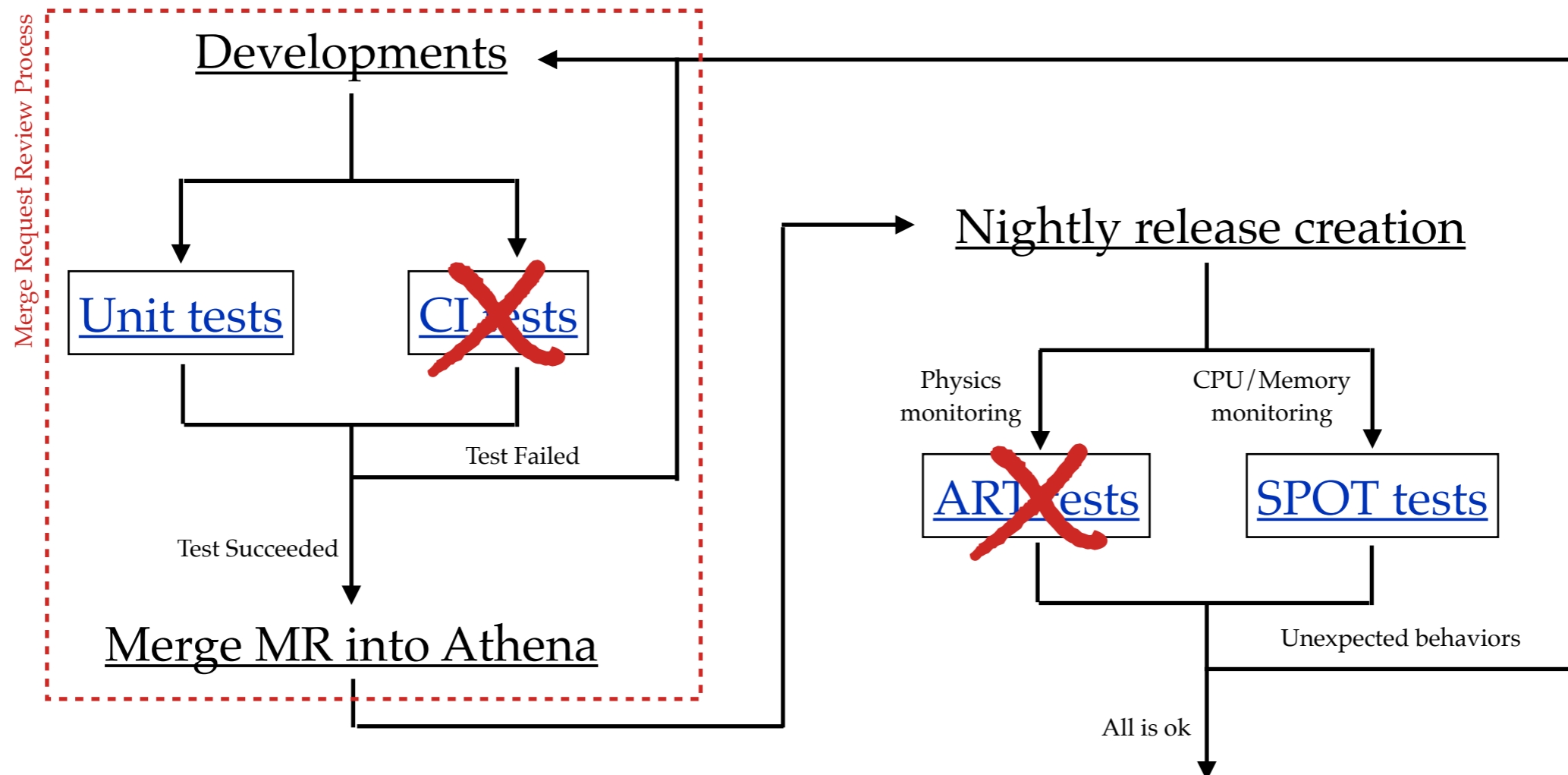
The Athena Infrastructure

- Developments in Athena rely on extensive tests in different stages of the process
 - Unit and CI tests are performed during the Merge Request review process.
Quick checks for testing specific modules or the entire code
 - ART and SPOT tests are performed after the MR has been merged on Athena nightly releases (produced once a day) for monitoring physics and CPU/memory performance respectively
- Some of these tests depends on ACTS or are explicitly for testing the ACTS-integrated pieces of code



The Athena Infrastructure

- An update on ACTS may break any of these tests
 - Extremely problematic if unit and CI tests gets broken → bump cannot be performed since, if merged, that will impact other unrelated MRs and block them too
 - Major issues may not be spotted at this stage but can break a nightly release
- A quick fix from ACTS may be needed to address the issue → immediate feedback and bidirectional communication is essential



The Athena Infrastructure

- Extremely pivotal to keep extending testing suite for ACTS related / dependent workflows from the experiment side!
 - The more corner cases we find the more robust our checks are
 - We are constantly adding tests with our integration effort to check the correct behaviors of all the ACTS-related code we insert in Athena

CITest_ACTS_ActsBenchmarkWithSpot-test
CITest_ACTS_ActsKfRefitting-test
CITest_ACTS_Propagation_ID-test
CITest_ACTS_Propagation_ITk-test
CITest_ACTS_ValidateActsCoreSpacePoints-test
CITest_ACTS_ValidateActsTrkSpacePoints-test

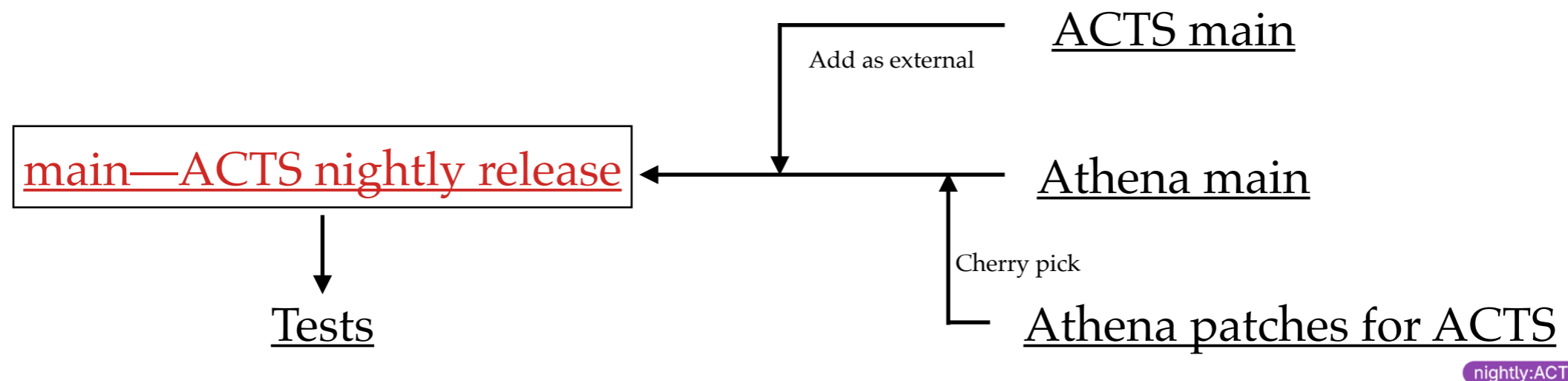
- Athena already deploys the ACTS vertex reconstruction in the main reconstruction workflow
 - So we may have effects also on non-ACTS-specific tests

CITest_ACTS_ValidateAmbiguityResolution-test	test_run4_acts_ckf_mu100GeV.sh	0	1	2	3	4	5	6	succeeded
CITest_ACTS_ValidateClusters-test	test_run4_acts_ckf_mu10GeV.sh	0	1	2	3	4	5	6	succeeded
CITest_ACTS_ValidateOrthogonalSeeds-test	test_run4_acts_ckf_mu1GeV.sh	0	1	2	3	4	5	6	succeeded
CITest_ACTS_ValidateSeeds-test	test_run4_acts_ckf_ttbar_PU0.sh						0	1	finished
CITest_ACTS_ValidateTracks-test	test_run4_acts_default_seeding_ttbar_PU200.sh						0	1	succeeded
CITest_ACTS_Workflow-test	test_run4_acts_ttbar_PU200.sh			0	1	2	3	succeeded	

The ACTS-Athena Infrastructure

The ACTS-Athena Infrastructure

- We want to catch failures as soon as possible
- Ideas
 - Monitor on a daily basis the effect on Athena (or any experiment ideally) given by the current ACTS main branch
 - Perform the Athena tests for every PR in ACTS before we create a new ACTS Tag
- We have a mechanism in place in Athena that allows to re-compile the software with a custom version of ACTS, add dedicated patches for Athena, and to produce a dedicated main—ACTS nightly release



The ACTS-Athena Infrastructure

The nightly release can then be used by developers for their developments

main—ACTS nightly release

Tests

Add as external

ACTS main

Athena main

Cherry pick

Athena patches for ACTS

nightly:ACTS

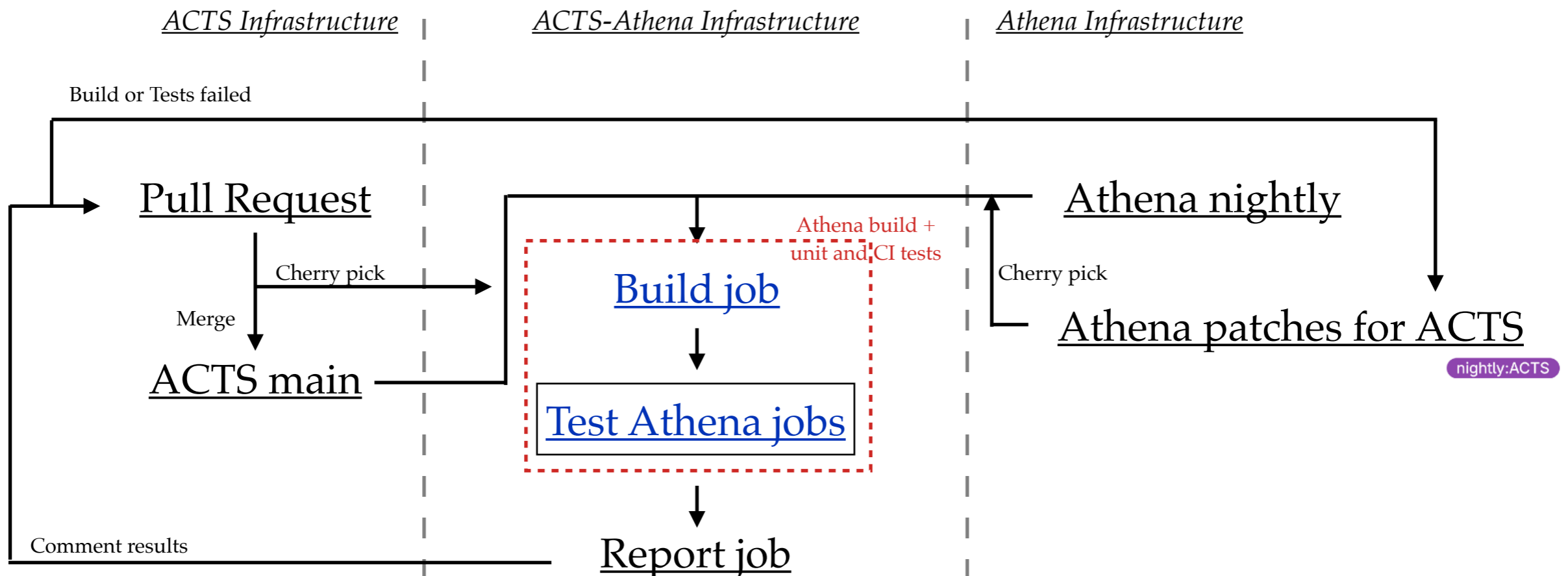
Release	Job time stamp	git clone	Extern. build	CMake config	Build time	Comp. errors (w/warn)	Install (CPack)	Test time	CTest errors (w/warn)	ART LOCAL	ART GRID	CVMFS (on server)	CVMFS (on client)	Host
2023-11-03T0401	2023/11/03 06:42	✔	✔	✔	2023/11/03 06:43	0 (0)	✔	2023/11/03 07:29	0 (0)	N/A	N/A			
2023-11-02T0401	2023/11/02 06:40	✔	✔	✔	2023/11/02 06:40	0 (0)	✔	2023/11/02 07:26	0 (0)	N/A	N/A			
2023-11-01T0401	2023/11/01 06:41	✔	✔	✔	2023/11/01 06:41	0 (0)	✔	2023/11/01 07:28	0 (0)	N/A	N/A	2023/11/01 07:45	2023/11/01 07:51	aibuild64-004
2023-10-31T0401	2023/10/31 06:41	✔	✔	✔	2023/10/31 06:41	0 (0)	✔	2023/10/31 07:28	0 (0)	N/A	N/A	2023/10/31 07:45	2023/10/31 07:51	aibuild64-004
2023-10-30T0401	2023/10/30 06:41	✔	✔	✔	2023/10/30 06:41	0 (0)	✔	2023/10/30 07:31	0 (0)	N/A	N/A	2023/10/30 07:48	2023/10/30 08:01	aibuild64-004
2023-10-28T0401	2023/10/28 06:39	✔	✔	✔	2023/10/28 06:39	0 (0)	✔	2023/10/28 07:29	0 (0)	N/A	N/A	2023/10/28 07:47	2023/10/28 07:51	aibuild64-004
2023-10-27T0401	2023/10/27 06:43	✔	✔	✔	2023/10/27 06:43	0 (0)	✔	2023/10/27 07:32	1 (1)	N/A	N/A			
2023-10-26T0401	2023/10/26 06:45	✔	✔	✔	2023/10/26 06:45	0 (0)	✔	2023/10/26 07:36	1 (1)	N/A	N/A			
2023-10-25T0401	2023/10/25 06:38	✔	✔	✔	2023/10/25 06:38	0 (0)	✔	2023/10/25 07:28	0 (0)	N/A	N/A	2023/10/25 07:46	2023/10/25 07:51	aibuild64-004

We test on daily base: we have at once all the PRs that have been merged in the previous 24 hours

Some tests fails. Need to check if they are ACTS related or issues from unrelated Athena areas

The ACTS-Athena Infrastructure

- But we can have better granularity by checking the single PRs
- Rely on the same (but light-weight) mechanism via API or Gitlab repository
 - We can compile Athena against ACTS main + merge PR or cherry picked still-to-merge PR
 - We run the proper Athena tests and report back to the original ACTS PR
 - We only compile a list of ACTS-related packages instead of the entire Athena software [thus the light-weight]



The ACTS-Athena Infrastructure



acts-project-service commented 3 days ago • edited ▾

Member ⋮

✓ Athena integration test results [5a05bca]

✓ All tests successful

status	job
●	run_workflow_tests_run2_data
●	run_unit_tests
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsBenchmarkWithSpot.sh 8 100
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsWorkflow.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateAmbiguityResolution.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateTracks.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateActsCoreSpacePoints.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateActsSpacePoints.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateSeeds.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateOrthogonalSeeds.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsValidateClusters.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsPersistifyEDM.sh
●	run_ci_tests: ../athena/AtlasTest/CITest/test/ActsKfRefitting.sh
●	run_ci_tests: python3 ../athena/Tracking/Acts/ActsGeometry/test/ActsExtrapolationAlgTest.py
●	run_ci_tests: python3 ../athena/Tracking/Acts/ActsGeometry/test/ActsITkTest.py
●	run_workflow_tests_run4_mc
●	run_workflow_tests_run3_mc
●	run_workflow_tests_run3_data



- The ACTS-Athena infrastructure will add a comment to the PR in ACTS
 - Details and links to the tests for checking the log files
- In case the build or tests failed labels will be added to the PR

Breaks Athena build

This PR breaks the Athena build

Fails Athena tests

This PR causes a failure in the Athena tests

Apogee

Apogee

- Validation Tool that collects commits from ACTS
 - Allows to keep track of pipelines triggered for PRs to ACTS
 - Groups triggered pipelines by ACTS commit
- It's now public (requires CERN and GitHub login)
 - <https://apogee.app.cern.ch>

List of all the PRs to ACTS main

The screenshot displays two entries from a list of pull requests. Each entry includes the PR title, author, timestamp, commit hash, and a pipeline status. The first entry is for PR #2592 by @andiwand, with pipeline #6434628. The second entry is for PR #2620 by @andiwand, with pipeline #6434323. A red dashed box highlights the action buttons (Reload, More, Revert) for the first entry.

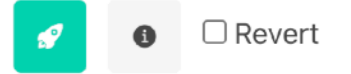
PR Title	Author	Timestamp	Commit Hash	Pipeline ID	Created	Last Updated	Actions
refactor!: Make `BoundaryCheck` constructor from `bool` explicit (#2592)	@andiwand	2023-11-04 11:48:42	2697f5a55	Pipeline #6434628	Created about 1 hour ago	last updated 43 minutes ago	Reload, More, Revert
refactor!: Rename and rearrange propagator options (#2620)	@andiwand	2023-11-04 08:24:55	c5789236e	Pipeline #6434323	Created about 4 hours ago	last updated 43 minutes ago	Reload, More, Revert

User can trigger actions and additional info via simple interface

Apogee

refactor!: Rename and rearrange propagator options (#2620)

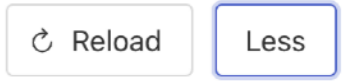
@andiwand , 2023-11-04 08:24:55 c5789236e



Add note

Pipeline #6434323

Created about 5 hours ago, last updated about 1 hour ago



Variable	Value
SOURCE_SHA	c5789236e9163b2c70f4a31ac6b234597935345a

```

build
  build_acts_athena

test
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsBenchmarkWithSpot.sh 8 100]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsKfRefitting.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsPersistifyEDM.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateActsCoreSpacePoints.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateActsSpacePoints.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateAmbiguityResolution.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateClusters.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateOrthogonalSeeds.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateSeeds.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsValidateTracks.sh]
  run_ci_tests: [../athena/AtlasTest/CITest/test/ActsWorkflow.sh]
  run_ci_tests: [python3 ../athena/Tracking/Acts/ActsGeometry/test/ActsExtrapolationAlgTest.py]
  run_ci_tests: [python3 ../athena/Tracking/Acts/ActsGeometry/test/ActsTkTest.py]
  run_unit_tests
  run_workflow_tests_run2_data
  run_workflow_tests_run3_data
  run_workflow_tests_run3_mc
  run_workflow_tests_run4_mc

report
  report_pull_request

```

Status of the ACTS-Athena pipeline is reported

Details on the specific successful/failing tests is available with links to log files

Apogee



Features

- Can associate patches for each ACTS main commit
- Can trigger pipelines out of order
 - Allows to include patches up to a specific commit and verify changes on a commit-by-commit base
- Can add notes to commits to keep track of what tests changed and accordingly set the labels to the PR

The screenshot displays a GitHub pull request interface. At the top, it shows '1 Open' and '75 Closed' pull requests. Below this is a list of commits with their titles, labels, and merge status. The commit '#2569' is highlighted, and a red arrow points to its 'Add note' button. The detailed view of this commit shows the title 'ci: Add clang-tidy check for &&, || and ! instead of and, or and not (#2569)', the author '@paulgessinger', the date '2023-10-31 11:42:14', and the commit hash '1f6dd58d0'. Below the commit details, there is a 'Pipeline #6414775' section with a 'Reload' button and a 'More' button. The bottom of the screenshot shows the start of another commit entry.

Apogee



- Can also monitor PRs before they are merged
 - Newly added feature to validate open PRs and their validation status
 - Allows to add patched to PRs

feat: allow custom precision for volume grid building (#2609)  

@asalzburger, created 2023-11-01 13:19:28, updated 2023-11-03 13:23:07

asalzburger:feat-small-detector-fixes @ bfbf7b5a2 , 10 commits, No pipeline

bfbf7b5a2 0852023bf c7236ac82 7441eb972 cc01012b2 51d41ee9d cce18d6fe 074ce8a40 847e674c6 5eec5e119

refactor: Use `std::size_t` consistently (#2624)  

@paulgessinger, created 2023-11-03 13:21:55, updated 2023-11-03 13:22:51

paulgessinger:use-std-size_t @ 656823a50 , 2 commits, No pipeline

656823a50 ababa3923

All the commits in the PR

Apogee

- Run the pipeline on specific commits

- The pipeline is not run automatically for every commit
- User has to manually trigger the pipeline
 - Possibility to select any commit (even old ones) and run the pipeline on it

The screenshot shows a GitHub pull request for the branch 'refactor: use time info in Kalman vertex fit (#2544)' by user '@felix-russo'. The pull request was created on 2023-10-14 and updated on 2023-10-31. It is associated with the repository 'felix-russo:always-d-time-fitting' and has 30 commits. A pipeline run is shown with ID '#6388828'. Below the commit list, there are two callout boxes: one pointing to a rocket icon labeled 'One can check the pipeline for a specific commit', and another pointing to a selected commit '136d9f777' labeled 'Select the commit and run the pipeline'.

refactor: use time info in Kalman vertex fit (#2544)
@felix-russo, created 2023-10-14 16:35:39, updated 2023-10-31 19:26:51
felix-russo:always-d-time-fitting @ fc3794688 , 30 commits, #6388828

a91cca172	3e0165663	63979d41b	256a1964d	6e14fecdd	a5099ec47	1b4f45463	68d8595e7	392a90e96	694360593
63c59d0fb	322d403d7	694ecb2f1	896cbd718	136d9f777	7ecb649ea	8e29ab5d7	1b4fb4956	600f20071	c8773b38e
cdac0246f	f9625fc6f	5a2527086	2cc3b4b99	ee3f31754	49551286d	a8931de6d	cc6824138	fbeefbb3e	441b25cc4

One can check the pipeline for a specific commit

Select the commit and run the pipeline

Conclusions

- The integration of ACTS into several experiments is not a trivial task
 - ACTS software must be reliable and deployable to all costumers. Not a trivial task
- Developed a strategy from our experience with Athena
 - Quite extensive infrastructure in place to monitor effects of PRs to ACTS main on the experiment
 - The goal is to catch problems as early as possible and to always offer a fully working ACTS tags for all experiments
 - The more experience we obtain, the more robust and complete the infrastructure is
- Quite a complex mechanism
 - Hidden behind a new validation tool [Apogee](#)
 - Provide an intuitive and easy to use API to improve the user experience
- Ideally, we will have one day the same mechanism extended to all the experiments