

# CDash query tutorial for triaging systemic Trilinos PR build and test failures



Roscoe A. Bartlett  
Department 1424  
Software Engineering and Research

October 27, 2022

Trilinos Users Group Meeting, Developers Day

# Current Trilinos Pull Request Testing Process?



## Most common observed Trilinos Pull Request (PR) process outline:

1. Push a topic branch to GitHub Trilinos fork
2. Create a Trilinos GitHub PR against the target branch 'develop'
3. Wait for PR autotester to run a set of PR builds which post to internal (SRN) CDash site [trilinos-cdash.sandia.gov](http://trilinos-cdash.sandia.gov)
4. Examine PR build & test failures on [trilinos-cdash.sandia.gov](http://trilinos-cdash.sandia.gov) (SRN only)
  1. Are these failures likely due to changes in the PR?
5. Resolve PR build & test failures:
  1. If failures are likely unique to this PR topic branch:
    1. Create new commit(s) to address the failures (offline testing common or rare?)
    2. Push new commits topic branch
  2. If failures are likely **not** unique to this PR topic branch:
    1. Assume failures are random and will go away if you run enough PR build iterations
6. Set the '**AT: RETEST**' label on the PR and **go to Step 3**

## Problems with setting '**AT: RETEST**' without any further investigation/actions:

- Errors may not be random and may take out your next PR build & test iteration
  - The computational cost of running the builds will be wasted
  - These doomed PR builds may actually be blocking a PR build for another PR that may fix things!
- Systemic problem with PR builds or testing process is not triaged and will break future PR test iterations (including your own future PRs!)

# Who should be responsible for triaging and reporting systematic Trilinos PR testing failures?



## Should the Trilinos Framework Team members do it?

- Trilinos Framework team members don't monitor the raw CDash results.
- Trilinos Framework team members are not familiar with the Trilinos test suite and understand how to interpret failures.
- Trilinos Framework team members don't understand the nature of the changes in a PR topic branch and are not well qualified to know what failures may be due to these changes.

## Should the Trilinos Developers associated with the failing PR builds do it?

- Trilinos developers **do** understand the nature of the changes in the PR topic branch and can typically determine if a build or test failure is due to the topic branch changes or not.
- Trilinos developers **do** understand the Trilinos test suite.

**Conclusion => The Trilinos developers looking at the PR build & test results on CDash are in the best place to triage and report systemic errors!**

# Proposed Trilinos Pull Request Process



1. Push a topic branch to GitHub Trilinos fork
2. Create a Trilinos GitHub PR against the target branch 'develop'
3. Wait for PR autotester to run a set of PR builds which post to internal (SRN) CDash site `trilinos-cdash.sandia.gov`
4. Examine PR build & test failures on `trilinos-cdash.sandia.gov` (SRN only)
  1. Are failures unique to this PR topic branch or are they impacting other recent PR builds as well?
  2. Are these infrastructure failures not related to any problem in Trilinos itself?
5. Resolve PR build & test failures:
  1. If failures are likely unique to this PR topic branch:
    1. Reproduce PR build and test failures offline
    2. Create new commit(s) to address the failures and verify in offline builds
    3. Push new commits topic branch
  2. If failures are due to problems in Trilinos 'develop':
    1. Run CDash queries to determine how wide spread and how long failures occurring
    2. Create Trilinos GitHub Issue for the failure and inform responsible parties
    3. Pin the new GitHub Issue to inform other Trilinos Developers
    4. **Wait for someone to address the problem causing the failures (if the failures are not random)**
6. Set the '**AT: RETEST**' label on the PR and **go to Step 3**

# Trilinos Failure Reporting and Resolution Process



<https://github.com/trilinos/Trilinos/wiki/Addressing-test-failures>

First, **create a Trilinos GitHub Issue that lists the full names for the failing tests and what builds they are occurring** (e.g. Issue #10847). (This allows any Trilinos developer to search the GitHub issues to see if an issue already exists for the failure.)

- NOTE: GitHub Issue searches do **not** support partial string matching (e.g. "Ifpack\_Container" will not match GitHub Issues/PRs that list "Ifpack\_Container\_MPI\_4")

Once the Trilinos GitHub issue has been created, the test failures can be addressed by one of following processes:

- A. Fix the failure (if the defect can be fixed quickly):
- B. Temporarily disable the failing code or test (if the defect is not critical or if it is a defective test)
  - Set the GitHub label "Disabled Tests"
- C. Back out the merge commit that caused the failures (if it is a build failure or some other critical defect)

# Reporting Trilinos Infrastructure Failures Process?



1. Create Trilinos GitHub Issue describing the problem (with appropriate level of details for external GitHub, consider pinning the issue)
2. Create internal Trilinos HelpDesk issue duplicating the GitHub Issue title and put in link to the GitHub Issue
3. Put in link from the GitHub Issue to the internal Trilinos HelpDesk issue

## Example:

Open Framework: Trilinos packages/subprojects ordering incorrect in Trilinos PR builds #10979  
bartlettroscoe opened this issue on Sep 1 · 0 comments

bartlettroscoe on Sep 1 · edited Member

### Bug Report

@trilinos/framework

#### Internal Issues

- TRILINOSH-197

CCR Help Center / Trilinos Help Desk / TRILINOSH-197

### Framework: Trilinos packages/subprojects ordering incorrect in Trilinos PR builds **WORK IN PROGRESS**

...

Details 01/Sep/22 1:28 PM

...

See:

- <https://github.com/trilinos/Trilinos/issues/10979>

Product Area(s)  
Framework

**Request for Trilinos Framework Team members:** Please update the linked GitHub issue to inform all Trilinos developers about the status of a systematic framework failures?

# Triaging Trilinos PR build failures on CDash



Are the build errors seen in my PR also seen in other PRs recently?

⇒ Run queries on <https://trilinos-cdash.sandia.gov/index.php?project=Trilinos>

Useful fields:

- "Group"
- "Site"
- "Build Name"
- "Build Errors"
- "Subprojects"

The screenshot shows the 'Filters' section of the CDash interface. A dropdown menu is open, displaying a list of fields and comparison operators. The 'Build Name' field is selected, and the 'contains' operator is chosen. The interface includes a search input field, minus and plus buttons, and a 'Help' link.

Field	Comparison Operator
Build Name	contains
Build Duration	– choose comparison –
Build Errors	contains
Build Warnings	does not contain
Build Name	is
Build Stamp	is not
Build Start Time	starts with
Build Type	ends with
Configure Duration	
Configure Errors	
Configure Warnings	
Expected	
Group	
Has Coverage	
Has CTest Notes	
Has Dynamic Analysis	
Has User Notes	
Label	
Revision	
Site	
Submission Client	
SubProjects	
Tests Duration	
Tests Failed	
Tests Not Run	
Tests Passed	
Tests Timing Failed	
Update Duration	
Updated Files	

# Example CDash build queries



[Trilinos GitHub #10836](#): Framework: Mysterious 6 failures in package Zoltan2Sphynx with no data on CDash going back to at least 2022-05-18

**Filters** Help

- Group  Pull Request
- SubProjects  Zoltan2Sphynx
- Build Errors  6

Begin  End

**Pull Request** *160 builds* [view timeline]

Site	Build Name	Configure			Build			Test			Start Time	Labels	
		Error	Warn	Time	Error	Warn	Time	Not Run	Fail	Pass			
ascic142	PR-10808-test-rhel7_sems-gnu-7.2.0-openmpi-1.10.1-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rtc_no-uvmm_deprecated-on_no-package-enables-883	0	1	1m 30s	6	0	4h 48m 24s	0	0	0	5m 34s	Aug 15, 2022 - 01:02 MDT	Zoltan2Sphynx
ascic142	PR-10802-test-rhel7_sems-gnu-7.2.0-openmpi-1.10.1-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rtc_no-uvmm_deprecated-on_no-package-enables-881	0	1	1m 27s	6	0	4h 4m	0	0	0	1s	Aug 14, 2022 - 20:36 MDT	Zoltan2Sphynx



# Triaging Trilinos PR test failures on CDash



Are the test errors seen in my PR also seen in other PRs recently?

⇒ Run queries on <https://trilinos-cdash.sandia.gov/queryTests.php?project=Trilinos>

Useful fields:

- "Group"
- "Build Name"
- "Site"
- "Status"
- "Test Output"

Hide Filters  
Show Matching  
Output

The screenshot shows the CDash query builder interface. It features a 'Filters' section with four filter rules:

Field	Comparison	Value	Operator
Group	is	Pull Request	and
Build Name	contains	<full-build-name>	and
Status	is	failed	and
Test Output	include regex		

The 'Test Output' dropdown menu is open, showing a list of fields: Build Name, Build Start Time, Details, Group, Label, Revision, Site, Status, Test Name, Test Output (highlighted), and Time. The 'include regex' option is selected in the dropdown.

# Example CDash test query



[Trilinos GitHub #11072](#): Ifpack: Ifpack\_Container tests failing in cuda-11.4.2 PR builds for all recorded history

Query Tests: 20 matches

[Hide Filters](#)  
[Hide Matching](#)  
[Output](#)

**Filters** Help

- Group is Pull Request
- Build Name contains cuda-11.4.2
  - Test Name is Ifpack\_Container\_LL\_MPI\_4
  - Test Name is Ifpack\_Container\_MPI\_4
- Status is failed
- Test Output include regex Test .TestContainer.exe. FAILED

Begin  End

Site	Build Name	Test Name	Status	Time	Details	Build Time	Matching Output
ascicgpu050	PR-10965-test-rhel7_sems-cuda-11.4.2-sems-gnu-10.1.0-sems-openmpi-4.0.5_release_static_Volta70_no-asan_complex_no-fpic_mpi_pt_no-rtc_no-uvdm_deprecated-on_no-package-enables-683	Ifpack_Container_LL_MPI_4	Failed	2s 360ms	Completed (Failed)	2022-09-27T11:38:58 MDT	Test `TestContainer.exe' FAILED! Container type = dense NumMyRows = 225, NumVectors = 3   x_exact - x  _2 = 6.11773 ===== Ifp
ascicgpu050	PR-10965-test-rhel7_sems-cuda-11.4.2-sems-gnu-10.1.0-sems-openmpi-4.0.5_release_static_Volta70_no-asan_complex_no-fpic_mpi_pt_no-rtc_no-uvdm_deprecated-on_no-package-enables-683	Ifpack_Container_MPI_4	Failed	2s 310ms	Completed (Failed)	2022-09-27T11:38:58 MDT	0 ===== Test `TestContainer.exe' FAILED! Test `TestContainer.exe' FAILED! Test `TestContainer.exe' FAILED! Test `TestContain

# Creating Trilinos GitHub Issue for PR test failures



Filters Help

Group	is	Pull Request	-	+	and
Build Name	contains	ats2_cuda-10.1.243-gnu-8.3.1-spmi-rolling_release_static_Volta70_Power9	-	+	and
Test Name	is	Stokhos_KokkosViewUQPCEUnitTest_Serial_MPI_1	-	+	and
Status	is	failed	-	+	and
Test Output	exclude regex	(Error initializing RM connection jsrun return value: 255)	-	+	

Begin  End

```
https://trilinos-cdash.sandia.gov/queryTests.php?project=Trilinos&begin=2022-05-01&end=2022-10-26&filtercount=5&showfilters=1&filtercombine=and&field1=groupname&compare1=61&value1=Pull%20Request&field2=buildname&compare2=63&value2=ats2_cuda-10.1.243-gnu-8.3.1-spmi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rtc_uvm_deprecated-on_no-package-enables-&field3=testname&compare3=61&value3=Stokhos_KokkosViewUQPCEUnitTest_Serial_MPI_1&field4=status&compare4=61&value4=failed&field5=testoutput&compare5=96&value5=(Error%20initializing%20RM%20connection%7Cjsrun%20return%20value%3A%20255)
```

Generate the Markdown for a new GitHub issue for these failing tests:

```
cd <some-base-dir>/
```

```
trilinosDir>/commonTools/framework/github_issue_creator/create_trilinos_github_test_failure_issue_driver.sh -u "https://trilinos-cdash.sandia.gov/queryTests.php?project=Trilinos&begin=2022-05-01&end=2022-10-26&..." "
```

```
emacs newGithubMarkdownIssueBody.md
```

# Creating Trilinos GitHub Issue for PR test failures



bartlettroscoe 21 days ago • edited

Member

CC: @trilinos/stokhos

## Next Action Status

## Description

As shown in [this query](#) (click "Shown Matching Output" in upper right) the tests:

- `Stokhos_KokkosViewUQPCEUnitTest_Serial_MPI_1`

in the unique GenConfig builds:

- `ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables`

started failing on testing day 2022-05-01.

The specific set of CDash builds impacted where:

- `PR-10472-test-ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables-911`
- `PR-10472-test-ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables-915`
- `PR-10571-test-ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables-1113`
- `PR-11086-test-ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables-1182`
- `PR-11099-test-ats2_cuda-10.1.243-gnu-8.3.1-spmpi-rolling_release_static_Volta70_Power9_no-asan_no-complex_no-fpic_mpi_pt_no-rdc_uvm_deprecated-on_no-package-enables-1211`



# Questions and Comments?