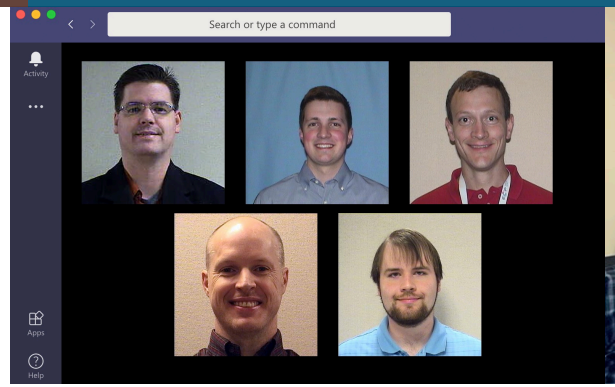
A Common Tool for Building Trilinos: Introduction to GenConfig



PRESENTED BY

Evan Harvey, Joshua Braun, James Willenbring

Sandia National Laboratories

Deploy
Research
Develop

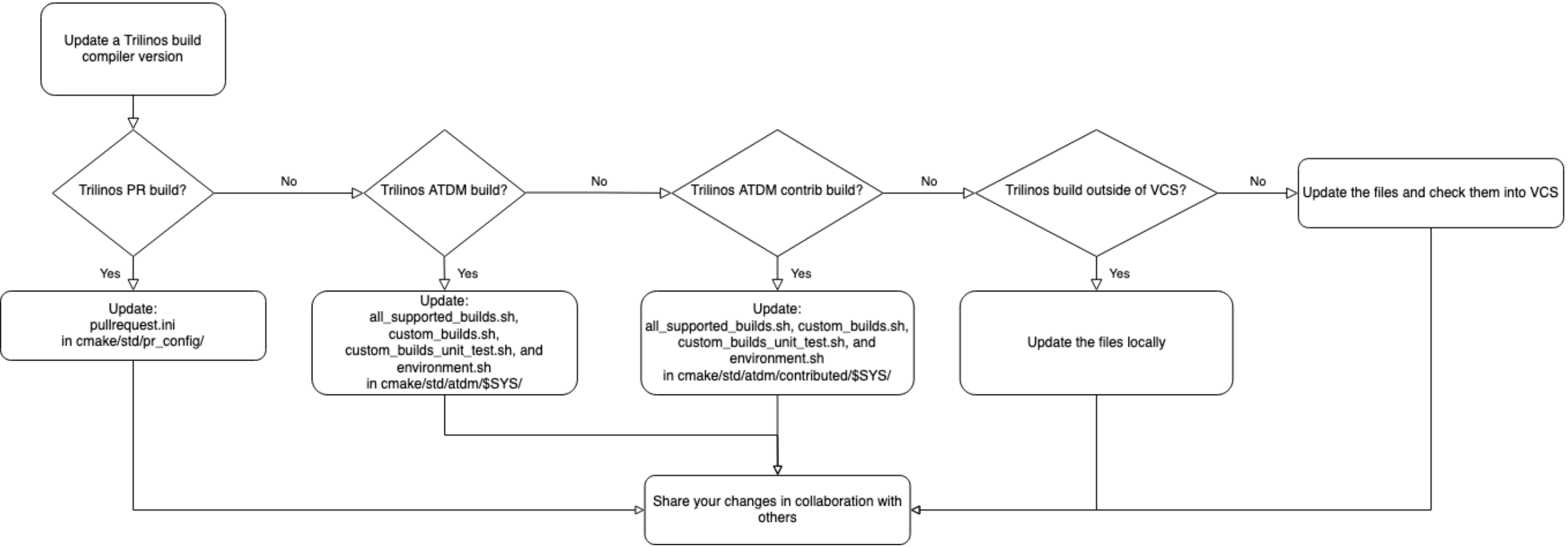
Software Engineering & Research
Department 1424

U.S. DEPARTMENT OF ENERGY
NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2021-14992 PE

Current approach to configuring Trilinos



This is complicated… I'll just send an email asking someone to update the compiler version.

Introduction to GenConfig

- Environment and cmake script tied to string
  - Complete configuration: *complete_config*
- Written in Python and unit-tested via pytest
  - 99% line coverage
- Offers two interfaces: CLI and API
  - CLI: *gen-config.sh*
  - API: *class gen_config.GenConfig*

How GenConfig uses LoadEnv

- LoadEnv also written in Python and unit-tested via pytest
  - 98% line coverage

- Environment tied to a
  - Complete environment name: *complete_env*
    - Note: *complete_config* begins with *complete_env*

- GenConfig loads environment via *class load_env.LoadEnv*
  - CLI: gen-config.sh modifies user's subshell
  - API: gen_config.py modifies python process

Developer use-case (gen-config.sh)

```
[josbrau@weaver11 build]$ source ~/GenConfig/gen-config.sh --list-configs
Using system 'rhel7' based on matching hostname 'weaver11'.

+=============================================================================+
|   INFO:  Please select one of the following complete configurations from
|          /home/josbrau/GenConfig/examples/trilinos/config-specs.ini
|
|     - rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_no-uvm_pr-framework
|     - 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-rdc_no-uvm_no-package-enables
|     - rhel7_sems-gnu-7.2.0-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-gnu-7.2.0-openmpi-1.10.1-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_pr
|     - rhel7_sems-gnu-7.2.0-openmpi-1.10.1-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-gnu-8.3.0-openmpi-1.10.1-openmp_release-debug_static_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_pr
|     - rhel7_sems-gnu-8.3.0-openmpi-1.10.1-openmp_release-debug_static_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-clang-7.0.1-openmpi-1.10.1-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-clang-9.0.0-openmpi-1.10.1-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-clang-10.0.0-openmpi-1.10.1-serial_release-debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling_release_static_Volta70_Power9_no-asan_complex_fpic_mpi_pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling_release_static_Volta70_Power9_no-asan_complex_fpic_mpi_pt_no-rdc_uvm_no-package-enables
|     - rhel7_sems-intel-17.0.1-mpich-3.2-serial_release-debug_static_no-kokkos-arch_no-asan_no-complex_fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|     - rhel7_sems-intel-19.0.5-mpich-3.2-serial_release-debug_static_no-kokkos-arch_no-asan_no-complex_fpic_mpi_no-pt_no-rdc_no-uvm_no-package-enables
|
+=============================================================================+
```

```
---------------------------------[ Examples ]---------------------------------

NOTE:  In each of the following examples, GenConfig first runs
       LoadEnv to load the correct environment.

Run CMake Using Configure Flags from GenConfig:

    source /path/to/gen-config.sh \
        <build-name> \
        /path/to/src

    NOTE:  /path/to/src must always be specified as the last command
           line argument UNLESS the --cmake-fragment flag is used.

Save CMake Fragment File to Use with CMake:

    source /path/to/gen-config.sh \
        --cmake-fragment foo.cmake \
        <build-name>

    cmake -C foo.cmake /path/to/src
```

Developer use-case (gen-config.sh)

```
[josbrau@weaver11 build]$ source ~/GenConfig/gen-config.sh --cmake-fragment cuda_10_1_105.cmake \
>   rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling_release_static_Volta70_Power9_no-asan_complex_fpic_mpi_pt_no-rdc_uvm_no-package-enables
```

```
----------------------------------[ Examples ]---------------------------------

NOTE:  In each of the following examples, GenConfig first runs
       LoadEnv to load the correct environment.

Run CMake Using Configure Flags from GenConfig:

    source /path/to/gen-config.sh \
        <build-name> \
        /path/to/src

    NOTE:  /path/to/src must always be specified as the last command
           line argument UNLESS the --cmake-fragment flag is used.

Save CMake Fragment File to Use with CMake:

    source /path/to/gen-config.sh \
        --cmake-fragment foo.cmake \
        <build-name>

    cmake -C foo.cmake /path/to/src
```

Developer use-case (gen-config.sh)

```
[josbrau@weaver11 build]$ source ~/GenConfig/gen-config.sh --cmake-fragment cuda_10_1_105.cmake \
>   rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling_release_static_Volta70_Power9_no-asan_complex_fpic_mpi_pt_no-rdc_uvm_no-package-enables
```

```
********************************************************************************
         E N V I R O N M E N T  L O A D E D  S U C E S S F U L L Y
********************************************************************************



********************************************************************************
         T Y P E  exit  T O  L E A V E  T H E  E N V I R O N M E N T
********************************************************************************


********************************************************************************
                    B E G I N  C O N F I G U R A T I O N
********************************************************************************


Please run:

  $ cmake -C /path/to/fragment.cmake /path/to/src

where "/path/to/fragment.cmake" is replaced with your generated cmake fragment file
and "/path/to/src" is replaced with your build source.

(rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling) $ ls
cuda_10_1_105.cmake
(rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling) $ cmake -C cuda_10_1_105.cmake ~/Trilinos/
```

```
---------------------------------[ Examples ]---------------------------------

NOTE:  In each of the following examples, GenConfig first runs
       LoadEnv to load the correct environment.

Run CMake Using Configure Flags from GenConfig:

    source /path/to/gen-config.sh \
        <build-name> \
        /path/to/src

    NOTE:  /path/to/src must always be specified as the last command
           line argument UNLESS the --cmake-fragment flag is used.

Save CMake Fragment File to Use with CMake:

    source /path/to/gen-config.sh \
        --cmake-fragment foo.cmake \
        <build-name>

    cmake -C foo.cmake /path/to/src
```

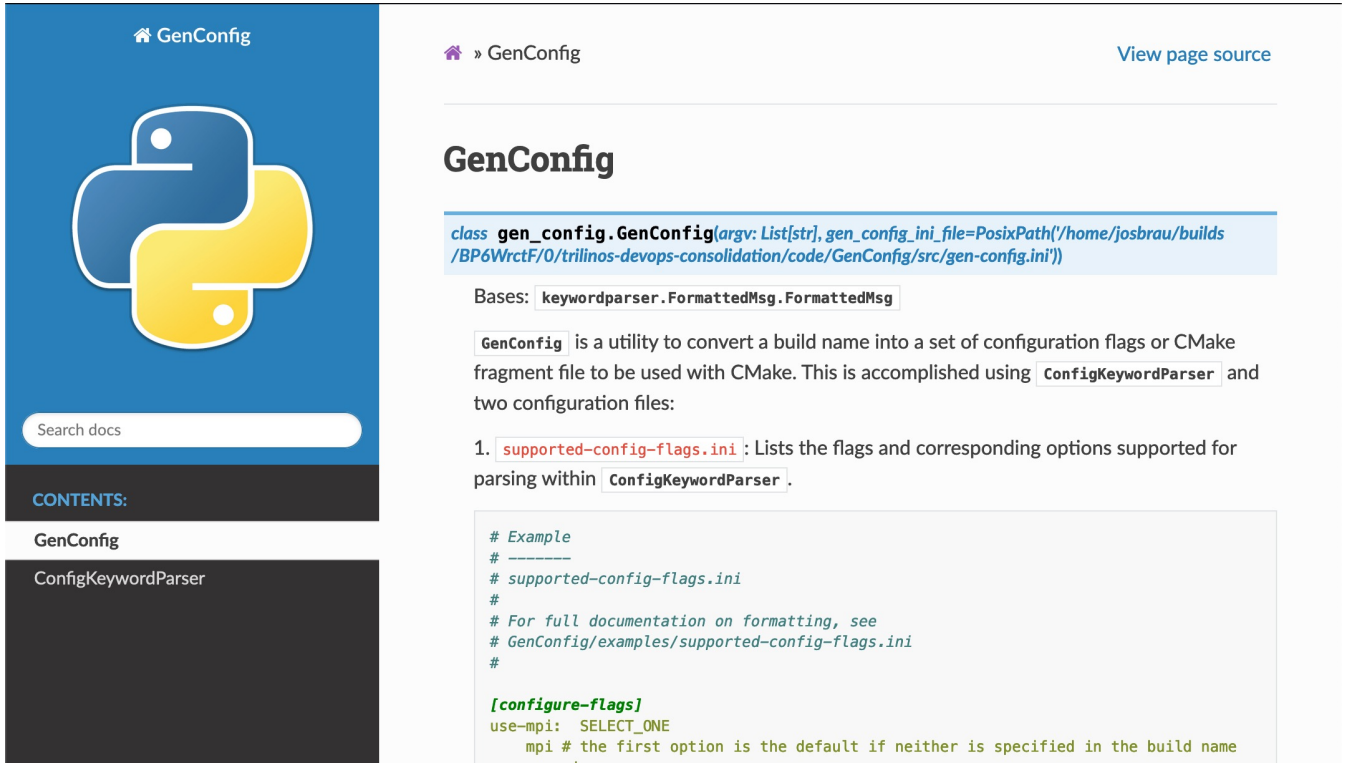
- GenConfig uses LoadEnv to put the user into a subshell with the right environment loaded.

PR testing use-case (class gen_config.GenConfig)

```
# example.py
from gen_config import GenConfig


build_name = "rhel7_cuda-10.1.105-gnu-7.2.0-spmpi-rolling_release_static_Volta70_Power9_no-asan_complex_fpic_mpi_pt_no-rdc_uvm_no-package-enables"
gc = GenConfig(["--cmake-fragment", "cuda_10_1_105.cmake", build_name])
gc.write_cmake_fragment()
```

More information in the GenConfig class documentation:



Introduction to config-specs.ini

- Single file where every *complete_config* is defined

- Focuses on:
  - human readability
  - reusability

- For example, this is the config for the *python-3* PR job:

```
[rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
use RHEL7_SEMS_COMPILER|GNU
use NODE-TYPE|SERIAL
use BUILD-TYPE|DEBUG
use RHEL7_SEMS_LIB-TYPE|SHARED
use KOKKOS-ARCH|NO-KOKKOS-ARCH
use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
use USE-COMPLEX|NO
use USE-RDC|NO

use COMMON
```

Introduction to config-specs.ini

- *complete_config*

```
1139  [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140  use RHEL7_SEMS_COMPILER|GNU
1141  use NODE-TYPE|SERIAL
1142  use BUILD-TYPE|DEBUG
1143  use RHEL7_SEMS_LIB-TYPE|SHARED
1144  use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145  use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
1146  use USE-COMPLEX|NO
1147  use USE-RDC|NO
1148
1149  use COMMON
```

Introduction to config-specs.ini

- *complete_env*

```
1139  [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140  use RHEL7_SEMS_COMPILER|GNU
1141  use NODE-TYPE|SERIAL
1142  use BUILD-TYPE|DEBUG
1143  use RHEL7_SEMS_LIB-TYPE|SHARED
1144  use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145  use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
1146  use USE-COMPLEX|NO
1147  use USE-RDC|NO
1148
1149  use COMMON
```

Introduction to config-specs.ini

- Configure flags

```
1139  [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140  use RHEL7_SEMS_COMPILER|GNU
1141  use NODE-TYPE|SERIAL
1142  use BUILD-TYPE|DEBUG
1143  use RHEL7_SEMS_LIB-TYPE|SHARED
1144  use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145  use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
1146  use USE-COMPLEX|NO
1147  use USE-RDC|NO
1148
1149  use COMMON
```

Introduction to config-specs.ini

- Common code for ‘debug’ build type:

```
409  [BUILD-TYPE|DEBUG]
410  opt-set-cmake-var CMAKE_BUILD_TYPE                STRING : DEBUG
411  opt-set-cmake-var Trilinos_ENABLE_DEBUG           BOOL   : ON
412  # Intrepid2 will not build with bounds check enabled:
413  # coreKokkos_Array.hpp:152:57: error: array subscript is below array bounds [-Werror=array-bounds]
414  opt-set-cmake-var Kokkos_ENABLE_DEBUG_BOUNDS_CHECK BOOL   : OFF
415  opt-set-cmake-var Kokkos_ENABLE_DEBUG             BOOL   : ON
```

- Build type flag

```
1139  [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140  use RHEL7_SEMS_COMPILER|GNU
1141  use NODE-TYPE|SERIAL
1142  use BUILD-TYPE|DEBUG
1143  use RHEL7_SEMS_LIB-TYPE|SHARED
1144  use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145  use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
1146  use USE-COMPLEX|NO
1147  use USE-RDC|NO
1148
1149  use COMMON
```

Changing config-specs.ini to use a new build type

Draft: Tug16 config spec demo

Overview 0 | Commits 1 | Pipelines 1 | **Changes 1**

Compare master and latest version — 1 file +2 -2

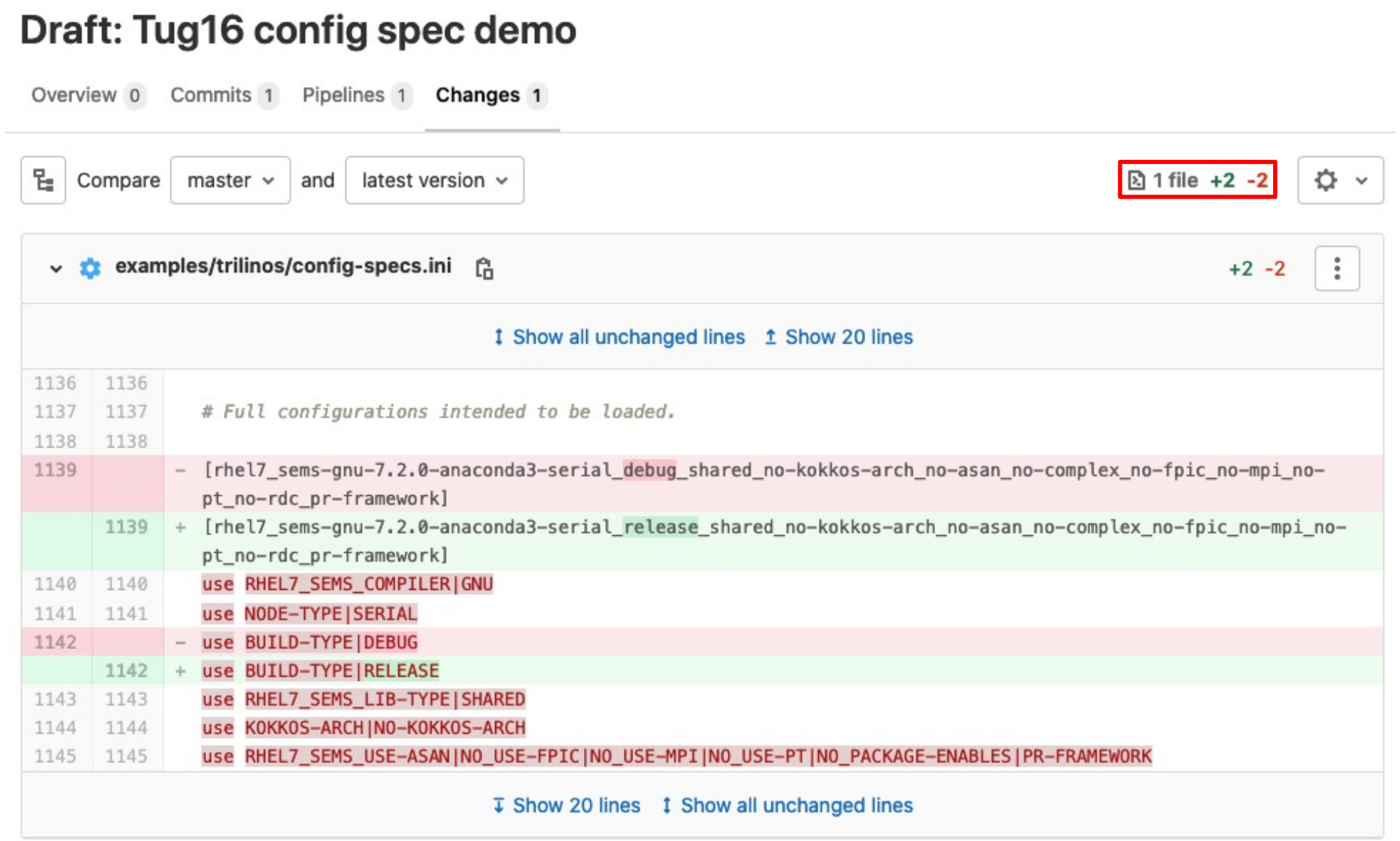
examples/trilinos/config-specs.ini +2 -2

↕ Show all unchanged lines ↥ Show 20 lines

```
1136 1136
1137 1137   # Full configurations intended to be loaded.
1138 1138
1139      - [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
     1139 + [rhel7_sems-gnu-7.2.0-anaconda3-serial_release_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140 1140   use RHEL7_SEMS_COMPILER|GNU
1141 1141   use NODE-TYPE|SERIAL
1142      - use BUILD-TYPE|DEBUG
     1142 + use BUILD-TYPE|RELEASE
1143 1143   use RHEL7_SEMS_LIB-TYPE|SHARED
1144 1144   use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145 1145   use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
```

↧ Show 20 lines ↕ Show all unchanged lines

Changing config-specs.ini to use a new build type



Draft: Tug16 config spec demo

Overview 0 | Commits 1 | Pipelines 1 | **Changes 1**

Compare master and latest version — 1 file +2 -2

examples/trilinos/config-specs.ini +2 -2

↕ Show all unchanged lines ↥ Show 20 lines

```
1136 1136
1137 1137   # Full configurations intended to be loaded.
1138 1138
1139      - [rhel7_sems-gnu-7.2.0-anaconda3-serial_debug_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
     1139 + [rhel7_sems-gnu-7.2.0-anaconda3-serial_release_shared_no-kokkos-arch_no-asan_no-complex_no-fpic_no-mpi_no-pt_no-rdc_pr-framework]
1140 1140   use RHEL7_SEMS_COMPILER|GNU
1141 1141   use NODE-TYPE|SERIAL
1142      - use BUILD-TYPE|DEBUG
     1142 + use BUILD-TYPE|RELEASE
1143 1143   use RHEL7_SEMS_LIB-TYPE|SHARED
1144 1144   use KOKKOS-ARCH|NO-KOKKOS-ARCH
1145 1145   use RHEL7_SEMS_USE-ASAN|NO_USE-FPIC|NO_USE-MPI|NO_USE-PT|NO_PACKAGE-ENABLES|PR-FRAMEWORK
```

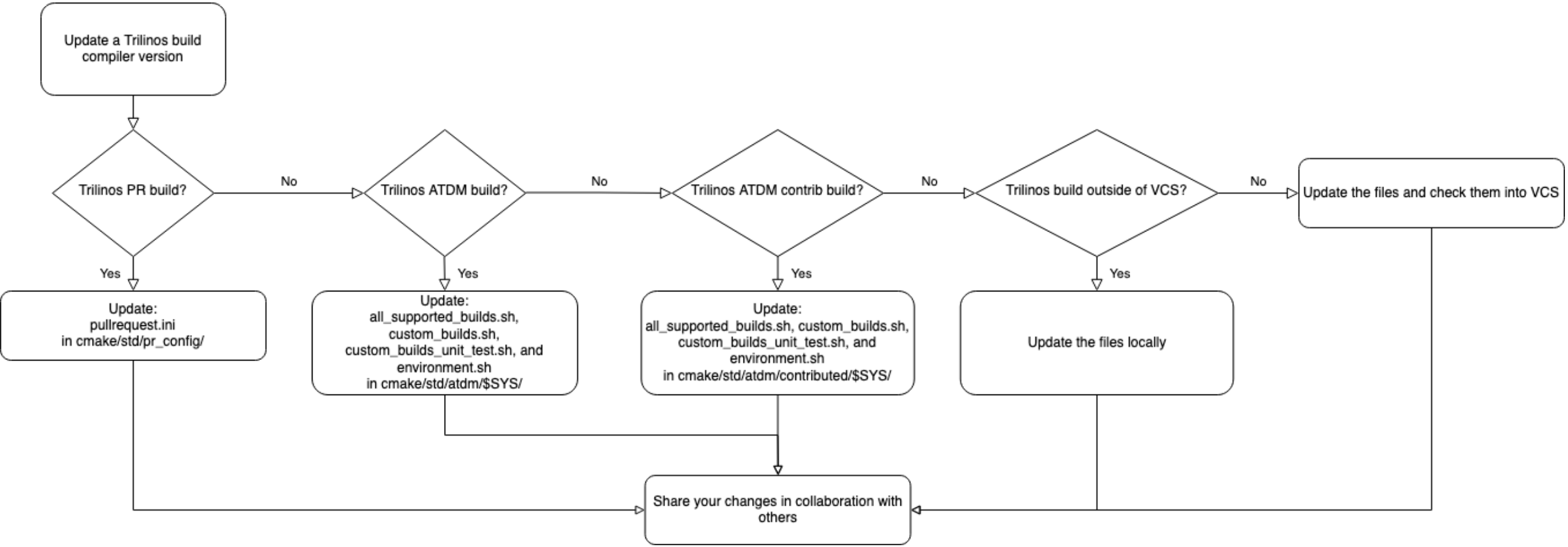
↧ Show 20 lines ↕ Show all unchanged lines

Changing config-specs.ini causes test failures

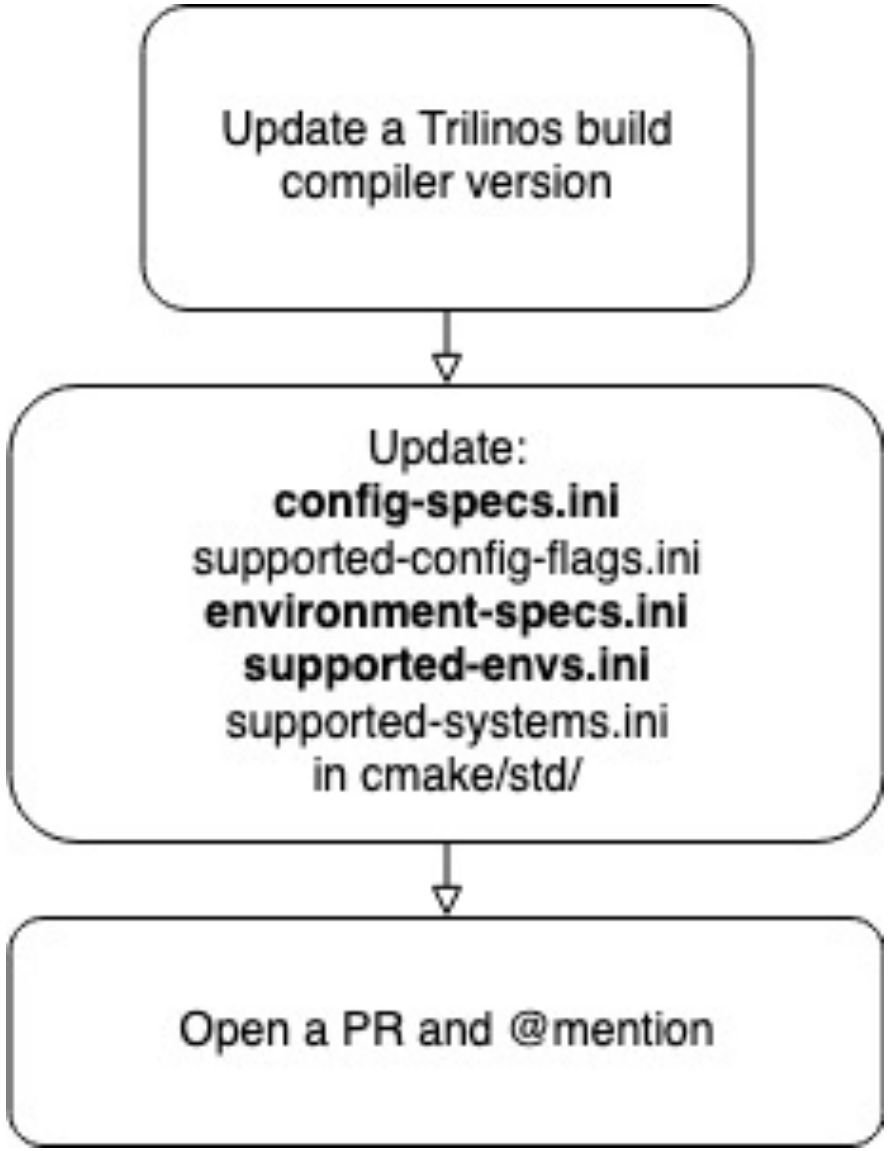
Since config-specs.ini is verified via pytest, the change causes this test failure:

```
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_all_configs_are_in_map FAILED [ 87%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_clang_10_0_0_openmpi_1_10_1_serial_release_debug_shared_no_kokkos_arch_no_asan_no_complex_no_fpic_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 88%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_gnu_7_2_0_anaconda3_serial_debug_shared_no_kokkos_arch_no_asan_no_complex_no_fpic_no_mpi_no_pt_no_rdc_pr_framework SKIPPED [ 90%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_gnu_7_2_0_openmpi_1_10_1_serial_release_debug_shared_no_kokkos_arch_no_asan_no_complex_no_fpic_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 91%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_gnu_7_2_0_serial_release_debug_shared_no_kokkos_arch_no_asan_no_complex_no_fpic_no_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 92%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_gnu_8_3_0_openmpi_1_10_1_openmp_release_debug_static_no_kokkos_arch_no_asan_no_complex_no_fpic_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 93%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_intel_17_0_1_mpich_3_2_serial_release_debug_static_no_kokkos_arch_no_asan_no_complex_fpic_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 95%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_rhel7_configs::test_rhel7_sems_intel_19_0_5_mpich_3_2_serial_release_debug_static_no_kokkos_arch_no_asan_no_complex_fpic_mpi_no_pt_no_rdc_no_package_enables SKIPPED [ 96%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_ats2_configs::test_all_configs_are_in_map PASSED [ 97%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_ats2_configs::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_no_package_enables SKIPPED [ 98%]
tests/verification_tests/test_verify_trilinos_configs.py::Test_verify_ats2_configs::test_ats2_cuda_10_1_243_gnu_8_3_1_spmpi_rolling_release_static_Volta70_Power9_no_asan_no_complex_no_fpic_mpi_pt_rdc_no_package_enables SKIPPED [100%]
================================== FAILURES ===================================
_____________ Test_verify_rhel7_configs.test_all_configs_are_in_map _____________
```

Old use-case


Update a Trilinos build compiler version
Trilinos PR build?
No
Trilinos ATDM build?
No
Trilinos ATDM contrib build?
No
Trilinos build outside of VCS?
No
Update the files and check them into VCS
Yes
Update: pullrequest.ini in cmake/std/pr_config/
Yes
Update: all_supported_builds.sh, custom_builds.sh, custom_builds_unit_test.sh, and environment.sh in cmake/std/atdm/$SYS/
Yes
Update: all_supported_builds.sh, custom_builds.sh, custom_builds_unit_test.sh, and environment.sh in cmake/std/atdm/contributed/$SYS/
Yes
Update the files locally
Share your changes in collaboration with others

New use-case: ini files as a common language



How GenConfig supports collaboration via ini files

- Parses single file: config-specs.ini
- Documentation resides in config-specs.ini
- Supports developer use-case:
  - Dev gets something working locally
  - Copy and paste into ini files under VCS
  - GenConfig takes care of translation
    - No need to translate from a cmake command line or cmake fragment file
- Rather than a potentially ambiguous email, config-specs.ini provides a testable common language

Special thanks to

- Dena Vigil
- Roscoe Bartlett
- Jason Gates
- David Collins
- Trilinos Framework Team