



# Geant4 Develop, Build and Test Tools Update

Ben Morgan







Open 14 Merged 1,951 Closed 190 All 2,155

 [Edit merge requests](#) [New merge request](#)

Recent searches ▾ Search or filter results...

Last updated

global-V10-07-24, mctruth-V10-07-01, modeling-V10-07-14, geommmng-V10-07-06: Initial clang-tidy of global category



 1 left
  31  
 updated 2 days ago

xrays-V10-06-10: fix too small step in Cerenkov

!2164 · created 4 days ago by Daren Sawkey · Y geant4-10-07\_patches · [Release Patch](#) [bug](#)







### Draft: Add tracking manager for EmStandardPhysics to TestEm3

1 left 3  
updated 1 week ago

### Draft: Add interface to register custom tracking managers


!2105 · created 3 weeks ago by Jonas Hahnfeld


## Draft: Test implementation of CRTP based allocatable types





 1 left
  22  
 updated 2 weeks ago

!1995 · created 2 months ago by Benjamin Morgan

geommg-V10-06-07, geant4py-V10-06-05: Support Geant4py bindings when using VecGeom



 1 left
  2  
 updated 2 weeks ago

!2133 · created 2 weeks ago by Benjamin Morgan Y geant4-10-07\_patches [Release Patch](#)

WIP: cmake-V10-07-29, vis\_toolssg-V10-07-06: Enable ToolsSG options with proper mutual exclusivity




 1 left
  8

# GitLab Status

Passed 2000 MRs, stable service from CERN

```
geant4-dev.git - zsh -- 100x42
[macbook]$ git log -p -n1
commit cafeece0588a72d643a018281ef75d3b649b0af8 (HEAD -> clang-tidy-tooling-global)
Author: Ben Morgan <Ben.Morgan@warwick.ac.uk>
Date: Mon Sep 13 16:25:11 2021 +0100

    Revert G4MTBarrier destructor to empty body

    Error from Windows CI regarding:

        G4MTRunManager.obj : error LNK2019: unresolved external symbol "const G4MTBarrier::vftable" (?
        ?_7G4MTBarrier@@6B@)

    Try reverting clang-tidy change to virtual destructor as main
    point of change that might have caused this.

diff --git a/source/global/management/include/G4MTBarrier.hh b/source/global/management/include/G4MT
Barrier.hh
index fe1377b79e..ab65b38731 100644
--- a/source/global/management/include/G4MTBarrier.hh
+++ b/source/global/management/include/G4MTBarrier.hh
@@ -125,7 +125,8 @@ class G4MTBarrier
 G4MTBarrier()
     : G4MTBarrier(1)
 {}
- virtual ~G4MTBarrier() = default;
+ // NOLINTNEXTLINE : may be needed by Visual Studio?
+ virtual ~G4MTBarrier() {}
 G4MTBarrier(const G4MTBarrier&) = delete;
 G4MTBarrier& operator=(const G4MTBarrier&) = delete;

[macbook]$
```

# Writing good Commit Messages and MR Descriptions

geant4 > geant4-dev > Merge requests > !2155

Merged Created 1 week ago by Daren Sawkey Options

## materials-V10-07-18: change material (const) properties map to a vector, etc.

All threads resolved

Overview 46 Commits 42 Changes 3

Update the material properties table to store the list of properties as vectors, rather than maps. This makes accessing the values a simple pointer dereference, rather than a find operation on the maps. The map keys were integers anyway. In tests with OpNovice2/electron.mac see about a 2% reduction in instructions.

If a material property is not defined, the vector of material property vectors contains a nullptr.

In contrast, material constant properties are G4double. In order to keep track of whether these are defined, a boolean is added. The boolean is true if the material constant property is defined. The vector of material constant properties is really a vector to std::pair<G4double, G4bool>.

Also some cleaning: variable names start with a lower case letter, and class members start with f (not G4!). Use C++11 style iteration and remove the private member iterators.

Edited 4 days ago by Daren Sawkey

No mile...

None

0

12

Really important for MR process and later maintenance

```
cmake — vim History — 100x42

=====
Geant4 - an Object-Oriented Toolkit for Simulation in C++
=====

Category History file

History file for Geant4 CMake build scripts.
It DOES NOT substitute the SVN log-message one should put at every
committal in the SVN repository !

Entries MUST use the following format:

MONTH DAY YEAR - INITIAL.SURNAME (TAG)
- FILECHANGED : Document changes made.

- OTHERFILECHANGED : Document changes made.

You should keep two lines between each date.
TAG is optional. If your changes fix a bug listed in the Geant4 Bugzilla,
add this in the documentation for the changed file as [BUGFIX #BUGNUMBER].

=====
* Reverse chronological order (last date on top), please *
=====

3 September 2021 - Ben Morgan (cmake-V10-04-36)
- geant4_add_test creates separate build/run tests when asked to do both
- Required to support valgrind/memcheck runs without valgrind-ing
the build part of the test.

1 September 2021 - Ben Morgan (cmake-V10-04-35)
- Remove obsolete geant4_{executable,link_library} options

26 August 2021 - Ben Morgan (cmake-V10-07-34)
- Remove obsolete network VRML driver option
"History" 3159L, 148035C
```



```
cmake — vim History — 100x42

# cmake Category History File

See CODING_GUIDELINES.rst for details on entry format/required info

## 2021-12-21 Ben Morgan (cmake-V11-00-01)
- Change to new History format
- Overall document is [Markdown](https://en.wikipedia.org/wiki/Markdown)
- Entry is a [Markdown level 2 heading](https://www.markdownguide.org/basic-syntax/#headings)
- Y-M-D for date
- Then name of comitter
- Then (tag)
- Entry is always formatted as a [Markdown unordered list](https://www.markdownguide.org/basic-syntax/#unordered-lists)

# History entries prior to 11.0

3 September 2021 - Ben Morgan (cmake-V10-04-36)
- geant4_add_test creates separate build/run tests when asked to do both
- Required to support valgrind/memcheck runs without valgrind-ing
the build part of the test.

1 September 2021 - Ben Morgan (cmake-V10-04-35)
- Remove obsolete geant4_{executable,link_library} options

26 August 2021 - Ben Morgan (cmake-V10-07-34)
- Remove obsolete network VRML driver option
- Remove obsolete network FukuiRenderer driver option

23 August 2021 - Guilherme Amadio (cmake-V10-07-33)
- Rename Geant4::HDF5 to hdf5:hdf5 to be compatible with
imported target provided since CMake 3.20.

20 August 2021 - Gunter Folger (cmake-V10-07-32)
- Switch to G4ParticleXS 4.0

22 July 2021 - Ivana Hrivnacova (cmake-V10-07-31)
- Fixed path to g4tools/fonts, now in source/externals


30 June 2021 - Ben Morgan (cmake-V10-07-30)
```


# New History file format for post-11.0 Development

Why? Consistency across categories,  
potential for automation/rendering.

[Discussion open on Issue #17](#)




 GitLab


 Menu

45

7


99+







G4


geant4-dev

 Project information


 Repository


 Issues


36


 Merge requests


14


 CI/CD


 Security & Compliance


 Deployments

 Monitor

 Infrastructure

 Packages & Registries

 Analytics

 Settings


<< Collapse sidebar


Overview 46

Commits 42

Changes 3

source/materials/src/G4MaterialPropertiesTable.cc



 Benjamin Morgan @bmorgan · 1 week ago

Suggested change ?

98- for(size\_t i = 0; i < kNumberOfPropertyIndex; ++i)


99- {

100- fMP.push\_back(nullptr);


101- }

98+ fMP.assign(kNumberOfPropertyIndex, nullptr);


I think this would work as fMP should not contain anything at this point?

 Vladimir Ivantchenko @vnivanch · 1 week ago

@bmorgan , Why not use a simple fMP.resize(kNumberOfPropertyIndex, nullptr) ?


 Benjamin Morgan @bmorgan · 1 week ago


Pure clarity in this case - it's construction, so we're not really resizing. Plus, `resize` won't change any existing elements in the range (0, newsize), so this `assign` is "safer" in this case.


 Daren Sawkey @dsawkey · 1 week ago


Nice, thanks.

Owner












Apply suggestion ▾


Developer










Owner









Developer










All threads resolved ▾


<<









No mile...




None



0



12



Use the Review Process!

Fast and effective way to work together to clarify changes, and improve Geant4 during development<sub>5</sub>

geant4 > geant4-dev > Merge requests > !2168

Open Created 19 minutes ago by Vladimir Grichine Reporter

Edit Mark as draft

## xrays-V10-07-07

Overview 1 Commits 1 Changes 6

G4GaussXTRadiator and G4VXTRenergyLoss - Gauss-distributed regular radiator and set/get methods for angle distribution range

Request to merge grichine:xrays-V10-07-07 into master

Open in Web IDE

Check out branch

Download

Approve Requires approval.

View eligible approvers

! Merge You can only merge once this merge request is approved.

The source branch will be deleted

- Only Category Coordinators/deputies can Approve at present.
- Reviewing how to allow all "Code Owners" as well (GitLab Permissions)

# MR Approval

Always approve MRs you are assigned to review, even if just by a comment in thread



Use latest CLHEP 2.4.5.1, and latst G4 view

Gunter Folger authored 1 week ago

f4b50ae1

Code owners ⓘ: [Gunter Folger](#) and [Soon Yung Jun](#)

Jenkinsfile-CI 4.9 KB

Edit

Web IDE

Lock

Replace

Delete



```
1 //
2 // This pipeline is running the CI triggered by gitlab.
3 // A Job per platform is started is started.
4 //
5 // Gunter Folger, June 2021
6 //
7
8
9 def g4
10
11 pipeline {
12     parameters {
13         string(name: 'EXTERNALS',           defaultValue: 'LCG_geant4ext20210906',
14                                     description: 'CVMFS directory keeping the external libraries')
15         string(name: 'BuildOptions',        defaultValue: 'none',
16                                     description: 'extra build options, like: staticlibs, UseVecGeom, BoundsCheck, EpCheck, UseInternalCLHEP, UseGra
17     }
18     options {
19         buildDiscarder(logRotator( daysToKeepStr: '5' ))
20         timestamps()
```

# Jenkins Pipelines

Better reporting to MRs, simpler,  
declarative scripts in geant4-dev

CONTRIBUTING.rst 51.8 KB

## Contributing to Geant4

The authoritative Geant4 Git repository is hosted on the [CERN GitLab](https://gitlab.cern.ch/geant4/geant4-dev) instance at <https://gitlab.cern.ch/geant4/geant4-dev>, accessible to members of the [Geant4 Collaboration](#). Contributions to Geant4 from outside the Collaboration are welcome, subject to successful testing and validation, via pull requests submitted through our GitHub mirror of published releases hosted at <https://github.com/Geant4/geant4>.

If you want to try out the Workflow documented below, the <https://gitlab.cern.ch/geant4/geant4-dev> repository provides a "sandbox" which can be used for experimentation.

If you have any issues with or questions on the workflow, please start an issue on the Issues Board at <https://gitlab.cern.ch/geant4/geant4-dev/issues>.

## Basic Tools

### Git

[Git](#) is the version control system used by Geant4. If it is not installed on your system already, it is usually available through package managers for your system (apt, yum, Homebrew, vcpkg) or comes bundled with IDEs such as XCode or Visual Studio. Failing this, the Git homepage provides a range of [binary bundles for Linux, MacOS, and Windows](#).

If you are new to git or need additional information, a wide range of documentation and tutorials are available, including:

- The [Atlassian Tutorials](#) are a good starting point, and also cover more advanced topics.
- The official [Pro Git book](#) offers greater depth and breadth. Chapters 2 and 3 covering the basics.

# Documentation on Merge Request Process

[A living document](#), so Issues/MRs to improve it are very welcome



Open Created 1 week ago by Ivana Hrivnacova Developer

Close issue

## Clang-tidy configuration

The start configuration in `geant4-dev/.clang-tidy` can produce a huge number of warnings. The purpose of this issue is to get agreed on a start configuration, reduced wrt what we have now.

In the suggested list of options below, the "readability" checks are commented-out and more "modernize" and "performance" checks are added, but only via commented lines.

The users can then uncomment one by one or use this list for running them via `run-clang-tidy.py` with the check selection parameter.

```
Checks: "-*, \  
#readability-braces-around-statements, \  
#readability-container-size-empty, \  
#readability-delete-null-pointer, \  
#readability-implicit-bool-conversion, \  
#readability-isolate-declaration, \  
#readability-redundant-member-init, \  
#readability-simplify-boolean-expr, \  
#readability-string-compare, \  
#readability-uniqueptr-delete-release, \  
#modernize-avoid-c-arrays, \  
modernize-make-shared, \  
modernize-make-unique. \  

```

# Issues: for Problems *and* Discussion *and* Work Items

Prefer Issues to email. Remember that you can link MRs to Issues to close them!

GitLab Menu

geant4-dev

- Project information
- Repository
- Issues 36
- List
- Boards
- Service Desk
- Milestones**
- Iterations
- Merge requests 14
- CI/CD
- Security & Compliance
- Deployments
- Monitor
- Infrastructure
- Packages & Registries
- Collapse sidebar

geant4 > geant4-dev > Milestones > Release 11.0

Open Milestone Edit Promote Close milestone Delete

## Release 11.0

Major release of Geant4 11.0

Issues 8 Merge requests 4 Participants 12 Labels 4

Work in progress (open and unassigned) 0	Waiting for merge (open and assigned) 1	Rejected (closed) 0	Merged 3
	global-V10-07-24, mctruth-V10-07-01, modeling-V10-07-14, geommmg-V10-07-06: Initial clang-tidy of global category !2161 enhancement		General: Minimal docs/config for Clang-tidy, Sanitizers and Coverity !2140 Staged vis-HepRep-V10-07-04, visman-V10-07-19, geant4py-V10-07-03 : Remove obsolete HepRep(WIRED) driver !2130 DAWN-V10-07-02. VRML-V10-

# Milestones/Projects

Workplan items -> Issues -> MRs -> Release! Make better use of these features in 2022?

SHIFTERS.rst 26.5 KB

Shift Tutorial planned for early 2022

## Managing Development/Testing Shifts

Code is integrated into Geant4 via developers submitting their changes for testing through GitLab [Merge Requests](#). Geant4 Working Group Coordinators are responsible for

1. Undertaking weekly shifts to manage and monitor the progress of these Merge Requests from submission to eventual integration (the *Shifter* role)
2. Reviewing in-progress Merge Requests making changes to their Category(ies) (the *Assignee*, or *Reviewer*, role)

This document provides a guide for working these shifts and review processes, which cover the ongoing mainline (the `master` Git branch) of Geant4 development. Shifts during testing and validation of new Releases and Patches are undertaken by the Release Manager, and are described in the [Release Management Guide](#). In both cases, it is assumed that you are familiar with the general git workflow for Geant4 described in the main [Contribution Guide](#).

If you have any questions about, or during, shifts and review, please [start an issue](#). For any problems encountered in specific Merge Requests during a shift, add a comment to the Merge Request comment thread and "mention" [@bmorgan](#) to notify them and get help, e.g. "[@bmorgan](#), there's a problem with *<describe it here>*, could you take look please?". Note that throughout Shifts, you should keep all notes and discussions on the Merge Request comment threads as these act as the Shift Log that other Shifters and developers will need to consult.

## Registering for Shifts

A calendar is available for shift registration. Please contact [@gunter](#) for sign up information. Shifts are taken in 7 day blocks, with each block starting Monday 00:00 (Geneva) each week.

# Get Involved!

[MR monitoring through Shifts:](#)  
volunteers welcome (and needed)!

# Build Status 1: Default to Multithreaded Build

- MT (very!) mature, and Tasking/G4RunManagerFactory make it easy to “build MT but run serial”
  - *Serial runs with MT build are a few % slower than pure Serial build*
  - *Exact difference measured in monthly profiling: <https://q4cpt.fnal.gov>*
  - *Will document this difference in Install/Application Developer Guide as appropriate for 11.0 Release*
- **For 11.0, the GEANT4\_BUILD\_MULTITHREADED CMake option will change to being ON by default**
- Change to CMake configuration in geant4-dev in MR **next week**
  - *No change to any existing build you have...*
  - *... but be aware that after change, new from-scratch builds will have MT enabled*

# Build Status 2:

## Modularization

---

- Migration to new CMake commands completed across all categories
  - *geant4\_add\_module,*  
*geant4\_module\_link\_libraries ...*
  - *Deliberately close to CMake*  
*add\_library/target\_link\_libraries*
- No longer need include\_directories, simply declare which modules yours needs to “link” to in `geant4_module_link_libraries`
- Documentation to be added in `CODING_GUIDELINES.rst`
- **Full discussion: Thursday Kernel Plenary**

# - `sources.cmake`

```
geant4_add_module(G4foo
PUBLIC_HEADERS
    G4Foo.hh
SOURCES
    G4Foo.cc
)
```

```
geant4_module_link_libraries(G4foo
PUBLIC G4globman
PRIVATE G4intercoms
)
```




# Build Status 3: GNUMake Retirement

- Internal use will finish in next few weeks as final changes are integrated to support Valgrind/Memcheck testing with CMake.
- **Removal from 11.0 for use by users in preparation:**
  - [Pkg-config](#) support files for non-CMake user code builds
  - *Migration of single example to demonstrate use*
  - *Documentation*
- **All these steps must be finalized before GNUMake removed in 11.0**
  - *In case of retention, will be marked as hard deprecation with **no** support.*



geant4 > geant4-dev > Merge requests > !1292

Merged
 Created 10 months ago by  **Jonathan Madsen**
Reporter

Edit

## global-V10-06-23 et al.: G4Backtrace Implementation

**Overview** 13
 Commits 2
 Pipelines 3
 Changes 5

- Added G4Backtrace.hh for printing backtraces from raised signals
- This is a more generic implementation of G4FPEDetection and can catch FPE issues but, by default, does not interfere with it.
- The default backtraced signals are: SIGQUIT, SIGILL, SIGABRT, SIGKILL, SIGBUS, SIGSEGV
- The default signals can be overridden within user-code before and after the creation of the run-manager (See "Usage" section in G4Backtrace.hh)
- The default signals can be overridden via environment variable G4BACKTRACE

[@gcosmo](#) [@gunter](#) I believe we discussed adding this capability previously. I am having quite a bit of trouble re-producing the tasking errors so this should help locate current and future segmentation faults in testing. E.g. if I intentionally create a seg-fault in G4TaskWorkerRunManager::DoWork(int), I get (with `-O2 -g`):

```

=====
--> G4TaskRunManager::CreateAndStartWorkers() --> Creating 12 tasks with 8 events/task...
=====
    
```

Backtrace:

```

[PID=7014, TID=6] [ 0/10]> 0  libsystem_platform.dylib      0x00007fff69fb55fd _sigtramp + 29
[PID=7014, TID=6] [ 1/10]> 1  ???                0x0000000000000000 0x0 + 0
[PID=7014, TID=6] [ 2/10]> 2  libG4tasking.dylib  0x00000000110659797 G4WorkerTaskRunManager::DoWork(int) + 295
[PID=7014, TID=6] [ 3/10]> 3  libG4tasking.dylib  0x00000000110653568 std::__1::packaged_task<void ()>::operator()() + 72
    
```

# Developer Tools 1: G4Backtrace

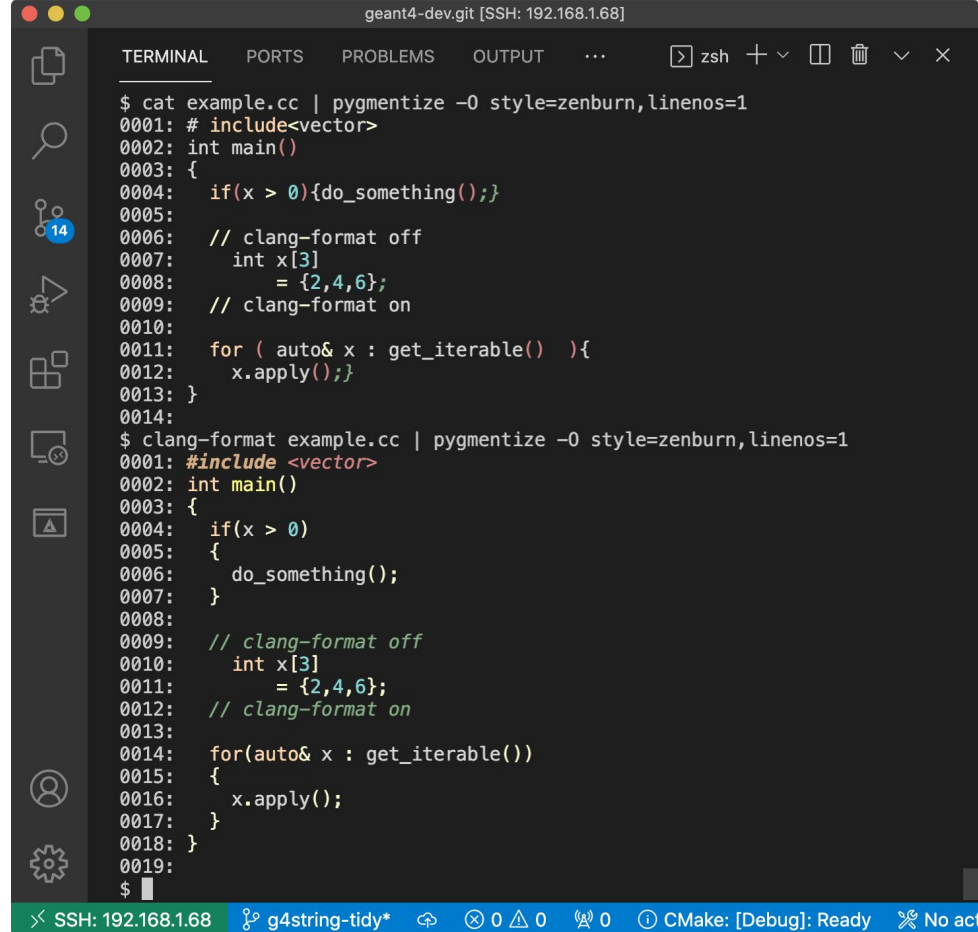
Default/Optional printing of stack trace to help in debugging



# Developer Tools 3:

## clang-format

- Tool for formatting C++ code, style defined in [.clang-format file in project root](#)
- **Why? Consistency across the project**
  - *Settings as close as possible to de-facto Geant4 style*
  - *Focus on what code **does***
- Easy to **disable**, e.g. for array/matrix data, with special comment blocks
- Can be integrated in development/MR process if required
- [Issue 97 on GitLab open to discuss](#) how to apply in development and docs/tooling to support use.



The screenshot shows a terminal window with a dark theme. The title bar at the top reads "geant4-dev.git [SSH: 192.168.1.68]". The terminal has tabs for "TERMINAL", "PORTS", "PROBLEMS", and "OUTPUT". The current tab is "TERMINAL". The prompt is "\$". The first command entered is "cat example.cc | pygmentize -O style=zenburn,linenos=1". The output shows a C++ code snippet with line numbers 0001 to 0014. The code includes a header, a main function, and some logic. The second command entered is "clang-format example.cc | pygmentize -O style=zenburn,linenos=1". The output shows the same code snippet, but with clang-format applied, resulting in a different indentation and formatting style. The status bar at the bottom shows "SSH: 192.168.1.68", "g4string-tidy\*", and other icons.

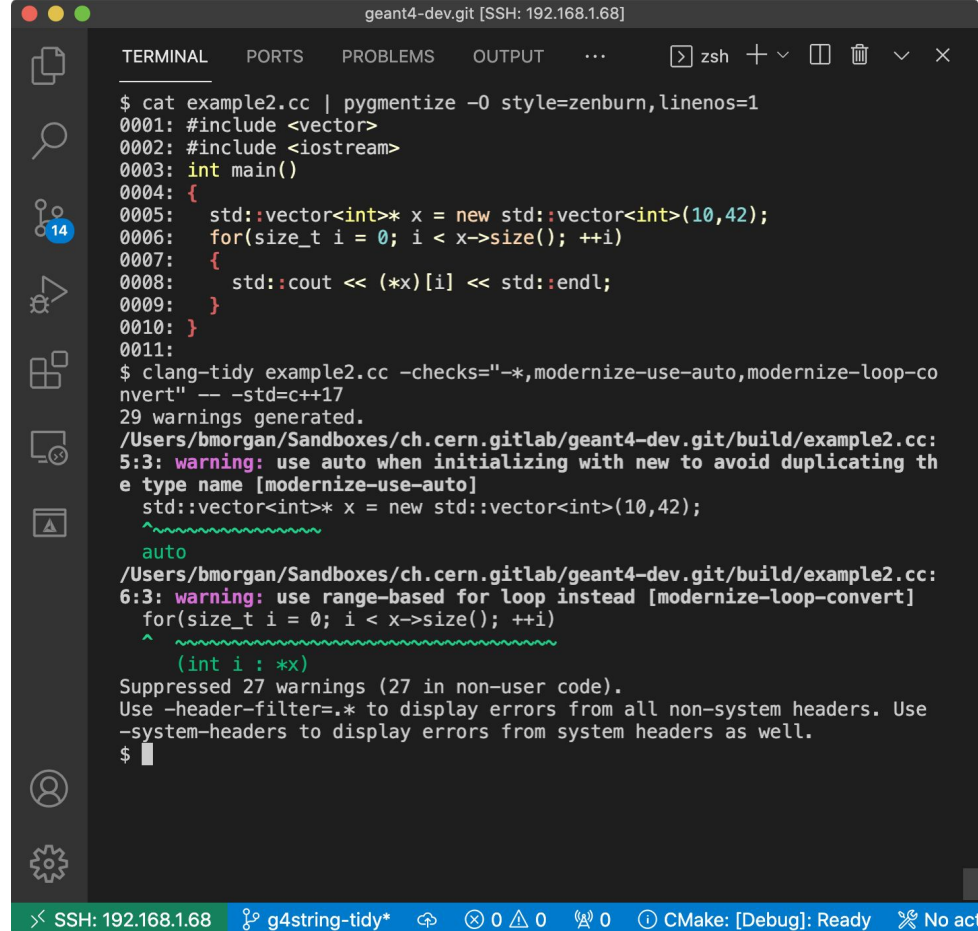
```
$ cat example.cc | pygmentize -O style=zenburn,linenos=1
0001: # include<vector>
0002: int main()
0003: {
0004:     if(x > 0){do_something();}
0005:
0006:     // clang-format off
0007:     int x[3]
0008:         = {2,4,6};
0009:     // clang-format on
0010:
0011:     for ( auto& x : get_iterable() ){
0012:         x.apply();}
0013: }
0014:

$ clang-format example.cc | pygmentize -O style=zenburn,linenos=1
0001: #include <vector>
0002: int main()
0003: {
0004:     if(x > 0)
0005:     {
0006:         do_something();
0007:     }
0008:
0009:     // clang-format off
0010:     int x[3]
0011:         = {2,4,6};
0012:     // clang-format on
0013:
0014:     for(auto& x : get_iterable())
0015:     {
0016:         x.apply();
0017:     }
0018: }
0019:
$
```

# Developer Tools 4:

## clang-tidy

- Tool for linting C++ code with a set of checks for “better practice” such as [clarity, modernity, performance](#).
  - Checks in `.clang-tidy` file or supplied on command line
- Why? Assist developer to identify code that could be improved
  - *Consistency as well - checks can be matched to, or complement, coding guidelines*
- Effectively acts as a higher layer of compiler warnings, also suggesting fixes...



The screenshot shows a terminal window with the following content:

```
geant4-dev.git [SSH: 192.168.1.68]
TERMINAL  PORTS  PROBLEMS  OUTPUT  ...  [zsh]
$ cat example2.cc | pygmentize -O style=zenburn,linenos=1
0001: #include <vector>
0002: #include <iostream>
0003: int main()
0004: {
0005:     std::vector<int>* x = new std::vector<int>(10,42);
0006:     for(size_t i = 0; i < x->size(); ++i)
0007:     {
0008:         std::cout << (*x)[i] << std::endl;
0009:     }
0010: }
0011:
$ clang-tidy example2.cc -checks="*,modernize-use-auto,modernize-loop-co
nvert" -- -std=c++17
29 warnings generated.
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
5:3: warning: use auto when initializing with new to avoid duplicating th
e type name [modernize-use-auto]
    std::vector<int>* x = new std::vector<int>(10,42);
    ^~~~~~
    auto
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
6:3: warning: use range-based for loop instead [modernize-loop-convert]
    for(size_t i = 0; i < x->size(); ++i)
    ^~~~~~
    for(int i : *x)
Suppressed 27 warnings (27 in non-user code).
Use -header-filter=. to display errors from all non-system headers. Use
-system-headers to display errors from system headers as well.
$
```

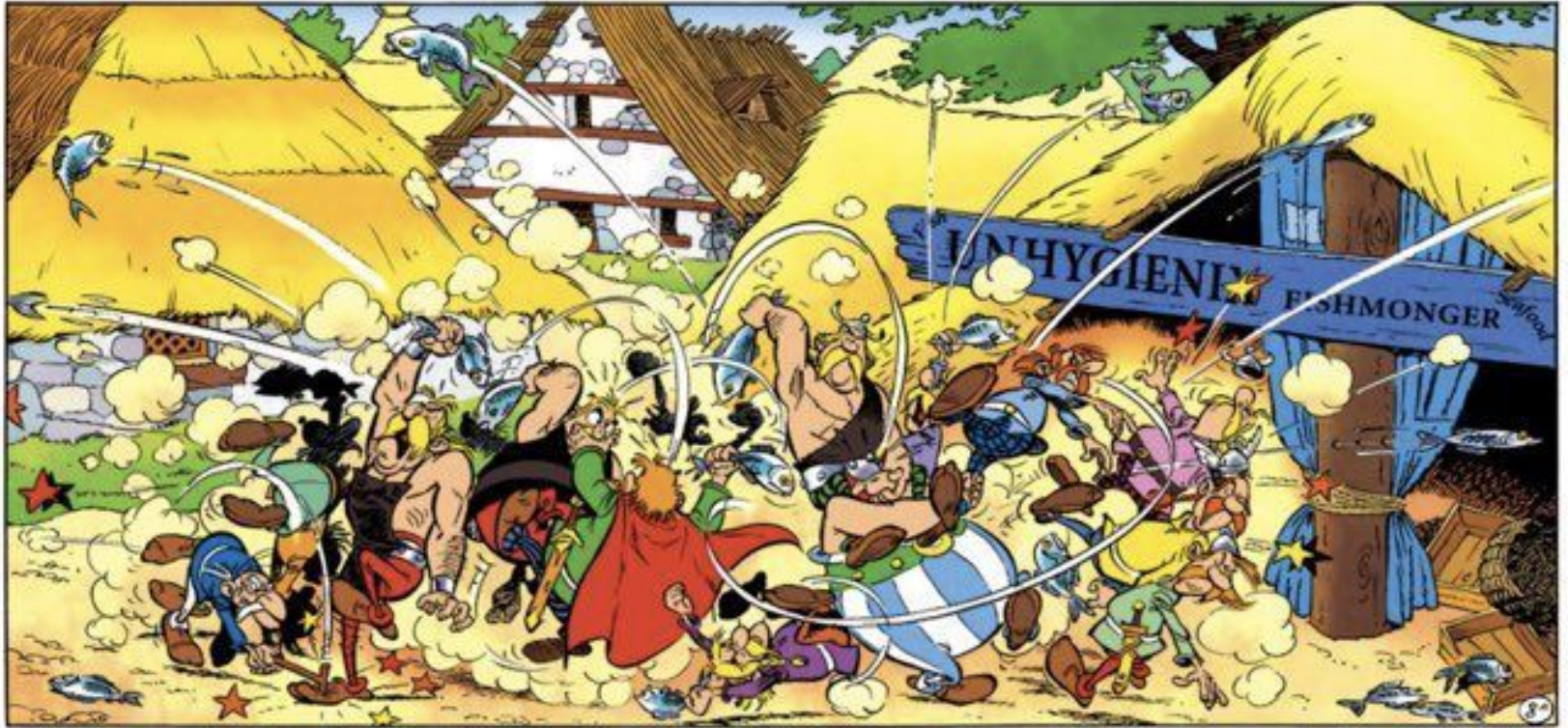


## Developer Tools 4: clang-tidy

- ... fixes can optionally be applied automatically.
  - *Not 100% reliable for more complex fixes, so some level of developer interaction needed*
- Checks can be disabled with `//NOLINT` comments on specific lines if needed
- Issue 94 opened to discuss:
  - *Initial set of checks to apply/fix (easy to add additional ones later)*
  - *What checks are better as new Coding Guidelines*
  - *Docs/Tooling to help developers use clang-tidy in their work*

```
geant4-dev.git [SSH: 192.168.1.68]

$ clang-tidy --quiet -fix example2.cc -checks="-*,modernize-use-auto,modernize-loop-convert" -- -std=c++17
29 warnings generated.
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
5:3: warning: use auto when initializing with new to avoid duplicating the type name [modernize-use-auto]
      std::vector<int>* x = new std::vector<int>(10,42);
      ~~~~~
      auto
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
5:3: note: FIX-IT applied suggested code changes
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
6:3: warning: use range-based for loop instead [modernize-loop-convert]
      for(size_t i = 0; i < x->size(); ++i)
      ^~~~~~
      for(int i : *x)
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
6:6: note: FIX-IT applied suggested code changes
      for(size_t i = 0; i < x->size(); ++i)
      ^
/Users/bmorgan/Sandboxes/ch.cern.gitlab/geant4-dev.git/build/example2.cc:
8:18: note: FIX-IT applied suggested code changes
      std::cout << (*x)[i] << std::endl;
      ^
clang-tidy applied 3 of 3 suggested fixes.
$ cat example2.cc | pygmentize -O style=zenburn,linenos=1
0001: #include <vector>
0002: #include <iostream>
0003: int main()
0004: {
0005:     auto* x = new std::vector<int>(10, 42);
0006:     for(int i : *x)
0007:     {
0008:         std::cout << i << std::endl;
0009:     }
0010: }
```



Open for Discussion on  
GitLab in [Issue 97](#), [Issue 94](#)!

... but please avoid holy wars or [bikeshedding](#)! Assisting developers, new and old, through consistency of style and practice is the **only** goal

# Questions, Discussion