

# CPPcheck And Coverity News

SHARMAZANASHVILI Alexander  
**Georgian Technical University**

TODUA Luka  
**Georgian Technical University**

PIRTSKHALAVA Mariam  
**Georgian Technical University**



# I. CPPcheck Scan

We have 2 scanning processes for CPPcheck:

1. Full scan of Athena repository

- Outputs: Jira Tickets for individual authors

  - .html* table with all defects for current scan

  - .html* table with all defects for all scans

  - .html* table with fixed defects

- Scan intensity: Weekly

2. MR individual scan

- Outputs: Filled templet /or xml

- Scan intensity: Follow up MR's

- Full scan process of Athena repository consists of 8 consecutive steps:
  1. Cloning Athena from gitlab by git pull
  2. Local CPPcheck scan of Athena and results generation in XML
  3. Comparison of current list of defects with previous list of defects and separation of new defects
  4. Searching of authors and e-mails for each defected code and XML modification
  5. Transformation of XML into *.html* table (current defects table)
  6. Creation of Jira tickets
  7. Modification of Full defects table - *.html* table
  8. Comparison of Full defects list with Current defects list and identification of fixed defects -> *.html* table generation

- Status of process automation

- 😊 1. Cloning Athena from gitlab by git pull
- 😊 2. Local CPPcheck scan of Athena and results generation in XML
- 😊 3. Comparison of current list of defects with previous list of defects and separation of new defects
- 😊 4. Searching of authors and e-mails for each defected code and XML modification
- 😊 5. Transformation of XML into *.html* table (current defects table)
- 😞 6. Creation of Jira tickets
- 😞 7. Modification of Full defects table - *.html* table
- 😞 8. Comparison of Full defects list with Current defects list and identification of fixed defects -> *.html* table generation

- Our latest CPPcheck defects analyses Week 04/02/2020–11/02/2020:

- Defects found: 5
- New authors added: Vakhtang Tsulaia  
William Axel Leight
- Jira tickets generated: 2

- We provide results in 2 ways:

1. *.html* table

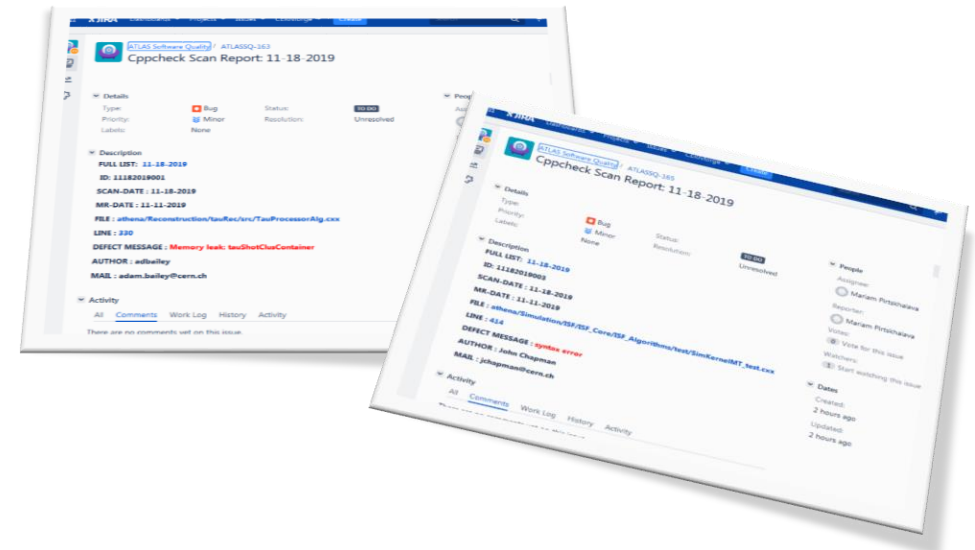
<https://cppcheck-list.web.cern.ch/cppcheck-list/>

<a href="#">021120201</a>	02-11-2020	<a href="#">athena/ControlAthenaServices/src/AthenaMtesEventLoopMgr.cxx</a>	02-06-2020	<a href="#">1357</a>	Memory leak: ready_message	Vakho Tsulaia	vakhtang.tsulaia@cern.ch
<a href="#">021120202</a>	02-11-2020	<a href="#">athena/MuonSpectrometer/MuonReconstruction/MuonSegmentMakers/src/MuonSegmentMakerAlgs/CscSegmentMakers/src/CscSegmentUtilTool.cxx</a>	02-10-2020	<a href="#">1572</a>	Invalid iterator 'icl' used.	William Axel Leight	william.axel.leight@cern.ch
<a href="#">021120203</a>	02-11-2020	<a href="#">athena/MuonSpectrometer/MuonReconstruction/MuonSegmentMakers/src/MuonSegmentMakerAlgs/CscSegmentMakers/src/CscSegmentUtilTool.cxx</a>	02-10-2020	<a href="#">1588</a>	Invalid iterator 'icl' used.	William Axel Leight	william.axel.leight@cern.ch

Opens Code on Gitlab

Opens Code on the defected string

2. Jira Tickets



- Objective is to read latest MR's from gitlab, initiate individual CPPcheck scanning and provide defects report back on gitlab
- We have developed process with 6 consecutive steps:
  - ☹️ 1. Access to latest MR from the Master Label page
  - ☹️ 2. Separation of latest MR and get access to file path
  - ☹️ 3. Cloning (updating) Athena locally
  - ☹️ 4. CPPcheck scanning start-up for the given code and path and generation of XML
  - ☹️ 5. Filling the templet from XML
  - ☹️ 6. Upload templet on gitlab into the analyzed MR section

Defects templet

issue create date	mm/aa/yyyy;
scan date	mm/dd/yyyy;
VR open date	mm/dd/yyyy;
VR Title	
VR Description	*** ** *
VR Last edit	Time/by
.abel	**** ** *
VR file path	*** ** */***
Defect message	*** ** ** *
Defect line	**** ** *
Author	**** ** *
Mail	@cern.ch

# CPPcheck Scan / Process-II

- We did test scans of latest 5 MR's from gitlab
- Defects are in xml
- Also filled templets are also ready

18/11/2019 MR-4 defects Templet

SUE CREATE DATE	20-11-2019
AN DATE	20-11-2019
R OPEN DATE	20-11-2019
R TITLE	18-11-2019
R DESCRIPTION	<a href="#">Add SCTInformation.h (ATLIDTRKCP-174)</a> Add SCTInformation.h, Use SCTInformation in SiSpacePointMakerTool.
R LAST EDIT	20-11-2019 by Susumu Oda
BEL	<a href="#">InnerDetector</a>
R FILE PATH	<a href="#">InnerDetector/InDetRecTools/SiClusterizationTool/SiClusterizationTool/SCT_ClusteringTool.h</a>
DEFECT MESSAGE	Code &apos;namespaceInDetDD(&apos; is invalid C code. Use --std or --language to configure the language.
DEFECT LINE	30
JTHOR	<a href="#">Susumu Oda</a>

18/11/2019 MR-2 defects

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <results version="2">
3   <cppcheck version="1.82"/>
4   <errors>
5     <error id="syntaxError" severity="error" msg="Code &apos;namespaceTrk(&apos;
6       is invalid C code. Use --std or --language to configure the language."
7       verbose="Code &apos;namespaceTrk(&apos; is invalid C code. Use --std or --
8       language to configure the language.">
9       <location
10        file="InnerDetector/InDetRecTools/SiSpacePointTool/SiSpacePointTool/Si
11        nMakerTool.h" line="21"/>
12     </error>
13   </errors>
14 </results>
15 </xml version="1.0" encoding="UTF-8"?>
16 <results version="2">
17   <cppcheck version="1.82"/>
18   <errors>
19     <error id="syntaxError" severity="error" msg="No pair for character (&quot;).
20       Can&apos;t process file. File is either invalid or unicode, which is currently not
21       supported." verbose="No pair for character (&quot;). Can&apos;t process file. File
22       is either invalid or unicode, which is currently not supported.">
23       <location
24        file="InnerDetector/InDetRawAlgs/InDetOverlay/python/PixelOverlayConfig.py"
25        line="17"/>
26     </error>
27   </errors>
28 </results>
```

18/11/2019 MR-1 defects

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <results version="2">
3   <cppcheck version="1.82"/>
4   <errors>
5     <error id="syntaxError" severity="error" msg="Code &apos;namespaceTrk(&apos;
6       is invalid C code. Use --std or --language to configure the language."
7       verbose="Code &apos;namespaceTrk(&apos; is invalid C code. Use --std or --
8       language to configure the language.">
9       <location
10        file="InnerDetector/InDetRecTools/SiSpacePointTool/SiSpacePointTool/Si
11        nMakerTool.h" line="21"/>
12     </error>
13   </errors>
14 </results>
15 </xml version="1.0" encoding="UTF-8"?>
16 <results version="2">
17   <cppcheck version="1.82"/>
18   <errors>
19     <error id="syntaxError" severity="error" msg="Code &apos;namespaceInDetDD(&apos;
20       is invalid C code. Use --std or --language to configure the language."
21       verbose="Code &apos;namespaceInDetDD(&apos; is invalid C code. Use --std or --
22       language to configure the language.">
23       <location
24        file="InnerDetector/InDetRecTools/SiClusterizationTool/SiClusterizationTool/S
25        CT_ClusteringTool.h" line="30"/>
26     </error>
27   </errors>
28 </results>
```

18/11/2019 MR-4 defects

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <results version="2">
3   <cppcheck version="1.82"/>
4   <errors>
5     <error id="syntaxError" severity="error" msg="No pair for character (&apos;).
6       Can&apos;t process file. File is either invalid or unicode, which is currently not
7       supported." verbose="No pair for character (&apos;). Can&apos;t process file. File
8       is either invalid or unicode, which is currently not supported.">
9       <location
10        file="InnerDetector/InDetExample/InDetRecExample/share/InDetRecLoadTools.py"
11        line="411"/>
12     </error>
13   </errors>
14 </results>
```

18/11/2019 MR-5 defects

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <results version="2">
3   <cppcheck version="1.82"/>
4   <errors>
5     <error id="syntaxError" severity="error" msg="Code &apos;namespaceInDetDD(&apos;
6       is invalid C code. Use --std or --language to configure the language."
7       verbose="Code &apos;namespaceInDetDD(&apos; is invalid C code. Use --std or --
8       language to configure the language.">
9       <location
10        file="InnerDetector/InDetRecTools/SiClusterizationTool/SiClusterizationTool/S
11        CT_ClusteringTool.h" line="30"/>
12     </error>
13   </errors>
14 </results>
```

18/11/2019 MR-3 defects

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <results version="2">
3   <cppcheck version="1.82"/>
4   <errors>
5     <error id="syntaxError" severity="error" msg="Code &apos;namespaceInDetDD(&apos;
6       is invalid C code. Use --std or --language to configure the language."
7       verbose="Code &apos;namespaceInDetDD(&apos; is invalid C code. Use --std or --
8       language to configure the language.">
9       <location
10        file="InnerDetector/InDetRecTools/SiClusterizationTool/SiClusterizationTool/S
11        CT_ClusteringTool.h" line="30"/>
12     </error>
13   </errors>
14 </results>
```



- In our SQ group we have 31 authors

1. Adam Baley
2. Adam Edward Barton
3. Ahmed Hasib
4. Andrei Sukharev
5. Apostolos Tsirigotis
6. Benedict Tobias Winter
7. Ban Nachman
8. Charles Barton
9. Chris Lee
10. Christos Anastopoulos
11. Dario Barberis
12. Edward Moyse
13. Goetz Gaycken
14. Hao Xu
15. John Derek Chapman
16. Matous Vozak
17. Nicolas Koehler
18. Nikita Belyaev
19. Pascal Boeschoten
20. Peter Onysi
21. Rafal Bielski
22. Ruth POTTgen
23. Scott Snyder
24. Shaun Roe
25. Soshi Tsuno
26. Susumo Oda
27. Tim Martin
28. Tomasz Bold
29. Walter Lampl
30. Vakhtang Tsulaia
31. William Axel Leight

- We have good feedbacks from authors:

The image displays three overlapping screenshots of GitHub issue discussions. The top-left screenshot shows a defect message for issue 10012019002, with a comment from Tomasz Bold dated 25/Oct/19. The top-right screenshot shows a defect message for issue 11182019001, with comments from Adam Bailey and Mariam Pirtskhalava. The bottom-center screenshot shows a comment from Ruth Pottgen dated 07/Jan/20 and a response from Benedict Tobias Winter dated 07/Jan/20.

**Issue 10012019002:**  
Description: FULL LIST: 10-01-2019, ID: 10012019002, SCAN-DATE: 10-01-2019, MR-DATE: 09-27-2019. FILE: https://gitlab.cern.ch/atlas/athena/blob/nightly/master/2019-09-30T2132/Calorimeter/CaloCondBlobObjs/test/CaloCondBlobBase\_test.cxx. LINE: 93. DEFECT MESSAGE: Shifting signed 32-bit value by 31 bits is undefined behaviour. AUTHOR: Tomasz Bold. MAIL: tomasz.bold@gmail.com.

**Issue 11182019001:**  
Description: FULL LIST: 11-18-2019, ID: 11182019001, SCAN-DATE: 11-18-2019, MR-DATE: 11-11-2019. FILE: athena/Reconstruction/tauRec/src/TauProcessorAlg.cxx. LINE: 330. DEFECT MESSAGE: Memory leak: tauShotClusContainer. AUTHOR: adbailey. MAIL: adam.bailey@cern.ch.

**Comments on Issue 10012019002:**  
Tomasz Bold added a comment - 25/Oct/19 12:46: Fixing by adding explicit cast to unit32\_t in CTPUnpa...  
Tomasz Bold added a comment - 21/Nov/19 5: This was fixed for the CTPUnpackingTool.cxx. Reassigning to Shaun for Calo.

**Comments on Issue 11182019001:**  
Adam Bailey added a comment - 21/Nov/19 3:12 PM: Hi. I see that we have several containers declared with new that should be deleted. However, when I simply try to delete the object at the end of execute it causes a seg fault. I think this could be due to these objects being used in WriteHandles. Is there a proper way to delete the objects in this case?  
Mariam Pirtskhalava added a comment - 22/Nov/19 1:40 PM: Hi Adam Bailey. Sorry i can't help you with that. I have no information about those containers and fixes. Maybe someone from group members can help you, if tagging them?

**Comments on Issue 12242019002:**  
Ruth Pottgen added a comment - 07/Jan/20 10:16 AM: I am not sure what to do with this. (I only merged the MR into 21.0.) It is not clear to me how the defect message fits to the indicated line. Moreover, that line was not changed by the MR. Maybe the author of the original MR, Benedict Tobias Winter, has an idea?  
Benedict Tobias Winter added a comment - 07/Jan/20 5:47 PM: Hi Ruth Pottgen and all, thanks for reporting. The warning is indeed thrown in a part of the code that was not touched recently. Perhaps the '<<' in combination with an int is interpreted as a binary shift operator instead of as an overload. It would be helpful to ask an expert for cppcheck what exactly causes the error and how to best avoid it.

## II. Coverity Scan

- Coverity in ATLAS was stopped in January 2019
- Before that day scanning processes were workable and we were doing weekly scans generating *.html* tables with a list of newly detected defects
- Athena full scan was taking ~30 hours on voatlas405 machine
- Coverity came back in October 2019 on new machine aibuild002.cern.ch
- Now we are trying to re-start Coverity scans

- Coverity needs preparation of different package then CPPcheck
- This is a binary Athena build which takes about 18 hour during the scanning
- Preparation of Athena binary build is complex work and we are not experts in this field
- In our shell's we are using external shells and some of them now starts failing. We do not always understand why it happens. So we need external assistance for successful re-start of Coverity scan

- Status of main start-up shell for Athena build is as follow:

Coverity Scan start-up shell from 2018

Setup build folders on aibuild002: done

Run ATLAS Setup on cvmfs: done

Get Athena clone from git: done

Athena build preparation: done

Run Coverity scan: current

```
1 #!/bin/bash
2 echo "cov-alex ID: $$"
3
4 # defaults
5 branch="master-alex"
6 proj="main"
7 DATESTAMP=`date +%d%m%y-%H%M`
8 CDIR=/build/Coverity/cov-sa
9 COVANA=$CDIR/bin/cov-analyze
10 COVCOM=$CDIR/bin/cov-commit-defects
11 PROJECT="Athena"
12 cd /build/alex1
13
14 touch log-cov-alex.txt
15 function log() {
16     echo "$@"
17     echo `date +%F %T`: "$@" >> log-cov-alex.txt }
18 # Initialize directories for BUILD
19 BUILDDIR="/build/alex1/$branch/$DATESTAMP/$proj"
20 COVBUILDDIR="$BUILDDIR/cov"
21 STREAM=$PROJECT
22 STRIPPATH="$BUILDDIR/athena/"
23
24 # Create directories for BUILD
25 mkdir -p $BUILDDIR
26 mkdir -p $COVBUILDDIR
27 cd $BUILDDIR
28
29 # Kerberos needed for Git clone?
30 #_kinit_
31
32 # ATLAS setup
33 log "ATLAS setup" $(date)
34 source /cvmfs/atlas.cern.ch/repo/ATLASLocalRootBase/user/atlasLocalSetup.sh
35 asetup none,gcc62,opt --cmakesetup
36 lsetup git
37
38 # Get snapshot of code
39 #log "Cloning Athena repo (https://@gitlab.cern.ch:8443/atlas/athena.git)"
40 git clone https://@gitlab.cern.ch:8443/atlas/athena.git
41 # symlink code snapshot (for testing only)
42 #ln -s /build/testing-wash/athena .
43
44 # Run externals
45 #log "Starting Athena Externals Build"
46 #./athena/Projects/Athena/build_externals.sh -b $BUILDDIR
47
48 # Run build with coverity option
49 #log "Starting Athena Build ./athena/Projects/Athena/build_cov.sh -c -m -i -b $BUILDDIR -x $COVBUILDDIR"
50 #./athena/Projects/Athena/build_cov.sh -c -m -i -b $BUILDDIR -x $COVBUILDDIR
51
52
53
54
55 echo "end of: $$"
```

- For the moment we have issue with `yampl.git` to ensure external access from machine
  - `yampl-gitclone.cmake:66` (message):  
**Failed to clone repository: 'https://github.com/vitillo/yampl.git'**
  - `94%] Built target Package_Gdbfatal: unable to access 'https://github.com/vitillo/yampl.git/': Failed connect to github.com:443; Connection timed out-- Had to git clone more than once: 3 times.CMake Error at /build/alex1/master/101219-1324/main/build/AthenaExternals/tmp/yampl-gitclone.cmake:66 (message): Failed to clone repository: 'https://github.com/vitillo/yampl.git'`
  - `Scanning dependencies of target G4processesmake[2]: *** [src/yampl-stamp/yampl-download] Error 1make[2]: Target `External/yampl/CMakeFiles/yampl.dir/build' not remade because of errors.make[1]: *** [External/yampl/CMakeFiles/yampl.dir/all] Error 2fatal: unable to access 'https://github.com/google/googletest.git/': Failed connect to github.com:443; Connection timed out-- Had to git clone more than once: 3 times.CMake Error at /build/alex1/master/101219-1324/main/build/AthenaExternals/tmp/GoogleTest-gitclone.cmake:66 (message): Failed to clone repository: 'https://github.com/google/googletest.git'`
- We have heard several options to solve it on the last ASCIG weekly meeting (thanks to everybody) and we will try to proceed accordingly

- Future Plans for Coverity:
  1. Re-build Coverity scanning of Full Athena repository
  2. Development of incremental scanning process to reduce time
  3. Development of Full list of current defects in *.html* table
  4. Development of *.html* table with all defects for all scans
  5. Development of comparison process of *.html* tables to generate the list of fixed defects
  6. Make automation of scanning processes

Comments are welcome,

Thanks!