Coverity Usage

This document describes:

1. the overview on Coverity
2. the instruction on the installation
3. The workflow of the Coverity use

Revision History

Rev	Description	Issuer
1.00	New	Kodai Muraki

Coverity is a static code analyser. Coverity can analyse all MISRA-C2012 rules including Amendment1.

Schema

Coverity consists of follows:

Coverity Connect:

- is server software with DB
- displays analysis results
- can be accessed by any web browser, no dedicated software is needed

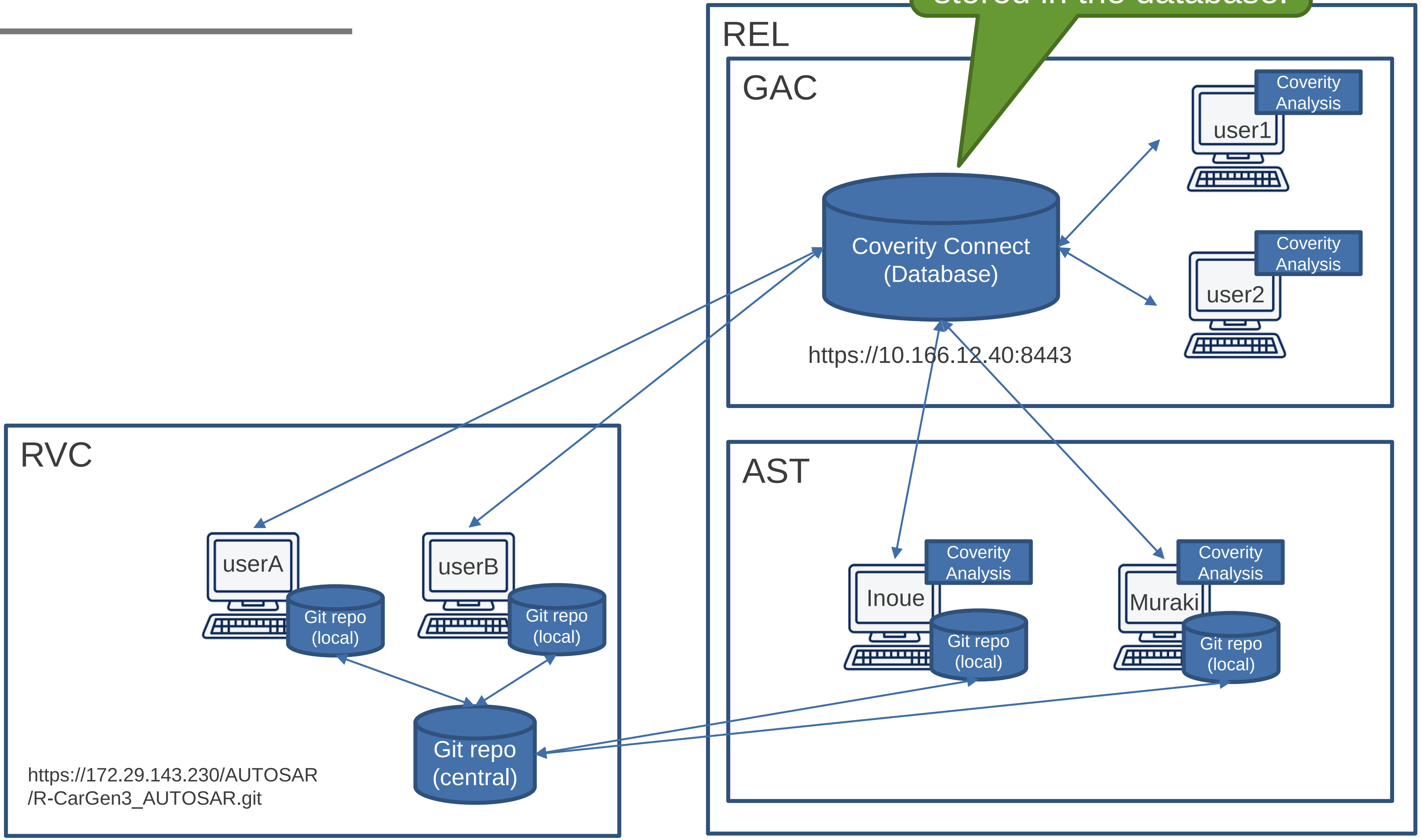
Coverity Analysis:

- is client software
- analyses source codes and push the analysis results to Coverity Connect

Git repository:

- stores source code

Schema

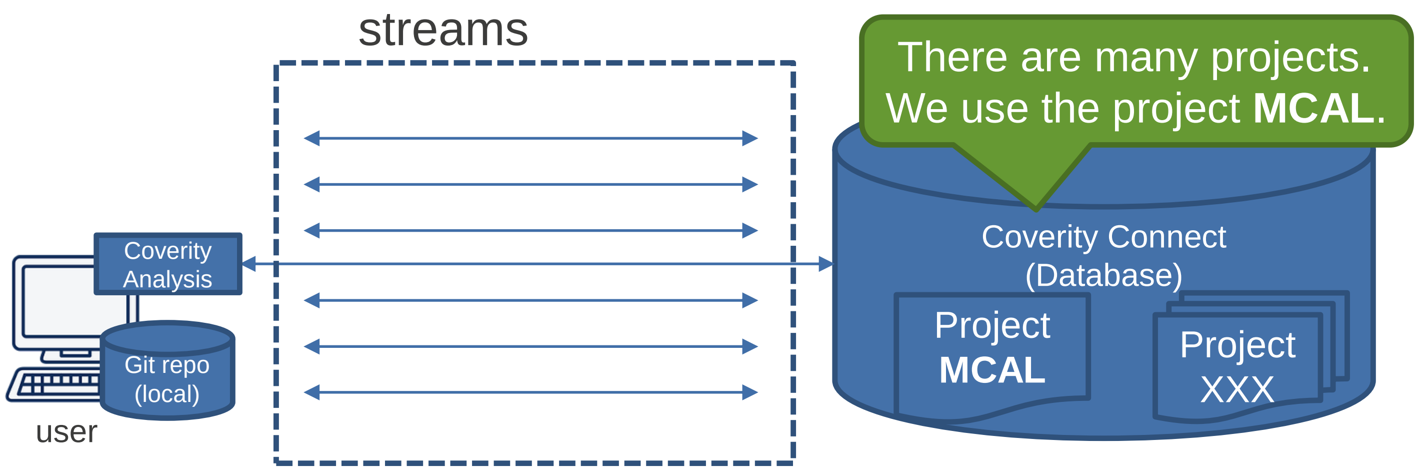


All analysis results are stored in the database.

How to work Coverity?

The essential workflow of Coverity is follows:

1) Coverity Analysis runs a make file of R-Car MCAL
2) Coverity Analysis analyses the making process
3) Coverity Analysis builds R-Car MCAL source by the Coverity own compiler
4) Coverity Analysis creates intermediate information including static analysis results
5) Coverity Analysis pushes the analysis result to Coverity Connect through a stream
6) Users can see the pushed analysis results on user's web browser



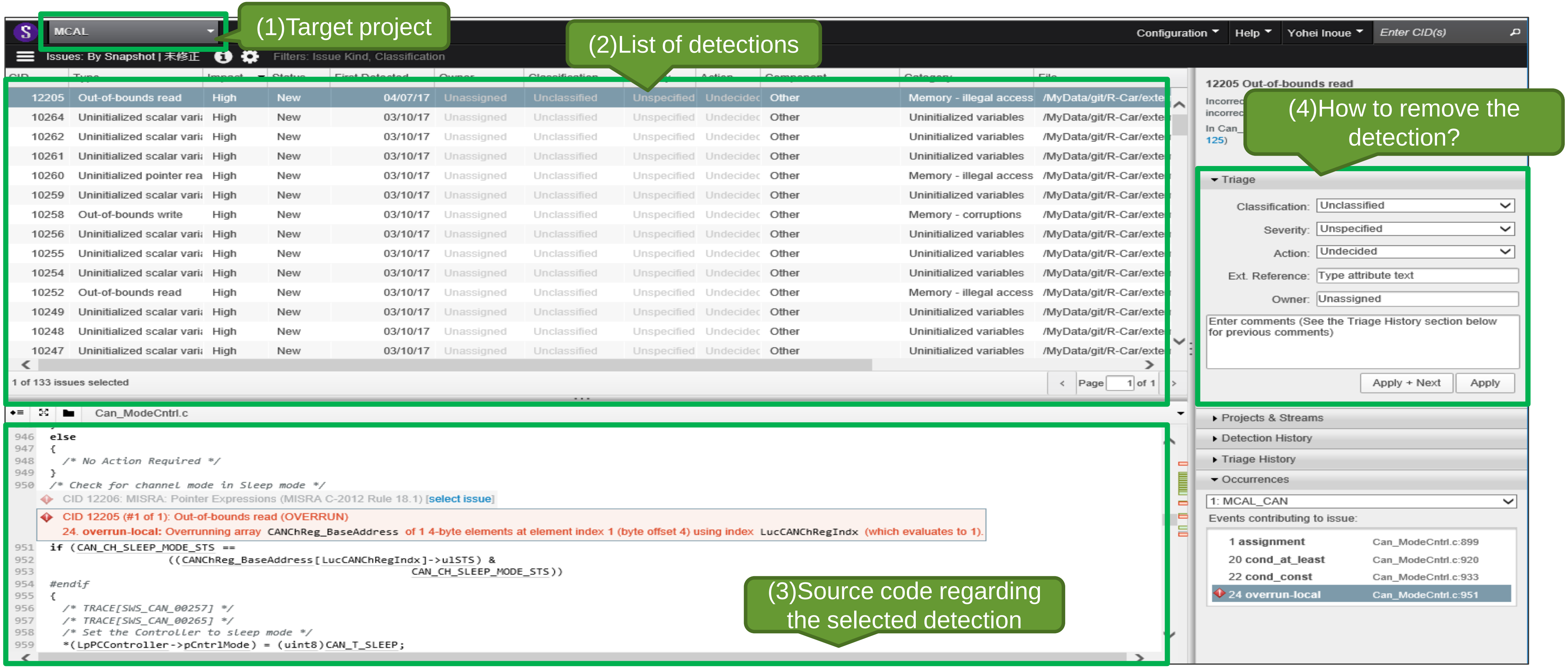
There are many projects.
We use the project **MCAL**.

One stream for one module,
One project for all MCAL-related streams
for comprehensibility and maintainability

Project	Stream	Module
MCAL	MCAL_ADC	V3x MCAL ADC module
	MCAL_CAN	V3x MCAL CAN module
	MCAL_DIO	V3x MCAL DIO module
	MCAL_ETH	V3x MCAL ETH module
	MCAL_FLS	V3x MCAL FLS module
	MCAL_GPT	V3x MCAL GPT module
	MCAL_MCU	V3x MCAL MCU module
	MCAL_PORT	V3x MCAL Port module
	MCAL_SPI	V3x MCAL SPI module
	MCAL_WDG	V3x MCAL WDG module
	MCAL_CDDCMT1	V3x CDD CMT1 module
	MCAL_CDDBus Monitor	V3x CDD Bus Monitor module
	MCAL_CDDCRC	V3x CDD CRS module
	MCAL_CDDTHS	V3x CDD THS module
	MCAL_CDDICCOM	V3x CDD ICCOM module
	MCAL_CDDSUCMT	V3x CDD SUCMT module

Overview on Coverity Connect

To see the analysis results, we need to visit Coverity Connect on our web browser.



MCAL
Configuration
Help
Yohei Inoue
Enter CID(s)
Issues: By Snapshot | 未修正
Filters: Issue Kind, Classification

(1)Target project

(2)List of detections

CID	Type	Impact	Status	First Detected	Owner	Classification	Severity	Action	Component	Category	File
12205	Out-of-bounds read	High	New	04/07/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/exte
10264	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10262	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10261	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10260	Uninitialized pointer rea	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/exte
10259	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10258	Out-of-bounds write	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - corruptions	/MyData/git/R-Car/exte
10256	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10255	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10254	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10252	Out-of-bounds read	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/exte
10249	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10248	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte
10247	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/exte

1 of 133 issues selected

Page 1 of 1

Can_ModeCntrl.c

```
946   else
947   {
948     /* No Action Required */
949   }
950   /* Check for channel mode in Sleep mode */
      CID 12206: MISRA: Pointer Expressions (MISRA C-2012 Rule 18.1) [select issue]
      CID 12205 (#1 of 1): Out-of-bounds read (OVERRUN)
      24. overrun-local: Overrunning array CANChReg_BaseAddress of 1 4-byte elements at element index 1 (byte offset 4) using index LucCANChRegIndx (which evaluates to 1).
951   if (CAN_CH_SLEEP_MODE_STS ==
952                    ((CANChReg_BaseAddress[LucCANChRegIndx]->ulSTS) &
953                                             CAN_CH_SLEEP_MODE_STS))
954   #endif
955   {
956     /* TRACE[SWS_CAN_00257] */
957     /* TRACE[SWS_CAN_00265] */
958     /* Set the Controller to sleep mode */
959     *(LpPController->pCntrlMode) = (uint8)CAN_T_SLEEP;
```

(3)Source code regarding the selected detection

12205 Out-of-bounds read

Incorrec
incorrec
In Can_
125)

(4)How to remove the detection?

Triage

Classification: Unclassified

Severity: Unspecified

Action: Undecided

Ext. Reference: Type attribute text

Owner: Unassigned

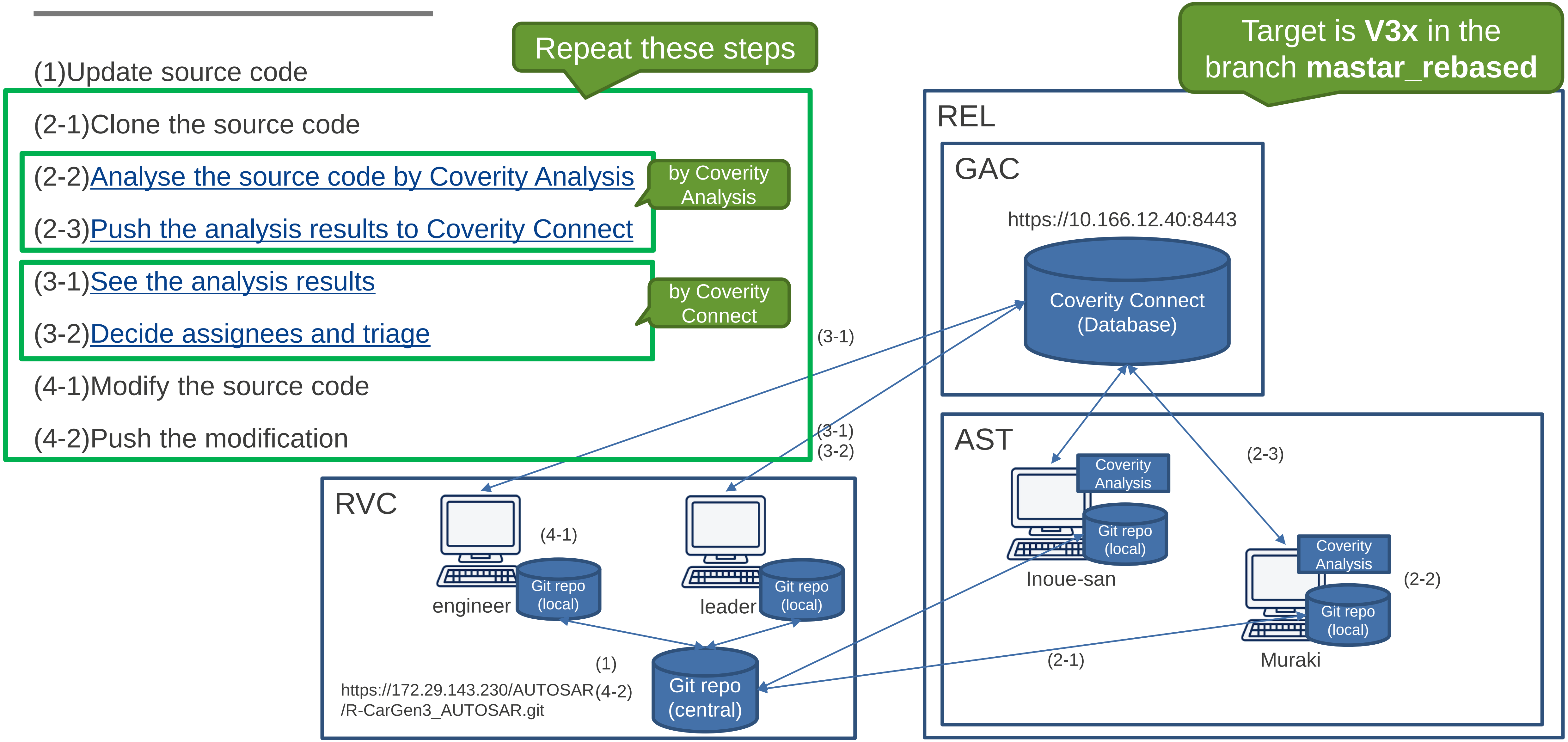
Enter comments (See the Triage History section below for previous comments)

1: MCAL_CAN

Events contributing to issue:

1 assignment	Can_ModeCntrl.c:899
20 cond_at_least	Can_ModeCntrl.c:920
22 cond_const	Can_ModeCntrl.c:933
24 overrun-local	Can_ModeCntrl.c:951

Workflow of analysing and modifying MCAL source code



(1)Update source code

Repeat these steps

(2-1)Clone the source code

(2-2)Analyse the source code by Coverity Analysis

(2-3)Push the analysis results to Coverity Connect

by Coverity Analysis

(3-1)See the analysis results

(3-2)Decide assignees and triage

by Coverity Connect

(4-1)Modify the source code

(4-2)Push the modification

Target is **V3x** in the branch **mastar_rebased**

Analyse the source code by Coverity Analysis and Push the results

First of all, we need to analyse MCAL source code by Coverity Analysis to detect issues. To analyse the MCAL source code, use either of:

- the Coverity Wizard project (.cwz)
- the batch file (.bat) to execute Coverity Analysis from the command line

All R-Car related software development SHALL follow the rules unanimously agreed among GAS and AST, because All R-Car related software shall achieve the same level of quality. The rules can be seen in the appendix section in this document.

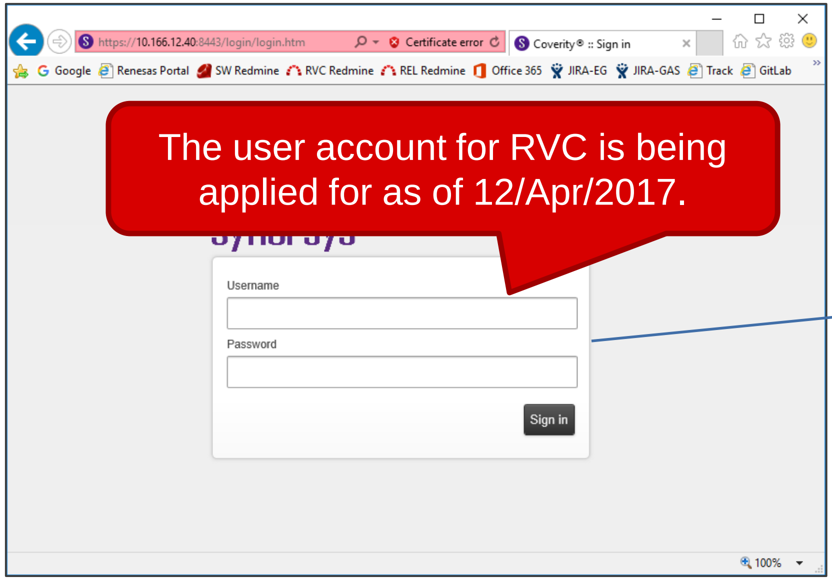
* The rule here means a detection rule.

Muraki already created both (.cwz, .bat) that strictly complies with the rules and analyses R-Car MCAL source code. They are stored in Muraki's OneDrive only as of 12/Apr/2017. They can be shared with RVC if necessary.

See the analysis result

To see the analysis result:

1) Run your web browser
2) Visit the Coverity Connect server at https://10.166.12.40:8443
3) Type the username and its password
4) Click on the **Sing in** button
5) Then, Coverity Connect will open

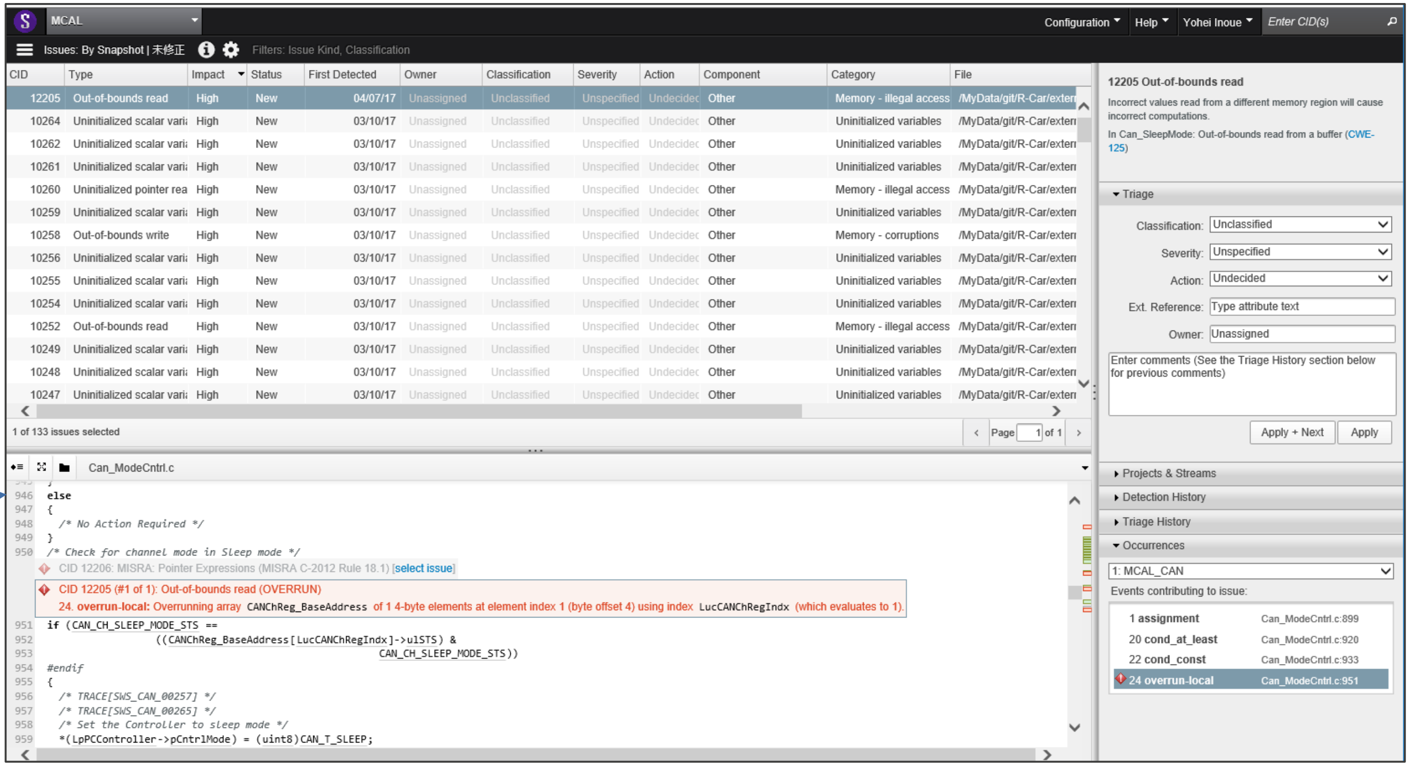


Username

Password

Sign in

The user account for RVC is being applied for as of 12/Apr/2017.



Issues: By Snapshot | 未修正

Filters: Issue Kind, Classification

CID	Type	Impact	Status	First Detected	Owner	Classification	Severity	Action	Component	Category	File
12205	Out-of-bounds read	High	New	04/07/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/extern
10264	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10262	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10261	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10260	Uninitialized pointer rea	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/extern
10259	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10258	Out-of-bounds write	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - corruptions	/MyData/git/R-Car/extern
10256	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10255	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10254	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10252	Out-of-bounds read	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Memory - illegal access	/MyData/git/R-Car/extern
10249	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10248	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern
10247	Uninitialized scalar vari	High	New	03/10/17	Unassigned	Unclassified	Unspecified	Undecidec	Other	Uninitialized variables	/MyData/git/R-Car/extern

1 of 133 issues selected

Page 1 of 1

Can_ModeCntrl.c

```
else
{
  /* No Action Required */
}
/* Check for channel mode in Sleep mode */
 CID 12206: MISRA: Pointer Expressions (MISRA C-2012 Rule 18.1) [select issue]
 CID 12205 (#1 of 1): Out-of-bounds read (OVERRUN)
 24. overrun-local: Overrunning array CANChReg_BaseAddress of 1 4-byte elements at element index 1 (byte offset 4) using index LucCANChRegIndx (which evaluates to 1).
if (CAN_CH_SLEEP_MODE_STS ==
              ((CANChReg_BaseAddress[LucCANChRegIndx]->ulSTS) &
                                                CAN_CH_SLEEP_MODE_STS))
#endif
{
  /* TRACE[SWS_CAN_00257] */
  /* TRACE[SWS_CAN_00265] */
  /* Set the Controller to sleep mode */
  *(LpPCController->pCntrlMode) = (uint8)CAN_T_SLEEP;
```

12205 Out-of-bounds read

Incorrect values read from a different memory region will cause incorrect computations.

In Can_SleepMode: Out-of-bounds read from a buffer (CWE-125)

Triage

Classification: Unclassified

Severity: Unspecified

Action: Undecided

Ext. Reference: Type attribute text

Owner: Unassigned

Enter comments (See the Triage History section below for previous comments)

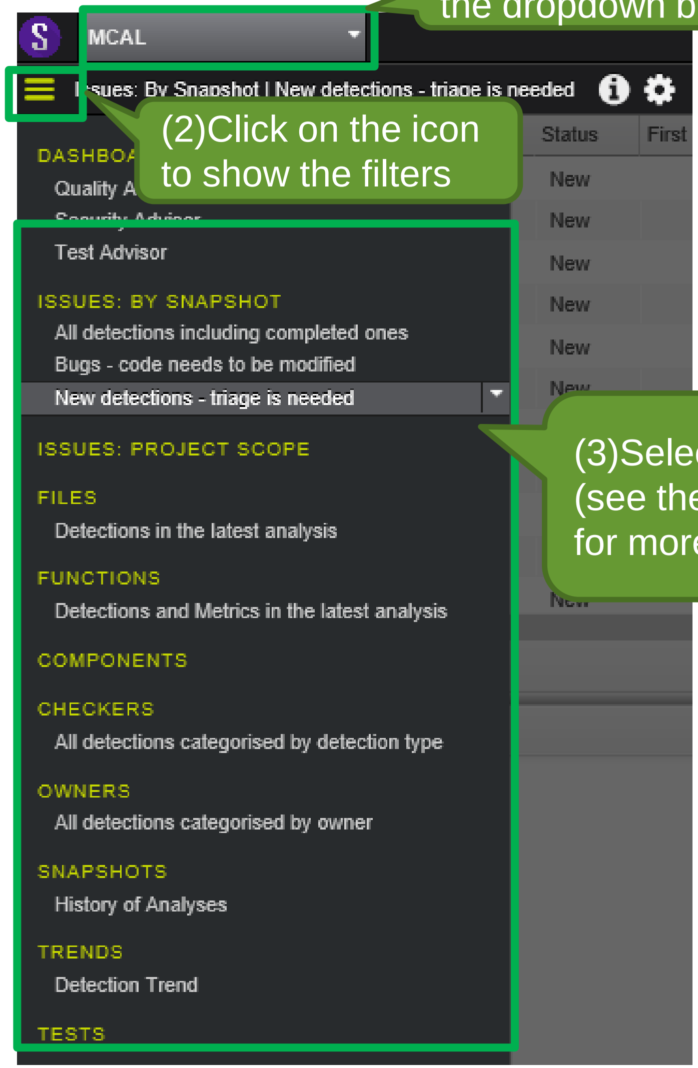
Apply + Next | Apply

1: MCAL_CAN

Events contributing to issue:

1 assignment	Can_ModeCntrl.c:899
20 cond_at_least	Can_ModeCntrl.c:920
22 cond_const	Can_ModeCntrl.c:933
24 overrun-local	Can_ModeCntrl.c:951

See the analysis result



(1)Select **MCAL** from the dropdown box

(2)Click on the icon to show the filters

(3)Select the filter (see the next page for more details)

MCAL

Issues: By Snapshot | New detections - triage is needed

DASHBOA…
- Quality A…
- Security Advisor
- Test Advisor

ISSUES: BY SNAPSHOT
- All detections including completed ones
- Bugs - code needs to be modified
- New detections - triage is needed

ISSUES: PROJECT SCOPE

FILES
- Detections in the latest analysis

FUNCTIONS
- Detections and Metrics in the latest analysis

COMPONENTS

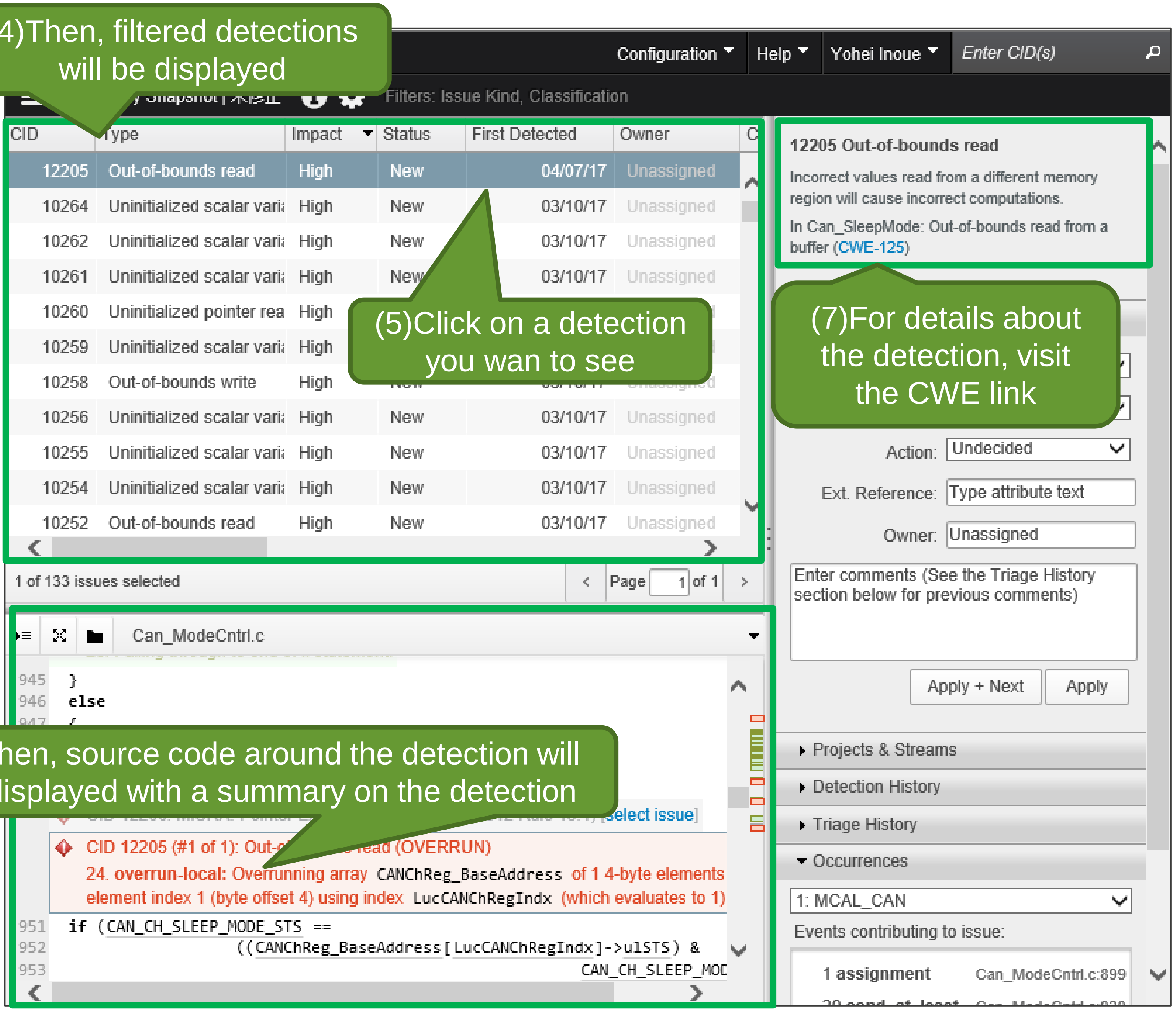
CHECKERS
- All detections categorised by detection type

OWNERS
- All detections categorised by owner

SNAPSHOTS
- History of Analyses

TRENDS
- Detection Trend

TESTS



(4)Then, filtered detections will be displayed

(5)Click on a detection you wan to see

(6)Then, source code around the detection will be displayed with a summary on the detection

(7)For details about the detection, visit the CWE link

CID	Type	Impact	Status	First Detected	Owner
12205	Out-of-bounds read	High	New	04/07/17	Unassigned
10264	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10262	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10261	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10260	Uninitialized pointer rea	High			
10259	Uninitialized scalar vari	High			
10258	Out-of-bounds write	High			
10256	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10255	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10254	Uninitialized scalar vari	High	New	03/10/17	Unassigned
10252	Out-of-bounds read	High	New	03/10/17	Unassigned

1 of 133 issues selected

Page 1 of 1

Can_ModeCntrl.c

```
}
else
{
```

[select issue]

CID 12205 (#1 of 1): Out-o… read (OVERRUN)

24. overrun-local: Overrunning array `CANChReg_BaseAddress` of 1 4-byte elements element index 1 (byte offset 4) using index `LucCANChRegIndx` (which evaluates to 1)

```
if (CAN_CH_SLEEP_MODE_STS ==
                ((CANChReg_BaseAddress[LucCANChRegIndx]->ulSTS) &
                                                        CAN_CH_SLEEP_MOD
```

12205 Out-of-bounds read

Incorrect values read from a different memory region will cause incorrect computations.

In Can_SleepMode: Out-of-bounds read from a buffer (CWE-125)

Action: Undecided

Ext. Reference: Type attribute text

Owner: Unassigned

Enter comments (See the Triage History section below for previous comments)

1: MCAL_CAN

Events contributing to issue:

1 assignment	Can_ModeCntrl.c:899

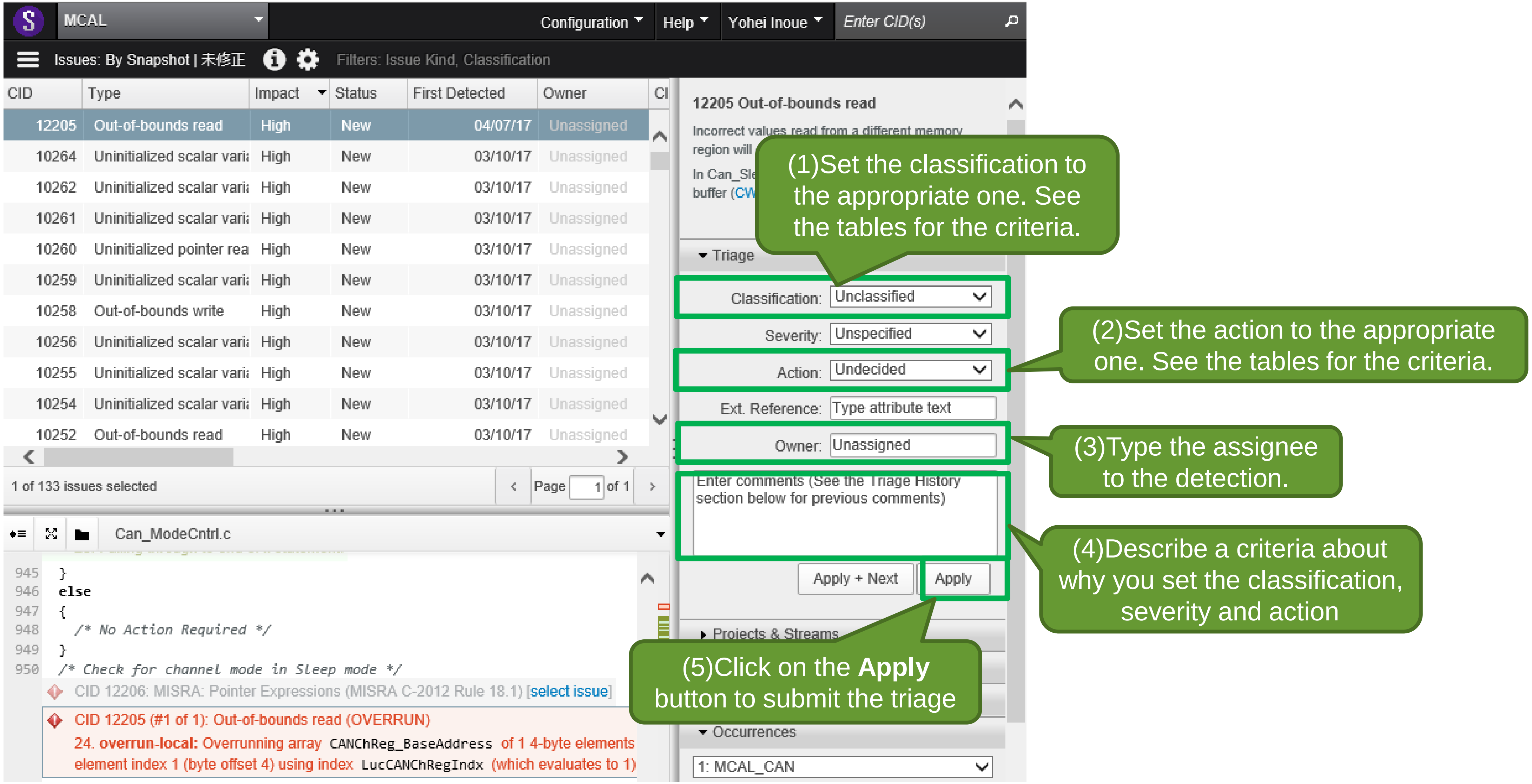
Filters provided

Filters can be used. Filters can be customised. Filters can be shared among users. The following customised filters are provided for MCAL users.

Filters

ID	Name	Purpose
F01	New detections - triage is needed	The detections are new. We need to decide their triage.
F02	Bugs - code needs to be modified	The code may cause a critical issue. The code shall be modified properly.
F03	History of Analyses	The analyses history helps us to know about their trend such as the number of the fixed detection or new detection.
F04	Metrics in the latest analysis	The metrics such as CCM - Cyclomatic Complexity Metric - help us to know about quality of the R-Car MCAL source code. Note: The metrics we need to follow can be seen at https://172.29.143.27/SS2/RCar_Autosar/99_Others/160606_FuSa Global Meeting MoM_5-7 MCAL metric targets.docx

Decide assignees and triage


S
MCAL
Configuration
Help
Yohei Inoue
Enter CID(s)
Issues: By Snapshot | 未修正
Filters: Issue Kind, Classification

CID	Type	Impact	Status	First Detected	Owner	Cl
12205	Out-of-bounds read	High	New	04/07/17	Unassigned	
10264	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10262	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10261	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10260	Uninitialized pointer rea	High	New	03/10/17	Unassigned	
10259	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10258	Out-of-bounds write	High	New	03/10/17	Unassigned	
10256	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10255	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10254	Uninitialized scalar vari	High	New	03/10/17	Unassigned	
10252	Out-of-bounds read	High	New	03/10/17	Unassigned	

1 of 133 issues selected

```
}
else
{
  /* No Action Required */
}
/* Check for channel mode in Sleep mode */
```

CID 12206: MISRA: Pointer Expressions (MISRA C-2012 Rule 18.1) [select issue]

CID 12205 (#1 of 1): Out-of-bounds read (OVERRUN)

24. **overrun-local:** Overrunning array `CANChReg_BaseAddress` of 1 4-byte elements element index 1 (byte offset 4) using index `LucCANChRegIndx` (which evaluates to 1)

12205 Out-of-bounds read

Incorrect values read from a different memory region will

In Can_Sle
buffer (CW

Triage

Classification:	Unclassified
Severity:	Unspecified
Action:	Undecided
Ext. Reference:	Type attribute text
Owner:	Unassigned

Enter comments (See the Triage History section below for previous comments)

(1)Set the classification to the appropriate one. See the tables for the criteria.

(2)Set the action to the appropriate one. See the tables for the criteria.

(3)Type the assignee to the detection.

(4)Describe a criteria about why you set the classification, severity and action

(5)Click on the **Apply** button to submit the triage

Decide assignees and triage

Classification

ID	Selection	Criteria to select
C0	Unclassified	The default value to be changed. We have to change to another one.
C1	Pending	Do not use.
C2	False Positive	The detection is not correct. Coverity detected by mistake.
C3	Intentional	The code is intentional. We do not have to modify the code.
C4	Bug	The code is a bug. We have to modify the code.

Action

ID	Selection	Criteria to select
A0	Unclassified	The default value to be changed. We have to change to another one.
A1	Fix Required	The source code has to be modified.
A2	Fix Submitted	T.B.D.
A3	Modeling Required	Do not use.
A4	Ignore	The detection can be ignored. We do nothing anymore.

Appendix

Installation of Coverity Static Analysis (client software)

The installer of Coverity Analysis can be obtain at:

T.B.D.

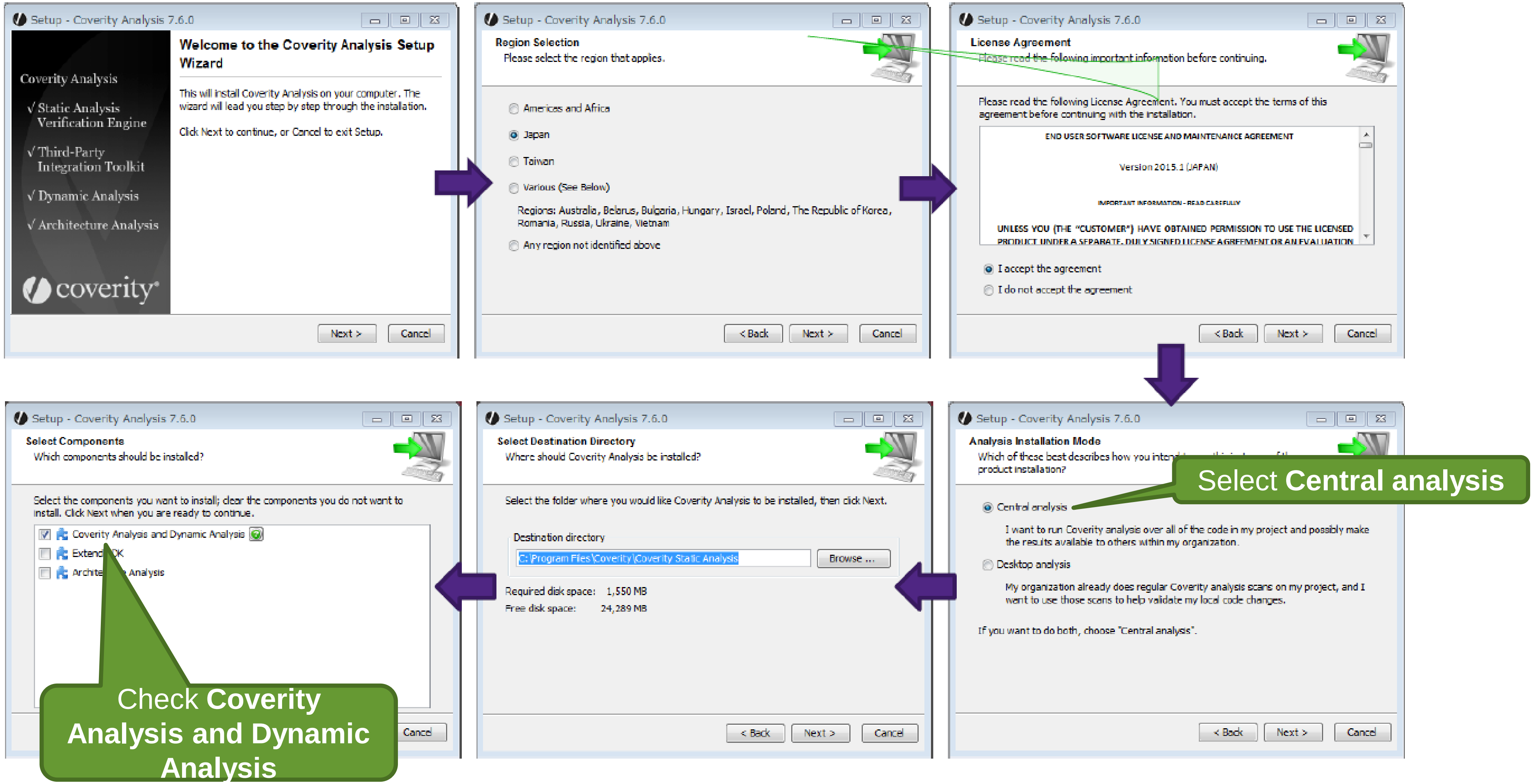
The license file for Coverity Analysis can be obtain at:

T.B.D.

REL/AST already obtained them, but their export control has not been finished yet.

Installation of Coverity Static Analysis (client software)

See the instruction below:



Setup - Coverity Analysis 7.6.0

Welcome to the Coverity Analysis Setup Wizard

This will install Coverity Analysis on your computer. The wizard will lead you step by step through the installation.

Click Next to continue, or Cancel to exit Setup.

Setup - Coverity Analysis 7.6.0

Region Selection

Please select the region that applies.

- ○ Americas and Africa
- ● Japan
- ○ Taiwan
- ○ Various (See Below)

Regions: Australia, Belarus, Bulgaria, Hungary, Israel, Poland, The Republic of Korea, Romania, Russia, Ukraine, Vietnam

- ○ Any region not identified above

Setup - Coverity Analysis 7.6.0

License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

END USER SOFTWARE LICENSE AND MAINTENANCE AGREEMENT

Version 2015.1 (JAPAN)

IMPORTANT INFORMATION - READ CAREFULLY

UNLESS YOU (THE "CUSTOMER") HAVE OBTAINED PERMISSION TO USE THE LICENSED PRODUCT UNDER A SEPARATE, DULY SIGNED LICENSE AGREEMENT OR AN EVALUATION

- ● I accept the agreement
- ○ I do not accept the agreement

Setup - Coverity Analysis 7.6.0

Analysis Installation Mode

Which of these best describes how you inten... product installation?

- ● Central analysis

  I want to run Coverity analysis over all of the code in my project and possibly make the results available to others within my organization.

- ○ Desktop analysis

  My organization already does regular Coverity analysis scans on my project, and I want to use those scans to help validate my local code changes.

If you want to do both, choose "Central analysis".

Select **Central analysis**

Setup - Coverity Analysis 7.6.0

Select Destination Directory

Where should Coverity Analysis be installed?

Select the folder where you would like Coverity Analysis to be installed, then click Next.

Destination directory

C:\Program Files\Coverity\Coverity Static Analysis

Required disk space: 1,550 MB

Free disk space: 24,289 MB

Setup - Coverity Analysis 7.6.0

Select Components

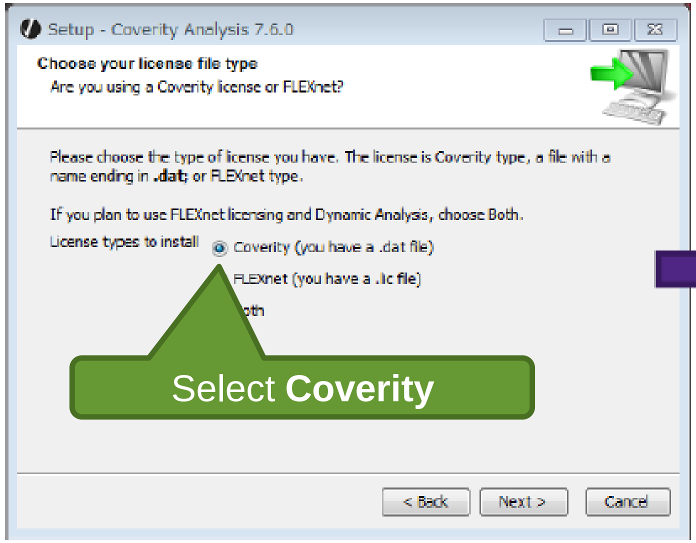
Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☑ Coverity Analysis and Dynamic Analysis
- □ Extend SDK
- □ Architecture Analysis

Check **Coverity Analysis and Dynamic Analysis**

Installation of Coverity Static Analysis (client software)



Setup - Coverity Analysis 7.6.0

Choose your license file type

Are you using a Coverity license or FLEXnet?

Please choose the type of license you have. The license is Coverity type, a file with a name ending in **.dat**; or FLEXnet type.

If you plan to use FLEXnet licensing and Dynamic Analysis, choose Both.

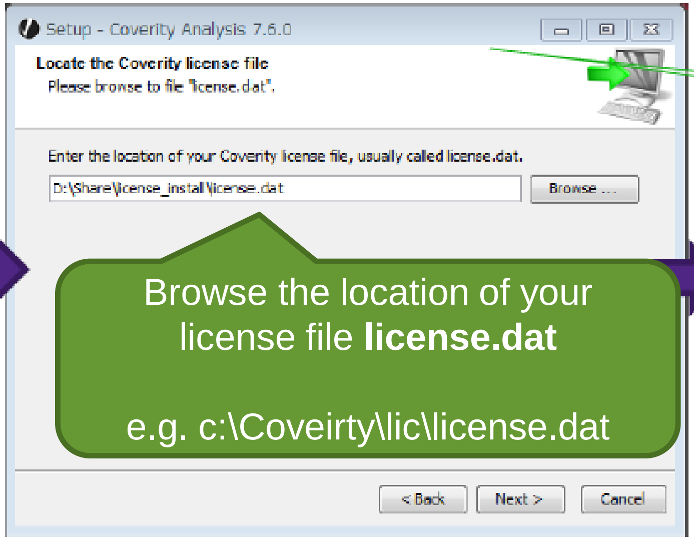
License types to install

● Coverity (you have a .dat file)

○ FLEXnet (you have a .lic file)

○ Both

Select **Coverity**



Setup - Coverity Analysis 7.6.0

Locate the Coverity license file

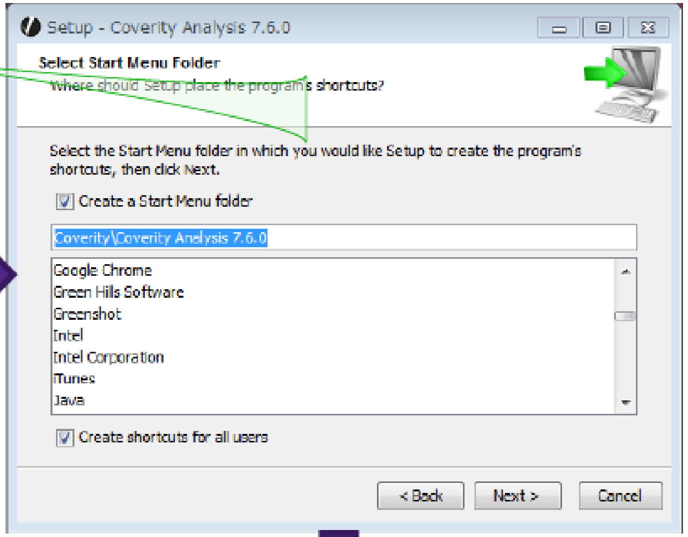
Please browse to file "license.dat".

Enter the location of your Coverity license file, usually called license.dat.

D:\Share\license_install\license.dat

Browse the location of your license file **license.dat**

e.g. c:\Coveirty\lic\license.dat



Setup - Coverity Analysis 7.6.0

Select Start Menu Folder

Where should Setup place the program's shortcuts?

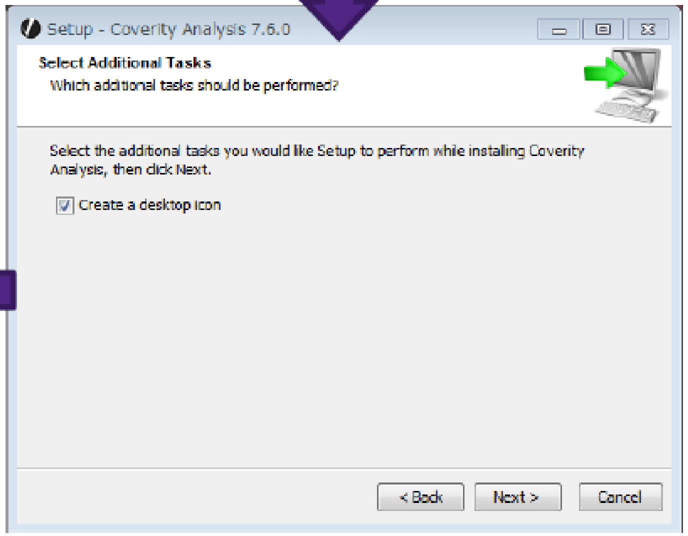
Select the Start Menu folder in which you would like Setup to create the program's shortcuts, then click Next.

☑ Create a Start Menu folder

Coverity\Coverity Analysis 7.6.0

Google Chrome
Green Hills Software
Greenshot
Intel
Intel Corporation
iTunes
Java

☑ Create shortcuts for all users



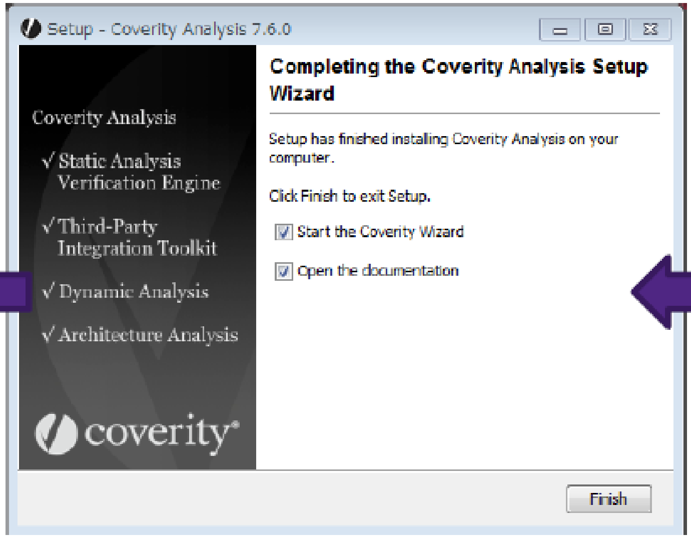
Setup - Coverity Analysis 7.6.0

Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Coverity Analysis, then click Next.

☑ Create a desktop icon



Setup - Coverity Analysis 7.6.0

Completing the Coverity Analysis Setup Wizard

Setup has finished installing Coverity Analysis on your computer.

Click Finish to exit Setup.

☑ Start the Coverity Wizard

☑ Open the documentation

You are ready to start using Coverity Analysis.

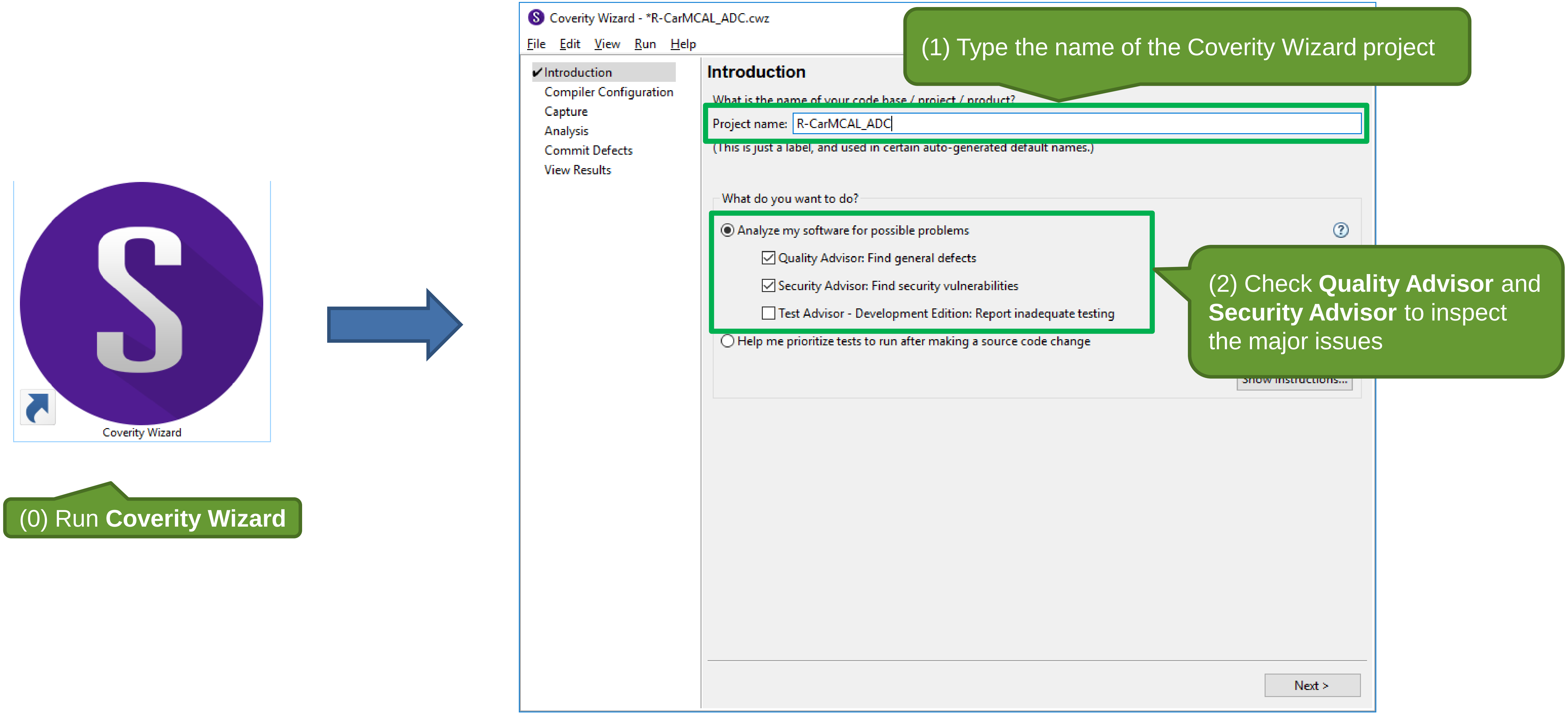
Workflow of Coverity Analysis

Prerequisites:

1. Source files to be analysed have been prepared.
2. The compiler to build the source files has been installed.
3. The make file or project file to build the source files has been prepared.
4. The MISRA-C configuration file has been duplicated for our customisation.

   The original configuration file can be seen at: <*Installation folder*>\config\MISRA\MISRA_c2012_7.config

Workflow of Coverity Analysis



(0) Run **Coverity Wizard**

Introduction

What is the name of your code base / project / product?

Project name: R-CarMCAL_ADC

(This is just a label, and used in certain auto-generated default names.)

What do you want to do?

● Analyze my software for possible problems

☑ Quality Advisor: Find general defects

☑ Security Advisor: Find security vulnerabilities

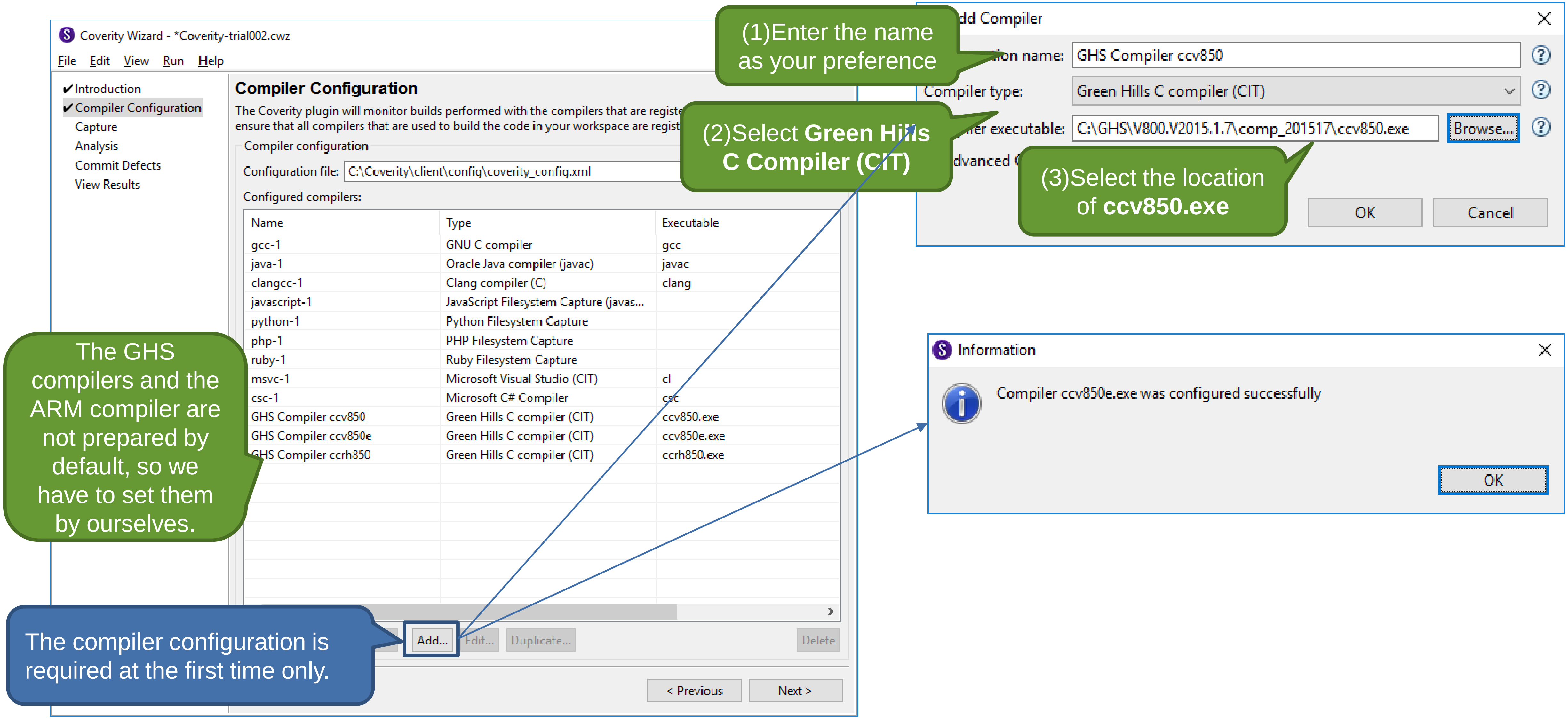
□ Test Advisor - Development Edition: Report inadequate testing

○ Help me prioritize tests to run after making a source code change

(1) Type the name of the Coverity Wizard project

(2) Check **Quality Advisor** and **Security Advisor** to inspect the major issues

Workflow of Coverity Analysis



Coverity Wizard - *Coverity-trial002.cwz

✔Introduction
✔Compiler Configuration
Capture
Analysis
Commit Defects
View Results

Compiler Configuration

The Coverity plugin will monitor builds performed with the compilers that are registe
ensure that all compilers that are used to build the code in your workspace are regist

Compiler configuration

Configuration file: C:\Coverity\client\config\coverity_config.xml

Configured compilers:

Name	Type	Executable
gcc-1	GNU C compiler	gcc
java-1	Oracle Java compiler (javac)	javac
clangcc-1	Clang compiler (C)	clang
javascript-1	JavaScript Filesystem Capture (javas...	
python-1	Python Filesystem Capture	
php-1	PHP Filesystem Capture	
ruby-1	Ruby Filesystem Capture	
msvc-1	Microsoft Visual Studio (CIT)	cl
csc-1	Microsoft C# Compiler	csc
GHS Compiler ccv850	Green Hills C compiler (CIT)	ccv850.exe
GHS Compiler ccv850e	Green Hills C compiler (CIT)	ccv850e.exe
GHS Compiler ccrh850	Green Hills C compiler (CIT)	ccrh850.exe

dd Compiler

ion name: GHS Compiler ccv850

Compiler type: Green Hills C compiler (CIT)

ler executable: C:\GHS\V800.V2015.1.7\comp_201517\ccv850.exe

Information

Compiler ccv850e.exe was configured successfully

(1)Enter the name as your preference

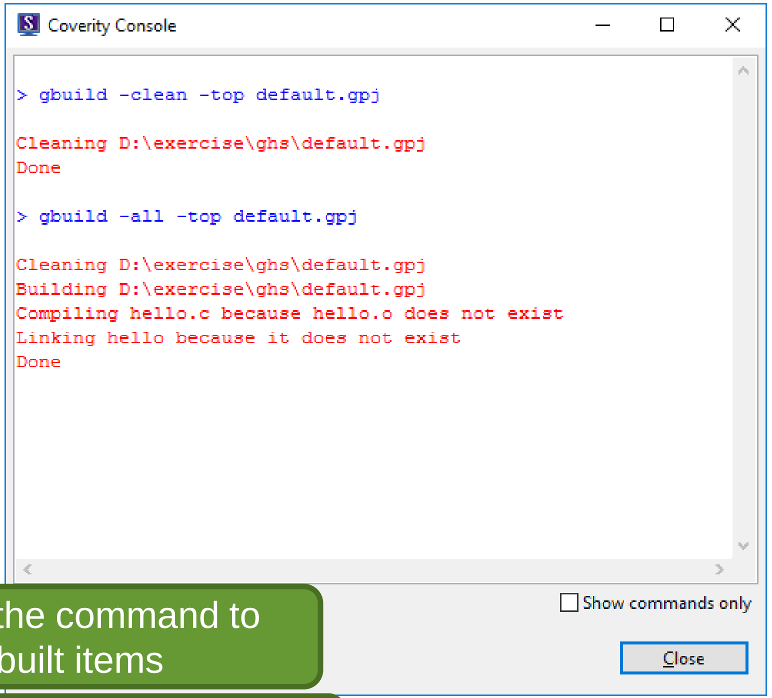
(2)Select **Green Hills C Compiler (CIT)**

(3)Select the location of **ccv850.exe**

The GHS compilers and the ARM compiler are not prepared by default, so we have to set them by ourselves.

The compiler configuration is required at the first time only.

Workflow of Coverity Analysis



Coverity Console

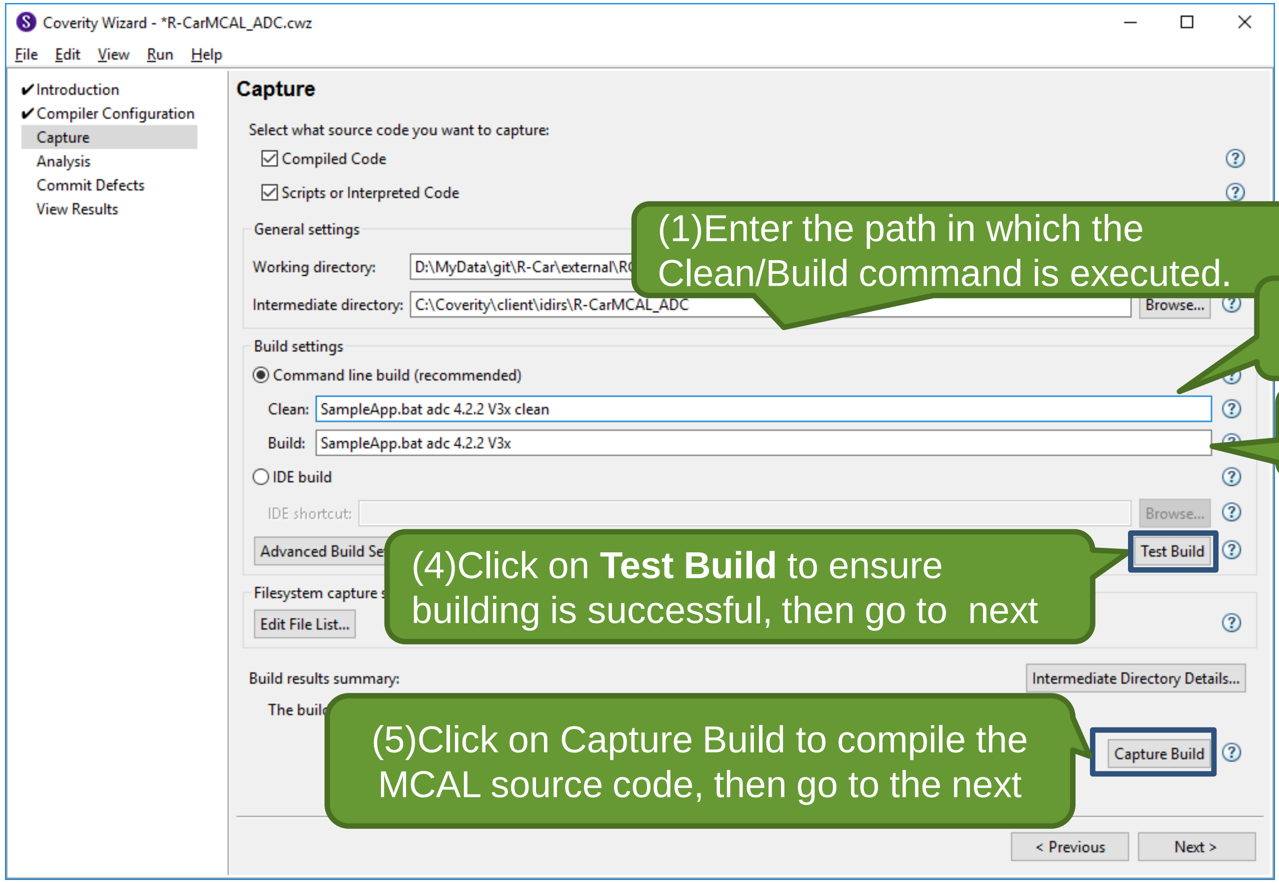
```
> gbuild -clean -top default.gpj

Cleaning D:\exercise\ghs\default.gpj
Done

> gbuild -all -top default.gpj

Cleaning D:\exercise\ghs\default.gpj
Building D:\exercise\ghs\default.gpj
Compiling hello.c because hello.o does not exist
Linking hello because it does not exist
Done
```

□ Show commands only



Coverity Wizard - *R-CarMCAL_ADC.cwz

Capture

Select what source code you want to capture:

☑ Compiled Code

☑ Scripts or Interpreted Code

General settings

Working directory: D:\MyData\git\R-Car\external\R

Intermediate directory: C:\Coverity\client\idirs\R-CarMCAL_ADC

Build settings

● Command line build (recommended)

Clean: SampleApp.bat adc 4.2.2 V3x clean

Build: SampleApp.bat adc 4.2.2 V3x

○ IDE build

IDE shortcut:

Filesystem capture

Build results summary:

The buil

(1)Enter the path in which the Clean/Build command is executed.

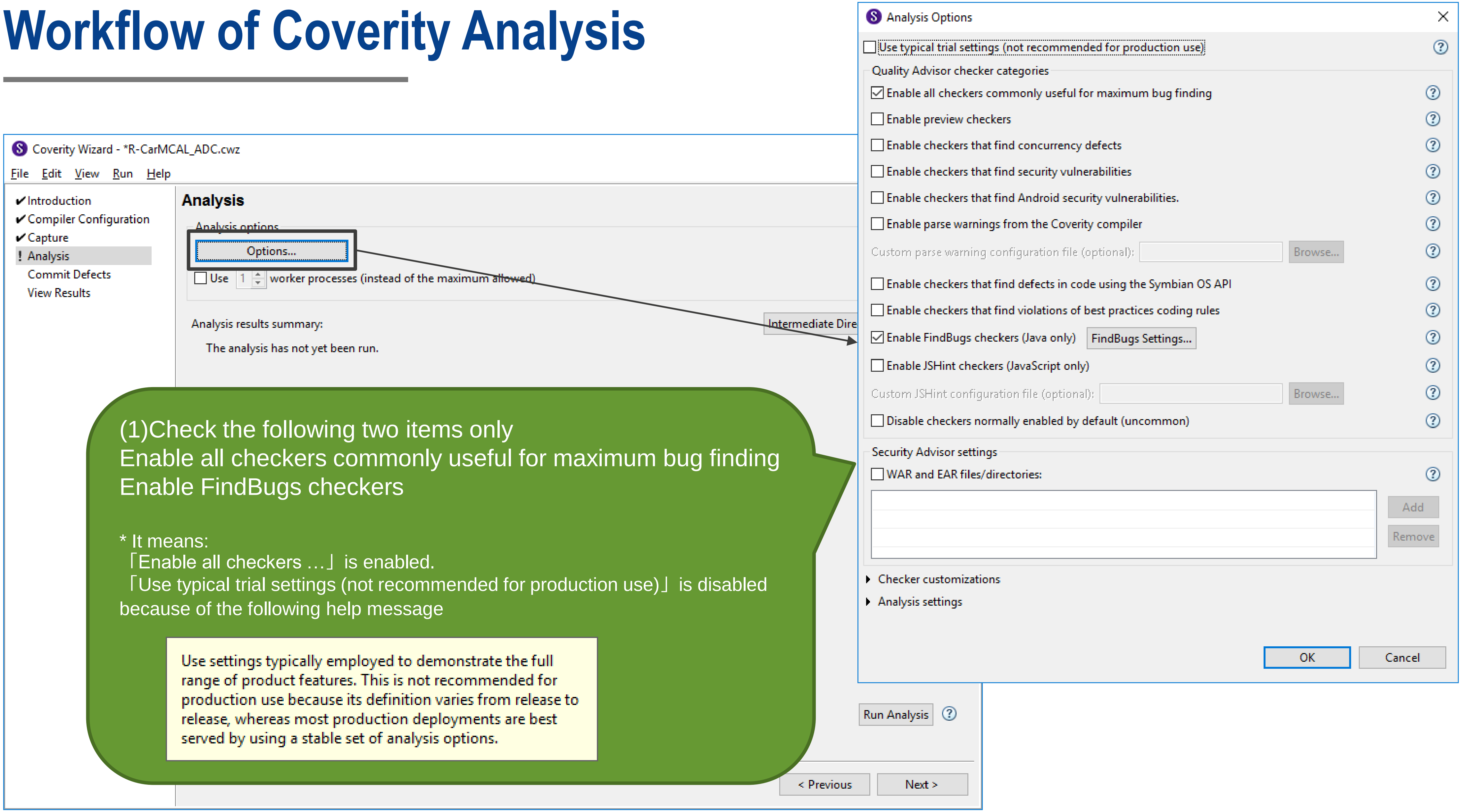
(2)Enter the command to clean all built items

(3)Enter the command to build the source codes

(4)Click on **Test Build** to ensure building is successful, then go to next

(5)Click on Capture Build to compile the MCAL source code, then go to the next

Workflow of Coverity Analysis



Analysis

Analysis options

Options...

□ Use 1 worker processes (instead of the maximum allowed)

Analysis results summary:

The analysis has not yet been run.

Analysis Options

□ Use typical trial settings (not recommended for production use)

Quality Advisor checker categories

☑ Enable all checkers commonly useful for maximum bug finding

□ Enable preview checkers

□ Enable checkers that find concurrency defects

□ Enable checkers that find security vulnerabilities

□ Enable checkers that find Android security vulnerabilities.

□ Enable parse warnings from the Coverity compiler

Custom parse warning configuration file (optional): Browse...

□ Enable checkers that find defects in code using the Symbian OS API

□ Enable checkers that find violations of best practices coding rules

☑ Enable FindBugs checkers (Java only) FindBugs Settings...

□ Enable JSHint checkers (JavaScript only)

Custom JSHint configuration file (optional): Browse...

□ Disable checkers normally enabled by default (uncommon)

Security Advisor settings

□ WAR and EAR files/directories:

Add

Remove

▸ Checker customizations

▸ Analysis settings

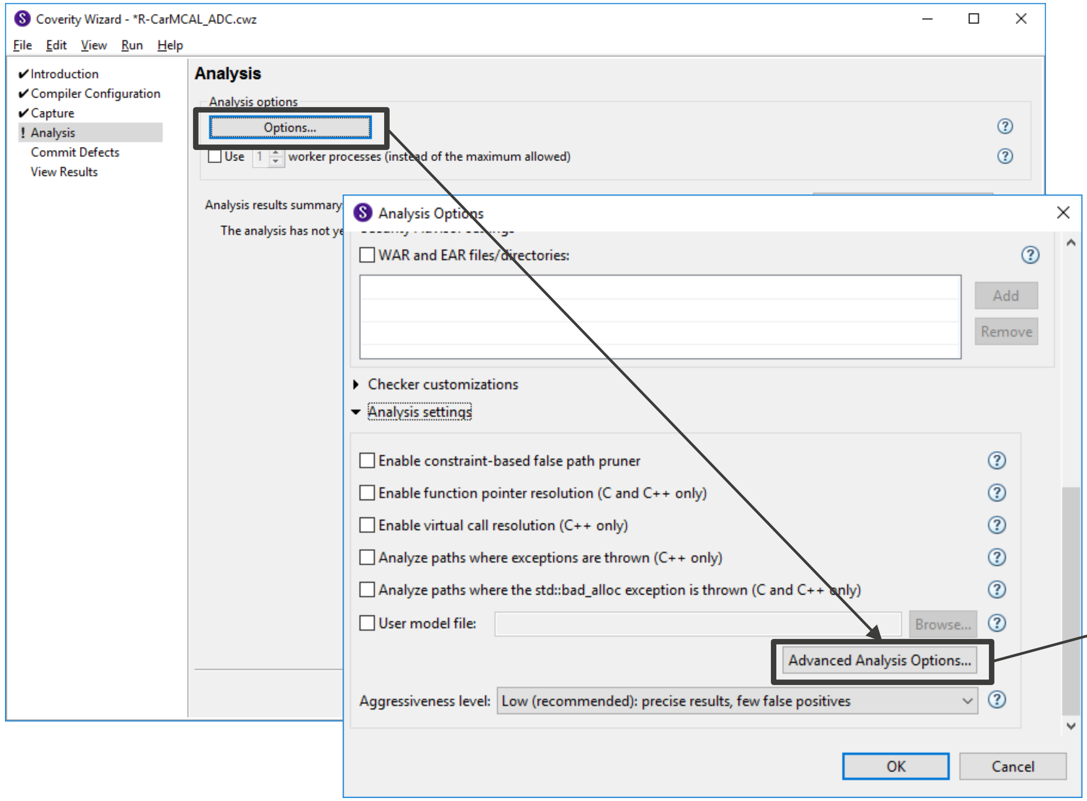
OK Cancel

(1)Check the following two items only
Enable all checkers commonly useful for maximum bug finding
Enable FindBugs checkers

* It means:
「Enable all checkers …」 is enabled.
「Use typical trial settings (not recommended for production use)」 is disabled because of the following help message

> Use settings typically employed to demonstrate the full range of product features. This is not recommended for production use because its definition varies from release to release, whereas most production deployments are best served by using a stable set of analysis options.

Workflow of Coverity Analysis



Coverity Wizard - *R-CarMCAL_ADC.cwz

Analysis

Analysis options

□ Use 1 worker processes (instead of the maximum allowed)

Analysis results summary

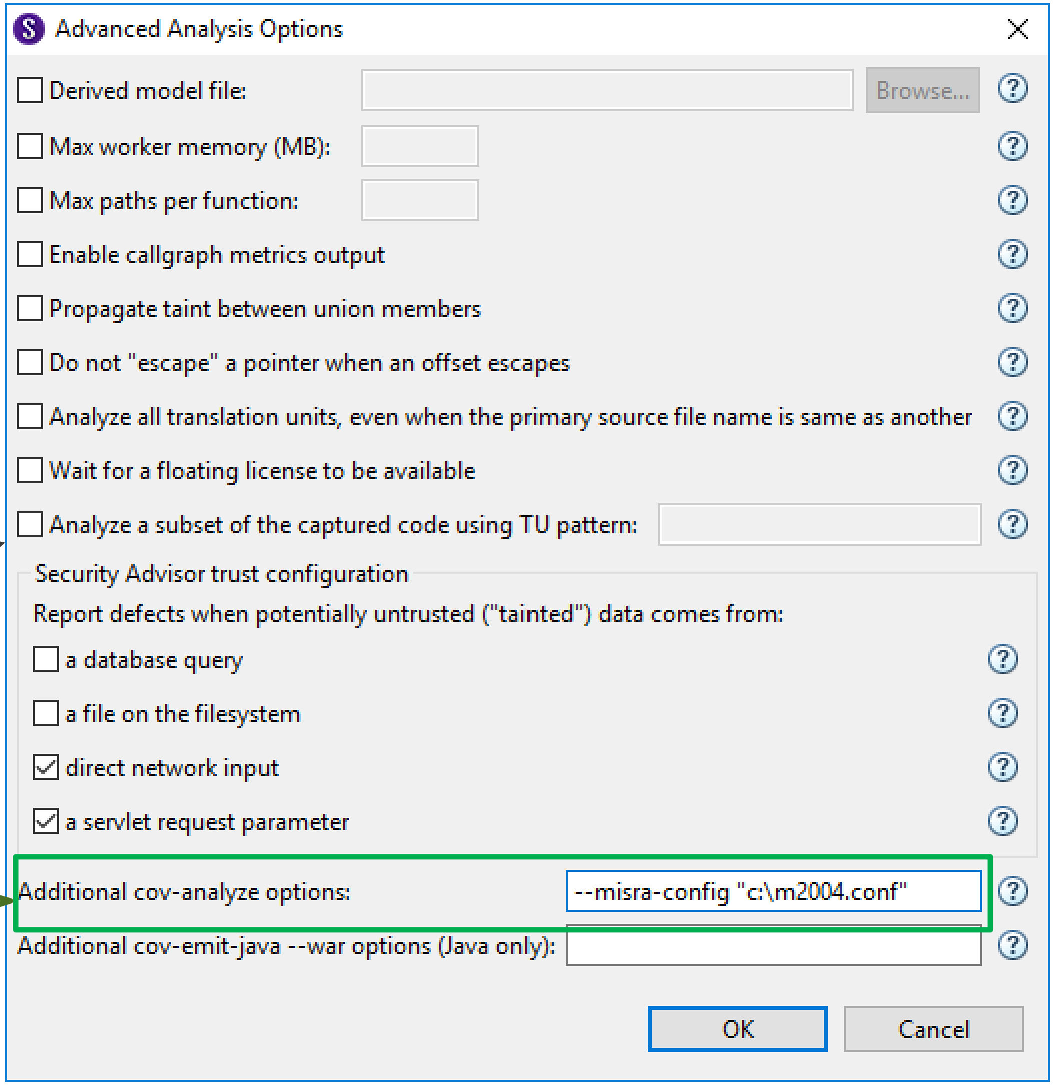
The analysis has not ye

Analysis Options

□ WAR and EAR files/directories:

- □ Enable constraint-based false path pruner
- □ Enable function pointer resolution (C and C++ only)
- □ Enable virtual call resolution (C++ only)
- □ Analyze paths where exceptions are thrown (C++ only)
- □ Analyze paths where the std::bad_alloc exception is thrown (C and C++ only)
- □ User model file:

Aggressiveness level: Low (recommended): precise results, few false positives



Advanced Analysis Options

- □ Derived model file:
- □ Max worker memory (MB):
- □ Max paths per function:
- □ Enable callgraph metrics output
- □ Propagate taint between union members
- □ Do not "escape" a pointer when an offset escapes
- □ Analyze all translation units, even when the primary source file name is same as another
- □ Wait for a floating license to be available
- □ Analyze a subset of the captured code using TU pattern:

Security Advisor trust configuration

Report defects when potentially untrusted ("tainted") data comes from:

- □ a database query
- □ a file on the filesystem
- ☑ direct network input
- ☑ a servlet request parameter

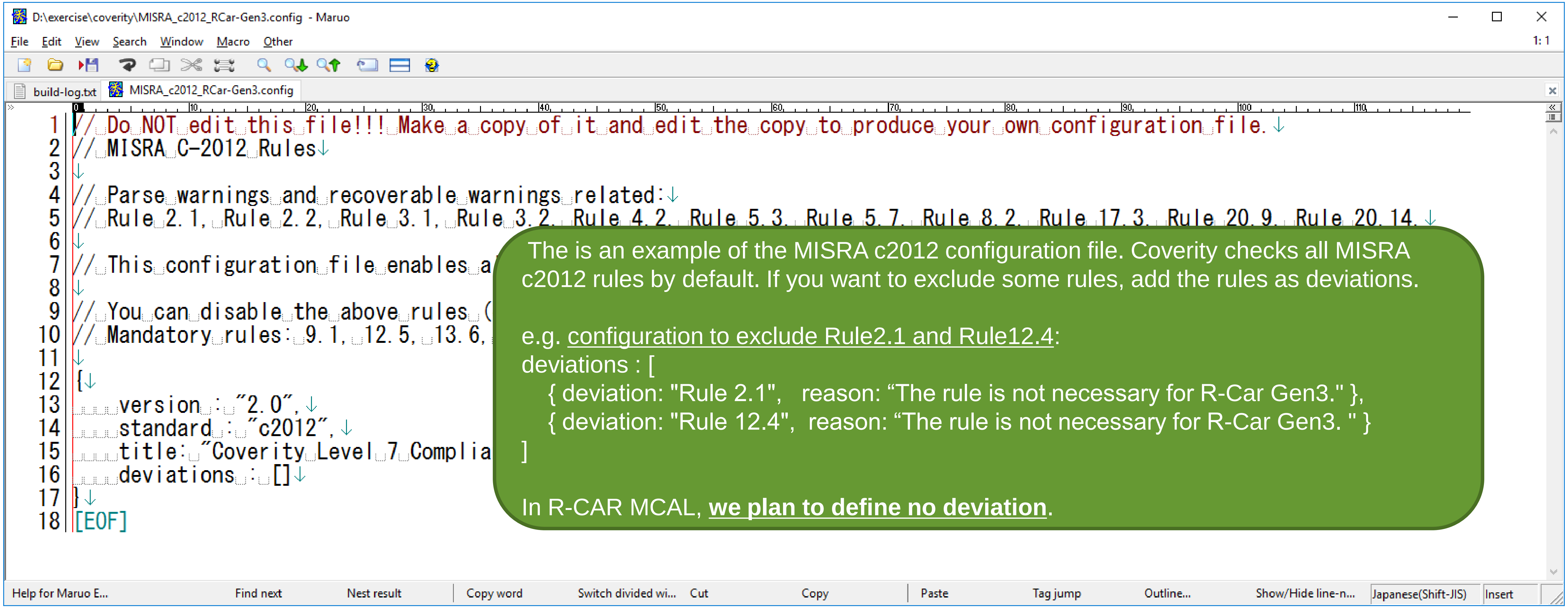
Additional cov-analyze options: --misra-config "c:\m2004.conf"

Additional cov-emit-java --war options (Java only):

Enter –misra-config “*<location of the misra configuration file>*”

For details about the MISRA configuration file, see the next slide.

Workflow of Coverity Analysis



```
// Do NOT edit this file!!! Make a copy of it and edit the copy to produce your own configuration file.
// MISRA C-2012 Rules

// Parse warnings and recoverable warnings related:
// Rule 2.1, Rule 2.2, Rule 3.1, Rule 3.2, Rule 4.2, Rule 5.3, Rule 5.7, Rule 8.2, Rule 17.3, Rule 20.9, Rule 20.14.

// This configuration file enables a

// You can disable the above rules (
// Mandatory rules: 9.1, 12.5, 13.6,

{
    version : "2.0",
    standard : "c2012",
    title: "Coverity Level 7 Complia
    deviations : []
}
[EOF]
```

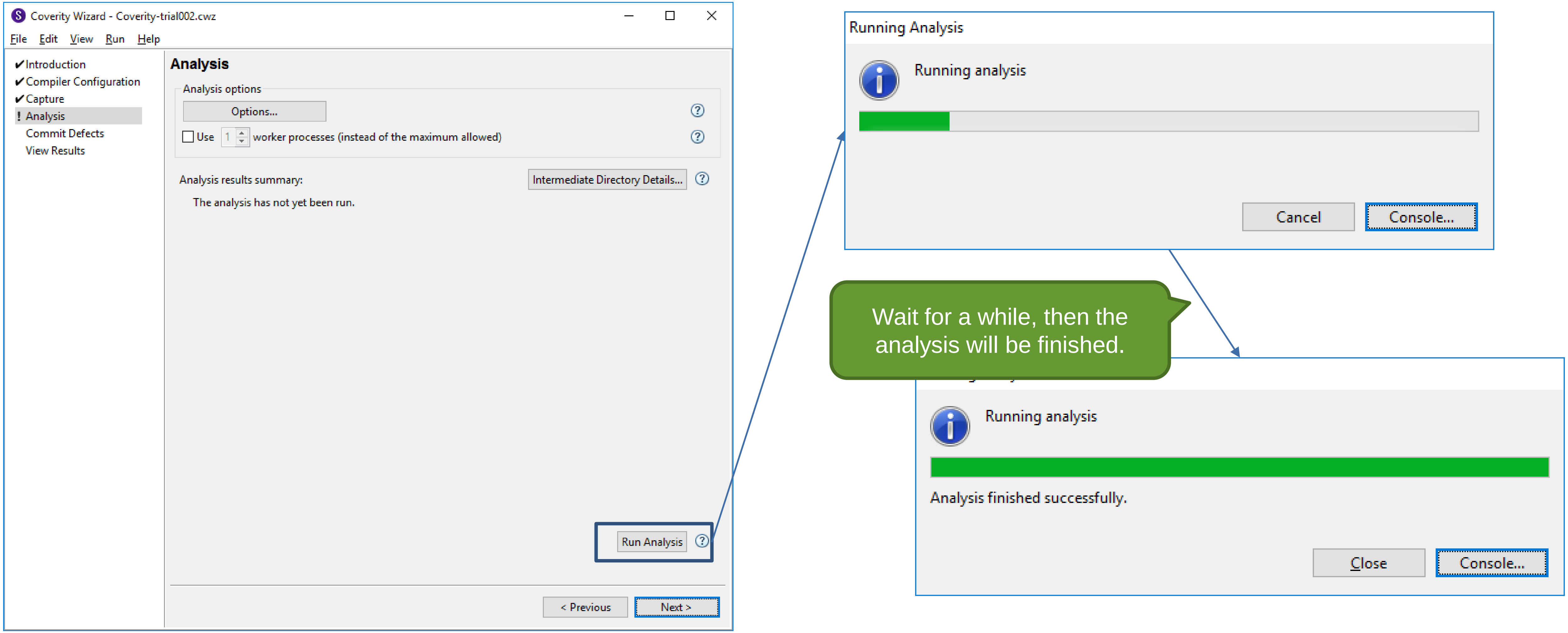
The is an example of the MISRA c2012 configuration file. Coverity checks all MISRA c2012 rules by default. If you want to exclude some rules, add the rules as deviations.

e.g. configuration to exclude Rule2.1 and Rule12.4:

```
deviations : [
    { deviation: "Rule 2.1",   reason: “The rule is not necessary for R-Car Gen3." },
    { deviation: "Rule 12.4",  reason: “The rule is not necessary for R-Car Gen3. " }
]
```

In R-CAR MCAL, **we plan to define no deviation**.

Workflow of Coverity Analysis



Coverity Wizard - Coverity-trial002.cwz

Analysis

Analysis options

□ Use 1 worker processes (instead of the maximum allowed)

Analysis results summary:

The analysis has not yet been run.

Running Analysis

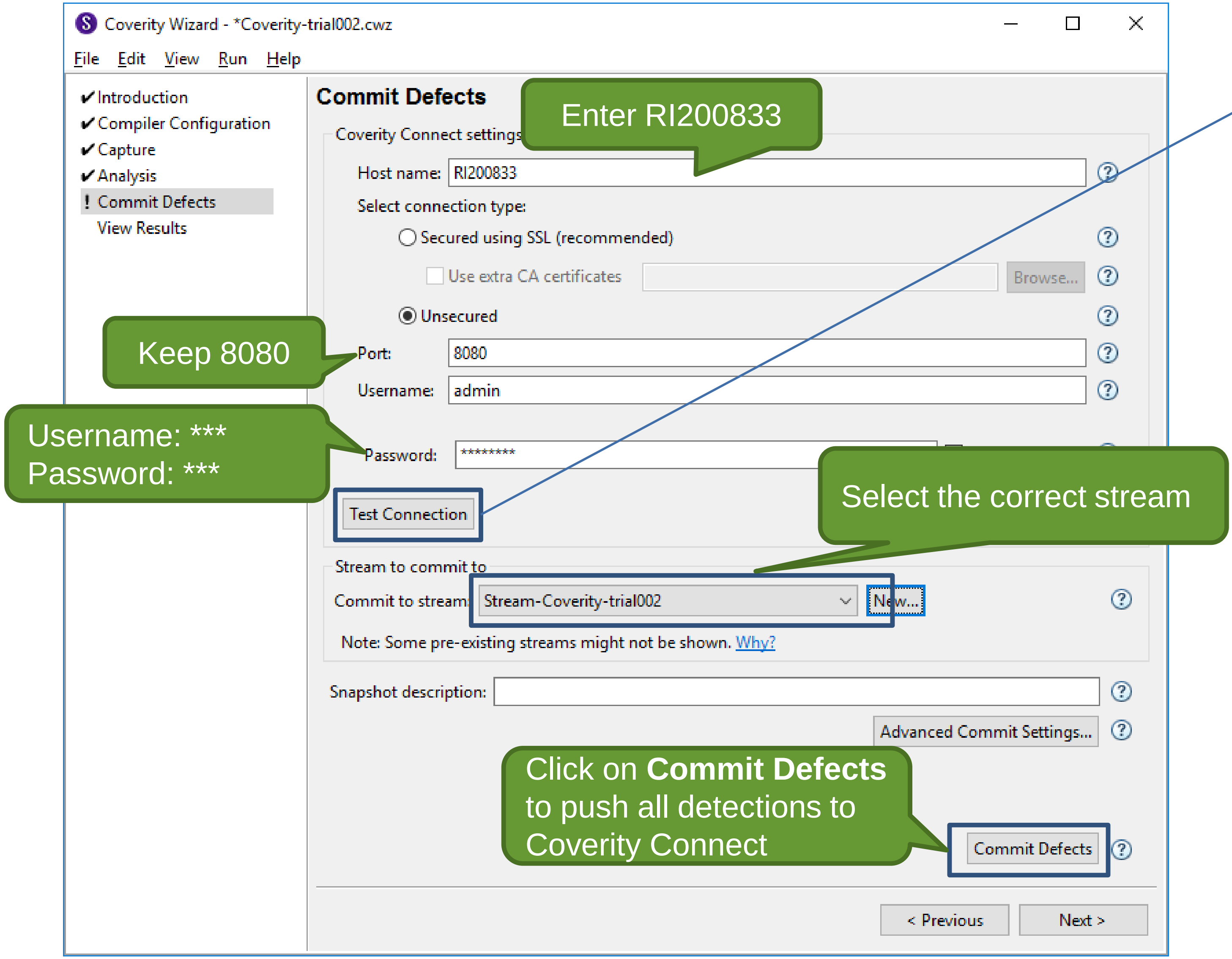
Running analysis

Wait for a while, then the analysis will be finished.

Running analysis

Analysis finished successfully.

Workflow of Coverity Analysis



Coverity Wizard - *Coverity-trial002.cwz

Commit Defects

Coverity Connect settings

Host name: RI200833

Select connection type:

○ Secured using SSL (recommended)

□ Use extra CA certificates

● Unsecured

Port: 8080

Username: admin

Password: ********

Stream to commit to

Commit to stream: Stream-Coverity-trial002

Note: Some pre-existing streams might not be shown. Why?

Snapshot description:

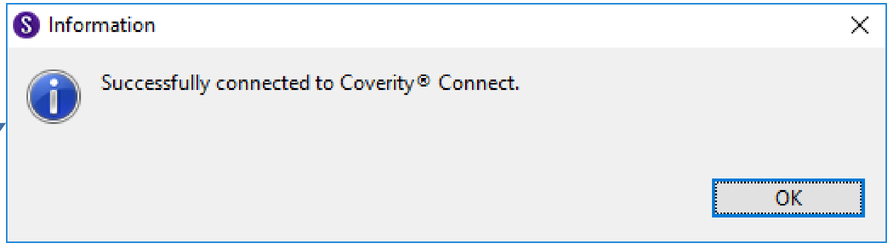
Enter RI200833

Keep 8080

Username: ***
Password: ***

Select the correct stream

Click on **Commit Defects** to push all detections to Coverity Connect

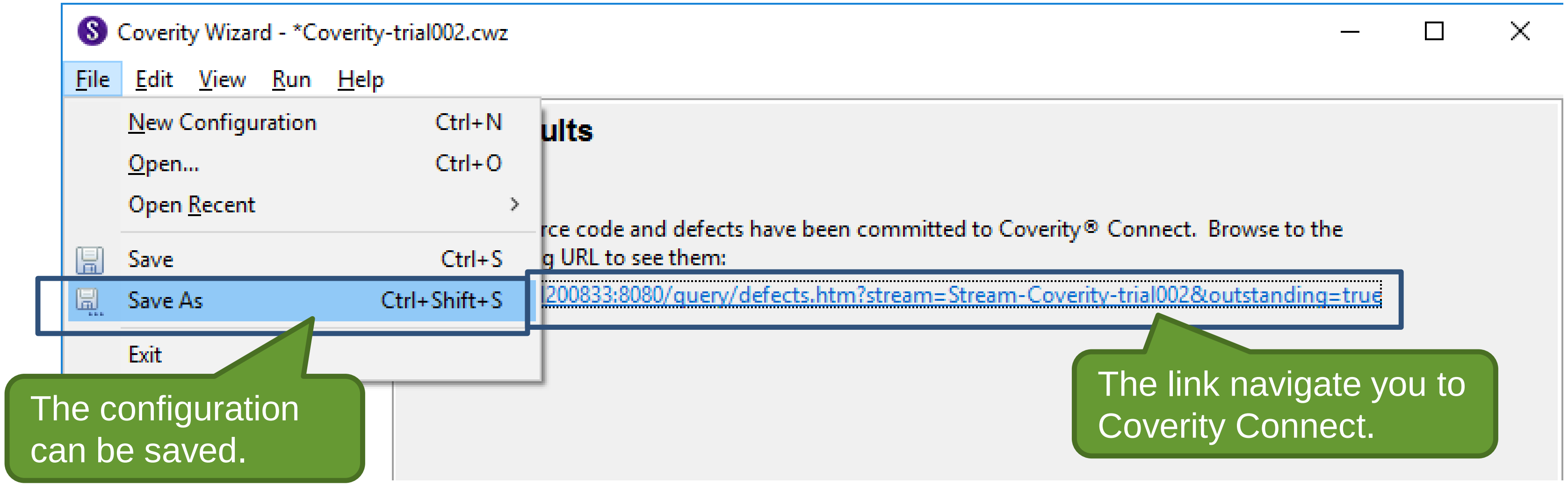


Information

Successfully connected to Coverity® Connect.

Project	Stream	Module
MCAL	MCAL_ADC	V3x MCAL ADC module
	MCAL_CAN	V3x MCAL CAN module
	MCAL_DIO	V3x MCAL DIO module
	MCAL_ETH	V3x MCAL ETH module
	MCAL_FLS	V3x MCAL FLS module
	MCAL_GPT	V3x MCAL GPT module
	MCAL_MCU	V3x MCAL MCU module
	MCAL_PORT	V3x MCAL Port module
	MCAL_SPI	V3x MCAL SPI module
	MCAL_WDG	V3x MCAL WDG module
	MCAL_CDDCMT1	V3x CDD CMT1 module
	MCAL_CDDBus Monitor	V3x CDD Bus Monitor module
	MCAL_CDDCRC	V3x CDD CRS module

Workflow of Coverity Analysis



rce code and defects have been committed to Coverity® Connect. Browse to the g URL to see them:

200833:8080/query/defects.htm?stream=Stream-Coverity-trial002&outstanding=true

The configuration can be saved.

The link navigate you to Coverity Connect.

Coverity Analysis commands

The execution of Coverity Analysis from the command line is simple and easy.

1) Build the MCAL source code by Coverity own compiler
2) Analyse the MCAL source code to detect static issues
3) Push the analysis results to Coverity Connect

1)
```
cov-build --dir tmp_dir --encoding UTF-8 SampleApp.bat MSN 4.2.2 V3x arm
```

2)
```
cov-analyze --dir tmp_dir --force --misra-config misra_conf --all --webapp-security --strip-path %USER_DIR_BUILD%
```

3)
```
cov-commit-defects --dir tmp_dir --host 10.166.12.40 --https-port 8443 --user user_name --password password --stream stream_name --description “desc"
```

where:

tmp_dir is a temporary folder to store intermediate files generated by Coverity Analysis
MSN is a module name of R-CAR MCAL/CDD
misra_conf is the location of a configuration file for MISRA-C analysis rules
user_name is a user name of Coverity Analysis
password is a password for ***user_name***
stream_name is a stream name
desc is a comment on a push