

Gradle and build systems for C language

28.4.2014

FI MUNI, Brno

Juraj Michálek

<http://georgik.sinusgear.com>

Grab the source code

<https://github.com/georgik/fimuni-c-cpp-examples.git>



Who am I?



SDL

Gradle

CMake

Nuget

tiobe.com - programming lang. index

Apr 2014	Apr 2013	Change	Programming Language	Ratings	Change
1	1		C	17.631%	-0.23%
2	2		Java	17.348%	-0.33%
3	4	▲	Objective-C	12.875%	+3.28%
4	3	▼	C++	6.137%	-3.58%
5	5		C#	4.820%	-1.33%
6	7	▲	(Visual) Basic	3.441%	-1.26%
7	6	▼	PHP	2.773%	-2.65%
8	8		Python	1.993%	-2.45%
9	11	▲	JavaScript	1.750%	+0.24%
10	12	▲	Visual Basic .NET	1.748%	+0.65%
11	10	▼	Ruby	1.745%	-0.23%

**Let's start with something
cool**

The Battle for Wesnoth



SDL

Simple Directmedia Layer



Multiplatform

SDL officially supports

Windows, Mac OS X, Linux, iOS, and Android.

SDL versions

1.2 stable - rock solid

2.x development - new features

Some basic concepts

SDL_init(flags)

SDL_INIT_TIMER - The timer subsystem

SDL_INIT_AUDIO - The audio subsystem

SDL_INIT_VIDEO - The video subsystem

SDL_INIT_CDROM - The cdrom subsystem

SDL_INIT_JOYSTICK - The joystick subsystem

SDL_INIT EVERYTHING - All of the above

SDL_INIT_NOPARACHUTE - Prevents SDL from catching fatal signals

SDL_INIT_EVENTTHREAD - Runs the event manager in a separate thread

Quit application

`SDL_quit()`

Window

```
SDL_CreateWindow("Hello World!", 100, 100,  
640, 480, SDL_WINDOW_SHOWN);
```

Load bitmap

```
SDL_Surface *bmp = NULL;  
bmp = SDL_LoadBMP("./smajlik.bmp");
```

Visual data

SDL_Renderer

SDL_Texture

Keyboard

`SDL_PollEvent(SDL_Event *event)`

`event.key.keysym.sym`

Timer

```
SDL_TimerID SDL_AddTimer(  
    Uint32          interval,  
    SDL_TimerCallback callback,  
    void*           param)
```

Mouse

```
SDL_GetMouseState(*x, *y);
```

Text

Not implemented

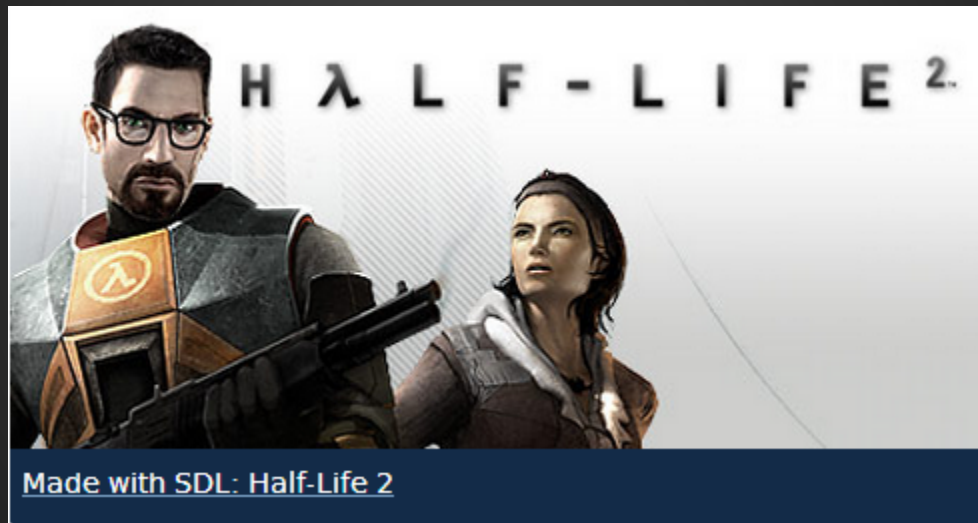


Extensions

extension for many languages:

C++, Java, Lua, Perl, PHP, Python, Ruby

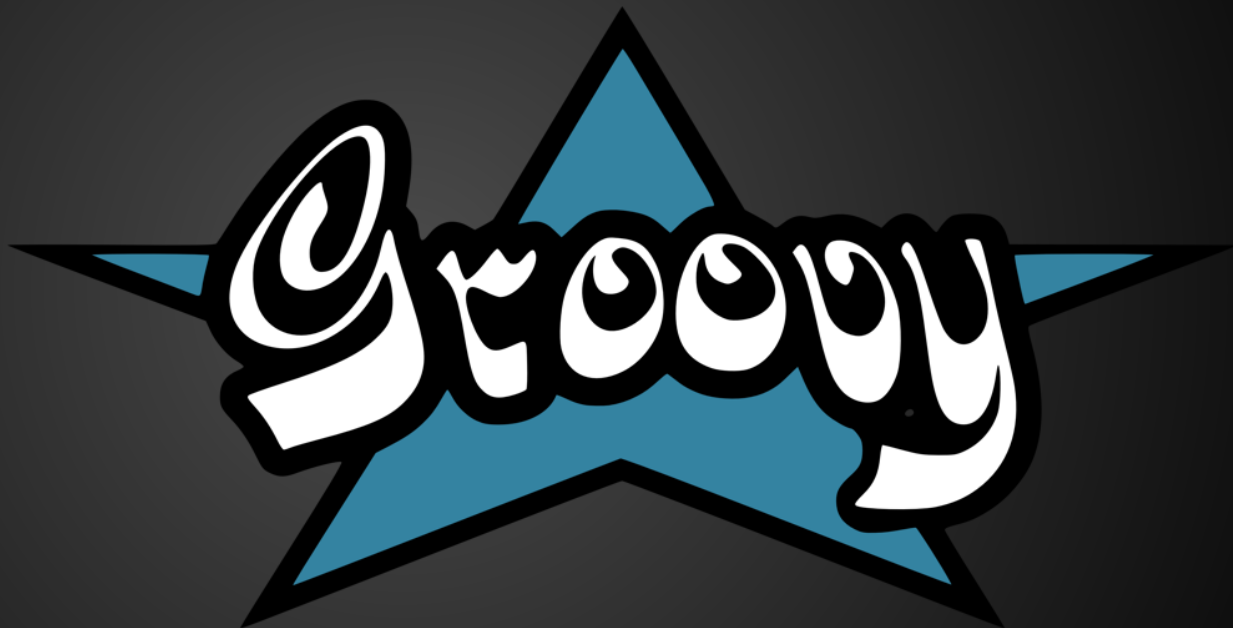
Made with SDL



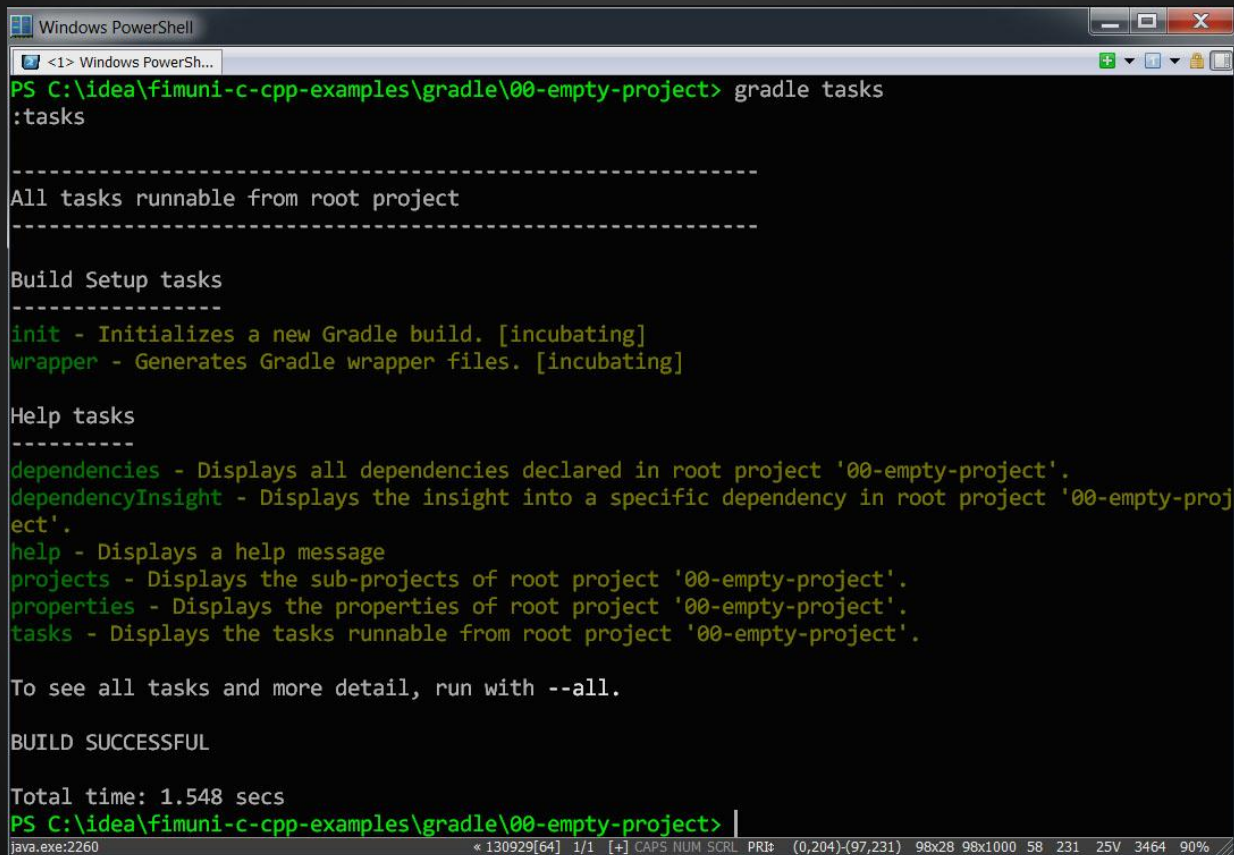
Automation Evolved



Domain Specific Language



gradle tasks



```
Windows PowerShell
<1> Windows PowerSh...
PS C:\idea\fmuni-c-cpp-examples\gradle\00-empty-project> gradle tasks
:tasks

-----
All tasks runnable from root project
-----

Build Setup tasks
-----
init - Initializes a new Gradle build. [incubating]
wrapper - Generates Gradle wrapper files. [incubating]

Help tasks
-----
dependencies - Displays all dependencies declared in root project '00-empty-project'.
dependencyInsight - Displays the insight into a specific dependency in root project '00-empty-project'.
help - Displays a help message
projects - Displays the sub-projects of root project '00-empty-project'.
properties - Displays the properties of root project '00-empty-project'.
tasks - Displays the tasks runnable from root project '00-empty-project'.

To see all tasks and more detail, run with --all.

BUILD SUCCESSFUL

Total time: 1.548 secs
PS C:\idea\fmuni-c-cpp-examples\gradle\00-empty-project> |
java.exe:2260 * 130929[64] 1/1 [+] CAPS NUM SCRL PRI# (0,204)-(97,231) 98x28 98x1000 58 231 25V 3464 90%
```

build.gradle

```
<1> build.gradle + ...  
task hello {  
    description = "Say Hello :-)"  
    println "Hello FI MUNI!"  
}
```

gradle tasks

Other tasks

hello - Say Hello :-)

gradle hello

```
PS C:\idea\fimuni-c-cpp-examples\gradle\01-hello-task> gradle hello  
Hello FI MUNI!  
:hello UP-TO-DATE  
  
BUILD SUCCESSFUL  
  
Total time: 1.53 secs
```

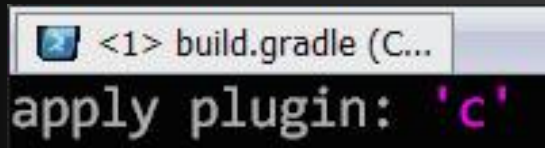
Plugin system

- focussed functionality is added by plugins
- reuse patterns and practices
- avoiding boilerplate build code



C language plugin

build.gradle



<1> build.gradle (C...
apply plugin: 'c'

gradle tasks

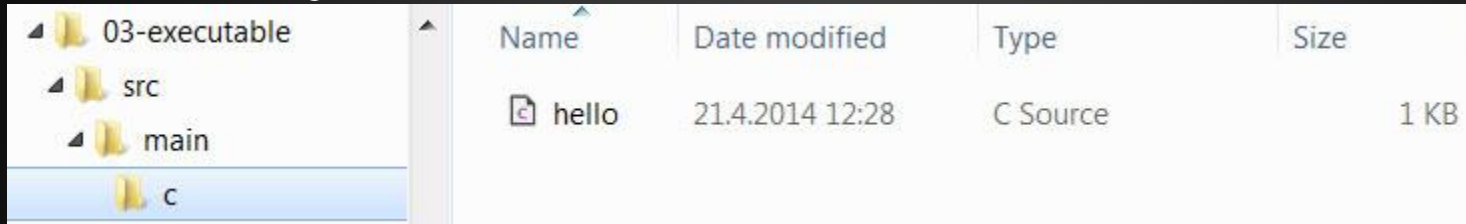
Build tasks

assemble - Assembles the outputs of this project.

clean - Deletes the build directory.

C build executable

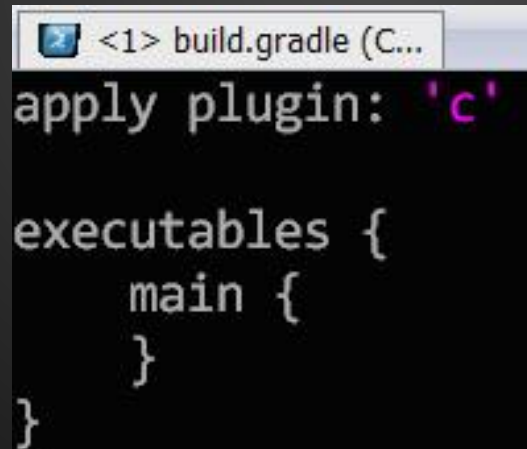
Directory structure



A screenshot of a file explorer window showing the directory structure of a project. The left pane shows a tree view with folders: 03-executable, src, main, and c. The 'c' folder is selected. The right pane shows a table of files within the 'c' folder.

Name	Date modified	Type	Size
hello	21.4.2014 12:28	C Source	1 KB

build.gradle



A screenshot of a code editor showing the content of a build.gradle file. The file is named '<1> build.gradle (C...'. The code defines a Gradle plugin named 'c' and its executables.

```
<1> build.gradle (C...
apply plugin: 'c'

executables {
    main {
    }
}
```

gradle tasks

Build tasks

assemble - Assembles the outputs of this project.

clean - Deletes the build directory.

installMainExecutable - Installs a development image of executable 'main:executable'

mainExecutable - Assembles executable 'main:executable'.

gradle mE

```
PS C:\idea\fimuni-c-cpp-examples\gradle\03-executable> gradle mE
```

```
:mainCExtractHeaders
```

```
:compileMainExecutableMainC
```

```
hello.c
```

```
:linkMainExecutable
```

```
:mainExecutable
```

```
BUILD SUCCESSFUL
```

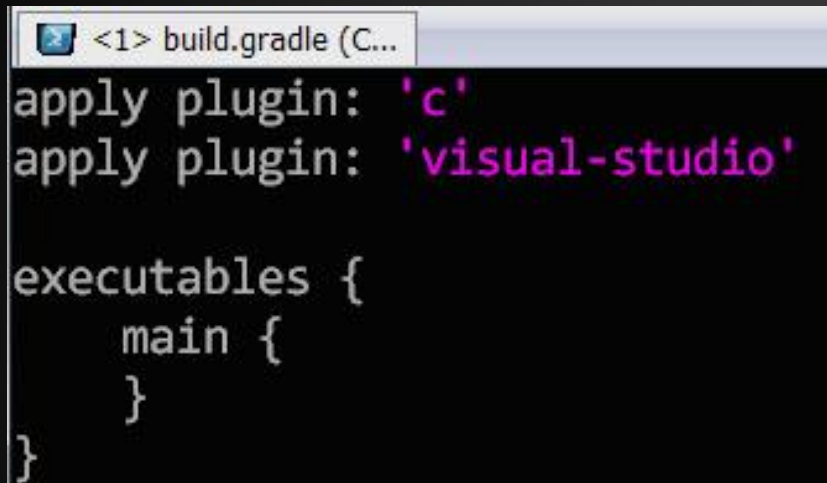
Windows

```
PS C:\idea\fimuni-c-cpp-examples\gradle\03-executable> cd .\build\binaries\mainExecutable
PS C:\idea\fimuni-c-cpp-examples\gradle\03-executable\build\binaries\mainExecutable> .\main.exe
Hello FI MUNI!
PS C:\idea\fimuni-c-cpp-examples\gradle\03-executable\build\binaries\mainExecutable> |
```

Linux

```
georgik@evicka4: ~/idea/fimuni-c-cpp-examples/gradle/03-executable/build/binaries/mainEx... x
File Edit View Search Terminal Help
┌(georgik@evicka4:pts/0)──(../fimuni-c-cpp-examples/gradle/03-executable)┐
└(:19:30:%)── cd build/binaries/mainExecutable ──(Mon, Apr21)┘
┌(georgik@evicka4:pts/0)──(../03-executable/build/binaries/mainExecutable)┐
└(:19:30:%)── ./main ──(Mon, Apr21)┘
Hello FI MUNI!
┌(georgik@evicka4:pts/0)──(../03-executable/build/binaries/mainExecutable)┐
└(:19:30:%)── ──(Mon, Apr21)┘
```

Visual Studio plugin

A screenshot of a code editor window titled "<1> build.gradle (C...)". The code is written in a dark-themed editor with syntax highlighting. It shows the application of two Gradle plugins: 'c' and 'visual-studio'. Below the plugin applications, there is a configuration block for 'executables' containing a 'main' executable.

```
<1> build.gradle (C...  
apply plugin: 'c'  
apply plugin: 'visual-studio'  
  
executables {  
    main {  
    }  
}
```

IDE tasks

cleanVisualStudio - Removes all generated Visual Studio project and solution files
mainVisualStudio - Generates the Visual Studio solution for executable 'main'.



mainExe - Microsoft Visual Studio



5

Quick Launch (Ctrl+Q)

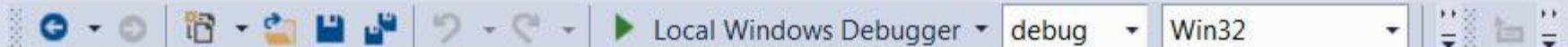


FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST ANALYZE

Juraj Michalek



WINDOW HELP



Local Windows Debugger

debug

Win32

Server Explorer
Toolbox

hello.c

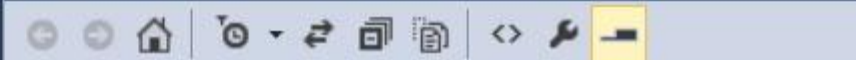
mainExe

(Global Scope)

```
#include <stdio.h>
```

```
int main(int argc, char** argv){  
    printf("Hello FI MUNI!\n");  
    return 0;  
}
```

Solution Explorer



Search Solution Explorer (Ctrl+;)

Solution 'mainExe' (1 project)

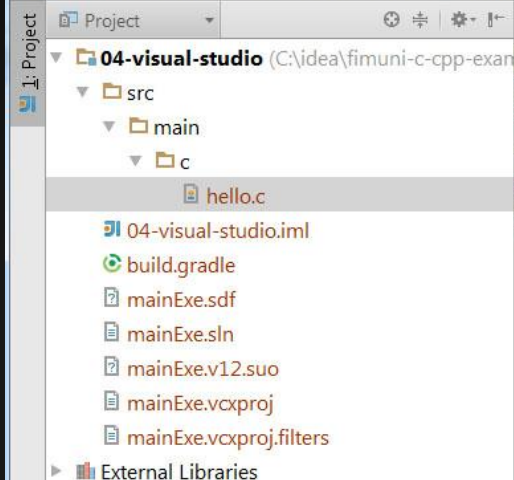
mainExe (Visual Studio 2010)

- External Dependencies
- Header Files
- Resource Files
- Source Files
 - build.gradle
 - hello.c



© 2000–2014 JetBrains s.r.o. All rights reserved.





```
1 #include <stdio.h>
2
3 int main(int argc, char** argv){
4     printf("Hello FI MUNI!\n");
5     return 0;
6 }
7
```

Gradle tasks

Recent tasks

- O4-visual-studio [mainExecuta...
- O4-visual-studio [clean]
- O4-visual-studio [cleanVisualS...

All tasks

- O4-visual-studio
 - assemble
 - clean
 - cleanVisualStudio
 - compileMainExecutableMa
 - installMainExecutable
 - linkMainExecutable
 - mainCExtractHeaders
 - mainExeVisualStudio
 - mainExeVisualStudioFilter:
 - mainExeVisualStudioProje
 - mainExeVisualStudioSolut
 - mainExecutable**
 - mainVisualStudio
- O4-visual-studio

Run O4-visual-studio [mainExecutable]

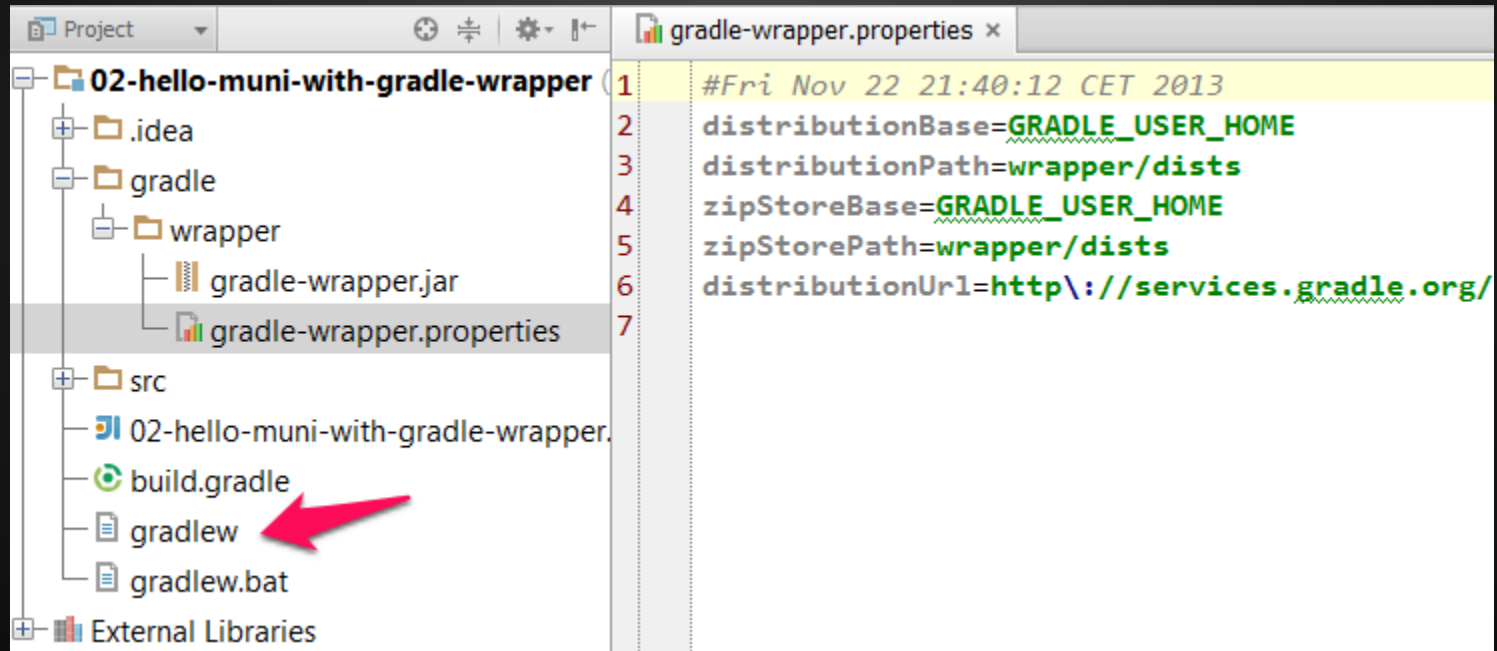
20:11:04: Executing external task 'mainExecut
:mainCExtractHeaders UP-TO-DATE
:compileMainExecutableMainC

Event Log

20:08:16 Gradle 'O4-visu
Cause: org.grad
Gradle settings

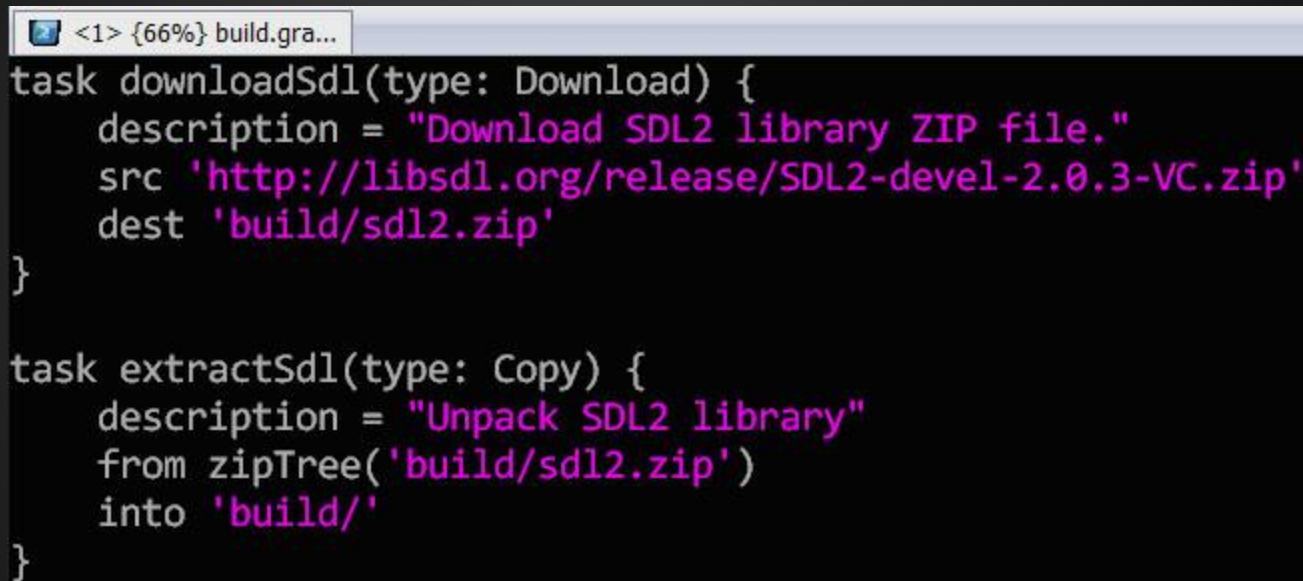
Gradle wrapper

Download and prepare infrastructure





Download SDL2 (for VS)



The screenshot shows a window titled "<1> {66%} build.gra...". It contains two task definitions in a build script. The first task, `downloadSdl`, is of type `Download` and describes downloading the SDL2 library ZIP file from a specific URL to a local build directory. The second task, `extractSdl`, is of type `Copy` and describes unpacking the downloaded ZIP file into the build directory.

```
task downloadSdl(type: Download) {  
    description = "Download SDL2 library ZIP file."  
    src 'http://libsdl.org/release/SDL2-devel-2.0.3-VC.zip'  
    dest 'build/sdl2.zip'  
}  
  
task extractSdl(type: Copy) {  
    description = "Unpack SDL2 library"  
    from zipTree('build/sdl2.zip')  
    into 'build/'  
}
```

```
PS C:\idea\fimuni-c-cpp-examples\sdl> gradle prepare
:downloadSdl
Download http://libsdl.org/release/SDL2-devel-2.0.3-VC.zip
:extractSdl
:copySdlInclude
:copySdlLib
:prepare

BUILD SUCCESSFUL

Total time: 8.01 secs
```

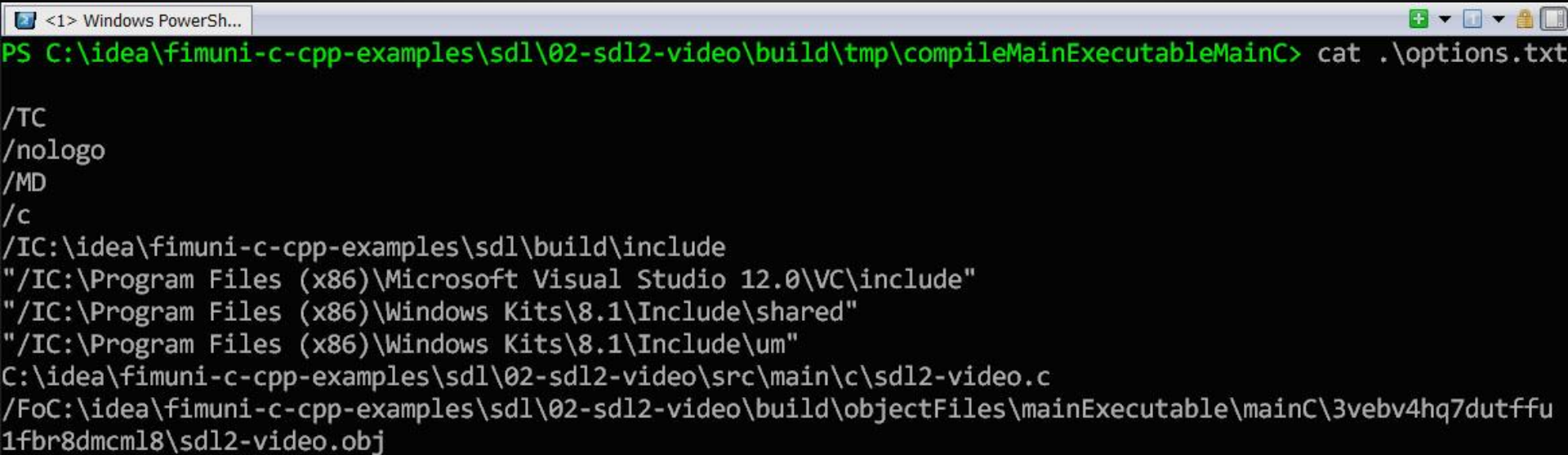
Sources

```
sources {  
  main {  
    c {  
      source {  
        // Include just source, avoid including *.swp  
        include "**/*.c"  
      }  
      exportedHeaders {  
        srcDir "../build/include"  
      }  
    }  
  }  
}
```

Toolchains

```
executables {  
    main {  
        binaries.all {  
            if (toolChain in VisualCpp) {  
                cCompiler.args "/MD"  
                linker.args "/SUBSYSTEM:CONSOLE", "/LIBPATH:../build/lib/",  
                    "SDL2main.lib", "SDL2.lib"  
            } else if (toolChain in Gcc) {  
                linker.args "-lSDL2"  
            }  
        }  
    }  
}
```

Helper files for compilers



A screenshot of a Windows PowerShell terminal window. The title bar shows "<1> Windows PowerSh...". The command prompt is at the path `PS C:\idea\fimuni-c-cpp-examples\sdl\02-sdl2-video\build\tmp\compileMainExecutableMainC>`. The user has entered the command `cat .\options.txt`. The output of the command is displayed in green text, listing various compiler flags and paths for a C compiler.

```
PS C:\idea\fimuni-c-cpp-examples\sdl\02-sdl2-video\build\tmp\compileMainExecutableMainC> cat .\options.txt

/TC
/nologo
/MD
/c
/IC:\idea\fimuni-c-cpp-examples\sdl\build\include
"/IC:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\include"
"/IC:\Program Files (x86)\Windows Kits\8.1\Include\shared"
"/IC:\Program Files (x86)\Windows Kits\8.1\Include\um"
C:\idea\fimuni-c-cpp-examples\sdl\02-sdl2-video\src\main\c\sdl2-video.c
/FoC:\idea\fimuni-c-cpp-examples\sdl\02-sdl2-video\build\objectFiles\mainExecutable\mainC\3vebv4hq7dutfu
1fbr8dmcml8\sdl2-video.obj
```

Build Debian/Ubuntu packages



Gradle Debian Plugin

```
debian {  
    packagename = "hello-fimuni"  
    publications = ['fimuni']  
    controlDirectory = "$projectDir/src/main/resources/control"  
    changelogFile = "$projectDir/src/main/resources/txt/changelog.txt"  
  
    data {  
        def baseDir = "$buildDir/linux"  
        dir {  
            name = baseDir  
        }  
    }  
}
```

<https://github.com/gesellix/gradle-debian-plugin.git>

Further Gradle training



<https://github.com/ysoftdevs/gradle-training>

PyGame

Power of C and Power of Python

<http://www.pygame.org>



Kivy - crossplatform framework for NUI



SDL + PyGame

<http://kivy.org>

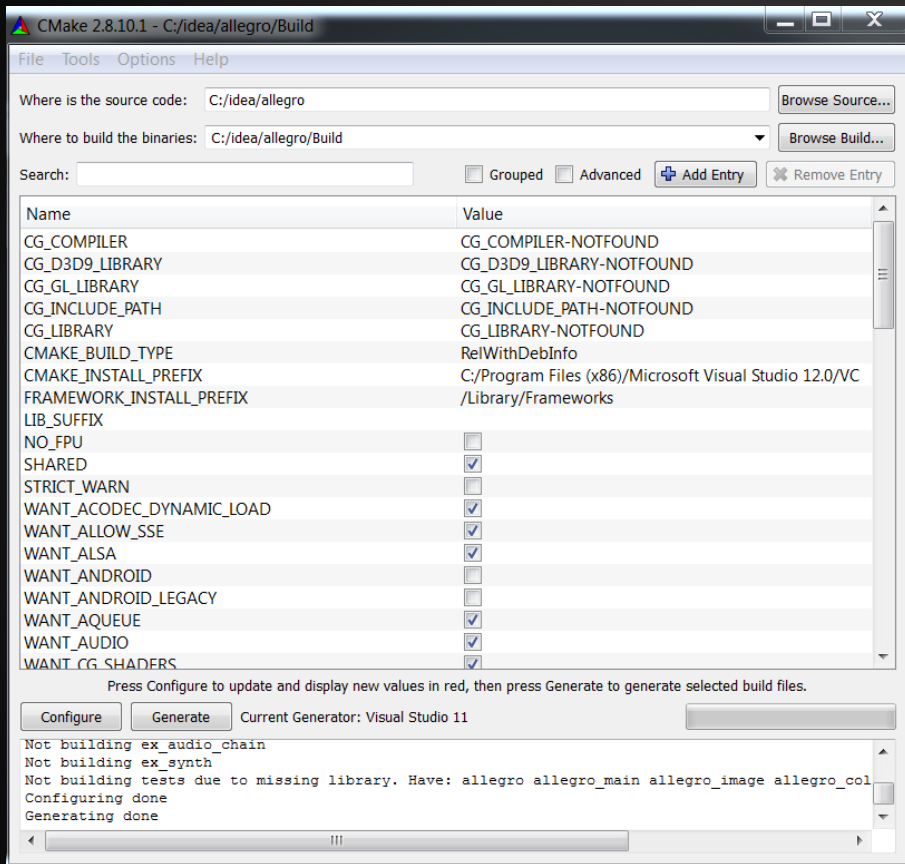




CMake

Allegro





Allegro 5.1

Win, Lin, Mac

iOS, Android

<http://alleg.sourceforge.net/a5docs/refman/>

Initialization

```
al_init();
```

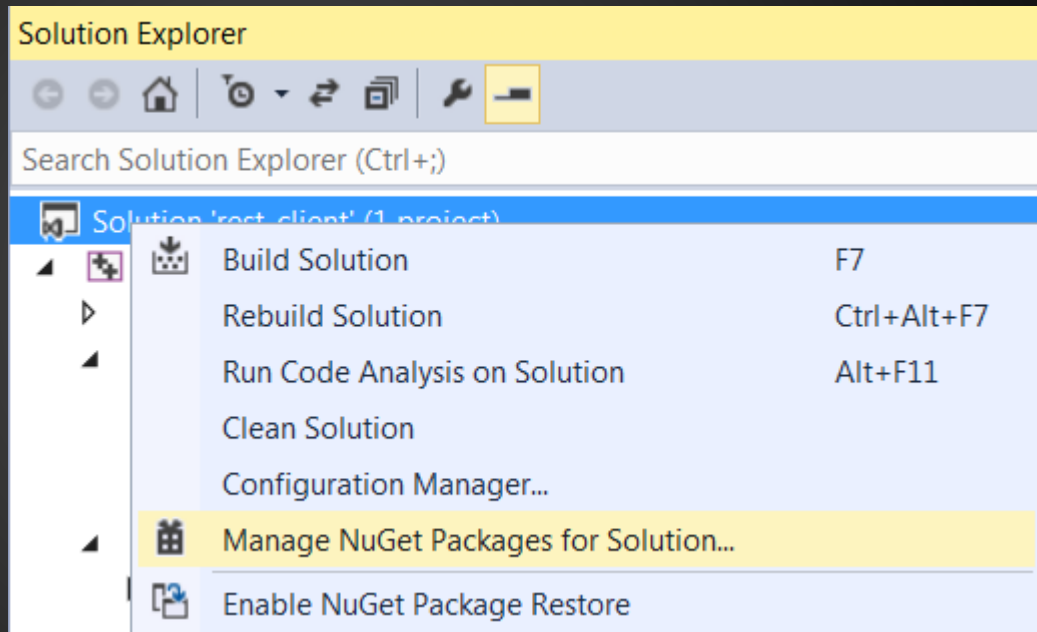
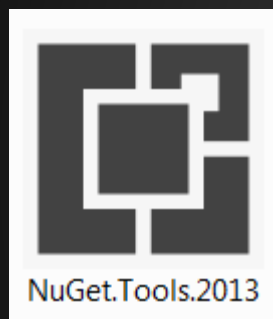
Graphic environment

```
al_create_display(int w, int h)
```



NuGet - <http://www.nuget.org>





▶ Installed packages

◀ Online

All

nuget.org

Microsoft and .NET

Search Results

▶ Updates

Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages.

Stable Only

Sort by: Relevance

sdl

**Simple DirectMedia Lay...**

Simple DirectMedia Layer is a cross-platform multi...

Install

**AntiXSS**

AntiXSS is an encoding library for .NET.

**SDL_image**

SDL_image loads images as SDL surfaces.

**DD4T Tridion 2013**

DD4T Providers for SDL Tridion 2013 GA

**DD4T Providers for Tridion 2011sp1**

Providers for SDL Tridion 2011 SP1

1

Created by: Sam Lantinga, SDL contributors

Id: [SDL](#)**Version:** 1.2.15.15**Last Published:** 5.7.2013**Downloads:** 314**License**[View License](#)

LGPL-2.1

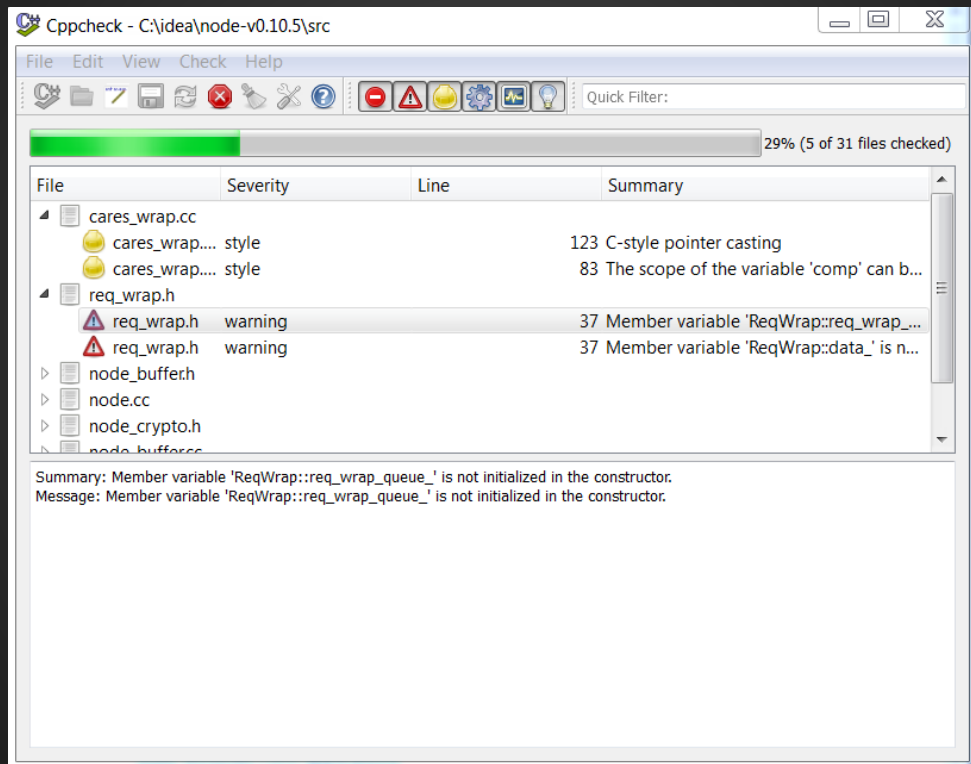
[Project Information](#)[Report Abuse](#)**Description:**

Simple DirectMedia Layer is a cross-platform multimedia library designed to provide low level access to audio, keyboard, mouse, joystick, 3D hardware via OpenGL, and 2D video framebuffer. Homepage: <http://www.libsdl.org/>

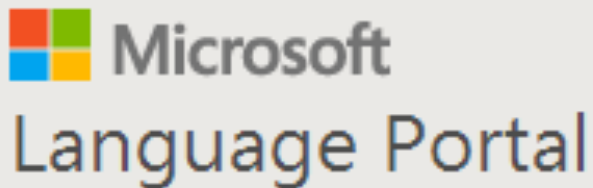
Settings

Close

Cppcheck



L10N - verify your translations

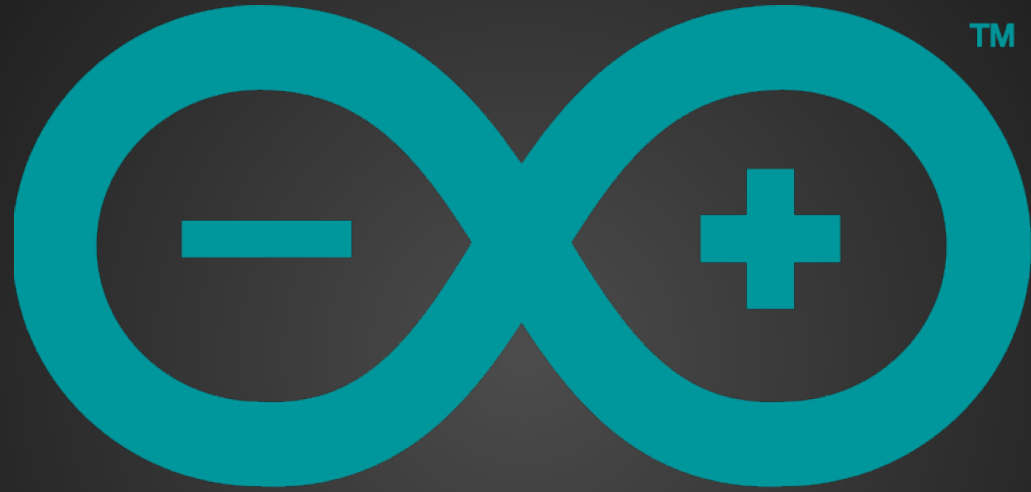


<http://www.microsoft.com/Language>

cURL

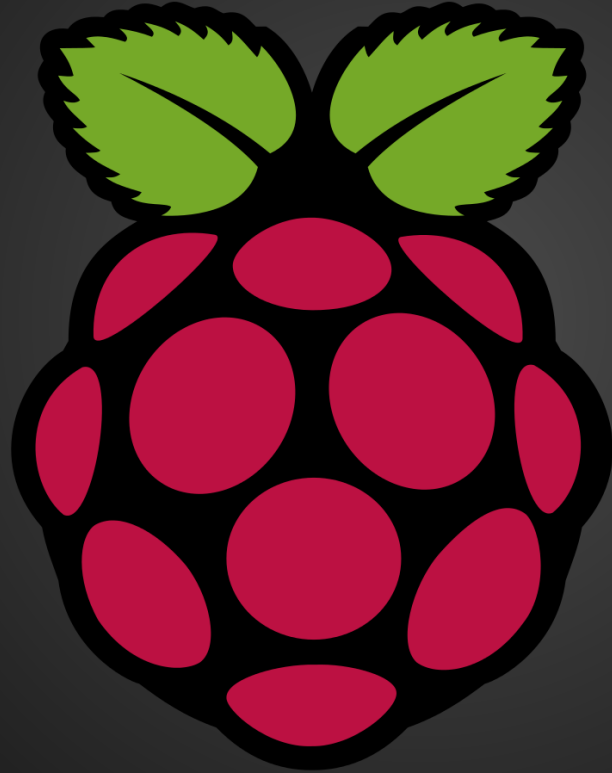
Continuous integration





ARDUINO

Raspberry Pi



Opportunity for innovative students



Bachelor and master thesis @Y Soft

Contact person: Andriy.Stetsko@ysoft.com