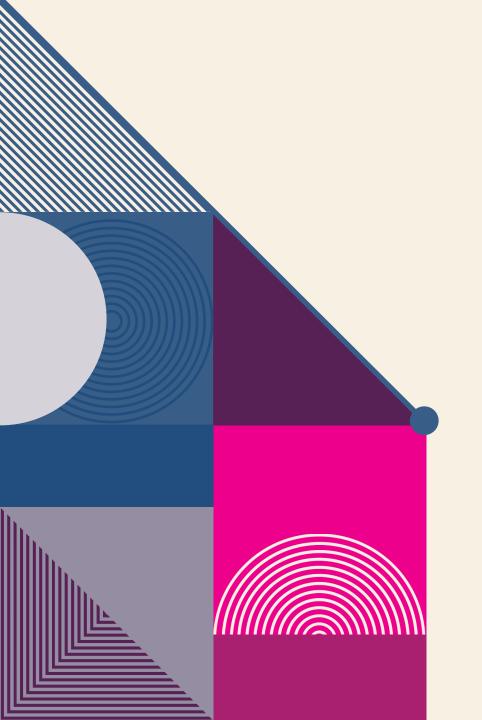# CMPLE++

A cross-platform C++ code generator/preprocessor.
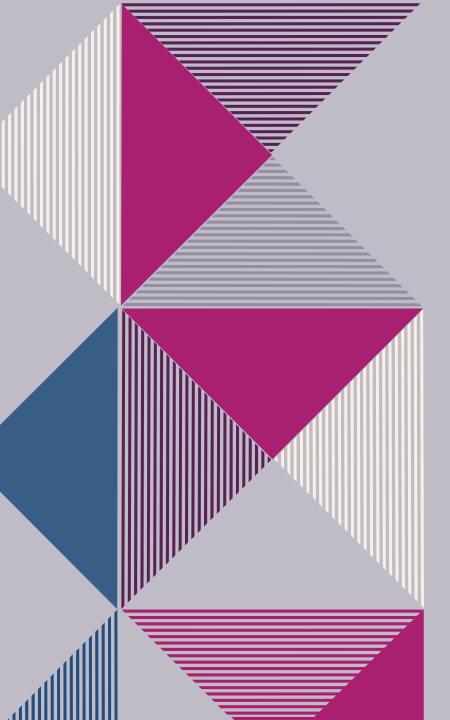
By Anton Dospekhov &
Igor Parfenov

ABOUT C++ DEVELOPMENT

Developing a complex graphical C++ project often happens to be frustrating due to amount of boilerplate code a programmer has to write.

The utilization of graphical capabilities requires from you extensive knowledge about GPU API or libraries that operate it.

PROBLEM

BOILERPLATE

You can spend whole day trying to set up a simple "Hello world" system

COMPLEX OOP SYSTEM

Developer has to use all capabilities of OOP to setup relations between components
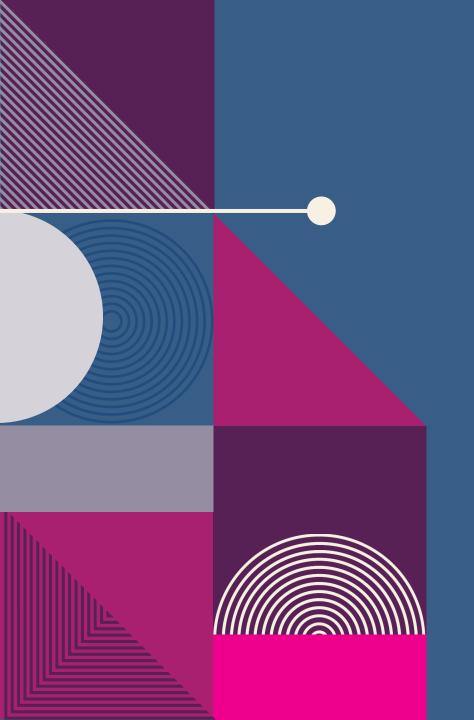
HIGH ENTRY LEVEL

Complicated workflow stops beginner game developers from learning game engine architecture

GRAPHICAL PIPELINE

Modern Graphical APIs require full understanding of rendering pipeline & resource management

NO SANDBOX

To work in simplified environment you need to build it yourself

SOLUTION

AUTO REFERENCING

Signatures and definitions are automatically linked, modules are automatically imported, if used

API CALLS ELISION

Narrowed set of commands allows beginner to learn minimum new functions

EASY TO USE

Simplified syntax requirement allows programmers to write the code they have to

MEMORY MANAGEMENT

Automatic resource allocation, distribution, relocation, optimization, defragmentation and release

PRODUCT OVERVIEW

PREPROCESSOR

Our tool runs on source code files before compilation and fills all necessary information

PROJECT SCOPE

All project files are processed at once, that allows components resolution on global level

MEMORY INJECTOR

All memory operations are performed through special resource management system to avoid leaks and corruption

LIBRARY DEPENDENCY

The project comes with set of essential and commonly accepted third-party tools and libraries that are imported when needed

Cmple++

PRODUCT BENEFITS

User can write light and "Cmple" code

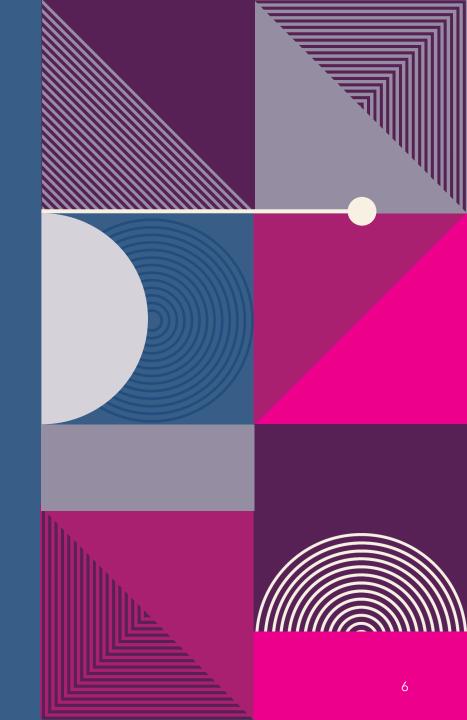No thinking about function declaration & definition matching

No thinking about header inclusion and guards

No thinking about libraries depenedcies

No thinking about access protection system

No thinking about memory issues

It is not a new language, it's same, but "Cmple"!

Cmple++

CONVERTER OVERVIEW

LOGIC VIEW

PHYSICAL VIEW

Test Suite

General libraries

Graphical libraries

Test Application
CmpleTest.exe

FMT library
fmt.lib

GLFW library
glfw.lib

Legend

Google Test library
gtest.lib

Free-Type library
freetype.lib

GLAD library
glad.lib

Executable

Open GL library
gl32.lib

Output

Library

Main Application
Cmple.exe

Cmple Core library
CmpleCore.lib

UML DIAGRAM

Converter

+ currentFile: FolderIterator
+ parser: FileParser
+ builder: ProjectBuilder
+ managers: vector<Manager*>

+ ReadNextFile(): void
+ ConvertTree(): void
+ WriteProject(string path): void

FolderIterator

+ currentIterator: path_iterator
+ rootIterator: path_iterator
+ sourceCount: size_t
+ resourceCount: size_t

+ Next(): string

ProjectBuilder

+ globalTree: TokenTree

+ addFile(TokenTree tree): void

FileParser

+ file: ProjectFile*
+ validator: Validator
+ field: type

+ LoadFile(string filename): int
+ RunChecks(): bool
+ GetTree(): TokenTree

<<interface>>
Manager

+ Transform(TokenTree project): TokenTree

<<interface>>
ProjectFile

+ file: fstream
+ pool: byte[]
+ tree: TokenTree

+ Process(): void
+ Read(): void
+ Save(string filename): void

DefinitionManager

+ checkList: map<string, bool>

+ Transform(TokenTree project): TokenTree

SCManager

+ classDatabase: map<string, pair<string, string>>
+ files: map<string, vector<string>>

+ Transform(TokenTree project): TokenTree

TokenTree

+ root: Node*

+ GetNode(string id): Node*

Validator

+ errors: vector<string>
+ warnings: vector<string>

+ GetReport(): vector<string>
+ Syntax(): bool
+ Globals(TokenTree: ProjectTree): bool
+ FileSystem(): bool
+ Misc(): bool

Node

+ id: string
+ parent: Node*
+ token: Token

MemoryManager

+ allocateInjections: vector<tuple<string, int, int>>
+ accessInjections: vector<tuple<string, int, int>>
+ releaseInjections: vector<tuple<string, int, int>>

+ Transform(TokenTree project): TokenTree

Token

+ value: string

SourceFile

+ Process(): void

ResourceFile

+ Process(): void

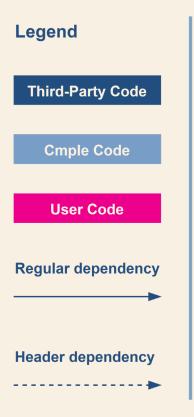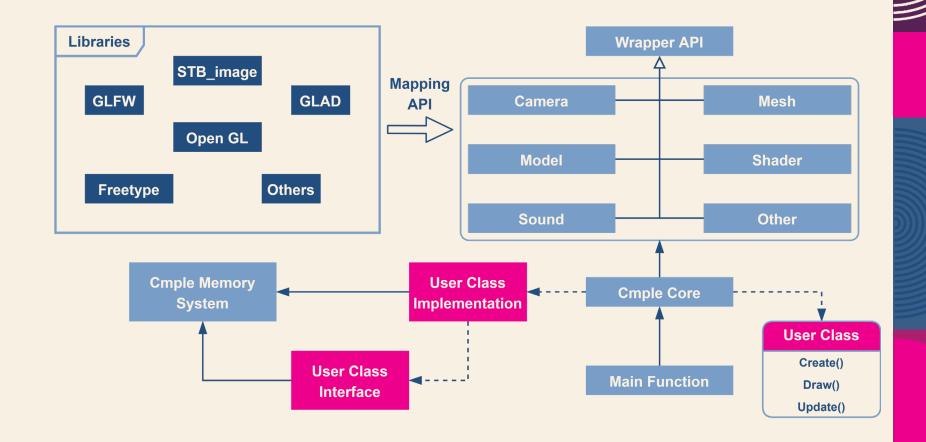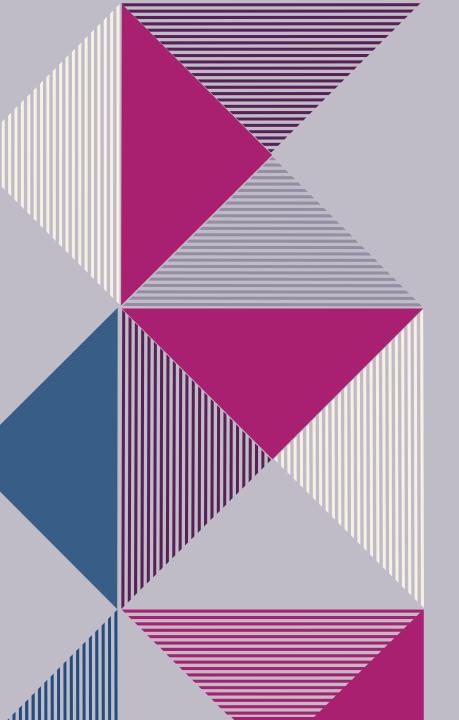# RUNTIME OVERVIEW

RUNTIME OVERVIEW

SUMMARY

Cmple tool is a quality of life tool. By using our code generator, we help beginner programmers to enter into world of graphical applications. It is essential for enthusiasts that want their own playground for experiments. As George Corrette said, "First learn computer science and all the theory. Next develop a programming style. Then forget all that and just write code."

- KEEP IT SIMPLE, STUPID (AND SHORT) -
KISS

THANK YOU

Anton Dospekhov

Igor Parfenov

a.dospekhov@innopolis.university

i.parfenov@innopolis.university