Hipreme Engine

BRINGING D EVERYWHERE

UNDERSTANDING THE PROJECT

Hipreme
Engine

WHO'S MARCELO MANCINI (HIPREME)

- GAME DEVELOPER;
- ENGINE PROGRAMMER;
- OPEN SOURCE;
- HOBBY MUSICIAN.

- D
- TYPESCRIPT
- LUA
- JAVA
- HAXE
- C++
- C#

Hipreme
Engine

WHAT IS HIPREME ENGINE

- D GAME ENGINE;
- 5% JAVA, C/C++, OBJ-C, JS;
- 2D;
- AVOIDS C;
- OOP BASED;
- HOT RELOADING;
- MUCH MORE...

- XBOX SERIES ;
- ANDROID;
- BROWSER (WASM);
- PS VITA;
- MACOS;
- WINDOWS;
- LINUX.

Available at

https://github.com/MrcSnm/HipremeEngine

Hipreme
Engine

THE START

Hipreme
Engine

INITIAL IDEA

- USE SDL 2;
- DO ABSTRACTIONS;
- DO IT THE "D" WAY;
- LEARN OPENGL WITH D.

MAIN REFERENCES

lib GDX
PHASER
UNREAL ENGINE
LÖVE
Unity
COCOS

PORTING PHASE 1 - ANDROID

- TESTING D;
- THE BUILD SYSTEM;
- THE LACK OF DOCS.

PORTING PHASE 1 - XBOX SERIES S

- TESTING D;
- THE BUILD SYSTEM;
- C++/CX AND UWP;
- DIRECTX.

PORTING PHASE A - LESSONS

- USE MORE THAN 1 LANGUAGE;
- SHARED LIBRARIES ROCKS;
- AVOID C DEPENDENCIES;
- USE D LIBRARIES.

THE PLAN

Hipreme
Engine

PORTING EVERYTHING FIRST

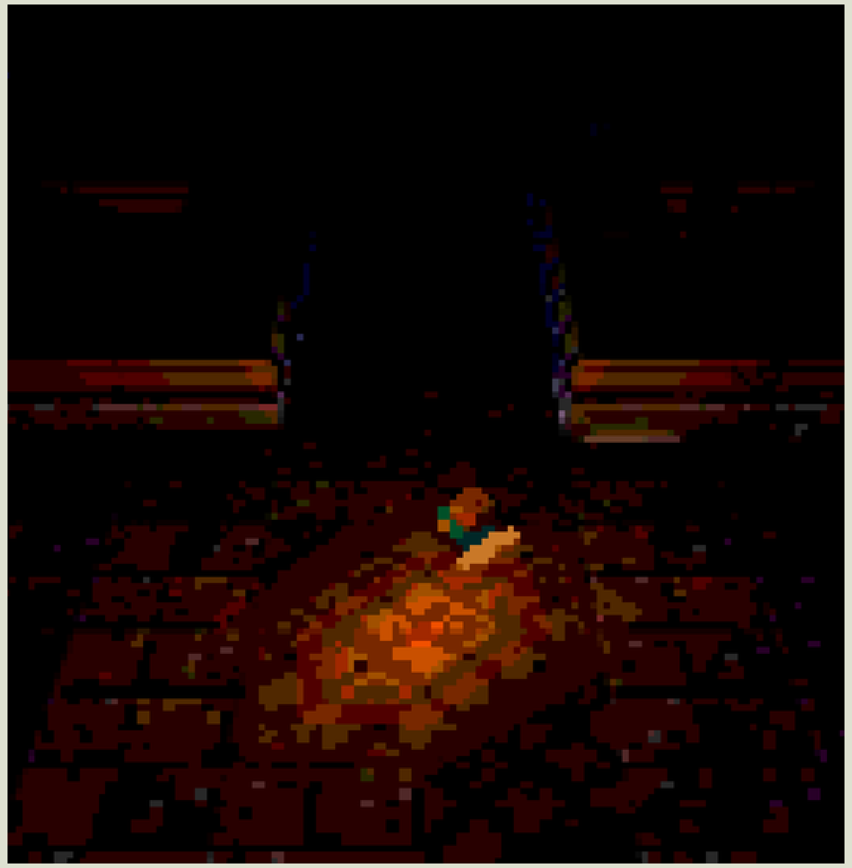
- LESS REFACTORS;
- MORE STABILITY;
- MORE THAN A SANDBOX;
- PROOF OF D CAPABILITIES.

TARGET WEBASSEMBLY - THE WALL

Hipreme
Engine

LOOKING INTO PRIOR WORK

UNDERRUN



Available in
https://skoppe.github.io/spasm/examples/underrun/

THE ART OF MACHINERY EMSCRIPTEN-D

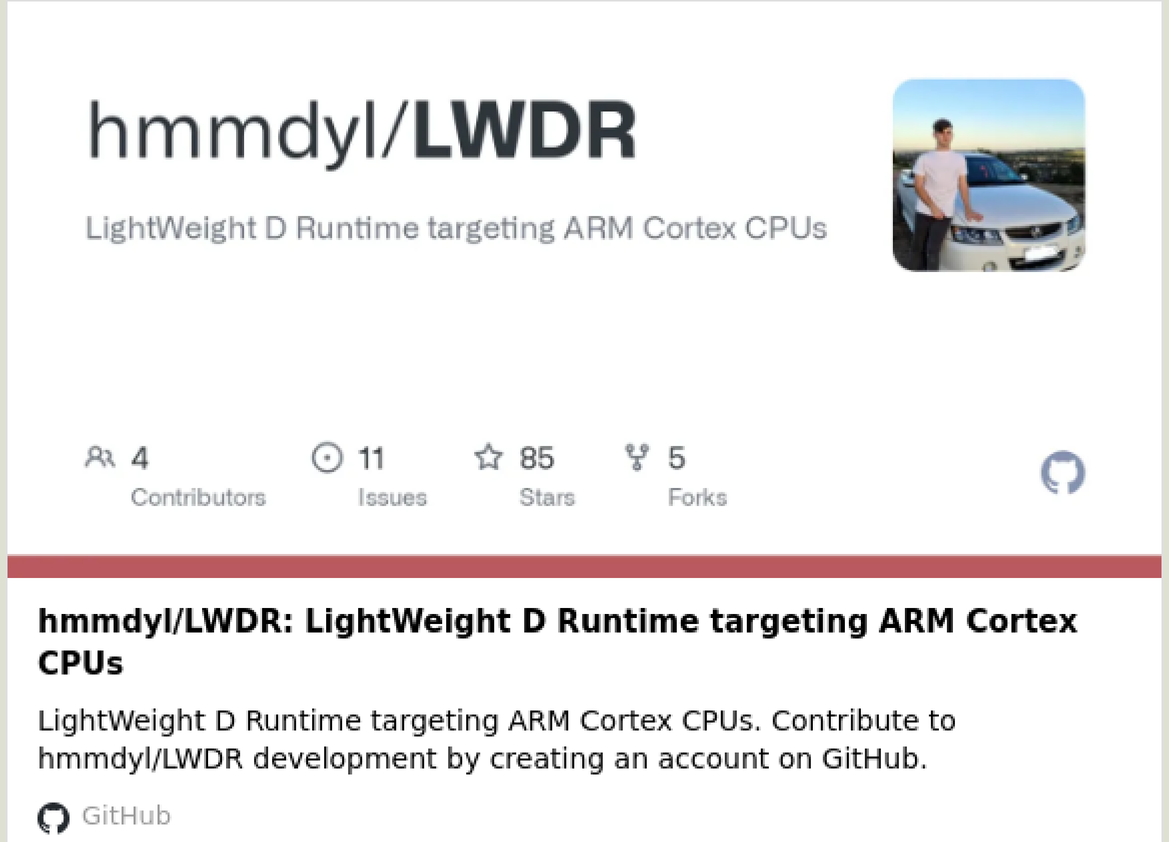
Available in
https://theartofmachinery.com/2018/12/20/emscripten_d.html

Hipreme
Engine

MAIN PROBLEMS

- NO D-RUNTIME;
- HAZY DOCUMENTATIONS;
- LACK OF REFERENCES;
- MOST PROJECTS WERE BIG.

SOME HOPE



hmmdyl/**LWDR**

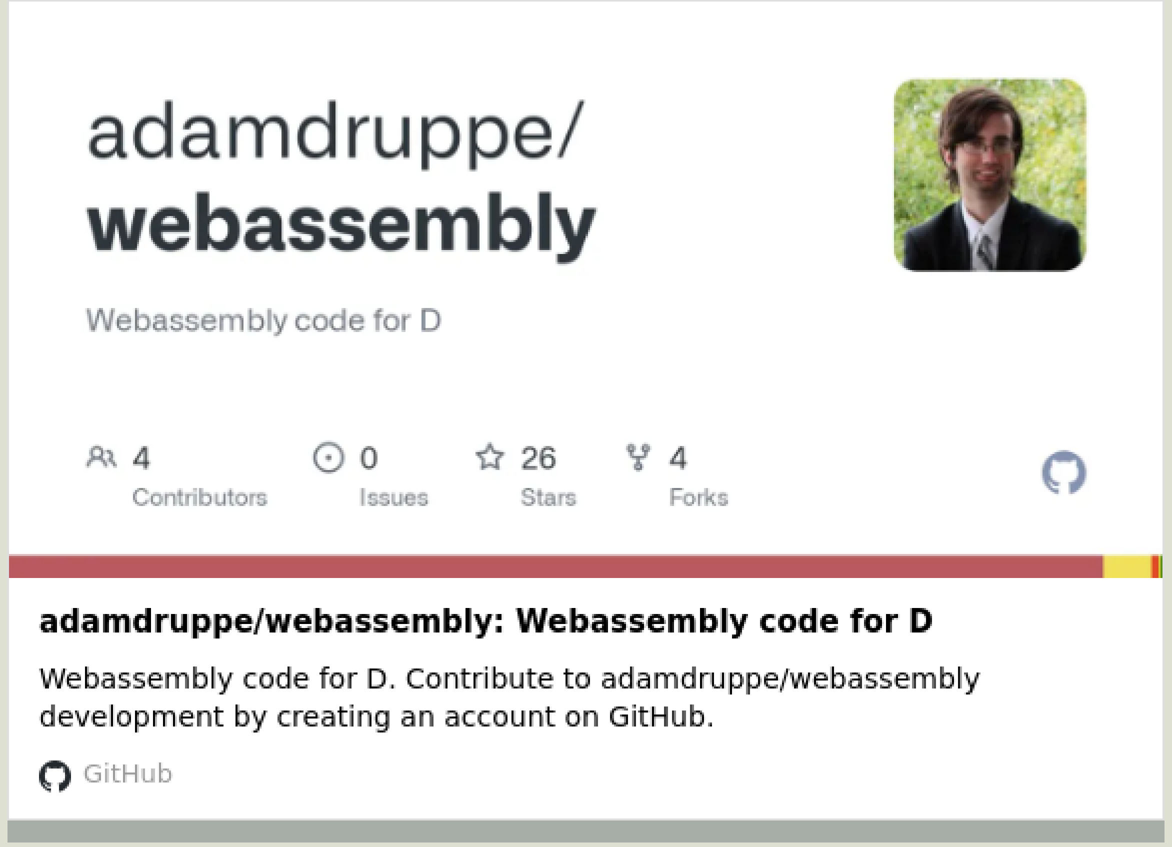
LightWeight D Runtime targeting ARM Cortex CPUs

4 Contributors · 11 Issues · 85 Stars · 5 Forks

hmmdyl/LWDR: LightWeight D Runtime targeting ARM Cortex CPUs

LightWeight D Runtime targeting ARM Cortex CPUs. Contribute to hmmdyl/LWDR development by creating an account on GitHub.

GitHub



adamdruppe/**webassembly**

Webassembly code for D

4 Contributors · 0 Issues · 26 Stars · 4 Forks

adamdruppe/webassembly: Webassembly code for D

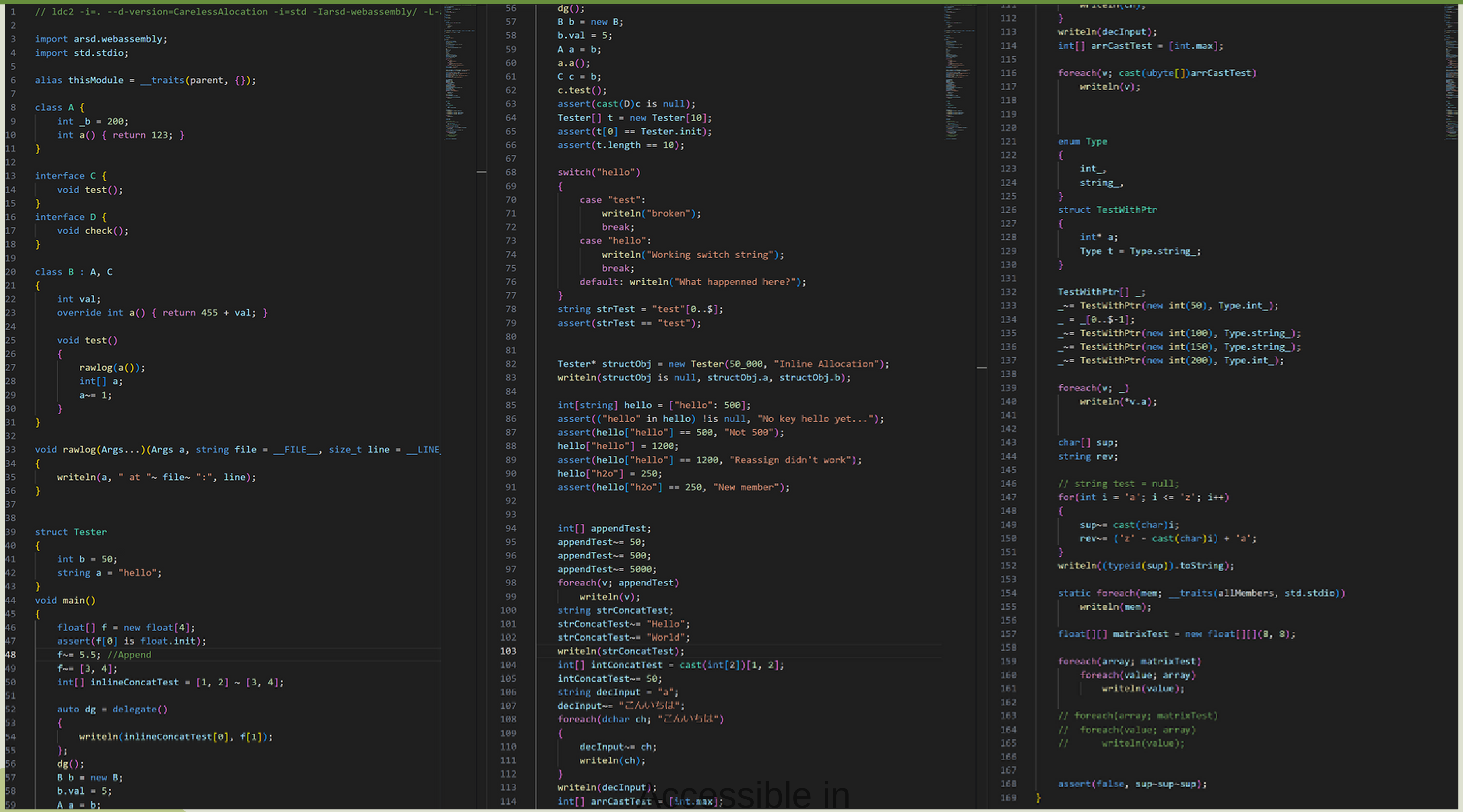
Webassembly code for D. Contribute to adamdruppe/webassembly development by creating an account on GitHub.

GitHub

Accessible in
https://github.com/hmmdyl/LWDR

Accessible in
https://github.com/adamdruppe/webassembly

EXTENSIVE TESTING



```d
// ldc2 -i=. --d-version=CarelessAlocation -i=std -Iarsd-webassembly/ -L-

import arsd.webassembly;
import std.stdio;

alias thisModule = __traits(parent, {});

class A {
    int _b = 200;
    int a() { return 123; }
}

interface C {
    void test();
}
interface D {
    void check();
}

class B : A, C
{
    int val;
    override int a() { return 455 + val; }

    void test()
    {
        rawlog(a());
        int[] a;
        a~= 1;
    }
}

void rawlog(Args...)(Args a, string file = __FILE__, size_t line = __LINE_
{
    writeln(a, " at "~ file~ ":", line);
}


struct Tester
{
    int b = 50;
    string a = "hello";
}
void main()
{
    float[] f = new float[4];
    assert(f[0] is float.init);
    f~= 5.5; //Append
    f~= [3, 4];
    int[] inlineConcatTest = [1, 2] ~ [3, 4];

    auto dg = delegate()
    {
        writeln(inlineConcatTest[0], f[1]);
    };
    dg();
    B b = new B;
    b.val = 5;
    A a = b;
```

```d
    dg();
    B b = new B;
    b.val = 5;
    A a = b;
    a.a();
    C c = b;
    c.test();
    assert(cast(D)c is null);
    Tester[] t = new Tester[10];
    assert(t[0] == Tester.init);
    assert(t.length == 10);

    switch("hello")
    {
        case "test":
            writeln("broken");
            break;
        case "hello":
            writeln("Working switch string");
            break;
        default: writeln("What happenned here?");
    }
    string strTest = "test"[0..$];
    assert(strTest == "test");


    Tester* structObj = new Tester(50_000, "Inline Allocation");
    writeln(structObj is null, structObj.a, structObj.b);

    int[string] hello = ["hello": 500];
    assert(("hello" in hello) !is null, "No key hello yet...");
    assert(hello["hello"] == 500, "Not 500");
    hello["hello"] = 1200;
    assert(hello["hello"] == 1200, "Reassign didn't work");
    hello["h2o"] = 250;
    assert(hello["h2o"] == 250, "New member");


    int[] appendTest;
    appendTest~= 50;
    appendTest~= 500;
    appendTest~= 5000;
    foreach(v; appendTest)
        writeln(v);
    string strConcatTest;
    strConcatTest~= "Hello";
    strConcatTest~= "World";
    writeln(strConcatTest);
    int[] intConcatTest = cast(int[2])[1, 2];
    intConcatTest~= 50;
    string decInput = "a";
    decInput~= "こんいちは";
    foreach(dchar ch; "こんいちは")
    {
        decInput~= ch;
        writeln(ch);
    }
    writeln(decInput);
    int[] arrCastTest = [int.max];
```

```d
        writeln(ch);
    }
    writeln(decInput);
    int[] arrCastTest = [int.max];

    foreach(v; cast(ubyte[])arrCastTest)
        writeln(v);



    enum Type
    {
        int_,
        string_,
    }
    struct TestWithPtr
    {
        int* a;
        Type t = Type.string_;
    }

    TestWithPtr[] _;
    _~= TestWithPtr(new int(50), Type.int_);
    _ = _[0..$-1];
    _~= TestWithPtr(new int(100), Type.string_);
    _~= TestWithPtr(new int(150), Type.string_);
    _~= TestWithPtr(new int(200), Type.int_);

    foreach(v; _)
        writeln(*v.a);


    char[] sup;
    string rev;

    // string test = null;
    for(int i = 'a'; i <= 'z'; i++)
    {
        sup~= cast(char)i;
        rev~= ('z' - cast(char)i) + 'a';
    }
    writeln((typeid(sup)).toString);

    static foreach(mem; __traits(allMembers, std.stdio))
        writeln(mem);

    float[][] matrixTest = new float[][](8, 8);

    foreach(array; matrixTest)
        foreach(value; array)
            writeln(value);

    // foreach(array; matrixTest)
    //  foreach(value; array)
    //      writeln(value);


    assert(false, sup~sup~sup);
}
```

Accessible in

https://github.com/MrcSnm/webassembly/blob/062b2555fbdb14b278f3e3d6a508a5749b6e4b5a/test_runtime.d

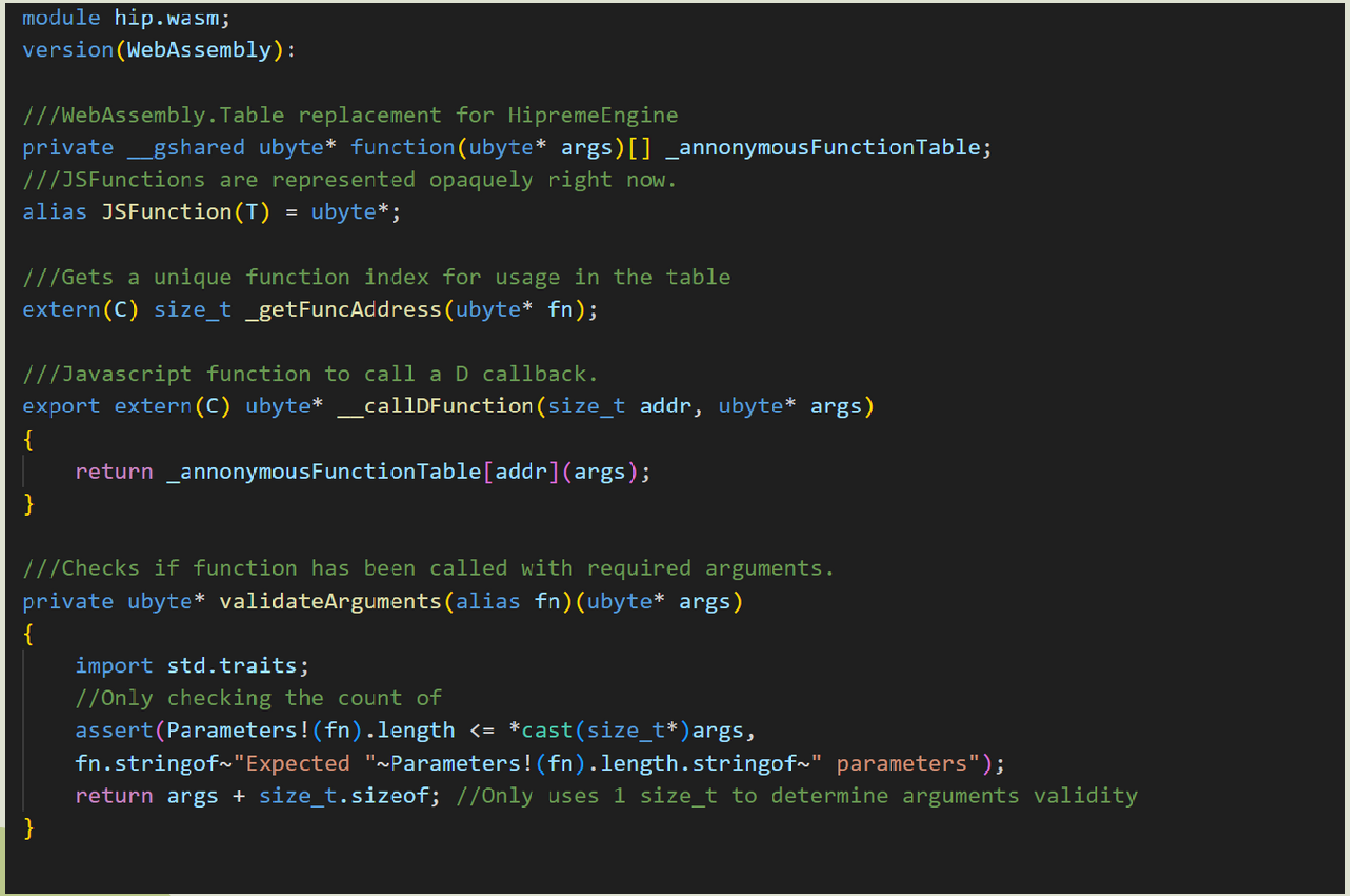
Hipreme Engine

PORTING PHASE 2 - WEBASSEMBLY

- D - JS COMMUNICATION;
- BYE-BYE, C STDLIB;
- THE WEB PHILOSOFY;
- ASSETS LOAD REFACTOR;
- ASYNC INTERFACE ONLY;
- NO-COLLECT GC.

Hipreme
Engine

WASM COMMUNICATION

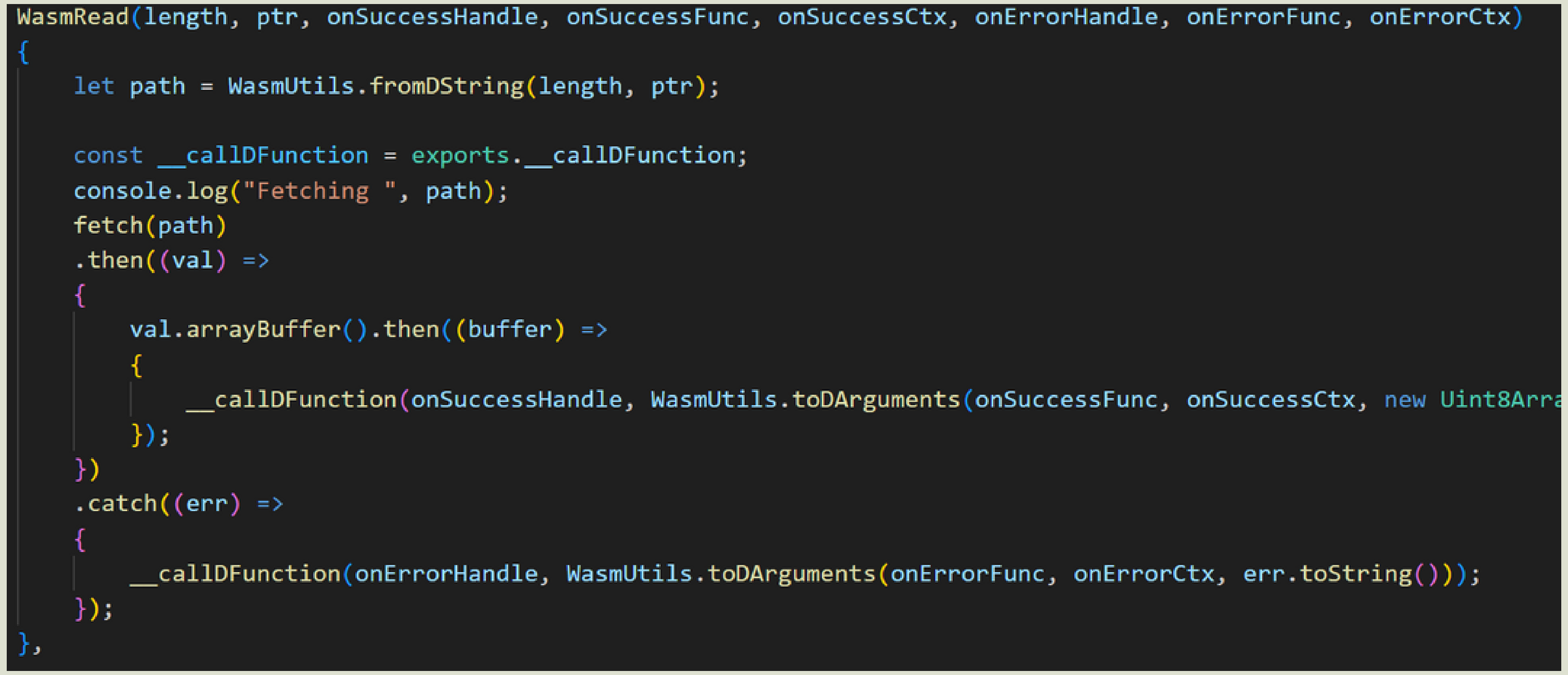


```
module hip.wasm;
version(WebAssembly):

///WebAssembly.Table replacement for HipremeEngine
private __gshared ubyte* function(ubyte* args)[] _annonymousFunctionTable;
///JSFunctions are represented opaquely right now.
alias JSFunction(T) = ubyte*;

///Gets a unique function index for usage in the table
extern(C) size_t _getFuncAddress(ubyte* fn);

///Javascript function to call a D callback.
export extern(C) ubyte* __callDFunction(size_t addr, ubyte* args)
{
    return _annonymousFunctionTable[addr](args);
}

///Checks if function has been called with required arguments.
private ubyte* validateArguments(alias fn)(ubyte* args)
{
    import std.traits;
    //Only checking the count of
    assert(Parameters!(fn).length <= *cast(size_t*)args,
    fn.stringof~"Expected "~Parameters!(fn).length.stringof~" parameters");
    return args + size_t.sizeof; //Only uses 1 size_t to determine arguments validity
}
```

Hipreme
Engine

Sending User Defined Callbacks

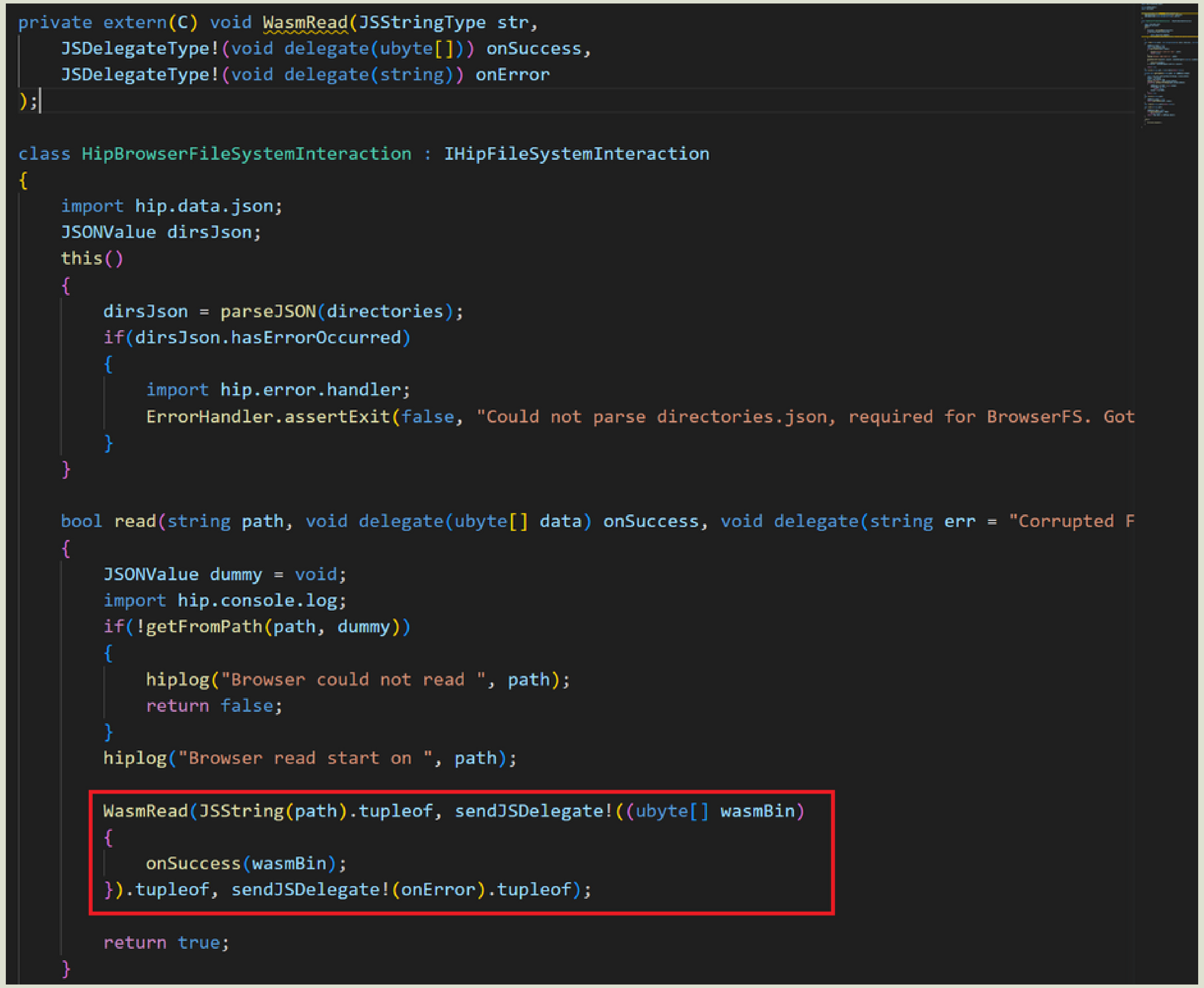
WASM COMMUNICATION



```
WasmRead(length, ptr, onSuccessHandle, onSuccessFunc, onSuccessCtx, onErrorHandle, onErrorFunc, onErrorCtx)
{
    let path = WasmUtils.fromDString(length, ptr);

    const __callDFunction = exports.__callDFunction;
    console.log("Fetching ", path);
    fetch(path)
    .then((val) =>
    {
        val.arrayBuffer().then((buffer) =>
        {
            __callDFunction(onSuccessHandle, WasmUtils.toDArguments(onSuccessFunc, onSuccessCtx, new Uint8Arra
        });
    })
    .catch((err) =>
    {
        __callDFunction(onErrorHandle, WasmUtils.toDArguments(onErrorFunc, onErrorCtx, err.toString()));
    });
},
```

Javascript Implementation

Receiving D Callback

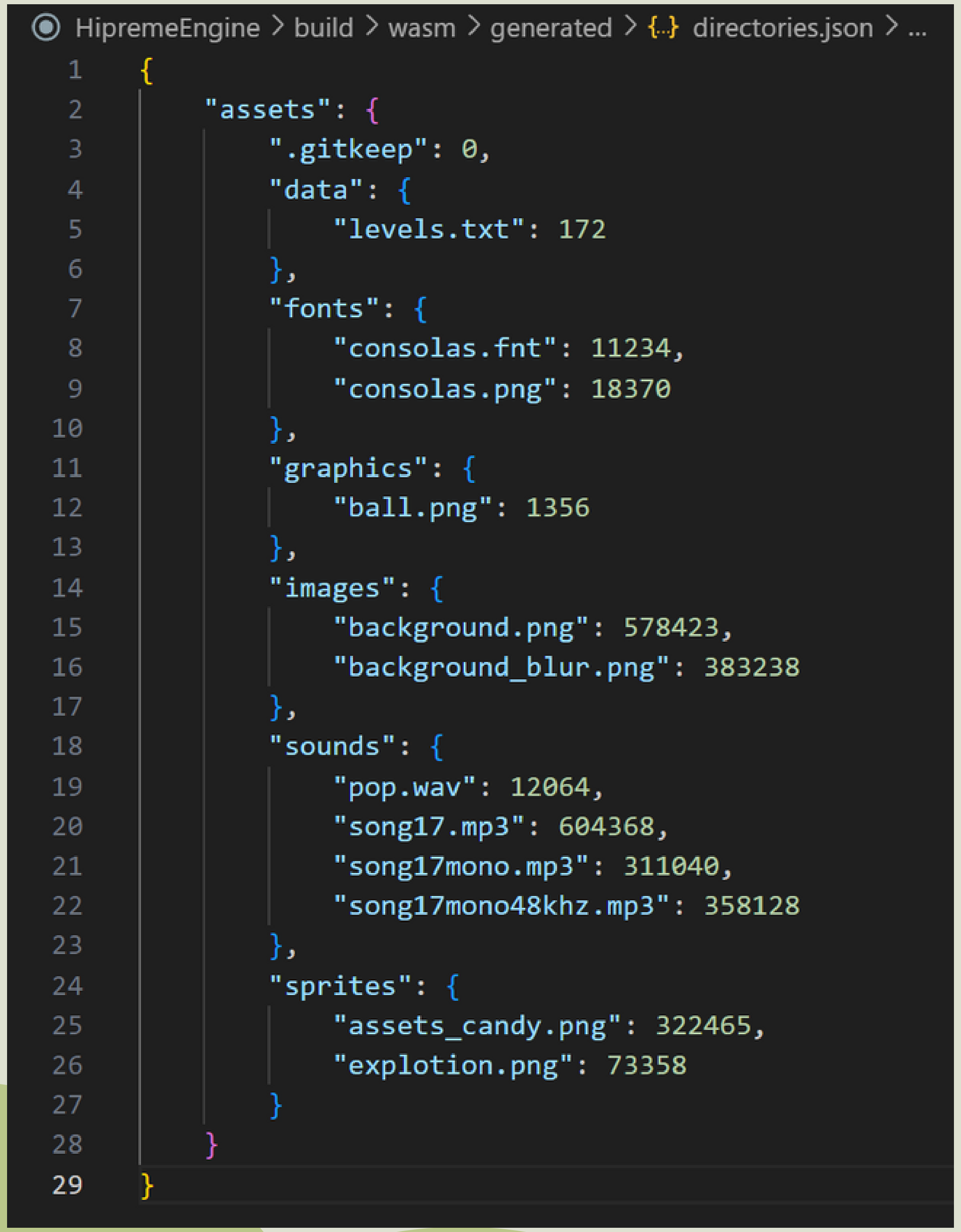


```
private extern(C) void WasmRead(JSStringType str,
    JSDelegateType!(void delegate(ubyte[])) onSuccess,
    JSDelegateType!(void delegate(string)) onError
);

class HipBrowserFileSystemInteraction : IHipFileSystemInteraction
{
    import hip.data.json;
    JSONValue dirsJson;
    this()
    {
        dirsJson = parseJSON(directories);
        if(dirsJson.hasErrorOccurred)
        {
            import hip.error.handler;
            ErrorHandler.assertExit(false, "Could not parse directories.json, required for BrowserFS. Got
        }
    }

    bool read(string path, void delegate(ubyte[] data) onSuccess, void delegate(string err = "Corrupted F
    {
        JSONValue dummy = void;
        import hip.console.log;
        if(!getFromPath(path, dummy))
        {
            hiplog("Browser could not read ", path);
            return false;
        }
        hiplog("Browser read start on ", path);

        WasmRead(JSString(path).tupleof, sendJSDelegate!((ubyte[] wasmBin)
        {
            onSuccess(wasmBin);
        }).tupleof, sendJSDelegate!(onError).tupleof);

        return true;
    }
```

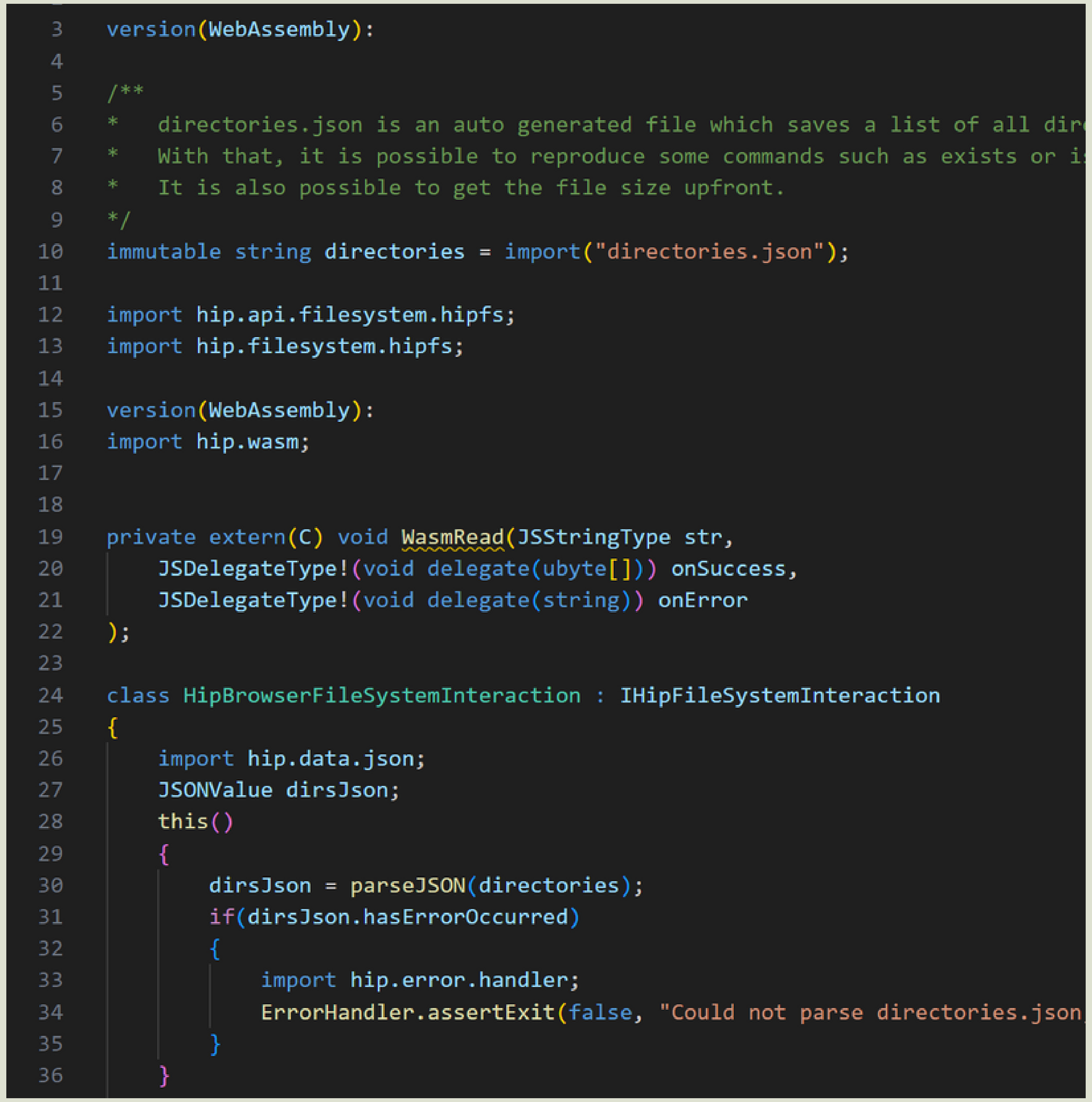
D Calling JS Function and sending its arguments

Hipreme Engine

WASM FILESYSTEM


HipremeEngine > build > wasm > generated > {..} directories.json > ...

```json
{
    "assets": {
        ".gitkeep": 0,
        "data": {
            "levels.txt": 172
        },
        "fonts": {
            "consolas.fnt": 11234,
            "consolas.png": 18370
        },
        "graphics": {
            "ball.png": 1356
        },
        "images": {
            "background.png": 578423,
            "background_blur.png": 383238
        },
        "sounds": {
            "pop.wav": 12064,
            "song17.mp3": 604368,
            "song17mono.mp3": 311040,
            "song17mono48khz.mp3": 358128
        },
        "sprites": {
            "assets_candy.png": 322465,
            "explotion.png": 73358
        }
    }
}
```

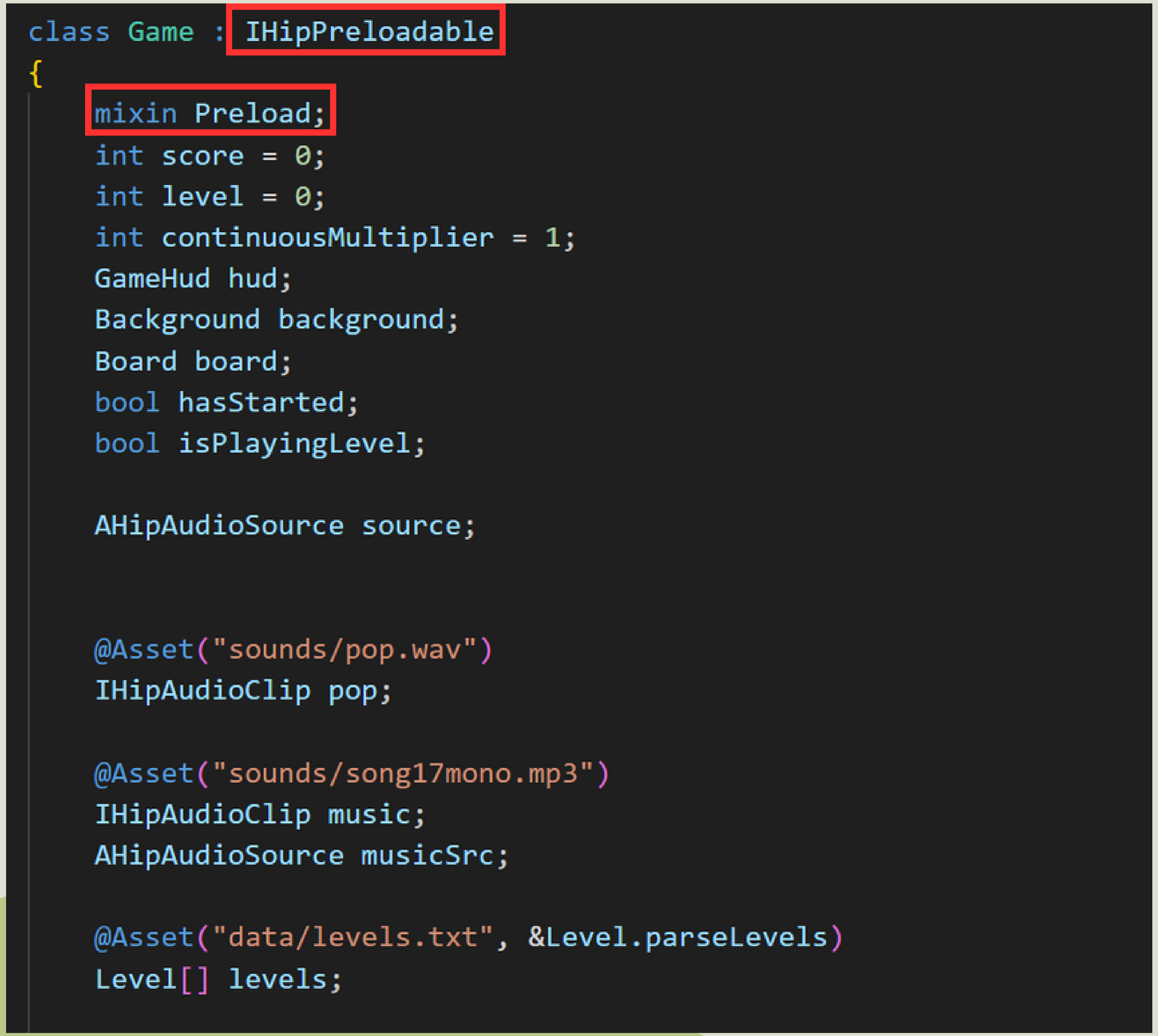


```d
version(WebAssembly):

/**
*   directories.json is an auto generated file which saves a list of all dir
*   With that, it is possible to reproduce some commands such as exists or i
*   It is also possible to get the file size upfront.
*/
immutable string directories = import("directories.json");

import hip.api.filesystem.hipfs;
import hip.filesystem.hipfs;

version(WebAssembly):
import hip.wasm;


private extern(C) void WasmRead(JSStringType str,
    JSDelegateType!(void delegate(ubyte[])) onSuccess,
    JSDelegateType!(void delegate(string)) onError
);

class HipBrowserFileSystemInteraction : IHipFileSystemInteraction
{
    import hip.data.json;
    JSONValue dirsJson;
    this()
    {
        dirsJson = parseJSON(directories);
        if(dirsJson.hasErrorOccurred)
        {
            import hip.error.handler;
            ErrorHandler.assertExit(false, "Could not parse directories.json
        }
    }
```

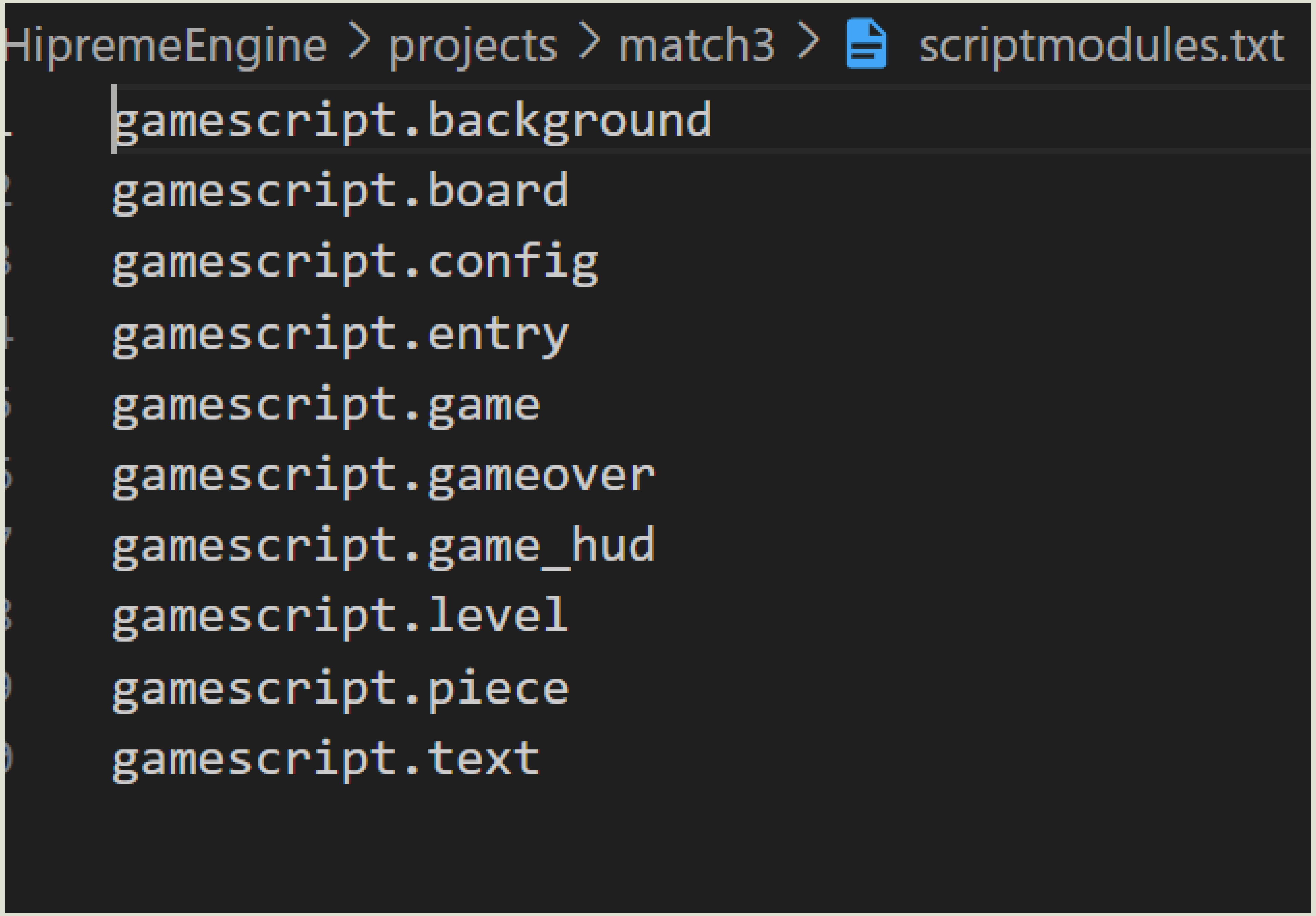
JSON Representation of the available FS

Compile Time JSON Import

WASM ASSET LOADING SOLUTION



```
class Game : IHipPreloadable
{
    mixin Preload;
    int score = 0;
    int level = 0;
    int continuousMultiplier = 1;
    GameHud hud;
    Background background;
    Board board;
    bool hasStarted;
    bool isPlayingLevel;

    AHipAudioSource source;


    @Asset("sounds/pop.wav")
    IHipAudioClip pop;

    @Asset("sounds/song17mono.mp3")
    IHipAudioClip music;
    AHipAudioSource musicSrc;

    @Asset("data/levels.txt", &Level.parseLevels)
    Level[] levels;
```

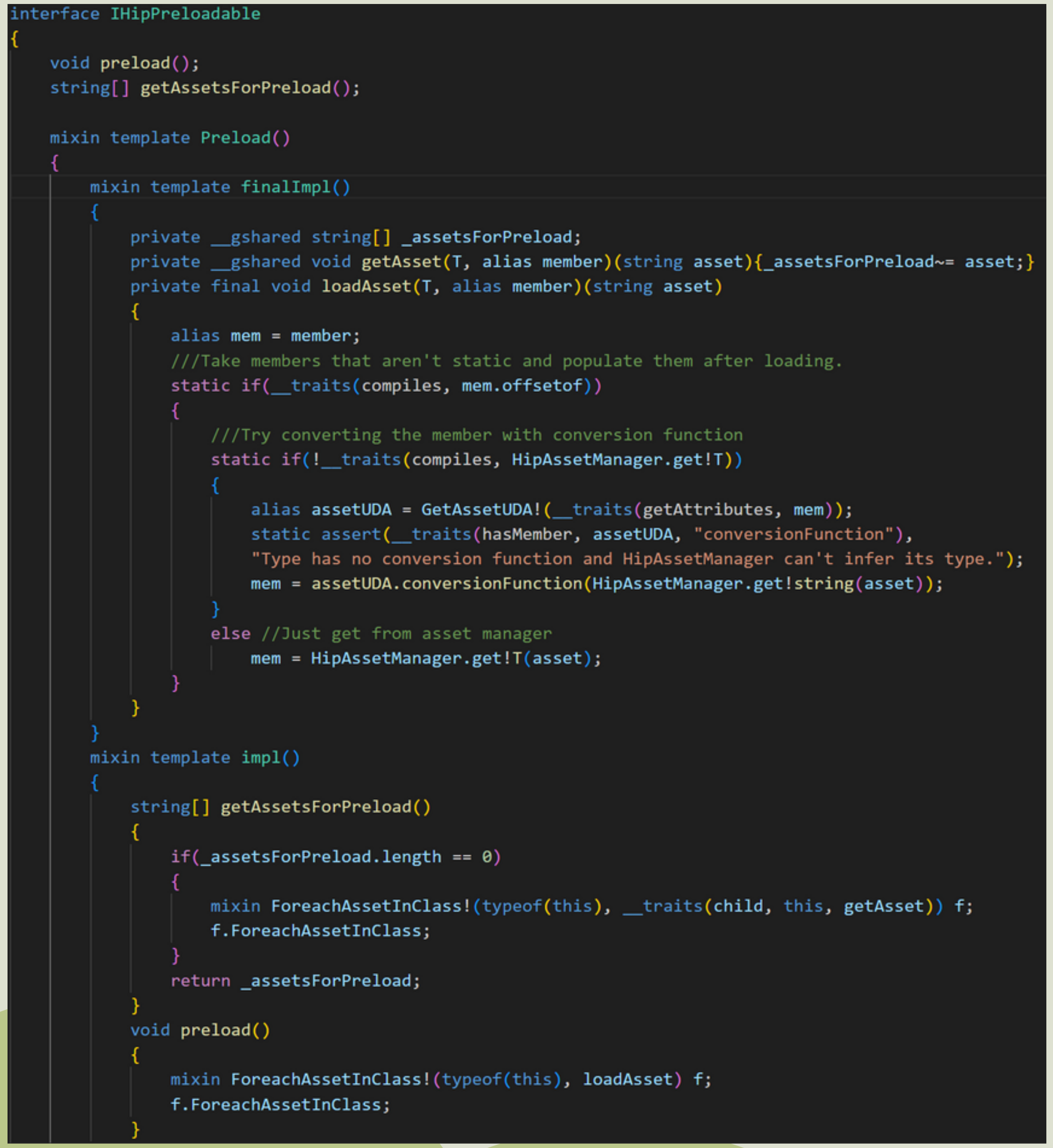


HipremeEngine > projects > match3 > scriptmodules.txt

```
gamescript.background
gamescript.board
gamescript.config
gamescript.entry
gamescript.game
gamescript.gameover
gamescript.game_hud
gamescript.level
gamescript.piece
gamescript.text
```

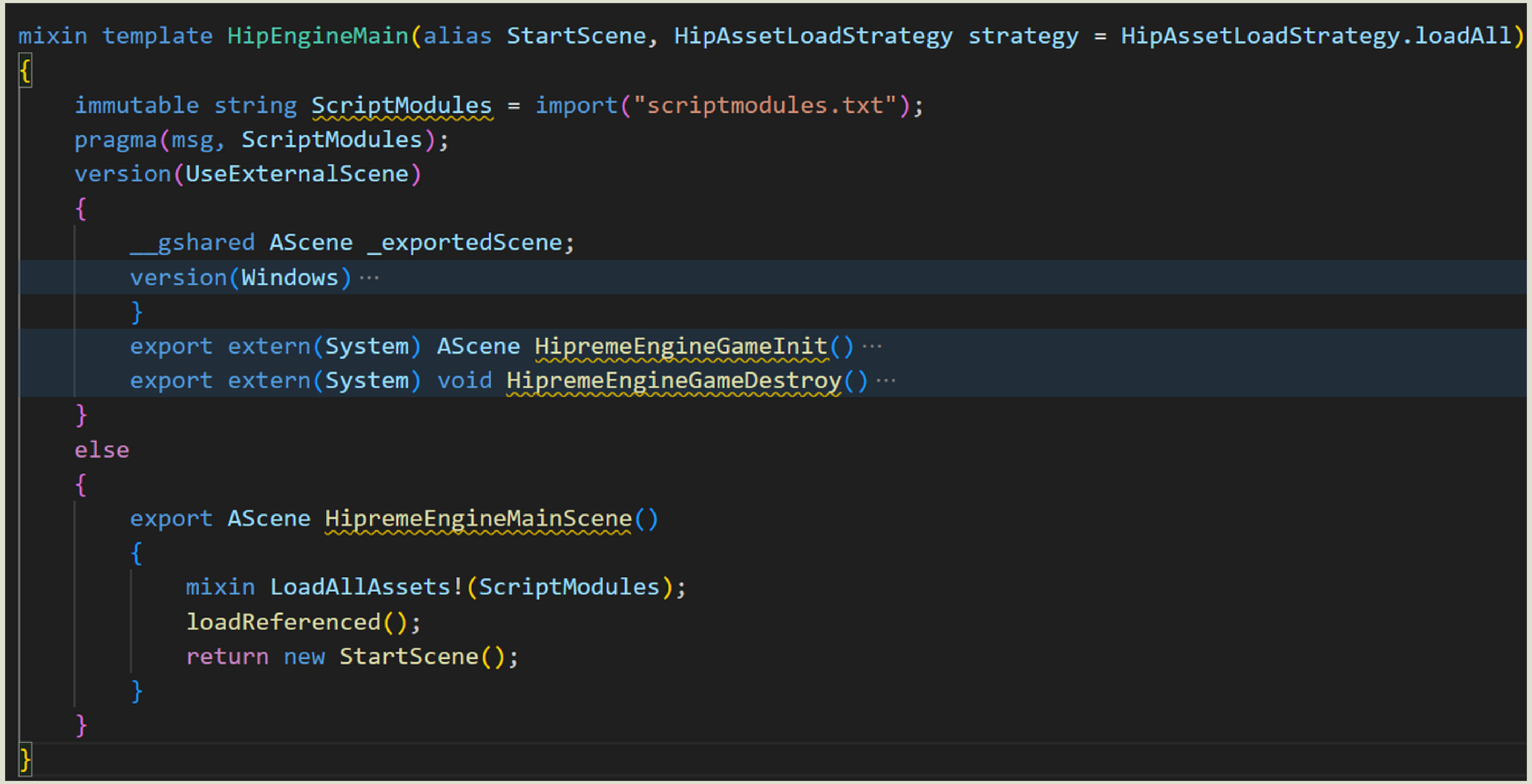
How the assets are found in the game

WASM ASSET LOADING SOLUTION



```d
interface IHipPreloadable
{
    void preload();
    string[] getAssetsForPreload();

    mixin template Preload()
    {
        mixin template finalImpl()
        {
            private __gshared string[] _assetsForPreload;
            private __gshared void getAsset(T, alias member)(string asset){_assetsForPreload~= asset;}
            private final void loadAsset(T, alias member)(string asset)
            {
                alias mem = member;
                ///Take members that aren't static and populate them after loading.
                static if(__traits(compiles, mem.offsetof))
                {
                    ///Try converting the member with conversion function
                    static if(!__traits(compiles, HipAssetManager.get!T))
                    {
                        alias assetUDA = GetAssetUDA!(__traits(getAttributes, mem));
                        static assert(__traits(hasMember, assetUDA, "conversionFunction"),
                        "Type has no conversion function and HipAssetManager can't infer its type.");
                        mem = assetUDA.conversionFunction(HipAssetManager.get!string(asset));
                    }
                    else //Just get from asset manager
                        mem = HipAssetManager.get!T(asset);
                }
            }
        }
        mixin template impl()
        {
            string[] getAssetsForPreload()
            {
                if(_assetsForPreload.length == 0)
                {
                    mixin ForeachAssetInClass!(typeof(this), __traits(child, this, getAsset)) f;
                    f.ForeachAssetInClass;
                }
                return _assetsForPreload;
            }
            void preload()
            {
                mixin ForeachAssetInClass!(typeof(this), loadAsset) f;
                f.ForeachAssetInClass;
            }
```

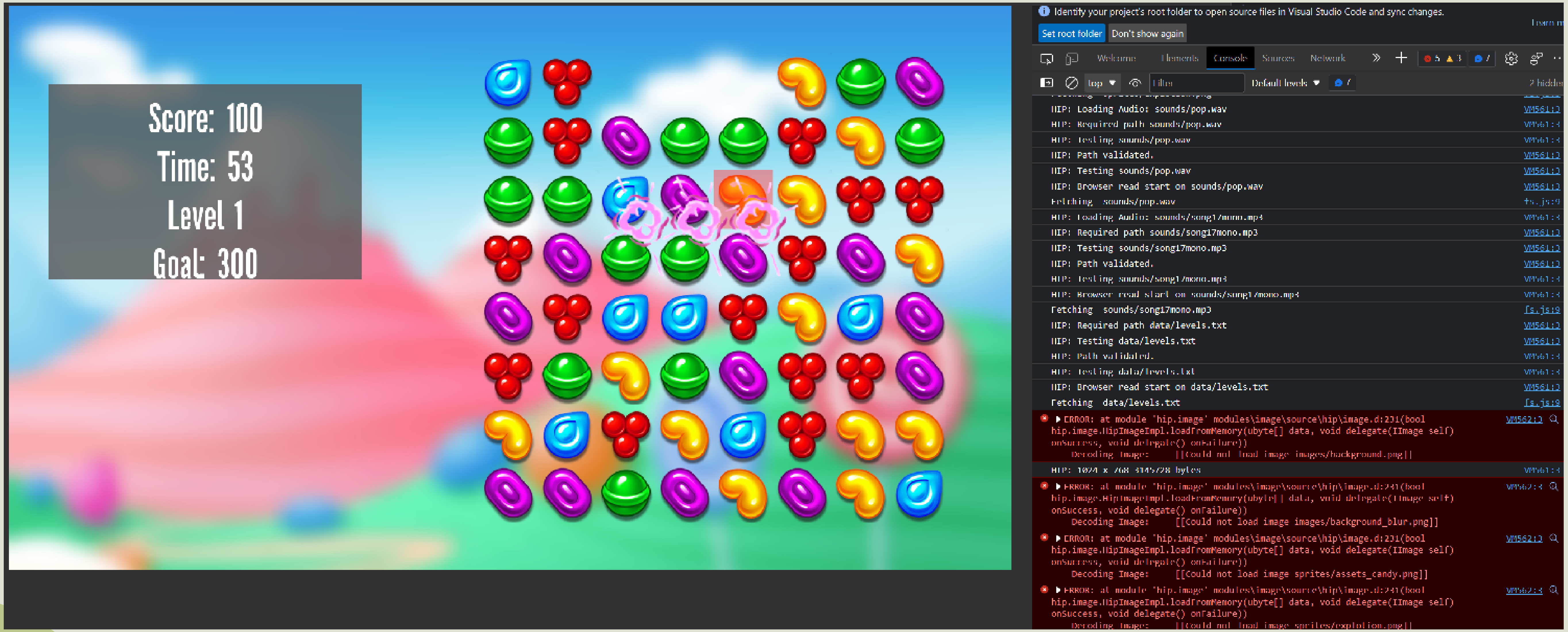
Mixin template on interface



```d
mixin template HipEngineMain(alias StartScene, HipAssetLoadStrategy strategy = HipAssetLoadStrategy.loadAll)
{
    immutable string ScriptModules = import("scriptmodules.txt");
    pragma(msg, ScriptModules);
    version(UseExternalScene)
    {
        __gshared AScene _exportedScene;
        version(Windows) ⋯
        }
        export extern(System) AScene HipremeEngineGameInit() ⋯
        export extern(System) void HipremeEngineGameDestroy() ⋯
    }
    else
    {
        export AScene HipremeEngineMainScene()
        {
            mixin LoadAllAssets!(ScriptModules);
            loadReferenced();
            return new StartScene();
        }
    }
}
```

Loads every asset before instantiating first scene

WEBASSEMBLY RESULT



Score: 100
Time: 53
Level 1
Goal: 300

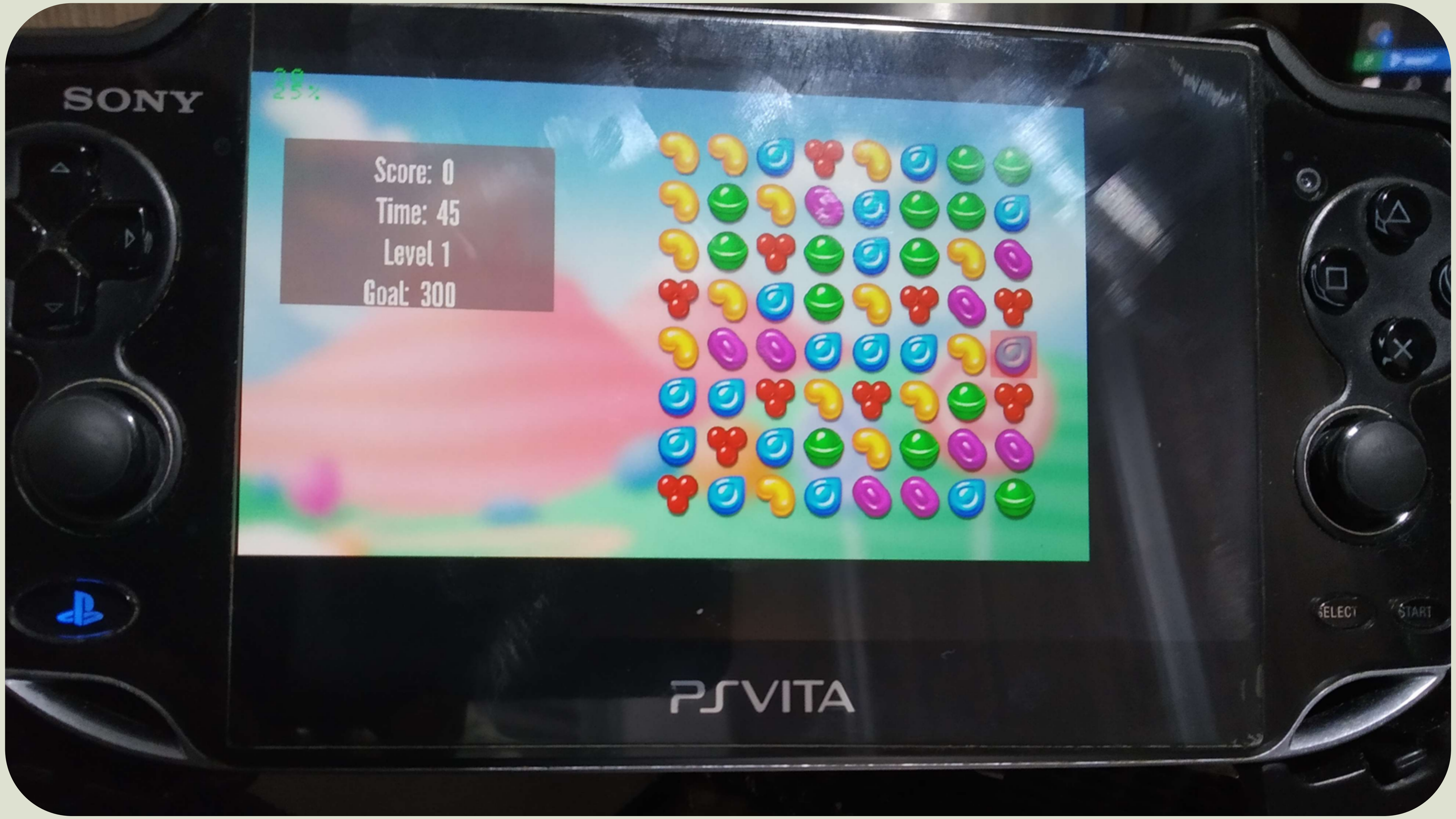
```
HIP: Loading Audio: sounds/pop.wav                                   VM561:3
HIP: Required path sounds/pop.wav                                    VM561:3
HIP: Testing sounds/pop.wav                                          VM561:3
HIP: Path validated.                                                 VM561:3
HIP: Testing sounds/pop.wav                                          VM561:3
HIP: Browser read start on sounds/pop.wav                            VM561:3
Fetching  sounds/pop.wav                                             fs.js:9
HIP: Loading Audio: sounds/song17mono.mp3                            VM561:3
HIP: Required path sounds/song17mono.mp3                             VM561:3
HIP: Testing sounds/song17mono.mp3                                   VM561:3
HIP: Path validated.                                                 VM561:3
HIP: Testing sounds/song17mono.mp3                                   VM561:3
HIP: Browser read start on sounds/song17mono.mp3                     VM561:3
Fetching  sounds/song17mono.mp3                                      fs.js:9
HIP: Required path data/levels.txt                                   VM561:3
HIP: Testing data/levels.txt                                         VM561:3
HIP: Path validated.                                                 VM561:3
HIP: Testing data/levels.txt                                         VM561:3
HIP: Browser read start on data/levels.txt                           VM561:3
Fetching  data/levels.txt                                            fs.js:9
▶ERROR: at module 'hip.image' modules\image\source\hip\image.d:231(bool  VM562:3
hip.image.HipImageImpl.loadFromMemory(ubyte[] data, void delegate(IImage self)
onSuccess, void delegate() onFailure))
    Decoding Image:     [[Could not load image images/background.png]]
HIP: 1024 x 768 3145728 bytes                                        VM561:3
▶ERROR: at module 'hip.image' modules\image\source\hip\image.d:231(bool  VM562:3
hip.image.HipImageImpl.loadFromMemory(ubyte[] data, void delegate(IImage self)
onSuccess, void delegate() onFailure))
    Decoding Image:     [[Could not load image images/background_blur.png]]
▶ERROR: at module 'hip.image' modules\image\source\hip\image.d:231(bool  VM562:3
hip.image.HipImageImpl.loadFromMemory(ubyte[] data, void delegate(IImage self)
onSuccess, void delegate() onFailure))
    Decoding Image:     [[Could not load image sprites/assets_candy.png]]
▶ERROR: at module 'hip.image' modules\image\source\hip\image.d:231(bool  VM562:3
hip.image.HipImageImpl.loadFromMemory(ubyte[] data, void delegate(IImage self)
onSuccess, void delegate() onFailure))
    Decoding Image:     [[Could not load image sprites/explosion.png]]
```

Match 3. Playable at

https://hipreme.itch.io/hipmatch3

Hipreme
Engine

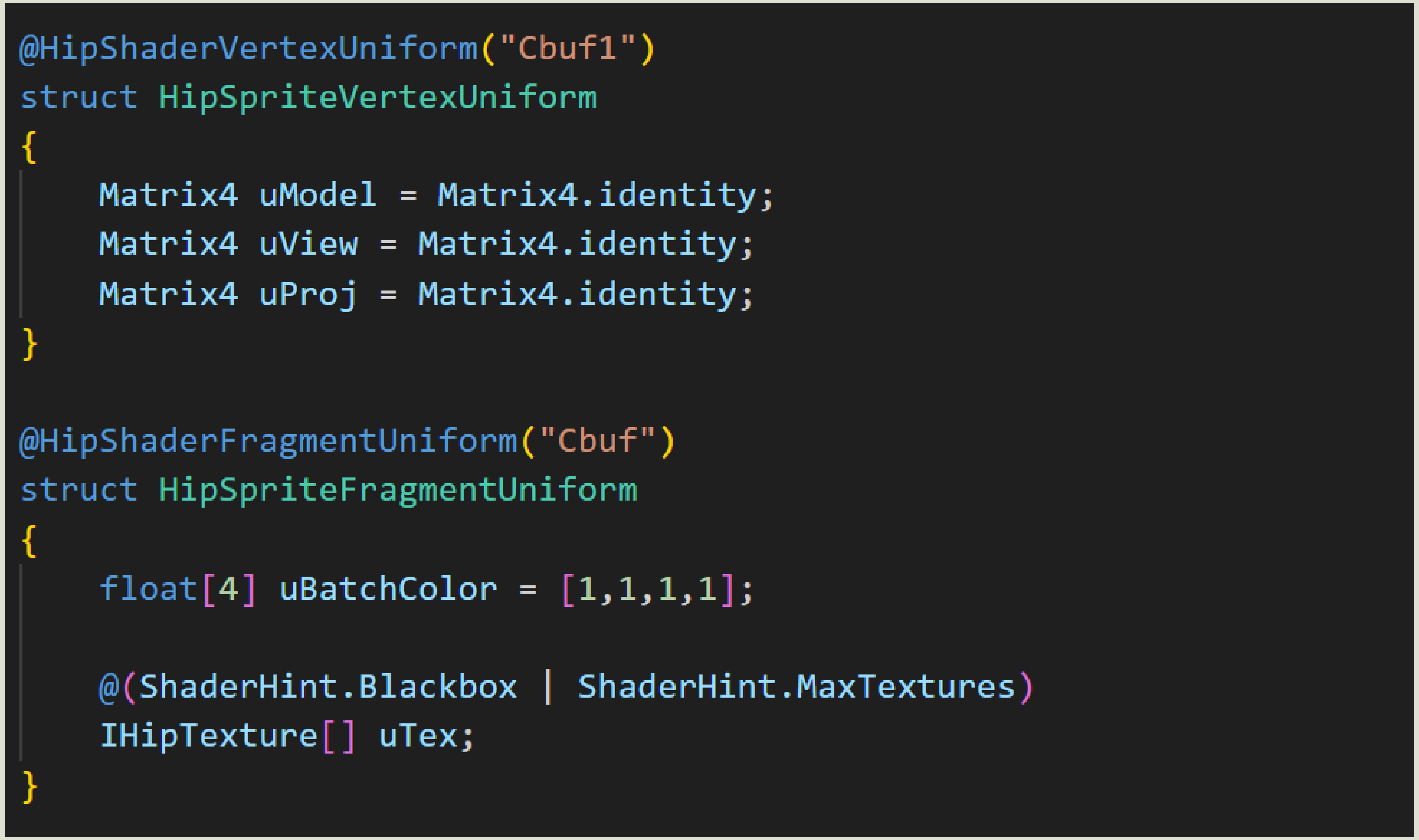
PSVITA RESULT (SAME CUSTOM RUNTIME)



PSVita Version - Same code.

PORTABILITY DESIGN

COMPILE TIME REFLECTION

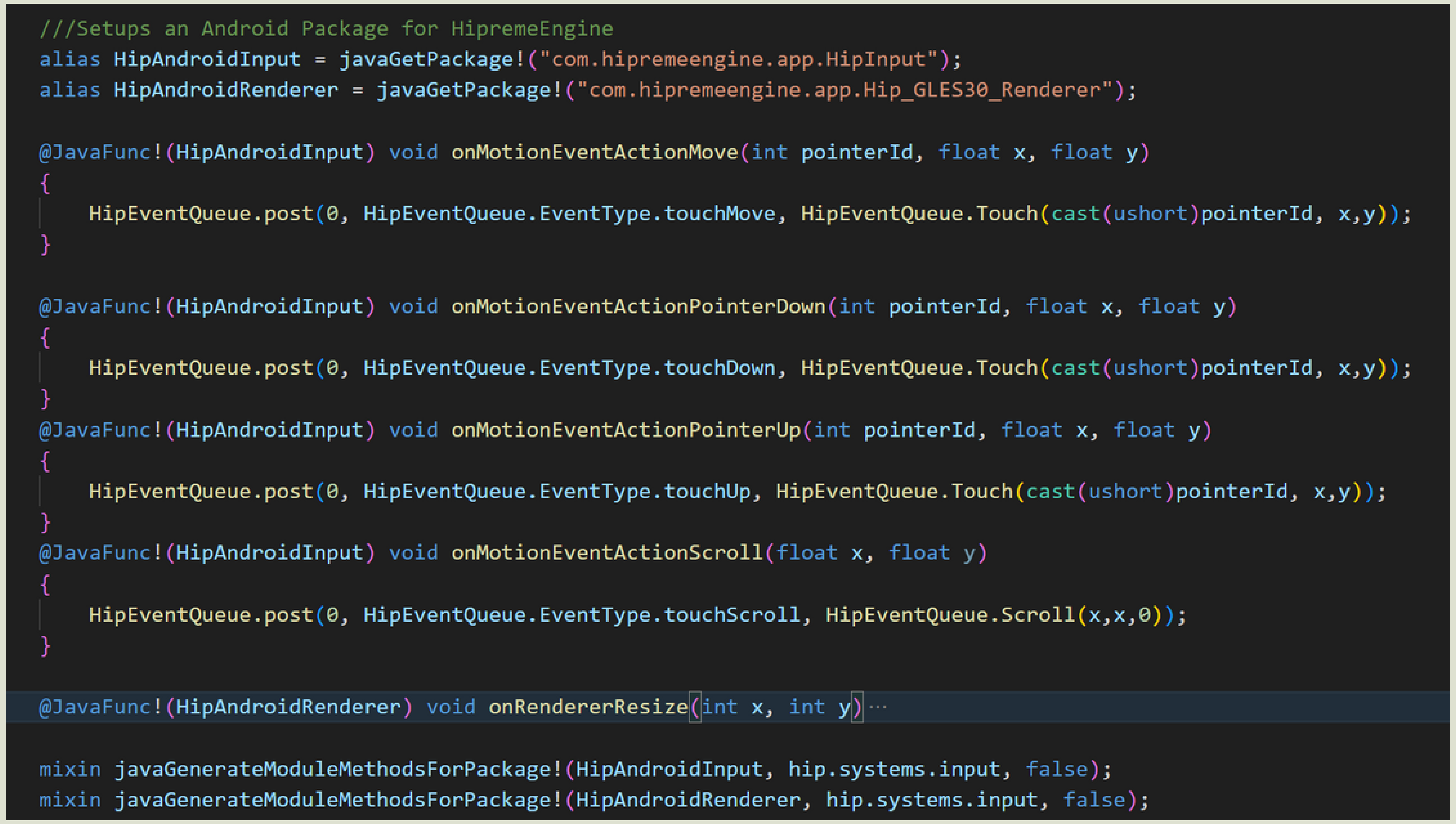


```
@HipShaderVertexUniform("Cbuf1")
struct HipSpriteVertexUniform
{
    Matrix4 uModel = Matrix4.identity;
    Matrix4 uView = Matrix4.identity;
    Matrix4 uProj = Matrix4.identity;
}

@HipShaderFragmentUniform("Cbuf")
struct HipSpriteFragmentUniform
{
    float[4] uBatchColor = [1,1,1,1];

    @(ShaderHint.Blackbox | ShaderHint.MaxTextures)
    IHipTexture[] uTex;
}
```

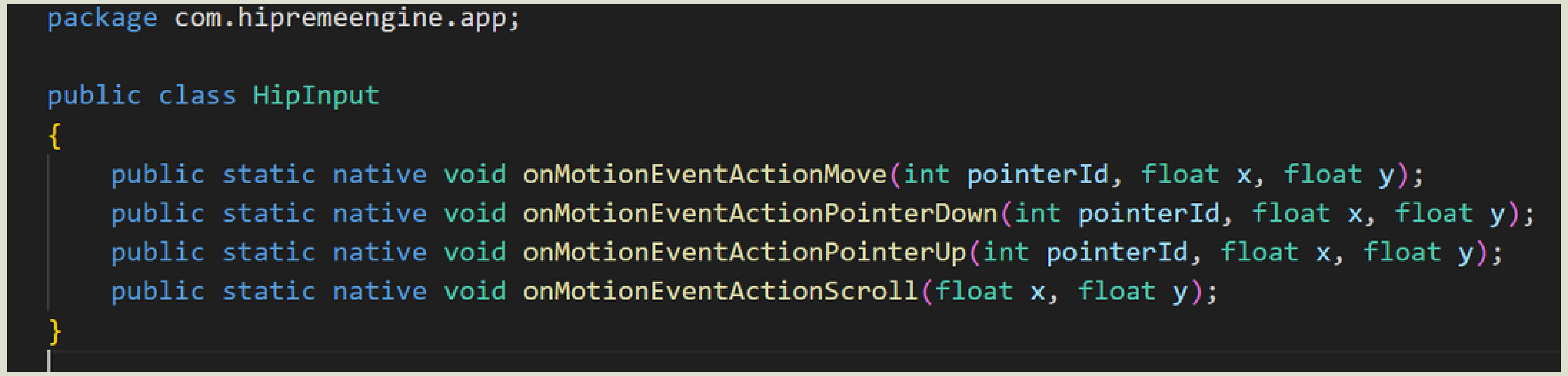
Using Reflection for Shader Vertex Buffer Descriptor

Hipreme
Engine

METAPROGRAMMING



```
///Setups an Android Package for HipremeEngine
alias HipAndroidInput = javaGetPackage!("com.hipremeengine.app.HipInput");
alias HipAndroidRenderer = javaGetPackage!("com.hipremeengine.app.Hip_GLES30_Renderer");

@JavaFunc!(HipAndroidInput) void onMotionEventActionMove(int pointerId, float x, float y)
{
    HipEventQueue.post(0, HipEventQueue.EventType.touchMove, HipEventQueue.Touch(cast(ushort)pointerId, x,y));
}

@JavaFunc!(HipAndroidInput) void onMotionEventActionPointerDown(int pointerId, float x, float y)
{
    HipEventQueue.post(0, HipEventQueue.EventType.touchDown, HipEventQueue.Touch(cast(ushort)pointerId, x,y));
}
@JavaFunc!(HipAndroidInput) void onMotionEventActionPointerUp(int pointerId, float x, float y)
{
    HipEventQueue.post(0, HipEventQueue.EventType.touchUp, HipEventQueue.Touch(cast(ushort)pointerId, x,y));
}
@JavaFunc!(HipAndroidInput) void onMotionEventActionScroll(float x, float y)
{
    HipEventQueue.post(0, HipEventQueue.EventType.touchScroll, HipEventQueue.Scroll(x,x,0));
}

@JavaFunc!(HipAndroidRenderer) void onRendererResize(int x, int y)···

mixin javaGenerateModuleMethodsForPackage!(HipAndroidInput, hip.systems.input, false);
mixin javaGenerateModuleMethodsForPackage!(HipAndroidRenderer, hip.systems.input, false);
```

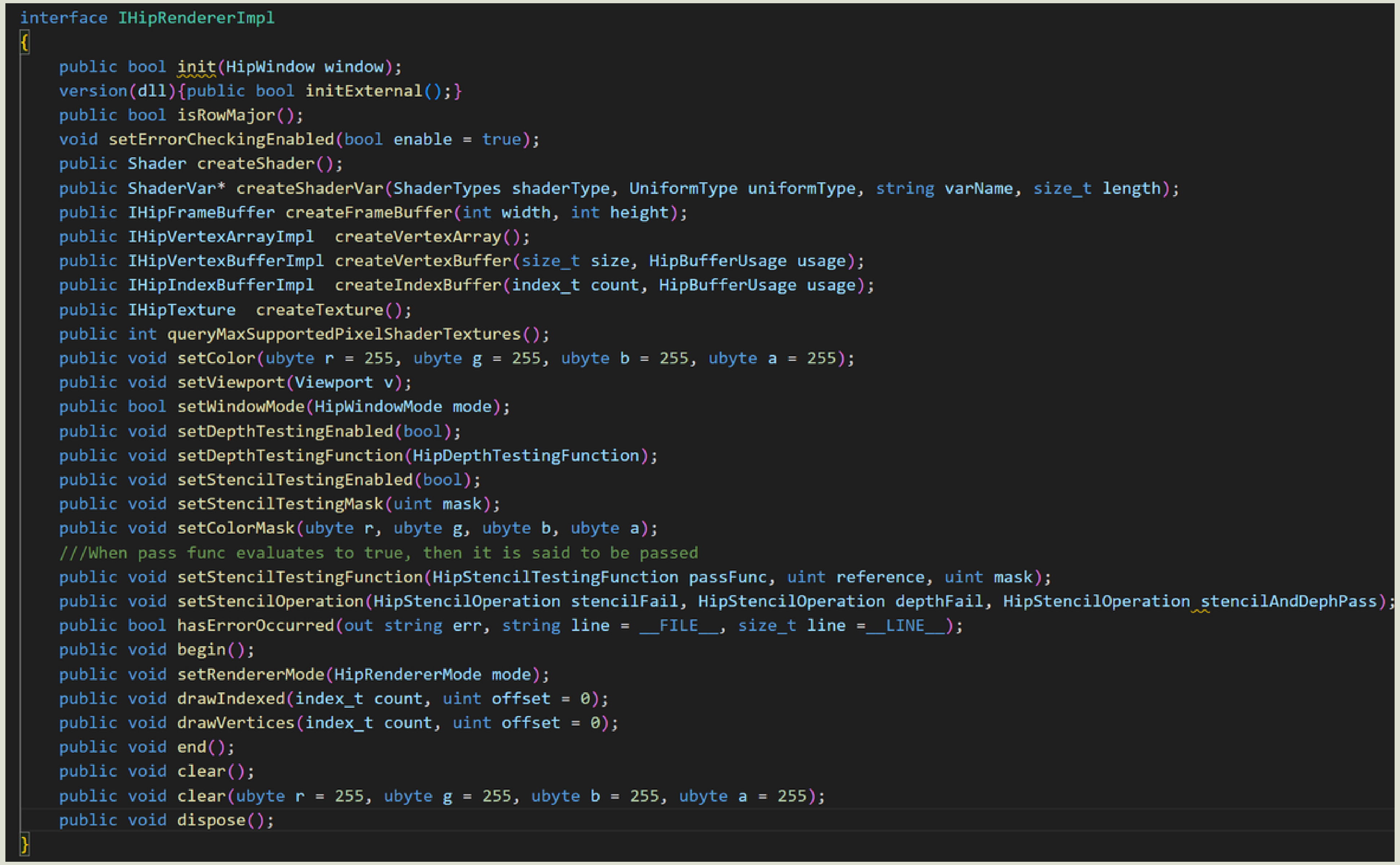


```
package com.hipremeengine.app;

public class HipInput
{
    public static native void onMotionEventActionMove(int pointerId, float x, float y);
    public static native void onMotionEventActionPointerDown(int pointerId, float x, float y);
    public static native void onMotionEventActionPointerUp(int pointerId, float x, float y);
    public static native void onMotionEventActionScroll(float x, float y);
}
```

Using Reflection + Metaprogramming for binding to Java.

Hipreme
Engine

API POLYMORPHISM

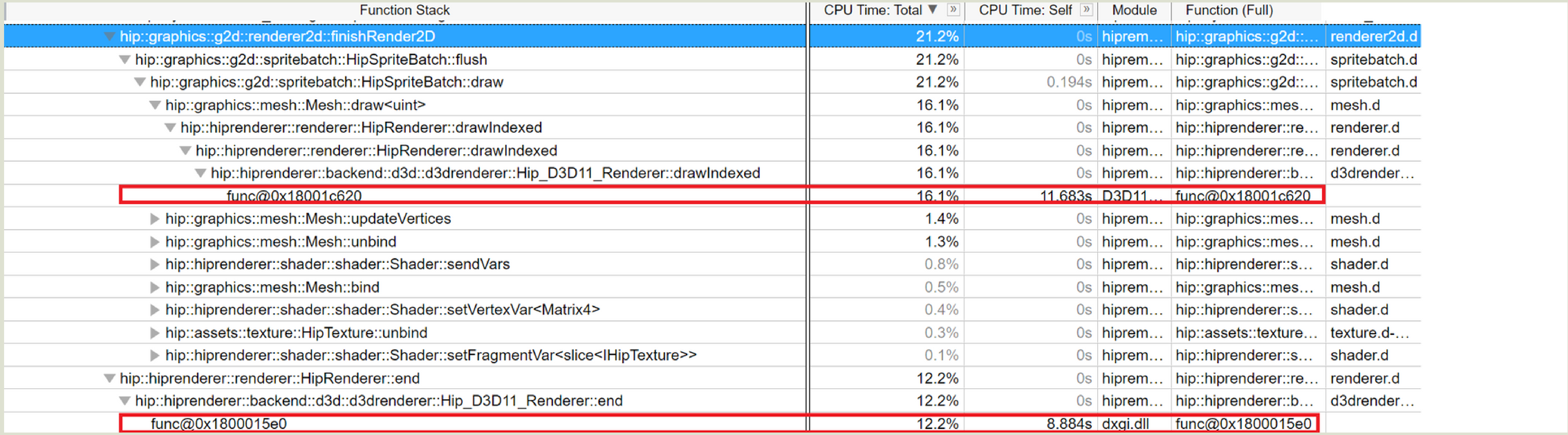


```
interface IHipRendererImpl
{
    public bool init(HipWindow window);
    version(dll){public bool initExternal();}
    public bool isRowMajor();
    void setErrorCheckingEnabled(bool enable = true);
    public Shader createShader();
    public ShaderVar* createShaderVar(ShaderTypes shaderType, UniformType uniformType, string varName, size_t length);
    public IHipFrameBuffer createFrameBuffer(int width, int height);
    public IHipVertexArrayImpl  createVertexArray();
    public IHipVertexBufferImpl createVertexBuffer(size_t size, HipBufferUsage usage);
    public IHipIndexBufferImpl  createIndexBuffer(index_t count, HipBufferUsage usage);
    public IHipTexture  createTexture();
    public int queryMaxSupportedPixelShaderTextures();
    public void setColor(ubyte r = 255, ubyte g = 255, ubyte b = 255, ubyte a = 255);
    public void setViewport(Viewport v);
    public bool setWindowMode(HipWindowMode mode);
    public void setDepthTestingEnabled(bool);
    public void setDepthTestingFunction(HipDepthTestingFunction);
    public void setStencilTestingEnabled(bool);
    public void setStencilTestingMask(uint mask);
    public void setColorMask(ubyte r, ubyte g, ubyte b, ubyte a);
    ///When pass func evaluates to true, then it is said to be passed
    public void setStencilTestingFunction(HipStencilTestingFunction passFunc, uint reference, uint mask);
    public void setStencilOperation(HipStencilOperation stencilFail, HipStencilOperation depthFail, HipStencilOperation stencilAndDephPass);
    public bool hasErrorOccurred(out string err, string line = __FILE__, size_t line =__LINE__);
    public void begin();
    public void setRendererMode(HipRendererMode mode);
    public void drawIndexed(index_t count, uint offset = 0);
    public void drawVertices(index_t count, uint offset = 0);
    public void end();
    public void clear();
    public void clear(ubyte r = 255, ubyte g = 255, ubyte b = 255, ubyte a = 255);
    public void dispose();
}
```

Interface describing what is needed to implement a new renderer.

Hipreme
Engine

IS OOP INHERENTLY SLOW?



Function Stack	CPU Time: Total ▼	CPU Time: Self	Module	Function (Full)	
▼ hip::graphics::g2d::renderer2d::finishRender2D	21.2%	0s	hiprem…	hip::graphics::g2d::…	renderer2d.d
▼ hip::graphics::g2d::spritebatch::HipSpriteBatch::flush	21.2%	0s	hiprem…	hip::graphics::g2d::…	spritebatch.d
▼ hip::graphics::g2d::spritebatch::HipSpriteBatch::draw	21.2%	0.194s	hiprem…	hip::graphics::g2d::…	spritebatch.d
▼ hip::graphics::mesh::Mesh::draw<uint>	16.1%	0s	hiprem…	hip::graphics::mes…	mesh.d
▼ hip::hiprenderer::renderer::HipRenderer::drawIndexed	16.1%	0s	hiprem…	hip::hiprenderer::re…	renderer.d
▼ hip::hiprenderer::renderer::HipRenderer::drawIndexed	16.1%	0s	hiprem…	hip::hiprenderer::re…	renderer.d
▼ hip::hiprenderer::backend::d3d::d3drenderer::Hip_D3D11_Renderer::drawIndexed	16.1%	0s	hiprem…	hip::hiprenderer::b…	d3drender…
func@0x18001c620	16.1%	11.683s	D3D11…	func@0x18001c620	
▶ hip::graphics::mesh::Mesh::updateVertices	1.4%	0s	hiprem…	hip::graphics::mes…	mesh.d
▶ hip::graphics::mesh::Mesh::unbind	1.3%	0s	hiprem…	hip::graphics::mes…	mesh.d
▶ hip::hiprenderer::shader::shader::Shader::sendVars	0.8%	0s	hiprem…	hip::hiprenderer::s…	shader.d
▶ hip::graphics::mesh::Mesh::bind	0.5%	0s	hiprem…	hip::graphics::mes…	mesh.d
▶ hip::hiprenderer::shader::shader::Shader::setVertexVar<Matrix4>	0.4%	0s	hiprem…	hip::hiprenderer::s…	shader.d
▶ hip::assets::texture::HipTexture::unbind	0.3%	0s	hiprem…	hip::assets::texture…	texture.d-…
▶ hip::hiprenderer::shader::shader::Shader::setFragmentVar<slice<IHipTexture>>	0.1%	0s	hiprem…	hip::hiprenderer::s…	shader.d
▼ hip::hiprenderer::renderer::HipRenderer::end	12.2%	0s	hiprem…	hip::hiprenderer::re…	renderer.d
▼ hip::hiprenderer::backend::d3d::d3drenderer::Hip_D3D11_Renderer::end	12.2%	0s	hiprem…	hip::hiprenderer::b…	d3drender…
func@0x1800015e0	12.2%	8.884s	dxgi.dll	func@0x1800015e0	

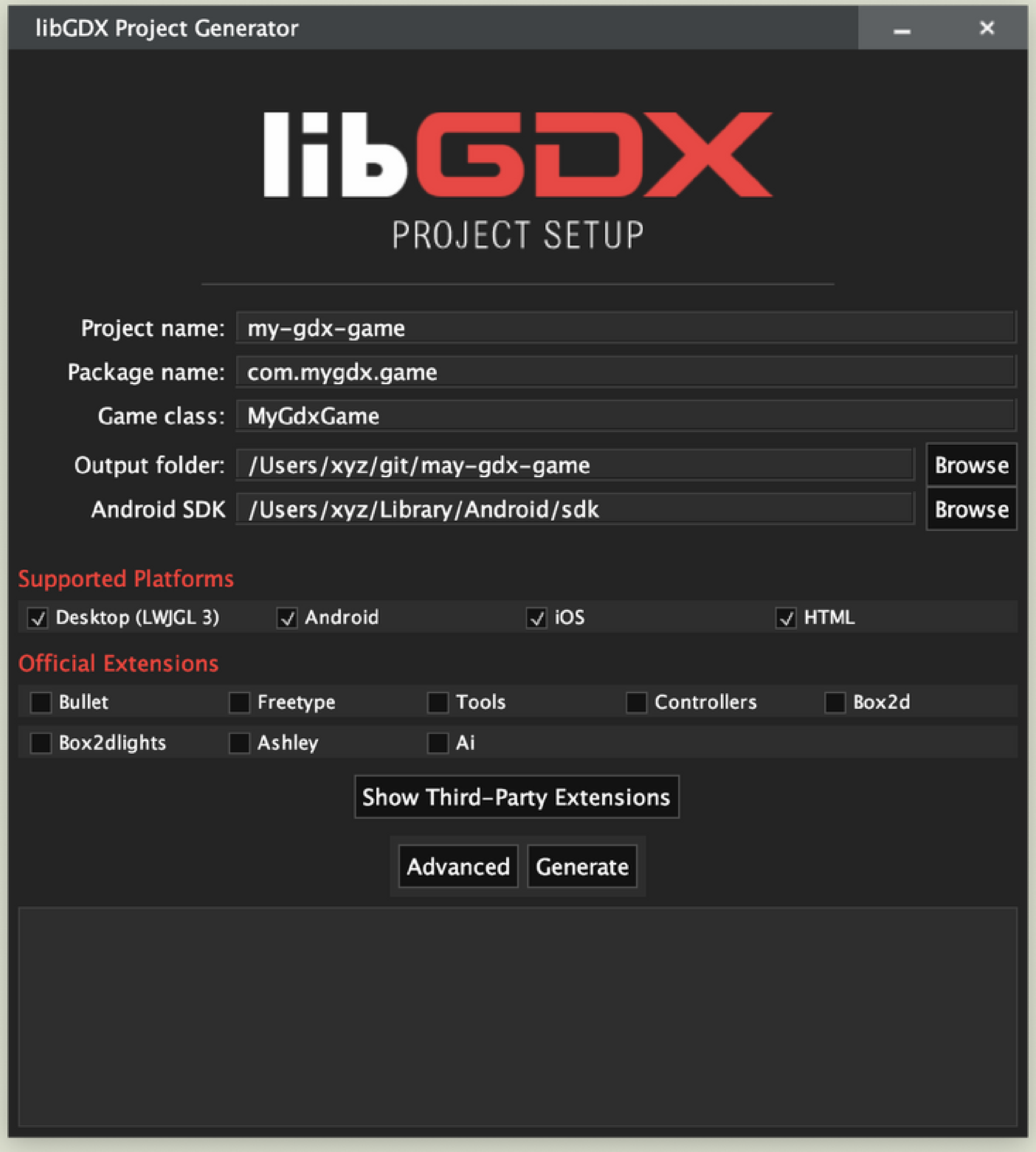
Intel VTune Profiler Statistics for Hipreme Engine.
Running for ~1 minute, 10K Dynamic Sprites.

Hipreme
Engine

BUILD AUTOMATION TOOL

Hipreme
Engine

MAIN REFERENCE



Project name: my-gdx-game
Package name: com.mygdx.game
Game class: MyGdxGame
Output folder: /Users/xyz/git/may-gdx-game
Android SDK /Users/xyz/Library/Android/sdk

Supported Platforms

☑ Desktop (LWJGL 3) ☑ Android ☑ iOS ☑ HTML

Official Extensions

□ Bullet □ Freetype □ Tools □ Controllers □ Box2d
□ Box2dlights □ Ashley □ Ai

libGDX Project Generator Tool

✓ POSITIVES:

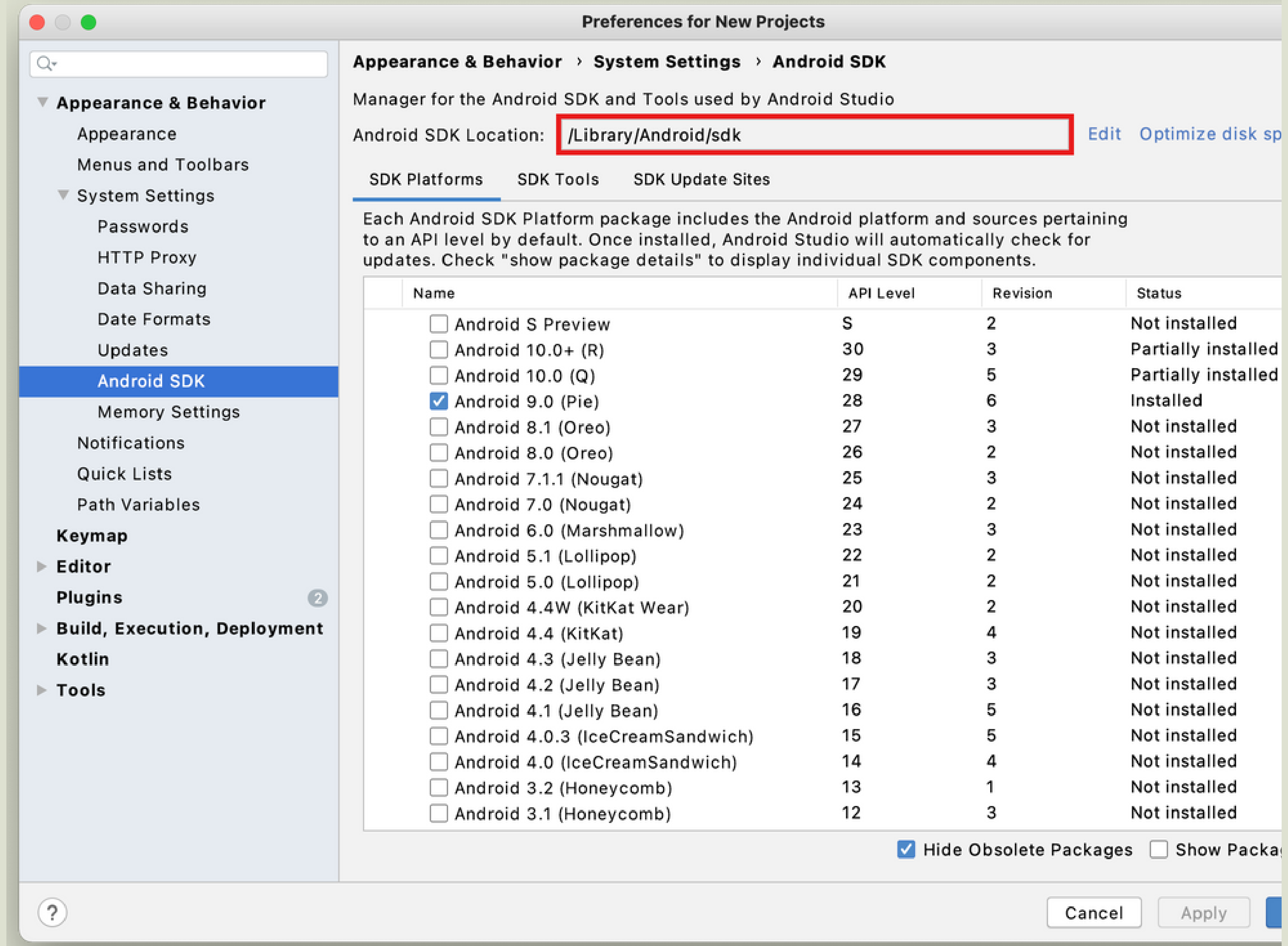
- PORTABLE
- EASY CONFIGURATION;

✗ NEGATIVES:

- NEEDS JAVA RUNTIME;
- NO ANDROID SDK INSTALL;
- NEEDS ENVIRONMENT VAR.

Hipreme Engine

ANDROID SDK INSTALLATION



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: /Library/Android/sdk Edit Optimize disk sp

SDK Platforms | SDK Tools | SDK Update Sites

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
☐ Android S Preview	S	2	Not installed
☐ Android 10.0+ (R)	30	3	Partially installed
☐ Android 10.0 (Q)	29	5	Partially installed
☑ Android 9.0 (Pie)	28	6	Installed
☐ Android 8.1 (Oreo)	27	3	Not installed
☐ Android 8.0 (Oreo)	26	2	Not installed
☐ Android 7.1.1 (Nougat)	25	3	Not installed
☐ Android 7.0 (Nougat)	24	2	Not installed
☐ Android 6.0 (Marshmallow)	23	3	Not installed
☐ Android 5.1 (Lollipop)	22	2	Not installed
☐ Android 5.0 (Lollipop)	21	2	Not installed
☐ Android 4.4W (KitKat Wear)	20	2	Not installed
☐ Android 4.4 (KitKat)	19	4	Not installed
☐ Android 4.3 (Jelly Bean)	18	3	Not installed
☐ Android 4.2 (Jelly Bean)	17	3	Not installed
☐ Android 4.1 (Jelly Bean)	16	5	Not installed
☐ Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
☐ Android 4.0 (IceCreamSandwich)	14	4	Not installed
☐ Android 3.2 (Honeycomb)	13	1	Not installed
☐ Android 3.1 (Honeycomb)	12	3	Not installed

☑ Hide Obsolete Packages ☐ Show Packa

Android Studio's SDK/NDK Installation

✅ POSITIVES:

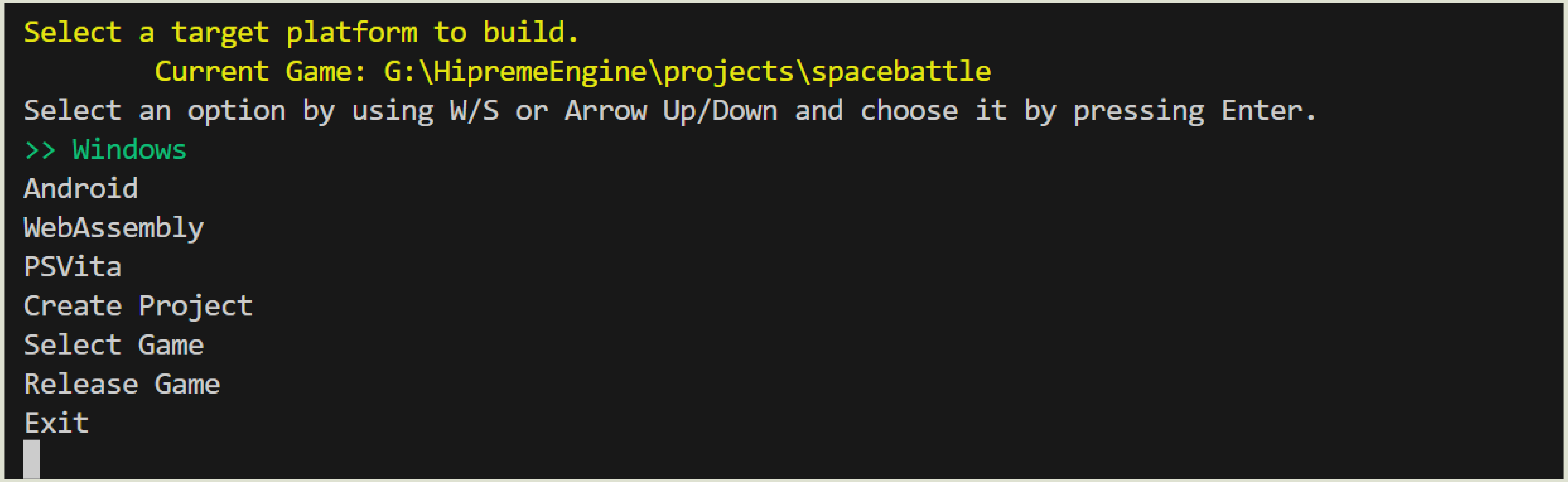
- WORKS WELL;

❌ NEGATIVES:

- NEEDS JAVA RUNTIME;
- REQUIRES ANDROID STUDIO;
- HARD TO BEGINNERS.

Hipreme
Engine

HIPREME ENGINE BUILD SELECTOR

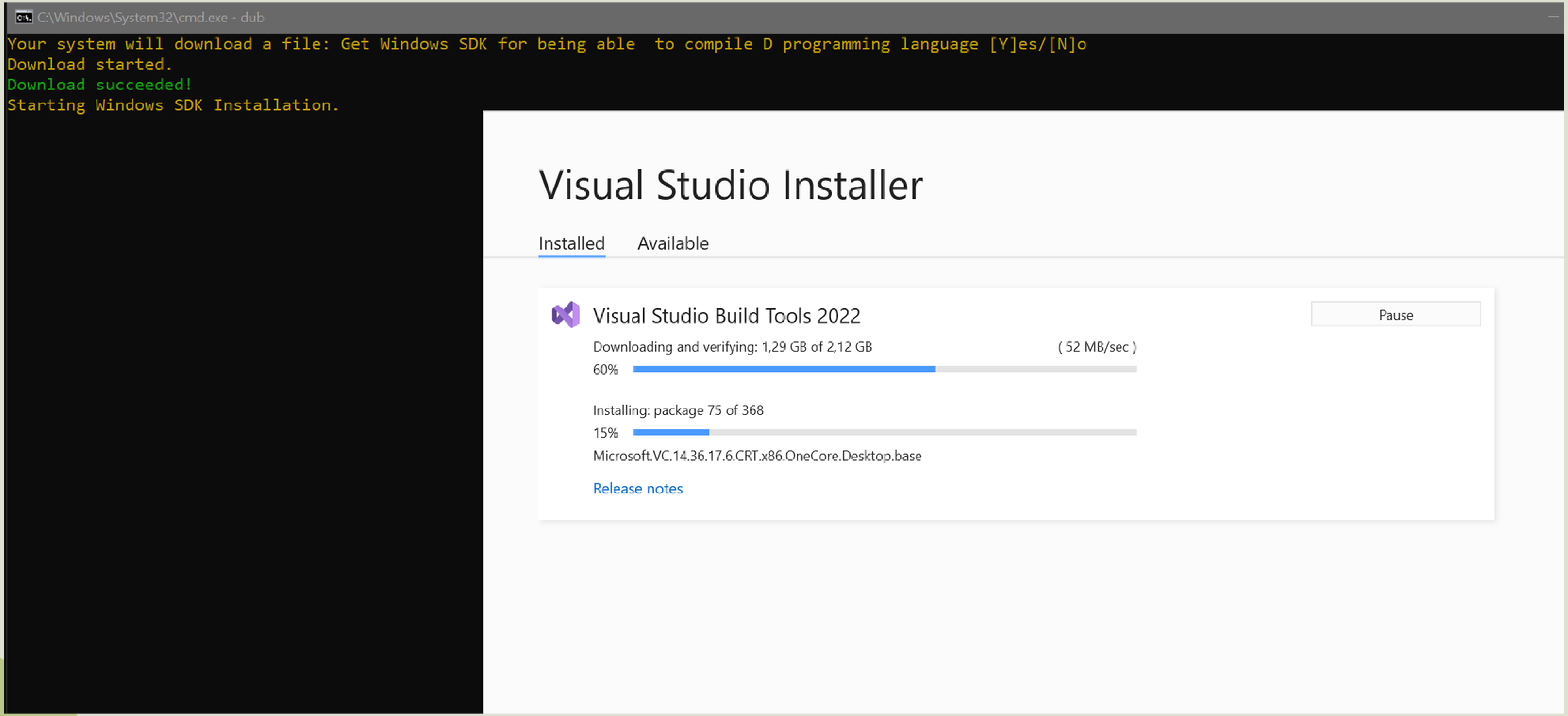


```
Select a target platform to build.
        Current Game: G:\HipremeEngine\projects\spacebattle
Select an option by using W/S or Arrow Up/Down and choose it by pressing Enter.
>> Windows
Android
WebAssembly
PSVita
Create Project
Select Game
Release Game
Exit
```

- Integrated project generator
- Android SDK Tools Auto Install
- Target DFLAGS Auto Configuration
- Java Runtime Auto Download
- Automatic D Compiler Install (LDC)
- Prebuilt Binary Distributed

- Easy project selector
- MSVC Runtime Auto Install
- MSBuild Auto Install
- Compatible with OSX, Windows and Linux
- Uses no Environment Var
- External Configuration File

HIPREME ENGINE BUILD SELECTOR



```
Your system will download a file: Get Windows SDK for being able  to compile D programming language [Y]es/[N]o
Download started.
Download succeeded!
Starting Windows SDK Installation.
```

Visual Studio Installer

Visual Studio Build Tools 2022

Downloading and verifying: 1,29 GB of 2,12 GB (52 MB/sec)

60%

Installing: package 75 of 368

15%

Microsoft.VC.14.36.17.6.CRT.x86.OneCore.Desktop.base

MSBuild Installation, required for specific DFLAGS.

Hipreme
Engine

THE FUTURE

Hipreme
Engine

SHORT TERM

- GAMES DEVELOPMENT;
- INCREASE VISIBILITY IN D;
- IOS PORT: EXTERN OBJECTIVE-C;
- DOCS;
- GAME FRAMEWORK.

Hipreme Engine

MID-LONG TERM

- ENGINE UI DEVELOPMENT;
- PS4/5 PORT;
- VISIBILITY OUTSIDE D;
- 3D

ACKNOWLEGEMENTS

QUESTIONS ?

Hipreme
Engine

THANKS FOR WATCHING!