

COCOS CREATOR 原生框架与绑定技术



by panda

主播集结令

征集在野 Cocos 项目开发者分享
自己的 Cocos 技巧、项目经验、
技术方案、游戏设计思路等

详情咨询微信号: *applesjump*



简介

- 热更新技术解惑
- Cocos Creator 的原生框架
- JSB 绑定技术
- 自动绑定环境配置
- 手动绑定简介
- C++ 调用 JS 方法简介
- 有抽奖～有抽奖～有抽奖～



热更新



运行热更新范例

.....

- 下载 tutorial hot update
- 搭建本地服务器
- 修改 local/remote manifest 的 url
- 打包原生版本
- 修改 main.js
- 运行范例



热更新技术解惑

.....

- 基本原理：浏览器对页面的更新
- 动态生成差异列表并更新
- 支持断点续传，并发下载等
- 无需也不支持增量包方式
- 两级进度：文件级和自节级
- 1.4 版本将改善稳定性

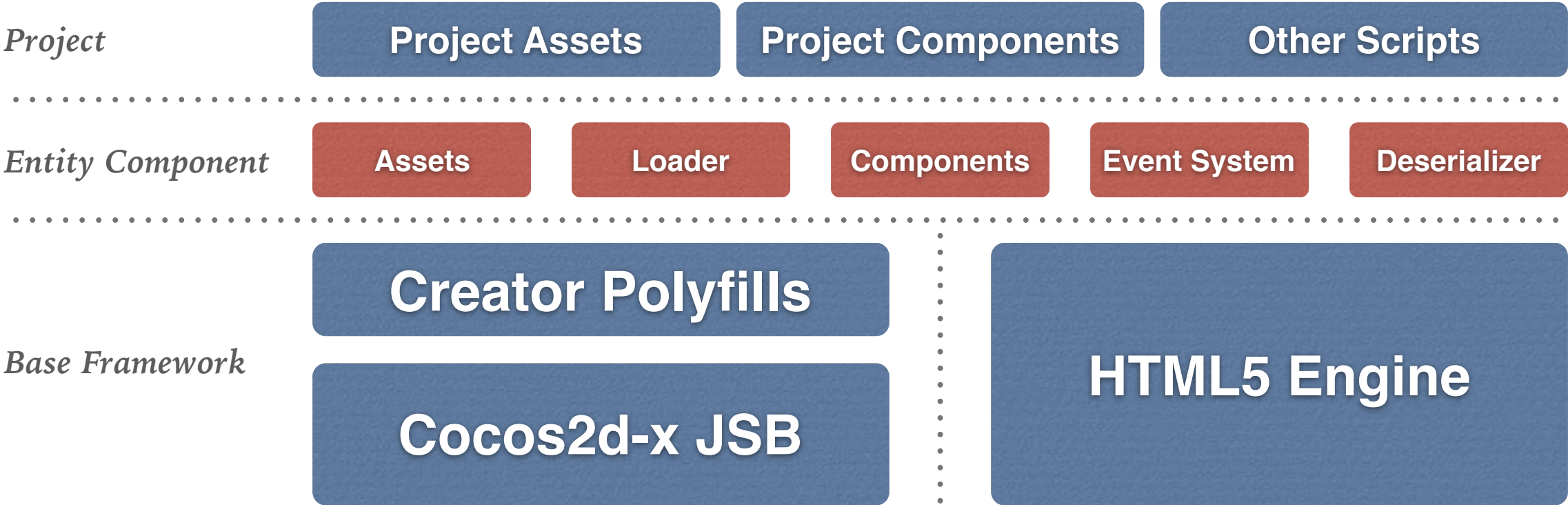


COCOS CREATOR 框架解析

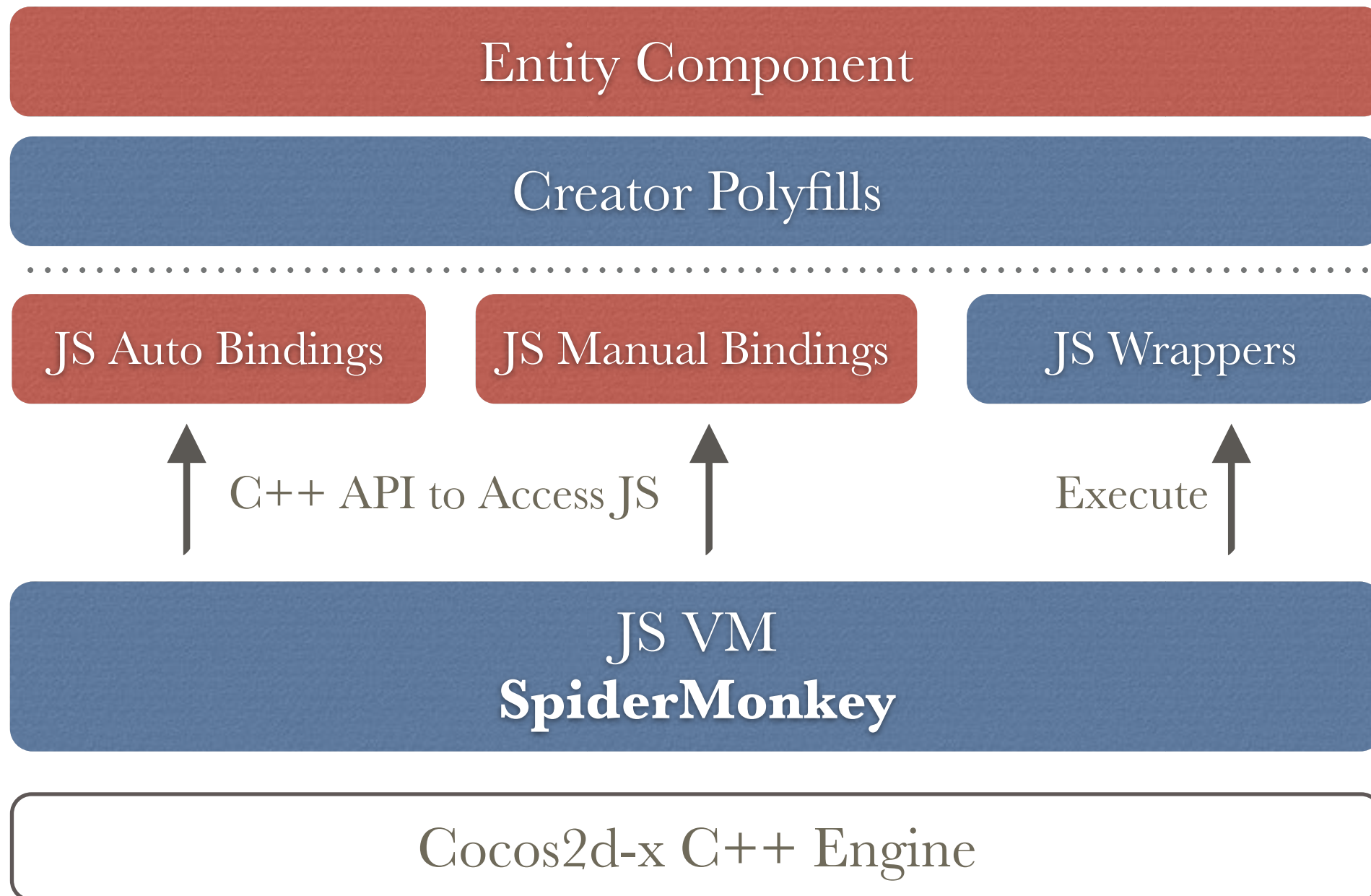


COCOS CREATOR 框架

.....



COCOS CREATOR 框架



COCOS CREATOR 扩展方式

.....

- JS 层扩展（可跨平台）
- 定制 HTML5 和 C++ 引擎
- 依赖 DOM 和浏览器 API 的扩展只可用于 WEB 平台
- 依赖文件读写，数据库，基础网络库扩展等功能只能在原生层实现



抽奖

.....



JSB 绑定技术



JSB 绑定技术

.....

- 将 C++ API 桥接到 JS 层 API
- JS API + 原生性能
- 强大的跨平台能力
- 绑定层 C++ API
- 自动绑定和手动绑定



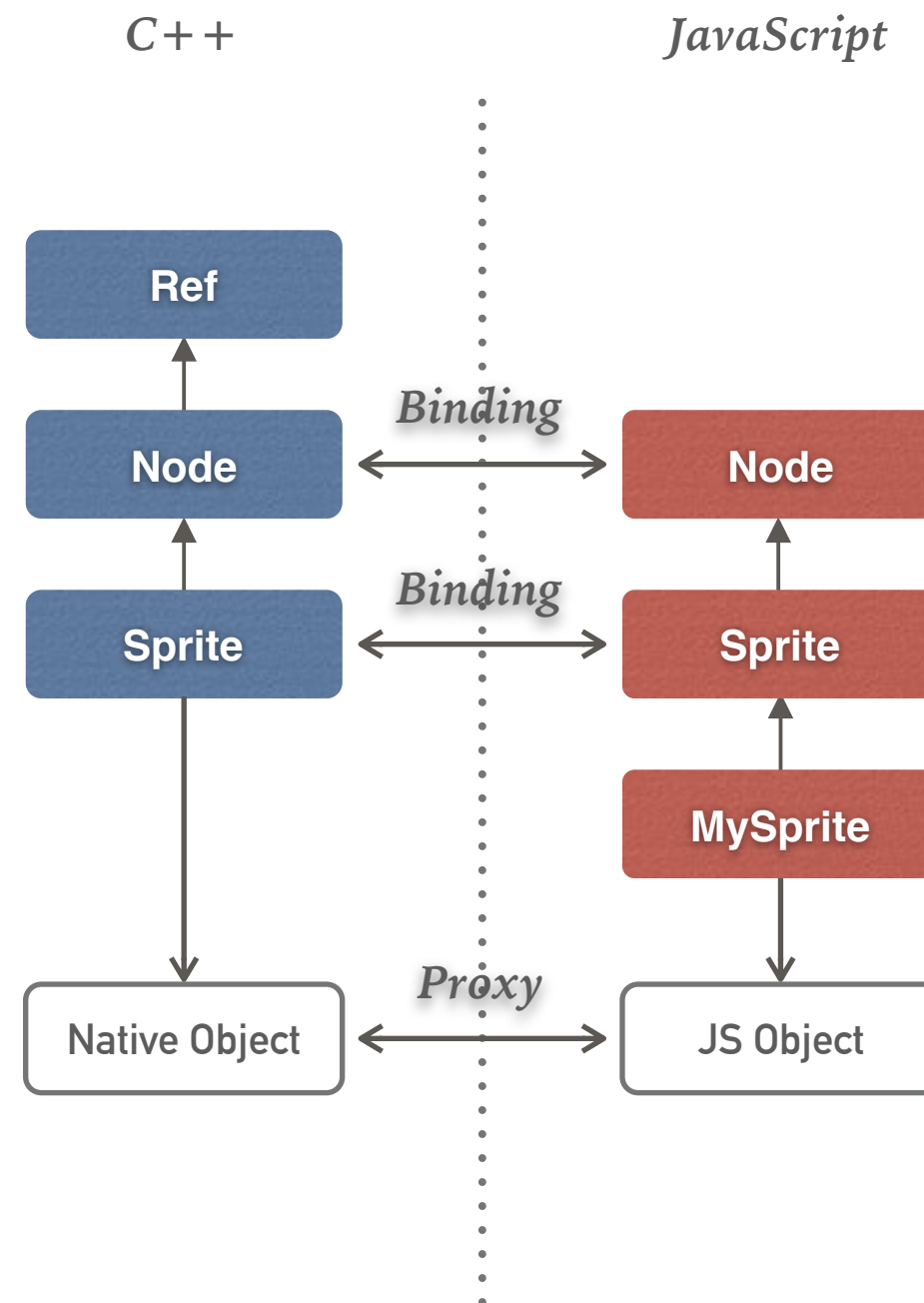
BINDINGS GENERATOR

.....

- 基于 Python 2.7
- windows 用户用 Python 32 bit
- 解析 C++ 类信息并以类为单位进行绑定
- 绑定函数
- 绑定属性 Getter/Setter
- 对象转换
- 类对象映射关系



类对象映射关系



自动绑定环境配置



手动绑定

.....

- 和自动绑定一样，手动写的绑定函数就是手动绑定~
- 因为自动绑定有一些限制：
 - 只能够针对类生成绑定，不支持结构体，独立函数
 - `void *` 类型参数或返回值
 - 不能够生成 Delegate 类型的 API
 - 子类中重写了父类的 API 的同时，又重载了这个 API
 - 一些未知的限制



手动绑定需要了解的技术

.....

- 接口绑定原理
- Spidermonkey API
- ScriptingCore API
- 对象转换原理
- 理解 JSB 内存模型



抽奖

.....



C++ 调用 JS 方法实例



实例

Simple way

```
auto engine = ScriptingCore::getInstance();
engine->evalString("cc.find('Canvas/Label').getComponent('cc.Label').string = 'Hello Cocos Live Fans ~'");
```

=

Complicated but powerful way

```
auto engine = ScriptingCore::getInstance();
auto cx = engine->getGlobalContext();

// var node = cc.find('Canvas/Label')
JS::RootedValue node(cx);
engine->evalString("cc.find('Canvas/Label')", &node);

// var label = node.getComponent('cc.Label')
JS::RootedValue label(cx);
jsval argsArr[1];
argsArr[0] = std_string_to_jsval(cx, "cc.Label");
JS::HandleValueArray args =
JS::HandleValueArray::fromMarkedLocation(1, argsArr);
engine->executeFunctionWithOwner(node,
"getComponent", args, &label);

// label.string = 'Hello Cocos Live Fans ~'
JS::RootedObject labelObj(cx,
label.toObjectOrNull());
JS::RootedValue strVal(cx, std_string_to_jsval(cx,
"Hello Cocos Live Fans ~"));
JS_SetProperty(cx, labelObj, "string", strVal);
```



资料

- 热更新范例: <https://github.com/cocos-creator/tutorial-hot-update>
- 热更新旧文档: <https://github.com/chukong/cocos-docs/blob/v3-unified-documentation/manual/framework/html5/v3/assets-manager/zh.md>
- Cocos2d 的绑定技术: <https://zhuanlan.zhihu.com/p/20525026>
- Cocos Creator 的自动绑定: <https://zhuanlan.zhihu.com/p/20525109>
- Cocos Creator 1.3 新内存模型: <https://zhuanlan.zhihu.com/p/21256591>
- 自动绑定工具及配置: <https://github.com/cocos2d/bindings-generator/>
- pyYaml 插件: <https://pypi.python.org/pypi/PyYAML/3.12>
- pyCheetah 插件: <https://pypi.python.org/pypi/Cheetah/2.4.4>
- Spidermonkey API: https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey/JSAPI_reference
- ScriptingCore API: http://cocos2d-x.org/docs/api-ref/cplusplus/v3x/dc/da6/class_scripting_core.html



预告

如何在2小时内完成一个
高逼格的插件？

下周二 20:00 ~ 22:00
by VisualSJ

