

链家组件化路由方案解析

陈曦



目录



- 1. 组件化起因
- 2. 业界方案
- 3. LJRouter 使用
- 4. LJRouter 实现
- 5. 优势与问题





• 1. 组件化起因



1.组件化起因

1.1既有问题

- 既有组件化成本略高
- 同一App多条业务线
- 不同App重用页面
- 直接引用耦合过高
- Webview/Push等Url调用困难

1.组件化起因

1.2预期目标

- 方便接入
- 安全健壮
- 自由传参
- 文档校验
- 其他基本需求

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



2. 业界方案

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE

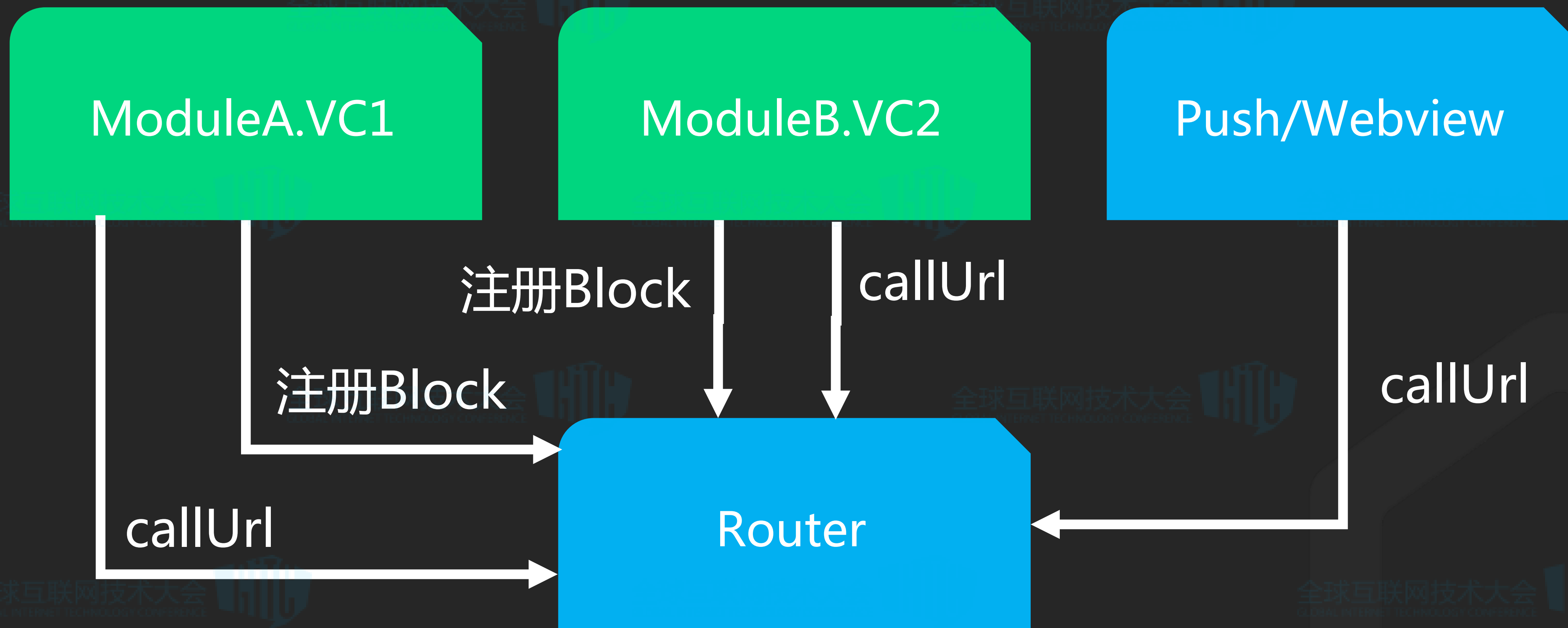


全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



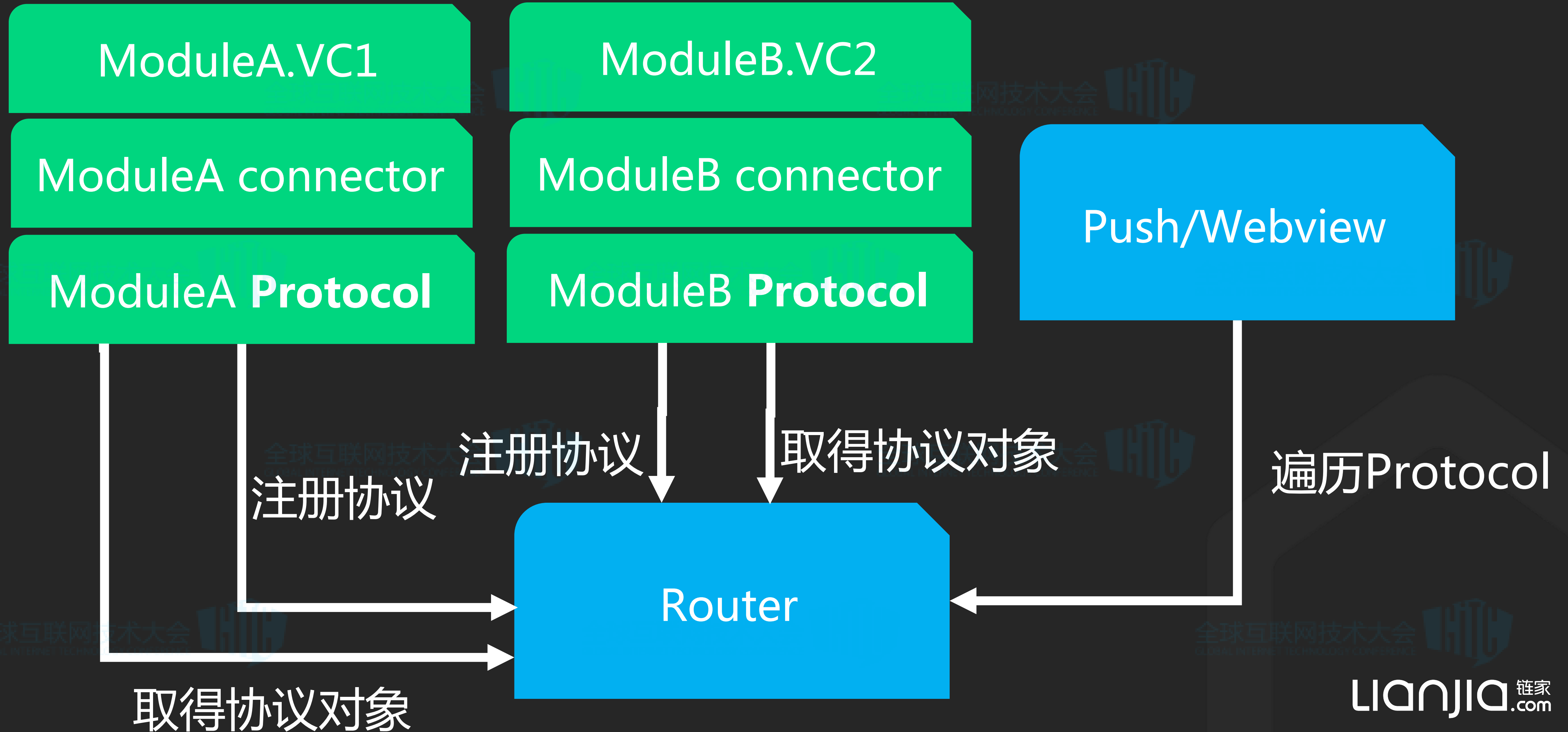
2. 业界方案

2.1 URL-Block



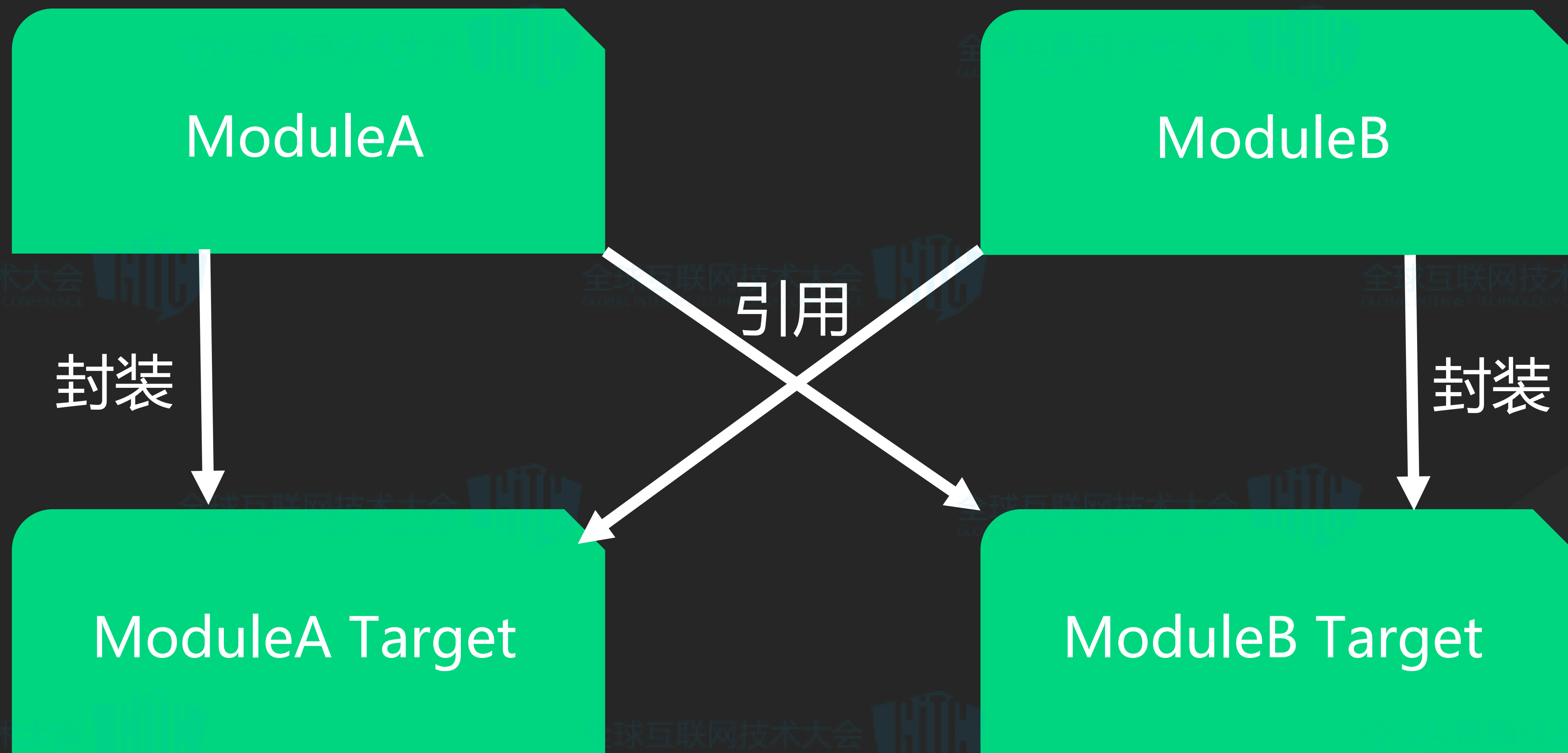
2. 业界方案

2.2 Protocol-class



2. 业界方案

2.3 Target-Action





3. LJRouter的使用



3. LJRouter的使用

```
// 注册
LJRouterPageInit(@"2017年新版首页",
                Home,
                (NSString*)title) {
    self = [super init];
    if (self) {
        self.title = title;
    }
    return self;
}

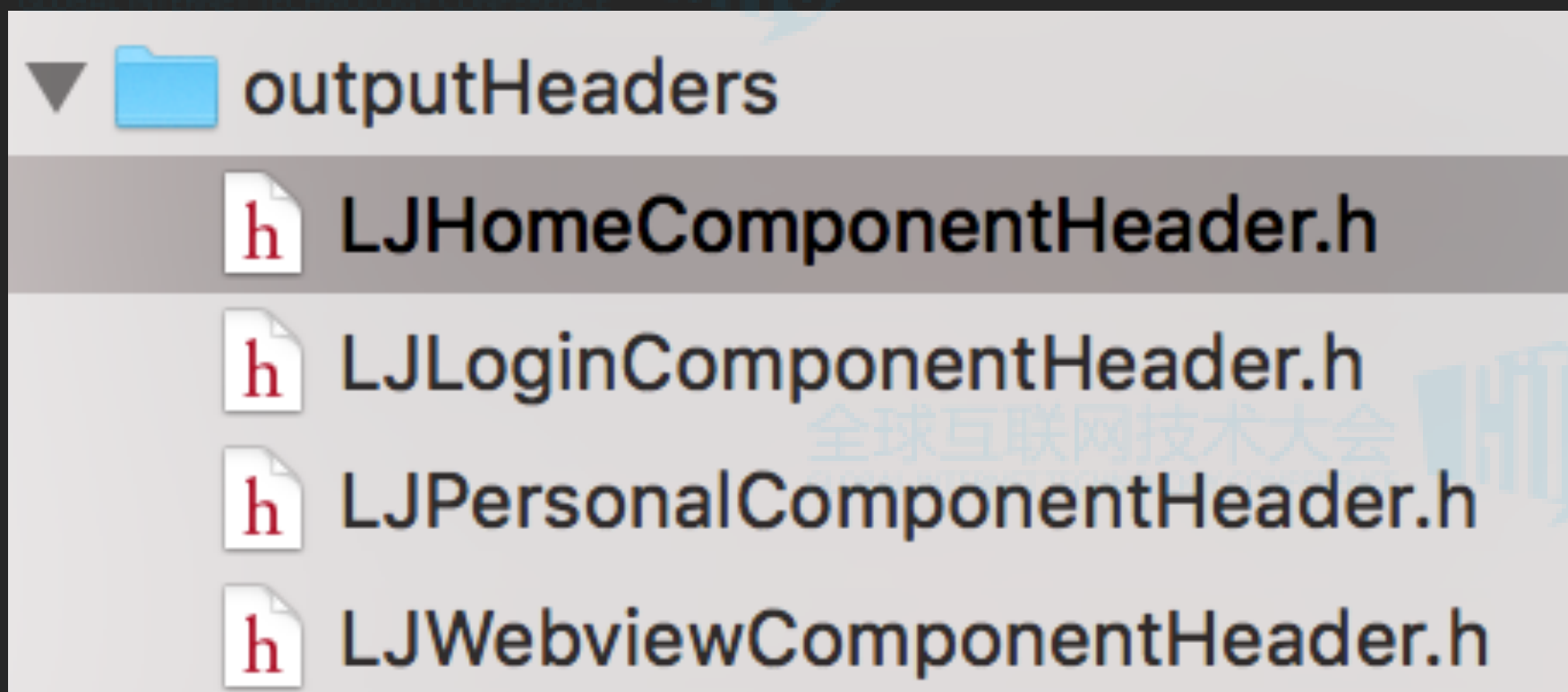
// 调用1
LJRouterUsePageObj(Home, (NSString*)title);
UIViewController *vc = get_Home_controller_with_title(@"我就是个title")

// 调用2
LJRouterUsePage(Home, (NSString*)title);
open_Home_controller_with_title(self, @"我还是个title");

// 调用3
[self openUrl:@"lianjia://home?title=我依旧是个title"];
```

3. LJRouter的使用

自动生成接口文件



```
// 页面 : // 首页tabbar  
LJRouterUsePage(homeTabbar);  
// 页面 : // 2017年新版首页  
LJRouterUsePage(Home, (NSString*)titlex);
```

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



4. LJRouter的实现

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



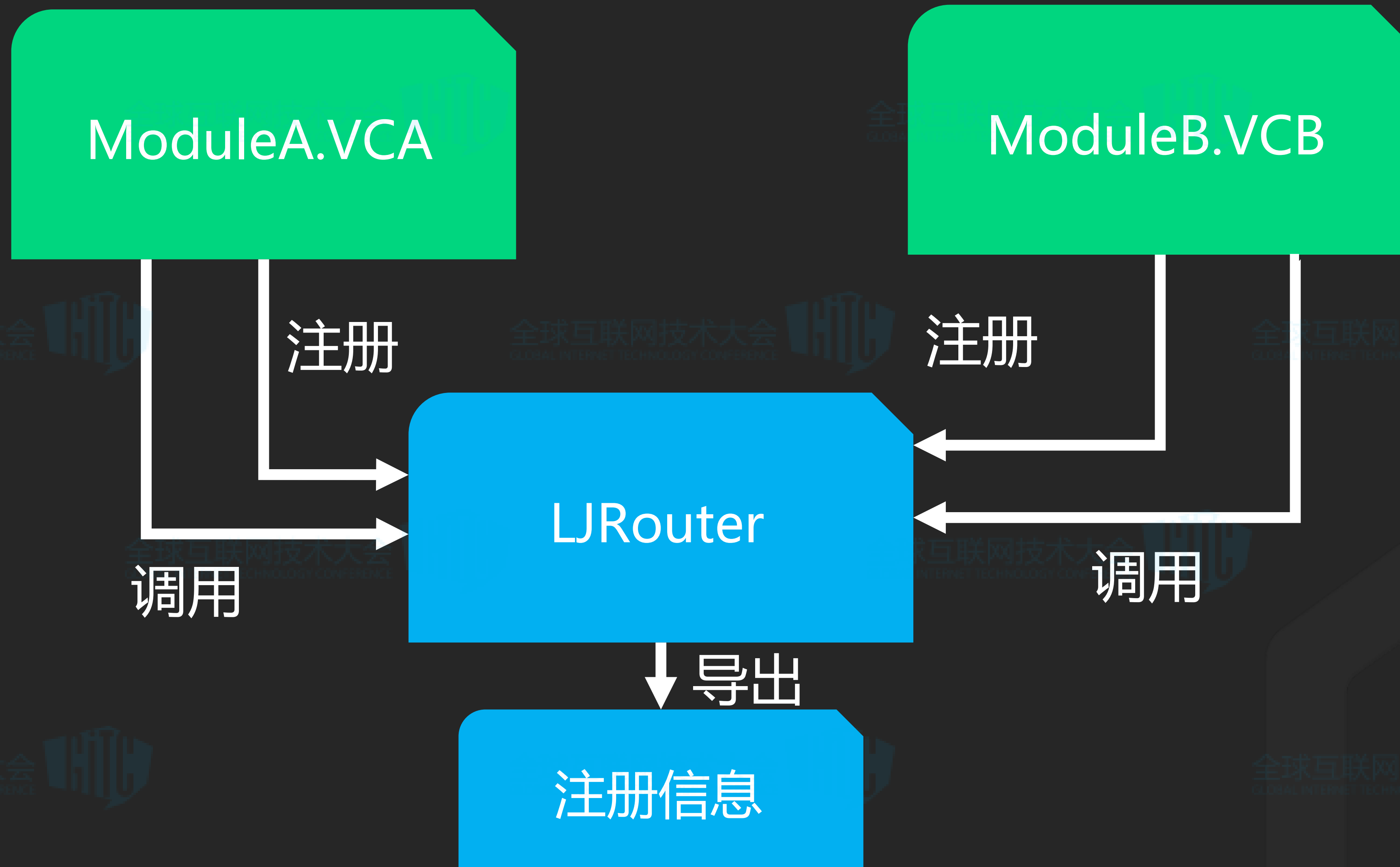
全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



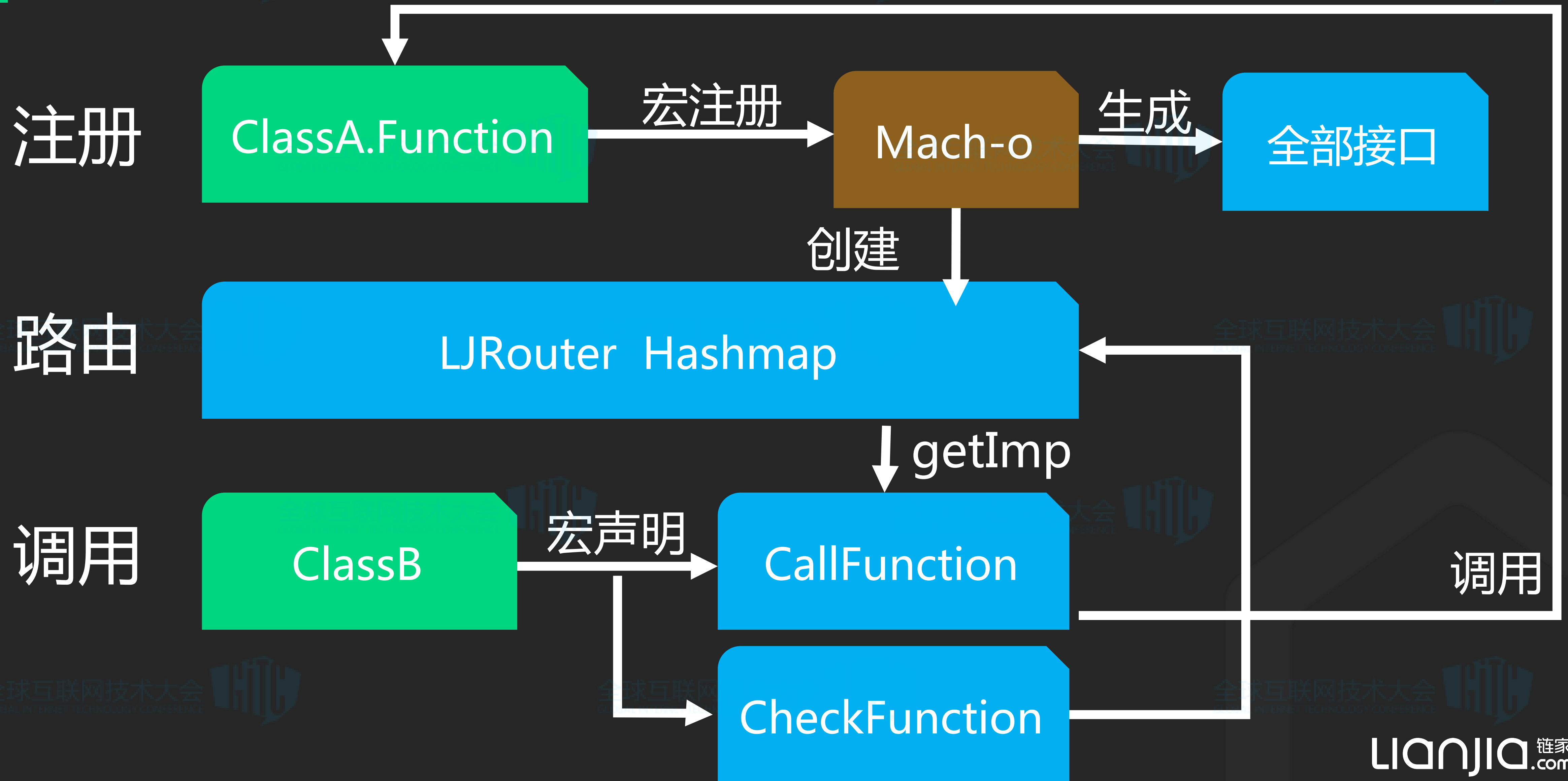
全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



4.LJRouter的实现-模块间调用



4.LJRouter的实现-模块间调用



4.LJRouter的实现-模块间调用

宏注册

1. 生成init函数
2. 生成+方法
3. 写入Mach-o

4.LJRouter的实现-模块间调用

```
+ (void) LJRouterHometitlexregInfo
{
    __attribute__((used)) static
    struct LJRouterRegister LJRouterHometitlexRegisterStruct
    __attribute__((used, section ("__DATA,__LJRouter"))) =
    {
        // 注册信息
    };
}

+ (instancetype) get_Home_controller_with_titlex : (NSString*) titlex
{
    return [[self alloc] initWith_Home_titlex : titlex];
}

- (instancetype) initWith_Home_titlex : (NSString*) titlex
{
    self = [super init];
    if (self)
    {
        self.title = titlex;
    }
    return self;
}
```

4.LJRouter的实现-模块间调用

路由初始化

1. 路由加载时读取Mach-o
2. 读取校验函数并执行

4.LJRouter的实现-模块间调用

```
// 注册所有的vc
```

```
uint8_t *data = getsectiondata(header, "__DATA", "__LJRouter",&size);
```

```
// 检查所有的函数类型
```

```
uint8_t *data = getsectiondata(header, "__DATA", "__LJRouterCheck",&size);
```

4.LJRouter的实现-模块间调用

宏使用页面/action

1. 调用真实函数
2. 生成校验函数

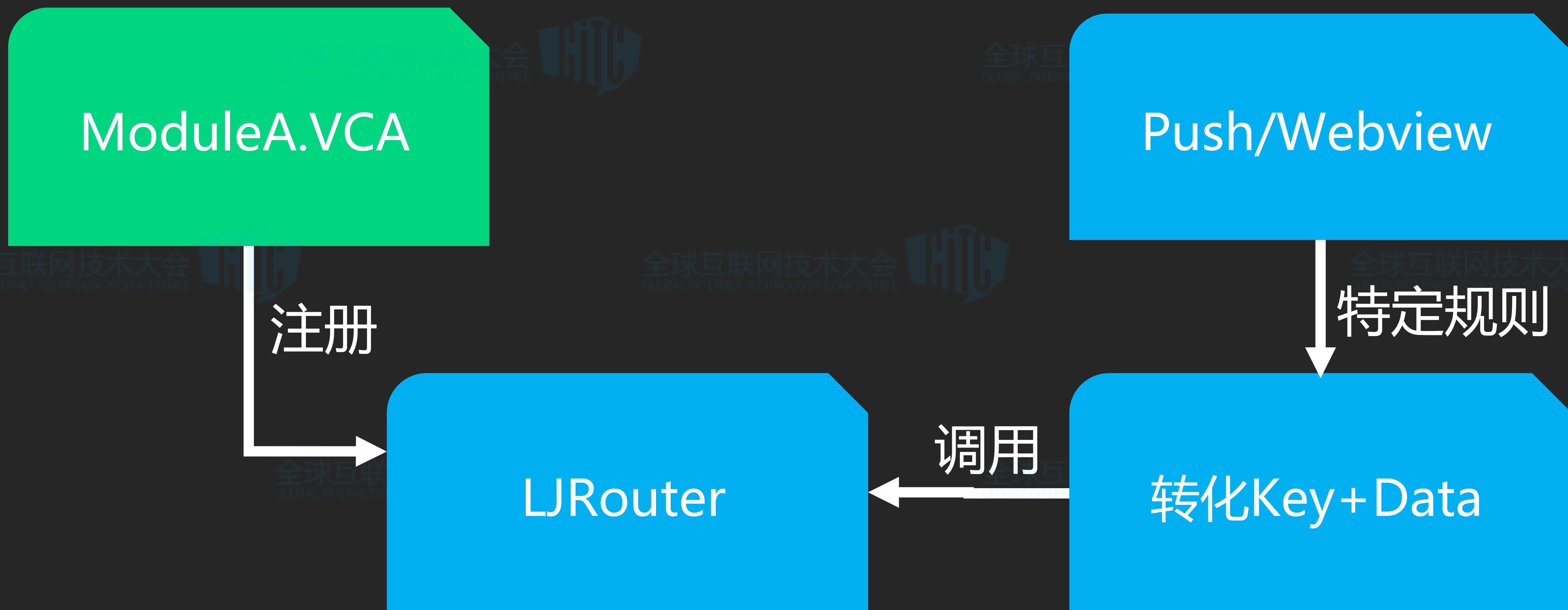
4.LJRouter的实现-模块间调用

```
// debug
static UIViewController* get_Home_controller_with_titlex ( typeof( NSString*) titlex)
{
    static void(^CheckFunctionName)() = ^(){
        // check
    };
    static void* CheckFunctionNameAddr __attribute__((used, section ("__DATA,__LJRouterCheck")))
        = &CheckFunctionName;

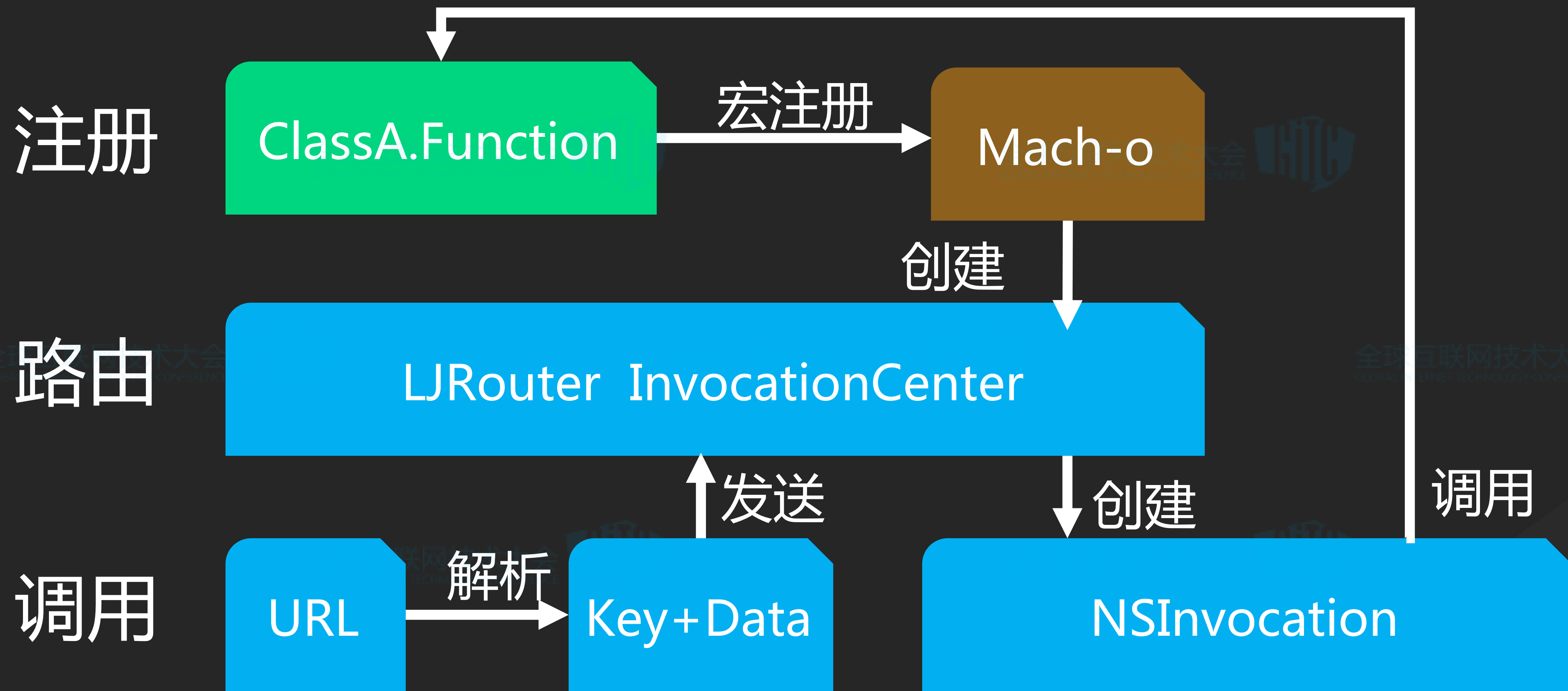
    // 调用真实imp
    Class theClass = [[LJRouter sharedInstance] classForKey:@"Home"];
    NSString *selName = @"get_Home_controller" "_with_" "titlex" ":";
    SEL sel = NSSelectorFromString(selName);
    UIViewController*(^imp)(id, SEL , typeof( titlex ))
        = (void*)class_getMethodImplementation(object_getClass(theClass), sel);
    return imp(theClass, sel , titlex);
}

// release
extern UIViewController* get_Home_controller_with_titlex ( typeof( NSString*) titlex);
```

4.LJRouter的实现-外部调用



4.LJRouter的实现-外部调用



4.LJRouter的实现-外部调用

外部调用

1.URL通用规则转化Key+Data

2.URL+js 转化Key+Data

3.其他格式转化Key+Data

4.LJRouter的实现-外部调用

InvocationCenter

1. 检查必要参数
2. 转化参数类型
3. 组装NSInvocation



5. 优势与问题



5.优势与问题

- 易于上手
- 自由传参
- 模块间有效性校验
- 尽可能细的粒度下保证模块感
- 支持 h5 / push 等外部调用以及回调
- 维护成本低
- 易于生成文档与文档自动校验
- 编译期注册 延迟加载

5.优势与问题

- Swift有限支持
- 框架级别未包含restful api解决方案.
- 编写风格稍有变化

Question & Answering



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



 Thanks!

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE

