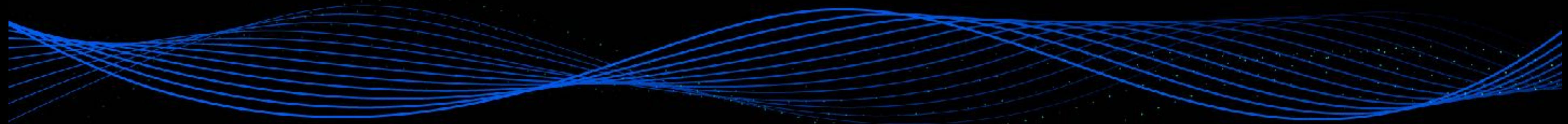




Swift在Airbnb的应用实践

陈坤 / Airbnb iOS高级工程师



Airbnb为什么使用Swift

新语言(2014发布), 安全编程模型为出发点, 包含很多现代编程特性

苹果大力支持和推广

和苹果建立良好的关系

准备重写UI

可以直接调用Objective-C的代码

时间轴



2015年以前
纯Objective-C

时间轴

2015年底
开始使用Swift构建新UI

2015年以前
纯Objective-C



时间轴

2015年底
开始使用Swift构建新UI

2015年以前
纯Objective-C

2017年1月
Swift 2->3

时间轴

2015年底
开始使用Swift构建新UI

2017年5月
支持Swift的BUCK构建系统

2015年以前
纯Objective-C

2017年1月
Swift 2->3

时间轴

2015年底
开始使用Swift构建新UI

2017年5月
支持Swift的BUCK构建系统

2015年以前
纯Objective-C


2017年1月
Swift 2->3

现在
少量Infra代码Objective-C


Swift的现状

3 Years, Version 4

Open Sourced, ~4,0000 Stars

 Watch ▾

2,445

 Unstar

38,695

 Fork

5,786

5	5		Python
6	7	▲	Visual Basic .NET
7	6	▼	PHP
8	8		JavaScript
9	11	▲	Delphi/Object Pascal
10	14	▲▲	Swift
11	9	▼	Perl
12	10	▼	Ruby
13	12	▼	Assembly language
14	16	▲	R
15	13	▼	Visual Basic
16	15	▼	Objective-C
17	48	▲▲	Go
18	18		MATLAB
19	19		PL/SQL
20	26	▲	Scratch

增长迅速

Alamofire

RxSwift

Kingsfisher

SwiftyJSON

ObjectMapper

PromiseKit

...

使用Swift的公司

Airbnb

Lyft

Uber

Linkedin

Coursera

IBM

Vimeo

Swift的主要特性以及在Airbnb的应用

安全

Optional Variable

Objective-C

```
59 @interface AIRUser : AIRModel
60
61 @property (nonatomic, copy, readonly) NSString *userId;
62 @property (nonatomic, copy, readonly) NSString *firstName;
63 @property (nonatomic, copy, readonly) NSString *lastName;
```

Optional Variable

Swift

```
public final class AIRUser {  
    let userId: String  
    let firstName: String  
    let lastName: String?
```

T 和T?是完全不同的两个类型


```
guard let lastName = user.lastName else { return user.firstName }  
  
// lastName's type is 'String' instead of 'String?' here  
return "\(user.firstName) \(lastName)"
```

Wrap和Unwrap

```
if let lastName = user.lastName {  
    // lastName's type is 'String' instead of 'String?' here  
    return "\(user.firstName) \(lastName)"  
} else {  
    // Can't access lastName here  
    return user.firstName  
}
```

let VS var

Immutable is better than mutable

```
let color: UIColor
if colorType == "red" {
    color = .red
}
// Compile error
```

```
let color: UIColor
if colorType == "red" {
    color = .red
} else {
    color = .blue
}
```

高级枚举类型

```
private enum DeepLinkState {  
    case notStarted  
    case started(instrumentId: String)  
    case queryingForCompletion(instrumentId: String, timer: Timer, startedTime: Date)  
    case queryingFailed(instrumentId: String)  
}
```

Protocol Oriented Programming

范型

中国联通 上午1:37

推荐 房源 体验 攻略

热门体验

¥ 434
Contemporary ar...
★★★★★ 33条评价

¥ 426 Cruise old wineries on a vint... &
★★★★★ 114条评价

好友看过

探索 心愿单 旅程 收件箱 个人资料

中国联通 上午1:37

推荐 房源 体验 攻略

房源

查看全部 >

¥ 477 奥斯图尼的整套房子
★★★★★ 82条评价

¥ 2,551 亚特兰房子
★★★★★ 231条评价

旅游目的地精选

LOS ANGELES

FIRENZE

LONDON

探索 心愿单 旅程 收件箱 个人资料

中国联通 上午1:37

推荐 房源 体验 攻略

旅游目的地精选

LOS ANGELES

洛杉矶

FIRENZE

佛罗伦萨

LONDON

伦敦

艺术与设计体验

查看全部 >

MARCEL
Bohemia CUBANA

探索 心愿单 旅程 收件箱 个人资料

```
public final class CarouselComponent<T>: UIView,  
StandardComponent,
```



where

```
T: UIView,  
T: StandardComponent,
```



```
{
```

CarouselComponent<HomeCardComponent>

CarouselComponent<ExperienceCardComponent>

CarouselComponent<GuidebookComponent>

其它特性

Memory Management

Less code

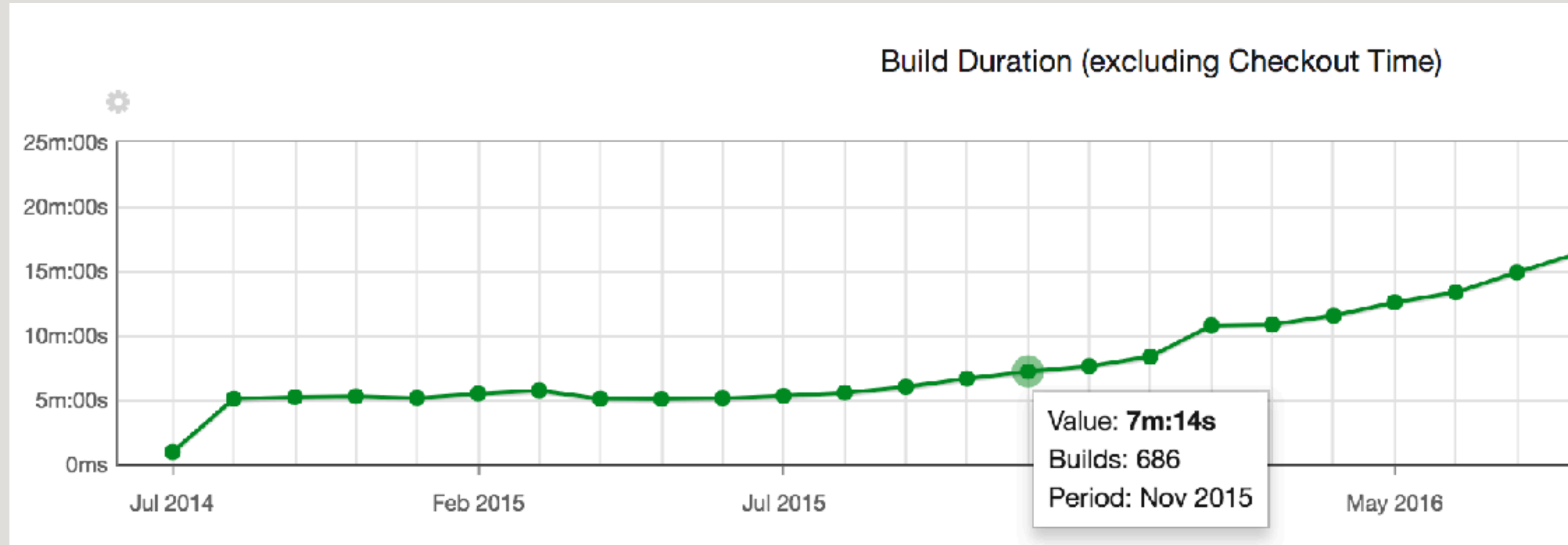
Faster

Tuples

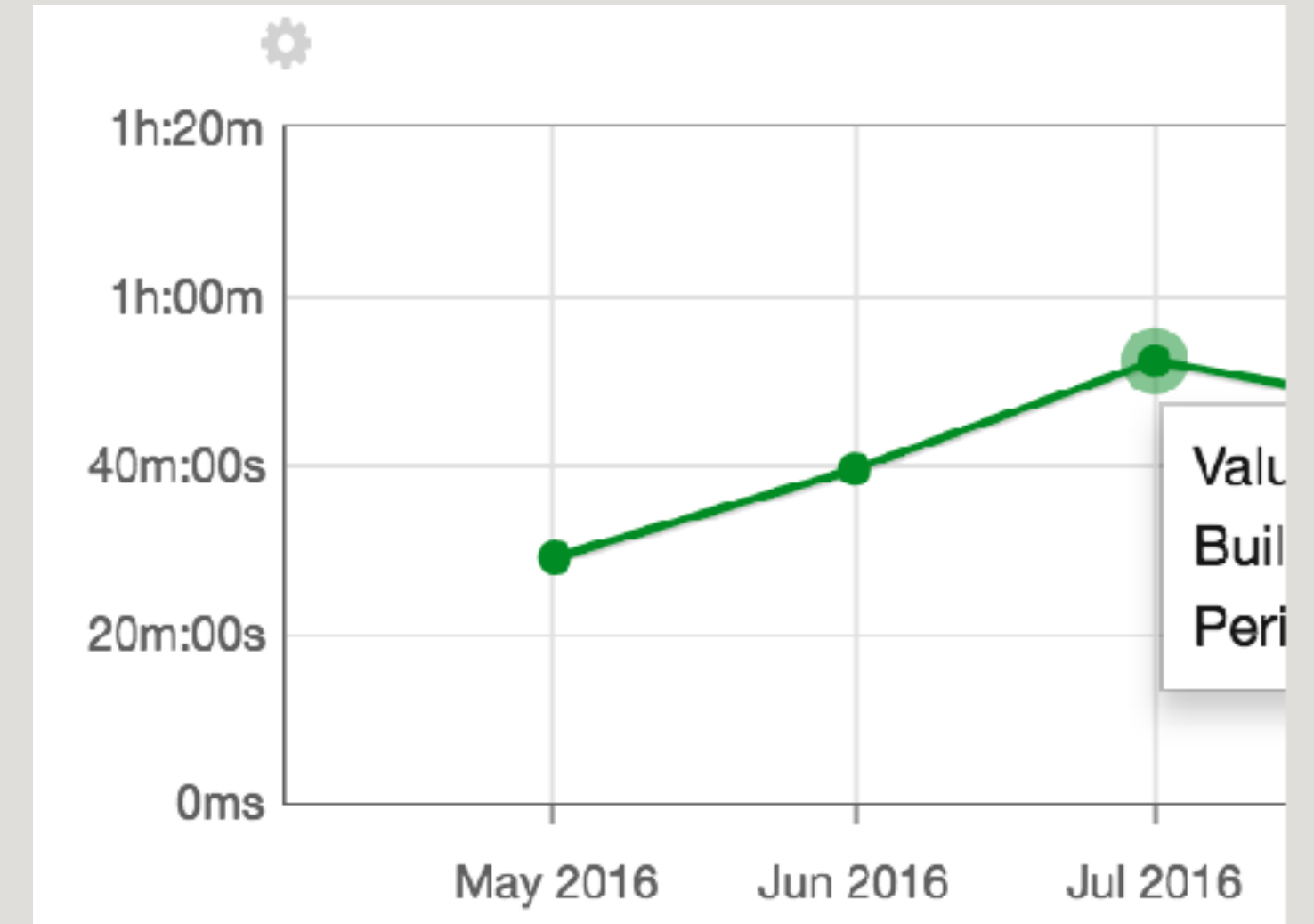
Access Control

Airbnb遇到的问题以及解决方案

编译时间



Debug Build

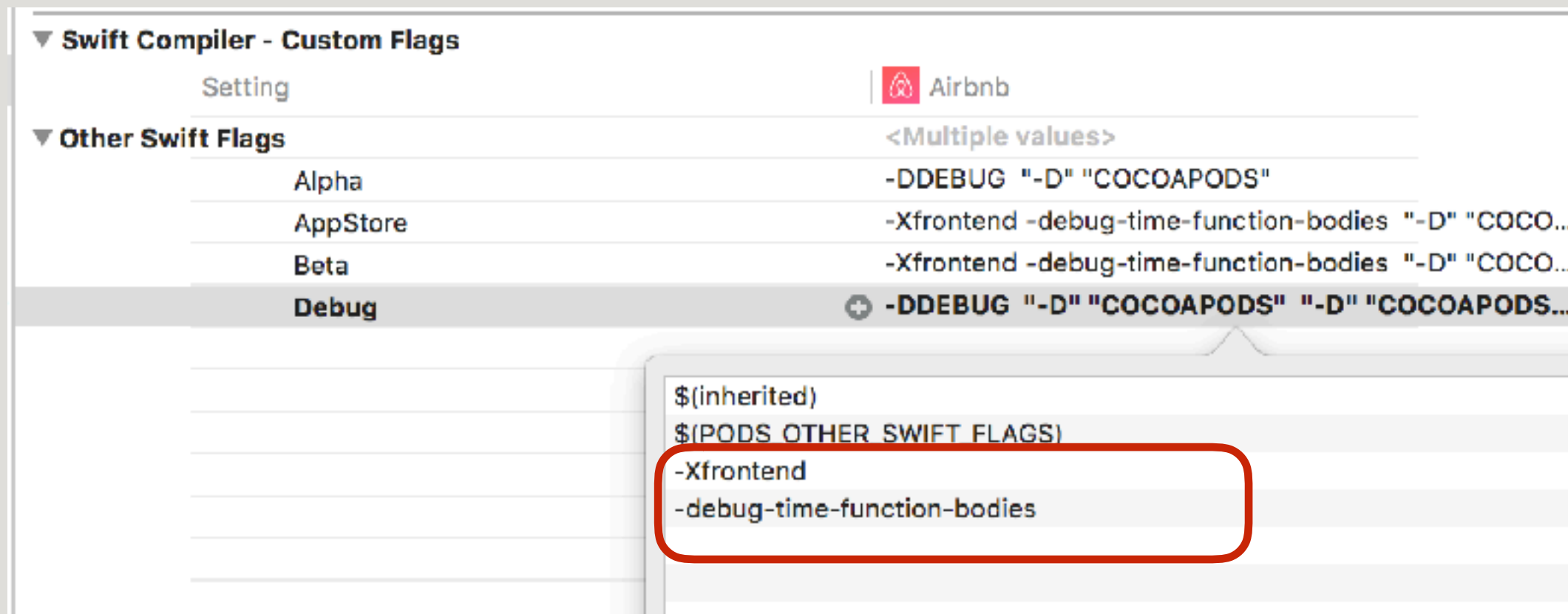


Release Build

类型推导以及类型检查

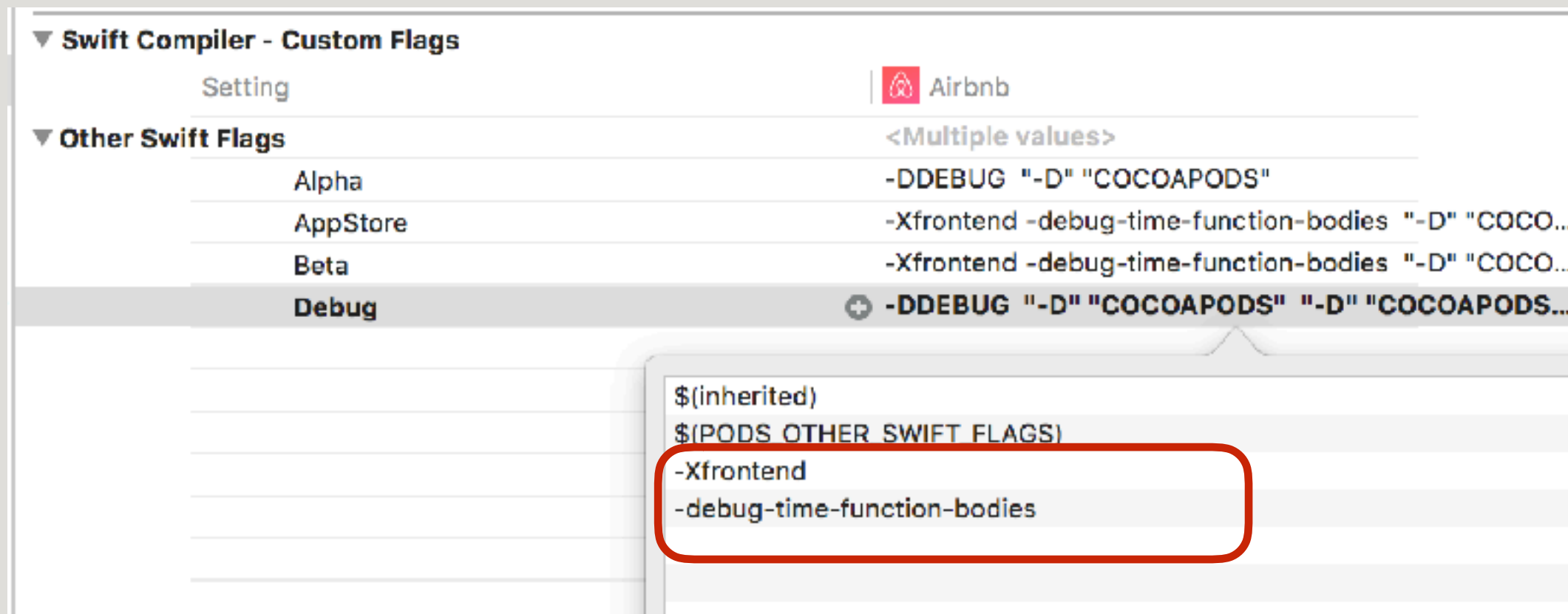
调试编译时间

OTHER_SWIFT_FLAGS: `-Xfrontend -debug-time-function-bodies`



调试编译时间

OTHER_SWIFT_FLAGS: -Xfrontend -debug-time-function-bodies



The screenshot shows the 'Build Time Analyzer' tool interface. The status is 'Completed'. A search bar is present. The main table displays the following data:

Cumulative time	Location	Occurrences	Function
2755.7ms	MemoryTrackingManager.swift:228:20	1	get {}
752.7ms	HomeSearchEventTracking.swift:292:23	1	private static func sendMParticleP2ImpressionEvent(searchMe...
383.8ms	ExploreSearchFiltersHorizontalCompon...	1	public func ==(lhs: ExploreSearchFiltersHorizontalStyle, rhs: Ex...
304.0ms	InlineRowRangeSliderComponent.swift:...	1	func 🔥 a18y 🔥 decrementMaxValue()
241.7ms	ChinaHomeCardComponent.swift:145:13	1	public func ==(lhs: ChinaHomeCardContent, rhs: ChinaHomeC...
200.1ms	ExploreSearchBarComponent.swift:369...	1	func ==(lhs: ExploreSearchBarStyle, rhs: ExploreSearchBarStyl...
193.0ms	QuickPayContent.swift:123:13	1	public func ==(lhs: QuickPayDynamicContent, rhs: QuickPayDy...
191.9ms	AutocompleteOptionsComponent.swift:...	1	public func ==(lhs: AutoCompleteOptionsContent, rhs: AutoCo...
177.7ms	AvailabilityButtonComponent.swift:346:...	1	public func ==(lhs: AvailabilityButtonStyle, rhs: AvailabilityButt...

Tool: BuildTimeAnalyzer-for-Xcode

调试编译时间

```
226 }  
227  
228 private lazy var slides: [TravelStoriesSlide.V1.TravelStoriesSlide] = {  
229     return self.memory.slides?.enumerated().map { index, slide in  
230         ...  
231     }  
232 }
```

Before

```
226 }  
227  
228 private lazy var slides: [TravelStoriesSlide.V1.TravelStoriesSlide] = {  
229     () -> [TravelStoriesSlide.V1.TravelStoriesSlide] in  
230     return self.memory.slides?.enumerated().map { index, slide in  
231         ...  
232     }  
233 }
```

After

调试编译时间

```
226 }
227
228 private lazy var slides: [TravelStoriesSlide.V1.TravelStoriesSlide] = {
229     return self.memory.slides?.enumerated().map { index, slide in
230         ...
231     }
232 }
```

Before

```
226 }
227
228 private lazy var slides: [TravelStoriesSlide.V1.TravelStoriesSlide] = {
229     () -> [TravelStoriesSlide.V1.TravelStoriesSlide] in
230     return self.memory.slides?.enumerated().map { index, slide in
231         ...
232     }
233 }
```

After

Build Time Analyzer

Status: Completed

Cumulative time	Location	Occurrences	Function
1731.8ms	MemoryTrackingManager.swift:228:20	1	get {}
752.7ms	HomeSearchEventTracking.swift:292:23	1	private static func

其它例子

```
let myCompany = [  
  "employees": [  
    "employee 1": ["attribute": "value"],  
    "employee 2": ["attribute": "value"],  
    "employee 3": ["attribute": "value"],  
    "employee 4": ["attribute": "value"],  
    "employee 5": ["attribute": "value"],  
    "employee 6": ["attribute": "value"],  
    "employee 7": ["attribute": "value"],  
    "employee 8": ["attribute": "value"],  
    "employee 9": ["attribute": "value"],  
    "employee 10": ["attribute": "value"],  
    "employee 11": ["attribute": "value"],  
    "employee 12": ["attribute": "value"],  
    "employee 13": ["attribute": "value"],  
    "employee 14": ["attribute": "value"],  
    "employee 15": ["attribute": "value"],  
    "employee 16": ["attribute": "value"],  
    "employee 17": ["attribute": "value"],  
    "employee 18": ["attribute": "value"],  
    "employee 19": ["attribute": "value"],  
    "employee 20": ["attribute": "value"],  
  ]  
]
```

12小时

```
let x = { String("\($0)" + "") +  
  String("\($0)" + "") }(0)
```

15秒

解决方案

1. Use CI system to monitor the build time.
2. Educate engineers to provide type information as much as possible.
3. Use Lint to prevent engineers from common issues.
4. Offline type inference (Use SourceKit).

其它技巧

Split Libraries

Airbnb目前有70个左右的Libraries

BUCK编译系统

大量缓存编译中间结果，CI编译时间大幅降低

不需要手动管理.xcodeproj文件

高级的依赖管理

开源

Uber和Airbnb都已使用

SWIFT_WHOLE_MODULE_OPTIMIZATION

▼ Optimization Level	<Multiple values> ⇅
Alpha	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
AppStore	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Beta	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Debug	✓ None [-Onone] - \${SWIFT_OPTIMIZATION_LEVEL_\${CONFIGURATION}} Fast, Single-File Optimization [-O]
	Fast, Whole Module Optimization [-O -whole-module-optimization]
▼ User-Defined	

SWIFT_WHOLE_MODULE_OPTIMIZATION

Optimization Level	<Multiple values> ⇅
Alpha	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
AppStore	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Beta	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Debug	✓ Non-Optimized [-O -whole-module-optimization] - \${SWIFT_OPTIMIZATION_LEVEL_\${CONFIGURATION}} Fast, Single-File Optimization [-O]
User-Defined	Fast, Whole Module Optimization [-O -whole-module-optimization]

SWIFT_WHOLE_MODULE_OPTIMIZATION

Optimization Level	<Multiple values> ⇅
Alpha	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
AppStore	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Beta	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
Debug	Fast, Whole Module Optimization [-O -whole-module-optimization] - \$
	✓ Non-Optimized [-O -whole-module-optimization] - \${SWIFT_OPTIMIZATION_LEVEL_\${CONFIGURATION}}
	Fast, Single-File Optimization [-O]
	Fast, Whole Module Optimization [-O -whole-module-optimization]

User-Defined	
Setting	Airbnb
▶ SWIFT_OPTIMIZATION_LEVEL_Alpha	-Owholemodule
SWIFT_OPTIMIZATION_LEVEL_AppStore	-Owholemodule
SWIFT_OPTIMIZATION_LEVEL_Beta	-Owholemodule
▼ SWIFT_WHOLE_MODULE_OPTIMIZATION	<Multiple values>
Alpha	
AppStore	
Beta	
Debug	YES

SWIFT_WHOLE_MODULE_OPTIMIZATION

Cumulative time	Location	Occurrences
7575.1ms	VerificationIdentityCoordinator.swift:86...	1
7550.1ms	ThemeStrings.swift:66:26	461
3983.1ms	ThemeStrings.swift:240:27	461
3473.4ms	ThemeStrings.swift:132:86	461
3339.1ms	ThemeStrings.swift:346:30	461
3317.8ms	ThemeStrings.swift:519:30	461
3280.3ms	EditListingCheckinViewController.swift:...	461
3179.5ms	EditListingCheckinViewController.swift:...	461
3055.2ms	ThemeStrings.swift:458:29	461
2790.1ms	ThemeStrings.swift:205:25	461
2613.3ms	AppContainmentViewController.swift:2...	461
2506.6ms	AppContainmentViewController.swift:2...	461

Before

Cumulative time	Location	Occurrences
6214.6ms	VerificationIdentityCoordinator.swift:86...	1
4599.1ms	PaymentOptionsDataSource.swift:113:16	1
282.9ms	SearchResultsContentPresenter.swift:4...	1
183.7ms	SessionManager.swift:77:8	1
128.5ms	PerformanceInsightsViewController.swi...	1
120.9ms	LegacyPerformanceInsightsViewContro...	1
112.5ms	SwiftJSON.swift:447:9	1
95.1ms	PerformanceInsightsViewController.swi...	1
65.4ms	LegacyPerformanceInsightsViewContro...	1

After

SWIFT_WHOLE_MODULE_OPTIMIZATION

13m9.414s

Before

4m24.634s

After

启动性能

测量pre-main时间

```
54 static NSInteger GetTimeSinceProcessStarted() {  
55     int mib[] = {CTL_KERN, KERN_PROC, KERN_PROC_PID , getpid()};  
56     struct kinfo_proc processInfo;  
57     size_t size = sizeof(processInfo);  
58     if (sysctl(mib, sizeof(mib) / sizeof(mib[0]), &processInfo, &size, NULL, 0) != -1) {  
59         struct timeval processStartTime = processInfo.kp_proc.p_un.__p_starttime;
```

调试dyld性能

Airbnb > iPhone 5s

Build
32 targets

Run
Debug

Test
Debug

Profile
Debug

Analyze
Debug

Archive
AppStore

Info Arguments

Arguments Passed On Launch

No Arguments

Environment Variables

Name	Value
<input checked="" type="checkbox"/> DYLD_PRINT_STATISTICS	1
<input checked="" type="checkbox"/> DYLD_PRINT_LIBRARIES	1
<input checked="" type="checkbox"/> OS_ACTIVITY_MODE	disabled

动态库合并

Dynamic Library Merging

基本原理: Modify Main Target's linker flags

删除“-l” flags

增加“-filelist” flag

1. Consolidate Assets (Pre)

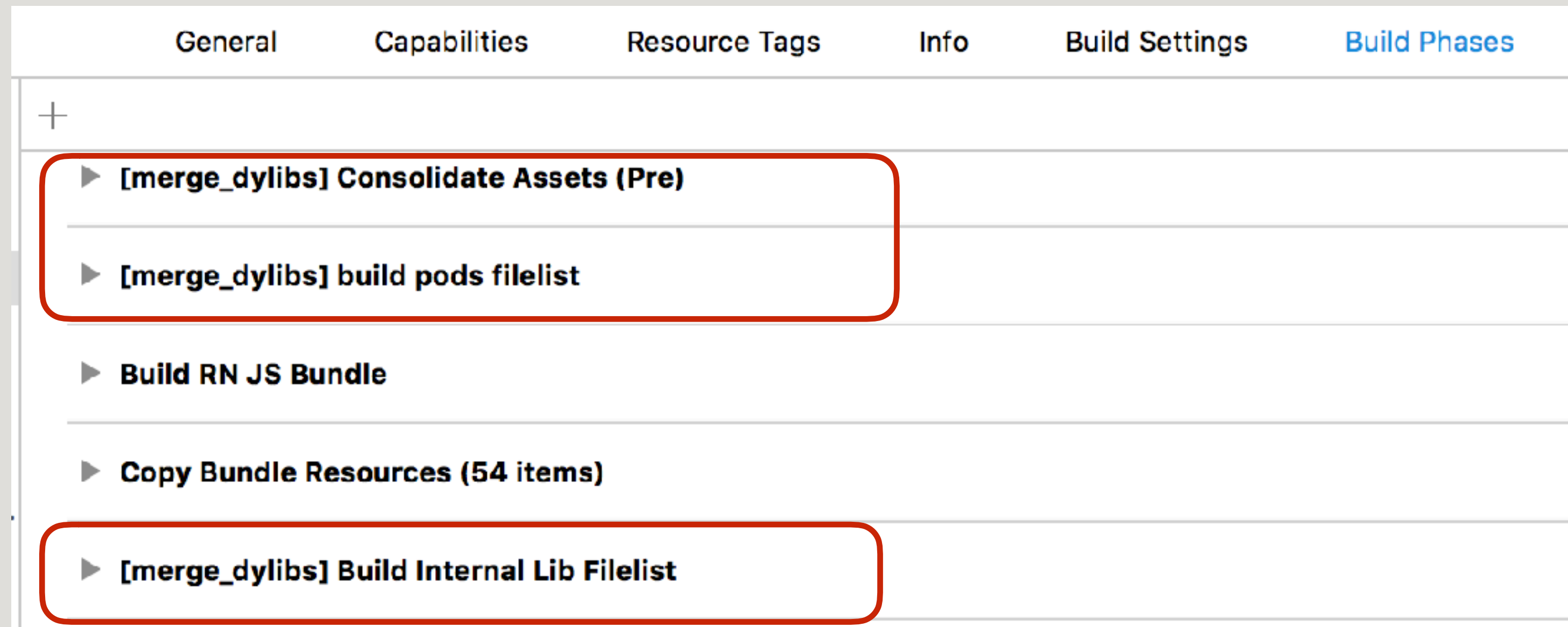
2. Build pods filelist

3. Build internal libraries filelist

4. Consolidate Assets (Post)

5. Remove unnecessary embembbed libraries

Xcode Build Phases例子



性能改善

-50%

Pre-main Time

-12MB

Install Size

Swift和Objective-C混用

Swift和Objective-C混用

1. 一定要在objective-c代码里面正确的标柱nullable!!!
2. 增量编译时间。

Thank you!

KUN.CHEN@AIRBNB.COM

