

蚂蚁财富Flutter多业务开发实践

楚斌

蚂蚁财富 - 平台与基础

自我介绍



楚斌，无线开发专家

蚂蚁财富-平台与基础 App端 iOS负责人

曾就职于网易有道，2017年加入蚂蚁金服，经历了蚂蚁财富发展的多个重大变革阶段，通过工具和工程方法推动了Flutter在财富BU内落地和跨BU的输出。

目录

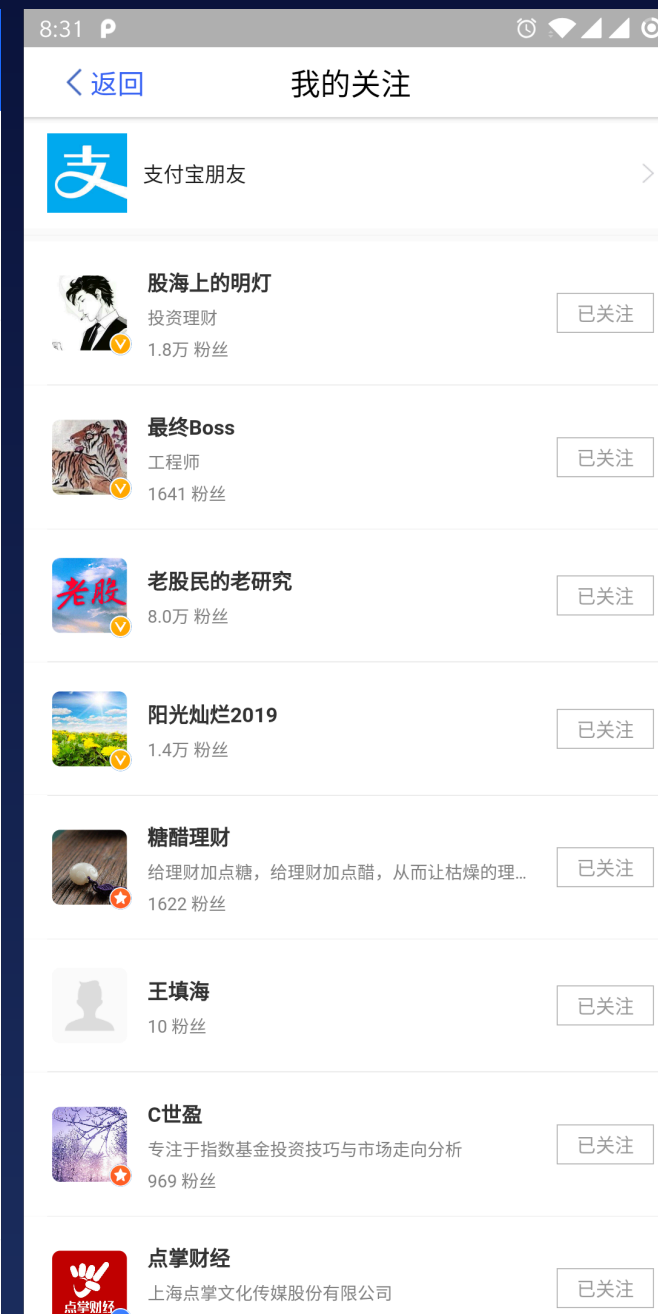
- 业务背景
- Flutter建设路线图
- 复杂生态融合
- 多业务技术体系
- 总结与展望

蚂蚁财富 - 国内领先的线上理财平台

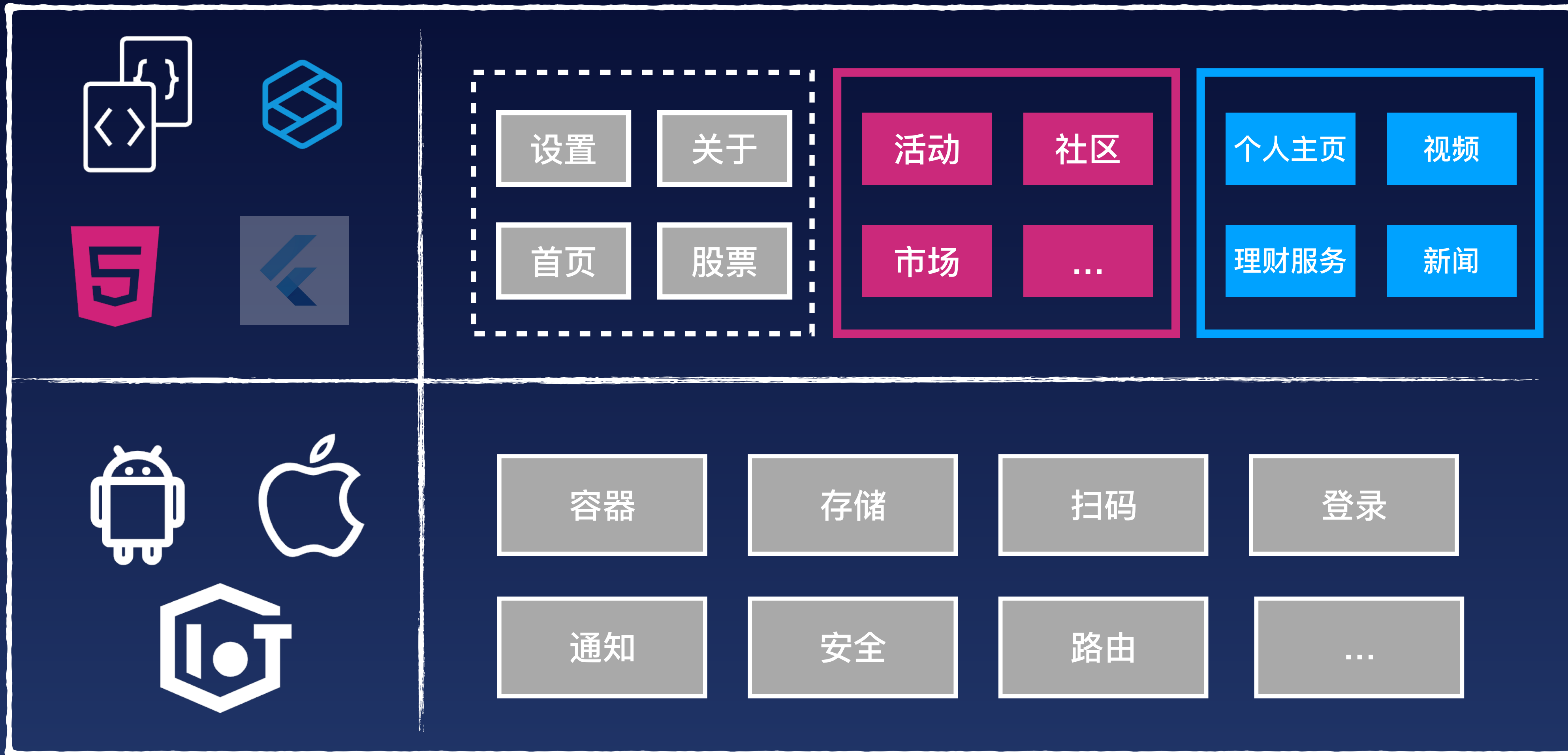
页面 + 区块 > 10

支撑业务 > 4

迭代版本 > 10



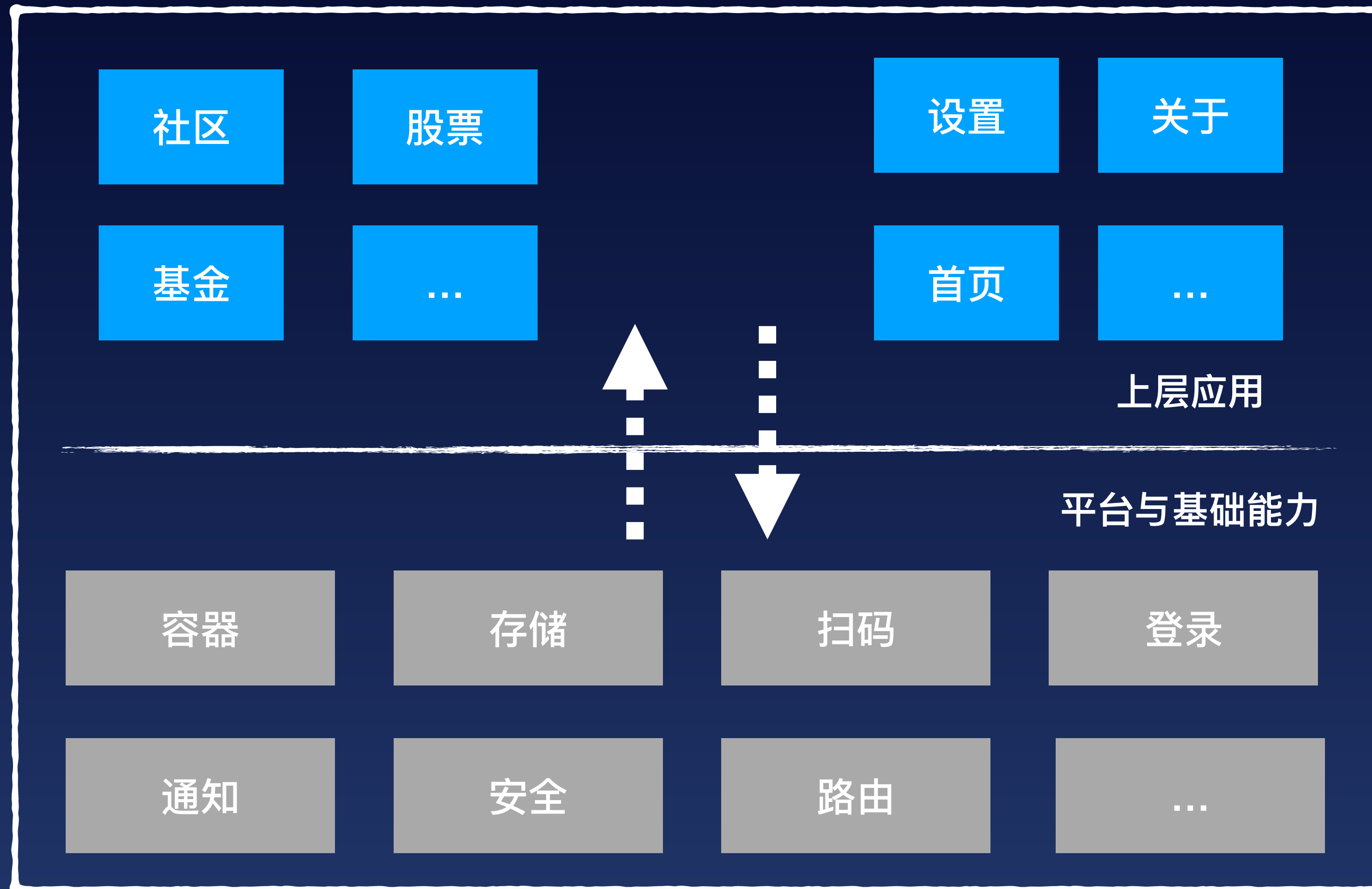
传统应用中Flutter的位置



???



Flutter落地面临的问题



- 如何开发?
- 开发体验? **别人怎么用?**
- 发布构建?

- Flutter如何接入?
- 平台能力? **自己怎么用?**
- 风控降级?
- 线上监控?

目录

- 业务背景
- Flutter建设路线图
- 复杂生态融合
- 多业务技术体系
- 总结与展望

建设路线图

一期 - 能力建设

- 容器能力对齐
- 混编脚本
- 混合栈
- 工程化
- 图片复用
- iOS内存
- 存储/通知
- 试点业务上线



二期 - 生态与运维

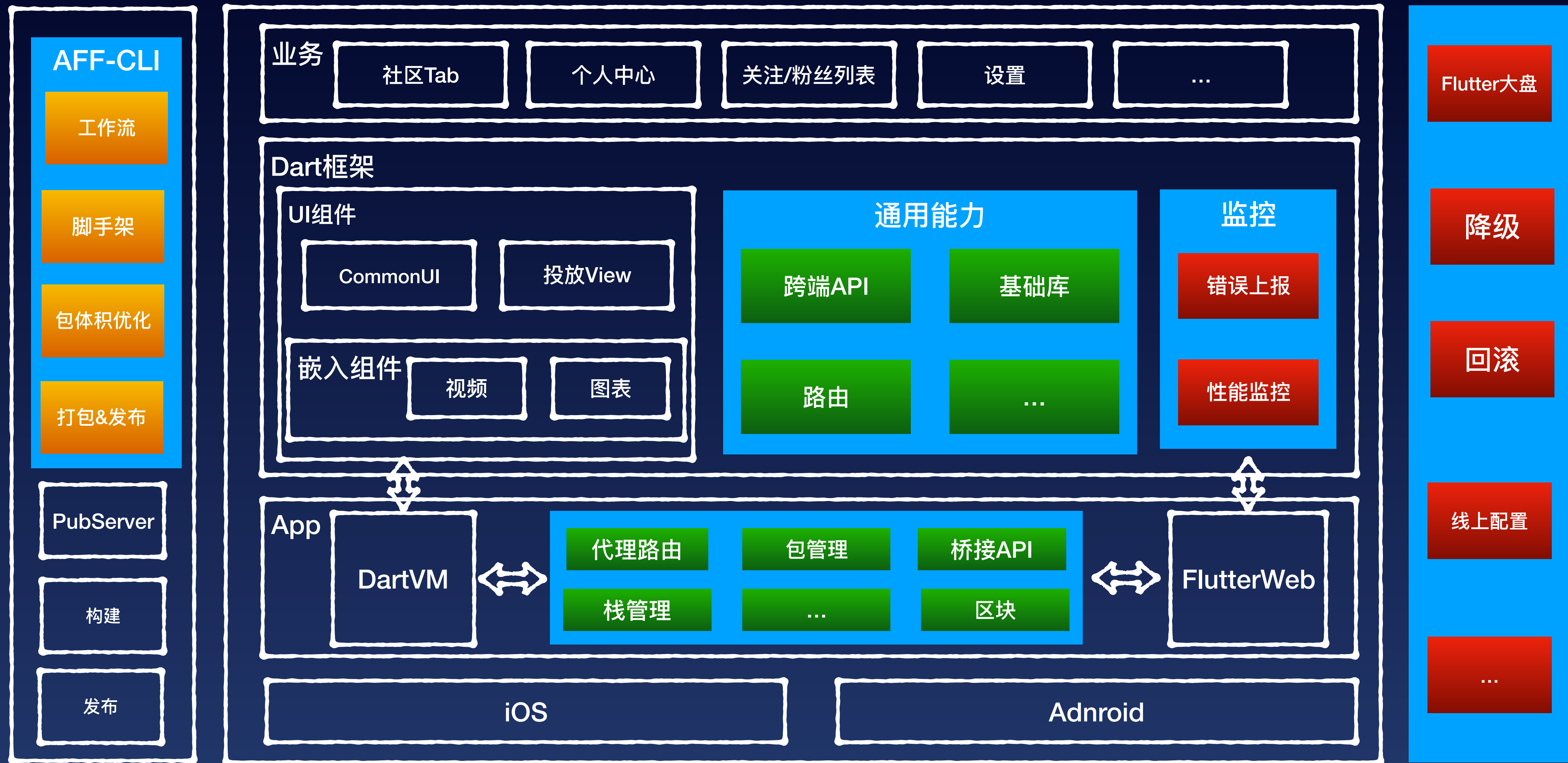
- UI组件
- 路由/降级
- 线上监控
- 私有pub
- workflow 沉淀
- 构建与发布
- 埋点&日志



三期 - 开放赋能

- 路由重构
- CI建设&优化
- CLI工具沉淀
- 业务组件沉淀
- 重点业务上线
- 技术输出
- 数据大盘建设

技术全景

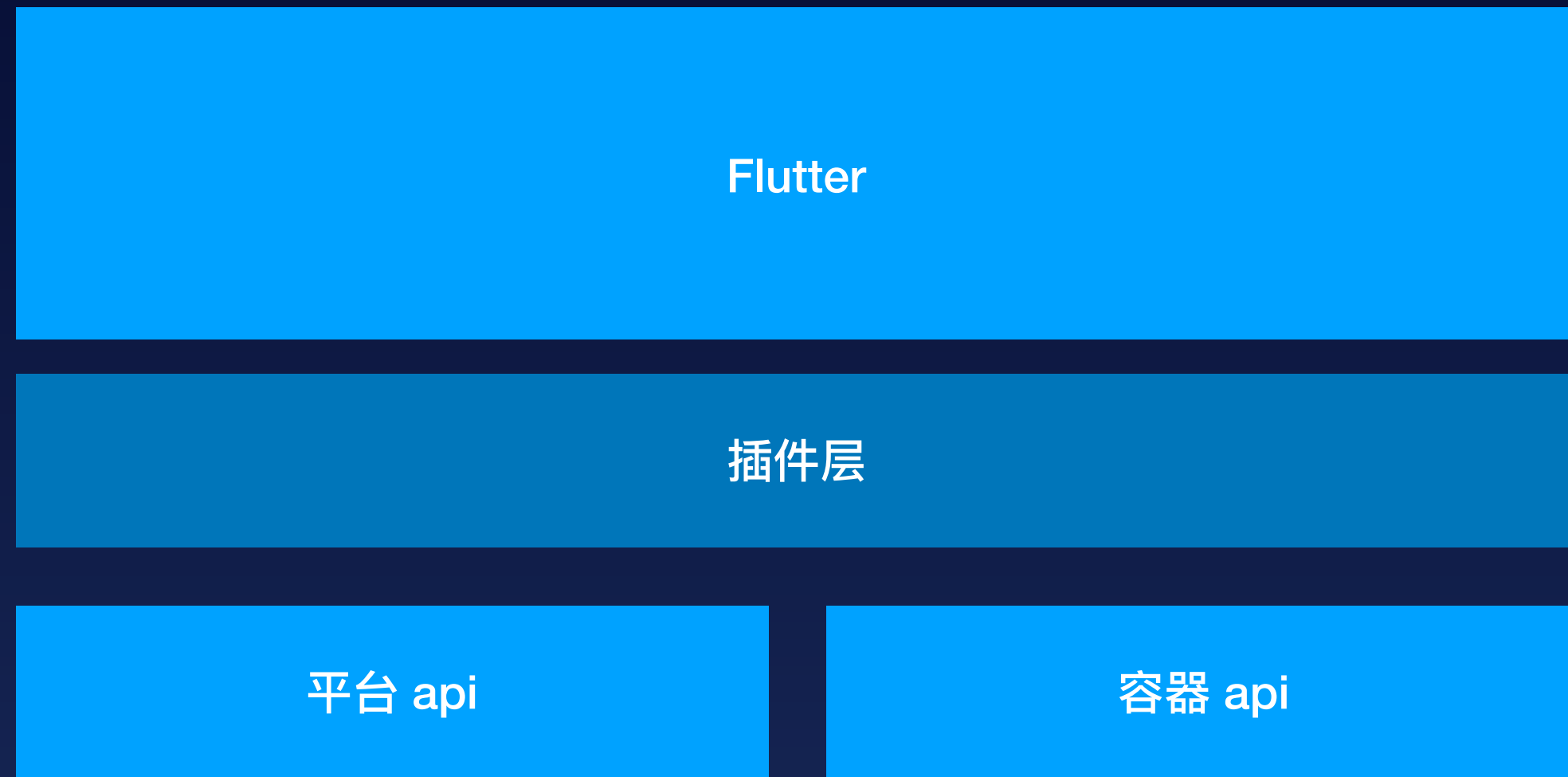


目录

- 业务背景
- 建设路线图
- 复杂生态融合
- 多业务技术体系
- 总结与展望

能力建设

容器能力对齐



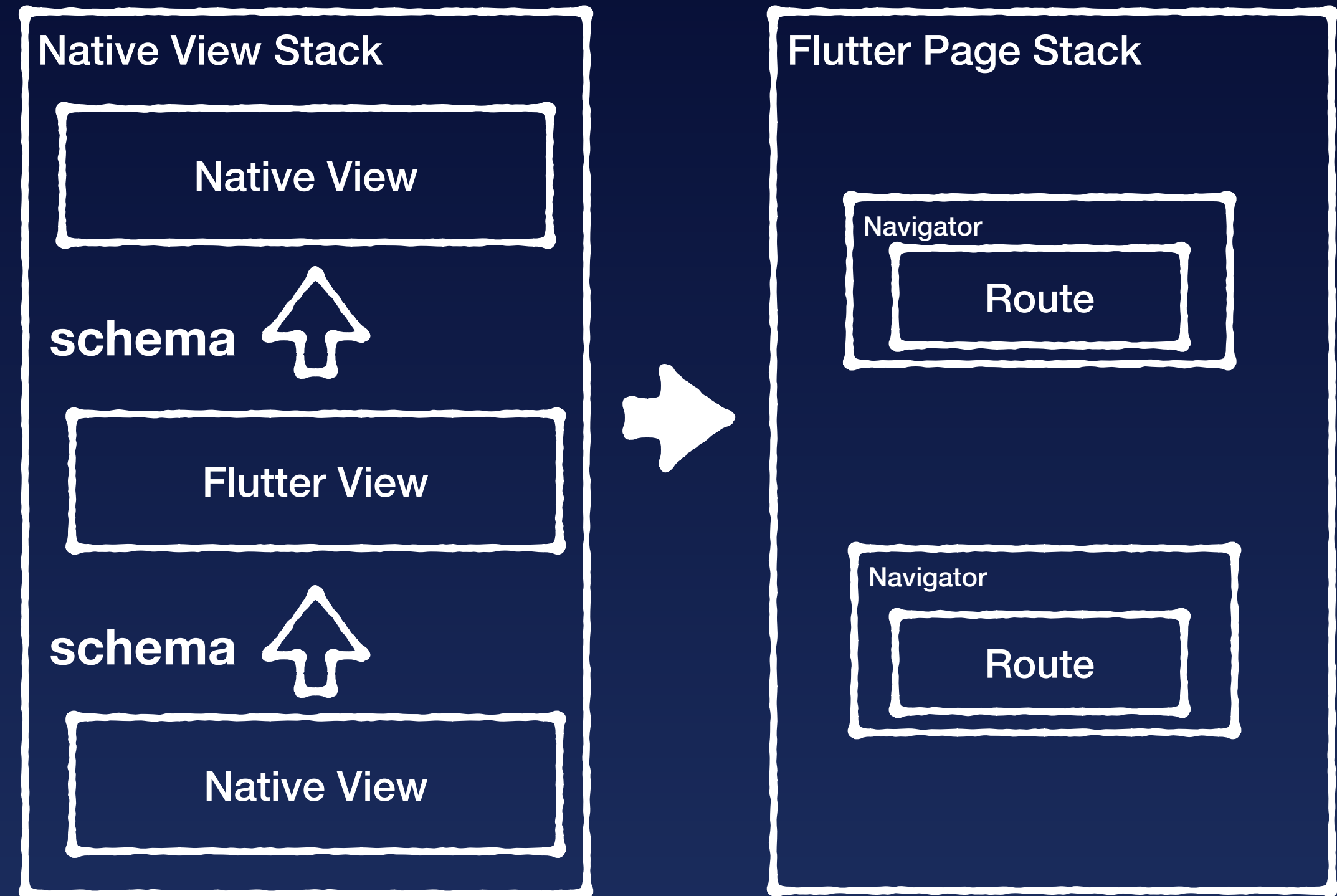
平台能力: MethodChannel

通知能力: EventSink

组件复用: PlatformView

```
flutterApi.call("rpc", {  
  operationType: "com.alipay. ....index.getHome"  
})
```

混合栈&路由



Flutter容器方案: FlutterBoost by 闲鱼

页面接入: scheme

视图接入: addSubview

能力建设 - Tip

通用方法名

Object.someMethod



```
flutterApi.getAppName();
```



noSuchMethod



```
@override  
dynamic noSuchMethod(Invocation invocation) { ...
```



NativeHandler(Symbol, Params)



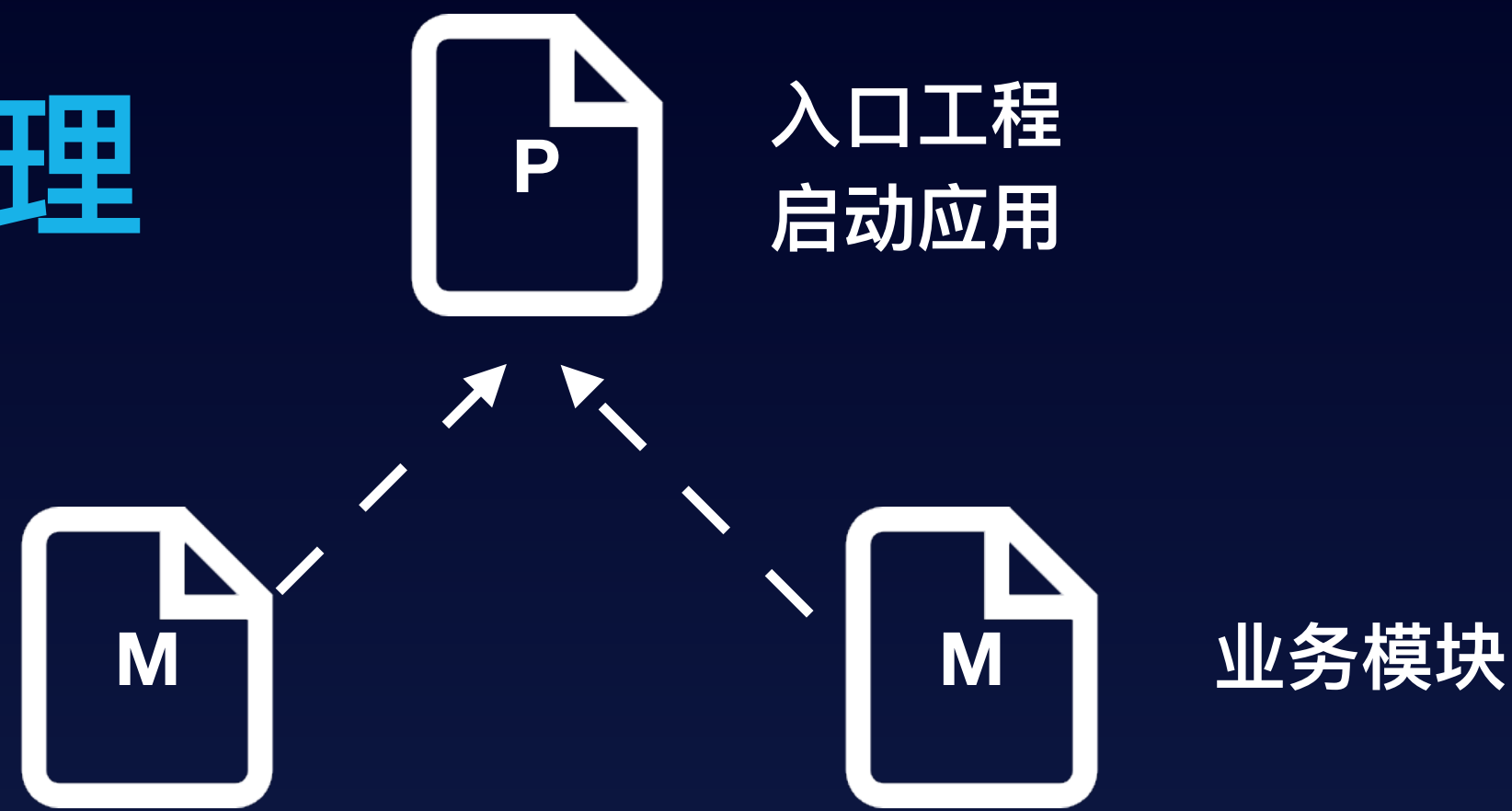
Callback

强校验入参

```
Future getSchemaUrl(@required String stageCode, String appId);  
// 获取展台url  
Future getStageSchemaUri(@required String stageCode, String appId);  
// 获取应用名称  
Future getAppName({String stageCode, String appId});
```

```
}  
await Future.wait([  
  flutterApi.getAppName({String stageCode, String appId}) → Future<dynamic>  
]);  
flutterApi.getAppName();
```

模块管理



普通依赖

```
dependencies:  
  af_flutter_setting:  
    git:  
      ref: 848135715009f5c...  
      url: git@gitlab.alipay-inc.com:flutter/af_flutter_setting.git  
  af_flutter_userinfo:  
    git:  
      ref: scroll_physics  
      url: git@gitlab.alipay-inc.com:flutter/af_flutter_userinfo.git  
  flutter:  
    sdk: flutter
```

依赖冲突

```
name: af_flutter_portal  
version: 0.0.1  
dependency_overrides:  
  af_flutter_api:  
    git:  
      ref: a5ae6751946c462b41281ec16c18440471d5b0e8  
      url: git@gitlab.alipay-inc.com:flutter/af_flutter_api.git
```



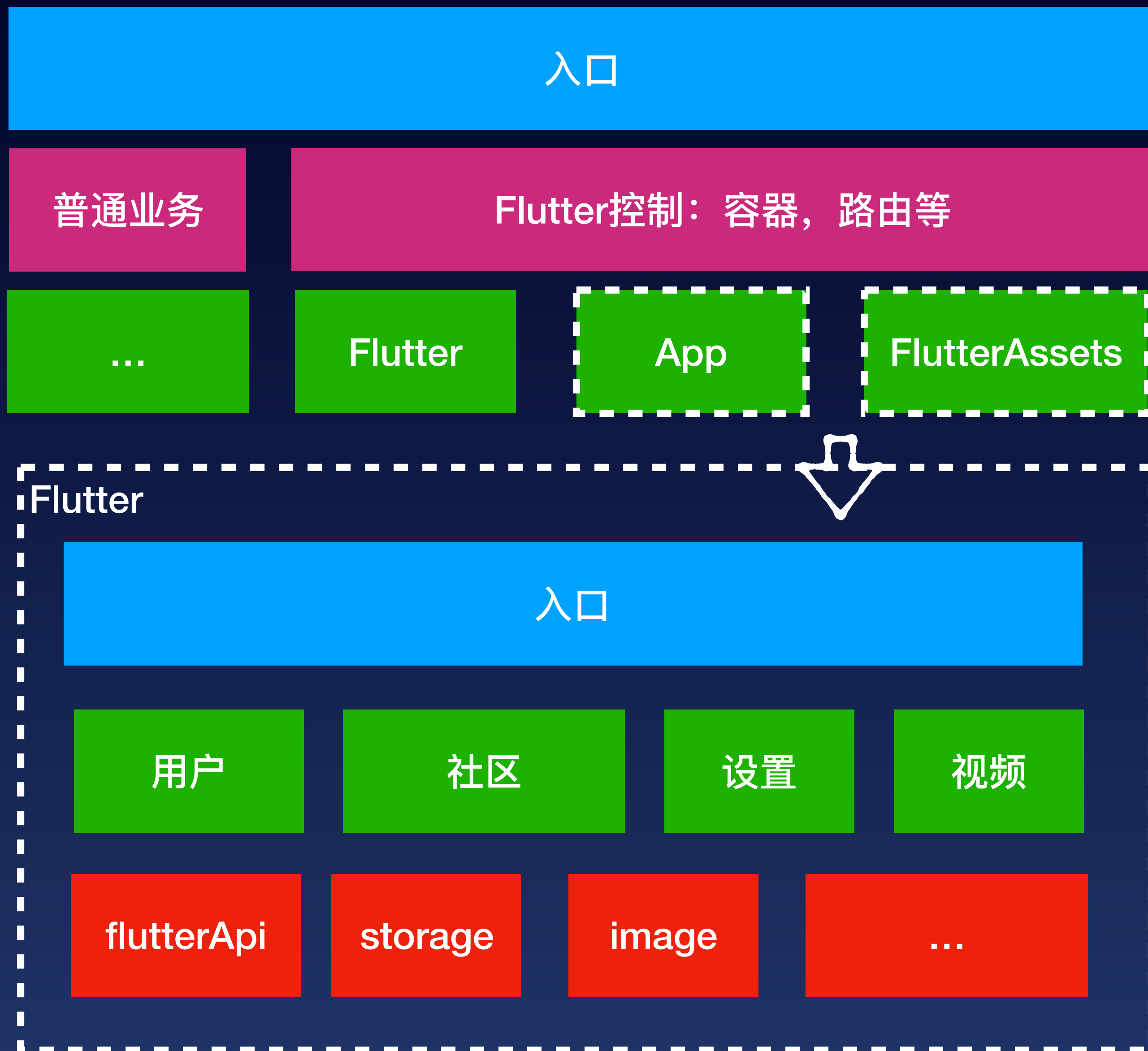
Flutter应用的一个版本



工程化

Native模块

Flutter模块

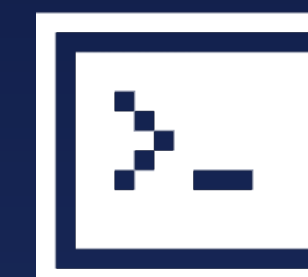
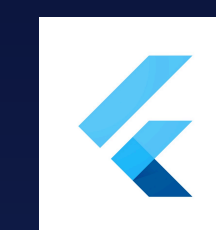


构建过程



Gradle Task / build phase

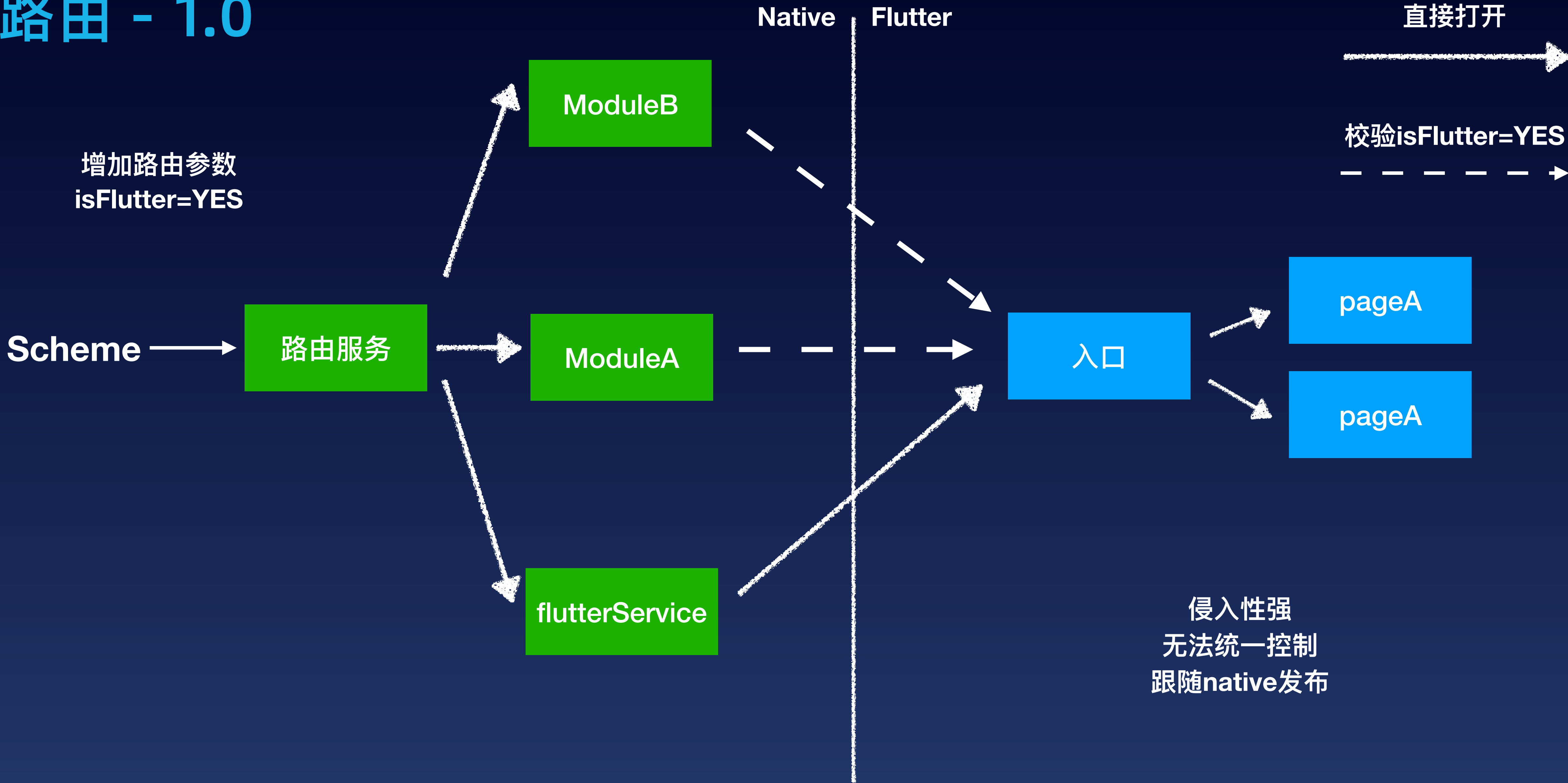
版本



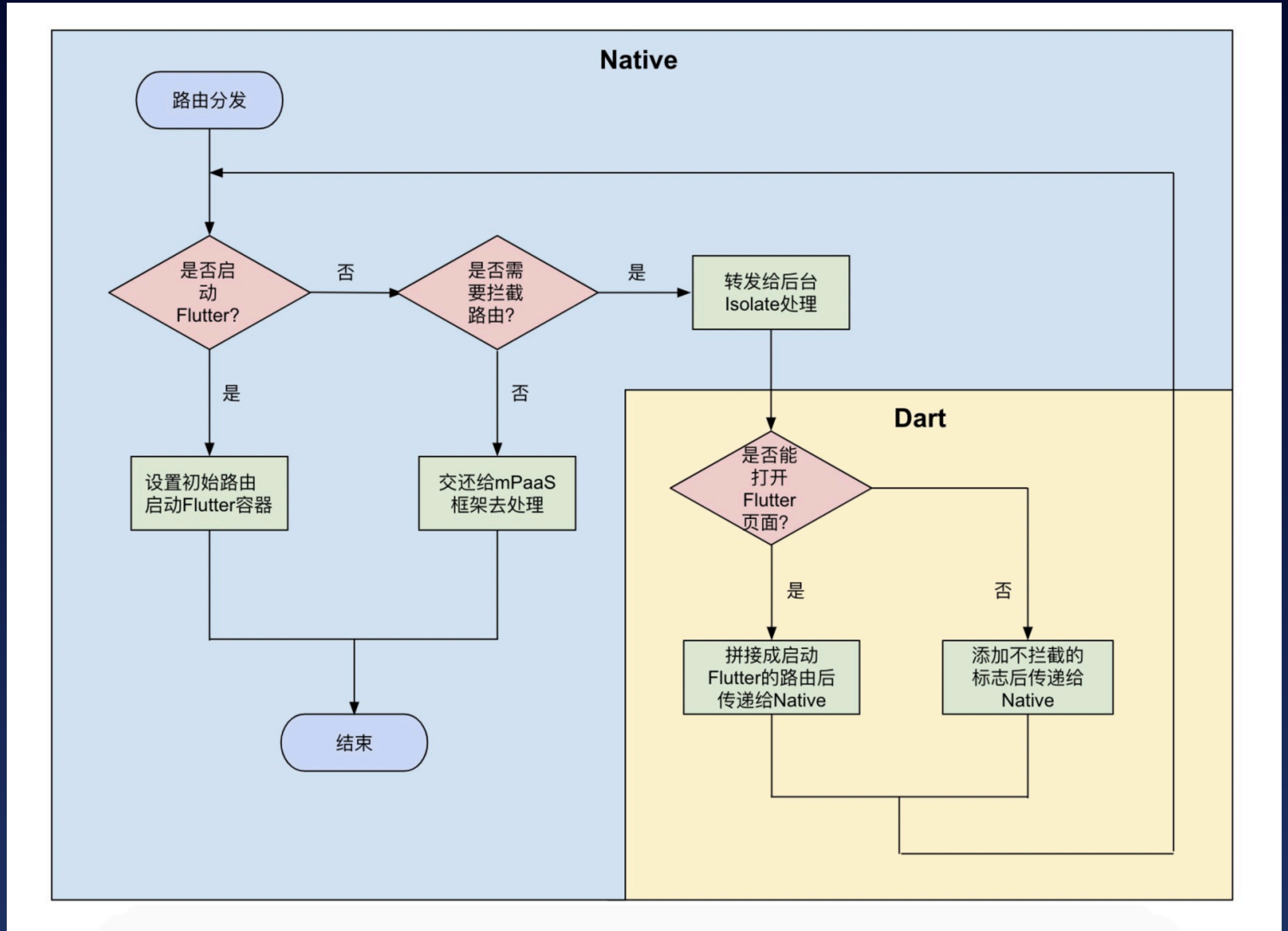
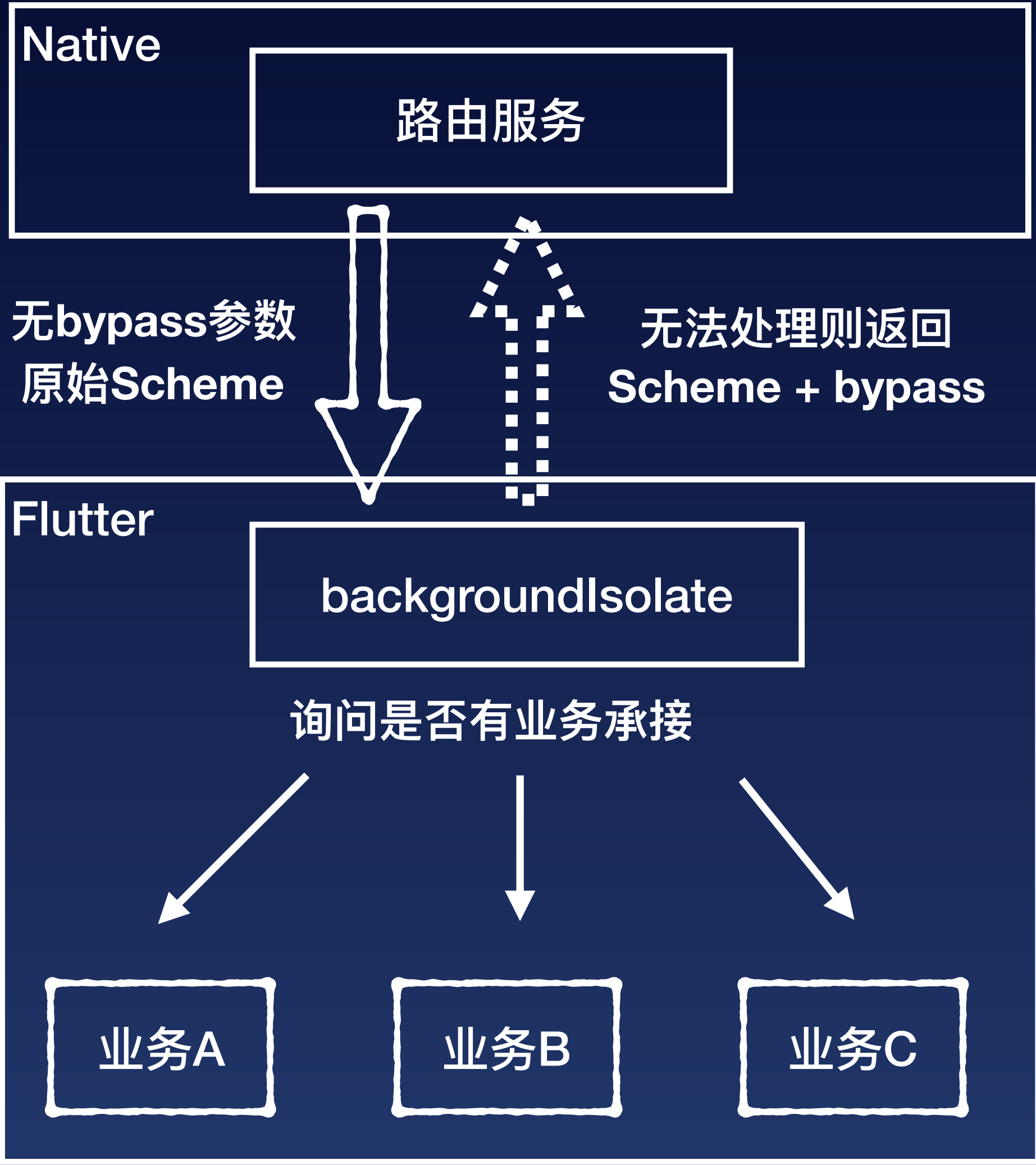
构建脚本



路由 - 1.0



路由反转 - 2.0



无侵入
Android动态上线

风险降级

区块业务

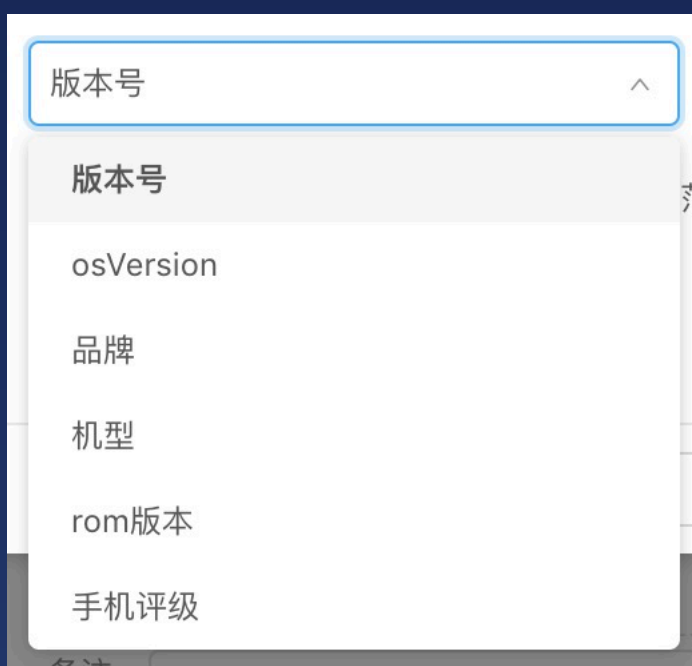


普通业务

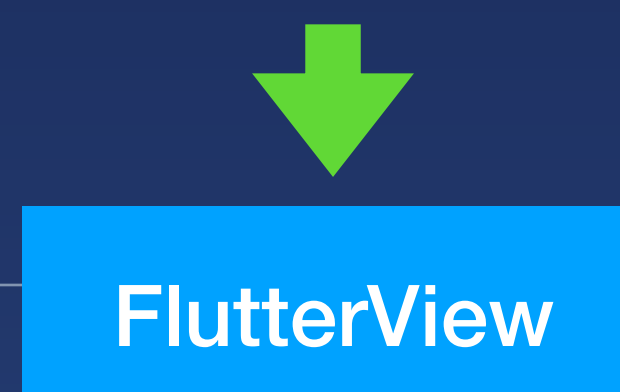
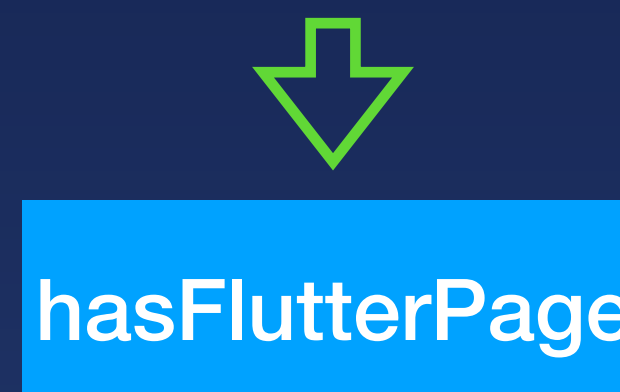
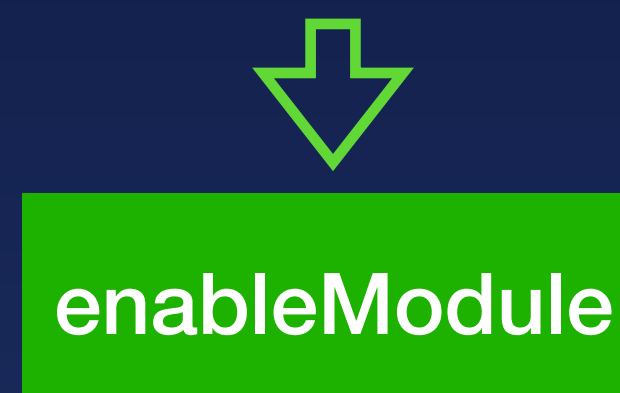
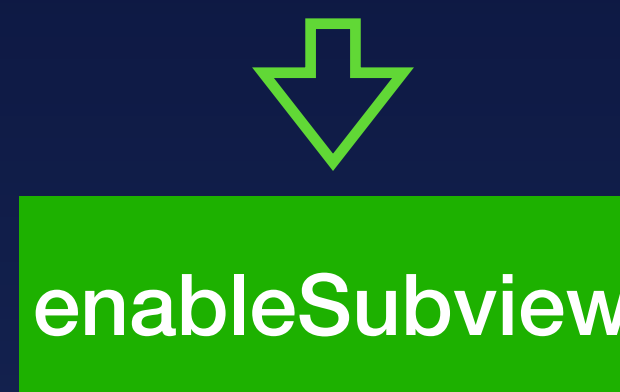
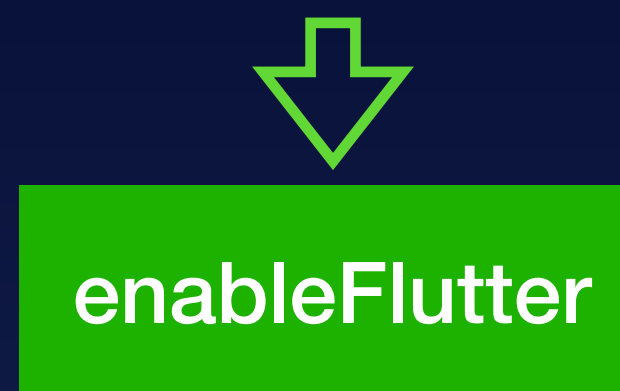


降级维度

通用规则：平台，白名单，百分比



Begin



降级

降级

降级

降级



Native

Flutter

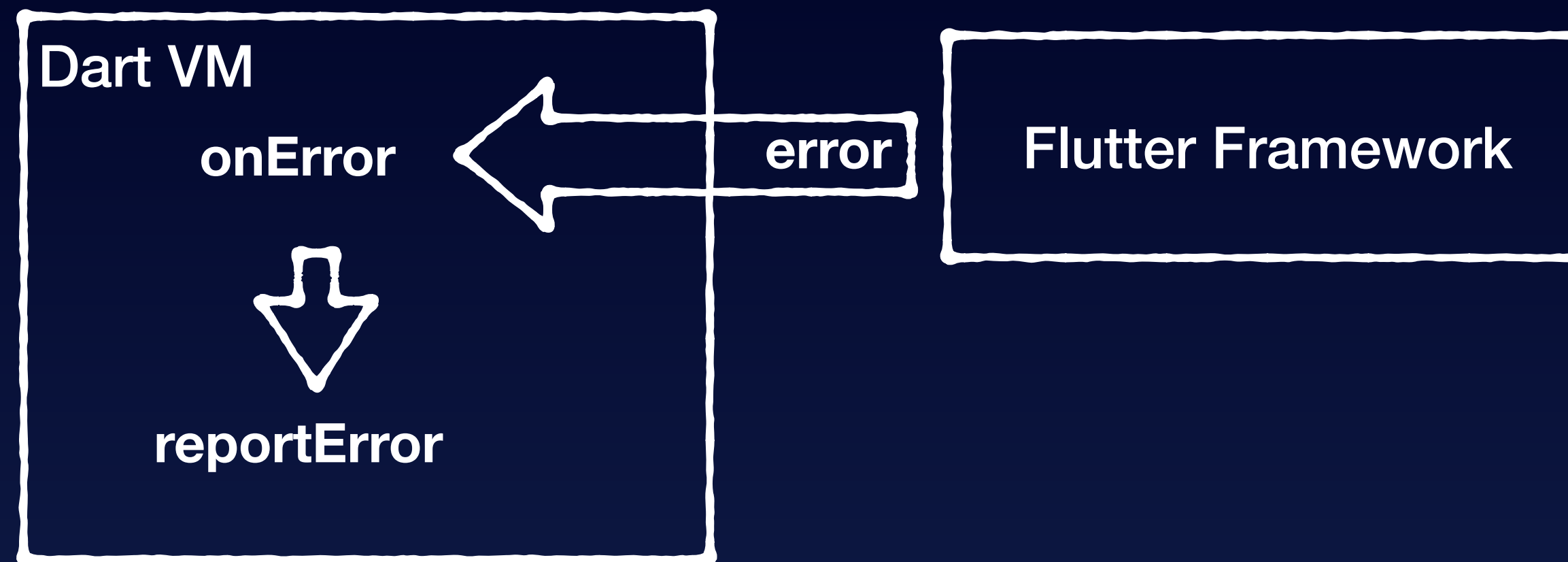
运维监控

```
main() {
  initBundles();

  runZoned(
    () {
      if (window.defaultRouteName == ROUTE_SCHEME_SERVICE) {
        SchemeService.startService();
        return;
      }
      runApp(GreatApp());
    },
    onError: isInDebugMode ? null : (error, stackTrace) {
      reportError(error, stackTrace);
    },
  );

  FlutterError.onError = (FlutterErrorDetails details) {
    if (isInDebugMode) {
      FlutterError.dumpErrorToConsole(details, forceReport: true);
    } else {
      Zone.current.handleUncaughtError(details.exception, details.stack);
    }
  };
};
```

- 线上运行稳定性监控
- 性能指标监控 - 卡顿, 内存等
- 清洗错误数据, 生成数据报表



```
#0 Canvas._drawImageRect (dart:ui/painting.dart:3480:72)
#1 Canvas.drawImageRect (dart:ui/painting.dart:3458:5)
#2 paintImage (package:flutter/src/painting/decoration_image.dart:442:14)
```



Top异常 / Top页面 更多

异常内容	统计次数	较昨日	用户数	近30日趋势	操作
The following NoSuchMethodError was thrown: The method 'markNeedsBuild' was called on null. Receiver...	2	↑ 100.00%	2		分析 明细
The following message was thrown: Object has been disposed.	1	↑ >99999%	1		分析 明细

目录

- 业务背景
- 建设路线图
- 复杂生态融合
- 多业务技术体系
- 总结与展望

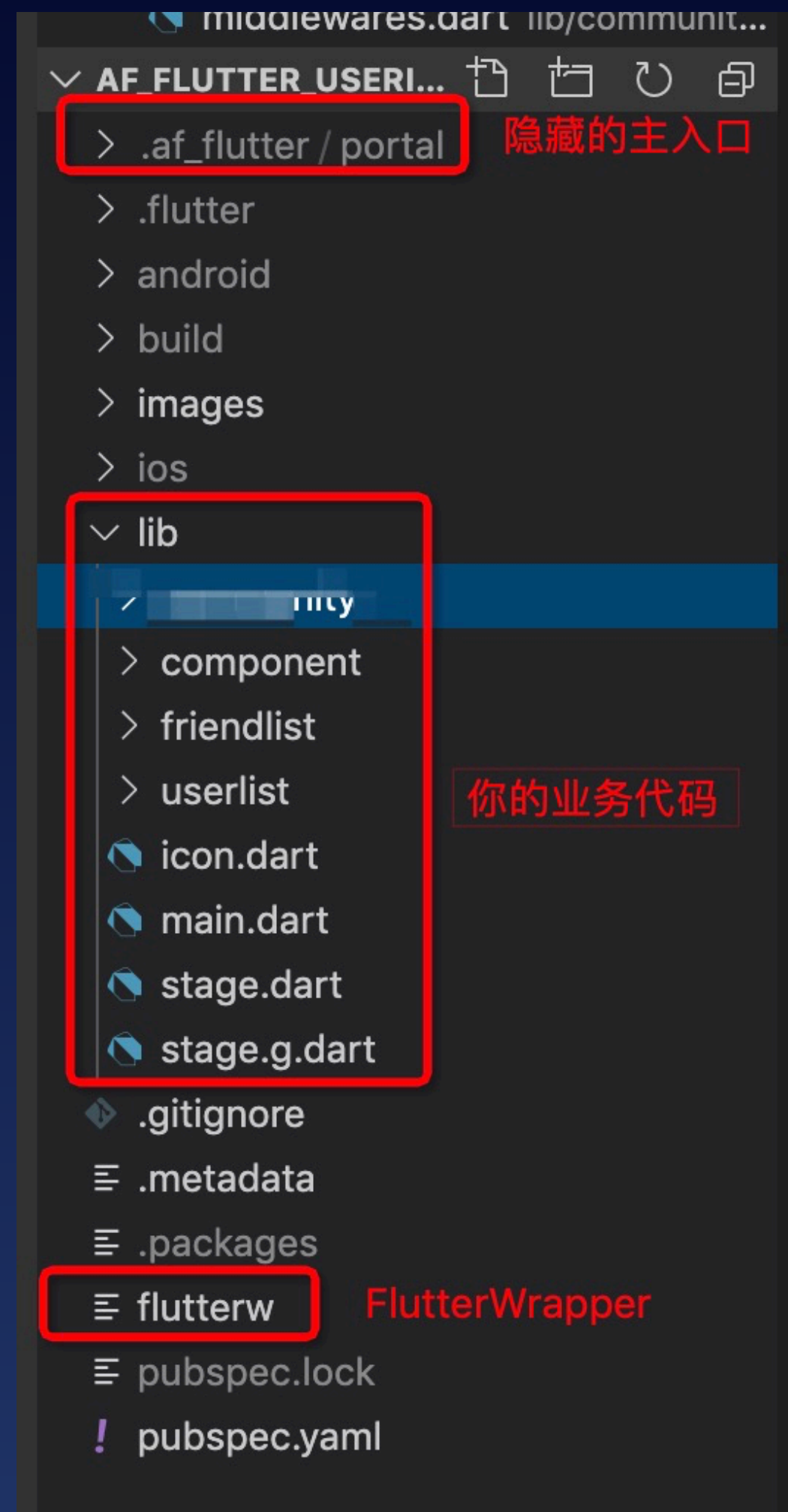
多业务技术体系要解决的问题

- 业务工程隔离
 - 代码隔离
 - 工程结构隔离
- 开发工具建设
 - 控制工具**维护成本**
 - 工具建设过程是**双向的**
- 多渠道（版本）支持
 - 约束**边界**，保留**定制能力**
 - 不同业务方对能力的述求完全不一样，支持多渠道**将选择能力还给业务方**
- 业务控制：接入方式，降级，灰度.....

业务工程隔离



开发时执行的run, attach
都是从该主工程发起



```
dependencies:  
  af_flutter_setting:  
    git:  
      ref: 848135715009f5c!  
      url: git@gitlab.alipa  
  af_flutter_userinfo:  
    path: ../../..  
  flutter:  
    sdk: flutter
```

flutterw指向的flutter版本
为本地flutter文件夹

开发工具建设指南 - AFF

Commands

create

build

attach

clean

run

publish

Flutter Wrapper
Installer

Available commands:

```
attach    Attach to a running application.
build     Build the Flutter project.
clean     Delete the build/ directories.
create    Create a new flutter project.
help      Display help information for aff.
hpm       Flutter app developing kit.
patch     Build the dynamic update patch.
publish   Publish your bundle to portal repository.
run       Run your flutter app on an attached device.
```

Tools

DevTools

SimUtil

FVM

Env

Dart

Node

...

基于FlutterTools 创建调试发布一条龙

DevTools: 能力更强的调试工具

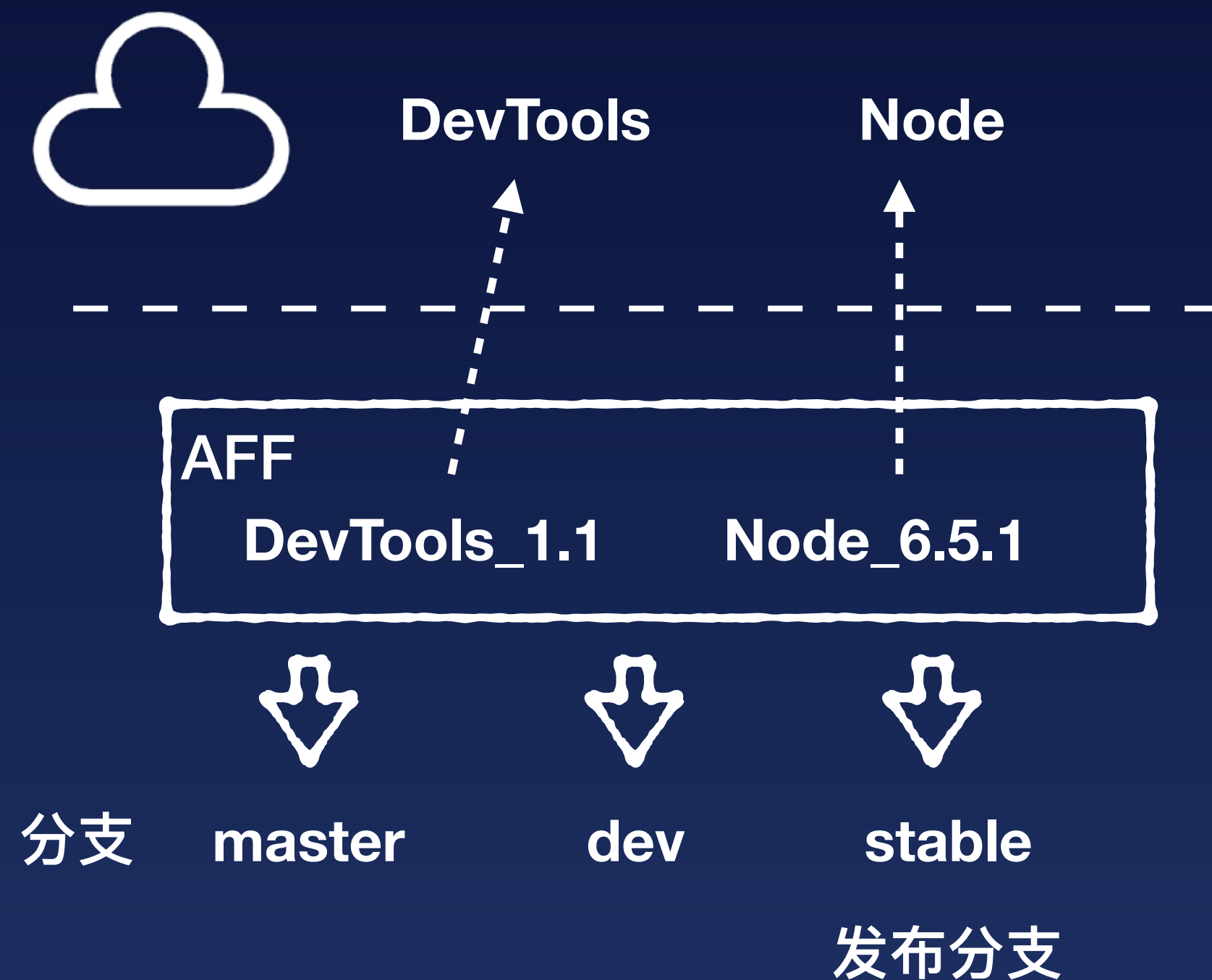
SimUtil: 集成模拟器操作

FVM: FlutterVersionManager

Publish时生成**新的应用版本**

工具建设 - Tip

在线更新



双向建设

```
# If not run in ~/.aff directory, then we assume we are running in development mode
if [[ "$TOOLS_ROOT" != "$AFF_HOME" ]]; then
  export AFF_DEV=true
fi
```

AFF未在默认安装目录~/.aff执行时进入开发模式
开发模式时，不强制要求在stable分支执行AFF
业务可自行开发cli与更新DevTools, Node

AFF运行时强制修改分支为远端发布分支

支持多渠道

本质上是对Flutter版本的管理 - FVM

实现原理：在本地文件夹中缓存多个版本

业务工程包

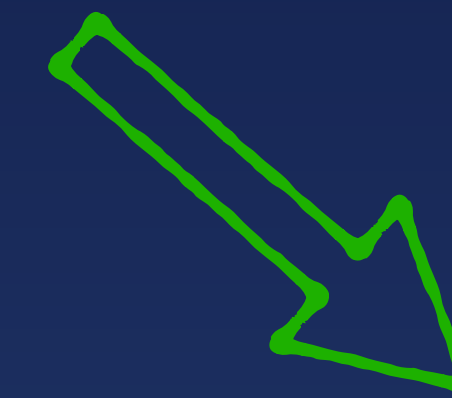
`aff create xxx --channel hk_wealth`

```
name: ant_flutter_userinfo
description: Manger Userinfo and detail pages.
version: 0.0.1
app_id: hk.antfin.wealth
channel: hk_wealth
```



AFF

```
...
* hk_wealth
master
performance_overlay
release_backup
stable
test_nic.xk
```



FVM

1.5.4-hotfix.2

1.7.8+hotfix.4

1.9.1+hotfix.6

目录

- 业务背景
- 建设路线图
- 复杂生态融合
- 多业务技术体系
- 总结与展望

总结

- 复杂生态融合
 - 混合工程接入
 - 平台桥接
 - 混合栈，路由统一
 - Flutter代码工程化
 - 模块化
 - 定义版本
 - 反向路由
 - 平台和业务多角度降级
 - 监控能力
 - Dart Error
 - Flutter Frame Error
- 多业务支撑
 - 业务隔离
 - 代码隔离
 - 工程结构隐藏
 - Flutter操作工具化
 - 一站式：无需感知native代码
 - 可参与共建
 - 支持多渠道

后续规划

- Flutter Web跟进
 - 达到可用标准后，这是一种平台认可的动态发布方式
- AFF的由内向外
 - 服务内部到服务社区
- Faas的持续探索
 - Dart在Server端的表现得到认可，值得继续投入
- 新型终端
 - Google新系统，5G新生态，新的时代在招手

THANKS

GMTC
全球大前端技术大会