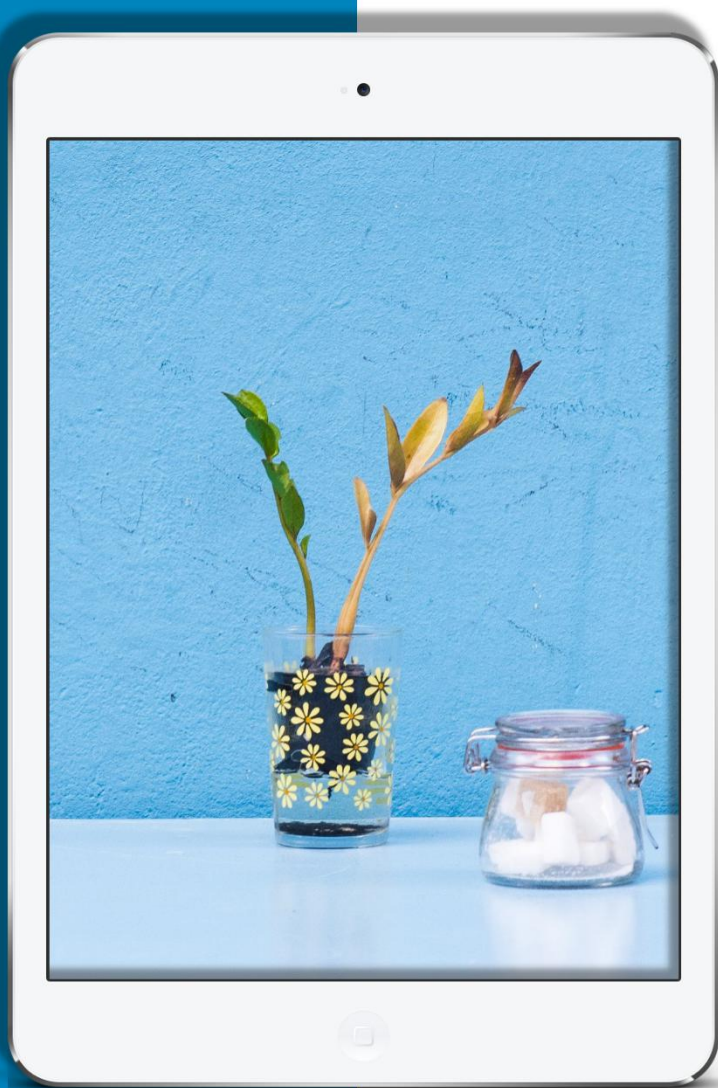


打包构建系统设计





打包构建系统

根据目前公司主要的业务以及开发流程,针对我们公司自己的实际需求设计了一套自己的打包构建系统,系统包括以下主要功能:

1. App 多环境打包;
2. App 自动发布热更新;
3. App 管理托管;

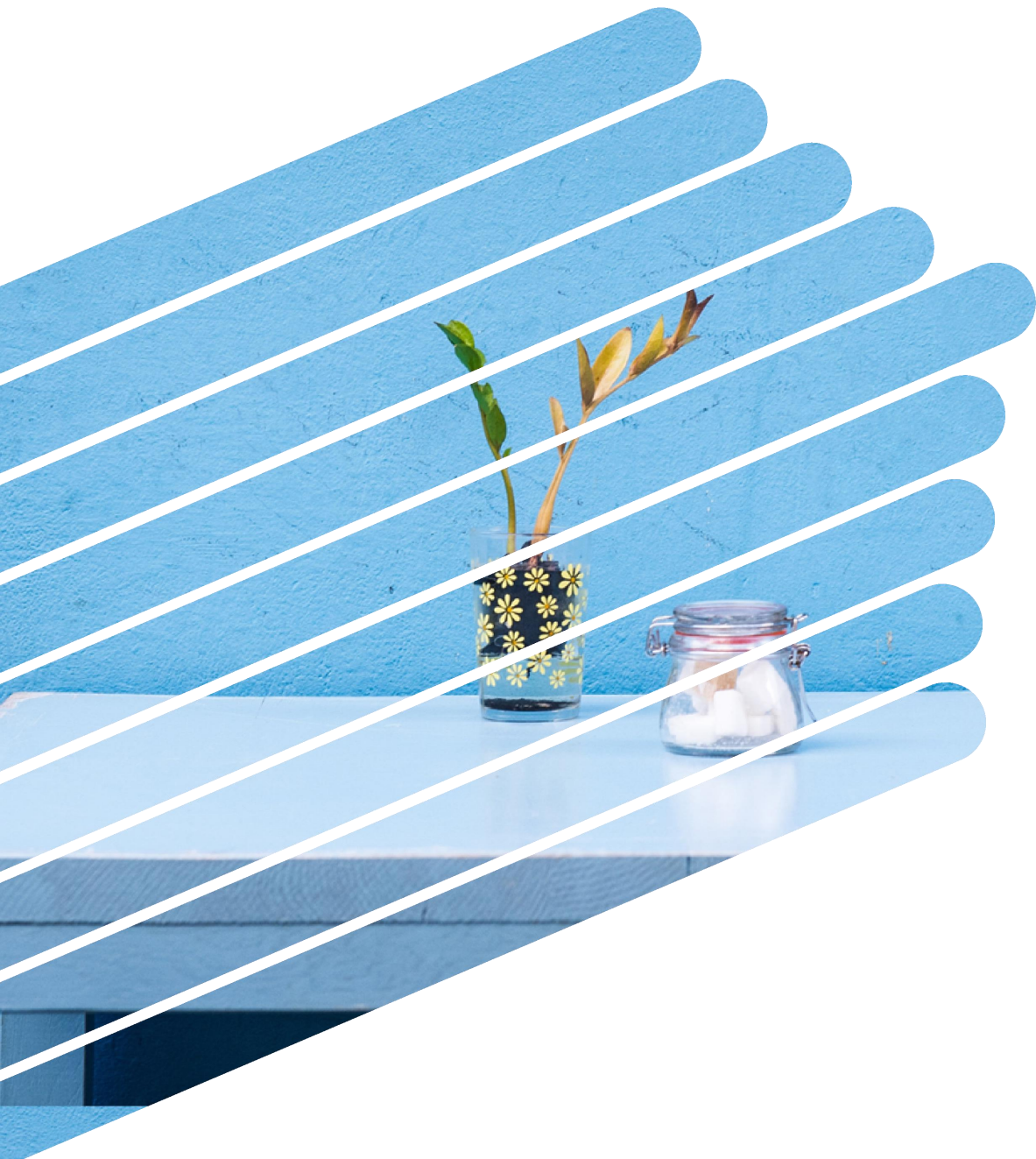
CONTENTS

由来 1

功能点 2

主要实现点 3

参考对象 4





1. 耗时长,效率低;
2. 重复性多,人工成本高;
3. 出错性概率大;

手动打包

VS

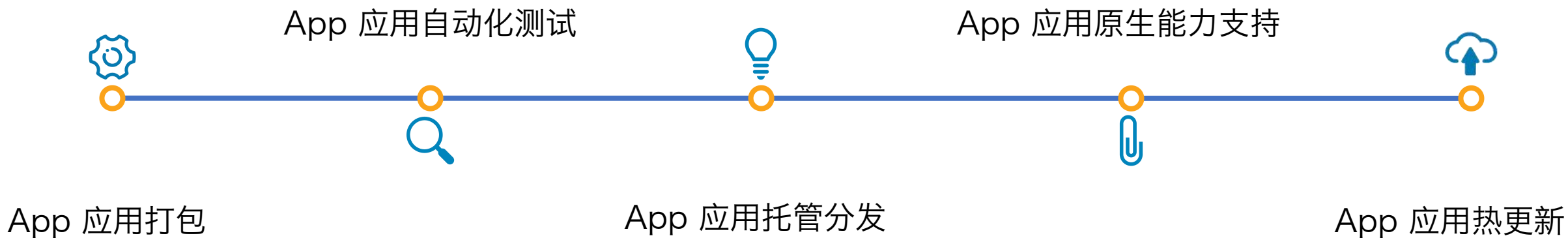


1. 自动化打包,解放双手;
2. 配置好后,一键操作;
3. 可配置一键远程打包;

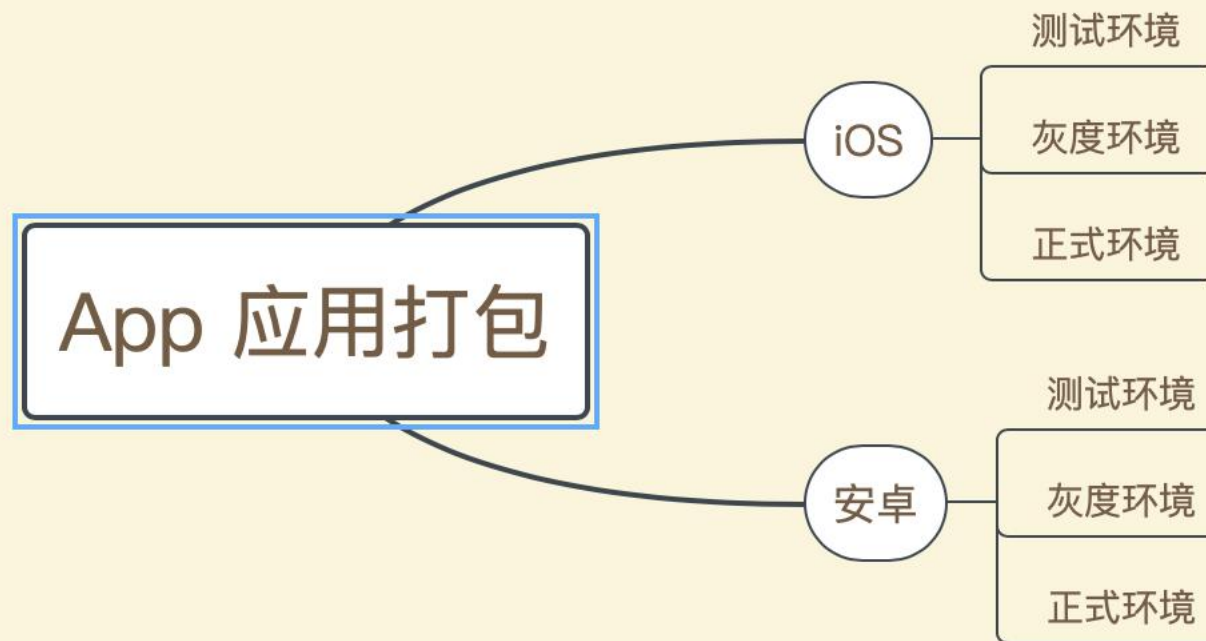
自动打包

打包构建系统由来

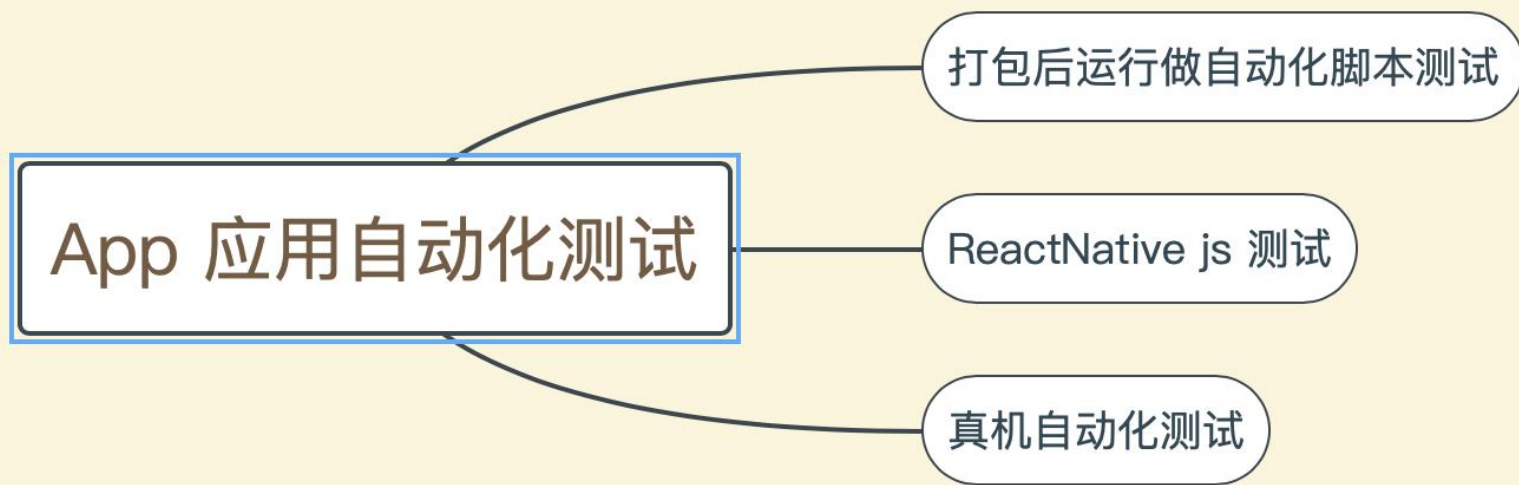
应用打包,发布热更新代码都是比较重复性的操作,而且应用托管和热更新的管理,所以就想做一套打包构建的系统



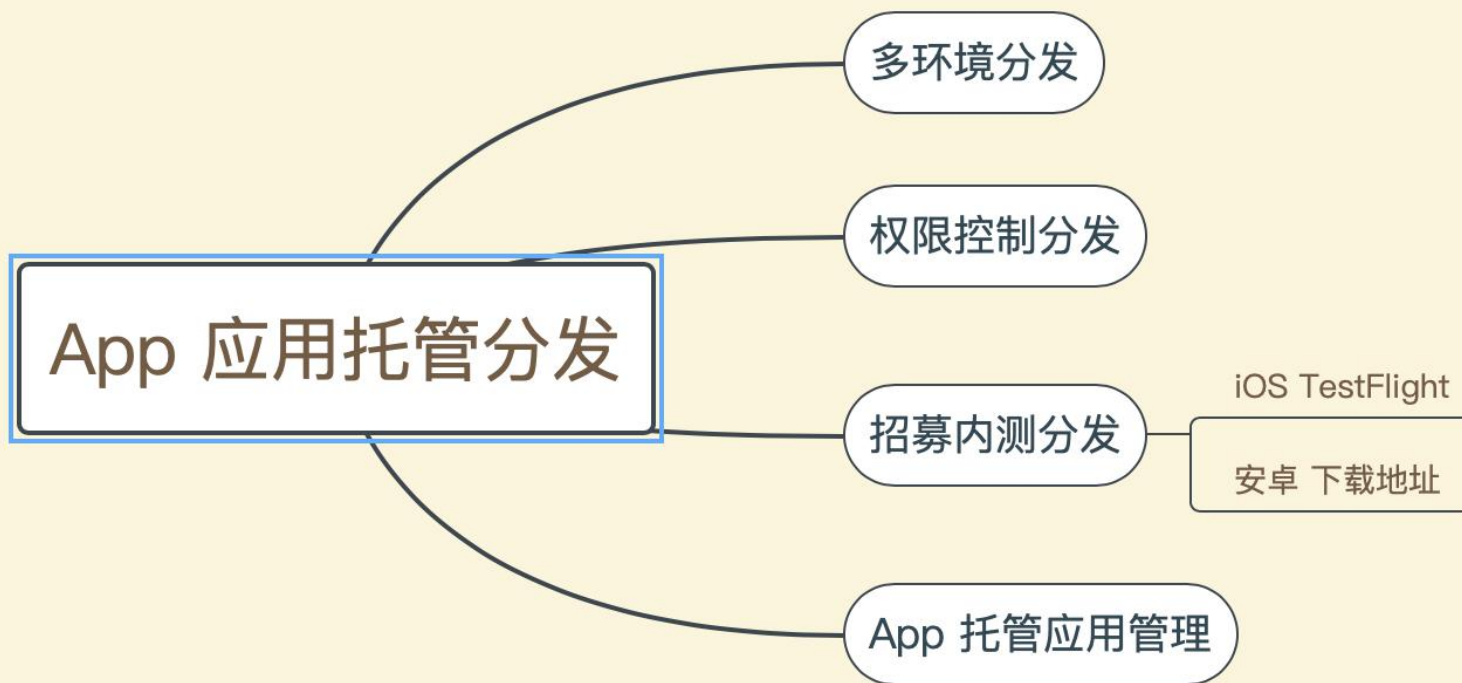
功能点 – App 应用打包



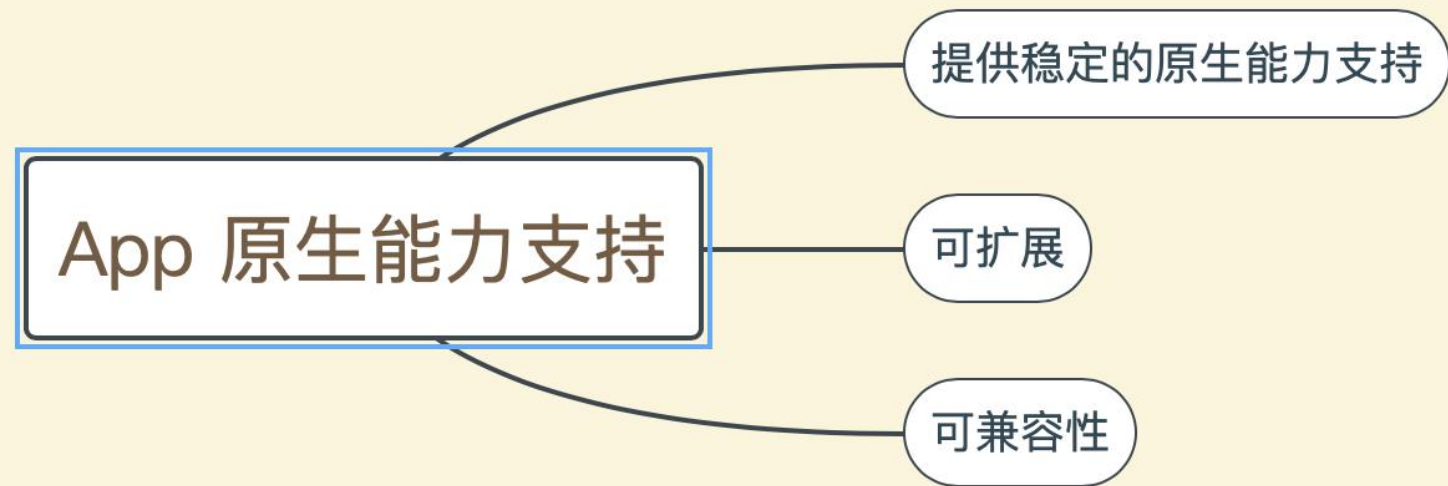
功能点 – App 应用自动化测试



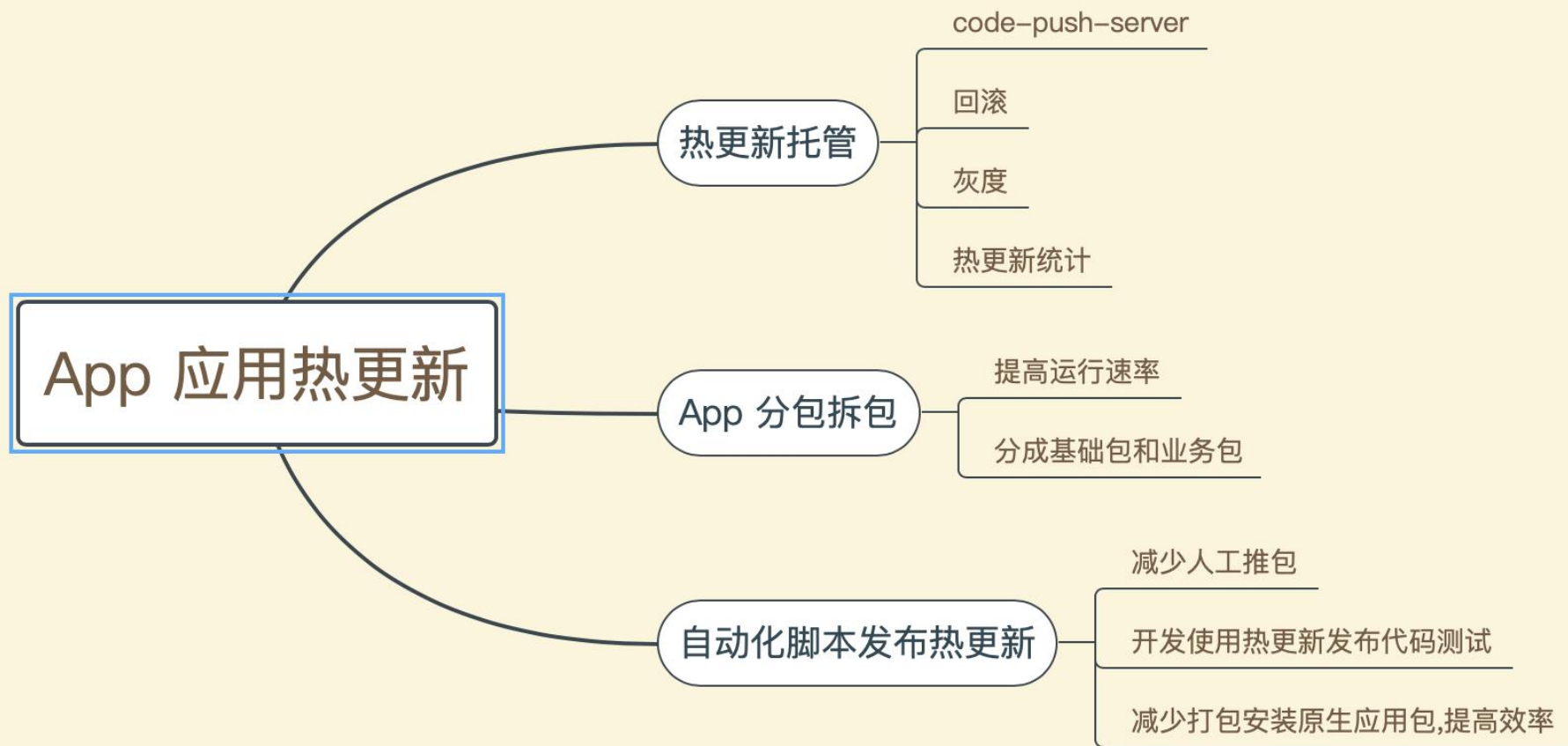
功能点 – App 应用托管分发







功能点 – App 原生能力支持



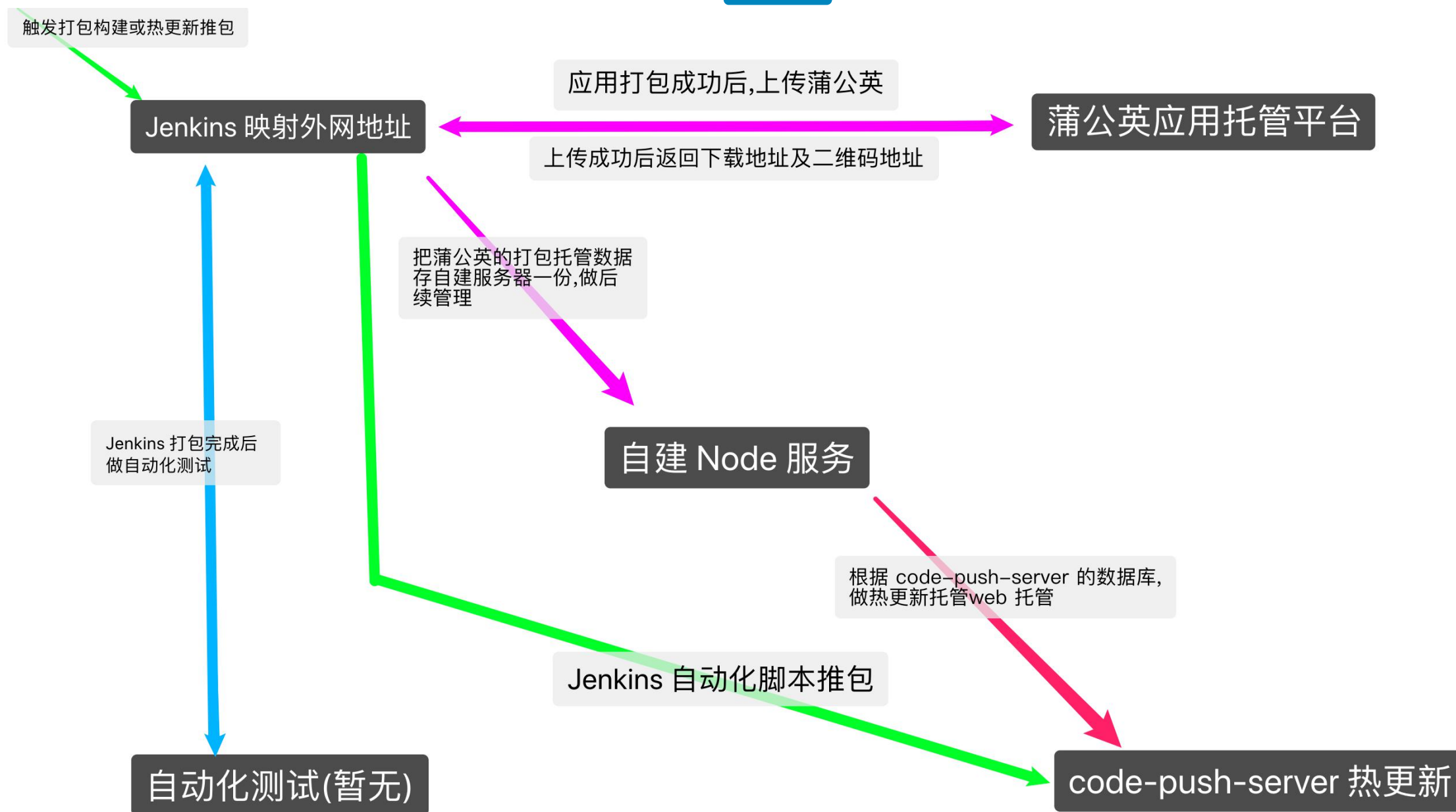
功能点 – App 应用热更新



主要实现点

App 应用打包		Jenkins 和脚本基本实现
App 应用托管		蒲公英托管(实现部分功能)
App 应用自动化测试		暂无
App 热更新		code-push(实现部分功能)

主要实现点



THANKS

