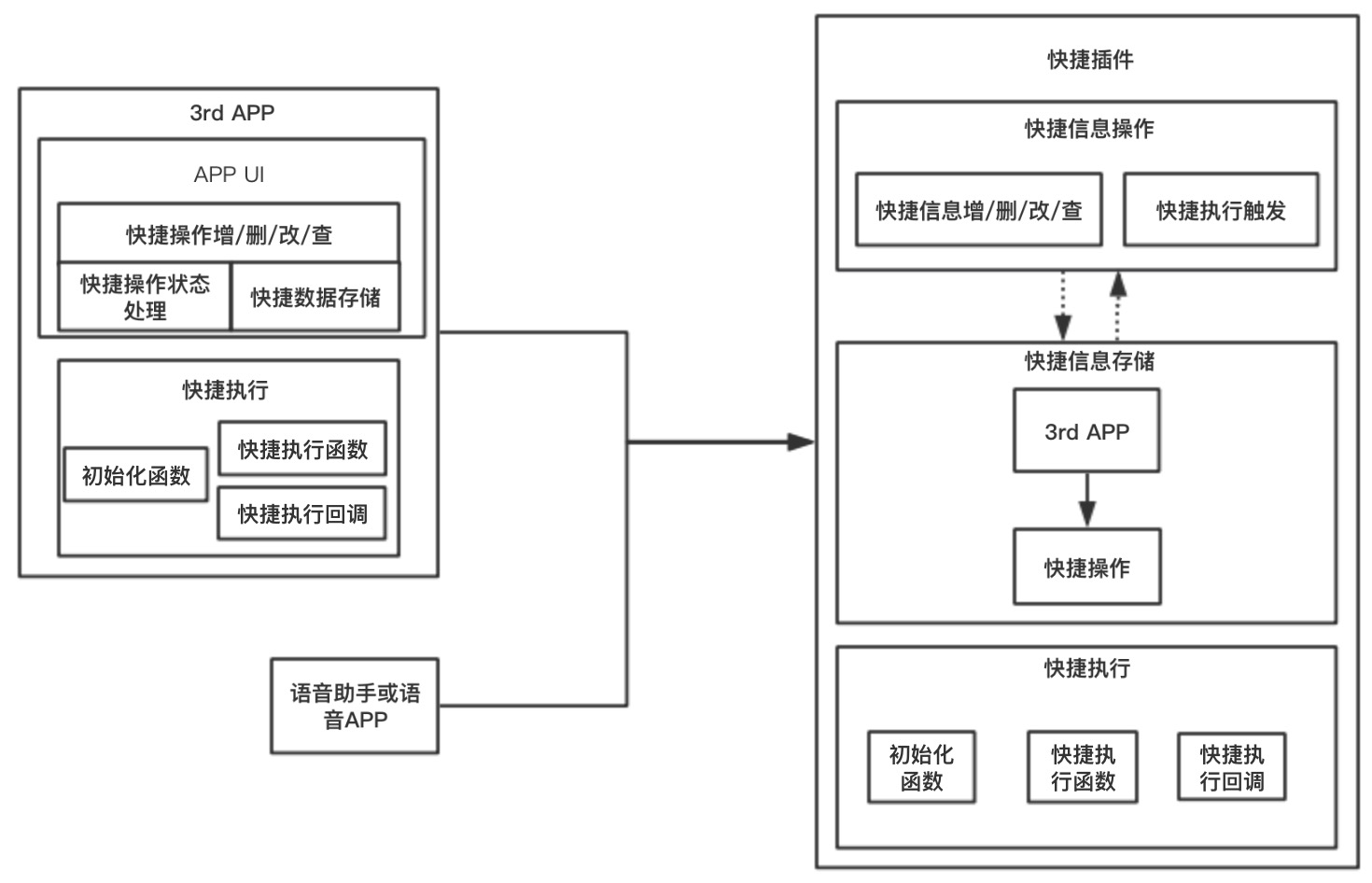
快捷控制设计文档

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修订内容 | 时间 | 修订人 |
| 1.0 | 创建文档 | 2019/5/29 | 邰光源 |
| 1.1 | 框架描述修改 | 2019/6/1 | 邰光源 |
| 1.2 | 更新框图 | 2019/6/5 | 邰光源 |
| 1.3 | 集成h5文档 | 2019/12/26 | 谢晨华 |

# 概述

通过调用语音助手或语音APP，在不启动APP的情况下完成对智能设备的快捷控制。

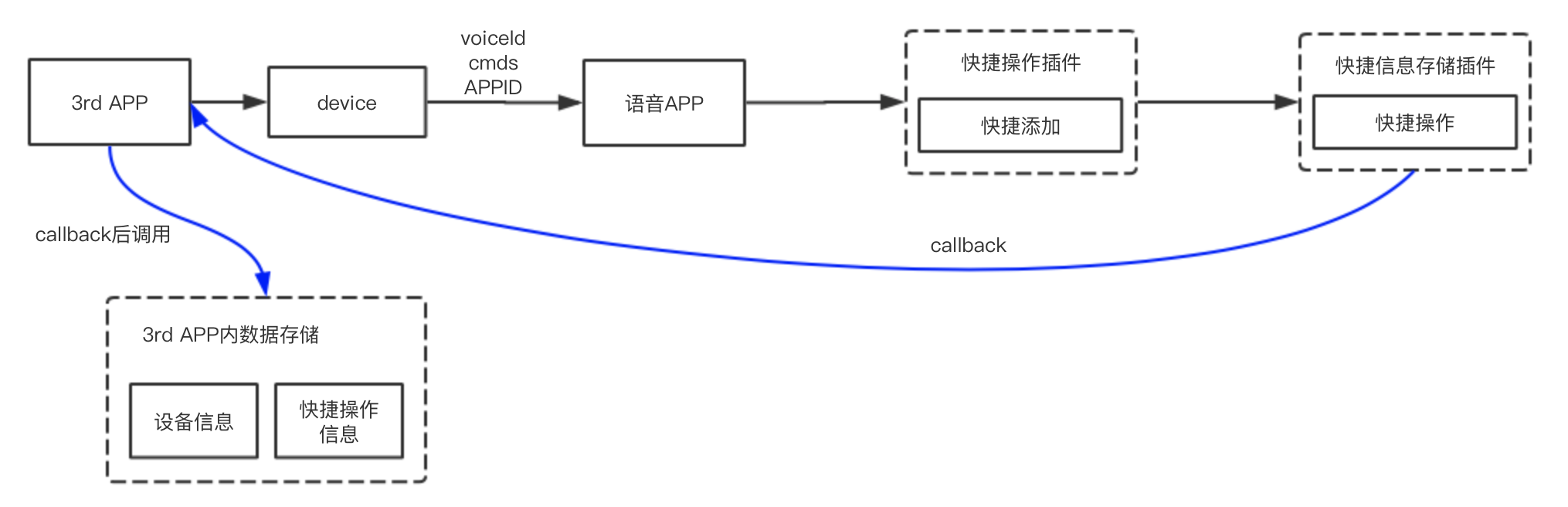
# 快捷控制整体框架



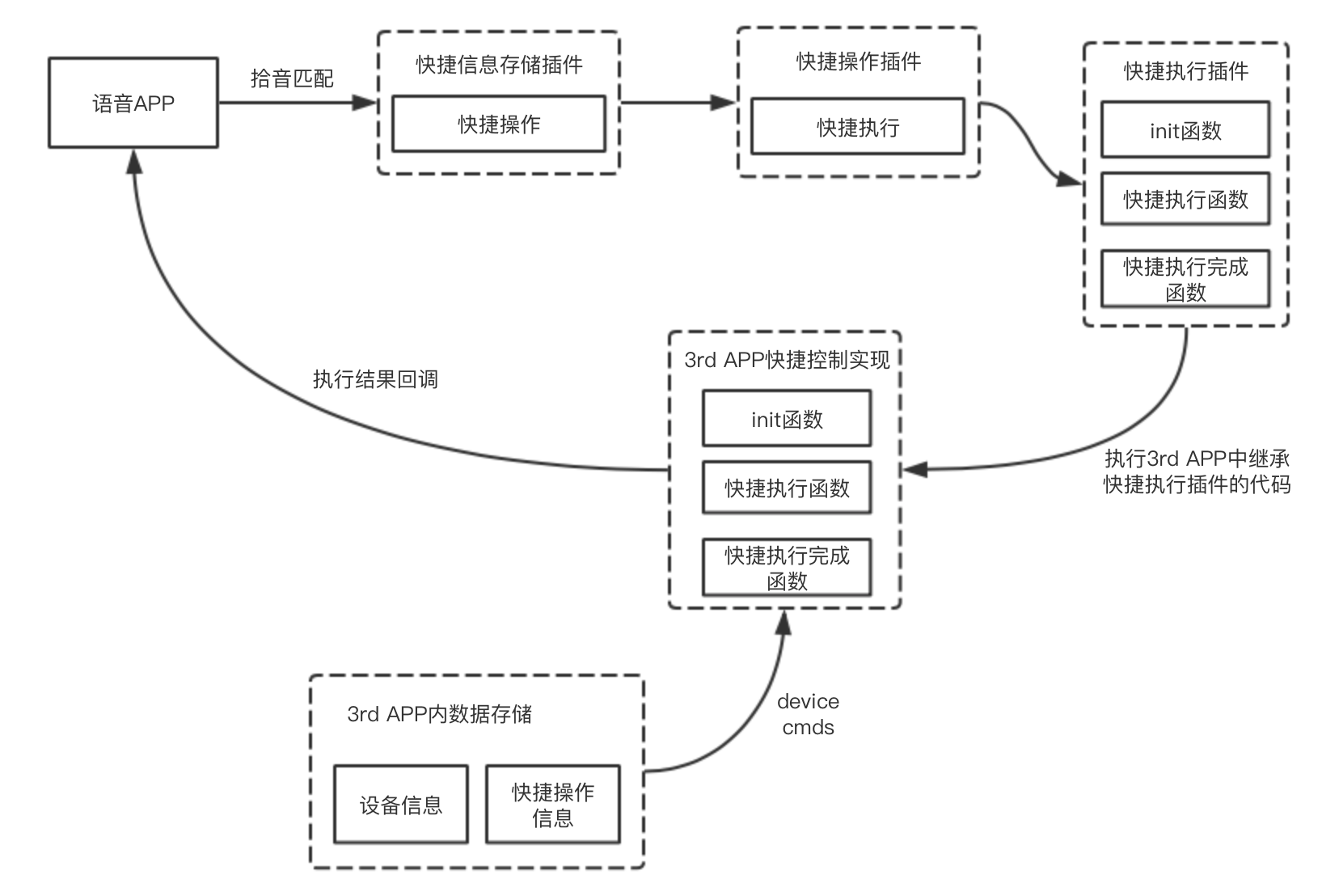
第三方APP在使用语音APP调用“快捷操作插件”及“快捷信息存储插件”完成快捷控制的信息录入。

在使用语音APP完成快捷控制时，需要先起“快捷控制服务”，在“快捷信息存储插件中”查找对应第三方APP的快捷操作信息，再调用“快捷操作插件”的“快捷执行”方法，通过“快捷执行”方法调用“快捷执行插件”完成快捷控制操作。

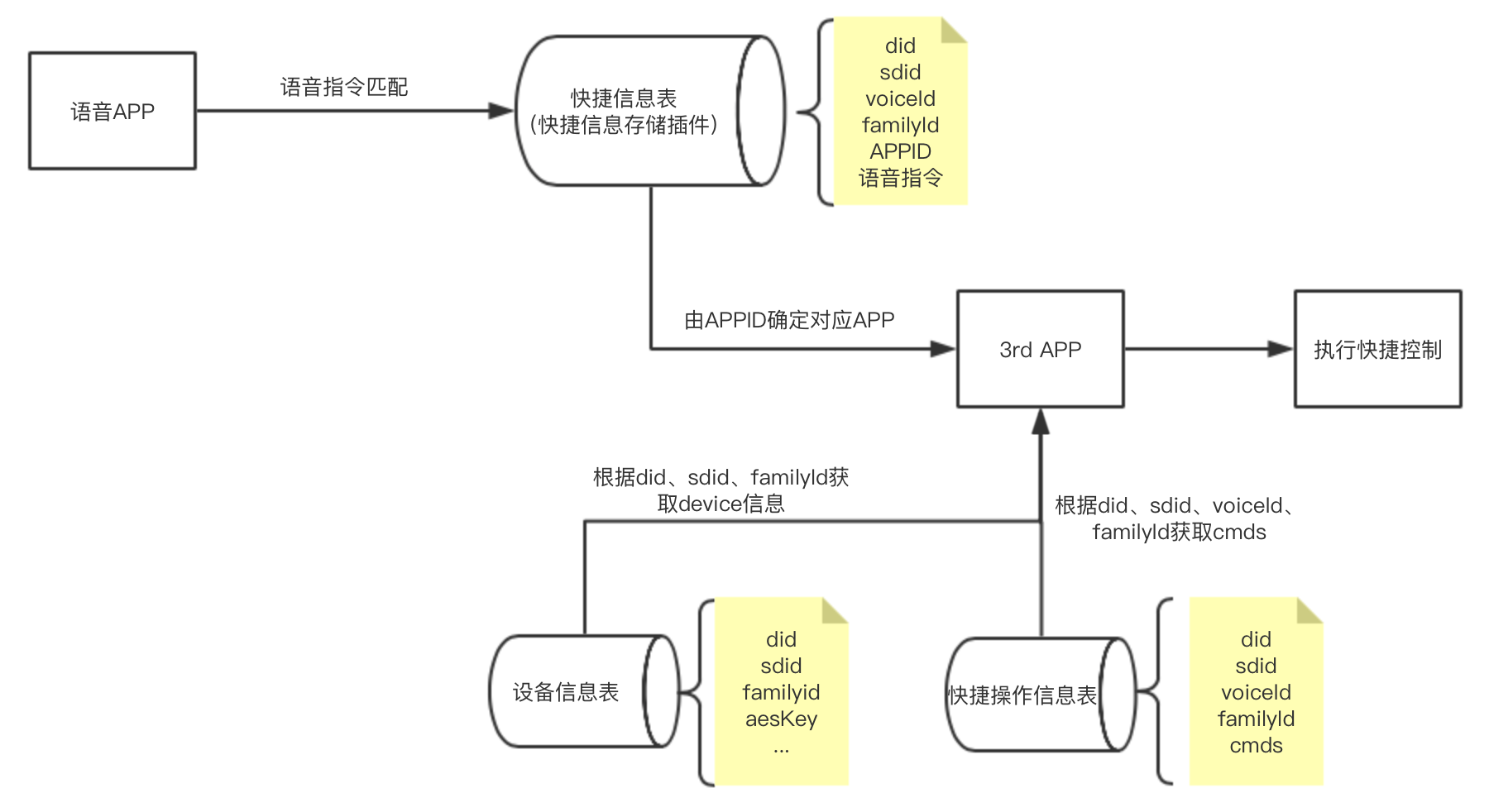
**快捷操作添加**



**快捷操作执行**



**数据关联关系**



# 数据库设计

完成快捷控制，需要用到两张表，一张是完整的设备信息表（包含did、sdid、familyId、aeskey、pid等），一张是快捷操作与设备信息的关联表。设备信息表不做过多阐述。

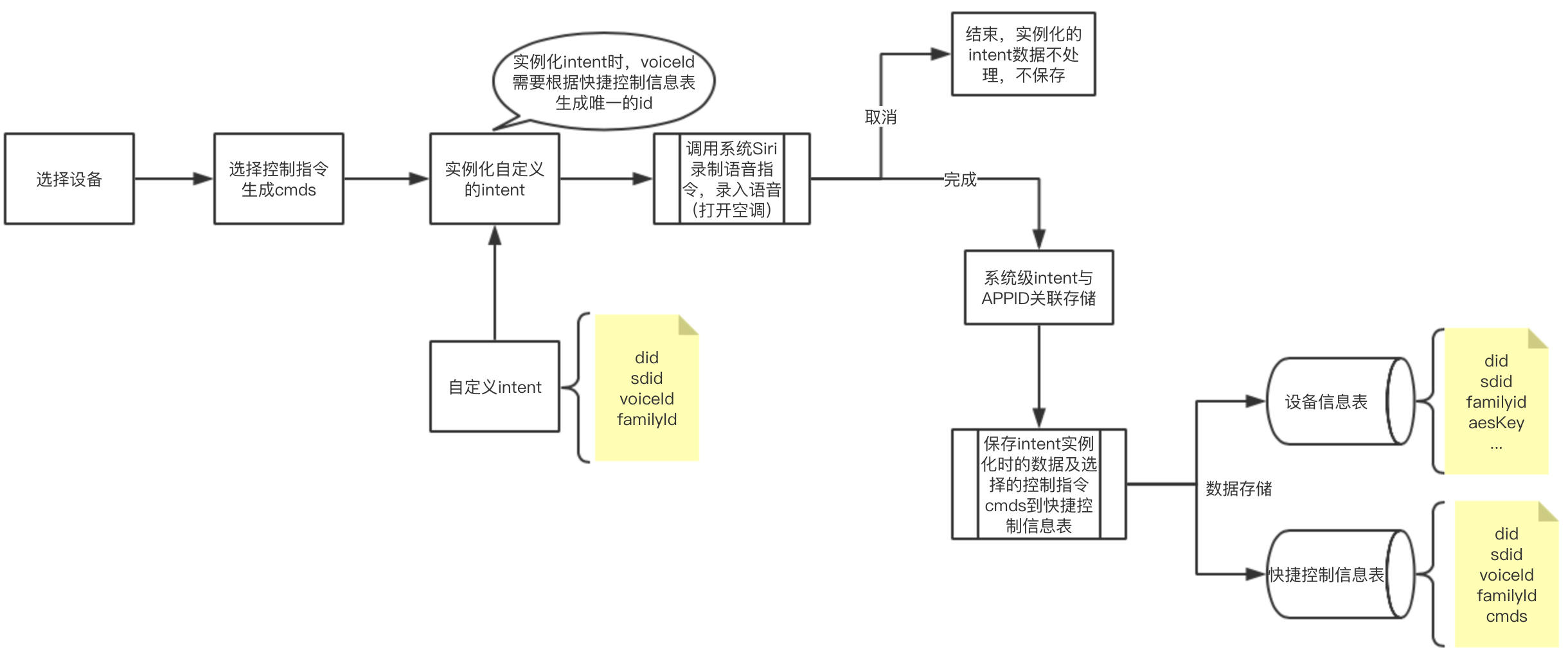
快捷操作信息数据库表结构

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| voiceId | long | 对应快捷控制的voiceId，native自动生成。 |
| did | varchar(32) | 设备did |
| sdid | varchar(32) | 设备的sdid |
| familyId | varchar(32) | 设备所在家庭id，防止单个设备属于多个家庭的情况发生 |
| cmds | varchar(512) | 快捷控制指令，通过appsdk发送；也可通过http接口发送，如云定时中的延时功能。 |

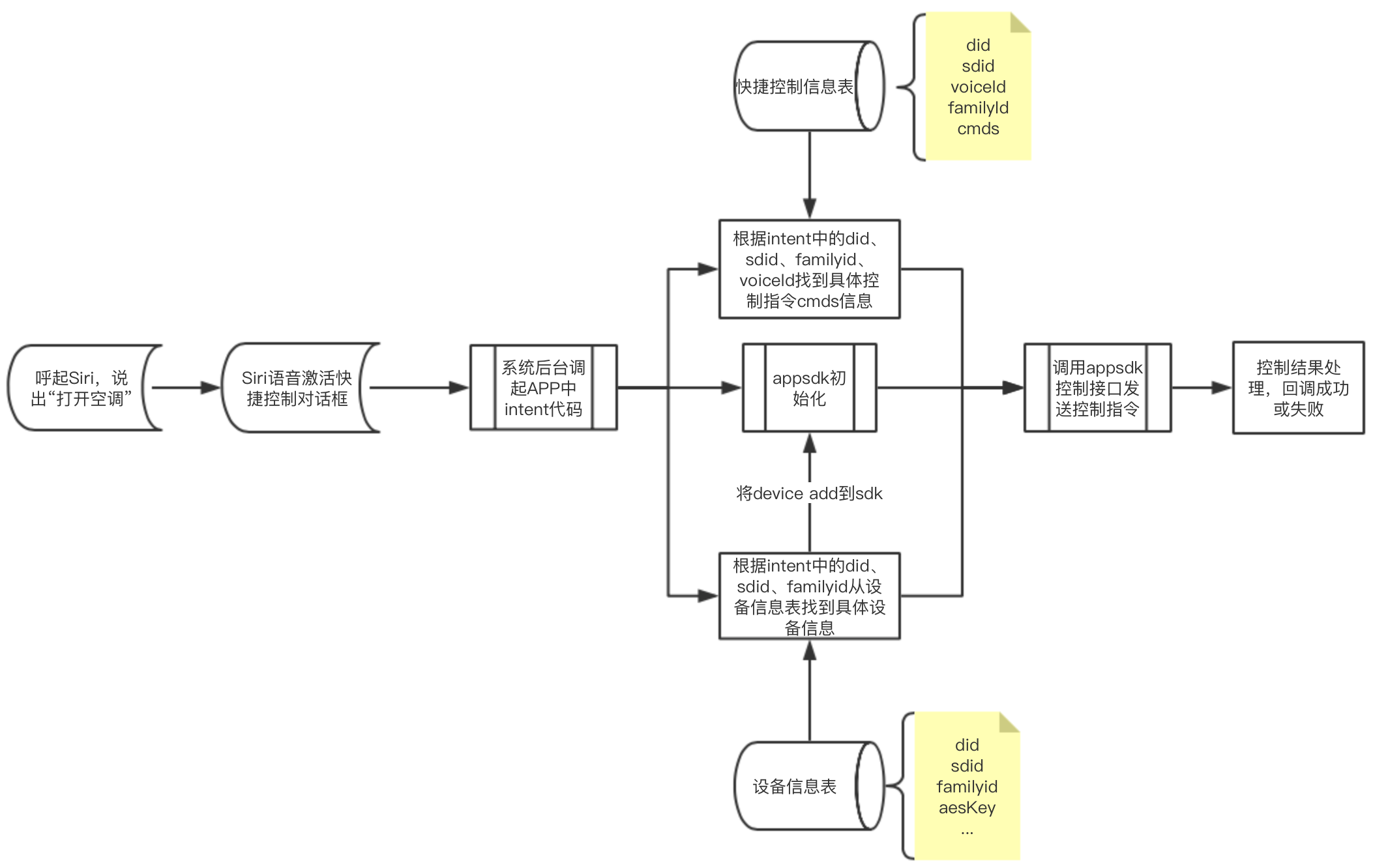
# iOS快捷操作的实现

以iOS为例，Siri Shortcuts是iOS 12中引入的新功能，允许开发人员使用Siri添加自定义的快捷方式，以便在未启动APP的情况下使用语音短语完成某些特定的快捷操作。

## Siri快捷控制添加



## Siri快捷控制实现

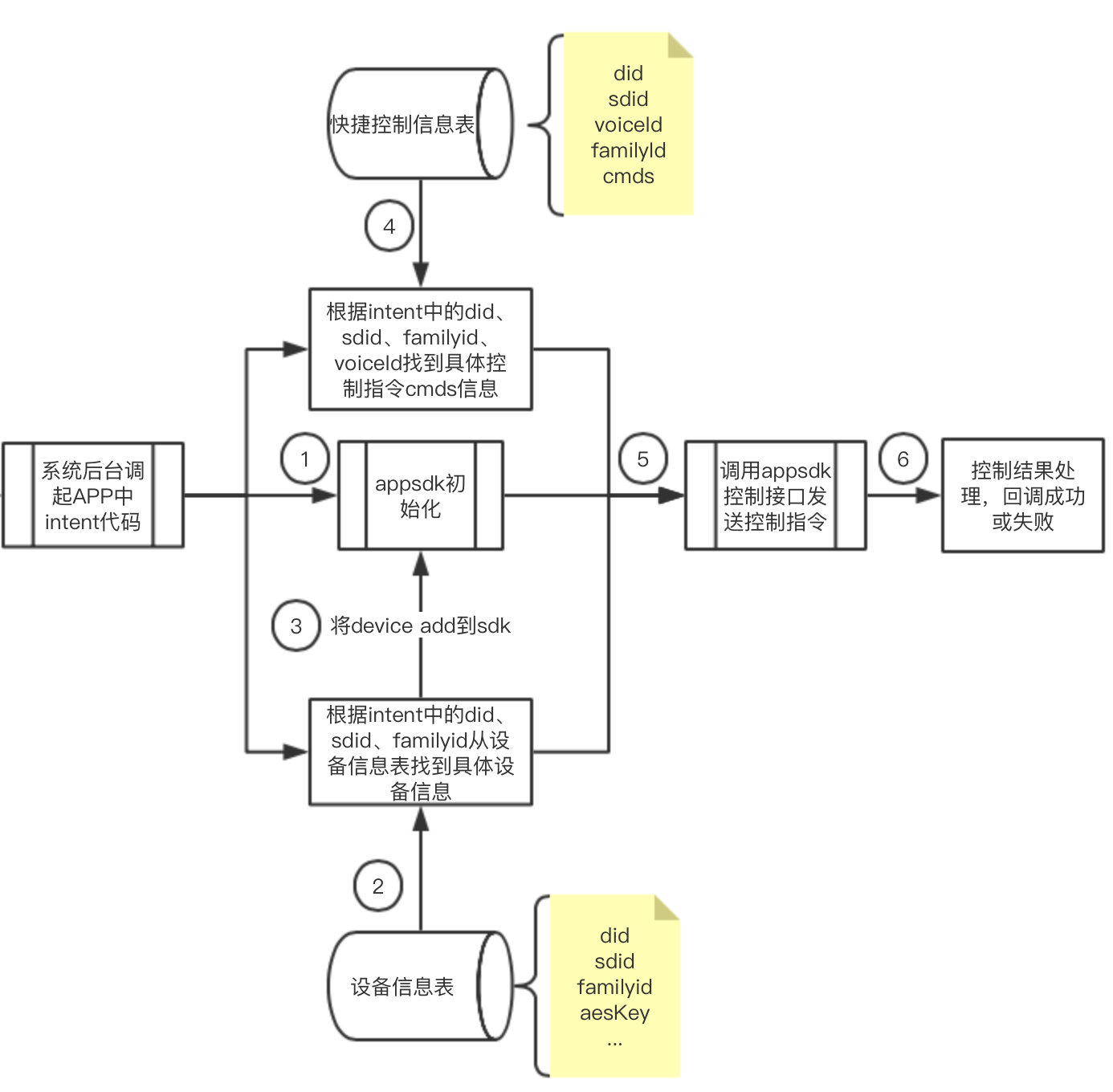


## APP数据通用处理

1. 设备信息表及快捷控制数据表为通用表，这里会对两张表进行封装打包成sdk（暂定SiriShortcutsDataSDK），暴露相应增、删、改、查接口。
2. Intent参数结构统一化处理，如DeviceControlIntent中必须有did、sdid、familyId、voiceId这四个字段。

|  |  |
| --- | --- |
| 字段 | 描述 |
| did | 设备的did信息 |
| sdid | 设备的sdid信息 |
| familyId | 设备所在家庭的familyId信息 |
| voiceId | 设备快捷控制语音id，由H5传入或APP自己生成，且必须保证唯一 |

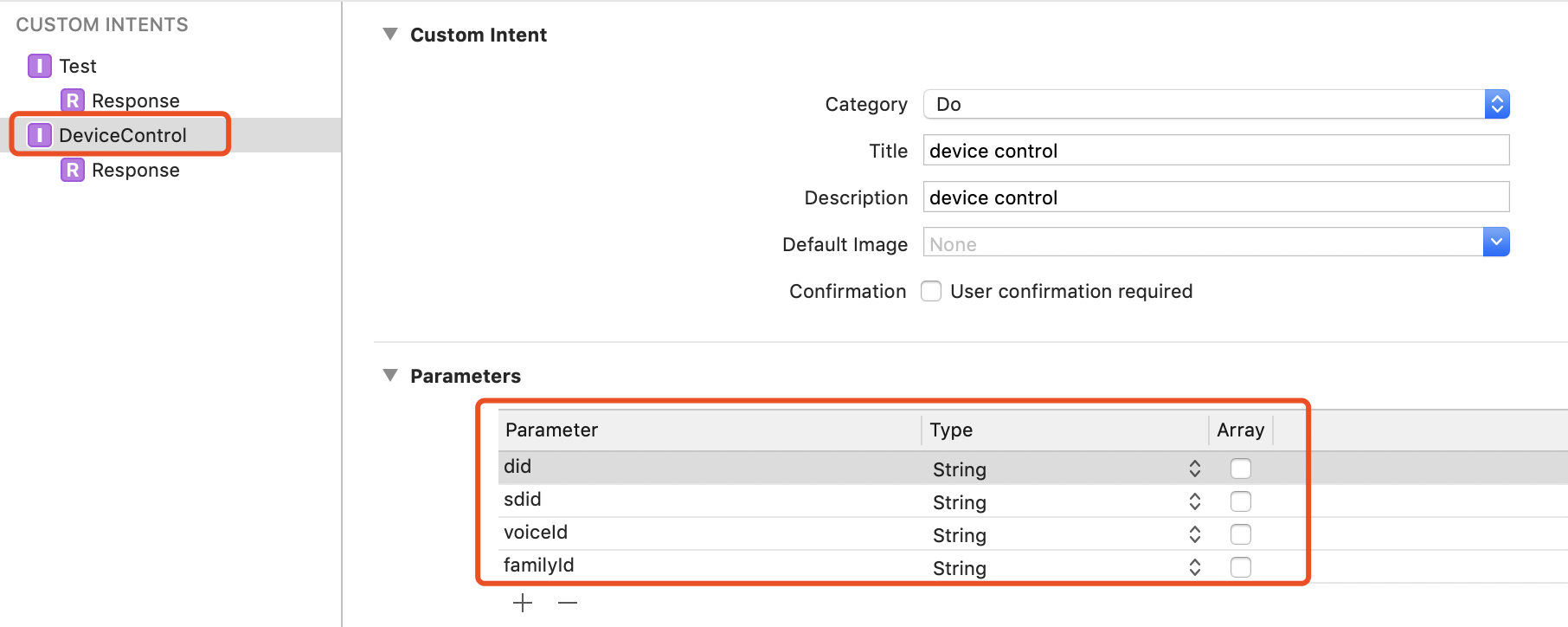
## APP通用控制逻辑处理



1. appsdk初始化逻辑一致，在IntentHandler文件中handlerForIntent方法里进行初始化
2. 从设备信息表中获取device信息。
3. 将device添加到appsdk中
4. 从快捷控制信息表中获取控制指令cmds
5. 调用appsdk的dnaControl接口发送控制指令
6. 处理返回结果

## APP非通用内容（自定义）

1. Intent需要每个工程单独创建，创建Intent时的名称（DeviceControl）可自定义，参数不可自定义，可新增不可删除，必须有did、sdid、familyId、voiceId字段

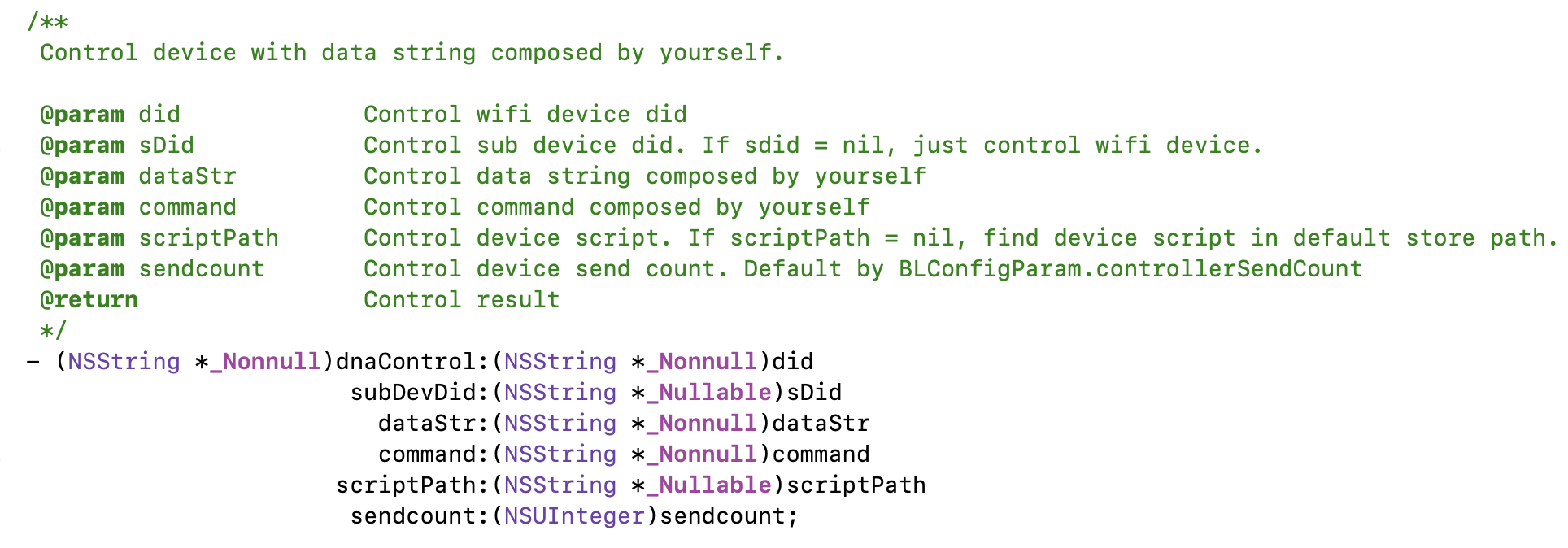


1. Intent UI可自定义
2. 其它除控制实现均为Xcode自动生成的统一内容

# 接口定义

控制相关使用appsdk现有控制接口，由于sdk需要在初始化成功后并且将device加入sdk后才可控制设备，所以在控制时初始化sdk及add device这两个步骤必不可少。延时控制采用云定时接口中的延时任务来实现，发送http请求。添加捷径则使用系统自有接口。

## 控制接口



## 延时控制接口

url:https://域名/appfront/v1/timertask/manage

方法:POST

header:

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 描述 |
| userId | string | 用户userid |
| loginsession | string | 登录获取的loginsession |
| familyId | string | 用户账号下家庭id |
| messageId | string | 随机生成 |

body:

{

"license": "",

"version":"v1", //v1表示老格式，v2表示新格式

"command":"upsert",//v1时有效

"type": 0, //0:devcontrol(设备控制); 1:scene(场景下发)

"endPoint":{

"did":"",

"pid":"",

"subdevdid":"",

"subdevpid":"",

"devicetypeflag":2, //0代表普通设备,1代表虚拟设备,2代表分组设备

"aeskey":"",//为空即可，从家庭中获取

"aesid":0,//为０即可，从家庭中获取

"extern":""//暂时不会使用

},

"payload":{

"type":0,//0定时任务，2周期任务

"index":3,　//如果有此参数，表示修改，不是新建，标示修改的序号

"enable":true,//任务开关

"timezone":"Asia/Chongqing",

"repeat":[1,2,3,4,5,6,7],//type为２关心，标示周期定时的执行时间，为空则在最近匹配时执行

"time":"yyyy-MM-dd hh:mm:ss",//格式：yyyy-MM-dd hh:mm:ss　或hh:mm:ss

"data":{"params":["pwr"],"vals":[[{"idx":1,"val":1}]]}

}

}

返回：

{

"status": 0,

"msg": "",

"payload":{

"timerlist":[

{

"index":0,

"timezone":"Asia/Chongqing",

"enable":true,//任务开关

"time":"yyyy-MM-dd hh:mm:ss"//格式：yyyy-MM-dd hh:mm:ss　或hh:mm:ss

}

],

"periodlist":[

{

"index":0,

"enable":true,//任务开关

"timezone":"Asia/Chongqing",

"repeat":[1,2],

"time":"yyyy-MM-dd hh:mm:ss"//格式：yyyy-MM-dd hh:mm:ss　或hh:mm:ss

}

]

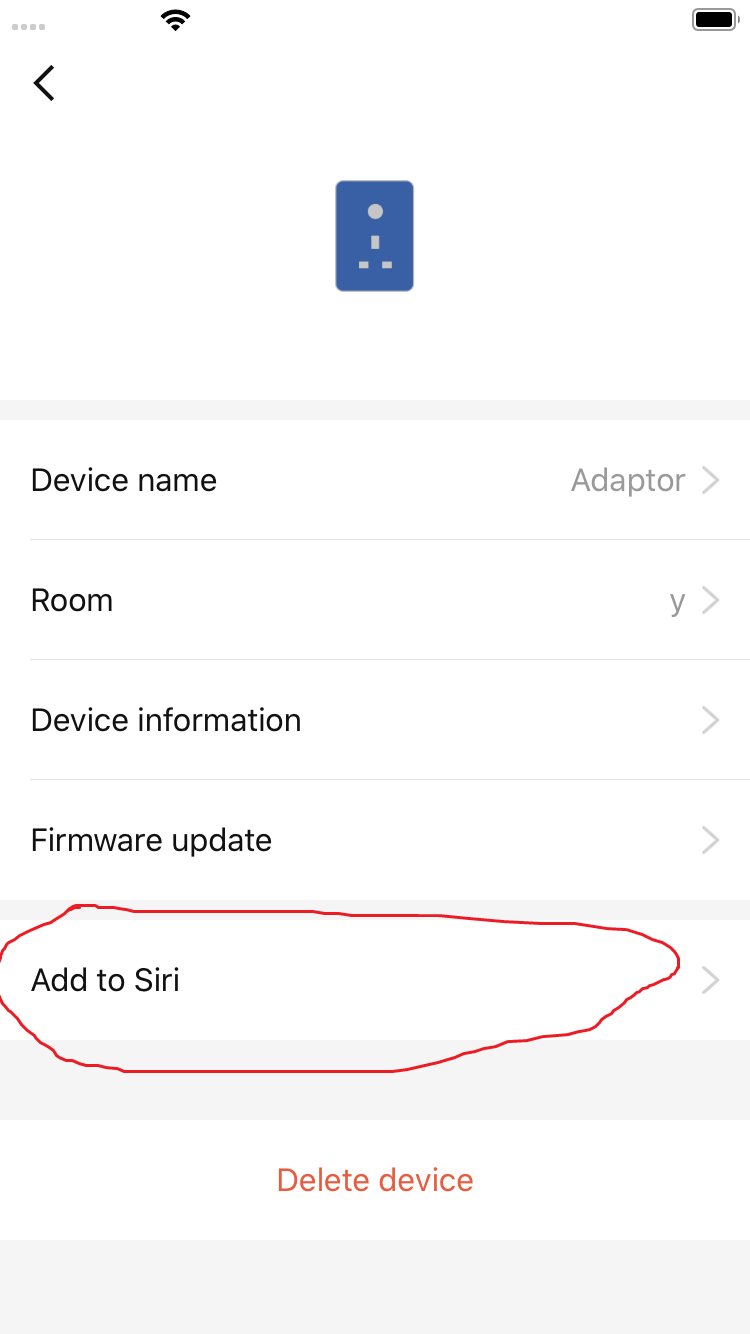
}

}

# H5实现

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 时间 | 说明 | 编辑人 |
| V1.0.0 | 2019/12/26 | 新建 | 谢晨华 |

## Siri入口

iOS 12之后可以通过Siri捷径功能快捷控制App中设备。用户可以设置设备中哪些功能可以被Siri捷径控制。Siri设置界面统一采用H5方式实现，app通过读取h5包zh-cn的voiceui.html文件来添加siri指令，在设备属性页点击“Add to Siri”来进入voiceui.html，  
 

Siri接口

app提供了‘增’、‘删’、‘查’接口提供给h5使用，若需要修改某条语音指令，可以下发‘增’接口指令并且携带需要修改的语音指令的voiceId（唯一）来实现。 需要注意的是，app提供“删”接口指令只能删除数据库里的语料，删除不了手机系统里的语料，所以需要app优化或者弃用；弃用后h5只能通过native弹出的删除按钮来删除，无法批量删除。

App新增加Cordova Plugin插件，名称：BLDeviceVoiceUIBridge

### 设置当前用户选择的指令(新增siri指令)

Action:

setVoiceCmd

请求数据：

{

"voiceId":"xxxxx",

"did": "aaaaaa",

"gatewayId": "bbbbbbbbbbbb",

"type":0,

"delay":0,

"cmds": [

{

"act":"set",

"params": [

"pwr1"

],

"vals": [

[

{

"idx": 1,

"val": 1

}

]

]

},

{

"params": [

"temp",

"mode"

],

"vals": [

[

{

"idx": 1,

"val": 0

}

],

[

{

"idx": 1,

"val": 0

}

]

]

}

]

}

参数说明：

|  |  |
| --- | --- |
| 参数名 | 说明 |
| voiceId | voiceUI中每条控制命令唯一标识（同一类型设备唯一即可）  如果是用户自定义的指令，并且voiceId值为空，需要Native自己生成 |
| type | 表示此指令是否是设备H5预定义  0：预定义  1：用户自定义添加 |
| delay | 单位秒  0表示Siri控制立即执行，其他表示多少秒执行（设置定时任务） |

### 获取当前用户已自定义的列表

Actoin:

getInitedVoices

请求参数：

无

返回数据：

{

"status":0,

"data":[

{

"voiceId":"xxxxx",

"did": "aaaaaa",

"gatewayId": "bbbbbbbbbbbb",

"shortcutText":"打开空调,

"type":0,

"delay":0,

"cmds": [

{

"act":"set",

"params": [

"temp",

"mode"

],

"vals": [

[

{

"idx": 1,

"val": 0

}

],

[

{

"idx": 1,

"val": 0

}

]

]

}

]

}

]

}

### ~~删除当前已设置的指令（只能删除数据库里的语料，删除不了手机系统里的语料）~~

已弃用，可在app的siri设置界面删除语料

~~Action:~~

~~deleteVoices~~

~~请求参数：~~

~~{~~

~~“voiceIds”:[“1”,”2”]~~

~~}~~

## **界面设计**

采用类似h5定时设置界面或h5自定义场景界面模板，分为2个界面：  
·siri指令列表页面（主页）

·siri指令添加页面

### siri指令列表页面（主页）

初始界面为已设置好的siri指令列表页面，内容为‘siri列表’以及‘添加按钮’。

点击某条设置好的语料即跳到app的siri设置界面，在app的siri设置界面可以修改该条siri指令的语料，也可删除该语料。

### siri指令添加页面

点击指令列表页面的‘添加按钮’跳到siri指令添加页面，该界面与h5设置定时指令页面应保持一致。

siri指令添加页面 内容分为“命令选择”以及“保存按钮” ；

“命令选择”可以随意选择想要的命令，选择命令后点击“保存按钮”后、调用“增”接口且携带参数跳到app的siri设置界面，在app的siri设置界面输入**语料**（可任意文字）后才可成功保存

注：siri指令添加页面的“命令选择”内容内，需携带‘时间选择框’来设置延时执行