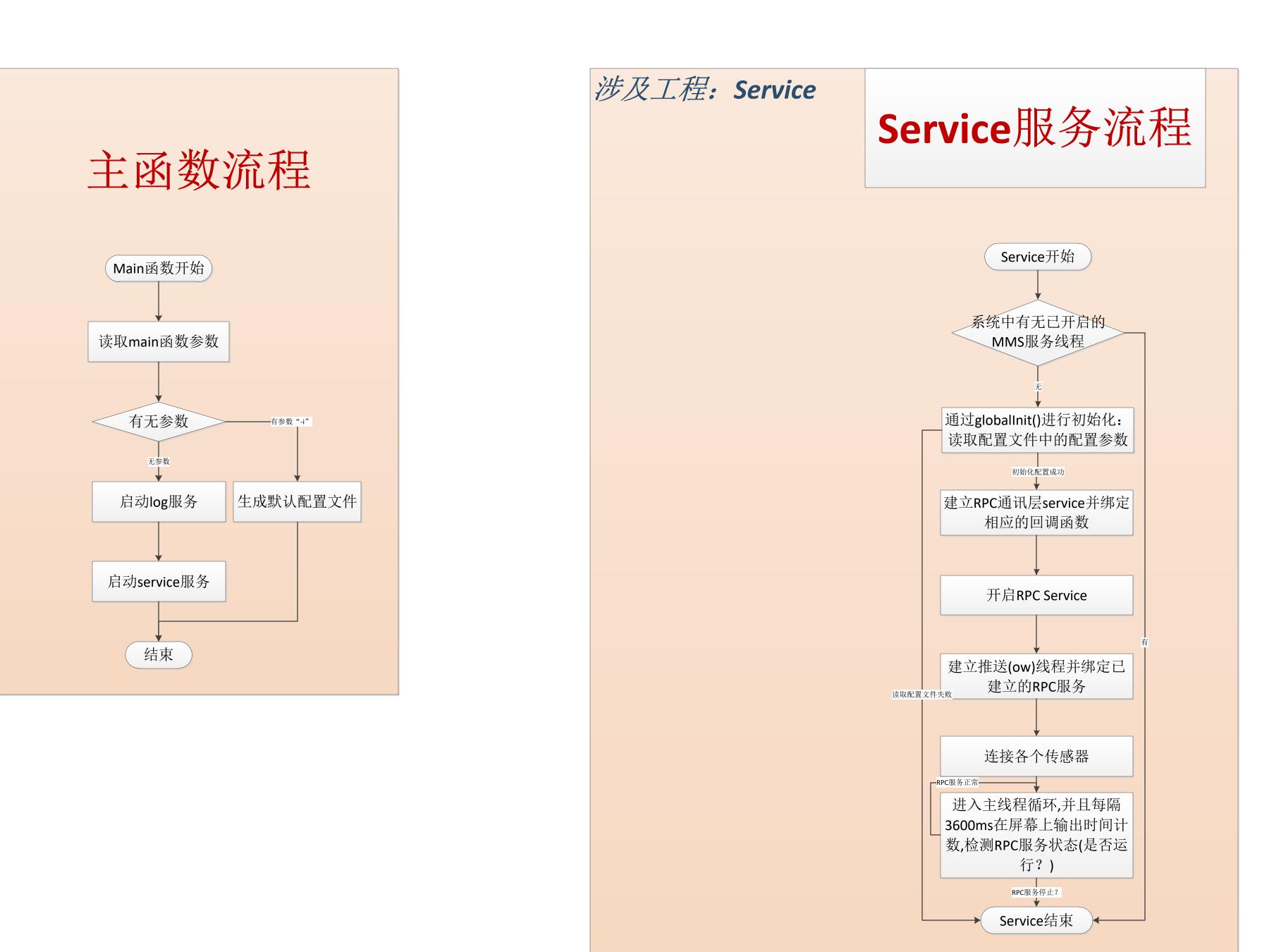
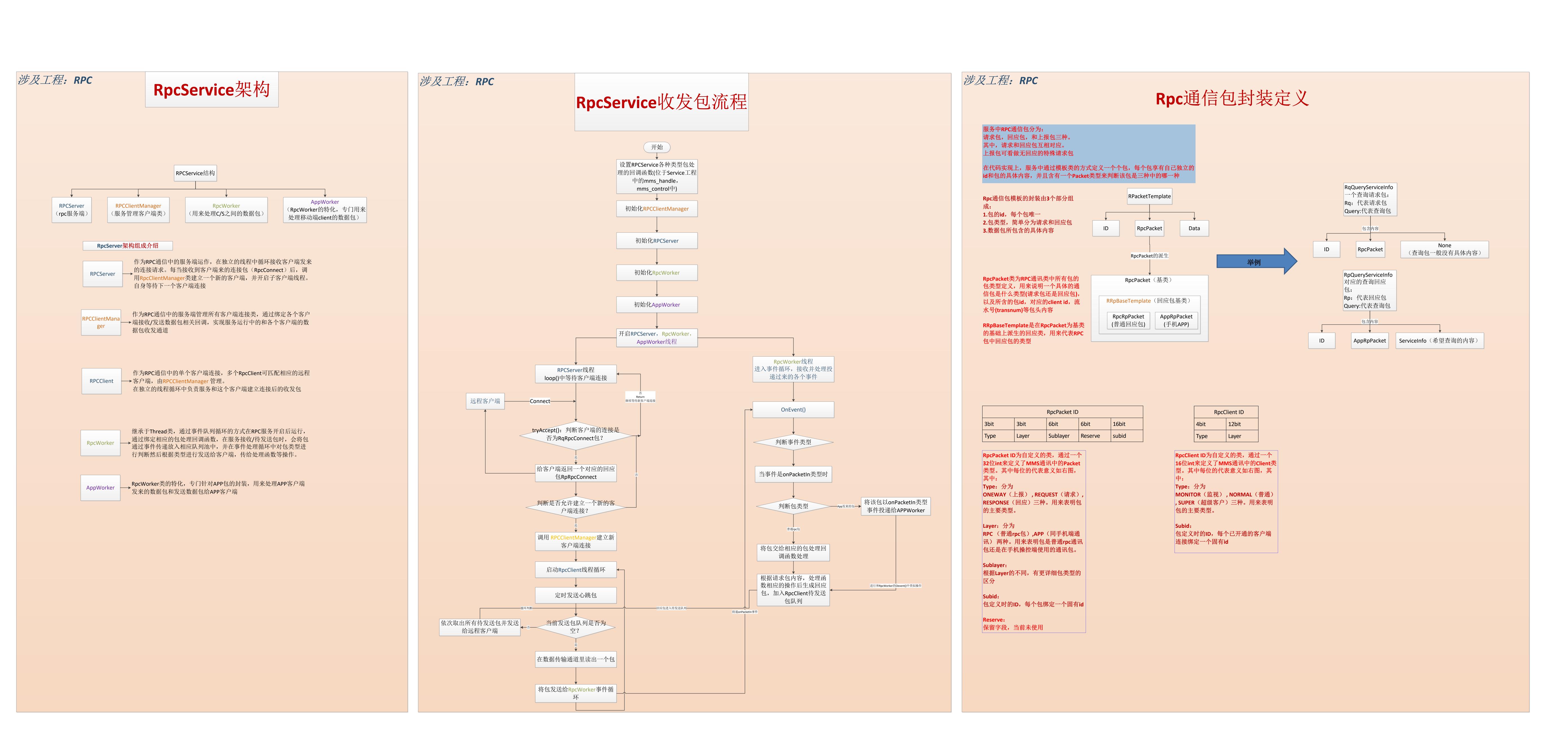
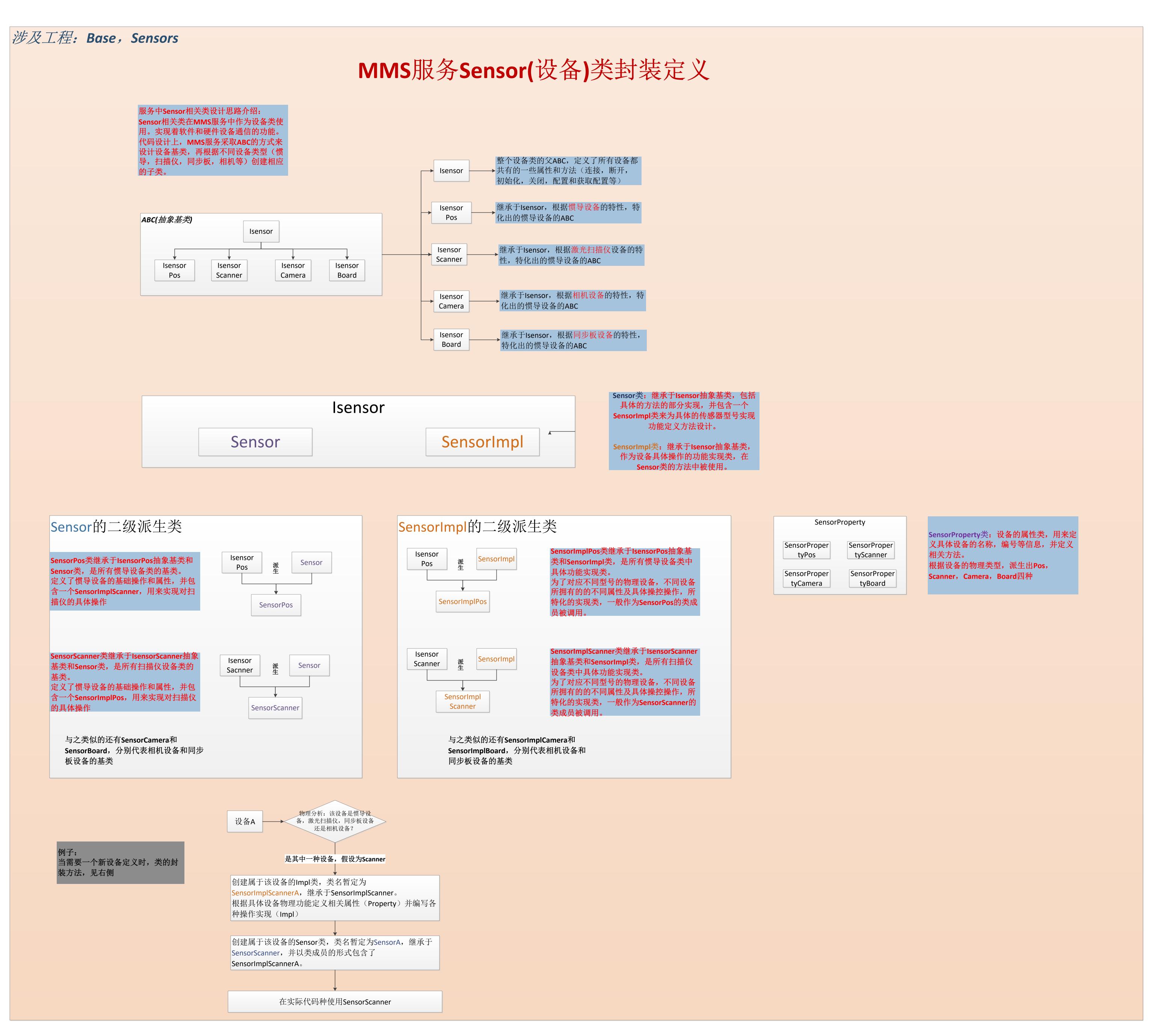
内容介绍。 该文件为MMS服务代码和工程的介绍文件。 由于当前MMS服务工程较多,设计中采用多层嵌套封装,并且多模块共同运作。初识服务可能会携不清方向,难于寻找入手点。因此编写该介绍的意向为对初次接触 服务代码的同事提供一个代码架构梳组,以达到快速熟悉服务代码架构,理解相关类设计以及知晓服务中各个工作流程的效果。 主要内容包括。 1.服务架构梳理。介绍服务中所有工程主要为容及用途,各个工程之间的联系等 2.重要类设计,封装介绍。对代码中重要的类封装,设计模式进行介绍 3.服务运行中的重要/生要流程的走向介绍(以流程图的方式),并对涉及到其中的重要类,方法进行有色标识和介。绍 注意: 本文件中内容,是以辅助熟悉代码内容为目的编写。当涉及到具体的相关代码、方法实现时,还是需要结合工程中具体代码及代码相关注释才能有深入了解。 因此希望新入手间事准而对大量代码手足无措时可在该文件内容下较快地理解服务代码中的设计。 作者:唐嘉豪

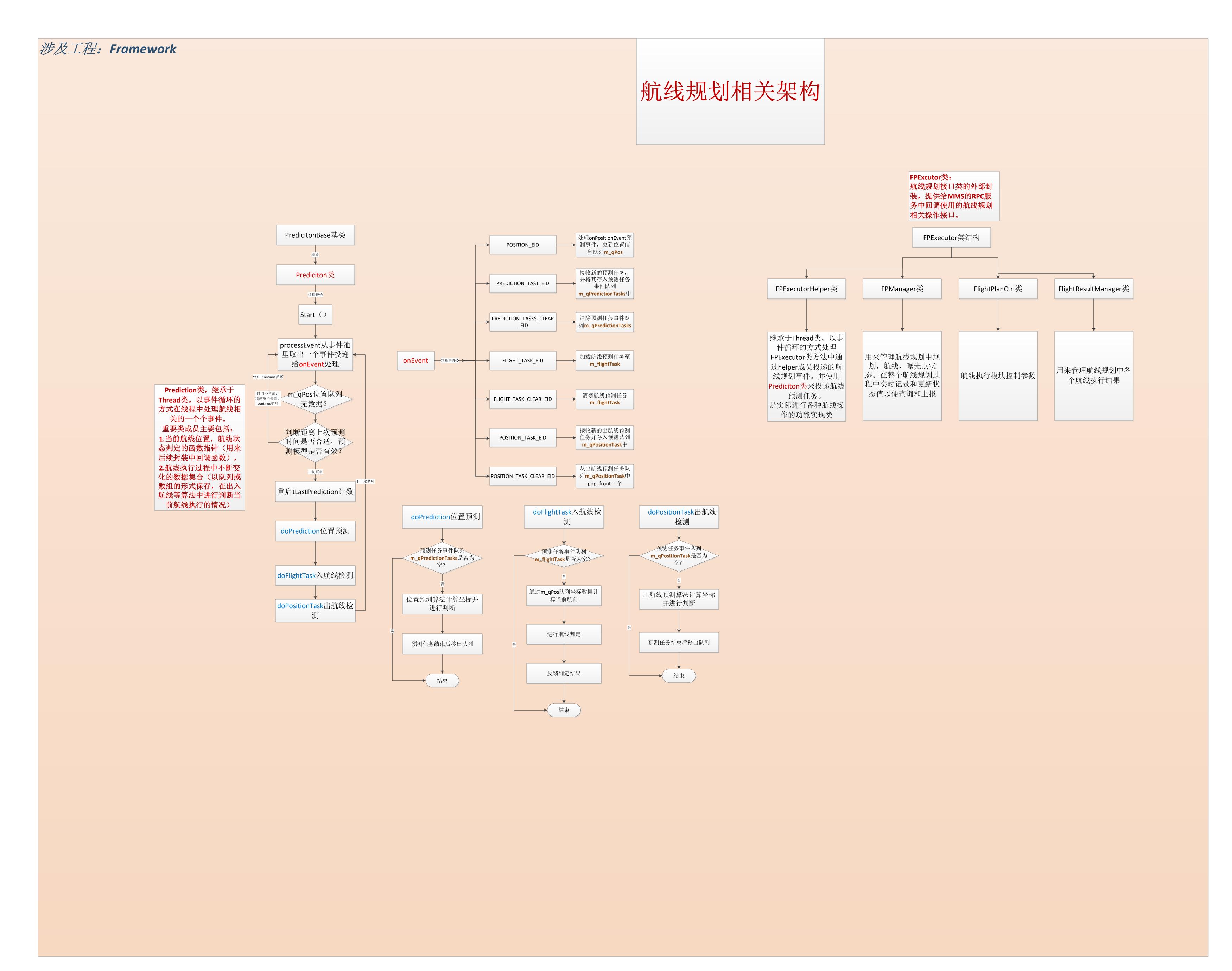




\_\_\_\_

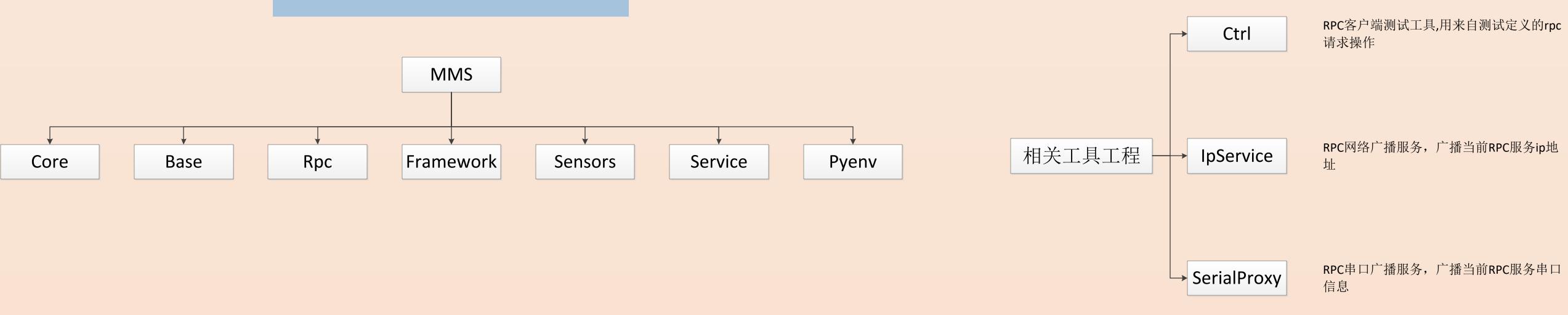






## MMS代码总架构

#### MMS服务中包含多个工程,每个 工程中内容及各个工程关系如下



Core

为整个系统的底层代码,主要包括:

1. 数据类型、数据基类定义及相关的数据转换(主要写于"Utils"和"types.h"中的代码中)

2. 日志系统代码(主要位于"debug"中的"logger.h")

3. 错误类型,异常定义(主要位于"debug"中的 "exception.h"和"str\_error.h")

4. 网络底层通信代码,串口通信的实现代码(主要位于"net"和"serial"中)

5. 线程底层代码(主要位于"thread"中)

6. 计时器,定时器代码实现(主要位于"time"中)

在Core的底层代码基础上,对整个操控服务涉及到的对采集 工程的定义,数据基类序列化,相关设备操控等需求建立的 各种基类代码的实现

Base

主要包括:

1. 针对设备操控的需求,各种设备基类的定义(主要位于 "sensor"和包含 "sensor"字段的代码页中)

2. 对各种数据类型,包括基本数据类型和操控服务中自定义的数据类型,实现各个数据类型之间序列化-----相互转换和运算符运算代码的重载(主要位于"serialize"中)

3. 针对工程采集中的一些计算需求和对采集工程的结构 定义,实现采集工程相关代码(主要位于"env"中)

4. 采集相关数据类型和操控服务返回码,状态码的定义 (主要位于"types.h","pos\_realtime.h"中)

RPC通讯层的实现代码

1. RPC通讯包的封装和定义(主要位于"packet"代码页中)

Rpc

2. 对各种数据类型,包括基本数据类型和操控服务中自定义的数据类型,作为包内容时的读包写包方式定义(主要位于"serialize"中)

3. 读写包的实际实现,包括读/写包中具体的操作,包 头包尾的判断和校验等(主要位于"io"中)

4.RPC通讯的服务和客户端,远程代理及相关功能类的实现(主要位于"service"和"proxy"中)

Framework

设备操控类的框架代码以及航线规划相关代码

主要包括:

1. 具体型号的设备类框架(主要位于"sensor") 2. 航线规划相关代码(主要位于包含"flight"字段的文 1) Sensor

各种型号设备的所有操控代码的具体实现,设备实际属性的定义以及服务运行期间实时数据处理操作的相关实现代码

主要包括:

1.设备功能实现类的代码(主要位于 "board(同步板)", "ilsp(扫描仪型号)", "riegl(扫描仪型号)", "velodyne(扫描仪型号)", "trimble(惯导型号)", "novatel(惯导型号)"中)

2. 实时数据处理,实时点云相关代码实现(主要位于 futility"中)

Service

用于RPC远程调用的回调函数定义,服务的状态机相关定义以及加载/保存服务配置类的实现

主要包括:

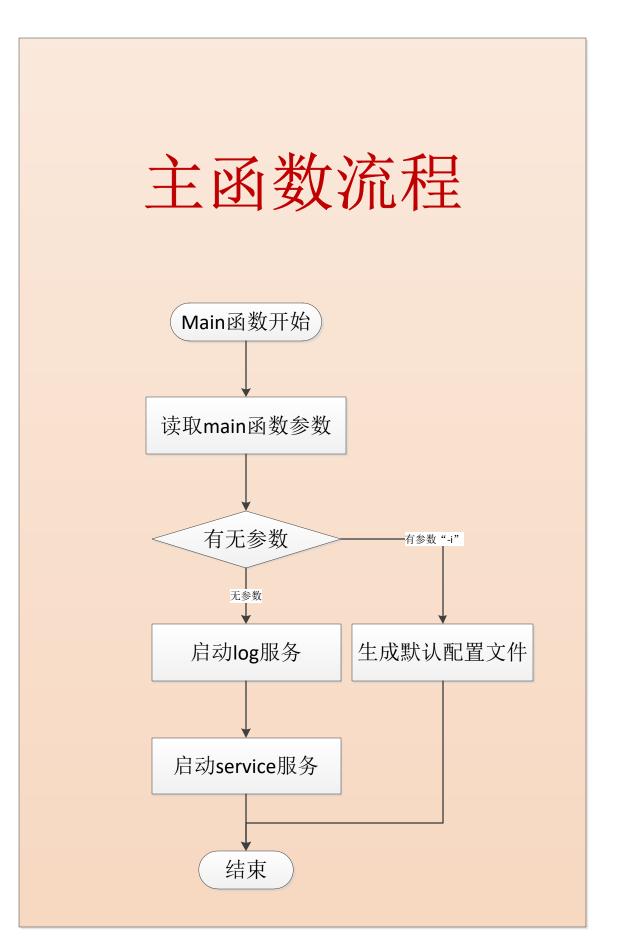
1.RPC回调定义(主要位于"mms-control","mms-handle","mms-oneway"中)

2. 服务状态机定义(主要位于"mms\_state\_machine"

· 3.服务配置类(位于"mms-settings"中) Pyenv

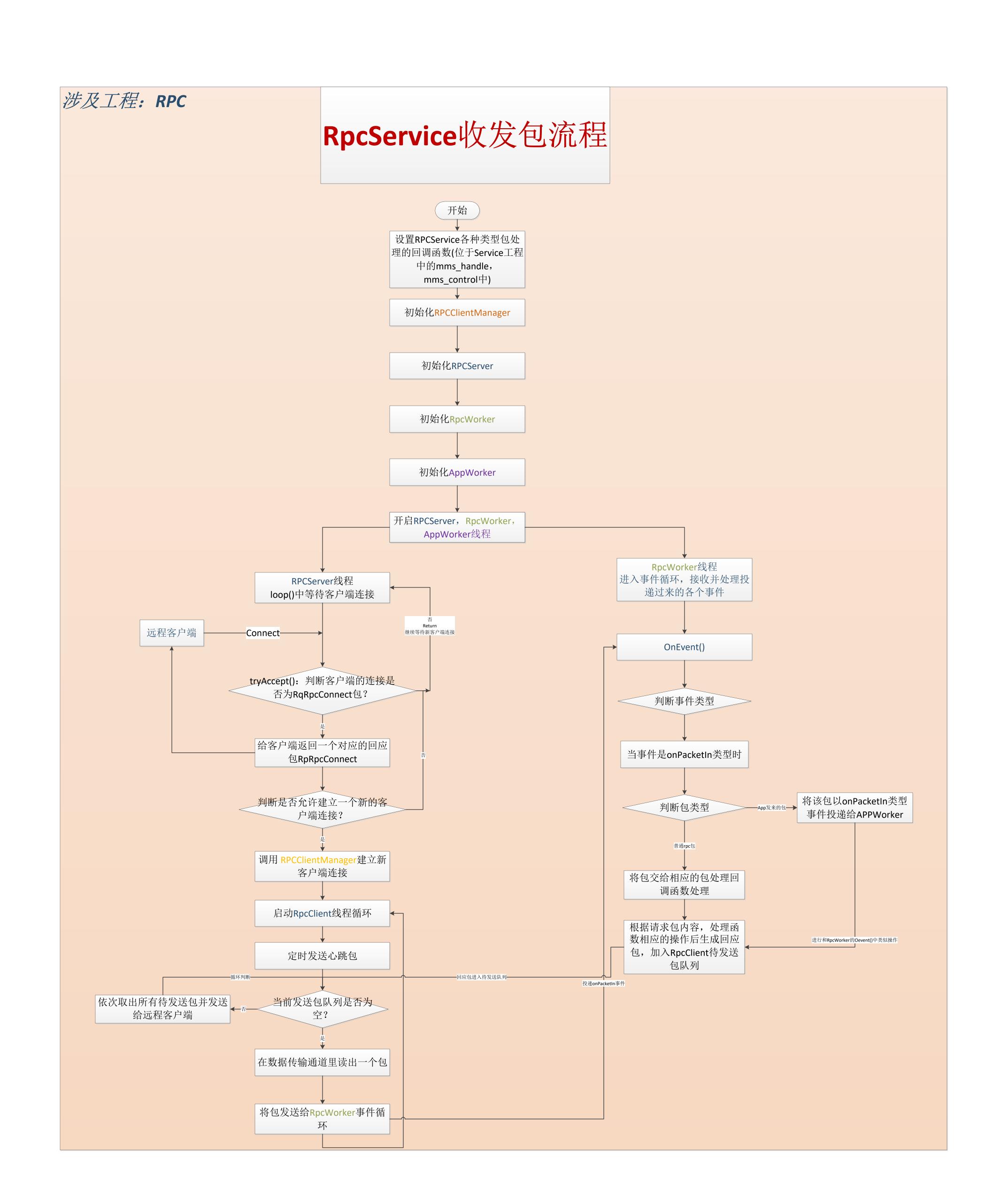
主要内容

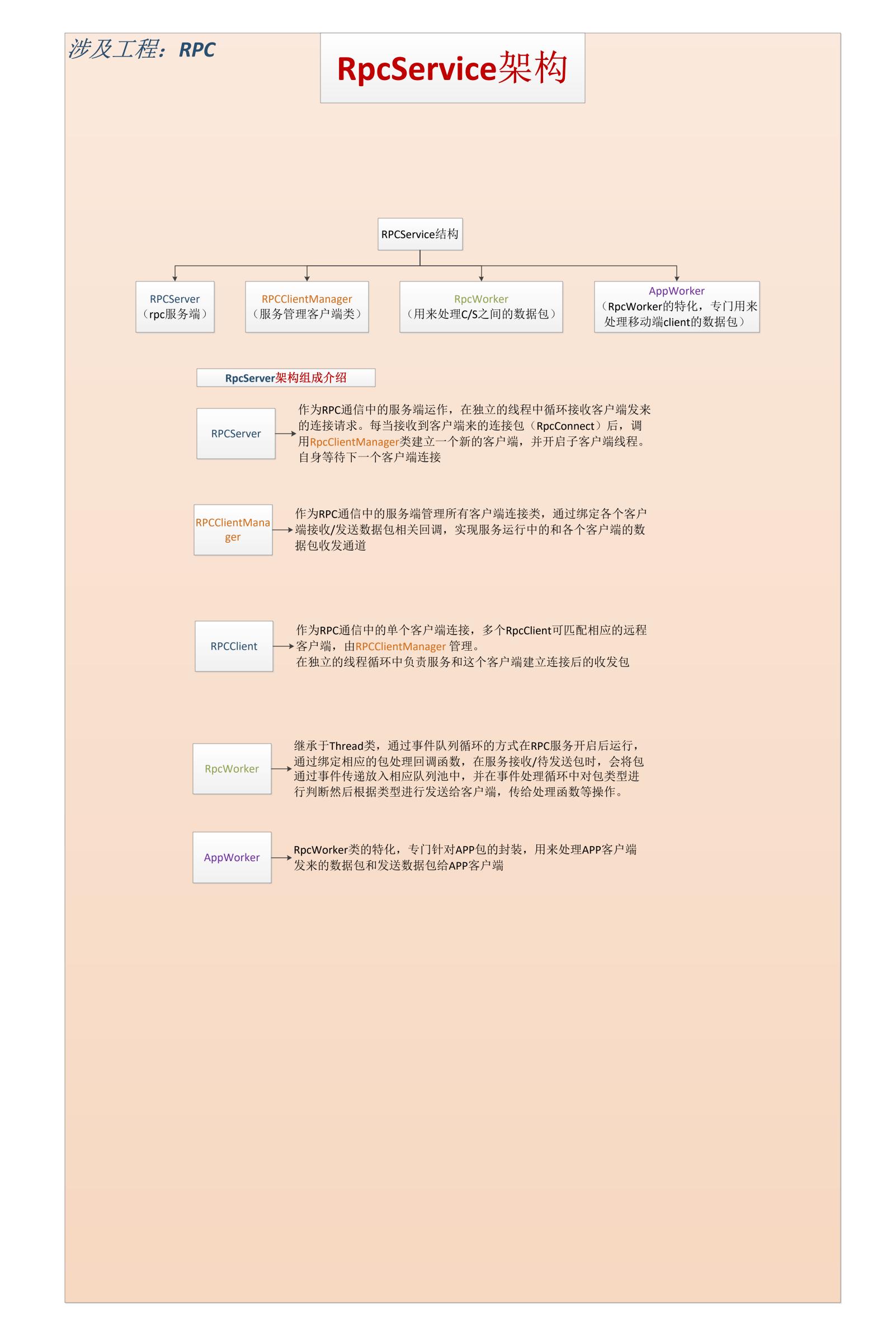
涉及到代码和Python相关兼容,当前未使用

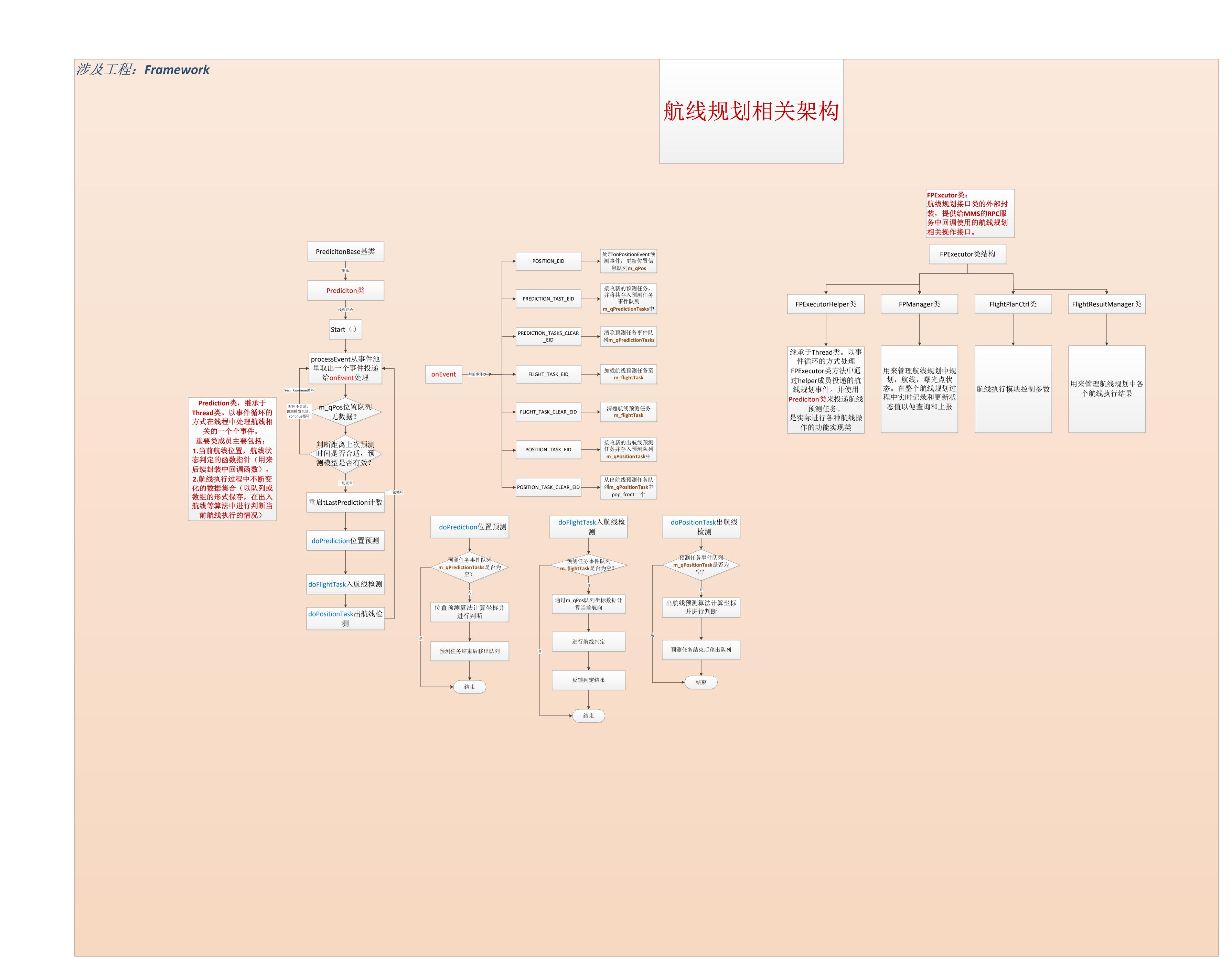


\_\_\_\_









#### 涉及工程: RPC

### RpcPacket通信包

RqQueryServiceInfo

一个查询请求包:

Rq: 代表请求包

Query:代表查询包

RpcPacket

RpQueryServiceInfo

对应的查询回应

Rp: 代表回应包

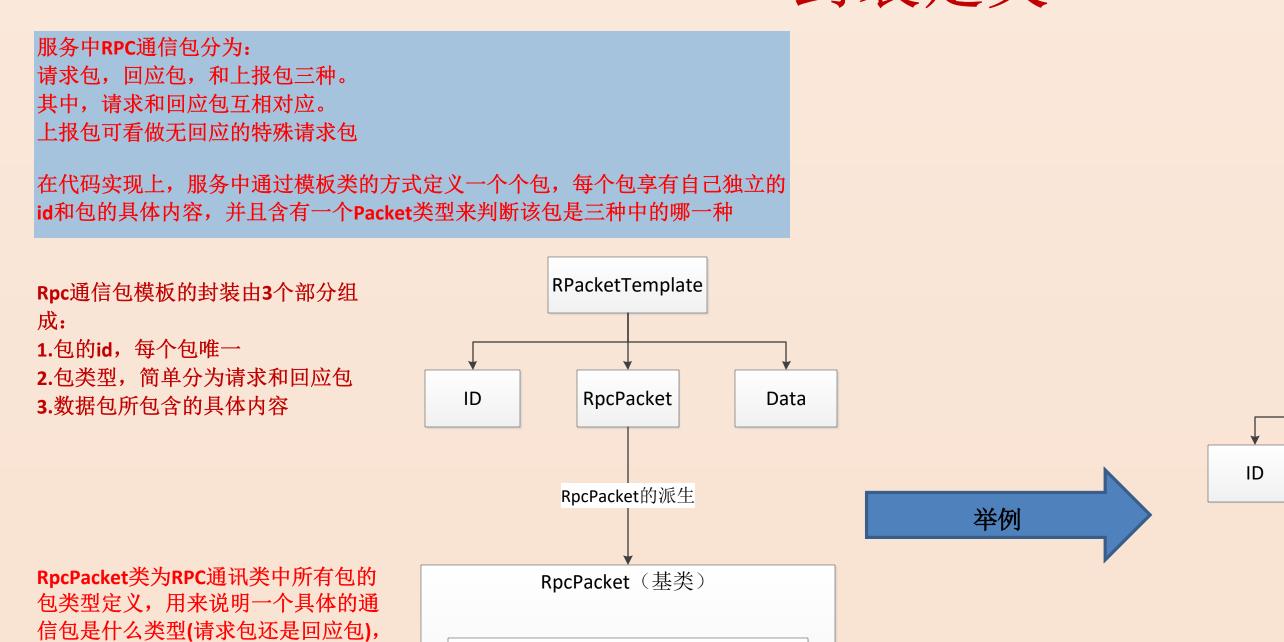
Query:代表查询包

ID

(查询包一般没有具体内容)

ServiceInfo(希望查询的内容)

## 封装定义



RRpBaseTemplate (回应包基类)

(普通回应包)

AppRpPacket

(手机APP)

RpcPacket ID					
3bit	3bit	6bit	6bit	16bit	
Туре	Layer	Sublayer	Reserve	subid	

RpcPacket ID为自定义的类,通过一个32位int来定义了MMS通讯中的Packet 类型。其中每位的代表意义如右图,其中:
Type: 分为
ONEWAY(上报),REQUEST(请求),RESPONSE(回应)三种。用来表明包的主要类型。

以及所含的包id,对应的client id,流

RRpBaseTemplate是在RpcPacket为基类

的基础上派生的回应类,用来代表RPC

水号(transnum)等包头内容

包中回应包的类型

Layer:分为 RPC(普通rpc包),APP(同手机端通讯)两种。用来表明包是普通rpc通讯包还是在手机操控端使用的通讯包。

Sublayer: 根据Layer的不同,有更详细包类型的 区分

Subid: 包定义时的ID,每个包绑定一个固有id Reserve:

保留字段, 当前未使用

RpcClient ID

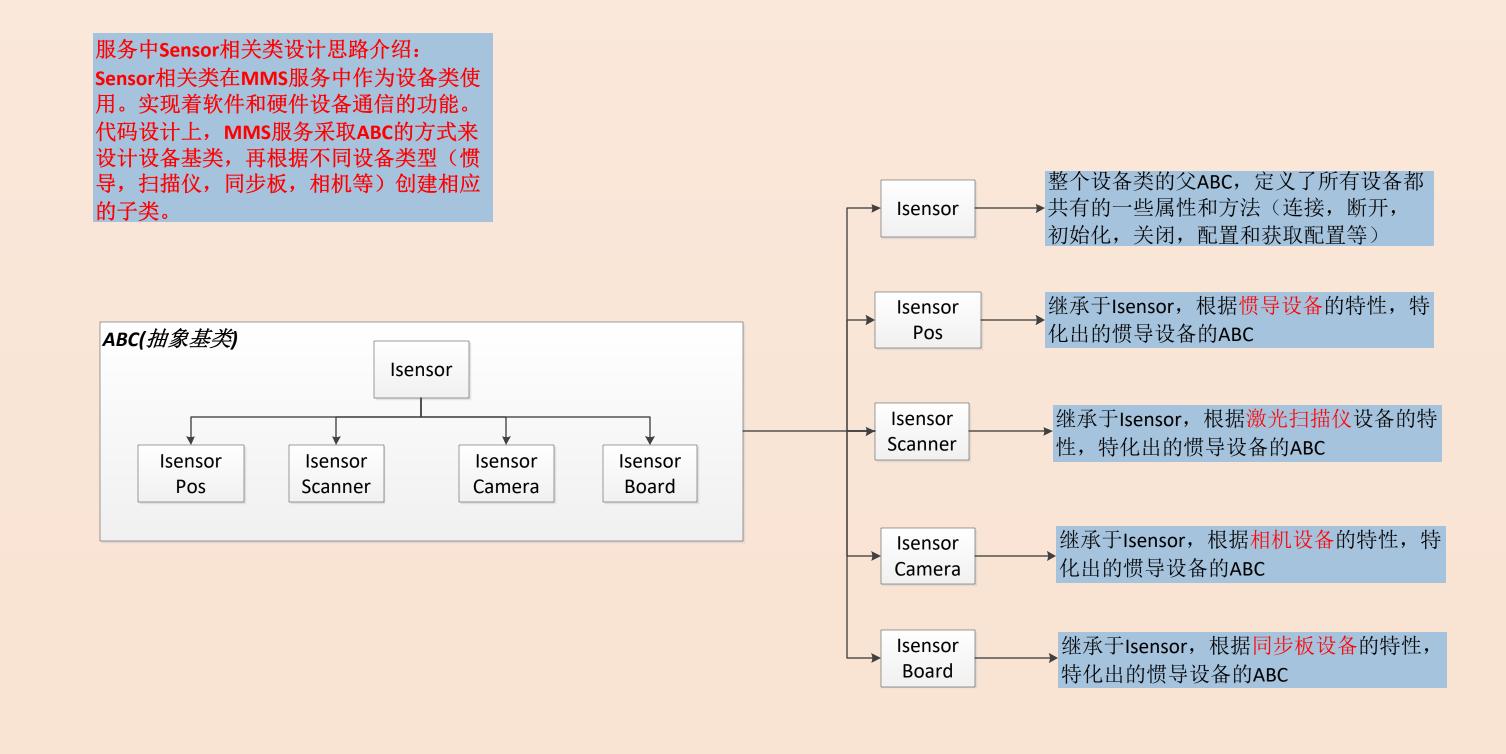
4bit 12bit

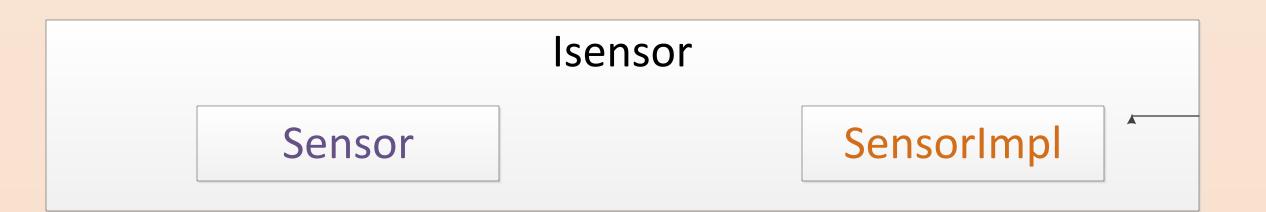
Type Subid

RpcClient ID为自定义的类,通过一个 16位int来定义了MMS通讯中的Client类型。其中每位的代表意义如右图,其中:
Type: 分为 MONITOR(监视), NORMAL(普通), SUPER(超级客户)三种。用来表明包的主要类型。

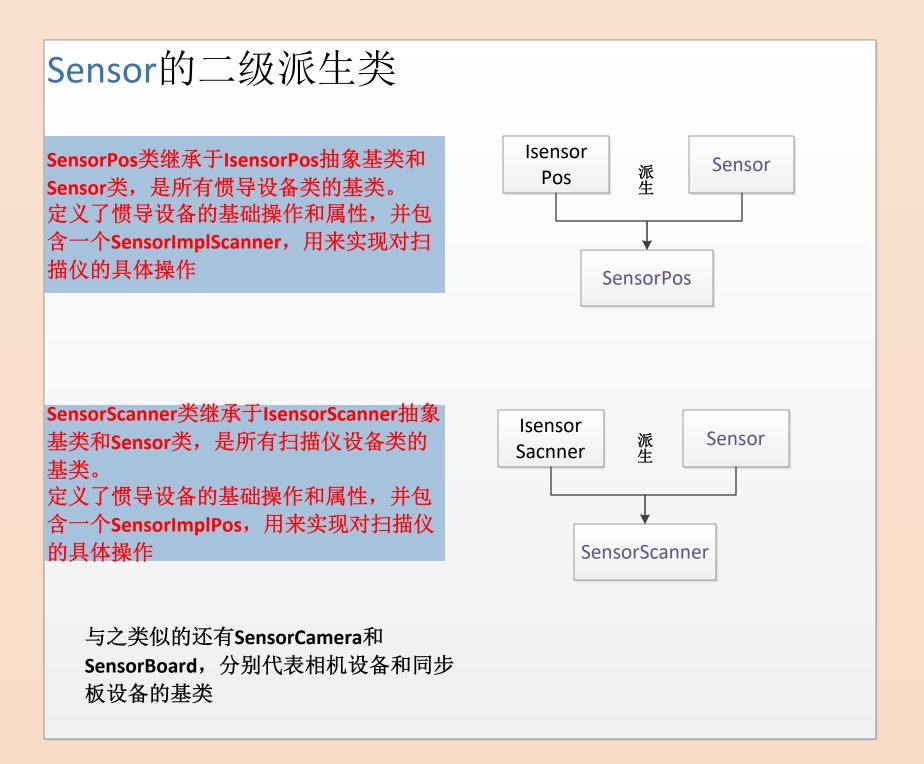
Subid: 包定义时的ID,每个已开通的客户端 连接绑定一个固有id 涉及工程: Base, Sensors

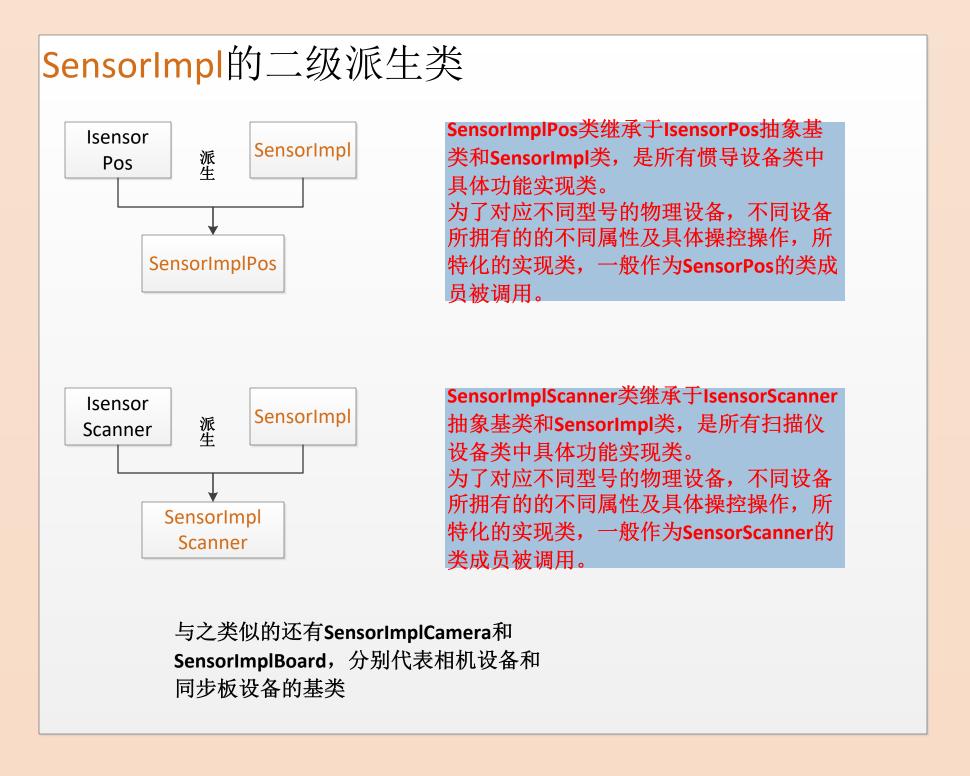
### MMS服务Sensor(设备)类封装定义

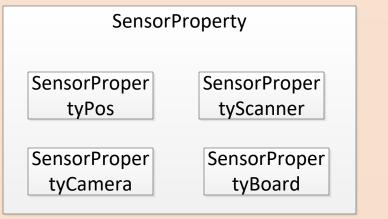




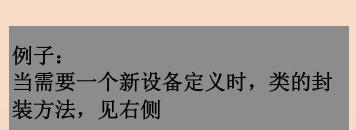
Sensor类:继承于Isensor抽象基类,包括 具体的方法的部分实现,并包含一个 SensorImpl类来为具体的传感器型号实现 功能定义方法设计。 SensorImpl类:继承于Isensor抽象基类, 作为设备具体操作的功能实现类,在 Sensor类的方法中被使用。

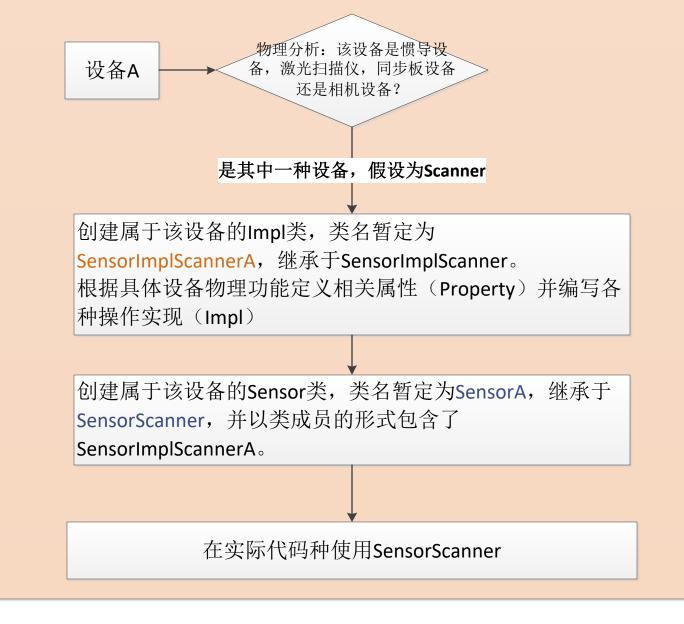


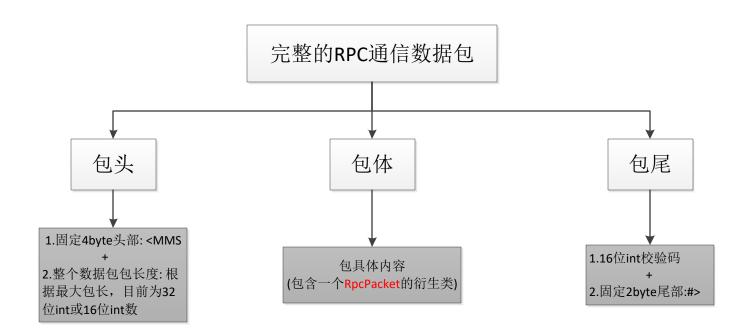




SensorProperty类:设备的属性类,用来定义具体设备的名称,编号等信息,并定义相关方法。根据设备的物理类型,派生出Pos,Scanner,Camera,Board四种







# 文件内容介绍