



```
int radio::callStateChangedInd(int slotId, int indicationType,
    int token, RIL_Errno e, void *response, size_t responseLen) {
    if (radioService[slotId] != NULL && radioService[slotId]->mRadioIndication !=
    NULL) {
        Return<void> retStatus = radioService[slotId]->mRadioIndication->
        callStateChanged(
            convertIntToRadioIndicationType(indicationType));
        radioService[slotId]->checkReturnStatus(retStatus);
    } else {
        RLOGE("callStateChangedInd: radioService[%d]->mRadioIndication == NULL", slotId);
    }
    return 0;
}
```

上面的代码逻辑主要是发起 radioService[slotId] → mRadioIndication → callStateChanged 调用，将调用 com.android.phone 进程中 IRadioIndication 服务的 callStateChanged 接口。

到此，已完成 RIL 中两种消息运行处理机制的解析，Solicited 和 UnSolicited 消息处理和响应机制总结如图 10-5 所示。

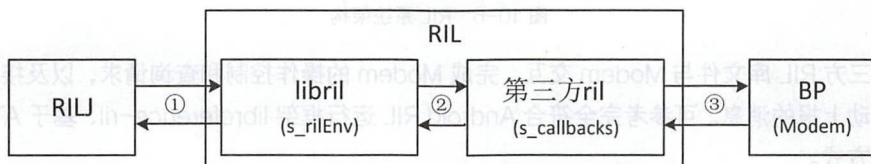


图 10-5 Solicited&UnSolicited 消息处理机制

图 10-5 所示的 Solicited&UnSolicited 消息处理机制，我们关注图中用数字标识的三种交互方式。

① phone 进程与 rild 进程交互，采用 HIDL 接口服务调用。com.android.phone 进程中提供 IRadioResponse 服务和 IRadioIndication 服务，rild 进程提供 IRadio 服务。

② RIL\_register 调用建立了 libril 与等三方 ril 互相持有对方的函数指针；在 rild 进程中，使用函数指针发起进程内的函数调用。

③ 在第三方 ril 动态链接库中，实现了将 RIL 请求转换为与 BP 的交互机制，不同厂家实现的机制不同；Nexus 6P 手机基于高通平台，使用了 QMI ( Qualcomm Message Interface ) 来完成 AP 与 BP 的通信。

## 本章小结

总结 RIL 的系统架构，如图 10-6 所示。

图 10-6 所示的 RIL 系统架构，我们需要掌握以下几点：

- RIL 主要分 RILJ 和 RILC 两部分。RILJ 运行在 com.android.phone 进程空间的 Telephony Framework 框架层，RILC 运行在 User Libraries 系统运行库层中的 HAL 子层。
- rild、libril 和第三方 RIL 实现都运行在 rild 进程中，通过 rild.rc 配置文件由 Linux init 进程进行加载和管理。
- RILJ 与 RILC 通过 IRadio、IRadioIndication 和 IRadioResponse 服务接口调用，完成 Solicited 和 UnSolicited 消息交互。



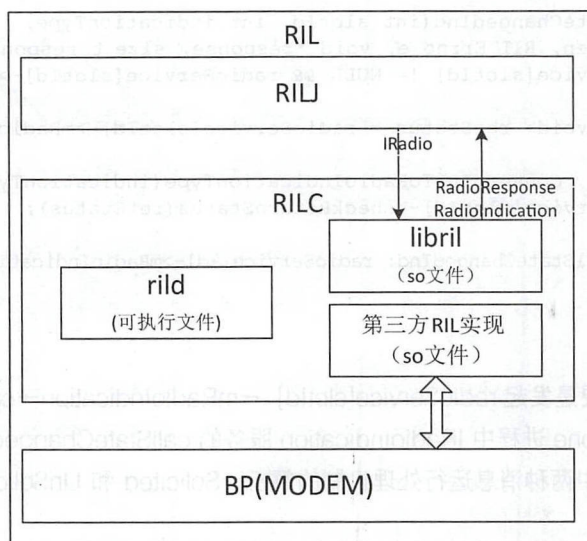


图 10-6 RIL 系统架构

- 第三方 RIL 库文件与 Modem 交互，完成 Modem 的操作控制和查询请求，以及接收 Modem 主动上报的消息。可参考完全符合 Android RIL 运行框架 libreference-ril、基于 AT 命令的实现方式。
- 掌握 Solicited 和 UnSolicited 消息在 RILJ 和 RILC 中的处理机制。

在 RILConstants.java 和 ril.h 中定义的 RILSolicited 和 UnSolicited 消息是一致的，分别是 RIL\_REQUEST\_XXX 和 RIL\_UNSOL\_XXX。

RILJ 对象 mRequestList 列表中的 RILRequest 对象与 libril 中 s\_pendingRequests 链表中的 RequestInfo 具有相同的处理逻辑，都是完成 Solicited Request 和 Solicited Response 消息处理。

UnSolicited 消息的处理机制相对简单，主要是通过服务调用完成消息主动上报。

## 本章小结



# Android Telephony

## 原理解析与开发指南



欣闻好友青平兄又出版了一本关于 Android Telephony 的著作。此书的前一版曾强烈吸引了高通等通信领域巨头们的关注，影响很大。这本新版的 Telephony 著作与时俱进，紧跟 Android 系统更新，相信它会超越前一版，赢得更多读者的喜爱。

青平的新书有三点特别值得一读：①内容新。全书基于 Android 8.1，紧跟 Google 官方版本发布的快节奏。②能实践。指导搭建开发环境、编译代码、烧写镜像到手机，让入门不再停留在表面。③挖掘深。多个手机大厂的一线工作经验，内容深入浅出让有经验的从业者亦能精进。

——**谢边 / 成都西纬科技总经理**

本书围绕 Android 最核心模块之一 Telephony 由简入深、以点带面，涉及从 Android 开发入门到深度剖析 Telephony 内部具体实现，同时清晰讲解相关模块间交互以及 Android 8.1 版本的新变化。内容纲目并举、娓娓道来，将原理与实践透彻展现，令我这个老司机读后为之虎躯一震、如沐春风。青平的又一大作，对初学者和精通者都是值得拜读的作品。

——**黄江涛 / 小米科技 Modem 专家**

青平有多年的 Android 系统研发经历，尤其对 Telephony 有丰富的实战经验。本书基于 Android 8.1 以通俗易懂的语言将 Telephony Framework 从应用层到 RIL 层的设计与实现做了详实的讲解和总结；以 Voice Call、ServiceState、Data Call 业务为主线，将 Telephony 各层有机串联起来。本书可以帮助 Android 系统初学者快速入门，对有经验的从业者来说，也是一本优秀的参考书、工具书。

——**谢亮 / Qualcomm 高通中国 IMS 客户支持工程师**

教材服务热线：010-81055256  
反馈/投稿/推荐信箱：315@ptpress.com.cn  
人邮教育服务与资源下载社区：www.rjiaoyu.com

人邮教育  
www.rjiaoyu.com

本书配套资源  
扫描二维码免费下载



封面设计：董志桢

定价：59.80 元

ISBN 978-7-115-48915-9

