

华为架构师解读：HarmonyOS 低时延高可靠消息传输原理

本文作者：zhangkesi，华为软件架构设计工程师

这是一篇 HarmonyOS 低时延高可靠消息传输原理的介绍，希望对你有所帮助。

01 一个近场通信的例子

1.1 全场景智慧生活的典型问题

在全场景智慧生活当中，设备种类和数量越来越多，各种富设备（如智慧屏、PC、PAD、音箱）以及各种瘦设备（如 IOT 的智能门锁、摄像头、智能灯、智能窗帘）的近场通信方式各不相同，有 wifi、蓝牙、NFC、usb、zigbee 等。

在这么多种近场通信方式选择上，如何让这些设备便捷、高效地通信，从而实现上层应用无需考虑设备差异，就如同使用“一个设备”一样，流畅地使用多个设备的能力，是全场景智慧生活中面临的一个典型问题。HarmonyOS 分布式软总线为这个问题提供了可靠的解决方案，并通过简单的 API 接口向开发者开放出来。

1.2 如何保障控制消息(Message)低时延高可靠

下图是一个家庭场景中典型的富瘦设备的组网图，主要包含两类业务，黑色线条的上网业务，红色线条的近场业务。横向的近场通信业务的物理通道，比纵向的上网业务的物理通道种类更多，带宽也不同，HarmonyOS 分布式软总线完全屏蔽了底层通信的差异，让上层应用

通过使用几个简单的软总线接口，就像使用本地接口一样，轻松实现多设备间高速通信。

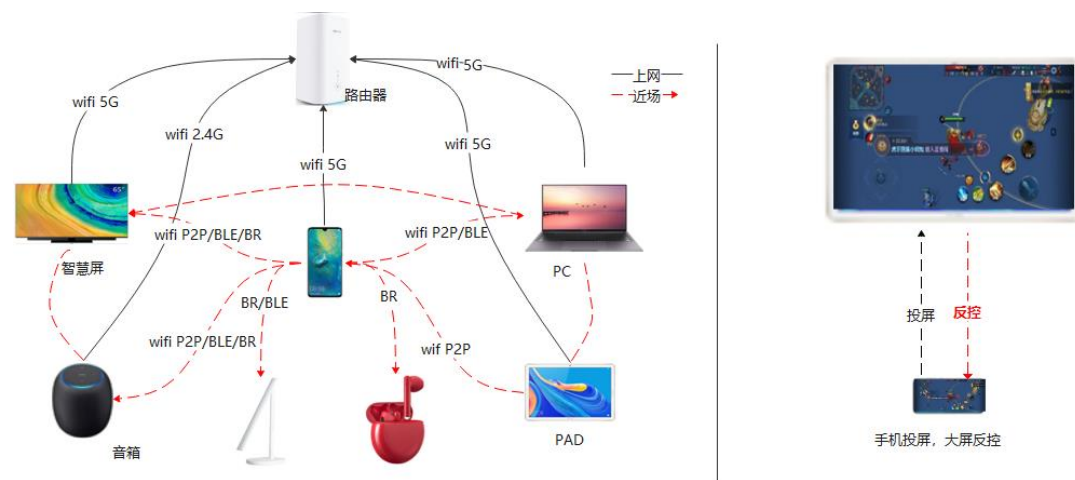


图 1

举个例子，将手机上的游戏 App 的操作界面投屏到 PAD 上，如何实现在 PAD 上进行手机上游戏 APP 的控制如在手机上控制一样的流畅？其中，使用软总线的 SendMessage 接口完成 PAD 到手机的反控操作（华为 Cast+技术）Message 的无延迟传输，起到了一个关键的作用。具体实现如下：

前提条件：

- 1、 手机、PAD 均搭载了 HarmonyOS，具备分布式软总线能力
- 2、 手机已经把游戏 APP 的操作界面投屏到 PAD 上

过程描述：

- 1、 手机首先使用软总线的发现能力发现 PAD 设备，并把手机上游戏 APP 的操作界面投屏到 PAD。
- 2、 因为游戏 APP 本身在手机上，所以在 PAD 上操作手机游戏 APP，就是从 PAD 到手机

的“反控操作”，即 PAD 上控制消息 Message 反馈到手机上执行，PAD 和手机之间需要通过软总线建立控制通道。软总线要选择最优传输通道，并保障该通道上的数据得到高优先级的传输。

3、 PAD 调用 SendMessage 接口把控制消息 Message 反馈给手机。

4、 手机收到 PAD 的反控消息并执行，并把执行后的结果再反馈到 PAD 上。整个过程的时延要求在百毫秒级。

上面描述的过程看似简单，实际上底层通信使用到了 HarmonyOS 分布式软总线的发现、连接和传输的能力。本次不讲发现和连接的技术点，仅对传输的实现原理进行解释。

02 近场 Message/Byte 传输实现原理

2.1 实现过程描述

HarmonyOS 分布式软总线提供了两个接口，分别用于近场通信场景下长短消息的传输，分别是 SendMessage 和 SendByte，实现原理相同，如下图所示：

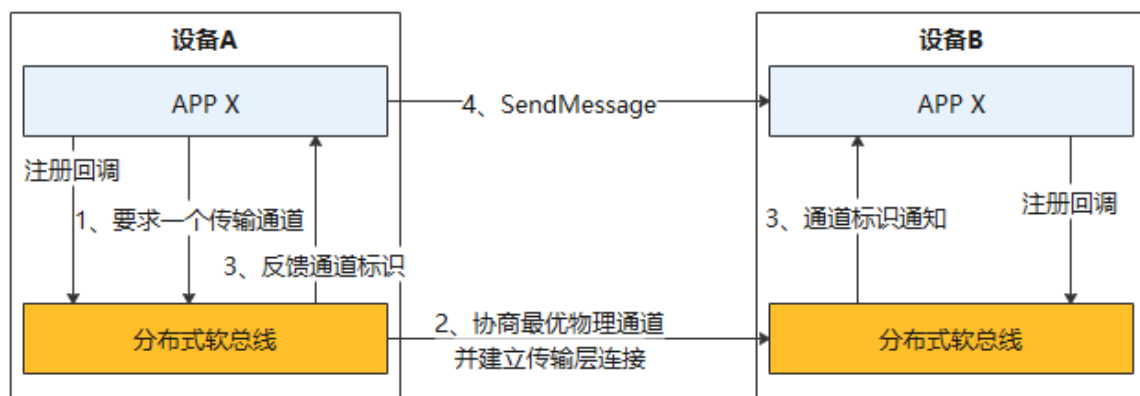


图 2

图中 APP X 统一代表不同的上层应用 App。具体过程描述：

- 1) 设备 A 和设备 B 的 APP X 会在初始化阶段向软总线注册回调通知接口，用于在传输通道打开、数据接收后通知到 APP X
- 2) 设备 A 的 APP X 要向设备 B 上的 APP X 发送消息，设备 A 的 APP X 首先把设备 B 的设备 ID 信息、以及标识 APP X 的信息传递给软总线，请求一个传输通道。
- 3) 软总线要根据当前两个设备已有的物理通道种类（BR/BLE/WIFI2.4/Wifi 5G/P2P），以及物理通道的负载和设备的状态，决策选择一个最优的传输通道的底层连接，同时完成传输层的连接建立，和传输标识的内核态到用户态的映射，最后把传输通道标识传递到两个设备的上层 APP X。
- 4) 设备 A 的 APP X 拿到通道标识后再调用 SendMessage/SendByte 接口和设备 B 的 APP X 进行通信。设备 B 的 APP X 也可以使用相同的方法和设备 A 进行通信。
- 5) 传输结束后，设备 A 的 APP X 可以调用关闭传输接口完成传输通道资源的释放。

2.2 Message/Byte 传输注意事项

- 1) Message 类型主要用于低时延、高可靠业务，比如游戏的控制命令、IoT 设备的开关（灯的开关、门窗的开关）等等，数据量最大不超过 4KB。
- 2) SendMessage 对 Message 类型消息的传输，HarmonyOS 软总线在底层实现按照最高优先级进行传输，例如空口使用最高优先级 VO 队列。因此在实际使用中，为了获得更低的时延，最好是一帧数据就能把 Message 消息发送完成。比如 1.5KB 大小，保证空口一帧就发送完成，减少空口的资源竞争和退避带来的时延开销。
- 3) Byte 类型主要用于传输比 Message 类型消息大，时延要求没那么高的业务。比如传输

一个图片的缩略图。通常最大不超过 4M 大小。具体大小取决于设备的内存大小，有些设备内存小，则其 Byte 类型消息不会超过 4M。

4) SendByte 除了用于时延要求不高的基本业务数据传输外，也可以用于探测网络端与端之间的时延，比如探测当前网络传输 1MB 数据需要多少时间。

5) 在支持多种物理链路的情况下，不建议上层应用指定具体的物理链路，让 HarmonyOS 系统自动选择，系统会根据当前的网络情况选择最优的传输通道。

6) 传输的回调接口，不要有阻塞性动作，特别是对于持续性的传输，如果在回调中有阻塞性动作，会导致传输性能下降。

本次为大家简单介绍 HarmonyOS Message/Byte 类型消息的底层传输原理，这两个都是数据量比较小（Byte/M）且非持续性的消息传输，对于规格比较大（G）且有持续性传输要求的 File 和 Stream 类型数据传输，会在后续技术解析文章中进行讲解，敬请期待！