

< HDC.Together >

华为开发者大会 2020

# 分布式软总线

## 构建低时延高带宽的本地多设备虚拟网络

# 支持“超级终端”的关键技术与能力



用户体验  
如同使用一个超级设备

三方开发者  
可基于抽象的超级终端开发服务

HarmonyOS



< HDC.Together >

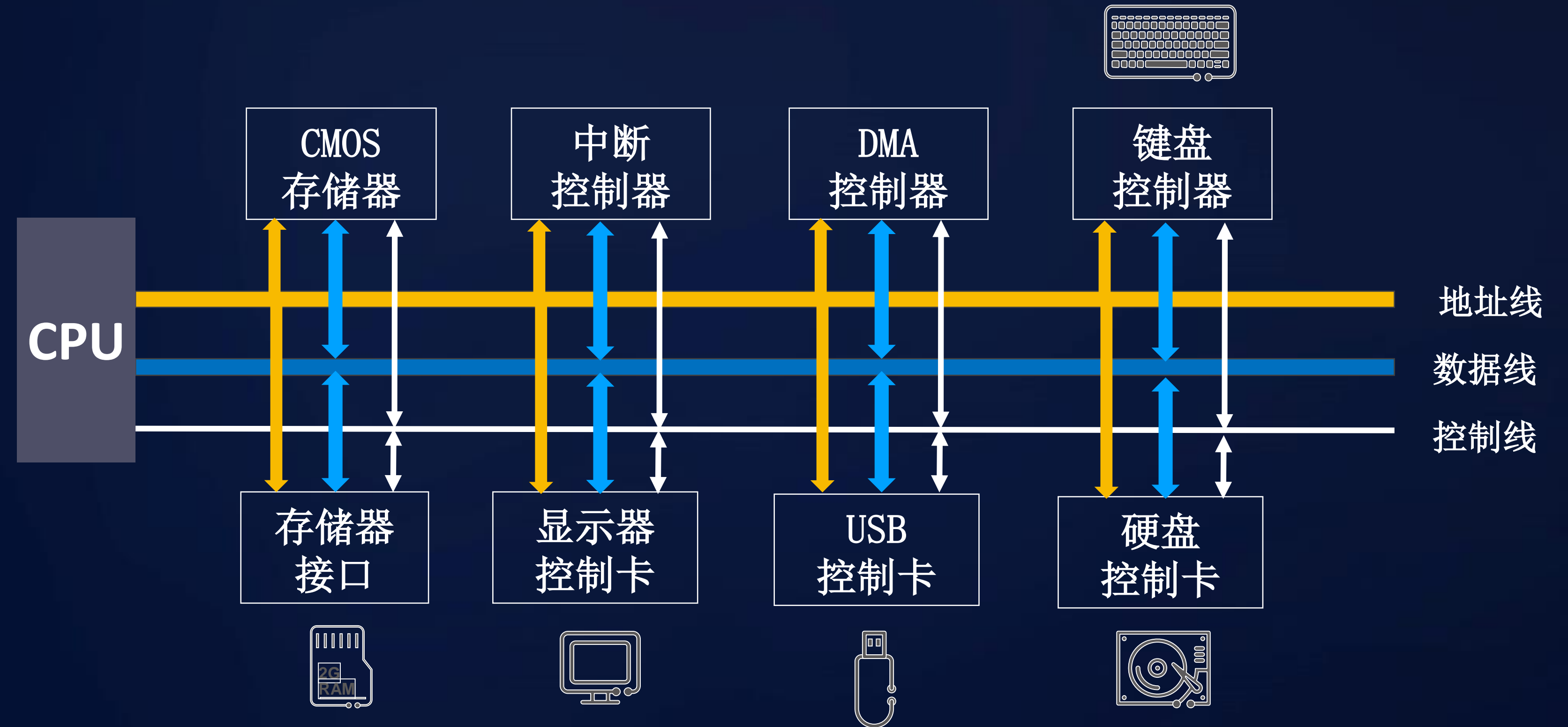
华为开发者大会 2020

# 计算机硬件总线结构



## 典型特征

- 即插即用
- 高带宽
- 低时延
- 高可靠
- 标准性



< HDC.Together >

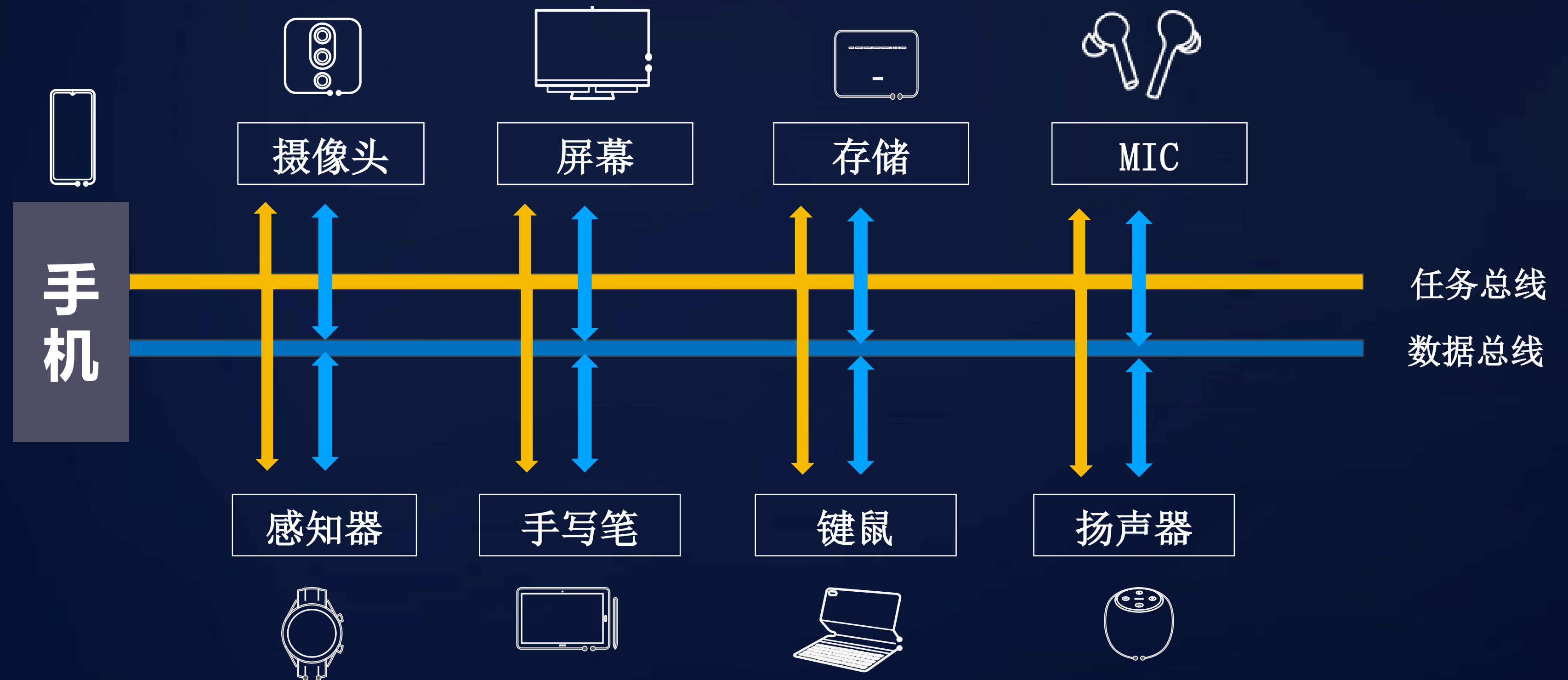
华为开发者大会 2020

# HarmonyOS分布式软总线



## 典型特征

- 自动发现/即连即用
- 高带宽
- 低时延
- 高可靠
- 开放/标准



< HDC.Together >

华为开发者大会 2020

# 分布式软总线架构





- 软总线之发现连接
- 软总线之组网
- 软总线之传输

# 软总线之发现连接：从手动发现，进化成自发现



## 手动触发

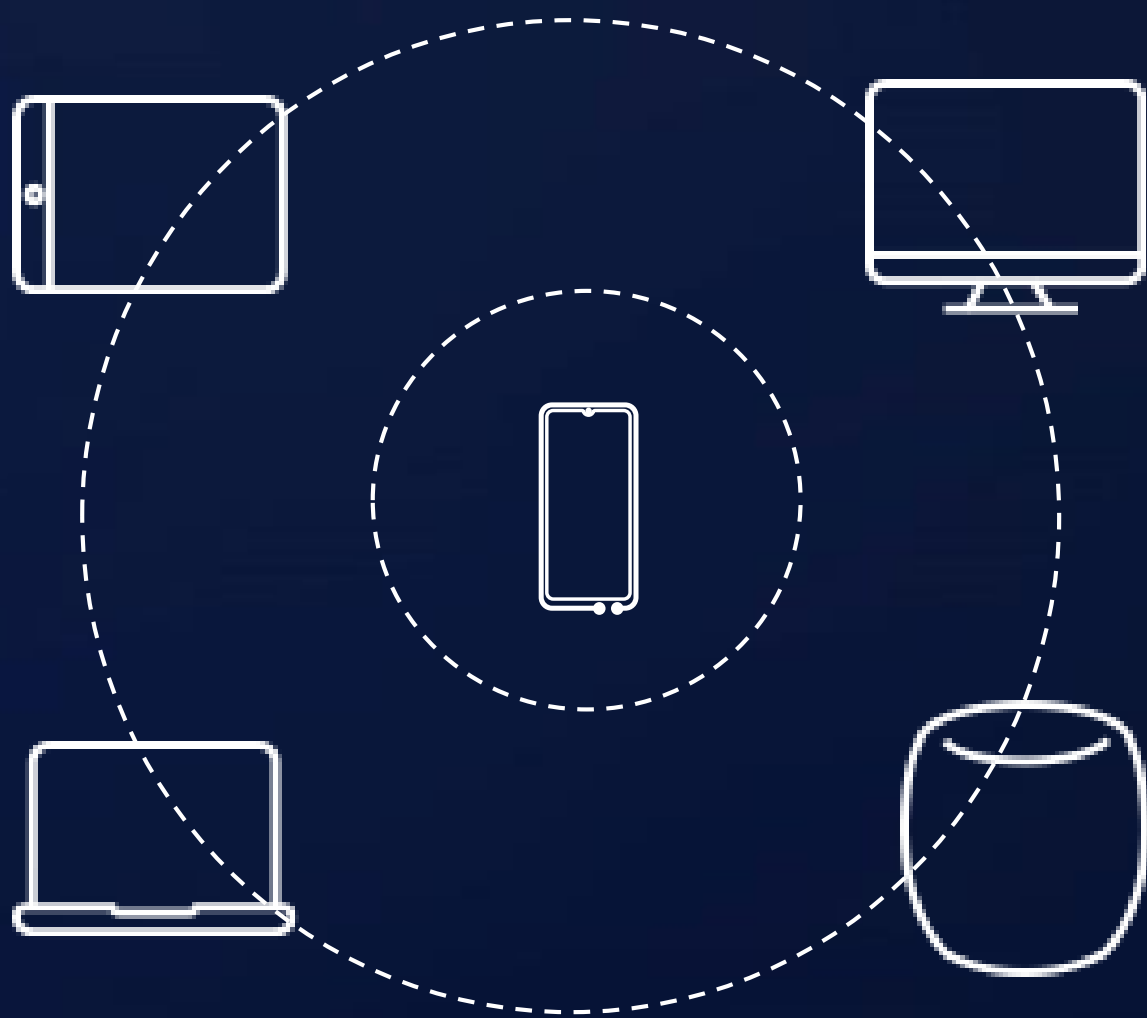
我是手机，有人在吗？



我呢，我是智慧屏



## 自动发现



< HDC.Together >

华为开发者大会 2020

# 用户零等待的自发现体验

- 附近同账号的设备自动发现
- 无软总线时，需要每次都搜索；软总线用户零等待

无软总线



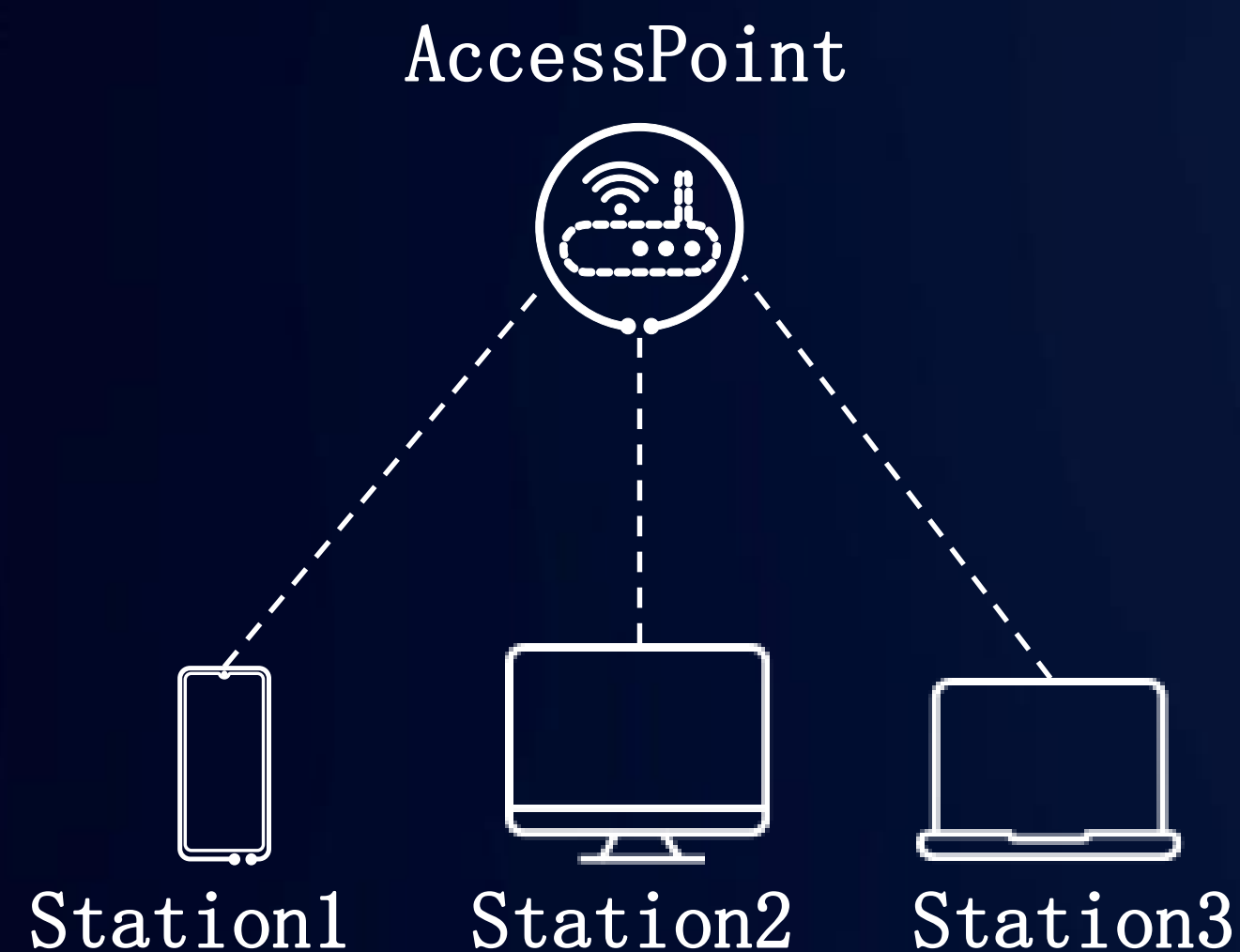
软总线



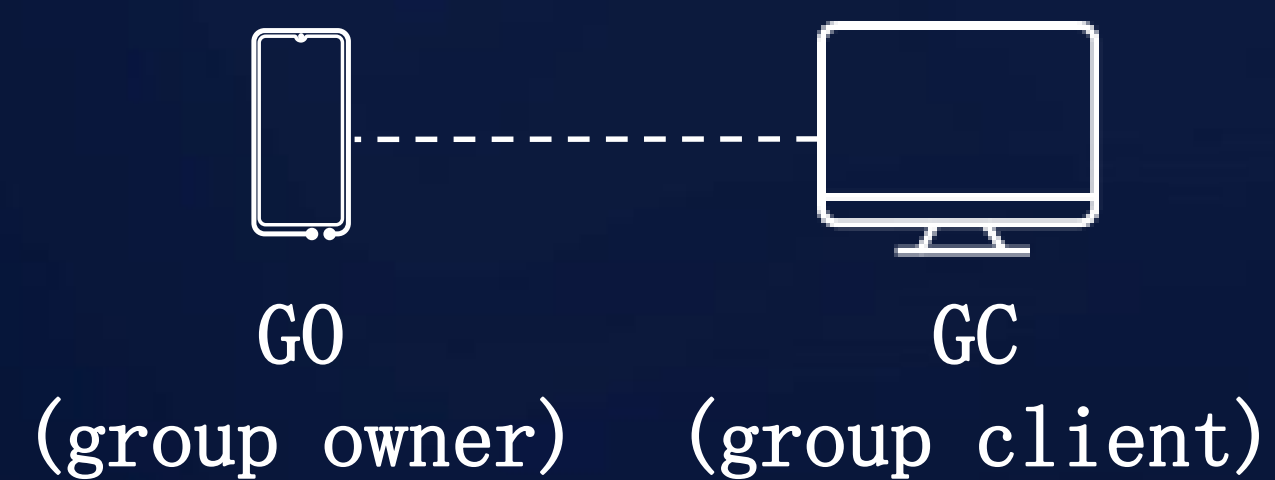


# 传统的组网模式

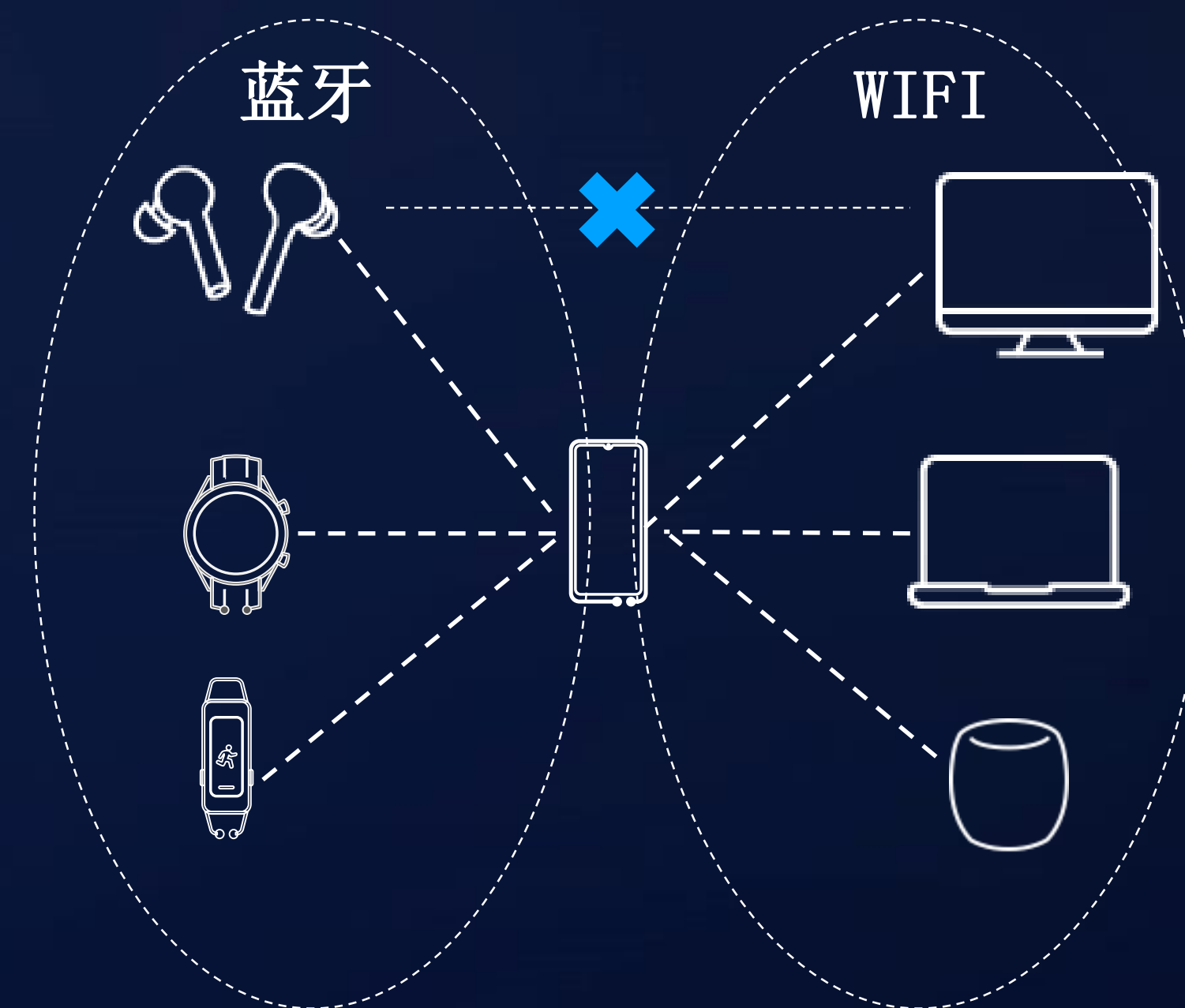
WIFI STA-AP  
依赖路由器



WIFI P2P  
仅点对点



蓝牙/WIFI网络隔离  
蓝牙WIFI无法互通



# 软总线组网关键技术-异构网络组网



- 目标：自动构建一个逻辑全连接网络，业务开发者无需关心组网方式与物理协议

- 组网API：设备上下线通知

```
public class DeviceManager {  
    SoftBus.registerDeviceListener(listener);  
}
```

```
public interface Listener {  
    onDeviceOnline (String NodeId);  
    onDeviceOffline(String NodeId);  
    onDeviceInfoChange(String NodeId);  
}
```



<HDC.Together>

华为开发者大会2020

# 软总线组网使能分布式开发新模式



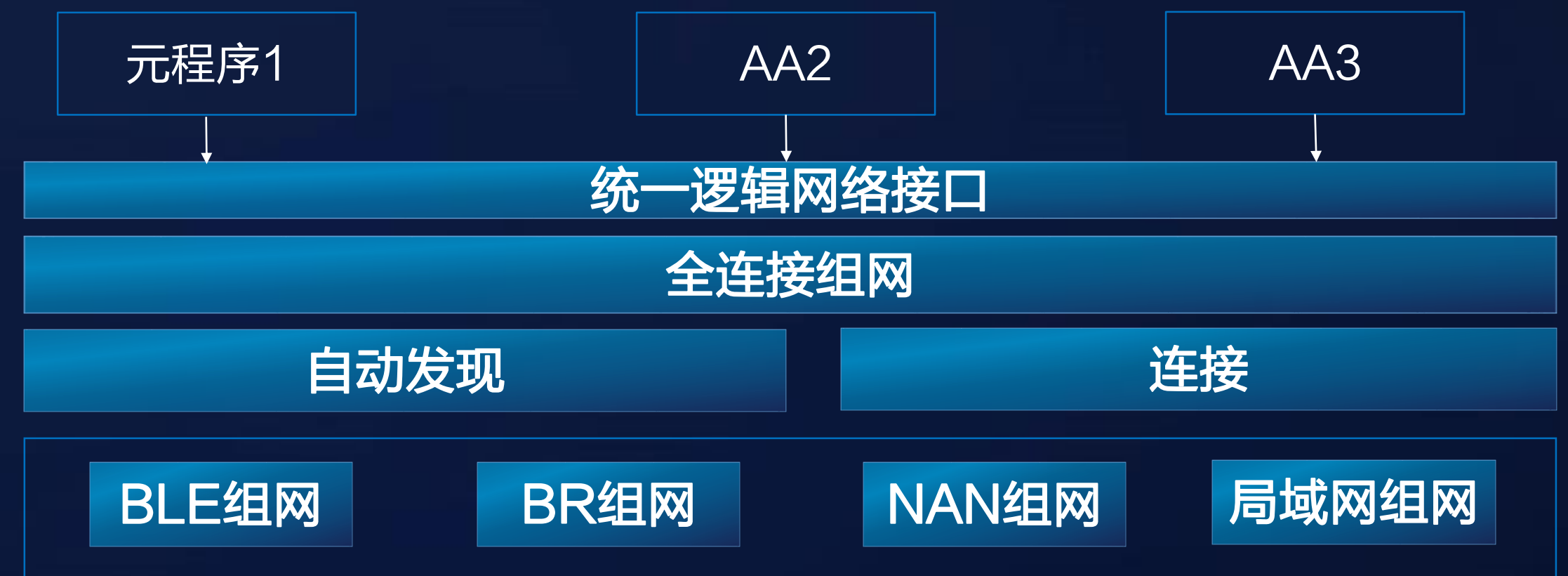
## 传统开发者模型

面向异构网络，多种协议



## HarmonyOS开发模型

面向一个逻辑网络，不感知协议



### 传统开发模式：

- 需要适配不同网络协议和标准规范，业务开发成本高
- 网络可扩展性差

### HarmonyOS分布式开发模式：

- 业务开发与设备组网解耦：业务仅需监听上下线，业务开发成本大幅降低
- 融合自组网：打破物理网络互通限制，异构网络自动组网，实现零等待发现组网

< HDC.Together >

华为开发者大会 2020



# 软总线之传输



高带宽 | High Speed



低时延 | Low Latency



高稳定 | High Reliability



传输极简API：文件、流、消息、字节

- `MySession.SendFile ( List<String> sourceFileList, List<String> destFileList )`
- `MySession.SendStream ( InputStream stream )`
- `MySession.SendByte ( byte[] data )`
- `MySession.SendMsg ( byte[] data, int timeoutMs )`

< HDC.Together >

华为开发者大会 2020



# 传输高带宽关键技术 – 极简协议

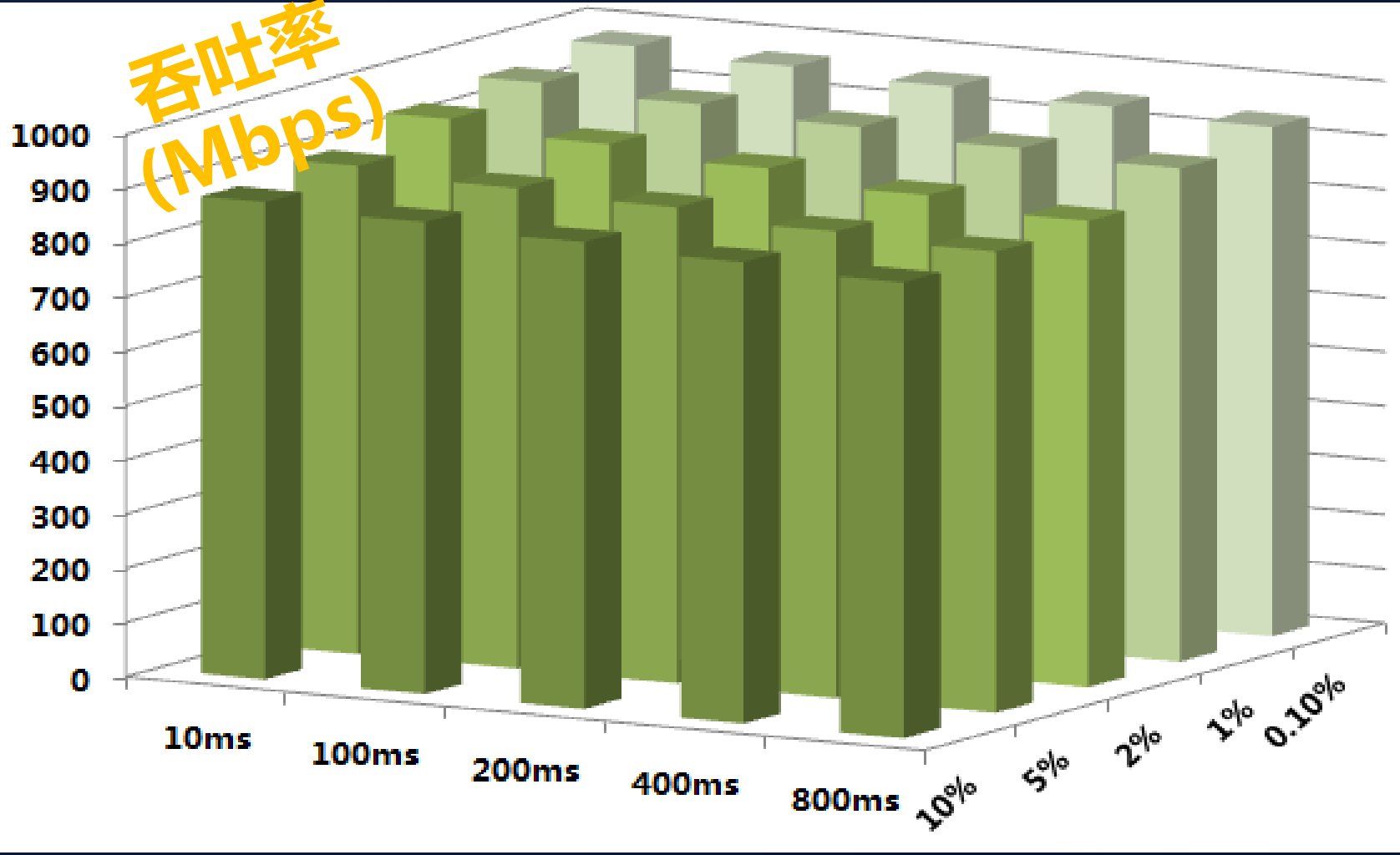


四层协议栈精简为一层 提升有效载荷



有效传输带宽提高20%

数据吞吐率  
不惧网络时延和丢包



- **流式传输**：基于UDP实现数据的保序和可靠传输
- **双轮驱动**：颠覆传统TCP每包确认机制
- **不惧网损**：摒弃传统滑动窗口机制，丢包快速恢复，避免阻塞
- **不惧抖动**：智能感知网络变化，自适应流量控制和拥塞控制

< HDC.Together >

华为开发者大会 2020

\*相关数据来源于华为实验室，模拟丢包与网络时延情况下测试对比，仅供参考

# 基于软总线的极简开发示意



手表当电视遥控器，抬腕即可控制，遥控器永远随身

开发难点：手表与电视无直接连接、跨蓝牙与WIFI网络



Step1: 基于自定义的服务名建立Session连接

```
session =  
OpenSession(deviceID, name);
```

Step2: 标准接口发送消息，直达对端服务

```
session.SendMessage(msg)
```

软总线跨物理网络通信

大屏应用处理  
消息

- 基于软总线开发：屏蔽组网和底层协议，代码量50行，业务开发者只需聚焦业务逻辑
  - 基于安卓开发：跨物理网络通信，需3~4K代码量达成商用，业务开发者需耗费大量精力实现底层通信
- < HDC.Together >

# HarmonyOS软总线—对开发者更友好



	A系统	HarmonyOS
发现连接	触发式发现，开发者感知WIFI/蓝牙等物理层协议	自动发现，开发者直接获取设备列表
组网	需要感知组网协议、跨物理介质网络无法直通	逻辑全连接网络，跨物理介质互通，开发者无需关心组网协议
传输	开发者感知WIFI/蓝牙等物理层协议，带宽、时延无保证	协议无关的文件、消息、流媒体传输通道，极简协议保证高带宽、低时延、高可靠

< HDC.Together >

华为开发者大会 2020



# 分布式软总线

- 超级终端通信底座，开启全场景智慧生活



Auto-Discovery  
自发现



Full Mesh Network  
全连接组网



Super Fast Transmission  
极速传输





欢迎关注HarmonyOS开发者微信公众号