

迎接物联网，拥抱新蓝海

—— 物联网操作系统漫谈

演讲者：杜玉杰

Huawei LiteOS 开源技术团队负责人

目录

1 物联网操作系统的机遇与挑战

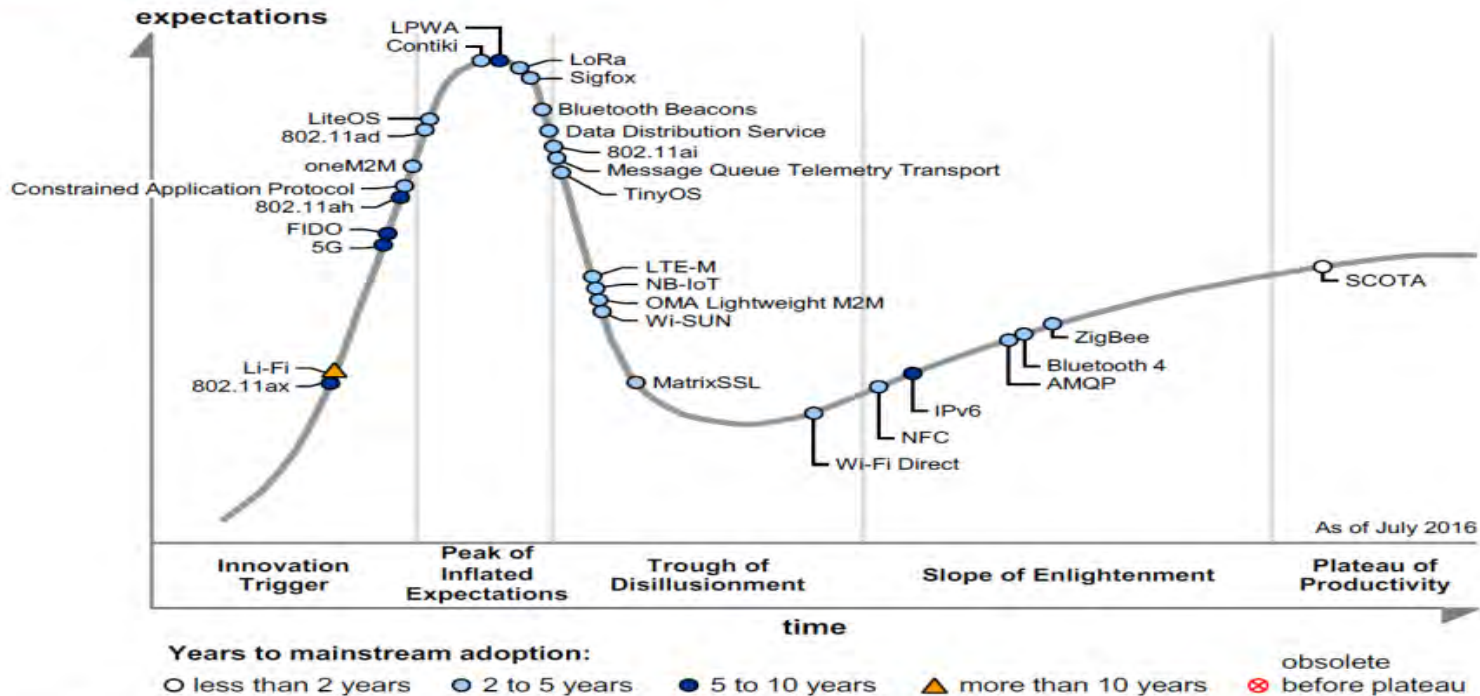
2 技术架构演进探讨

3 商业生态架构思考

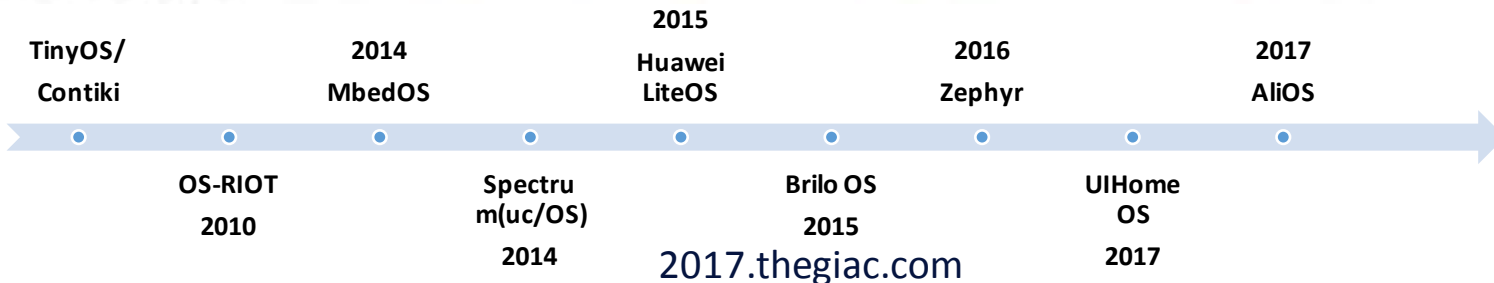
从IoT产业看，随着IoT产业升温各大厂商纷纷加大投入

Figure 1. Hype Cycle for IoT Standards and Protocols, 2016

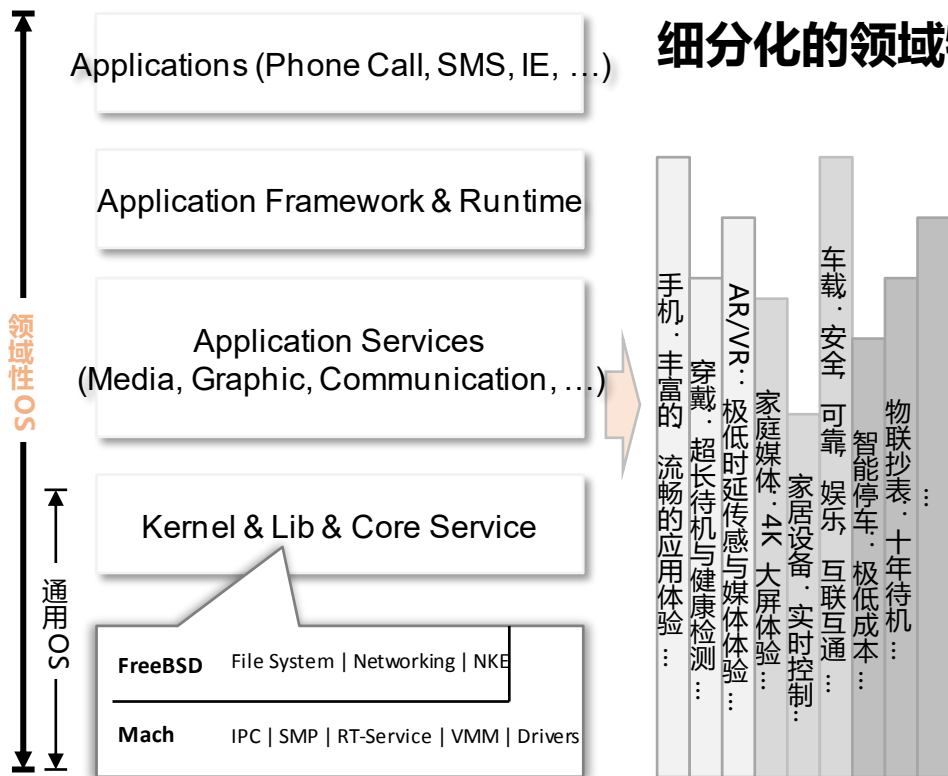
Source: Hype Cycle for IoT Standards and Protocols, 2016 (Gartner)



Source: Gartner (July 2016)



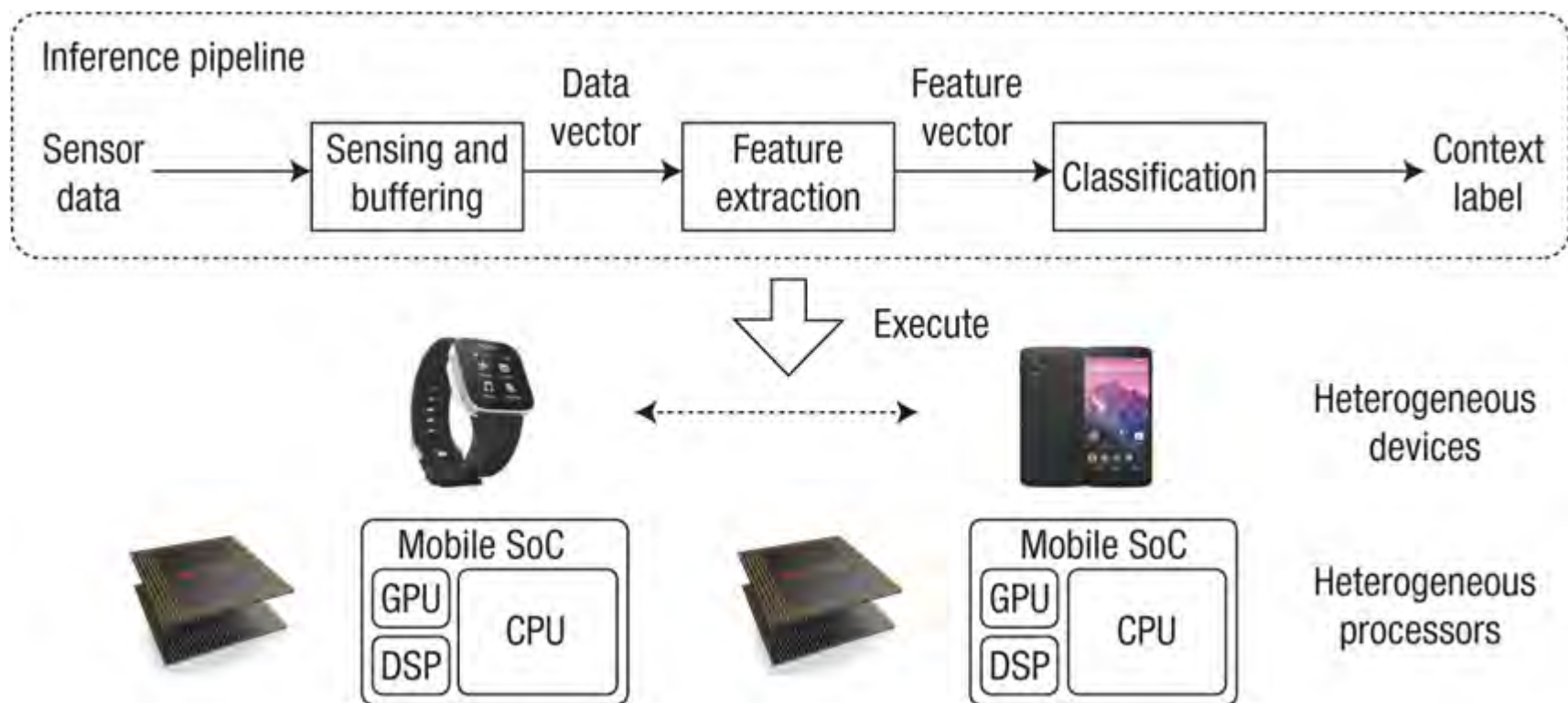
从IoT应用看，领域概念进一步细分、技术栈进一步延展



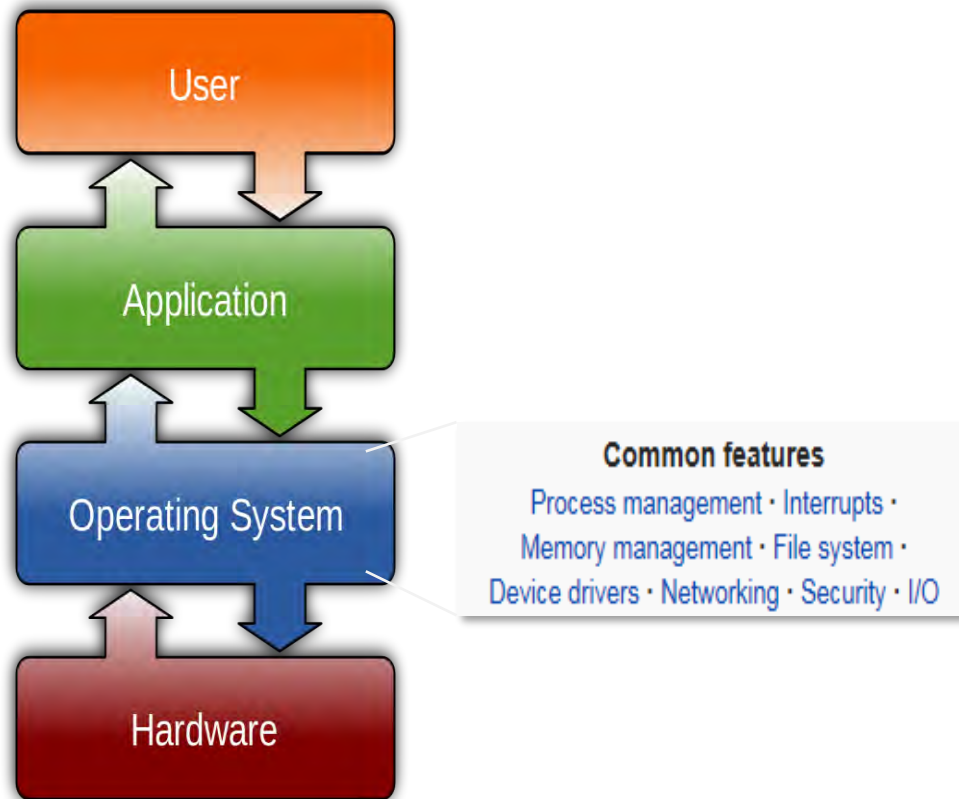
- 领域性 OS 一般包括通用操作系统内核和库、中间件、应用框架与引擎、应用服务、默认的应用和算法；优秀实践下沉，更加完善的技术栈，成为取得商业成功的关键因素，因为业务层面不再重复造轮子就可以直接获取最佳实践和技术；这个符合整个产业链条在做垂直整合的大趋势（包括软硬件的垂直整合，例如：Google 做 TPU 芯片的例子）。
- 当前大量领域并不存在领域性操作系统或者不完善（如没有固化优秀实践），特别是在终端侧，例如：智能家居
- 领域概念进一步细分，使得技术栈呈现更多发挥空间

人工智能可能成为IoT断代标志，设备端推理的应用场景更为多样化

智能手机、ADAS、智能摄像头、语音交互、VR/AR等设备需求各异，需要更为定制化、低功耗、低成本的芯片、操作系统支撑业务算法各个层面的技术颠覆



回顾：操作系统技术本质



https://en.wikipedia.org/wiki/Operating_system

- 从本质上讲，操作系统（OS）就是硬件资源与应用之间的“枢纽”，基于这个枢纽理念所进行的资源调度、进程管理、内存管理、终端管理、文件系统、I/O 处理、安全、驱动与设备管理等
- 经历40年的发展，当前存在 50+ 操作系统变体类型，30+ 商业产品；另外，单独统计 RTOS 领域就有180+ 实现或者开源项目；同时，我们也看到 180+ 项目中慢慢消失的操作系统实现已经超过 50%，为什么这么多会落寞？为什么最终商业成功只有那几个？
- LiteOS 从诞生到现在，一直都在思考这几个问题；

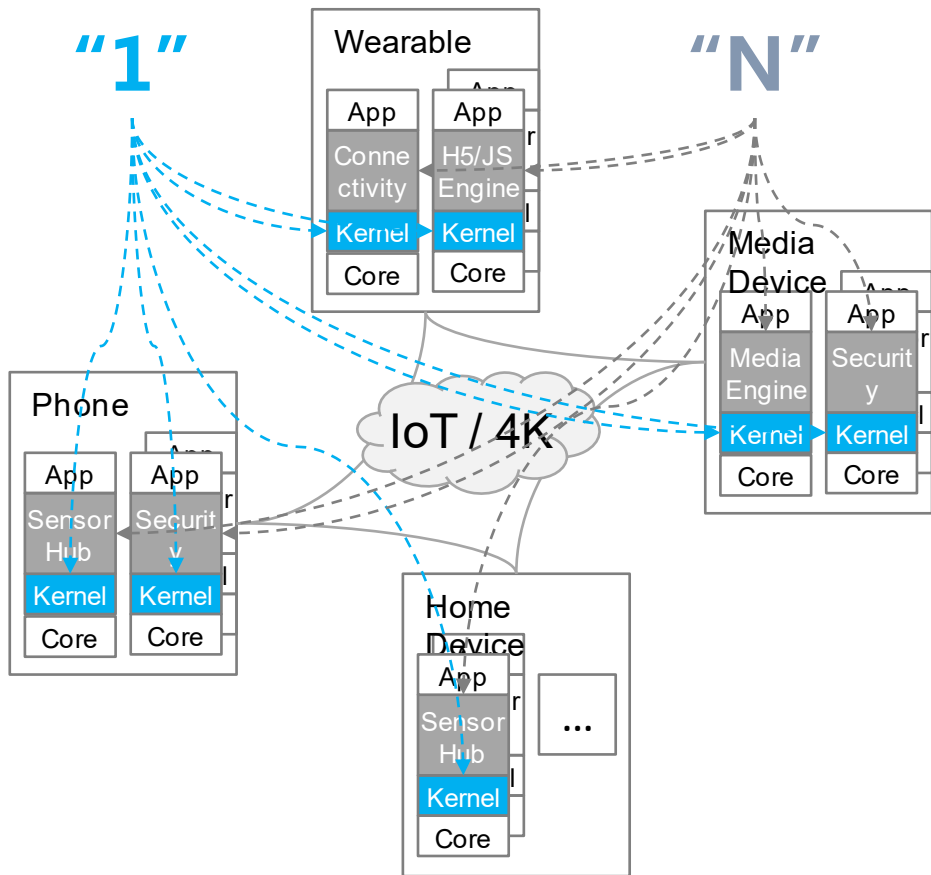
目录

1 物联网操作系统的机遇与挑战

2 技术架构演进探讨

3 商业生态架构思考

IoT OS技术架构演进



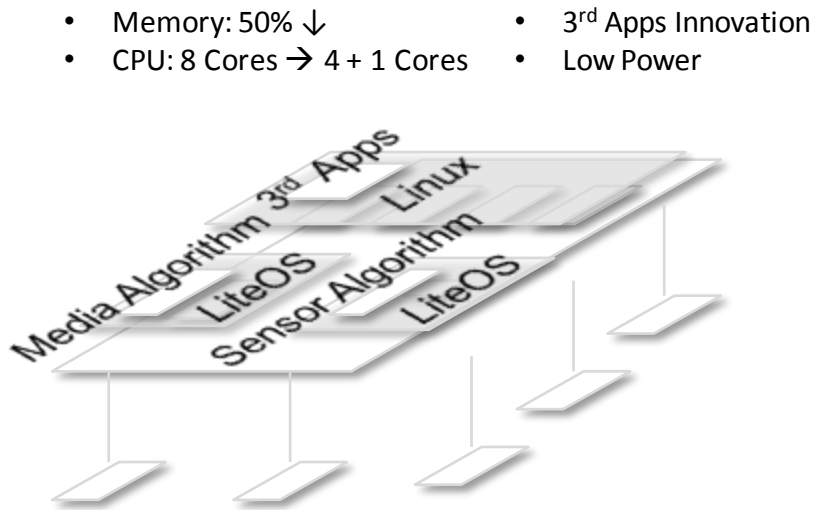
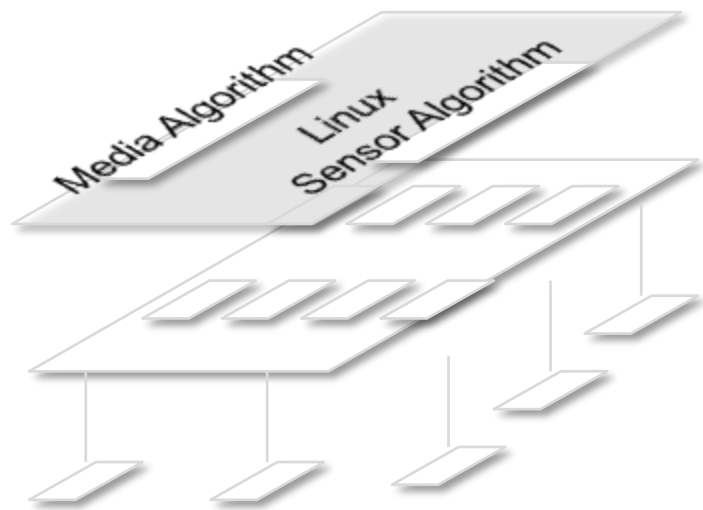
1. 业务诉求^{IPC}

2. 技术演进

1990 ~ 2010
2010 ~ 2030

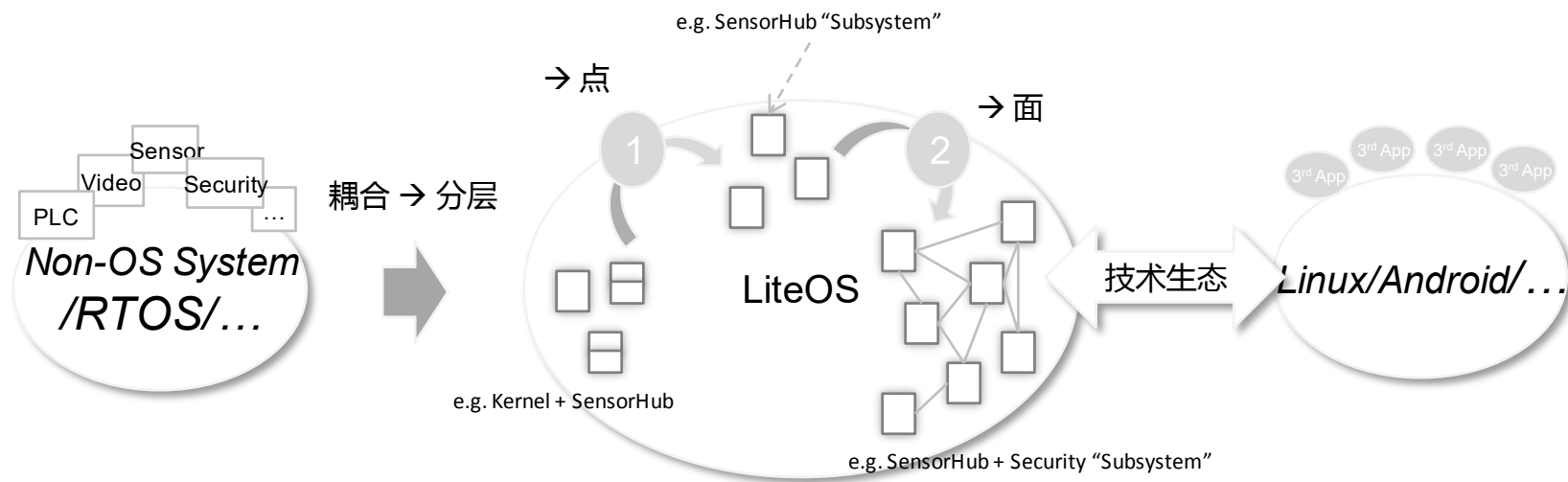
3. AIoT趋势

业务诉求：以手机为例



- LiteOS 在此案例中，共接管了 3 大子系统（媒体、传感、安全），接管后首先基于单点的优化，拉通上层媒体算法、传感算法等与芯片和硬件进行联合设计，并就整体软件包进行裁剪，实现了存储空间和内存占用各降低 30%、50%的效果，CPU 资源也减少了近40%，整体成本降低 ~1/3，并实现了超低功耗待机和第三方算法加载能力；
- 在新增的第三方算法支持方面，媒体能力和传感能力可以直接与第三方算法或者应用对接，支撑新业务创新

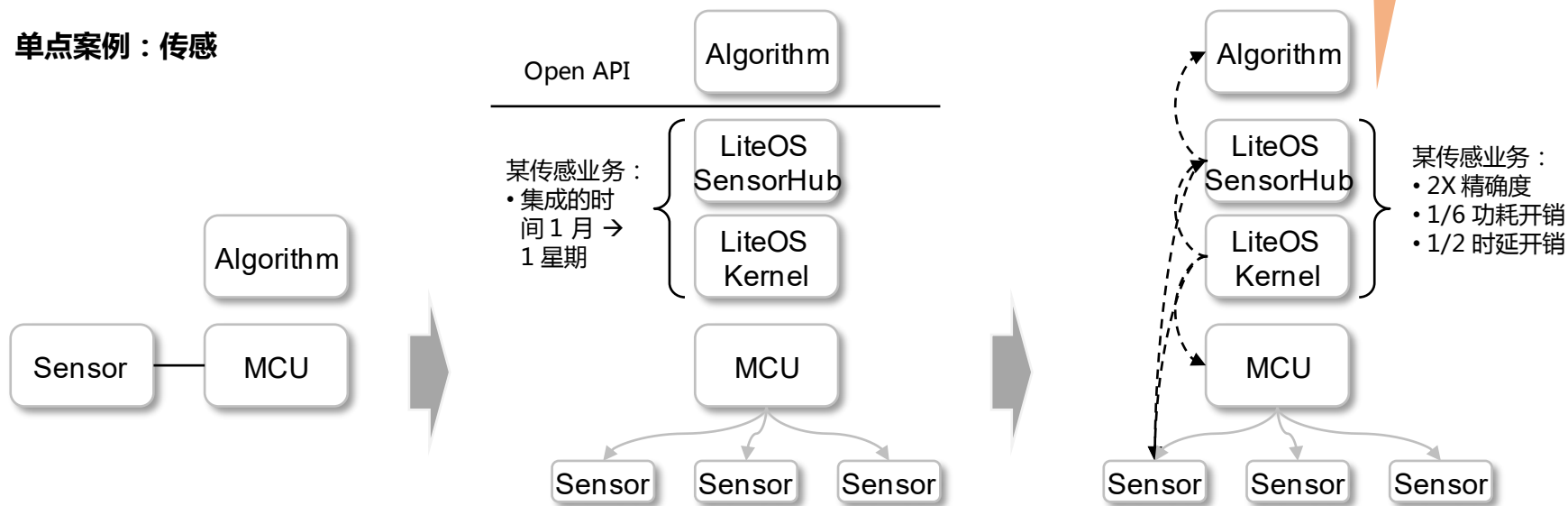
技术演进：拉通专有系统与通用领域的技术生态



- LiteOS 内部就单点能力形成最佳优化实践，包括传感、互联、安全、消息、Runtime 等子系统可以独立对外呈现为业务能力，而且在性能、功耗、时延等方面做到极致；另外在各个单点能力基础上，形成相互协同实现业务创新（手势解锁就是一个安全与传感结合的例子），并最终形成一个“面”整体对外呈现业务能力；这里的“点”、“面”都是完整的“麻雀”……
- LiteOS 与其他系统协同、优势互补，例如与 Linux/Android 系统的互补，进一步拉通特定领域与现有通用领域的技术生态。

点：垂直整合，发挥极限能力

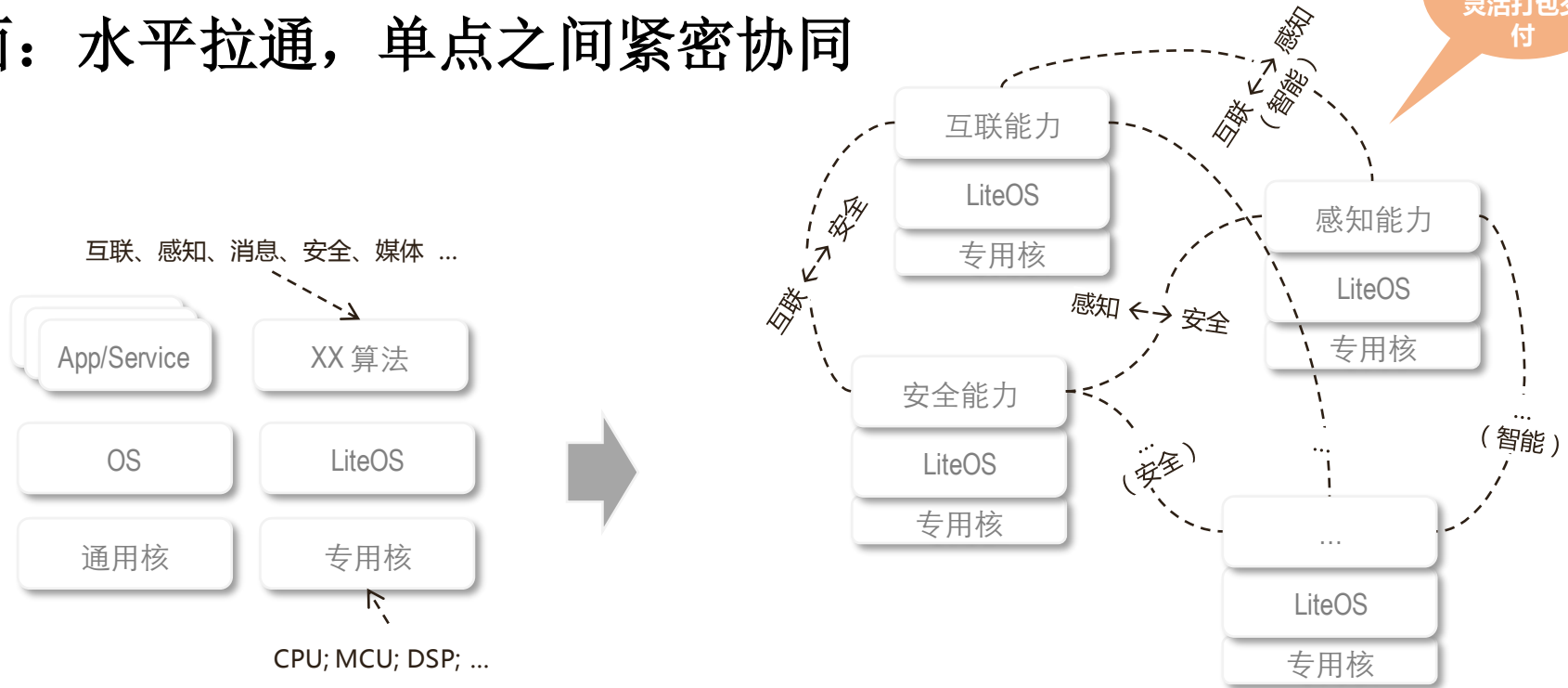
单点案例：传感



- LiteOS 单点能力就是在分层的基础上，针对该点的业务特征（比如传感、互联等），将从芯片/硬件、LiteOS 内核、LiteOS Framework 等部件垂直拉通优化，实现业务极限能力的同时，减少系统开销（CPU、内存、功耗等）；因此单点的整合，就是某个业务子系统的垂直优化整合，是具备业务特征的 E2E 的优化；
- 除了 LiteOS 本身的优化外，也规范了芯片/硬件的架构和设计，以及规范了上层业务逻辑和算法的设计，将最佳实践固化下来

LiteOS 系
统独立部署
灵活打包交
付

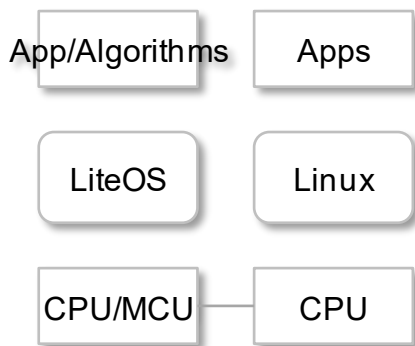
面：水平拉通，单点之间紧密协同



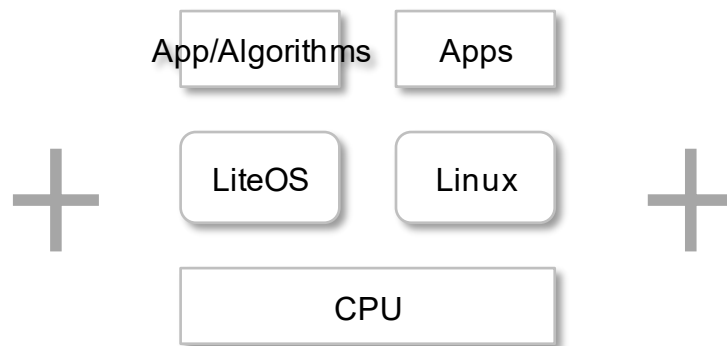
- LiteOS 点 → “面” 的能力就是针对复杂业务场景，特别是进行跨领域的业务创新，水平拉通所需的技术栈和数据通道，形成一个平面的对外能力；这个平面是建立在单点优化到极致的基础上，因此是一个“虚拟的平面”（原则：不损害单点极限能力）.....
- 随着平面能力的不断完善，反过来不断牵引单点能力的不断优化，从而实现单点能力在自身做到极限优化的同时，也确保其支撑的平面能力也能达到极限；类似过程不断的反复，并最终在所针对的业务领域达到最佳的平衡（后面会详细介绍到领域技术栈）。

与 Linux 协同， LiteOS for Linux

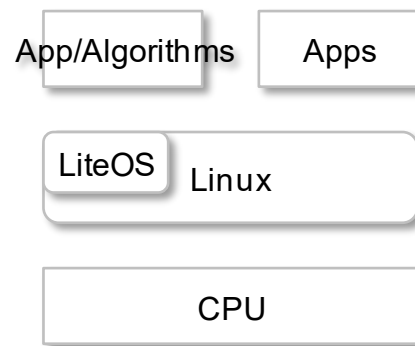
独立部署协同模式



类似 AMP 部署协同模式

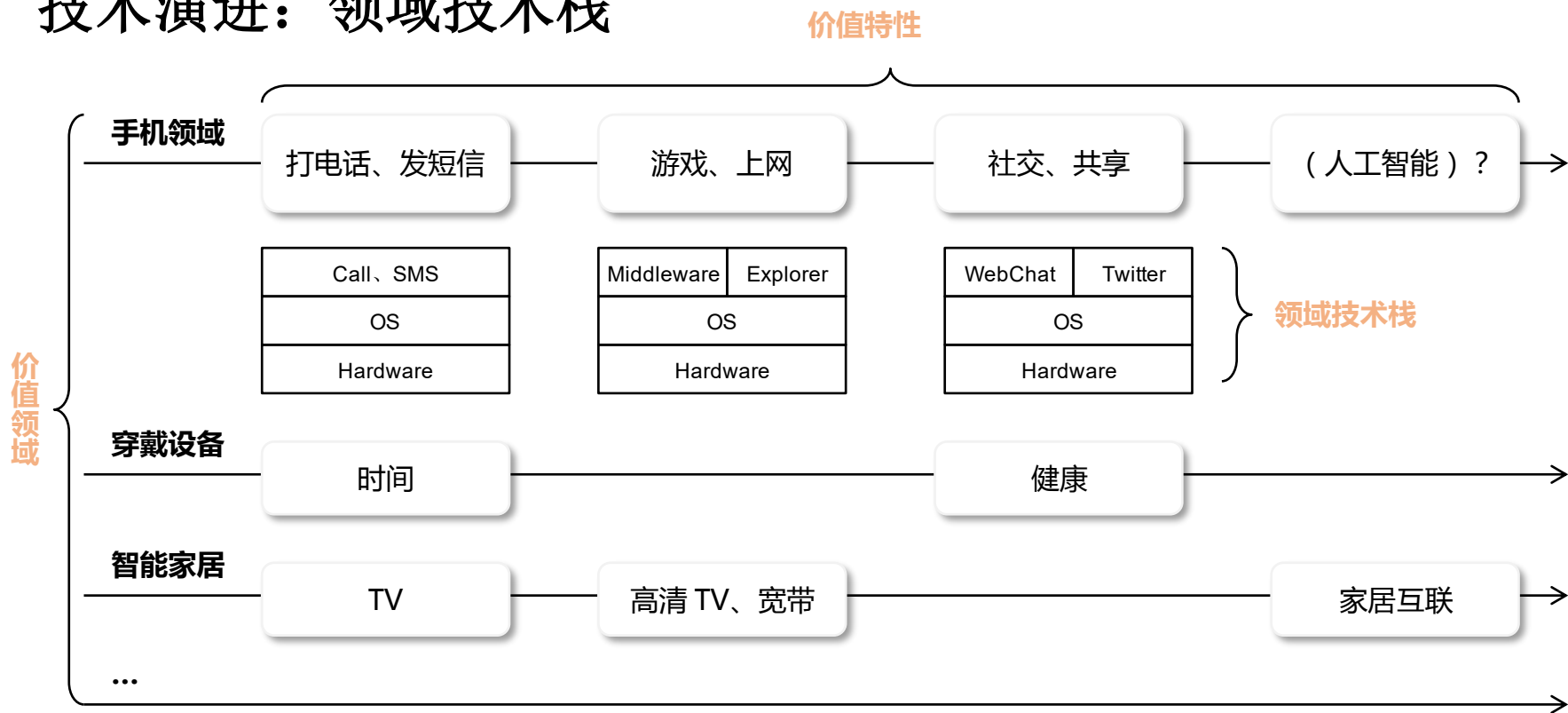


技术组件内嵌协同模式



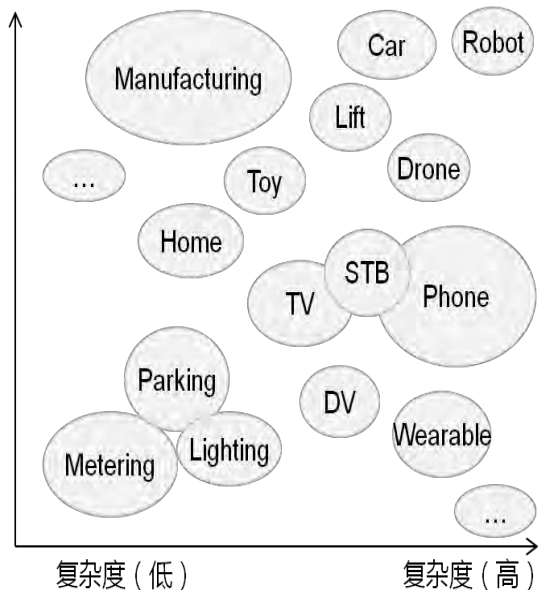
- 由于 LiteOS 支撑的业务特征关系，LiteOS 诞生之初就考虑了与 Linux 等系统的协同关系，这种协同关系将在很长时间内存在；目前 LiteOS 已经在支撑的某产品验证过独立部署协同模式、类似 AMP 部署协同模式，技术组件内嵌协同模式也正在其他商用产品上验证中，基于当前的验证结果，从最终用户角度出发体验收益较大，产品成本也因为整体的协同而降低；
- 随着未来物联智能设备硬件资源越来越丰富，这种协同关系将会更加紧密，LiteOS 将持续挖掘与 Linux 等操作系统的协同点

技术演进：领域技术栈

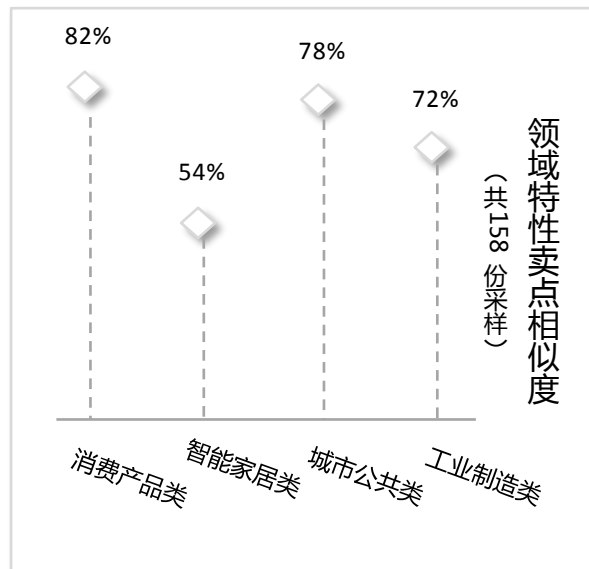


- 像手机操作系统一样，领域性操作系统将实现完整的领域性“技术栈”（包括领域性基础应用），部署即可使用；而随着业务诉求的变化，不同阶段的领域性业务和特性有不同价值呈现，因此构建该阶段的高价值领域“技术栈”将成为领域性操作系统成功的关键，即高度定制优化的“技术栈”应该聚焦支撑该价值领域和该价值领域特性

为什么 LiteOS 要支撑领域技术栈

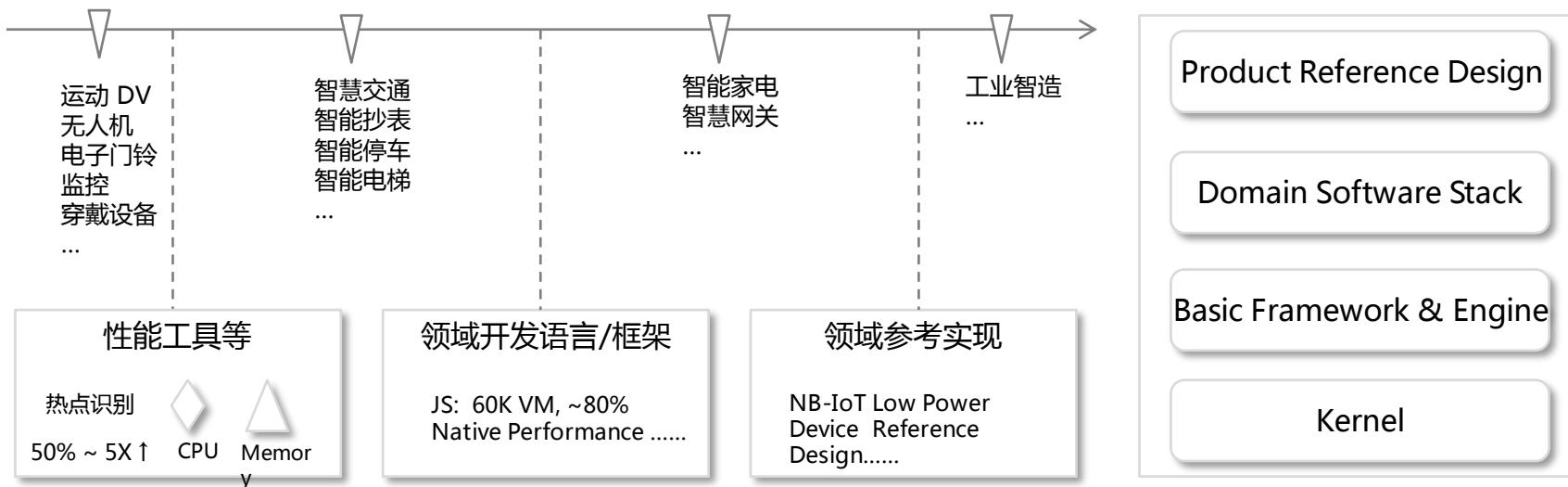


- 手机：丰富的、流程的应用体验 ...
- 穿戴：超长待机与健康检测 ...
- AR/VR：极低时延传感与媒体体验 ...
- 家庭媒体：4K 大屏体验 ...
- 家居设备：实时控制...
- 车载：安全，可靠，娱乐，互联互通 ...
- 智能停车：极低成本 ...
- 物联抄表：十年待机 ...
- ...



- 从技术上讲，LiteOS 内核和框架需要满足价值领域特性的最佳诉求，需要从技术层面解决价值领域特性最具技术难度的点，这个是判断单点与面是否做到体验极致的标准，也是 LiteOS 作为领域性操作系统的关键价值所在
- 从业务上讲，LiteOS 支撑价值领域技术栈，也就解决了领域性 50%~80% 的共性问题，可以大大节省领域业务持续创新的时间。

如何支撑领域技术栈



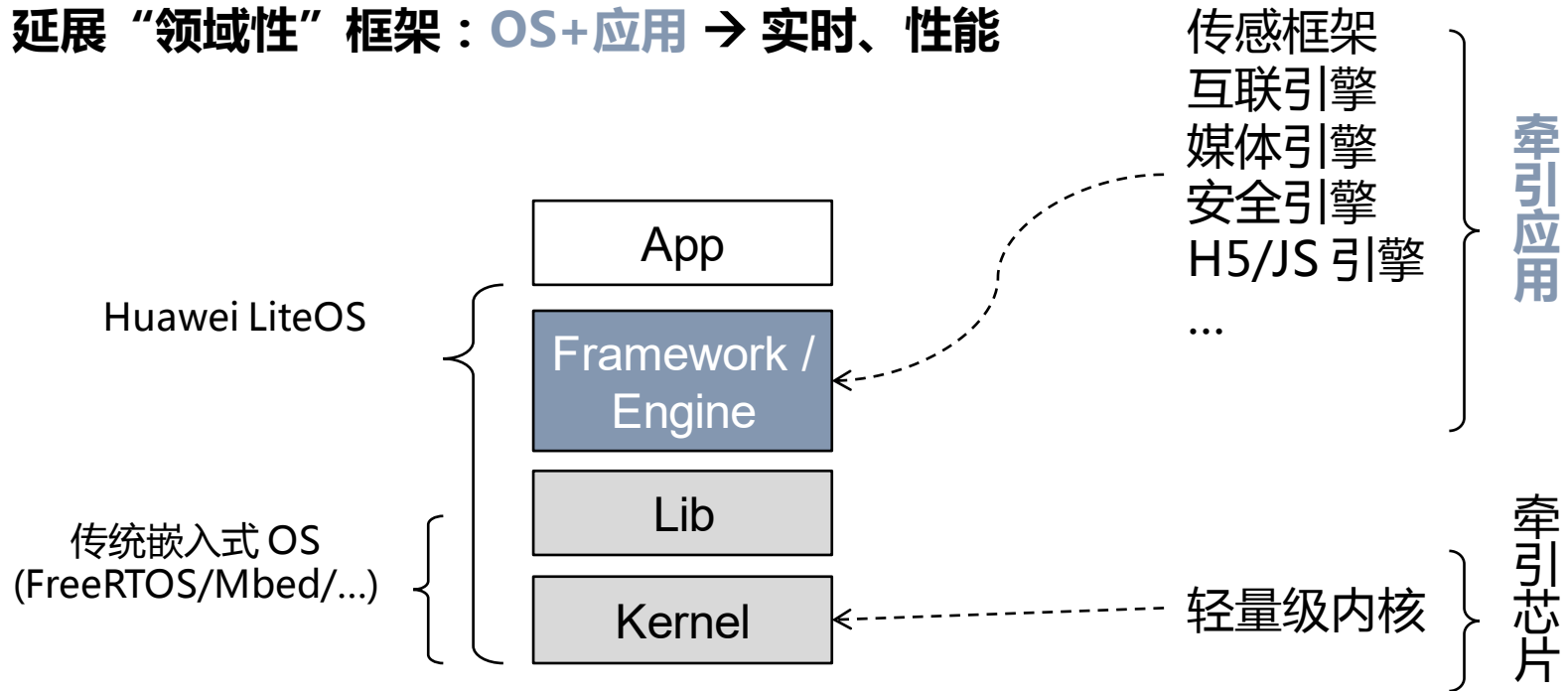
- LiteOS 会在支撑的产品的基础上，不断总结技术经验，并形成领域性技术栈，结合当前 1+N 的技术架构（一个轻量级内核，N 个基础技术框架和引擎），支撑领域性关键特性实现最佳体验；
- 从技术层面，首先提供一系列优化工具，专门识别当前相关领域的问题瓶颈点；其次会通过抽象领域性概念形成 API 和框架，固化优化经验；最后会打造一个完整的参考实现（与合作伙伴构建），基于参考实现验证领域技术栈的最极限能力

解释一点， **LiteOS** 不碰数据、不做业务

- LiteOS 支撑领域技术栈并不涉及上业务的构建和业务数据的获取，只考虑如何更好的通过领域技术栈加速领域业务的构建和体验的提升，同时减小价值领域、价值特性所需技术栈的碎片化问题；
- LiteOS 跟合作伙伴一起定义领域性操作系统，LiteOS 作为核心技术平台支撑合作伙伴构建最佳体验的领域性操作系统，即 LiteOS 提供领域技术栈，合作伙伴提供领域业务和价值特性；
- 对于新的创新领域，LiteOS 可以和合作伙伴一起尝试制定相关领域性操作系统的参考实现

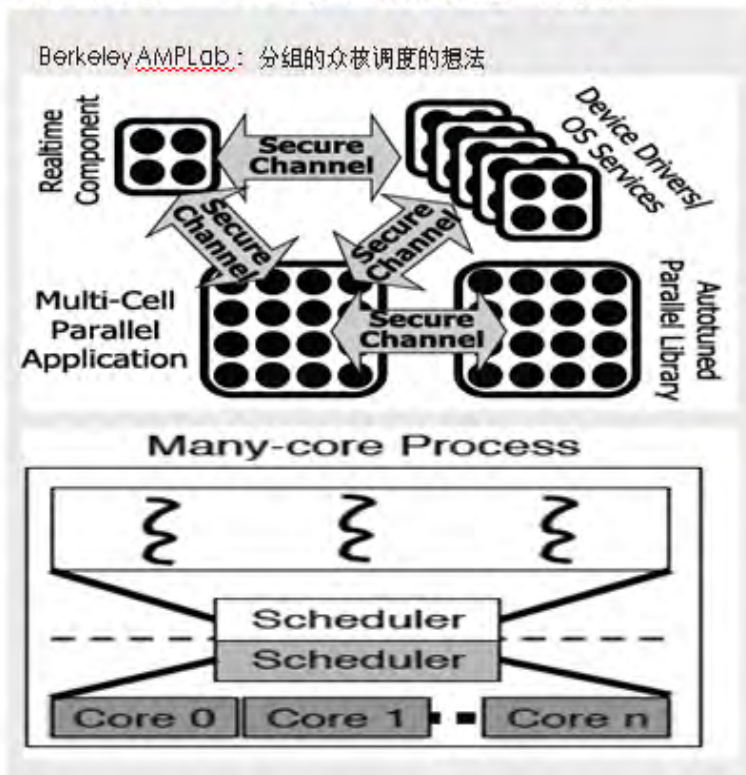
同时，OS 范围将从聚焦内核，发展为聚焦内核 + 应用框架

延展“领域性”框架：OS+应用 → 实时、性能



并且，基于“同构”扩展，演进为基于“众核+异构”扩展

随着海量数据处理、高性能计算进入普通业务应用场景，要求 OS 基于众核提供不同特征任务（实时性、并发性等）的分类调度和针对性能优化能力（预计5年后：>100核，10年后：~1000核），确保高性能、高效率。

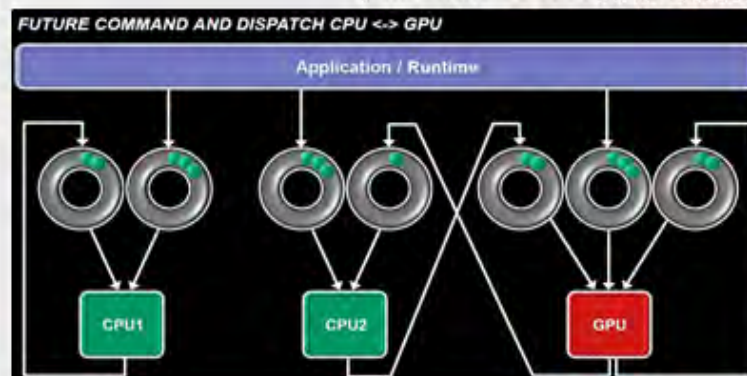


随着应用复杂度提升，网络设备功能融合，要求 OS 根据不同功耗、不同特征（计算密集、I/O 密集、媒体处理等）的功能模块到异构核执行，确保高效率、低功耗。

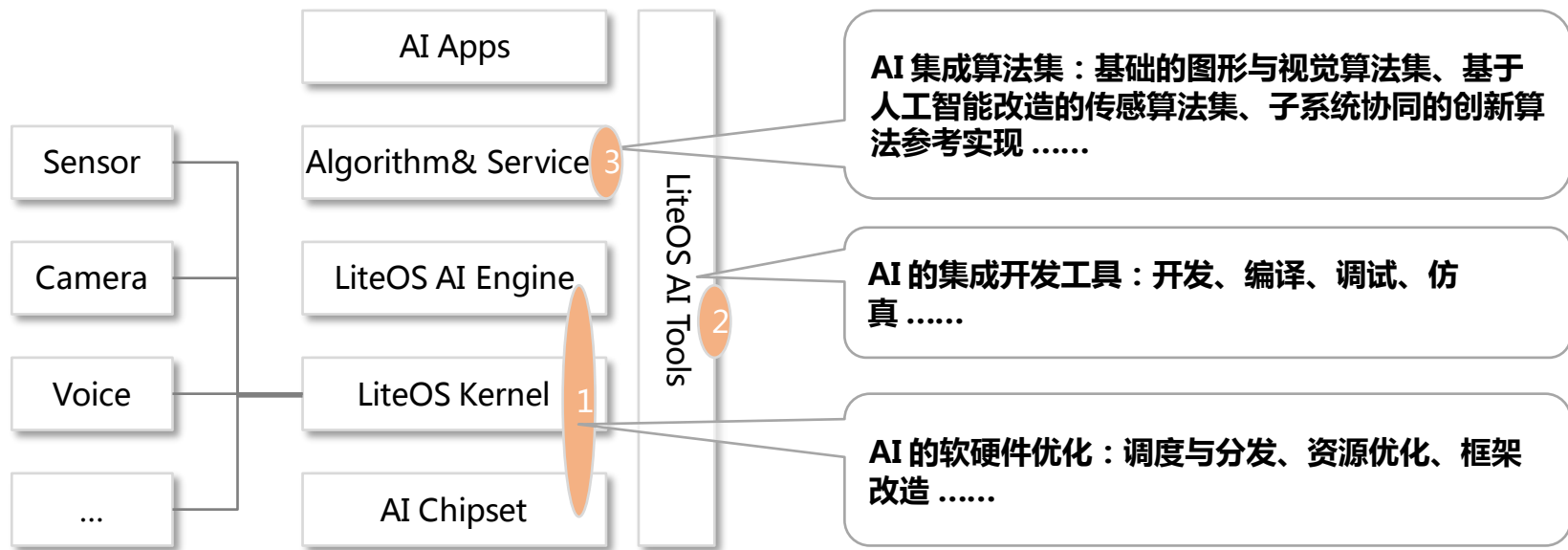
- 通用 CPU + 低功耗 CPU + GPU 等；
- 针对计算密集型、I/O 密集型、混合型等优化；
- 编程接口简化，应用无感知异构核差异；
- OS 负责任务在不同处理器核间无缝迁移，并针对性优化（性能、能耗、可靠性等）。



(AMD Fusion E, T, A 系列异构核思路)

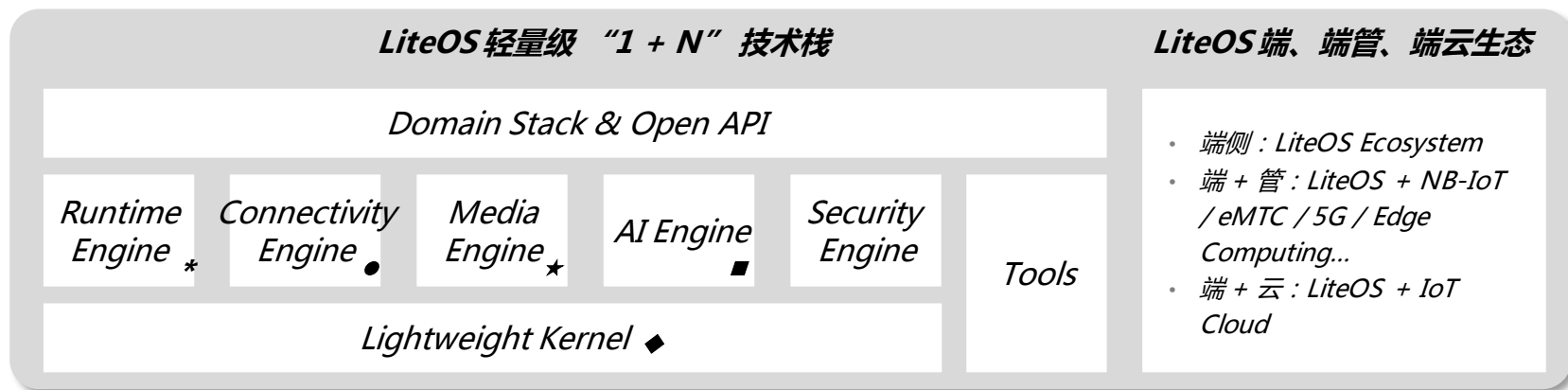


LiteOS 在人工智能领域的技术思考



- LiteOS 会针对终端侧改造当前的技术栈，包括硬件协同、配套工具构建、基础算法集（与合作伙伴一起实施）
- 同时，会考虑现有支撑的场景的算法改造，包括传感、视频与图像、语音等方面，结合人工智能的技术，尝试改造现有的算法，确保其在基本的特性之上，具备人工智能特征；在这个过程中，将充分拉通传感子系统、Camera 子系统、语音子系统等方面与人工智能子系统的协同；后续会针对不断拓展的业务场景进行基础算法改造和集成。

“1+N” 架构打造领域技术栈提升端云协同的开源开放能力



Hisilicon (BVT / STB / DPT / HCT Chipsets, Kirin 9xx Chipset), 3rd Chipset

未来3~5年，LiteOS 希望打造端侧领先的连接、传感、智能等“系统级”体验和生态，塑造物联领域的“Android”！

目录

1

物联网操作系统的机遇与挑战

2

技术架构演进探讨

3

商业生态架构思考

生态系统正从封闭走向开放，企业的关键能力正在发生变化

开放的含义包括商业概念、研发平台，不仅仅是接口...

1

设备制
造商

封闭、内生

关键：技术实力
抓手：标准、专利

10³

vmware + CISCO + EMC
+
Microsoft + Intel

联盟、合作

关键：联盟能力
抓手：标准、专利、
联盟、商业模式

10
>100

android
amazon
webservices

开放、平台

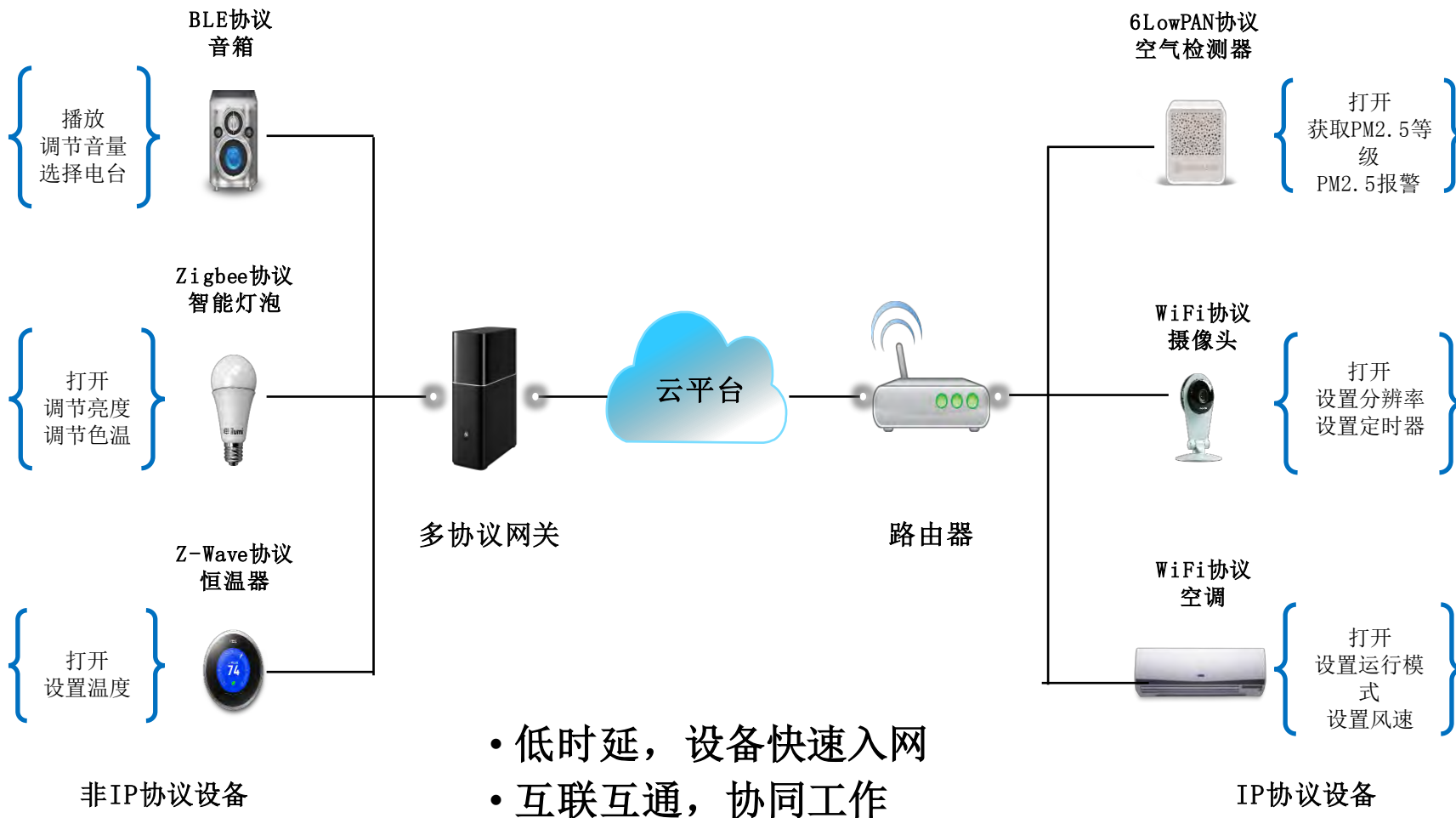
关键：生态系统构建
抓手：用户、数据、
开发者

技术及商业门槛降低，促进生态从内生型向开放性演进

“在任何一个新的、大的市场上，我们不知道接下来市场发展的方向，我们也不知道会有什么样的应用，我们也不知道谁有非常好的方法，我们唯一能够探索的就是这样一种新的行业方式，就是**创建一个生态系统，让每个人都可以合作创新，一起来界定未来。**”

---- 前《连线》杂志总编，《长尾理论》、《创客》等作者，3D Robotics公司首席执行官克里斯·安德森 Chris Anderson

共建联接、开放、合作共赢的智慧家庭



连接+共享单车平台孕育更多商业模式

NB-IoT战略合作，平台共赢，服务共赢



2月22日，ofo宣布与中国电信、华为达成合作，三方将共同研发基于新一代物联网NB-IoT技术的新型共享单车，共同打造全球首个共享单车行业标准



ofo 创始人兼CEO戴威表示：“经过一年半时间的发展，ofo共享单车已经成为全球最大、市场占有率第一的**共享单车平台**，此次跟中国电信、华为达成三强联合，三方将以全球领先的实力探索共享单车行业新高度，继续引领行业发展，为世界用户提供最好的共享单车产品和服务。”



中国电信北京公司总经理张志勇表示：“中国电信北京公司将与物联网行业伙伴、客户强强联手承接国家创新战略，为政府及企业信息化建设提供综合智能信息服务。基于中国电信800MHz频段的蜂窝窄带物联网技术，三方开展共享单车联网合作，相信未来一定能够提供更便捷的出行服务。”



华为无线网络产品线首席战略官余泉表示：“共享单车是典型的低功耗广域物联网应用，NB-IoT是专为此类场景设计的物联网技术。此次共享单车的合作是NB-IoT技术的又一个应用突破。”

未来，更广泛的生态与技术覆盖，更多智慧化业务

基于ITU定义的智慧家庭网络业务及终端



华为开发者使能“沃土计划”：构建更加强大、可持续的生态系统



GIAC | 全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



扫码关注GIAC公众号

2017.thegiac.com