

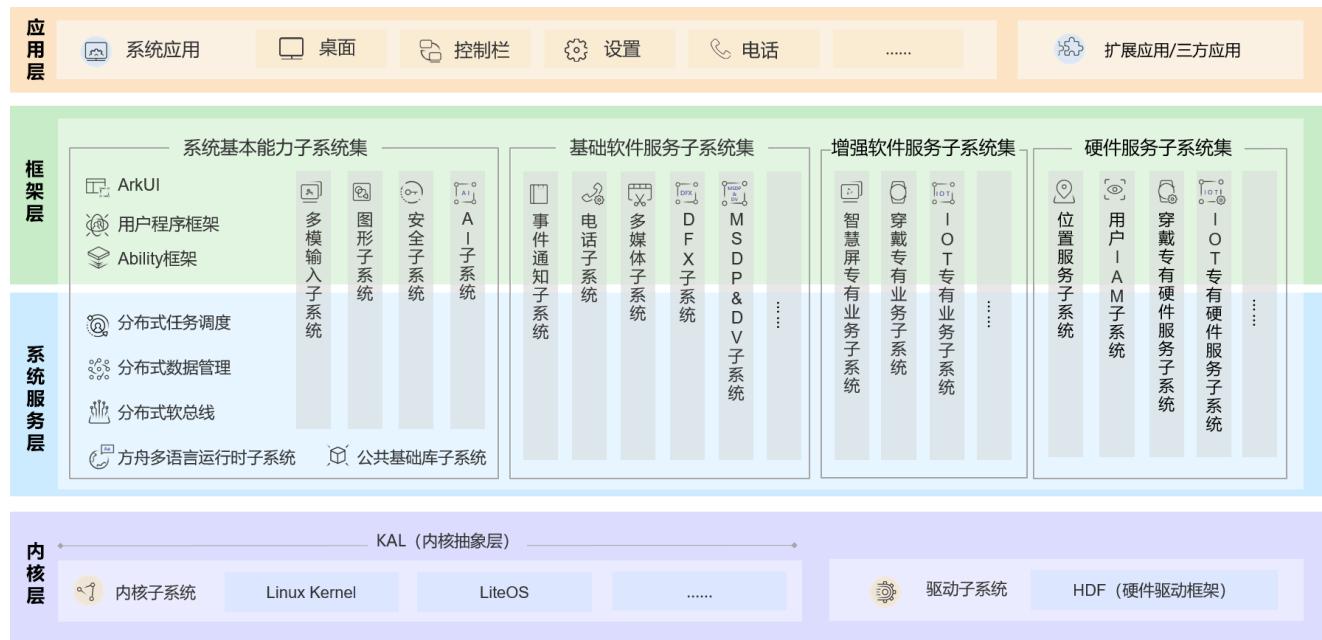
鸿蒙优化-调研报告

鸿蒙系统简介

鸿蒙操作系统简介：

1. **4.X** 版本之前兼容安卓 **AOSP**, **4.X** 版本之后华为设计了双系统，一套是原生安卓，用于维护安卓生态；一套是自研 **LiteOS** 微内核，用于服务物联网轻量设备
2. **5.0** 版本之后采用纯血鸿蒙，也就是全链路自研，不再兼容安卓

鸿蒙的设计架构如下：



详细解释每一层：

1. 内核层：内存管理，文件系统，进程管理，设备管理等使用 **Linux**（手机，平板等高性能设备）/ **LiteOS**（摄像头等嵌入式设备）；把手机上的蓝牙设备、网卡、摄像头、屏幕等和内核的 **C API** 接口封装到一起，对上层提供统一的 **C API**，叫做 **KAL**（内核抽象层）
2. 系统服务层：把内核抽象API进一步以服务类型划分进行封装，比如电话、消息、相机、智慧屏、定位等各封装出一套调用库，方便上层使用
3. 框架层：把基础服务综合到一起，抽象出一套APP 生命周期管理、资源分配、UI绘制的开发套件(SDK)共用户开发
4. 应用层：使用SDK开发

任务测试

目标任务

优化Top6在华为机器上的冷启动性能

跟华为的会议记录显示：**Ability** 加载阶段性能差，页面加载和刷新性能差

思路：

1. 分析 **Ability** 框架性能瓶颈, 优化 **Ability** 框架性能
2. 优化华为的虚拟机平台(即开发语言 **ArkTS**)的性能, 这一部分设计到编译器设计, 与我们实验室差太远, 不予考虑。

目标机测试

机器配置如下所示:

- 机器型号: HUAWEI Pura70 Pro
- 鸿蒙内核版本: 4.2.0
- 运行内存: 12G

测试数据如下:

1. 抖音: 冷启动 **705ms**

```
LaunchState: COLD
Activity: com.ss.android.ugc.aweme/.main.MainActivity
TotalTime: 691
WaitTime: 705
```

2. 微信: **688ms**

```
LaunchState: COLD
Activity: com.tencent.mm/.plugin.account.ui.WelcomeActivity
TotalTime: 671
WaitTime: 688
```

3. 京东: **874ms**

```
LaunchState: COLD
Activity: com.jingdong.app.mall/.MainFrameActivity
TotalTime: 865
WaitTime: 874
```

4. 支付宝: **450ms**

```
LaunchState: COLD
Activity: com.eg.android.AlipayGphone/.AlipayLogin
TotalTime: 431
WaitTime: 450
```

5. 快手: 875ms

```
LaunchState: COLD
Activity: com.smile.gifmaker/com.yxcorp.gifshow.HomeActivity
TotalTime: 864
WaitTime: 875
```

6. B站: 1145ms

```
LaunchState: COLD
Activity: tv.danmaku.bili/.MainActivityV2
TotalTime: 1133
WaitTime: 1145
```

7. 小红书: 612ms

```
LaunchState: COLD
Activity: com.xingin.xhs/.index.v2.IndexActivityV2
TotalTime: 588
WaitTime: 612
```

结合视频可以看出，启动速度是非常丝滑的，不存在华为所说的Ability层延迟高问题。

结果分析：鸿蒙**4.2**还是使用安卓的框架层，鸿蒙**5**,鸿蒙**6**使用华为自研的框架层，华为框架层设计的太差了，性能低下。

后续调研

思路一：升级到鸿蒙5或鸿蒙6进行数据分析

目前的测试机因为启用了Root权限，不允许升级(纯血鸿蒙使用混合内核模型，允许受控应用直接访问内核数据，因此对安全管理严格)

跟鲁博讨论后，准备购买符合条件的测试机: **8GB** 内存低端机

但是由于鸿蒙系统内存占用高达 **3-4GB**，用户体验差，目前的低端机型，如畅想，**nove 12-14** 等都回退系统版本到 **4.2**

只有高端旗舰手机搭载了纯血鸿蒙(软件不够，硬件凑，割有钱人)

思路二：调研华为框架层代码：

开源仓库链接：[ability_ability_runtime: 元能力子系统实现对Ability的运行及生命周期进行统一的调度和管理，应用进程能够支撑多个Ability，Ability具有跨应用进程间和同一进程内调用的能力。 - Gitee.com](#)

由于只涉及到**Ability**模块，所以性能测试依赖官方提供的样例和**benchmark**存在如下问题：

1. 没有完善负载进行性能测试分析：目前官方仓库只有一个单页面的APP，跟实际应用相差甚远，没法进行性能分析
2. 开源鸿蒙整个项目依赖华为自研工具链，需要使用开源鸿蒙编译工具：[build/docs at master · openharmony/build](#)。工具链无法在实验室机器上构建，目前只支持Ubuntu18.04和Ubuntu20.04

问题

为什么周游老师实验室的同学只需要修改open-harmony

无论如何，鸿蒙系统都建立在Linux之上，鸿蒙系统依赖Linux进行文件管理，只要不修改Linux文件系统的接口，只是优化性能，鸿蒙系统就可以享受到。只需要将新的Linux镜像刷到机器里去就行。

冷启动优化的瓶颈在于框架层，修改Linux内核对于性能提升不明显（内核不是瓶颈，也无法直接定位到内核）

意见与建议

1. 系统冷启动，应用后台保活等都跟鸿蒙的框架层严格绑定，需要深入嵌入华为生态(以后就跟华为绑定了)；华为提供的模块代码没有合格的性能测试集，即使改了这一块也没法实际分析；只有华为的官方文档可以看，AI也帮不上忙，文档很多时候落后于API；
2. 最好是能够拿到和Linux内核强相关的任务：如文件系统等，这是通用的，而且易于修改，有完善的性能测试