

移动应用开发（二）

张凯斌

福建农林大学
资源与环境学院

2025.2.28

引言

□ 对教科书的说明

- 计算机作为一个专业
 - ✓ 鲜活的
 - ✓ 动态的
 - ✓ 充满活力与希望
- 课程会以教课书为基础更深入、更灵活、更贴近生活

□ 本次课的两个目标

- 介绍鸿蒙HormanyOS操作系统
- 基于鸿蒙的移动应用开发基础

历史背景

- 2019年8月9日华为开发者大会
 - 发布HarmonyOS 3.1 开发者尝鲜版



是什么？ (1/2)

□ HarmonyOS是一个全端、全场景、分布式操作系统

- 全端：多种终端形态
 - ✓ PC、笔记本、平板、智能穿戴
- 全场景：多种使用场景
 - ✓ 社交通信、移动办公、媒体娱乐、运动健康
- 分布式技术
 - ✓ 整合各种终端，形成一个虚拟的统一整体—超级终端

□ 尤其是物联网

- 当前物联网受硬件资源限制
 - ✓ 计算能力弱
 - ✓ 存储空间小
- 各种软件协议和其他计算机类型大相径庭

是什么？ (2/2)

- ❑ 对消费者而言
 - 从根本上解决了在不同场景中使用不同终端体验割裂的问题
- ❑ 对应用开发者而言
 - 超级终端屏蔽了不同硬件间的差异
 - 开发者能够专注业务逻辑
- ❑ 对设备开发者而言
 - HarmonyOS是组件化设计
 - 开发时按需调用HarmonyOS提供的各种组件

- ❑ 全新的全场景设备体验，正深入改变消费者的使用习惯
 - 应用开发者面临设备底座从手机单设备到全场景多设备的转变
- ❑ 不同设备类型意味着
 - 传感器能力、硬件能力、屏幕尺寸、操作系统和开发语言
 - 还意味着差异化的交互方式
- ❑ 具体的当前移动应用开发中遇到的主要挑战包括
 - 针对不同设备上的不同操作系统，重复开发，维护多套版本
 - 多种语言栈，对人员技能要求高
 - 多种开发框架，不同的编程范式
 - 命令式编程，需关注细节，变更频繁，维护成本高

鸿蒙生态核心理念 (1/4)

- 鸿蒙系统结合移动生态发展的趋势，提出了三大技术理念
 - 一次开发，多端部署
 - 可分可合，自由流转
 - 统一生态，原生智能

鸿蒙生态核心理念 (2/4)

□ 一次开发，多端部署



鸿蒙生态核心理念 (3/4)

□ 可分可合，自由流转



鸿蒙生态核心理念 (4/4)

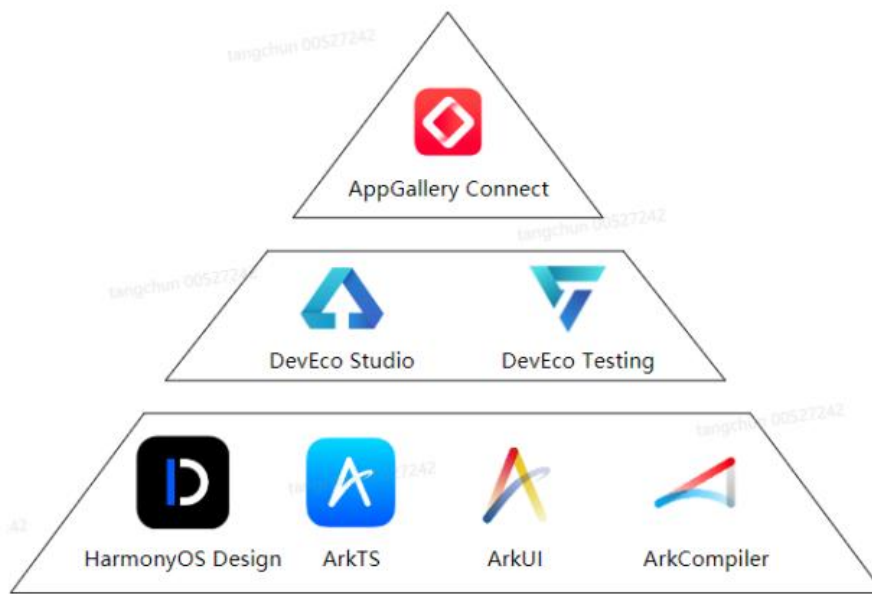
□ 统一生态，原生智能



鸿蒙应用开发生态链

□ 完整的应用开发生态链

- 鸿蒙开发套件包含设计、开发、测试、运维套件以及 OS 开放能力集
- 通过鸿蒙开发套件，开发者可以高效开发鸿蒙生态应用、元服务

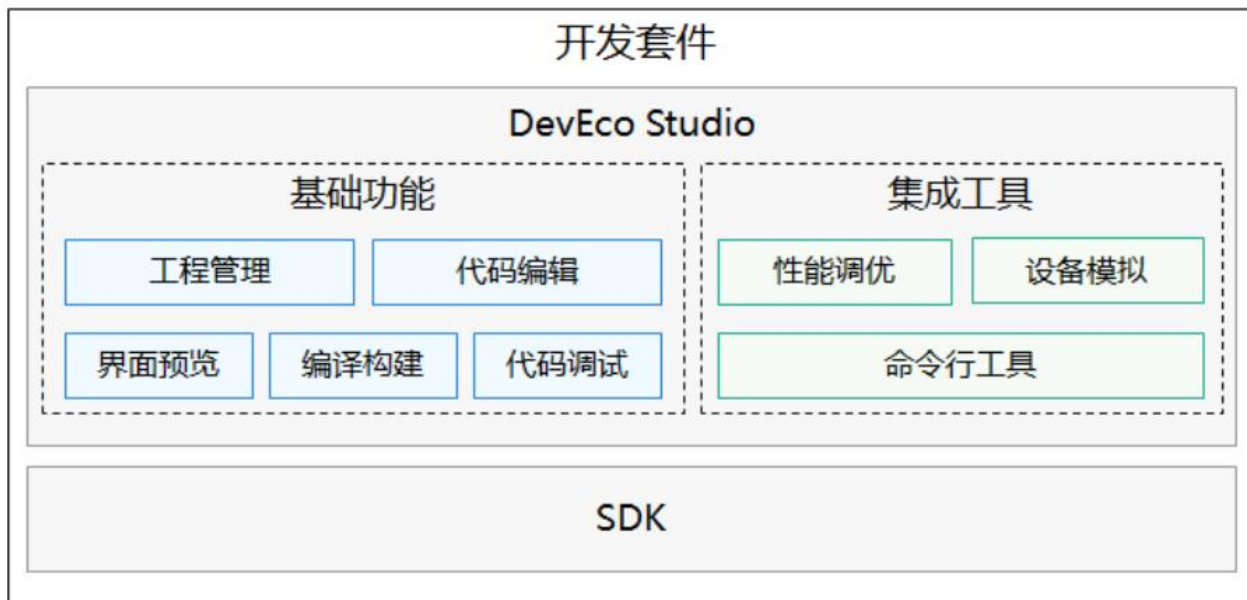


□ HarmonyOS Design提供全栈式设计

- 设计规范
 - ✓ 设计理念、人因研究、应用架构、人机交互、视觉风格、动效、音效、振动、多态控件、界面用语、全球化、无障碍、隐私设计等
- 设计资源
 - ✓ HarmonyOS 字体、HarmonyOS Symbol 和 HarmonyOS 音效库
- 设计工具
 - ✓ 在线设计工具、在线组件库、在线样式库、Symbol 插件、主题换肤插件、规范检查插件

鸿蒙开发套件 (1/2)

- ❑ 开发者在应用开发过程中使用到的产品集合
 - 包含 HUAWEI DevEco Studio 以及HUAWEI DevEco Studio 集成的性能调优、设备模拟、命令行工具和 SDK



鸿蒙开发套件 (2/2)



❑ HUAWEI DevEco Studio

- 鸿蒙生态应用、元服务开发配套的集成开发环境 (IDE)
 - ✓ 提供了工程管理、代码编辑、界面预览、编译构建、代码调试等基础功能
 - ✓ 同时还集成了性能调优工具、设备模拟工具、命令行工具等帮助开发者解决特定领域的问题
- SDK
 - ✓ 集成在 HUAWEI DevEco Studio 中
 - ✓ 包含开发者可以使用的 API 定义和调试编译等基础的工具链

其他鸿蒙套件

- ❑ 鸿蒙测试套件
- ❑ 鸿蒙运维套件



OS开发能力集 (1/6)

- ❑ OS 开放能力集通过 SDK 的形式对开发者呈现，提供应用开发所需的一系列系统开放能力
 - 包括 UI 框架、Ability、分布式服务、安全、方舟编译器、应用服务、基础软件服务、系统服务等
- ❑ ArkUI框架
 - ArkUI 是一套声明式开发框架，它具备简洁自然的 UI 信息语法、丰富的 UI 组件、多维状态管理，以及实时多维度预览等能力
- ❑ Ability
 - 应用所具备能力的抽象，是应用程序的基本组成部分
 - ✓ 主要包括组件生命周期回调、系统环境变化通知、应用跳转、万能卡片开发等能力

OS开发能力集 (2/6)



□ 分布式服务

- 由于具体场景的差异，以及技术演进、生态构建等各种复杂因素的存在，多设备之间的通信方式各不相同
- 分布式软总线
 - ✓ 支持多设备间协同实现异构组网，对开发者屏蔽不同通信介质和通信协议带来的差异
 - ✓ 提供高吞吐、低时延、高可靠、安全可信的通信通道，克服无线通信不可靠、不稳定的挑战

OS开发能力集 (3/6)



□ 分布式服务 (contd.)

- 分布式文件：基于分布式软总线，为应用跨设备文件相互访问提供完整解决方案
 - ✓ 分布式文件系统是一个直接构建在内核态，无中心、高性能的文件系统
- 分布式数据库：基于分布式软总线实现数据的分布式管理
 - ✓ 用户数据不再与单一物理设备绑定，跨设备的数据处理如同本地数据处理一样方便快捷

OS开发能力集 (4/6)

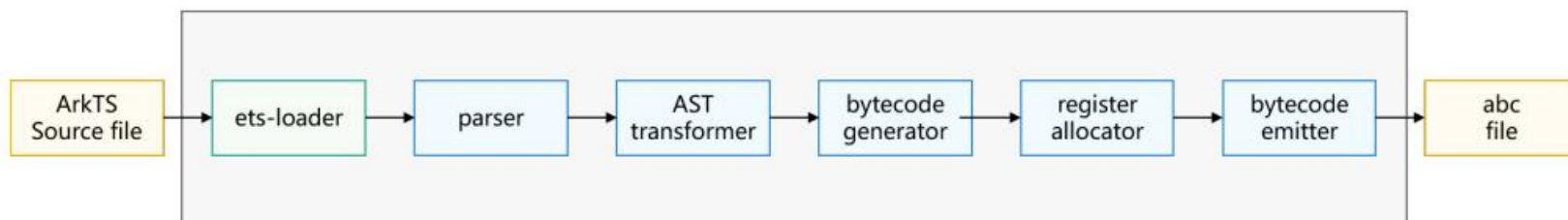
□ 分布式服务 (contd.)

- 分布式硬件：打破单一设备的硬件边界，是软件定义各种新产品形态和体验的“新硬件”所需的关键技术
 - ✓ 分布式硬件能够将硬件设备化整为零，形成“超级终端”
 - ✓ 硬件资源池，供多个设备共享使用，真正达到软件定义硬件、设备间实现系统级融合并灵活按需适应不同场景的目的
- 融合感知：构筑在传感器技术和多模感知技术上的系统开放能力
 - ✓ 提供了对设备、环境、用户等的各类状态的感知能力
 - 例如设备的摆放姿态、操作行为、多设备间的空间拓扑关系、用户的移动状态、位置信息，甚至是用户的情绪等

OS开发能力集 (5/6)

❑ 方舟编译器 (ArkCompiler)

- 支持多种编程语言、多种芯片平台联合编译、运行而设计的统一编译运行时平台
 - ✓ 支持包括动态类型和静态类型语言在内的多种编程语言，如 ArkTS、TS、JS
- 方舟编译器主要分成两个部分，编译工具链与运行时
 - ✓ 编译工具链以ArkTS/TS/JS 源码作为输入，将其编译生成为ABC (ArkCompiler Bytecode, 即方舟字节码) 文件
 - ✓ 运行时直接运行字节码文件，实现对应语言规范的语义逻辑



OS开发能力集 (6/6)



□ 基础软件服务

- 鸿蒙系统为开发者提供了通用的基础软件服务，包括多媒体、通信、图形、文件存储、升级、无障碍等

□ 系统服务

- 系统服务通过屏蔽硬件厂家接口差异，为应用提供统一的接口规范，从而使开发者便捷地控制外设

□ 应用服务

- 应用服务是为鸿蒙生态应用开发者提供的商业化相关服务
 - ✓ 如鸿蒙生态应用的账号、消息通知、支付、广告、地图等各种业务快速开通和实现

典型开发场景 (1/11)

- ❑ 基于鸿蒙系统原生开发框架，开发者可以根据自己实际的业务情况进行选择
 - 独立开发一个应用
 - 独立开发一个元服务
 - 同时开发应用和元服务
- ❑ 开发者可以选择开发简单、场景聚焦的元服务，渐进迭代演进，按需组合元服务成为一个复杂的应用
 - 对于大型游戏类应用，可以直接开发鸿蒙生态应用
 - 针对智慧屏、车机、手表等设备，可以考虑开发元服务并提供万能卡片，让信息更加直接的呈现给用户

典型开发场景 (2/11)



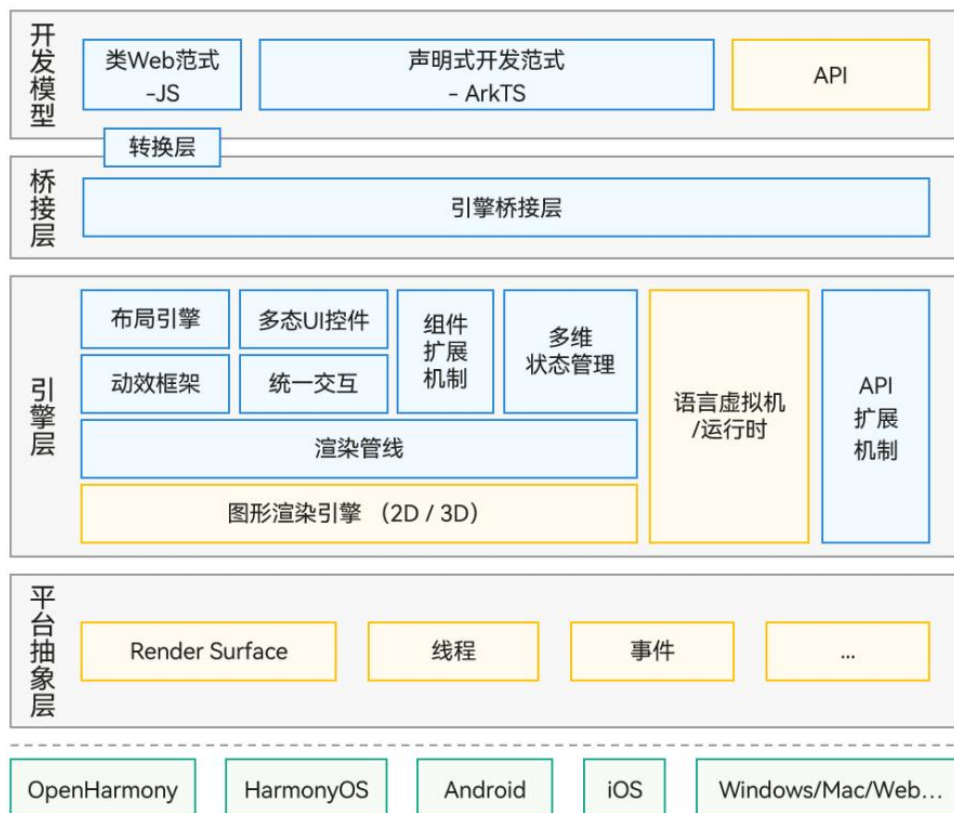
□ ArkTS 语言

- ArkTS 是鸿蒙生态应用的开发语言，使用.ets 作为 ArkTS 语言源码文件后缀
- 提供了声明式 UI、状态管理等相应的能力，让开发者以更简洁、更自然的方式开发高性能应用
- ArkTS 的重要特性之一是静态类型
 - ✓ 相比于 TS 只在编译时进行类型检查，ArkTS 将编译时所确定的类型应用到运行性能优化中

典型开发场景 (3/11)

□ ArkUI 框架

- ArkUI 是鸿蒙生态原生的 UI 开发框架



典型开发场景 (4/11)

□ ArkUI 框架 (contd.)

- ArkUI 框架提供给开发者两种开发方式
 - ✓ 基于 ArkTS 的声明式开发范式
 - 声明式开发范式更加简洁高效
 - ✓ 基于 JS 扩展的类 Web 开发范式
 - 类 Web 开发范式对 Web 及前端开发者更友好

□ 声明式开发范式

- 声明式开发范式的核心思想是数据驱动 UI 变化，通过提供的状态进行数据管理，
 - ✓ 这里状态管理指的是，管理数据发生变化时，框架能自动更新这些数据关联的最小范围的 UI

典型开发场景 (5/11)

□ 声明式开发范式 (contd.)

● 布局

- ✓ 布局指用特定的组件或者属性来管理用户页面所放置 UI 组件的大小和位置

● 组件化

- ✓ 组件是 ArkUI 框架中的基础显示单元，一切 UI 显示的内容都是组件
- ✓ ArkUI 框架提供多种开箱即用的 UI 组件，如文本显示、图片显示、按键等，并提供了面向多种设备形态的多态 UI 能力

典型开发场景 (6/11)

□ 声明式开发范式 (contd.)

● 装饰器

- ✓ 自定义组件的场景中，通常会遇到需要动态传入不同的 UI 元素的情况
- ✓ 为满足该场景 ArkUI 框架同时提供了动态构建 UI 元素的能力

● 动效

- ✓ 声明式的动画是由数据变化驱动动画启动，而不再是直接控制动画的播放

● 事件交互

- ✓ ArkUI 框架提供了很多交互事件，这些事件提供了不同的信息用于处理相关程序交互逻辑

典型开发场景 (7/11)

□ 声明式开发范式 (contd.)

- 绘制能力

- ✓ ArkUI 框架提供两种 2D 自定义绘制能力

- 混合开发

- ✓ 应用的场景是多样的，部分场景直接采用 UI 组件组合无法满足诉求

- 例如游戏、地图这种需要依赖 C++ SDK 进行独立渲染

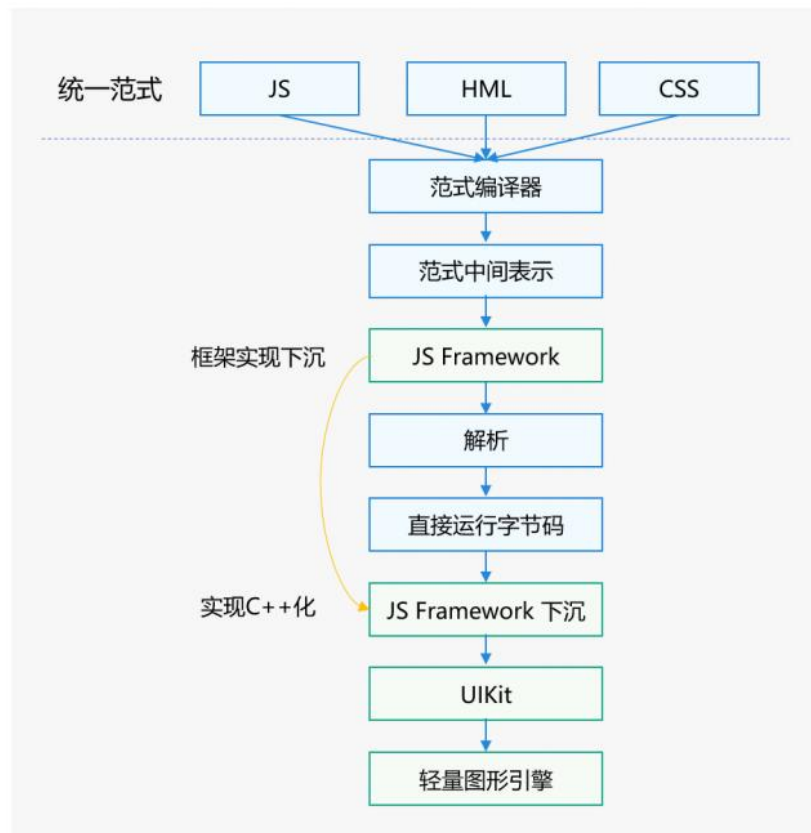
- 又或者开发相机、视频播放器这种需要独立纹理填充的场景

- ✓ 因此需要框架提供一种能够在 C++ 侧进行自定义绘制的组件

典型开发场景 (8/11)

□ 类 Web 开发范式

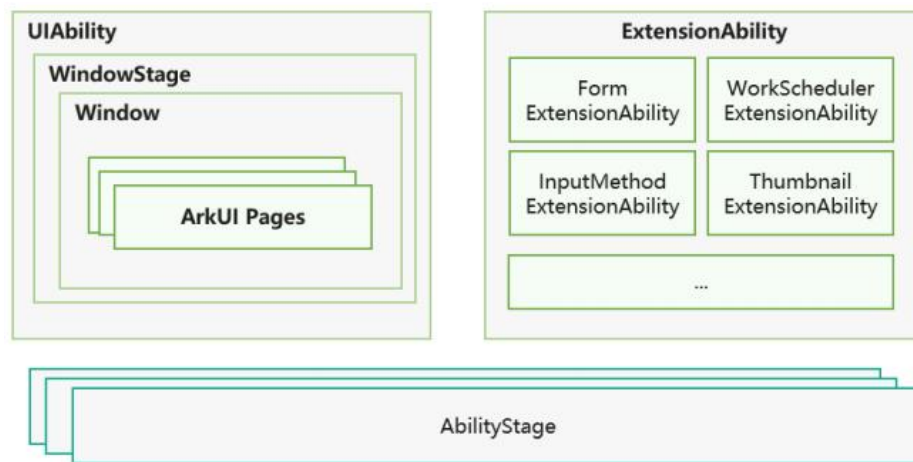
- 使用 HML 标签文件进行布局搭建, 使用 CSS 文件进行样式描述, 使用 JS 文件进行逻辑处理
- UI 组件与数据之间通过单向数据绑定的方式建立关联
 - ✓ 当数据发生变化时, UI 界面自动触发更新
 - ✓ 此种开发范式, 对 Web 前端开发者更为友好



典型开发场景 (9/11)

□ 应用程序框架

- 应用程序框架定义了应用程序的模型与结构
 - ✓ 鸿蒙系统上的应用模型称之为 “Stage 模型”
 - ✓ 应用程序框架定义了应用的全生命周期
 - ✓ 鸿蒙系统支持1+8+N 多设备的统一操作
 - ✓ Ability 是应用程序框架中最基本的抽象单位，是能够完成独立功能的应用组件



典型开发场景 (10/11)

□ HarmonyOS SDK

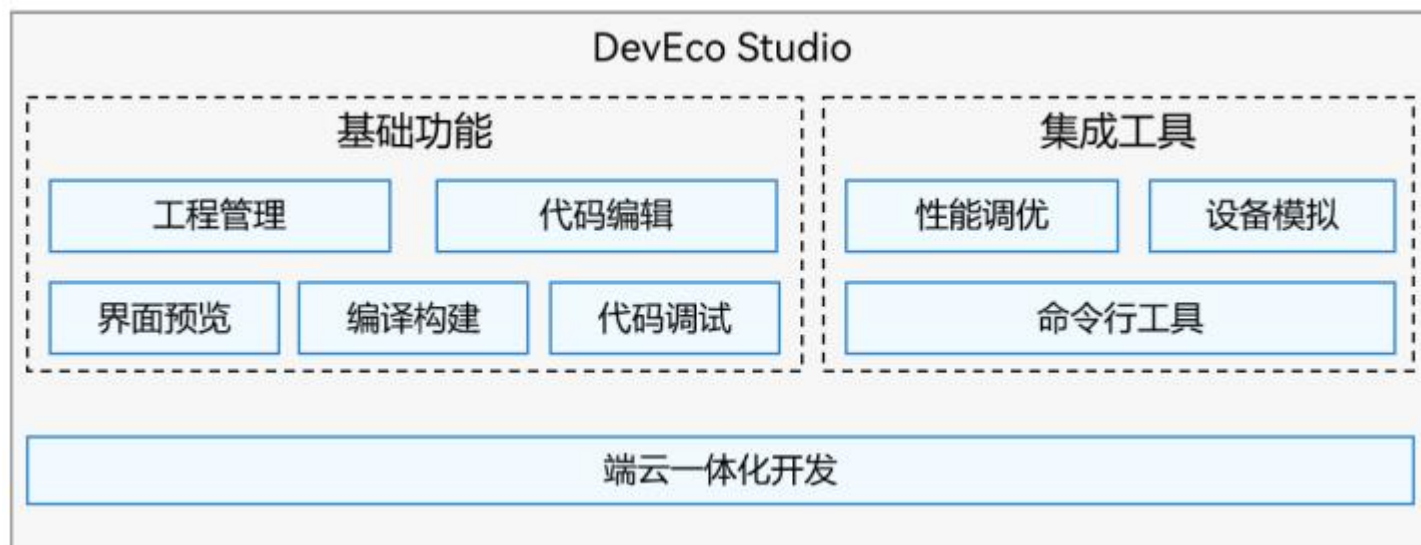
- HarmonyOS SDK 是为了支撑 Harmony OS 应用开发而提供的一系列开放能力
 - ✓ 包括API 接口定义, 资料文档, 样例代码, 基础工具包等。
- HarmonyOS SDK 是鸿蒙生态建设的重要组成部分



典型开发场景 (11/11)

□ 集成开发环境

- HUAWEI DevEco Studio 是面向鸿蒙生态的集成开发环境，提供了一站式的鸿蒙生态应用、元服务开发能力



Q&A

Good Luck!