

本章将详细讲解鸿蒙 ACE Java UI 框架的使用,读者通过本章的学习将对鸿蒙 Java UI 框架有深入的了解。ACE Java UI 框架是鸿蒙 ACE(Ability Cross-platform Environment,鸿蒙 面向元能力(Ability)的跨平台开发运行环境)的重要组成部分,用来开发鸿蒙富设备应用 程序。

3.1 ACE 运行时简介

ACE 全称为 Ability Cross-platform Environment,是鸿蒙面向元能力(Ability)开发的 跨平台运行环境,如图 3-1 所示。



图 3-1 鸿蒙 ACE Run-time 架构图

ACE 是为华为鸿蒙操作系统定制设计的一套针对全场景环境的跨平台应用开发框架。 ACE 支持目前两种主流语言: Java 和 JavaScript。未来将支持华为自己的开发语言 "仓颉"。

3.1.1 ACE 针对全场景开发

鸿蒙应用开发针对的是1+8+N的全场景设备:1指的是手机;8代表车机、音箱、耳机、手表/手环、平板、智慧屏、PC、AR/VR;N泛指其他IoT设备。

这里可以把这些 IoT 终端设备分为富设备和轻设备,对于富设备开发鸿蒙提供了基于 Java、C++语言的开发框架,对于一些内存有限,功能有限的轻设备则提供了基于 JavaScript 的开发框架,这样可以让一些低端硬件配置的 IoT 设备运行鸿蒙应用程序。

3.1.2 ACE 支持的两种 UI 框架

鸿蒙跨平台运行时(ACE)为 Java 开发者和 JavaScript 开发者提供了一站式的开发环境。

1. 基于 Java 的应用开发框架(ACE Java UI)

鸿蒙的应用开发可以基于 Java 语言+XML 的方式进行开发,此方式和安卓类似,如果 开发者熟悉 Java 语言或者 Android 开发,那么使用此方式可以很快进行应用开发。ACE Java UI 框架的架构图如图 3-2 所示。



图 3-2 鸿蒙 ACE Java 总体架构图

2. 基于 JavaScript 的应用开发框架(ACE JavaScript UI)

鸿蒙开发同时还提供了 JavaScript 形式的开发。JavaScript 的语法规则基本遵循 ES6 的语法规范。如果是前端开发人员,则可以使用 HTML+CSS+JavaScript 的形式进行应用的开发,但是对于后端一些服务能力,目前只能使用 Java 语言进行开发,所以对于 JavaScript 的开发者,依然需要 Java 语言的辅助。ACE JavaScript UI 框架的架构图如 图 3-3 所示。



图 3-3 鸿蒙 ACE JavaScript 总体架构图

3.2 ACE开发中的核心概念

在学习 ACE 开发框架前,需要了解 ACE 中的一些核心概念。

3.2.1 Ability 和 Slice

Ability 是 HarmonyOS 开发的核心,首先了解什么是 Ability,我们用生活中的例子来做一个类比,Ability 就好像是一个画板,初始什么都没有,一片空白,我们可以找一张画纸进行绘画,画完的画纸夹在画板上,画板上就会有我们创作的图画了。

HarmonyOS中的 Ability 其实就相当于画板,它用来承载画纸,那么画纸又是什么? 标题中还有一个 Slice,它就相当于画纸,我们写应用的 UI,写完后通过 Slice 加载布局,就相 当于在画纸上绘画,完成后把画纸放在画板上进行展示。

将对应的概念引申到程序中,我们有一个 Ability(画板)用来向用户展示新闻,同时这 个 Ability 拥有两个 Slice(画纸),一个用来展示新闻标题,另一个用来展示新闻详情。那么 我们就需要在这两个 Slice 中分别画上新闻的标题和详情,然后通过单击或其他交互方式来 切换两个 Slice,相当于给画板换另一张画纸。

3.2.2 Ability 分类

有了上述概念,我们基本知道 Ability 能做些什么了,但是还要具体了解一下这个核心

组件,在 HarmonyOS中,Ability可以分为两类,一类叫FA(Feature Ability),另一类叫作 PA(Particle Ability),这两类 Ability 最直观的区别是FA 有界面,可以和用户交互,而PA 无界面,它提供一些应用运行需要的能力,例如读取数据。这两类 Ability 在创建的时候都 有相应的模板。

Ability 的模板种类。

(1) 对于 FA 类型的 Ability, Page 类型的模板是其唯一的模板类型,这种类型的模板 提供了和用户交互的能力。

(2) 对于 PA 类型的 Ability, 它有两类模板。

- Service 模板:用于提供后台运行任务的能力。
- Data 模板:用于对外部提供统一的数据访问抽象。

3.2.3 鸿蒙应用包结构

鸿蒙应用软件包以 App Pack(Application Package)形式发布,它是由一个或多个 HAP (HarmonyOS Ability Package)及描述每个 HAP 属性的 pack. info 组成。HAP 是 Ability 的部署包,HarmonyOS 应用代码围绕 Ability 组件展开。一个 HAP 是由代码、资源、第三 方库及应用配置文件组成的模块包,可分为 entry 和 feature 两种模块类型,如图 3-4 所示。



图 3-4 鸿蒙应用包结构

entry:应用的主模块。在一个 App 中,对于同一设备类型必须有且只有一个 entry 类型的 HAP,可独立安装运行。

feature: 应用的动态特性模块。一个 App 可以包含一个或多个 feature 类型的 HAP, 也可以不含。只有包含 Ability 的 HAP 才能够独立运行。

有了上述概念,3.3节将分析之前创建的项目的结构和项目中配置文件的配置内容具体有什么含义。

3.3 创建一个 ACE Java 项目

在对 HarmonyOS 应用程序有了一个初步认知之后,我们使用 DevEco Studio 来创建 一个项目,把项目运行起来,先从整体上来了解一下 HarmonyOS 项目的整体结构及开发工 具的基本使用。

环境和工具配置好后,就可以创建一个项目了。先创建一个项目,从整体上了解一下 HarmonyOS应用的整体框架。我们在一个布局里放置一个文本框,用于显示一个数字,再 添加一个按钮,每次单击按钮让文本框中的数字加1。通过这样一个小程序简单演示工具 的使用和项目的基本框架。

3.3.1 新建 ACE Java 项目

选择 File→New→New Project 进行项目创建,会弹出如图 3-5 所示的窗口。

Choose your	ability temp	late					
			Ģ	Q	¢	0	
Phone	Tablet	Car	TV	Wearable	Lite Wearable	Smart Vision	
Empty Feature	CS Ability(JS)	Empty Feature A	bility(Java)	Business Card A	bility(Java)	Category List Ability(Java)	
		•				-	

图 3-5 新建 ACE Java 项目

创建项目窗口可分为两块,其中 Device 表示目前支持的设备。设备列表中从左到右依 次为手机、平板、车机、电视机、智慧屏、穿戴设备、轻型穿戴设备等。

上边是新建项目时供选择的模板,因为笔者使用 Java 开发,所以选择第二个(Empty Feature Ability Java),单击 Next 按钮进入下个页面,如图 3-6 所示。

配置完项目名、包名、使用的 SDK 版本及项目的保存路径后,单击 Finish 按钮即可,创 建完成后项目会自动构建。构建成功后项目整体结构如图 3-7 所示。

视频讲解

Project Name first_demo Package Name com.charjedu.first_demo Save Location D:\work2020'\29-HarmonyOS\book_kode\first_demo	
first_demo Package Name com.chaijedu.first_demo Save Location D:work2020/29-HarmonyOS\book_book_book_book_book_book_book_book	
Package Name com.charjedu.first_demo Save Location D\work2020/29-HarmonyOS\book_bode\first_demo	
com.charjedu.first_demo Save Location D\work2020/29-HarmonyOS\book_bode\first_demo	
Save Location D:\work2020/29-HarmonyOS:book_bode\first_demo	
D:\work2020\29-HarmonyOS\book_code\fint_demo	
	Đ
Compatible API Version	
SDK: API Version 3	*

图 3-6 选择项目模板



图 3-7 ACE Java 项目目录结构

先完成我们的功能,后续再了解目录及其作用。首先打开 resources/base/layout 目录中自动生成的布局文件,然后单击工具右侧的预览,如图 3-8 所示。

Phone (1080 x 2250) Hello World	entry : ability_n	nain.xml	Q ⊕
Hello World		Phone (1080 x 2250)	© Preview
		Hello World	er

图 3-8 ACE Java 项目预览

这是工具提供的一个预览器,当写 xml 布局的时候可以近乎实时地看到我们所写代码的效果。

3.3.2 编写界面布局

在 ability_main. xml 文件中,添加一个 Text 组件和一个 Button 组件。这里为 Button 组件添加了一个背景文件,通过 Button 属性 background_element 引入。

background_element 的值是 \$ graphic:background_ability_main。 \$ graphic 表示引用 graphic 文件夹,background_ability_main 是 graphic 文件夹下的 xml 文件,如代码示例 3-1 所示。

代码示例 3-1 编写界面布局: first_demo/entry/layout/ability_main.xml

```
<?xml version = "1.0" encoding = "utf - 8"?>
< DirectionalLayout
    xmlns:ohos = "http://schemas.huawei.com/res/ohos"
    ohos:height = "match_parent"
```

```
ohos:width = "match_parent"
    ohos:orientation = "vertical">
< Text
        ohos:id = " $ + id:text"
        ohos:height = "match_content"
        ohos:width = "match_content"
        ohos:layout_alignment = "horizontal_center"
        ohos:text = "1"
        ohos:text_size = "60fp"
        />
< Button
        ohos:height = "match_content"
        ohos:width = "match content"
        ohos:text_size = "18fp"
        ohos:id = " $ + id:btn_add"
        ohos:layout_alignment = "horizontal_center"
        ohos:background_element = " $ graphic:background_ability_main"
        ohos:text color = "white"
        ohos:padding = "15vp"
        ohos:text="单击加 1"/>
</DirectionalLayout >
```

此外还需要修改 graphic 目录下 background_ability_main. xml 文件,将背景设置为 10 像素的圆角,将填充色设置为 # 007CFD,如代码示例 3-2 所示。

代码示例 3-2 设置背景: first_demo/entry/layout/background_ability_main.xml

```
<?xml version = "1.0" encoding = "utf - 8"?>
< shape xmlns:ohos = "http://schemas.huawei.com/res/ohos"
ohos:shape = "rectangle">
< corners
ohos:radius = "10"/>
< solid
ohos:color = " # 007CFD"/>
</shape >
```

此时在预览器中看到的效果如图 3-9 所示。

3.3.3 编写界面逻辑代码

接下来实现"单击加1"功能,打开由项目自动创建的 MainAbilitySlice 文件,如代码示例 3-3 所示。这里通过 Button 事件,每单击一次,设置"count++;"。



图 3-9 预览效果图

代码示例 3-3 设置背景: first_demo/entry/java/MainAbilitySlice. java

```
public class MainAbilitySlice extends AbilitySlice {
   private int count = 1;
@Override
public void onStart(Intent intent) {
        super.onStart(intent);
        super.setUIContent(ResourceTable.Layout ability main);
        Text text = (Text) findComponentById(ResourceTable.Id_text);
        Button button = (Button) findComponentById(ResourceTable.Id_btn_add);
        button.setClickedListener(new Component.ClickedListener() {
            @Override
            public void onClick(Component component) {
                count++;
                 text.setText(count + "");
        });
    }
    @Override
```

```
public void onActive() {
    super.onActive();
}
@Override
public void onForeground(Intent intent) {
    super.onForeground(intent);
}
}
```

3.3.4 通过模拟器预览效果

到此为止,功能已经开发完毕,单击工具上方菜单栏的 Tools 下边的 HVD Manager,弹出华为账号授权界面,如图 3-10 所示。

⊦ £	HUAWEI DevEco Studio想要访问您的 华为账号
¢)
	初接账号
ž	这样即可让HUAWEI DevEco Studio:
9	查看和管理您在华为账号平台中的数据
51	论开HUAWEI DevEco Studio访问您的账号吗?
の思	唐 "允许",即表示您允许此应用模模其服务条款和隐私政策使用您的信 。
	取消 允许

图 3-10 华为账号授权

这里需要登录已注册的华为账号并且授权,单击"允许"按钮,跳转到网页授权,授权完成后,会弹出下面的界面,如图 3-11 所示。

这里可以选择对应的远程设备,因为我们创建项目时选的是手机类型的项目,因此这里选手机 P40,单击后边的蓝色箭头,之后在工具右侧的预览区会变成所选择的手机 P40。

选择手机 P40 模拟器,如图 3-12 所示,等待模拟器启动完成后,就可以单击工具右上方的按钮启动项目了。

单击工具右上方的按钮启动项目,如图 3-13 所示。