

撰写人：赵春春

日期： 2021-4-13

**HiARSiri**

**概要设计**

**HiARSiri概要设计**

目录

[HiARSiri概要设计 1](#_Toc24232)

[1. 引言 1](#_Toc11928)

[1.1 编写目的 1](#_Toc23665)

[1.2 项目范围 1](#_Toc4471)

[1.3 项目背景 1](#_Toc24357)

[1.4 缩略语 1](#_Toc30341)

[2. 软件概述 2](#_Toc6936)

[2.1 HiARSiri概述 2](#_Toc27004)

[2.2 功能1：全局模式下的语音交互操作 2](#_Toc5484)

[2.3 功能2：支持客户定制个别简单语音操作命令 2](#_Toc10744)

[2.4 功能3：对于其他普通的控件进行序号辅助识别 3](#_Toc18987)

[2.5 功能4：对于特殊应用和特殊控件可以直接语音操作，不需要辅助识别 3](#_Toc6967)

[3. 需求分析 4](#_Toc29077)

[3.1 全局模式下的语音命令 4](#_Toc13990)

[3.2 支持客户定制简单的识别词 4](#_Toc24800)

[3.3 对于普通控件的辅助识别 4](#_Toc16412)

[3.4 对于HiAR的特定应用 5](#_Toc27163)

[4. 总体结构设计 6](#_Toc12469)

[4.1 HiARSiriService 6](#_Toc31764)

[4.2 HiARSiriActivity 6](#_Toc13730)

[5. 软件流程设计 7](#_Toc2665)

[5.1 HiARSiriService初始化流程 7](#_Toc19057)

[5.2 HiARSiri响应Accessibility Event流程 8](#_Toc18226)

[5.3 HiARSiri响应语音识别service的callback的流程 9](#_Toc29335)

[6. 软件类图和接口 10](#_Toc17122)

[6.1 整个项目的类图 10](#_Toc29760)

[6.2 关键类中的主要接口 10](#_Toc27733)

# **引言**

#### 编写目的

本文档是Bertha项目上HiARSiri软件功能的概要设计，主要是介绍HiARSiri的软件功能和软件架构以及软件中各个模块的接口设计。

#### **项目范围**

本文档适用于Bertha项目，对于后续项目的适用度，需要再次进行项目评估。

#### **项目背景**

HiARSiri模块被用于整个Bertha项目上的智能设备，后期也可以用于其他项目上基于Android开发的智能设备。该模块需要HiAR的语音识别模块的支持，所以其他Android智能设备必须也要集成了HiAR的语音识别模块。

#### **缩略语**

|  |  |  |
| --- | --- | --- |
| **缩略语** | **英文全名** | **中文解释** |
|  |  |  |
|  |  |  |
|  |  |  |

# **软件概述**

#### **HiARSiri概述**

HiARSiri是亮风台公司在Bertha项目上新开发的软件，该软件主要是为了解放AR眼镜使用者，方便使用者更快，更便捷的使用AR眼镜，也是在AR眼镜上提供了一个全新的交互方式。HiARSiri提供了全新的语音操作功能，可以使用“全局命令 + 自定义命令 + 序号辅助识别”的模式给用户提供更细致，更简单的交互方式。

#### **功能1：全局模式下的语音交互操作**

对于一些快捷的操作以及一些使用频繁的操作，HiARSiri提供了在全局任何情况下都可以识别并交互的能力。

例如：

快速设置：在当前页面上层叠加打开快速设置功能，可进行亮度、音量、开关降噪、关闭麦克风等的快速设置；

返回桌面：返回至桌面；

返回上一级：返回至当前页面的上一级页面；

后台应用：显示后台在运行的应用程序；

显示帮助：显示当前页面可用的语音命令词；

#### **功能2：支持客户定制个别简单语音操作命令**

对于一些客户需要定制的特殊语音识别，，HiARSiri提供了在客户可以定制的简单操作。这部分功能目前还没具体定义，后续需要看情况是否支持。

在任何界面都能被识别并响应的语音命令词，比如“呼叫XXX”，其中“呼叫”是固定命令词，后面的“XXX”是自定义命令词，两个组合构成一个完整命令词。

#### **功能3：对于其他普通的控件进行**序号辅助识别

当目前的操作界面中控件比较多，而且不具有通用性的控件，会重新绘制一个overlayer界面，并对每一个控件进行编号，在控件的左上角显示对应的编号，这样可以通过打开/点击第X项的命令对其进行操作。

#### **功能4：对于特殊应用和特殊控件可以直接语音操作，不需要辅助识别**

对于HiAR的一些特定应用，因为有app的支持，所以可以做到对特定的app和特定的控件可以做到直接语音操作，不需要添加辅助识别编号。

# **需求分析**

#### **全局模式下的语音命令**

对于全局模式下的语音命令，一定是固定的简单操作，包括返回桌面，返回上一级，打开帮助等等，这些识别语音可以有两种实现方法：

在每次设置识别关键词的时候都把这些固定的识别命令都设置进去，这样可能会影响识别，因为可能在设置识别关键词的时候把前面识别的冲掉；

在语音识别引擎端提前配置好，这样识别速度会加快；

但是目前使用的是第一种方案，因为这个方案实现比较简单，后面也可以支持客户修改定制；

#### **支持客户定制简单的识别词**

目前这个功能还没有实现，等待后续软件定义正是下来，或者真的需要的时候在进行添加。

#### **对于普通控件的辅助识别**

对于一些普通的应用，因控件的千差万别，所以不能够做到控件的语义识别，只能通过对每个控件进行编号，从而到达辅助识别，可以通过打开/点击第X项的命令对其进行操作。

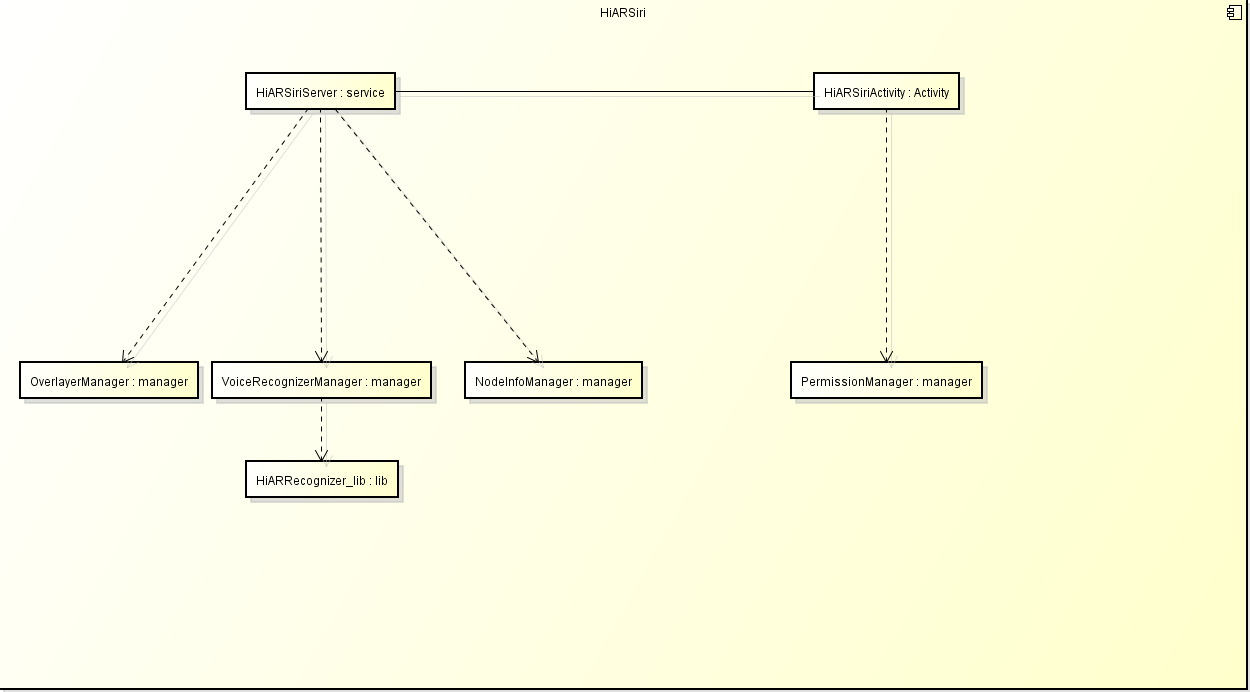


#### **对于HiAR的特定应用**

对应HiAR的一些特定应用，因为有APP的支持，所以可以做到某些特定控件的语义识别，但是因为目前还没有更明确的定义，所以还没有相关功能的实现，这个是后期可以扩展的功能。

# **总体结构设计**

目前HiARSiri的整体架构设计如下图：



HiARSiri主要包括两个部分：HiARSiriService和HiARSiriActivity。

#### **HiARSiriService**

HiARSiriService是HiARSiri的主service，负责界面绘制/语音识别等功能。下面又包含了HiARNodeInfoManager，HiARVoiceManager和HiAROverlayManager三个部分。

HiARNodeInfoManager主要是用来解析手机界面上的Nodeinfo，对Nodeinfo进行分类，并对Node的参数和编号进行重新赋值。

HiARVoiceManager主要用来和语音识别service进行交互，有语音识别的callback，根据callback返回的值，进行对识别的到动作进行更进一步的操作。

HiAROverlayManager是进行Overlayer界面的绘制，根据不同的操作，显示界面上不同的操作，同时也会绘制node上的编号。

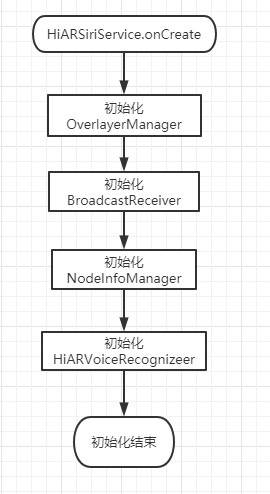
#### **HiARSiriActivity**

HiARSiriActivity主要是HiARSiri启动的activity，为了方便用户关闭和打开HiARSiriService。

HiARSiriActivity下主要包含HiARPermissionManager这个，这个类里的主要功能包含：checkHiARPermission，requestHiARPermission，checkOverlayPermission，requestOverlayPermission。

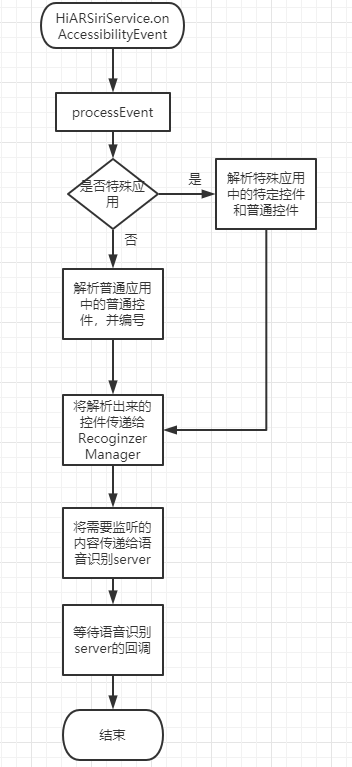
# **软件流程设计**

#### **HiARSiriService初始化流程**



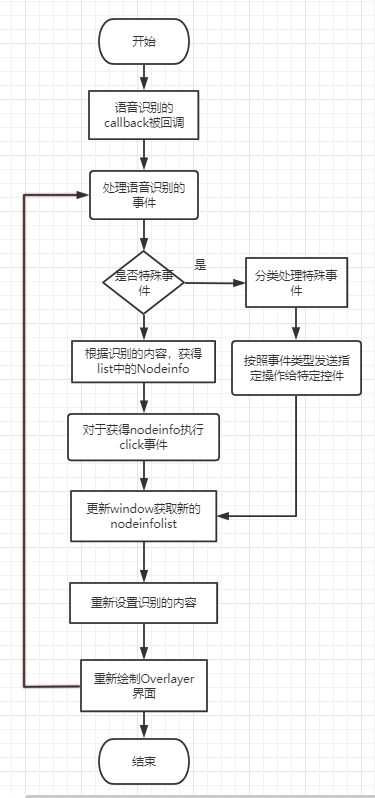
HiARSiriService在被启动后，并被连接上后，会启动初始化流程，会去初始化相关的几个模块，其中包括Overlayermanager，VoiceRecognizerManager等，这个时候会尝试把Overlayer的layout绘制出来。

#### **HiARSiri响应Accessibility Event流程**



1. HiARSiriservice在onAccessibilityEvent中会受到AccessibilityServiceManager发送过来的Event，这里对Event进行过滤，只处理TYPE\_WINDOW\_CONTENT\_CHANGED类型的消息。
2. 判断当前APP是否HiAR的特殊APP
   1. 对于特殊APP，会对特殊的控件，设置特殊的识别语句；
   2. 对于普通的APP，设置识别语句依靠标号；
3. 把解析出来的nodeinfolist传递给RecognizerManager；
4. RecognizerManager把NodeinfoList中每个Node的识别语句以及全局的识别语句传递给语音识别服务；
5. 等待语音识别的callback；

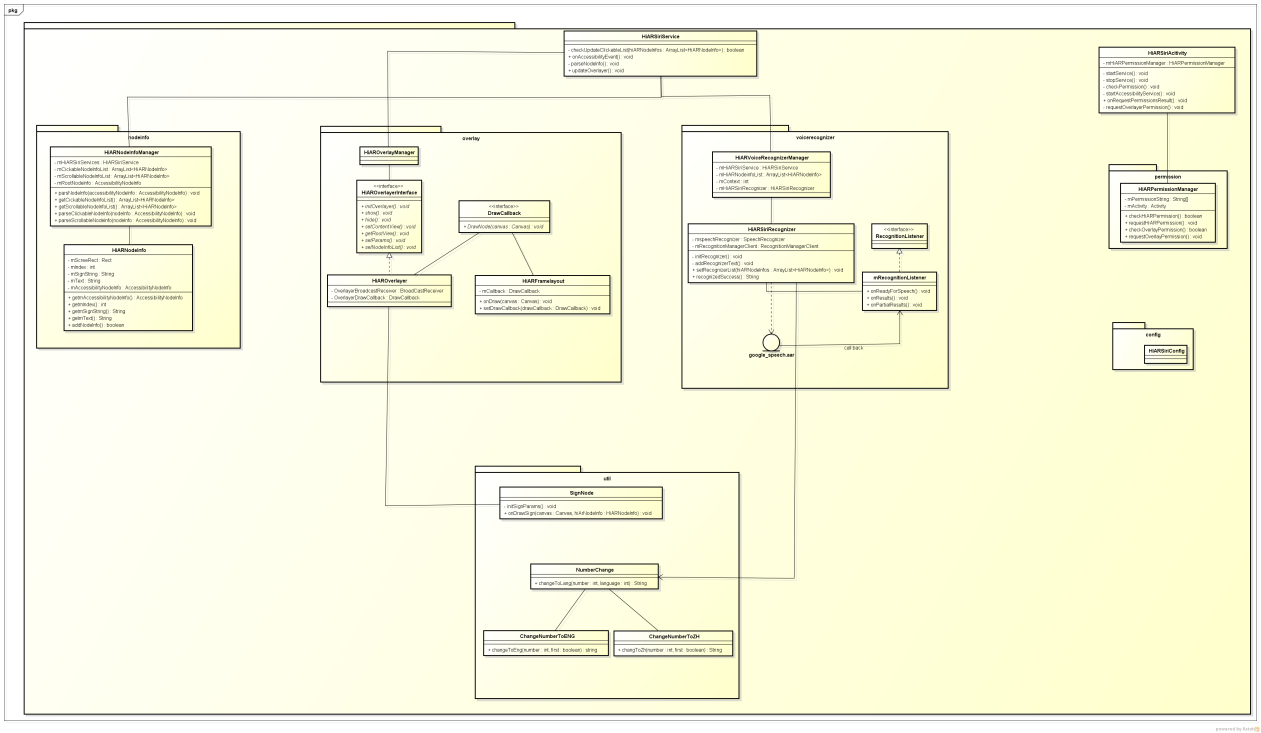
#### **HiARSiri响应语音识别service的callback的流程**



1. RecognizerManager中的callback被语音识别回调后，会通过Result带过的事件，判断该事件是否特殊事件；
   1. 对于特殊事件，按照事件类型发送给指定的控件；
   2. 对于普通的click事件，根据识别的内容，获取对应的nodeinfo，并执行click 事件；
2. 更新window界面，同时获取新的nodeinfolist
3. 重新设置识别的内容；
4. 重新绘制Overlayer的界面；
5. 结束

# **软件类图和接口**

#### **整个项目的类图**

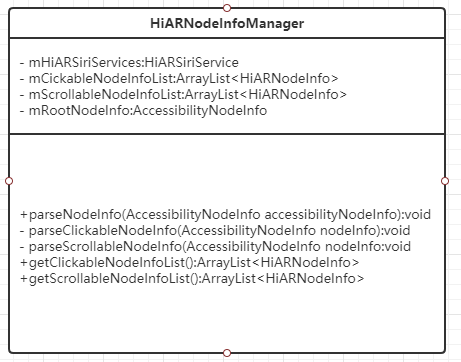


该图是整个项目中的关键类的类图，可以看到主要有五个package：overlay，voicerecognizer，nodeinfo，util，permission。

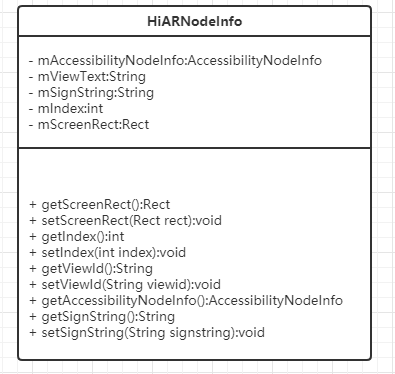
其中overlay中主要涉及到overlaymanager，voicerecognizer中主要是voicerecognizermanager，nodeinfo是主要是nodeinfomanager，util是一些通用的工具类和接口，permission中是permissionmanager。

#### **关键类中的主要接口**

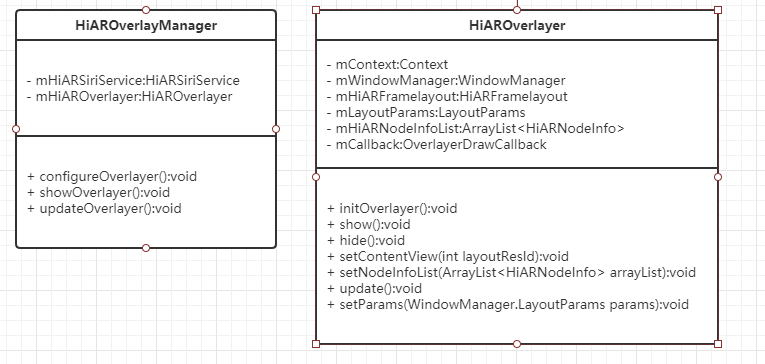
HiARNodeInfoManager的主要变量和函数：



HiARNodeInfo类的主要变量和函数：



HiAROverlayManager和HiAROverlayer类的主要变量和函数：



HiARSiriRecognizer类的主要变量和函数：

