2016

AccessibilityManager

中国信息无障碍产品联盟&信息无障碍研究会　译制

20161111

目录

翻译声明.....1
AccessibilityManager 2
1. 概要.....3
1.1 嵌套类.....3
1.2 公有方法.....3
1.3 继承方法.....5
2.公有方法.....7
2.1 addAccessibilityStateChangeListener.....7
2.2 addTouchExplorationStateChangeListener.....7
2.3 getAccessibilityServiceList.....8
2.4 getEnabledAccessibilityServiceList 8
2.5 getInstalledAccessibilityServiceList.....9
2.6 interrupt.....9
2.7 isEnabled.....10
2.8 isTouchExplorationEnabled.....10
2.9 removeAccessibilityStateChangeListener 10
2.10 removeTouchExplorationStateChangeListener 11
2.11 sendAccessibilityEvent 11

翻译声明

翻译机构：信息无障碍研究会（ARA） 中国信息无障碍产品联盟（CAPA）

译者：刘辉

审阅：李鸿利、刘彪、沈广荣

本文档翻译自谷歌官方文档《AccessibilityManager》。如您对翻译文档内容有异议，请将原文文档做为主要参考，原文版权由 Google 持有并保留。

AccessibilityManager

添加于 API 级别 4。

public final class AccessibilityManager

extends object

java.lang.Object

↳ android.view.accessibility.AccessibilityManager

系统级别服务，为 AccessibilityEvents 分发事件，并为查询系统的无障碍状态提供设施。当在用户界面上发生某些明显事件时，生成无障碍事件，例如一个 Activity 启动，一个 View 的焦点或选择改变，等等。对处理无障碍事件感兴趣的组织可以实现和注册一个继承自 AccessibilityService 的无障碍服务。

为了获取一个无障碍管理的权限，需要实现：
AccessibilityManager accessibilityManager = (AccessibilityManager) context.getSystemService(Context.ACCESSIBILITY_SERVICE);

参见：

AccessibilityEvent
AccessibilityNodeInfo
AccessibilityService
getSystemService(Class)
ACCESSIBILITY_SERVICE

1. 概要

1.1 嵌套类

interface	AccessibilityManager.AccessibilityStateChangeListener 系统无障碍状态的监听器。
interface	AccessibilityManager.TouchExplorationStateChangeListener 系统触摸浏览状态的监听器。

1.2 公有方法

boolean	addAccessibilityStateChangeListener (AccessibilityManager.AccessibilityStateChangeListener listener) 为系统全局无障碍状态的改变，注册一个 AccessibilityManager.AccessibilityStateChangeListener。
boolean	addTouchExplorationStateChangeListener (AccessibilityManager.TouchExplorationStateChangeListener listener) 为系统全局触摸浏览状态的改变，注册一个 AccessibilityManager.TouchExplorationStateChangeListener 。
List<ServiceInfo>	getAccessibilityServiceList() *该方法在 API 级别 14 时被弃用。使用*

	getInstalledAccessibilityServiceList()替代。
List<AccessibilityServiceInfo>	getEnabledAccessibilityServiceList(int feedbackTypeFlags) 为给定反馈类型，返回已启用无障碍服务的 AccessibilityServiceInfo 列表。
List<AccessibilityServiceInfo>	getInstalledAccessibilityServiceList() 返回已安装的无障碍服务的 AccessibilityServiceInfo 列表。
void	interrupt() 从所有无障碍服务中，请求反馈中断。
boolean	isEnabled() 返回系统中的无障碍是否已启用。
boolean	isTouchExplorationEnabled() 返回系统中的触摸浏览是否已启用。
boolean	removeAccessibilityStateChangeListener (AccessibilityManager.AccessibilityStateChangeListener listener) 注销一个 AccessibilityManager.AccessibilityStateChangeListener。
boolean	removeTouchExplorationStateChangeListener(AccessibilityManager.TouchExplorationStateChangeListener listener) 注销一个 AccessibilityManager.TouchExplorationStateChangeListener 。
void	sendAccessibilityEvent(AccessibilityEvent event)

	发送一个 AccessibilityEvent。

1.3 继承方法

继承自 java.lang.Object 类。

Object	clone() 创建并返回该对象的复本。
boolean	equals(Object obj) 标识某些其他对象是否"等同于"该对象。
void	finalize() 当垃圾收集器确认再也没有该对象的引用时，垃圾收集器调用该方法。
final Class<?>	getClass() 返回该对象的运行类。
int	hashCode() 为该对象返回哈希编码值。
final void	Notify() 唤醒在对象监视器上等待的单线程。
final void	notifyAll() 唤醒在对象监视器上等待的所有线程。
String	toString()

	返回对象的字符串表示。
final void	wait(long millis, int nanos) 导致当前线程等待，直到另一线程为该对象调用 Notify()方法或 notifyAll()方法，或某些其他线程中断当前线程，或一定数量的实时运行已经停止。
final void	wait(long millis) 导致当前线程等待，直到另一线程为该对象调用 Notify()方法或 notifyAll()方法，或一定数量的实时运行已经停止。
final void	wait() 导致当前线程等待，直到另一线程为该对象调用 Notify()方法或 notifyAll()方法。

2.公有方法

2.1 addAccessibilityStateChangeListener

添加于 API 级别 14。

boolean addAccessibilityStateChangeListener (AccessibilityManager.AccessibilityStateChangeListener listener)

为系统全局无障碍状态的改变，注册一个 AccessibilityManager.AccessibilityStateChangeListener。

参数：

Listener	AccessibilityManager.AccessibilityStateChangeListener: 监听器。

返回值：

boolean	如果注册成功返回 true。

2.2 addTouchExplorationStateChangeListener

添加于 API 级别 19。

boolean addTouchExplorationStateChangeListener (AccessibilityManager.TouchExplorationStateChangeListener listener)

为系统全局触摸浏览状态的改变，注册一个 AccessibilityManager.TouchExplorationStateChangeListener。

参数：

listener	AccessibilityManager.TouchExplorationStateChangeListener: 监听器。

返回值：

Boolean	如果注册成功返回 true。

2.3 getAccessibilityServiceList

添加于 API 级别 4。

List<ServiceInfo> getAccessibilityServiceList ()

注： *该方法在 API 级别 14 被弃用。现在使用 getInstalledAccessibilityServiceList() 替代。*

返回已安装的无障碍服务的 ServiceInfo 列表。

返回值：

List<ServiceInfo>	一个具有 ServiceInfo 的不可修改的列表。

2.4 getEnabledAccessibilityServiceList

添加于 API 级别 14。

List<AccessibilityServiceInfo> getEnabledAccessibilityServiceList (int feedbackTypeFlags)

为一个给定反馈类型，返回已启用无障碍服务的 AccessibilityServiceInfo 列表。

参数：

feedbackTypeFlags	int: 反馈类型标识。

返回值：

List<AccessibilityServiceInfo>	一个具有 AccessibilityServiceInfo 的不可修改的列表。

参见：

FEEDBACK_AUDIBLE

FEEDBACK_GENERIC

FEEDBACK_HAPTIC

FEEDBACK_SPOKEN

FEEDBACK_VISUAL

FEEDBACK_BRAILLE

2.5 getInstalledAccessibilityServiceList

添加于 API 级别 14。

List<AccessibilityServiceInfo> getInstalledAccessibilityServiceList ()

返回一个已安装的无障碍服务的 AccessibilityServiceInfo 列表。

返回值：

List<AccessibilityServiceInfo>	一个具有 AccessibilityServiceInfo 的不可修改的列表。

2.6 interrupt

添加于 API 级别 4。

void interrupt ()

从所有无障碍服务中请求反馈中断。

2.7 isEnabled

添加于 API 级别 4。

boolean isEnabled ()

返回系统中的无障碍是否已启用。

返回值：

boolean	如果已启用无障碍，返回 true，否则，返回 false。

2.8 isTouchExplorationEnabled

添加于 API 级别 14。

boolean isTouchExplorationEnabled ()

返回系统中的触摸浏览是否已启用。

返回值：

boolean	如果触摸浏览已启用，返回 true，否则，返回 false。

2.9 removeAccessibilityStateChangeListener

添加于 API 级别 14。

boolean removeAccessibilityStateChangeListener (AccessibilityManager.AccessibilityStateChangeListener listener)

注销一个 AccessibilityManager.AccessibilityStateChangeListener。

参数：

Listener	AccessibilityManager.AccessibilityStateChangeListener: 监听器。

返回值：

boolean	如果注销成功，返回 true。

2.10 removeTouchExplorationStateChangeListener

添加于 API 级别 19。

boolean removeTouchExplorationStateChangeListener (AccessibilityManager.TouchExplorationStateChangeListener listener)

注销一个 AccessibilityManager.TouchExplorationStateChangeListener。

参数：

listener	AccessibilityManager.TouchExplorationStateChangeListener: 监听器。

返回值：

boolean	如果注销成功，返回 true。

2.11 sendAccessibilityEvent

添加于 API 级别 4。

void sendAccessibilityEvent (AccessibilityEvent event)

发送一个 AccessibilityEvent。

参数：

event	AccessibilityEvent: 要发送的事件。

抛出：

IllegalStateException	如果无障碍没有被启用。注：发送自定义无障碍事件的偏好设置是通过调用 requestSendAccessibilityEvent(View, AccessibilityEvent)，而不是该方法，来允许父视图增加/过滤被其后代发送的事件。