

AccessibilityRecord

中国信息无障碍产品联盟&信息无障碍研究会 译制

20170120

目录

翻译声明.....	1
AccessibilityRecord	2
1.开发者指南.....	3
2.概要.....	4
2.1 公有方法.....	4
2.2 继承方法.....	9
3.公有方法.....	11
3.1getAddedCount	11
3.2getBeforeText.....	11
3.3getClassName	11
3.4getContentDescription	12
3.5getCurrentItemIndex	12
3.6getFromIndex	12
3.7getItemCount.....	13
3.8getMaxScrollX.....	13
3.9getMaxScrollY	13
3.10getParcelableData	14
3.11getRemovedCount.....	14
3.12getScrollX	14
3.13getScrollY	15
3.14getSource.....	15
3.15getText.....	16
3.16getToIndex	16
3.17getWindowId.....	16
3.18isChecked.....	17
3.19isEnabled.....	17
3.20isFullScreen.....	17
3.21isPassword.....	18
3.22isScrollable.....	18
3.23obtain.....	18
3.24obtain.....	19
3.25recycle	19

3.26setAddedCount.....	20
3.27setBeforeText	20
3.28setChecked	21
3.29setClassName	21
3.30setContentDescription.....	21
3.31setCurrentItemIndex	22
3.32setEnabled	22
3.33setFromIndex	23
3.34setFullScreen.....	23
3.35setItemCount	24
3.36setMaxScrollX	24
3.37setMaxScrollY	25
3.38setParcelableData.....	25
3.39setPassword.....	25
3.40setRemovedCount	26
3.41setScrollX.....	26
3.42setScrollY.....	27
3.43setScrollable	27
3.44setSource	27
3.45setSource	28
3.46setToIndex.....	28
3.47toString.....	29

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：刘彪、沈广荣、朱志杰

本文档翻译自谷歌官方文档《[AccessibilityRecord](#)》。如您对翻译文档内容有异议，请将原文文档作为主要参考，原文版权由 Google 持有并保留。

本翻译文档使用请参见 [CC BY-NC-SA 3.0](#)。文档可以免费使用、分享，但请保留本链接，如您对内容上有任何的意见或疑问，请发送邮件至 liuhui@siaa.org.cn，我们只是希望文档内容能够统一完整，真正帮助开发者完善产品的信息无障碍。

AccessibilityRecord

添加于 [API 级别 14](#)。

public class AccessibilityRecord

extends [Object](#)

[java.lang.Object](#)

↳ [android.view.accessibility.AccessibilityRecord](#)

已知直接子类

AccessibilityEvent	该类代表当用户界面发生某些明显的事件时，由系统发送的无障碍事件。
------------------------------------	----------------------------------

在 [AccessibilityEvent](#) 中代表一个记录，包含源 [View](#) 状态改变的信息。当一个视图发送一个无障碍事件，需要从其父视图请求调度构造事件。为了给 [AccessibilityService](#) 提供更多的上下文，父元素可能会选择性地添加一个记录。因此，无障碍服务可以通过添加无障碍记录来增强反馈。

一旦包含一个记录的无障碍事件被调度，该记录就不可改变，且调用一个状态改变方法会产生错误。

注意：并不是所有的属性都适用于所有无障碍事件类型。更多详细信息，参见 [AccessibilityEvent](#)。

1. 开发者指南

关于创建和处理无障碍记录的更多信息，参见《[Android 无障碍开发指南](#)》。

参见：

[AccessibilityEvent](#)

[AccessibilityManager](#)

[AccessibilityService](#)

[AccessibilityNodeInfo](#)

2. 概要

2.1 公有方法

int	getAddedCount() 获取被添加字符的数目。
CharSequence	getBeforeText() 在改变前，获取文本。
CharSequence	getClassName() 获取源的类名。
CharSequence	getContentDescription() 获取源的描述。
int	getCurrentItemIndex() 获取可访问项目列表中的源的索引。
int	getFromIndex() 获取变更序列的第一个字符的索引，或者文本选择起始的索引，或滚动时第一个可见项的索引。
int	getItemCount() 获取可访问项目的数目。
int	getMaxScrollX() 获取源左边界的最大滚动偏移像素数。

int	<code>getMaxScrollY()</code> 获取源上边界的最大滚动偏移像素数。
Parcelable	<code>getParcelableData()</code> 获取 Parcelable 数据。
int	<code>getRemovedCount()</code> 获取被移除字符的数目。
int	<code>getScrollX()</code> 获取源左边界的滚动偏移像素数。
int	<code>getScrollY()</code> 获取源上边界的滚动偏移像素数。
AccessibilityNodeInfo	<code>getSource()</code> 获取事件源的 AccessibilityNodeInfo。
List<CharSequence>	<code>getText()</code> 获取事件的文本。
int	<code>getToIndex()</code> 获取文本选择末尾的索引，或滚动时最后一个可见项的索引。
int	<code>getWindowId()</code> 获取事件来自窗口的 id。
boolean	<code>isChecked()</code> 获取源是否已选中。

boolean	isEnabled() 获取源是否已启用。
boolean	isFullScreen() 获取源是否占据整个屏幕。
boolean	isPassword() 获取源是否是个密码字段。
boolean	isScrollable() 获取源是否可滚动。
static AccessibilityRecord	obtain() 如果可获得，返回一个缓存实例，否则初始化一个新的。
static AccessibilityRecord	obtain(AccessibilityRecord record) 如果可获得，返回一个缓存实例，否则初始化一个新的。
void	recycle() 返回一个重复使用的实例。
void	setAddedCount(int addedCount) 设置被添加字符的数目。
void	setBeforeText(CharSequence beforeText) 在改变前，设置文本。
void	setChecked(boolean isChecked)

	如果源被选中，设置。
void	<code>setClassName(CharSequence className)</code> 设置源的类名称。
void	<code>setContentDescription(CharSequence contentDescription)</code> 设置源的描述。
void	<code>setCurrentItemIndex(int currentItemIndex)</code> 设置可访问项目列表中的源的索引。
void	<code>setEnabled(boolean isEnabled)</code> 如果源已启用，设置。
void	<code>setFromIndex(int fromIndex)</code> 设置变更序列第一个字符的索引，或文本选择的起始，或滚动时第一个可见项目的索引。
void	<code>setFullScreen(boolean isFullScreen)</code> 如果源占用了整个屏幕，设置。
void	<code>setItemCount(int itemCount)</code> 设置可访问项目的数目。
void	<code>setMaxScrollX(int maxScrollX)</code> 设置源左边界的最大滚动偏移像素数。
void	<code>setMaxScrollY(int maxScrollY)</code> 设置源上边界的最大滚动偏移像素数。

void	setParcelableData (Parcelable parcelableData) 设置事件的 Parcelable 数据。
void	setPassword (boolean isPassword) 如果源是个密码字段，设置。
void	setRemovedCount (int removedCount) 设置被移除字符的数目。
void	setScrollX (int scrollX) 设置源左边界的滚动偏移像素数。
void	setScrollY (int scrollY) 设置源上边界的滚动偏移像素数。
void	setScrollable (boolean scrollable) 如果源可滚动，设置。
void	setSource (View root, int virtualDescendantId) 将源设置为给定 root 的一个虚拟子元素。
void	setSource (View source) 设置事件源。
void	setToIndex (int toIndex) 设置文本选择末尾的索引，或滚动时，最后一个可见项目的索引。
String	toString () 返回对象的字符串代表。

2.2 继承方法

继承自 `java.lang.Object` 类。

Object	<code>clone()</code> 创建并返回该对象的复本。
boolean	<code>equals(Object obj)</code> 标识某些其他对象是否"等同于"该对象。
void	<code>finalize()</code> 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<?>	<code>getClass()</code> 返回该对象的运行类。
int	<code>hashCode()</code> 为该对象返回哈希编码值。
final void	<code>notify()</code> 唤醒在对象监视器上等待的单线程。
final void	<code>notifyAll()</code> 唤醒在对象监视器上等待的所有线程。
String	<code>toString()</code> 返回对象的字符串表示。
final void	<code>wait(long millis, int nanos)</code>

	使当前线程等待，直到另一线程为该对象调用 <code>notify()</code> 方法或 <code>notifyAll()</code> 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。
<code>final void</code>	<code>wait(long millis)</code> 使当前线程等待，直到另一线程为该对象调用 <code>notify()</code> 方法或 <code>notifyAll()</code> 方法，或一定数量的实时运行已经停止。
<code>final void</code>	<code>wait()</code> 使当前线程等待，直到另一线程为该对象调用 <code>notify()</code> 方法或 <code>notifyAll()</code> 方法。

3. 公有方法

3.1 getAddedCount

添加于 [API 级别 4](#)。

`int getAddedCount ()`

获取被添加字符的数目。

返回值：

<code>int</code>	被添加字符的数目。
------------------	-----------

3.2 getBeforeText

添加于 [API 级别 4](#)。

`CharSequence getBeforeText ()`

在改变前，获取文本。

返回值：

<code>CharSequence</code>	改变前文本。
---------------------------	--------

3.3 getClassNames

添加于 [API 级别 4](#)。

`CharSequence getClassNames ()`

获取源的类名。

返回值:

CharSequence	类名。
------------------------------	-----

3.4getContentDescription

添加于 [API 级别 4](#)。

[CharSequence](#) getContentDescription ()

获取源的描述。

返回值:

CharSequence	描述。
------------------------------	-----

3.5getCurrentItemIndex

添加于 [API 级别 4](#)。

int getCurrentItemIndex ()

获取可访问项目列表中的源的索引。

返回值:

int	当前项目索引。
-----	---------

3.6getFromIndex

添加于 [API 级别 4](#)。

int getFromIndex ()

获取变更序列的第一个字符的索引，或者文本选择起始的索引，或当滚动时第一个可见项的索引。

返回值：

int	第一个字符，或选择起始，或第一个可见项的索引。
-----	-------------------------

3.7getItemCount

添加于 [API 级别 4](#)。

`int getItemCount ()`

获取可访问项目的数目。

返回值：

int	项目的数目。
-----	--------

3.8getMaxScrollX

添加于 [API 级别 15](#)。

`int getMaxScrollX ()`

获取源左边界的最大滚动偏移像素数。

返回值：

int	最大滚动。
-----	-------

3.9getMaxScrollY

添加于 [API 级别 15](#)。

`int getMaxScrollY ()`

获取源上边界的最大滚动偏移像素数。

返回值：

<code>int</code>	最大滚动。
------------------	-------

3.10 getParcelableData

添加于 [API 级别 4](#)。

`Parcelable getParcelableData ()`

获取 `Parcelable` 数据。

返回值：

<code>Parcelable</code>	<code>Parcelable</code> 数据。
-------------------------	-----------------------------

3.11 getRemovedCount

添加于 [API 级别 4](#)。

`int getRemovedCount ()`

获取被移除字符的数目。

返回值：

<code>int</code>	被移除字符的数目。
------------------	-----------

3.12 getScrollX

添加于 [API 级别 14](#)。

`int getScrollX ()`

获取源左边界的滚动偏移像素数。

返回值：

<code>int</code>	滚动。
------------------	-----

3.13getScrollY

添加于 [API 级别 14](#)。

`int getScrollY ()`

获取源上边界的滚动偏移像素数。

返回值：

<code>int</code>	滚动。
------------------	-----

3.14getSource

添加于 [API 级别 14](#)。

[AccessibilityNodeInfo](#) getSource ()

获取事件源的 [AccessibilityNodeInfo](#)。

注意：通过调用 [AccessibilityNodeInfo#recycle\(\)](#)回收接收到的信息，避免创建多个实例，是客户端的责任。

返回值：

AccessibilityNodeInfo	源的信息。
---------------------------------------	-------

3.15 getText

添加于 [API 级别 4](#)。

`List<CharSequence> getText ()`

获取事件的文本。列表中的索引，代表文本的优先级。具体来说，索引越低，优先级越高。

返回值：

<code>List<CharSequence></code>	文本。
---------------------------------------	-----

3.16 getToIndex

添加于 [API 级别 14](#)。

`int getToIndex ()`

获取文本选择末尾的索引，或滚动时最后一个可见项的索引。

返回值：

<code>int</code>	选择末尾的索引，或最后项目的索引。
------------------	-------------------

3.17 getWindowId

添加于 [API 级别 14](#)。

`int getWindowId ()`

获取事件来自窗口的 id。

返回值：

int	窗口 id。
-----	--------

3.18isChecked

添加于 [API 级别 4](#)。

int getScrollY ()

获取源是否已选中。

返回值：

boolean	如果视图已选中返回 true，否则返回 false。
---------	----------------------------

3.19isEnabled

添加于 [API 级别 4](#)。

boolean isEnabled ()

获取源是否已启用。

返回值：

boolean	如果视图已启用返回 true，否则返回 false。
---------	----------------------------

3.20isFullScreen

添加于 [API 级别 4](#)。

boolean isFullScreen ()

获取源是否占据整个屏幕。

返回值:

boolean	如果视图是全屏返回 true，否则返回 false。
---------	----------------------------

3.21isPassword

添加于 [API 级别 4](#)。

boolean isPassword ()

获取源是否是个密码字段。

返回值:

boolean	如果视图是个密码字段返回 true，否则返回 false。
---------	-------------------------------

3.22isScrollable

添加于 [API 级别 14](#)。

boolean isScrollable ()

获取源是否可滚动。

返回值:

boolean	如果源可滚动返回 true，否则返回 false。
---------	---------------------------

3.23obtain

添加于 [API 级别 14](#)。

[AccessibilityRecord](#) obtain ()

如果可获得，返回一个缓存实例，否则初始化一个新的。

返回值：

AccessibilityRecord	一个实例。
-------------------------------------	-------

3.24obtain

添加于 [API 级别 14](#)。

[AccessibilityRecord](#) obtain ([AccessibilityRecord](#) record)

如果可获得，返回一个缓存实例，否则初始化一个新的。该实例初始化自给定记录中的数据。

参数：

record	AccessibilityRecord
--------	-------------------------------------

返回值：

AccessibilityRecord	一个实例。
-------------------------------------	-------

3.25recycle

添加于 [API 级别 14](#)。

void recycle ()

返回一个重复使用的实例。

注意：调用该函数之后，不能触摸对象。

抛出：

IllegalStateException	如果记录已被回收。
---------------------------------------	-----------

3.26 setAddedCount

添加于 [API 级别 4](#)。

`void setAddedCount (int addedCount)`

设置被添加字符的数目。

参数：

<code>addedCount</code>	<code>int</code> : 被添加字符的数目。
-------------------------	------------------------------

抛出：

IllegalStateException	如果从一个 <code>AccessibilityService</code> 调用。
---------------------------------------	---

3.27 setBeforeText

添加于 [API 级别 4](#)。

`void setBeforeText (CharSequence beforeText)`

在改变前，设置文本。

参数：

<code>beforeText</code>	<code>CharSequence</code> : 改变前文本。
-------------------------	------------------------------------

抛出：

IllegalStateException	如果从一个 <code>AccessibilityService</code> 调用。
---------------------------------------	---

3.28setChecked

添加于 [API 级别 4](#)。

void setChecked (boolean isChecked)

如果源被选中，设置。

参数：

isChecked	boolean: 如果视图被选中为 true，否则为 false。
-----------	-----------------------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.29setClassName

添加于 [API 级别 4](#)。

void setClassName ([CharSequence](#) className)

设置源的类名。

参数：

className	CharSequence : 类名。
-----------	------------------------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.30setContentDescription

添加于 [API 级别 4](#)。

`void setContentDescription (CharSequence contentDescription)`

设置源的描述。

参数：

<code>contentDescription</code>	CharSequence: 描述。
---------------------------------	-------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.31 setCurrentItemIndex

添加于 [API 级别 4](#)。

`void setCurrentItemIndex (int currentItemIndex)`

设置可访问列表项中的源的索引。

参数：

<code>currentItemIndex</code>	int: 当前项目索引。
-------------------------------	--------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.32 setEnabled

添加于 [API 级别 4](#)。

`void setEnabled (boolean isEnabled)`

如果源已启用，设置。

参数：

isEnabled	boolean: 如果视图已启用，为 true，否则为 false。
-----------	------------------------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.33 setFromIndex

添加于 [API 级别 4](#)。

void setFromIndex (int fromIndex)

设置变更序列第一个字符的索引，或文本选择的起始，或滚动时第一个可见项目的索引。

参数：

fromIndex	int: 第一个字符的索引，或选择起始，或第一个可见项目。
-----------	-------------------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.34 setFullScreen

添加于 [API 级别 4](#)。

void setFullScreen (boolean isFullScreen)

如果源占据整个屏幕，设置。

参数:

isFullScreen	boolean: 如果源是全屏, 为 true, 否则为 false。
--------------	-------------------------------------

抛出:

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.35 setItemCount

添加于 [API 级别 4](#)。

```
void setItemCount (int itemCount)
```

设置可访问项目的数目。

参数:

itemCount	int: 项目数目。
-----------	------------

抛出:

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.36 setMaxScrollX

添加于 [API 级别 15](#)。

```
void setMaxScrollX (int maxScrollX)
```

设置源左边界的最大滚动偏移像素数。

参数:

maxScrollX	int: 最大滚动。
------------	------------

3.37 setMaxScrollY

添加于 [API 级别 15](#)。

```
void setMaxScrollY (int maxScrollY)
```

设置源上边界的最大滚动偏移像素数。

参数：

maxScrollY	int: 最大滚动。
------------	------------

3.38 setParcelableData

添加于 [API 级别 4](#)。

```
void setParcelableData (Parcelable parcelableData)
```

设置事件的 [Parcelable](#) 数据。

参数：

parcelableData	Parcelable : parcelable 数据。
----------------	---

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--

3.39 setPassword

添加于 [API 级别 4](#)。

```
void setPassword (boolean isPassword)
```

如果源是个密码字段，设置。

参数:

isPassword	boolean: 如果视图是个密码字段, 为 true, 否则为 false。
------------	---

抛出:

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.40 setRemovedCount

添加于 [API 级别 4](#)。

void setRemovedCount (int removedCount)

设置被移除字符的数目。

参数:

removedCount	int: 被移除字符的数目。
--------------	----------------

抛出:

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.41 setScrollX

添加于 [API 级别 14](#)。

void setScrollX (int scrollX)

设置源左边界的滚动偏移像素数。

参数:

scrollX	int: 滚动。
---------	----------

3.42 setScrollY

添加于 [API 级别 14](#)。

```
void setScrollY (int scrollY)
```

设置源上边界的滚动偏移像素数。

参数：

scrollY	int: 滚动。
---------	----------

3.43 setScrollable

添加于 [API 级别 14](#)。

```
void setScrollable (boolean scrollable)
```

如果源可滚动，设置。

参数：

scrollable	boolean: 如果源可滚动，为 true，否则为 false。
------------	-----------------------------------

抛出：

IllegalStateException	如果从一个 AccessibilityService 调用。
---------------------------------------	--------------------------------

3.44 setSource

添加于 [API 级别 16](#)。

```
void setSource (View root, int virtualDescendantId)
```

将源设置为给定 root 的一个虚拟子元素。如果 virtualDescendantId 等于

`NO_ID`, `root` 被设置为源。

一个虚拟子元素是个虚构视图, 为无障碍目的, 被报告为视图层次的一部分。这让绘制复杂内容的自定义视图可以将自己报告为一个虚拟视图树, 然后传递其逻辑结构。

参数:

<code>root</code>	View: 虚拟子树的 <code>root</code> 。
<code>virtualDescendantId</code>	int: 虚拟子元素的 <code>id</code> 。

3.45 setSource

添加于 [API 级别 14](#)。

`void setSource (View source)`

设置事件源。

参数:

<code>source</code>	View: 源。
---------------------	-----------------

抛出:

IllegalStateException	如果从一个 <code>AccessibilityService</code> 调用。
---------------------------------------	---

3.46 setToIndex

添加于 [API 级别 14](#)。

`void setToIndex (int toIndex)`

设置文本选择末尾的索引, 或滚动时, 最后一个可见项目的索引。

参数:

toIndex	int: 选择末尾的索引, 或最后项目的索引。
---------	-------------------------

3.47 toString

添加于 [API 级别 14](#)。

String toString ()

返回对象的一个字符串代表。一般来说, **toString** 方法返回一个字符串, 以文本代表该对象。结果应该是个简洁翔实的代表, 且容易被用户阅读。推荐所有子类重写该方法。

Object 类的 **toString** 方法, 返回一个字符串, 包含实例对象类的名称, 字符 '@', 和对象哈希编码的无符号十六进制代表。换句话说, 该方法返回一个字符串, 等同于:

```
getClass().getName() + '@' + Integer.toHexString(hashCode())
```

返回值:

String	对象的字符串代表。
---------------	-----------