

UI Automator deep diving

Светлана Смелъчакова

LG Russia R&D Lab

О чем буду говорить?

- Что такое UI Automator
- Как писать тесты
- UI Automator и Accessibility Service
- Custom controls
- API UI Automator v1 и v2
- Проблемы с UI Automator
- Настройка тестов

Что такое UI Automator?

UI Automator - библиотека для поиска компонентов на экране и эмуляции действий пользователя (клики, тапы, ввод текста).



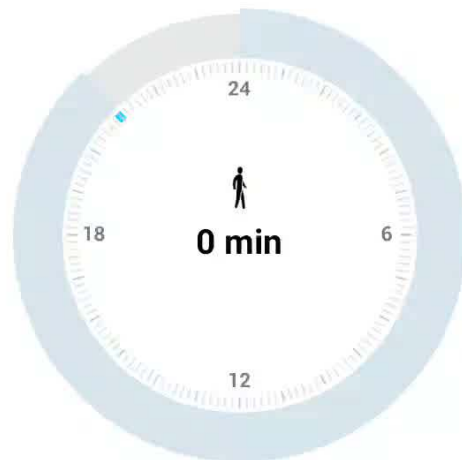
Применение UI Automator

- Если ваше приложение взаимодействует с другими приложениями. (сделать фото, открыть карту, permissions, gms)

LG Health



0 steps (0 Cal, 0.00 km)
Goal: 6573



Get active! (+)

CHALLENGES

Get started on the life you want

WEIGHT

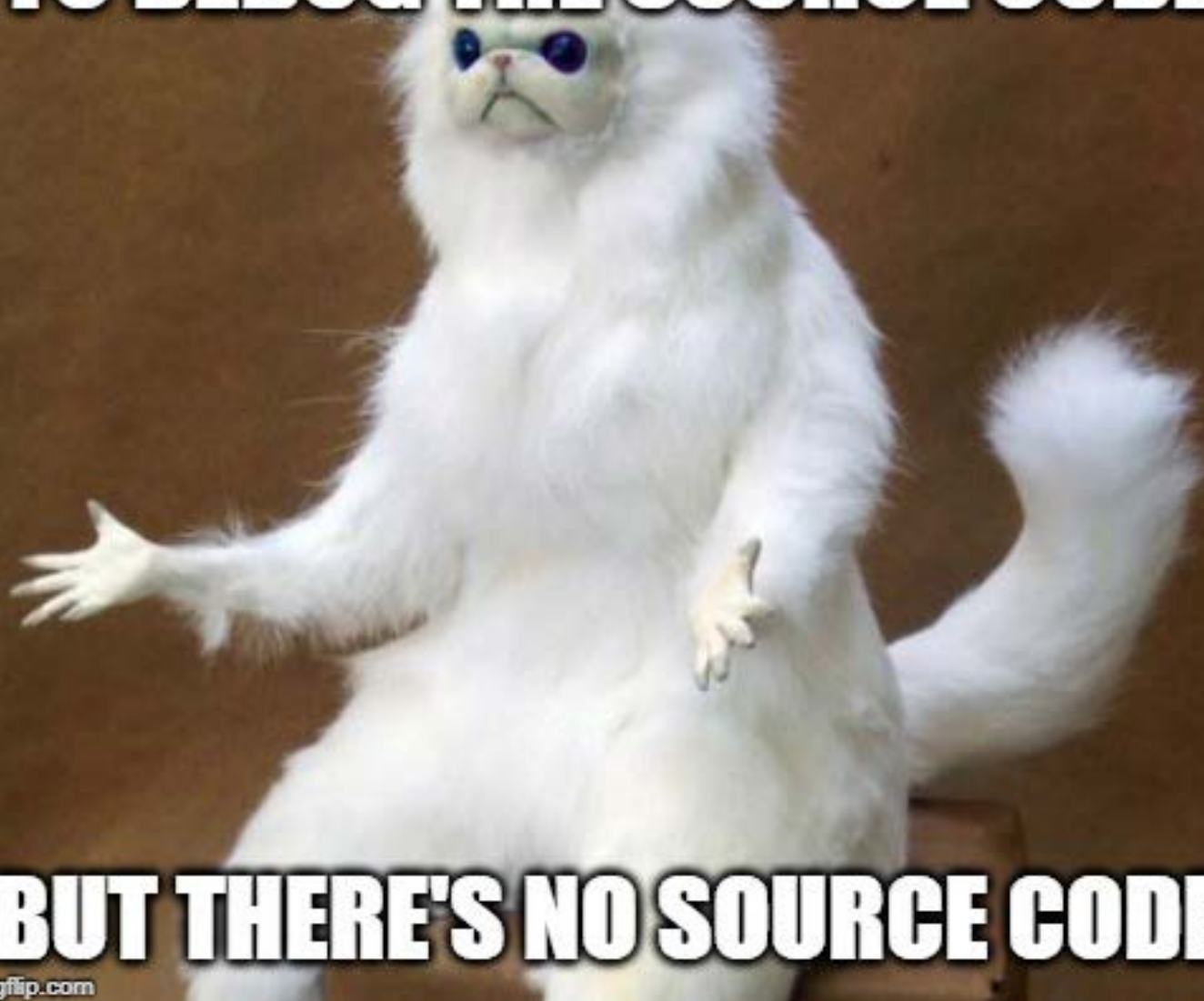


RANKING



- Если ваше приложение взаимодействует с другими приложениями. (сделать фото, открыть карту, permissions)
- Нет исходников

**WHEN THEY ASK YOU
TO DEBUG THE SOURCE CODE**



BUT THERE'S NO SOURCE CODE

- Если ваше приложение взаимодействует с другими приложениями. (сделать фото, открыть карту, permissions)
- Нет исходников
- Вендоры, тестирование девайсов



Отличия от Espresso

	UI Automator	Espresso
Принцип	Черный ящик	Белый ящик
Количество приложений	∞	1
Изменение настроек девайса	✓	✗
Notifications	✓	✗
Скриншоты	✓	✗
Скорость тестов	✗	✓

Тест на UI Automator

02:33

Чт, 21 ноября



Облачно 3°C



Google



Telegram



Hangouts



AIReader



Skype



ВКонтакте



Chrome



Инструменты



Галерея



Настройки



Play Маркет



Яндекс



Тест на Espresso

02:17

Чт, 21 ноября



Облачно 3°C



Google



Telegram



Hangouts



AIReader



Skype



ВКонтакте



Chrome



Инструменты



Галерея



Настройки



Play Маркет



Яндекс



Минусы UI Automator

- медленнее Espresso

Простой тест:

Espresso – 3,6 с

UI Automator – 21 с

- не совместим с любыми отрисованными приложениями (OpenGL, Unity)
- кастомные контролы



UI Automator и Espresso



UI Automator

+



UI TESTING FOR ANDROID

espresso

Как писать тесты на UI Automator



Android Studio, Instrumented tests



Appium



Сбилдить jar, запустить из adb

Запускаем из Android Studio Build.gradle

```
android {  
    defaultConfig {  
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
    }  
}  
dependencies {  
    androidTestImplementation 'androidx.test:runner:1.3.0-alpha02'  
    androidTestImplementation 'androidx.test:uiautomator:2.2.0'  
}
```

Tect UI Automator c AndroidJUnit4

```
@RunWith(AndroidJUnit4.class)
```

```
public class ChangeTextBehaviorTest_ {
```

```
    @Test
```

```
    public void testChangeText_sameActivity() {
```

Tect UI Automator c AndroidJUnit4

```
@RunWith(AndroidJUnit4.class)
```

```
public class ChangeTextBehaviorTest_ {
```

```
    @Test
```

```
    public void testChangeText_sameActivity() {
```

```
        mDevice.findObject(
```

```
            By.res(BASIC_SAMPLE_PACKAGE, "editTextInput"))
```

```
                .setText("UI Automator is awesome");
```

Tect UI Automator c AndroidJUnit4

```
@RunWith(AndroidJUnit4.class)
```

```
public class ChangeTextBehaviorTest_ {
```

```
@Test
```

```
public void testChangeText_sameActivity() {
```

```
    mDevice.findObject(
```

```
        By.res(BASIC_SAMPLE_PACKAGE, "editTextInput"))
```

```
        .setText("UI Automator is awesome");
```

```
    mDevice.findObject(
```

```
        By.res(BASIC_SAMPLE_PACKAGE, "changeTextBt"))
```

```
        .click();
```

Запуск теста

```
adb shell am instrument -w -r package_name  
/androidx.test.runner.AndroidJUnitRunner
```

UI Automator Viewer

UI Automator Viewer

22:31

00:00:00

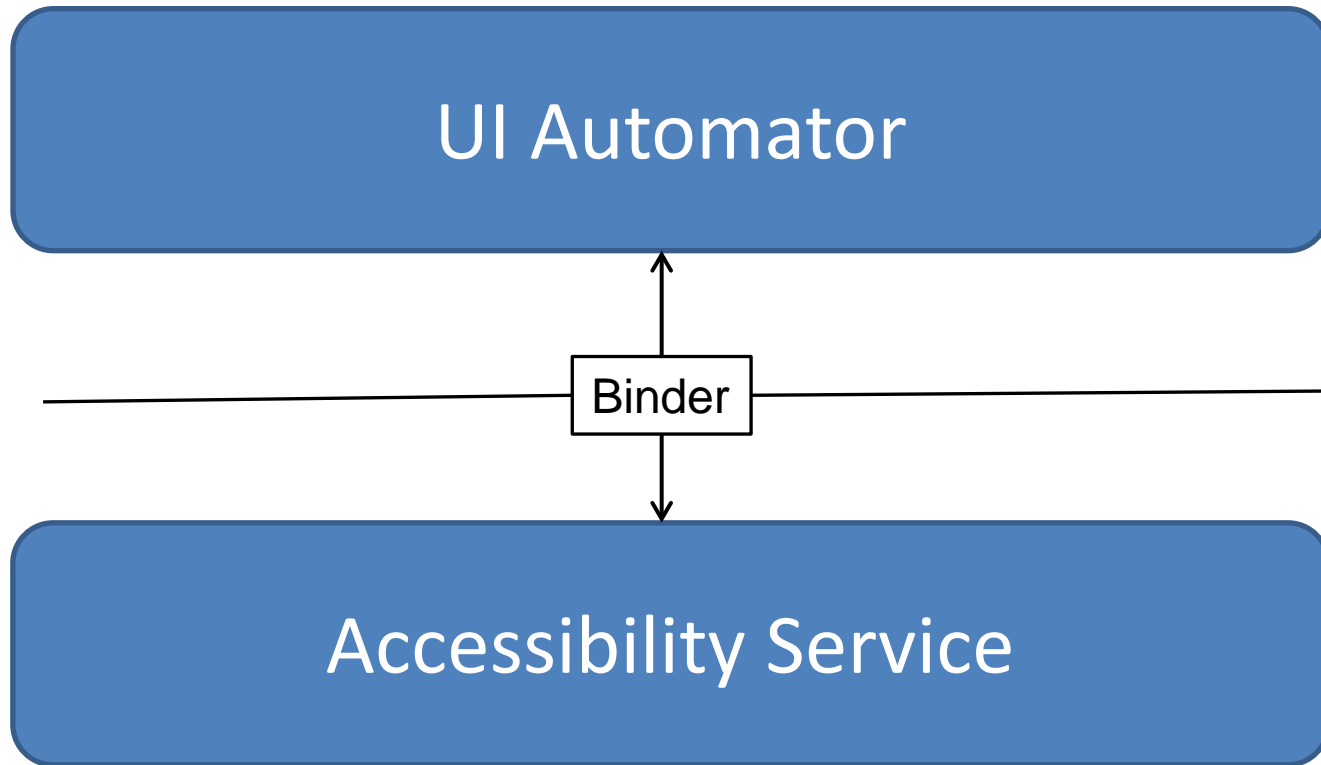
Стандартное качество | HD

Node Detail

index	1
text	
resource-id	<u>com.android.soundrecorder/id/btn_record</u>
class	<u>android.widget.ImageView</u>
package	com.android.soundrecorder
content-desc	<u>Запись</u>
checkable	false
checked	false
clickable	true
enabled	true
focusable	true
focused	false
scrollable	false
long-clickable	false
password	false
selected	false
bounds	[454,1847][626,2019]

Accesibility services - службы специальных возможностей, призванные облегчить работу с устройствами для людей, чьи возможности так или иначе ограничены.





 **TalkBack****Вкл.****Ярлык TalkBack**

Чтобы включить или выключить режим TalkBack, нажмите и удерживайте обе клавиши громкости более 3 секунд.

**Приватный экран**

Отключить подсветку на всех экранах. Дважды коснитесь экрана тремя пальцами, чтобы включить или выключить его.

**Ввод одним касанием клавиатуры**

Если пользуетесь клавиатурой или панелью LG, для ввода клавиши коснитесь только один раз

**Преобразование текста в речь**

Определить настройки функции голосового воспроизведения текста (TTS)

Настройки

Application

View

View

View

View

View

View



Accessibility
Service

AccessibilityNodeInfo

AccessibilityNodeInfo

AccessibilityNodeInfo

AccessibilityNodeInfo

AccessibilityNodeInfo

AccessibilityNodeInfo

```
public class AccessibilityNodeInfo implements Parcelable {  
    private final Rect mBoundsInScreen = new Rect();  
    private CharSequence mClassName;  
    private CharSequence mText;  
    private String mViewIdResourceName;  
    ...  
}
```

Класс android.view.View.java

```
void onInitializeAccessibilityNodeInfoInternal(AccessibilityNodeInfo info) {  
    info.setBoundsInParent(bounds);  
    info.setPackageName(mContext.getPackageName());  
    info.setClassName(View.class.getName());  
    info.setContentDescription(getContentDescription());  
    ...  
}
```

Класс android.view.View.java

@UiThread

```
public class View implements AccessibilityEventSource {
```

```
    int mAccessibilityViewId = NO_ID;
```

```
    public void sendAccessibilityEvent(int eventType) {
```

```
        if (mAccessibilityDelegate != null) {
```

```
            mAccessibilityDelegate.sendAccessibilityEvent(this, eventType);
```

```
        } else {
```

```
            sendAccessibilityEventInternal(eventType);
```

```
        }
```

```
    }
```

```
}
```


Класс android.View.View.java

```
public boolean performClick() {  
    sendAccessibilityEvent(AccessibilityEvent.TYPE_VIEW_CLICKED);  
    ...  
}  
public boolean performLongClick() {  
    sendAccessibilityEvent(AccessibilityEvent.TYPE_VIEW_LONG_CLICKED);  
    ...  
}  
protected void onFocusChanged(...) {  
    if (gainFocus) {  
        if (AccessibilityManager.getInstance(mContext).isEnabled()) {  
            sendAccessibilityEvent(AccessibilityEvent.TYPE_VIEW_FOCUSED);  
            requestAccessibilityFocus();  
        }  
    }  
    ...  
}
```

```
public final class AccessibilityEvent implements Parcelable {  
    private @EventType int mEventType;  
    private CharSequence mPackageName;  
    private long mEventTime;  
    ...  
}
```

← Vision

TalkBack

Provide spoken feedback for UI interactions



Message/call voice notifications

Read caller info and messages out loud



Font

Display size

Set items on the screen to a size easy for you to see

Touch zoom

Zoom in or out by triple-tapping the screen



Window zoom

Zoom in or out within a window and invert the color



Large mouse pointer



High contrast screen

Increase screen contrast with a black background. Some screens may not change.





APPLICATION FRAMEWORK

BINDER IPC PROXIES

ANDROID SYSTEM SERVICES

MEDIA SERVER

AudioFlinger

Camera Service

MediaPlayer Service

Other Media Services

SYSTEM SERVER

Search Service

Activity Manager

Window Manager

Other System Services & Managers

HAL

Camera HAL

Audio HAL

Graphics HAL

Other HALs

LINUX KERNEL

Camera Driver

Audio Driver (ALSA, OSS, etc.)

Display Drivers

Other Drivers

Application

Phone
Camera
Launcher

Alarm
Gallery
Email

App

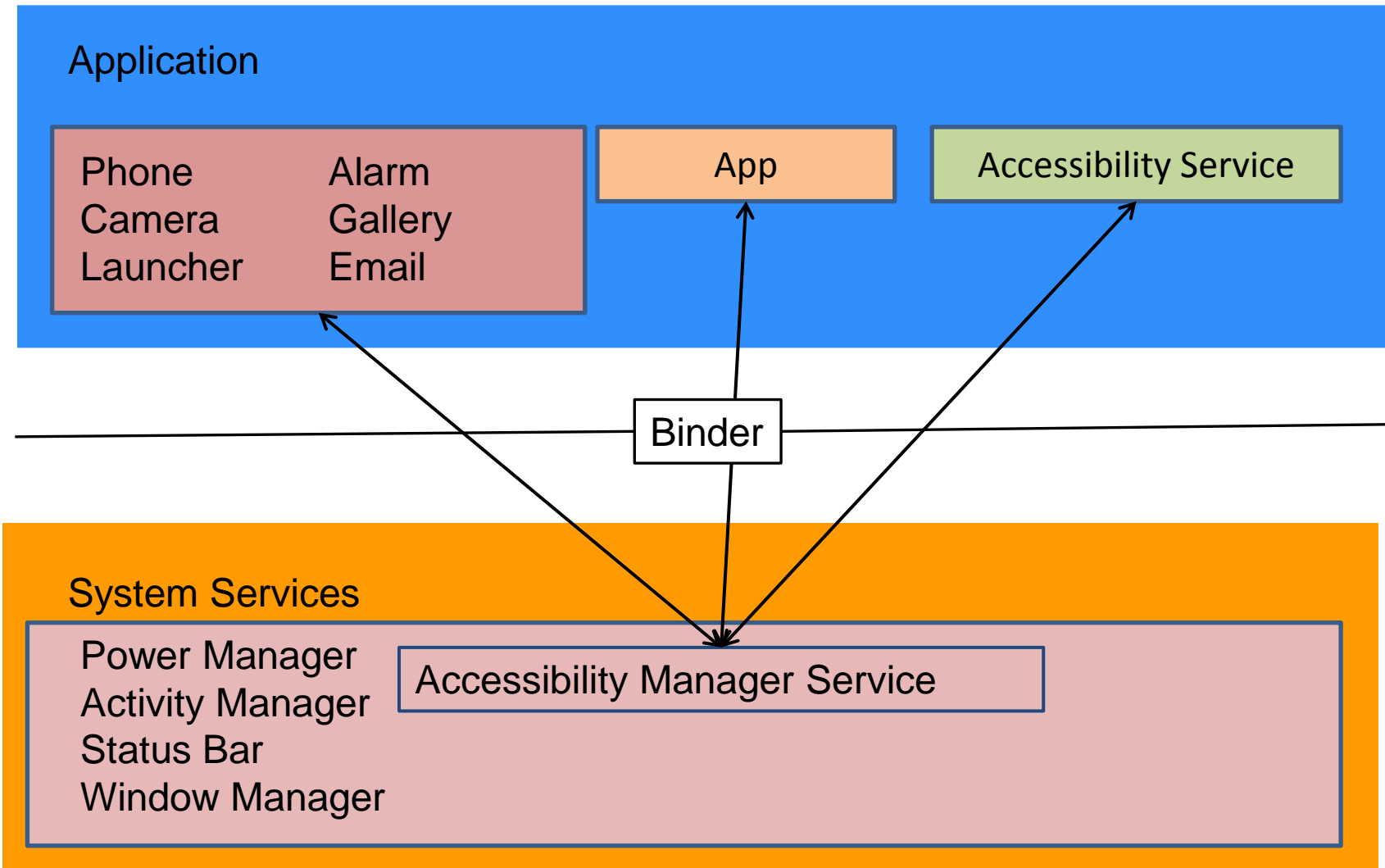
Accessibility Service

Binder

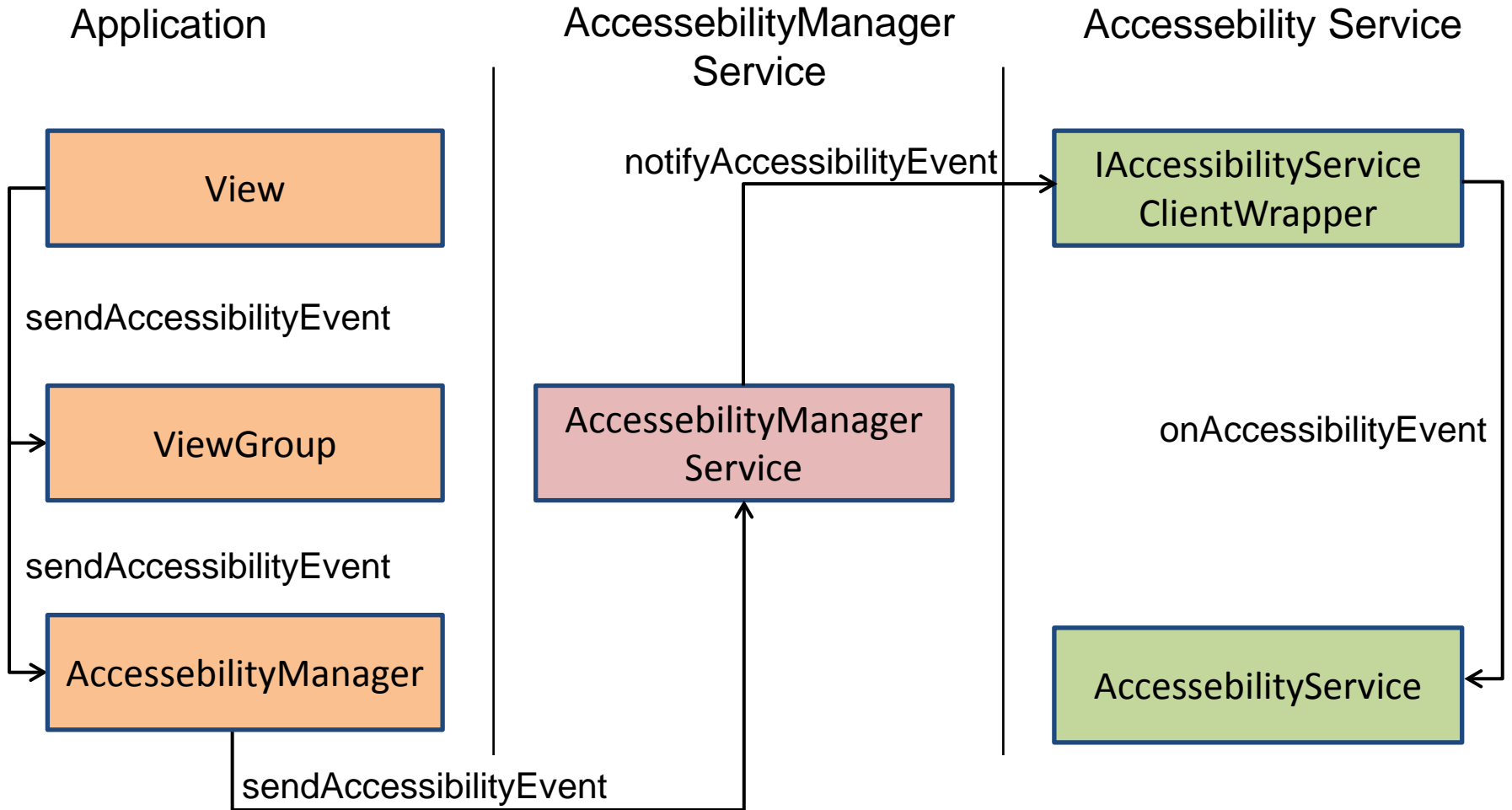
System Services

Power Manager
Activity Manager
Status Bar
Window Manager

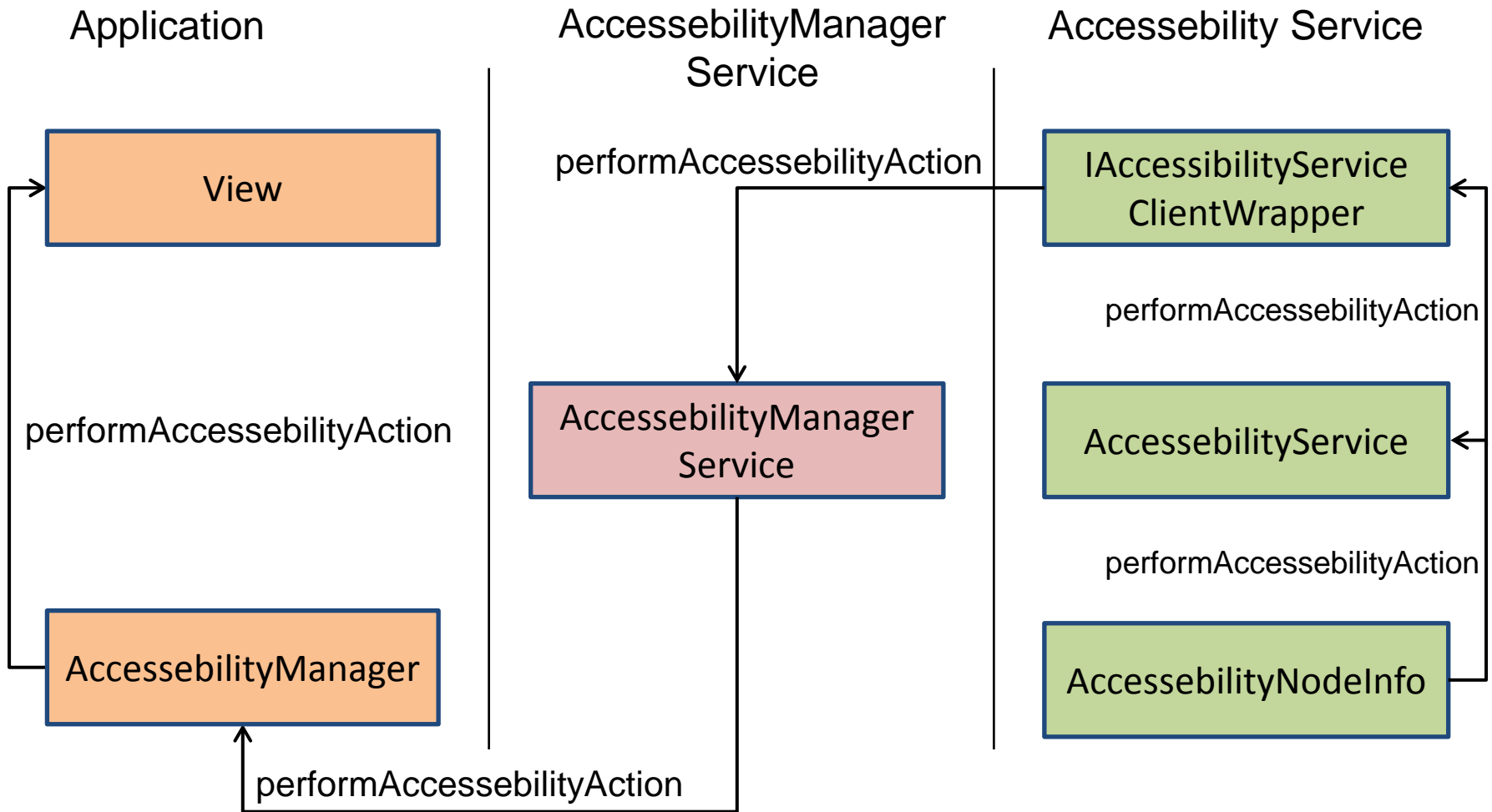
Accessibility Manager Service



Application to Accessibility Service



Accessibility Service to Application



Класс View

```
public boolean performAccessibilityAction(int action) {  
    switch (action) {  
        case AccessibilityNodeInfo.ACTION_CLICK: {  
            onTouchEvent(event);  
        } break;  
        case AccessibilityNodeInfo.ACTION_FOCUS: {  
            if (!hasFocus()) {  
                getViewRootImpl().ensureTouchMode(false);  
                return requestFocus();  
            }  
        } break;  
        ...  
    }
```



Custom controls

Custom controls must implements methods:

`dispatchPopulateAccessibilityEvent()`

`onPopulateAccessibilityEvent()`

`onInitializeAccessibilityEvent()`

`onInitializeAccessibilityNodeInfo()`



ExploreByTouchHelper

```
mExploreByTouchHelper = new ExploreByTouchHelper(this) {  
    @Override  
    int getViewAt(float x, float y) { ..}
```

```
    @Override  
    boolean onPerformActionForVirtualView(int virtualViewId, int action) {  
        if (action == AccessibilityNodeInfoCompat.ACTION_CLICK) {  
            mExploreByTouchHelper.sendEventForVirtualView(virtualViewId,  
                AccessibilityEvent.TYPE_VIEW_CLICKED);  
            return true;  
        }  
        return false;  
    }  
}
```

```
ViewCompat.setAccessibilityDelegate(this, mExploreByTouchHelper);
```

Overview

Tutorials

Getting started

Accessibility

Overview

[Make your app accessible](#)

App Maker accessibility features

Data

Pages

Widgets

Scripts

Security

Publish and manage

Make your app accessible



Contents

[Accessible design](#)[App Maker accessibility tools](#)[Customize UI descriptions with ARIA labels](#)[Provide text to screen readers](#)

App Maker apps should be usable by everyone, including people with disabilities. Common disabilities that affect a person's use of an app include blindness or low vision, color blindness, deafness or impaired hearing, and restricted motor skills.

When you develop apps with accessibility in mind, you improve the user experience for all users, not only for people with disabilities.

Accessible design



Accessible design includes the layout, size, color, and navigation of your app, as well as its interaction with adaptive and assistive technologies.

To enable users of all abilities to use your app successfully, use the Material Design [accessibility best practices](#). The following table describes the main principles of accessibility





power

smart auto-charging

LG

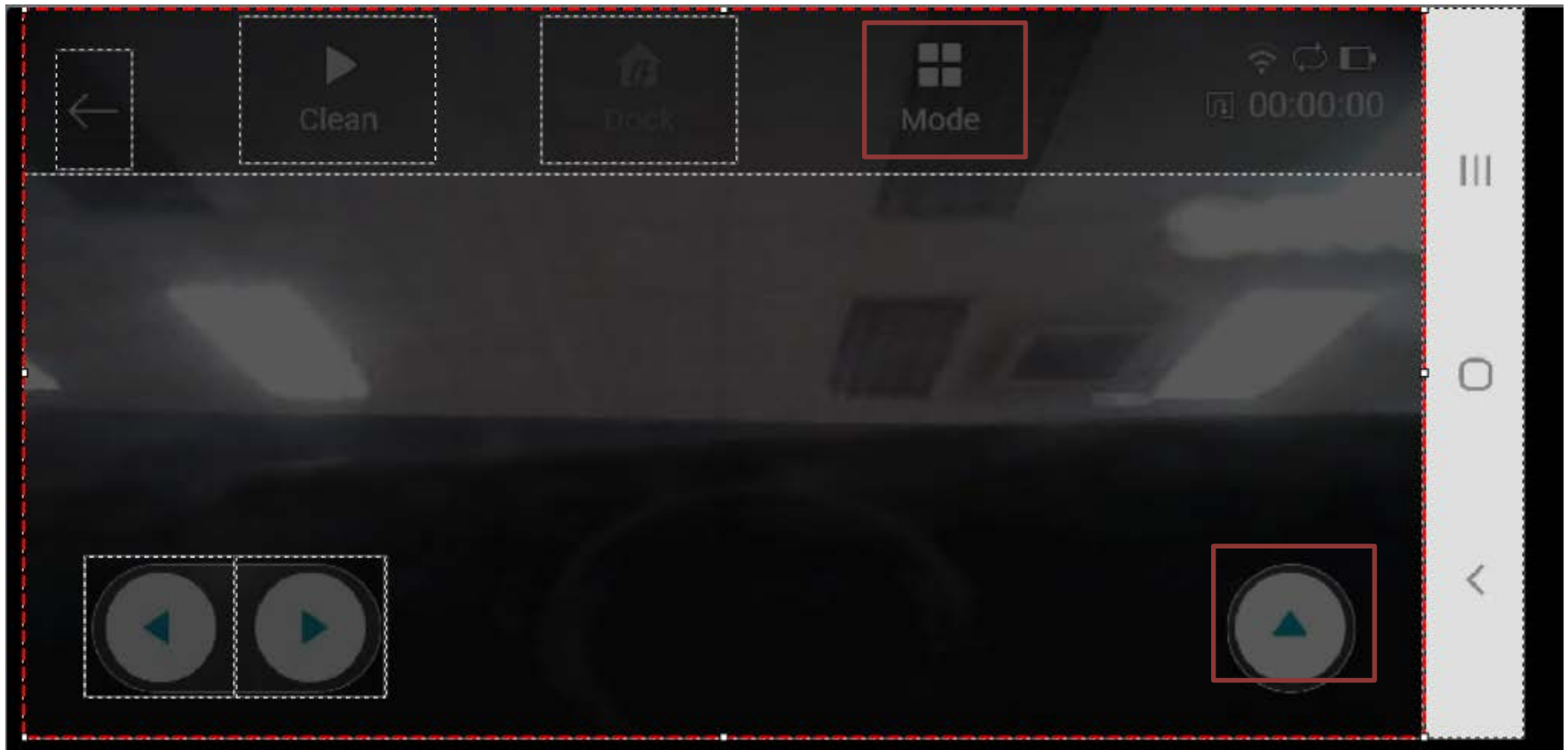
TURBO 3sec
MODE
HOME Language 3sec
START STOP

Smart Inverter MOTOR
10 YEAR WARRANTY

HOM-BOT

PUSH

LG SmartThinQ app



Custom Accessibility Service

```
<accessibility-service
```

```
  android:description="@string/accessibility_service_description"
```

```
  android:packageNames="com.example.android.apis"
```

```
  android:accessibilityEventTypes="typeAllMask"
```

```
  android:accessibilityFlags="flagDefault"
```

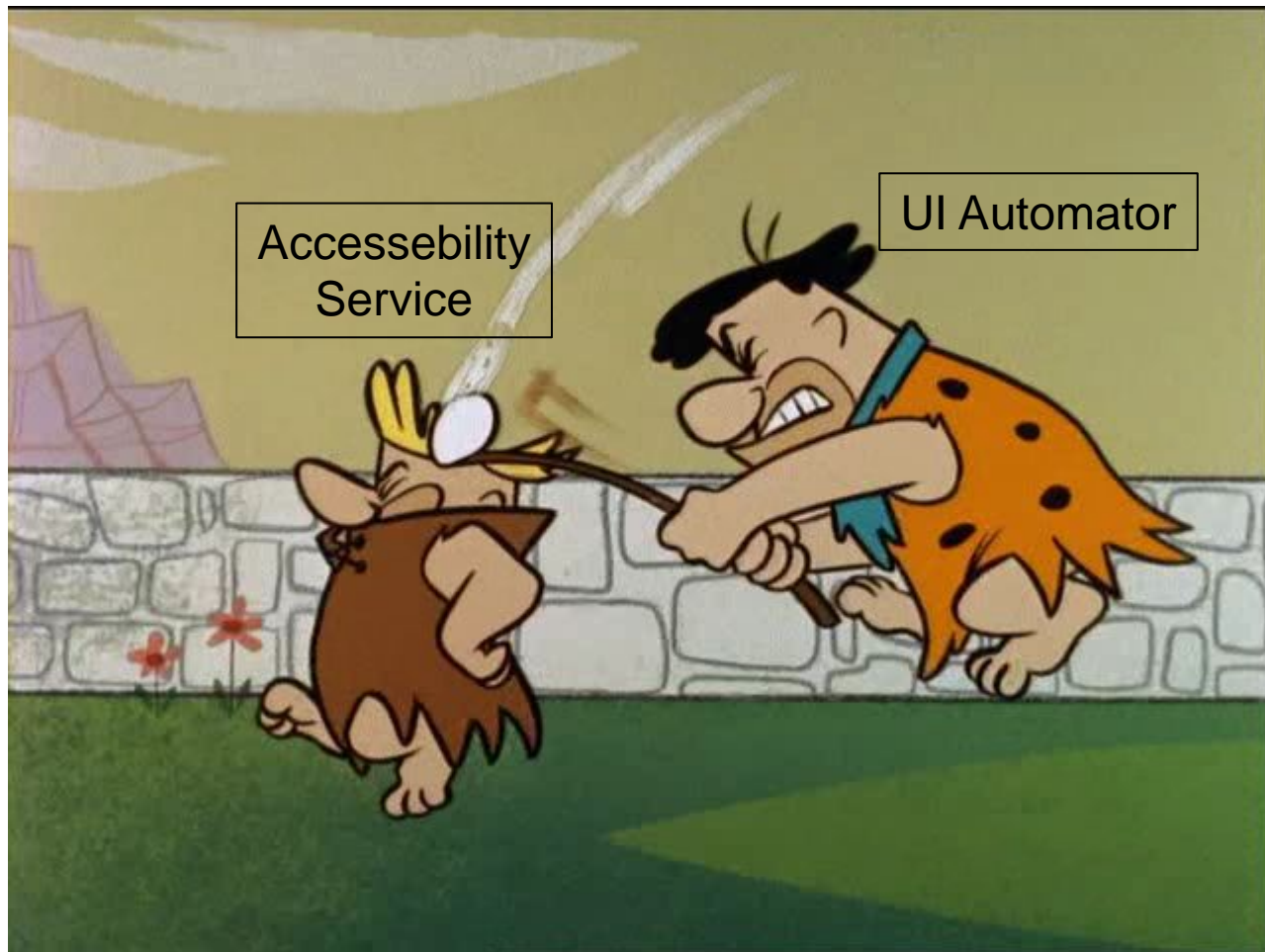
```
  android:accessibilityFeedbackType="feedbackSpoken"
```

```
  android:notificationTimeout="100"
```

```
  android:canRetrieveWindowContent="true"
```

```
/>
```

До SDK 24 мог быть запущен только один Accessibility



SDK 24+

```
UiAutomation uiAutomation = getInstrumentation().  
    getUiAutomation(UiAutomation.  
        FLAG_DONT_SUPPRESS_ACCESSIBILITY_SERVICES)
```



UI Automator API

- UiDevice
- UiSelector
- UiObject
- BySelector
- UiObject2
- Configurator
- UiAutomation

UiDevice

```
mDevice = UiDevice.getInstance(getInstrumentation());
```

Public methods

openNotification()

pressBack()

setOrientationLeft()

takeScreenshot(File storePath, float scale, int quality)


```
mDevice = UiDevice.getInstance(getInstrumentation());  
mDevice.setOrientationRight();  
mDevice.takeScreenshot(new File("D:/screen.png"));  
mDevice.pressMenu();
```


Настройки

🔍 Поиск настроек

 О телефоне MIUI Global 11.0.6 >


 Обновление компонентов 1 >

 Состояние защиты >

 SIM-карты и мобильные сети >

 Wi-Fi experia >

 Bluetooth Вкл >

 Точка доступа Wi-Fi Откл >

 Подключение и общий доступ >

 Активный экран и блокировка >

UiSelector

```
UiSelector().text("Wi-Fi")
```

Public methods

```
checkable(boolean val)
```

```
className(String className)
```

```
descriptionContains(String desc)
```

```
resourceId(String id)
```

UiObject

```
UiObject awesomeButton =  
    mDevice.findObject(UiSelector().text("Wi-Fi"))
```

Public methods

click()

exists()

getBounds()

isEnabled()

setText(String text)


```
mDevice = UiDevice.getInstance(  
                                getInstrumentation());  
UiObject textObject= mDevice.findObject(  
                    new UiSelector().text("Wi-Fi"));  
textObject.click();
```

Настройки

🔍 Поиск настроек

 О телефоне MIUI Global 11.0.6 >


 Обновление компонентов 1 >

 Состояние защиты >


 SIM-карты и мобильные сети >

 Wi-Fi experia >

 Bluetooth Вкл >

 Точка доступа Wi-Fi Откл >

 Подключение и общий доступ >

 Активный экран и блокировка >

BySelector

By.text("Wi-Fi")

Public methods

checkable(boolean isCheckable)

clazz(String packageName, String className)

descContains(String substring)

text(String textValue)

UiObject2

```
UiObject2 awesomeButton = mDevice.findObject(  
    By.text("Wi-Fi"))
```

Public methods

click()

getBounds()

isEnabled()

setText(String text)

mDevice =

`UiDevice.getInstance(getInstrumentation());`

`UiObject2 textObject=`

`mDevice.findObject(By.text("Wi-Fi"));`

`textObject.click();`

Настройки

🔍 Поиск настроек

 О телефоне MIUI Global 11.0.6 >


 Обновление компонентов 1 >

 Состояние защиты >


 SIM-карты и мобильные сети >

 Wi-Fi experia >

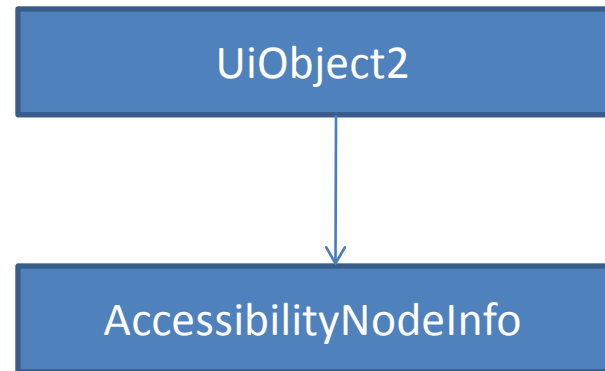
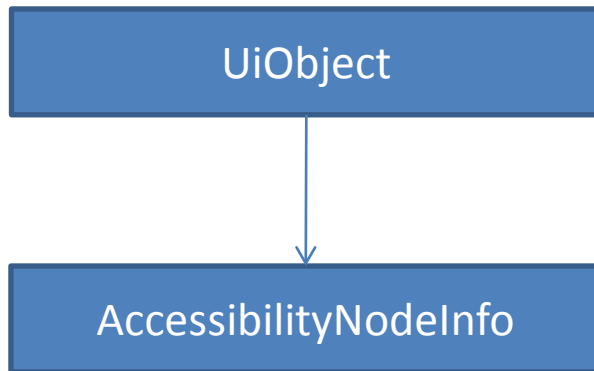
 Bluetooth Вкл >

 Точка доступа Wi-Fi Откл >

 Подключение и общий доступ >

 Активный экран и блокировка >

UiObject vs UiObject2



UI Automator v1 vs UI Automator v2

UiObject	UiObject2
Запускается поиск по дереву элементов	Кэшированный объект
<pre>UiObject awesomeButton = new UiObject(new UiSelector().text("Display"))</pre>	<pre>UiObject2 awesomeButton = mDevice.findObject(By.text("Display"))</pre>

Settings



Connections

Wi-Fi, Bluetooth, Data usage, Airplane mode



Sounds and vibration

Sound mode, Ringtone, Volume



Notifications

Block, allow, prioritize



Display

Brightness, Blue light filter, Home screen



Wallpapers and themes

Wallpapers, Themes, Icons

Display



Brightness



Adaptive brightness

On



Blue light filter

Reduce eye strain by limiting the amount of blue light emitted by the screen.



Night mode

Use a darker theme to keep your eyes comfortable at night.



Screen mode

Adaptive display

Font size and style

Screen zoom

Screen resolution

WQHD+ (2960 x 1440)

Класс QueryController

```
AccessibilityNodeInfo findAccessibilityNodeInfo(UiSelector selector,
    boolean isCounting) {
    mUiAutomatorBridge.waitForIdle();
    initializeNewSearch();
    synchronized (mLock) {
        AccessibilityNodeInfo rootNode = getRootNode();
        if (rootNode == null) {
            Log.e(LOG_TAG, "Cannot proceed when root node is null. Aborted search");
            return null;
        }
        // Copy so that we don't modify the original's sub selectors
        UiSelector uiSelector = new UiSelector(selector);
        return translateCompoundSelector(uiSelector, rootNode, isCounting);
    }
}
```

Configurator

Public methods

`getWaitForIdleTimeout()`

`setKeyInjectionDelay(long delay)`

`setWaitForIdleTimeout(long timeout)`

`setWaitForSelectorTimeout(long timeout)`

UiAutomation

Public methods

`executeShellCommand(String command)`

`getRootInActiveWindow()`

`grantRuntimePermission(String packageName, String permission)`

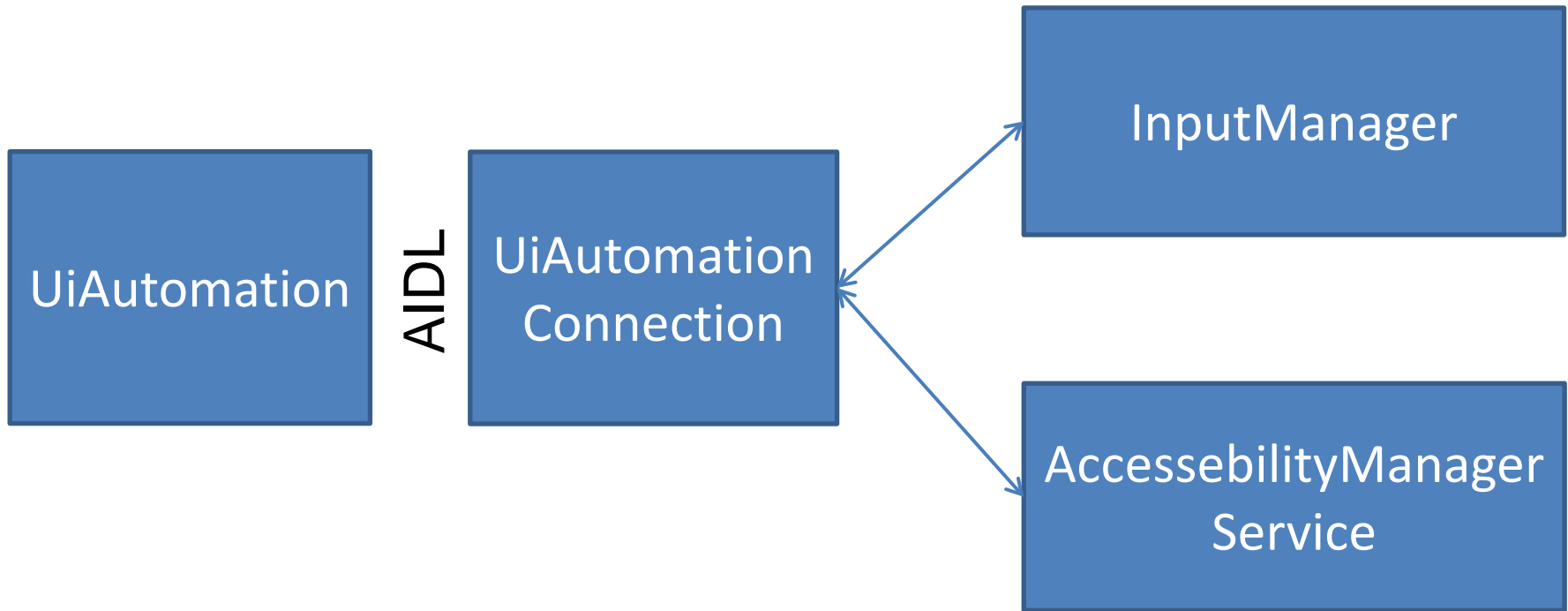
`injectInputEvent(InputEvent event, boolean sync)`

`waitForIdle(long idleTimeoutMillis, long globalTimeoutMillis)`

UiAutomation vs Instrumentation

- `UiAutomation.injectInputEvent()` работает для всех приложений
- `Instrumentation.injectInputEvent()` работает только для тестируемого приложения

UiAutomation



Ввод текста

SDK 18<

`InputManager.injectEvent()`

SDK 19+

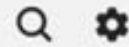
`AccessibilityNodeInfo.performAction()`

Notifications

Notifications

```
mDevice.openNotification();
```

2:26 Tue, October 8



IoT@home ▾

5

IoT@home [10번 맘카] 움직임이 감지되었습니다.

IoT@home [10번 맘카] 움직임이 감지되었습니다.

Samsung Health

0 steps

Start walking and grow your step count!

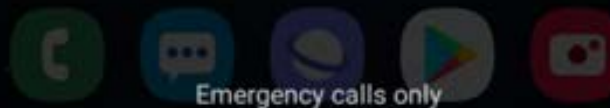


Android System USB for file transfer ▾

Settings Cable charging (7 m until fully cha... ▾

Notification settings

Clear



Подведение итогов

- UI Automator от Accessibility через View получает данные от всех приложений.
- UI Automator универсален и может тестить все
- Ui Automator v2 быстрее, но на быстро меняющихся экранах лучше Ui Automator v1

Спасибо!
Вопросы?