Технологии Java

Android: Activity и Ресурсы

Кузнецов
Андрей Николаевич

Санкт-Петербургский Государственный
Политехнический Университет

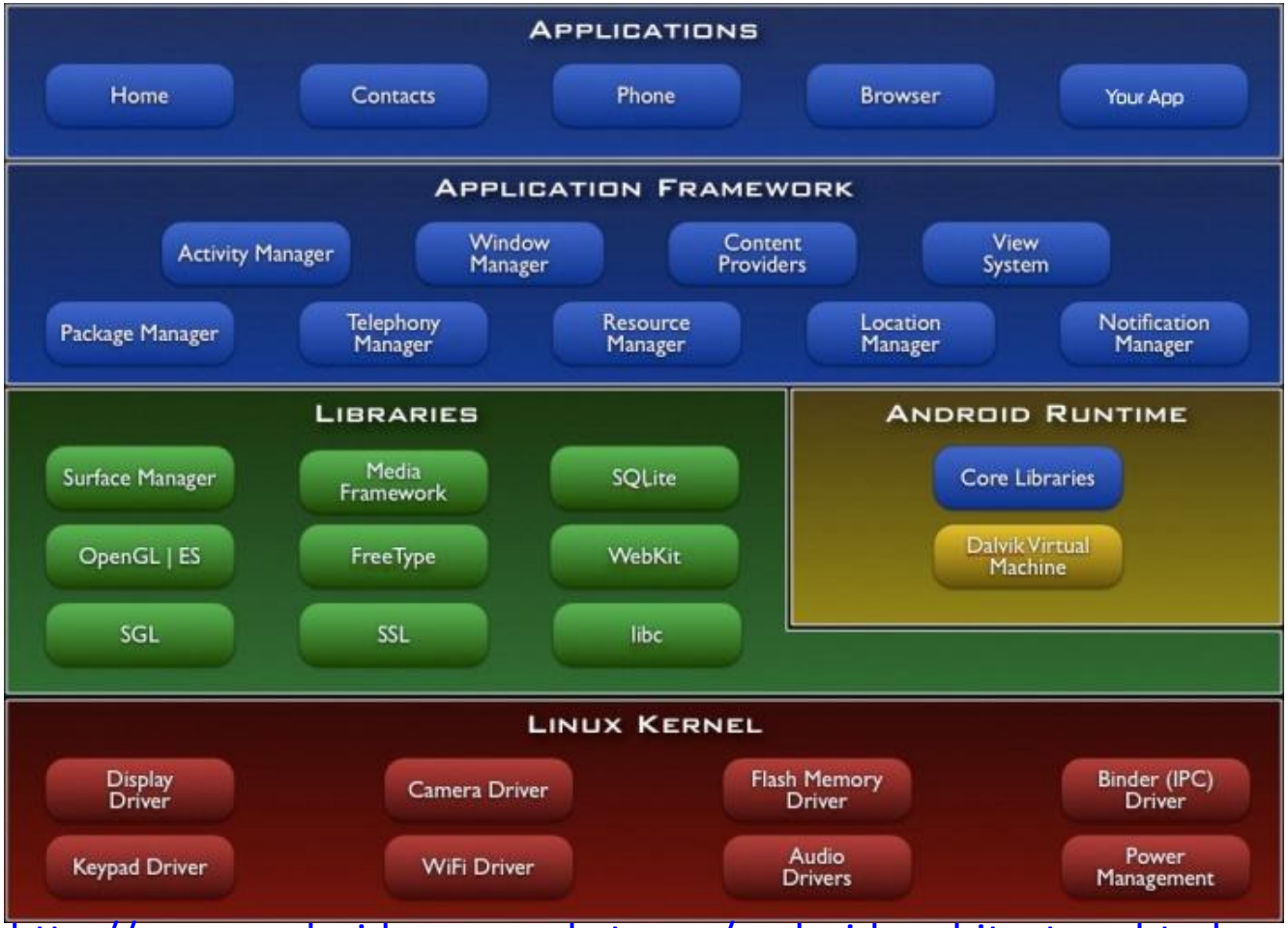
В предыдущих лекциях...

- Android SDK r21
  - http://developer.android.com/sdk/index.html
- Eclipse IDE for Mobile Developers
  - http://eclipse.org/mobile/
- ADT Plugin для Eclipse
  - https://dl-ssl.google.com/android/eclipse/
- Java SE Development Kit 7
  - http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

В предыдущих лекциях...

- Android SDK r21
  - http://developer.android.com/sdk/index.html
- Eclipse IDE for Mobile Developers
  - http://eclipse.org/mobile/
- ADT Plugin для Eclipse
  - https://dl-ssl.google.com/android/eclipse/
- Java SE Development Kit 7
  - http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

В предыдущих лекциях...



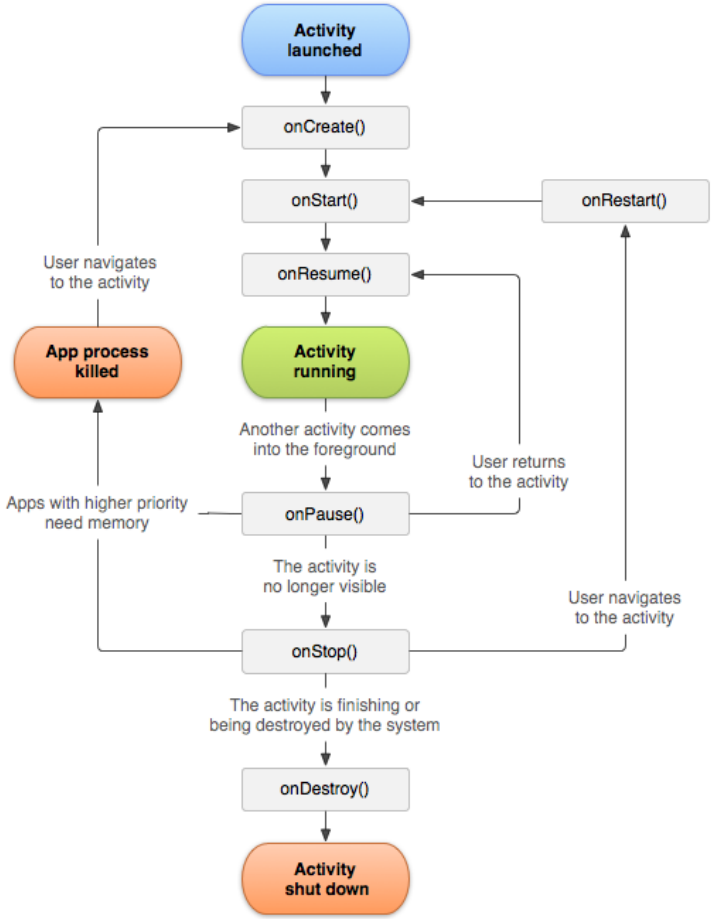
See http://www.android-app-market.com/android-architecture.html

В предыдущих лекциях...

- Activities
- Services
- Content Providers
- Broadcast Receivers
- Intents

As a developer we need only to call and extend these already defined classes to use in our application.

В предыдущих лекциях...

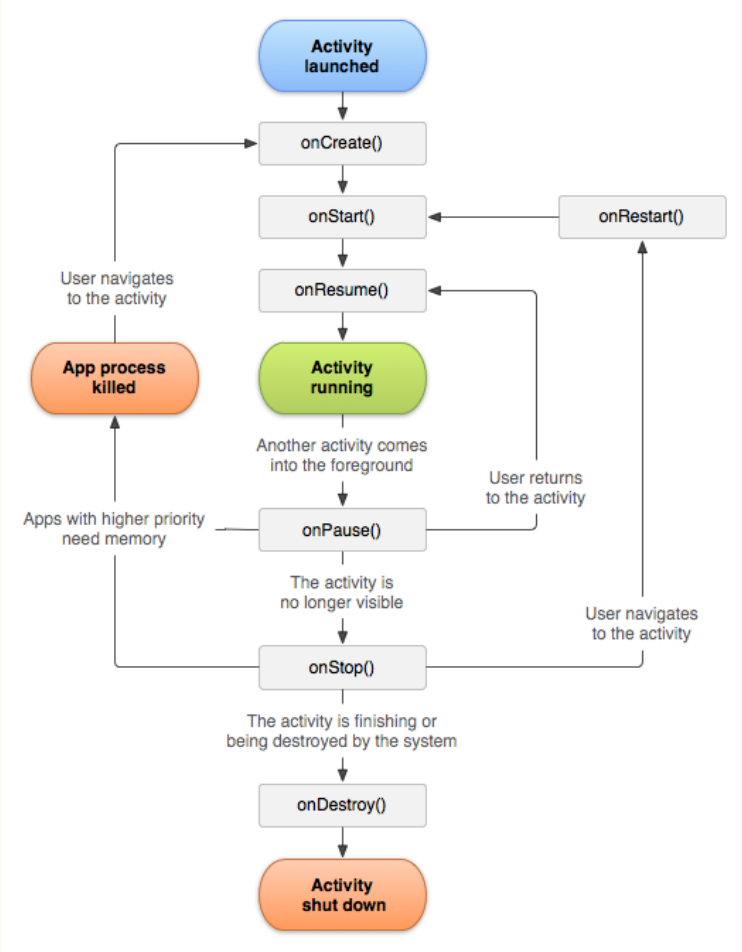


В предыдущих лекциях...

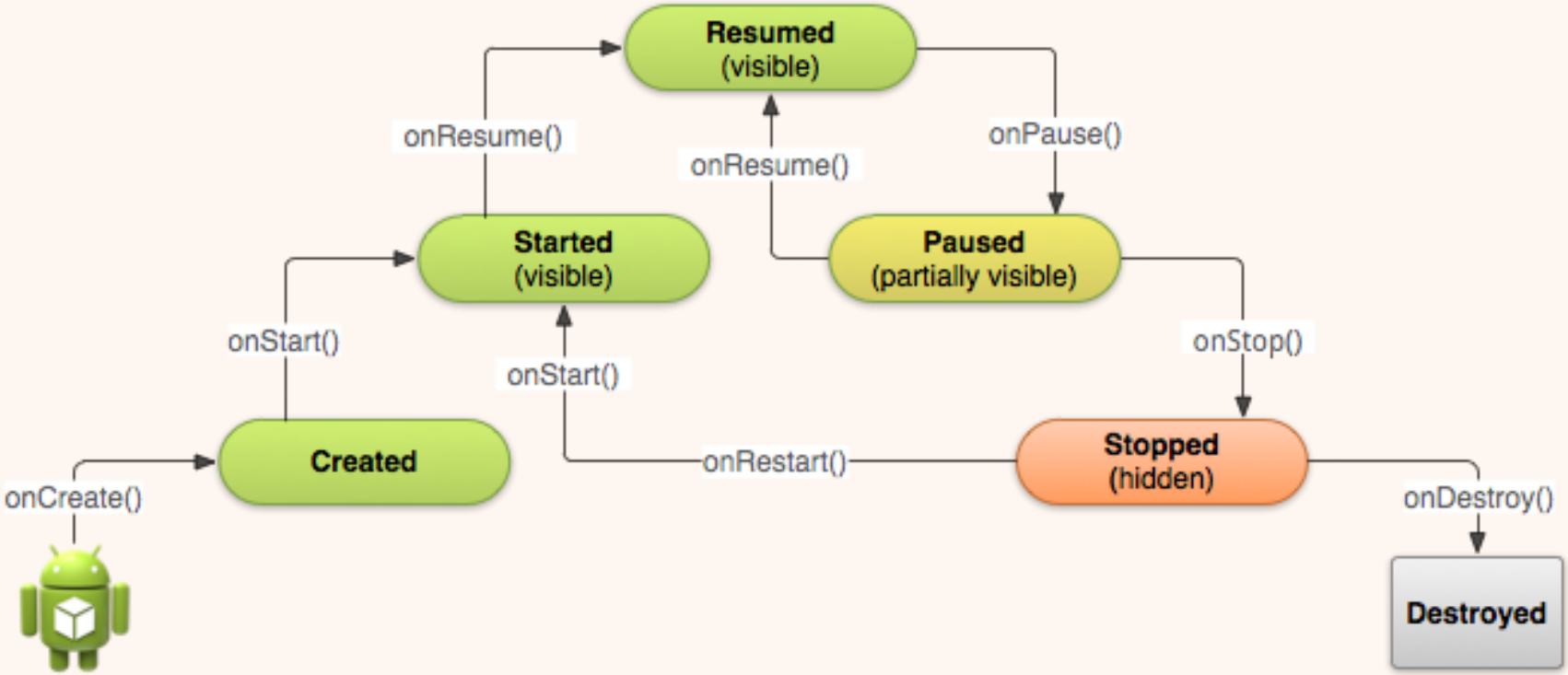
- ДЗ (?)

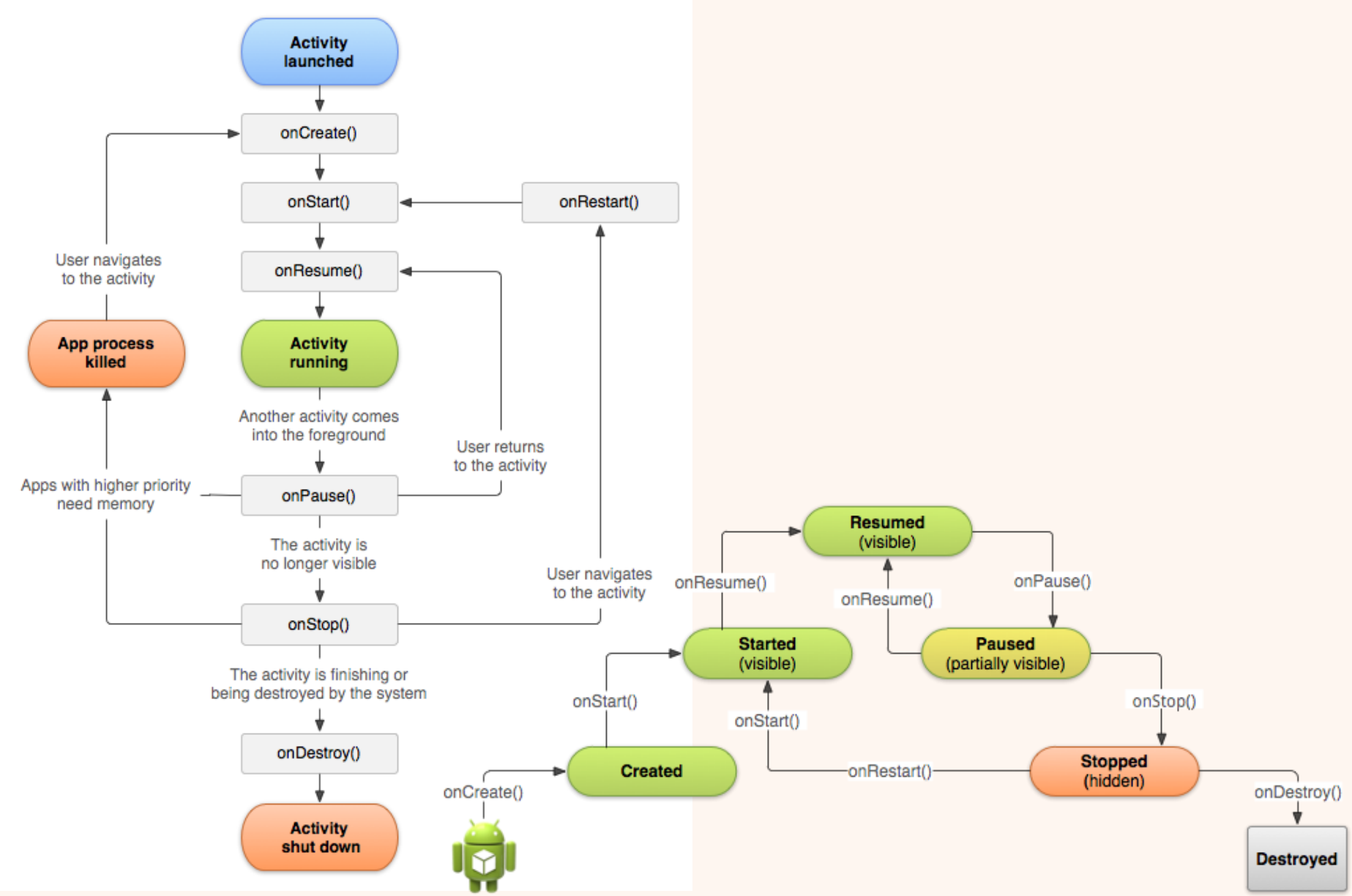
ACTIVITIES IN ANDROID

Жизненный Цикл Activity (1)



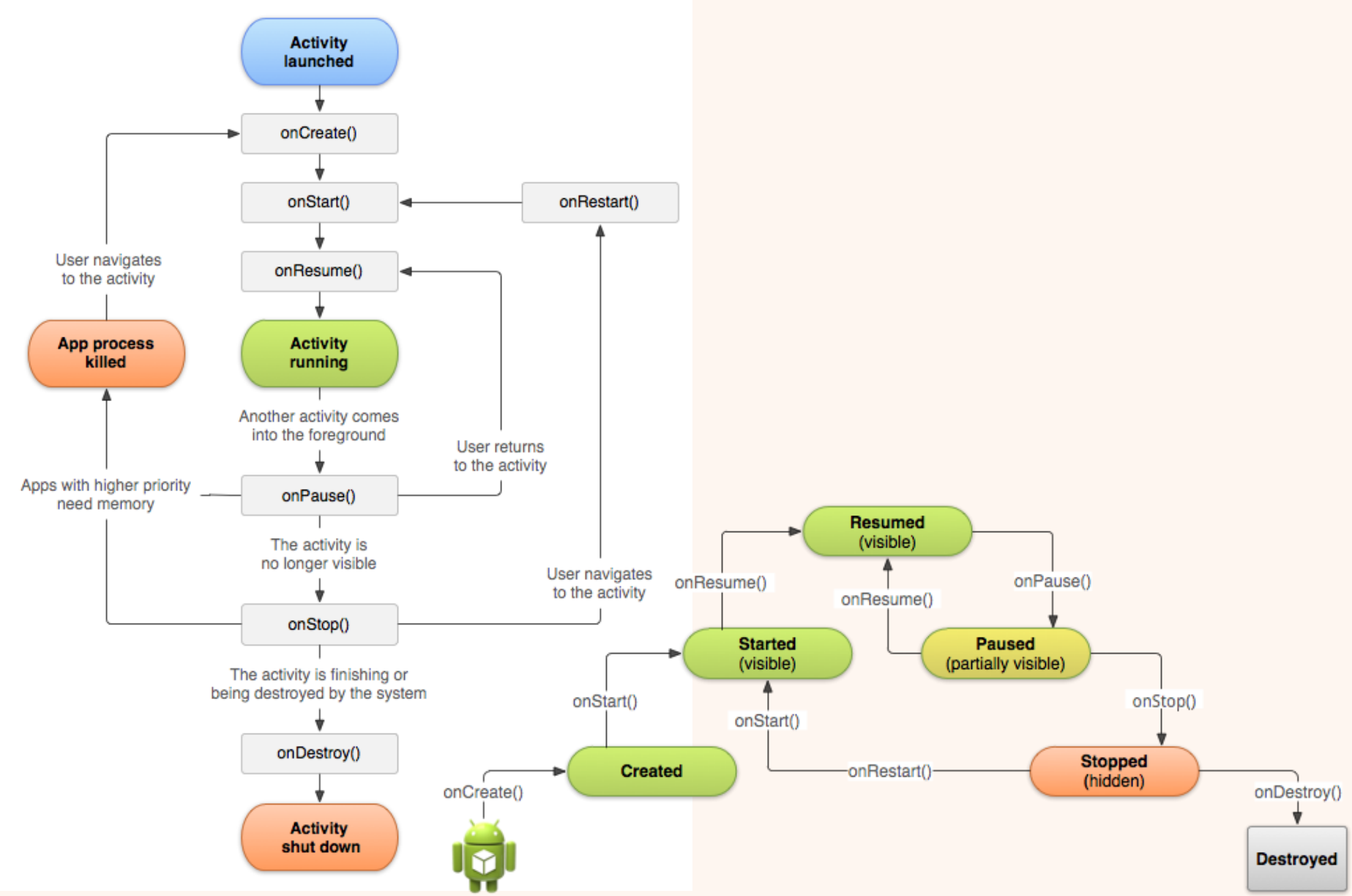
Жизненный Цикл Activity (2)





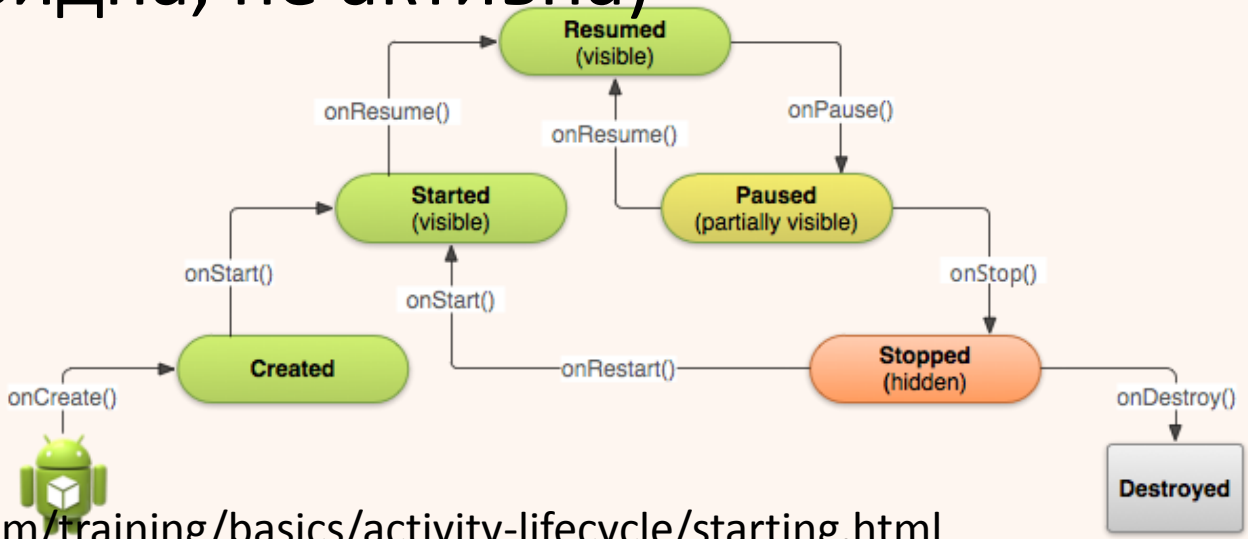
Жизненный Цикл Activity (3)

- Does not crash if the user receives a phone call or switches to another app while using your app
- Does not consume valuable system resources when the user is not actively using it
- Does not lose the user's progress if they leave your app and return to it at a later time
- Does not crash or lose the user's progress when the screen rotates between landscape and portrait orientation



Стабильные Состояния

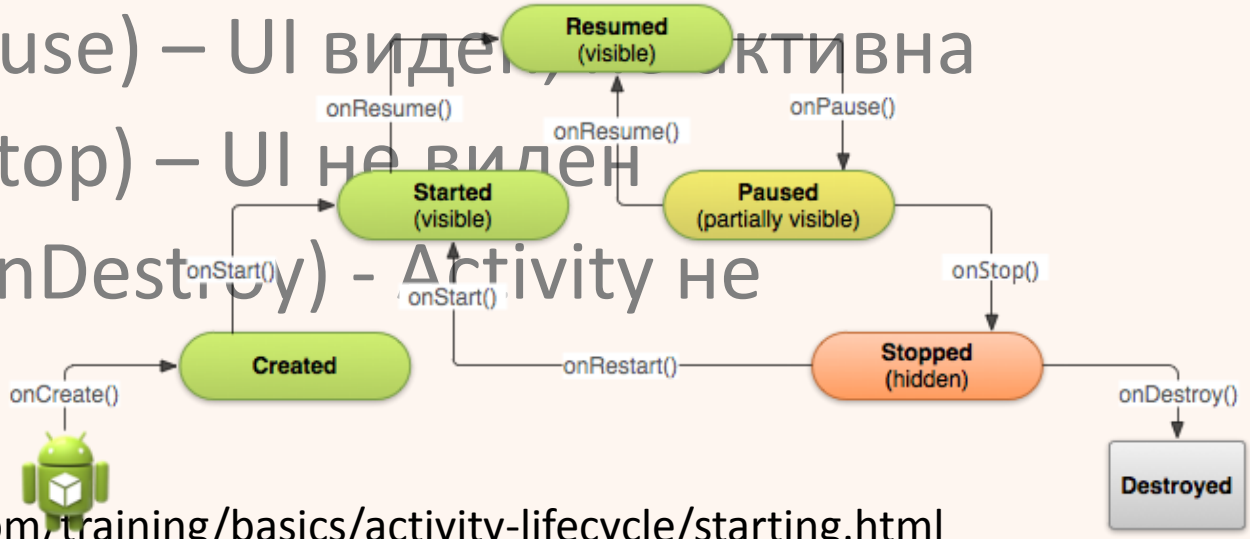
- Destroyed (не создана/разрушена)
- Resumed (Running – видна, активна)
- Paused (частично видна, не активна)
- Stopped (не видна, не активна)



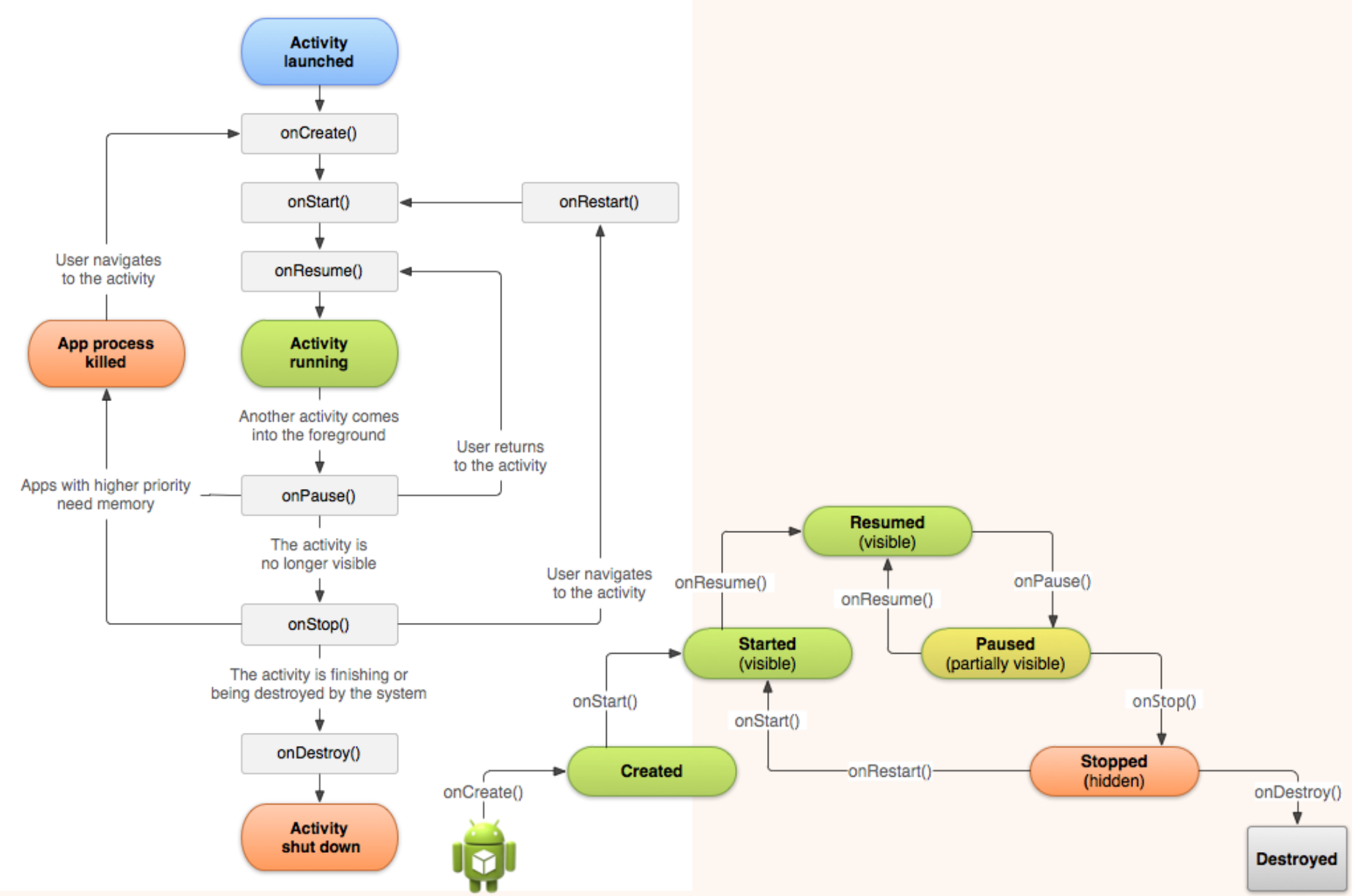
http://developer.android.com/training/basics/activity-lifecycle/starting.html

Переходные Состояния

- Destroyed () – Activity не существует
- Created (onCreate) – UI не виден
- Started (onStart) – UI виден, не активна
- Resumed (onResume) = RUNNING – активна
- Paused (onPause) – UI виден, не активна
- Stopped (onStop) – UI не виден
- Destroyed (onDestroy) - Activity не существует

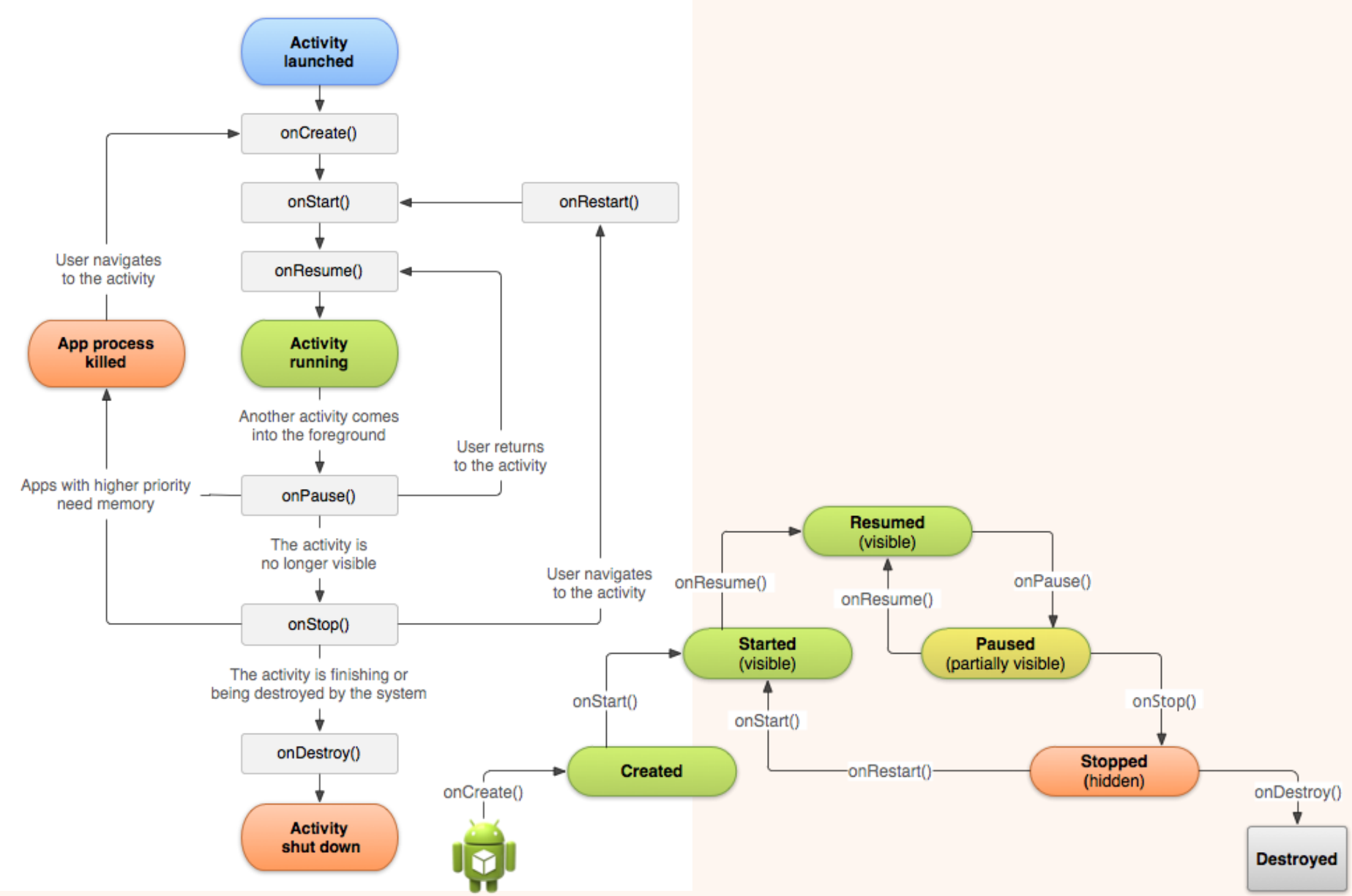


http://developer.android.com/training/basics/activity-lifecycle/starting.html



onCreate/onDestroy

- onCreate
  - Логика, выполняющаяся только 1 раз за всю жизнь Activity: создание UI, инстанциирование членов класса и т.п.
- onDestroy
  - Остановка потоков и освобождение прочих ресурсов, занятых/созданных в onCreate
  - Обычно перегружать не требуется

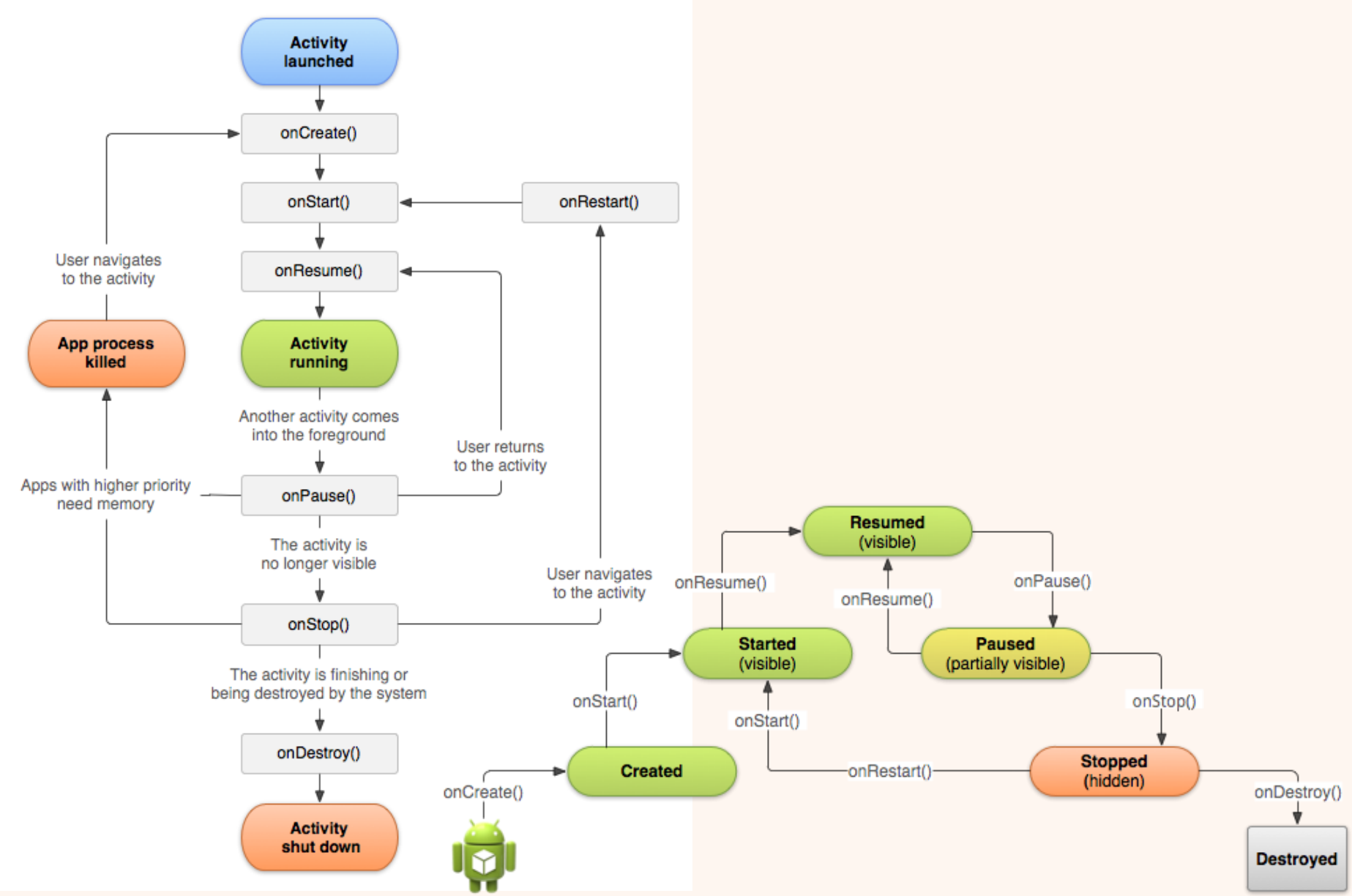


onResume/onPause

- onPause – остановить все процесы и сервисы, потребляющие CPU и батарею
  - Остановить анимацию и прочие действия потребляющие большое время CPU
  - Освободить системные ресурсы: broadcast receivers, handles to sensors (like GPS), и прочие сенсоры, которые могут сократить время жизни батареи
- В реализации метода onPause избегать ресурсоемких действий (например, запись в БД)
  - Такие действия лучше проводить в onStop

onResume/onPause

- onResume
  - Инициализировать/создать ресурсы, освобожденные в onPause (анимация, сенсоры, broadcast receivers и т.п.)
  - Инициализировать/создать ресурсы, которые должны быть обновлены перед тем, как Activity начнет работать
    - Например, обновить громкость звука

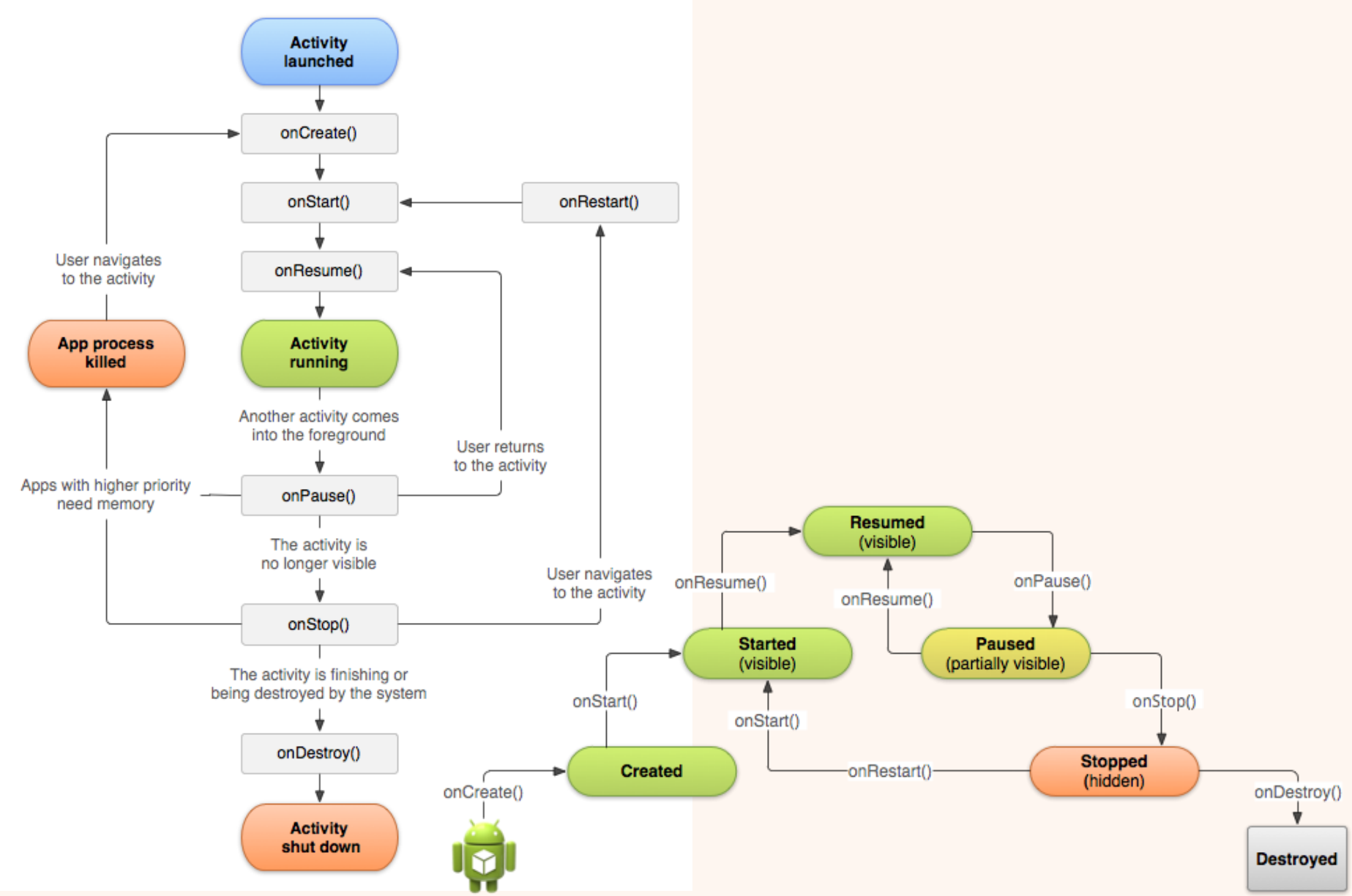


onStop/onRestart/onStart

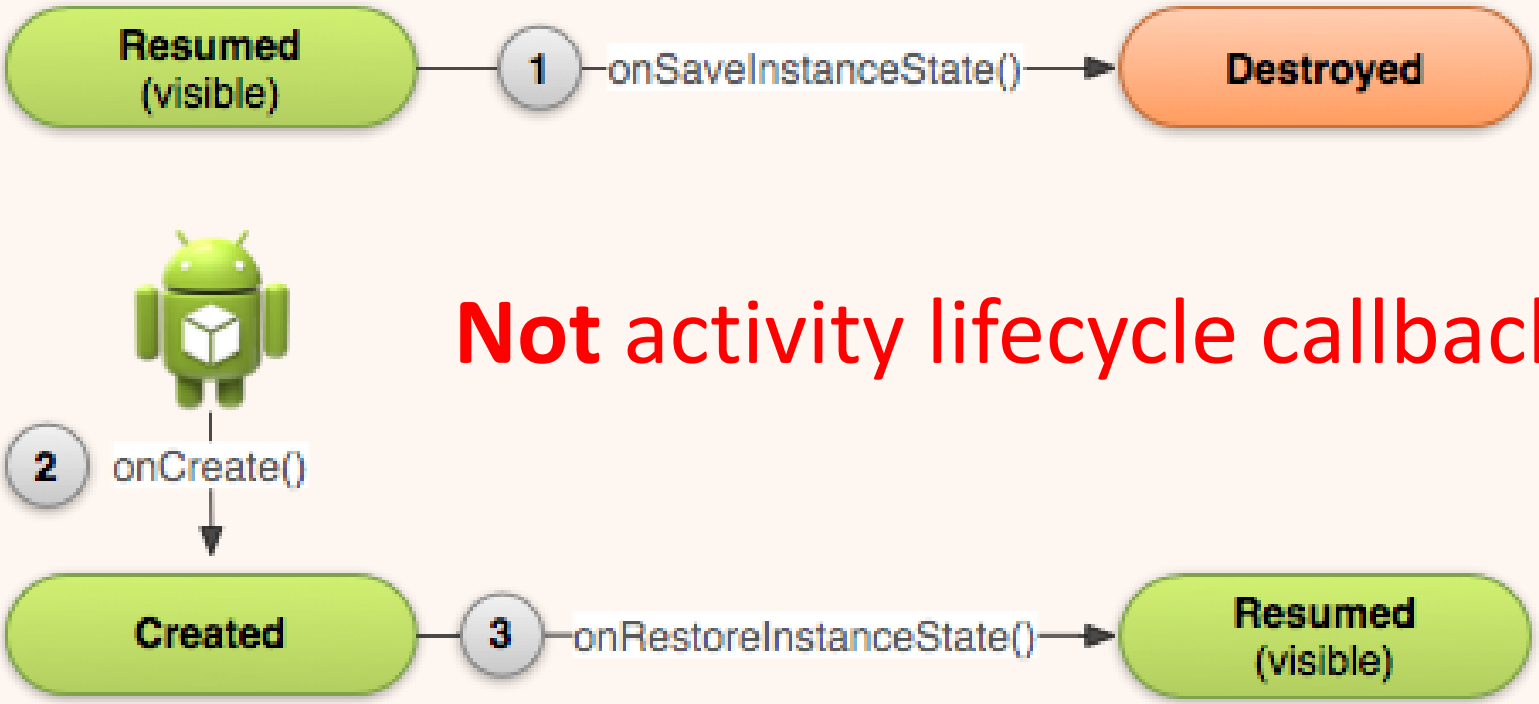
- onStop
  - Освободить все системные ресурсы, которые не требуются Activity, когда она не видна на экране
  - Выполнение больших ресурсоемких операций по остановке Activity (например, запись данных в БД)
- onStart
  - Создать/захватить ресурсы

onStop/onRestart/onStart

- onRestart
  - Используется крайне редко



Recreating an Activity



Not activity lifecycle callbacks!

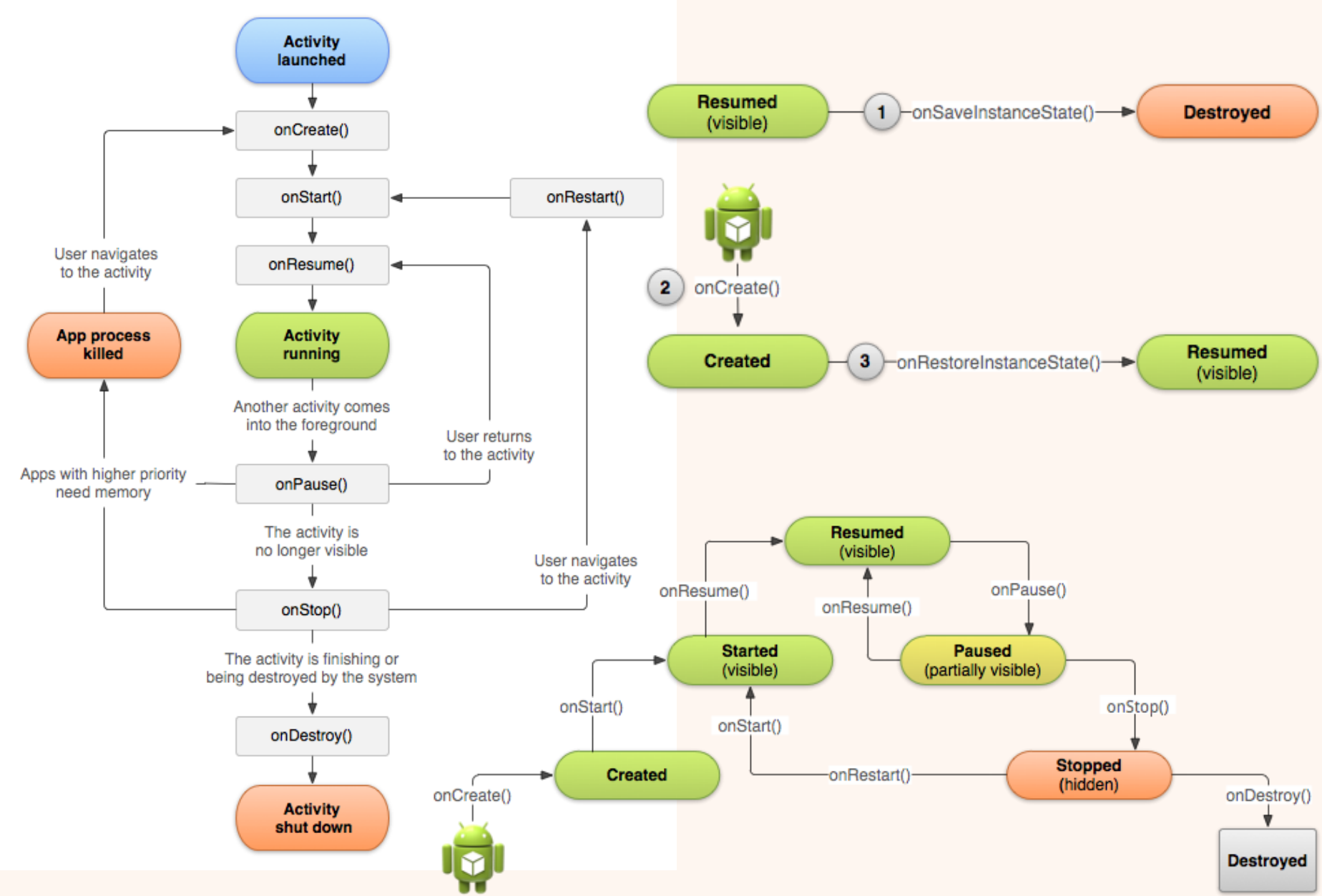
onRestoreInstanceState/ onSaveInstanceState

- onSaveInstanceState (Bundle outState)
  - Called to retrieve per-instance state from an activity before being killed so that the state can be restored in onCreate(Bundle) or onRestoreInstanceState(Bundle)
  - If called, this method will occur before onStop()

onRestoreInstanceState/ onSaveInstanceState

- onRestoreInstanceState (Bundle savedState)
  - This method is called after onStart() when the activity is being re-initialized from a previously saved state
  - Most implementations will simply use onCreate(Bundle) to restore their state

При перегрузке любого метода сначала вызвать метод базового класса!



RESOURCES IN ANDROID

Типы ресурсов (1)

- Все ресурсы хранятся в каталоге “res”
- Тип ресурса определяется **именем подкаталога**

```
MyProject/
    src/
        MyActivity.java
    res/
        drawable/
            icon.png
        layout/
            main.xml
            info.xml
        values/
            strings.xml
```

Типы ресурсов (2)

- ./animator/*
- ./anim/*
- ./xml/*
- ./drawable/*
  - Bitmap files (png, 9.png, jpg, gif)
  - State lists
  - Shapes
  - Other drawables

Типы ресурсов (3)

- ./layout/*
- ./menu/*
- ./raw/*
- ./values/*
  - arrays.xml
  - colors.xml
  - dimens.xml
  - strings.xml
  - styles.xml

Drawable

- Bitmap files (png, 9.png, jpg, gif)
  - See \Android\android-sdk\tools\draw9patch.bat
- State lists
- Shapes

Drawable: State Lists

- Layer List
  - A Drawable that manages an array of other Drawables. These are drawn in array order, so the element with the largest index is be drawn on top.
- State List
  - An XML file that references different bitmap graphics for different states (for example, to use a different image when a button is pressed)
- Level List
  - An XML file that defines a drawable that manages a number of alternate Drawables, each assigned a maximum numerical value

Drawable: Layer List

```
<?xml version="1.0" encoding="utf-8"?>
<layer-list
   xmlns:android="http://schemas.android.com/apk/res/android" >
   <item
      android:drawable="@[package:]drawable/drawable_resource"
      android:id="@[+][package:]id/resource_name"
      android:top="dimension"
      android:right="dimension"
      android:bottom="dimension"
      android:left="dimension" />
</layer-list>
```

Drawable: State List (1)

```
<?xml version="1.0" encoding="utf-8"?>
<selector
xmlns:android="http://schemas.android.com/apk/res/android"
   android:constantSize=["true" | "false"]
   android:dither=["true" | "false"]
   android:variablePadding=["true" | "false"] >
   <item
      android:drawable="@[package:]drawable/drawable_resource"
      android:state_pressed=["true" | "false"]
      android:state_focused=["true" | "false"]
      android:state_hovered=["true" | "false"]
      android:state_selected=["true" | "false"]
      android:state_checkable=["true" | "false"]
      android:state_checked=["true" | "false"]
      android:state_enabled=["true" | "false"]
      android:state_activated=["true" | "false"]
      android:state_window_focused=["true" | "false"] />
</selector>
```

Drawable: State List (2)

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
   <item android:state_pressed="true"
       android:drawable="@drawable/button_pressed" /> <!-- pressed -->
   <item android:state_focused="true"
       android:drawable="@drawable/button_focused" /> <!-- focused -->
   <item android:state_hovered="true"
       android:drawable="@drawable/button_focused" /> <!-- hovered -->
   <item android:drawable="@drawable/button_normal" /> <!-- default -->
</selector>
```

Drawable: Level List (1)

```
<?xml version="1.0" encoding="utf-8"?>
<level-list
   xmlns:android="http://schemas.android.com/apk/res/android" >
   <item
      android:drawable="@drawable/drawable_resource"
      android:maxLevel="integer"
      android:minLevel="integer" />
</level-list>
```

Drawable: Level List (2)

```
<?xml version="1.0" encoding="utf-8"?>
<level-list xmlns:android="http://schemas.android.com/apk/res/android" >
   <item
      android:drawable="@drawable/status_off"
      android:maxLevel="0" />
   <item
      android:drawable="@drawable/status_on"
      android:maxLevel="1" />
</level-list>
```

Drawable: Other (1)

- Transition Drawable
  - An XML file that defines a drawable that can cross-fade between two drawable resources
- Inset Drawable
  - An XML file that defines a drawable that insets another drawable by a specified distance. This is useful when a View needs a background drawble that is smaller than the View's actual bounds.

Drawable: Other (2)

- Clip Drawable
  - An XML file that defines a drawable that clips another Drawable based on this Drawable's current level value
- Scale Drawable
  - An XML file that defines a drawable that changes the size of another Drawable based on its current level value

Drawable: Shape (1)

```
<?xml version="1.0" encoding="utf-8"?>
<shape
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:shape=["rectangle" | "oval" | "line" | "ring"] >
   <corners
      android:radius="integer"
      android:topLeftRadius="integer"
      android:topRightRadius="integer"
      android:bottomLeftRadius="integer"
      android:bottomRightRadius="integer" />
   <gradient
      android:angle="integer"
      android:centerX="integer"
      android:centerY="integer"
      android:centerColor="integer"
      android:endColor="color"
      android:gradientRadius="integer"
      android:startColor="color"
      android:type=["linear" | "radial" | "sweep"]
      android:useLevel=["true" | "false"] />
```

Drawable: Shape (2)

```
    <padding
        android:left="integer"
        android:top="integer"
        android:right="integer"
        android:bottom="integer" />
    <size
        android:width="integer"
        android:height="integer" />
    <solid
        android:color="color" />
    <stroke
        android:width="integer"
        android:color="color"
        android:dashWidth="integer"
        android:dashGap="integer" />

</shape>
```

Drawable: Shape (3)

```
<?xml version="1.0" encoding="utf-8"?>
<shape
xmlns:android="http://schemas.android.com/apk/res/android"
   android:shape="rectangle">
   <gradient
      android:startColor="#FFFF0000"
      android:endColor="#80FF00FF"
      android:angle="45"/>
   <padding android:left="7dp"
      android:top="7dp"
      android:right="7dp"
      android:bottom="7dp" />
   <corners android:radius="8dp" />
</shape>
```

Values

- String, String Array, Quantity Strings (Plurals)
- Style
- Bool
- Color
- Dimesion
- ID
- Integer
- Integer Array
- Typed Array

Values: String (1)

- String
  - XML resource that provides a single string
- String Array
  - XML resource that provides an array of strings
- Quantity Strings (Plurals)
  - XML resource that carries different strings for pluralization

Values: String (2)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <string
      name="string_name"
      >text_string</string>
</resources>
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <string name="hello">Hello!</string>
</resources>
```

Values: String Array (1)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <string-array
      name="string_array_name">
      <item
         >text_string</item>
   </string-array>
</resources>
```

Values: String Array (2)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="planets_array">
        <item>Mercury</item>
        <item>Venus</item>
        <item>Earth</item>
        <item>Mars</item>
    </string-array>
</resources>
```

Values: Quantity Strings (Plurals) (1)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <plurals
        name="plural_name">
        <item
            quantity=["zero" | "one" | "two" | "few" |
"many" | "other"]
            >text_string</item>
    </plurals>
</resources>
```

Values: Quantity Strings (Plurals) (2)

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <plurals name="numberOfSongsAvailable">
        <item quantity="one">One song found.</item>
        <item quantity="other">%d songs found.</item>
    </plurals>

</resources>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <plurals name="numberOfSongsAvailable">
      <item quantity="one">Znaleziono jedną piosenkę.</item>
      <item quantity="few">Znaleziono %d piosenki.</item>
      <item quantity="other">Znaleziono %d piosenek.</item>
   </plurals>
</resources>
```

Values (other)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <bool name="screen_small">true</bool>
    <color name="translucent_red">#80ff0000</color>
    <item type="id" name="button_ok" />
    <integer name="max_speed">75</integer>
    <integer-array name="bits">
        <item>4</item>
        <item>8</item>
    </integer-array>
    <array name="icons">
        <item>@drawable/home</item>
        <item>@drawable/settings</item>
    </array>
    <array name="colors">
        <item>#FFFF0000</item>
        <item>#FF00FF00</item>
    </array>
</resources>
```

Values: Dimension (1)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="textview_height">25dp</dimen>
    <dimen name="textview_width">150dp</dimen>
    <dimen name="ball_radius">30dp</dimen>
    <dimen name="font_size">16sp</dimen>

</resources>
```

Values: Dimension (2)

- “dp”
  - Density-independent Pixels - An abstract unit that is based on the physical density of the screen.
- “sp”
  - Scale-independent Pixels - This is like the dp unit, but it is also scaled by the user's font size preference.
- “pt”
  - Points - 1/72 of an inch.

Values: Dimension (3)

- “px”
  - Pixels
- “mm”
  - Millimeters
- “in”
  - Inches

Layout

```
<?xml version="1.0" encoding="utf-8"?>
<ViewGroup xmlns:android="http://schemas.android.com/apk/res/android"
   android:id="@[+][package:]id/resource_name"
   android:layout_height=["dimension" | "fill_parent" | "wrap_content"]
   android:layout_width=["dimension" | "fill_parent" | "wrap_content"]
   [ViewGroup-specific attributes] >
   <View
      android:id="@[+][package:]id/resource_name"
      android:layout_height=["dimension" | "fill_parent" |
"wrap_content"]
      android:layout_width=["dimension" | "fill_parent" | "wrap_content"]
      [View-specific attributes] >
      <requestFocus/>
   </View>
   <ViewGroup >
      <View />
   </ViewGroup>
   <include layout="@layout/layout_resource"/>
</ViewGroup>
```

Layout: Root Element

- ViewGroup
  - LinearLayout, RelativeLayout, FrameLayout, etc.
- View
  - TextView, Button, CheckBox, etc.
- <merge>
  - An alternative root element that is not drawn in the layout hierarchy

<include>

```
<include layout="@layout/layout_resource"
    android:id="@[+][package:]id/resource_name"
    android:layout_height=["dimension" |
  "fill_parent" | "wrap_content"]
    android:layout_width=["dimension" |
  "fill_parent" | "wrap_content"] />
```

- <merge>
  - An alternative root element that is not drawn in the layout hierarchy

android:id

- "@+id/*name*“
- Valid ID resource

android:layout_height and android:layout_width

Value	Description
match_parent	Sets the dimension to match that of the parent element. Added in API Level 8 to deprecate fill_parent.
fill_parent	Sets the dimension to match that of the parent element.
wrap_content	Sets the dimension only to the size required to fit the content of this element.

Menu

- http://developer.android.com/guide/topics/resources/menu-resource.html

Пример

- Eclipse