



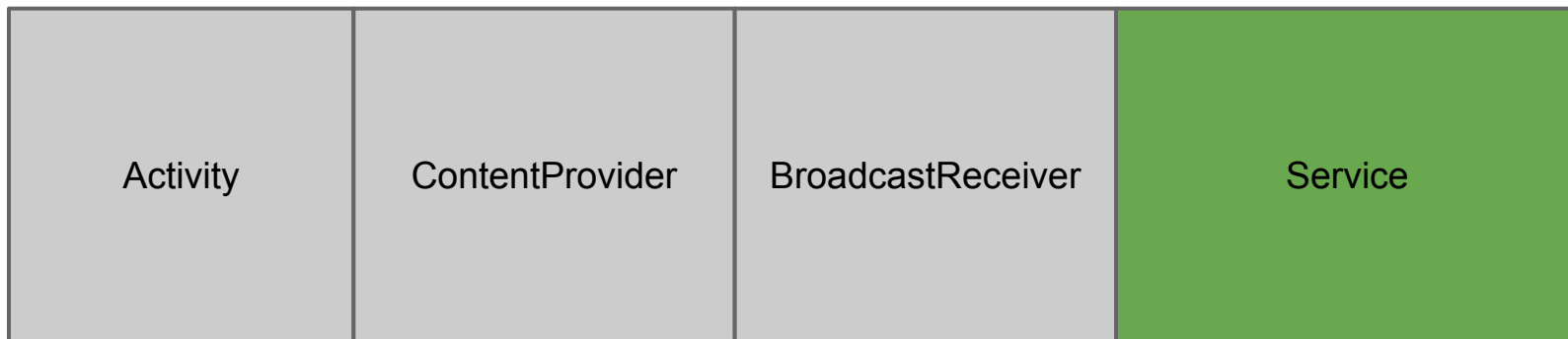
УНИВЕРСИТЕТ ИТМО

# Сервисы в Android

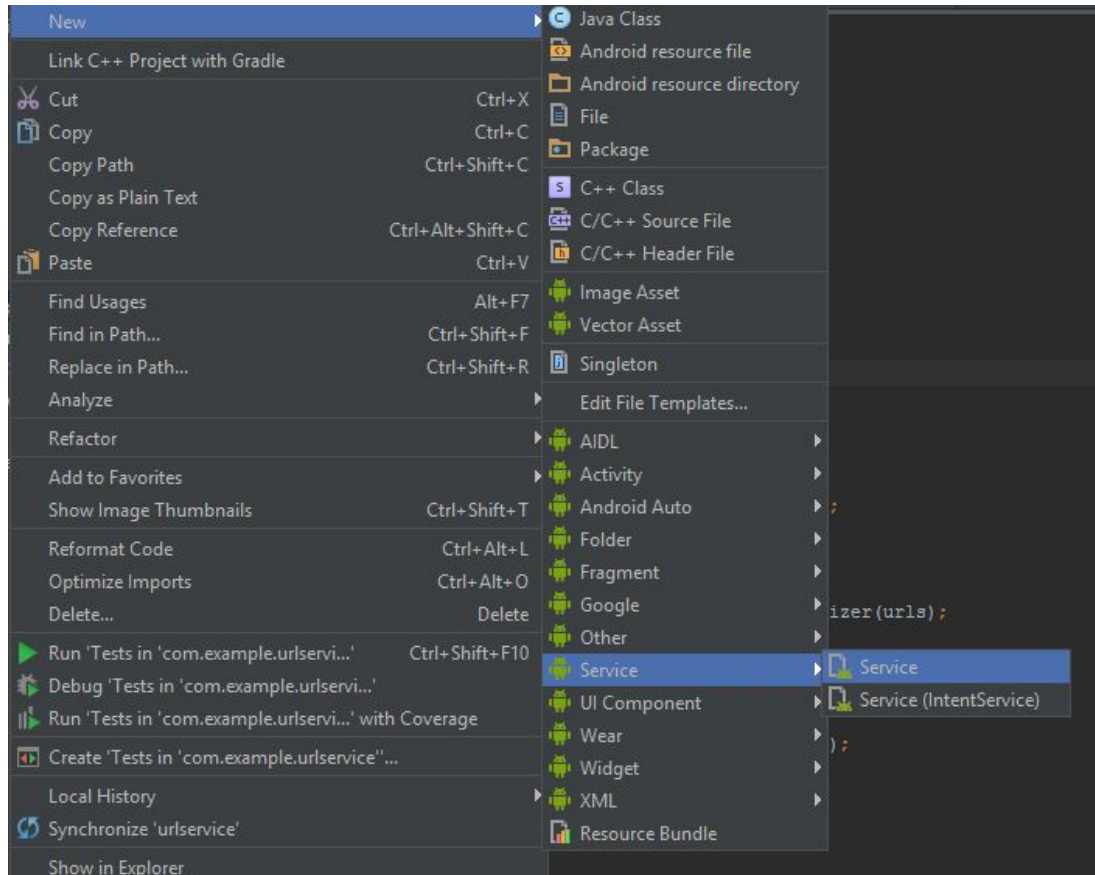
Санкт-Петербург, 2016

# Service

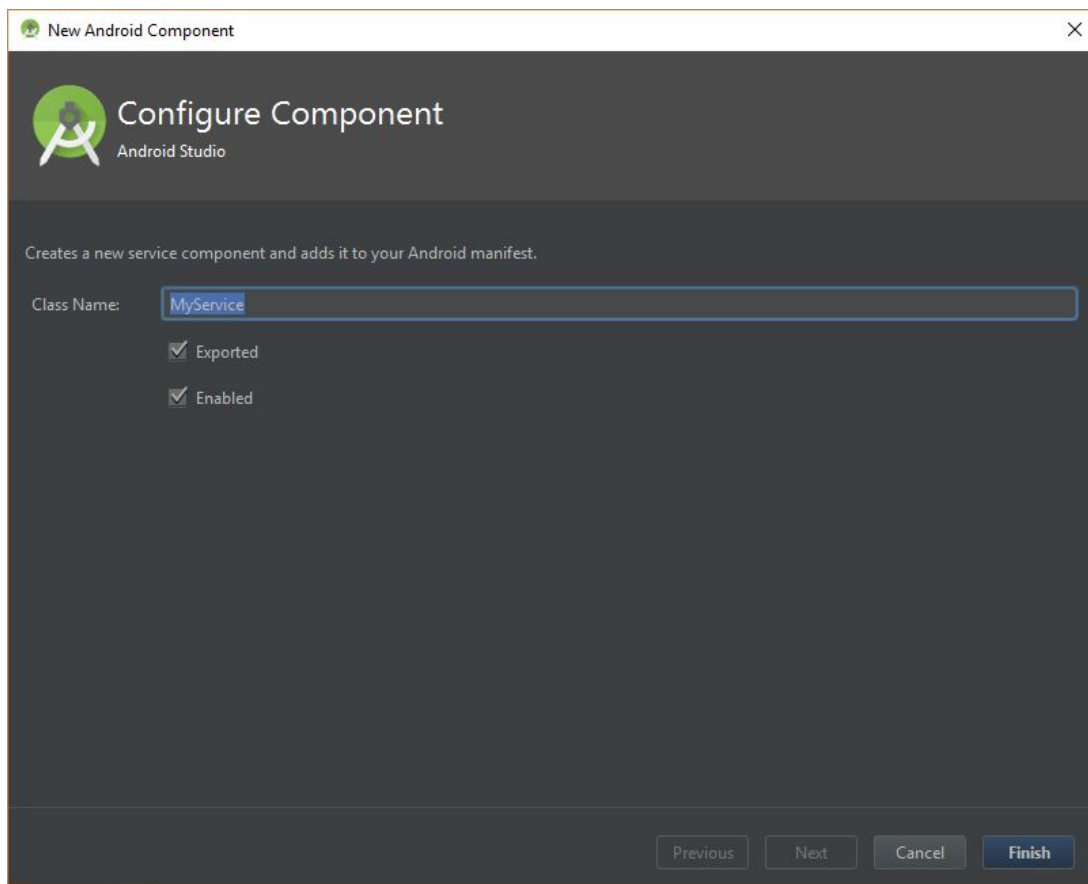
Сервис (служба) - один из четырех компонентов Android приложения, предназначенных для выполнения операции в фоновом режиме.




# Создание сервиса:



# Создание сервиса:



New Android Component

 **Configure Component**  
Android Studio

Creates a new service component and adds it to your Android manifest.

Class Name:

Exported

Enabled

# Реализация сервиса:

1.

```
import android.app.Service;
import android.content.Intent;
import android.os.IBinder;

public class MyService extends Service {
    public MyService() {
    }

    @Override
    public IBinder onBind(Intent intent) {
        // TODO: Return the communication channel to the service.
        throw new UnsupportedOperationException("Not yet implemented");
    }
}
```

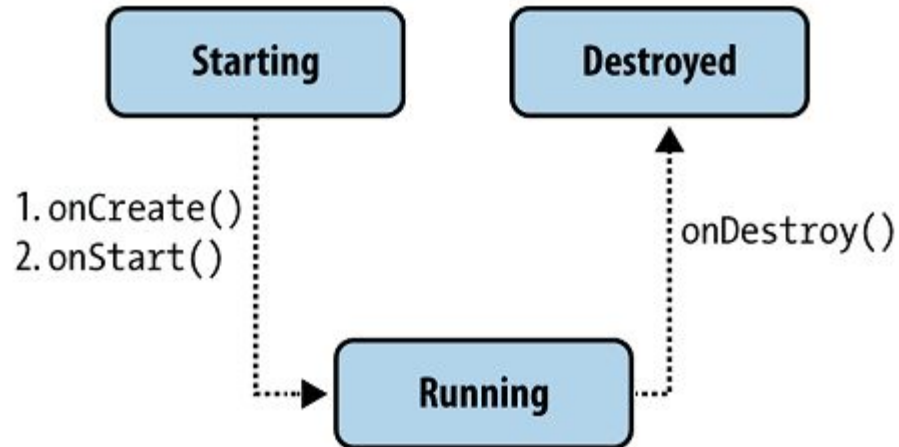
2.\*

```
<service
    android:name=".MyService"
    android:enabled="true"
    android:exported="true">
</service>
```

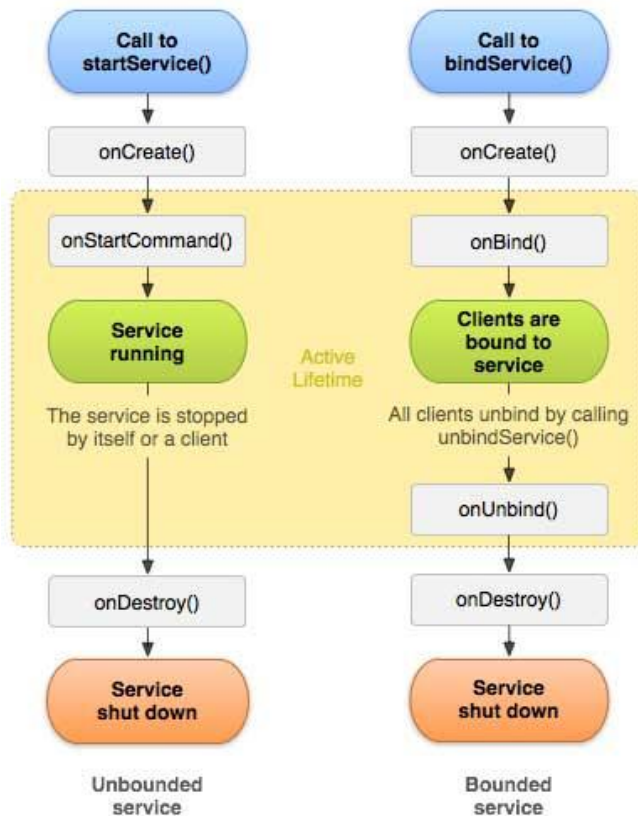
\*объявление службы в AndroidManifest.xml

# Методы жизненного цикла:

- onCreate()
- onStartCommand()
- onDestroy()

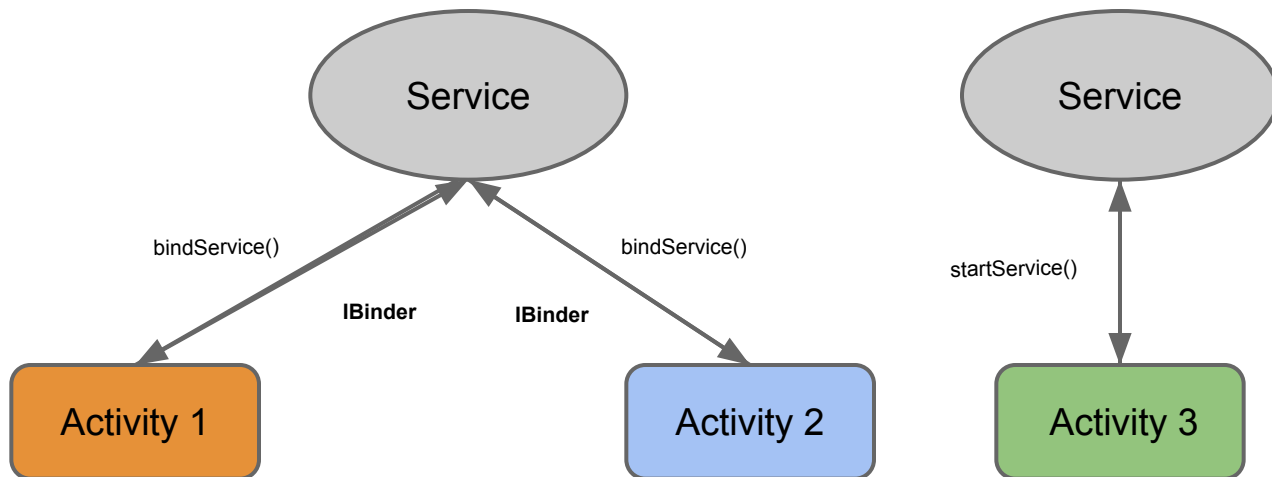


# Жизненный цикл служб



# Условно есть два вида взаимодействия с сервисом:

- `startService()`
- `bindService()`





# Внимание!

- **Служба** работает в основном потоке ведущего процесса — служба **не** создает своего потока и **не** выполняется в отдельном процессе (если вы не указали иное). Это означает, что если ваша служба собирается выполнять любую работу с высокой нагрузкой ЦП или блокирующие операции (например, воспроизведение MP3 или сетевые операции), вы должны создать в службе новый поток для выполнения этой работы. Используя отдельный поток, вы снижаете риск возникновения ошибок «Приложение не отвечает», и основной поток приложения может обрабатывать взаимодействие пользователя с вашими операциями

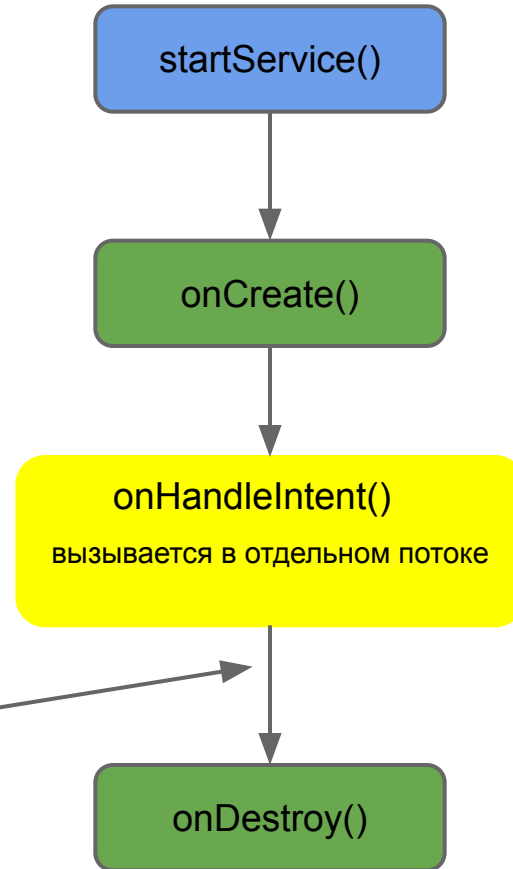
# IntentService

Это подкласс обычного Service.

-Асинхронность

-Остановка сервиса после выполнения.

завершится автоматически



# Запуск службы

- Пример для явного вызова службы с именем **HelloService**:

```
Intent intent = new Intent(this, HelloService.class);  
startService(intent);
```

- Пример для неявного вызова службы с именем **MyService**:

```
startService(new Intent(MyService.SERVICE_ACTION));
```

# Уничтожение процессов

процессы уничтожаются в особом приоритете.

[startForeground \(int id, Notification notification\).](#)

[stopForeground \(boolean removeNotification\).](#)

Activity

уничтожаются первыми

Service

Foreground  
Service

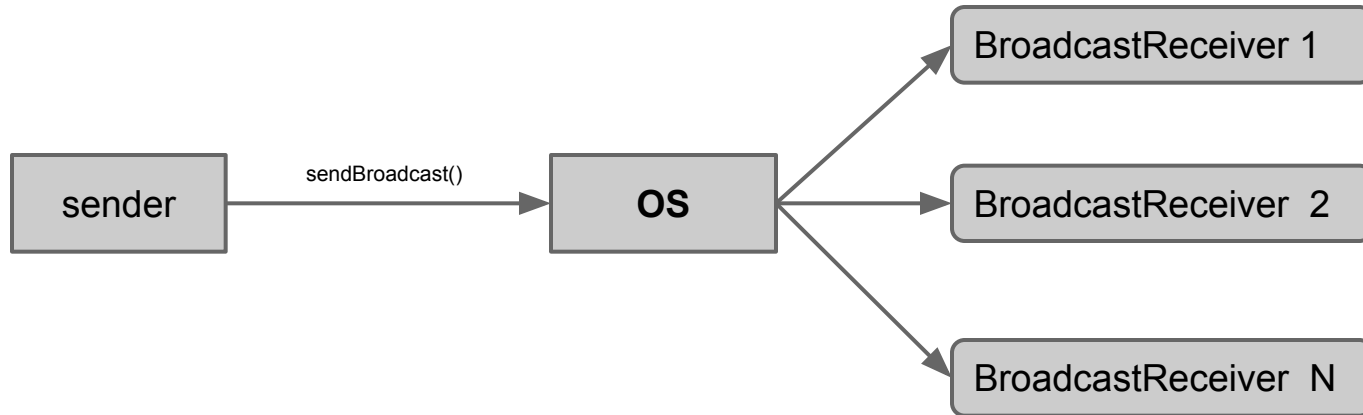
вероятность уничтожения  
стремится к 0

# BroadcastReceiver

## Широковещательные сообщения

Приёмник широковещательных сообщений — это компонент для получения внешних событий и реакции на них.

- BroadcastReceiver базовый класс для всех приемников сообщений.
- Инициализировать передачи могут другие приложения или службы



# Что важно понять

- сервис - не отдельный процесс.
- сервис выполняется в главном потоке приложения.
- сервис помогает сообщить системе, что приложение желает совершать операции в фоновом режиме.