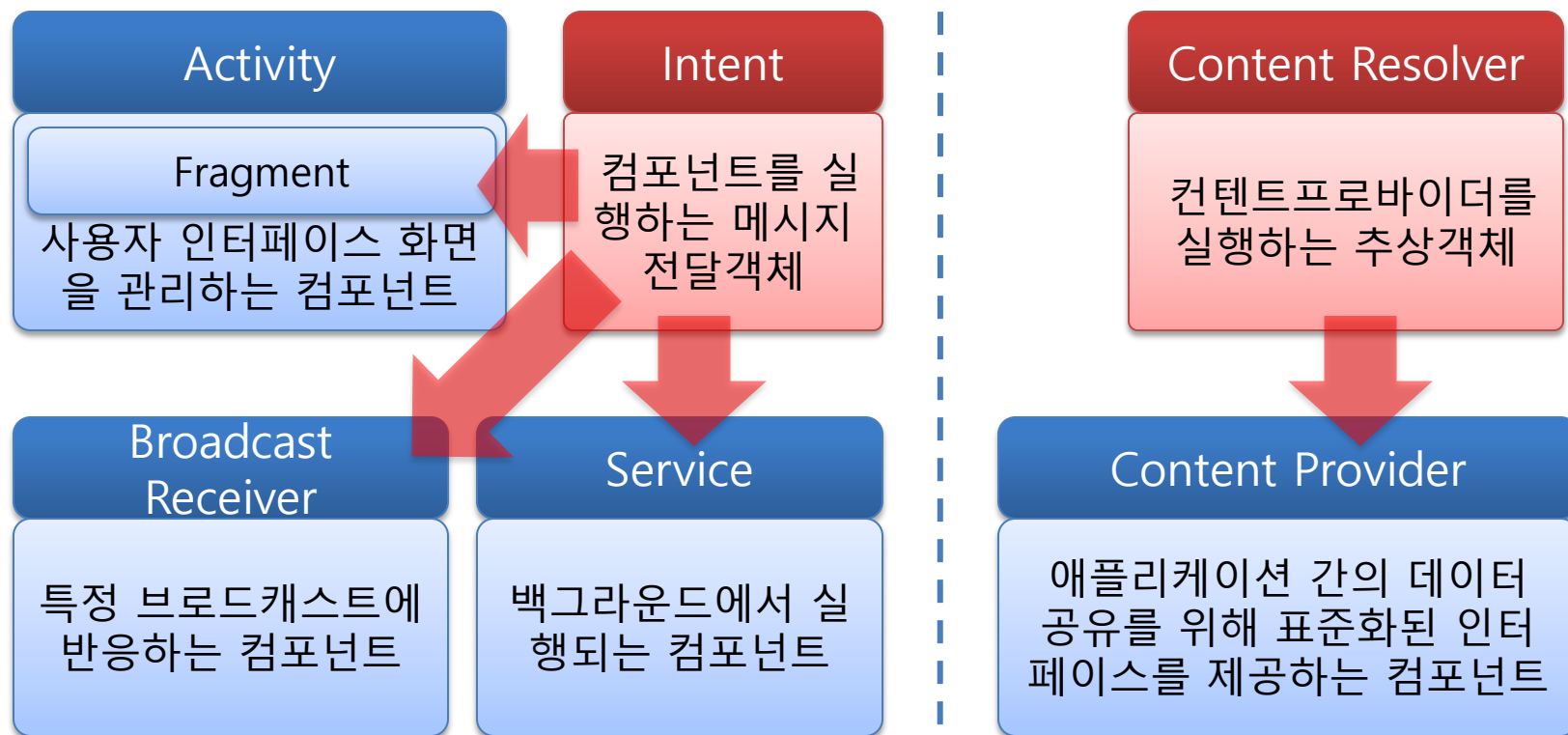


# Android 009 Service

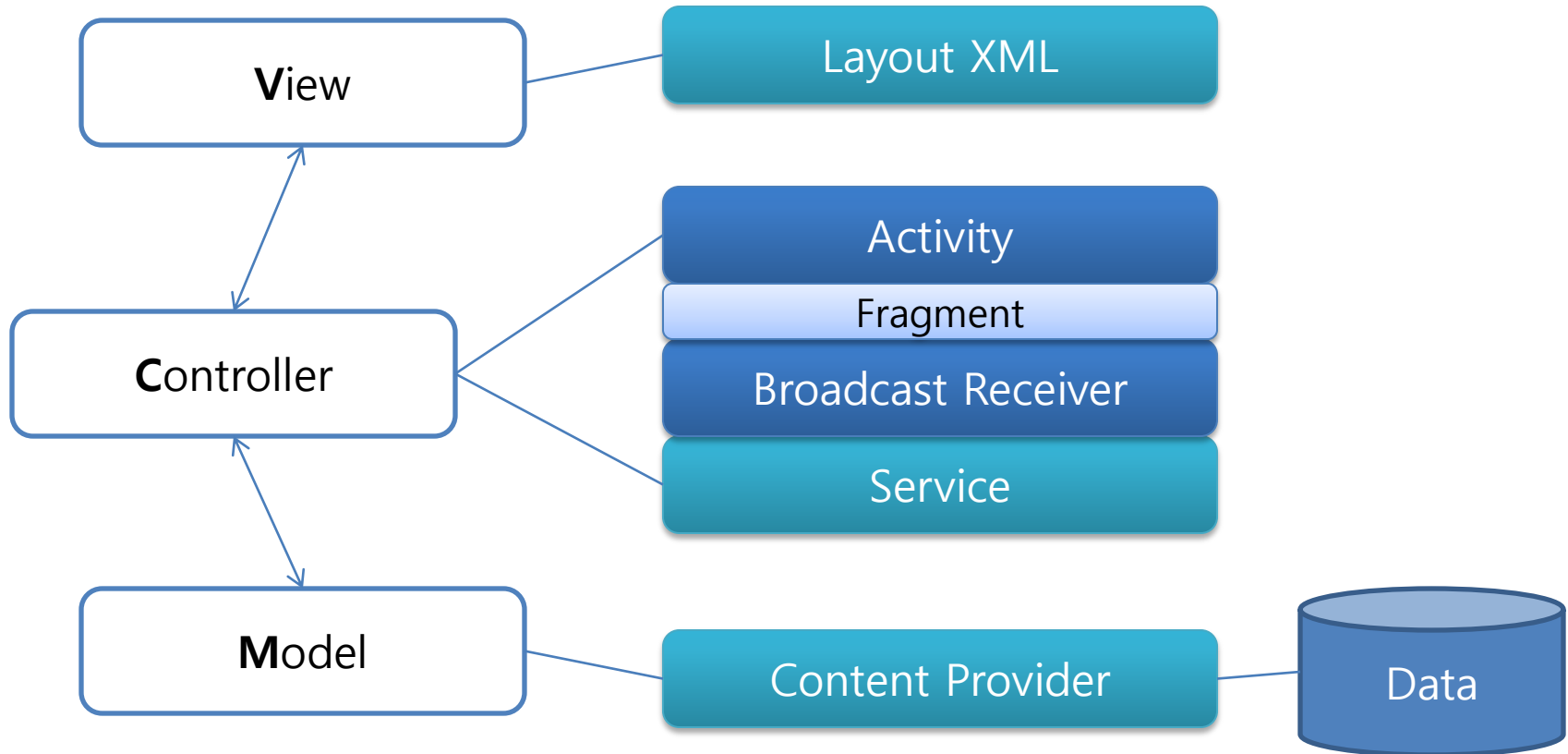
## 1. App Components

- 안드로이드는 App 을 구성하는 4개의 핵심 컴포넌트가 있으며 이들은 각각의 독립적인 Life Cycle 에 의해서 실행된다. 왼쪽의 세개의 컴포넌트는 Intent 에 의해, 나머지 한개인 Content Provider 는 Resolver에 의해 실행된다



## 2. App Architecture

- 안드로이드 어플리케이션은 기본적으로 Model(데이터)-View(화면)-Controller(제어) 의 MVC 아키텍처로 구성된다



## 3. Service Life Cycle

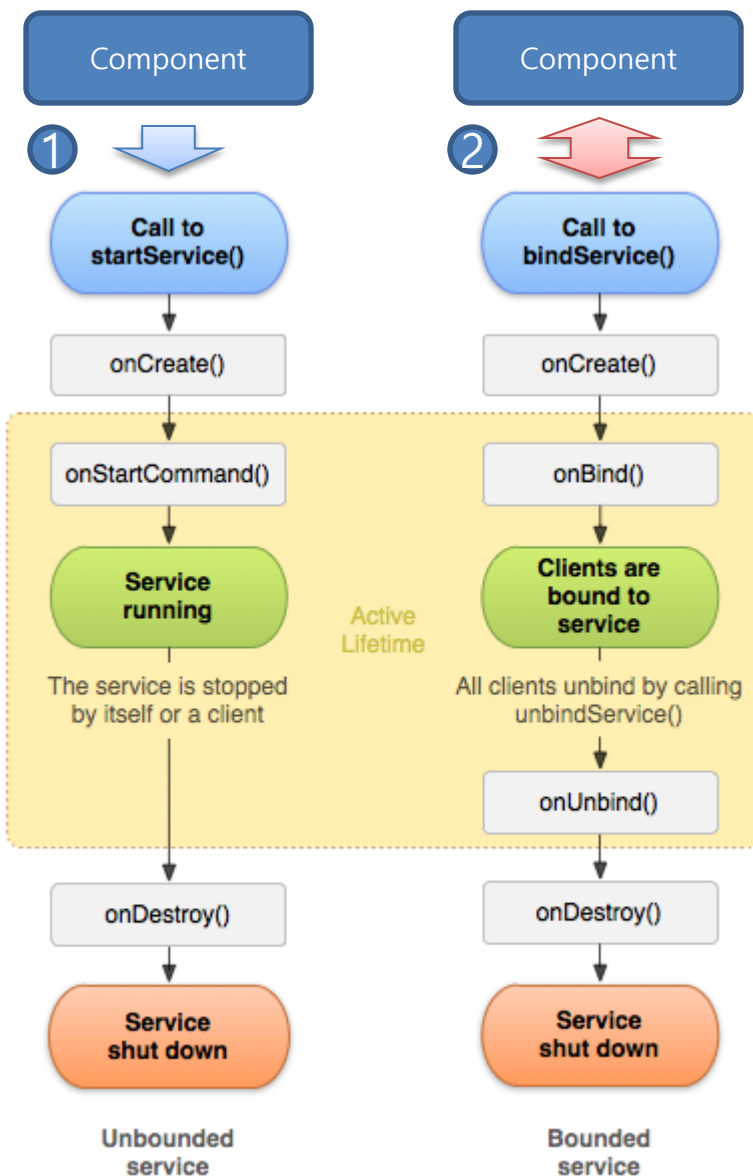
- 서비스는 오른쪽 그림과 같이 두가지 형태로 제공된다

### ① Started

- `startService()` 메서드를 통해 생성된다. 일반적으로 한개의 컴포넌트에서 백그라운드 프로세스를 연결하고자 할 때 사용된다.
- `stopService()`로 종료

### ② Bound

- `bindService()` 메서드로 연결하고, 여러개의 컴포넌트에서 연결을 만들 수 있다. 주로 호출측 컴포넌트와 bind 되어 상호작용이 있을경우 사용된다.



## 3.1. Service Life Cycle

### \* 주의사항

안드로이드의 서비스는 기본적으로 Main Thread에서 실행된다.

따라서 CPU 사용량이 많은 로직을 가진 서비스의 경우 내부에서 별도로 Thread를 생성해서 사용해야지만 화면지연 등의 오동작 없이 정상적으로 동작하게 된다

### 1) Started Service

method	설명
onCreate	서비스 생성
onStartCommand	서비스 시작
onDestroy	서비스 종료

### 2) Bound Service

method	설명
onCreate	서비스 생성
onBind	서비스 연결
onUnbind	서비스 연결해제
onRebind	재연결
onDestroy	서비스 종료

## 3.2. Service Code Example

- Started Service

startService(Intent) 명령어로 실행

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

- Bound Service

bindService(Intent, ServiceConnection, 플래그) 명령어로 실행

```
Intent intent = new Intent(this, LocalService.class);  
bindService(intent, mConnection, Context.BIND_AUTO_CREATE);  
// 대상이 되는 서비스를 자동으로 생성하는 플래그값
```

## 4. Intent Service

- 일반적인 Service는 하나의 Main Thread 에서 실행되기 때문에 비동기로 동작시키기 위해서는 내부에 또 다른 Thread를 구현해야만 한다. 이런 점을 보완하기 위해서 나온것이 Intent Service 이다

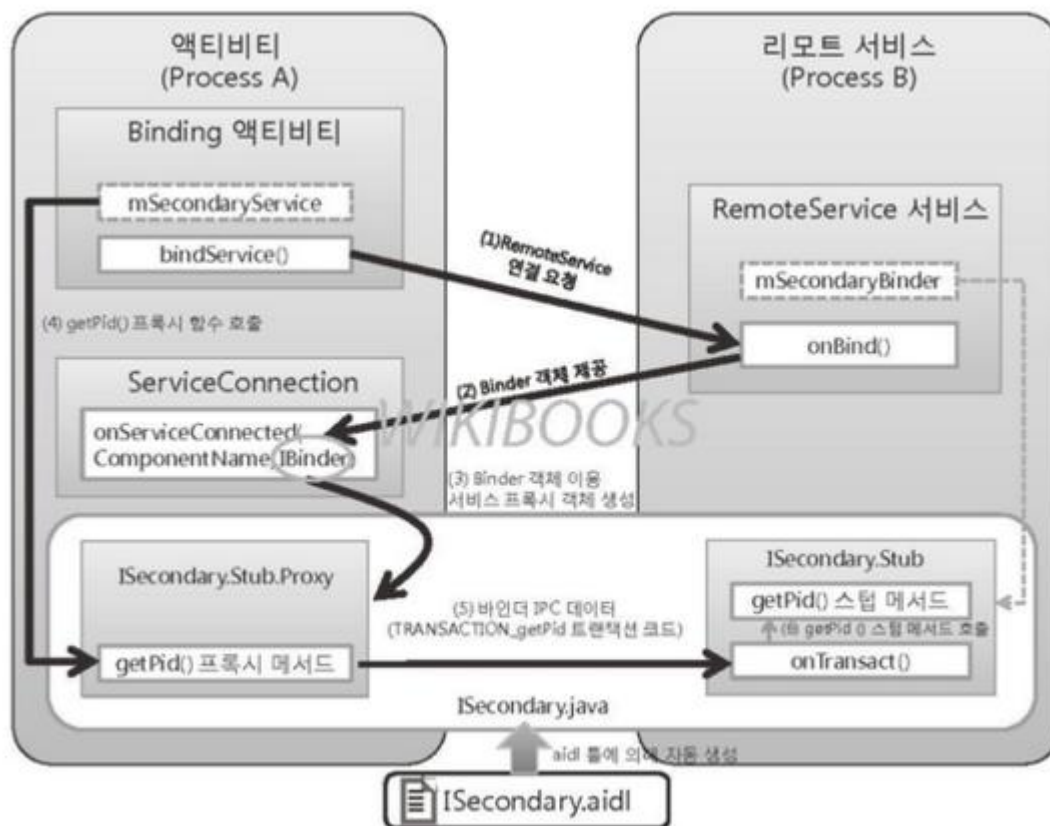
-Intent Service는 완전한 비동기 처리를 위해 내부에 백그라운드 태스크를 포함하고 있다

```
public class SimpleIntentService extends IntentService {  
    public SimpleIntentService() {  
        super(SimpleIntentService.class.getName());  
        setIntentRedelivery(true);  
    }  
  
    @Override  
    protected void onHandleIntent(Intent intent) {  
        // 백그라운드 스레드에서 호출됨  
    }  
}
```

## 5. Remote Service

- 앞의 예제와 같이 동일한 프로세스 내에서 공유되는 서비스를 Local Service, 서로 다른 프로세스에서 사용하는 서비스를 Remote Service 라고한다

-안드로이드는  
AIDL을 통해서 서로 다른  
프로세스에 존재하는  
서비스와 통신할 수 있다



\* AIDL

Android Interface Definition Language