

Dependency injection made easy with Dagger2

Alessandro Candolini

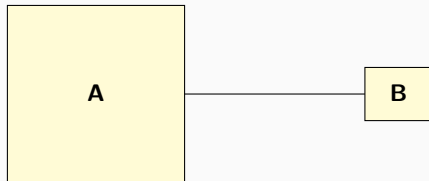
January 25, 2018

Agenda

1. Dependency injection principles
2. Dagger2
3. Dagger2 Android

Dependency injection principles

What is a dependency?



```
/** Class A */  
class A {  
    // ....  
    fun doSomething() {  
        b.log("text")  
    }  
}  
  
/** Class B (dependency) */  
class B {  
    fun log(text : String) {  
    }  
}
```

```
/** Class A */  
class A {  
    fun doSomething() {  
        B.log("text") // <- static method  
    }  
}  
  
/** Class B (Dependency) */  
class B {  
    companion object {  
        fun log(text: String) {  
        }  
    }  
}
```

Examples:

- Helper classes
- Utils classes
- Manager classes, etc. . .

Drawbacks:

- *A* not testable in isolation
- *A* *tightly coupled* to *B*
- Lack of encapsulation (backdoor)
- *Hidden* dependency

More Examples:

- `Application.getStaticContext()\`
- migrating one class leads to migrate 100 classes...

Dagger2

Dagger2 Android

Questions?