

Git-Hub Folder :

Project-Template

Development

Testing

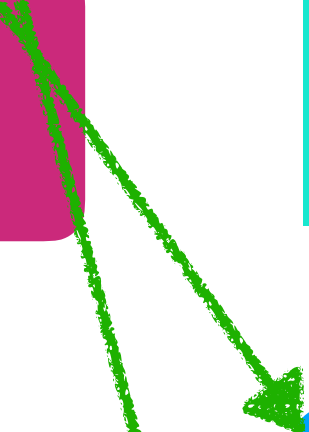
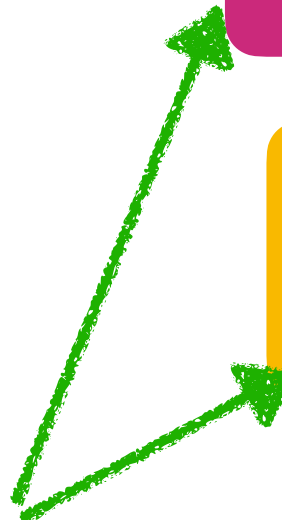
Testing-SignOff08

Testing-DenQA

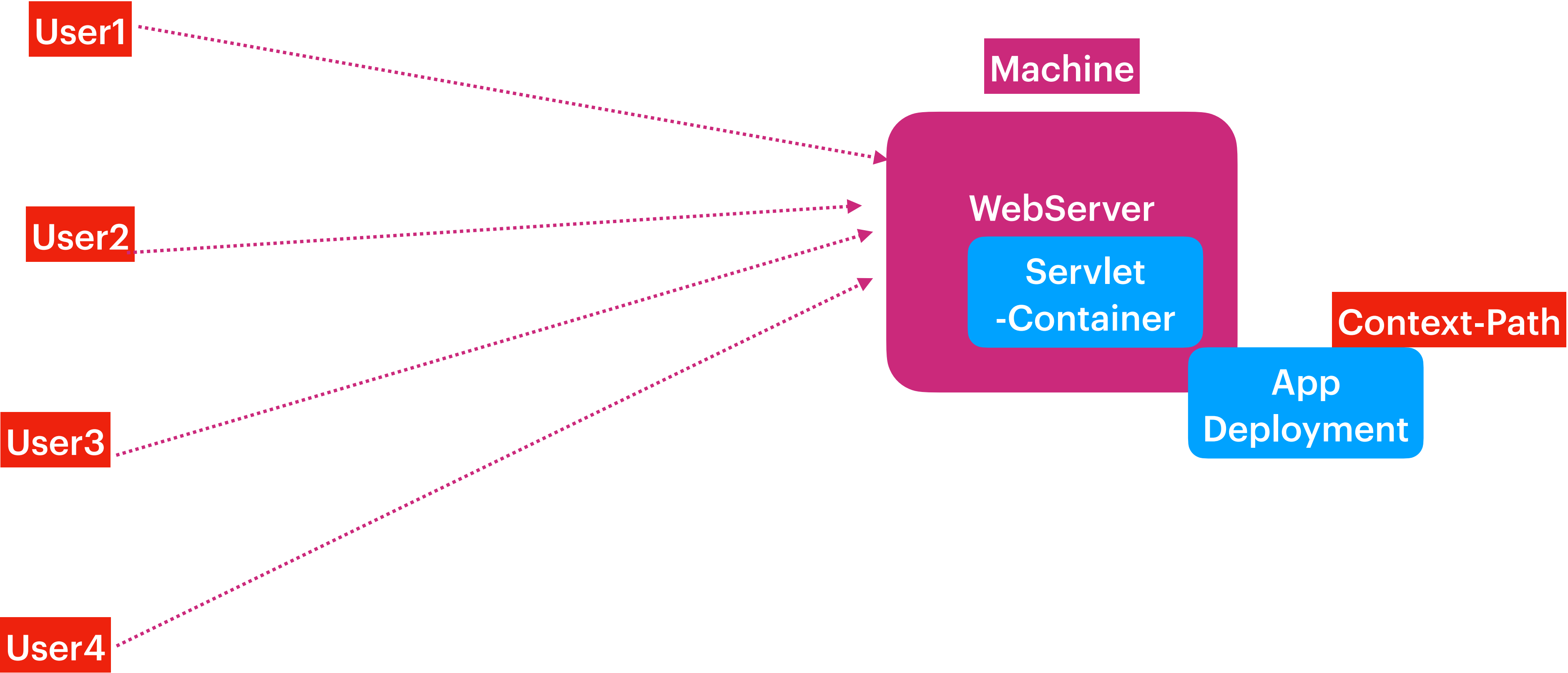
Configuration
Management

Testing-PreProd
(UAT)

Deployment in
Production

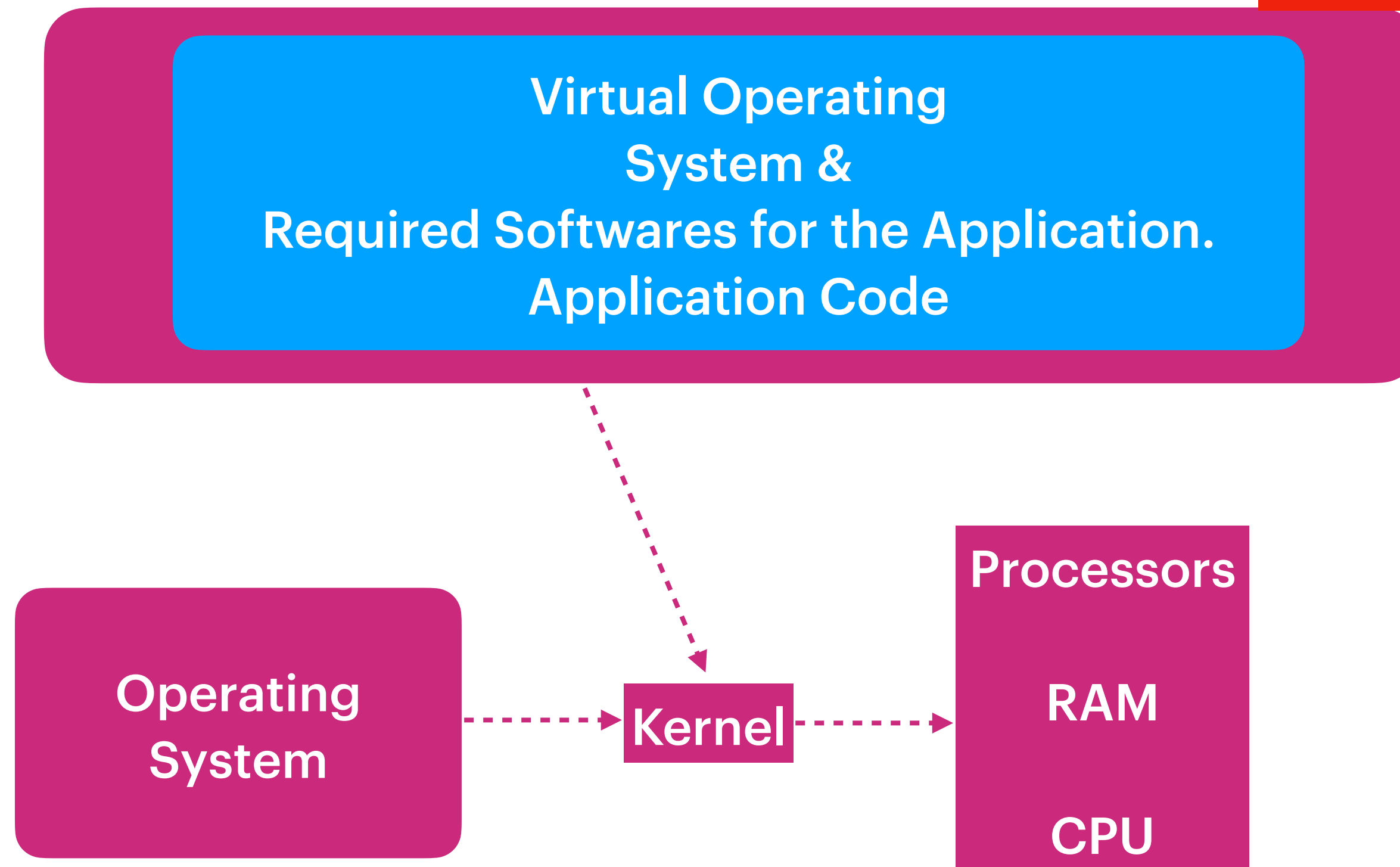


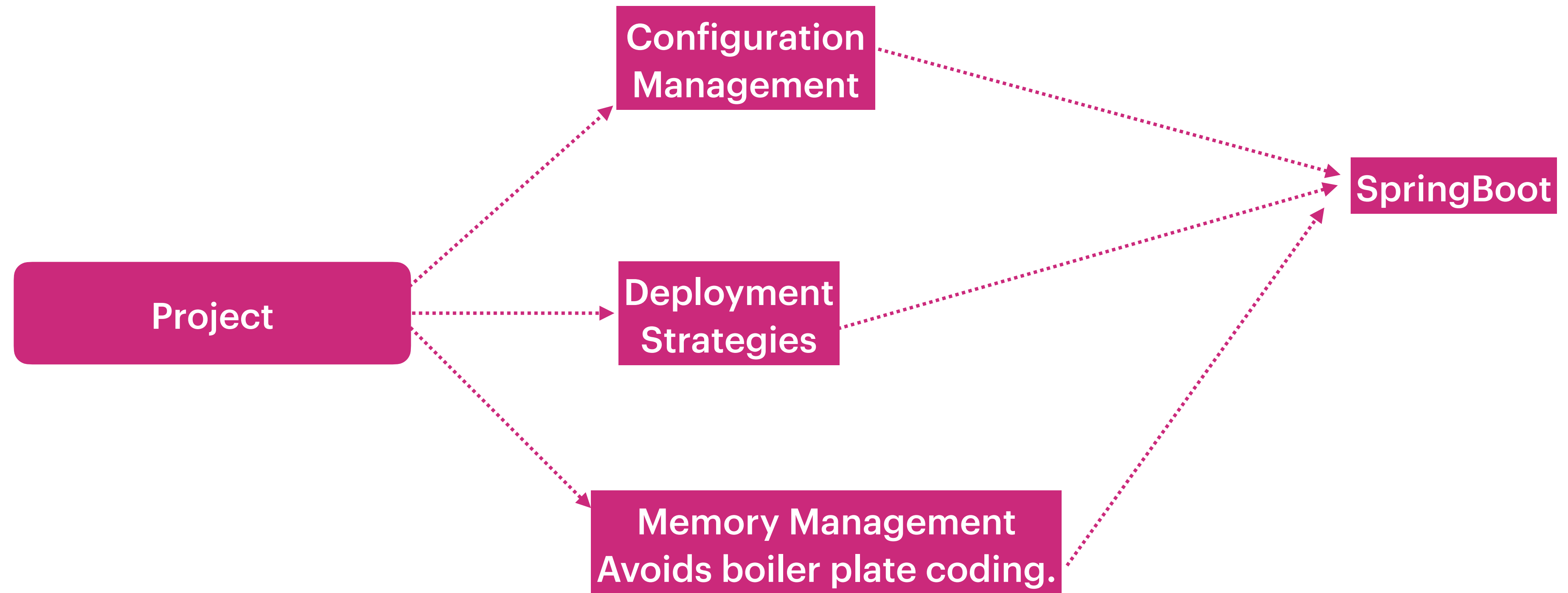
<https://ip-address-of-server-machine:webserver-port/app-name/<url-pattern>>

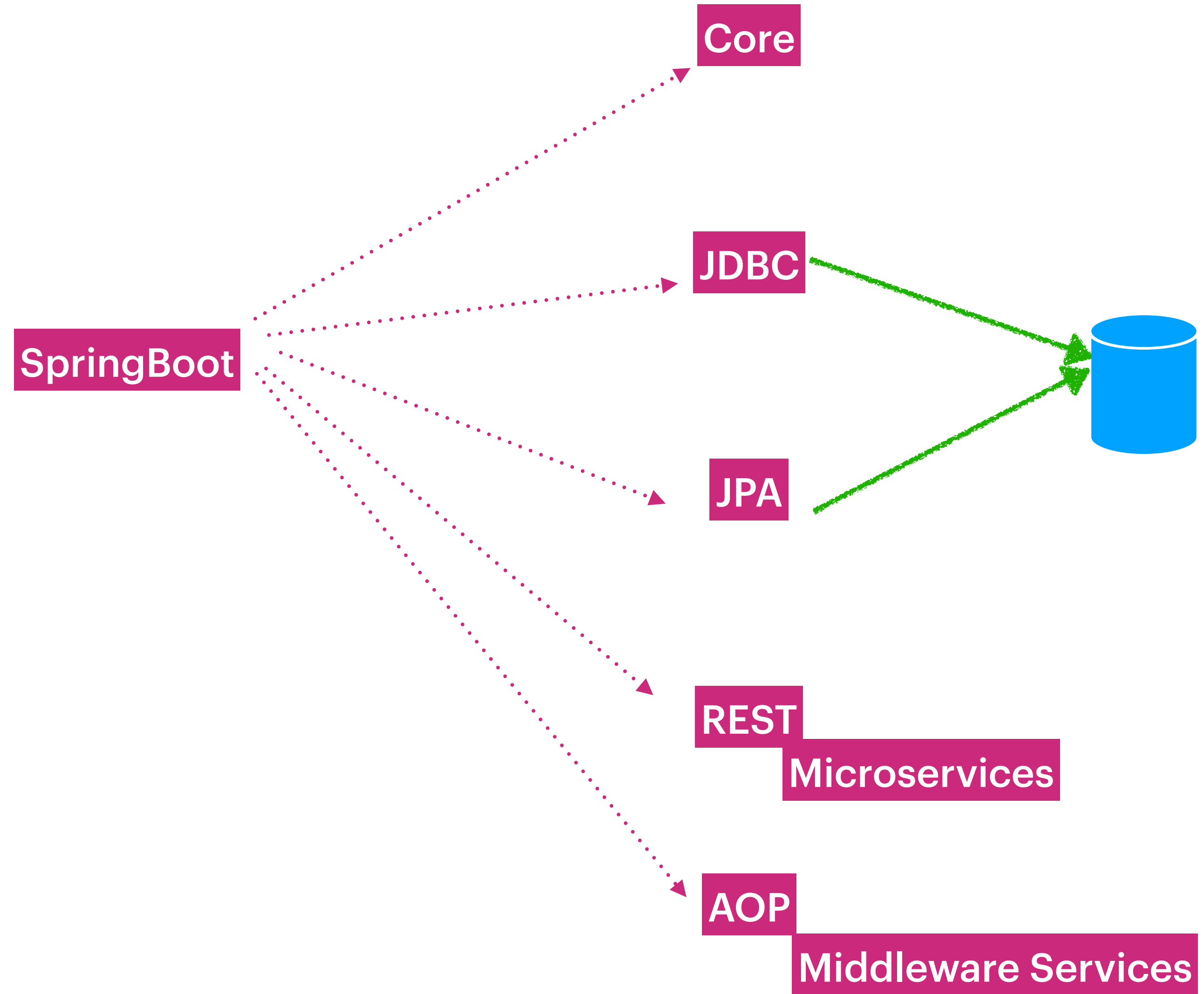


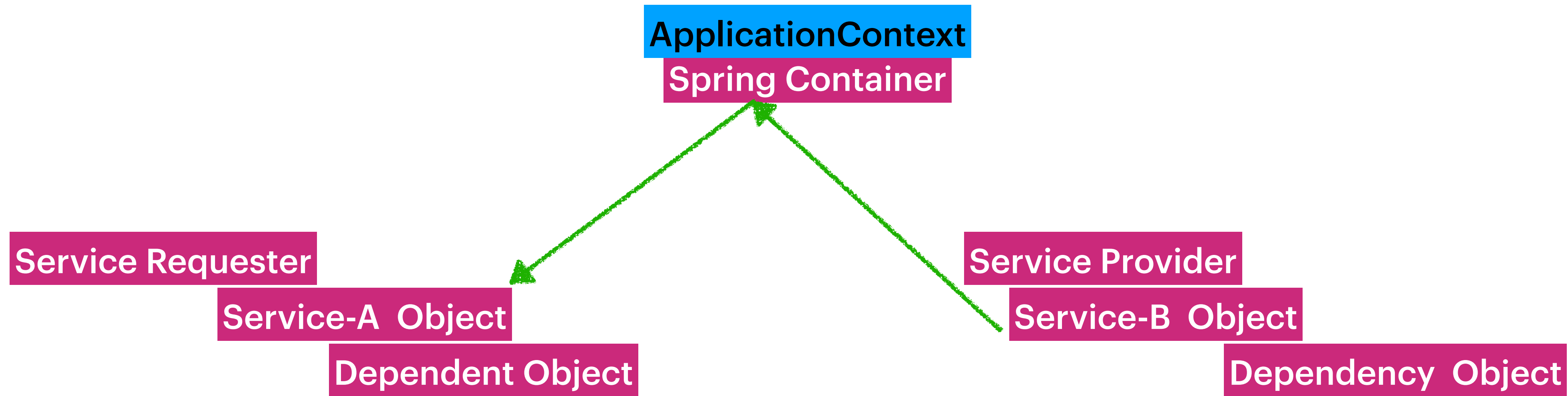
Dockerization

Container



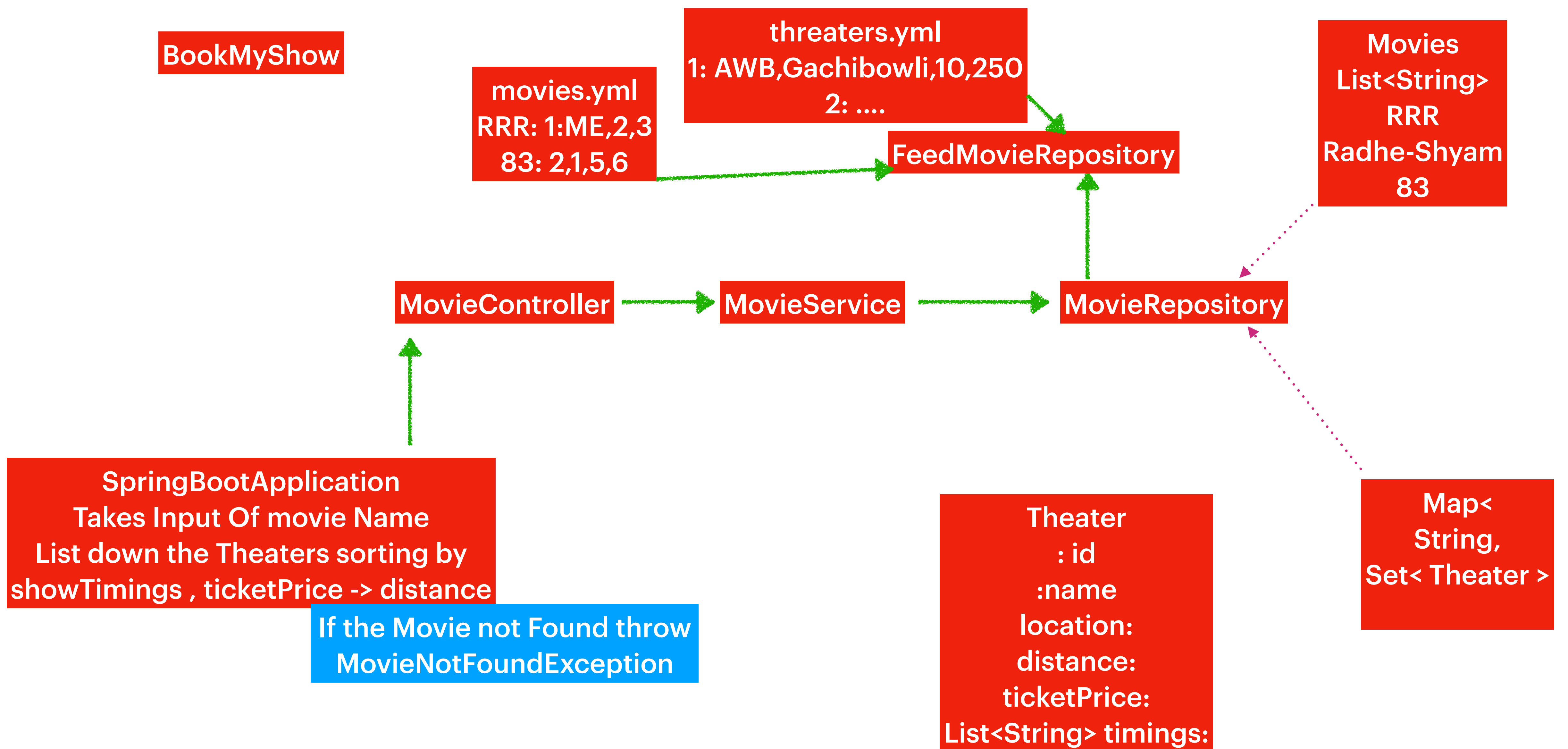


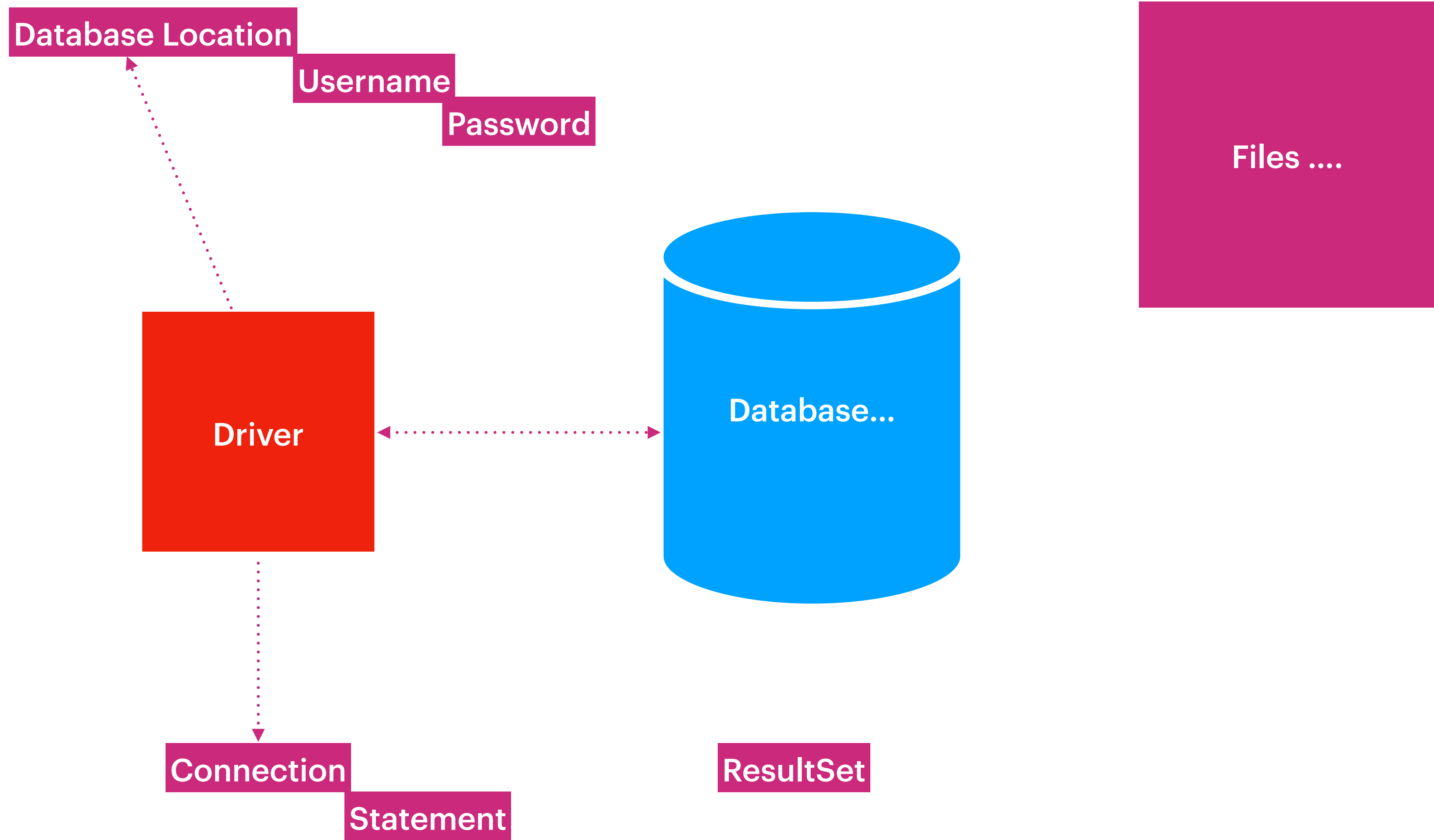


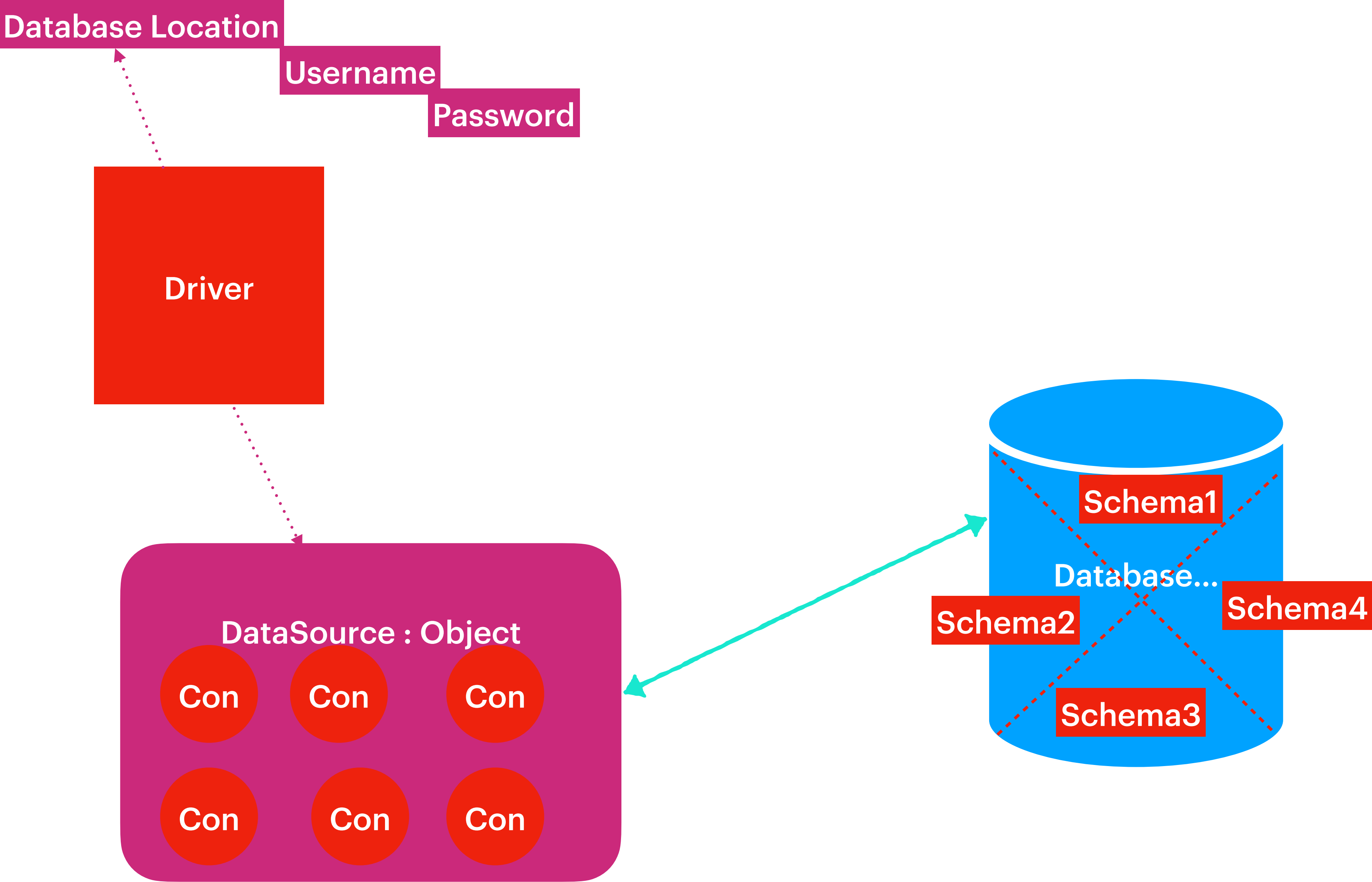


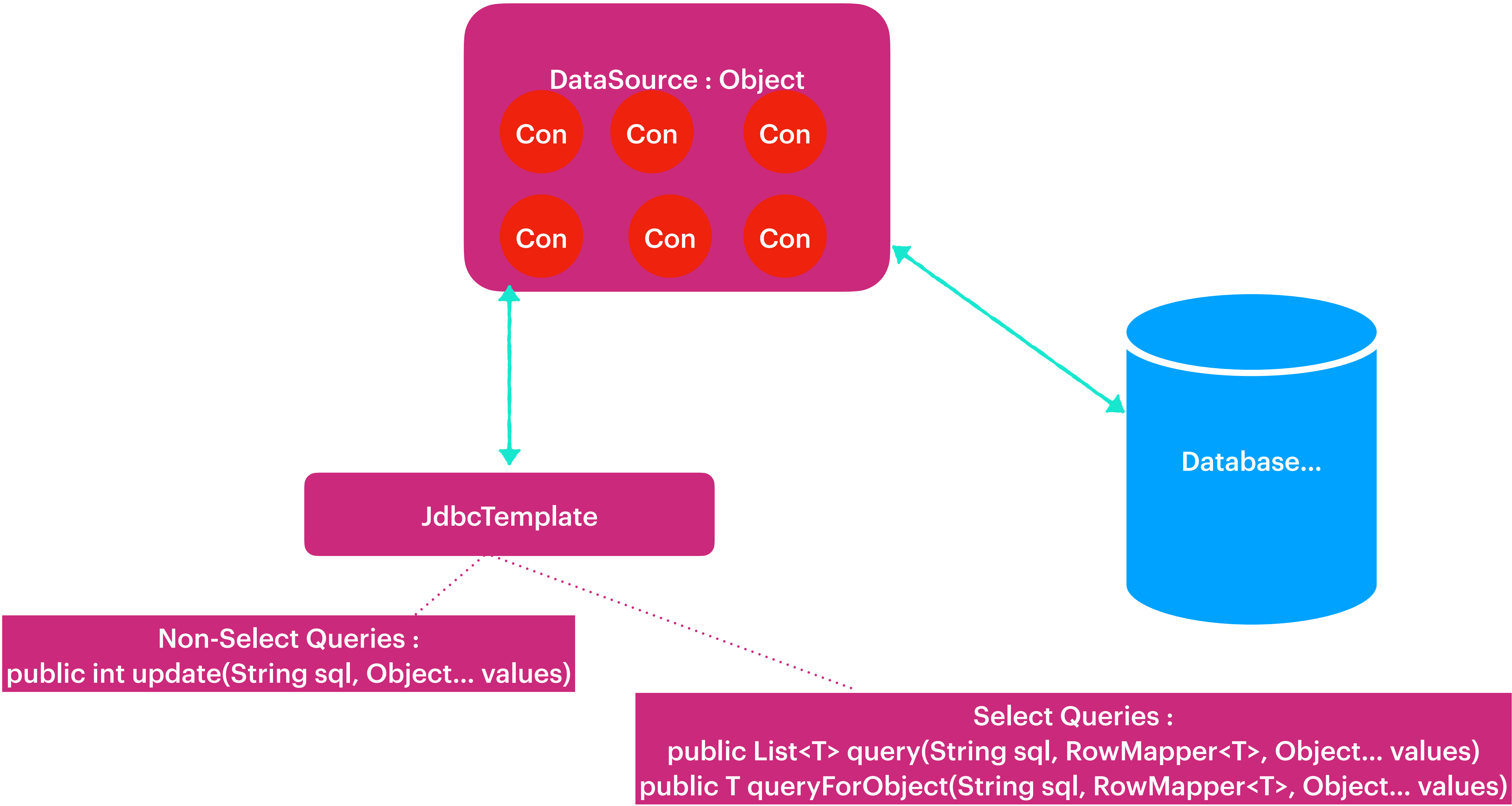
Dependency Injection :
Injecting Dependencies to the Dependents through Spring Container is Called Dependency-Injection.

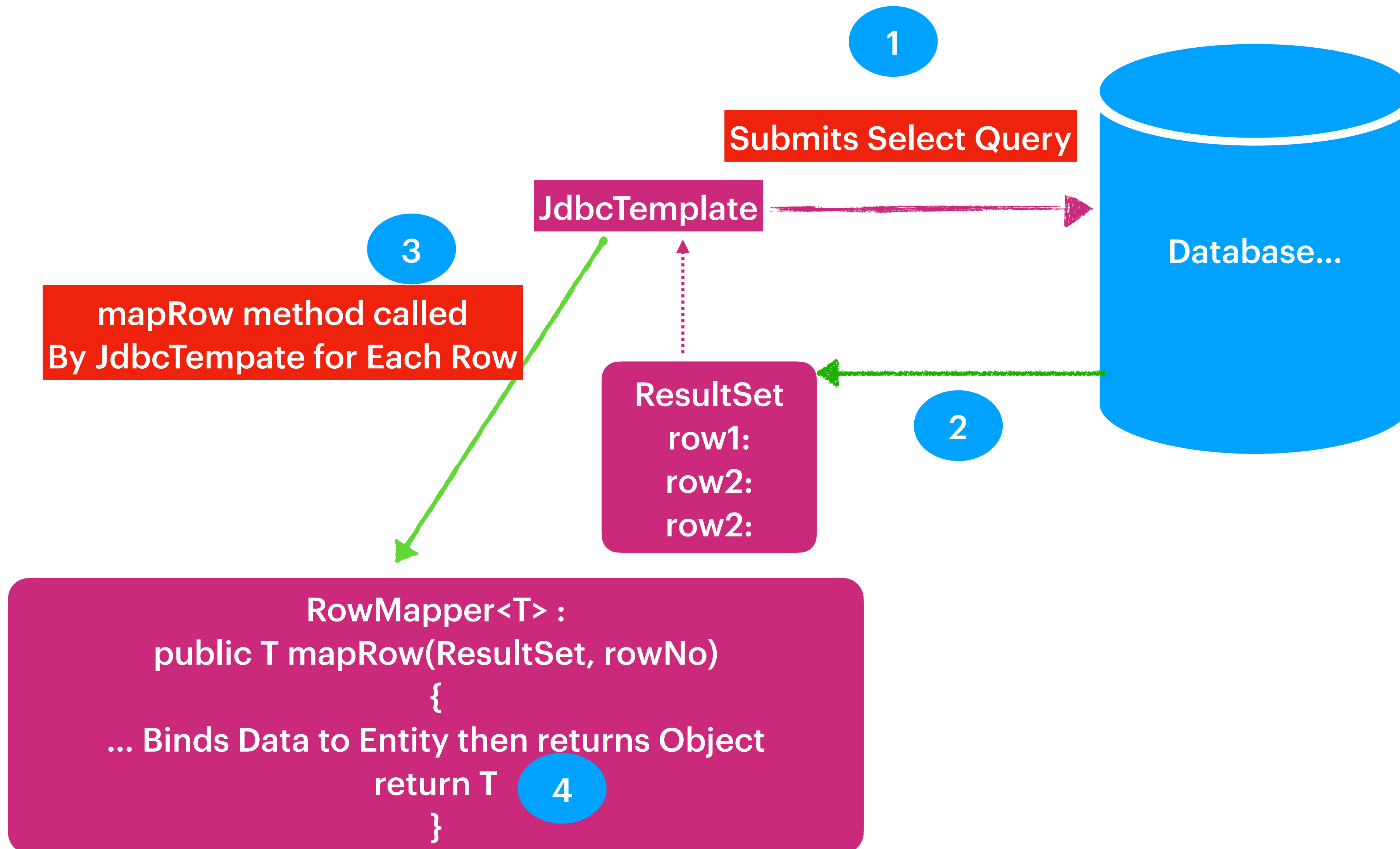
Setter - Injection
Constructor - Injection

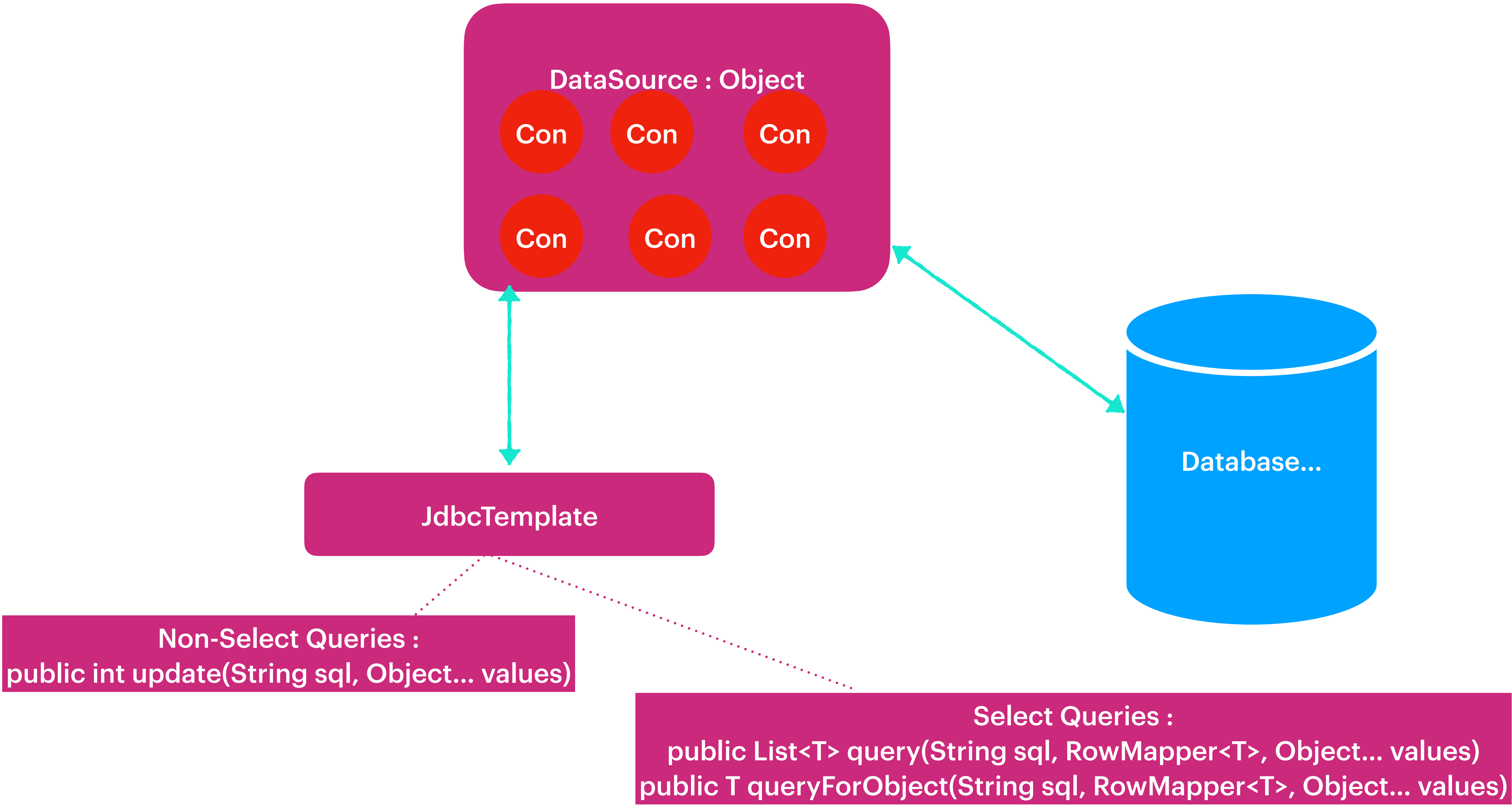


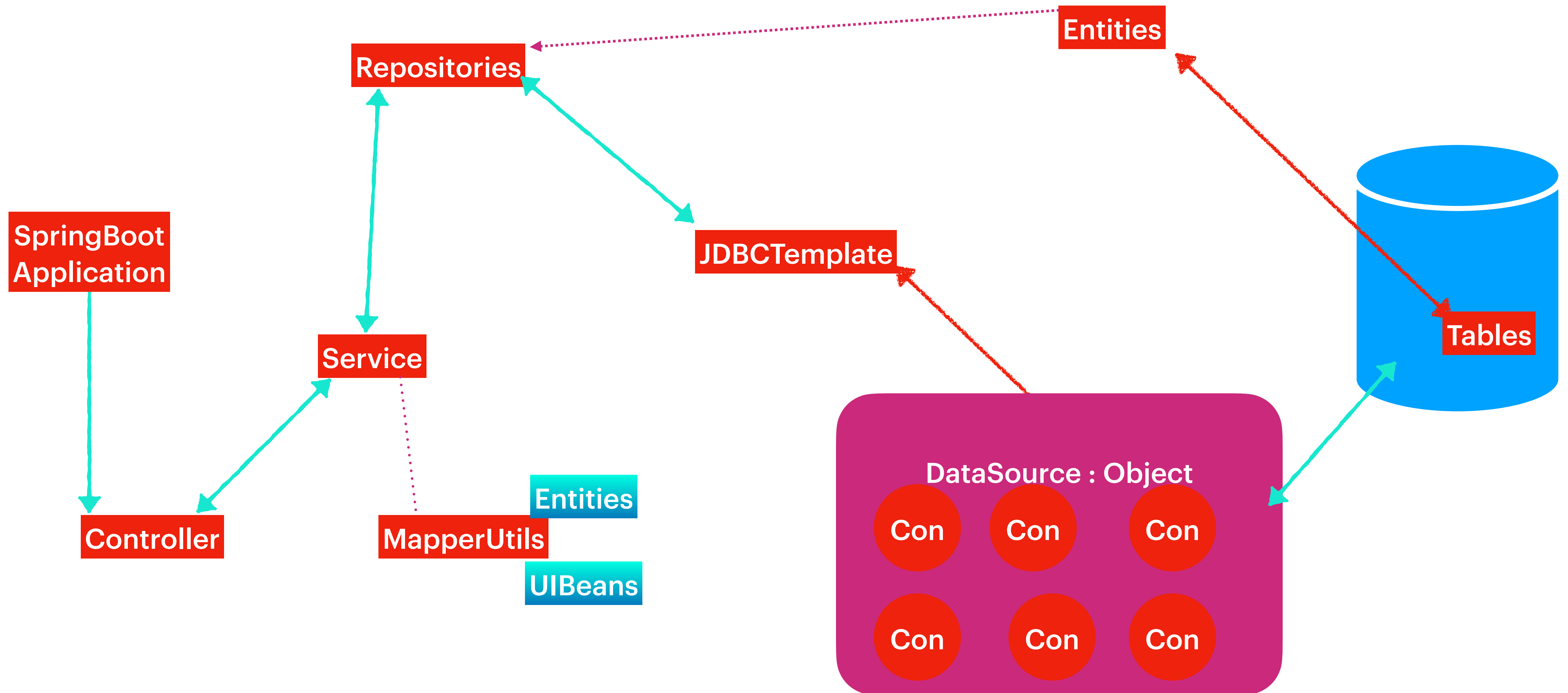


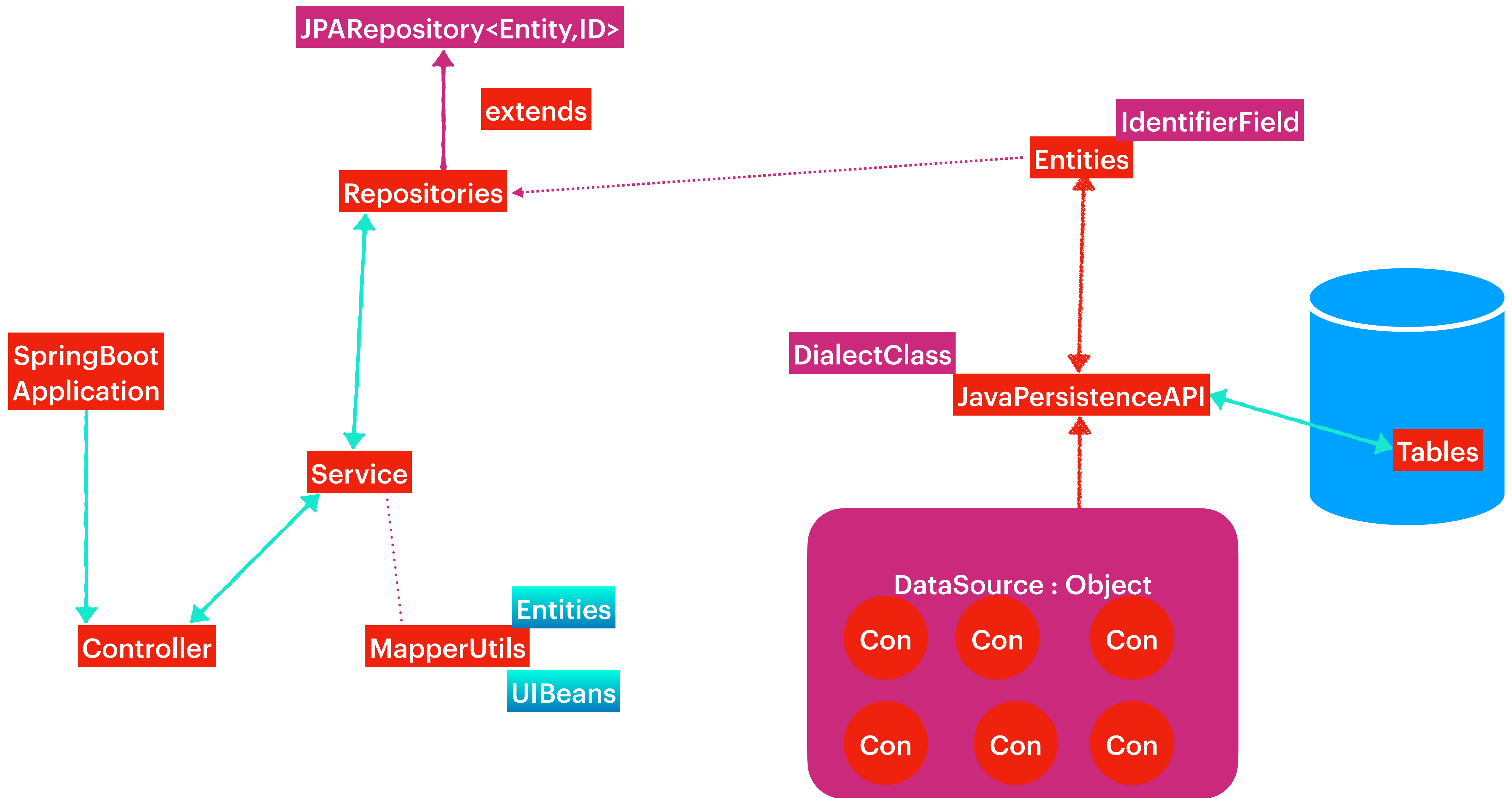


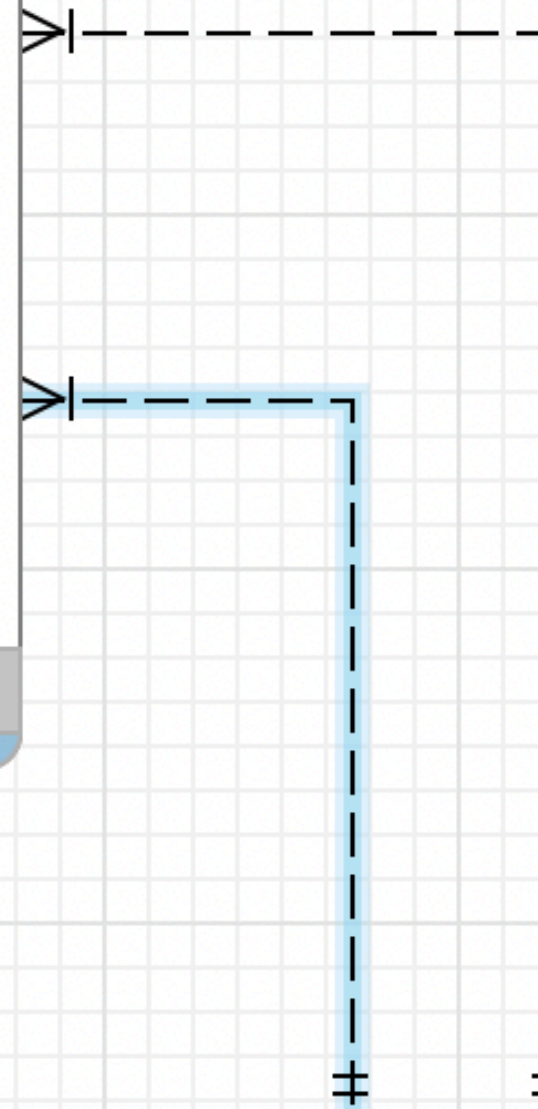
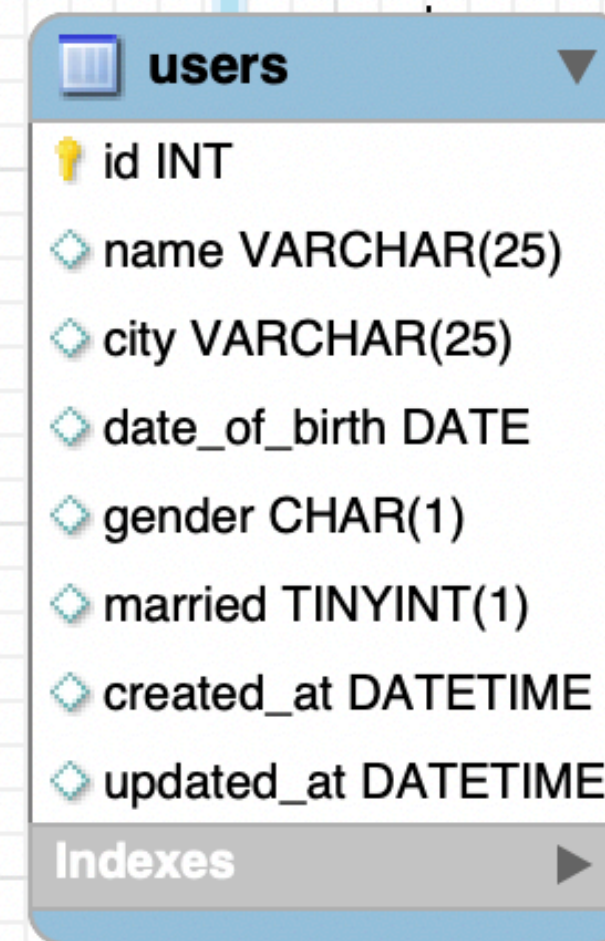
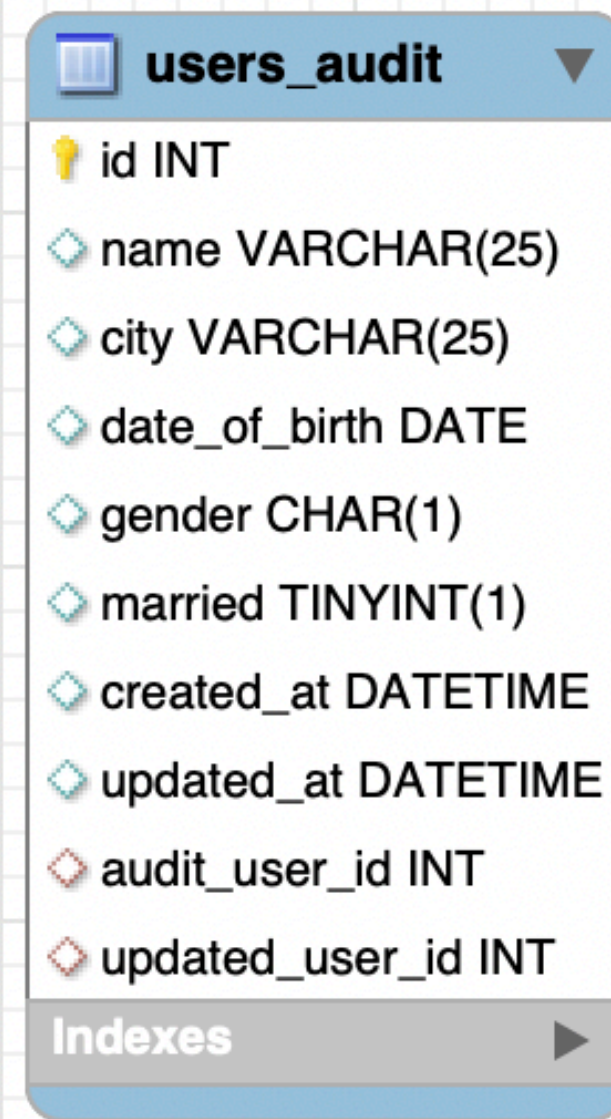


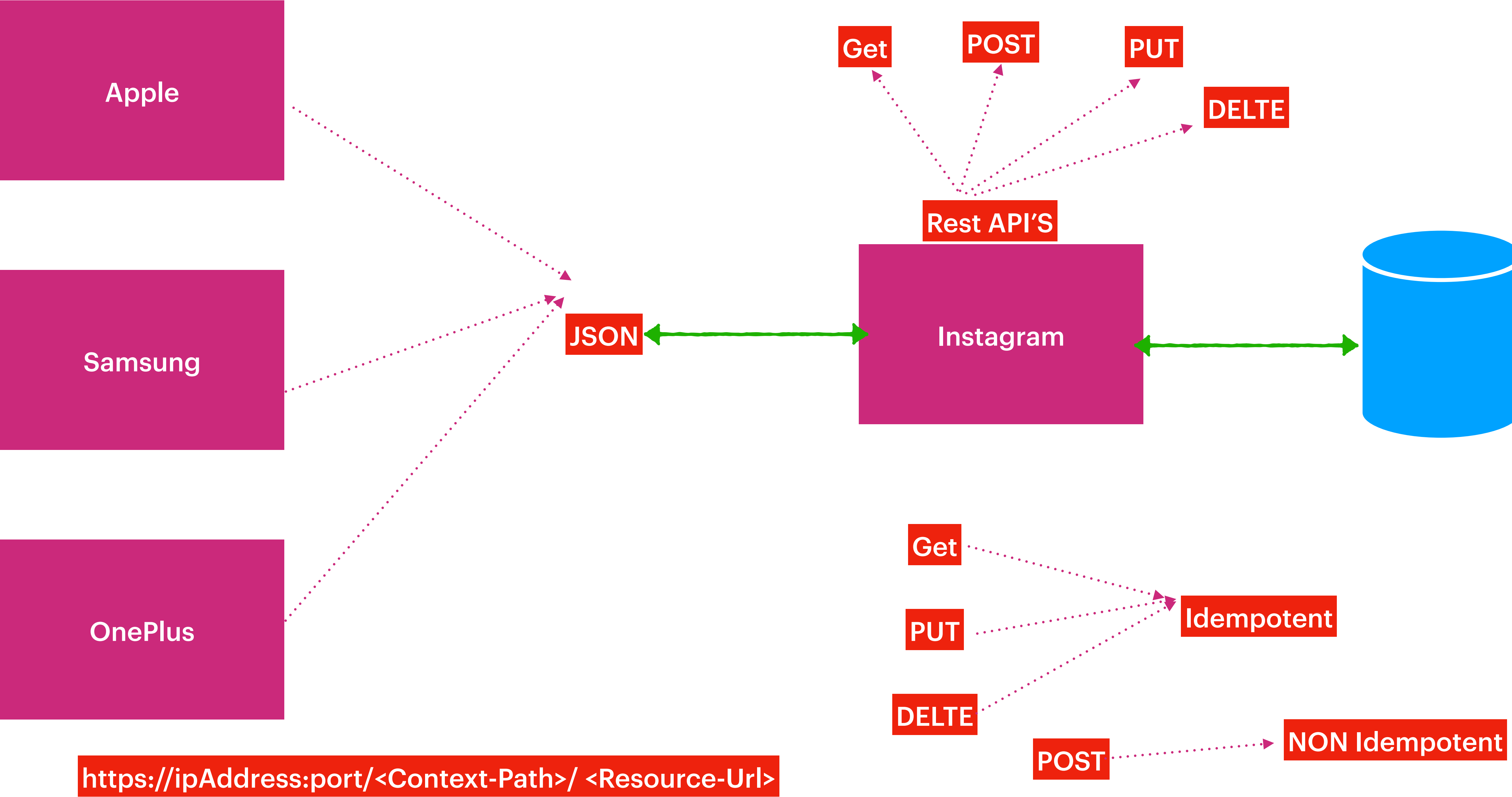




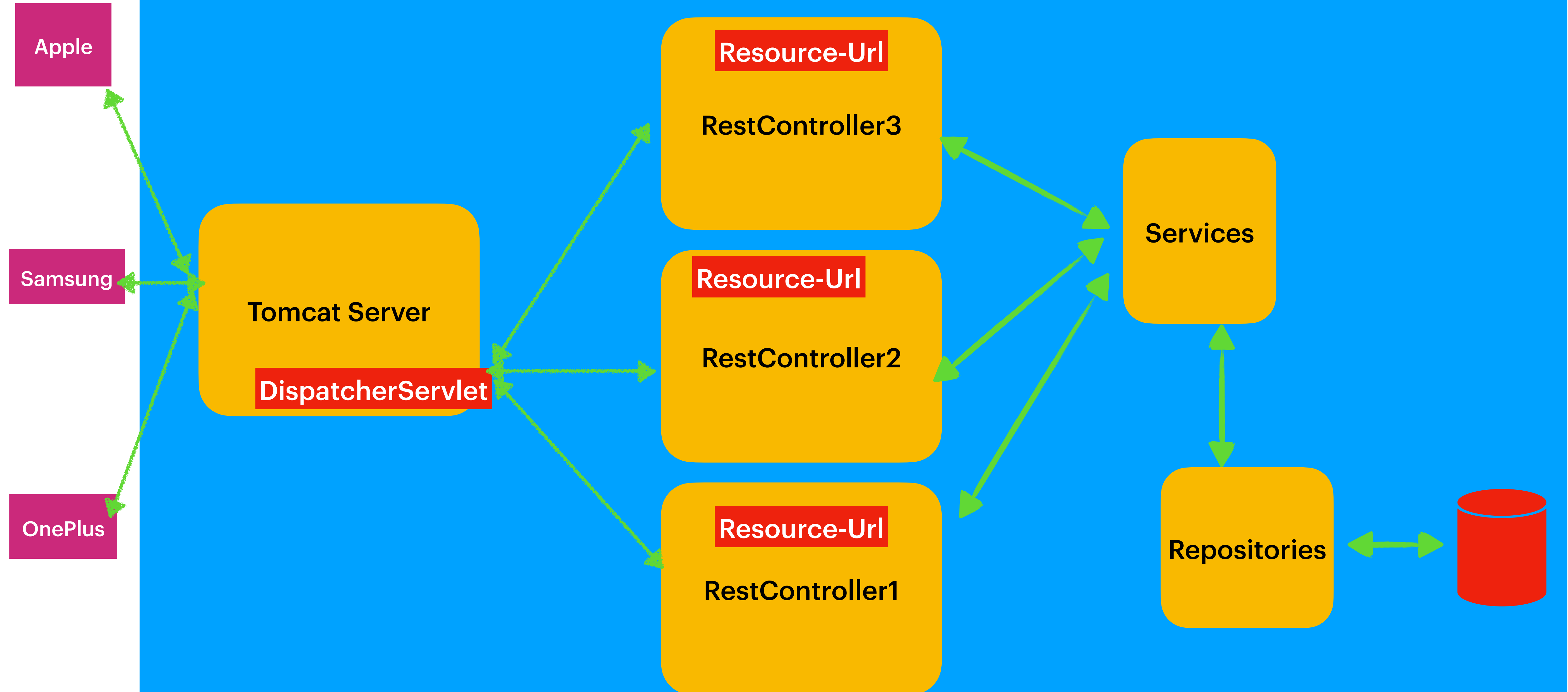


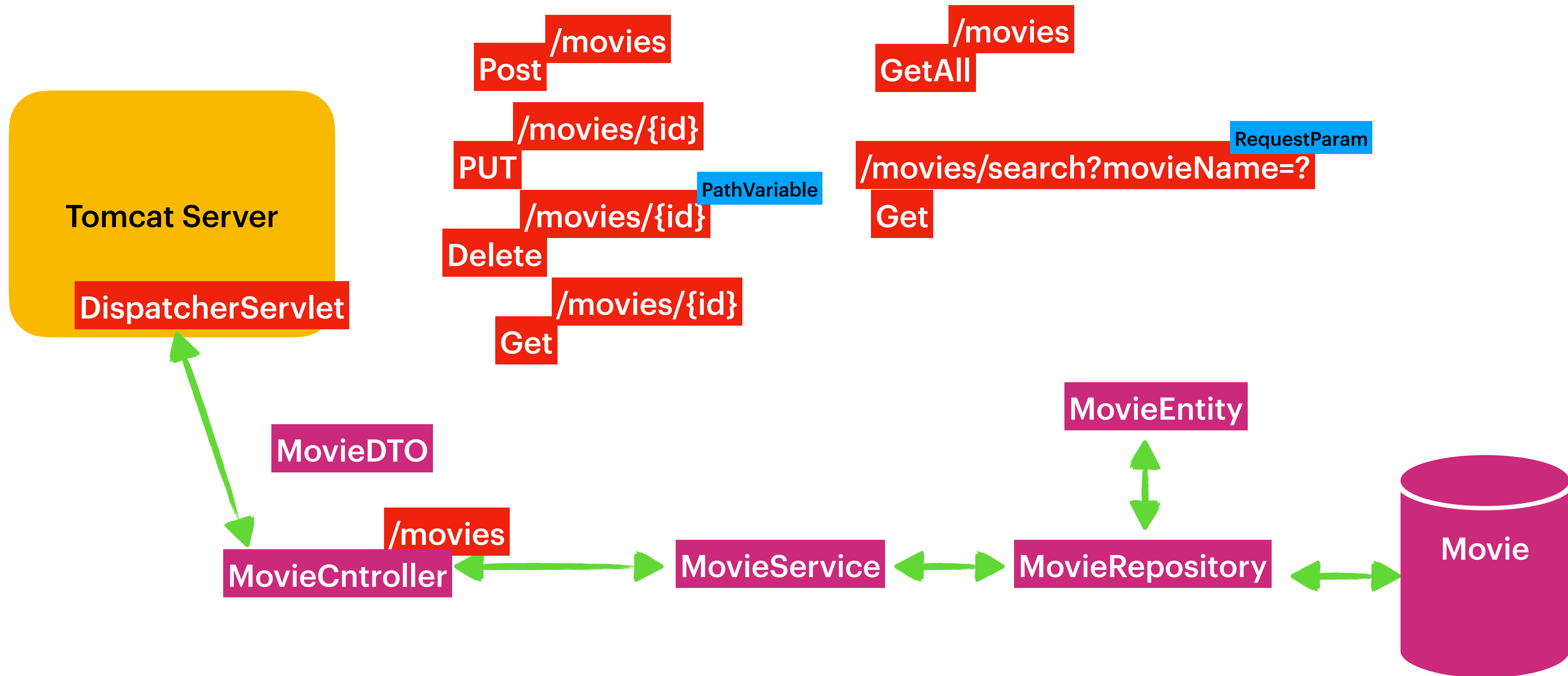


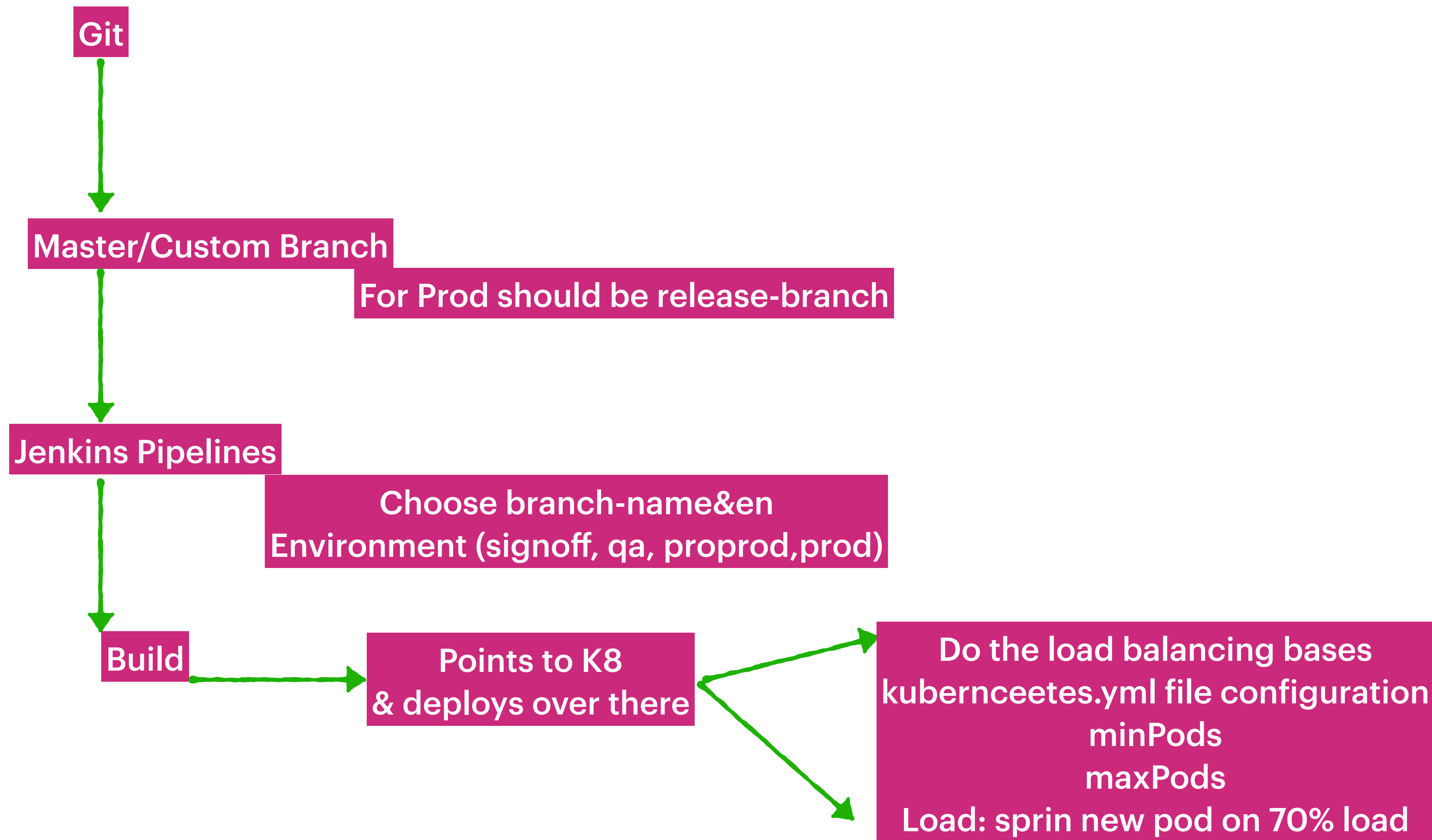




SpringBoot Application(Instagram)



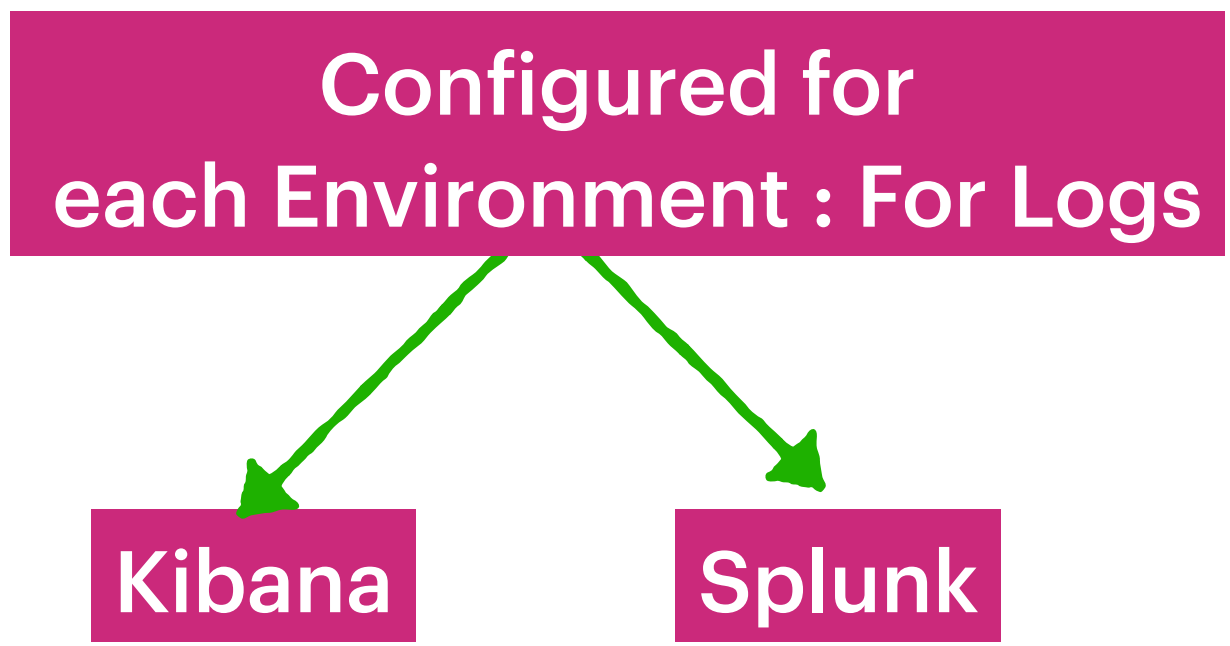




Configured for
each Environment : For Logs

Kibana

Splunk



```
@Bean("oracleDataSource")
public DataSource oracleDataSource()
{
    Return DataSource.build().un....;
}
```

```
@Bean("mySqlDataSource")
public DataSource oracleDataSource()
{
    Return DataSource.build().un....;
}
```

```
@Bean("oracleEtityMgr")
Public EntityMamanger enmgr()
{
entityManager.setDAtaSource(mySqlDataSource);
}
```

```
@Default
@Bean
Public EntityMamanger oraracenmgr()
{
entityManager.setDAtaSource(oracleDataSource);
}
```