



# Mybatis总结

T A H N K   Y O U   F O R   W A T C H I N G

 主讲老师Lison : 525765982

 课程咨询依娜老师 : 2470523467



# 目录

## CONTENTS



### 看源码的正确姿势

论学习源码的重要性  
学习源码的方法论



### MyBatis源码概述

**MyBatis**源码概述  
初始化阶段  
动态代理阶段  
数据库访问阶段



### 手写MyBatis解析

手写**MyBatis**解析



为什么使用mapper接口就能对数据库进行访问？

为什么在spring容器中，并没有出现sqlSession的身影？



# 站在巨人的肩膀上你能尿的更远



## 论学习源码的目的：

1. 编写优雅、高效的代码经验；
2. 提升微观的架构设计能力，重点在思维 and 理念；
3. 解决工作中、学习中的各种疑难杂症；
4. 在BATJ一线互联网公司面试中展现优秀的自己；

# 学习源码的步骤



1. 精心挑选要阅读的源码项目；
2. 饮水思源——官方文档，先看文档再看源码；
3. 下载源码，安装到本地，保证能编译运行；
4. 从宏观到微观，从整体到细节；
5. **找到入口**，抓大放小，梳理核心流程；
6. 源码调试，找到核心数据结构和关键类；
7. 勤练习，多折腾；



# 目录

## CONTENTS



### 看源码的正确姿势

论学习源码的重要性  
学习源码的方法论



### MyBatis源码概述

**MyBatis**源码概述  
初始化阶段  
动态代理阶段  
数据库访问阶段



### 手写MyBatis解析

手写**MyBatis**解析



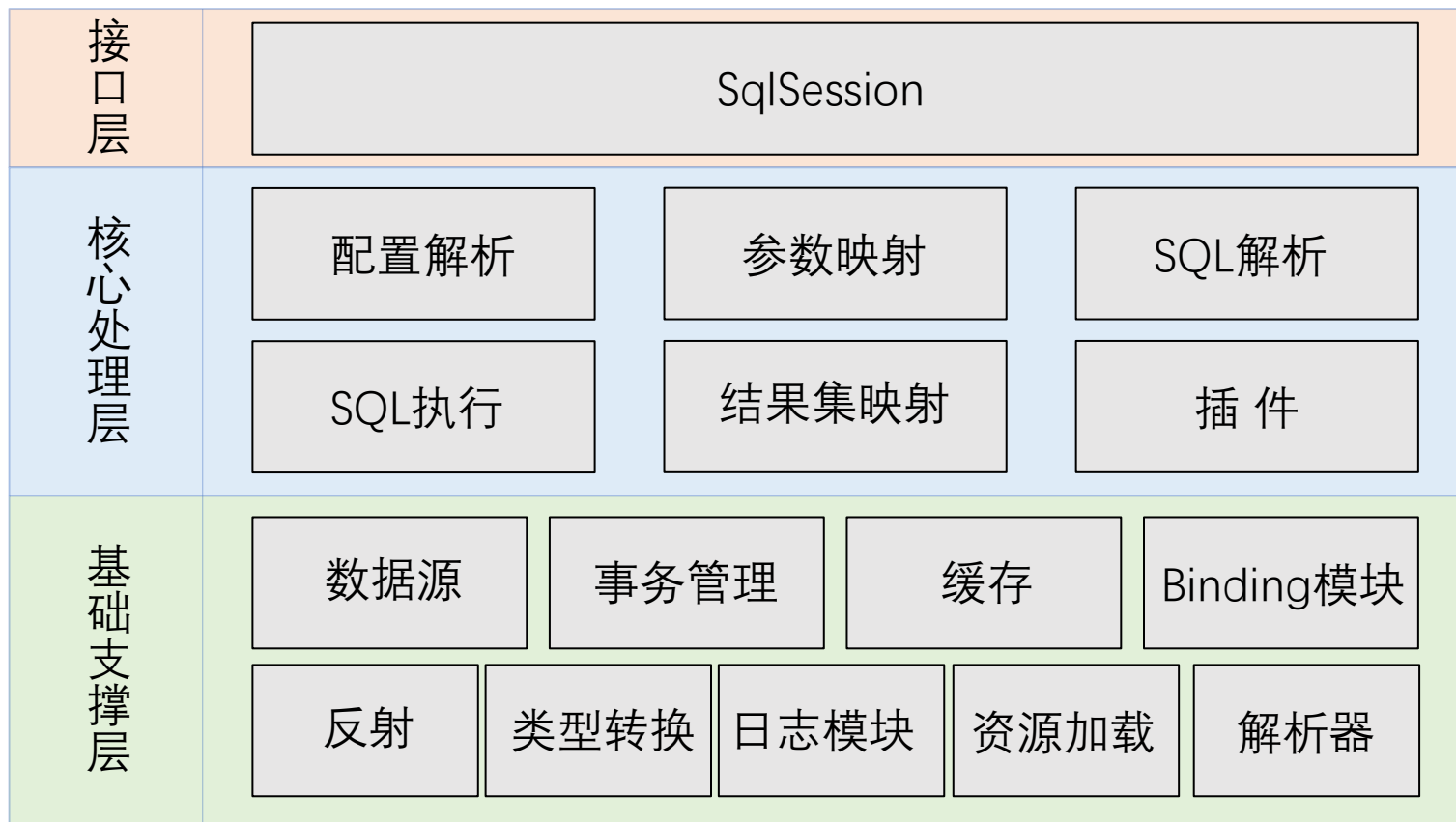
# MyBatis源码整体架构



MyBatis 源码下载地址：<https://github.com/mybatis/mybatis-3>



MyBatis源码结构x







# mybatis核心流程三大阶段

## 初始化阶段

读取XML配置文件和注解中的配置信息，创建配置对象，并完成各个模块的初始化的工作；

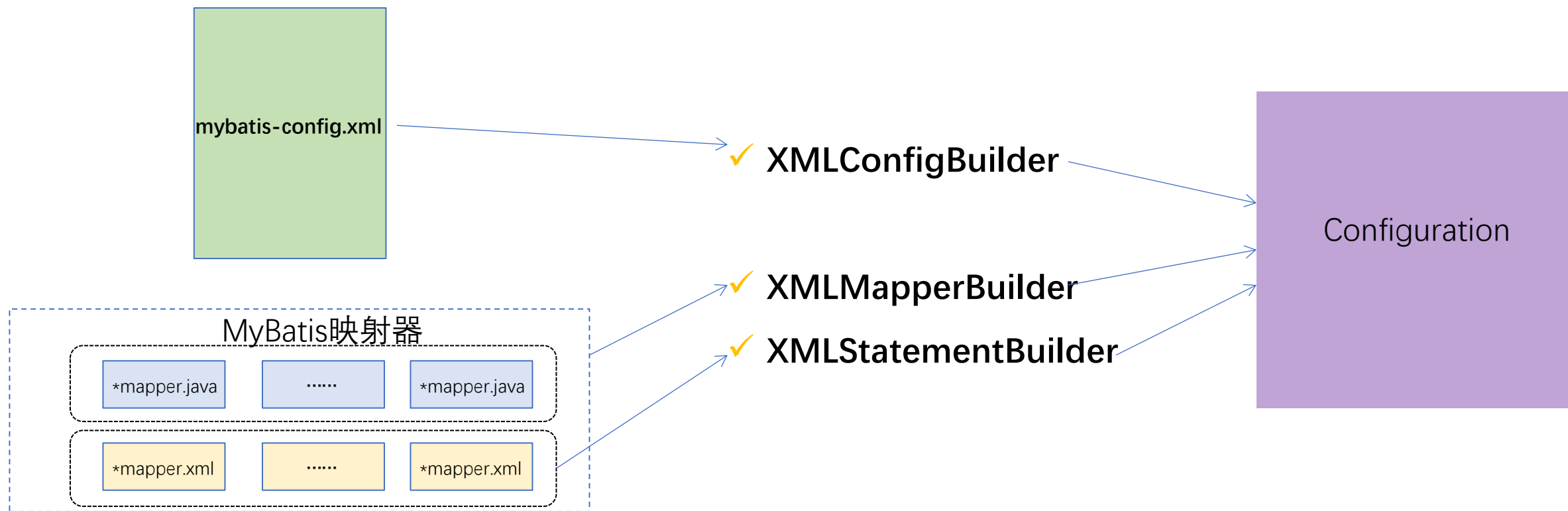
## 代理阶段

封装iBatis的编程模型，使用mapper接口开发的初始化工作；

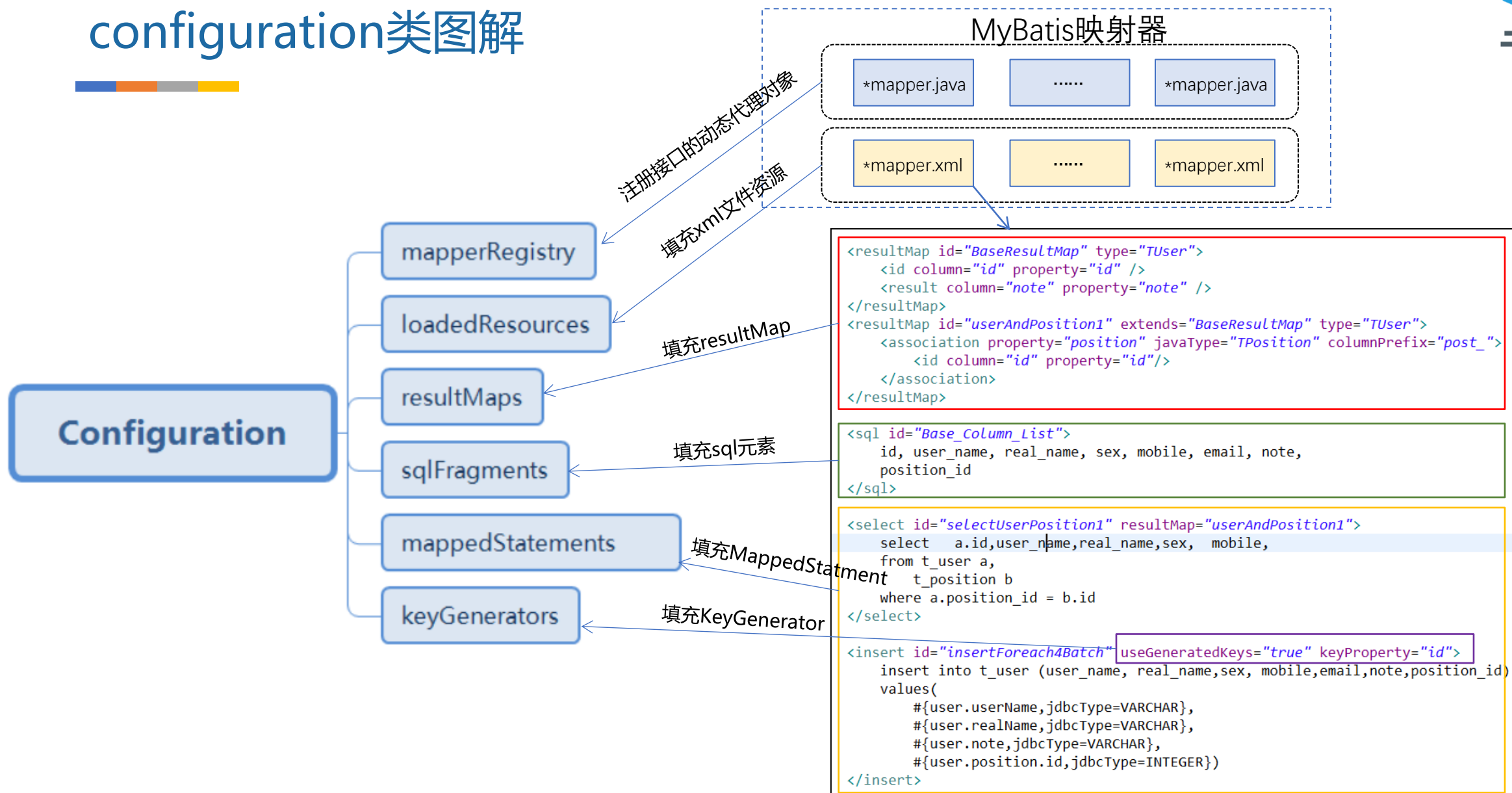
## 数据读写阶段

通过SqlSession完成SQL的解析，参数的映射、SQL的执行、结果的解析过程；

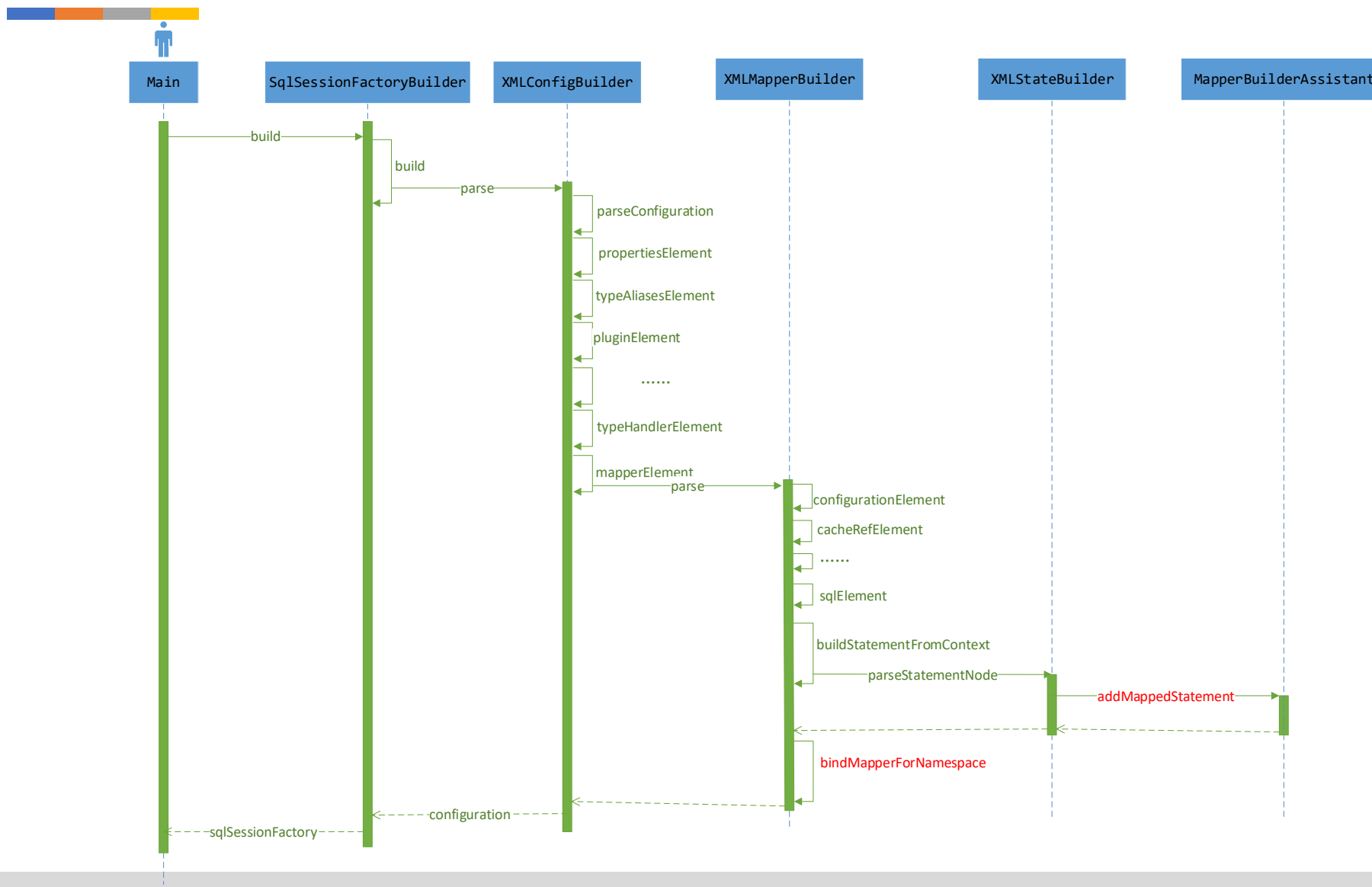
# Mybatis的初始化



# configuration类图解

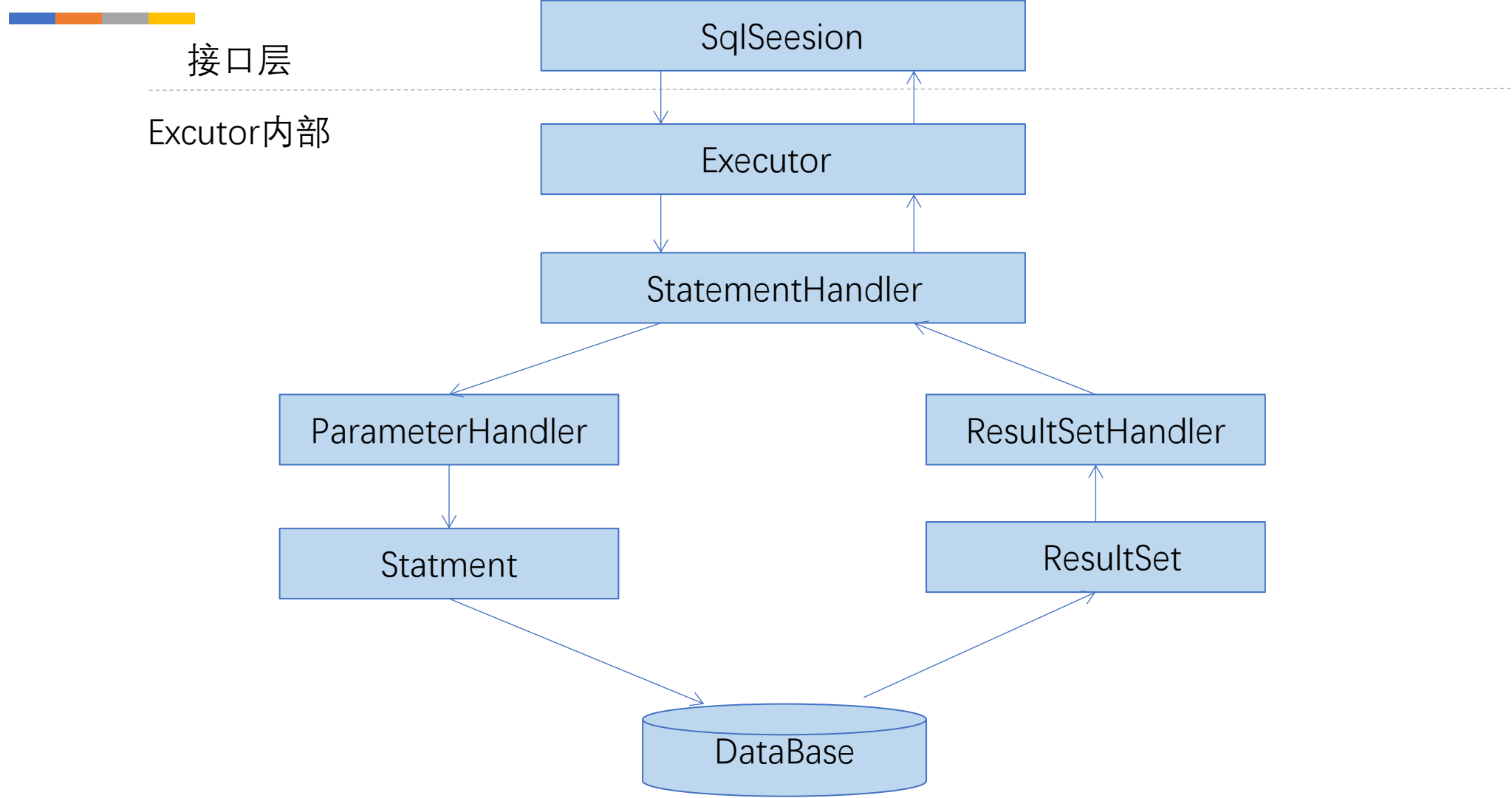


# MyBatis初始化过程





# SqlSeesion对外提供的访问接口



# 为什么使用mapper接口就能操作数据库？



配置文件解读 + 动态代理的增强



## 快速入门




## 追本溯源

```
@Test
// 快速入门
public void quickStart() throws IOException {
    // 2.获取sqlSession
    SqlSession sqlSession = sqlSessionFactory.openSession();
    // 3.获取对应mapper
    TUserMapper mapper = sqlSession.getMapper(TUserMapper.class);
    // 4.执行查询语句并返回结果
    TUser user = mapper.selectByPrimaryKey(1);
    System.out.println(user.toString());
}
```

```
@Test
// 快速入门 本质分析
public void originalOperation() throws IOException {
    // 2.获取sqlSession
    SqlSession sqlSession = sqlSessionFactory.openSession();
    // 3.执行查询语句并返回结果
    TUser user = sqlSession.selectOne("com.enjoylearning.mybatis.mapper."
                                     + "TUserMapper.selectByPrimaryKey", 1);
    System.out.println(user.toString());
}
```

翻译

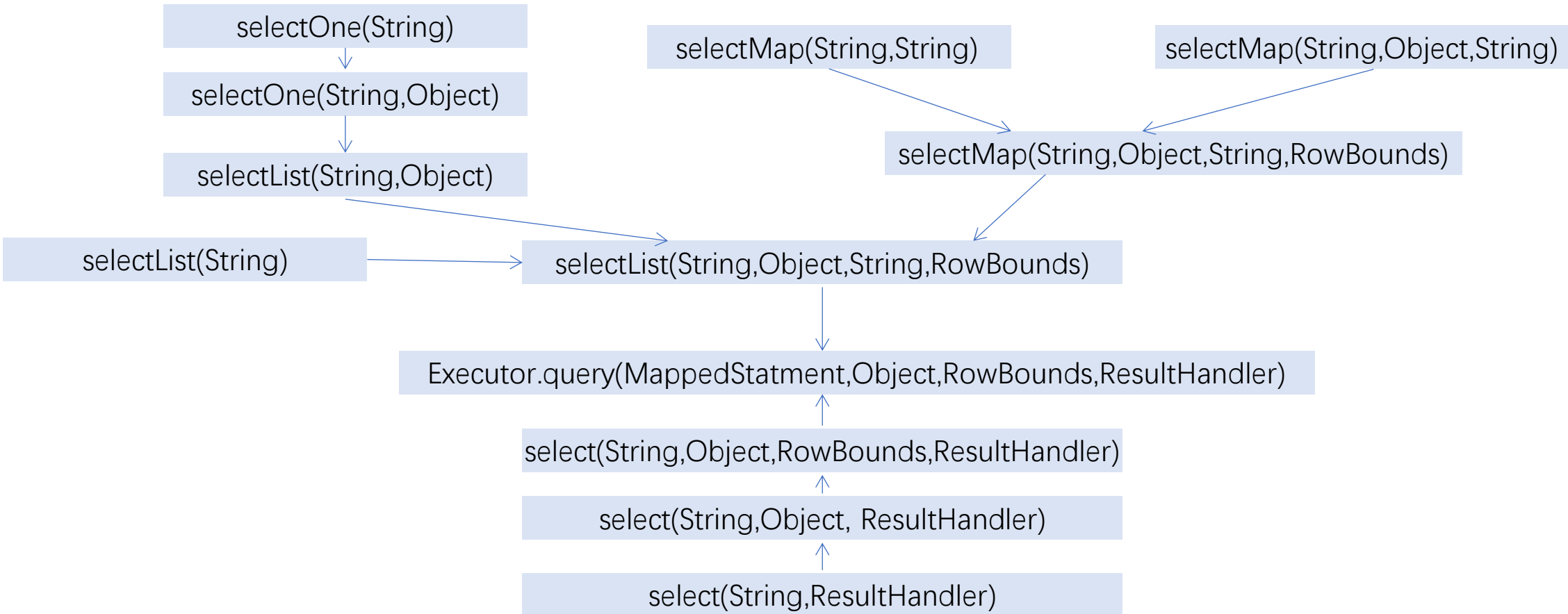
- ✓ 找到session中对应的方法执行；
- ✓ 找到命名空间和方法名
- ✓ 传递参数

- ✓ 找到session中对应的方法执行 ;  MapperMethod.**SqlCommand**. type + MapperMethod.MethodSignature.returnType
- ✓ 找到命名空间和方法名  MapperMethod.**SqlCommand**. name
- ✓ 传递参数  MapperMethod.MethodSignature.paramNameResolver





# SqlSession查询接口嵌套关系



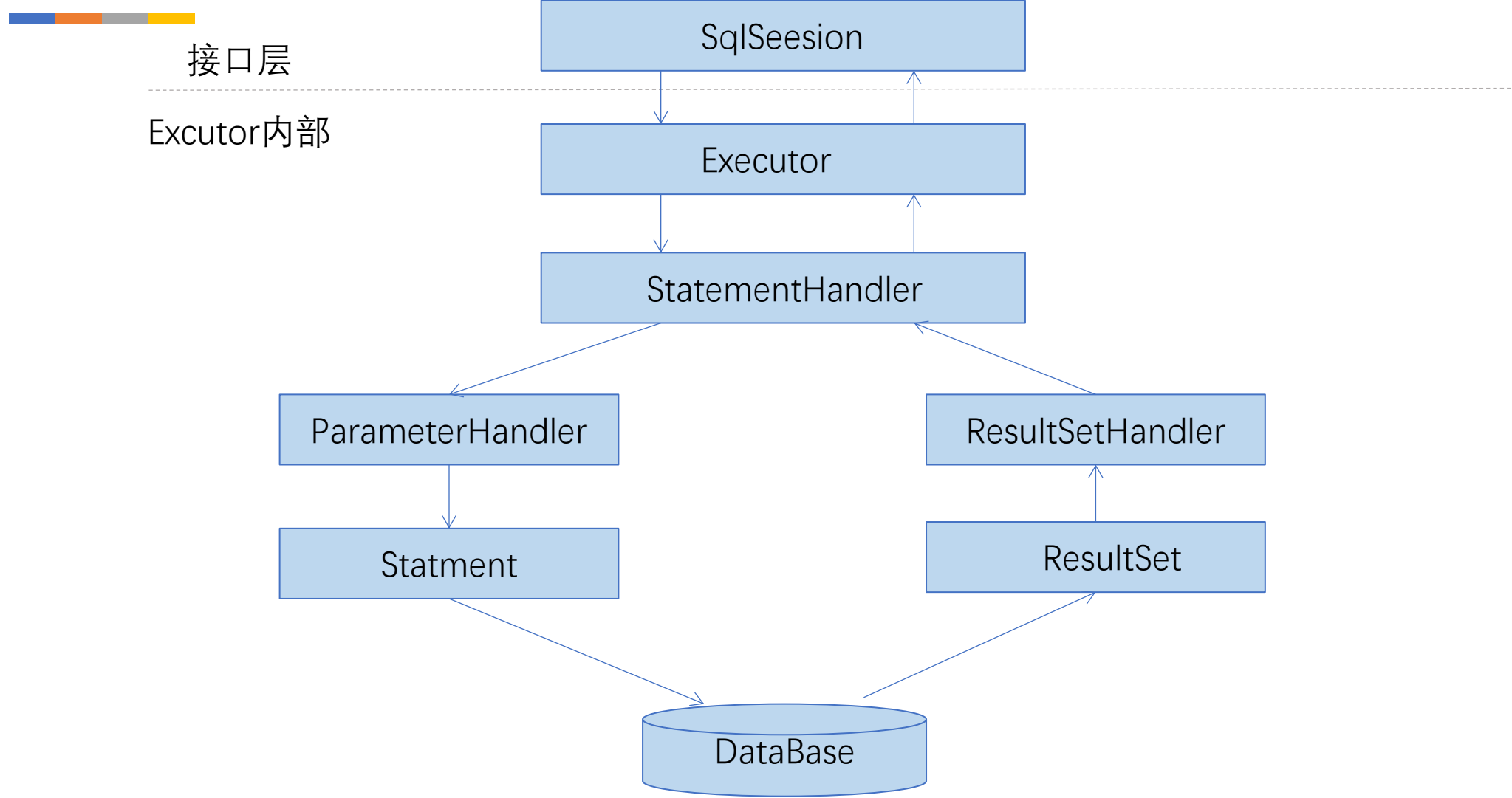
# Executor的三个重要小弟



- 通过对SimpleExecutor doQuery()方法的解读发现，Executor是个指挥官，它在调度三个小弟工作：
  - StatementHandler：它的作用是使用数据库的Statement或PreparedStatement执行操作，启承上启下作用；
  - ParameterHandler：对预编译的SQL语句进行参数设置，SQL语句中的占位符“？”都对应BoundSql.parameterMappings集合中的一个元素，在该对象中记录了对应的参数名称以及该参数的相关属性
  - ResultSetHandler：对数据库返回的结果集（ResultSet）进行封装，返回用户指定的实体类型；



# Executor内部运作过程





# 目录

## CONTENTS



### 看源码的正确姿势

论学习源码的重要性  
学习源码的方法论



### MyBatis源码概述

**MyBatis**源码概述  
初始化阶段  
动态代理阶段  
数据库访问阶段



### 手写MyBatis解析

手写**MyBatis**解析

# 关键核心类



1. Configuration ;
2. SqlSessionFactory ;
3. SqlSession ;
4. Executor ;
5. StatmentHandler
6. ParameterHandler;
7. ResultSetHandler;