

bind方法的实现猜想

- 初始化NioServerSocketChannel
- 初始化EventLoop
- NioServerSocketChannel绑定监听端口
- NioServerSocketChannel注册到selector
- NioServerSocketChannel中的handler构建
- NioSocketChannel中的handler构建

ServerSocketChannel初始化过程

1. 验证参数的合法性
2. 通过ChannelFactory构建NioServerSocketChannel实例，而这个NioServerSocketChannel是根据ServerBootstrap初始化绑定的class有关系
3. 初始化NioServerSocketChannel中的Handler，Pipeline，这个pipeline中会把服务端的Handler添加到pipeline中 (init())
 1. ServerBootstrapAcceptor
4. register方法，将NioServerSocketChannel注册到NioEventLoop中的Selector中，并且监听的事件是 OP_ACCEPT
5. 如果register注册成功，则调用 ServerSocketChannel.bind方法，绑定ip和端口

图解服务启动初始化过程

图片地址: <https://www.processon.com/view/link/614b411d1efad43669d5cc14>

阅读源码

- 目的
 - 了解原理背后的实现方法，加深对原理的理解
 - 借鉴和学习落地的代码设计
 - 开源代码的维护
- 如何学习
 - 熟练并掌握该框架的使用
 - 了解该框架的基本原理
 - 下载源码阅读
 - 跟着源码部分的内容学习，找到源码阅读的方法
 - 自己多看
 - 边猜想边阅读

