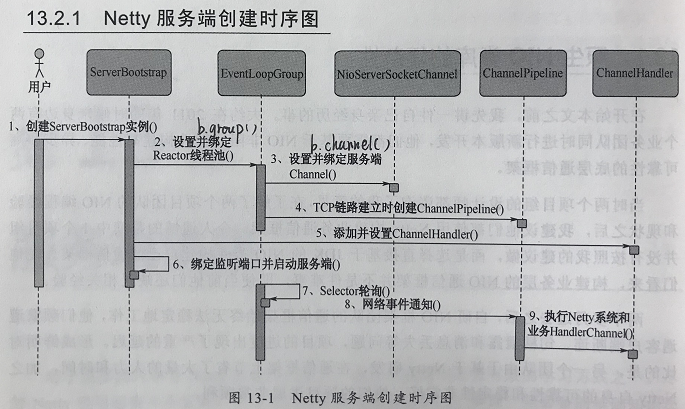
# Netty

Netty 是一款**异步的事件驱动**的网络应用程序框架，支持快速地开发可维护的高性能的面向协议的服务器和客户端。

# 服务端创建



## 创建ServerBootstrap

ServerBootstrap是Netty服务端的启动辅助类，提供了一系列的方法用于设置服务端启动相关的参数。

## 设置并绑定Reactor线程池

Netty的Reactor线程池是EventLoopGroup，实际就是**EventLoop的数组**。

EventLoop的职责是处理所有注册到**本线程多路复用器**Selector上的Channel，Selector的轮询操作由绑定的EventLoop线程run方法驱动，在一个循环体内循环执行。

## 设置并绑定服务器Channel

**b.channel();**

NIO服务端需要创建ServerSocketChannel，Netty对其进行了封装成为NioServerSocketChannel。

ServerBootstrap提供了channel方法用于指定服务器Channel的类型。Channel用于端口监听和客户端链路接入。

## 链路建立的时候创建并初始化ChannelPipeline

它本质就是一个负责处理网络事件的责任链，负责管理和执行ChannelHandler。网络事件以事件流的形式在ChannelPipeline中流转，ChannelPipeline调度ChannelHandler执行。

## 添加并设置ChannelHandler

ChannelHandler是用户定制和扩展的关键接口，可以完成消息编解码、心跳、安全认证、流量控制等。

## 绑定并启动监听端口

在绑定监听端口之前系统会做一系列初始化和检测工作。ServerSocketChannel注册到Selector上监听客户端连接。

## Selector轮询

由Reactor线程NioEventLoop负责调度和执行Selector轮询操作，选择就绪的Channel集合。

轮询到就绪的channel后，由Reactor线程NIOEventLoop执行ChannelPipeline的相应方法，最终调度并执行ChannelHandler。