

我是一段不羁的公告！  
记得给苏苏这 3 个项目加油，添加一个 STAR 噢。  
<https://github.com/YunaiV/SpringBoot-Labs>  
<https://github.com/YunaiV/onemail>  
<https://github.com/YunaiV/ruoyi-vue-pro>

• NETTY

# 精尽 Netty 源码分析 —— Netty 简介（一）之项目结构

## 1 概述

### 文章目录

- 1. 概述
- 2. 代码统计
- 3. 架构图
- 4. 项目依赖图
- 5. common
- 6. buffer
- 7. transport
- 8. codec
- 9. handler
- 10. example
- 11. 其它项目
- 666. 彩蛋

项目结构。  
对 Netty 的整体项目有个简单的了解。  
我们会发现拆分了**好多** Maven 项目。是不是内心一紧，产生了恐惧感？不要方，我们就是继续怼。





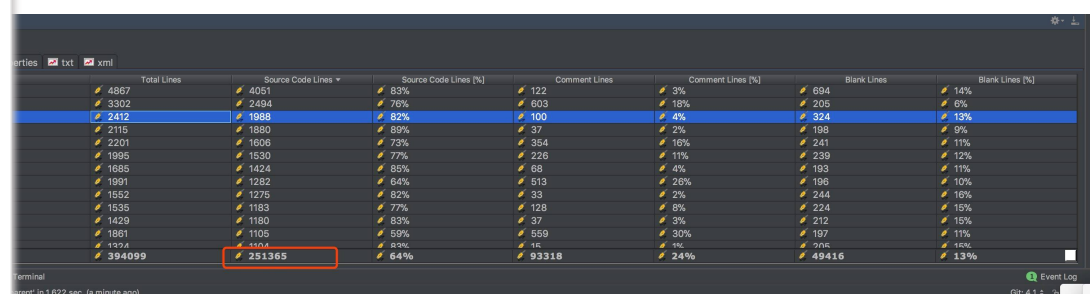
## 2. 代码统计

## 文章目录

1. 概述
2. 代码统计
3. 架构图
4. 项目依赖图
5. common
6. buffer
7. transport
8. codec
9. handler
10. example
11. 其它项目

下。笔者在开始源码学习时，会首先了解项目的代码量。

### 3. A Statistic 插件，统计整体代码量。



总的代码量在 251365 行。这其中还包括单元测试，示例等等代码。

ell 脚本命令逐个 Maven 模块统计。

一般情况下，笔者使用 `find . -name "*.java"|xargs cat|grep -v -e ^$ -e ^\s*\|\/\|.*$|wc -l`。这个命令只过滤了部分注释，所以相比 [IDEA Statistic](#) 会偏多。

当然，考虑到准确性，胖友需要手动 `cd` 到每个 Maven 项目的 `src/main/java` 目录下，以达到排除单元测试的代码量。

模块 1	模块 2	包 1	代码量 1
buffer	buffer		25005
codec	codec		21944
	codec-dns		
	codec-haproxy		
	codec-http		
	codec-http2		
	codec-memcache		
	codec-mqtt		
	codec-redis		
	codec-smtp		
	codec-socks		
	codec-stomp		
	codec-xml		
common	common		23064
example	example		
handler	handler		21504
	handler-proxy		
microbench	microbench		
resolver	resolver		
	resolver-dns		
testsuite	testsuite		
	testsuite-autobahn		
	testsuite-http2		
	testsuite-osgi		
transport	transport		27068
	transport-native-epoll		
	transport-native-kqueue		
	transport-native-unix-common		
	transport-native-unix-common-tests		
	transport-rxtx		
	transport-sctp		
	transport-udt		
合计			118585

- 😈 偷懒了下，暂时只统计**核心**模块，未统计**拓展**模块。

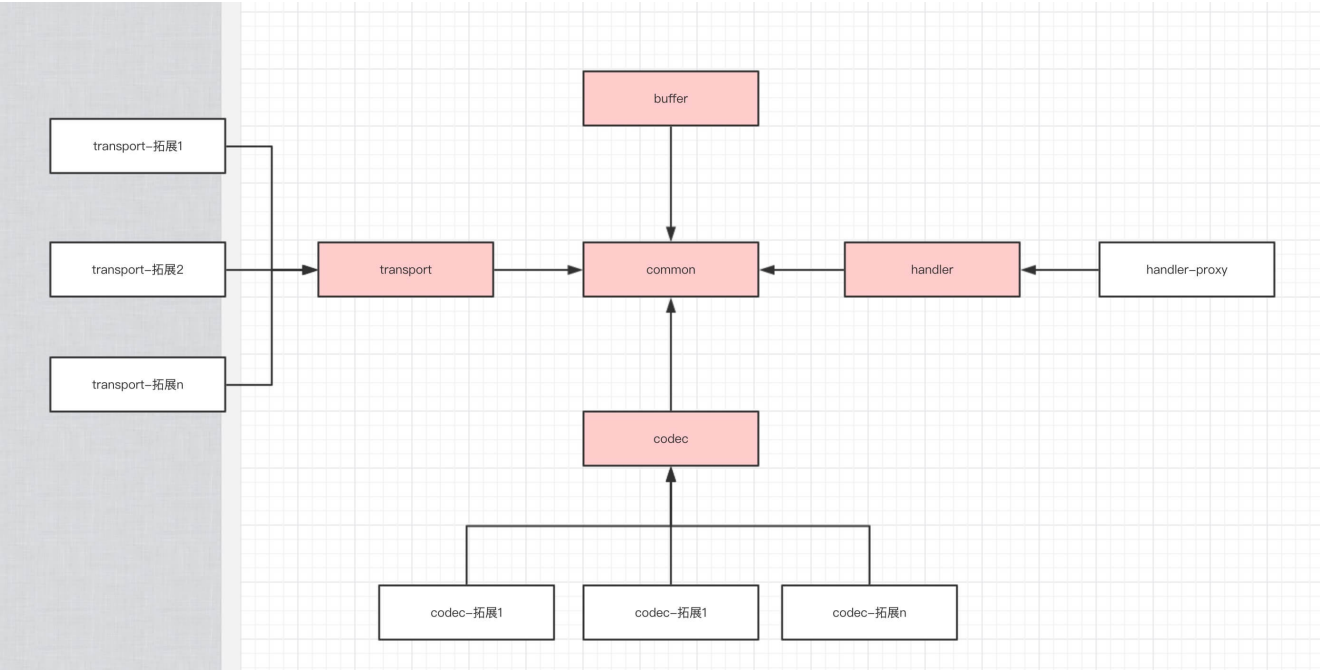
### 3. 架构图

在看具体每个 Netty 的 Maven 项目之前，我们还是先来看看 Netty 的整体架构图。



### 4. 项目依赖图

Netty 的 Maven 项目之间**主要依赖**如下图：



- 本图省略**非主要依赖**。例如， handler-proxy 对 codec 有依赖，但是并未画出。
- 本图省略**非主要的项目**。例如， resolver 、 testsuite 、 example 等等。

下面，我们来详细介绍每个项目。

## 5. common

`common` 项目，该项目是一个通用的工具类项目，几乎被所有的其它项目依赖使用，它提供了一些数据类型处理工具类，并发编程以及多线程的扩展，计数器等等通用的工具类。

### 文章目录

1. 概述
2. 代码统计
3. 架构图
4. 项目依赖图
5. common
6. buffer
7. transport
8. codec
9. handler
10. example
11. 其它项目
666. 彩蛋

# 并发

## 文章目录

1. 概述
  2. 代码统计
  3. 架构图
  4. 项目依赖图
  5. common
  6. buffer
  7. transport
  8. codec
  9. handler
  10. example
  11. 其它项目
666. 彩蛋

common [netty-common]

src

main

java

io.netty.util

concurrent

internal

AbstractConstant

AbstractReferenceCounted

AsciiString

AsyncMapping

Attribute

AttributeKey

AttributeMap

BooleanSupplier

ByteProcessor

ByteProcessorUtils

CharsetUtil

Constant

ConstantPool

DefaultAttributeMap

DomainMappingBuilder

DomainNameMapping

DomainNameMappingBuilder

HashedWheelTimer

HashingStrategy

IllegalReferenceCountException

IntSupplier

Mapping

NettyRuntime

NetUtil

package-info.java

Recycler

ReferenceCounted

ReferenceCountUtil

ResourceLeak

ResourceLeakDetector

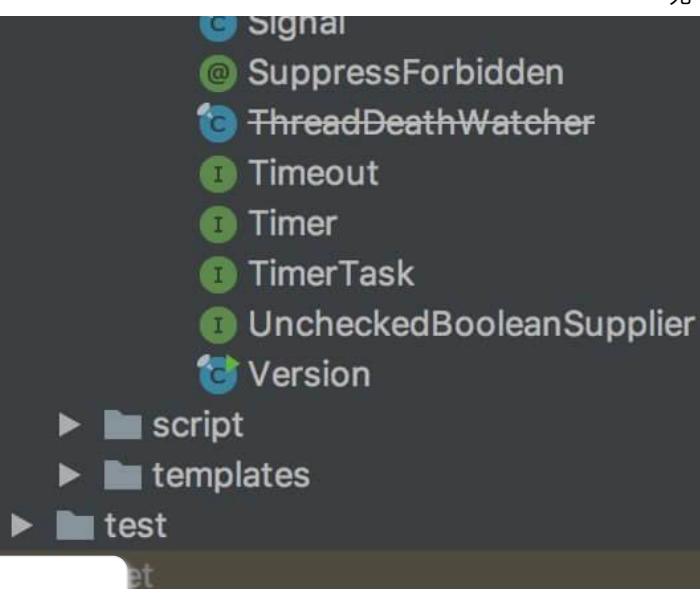
ResourceLeakDetectorFactory

ResourceLeakException

ResourceLeakHint

ResourceLeakTracker





### 文章目录

1. 概述
2. 代码统计
3. 架构图
4. 项目依赖图
5. common
6. buffer
7. transport
8. codec
9. handler
10. example
11. 其它项目
666. 彩蛋

见了 Netty 架构图中的 Zero-Copy-Capable Rich Byte Buffer 。

目下是 Netty 自行实现的一个 Byte Buffer 字节缓冲区。该包的实现相对于 JDK 自带的 ByteBuffer 有很  
的功能，使用体验，性能都要更加优秀。它提供了一系列(多种)的抽象定义以及实现，以满足不同场景

▼ buffer [netty-buffer]

▼ src

▼ main

▼ java

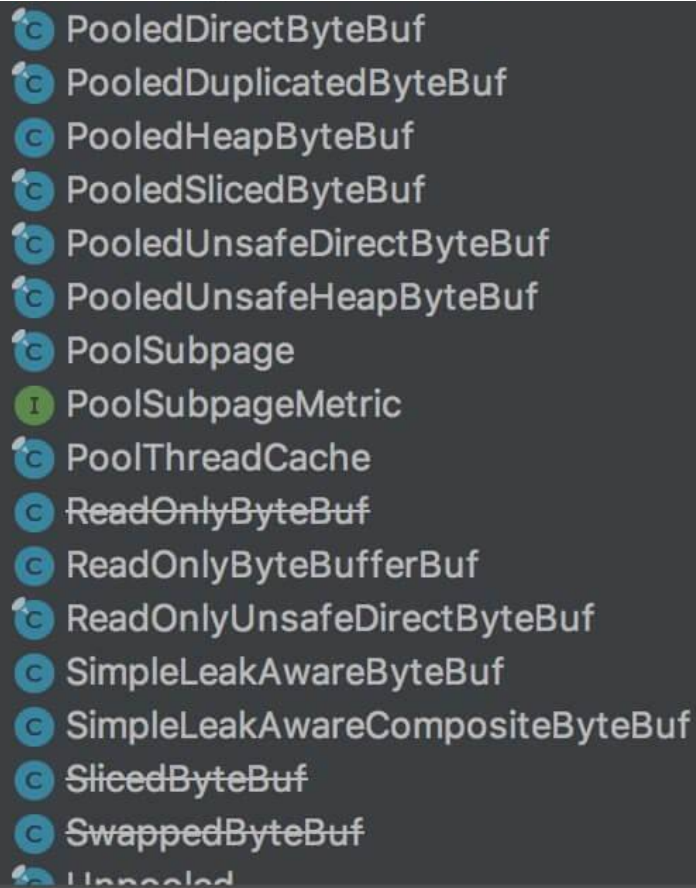
▼ io.netty.buffer

- (C) AbstractByteBuf
- (C) AbstractByteBufAllocator
- (C) AbstractDerivedByteBuf
- (C) AbstractPooledDerivedByteBuf
- (C) AbstractReferenceCountedByteBuf
- (C) AbstractUnpooledSlicedByteBuf
- (C) AbstractUnsafeSwappedByteBuf
- (C) AdvancedLeakAwareByteBuf
- (C) AdvancedLeakAwareCompositeByteBuf
- (C) **ByteBuf**
- (I) ByteBufAllocator
- (I) ByteBufAllocatorMetric
- (I) ByteBufAllocatorMetricProvider
- (I) ByteBufHolder
- (C) ByteBufInputStream
- (C) ByteBufOutputStream
- (I) ByteBufProcessor
- (C) ByteBufUtil
- (C) CompositeByteBuf
- (C) DefaultByteBufHolder
- (C) DuplicatedByteBuf
- (C) EmptyByteBuf
- (C) FixedCompositeByteBuf
- (C) HeapByteBufUtil
- package-info.java
- (C) PoolArena
- (I) PoolArenaMetric
- (C) PoolChunk
- (C) PoolChunkList
- (I) PoolChunkListMetric
- (I) PoolChunkMetric
- (C) PooledByteBuf
- (C) PooledByteBufAllocator
- (C) PooledByteBufAllocatorMetric

### 文章目录

1. 概述
  2. 代码统计
  3. 架构图
  4. 项目依赖图
  5. common
  6. buffer
  7. transport
  8. codec
  9. handler
  10. example
  11. 其它项目
666. 彩蛋





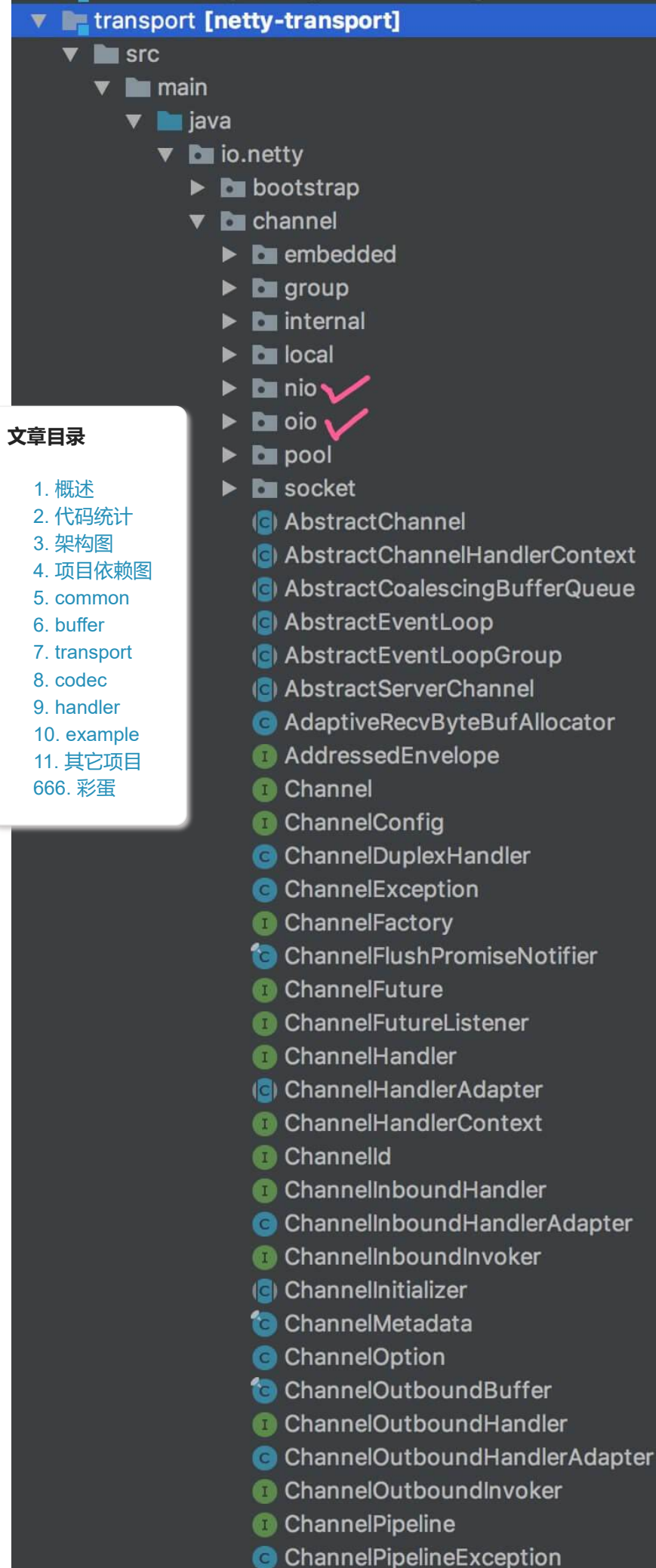
PooledDirectByteBuffer  
PooledDuplicatedByteBuffer  
PooledHeapByteBuffer  
PooledSlicedByteBuffer  
PooledUnsafeDirectByteBuffer  
PooledUnsafeHeapByteBuffer  
PoolSubpage  
PoolSubpageMetric  
PoolThreadCache  
ReadOnlyByteBuffer  
ReadOnlyByteBufferBuf  
ReadOnlyUnsafeDirectByteBuffer  
SimpleLeakAwareByteBuffer  
SimpleLeakAwareCompositeByteBuffer  
SlicedByteBuffer  
SwappedByteBuffer  
Unpooled

### 文章目录

1. 概述
2. 代码统计
3. 架构图
4. 项目依赖图
5. common
6. buffer
7. transport
8. codec
9. handler
10. example
11. 其它项目
666. 彩蛋

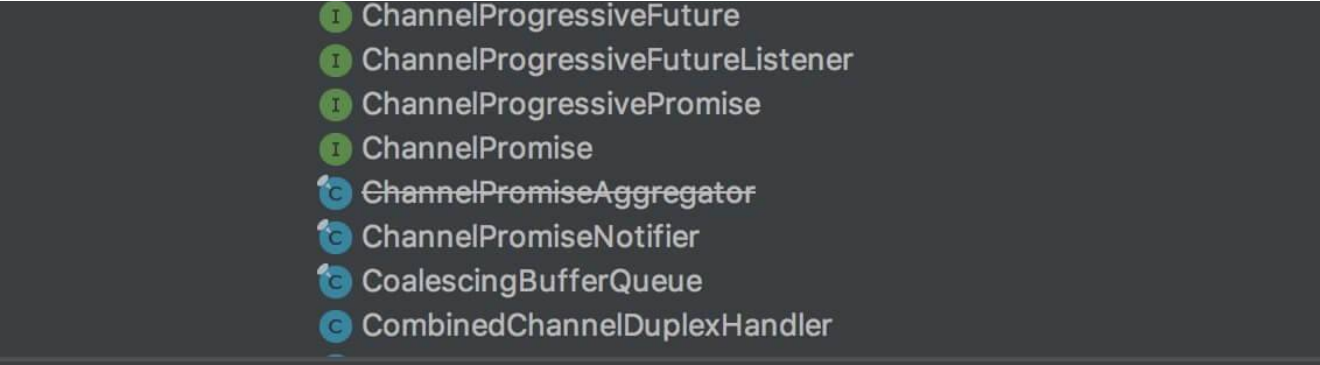
本项目，实现了 Netty 架构图中 Transport Services、Universal Communication API 和 Extensible Event Model 等多部分内容。

transport 项目，该项目是网络传输通道的抽象和实现。它定义通信的统一通信 API，统一了 JDK 的 OIO、NIO (不包括 AIO) 等多种编程接口。



### 文章目录

1. 概述
  2. 代码统计
  3. 架构图
  4. 项目依赖图
  5. common
  6. buffer
  7. transport
  8. codec
  9. handler
  10. example
  11. 其它项目
666. 彩蛋



- ChannelProgressiveFuture
- ChannelProgressiveFutureListener
- ChannelProgressivePromise
- ChannelPromise
- ChannelPromiseAggregator
- ChannelPromiseNotifier
- CoalescingBufferQueue
- CombinedChannelDuplexHandler

另外，它提供了多个子项目，实现不同的传输类型。例如： `transport-native-epoll` 、 `transport-native-kqueue` 、 `transport-rxtx` 、 `transport-udt` 和 `transport-sctp` 等等。

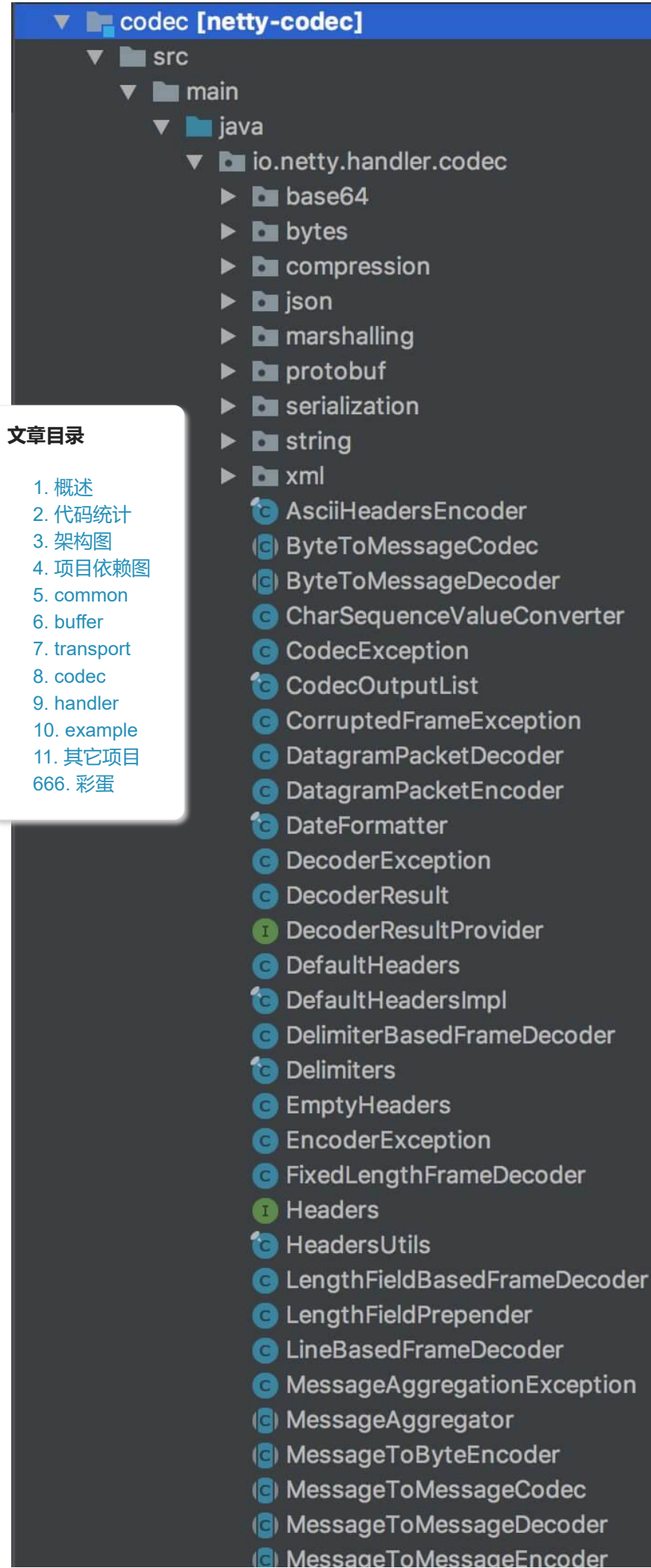
## 8. codec

### 文章目录

- 1. 概述
- 2. 代码统计
- 3. 架构图
- 4. 项目依赖图
- 5. common
- 6. buffer
- 7. transport
- 8. codec
- 9. handler
- 10. example
- 11. 其它项目
- 666. 彩蛋

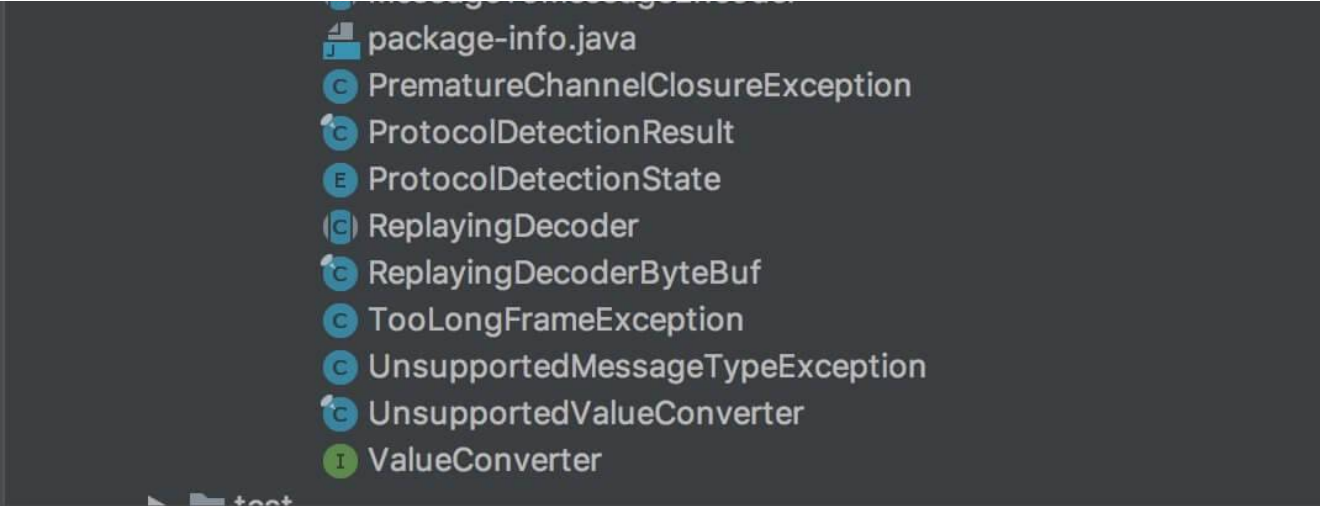
见了Netty 架构图中的 Protocol Support 。

是协议编解码的抽象与**部分**实现：JSON、Google Protocol、Base64、XML 等等。



### 文章目录

1. 概述
  2. 代码统计
  3. 架构图
  4. 项目依赖图
  5. common
  6. buffer
  7. transport
  8. codec
  9. handler
  10. example
  11. 其它项目
666. 彩蛋

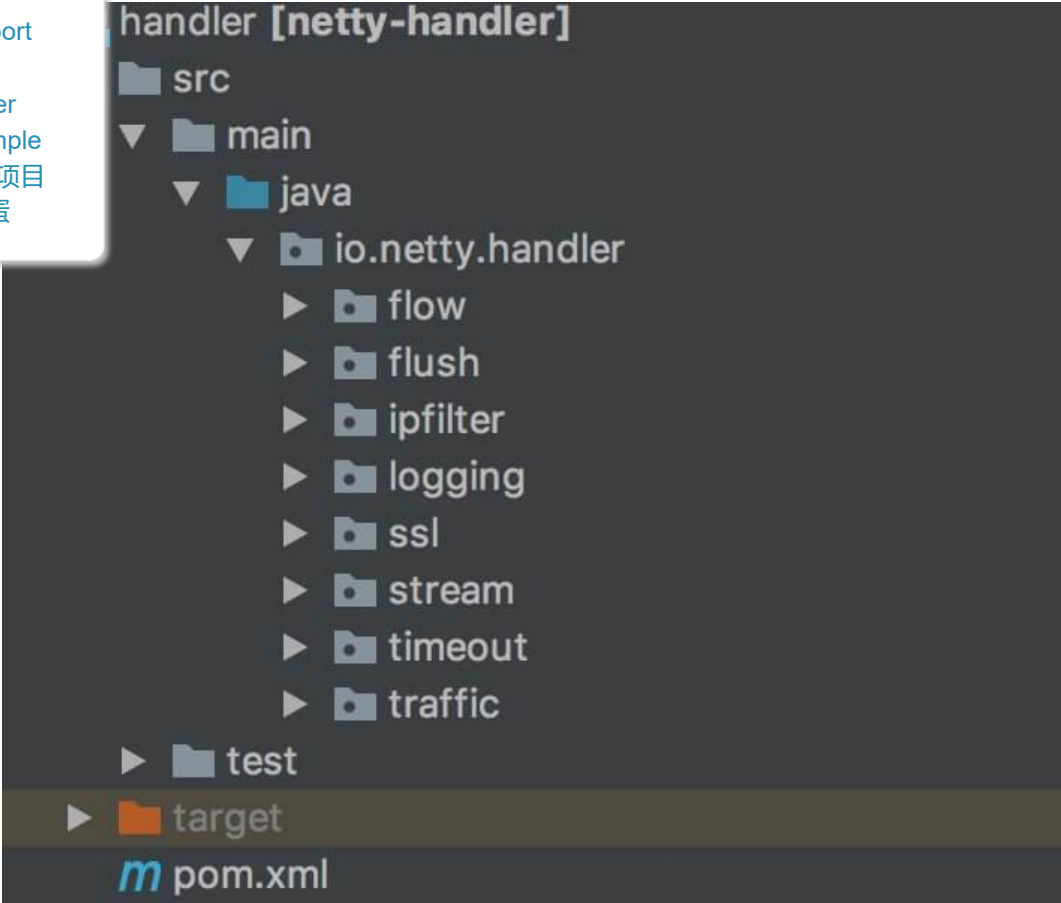


另外，它提供了多个子项目，实现不同协议的编解码。例如： codec-dns 、 codec-haproxy 、 codec-http 、 codec-redis 、 codec-memcached 、 codec-smtp 、 codec-socks 、 codec-stomp 、

文章目录

- 1. 概述
- 2. 代码统计
- 3. 架构图
- 4. 项目依赖图
- 5. common
- 6. buffer
- 7. transport
- 8. codec
- 9. handler
- 10. example
- 11. 其它项目
- 666. 彩蛋

项目是提供内置的连接通道处理器( ChannelHandler )实现类。例如： SSL 处理器、日志处理器等等。



另外，它提供了一个子项目 handler-proxy ，实现对 HTTP、Socks 4、Socks 5 的代理转发。

## 10. example

example 项目，该项目是提供各种 Netty 使用示例，良心开源项目。



### 文章目录

1. 概述
2. 代码统计
3. 架构图
4. 项目依赖图
5. common
6. buffer
7. transport
8. codec
9. handler
10. example
11. 其它项目
666. 彩蛋

## 11. 其它项目

Netty 中还有其它项目，考虑到不是本系列的重点，就暂时进行省略。

- all : All In One 的 pom 声明。
- bom : Netty Bill Of Materials 的缩写，不了解的胖友，可以看看 [《Maven 与Spring BOM\( Bill Of Materials \)简化Spring 版本控制》](#)。
- microbench : 微基准测试。
- resolver : 终端( Endpoint )的地址解析器。
- resolver-dns
- tarball : All In One 打包工具。
- testsuite : 测试集。

测试集( TestSuite )：测试集是把多个相关测试归入一个组的表达方式。在 Junit 中，如果我们没有明确的定义一个测试集，那么 Juint 会自动的提供



一个测试集。一个测试集一般将同一个包的测试类归入一组。

- testsuite-autobahnn
- testsuite-http2
- testsuite-osgi

## 666. 彩蛋

本文基于杨武兵大佬的 [《Netty 源码分析系列 —— 概述》](#) 进行修改。

### 文章目录

- 1. 概述
- 2. 代码统计
- 3. 架构图
- 4. 项目依赖图
- 5. common
- 6. buffer
- 7. transport
- 8. codec
- 9. handler
- 10. example
- 11. 其它项目
- 666. 彩蛋

访客数 次 && 总访问量 次