

CommunityOverCode

THE ASF CONFERENCE

Tomcat的技术内幕和在喜马拉雅的实践

 彭荣新，喜马拉雅资深架构师

CONTENTS

1. IO Thread Model
2. Memory Model
3. Classload Feature
4. Performance Optimization



CommunityOverCode

THE ASF CONFERENCE

Part 01

IO Thread Model

IO Thread Model—Protocol

> Http

> Http nio

> 同步IO

> 大部分非阻塞

> Sendfile 零copy

> http nio2

> 用户态实现的伪AIO

> 同步IO

> APR

> 堆外内存

> 依赖本地库实现

> Http2

> base https

> WebSocket

> Nginx + tomcat

> Https

> 很少用

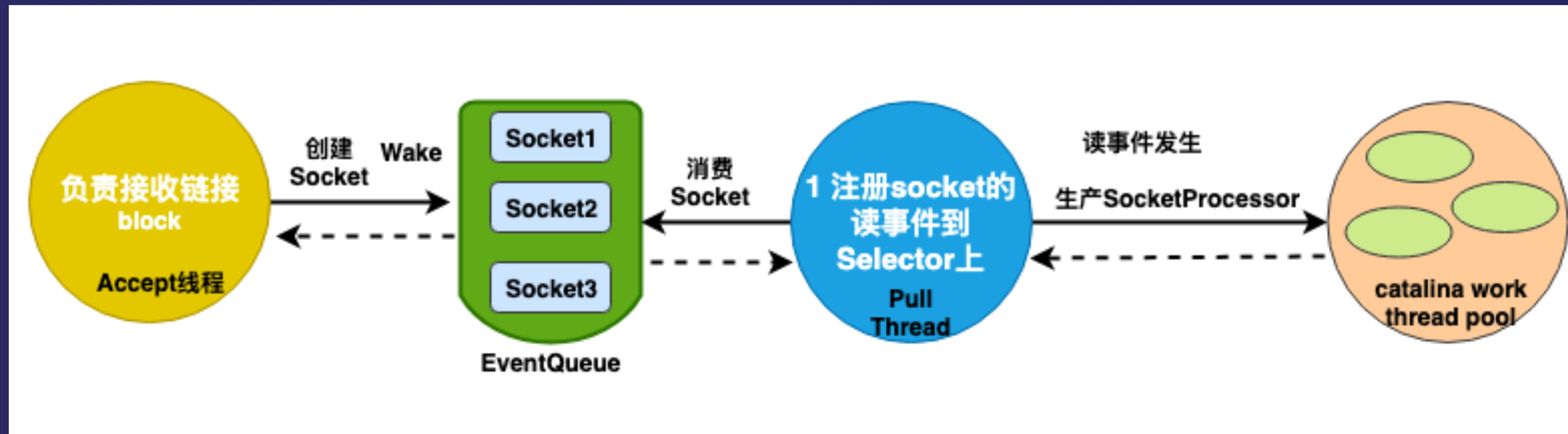
IO Thread Model—NIO

> Accept Thread

- > Accept connection
- > Put EventQueue

> Wake Pull Thread

- > No connection blocked

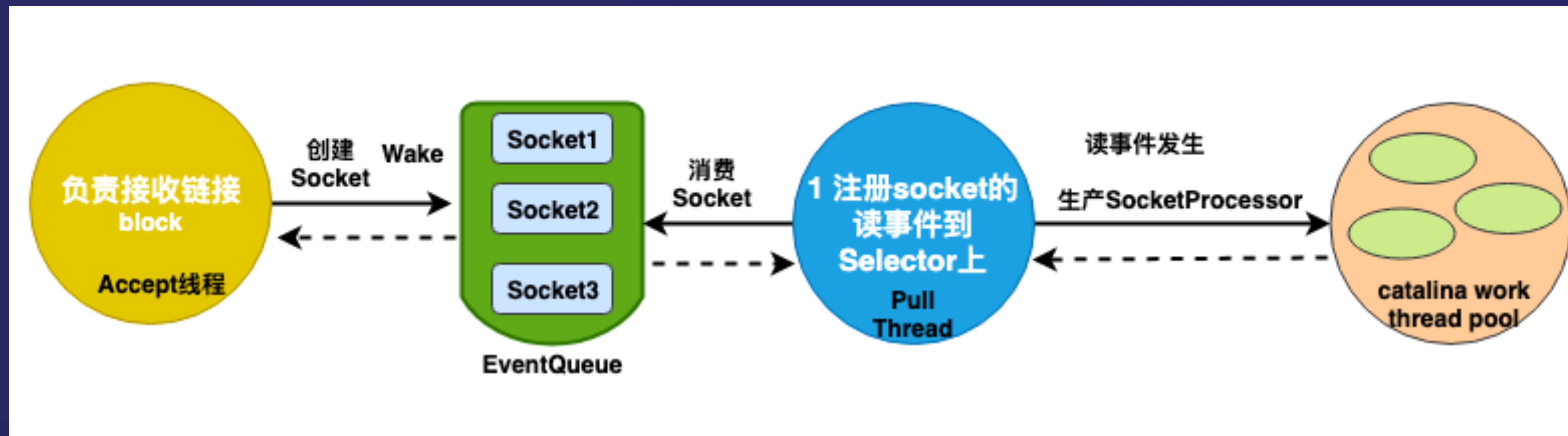


IO Thread Model—NIO

> Pull Thread

- > Register read event to selector
- > Create SocketProcessor submit Catalina thread pool

- > unregister read event from selector



IO Thread Model—NIO

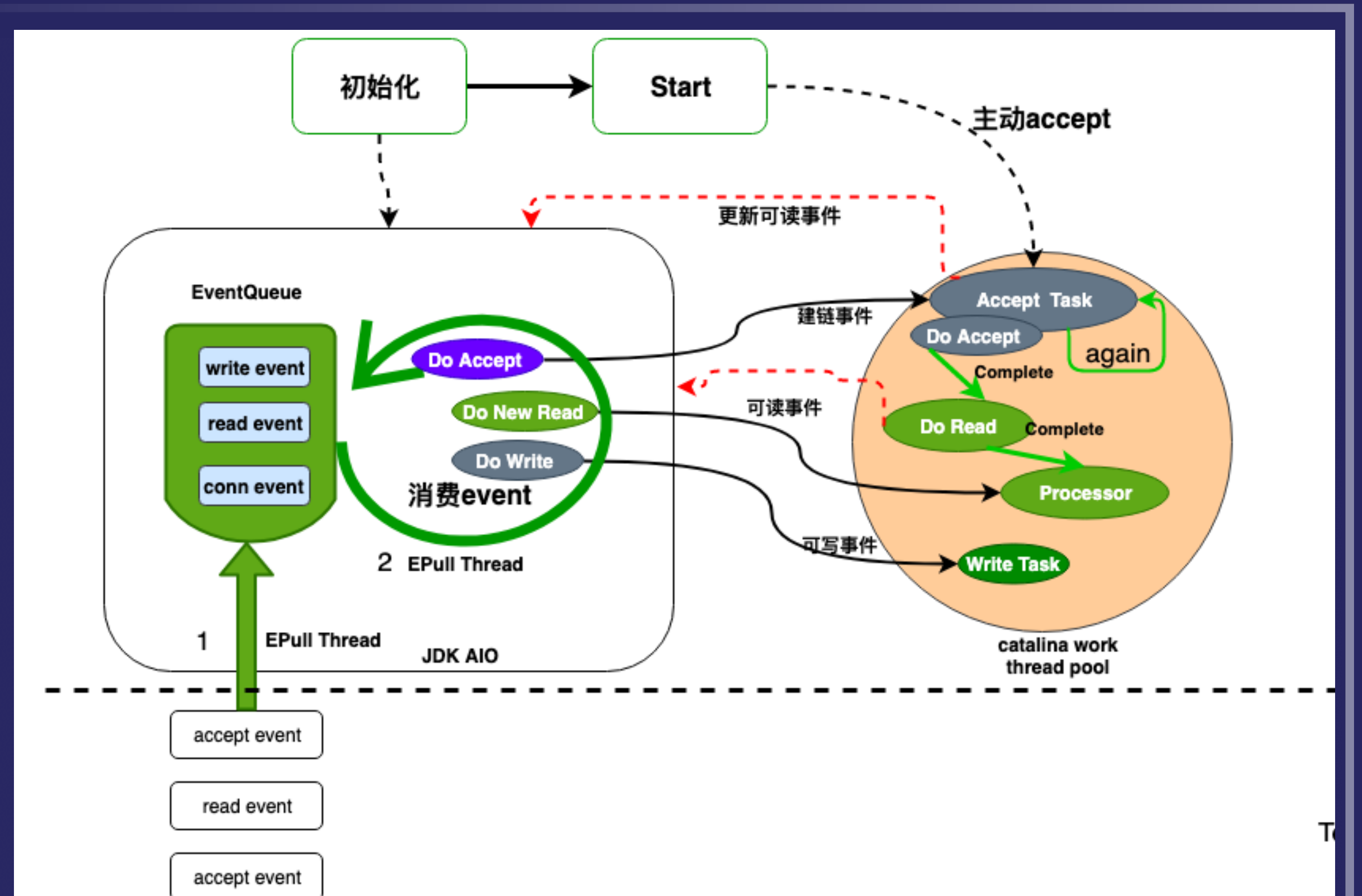
> Catalina Work Thread

- > Parse req line no block
- > Parse header no block
- > Router for filter
- > Router for servlet
- > Reader body blocked

```
while (!timedout) {
    if (keycount > 0) { //only read if we were registered for a read
        read = socket.read(buf);
        if (read != 0) {
            break;
        }
    }
    try {
        if (att.getReadLatch()==null || att.getReadLatch().getCount()==0) {
            att.startReadLatch( cnt: 1);
        }
        //注册读事件到block selector, 等待数据到来被唤醒。
        poller.add(att, SelectionKey.OP_READ, reference);
        //阻塞当前线程, 如果后面数据来了, 这个poller会通知该socket的latch, 继续执行
        att.awaitReadLatch( AbstractEndpoint.toTimeout(readTimeout), TimeUnit.MILLISECONDS);
    } catch (InterruptedException ignore) {
        // Ignore
    }
}
if ( att.getReadLatch()!=null && att.getReadLatch().getCount()> 0) {
    //检查是否被中断唤醒的
    //we got interrupted, but we haven't received notification from the poller.
    keycount = 0;
} else {
    //latch countdown has happened
    keycount = 1;
    att.resetReadLatch();
}
if (readTimeout >= 0 && (keycount == 0)) {
    timedout = (System.currentTimeMillis() - time) >= readTimeout;
}
}
```


IO Thread Model—NIO2

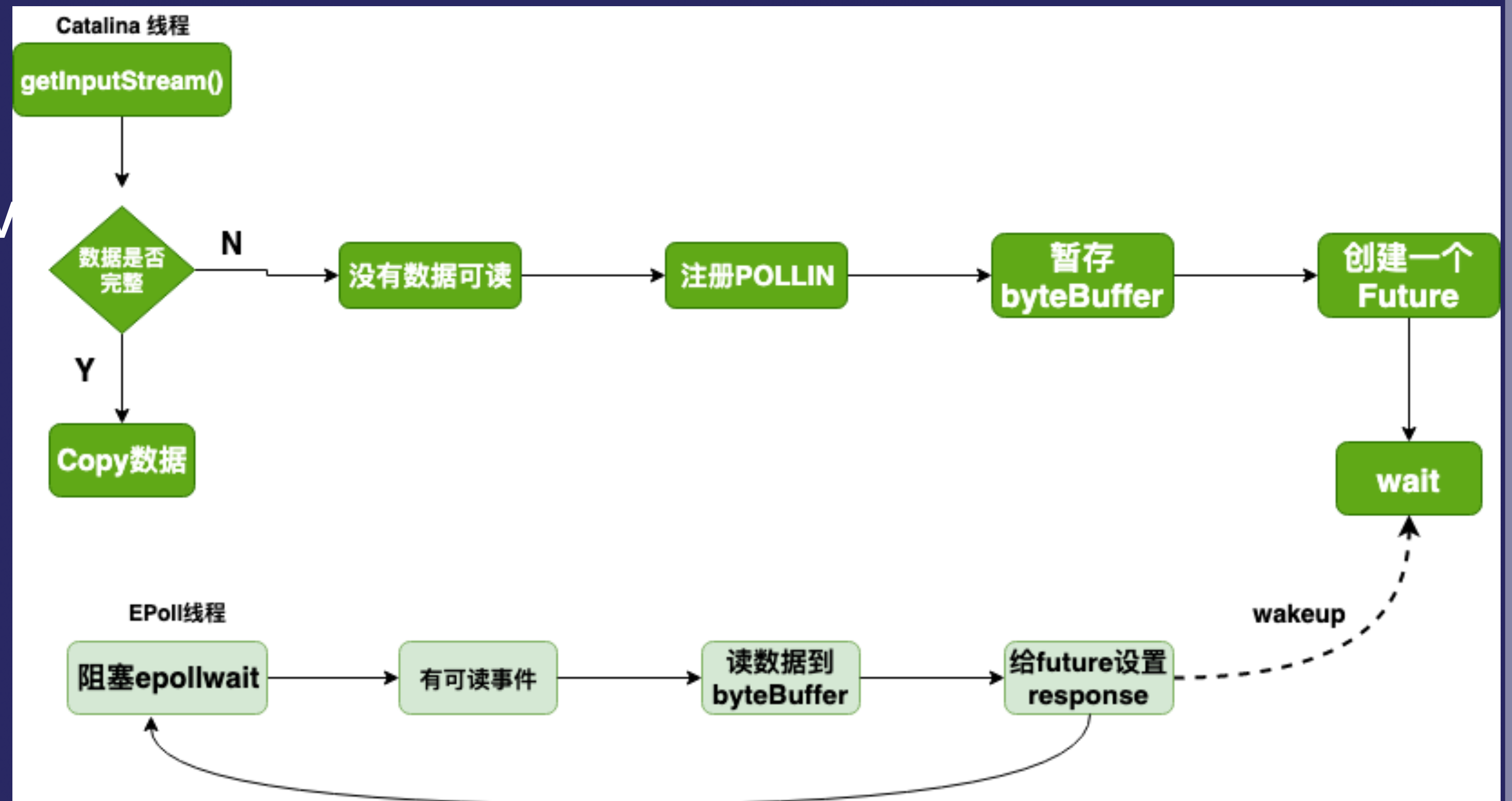
- > EPuller Thread
 - > Receiver put eventQueue
 - > Accept connection
 - > Data available pool thread read
 - > No data register epoll event and poll thread read
 - > Blocked at poll wait



IO Thread Model—NIO2

> Catalina Thread

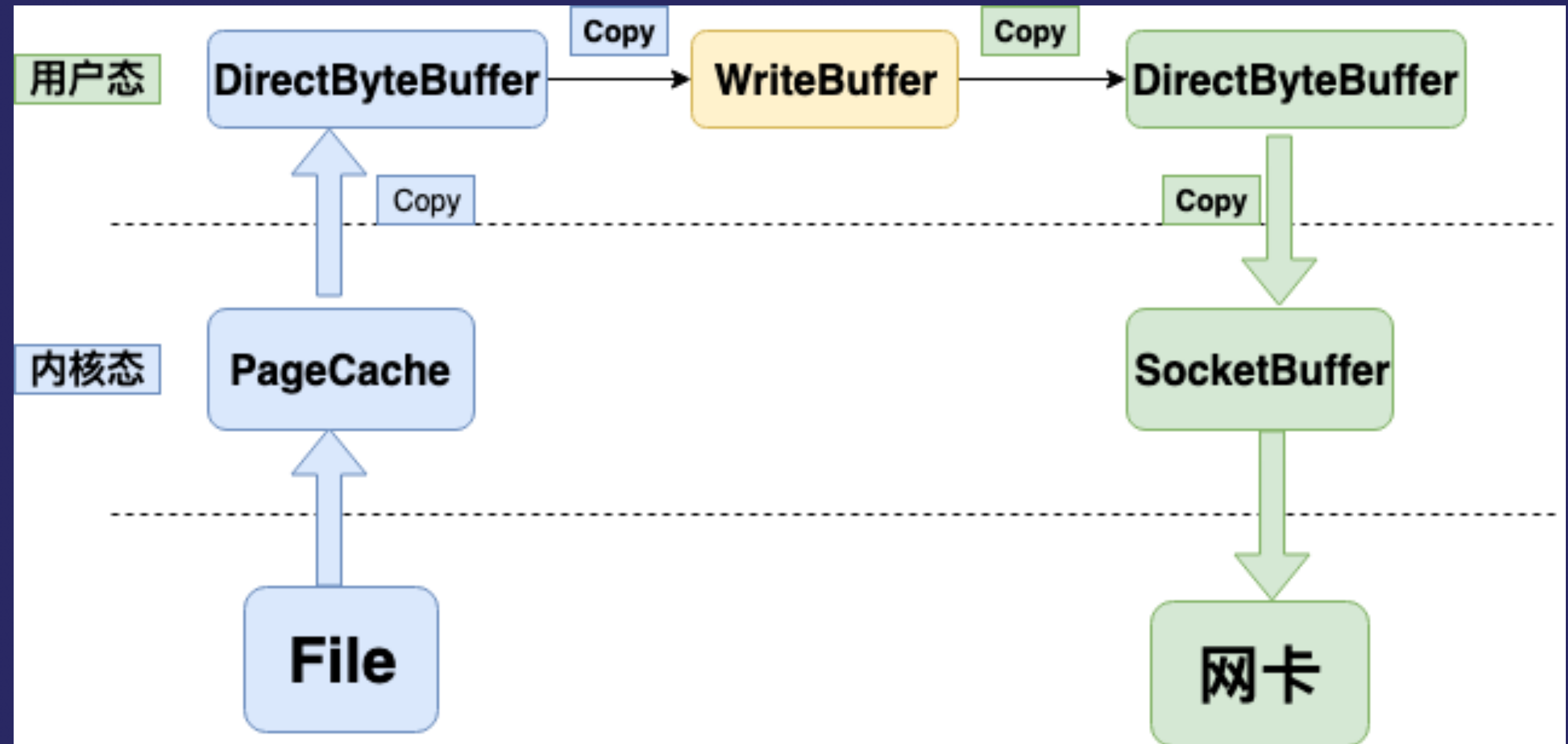
- > First accept connection
- > No data register epoll event
- > Read body blocked



IO Thread Model—NIO2

> SendFile

- > 普通IO Read/Write
- > 读2次Copy
- > 写2次Copy



IO Thread Model—Compare

> NIO2 VS NIO

> 优势

- > Context Switch 减少
- > System call次数减少

> NIO2 or NIO

- > Tomcat 9,10 default nio
- > Suggest nio for stable

> 缺点

- > 容易受慢客户端影响
- > EPoll read 容易成为瓶颈
- > 增加了复杂性
- > SendFile 性能低

CommunityOverCode

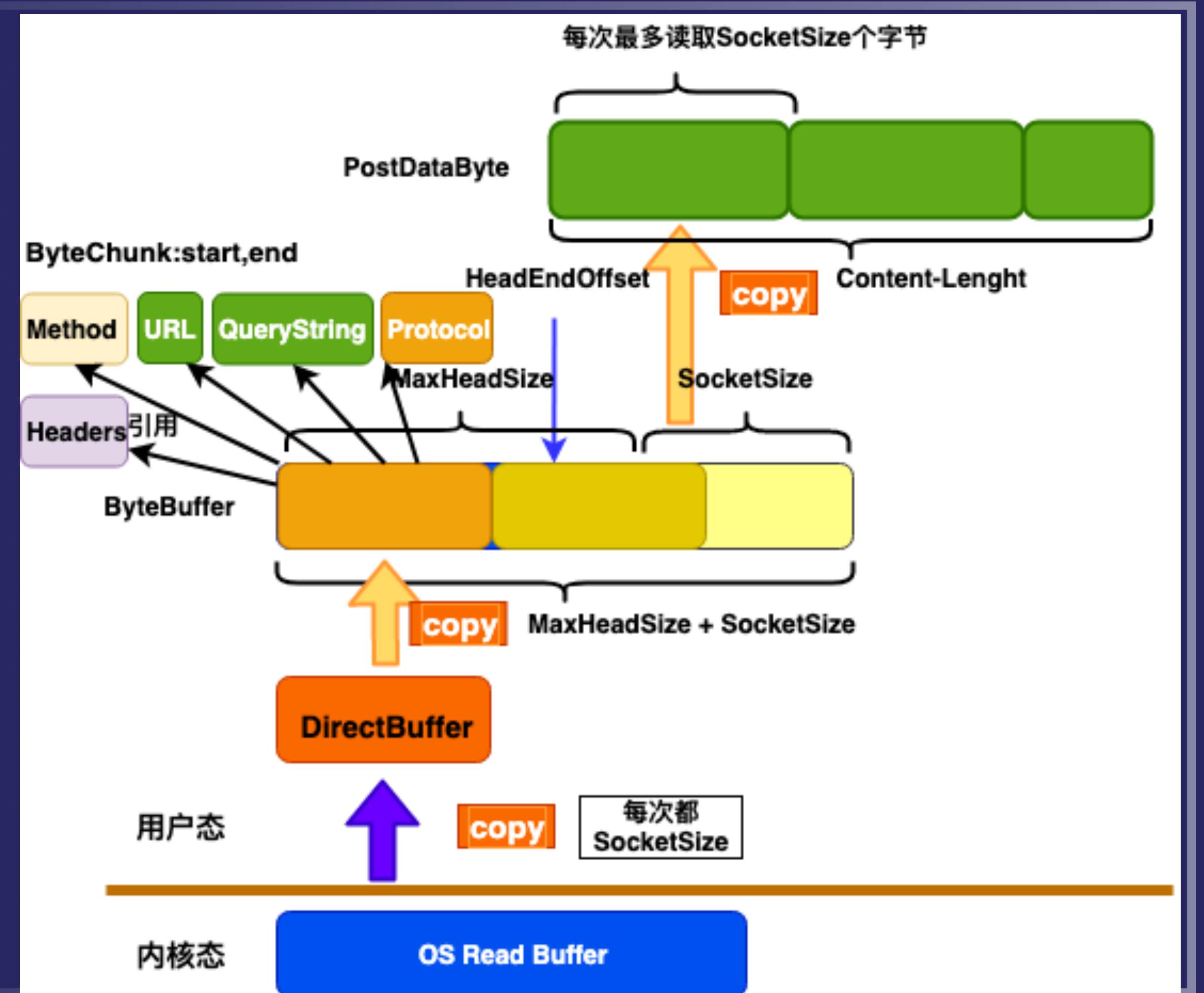
THE ASF CONFERENCE

Part 02

Tomcat Memory Model

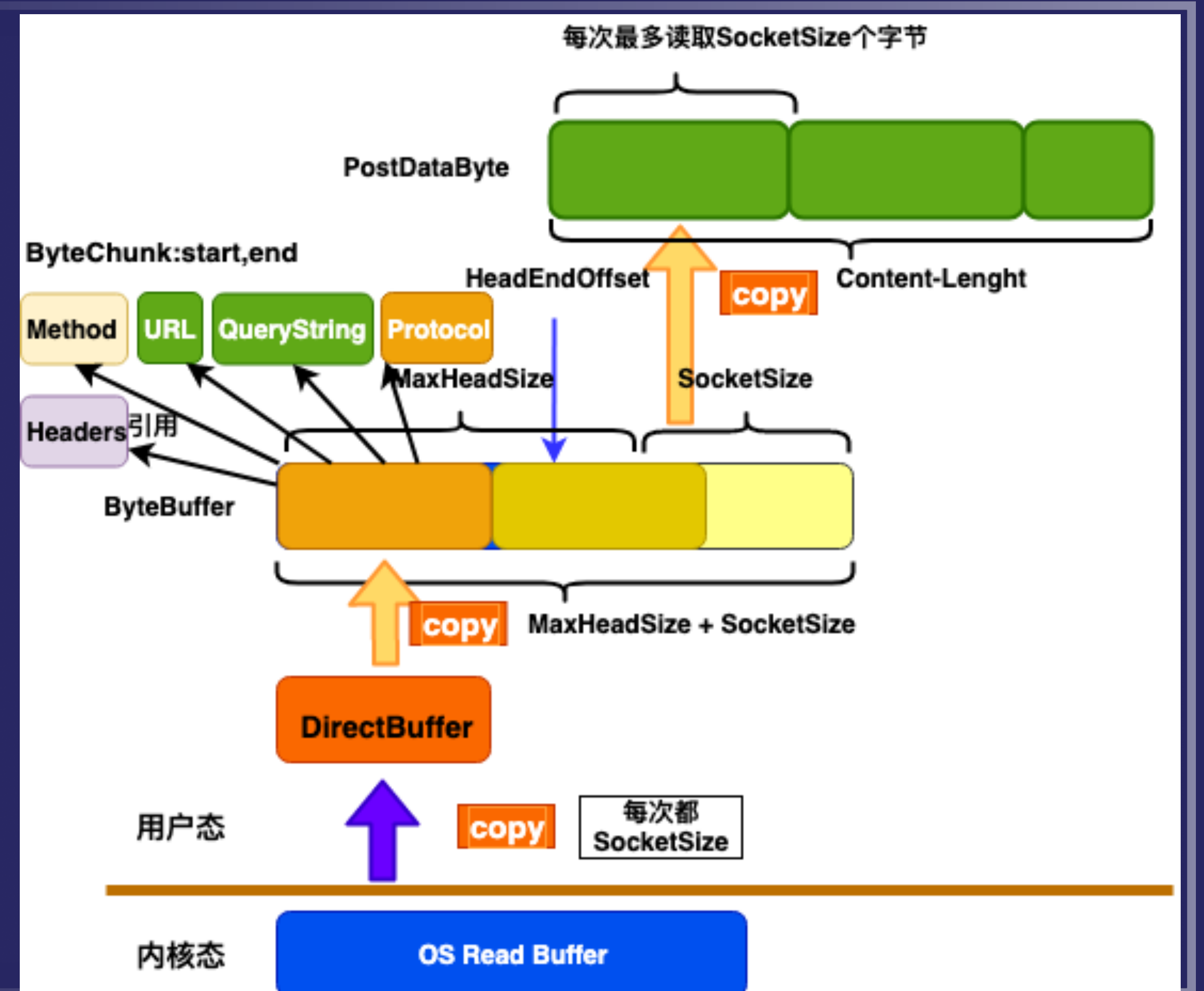
Tomcat Memory Model—Buffer

- > **ByteBuffer Req**
 - > Cache req line, header and body
 - > MaxHeadSize default 8k
 - > SocketSize default 8k
 - > Every req create one



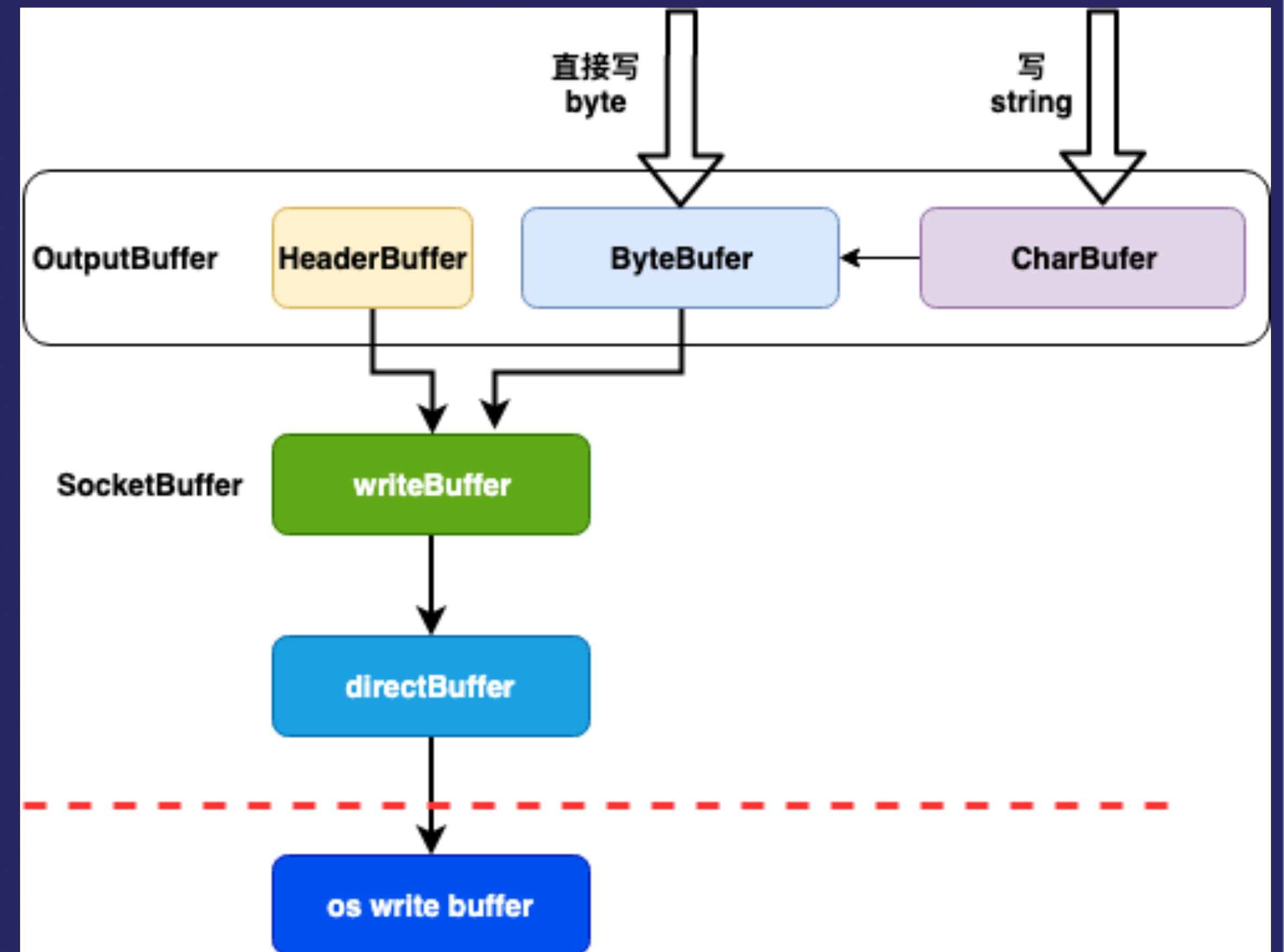
Tomcat Memory Model—Buffer

- > ByteBuffer Release
- > Request ByteChunk set 0
- > ByteBuffer limit, position set 0
- > ByteBuffer return pool



Tomcat Memory Model—Write

- > **OutputBuffer**
 - > headerBuffer for line and header
 - > Bytebuffer for body
- > **SocketBuffer**
 - > 聚合header and body
- > **Body Encode**
 - > Transfer-Encoding: chunked
 - > Content-Length



CommunityOverCode

THE ASF CONFERENCE

Part 03

ClassLoader Feature

ClassLoader

> ClassPath

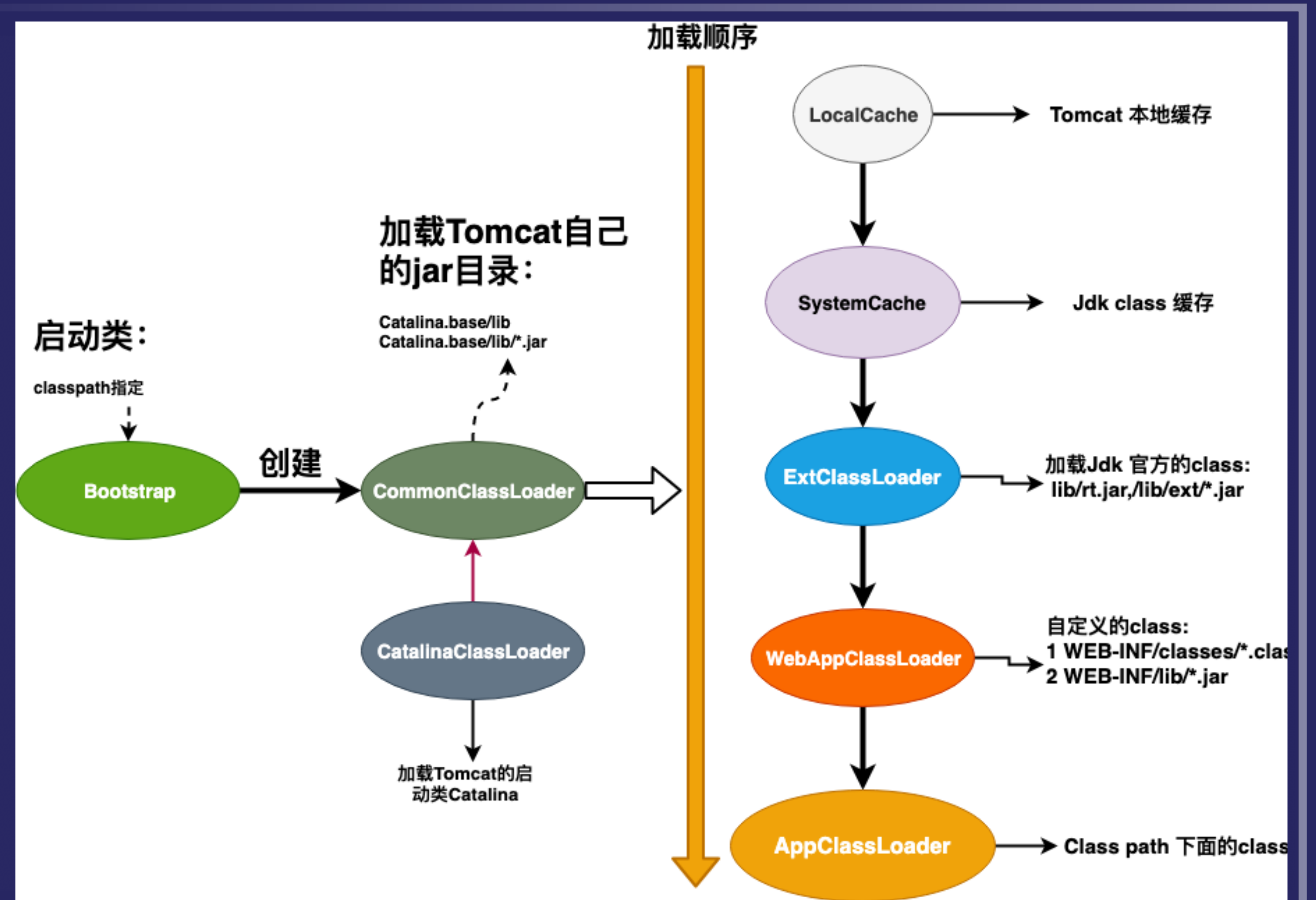
> home/bin/bootstrap.jar

> home/bin/tomcat-juli.jar

> CommonClassLoader

> home/lib

> home/lib/*.jar



CommunityOverCode

THE ASF CONFERENCE

Part 04

Performance optimization

Performance optimization—Cache

> NioChannels

- > Cache socks channel

- > Default size 128, limit 500, size >128 do expand

- > Tomcat 10 maxMemory / 32

> Nio1 EventCaches

- > Cache poll event

- > size 128, limit 500, size >128 do expand

- > Tomcat 10 default 0 disabled

Performance optimization—Cache

> ProcessorCache

- > Cache catalina task socketProcessor
- > default size 128, limit 500, size >128 do expand
- > Tomcat 10 default 0 disabled














> RecycledProcessor

- > Cache http11Processor
- > default size 200, limit -1 no limit
- > Register jmx for collected global request
- > Exist memory leak at hight concurrent

Performance optimization

```
@SuppressWarnings("sync-override") // Size may exceed cache size a Reader I
@Override
public boolean push(Processor processor) {
    int cacheSize = handler.getProtocol().getProcessorCache();
    //这里存在并发问题，并发高的情况下，会存在超量。
    boolean offer = cacheSize == -1 ? true : size.get() < cacheSize;
    // avoid over growing our cache or add after we have stopped
    boolean result = false;
    if (offer) {
        //limit 是-1, push都能成功。
        result = super.push(processor);
        if (result) {
            size.incrementAndGet();
        }
    }
    //push 成功，就不unregister，超量的不能释放
    if (!result) {
        handler.unregister(processor);
    }
}
```


Performance optimization

Class Name	Ref. Objects	Shallow Heap
 java.lang.Thread @ 0x7116bd4b0 catalina-exec-362 Thread	1,462	112
 <Java Local> org.apache.coyote.http11.Http11Protocol\$Http11ConnectionHandler @ 0x70e45bed0	1,461	32
 global org.apache.coyote.RequestGroupInfo @ 0x70e45cc88	1,459	56
 processors java.util.ArrayList @ 0x70e4608b0	1,459	24
 elementData java.lang.Object[1529] @ 0x76d811160	1,459	6,136
 [1332] org.apache.coyote.RequestInfo @ 0x76d07f128 »	1	88
 [1065] org.apache.coyote.RequestInfo @ 0x76c4239e0 »	1	88
 [939] org.apache.coyote.RequestInfo @ 0x76d3cd618 »	1	88
 [632] org.apache.coyote.RequestInfo @ 0x75eff6688 »	1	88
 [286] org.apache.coyote.RequestInfo @ 0x757517138 »	1	88
 [755] org.apache.coyote.RequestInfo @ 0x76a19b600 »	1	88
 [179] org.apache.coyote.RequestInfo @ 0x75888b6f8 »	1	88
 [610] org.apache.coyote.RequestInfo @ 0x76a19b9c8 »	1	88

Performance Optimization

- Web Container
 - Tomcat 镜像
 - 个性化配置
 - 线程数, 阻塞型web服务可以2k
 - Backlog, 连接队列1024, gc影响建连
 - maxKeepAliveRequests 1024, 提升连接复用次数
 - 超过次数response connection:closed header
 - 需要close 否则400
 - maxHttpHeaderSize 32k , 和网关一致, 减少400 code

CommunityOverCode

THE ASF CONFERENCE

Thanks

彭荣新 喜马拉雅资深架构师

微信号: gavinpeng2014

WWW.COMMUNITYOVERCODE.ORG

