

# Ch. 7 从设计到实现

2017.5

# 进度回顾





# 比特币勒索 Wanna Cry



## § 7.0 Hacker & Cracker

- 技术高手
- 高手+大坏蛋



Dennis Ritchie



Edsger Wybe Dijkstra



Kevin Mitnick



# 计算机病毒

- Virus
- Worm
- Trojan(Trojan Horse)

# 目录

- § 7.1 设计的层次
- § 7.2 大型软件项目的架构
- § 7.3 一些实现方式



# 设计的层次

- 架构设计
- 模块设计
- 类设计
- 流程设计
- 代码结构设计

# Architecture

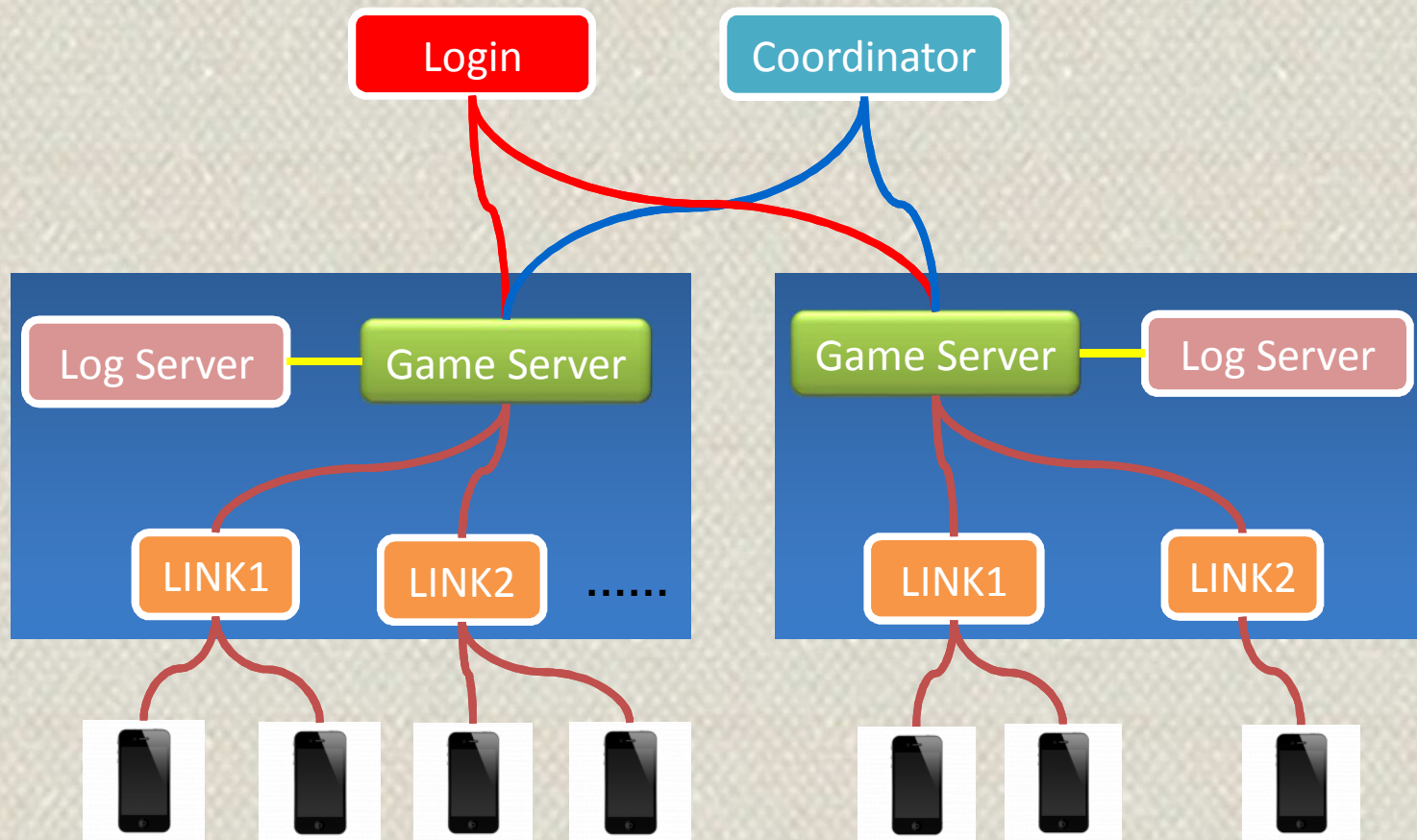
- 系统结构
- 子系统
- 大模块
- 各大模块之间的关联



# 架构设计的表达

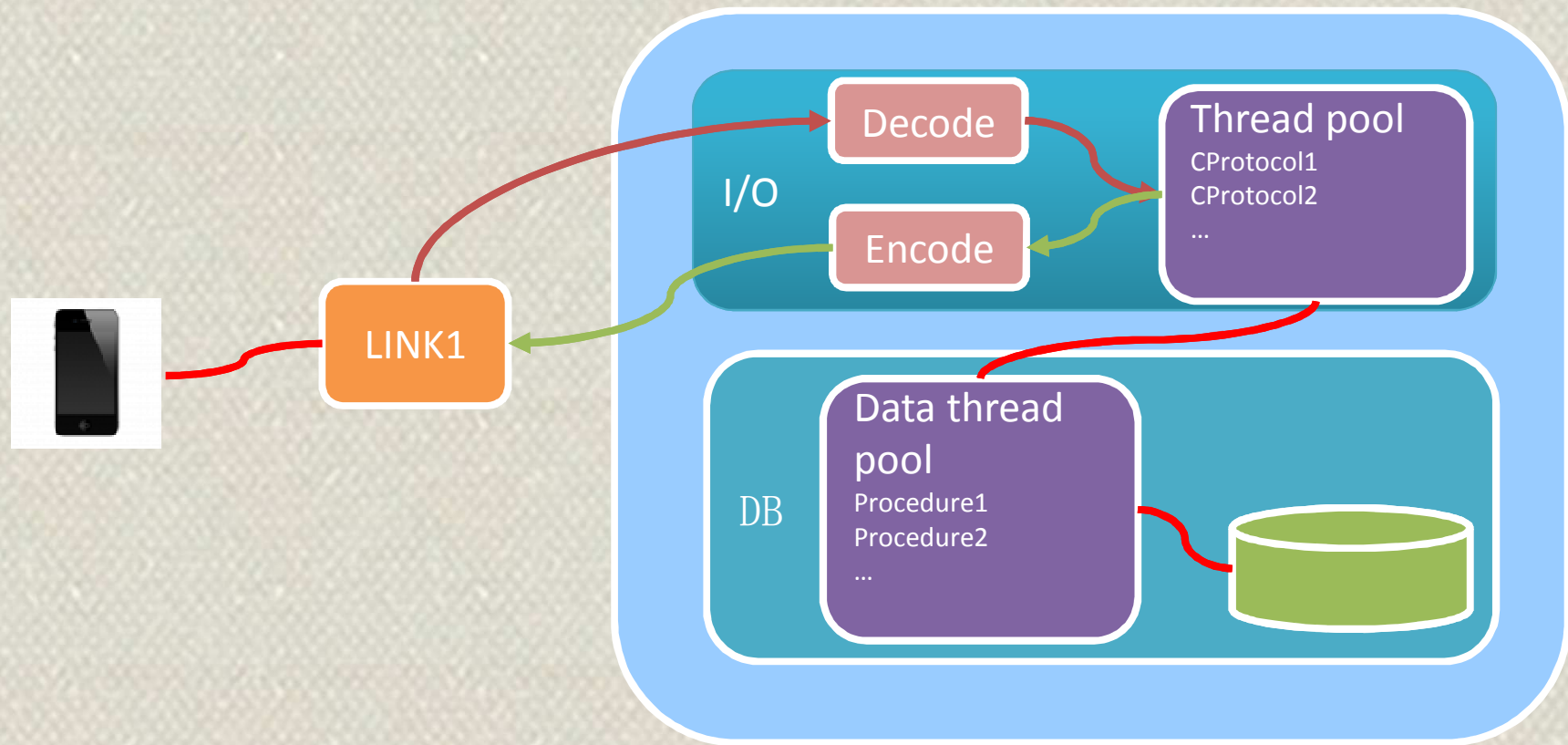
- 文档描述
- 结构图

# Architecture Overview





# Game Server Architecture



# 目录

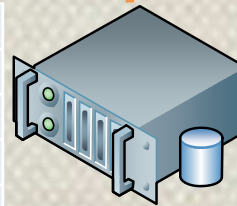
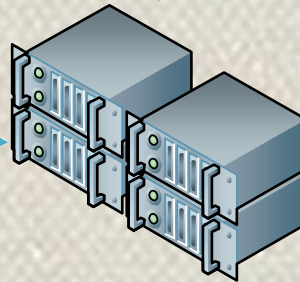
- § 7.1 设计的层次
- § 7.2 大型软件项目的架构
- § 7.3 一些实现方式



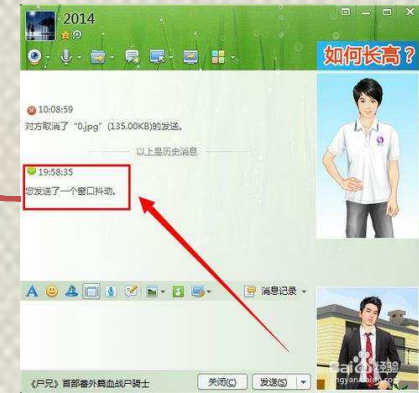
# 大型软件的架构

- 单机架构(Standalone)
- C/S架构(Client / Server)
- B/S架构(Browser/Server)
- Peer to Peer
- 分布式系统架构(Distributed System)
- 混合架构(Hybrid Architecture)

# C/S架构

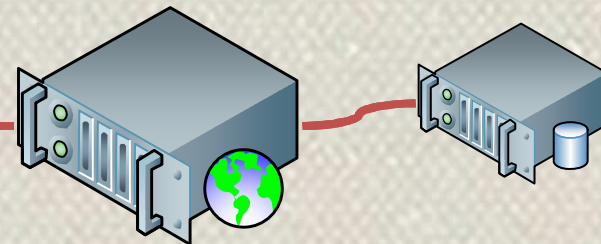
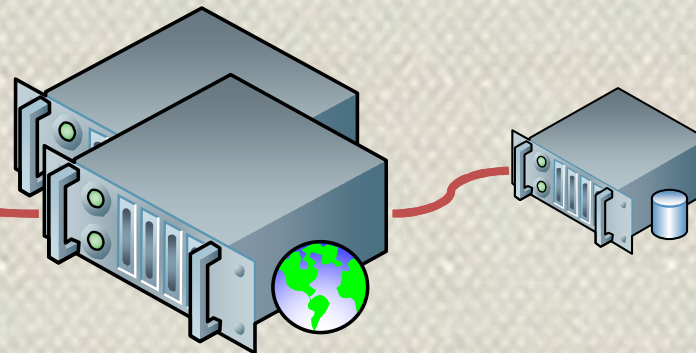


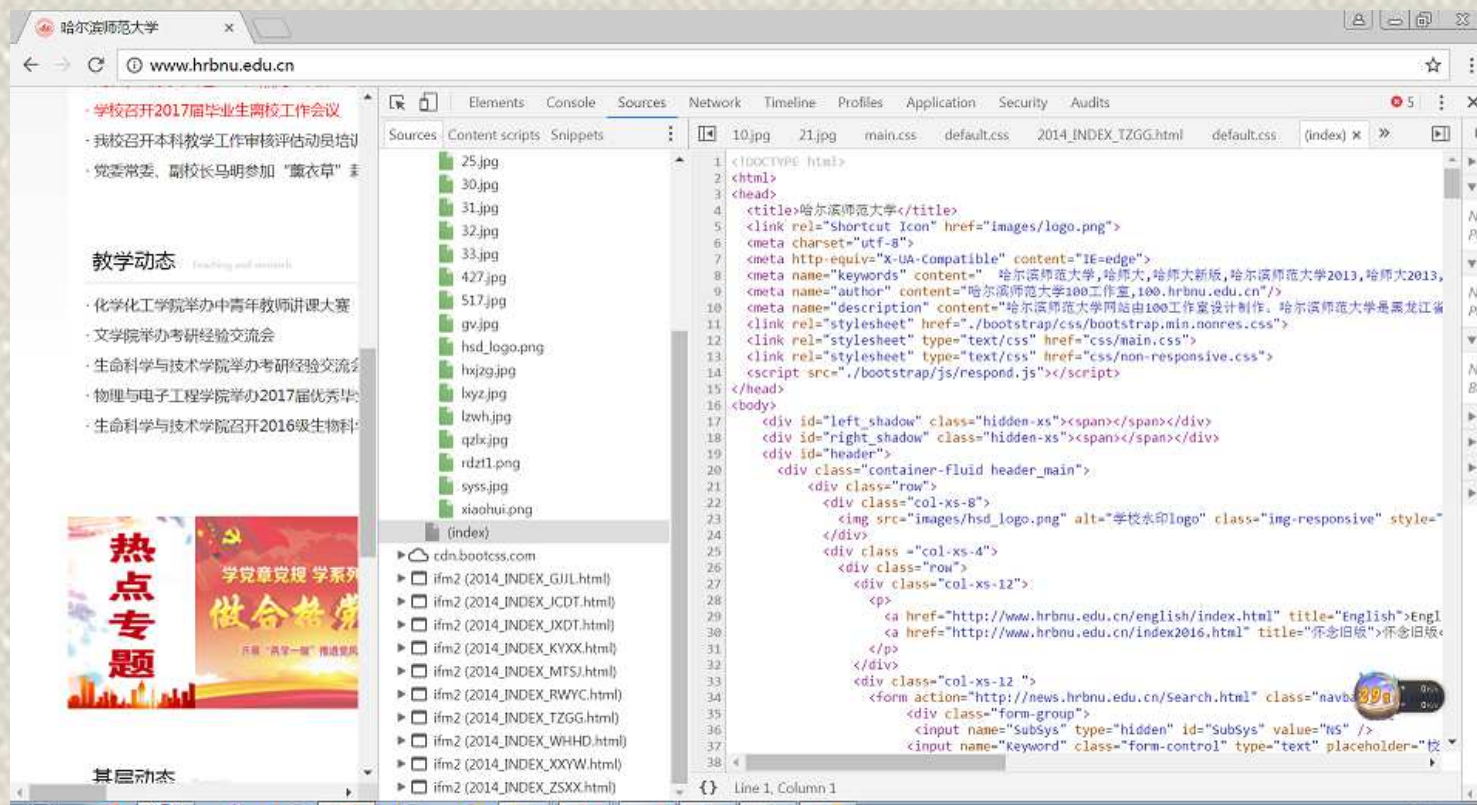
accid	password	name	signature	
1369234	#wulal896siw	快乐的程序员	之所以快乐, .....	
13695413	9Tfit8^bfatIO	小鼠标	人生如戏, 戏如人生	
13695414	5ajfit8bfrad	哈哈镜	照鬼不照人	
13695415	gwjit45eblgoa	蜜桃	你胖你先吃	
13695416	weafadbffwa	开心鬼 ^_^		
13695417	56esdt8OTe	秀个够够		





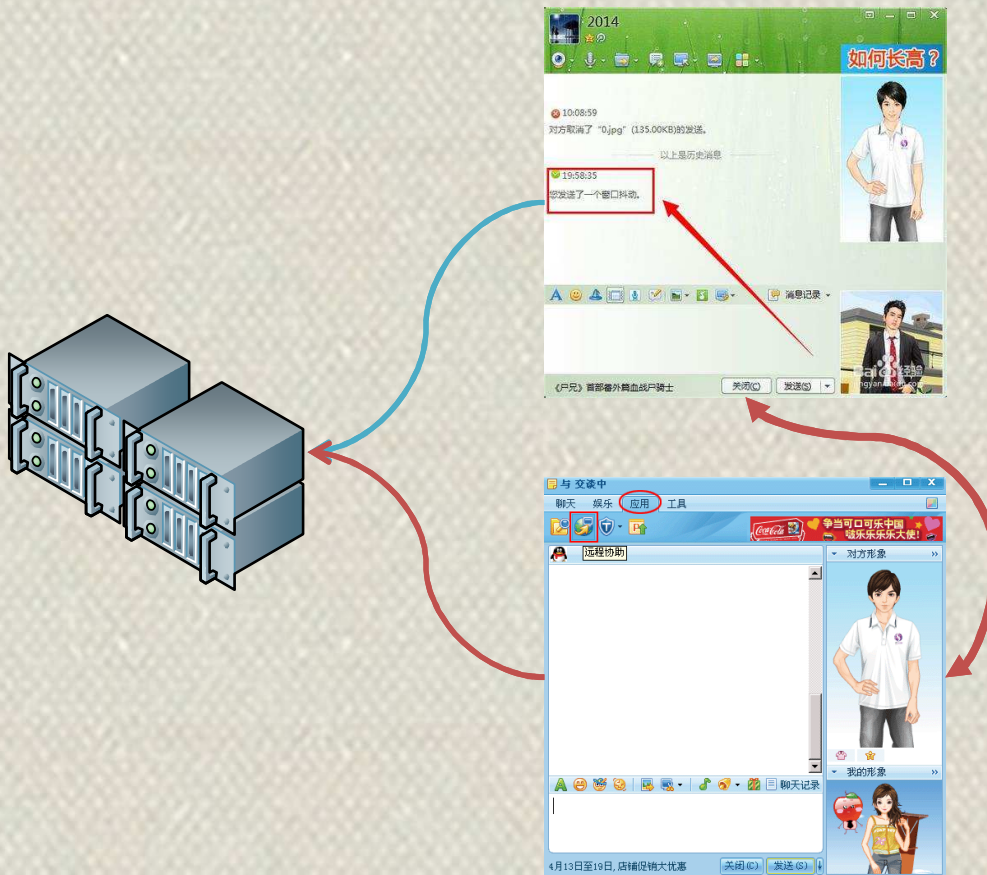
# B/S架构







# P2P



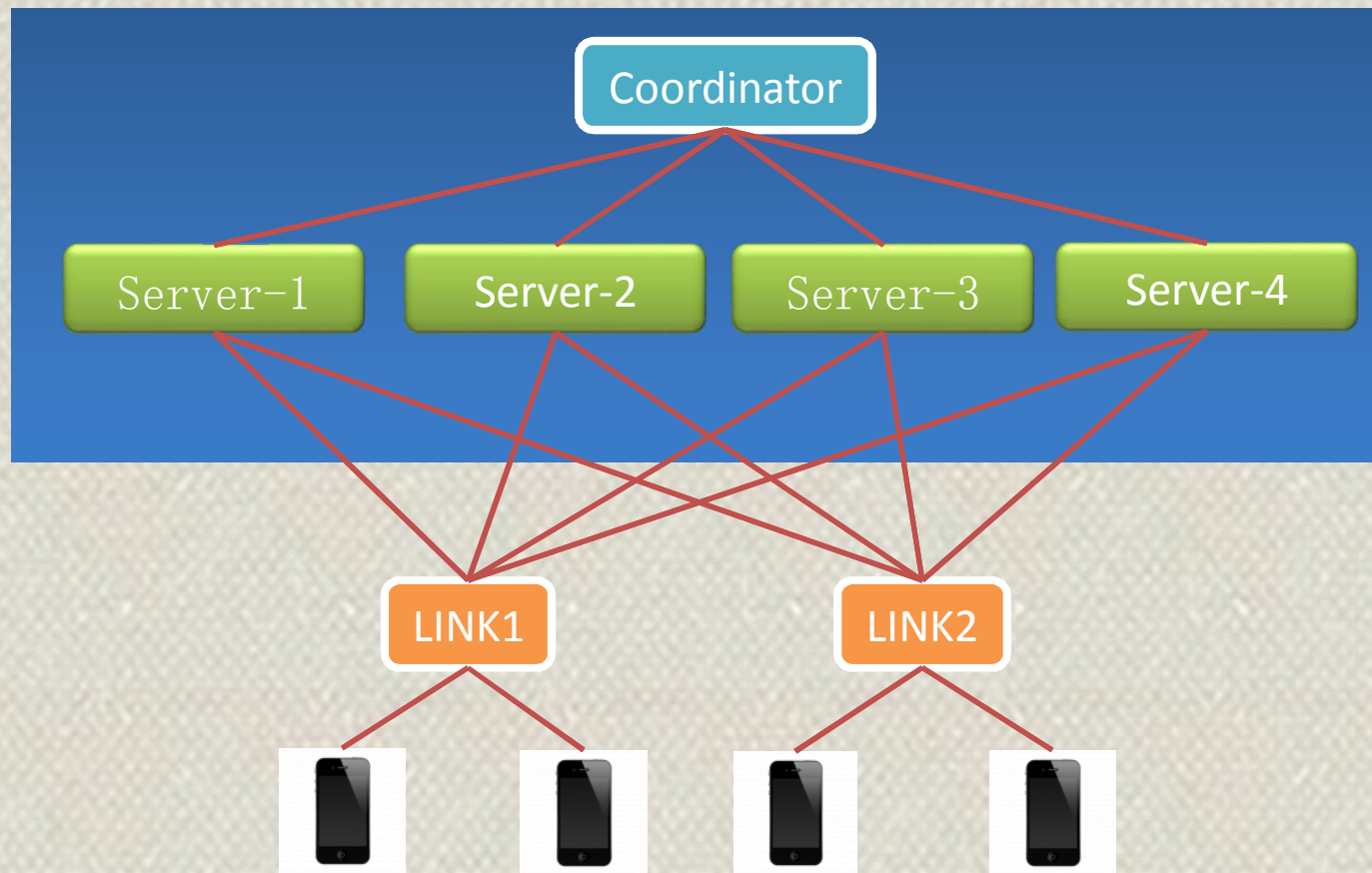
- 甲请求登录，S记录
- 乙请求登录，S记录
- 甲请求与乙交谈
- S把乙的信息发给甲
- 甲和乙建立连接
- 开始交谈

# P2P的问题

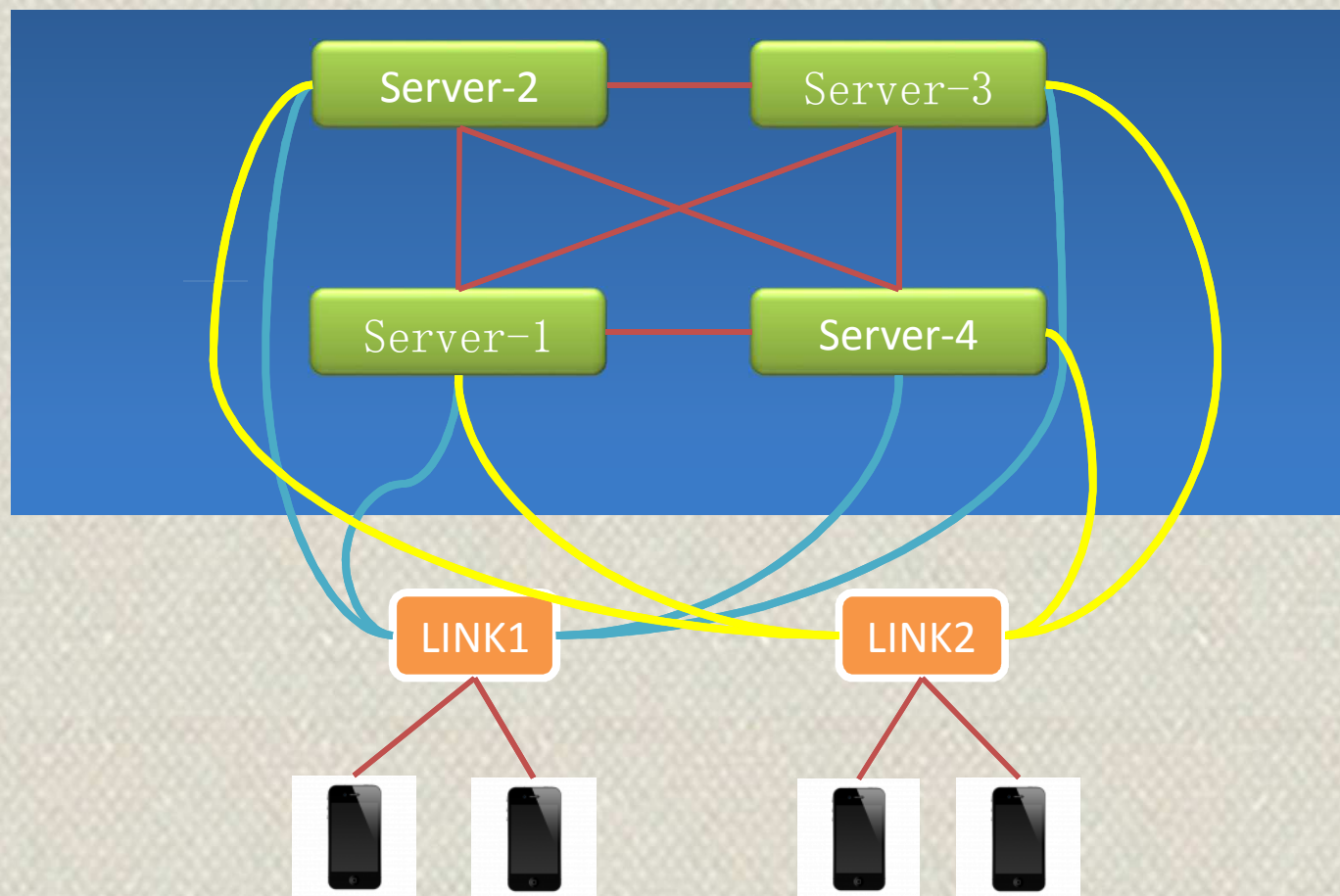
- 优点很明显
- 有什么局限？



# 分布式系统

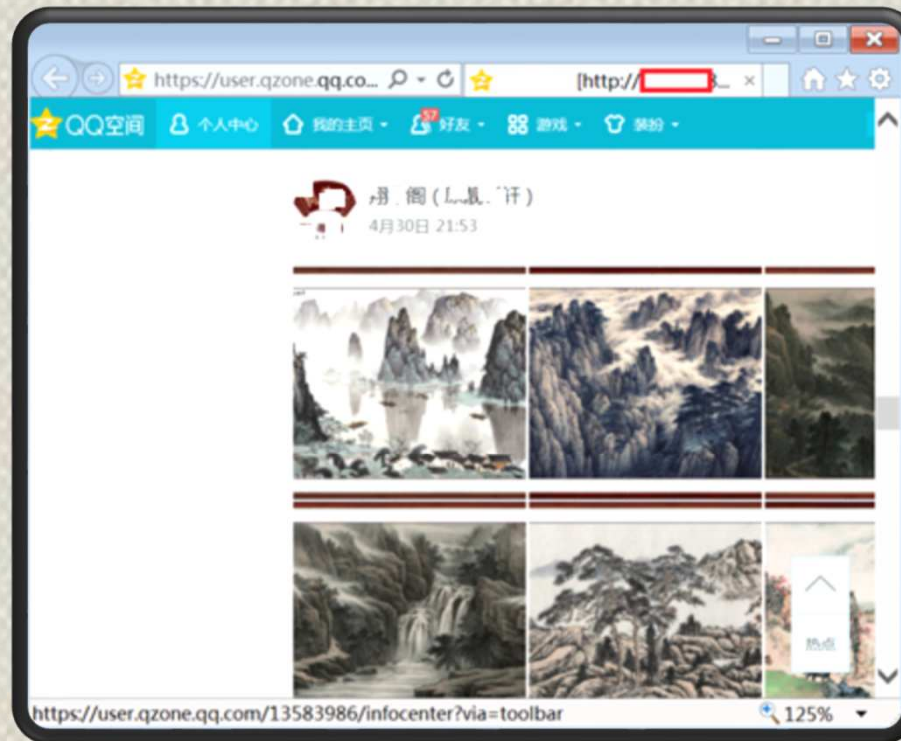


# 分布式系统





# 混合架构



# 目录

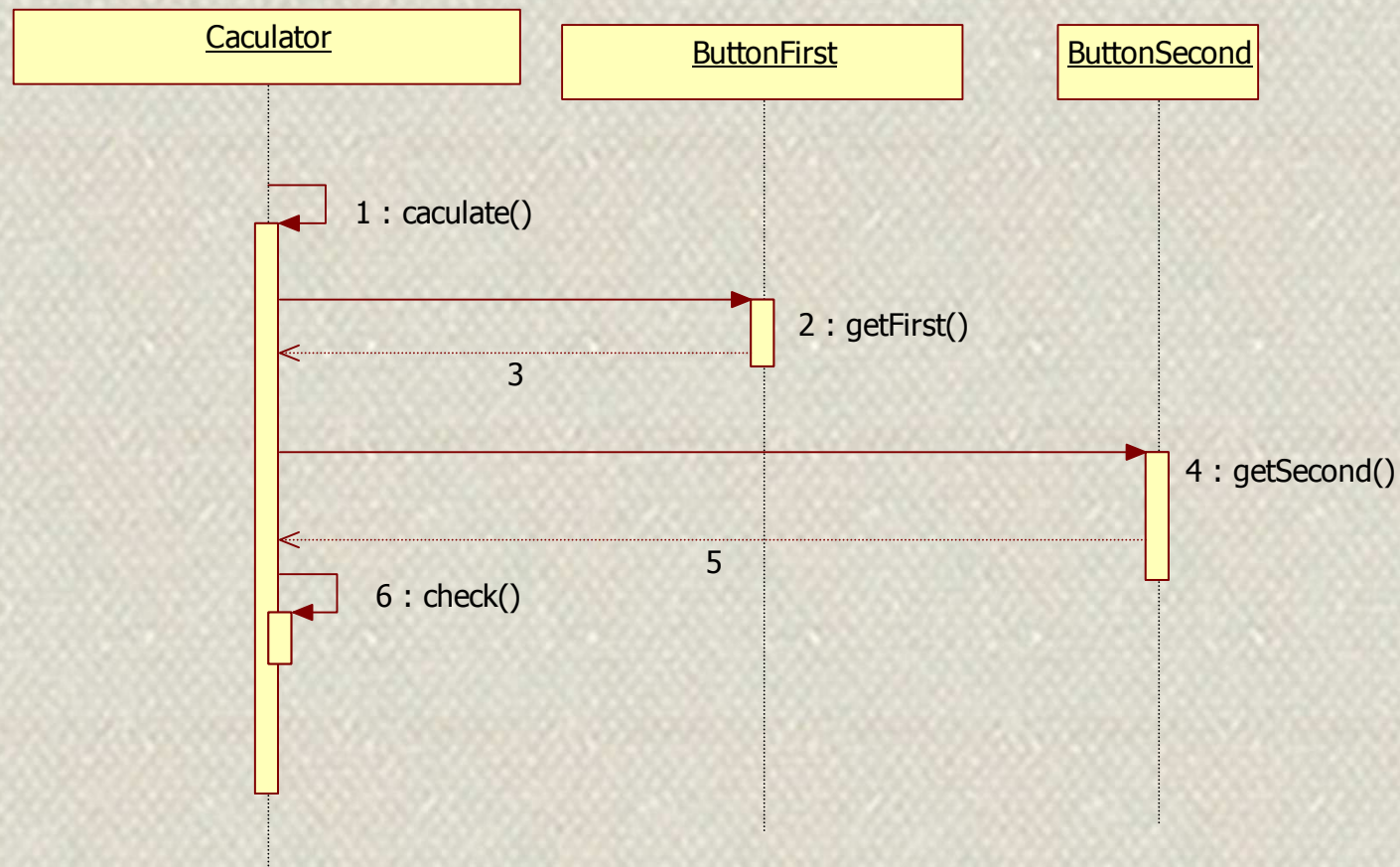
- § 7.1 设计的层次
- § 7.2 大型软件项目的架构
- § 7.3 一些实现方式



# 与最终写代码的距离还有多远

- 直接写代码
- 移植代码
- 流程图
- 交互图

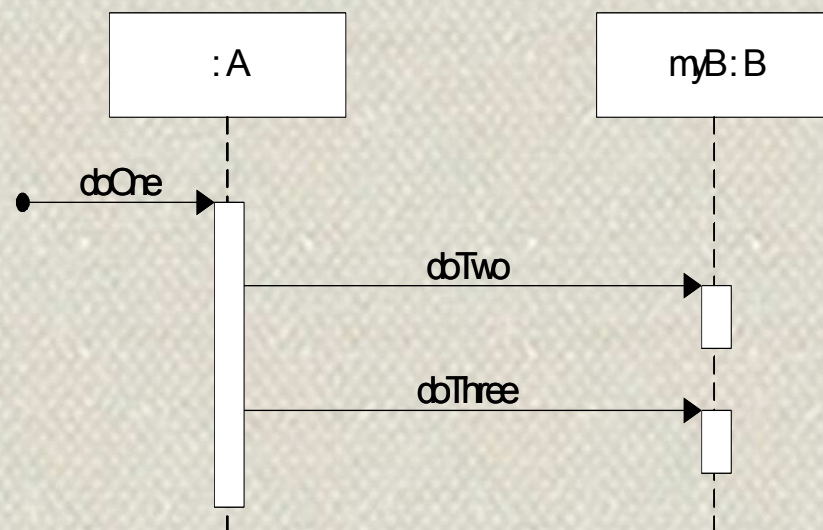
# 时序图





# 时序图

- 显示用例的行为顺序
- 显示了用例流程中不同对象之间的调用关系
- 对象、类和参与者都在顺序图中进行描述

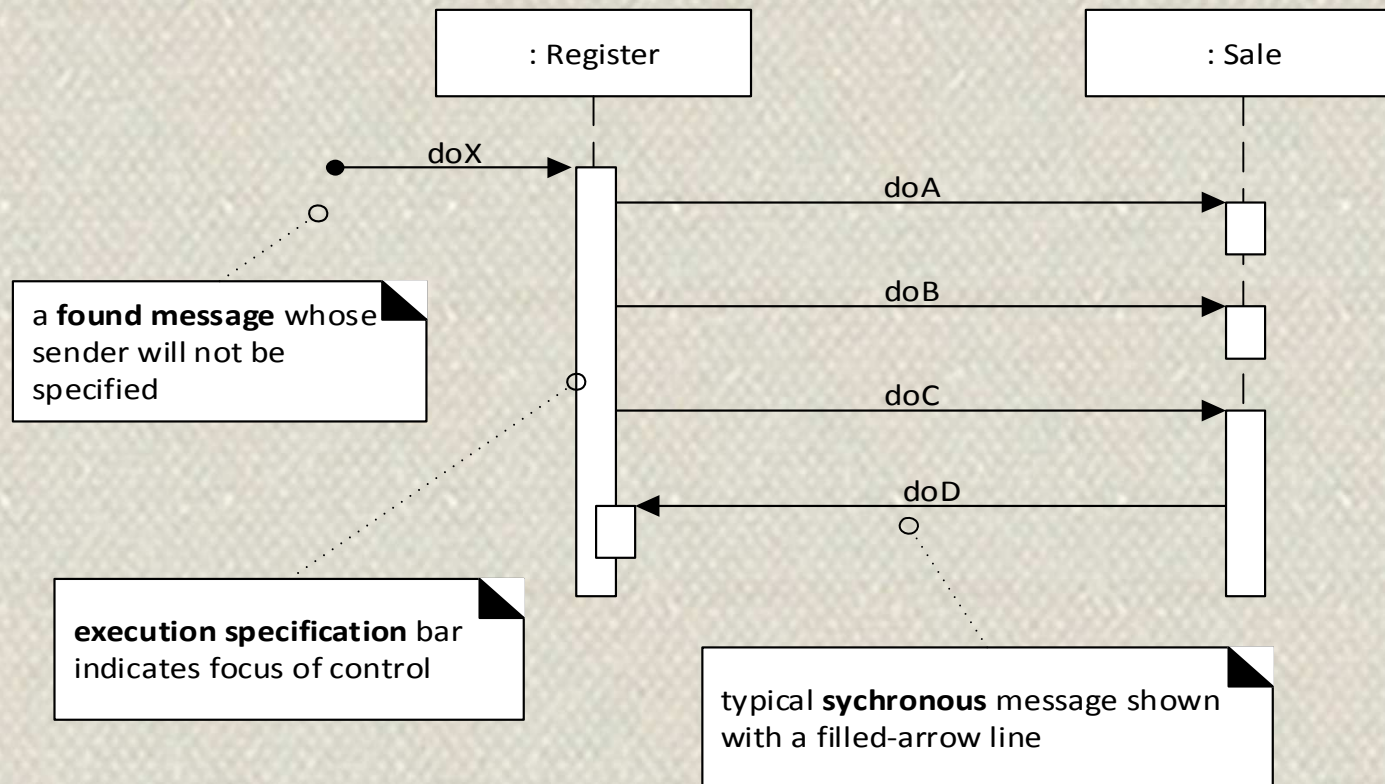


```
1  public class A
2  {
3      private B myB = new B();
4      public void doOne()
5      {
6          myB.doTwo();
7          myB.doThree();
8      }
9  }
```

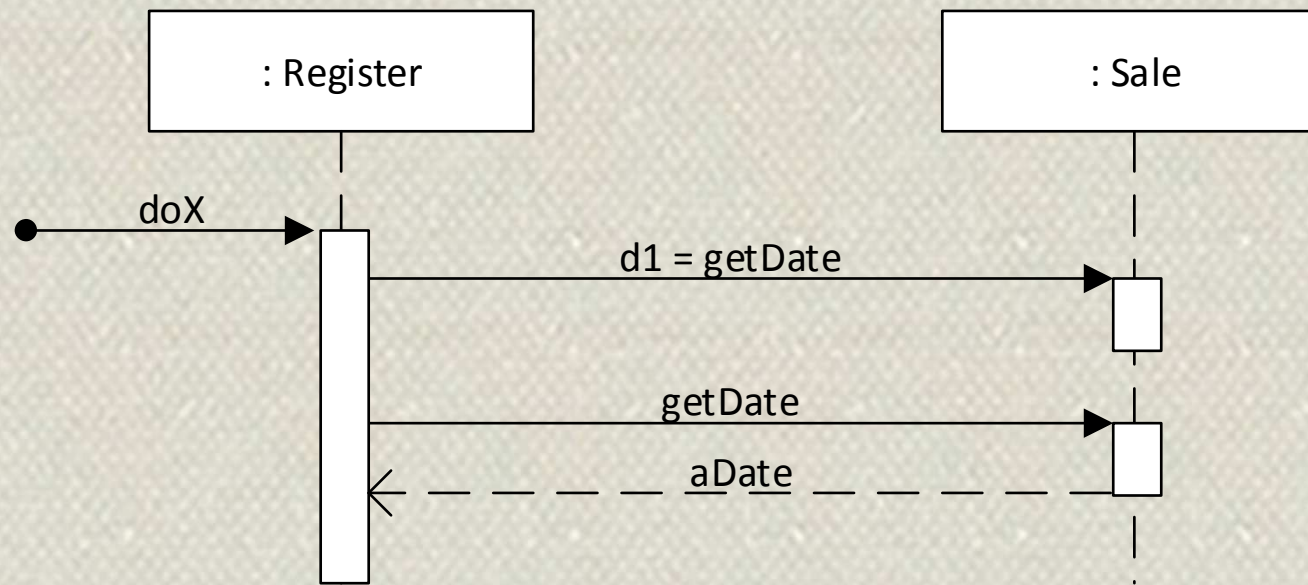


# 顺序图基本表示法

- 生命线框、消息、控制期和执行活动条



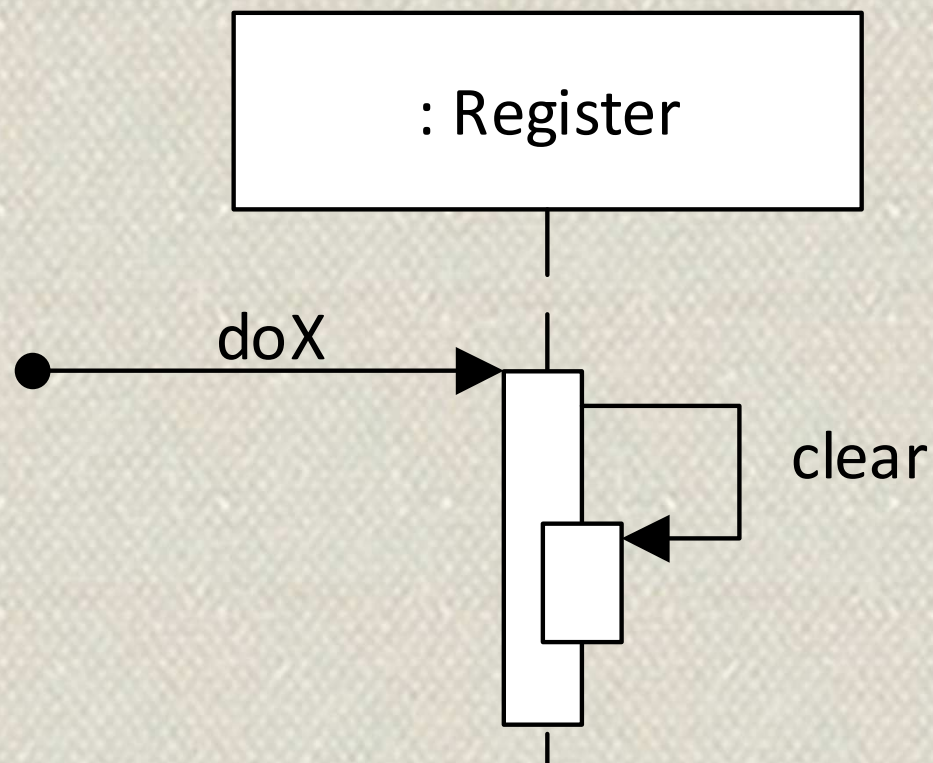
# 消息返回或应答表示





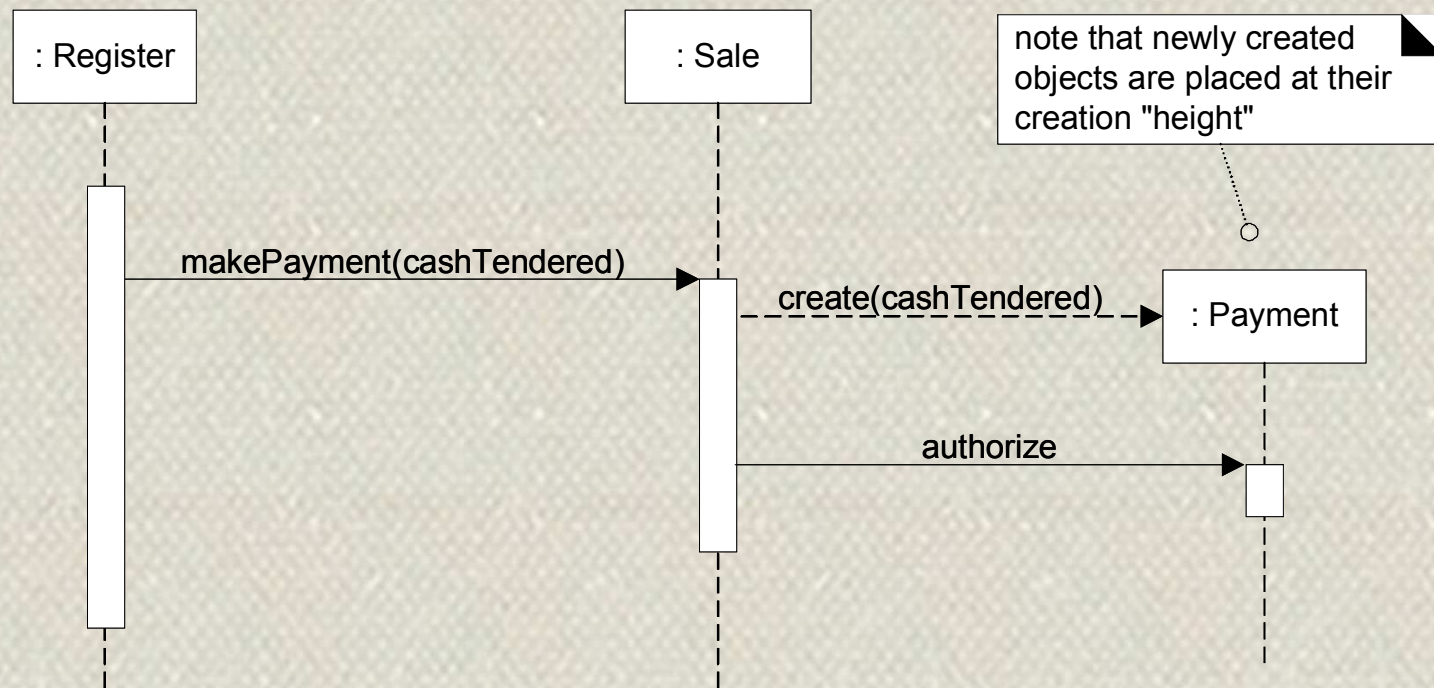
# 发送给“自身”的消息

- 使用嵌套的活动条表示对象发送给自己的消息



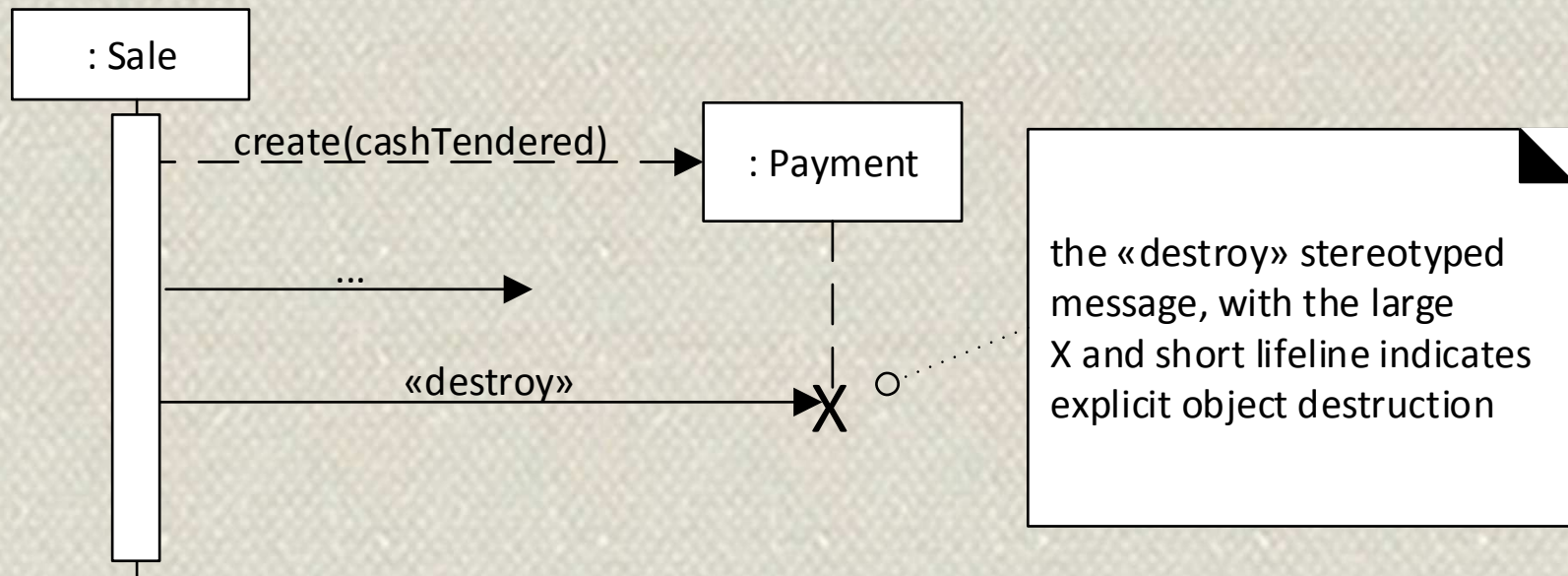
# 创建

- Create消息：创建对象或是new

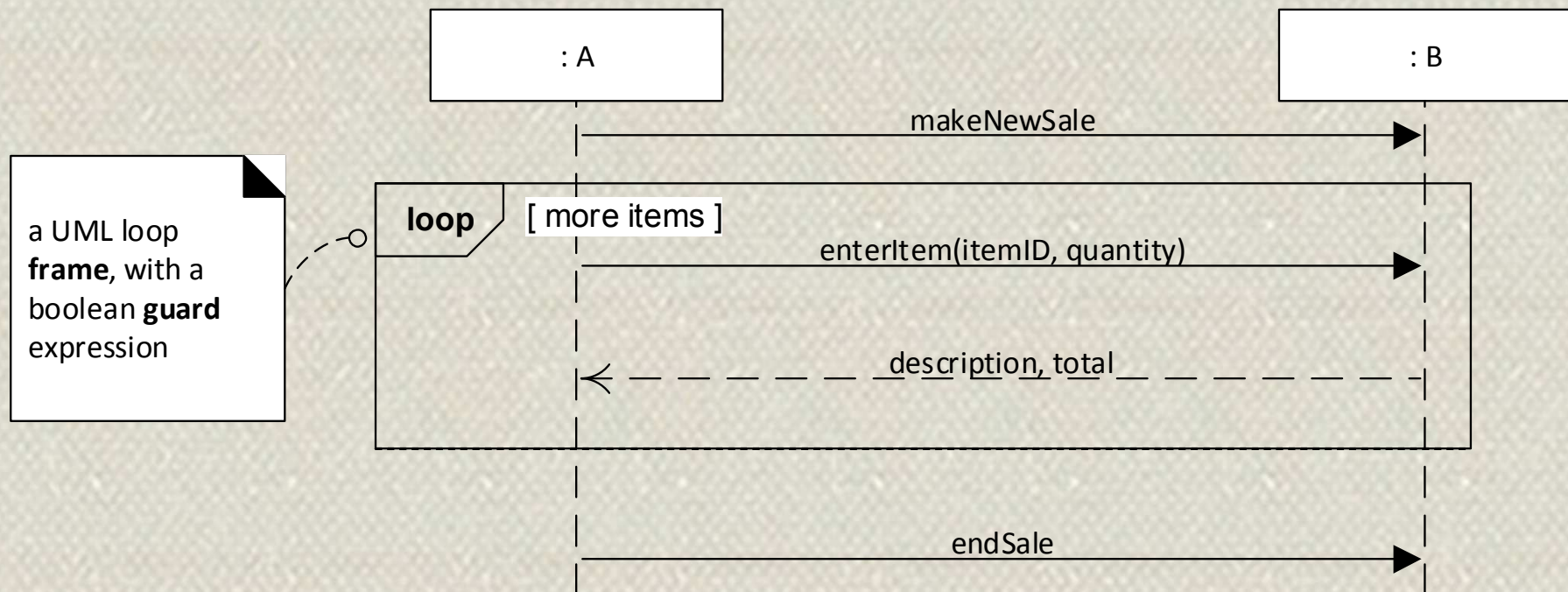




# 对象的销毁

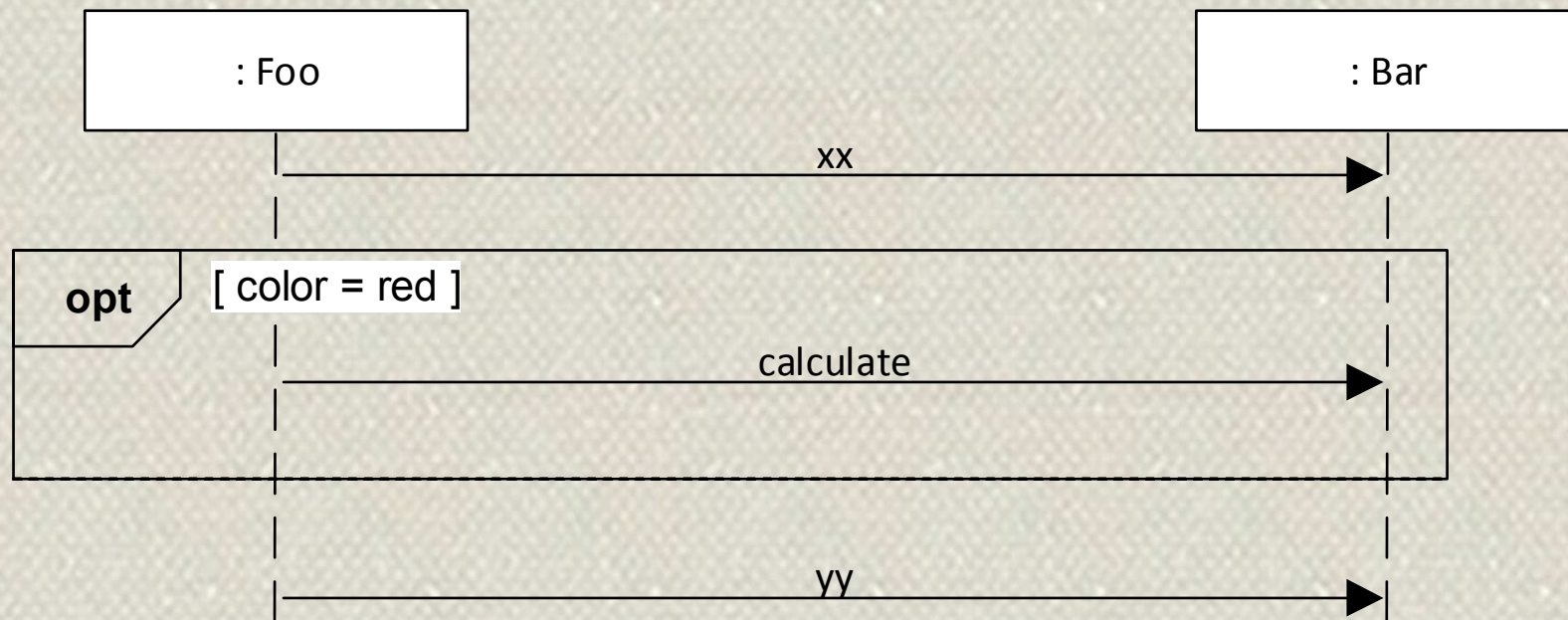


# 帧 (frame)





# 条件帧



# 互斥帧

