1、 分布式系统中的体系结构样式有那几种？并简述之。

1)分层体系结构：组件组成了不同的层，其中 $\mathbf{L_i}$ 层的组件可以调用下面的层 $\mathbf{L_{i-1}}$. 其中的关键因素是，其控制系统是从一层到别一层的：请求是从上往下，而请求结果是从下向上。

2）基于对象的体系结构：是一种很松散的组织结构。基本上每一个对象都对应一个组件，这些组件是通过过程调用机制来连接的。分层和面向对象的体系结构仍然是大型软件系统最重要的样式。

3)以数据为中心的体系结构，其发展思想：进程通信需要通过一个公用（被动或主动的）仓库。可以分成两个关键的部分:展现当前状态的中央数据结构和一个对数据进行操作的独立组件的集合

4）基于事件的体系结构，进程基本上是通过事件的传播来通信的，事件传播还可以有选择地携带数据。基本思想是：进程发布事件，然后，中间件将确保那些订阅了这些事件的进程才接收它们，优点是进程是松散耦合的。

2、 简述分布式系统设计所面临的问题及遇到的挑战。

1)：难以合理设计分配策略,在集中式系统中，所有的资源都由系统系统管理和分配，但在分布式系统中，资源属于局部工作站或个人计算机，所以调度的灵活性不如集中式系统，资源的物理分布可能与服务器的分布不匹配，某些资源可能空闲，而另外一些资源可能超载。

2)：部分失效问题：由于分布式系统通常是由若干部分组成的，各个部分由于各种各样的原因可能发生故障，如硬件故障。如果一个分布式系统不对这些故障对这些问题进行有效的处理，系统某个组成部分的故障可能导致整个系统的瘫痪。

3）性能和可靠性过分依赖于网络：由于分布式系统是建立在网络之上的，而网络本身是不可靠的，可能经常发生故障，网络故障可能导致整个系统的终止；另外，网络超负荷会导致性能下降，增加系统的响应时间。

4）缺乏统一控制：一个分布式系统的控制通常是一个典型的分散式控制，没有统一的中心控制。因此，分布式系统通常需要相应的同步机制来协调系统中各个部分的工作；

5）安全保密性问题：为了获得可扩展性，分布式系统中的许多软件接口都提供给用户，这样的开放结构对于开发人员非常有价值，担同时也为破坏者打开了方便之门

挑战：设计与实现一个对用户来说是透明的且具有容错能力的分布式系统。

挑战：1. Heterogeneity 异构性，包括网络、硬件、操作系统、编程语言、Implementation 组件，<中间件的作用就是隐藏这些异构，并提供一致的计算模式（模块）>，<虚拟机，编译器成虚拟机使用的 code>；2. Openness 开放性：提供 Services, Syntax, Semantics，合适的接口定义需要兼顾完整性和中立性，同时互操作性和便携性也是重要的，分布式系统需要 flexible，即易于配置且把策略和结构分开来获得 flexible；3. Security 安全性：安全性包括有效性，机密性，完整性三个要素，拒绝服务攻击和移动代码安全是两个安全方面的挑战。4. Scalability 可扩展性（可测性），即在资源和用户增加的时候保持效率，size、geographically、administratively 的 scaleble。要求分散（分布式）算法，没有机器有系统状态的完整信息，机器做决定仅仅是取决于本地信息，一个机器的错误不会毁掉整个算法，没有统一时序。5. Failure Handling 错误处理。6. Concurrency 协力，一致（并发？） 7. Transparency 透明度

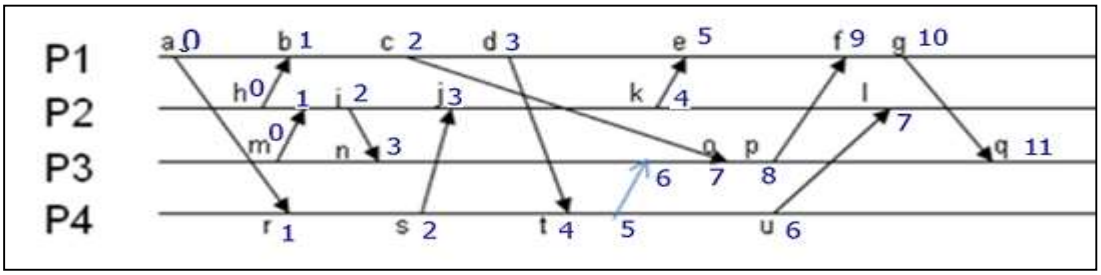
3、 **简述远程过程调用的步骤。**

Client 端:1)发送远程过程调用的消息(以消息包形式)给远程的 server 端;2) 等待, 直到收到 server 端对该请求的回复;3) 一旦接收到来自 server 端的返回执行结果, 就继续执行后面的程序.Server 端:1) 倾听状态, 等待 client 端发送过程调用消息;2)一旦接收到过程调用消息, server 就抽取参数并分析它, 然后执行所请求的过程;3)将执行结果以消息包形式回送给

client.
1、客户进程以正常的方式调用客户存根；
2、客户存根生成一个消息，然后调用 本地操作系统；
3、客户端操作系统将消息发送给远程操作系统；
4、远程操作系统将消息交给服务器存根；
5、服务器存根将参数提取出来，然后调用服务器；
6、服务器执行要求的操作，操作完成后将结果返回给服务器存根；
7、服务器存根将结果打包成一个消息，然后调用本地操作系统；
8、服务器操作系统将含有结果的消息发送回客户端操作系统；
9、客户端操作系统将消息交给客户存根；
10、客户存根将结果从消息中提取出来，返回给调用它的客户过程。

4、 Consider **Figure 1** that shows four processes ***(P1, P2, P3, P4)*** with events ***a, b, c,*** … and messages communicating between them. Assume that initial logical clock values are all initialized to ***0***.

a) List the **Lamport timestamps** for each event shown in Figure 1. Assume that each process maintains a logical clock as a single integer value as a Lamport clock. Provide Lamport clock for all labeled events (but consider all labeled and unlabeled events).



b) List the **Total Ordering Logical Clock timestamps** for each event shown in Figure 1. . Provide Total Ordering Logical Clock for all labeled events (but consider all labeled and unlabeled events).

c) Is there the potential for a causal violation? Explain why.

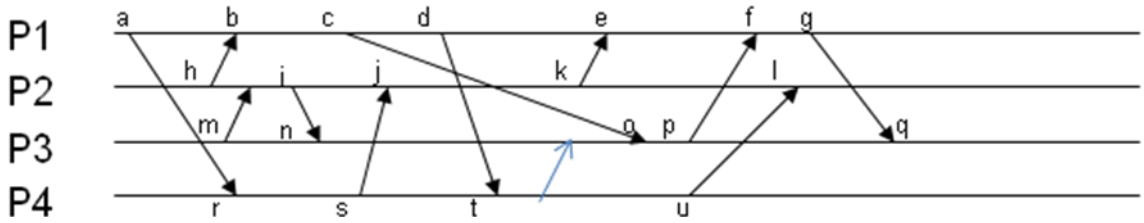


Figure 1: Four *Processes P1, P2, P3, P4 run events a,b,c,d,…. to send and receive messages*

5、 a, b, and c are events and no two events belong to the same process. Prove or disprove (give counter-example) the following:

a. a is concurrent with b and b is before c implies that a is before c.

b. a is concurrent with b and b is concurrent with c implies that a is concurrent with c.

6、 Consider a group of distributed systems, P1, P2, P3, P4 and P5 (P5 is currently the coordinator). P5 fails and P2 notices the failure. If the Bully algorithm is used for election of a new coordinator and the election attribute is the (Max of) processor numbers, show the set of all messages communicated through each communication channel Pij i,j = 1..5 for this election. Show the type of each message as “election”, “response”, “coordinator”.(CH10)

7、 How Web Services work？

1 客户发现服务器

2 客户通过建立 TCP 连接来绑定服务器

3 客户建立 SOAP 请求

4 SOAP 路由器路由请求给合适的服务器

5 服务器把请求解包,然后处理,并返回结果

6 结果通过相反的路径返回到客户端

8、 简述 CDN？CDN 的基本类型有哪几种，并简述之？

9、 The Byzantine generals problem for 3 loyal generals and 1 traitor.

a) what they send to each other？
b) what each one got from the other？
c) what each one got in second step？

10、 Consider Figure 2.(C9)

a) Where does the waiting/blocking occur?
b) What happens in case of a crash? How do we detect a crash?

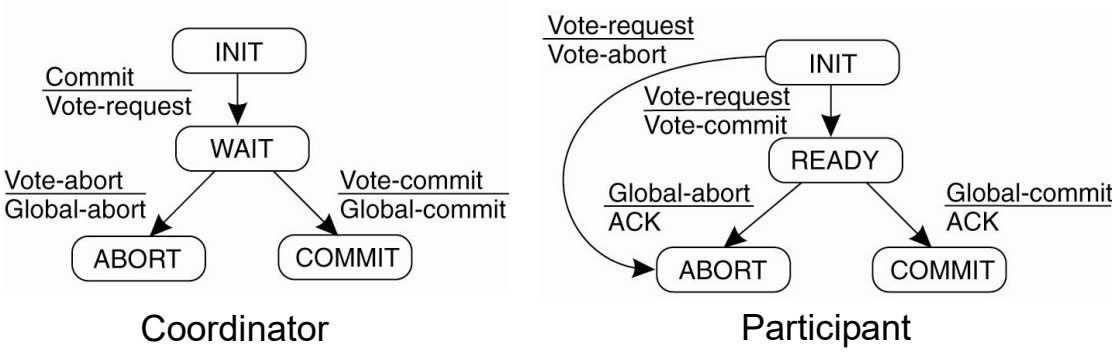


Figure 2: Coordinator and Participant

11、 以 google 文件系统为例，说明分布式文件系统的设计与应用场景密切相关。

12、 At 10:27:540 (hr, min, sec.), Processor B requests time from a time server. At 10:27:610 it receives a reply with time stamp of 10:27:375.

a) Find out the dirft of B's clock with respect to the server 's clock (assume there is no processing time at the server for time service).
b) Is B's clock going too fast or too slow?
c) How should B adjust its clock?

1.B 的发送时间应该是 10：27：140 所以该时钟早了 ：400SEC

2.TOO FAST

3.将时钟的的 SEC 减去 200 就可以了

13、 Compile a summary for the course

a) It should be a summary summarizing what we have gone through for the semester.
b) This summary will serve as a quick review on distributed systems and a prefect cue card/note before a job interview that is related to distributed computing.
c) I will look at the organization of the summary as well as some intelligent ways for reminding what you have learnt in the course.

14、 "End-To-End Arguments"主要观点是什么？

主要是关于可靠性的讨论。网上定义为：一种应用功能只有当其知识和帮助置于通信系统的边缘才能完全和正确地实现，因此将提出这种应用功能作为通信系统本身的性质是不可能的。当时提出这种论断的依据是网络是不可靠的，最终检查是否正确执行只能在处于传输终端的应用层。让网络核心部分只做最通用的数据传输而不实现特殊应用有不少优点：如降低核心网络复杂性，便于升级；提高网络通用性和灵活性，增加新应用不必改变核心网络；提高可靠性等等。但是 20 年后网络应用环境已大大改变。由于用户急剧增加而互不了解，Internet 已变成没有信用的世界，必须在网络的核心部分增加认证、授权等机制使网络更可信。尽力服务不能保证服务质量，特别是流媒体服务质量，需要在网络中间增加存储节点。

15、 What problems does naming solve?

1)、增加实体的可读性

2)、隐藏复杂性和动态变化：

a)多个低层实体可以可以使用一个名字

b)多个低层实体的改变可以隐藏。

3)允许通过多种方式发现一个对象，既一个对象可以有多个名字。

16、 Why is DNS iterative and not recursive?

17、 简述 Ring Algorithm 和 Bully Algorithm，并分析其复杂度？

Bully Algorithm：

Bully Algorithm：

当任何一个进程发现协作者不再响应时，它就发起一次选举。进程 P 按如下过程主持一次选举：

1）P 向所有编号比它大的进程发送一个 election 消息

2）如果无人响应，P 获胜并成为协作者；

3）如果有编号比它大的进程响应，则由响应者接管选举工作。P 的工作完成。
任何一个时候，一个进程只能从编号比它小的进程得到一个 election 消息。当有 election 消息到达时，接收者回送一个 OK 消息给发送者，表明它仍然在运行，并且接管选举工作。然后接收者主持一个选举，除非它正有一个选举。最终，除了一个进程外，其它所有进程都将放弃，那个进程就是新协作者。它将选举获胜的消息发送给所有进程，通知这些进程自己就是新的协作者。
当一个以前崩溃的进程现在恢复过来时，它将主持一次选举。如果该进程恰好是当前正在运行的进程中进程号最大的进程，它将赢得此次选举，接管协作者的工作。这样大的进程总是获胜。
Ring Algorithm：
当任何一个进程注意到协作者不工作时，它就构造一个带有它自己的进程号的 election 消息，并将该消息发送给它的后继者。如果后继者崩溃了，发送者沿者此环跳过它的后继者发送给下一个进程，或者再下一个，直到找到一个正运行的进程。在每一步中，发送者都将自己的进程号加到该消息中，以使自己成为协作者的候选人之一。
最终消息返回到发起此次选举的进程。当发起进程接收到一个包含它自己进程号的消息时，它识别出这个事件。此时，消息类型变为 coordinate 消息，并再一次绕环运行，向所有进程通知谁是协作者（成员列表中进程号最大的那个）以及新环中的成员都有谁。这个消息在循环一周后被删除，随后每个进程恢复原来的工作。
18、　简述分布式系统的典型实现？