

# 分布式系统

## What

### 1. 概念

1. 分布式软件系统 (Distributed Software Systems) , 是支持分布式处理的软件系统, 是在由通信网络互联的多处理机体系结构上执行任务的系统。它包括**分布式操作系统**, 分布式程序设计语言及其编译 (解释) 系统、**分布式文件系统**和分布式数据库系统等。
2. **分布式操作系统**负责管理分布式处理系统资源和控制分布式程序运行。他和集中式操作系统的区别在于资源管理、进程通信和系统结构等方面。
3. 分布式程序设计语言用于编写运行于分布式计算机系统上的分布式程序。一个分布式程序由若干个可以独立执行的程序模块组成, 它们分布于一个分布式处理系统的多台计算机上被同时执行。它与集中式的程序设计语言相比有三个特点: 分布性、通信性和稳健性。

分布式文件系统具有执行远程文件存取的能力, 并以透明方式对分布在网络上的文件进行管理和存取。

分布式数据库系统由分布于多个计算机结点上的若干个数据库系统组成, 它提供有效的存取手段来操纵这些结点上的子数据库。分布式数据库在使用上可视为一个完整的数据库, 而实际上它是分布在地理分散的各个结点上。当然, 分布在各个结点上的子数据库在逻辑上是相关的。

分布式数据库系统是由若干个站集合而成。这些站又称为节点, 它们在通讯网络中联接在一起, 每个节点都是一个独立的数据库系统, 它们都拥有各自的数据库、中央处理机、终端, 以及各自的局部数据库管理系统。因此分布式数据库系统可以看作是一系列集中式数据库系统的联合。它们在逻辑上属于同一系统, 但在物理结构上是分布式的。

### 4. 鸿蒙是分布式操作系统

5. NI LabVIEW 8 的分布式智能结合了相关的技术和工具, 解决了分布式系统开发会碰到的一些挑战。
6. 网络和分布式系统之间的区别更多的在于高层软件 (特别是操作系统) , 而不是硬件。

### 7. 优点: 并行, 容错性

### 8. 分布式操作系统是有着比较明确的定义:

- 工作在分布式的网络上, 由网络联通各个节点
- 是一套软件, 在各个节点上是这个软件的一个子集, 每个子集包含两部分, 一部分是控制本地资源的 (通常为一个操作系统内核或者微内核) , 另一部分是上边的一层分布式的控制管理机制。
- 分布式操作系统提供 API 给用户应用程序

### 9. hadoop

10. 在过去, 分布式操作系统的主要目标是整合各个节点的计算资源, 完成一件繁重的任务。

### 11. RPC: remote procedure call, 远程过程调用

- UNIX 的 corba, Windows 的 DCOM
- 它的实现需要具有一定的位置透明性
- 一个需要考虑的因素是 RPC 的时延, 远程调用通常比本地调用高几个数量级

### 12. 分布式调度: distributed scheduling

- **全局调度器**负责负载的整体协调, 并负责对用户应用提供接口
- 本地调度器负责分配到本地的计算任务的调度, 负责对本地资源的状态管理
- 全局状态控制单元负责维持一张表, 记录全局的任务分配图
- 全局调度器并不位于某单个机器, 而是在各个节点之间共享的一套软件资源

### 13. 分布式文件系统

- 分布式文件系统是分布在多个文件服务器或多个位置上的文件系统
- 它允许程序像访问本地文件一样访问或存储隔离的文件，从而使程序员可以从任何网络或计算机访问文件
- 位置透明性
- 冗余
- 存在的问题：不同机器上的文件重名问题，某个存储设备下线问题
- 操作的是 vfs，即虚拟文件系统。利用 RPC 的机制，把远端的文件挂载到我本地的 vfs 上，实现访问远端文件如同访问本地文件

### 14. 分布式数据库

- 分布式数据库是一组分布在计算机网络或者 Internet 上的互联数据库。分布式数据库管理系统 (DDBMS) 管理分布式数据库并提供机制，以使数据库对用户透明。在这些系统中，数据有意识地分布在多个节点之间，以便可以最佳地使用组织地所有计算资源。
- 依赖于通信

### 15. 分布式安全

- 需要额外加一些机制去保护
- 加密解密，安全通道 (SSL)，访问控制 (分布式 ID 注册)，安全管理 (接入限制)

### 16. 服务器分布式 与 终端分布式

项目	服务器分布器	终端分布式
计算资源	对等计算能力	差异巨大
计算架构	同架构	异构
网络情况	可靠网络	脆弱网络
用户界面	面向计算任务	面向用户操作
负载均衡	核心要务	能者多劳
共识机制	多种多样	中心节点

## 2. 鸿蒙

1. 归属于系统服务层，在鸿蒙中会提供几个系统进程，进程会注册一系列的系统服务，应用就可以调用这些系统服务去实现一些功能

2. 分成了：分布式软总线，分布式数据管理，分布式任务调度

### 3. 分布式软总线（没有开源）

- 自动发现，简化连接
- 总线中枢对设备网络进行管理，类似信令管理
- 任务盒数据总线是承载结构
- 协议货架为华为定制的
- 与海思系列芯片配合
- 任务&数据总线：文件传输，消息传输，流媒体传输，字节流传输

### 4. 分布式数据库

- 基于 KV 模型的分布式数据库：key-value，实现了在不同设备上的数据库，拥有统一的视角
- kvStore 是管理盒同步的对象单元
- 支持简单数据类型和复杂数据类型
- 提供统一的分布式数据库 API

### 5. 分布式文件系统

- 与基于 vfs 的分布式文件系统比较类似
- 支持兼容 POSIX 接口：Unix 和 Linux 定义的一套在应用和操作系统之间访问的函数的规范
- 位置透明，冗余

## 6. 分布式融合搜索

- 给定关键字后，无需指定要搜索的设备，这个融合搜索会帮助把所有相关的东西都搜出来

## 7. 分布式安全

- 正确的人在正确的设备上正确使用数据

# Why

1. 解决组织机构分散而数据需要相互联系的问题

## 暂时的一些想法

- 分布式操作系统和分布式文件系统这两个都可以做，都是和本门课程联系比较紧密的课题，总结来说，他们要实现的一个目的都是“透明”——调用远程系统调用就像本地一样，操作远程文件就像操作本地文件一样。虽然网络和通信原因，导致远程操作的延迟比本地高几个数量级，如何给他们提速，或许是一个可以深入探讨的方向。这其中涉及到的网络延迟，由于与本课程关联不大，可以暂时忽略。
- 鸿蒙系统是分布式操作系统，我们不一定要用它（因为到现在我也没能找到它的源码），但是它的设计中或许有值得我们学习的。比如果我们自己做一个简单的分布式系统，或者在一些小项目的基础上进行改进，鸿蒙的一些设计理念，我们或许可以参考。其中，部分设计理念，见本文档前面的“鸿蒙部分”。
- 看到 tjf 同学在他的第二次调研提出的疑问，如何实现分布式、实现哪一种分布式以及会不会偏离操作系统。说下我的一些看法，第三个问题，会不会偏离操作系统，分布式操作系统的大部分行为和和本地操作系统是一致的，只要不是思维过于飞跃，不会偏离本课程，而且还有老师帮忙把关。前两个问题，现在回答可能稍早一些，不过这两个问题也是我的问题，我对这两个问题的想法是，从我们组确定走分布式系统的方向，到今天，只有一周时间，与此同时，我们的操作系统正课（也就是从老师讲完了机器狗和操作系统历史之后算起），到现在也只有两周，现在感觉没个抓手，没有具体看得见摸得着的东西，是很正常的。