系统架构设计师考前20问

一、问：信息化需求的3个层次是什么？

答：（1）战略需求。组织信息化的目标是提升组织的竞争能力、为组织的可持续发展提供一个支持环境。从某种意义上来说，信息化对组织不仅仅是服务的手段和实现现有战略的辅助工具；信息化可以把组织战略提升到一个新的水平，为组织带来新的发展契机。特别是对于企业，信息化战略是企业竞争的基础。

（2）运作需求。组织信息化的运作需求是组织信息化需求非常重要且关键的一环，它包含三方面的内容：1.实现信息化战略目标的需要；2.运作策略的需要。3.人才培养的需要。

（3）技术需求。由于系统开发时间过长等问题在信息技术层面上对系统的完善、升级、集成和整合提出了需求。也有的组织，原来基本上没有大型的信息系统项目，有的也只是一些单机应用，这样的组织的信息化需求，一般是从头开发新的系统。

组织的三个层次的需求并不是相互孤立的，而是有着内在的联系。信息化需求的获取是一个自上而下的过程，需要对这些需求进行综合分析，才能把握组织对信息化建设的方向。

二、问：面向信息集成、面向过程集成、面向服务集成与界面集成、数据集成、应用集成、过程集成有什么区别和联系？

答：按照集成的层次进行分类的，所以将企业内的集成分为：界面集成、平台集成、数据集成、应用集成、过程集成等。

按照集成技术来分为：面向信息的集成、面向过程的集成和面向服务的集成。

对于二者的联系： 一般而言，界面集成、平台集成、数据集成、应用集成一般可通过面向信息的集成技术实现，过程集成一般可通过面向过程的集成技术可实现，但其中又常包含接口集成等面向信息的集成技术。当然，应用集成和过程集成也可以通过面向服务的集成技术来实现。

三、问：演化模型跟螺旋模型，增量模型，原型法开发的关系？

答：首先是原型化，原型化又可以分为两种，抛弃式原型和演化式原型，其中最终演化成一个产品的就演变成为演化模型。螺旋模型式是通过演化模型和瀑布模型的结合所产生的，并且螺旋模型强调了风险管理。增量模型是原型模型和瀑布模型所结合而产生的。其中螺旋模型和增量模型的区别：螺旋模型他是一直旋下去，旋到最后才是一个产品，而增量模型每一个增量都发布了一个可操作的产品，这是它的一个细微的区别。

四、问：传统五大类风格有哪些？是怎么分类的？

答：

|  |  |
| --- | --- |
| 五大架构风格 | 子风格 |
| 数据流风格 | 批处理、管道-过滤器 |
| 调用/返回风格 | 主程序/子程序、面向对象、层次结构 |
| 独立构件风格 | 进程通信、事件驱动系统（隐式调用） |
| 虚拟机风格 | 解释器、规则系统 |
| 仓库风格 | 数据库系统、黑板系统、超文本系统 |

五、问：虚拟机架构风格解释器和规则系统之间的差异点在哪里？

答：他们都是虚拟机风格，两者都可以灵活应对自定义场景。

解释器风格是一个用来执行其它程序的程序。解释器可针对不同的硬件平台实现一个虚拟机，将高抽象层次的程序翻译为低抽象层次所能理解的指令，以消除在程序语言与硬件之间存在的语义差异。适用于需要“自定义规则”的场合。

基于规则的系统是以规则为中心的虚拟机系统。而规则系统是适合于专家系统和DSS的。

六、问：REST的五个原则是什么？

答：REST（Representational State Transfer，表述性状态转移）是一种只使用HTTP和XML进行基于Web通信的技术，可以降低开发的复杂性，提高系统的可伸缩性。

REST的5个原则：

（1）网络上的所有事物都被抽象为资源。

（2）每个资源对应一个唯一的资源标识。

（3）通过通用的连接件接口对资源进行操作。

（4）对资源的各种操作不会改变资源标识。

（5）所有的操作都是无状态的。

七、问：微服务的优势是什么？

答：微服务之所以能盛行，必然是有它独特优势的，微服务的优势如下：

（1）技术异构性：在微服务架构中，每个服务都是一个相对独立的个体，每个服务都可以选择适合于自身的技术来实现。如，要开发一个社交平台，此时，我们可能使用文档型数据库来存储帖子的内容，使用图数据来存储朋友圈的这些关系等，这样可以把每一块的性能都充分发挥出来。同时，在应用新技术时，微服务架构也提供了更好的试验场。因为对于单块的系统而言，采用一个新的语言、数据库或者框架都会对整个系统产生巨大的影响，这样导致我们想尝试新技术时，望而却步。但微服务不同，我们完全可以只在一个微服务中采用新技术，待技术使用熟练之后，再推广到其它服务。

（2）弹性：弹性主要讲的是系统中一部分出现故障会引起多大问题。在单块系统中，一个部分出现问题，可能导致整体系统的问题。而微服务架构中，每个服务可以内置可用性的解决方案与功能降级方案，所以比单块系统强。

（3）扩展：单块系统中，我们要做扩展，往往是整体进行扩展。而在微服务架构中，可以针对单个服务进行扩展。

（4）简化部署：在大型单块系统中，即使修改一行代码，也需要重新部署整个应用系统。这种部署的影响很大、风险很高，因此不敢轻易的重新部署。而微服务架构中，每个服务的部署都是独立的，这样就可以更快地对特定部分的代码进行部署。

（5）与组织结构相匹配：我们都知道，团队越大越难管理，同时团队越大也代表系统规模越大代码库越大，这样容易引起一系列的问题。且当团队是分布式的时候，问题更严重。微服务架构就能很好的解决这个问题，微服务架构可以将架构与组织结构相匹配，避免出现过大的代码库，从而获得理想的团队大小及生产力。服务的所有权也可以在团队之间迁移，从而避免异地团队的出现。

（6）可组合性：在微服务架构中，系统会开放很多接口供外部使用。当情况发生改变时，可以使用不同的方式构建应用，而整体化应用程序只能提供一个非常粗粒度的接口供外部使用。

（7）对可替代性的优化：在单块系统中如果删除系统中的上百行代码，也许不知道会发生什么，引起什么样的问题，因为单块系统中关联性很强。但在微服务架构中，我们可以在需要时轻易的重写服务，或者删除不再使用的服务。

八、问：微服务与SOA的对比

答：

|  |  |
| --- | --- |
| 微服务 | SOA |
| 能拆分的就拆分 | 是整体的，服务能放一起的都放一起 |
| 纵向业务划分 | 是水平分多层 |
| 由单一组织负责 | 按层级划分不同部门的组织负责 |
| 细粒度 | 粗粒度 |
| 两句话可以解释明白 | 几百字只相当于SOA的目录 |
| 独立的子公司 | 类似大公司里面划分了一些业务单元（BU） |
| 组件小 | 存在较复杂的组件 |
| 业务逻辑存在于每一个服务中 | 业务逻辑横跨多个业务领域 |
| 使用轻量级的通信方式，如HTTP | 企业服务产总线（ESB）充当了服务之间通信的角色 |

|  |  |
| --- | --- |
| 微服务架构实现 | SOA实现 |
| 团队级，自底向上开展实施 | 企业级，自顶向下开展实施 |
| 一个系统被拆分成多个服务，粒度细 | 服务由多个子系统组成，粒度大 |
| 无集中式总线，松散的服务架构 | 企业服务总线，集中式的服务架构 |
| 集成方式简单（HTTP/REST/JSON） | 集成方式复杂（ESB/WS/SOAP） |
| 服务能独立部署 | 单块架构系统，相互依赖，部署复杂 |

九、问：参与特定领域软件架构（Domain Specific Software Architecture，DSSA）的人员可以划分为哪四类角色？

答：（1）领域专家

领域专家可能包括该领域中系统的有经验的用户、从事该领域中系统的需求分析、设计、实现以及项目管理的有经验的软件工程师等。

领域专家的主要任务包括提供关于领域中系统的需求规约和实现的知识，帮助组织规范的、一致的领域字典，帮助选择样本系统作为领域工程的依据，复审领域模型、DSSA等领域工程产品，等等。

（2）领域分析人员

领域分析人员应由具有知识工程背景的有经验的系统分析员来担任。

领域分析人员的主要任务包括控制整个领域分析过程，进行知识获取，将获取的知识组织到领域模型中，根据现有系统、标准规范等验证领域模型的准确性和一致性，维护领域模型。

领域分析人员应熟悉软件重用和领域分析方法；熟悉进行知识获取和知识表示所需的技术、语言和工具；应具有一定的该领域的经验，以便于分析领域中的问题及与领域专家进行交互；应具有较高的进行抽象、关联和类比的能力；应具有较高的与他人交互和合作的能力。

（3）领域设计人员

领域设计人员应由有经验的软件设计人员来担任。

领域设计人员的主要任务包括控制整个软件设计过程，根据领域模型和现有的系统开发出DSSA，对DSSA的准确性和一致性进行验证，建立领域模型和DSSA之间的联系。

领域设计人员应熟悉软件重用和领域设计方法；熟悉软件设计方法；应有一定的该领域的经验，以便于分析领域中的问题及与领域专家进行交互。

（4）领域实现人员

领域实现人员应由有经验的程序设计人员来担任。

领域实现人员的主要任务包括根据领域模型和DSSA，或者从头开发可重用构件，或者利用再工程的技术从现有系统中提取可重用构件，对可重用构件进行验证，建立DSSA与可重用构件间的联系。

领域实现人员应熟悉软件重用、领域实现及软件再工程技术；熟悉程序设计；具有一定的该领域的经验。

十、问：常考的质量属性以及相关设计策略有哪些？

答：（1）性能

性能（performance）是指系统的响应能力，即要经过多长时间才能对某个事件做出响应，或者在某段时间内系统所能处理的事件的个数。例如：1.同时支持1000并发；2.响应时间小于1s；3.显示分辨率达到4K。

代表参数：响应时间、吞吐量 设计策略：优先级队列、资源调度

（2）可用性

可用性（availability）是系统能够正常运行的时间比例。经常用两次故障之间的时间长度或在出现故障时系统能够恢复正常的速度来表示。例如：1.主服务器故障，1分钟内切换至备用服务器；2.系统故障，1小时内修复；3.系统支持7╳24小时工作。

（可靠性与可用性意义相近，一般选择优先选择可用性。可靠性要求比可用性更高。可靠则必可用，而可用不一定可靠。可用性是可靠性的一个指标。可参照概念整理《第5章 系统可靠性分析与设计》理解。）

代表参数：故障间隔时间 设计策略：冗余、心跳线

（3）安全性

安全性（security）是指系统在向合法用户提供服务的同时能够阻止非授权用户使用的企图或拒绝服务的能力。安全性又可划分为机密性、完整性、不可否认性及可控性等特性。例如：1.可抵御SQL注入攻击；2.对计算机的操作都有完整记录；3.用户信息数据库授权必须保证99.9%可用。

设计策略：追踪审计

（4）可修改性

可修改性（modifiability）是指能够快速地以较高的性能价格比对系统进行变更的能力。通常以某些具体的变更为基准，通过考察这些变更的代价衡量可修改性。（可扩展性与之相近）例如：1.更改系统报表模块，必须在2人周内完成；2.对Web界面风格进行修改，修改必须在4人月内完成。

主要策略：信息隐藏（二义性：良好的封装能够做到信息隐藏，一般归于可修改性策略；信息隐藏也能够体现在安全性当中）

十一、问：产品线的建立方式有哪几种？

答：有四种，如下：

（1）将现有产品演化为产品线

（2）用软件产品线替代现有产品集

（3）全新软件产品线的演化

（4）全新软件产品线的开发

|  |  |  |
| --- | --- | --- |
|  | 演化方式 | 革命方式 |
| 基于现有产品 | 基于现有产品架构设计产品线的架构，经演化现有构件，开发产品线构件 | 核心资源的开发基于现有产品集的需求和可预测的、将来需求的超集 |
| 全新产品线 | 产品线核心资源随产品新成员的需求而演化 | 开发满足所有预期产品线成员的需求的核心资源 |

十二、问：成功实施产品线主要决定因素有哪些？

答：（1）对该领域具备长期和深厚的经验

（2）一个用于构建产品的好的核心资源库

（3）好的产品线架构

（4）好的管理（软件资源、人员组织、过程）支持

十三、问：被动攻击和主动攻击分别有哪些？

答：被动攻击：收集信息为主，破坏保密性。

|  |  |  |
| --- | --- | --- |
| 攻击类型 | 攻击名称 | 描述 |
| 被动攻击 | 窃听（网络监听） | 用各种可能的合法或非法的手段窃取系统中的信息资源和敏感信息。 |
| 业务流分析 | 通过对系统进行长期监听，利用统计分析方法对诸如通信频度、通信的信息流向、通信总量的变化等参数进行研究，从而发现有价值的信息和规律。 |
| 非法登录 | 有些资料将这种方式归为被动攻击方式。 |

主动攻击：主动攻击的类别主要有：中断（破坏可用性），篡改（破坏完整性），伪造（破坏真实性）

|  |  |  |
| --- | --- | --- |
| 攻击类型 | 攻击名称 | 描述 |
| 主动攻击 | 假冒身份 | 通过欺骗通信系统（或用户）达到非法用户冒充成为合法用户，或者特权小的用户冒充成为特权大的用户的目的。黑客大多是采用假冒进行攻击。 |
| 抵赖 | 这是一种来自用户的攻击，比如：否认自己曾经发布过的某条消息、伪造一份对方来信等。 |
| 旁路控制  【旁路攻击】 | 密码学中是指绕过对加密算法的繁琐分析，利用密码算法的硬件实现的运算中泄露的信息。如执行时间、功耗、电磁辐射等，结合统计理论快速的破解密码系统。 |
| 重放攻击 | 所截获的某次合法的通信数据拷贝，出于非法的目的而被重新发送。  加时间戳能识别并应对重放攻击。 |
| 拒绝服务（DOS） | 对信息或其它资源的合法访问被无条件地阻止。 |

十四、问：恢复块方法与N版本程序设计有什么区别？

答：

|  |  |  |
| --- | --- | --- |
|  | 恢复块方法 | N版本程序设计 |
| 硬件运行环境 | 单机 | 多机 |
| 错误检测方法 | 验证测试程序 | 表决 |
| 恢复策略 | 后向恢复 | 前向恢复 |
| 实时性 | 差 | 好 |

其中：

前向恢复：使当前的计算继续下去，把系统恢复成连贯的正确状态，弥补当前状态的不连贯情况。

后向恢复：系统恢复到前一个正确状态，继续执行。

十五、问：总时差和自由时差分别是什么，怎么进行计算？

答：总时差（松弛时间）：

在不延误总工期的前提下，该活动的机动时间。活动的总时差等于该活动最迟完成时间与最早完成时间之差，或该活动最迟开始时间与最早开始时间之差。

自由时差：

在不影响紧后活动的最早开始时间前提下，该活动的机动时间。

对于有紧后活动的活动，其自由时差等于所有紧后活动最早开始时间减本活动最早完成时间所得之差的最小值。

对于没有紧后活动的活动，也就是以网络计划终点节点为完成节点的活动，其自由时差等于计划工期与本活动最早完成时间之差。

对于网络计划中以终点节点为完成节点的活动，其自由时差与总时差相等。此外，由于活动的自由时差是其总时差的构成部分，所以，当活动的总时差为零时，其自由时差必然为零，可不必进行专门计算。

十六、问：CISC（复杂指令集）和RISC（精简指令集） 的区别是什么？

答：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令系统类型 | 指令 | 寻址方式 | 实现方式 | 其它 |
| CISC（复杂） | 数量多，使用频率差别大，可变长格式 | 支持多种 | 微程序控制技术（微码） | 研制周期长 |
| RISC（精简） | 数量少，使用频率接近，定长格式，大部分为单周期指令，操作寄存器，只有Load/Store操作内存 | 支持方式少 | 增加了通用寄存器；硬布线逻辑控制为主；适合采用流水线 | 优化编译，有效支持高级语言 |

十七、问：板级支持包 （BSP）是什么？

答：（1）概念

板级支持包（BSP）是介于主板硬件和操作系统中驱动层程序之间的一层，一般认为它属于操作系统的一部分，主要是实现对操作系统的支持，为上层的驱动程序提供访问硬件设备寄存器的函数包，使之能与硬件主板更好地运行。在嵌入式系统软件的组成中，就有BSP。BSP是相对于操作系统而言的，不同的操作系统对应不同定义形式的BSP,例如VxWorks的BSP和Linux的BSP相对于某一CPU来说尽管实现的功能一样，可是写法和接口定义是完全不同的，所以写BSP一定要按照该系统BSP的定义形式来写（BSP的编程过程大多数是在某一个成型的BSP模板上进行修改）。这样才能与上层OS保持正确的接口，良好地支持上层OS。

（2）功能

BSP主要功能为屏蔽硬件，提供操作系统及硬件驱动，具体功能包括：

单板硬件初始化，主要是CPU的初始化，为整个软件系统提供底层硬件支持；

为操作系统提供设备驱动程序和系统中断服务程序；

定制操作系统的功能，为软件系统提供一个实时多任务的运行环境；

初始化操作系统，为操作系统的正常运行做好准备。

（3）BSP特点

板级支持包一般包含相关底层硬件的初始化、数据的输入／输出操作和硬件设备的配置等功能，它主要具有以下两个特点。

硬件相关性，因为嵌入式实时系统的硬件环境具有应用相关性，而作为上层软件与硬件平台之间的接口，BSP需为操作系统提供操作和控制具体硬件的方法。

操作系统相关性，不同的操作系统具有各自的软件层次结构，因此不同操作系统具有特定的硬件接口形式。

十八、问：封锁技术和封锁协议分别是什么？

答：（1）封锁技术

1.排他型封锁（简称X锁，也称为写锁）：如果事务T对数据A（可以是数据项、记录、数据集、乃至整个数据库）实现了X封锁，那么只允许事务T读取和修改数据A，其他事务不能对其加锁和操作。

2.共享型封锁（简称S锁，也称为读锁）：如果事务T对数据A实现了S封锁，那么只允许事务T读取数据A，不能修改数据A，其他事务可以读取数据A，并且可以对数据A加S锁，但不能加X锁。

（2）封锁协议

1.一级封锁协议。事务T在修改数据R之前必须先对其加X锁，直到事务结束才释放。可防止丢失修改。

2.二级封锁协议。一级封锁协议加上事务T在读取数据R之前先对其加S锁，读完后即可释放S锁。可防止丢失修改，还可防止读“脏”数据。

3.三级封锁协议。一级封锁协议加上事务T在读取数据R之前先对其加S锁，直到事务结束才释放。可防止丢失修改、防止读“脏”数据与防止数据重复读。

两段锁协议。可串行化的。可能发生死锁。

十九、问：冷备份和热备份的优缺点分别是什么？

答：

|  |  |  |
| --- | --- | --- |
| 优缺点  备份方式 | 优点 | 缺点 |
| 冷备份 | 非常快速的备份方法(只需复制文件)；容易归档(简单复制即可)；容易恢复到某个时间点上(只需将文件再复制回去)；能与归档方法相结合，做数据库“最佳状态”的恢复 ；低度维护，高度安全 | 单独使用时，只能提供到某一时间点上的恢复；在实施备份的全过程中，数据库必须要作备份而不能做其他工作；若磁盘空间有限，只能复制到磁带等其他外部存储设备上，速度会很慢；不能按表或按用户恢复 |
| 热备份 | 可在表空间或数据库文件级备份，备份的时间短；备份时数据库仍可使用；可达到秒级恢复(恢复到某一时间点上)；可对几乎所有数据库实体做恢复；恢复是快速的 | 不能出错，否则后果严重；若热备份不成功，所得结果不可用于时间点的恢复；因难于维护，所以要特别小心，不允许“以失败告终” |

（1）冷备份也称为静态备份，是将数据库正常关闭，在停止状态下，将数据库的文件全部备份（复制）下来。

（2）热备份也称为动态备份，是利用备份软件，在数据库正常运行的状态下，将数据库中的数据文件备份出来。

（3）完全备份：备份所有数据

（4）差量备份：仅备份上一次完全备份之后变化的数据

（5）增量备份：备份上一次备份之后变化的数据

（6）日志文件 ：事务日志是针对数据库改变所做的记录，它可以记录针对数据库的任何操作，并将记录结果保存在独立的文件中。

二十、问：分区和分表有什么区别和联系？分区有什么策略？

答：（1）分区与分表的区别与联系：

两者都针对数据表，将数据做到分布式，提高数据检索的效率，降低数据库的频繁I/O压力值。

分表是真正的生成数据表，是将一张大数据量的表分成多个小表实现数据均衡。

分区并不是生成新的数据表，而是将表的数据均衡分摊到不同的硬盘，系统或是不同服务器存储介质中，实际上还是一张表。

（2）分区的优点：

相对于单个文件系统或是硬盘，分区可以存储更多的数据。

数据管理比较方便，比如要清理或废弃某年的数据，就可以直接删除该日期的分区数据即可。

精准定位分区查询数据，不需要全表扫描查询，大大提高数据检索效率。

可跨多个分区磁盘查询，来提高查询的吞吐量。

在涉及聚合函数查询时，可以很容易进行数据的合并。

（3）分区的策略：

1.范围分区（RANGE）：就是根据数据库表中某一字段的值的范围来划分分区。如：年份小于2016的分成一个区，其它分成另一个区。

2.散列分区（HASH）：散列分区是根据字段的hash值进行均匀分布，尽可能的实现各分区所散列的数据相等。

3.列表分区（LIST）：列表分区明确指定了根据某字段的某个具体值进行分区，而不是像范围分区那样根据字段的值范围来划分的。如：长沙、武汉分成一个区，北京一个区。