

2015 年系统架构师考试科目二：案例分析

1. 阅读以下关于软件架构评估的说明，在答题纸上回答问题 1 和问题 2。

【题目】

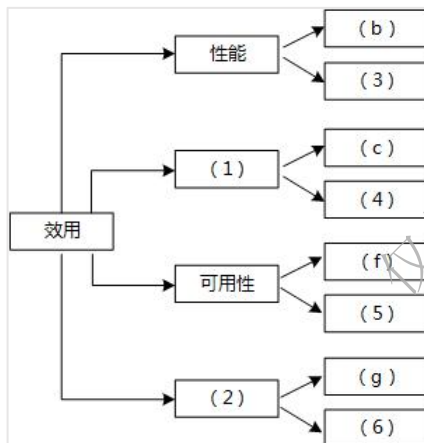
某软件公司拟为某市级公安机关开发一套特种车辆管理与监控系统，以提高特种车辆管理的效率和准确性。在系统需求分析与架构设计阶段，用户提出的部分需求和关键质量属性场景如下：

- (a)系统用户分为管理员、分管领导和普通民警等三类；
- (b)正常负载情况下，系统必须在 0.5 秒内对用户的车辆查询请求进行响应；
- (c)系统能够抵御 99.999%的黑客攻击；
- (d)系统的用户名必须以字母开头，长度不少于 5 个字符；
- (e)对查询请求处理时间的要求将影响系统的数据传输协议和处理过程的设计；
- (f)网络失效后，系统需要在 2 分钟内发现并启用备用网络系统；
- (g)在系统升级时，需要保证在 1 个月内添加一个新的消息处理中间件；
- (h)查询过程中涉及到的车辆实时视频传输必须保证 20 帧 / 秒的速率，且画面具有 600*480 的分辨率；
- (i)更改系统加密的级别将对安全性和性能产生影响；
- (j)系统主站点断电后，需要在 3 秒内将请求重定向到备用站点；
- (k)假设每秒中用户查询请求的数量是 10 个，处理请求的时间为 30 毫秒，则“在 1 秒内完成用户的查询请求”这一要求是可以实现的；
- (l)对用户信息数据的授权访问必须保证 99.999%的安全性；
- (m)目前对“车辆信息实时监控”业务逻辑的描述尚未达成共识，这可能导致部分业务功能模块的重复，影响系统的可修改性；
- (n)更改系统的 Web 界面接口必须在 1 周内完成；
- (o)系统需要提供远程调试接口，并支持系统的远程调试。

在对系统需求和质量属性场景进行分析的基础上，系统的架构师给出了三个候选的架构设计方案。公司目前正在组织系统开发的相关人员对系统架构进行评估。

【问题 1】(12 分)

在架构评估过程中，质量属性效用树(utility tree)是对系统质量属性进行识别和优先级排序的重要工具。请给出合适的质量属性，填入图 1-1 中(1)-(2)空白处；并选择题干描述中的(a)~(o)，将恰当的序号填入(3)~(6)空白处，完成该系统的效用树。



【问题 1 解析】【与 2017 年第 1 题、2014 年第 4 题类似】

质量属性效用包括: 性能、安全性、可用性、可修改性。

(1)~(2)空白处分别为安全性、可修改性。

(3)~(6)空白处分别为:

(3)~(h)—性能: 查询过程中涉及到的车辆实时视频传输必须保证 20 帧/秒的速率, 且画面具有 600*480 的分辨率。

(4)~(l)—安全性: 对用户信息数据的授权访问必须保证 99.999% 的安全性。

(5)~(j)—可用性: 系统主站点断电后, 需要在 3 秒内将请求重定向到备用站点。

(6)~(n)—可修改性: 更改系统的 Web 界面接口必须在 1 周内完成。

【问题 2】(13 分)

在架构评估过程中: 需要正确识别系统的架构风险、敏感点和权衡点, 并进行合理的架构决策。请用 300 字以内的文字给出系统架构风险、敏感点和权衡点的定义, 并从题干描述中的 (a)~(o) 各选出 1 个属于系统架构风险、敏感点和权衡点的描述。

【问题 2 解析】

系统架构风险: 架构设计中潜在的、存在问题的架构决策所带来的隐患(m)。

系统架构敏感点: 为了实现某种特定的质量属性, 一个或多个构件所具有的特性(e)。

系统架构权衡点: 影响多个质量属性的特征, 是多个质量属性的敏感点(i: 安全性和性能)。

2. 以下关于系统设计建模的说明, 在答题纸上回答问题 1 至问题 3。

【题目】

某公司拟研制一款高空监视无人直升机, 该无人机采用遥控—自主复合型控制实现垂直升降。该直升机飞行控制系统由机上部分和地面部分组成, 机上部分主要包括无线电传输设备、飞控计算机、导航设备等, 地面部分包括遥控操纵设备、无线电传输设备以及地面综合控制计算机等。其主要工作原理是地面综合控制计算机负责发送相应指令, 飞控计算机按照预定程序实现相应功能。经过需求分析, 对该无人直升机控制系统纵向控制基本功能整理如下:

(a) 飞控计算机加电后, 应完成系统初始化, 飞机进入准备起飞状态;

(b) 在准备起飞状态中等待地面综合控制计算机发送起飞指令, 飞控计算机接收到起飞指令后, 进入垂直起飞状态;

(c) 垂直起飞过程中如果飞控计算机发现飞机飞行异常, 飞行控制系统应转入无线电遥控飞行状态, 地面综合控制计算机发送遥控指令;

(d) 垂直起飞达到预定起飞高度后, 飞机应进入高度保持状态;

(e) 飞控计算机在收到地面综合控制计算机发送的目标高度后, 飞机应进入垂直升降状态, 接近目标高度; 垂直升降过程中出现飞机飞行异常, 控制系统应转入无线电遥控飞行;

(f) 飞机到达目标高度后, 应进入高度保持状态, 完成相应的任务;

(g) 飞机在接到地面综合控制计算机发送的任务执行结束指令后, 进入飞机降落状态;

(h) 飞机降落过程中如果出现飞机飞行异常, 控制系统应转入无线电遥控飞行;

(i) 飞机降落到指定着陆高度后, 进入飞机着陆状态, 应按照预定着陆算法, 进行着陆;

(j) 无线电遥控飞行中, 地面综合控制计算机发送着陆指令, 飞机进入着陆状态, 应按照预定着陆算法, 进行着陆。

【问题 1】(6 分)

状态图和活动图是软件系统设计建模中常用的两种手段, 请用 200 字以内文字简要说明状态图和活动图的含义及其区别。

【问题 1 解析】

状态图主要用于描述一个对象在其生存期间的动态行为,表现一个对象所经历的状态序列,引起状态转移的事件(event),以及因状态转移而伴随的动作(action)。

活动图可以用于描述系统的工作流程和并发行为。活动图其实可看作状态图的特殊形式,活动图中一个活动结束后将立即进入下一个活动(在状态图中状态的转移可能需要事件的触发)。

两者最大的区别是:状态图侧重于描述行为的结果,而活动图侧重描述行为的动作。其次活动图可描述并发行为,而状态图不能。

【问题 2】(10 分)

根据题干中描述的基本功能需求,架构师王工通过对需求的分析和总结给出了无人直升机控制系统纵向控制状态图(图 2-1)。请根据题干描述,提炼出相应状态及条件,并完善图 2-1 所示状态图中的(1)~(5),将答案填写在答题纸中。

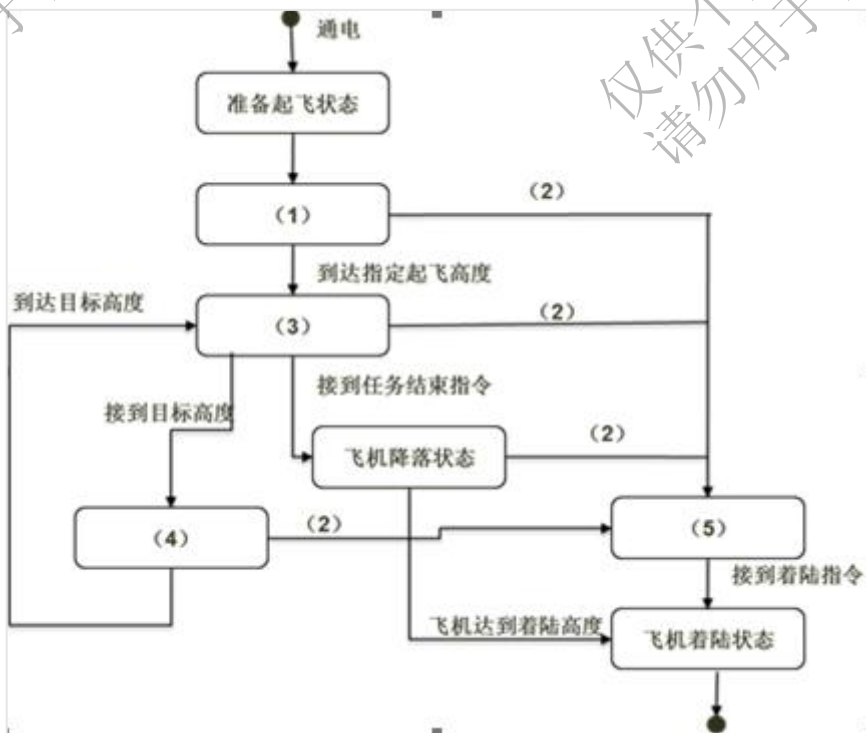
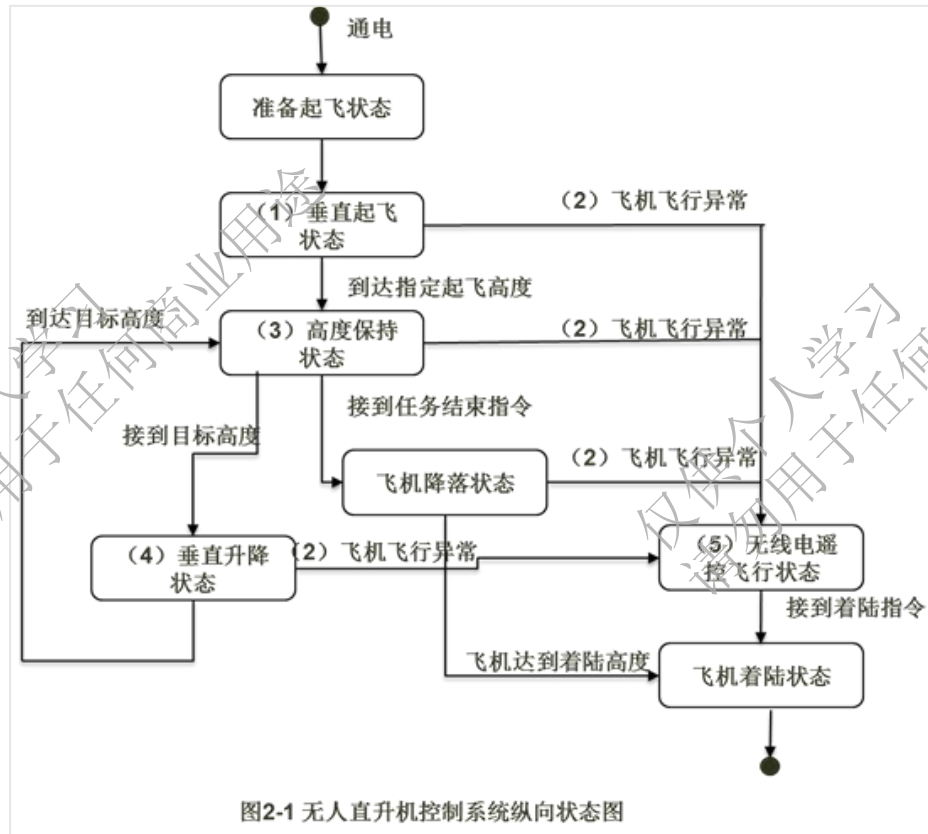


图 2-1 无人直升机控制系统纵向控制状态图

【问题 2 解析】

在解答本题时,首先需要对题目中描述的基本功能需求(a)~(j)进行分析与梳理,确定系统控制中的所有状态以及状态间的转换条件,再结合问题 2 中已经给出的状态,完成其余状态及条件的设计。



【问题 3】(9 分)

根据题目中描述的基本功能需求，架构师王工给出了无人直升机控制系统纵向控制的顶层活动图(图 2-2)。请根据题干描述，完善图 2-2 活动图的(1)-(9)，将答案填写在答题纸中。

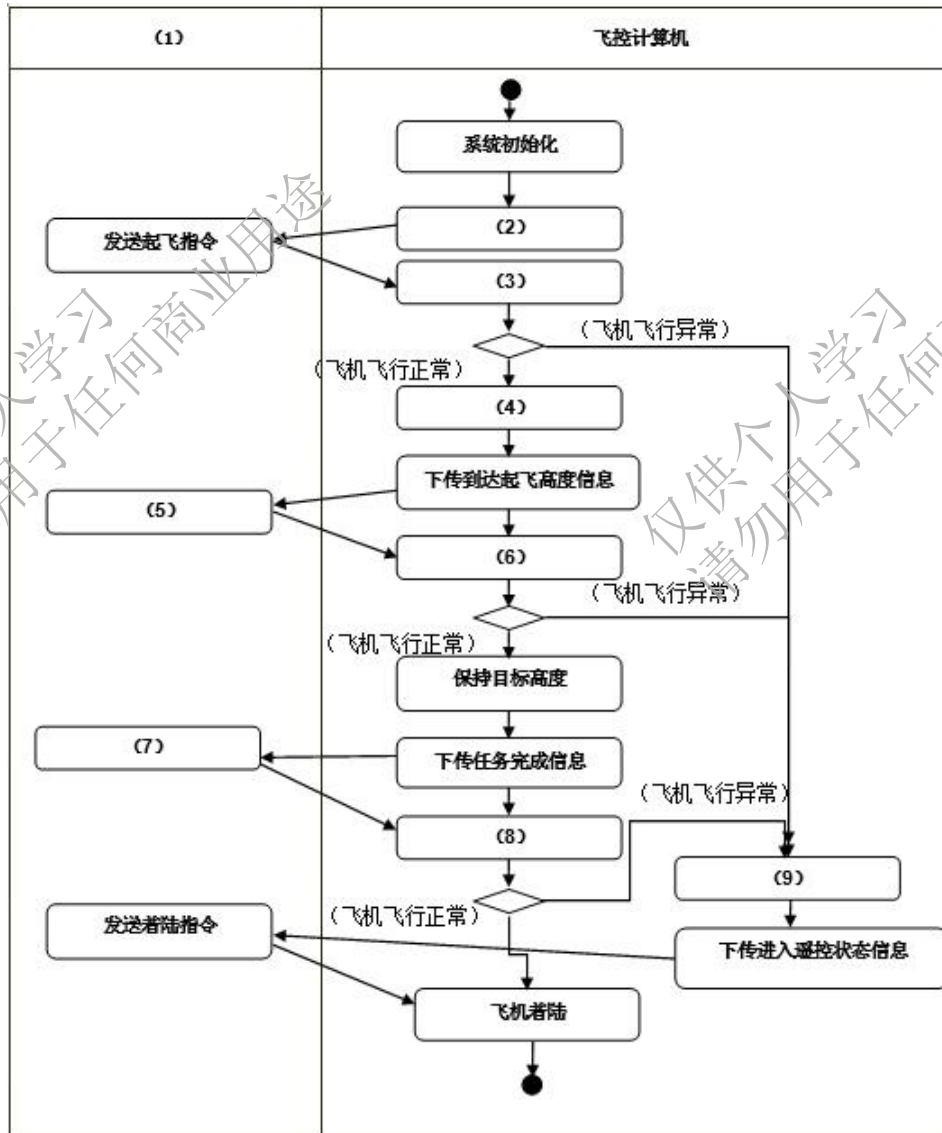
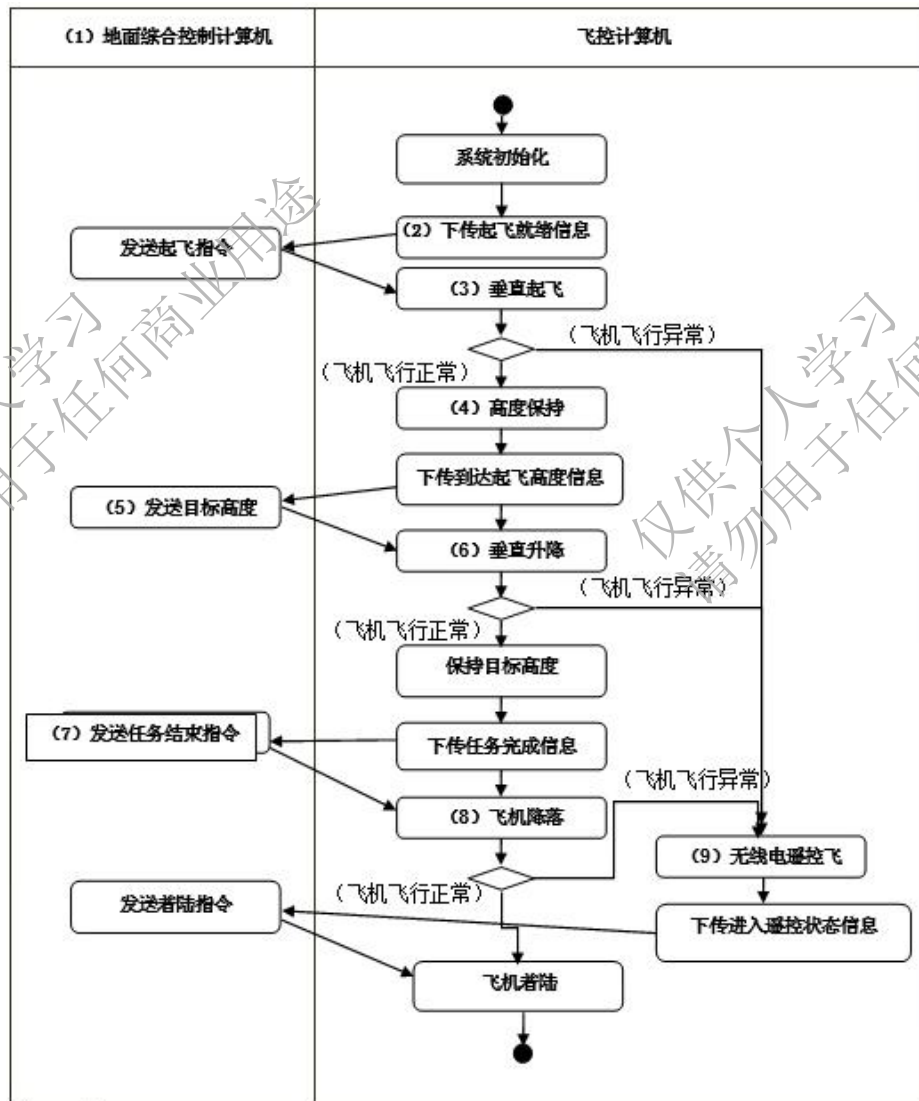


图 2-2 无人直升机控制系统纵向控制顶层活动图

【问题 3 解析】

本问题考查系统建模中活动图的设计与应用。考生应该掌握泳道活动图的概念并且学会应用。泳道活动图，是将一个活动图中放入活动状态进行分组，每一组表示一个特定的类或者对象，它们负责完成组内的活动。每个活动都明确属于一个泳道，不可以跨越泳道，而转移则可以跨越泳道。

在解答本题时，首先需要对题目中描述的基本功能需求进行分析与梳理，确定题目中存在哪些硬件设备与飞控计算机进行交互，以及设备间的交互关系，再结合问题 3 中已经给出的活动，完成其余活动及设备的设计。



3. 阅读以下关于嵌入式系统可靠性设计方面的描述, 回答问题 1 至问题 3。

【题目】

某宇航公司长期从事宇航装备的研制工作, 嵌入式系统的可靠性分析与设计已成为该公司产品研制中的核心工作, 随着宇航装备的综合化技术发展, 嵌入式软件规模发生了巨大变化, 代码规模已从原来的几十万扩展到上百万, 从而带来了由于软件失效而引起系统可靠性降低的隐患。公司领导非常重视软件可靠性工作, 决定抽调王工程师等 5 人组建可靠性研究团队, 专门研究提高本公司宇航装备的系统可靠性和软件可靠性问题, 并要求在三个月内, 给出本公司在系统和软件设计方面如何考虑可靠性设计的方法和规范。可靠性研究团队很快拿出了系统及硬件的可靠性提高方案, 但对于软件可靠性问题始终没有研究出一种普遍认同的方法。

【问题 1】(共 9 分)

请用 200 字以内文字说明系统可靠性的定义及包含的 4 个子特性, 并简要指出提高系统可靠性一般采用哪些技术?

【问题 1 解析】

可靠性(Reliability)是指产品在规定的条件下和规定的时间内完成规定功能的能力。

子特性: 成熟性, 容错性, 易恢复性, 可靠性的依从性。

提高可靠性的技术：

- (1) N 版本程序设计
- (2) 恢复块方法
- (3) 防卫式程序设计
- (4) 双机热备或集群系统
- (5) 冗余设计

【问题 2】(共 8 分)

王工带领的可靠性研究团队之所以没能快速取得软件可靠性问题的技术突破，其核心原因是他们没有搞懂高可靠性软件应具备的特点。软件可靠性一般致力于系统性地减少和消除对软件程序性能有不利影响的系统故障。除非被修改，否则软件系统不会随着时间的推移而发生退化。请根据你对软件可靠性的理解，给出表 3-1 所列出的硬件可靠性特征对应的软件可靠性特征之间的差异或相似之处，将答案写在答题纸上。

表3-1 硬件和软件可靠性对比

序号	硬件可靠性	软件可靠性
1	失效率服从浴缸曲线。老化状态类似于软件调试状态	(1)
2	即使不适用，材料劣化也会导致失效	(2)
3	硬件维修会恢复原始状态	(3)
4	硬件失效之前会有报警	(4)

【问题 2 解析】

- (1) 不考虑软件演化的情况下，失效率在统计上是非增的
- (2) 如果不使用该软件，永远不会发生失效
- (3) 软件维护会创建新的软件代码
- (4) 软件失效之前很少会有报警

【问题 3】(共 8 分)

王工带领的可靠性研究团队在分析了大量相关资料基础上，提出软件的质量和可靠性必须在开发过程构建到软件中，也就是说，为了提高软件的可靠性，必须在需求分析、设计阶段开展软件可靠性筹划和设计。研究团队针对本公司承担的飞行控制系统制定出了一套飞控软件的可靠性设计要求。飞行控制系统是一种双余度同构型系统，输入采用了独立的两路数据通道，在系统内完成输入数据的交叉对比、表决、制导率计算，输出数据的交叉对比、表决、一输出等功能，系统的监控模块实现对系统失效或失步的检测与定位。其软件的可靠性设计包括恢复块方法和 N 版本程序设计方法。请根据恢复块方法工作原理完成图 3-1，在(1)~(4)中填入恰当的内容。并比较恢复块方法与 N 版本程序设计方法，将比较结果(5)~(8)填入表 3-2 中。

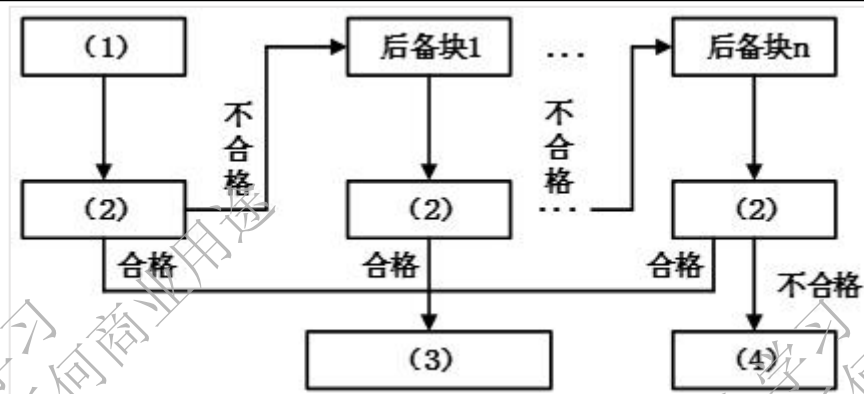


图 3-1 恢复块方法

表 3-2 恢复块方法与 N 版本程序设计的比较

	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	(5)
恢复策略	(6)	向前恢复
实时性	(7)	(8)

【问题 3 解析】

- (1)主块
- (2)验证测试
- (3)输出正确结果
- (4)异常处理

	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

- (5)表决
- (6)后向恢复
- (7)差
- (8)好

4. 阅读以下关于应用系统数据架构的说明，在答题纸上回答问题 1 至问题 3。

【题目】

某软件公司拟开发一套贸易综合管理系统,包括客户关系管理子系统和商品信息管理系统两部分。客户关系管理子系统主要管理客户信息,并根据贸易业务需要频繁向客户发送相关的电子邮件、短信等提醒信息。商品信息管理系统主要为客户提供商品信息在线查询功能,包括商品基本信息、实时库存与价格等。

在对系统进行数据架构设计时,公司项目组的架构师王工主张采用文件系统进行数据管理,原因是目前公司客户和商品数量不大,且系统功能较为简单,采用文件系统进行数据管理简单直观,开发周期短。架构师李工则建议采用关系数据库进行数据管理,原因在于公司目前正处在高速扩张期,虽然目前的客户和商品数量不大,但随着公司快速发展,需要管理的数据必然飞速膨胀,采用关系数据库作为数据存储层,系统的扩展性更强,并能够对未来可能增加的复杂业务提供有效支持。经过讨论,项目组初步采纳了李工的意见,决定采用关系数据库存储客户数据,并针对业务特征对系统性能进行优化。

【问题 1】(8 分)

请从设计难度、数据冗余程度、数据架构、应用扩展性等 4 个方面对关系型数据库管理系统和文件系统两种数据存储方式进行比较,填写表 4-1 中(1)~(4)。

表 4-1 关系型数据库管理系统和文件系统存储方式比较

	设计难度	数据冗余程度	数据架构	应用扩展性
关系型数据库	(1)	遵守数据库范式,数据冗余较少	以数据库为中心组织、管理数据	(4)
文件系统	针对特定应用系统设计,难度较小	(2)	(3)	符合特定应用系统要求的文件数据很难在不同的应用系统之间共享

【问题 1 解析】

- (1) 针对特定应用系统,难度较大。
- (2) 可能在多个文件中复制相同的数据属性,数据冗余较大。
- (3) 以应用为中心管理数据。
- (4) 数据库独立于应用系统,数据库系统接口标准化,易于在不同应用之间共享数据。

【问题 2】(13 分)

对系统的核心业务需求进行认真分析后,公司的资深架构师张工提出一种内存数据库和关系数据库的混合存储架构,其核心思想是将需要频繁读写的数据存入内存数据库,而将相对固定不变的数据存入关系数据库。请首先分析比较内存数据库和关系数据库在数据模型、读写性能、存储容量、可靠性等方面的差异,填写表 4-2 中(1)~(4)的空白,并根据张工的思路指定各种业务数据的存储方式,填写表 4-3 中(5)~(9)中的空白。

表 4-2 内存数据库和关系数据库比较

	主要数据模型	读写性能	存储容量	可靠性
内存数据库	(1)	内存直接读写,性能相对较高	(3)	(4)
关系数据库	关系模式	(2)	基于磁盘存储,存储容量大	内建恢复机制,可靠性较高

表 4-3 业务数据存储方式

业务数据	存储方式
客户基本信息	关系数据库
客户电子邮件	(5)
客户联系电话	(6)
商品基本信息	(7)
商品库存信息	(8)
商品价格信息	(9)

【问题 2 解析】

第一小问：

主要数据模型	读写性能	存储容量	可靠性	
内存数据库	Key-Value 模式 (键-值对模式)	内存直接读写， 性能相对较高	基于内存存储， 存储容量受限	恢复机制复杂， 可靠性较低
关系数据库	关系模式	外存读写，性能 相对较低	基于存盘存储， 存储容量大	内建恢复机制， 可靠性较高

(1)Key-Value 模式

(2)外存读写，性能相对较低

(3)基于内存存储，存储容量受限

(4)恢复机制复杂，可靠性较低

第二小问：

根据贸易综合管理系统的需求描述，需要频繁向客户发送相关的电子邮件、短信等提醒信息，并实时更新商品库存信息和商品价格信息，因此混合存储架构中，将客户电子邮件、客户联系电话、商品库存信息、商品价格信息等数据存入内存数据库；客户基本信息，商品基本信息相对稳定、访问频率较低，存入关系型数据库。

(5)内存数据库

(6)内存数据库

(7)关系数据库

(8)内存数据库

(9)内存数据库

【问题 3】(4 分)

系统开发完成进行压力测试时，发现在大数据量的情况下，部分业务查询响应时间过长，经过分析发现其主要原因是部分 SQL 查询语句效率低下。请判断表 4-4 中的 SQL 语句设计策略哪些可能会提升查询效率，哪些可能会降低查询效率，在(1)~(4)中填入“提升”或“降低”。

表 4-4 SQL 设计策略对性能的影响

SQL语句设计策略	查询性能影响情况
查询时不要返回不需要的行、列	(1)
需要进行多表连接查询时，尽量采用子查询结构	(2)
尽量采用NOT IN、NOT EXIST、LIKE等操作	(3)
尽量避免使用DISTINCT关键字	(4)

【问题 3 解析】

SQL 语句设计时, 影响查询效率的设计原则是:

- 查询时尽量不要返回不需要的行、列;
- 需要进行多表连接查询时, 尽量使用连接查询, 避免使用子查询结构;
- 尽量避免采用 NOT IN、NOT EXIST、LIKE 等使用全表查询的操作;
- 尽量避免使用 DISTINCT 关键字

(1)提升

(2)降低

(3)降低

(4)提升

5. 阅读以下关于 Web 应用系统架构设计的叙述, 在答题纸上回答问题 1 至问题 3。

【题目】

某信息技术公司计划开发一套在线投票系统, 用于为市场调研、信息调查和销售反馈等业务提供服务。该系统计划通过大量宣传和奖品鼓励的方式快速积累用户, 当用户规模扩大到一定程度时, 开始联系相关企业提供信息服务, 并按照信息服务种类和用户投票数量收取费用。

为了降低开发成本和提高开发效率, 项目组经过讨论后决定采用轻量级 Java EE 开发框架设计系统应用架构。在应用架构设计中, 除了满足系统主要功能需求, 还需要考虑的因素包括:

- (1)项目开发采用 MySQL 数据库存储数据, 一旦将来可能移植到其它数据库平台;
- (2)系统开发过程中尽可能降低或者消除 SQL 语句开发的工作量;
- (3)投票系统中数据之间的关系复杂, 需要支持数据对象的聚合和继承等关系。

项目组基于 MVC 模式设计出了投票系统的架构, 包括表示层、业务逻辑层、数据持久层和数据层。在具体讨论数据持久层采用哪种技术方案时, 老王建议采用成熟的 Hibernate 框架, 小李则认为 iBatis 更加灵活, 更适合作为投票系统数据持久层开发技术。

【问题 1】(5 分)

请用 300 以内文字说明什么是数据持久层, 使用数据持久层能够为项目开发带来哪些好处?

【问题 1 解析】

数据持久层是一组软件服务, 将应用程序与该程序所使用的数据库源分离, 为整个项目提供一个统一、安全、并发的数据持久机制。

好处:

- 1、程序代码重用性强, 即使更换数据库, 只需要更改配置文件, 不必重写程序代码。
- 2、业务逻辑代码可读性强, 在代码中不会有大量的 SQL 语言, 提高程序的可读性。
- 3、持久化技术可以自动优化, 以减少对数据库的访问量, 提高程序运行效率。
- 4、简化开发工作, 让开发人员更关注于业务逻辑的开发。
- 5、通过对象/关系映射向业务逻辑提供面向对象的数据访问。

【问题 2】(12 分)

针对在线投票系统的实际应用需求和要求, 项目组应选用哪种技术实现数据持久层? 请用 200 字以内文字说明其采用该技术的原因。

【问题 2 解析】

- 1、项目组应选 Hibernate 框架
- 2、选择该技术的原因是:

(1)从移植的角度来看使用 Hibernate 更容易移植到其它数据库平台。

Hibernate 与具体数据库的关联只需在 XML 文件中配置即可, 所有的 HQL 语句与具体使用的数据库无关, 移植性很好。MyBatis 项目中所有的 SQL 语句都是依赖所用的数据库的, 所以不同数据库类型的支持不好。

- (2)使用 Hibernate 能降低或者消除 SQL 语句开发工作量, Hibernate 提供了方法完成持久层操作, 程序员不需要对 SQL 的熟练掌握, 便可完成任务。
- (3)Hibernate 提供了对象状态管理的功能, 使开发者不再需要理会底层数据库系统的细节, 而 MyBatis 在这一块没有文档说明, 用户需要对对象自己进行详细的管理。

【问题 3】(8 分)

数据持久层是 Web 应用系统框架中重要的组成部分, 主流的数据持久层技术分别基于不同的技术方案, 请在表 5-1 中(1)-(4)处分别根据(a)~(d)所列技术的方案类别填入其序号。

表 5-1 数据持久层技术分类

技术方案	实现技术
JDBC封装	(1)
SQL Mapping	(2)
O/R Mapping	(3)
Entity Bean	(4)

- (a) BMP, CMP
- (b) iBatis/MyBatis
- (c) SprmgJdbcTemplate
- (d) TopLink,JDO,Hibernate

【问题 3 解析】

- (1) (c)
- (2) (b)
- (3) (d)
- (4) (a)

【问题分析】

iBatis 是 apache 的一个开源项目, 一个 O/R Mapping 解决方案, iBatis 最大的特点就是小巧, 上手很快。如果不需要太多复杂的功能, iBatis 是能够满足你的要求又足够灵活的最简单的解决方案, 现在的 iBatis 已经改名为 Mybatis 了。

EJB 有两种主要类型 BMP(Bean managed persistence) 和 CMP(Container managed persistence),这两种类型各有优缺点。

BMP 是在 Bean 中完成对数据库 JDBC 的各种调用, 也就是说, 在你的实体 bean(entity bean)中, 明确写入了 SQL 语句, 如"insert .."或"select ..",并且使用 Datasource 获得一个数据库资源以及连接(connection)从而对数据库直接进行增加删除修改。

CMP 是由 EJB 容器自动完成对数据库的操作, 你所有做的, 就是在实体 bean 重写入 SetXXX 或 getXXX 方法, 然后在 ejb-jar.xml 中定义 cmp-field 就可以。

Hibernate 对数据库结构提供了较为完整的封装,Hibernate 的 O/R Mapping 实现了 POJO 和数据库表之间的映射, 以及 SQL 的自动生成和执行。程序员往往只需定义好了 POJO 到数据库表的映射关系, 即可通过 Hibernate 提供的方法完成持久层操作。程序员甚至不需要对 SQL 的熟练掌握, Hibernate/OJB 会根据制定的存储逻辑, 自动生成对应的 SQL 并调用 JDBC 接口加以执行。

iBATIS 的着力点, 则在于 POJO 与 SQL 之间的映射关系。然后通过映射配置文件, 将 SQL 所需的参数, 以及返回的结果字段映射到指定 POJO。相对 Hibernate "O/R" 而言, iBATIS 是一种 "Sql Mapping" 的 ORM 实现。

Hibernate 的调优方案

- 制定合理的缓存策略;
- 尽量使用延迟加载特性;
- 采用合理的 Session 管理机制;
- 使用批量抓取, 设定合理的批处理参数 (batch_size);
- 进行合理的 O/R 映射设计

Mybatis 调优方案

MyBatis 在 Session 方面和 Hibernate 的 Session 生命周期是一致的, 同样需要合理的 Session 管理机制。MyBatis 同样具有二级缓存机制。MyBatis 可以进行详细的 SQL 优化设计。

SQL 优化方面

Hibernate 的查询会将表中的所有字段查询出来, 这一点会有性能消耗。Hibernate 也可以自己写 SQL 来指定需要查询的字段, 但这样就破坏了 Hibernate 开发的简洁性。而 Mybatis 的 SQL 是手动编写的, 所以可以按需求指定查询的字段。

Hibernate HQL 语句的调优需要将 SQL 打印出来, 而 Hibernate 的 SQL 被很多人嫌弃因为太丑了。MyBatis 的 SQL 是自己手动写的所以调整方便。但 Hibernate 具有自己的日志统计。Mybatis 本身不带日志统计, 使用 Log4j 进行日志记录。

扩展性方面

Hibernate 与具体数据库的关联只需在 XML 文件中配置即可, 所有的 HQL 语句与具体使用的数据库无关, 移植性很好。MyBatis 项目中所有的 SQL 语句都是依赖所用的数据库的, 所以不同数据库类型的支持不好。

对象管理

Hibernate 是完整的对象/关系映射解决方案, 它提供了对象状态管理 (state management) 的功能, 使开发者不再需要理会底层数据库系统的细节。也就是说, 相对于常见的 JDBC/SQL 持久层方案中需要管理 SQL 语句, Hibernate 采用了更自然的面向对象的视角来持久化 Java 应用中的数据。

换句话说, 使用 Hibernate 的开发者应该总是关注对象的状态 (state), 不必考虑 SQL 语句的执行。这部分细节已经由 Hibernate 掌管妥当, 只有开发者在进行系统性能调优的时候才需要进行了解。

而 MyBatis 在这一块没有文档说明, 用户需要对对象自己进行详细的管理。

抓取策略

Hibernate 对实体关联对象的抓取有着良好的机制。对于每一个关联关系都可以详细地设置是否延迟加载, 并且提供关联抓取、查询抓取、子查询抓取、批量抓取四种模式。它是详细配置和处理的。

而 Mybatis 的延迟加载是全局配置的。