选择题 40 分 主要是 EJB 和 JPA 的部分，下面给的是一些我在网上做的练习题和答案，如果有相同题目最好，没有就蒙吧

1 有状态 bean 的生命周期状态是：dose not exist,method ready,method ready in transaction,passive 2 无状态 bean 的生命周期状态是：dose not exist,method ready pool 3 实体 bean:pooled,ready 4 what is the sequence of steps in the life cycle of a stateless session bean? class.newInstance().setSessionContext(ctx),ejbcreat() 5 HTTPS is Http over ssl and http is connection base and stateless 6 Asynchronous communication is achieved by using Message Oriented Middleware,Decentralized messaging using Secure Sockets Layer is not part of Messaging architecture.7 in P2P,client send message to A designated queue.8.Security is an Architecture non-functional requirement and the mandatory checking account a functional design requirement.9.what non-functional requirements were discussed?Performance,availability,scalability and security 10if a computer can have exactly 1 processor,1 HDD and 1 memory stick,what pattern would be best used here? Builder 11.in which of the following cases would an application not necessarily benefit from the use of EJB? Small scale deployment and no transactional requirements 13.The container applies what memory management techniques in the case of Session beans?Bean pooling and bean passivation 14.what happens when the remove() method is called on the home interface of an entity bean?the remote reference is invalidated and the data represented by the bean instance id deleted from the database. 16.which two improve software maintainability？ Code factoring and reusing components .17what decreases software maintainability?Data sharing and high coupling .18whilch two features of a firewall might interfere with the operation of IIOP?Port filtering and address filtering 20.As an architecture,you are interested in the architecture characteristics of a system.which two are architectural characteristics? Reliability and performance.21.which two statements are true about management of an EJB's resources? The reference to home object is obtained through JNDI to improve maintainability and flexibility and the EJB container can manage the instance's access to its resourced because the remote object acts as a proxy. 24.what is benefit of bean poolong in an EJB container?it reduces the memory allocation and garbage-collection cycles. 25.which statement about an EJB container's life-cycle management of session beans is true?the container can passivate a stateful session bean to free limited resources. 26.what statement about passivating beans is true?the passivated stateful session bean should close all connections before being passivated 17.How would you encapsulate interaction with that legacy system?with an ejb session bean that uses JMS to message the legacy system. 18.which three statments about container-managed persistence are true?a container-managed enterprise bean requires less code,container-managed persistence provides less bean control,container-managed persistence an require complex mapping techniques. 19.what acts a proxy to an EJB? Remote instance.30.what is an alternative solution?use a stateful session bean as a Mediator to the entity beans.31.a client makes a transcationless method call an EJB that has a transaction attribute of required in the deployment descriptor.which state is true?the container creates a new transaction.31.which two statements about stateless session beans are true?they provide a generic service and they may provide high performance by being available for multiple clients.32.which pattern provides a means to access the elements on an aggregate object sequentially without exposing the undelying structure?literator.33.which pattern will solve the problem?Mediator.34.what are two benefits of the Facade pattern?it hides complex subsystem from client and it encourages weak coupling between the client and the subsystem.35.which design pattern is represented by EJB Home Interface?abstract factory.16.what are three benefits of design pattern?they act as a learning aid,they provide a common design vocabulary,they standardize the way designs are developed.17.what are two clear advantages to using message services in an application?Provides scalability ,allows loose coupling between components.18.which two statements about JMS are true?JMS supports Publish/Subscribe,JMS uses JNDI to find the destination.40.which three architectural decisions adhere to he requirements?(3)Make Order and account an entity EJB,Make customer a staeful-session EJB,make customer a session EJB and put business logic in it 41.which two services dose EJB provide?Transaction services,Life-cycle management..42.what types of EJB cannot use bean-managed transaction demarcation?BMP entity bean,CMP entity bean.43.Given an Order entity EJB and a single class loader, which of the following is a Singleton?OrderBemote 44.An EJB is required to process credit card purchases. All purchase-related information such as card number and amount will be supplied as parameters to a single business method invocation that processes the purchase. Server performance is critical. Which bean type BEST suits the requirement? A stateful session bean45.A container-managed persistence (CMP) bean A has a one-to-many container-managed relationship (CMR) with another container-managed persistence (CMP) bean B. Select the interface that will expose the methods related to this relationship. The home interface of bean A.46.Which of the following represent valid state transitions through an EJB method for a CMP entity Bean, according to its life cycle? "Does Not Exist" to "Pooled" via Class.newInstance() then setEntityContext(), "Pooled" to "Ready" via ejbCreate() then ejbPostCreate()47.A JMS session can implement: the TopicSession interface to support the publish-subscribe model,the QueueSession interface to support the point-to-point model.48.Which of the following is NOT a required step to get a reference to an existing EJB's home?Use the RMIRegistry lookup() method to get a remote reference to the EJB home.48. A stateful session bean is to be created. Which of the following objects are necessary?home interface,bean class, component interface 49.Given that a data source "FinanceDB" is already registered with JNDI, what are the steps that a JDBC application needs to take to create a connection to the database? "FinanceDB"Context ctx = new InitialContext(); DataSource ds = (DataSource)ctx.lookup("java:comp/env/jdbc/FinanceDB"); Connection con = ds.getConnection("j2ee", "hello");50.An EJB that is deployed with container-managed transaction demarcation has a business method that performs an operation that might throw a checked exception. The bean cannot recover from this checked exception and should rollback. Which implementation accomplishes this with the least amount of code and a maximum of information for the EJB client? public void businessMethod() throws EJBException {try {// operation throwing SomeCheckedException goes here} catch (SomeCheckedException ae) {context.setRollbackOnly();throw new EJBException(ae);}}51.A CMP entity bean implements a BankAccount record. The bean requires a composite primary key consisting of a branch number and an account number. Both values are of type int. The class AccountKey has been created to hold these values. Which of the following declarations of ejbCreate() would be valid ways to create a BankAccount bean, given that neither field can be automatically generated ?ejbCreate(int branchNumber, int accountNumber), ejbCreate(AccountKey key)52.Which of the following method declarations would MOST likely be found in the remote home interface of the entity bean named Customer?Customer createWithSSN(int id, String socialSecurityNumber) throws CreateException, RemoteException;Collection findAllByAge(int age) throws FinderException, RemoteException;52.In a Web application where performance is key, JDBC is used to query a database. The database is queried with parameters that are specified by users on a form. Which of the following are valid? aPreparedStatement.setString(2, "data");,aStatement.setString(2, "data");53.Which of the following are not considered tiers in a J2EE based n-tier model?EJB Integration Tier,Legacy Connectivity Tier55.Which of the following are not service level requirements that affect software architecture?detailed Design,Training,Design patterns55.Which of the following statements are true about the Entity class?Entity class must be declared as top level class,Enum can be declared as Entity56Which of the following are correct?@entity public class employees{..},public class employees implements serializable{...}57If you want to send an entity object as the pass by value through a remote interface, which of the following statements are valid?@entity public class employees implements serializable{...}58. Which of the following statements are correct?all the above 59Which of the following statements are correct?Entities supports inheritanc e,Abstract class and concrete classes can be entities,Instance variables of an entity class must be private,protected or package visibility60 Which of the following statements are correct?Every entity must have a primary key61.Select the valid primary key types?all the above62.Which of the following are true about composite primary keys?All the above63.Which of the following are not true about composite primary keys?The primary key class may not define equals and hashCode methods,May not have public no-arg constructor.

第二部分是 60 分 应该是两个 10 分的小题和一个 40 分的大题。根据重要性两个小题一个应该是关于 pattern（模式）的选择，就是给你一段描述或者一个场景，问应该要用哪个 pattern，下面给出重要的 pattern 描述。

数据流-管道过滤器风格，每个构件都有一组输入和输出，构件读取输入的数据流，经过内部处理产生数据流。优点：是的软件构建具有良好的隐蔽性和高内聚低耦合的特点，有助于设计者将整个系统的输入输出行为看成多个过滤器的行为的简单合成，支持软件重用，易于系统维护和增强系统性能，允许对一些如吞吐量，死锁等属性的分析，支持并行。缺点：通常导致进程成为批处理的操作，不适合处理交互应用，数据传输上没有通用的标准，每个过滤器都增加了解析和合成数据的工作，导致了系统性能下降，增加了编写过滤器的复杂性，使数据传输量增加。应用场景：编译器，Unix 管道，图像处理，信号处理，声音图像处理。数据集中式-黑板模式和仓库模式：有两种不同的构件，中央数据结构说明当前状态，独立构件在中央数据存贮上执行，仓库与外构件间的相互作用在系统中会有很大的变化。优点：善于管理数据信息，适合大量数据的应用场合，适用于复杂的逻辑系统，黑板系统模型能更容易处理任务间的协作。缺点是：不同数据源代理要对于共享数据达成一致，需要同步加锁机制，增大系统复杂性。每个程序试图解决总体任务的一部分，通过合作完成系统任务，没有明确调用顺序。当输入事务决定进行何种处理并把处理结果存储到中央数据结构中称为仓库，若中央数据结构决定何种处理称为黑板。应用场景：信号处理，数据库系统，语音识别，模式识别等。层系统风格：每一层为上一层服务并作为下层的客户，层次之间存在接口，各层模块具有不同类型的功能，将一个复杂问题分解成一个增量步骤序列的实现，为软件复用提供强大的支持。优点：使设计者可以把一个复杂系统按递增的步骤进行分解，支持功能增强，支持复用，对标准化的支持，可测试性。缺点：并不是每个系统都容易划出分层，效率会降低，很难找到合适的，正确的层次抽象方法。订阅模式（基于事件的风格，又称 publish-and-subscribe）notification architecture 定义对象间一种一对多的依赖关系。当一个对象被状态更新后，所有依赖他的对象都得到通知并更新，涉及到多个对象都对一个特殊对象的数据变化感兴趣，追踪那个对象的数据变化，是一种基于的风格。优点：软件服用以及改进系统，缺点：构件放弃了对系统计算的控制，数据交换存在问题，是否可以正确性推理。 C/S 风格 基于资源不对等且实现共享而提出来的，C/S 结构将应用一分为二。服务器负责数据管理，客户完成与用户的交互，C/S 体系结构具有强大的数据操作能力和事务处理能力。优点：降低通信量，采用请求响应而不是文件传输，多用户同时访问。缺点：系统伸缩性差，用户量过大会导致性能下降，更新成本大。 Interpreter 解释器模式：一个解释器就是一个虚拟机，有利于实现程序的可移植性和语言的的跨平台性，但影响性能。Broker 代理模式：代理者的作用仅仅是在客户机和服务之间的建立通信链。在分布式系统中，服务器向代理注册，使其接口能被客户机使用。客户机通过代理发送请求访问服务器功能。MVC（模型-视图-控制器）风格主要处理用户界面开发中面临的问题，视图为用户显示模型信息，视图从模型获取数据，一个模型可以有多个试图。模型是应用程序的核心，封装数据与状态，对模型的修改扩散到所有视图中。控制器接受用户的输入并调用模型去完成用户需求，控制器本身不输出任何东西和做任何处理。优点：将各方面问题分解考虑，简化系统设计，保证系统可扩展性。改变界面但不影响应用程序的功能内核，使系统易于演化开发，可维护性好。

注意：要是实在看不懂或者不会选，其中层模型，管道-过滤器模型，黑板模型，MVC 模型还有 C/S 模型是出现频率最高的，尽量在这 5 个里面挑一个，然后把上面的描述写上去，就算是选择这个模型的原因，这样卷子也会满一点。如果是多个模型的选择，比如有 5 个选择，一个选择 2 分那种，也从这里面选。

第二种小题 可能是给一个场景，主人公的系统因为升级或者某些客观原因导致了需要提升性能，会问你如何提升系统或者说系统有什么潜在的风险是选择什么模式或者哪个质量属性最重要。先给出各种质量属性。QA 运行时质量属性 performance(性能),usability(易用性),reliability(可靠性),security(安全性)

QA 非运行时质量属性 availability（可用性）maintainability(可维护性),evolvability(可进化性),testability(可测试性),reusability(复用性),integrability(可集成性),configurability(可配置性),scalaility(可伸缩性)。这个问题需要把题目读懂，然后自己编，比如说系统有什么潜在危险就说系统负载过大会崩溃，信息没有持久化可能会丢失，系统安全性不强等等，就是按照自己的话写，关键是要把题目读懂随机应变。这种问题就算不会也要把题目写满。

第三大题最重要，有 40 分，给两个在网上做的两个类似的题目：

Read the description below and do the following:4.1 Identify the most important quality attribute(s) for the system described below.4.2 Identify architectural driver for the system described below.4.3 Choose and describe suitable architectural tactics for the problem described below, and describe how the tactics affect the quality attributes.4.4 Create architecture views of the system described below. The architecture must be described in two views according to the 4+1 view model: Scenario view and Logical view. **Motivate for your choice of quality attributes, architectural drivers and the architectural tacticsused in your architecture. Software for Digital Camera ANSWER:**4. 1 Most important quality attributes: Modifiability, availability and performance are the main quality focus areas of the system. For modifiability it is important that you can use various kinds of hardware components (buttons, screens, data storage and optical components). For availability it is important that the software does not crash and the user of a camera expect no failure. For performance, it is important that the camera is responsive so it does not take too long before the user can take pictures or use the functionality of the camera. 4. 2 Architectural drivers for this system are: Variation in hardware components and configurations, and availability of the software. 4. 3 Architecture tactics: 4. 3. 1 Modifiability: Maintain semantic coherence, especially on the variation points for hardware components and interfaces. Hide information. Divide functionality of the camera into building blocks that can be added or removed on configuration. 4. 3. 2 Availability: Exceptions, heartbeat, checkpoint/rollback. 4. 3. 3 Performance: Schedule performance critical functionality. Scenario view: The scenario view shows the main actors (User, screen, optics and storage)and the main functionality of the camera. Logical view: The motivation for the logical view is to have good support for productline and modifiability. Dig.scr = digital screen, Opt.dev = optical device, Sto.dev = storage device.The extra functionality is divided into simple, medium and advanced to distinguish between different price segments of cameras with different level of functionality（这道题描述的是关于数字照相机系统的一些信息，因为篇幅原因不会全部贴上，如果你看到材料的标题是 **Software for Digital Camera，**很可能是原题，如果是原题，一定要把上面的答案和下面给出的图翻译成中文）

分析：正常第一问就是问你哪种属性最重要：如果题目中有说到系统要在各种环境下都能运行，什么情况下都能运行，就是可用性。如果对系统对响应时间有要求，那就是性能。如果说系统要在各个平台上，如手机，PC 都能运行，就是可维护性。如果你实在看不懂，上面三种情况都写上。第二问会问是是架构驱动，这主要看系统的侧重点，就是第一问里面的哪种属性最重要，这是基于对文章的判断而得到的，如果不会写，就写可用性。第三问会问实现这些属性的战术（tatics），下面给出各个属性的战术图：举个例子，写可用性的战术，这样写：可用性战术分为：错误检测（包括：命令/响应，心跳，异常），错误恢复（表决，主动冗余，被动冗余，shadow，状态再同步，回滚），错误预防包括（从服务中删除，事务，进程监视器）。可修改性和性能类似。

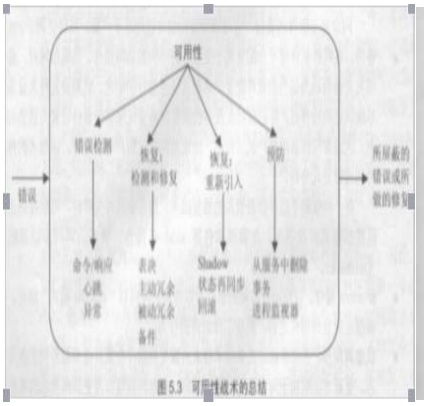


图 5.3 可用性战术的总结

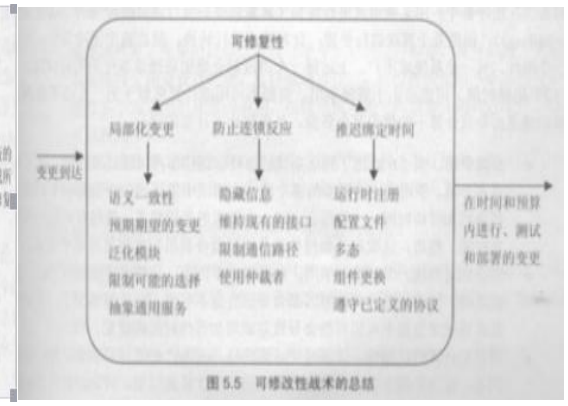


图 5.5 可修改性战术的总结

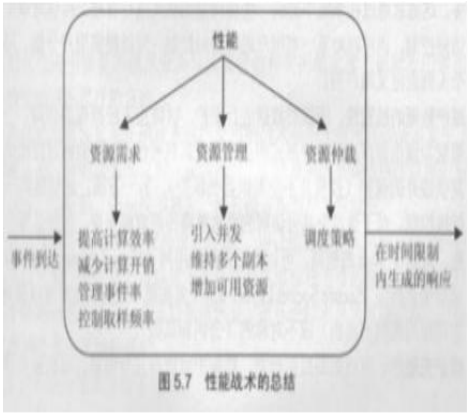
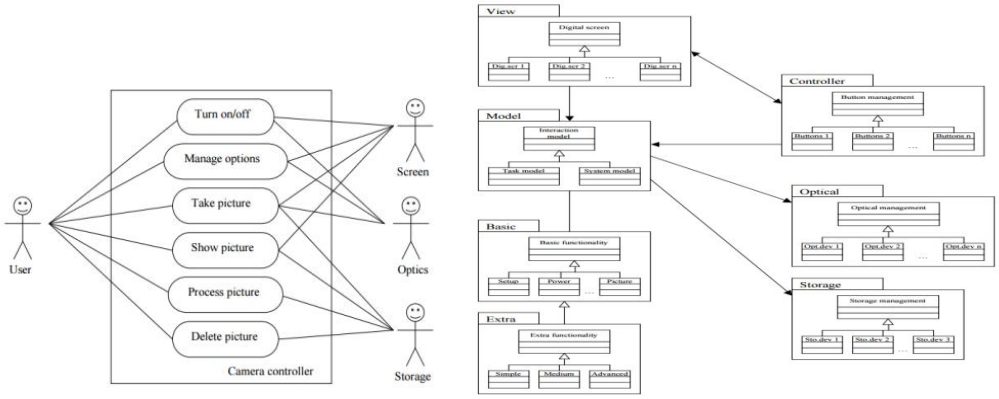
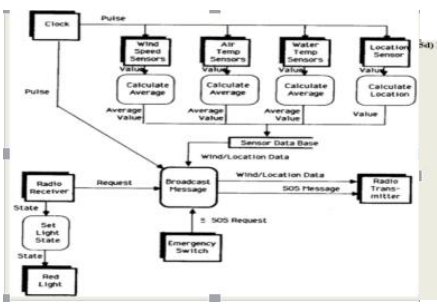


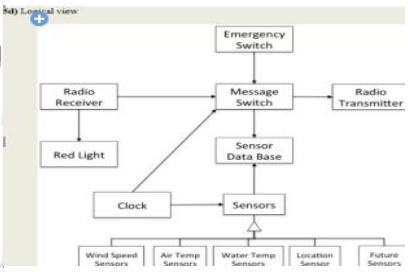
图 5.7 性能战术的总结

第四问和第五问通常都是画两个图，其中有逻辑视图，过程视图，物理视图，场景视图，过程试图这五种里面选两个。这个例题要画的是场景视图和逻辑视图（Create architecture views of the system described below. The architecture must be described in two views according to the 4+1 view model: Scenario view and Logical view.）答案是这样画的：场景视图应该就是像图一这样，外面的是使用这个系统的单位，方框里面是系统的功能以及每个单位可以使用系统的哪些性能。逻辑试图就是整个系统各个部件的逻辑，就是谁和谁交互，谁又控制谁的意思。（这种题目还是很难的，真不行就瞎画吧，答案不唯一，其实最关键是把题目读懂，读懂了知道每个视图什么意思总能画点东西出来）



例题二：Read the description below and do an architectural design. Your answer must include: a) Architectural drivers b) Architectural tactics and patterns – 3 points c) Process view d) Logical view Motivate for your choices and state your assumptions. **Software for sea buoys support for navigation at sea** .5a) Architectural drivers: Important quality attributes are * Availability: Operating on its own out in the sea * Performance: Timing requirements * Modifiability: Work with various sensors, might add new sensors in future with additional new functionality.5b) Architectural tactics and patterns Architectural tactics and patterns The architecture is based on the Task Control Architectural patterns, which basically is a way of communicating with various tasks by sending messages to a central server (named Message Switch in the architecture). Various tactics are used to achieve quality in various areas: * Modifiability: - Generalized interface for sensors: Makes it easier to add and change sensors - Increase semantic coherence: Clear separation of concerns and all classes has distinct and not overlapping responsibilities - Encapsulation: Interfaces only provides API to methods accessed by other classes. Rest remain private. * Availability: - Heartbeat: The Clock of the system is provided by separate hardware and can be used as a heartbeat and force restart if system does not react within time. - Self-test: System will do a self-test when starting up. - Exception Handling: Catch exceptions in the system, which can cause a restart of the system or broadcasting error if unfixable error. - Simplicity: Keep the system simple to make it easier to test and find errors. * Performance: - Reduce overhead: Keep the code simple - Schedule resources: Use the Clock to trigger events at the right time. Schedule so that sensors are read in sequence and not at the same time. 5c) Process view 这道题说的是一个海洋的定位系统,前三问就参考上一题的思路，然后出画图，要画的是（过程试图和逻辑试图）过程试图主要是说各个组件在系统的先后运行顺序图。





最后那些图基本都不怎么会画，要是考的不一样就参考那些方法（应该不会是原题的），最起码前三问按照说的方法来肯定有分，最后一题不会画图就随便画吧。