

1、什么是构件接口，在软件构件模型中如何定义其接口。

由于构件是一些二进制代码，它隐藏了内部的实现细节，而构件接口是构件提供服务的唯一通道，是外部可以唯一对构件所作的假设，包括对外提供的功能、对外请求的功能和服务。只有通过构件接口才能访问构件的服务以及能与其他构件一同工作。

构件接口：1、是对具有共同目标的服务的划分，描述了客户端如何与构件进行交互，同时隐藏了底层的实现细节。2、使潜在客户根据接口信息判断构件是否满足需求，客户程序可以使用一个接口中的部分或者全部服务。3、定义良好的接口可以促进构件开发商之间的竞争，也有利于灵活设计方式的建立。

如何定义接口：1、只有通过**对关键活动对象行为**的理解才能构造出有用的接口来。2、需要对**系统行为**进行抽象，定义独立的服务和描述它们之间的协作关系等。3、可以将**领域模型**中的对象及其协作关系，与已经定义好的接口规格说明相对照。当接口用于领域模型中时，接口之间的交互协议必须在领域模型中更为精确地表示出来。

构件接口定义可采用的方案：1、定义不变的接口（服务的不可删除性）2、定义自描述的接口（一般采用XML）3、定义元数据性质的接口4、构件设计成可定制的

2、采用 C++如何实现一个单对象支撑下的原子构件。

1、**定义接口规约：**在构件接口规约中定义其对外提供的功能和要求的外部功能。2、**制作构件实体：**将构件接口规约中定义的对外提供的功能和对外要求的功能，对应到构件的 C++ 实现中：①建立一个 C++ 的类，类名对应构件名②构件对外提供的功能，对应到类的 public 部分中的函数声明③构件对外要求的功能，对应到类的 protected 部分的一个名为 RequiredFunctions 的函数数组，每个数组元素存放一个对外要求的功能，并按照规约中描述的顺序在数组中存放。同时，在类的 public 部分包括一个 Initialize 函数，负责对 RequiredFunctions 数组进行初始化。④实现对外提供的功能对应的各个函数，当涉及到对外要求的功能时，使用 RequiredFunctions 数组中对应的函数。

3、什么是领域工程，叙述基本过程。

领域工程是为一组相似或相近系统的应用工程建立基本能力和必备基础的过程，它覆盖了建立可复用软件构件的所有活动，针对一个领域中的所有系统，而不局限于某个特定的系统。**领域过程**包括三个主要的阶段：①**领域分析：**这个阶段的主要目标是获得领域模型。领域模型描述领域中系统之间的共同的需求，并尽可能地识别和描述领域变化性。②**领域设计：**这个阶段的目标是获得领域构架（DSSA）。DSSA 描述在领域模型中表示的需求的解决方案，它不是单个系统的表示，而是能够适应领域中多个系统的需求的一个高层次的设计。③**领域实现：**实现 DSSA 和构件，并依据领域分析模型和 DSSA 组织可复用构件库。

4、什么是应用工程，叙述基本过程。

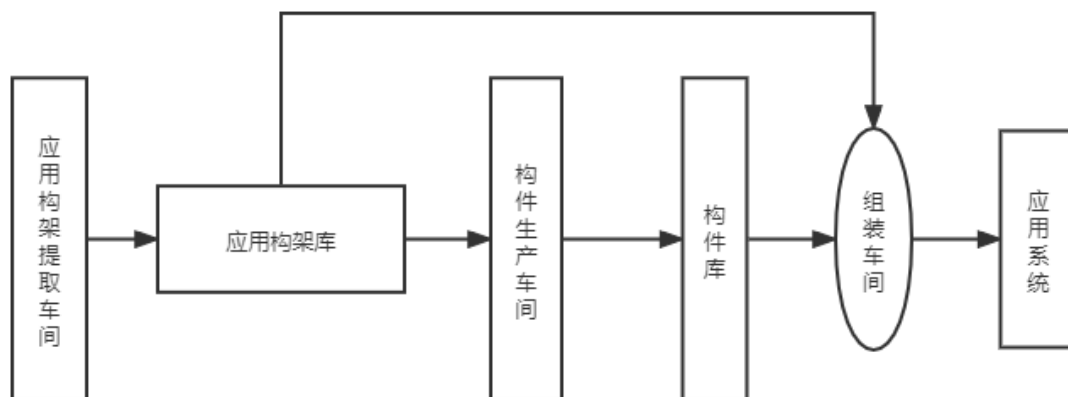
应用工程是开发单个特定应用系统的活动，针对一组特定的需求，产生一个特定的解决方案。应用过程由产品空间、构件与体系结构、生产计划、需求、产品和管理组成。

应用工程的过程如下：①**采用复用式开发。**复用的采用对开发过程会产生重大影响，这种影响取决于所使用的开发原则是自顶向下还是自底向上。一般来说，采用以复用为基础的自顶向下的应用系统开发原则要经过以下几个阶段：需求规格说明，用来描述应用系统的需求；产品设计，用来细化系统的设计，细化到能够确定可以在复用库中找到潜在设计单元的程度；检索，用来在复用库中查找单元，然后评估这些单元是否满足设计要求；编写没有找到的单元代码（在复用库中没有找到的单元）；最后是集成和测试。基于复用的自底向上原则与上述使用复用库的方式不同，它在需求规格说明之后开始使用复用库，而不是在设计之后查找复用库，而是在设计之前查找复用库。也就是说，要先熟悉复用库的内容，然后才能充分利用复用库资源，之后的步骤跟自顶向下设计相同。②**描述可复用资源。**在以应用为中心的结构中，领域工程团队受委托进行可复用资源的开发。这些可复用资产可以针对具体项目做出的委托，同时应用工程团队应理解这些可复用资源的长期复用潜力。③**复用的再工程。**作为开发项目的一个副产品，应用工程团队可以采用为特定应用系统开发的构件，并选择复用。构件的选择依据构件的复用潜力。在构件被集成到复用库之前，需要进行再工程，以满足复用库的可复用标准要求。

5、阐述基于 UML 模型驱动的软件开发过程。

①了解总体需求；②初步选择大粒度的 SA（软件体系结构）风格；③在 SA 风格的指导下，获取更详细的需求；④完成需求到 SA 各要素之间的映射；⑤详细设计 SA 各个要素之间的映射（构件、连接件及配置）；⑥设计 SA 的内在布局和功能配置；⑦正确性验证；⑧形成文档 SA 并进行评估；⑨对设计进行迭代优化。

6、使用图来描述青鸟软件产品线的构成。



青鸟软件产品线的构成包含“3个3”：

① 三个车间：构架生产车间、构件生产车间、产品组装车间。② 三类人员：构架生产者、构件生产者、产品组装者。③ 三种仓库：构架仓库、构件仓库、产品仓库。

7、阐述软件体系结构的定义。

从构造的角度审视软件体系结构：以下两个定义都强调了体系结构是由构件、连接件及其约束（或连接语义）组成的。

① Garlan & Shaw：软件体系结构是一个三元组：SA={components, connectors, constrains}（构件，连接件，约束）。

② CFRP：软件体系结构是一个四元组：SA={elements, interfaces, connections, connection-semantics}（元素，接口，连接，连接语义）

出于可用性的考虑，通过强调体系结构风格和规则等精化了以上两个定义：

③ Perry & Wolf：软件体系结构是一个三元组：SA={elements, form, rational}（元素，形式，准则）

④ Vestal：软件体系结构是一个三元组：SA={components, idioms/styles, common patterns of interaction}（构件，固定术语/风格，通用互操作模式）

以质量角度审视软件体系结构，以下两个定义不仅强调了体系结构的基本构成，还强调了体系结构的环境。不同的环境（需求、设计决策）将影响设计与演化的原理。

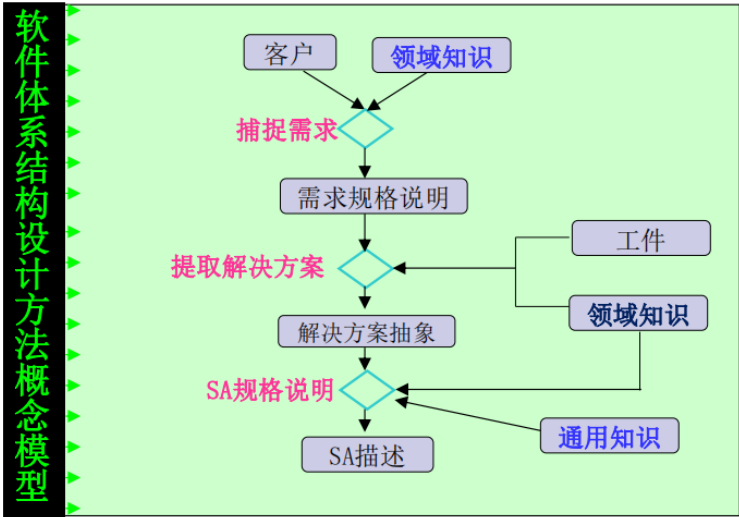
⑤ IEEE 610.12-1990 软件工程标准词汇中的定义：SA={component, connector, environment, principle}，即软件体系结构是一个四元组，是以：构件，构件之间的关系，构建与环境之间的关系为内容的某一系统的基本组织结构，以及指导上述内容设计与演化的原理。

⑥ Boehm：软件体系结构是一个五元组：SA={components, connections, constraints, stakeholders' needs, rationale}（构件，连接件，约束，涉众需求，能够展示由构件、连接件和约束所定义的系统在实现时如何满足系统不同人员需求的原理）。

总的来说软件体系结构的要素，主要涉及： * 构成成分(构件、连接件);* 拓扑结构(构件和连接件构成);* 限制;* 设计原则与指导方针

8、 阐述软件体系结构设计方法的统一模型。

SA设计的统一模型



9、 阐述软件体系结构编档的视图方法体系。

视图是对一组系统元素及其关系的描述。视图类型是定义描述 SA 的元素类型和关系类型。

- ① 模块视图类型：软件是如何设计和构造的，为系统的设计和实现单元编档。
- ② 构件-连接件（C&C）视图类型：软件作为一个运行时行为和交互作用的元素集是如何构造的，为系统的执行单元编档。
- ③ 分配视图类型：软件是如何在它所处的环境中与非软件结构产生联系的，为系统与环境之间的关系编档。

风格是对元素和关系类型的特化，以及如何使用这些元素和关系的一组限制条件。

视图类型、风格和视图的关系如下图所示：

	模块视图类型	C&C 视图类型	分配视图类型
风格	分解风格 使用风格 泛化风格 分层风格	管道-过滤器 共享数据风格 发布-订阅风格 C/S 风格 对等关系风格 通讯进程风格	部署风格 实现风格 工作任务风格

10、什么是涉众， 阐述 ATAM 方法。

涉众就是该 SA 及根据该 SA 开发的系统中有即得利益的人，包括系统的生产者，如开发人员、维护人员、集成人员、测试人员等；还包括系统的消费者，如客户、最终用户、应用开发等；还有系统服务人员，如系统管理员、网络管理员、服务代表；还有接触系统或与系统交互的人，如设备专家等。

ATAM 方法有 9 个步骤分别为：①ATAM 方法的表述：评估负责人向涉众介绍 ATAM 评估方法②商业动机的表述：项目决策者(最好是项目经理)从商业的角度介绍③SA 的表述：首席设计师对 SA 进行详略适当的介绍④确定 SA 方法：设计师对评估小组清楚地说明⑤生成质量属性效用树：评估小组和项目决策者(SA 小组,经理和客户代表)：共同确定出最重要的质量属性，确定优先级，用效用树表示⑥分析 SA 方法：评估小组对每一个 SA 方法探测足够多的信息，对第 4 步的 SA 方法、和第 5 步的质量需求的匹配情况进行分析⑦集体讨论并确定场景优先级：从场景开始，最终确定出它们要表达的质量属性⑧分析 SA 方法：将新得到的最高优先级的场景与尚未得到的 SA 工作产品对应起来⑨结果的表述：将质量需求的优先级划分、使用的 SA 方法、SA 方法和质量属性的待分析问题，效用树，有无风险决策、敏感点与权衡点等用 PPT+口头形式或者书面报告给出

ATAM 方法实施的 4 个阶段分别为：①建立评估小组、确立评估组织、以及组织之间的关系②以 SA 为中心，获取 SA 信息并对其进行分析③以涉众为中心，获取涉众的观点，并对第 1 阶段进行验证④形成报告，对存在的后续工作进行规划，并实现文档和经验的更新

11、描述常见的 6 中软件质量属性。

- ①可用性：系统能正常运行的时间比。经常用两次故障之间的时间长度、或出现故障时系统能恢复正常的速度来表示。
- ②可修改性：能快速的以较高的性价比对系统运行更改的能力。通常以某些更改为基准，通过考察这些更改的代价来衡量。
- ③性能：系统的响应能力。常用单位时间段内处理事件的个数或完成某个事务处理所需的时间来表示。
- ④安全性：系统向合法用户提供服务的同时，能阻止非法授权使用的企图或拒绝服务的能力。安全性是根据系统受到的安全威胁的类型来分类的。
- ⑤可测试性：经过测试揭示软件缺陷的容易程度。可测试语句的百分比；出现故障的概率；执行测试的时间等。
- ⑥易用性：系统能够长时间运行的能力。用平均无故障时间来衡量。

12、如何实现软件功能属性与质量属性在 UML 建模中的统一。

在具体的场景下软件的功能属性与质量属性可以统一。在 UML 建模中实现软件功能属性与质量属性的统一，即对具体场景进行 UML 建模。具体实现可以在需求分析阶段考虑软件应用的具体场景下的功能属性需求以及质量属性需求；在分析功能属性实现的可行性的同时分析质量属性实现的可行性。即通过模型映射的方式将具体的场景一层层映射到各个模型上从而进行 UML 建模。