

系统架构设计师

第8章 软件质量属性与架构评估

授课：王建平

目录

1

软件系统质量属性

2

系统架构评估

3

ATAM方法架构评估实践

目录

1

软件系统质量属性

2

系统架构评估

3

ATAM方法架构评估实践

软件系统的质量属性

质量属性概念

◆软件系统的质量就是“软件系统与明确地和隐含地定义的需求相一致的程度”。

◆影响软件质量的主要因素划分为6个维度特性：功能性、可靠性、易用性、效率、维护性与可移植性。（★★）

a) 功能性包括适合性、准确性、互操作性、依从性、安全性；

b) 可靠性包括容错性、易恢复性、成熟性；

c) 易用性包括易学性、易理解性、易操作性；

d) 效率包括资源特性和时间特性；

e) 维护性包括可测试性、可修改性、稳定性和易分析性；

f) 可移植性包括适应性、易安装性、一致性和可替换性。

口诀：适准作一安，错易成，学姐（解）坐（作），圆石（源时），试改定分，应装一蹄（替）

开发期质量属性

◆基于软件系统的生命周期，可以将软件系统的质量属性分为开发期质量属性和运行期质量属性2部分。

◆开发期质量属性（★★★）

开发期质量属性指在软件开发阶段所关注的质量属性，包括：

- (1)易理解性：指设计被开发人员理解的难易程度。
- (2)可扩展性：软件因适应新需求或需求变化而增加新功能的能力，也称为灵活性。
- (3)可重用性：指重用软件系统或某一部分的难易程度。
- (4)可测试性：对软件测试以证明其满足需求规范的难易程度。
- (5)可维护性：当需要修改缺陷、增加功能、提高质量属性时，识别修改点并实施修改的难易程度。
- (6)可移植性：将软件系统从一个运行环境转移到另一个不同的运行环境的难易程度。

运行期质量属性

◆运行期质量属性（★★★）

◆运行期质量属性指在软件运行阶段所关注的质量属性，包含：

- (1)性能：指软件系统及时提供相应服务的能力，如速度、吞吐量、容量等要求。
- (2)安全性：指软件系统同时兼顾向合法用户提供服务，以及阻止非授权使用的能力。
- (3)可伸缩性：指当用户数和数据量增加时，软件系统维持高服务质量的能力。例如通过增加服务器来提高能力。
- (4)互操作性：指本软件系统与其他系统交换数据和相互调用服务的难易程度。
- (5)可靠性：指软件系统在一定的时间内持续无故障运行的能力。
- (6)可用性：指系统在一定时间内正常工作的时间所占的比例。可用性会受到系统错误、恶意攻击，高负载等问题影响。
- (7)鲁棒性：指软件系统在非正常情况下(如用户进行了非法操作、相关的软硬件系统发生了故障等)，仍能够正常运行的能力，也称健壮性或容错性。

典型真题

例：软件系统质量属性(Quality Attribute)是一个系统的可测量或者可测试的属性，它被用来描述系统满足利益相关者需求的程度，其中，()关注的是当需要修改缺陷、增加功能、提高质量属性时，定位修改点并实施修改的难易程度；()关注的是当用户数和数据量增加时，软件系统维持高服务质量的能力。

- | | | | |
|-------|--------|---------|--------|
| A.可靠性 | B.可测试性 | C.可维护性. | D.可重用性 |
| A.可用性 | B.可扩展性 | C.可伸缩性. | D.可移植性 |

参考答案：C C

面向架构评估的质量属性

◆面向架构评估的质量属性

为了评价一个软件系统，特别是软件系统的架构，需要进行架构评估。评估方法关注的质量属性有以下几种。（★★★）

- 1.性能
- 2.可靠性
- 3.可用性
- 4.安全性
- 5.可修改性
- 6.功能性
- 7.可变性
- 8.互操作性

软件架构评估-质量属性-性能

◆性能：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。（★★★）

◆常见场景分析：（★★★）

- （1）同时支持1000并发。
- （2）响应时间应小于1s。
- （3）显示分辨率达到4k。

◆性能战术（★★★）

设计策略：

- 资源需求：**提高计算效率**、减少处理事件流所需的资源、减少所处理事件的数量、控制资源的使用
- 资源管理：**引入并发**、维持数据或计算的多个副本、增加可用资源
- 资源仲裁：**资源调度策略**、先进先出、固定优先级调度、动态优先级调度、静态调度

软件架构评估-质量属性-可靠性

◆可靠性 (★★★)

可靠性是软件系统在面对应用或系统错误时，在意外或错误使用的情况下维持软件系统的功能特性的基本能力。通常用来衡量在规定的条件和时间内，软件完成规定功能的能力。

(1)容错。容错的目的是在错误发生时确保系统正确的行为，并进行内部“修复”。

(2)健壮性。这里说的是保护应用程序不受错误使用和错误输入的影响，在发生意外错误事件时确保应用系统处于预先定义好的状态。

◆可靠性指标： (★★★)

- ✓ 平均失效间隔时间(MTBF)：是指相邻两次故障之间的平均时间，也称为平均故障间隔。
- ✓ 平均失效等待时间(MTTF)也称平均无故障时间，是指某个元件预计的可运作平均时间
- ✓ 平均故障修复时间(MTTR)：是描述产品由故障状态转为工作状态时修理时间的平均值。表示计算机的可维修性。

平均失效间隔时间(MTBF)常与MTTF发生混淆。因为两次故障之间必然有修复行为，因此，MTBF中应包含MTTR。

$MTBF = MTTR + MTTF$ 实际应用中MTTR很小，所以 $MTBF \approx MTTF$

软件架构评估-质量属性-可用性

◆可用性：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。如故障间隔时间。（★★★）

◆常见场景分析：（★★★）

- （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。

◆可用性战术（★★★）

■错误检测：**命令/响应**、心跳线、异常处理

■错误恢复：**表决**（通过冗余构件与表决器相联）、主动冗余、被动冗余、备件、状态再同步、检查点/回滚

■错误预防：从服务中删除、事务（事务保证一致性）、**进程监视器**

软件架构评估-质量属性-安全性

◆安全性：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。如保密性、完整性、不可抵赖性、可控性。（★★★）

◆常见场景分析：（★★★）

（1）可抵御SQL注入攻击。（2）对计算机的操作都有完整的记录。（3）用户信息数据库授权必须保证 99.9%可用。

◆设计策略：入侵检测、用户认证、用户授权、追踪审计。（★★★）

■抵抗攻击：对用户身份进行验证、授权、维护数据的机密性、完整性、限制暴露的信息、限制访问

■检测攻击：“**入侵检测**”

■从攻击中恢复：识别攻击者（**审计追踪**）、恢复（冗余）

软件架构评估-质量属性-可修改性

◆可修改性：指能够快速地以较高的性价比对系统进行变更的能力。通常以某些具体的变更为基准，通过考查变更的代价来衡量可修改性。可修改性包含以下 4个方面：（★★★）

(1)可维护性 (2)可扩展性 (3)结构重组 (4)可移植性

◆常见场景分析（★★★）

(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。

◆可修改性战术（★★★）

■ 局部化修改：**维持语义的一致性**、预期期望的变更、泛化该模块、限制可能的选择

■ 防止连锁反应：**信息隐藏**、维持现有的接口、限制通信路径、仲裁者的使用

■ 推迟绑定时间：运行时注册、配置文件、多态、构件更换、遵守已定义的协议。

软件架构评估-质量属性-功能、可变、互操作性

◆功能性 (★★)

功能性(Functionality)是系统能完成所期望的工作的能力。一项任务的完成需要系统中许多或大多数构件的相互协作。

◆可变性 (★★)

可变性(Changeability)是指架构经扩充或变更而成为新架构的能力。这种新架构应该符合预先定义的规则，在某些具体方面不同于原有的架构。当要将某个架构作为一系列相关产品(例如，软件产品线)的基础时，可变性是很重要的。

◆互操作性 (★★)

作为系统组成部分的软件不是独立存在的，通常与其他系统或自身环境相互作用。为了支持互操作性，软件架构必须为外部可视的功能特性和数据结构提供精心设计的软件入口。程序和用其他编程语言编写的软件系统的交互作用就是互操作性的问题，这种互操作性也影响应用的软件架构。

软件架构评估-质量属性-补

◆易用性：软件系统易于被使用的程度。（★★）

1、常见场景（1）界面友好。（2）新用户学习使用系统时间不超过2小时。

2、易用性战术

- 运行时：任务的模型、用户的模型、系统的模型（维护任务、用户、系统信息，了解系统用户的需求等）
- 设计时：接口与其余部分分离
- 支持用户主动操作：如**取消**、**撤销**等

◆可测试性：对软件系统进行测试以证明满足需求规格的能力。指的是通过测试揭示软件缺陷的容易程度。（★★）

1、常见场景：提供远程调试接口，支持远程调试。

2、可测试性策略

- 输入/输出：记录回放、**接口与实现分离**、优化访问线路接口
- 内部监控

典型真题

某公司欲开发一个人员管理系统，在架构设计阶段，公司的架构师识别出 3 个核心质量属性场景。其中“管理系统遭遇断电后，能够在 15 秒内自动切换至备用系统并恢复正常运行”主要与（1）质量属性相关，通常可采用（2）架构策略实现该属性；“系统正常运行时，人员信息查询请求应该在 2 秒内返回结果”主要与（3）质量属性相关，通常可采用（4）架构策略实现该属性；“系统需要对用户的操作情况进行记录，并对所有针对系统的恶意操作行为进行报警和记录”主要与（5）质量属性相关，通常可采用（6）架构策略实现该属性。

- | | | | | |
|-----|---------|-------------|---------|----------|
| (1) | A.可用性. | B.性能 | C.易用性 | D.可修改性 |
| (2) | A.抽象接口 | B.信息隐藏 | C.主动冗余. | D.影子操作 |
| (3) | A.可测试性 | B.易用性 | C.可用性 | D.性能. |
| (4) | A.记录/回放 | B.操作串行化 | C.心跳 | D.资源调度. |
| (5) | A.可用性 | B.安全性. | C.可测试性 | D.可修改性 |
| (6) | A.追踪审计. | B.Ping/Echo | C.选举 | D.维护现有接口 |

参考答案：（1）A，（2）C，（3）D（4）D，（5）B，（6）A

典型真题

某公司欲开发一个电子交易清算系统，在架构设计阶段，公司的架构师识别出3个核心质量属性场景。

其中“数据传递时延不大于1s，并提供相应的优先级管理”主要与（1）质量属性相关，通常可采用

（2）架构策略实现该属性：“系统采用双机热备，主备机必须实时监测对方状态，以便完成系统的实

时切换”主要与（3）质量属性相关，通常可采用（4）架构策略实现该属性：“系统应能够防止99%

的黑客攻击”主要与（5）质量属性相关，通常可采用（6）架构策略实现该属性。

(1) A. 可用性 B. 性能. C. 安全性 D. 可修改性

(2) A. 限制资源 B. 引入并发 C. 资源仲裁. D. 限制访问

(3) A. 可用性. B. 性能 C. 安全性 D. 可修改性

(4) A. 记录/回放 B. 操作串行化 C. 心跳. D. 资源调度

(5) A. 可用性 B. 性能 C. 安全性. D. 可修改性

(6) A. 检测攻击. B. Ping/Echo C. 选举 D. 权限控制

参考答案：BCACCA

质量属性场景描述

◆质量属性场景描述

为了精确描述软件系统的质量属性，通常采用质量属性场景作为描述质量属性的手段。质量属性场景是一个具体的质量属性需求，是利益相关者与系统的交互的简短陈述。

◆它由 6 部分组成：（★★）

- ✓ 刺激源：这是某个生成该刺激的实体(人、计算机系统或者任何其他刺激器)。
- ✓ 刺激：该刺激是当刺激到达系统时需要考虑的条件。
- ✓ 环境：该刺激在某些条件内发生。当激励发生时，系统可能处于过载、运行或者其他情况。
- ✓ 制品：某个制品被激励。可能是整个系统，也可能是系统的一部分。
- ✓ 响应：该响应是在激励到达后所采取的行动。
- ✓ 响应度量：当响应发生时，应当能够以某种方式对其进行度量，以对需求进行测试。

◆质量属性场景主要关注可用性、可修改性、性能、可测试性、易用性和安全性等 6 类质量属性。（★★★）

质量属性场景描述

◆可用性质量属性场景（★）

场景要素	可能的情况
刺激源	系统内部、系统外部
刺激	疏忽、错误、崩溃、时间
环境	正常操作、降级模式
制品	系统处理器、通信信道、持久存储器、进程
响应	系统应该检测事件、并进行如下一个或多个活动： 将其记录下来通知适当的各方，包括用户和其他系统；根据已定义的规则禁止导致错误或故障的事件源 在一段预先指定的时间间隔内不可用，其中，时间间隔取决于系统的关键程度在正常或降级模式下运行
响应度量	系统必须可用的时间间隔 可用时间 系统可以在降级模式下运行的时间间隔 故障修复时间

质量属性场景描述

◆可修改性质量属性场景（★）

场景要素	可能的情况
刺激源	最终用户、开发人员、系统管理员
刺激	希望增加、删除、修改、改变功能、质量属性、容量等
环境	系统设计时、编译时、构建时、运行时
制品	系统用户界面、平台、环境或与目标系统交互的系统
响应	查找架构中需要修改的位置，进行修改且不会影响其他功能，对所做的修改进行测试，部署所做的修改
响应度量	根据所影响元素的数量度量的成本、努力、资金；该修改对其他功能或质量属性所造成影响的程度

质量属性场景描述

◆性能质量属性场景（★）

场景要素	可能的情况
刺激源	用户请求，其他系统触发等
刺激	定期事件到达、随机事件到达、偶然事件到达
环境	正常模式、超载（Overload）模式
制品	系统
响应	处理刺激、改变服务级别
响应度量	等待时间、期限、吞吐量、抖动、缺失率、数据丢失率

质量属性场景描述

◆可测试性质量属性场景

场景要素	可能的情况
刺激源	开发人员、增量开发人员、系统验证人员、客户验收测试人员、系统用户
刺激	已完成的分析、架构、设计、类和子系统集成；所交付的系统
环境	设计时、开发时、编译时、部署时
制品	设计、代码段、完整的应用
响应	提供对状态值的访问，提供所计算的值，准备测试环境
响应度量	已执行的可执行语句的百分比 如果存在缺陷出现故障的概率 执行测试的时间 测试中最长依赖的长度 准备测试环境的时间

质量属性场景描述

◆易用性质量属性场景（★）

场景要素	可能的情况
刺激源	最终用户
刺激	想要学习系统特性、有效使用系统、使错误的影响最低、适配系统、对系统满意
环境	系统运行时或配置时
制品	系统
响应	<p>(1) 系统提供以下一个或多个响应来支持“学习系统特性”： 帮助系统与环境联系紧密；界面为用户所熟悉；在不熟悉的环境中，界面是可以使用的</p> <p>(2) 系统提供以下一个或多个响应来支持“有效使用系统”： 数据和（或）命令的聚合；已输入的数据和（或）命令的重用；支持在界面中的有效导航；具有一致操作的不同视图；全面搜索；多个同时进行的活动</p> <p>(3) 系统提供以下一个或多个响应来“使错误的影响最低”： 撤销；取消；从系统故障中恢复；识别并纠正用户错误；检索忘记的密码；验证系统资源</p> <p>(4) 系统提供以下一个或多个响应来“适配系统”： 定制能力；国际化</p> <p>(5) 系统提供以下一个或多个响应来使客户“对系统的满意”： 显示系统状态；与客户的节奏合拍</p>
响应度量	任务时间、错误数量、解决问题的数量、用户满意度、用户知识的获得、成功操作在总操作中所占的比例、损失的时间 / 丢失的数据量

质量属性场景描述

◆安全性质量属性场景（★）

场景要素	可能的情况
刺激源	正确识别、非正确识别身份未知的来自内部 / 外部的个人或系统；经过了授权 / 未授权它访问了有限的资源 / 大量资源
刺激	试图显示数据，改变 / 删除数据，访问系统服务，降低系统服务的可用性
环境	在线或离线、联网或断网、连接有防火墙或者直接连到了网络
制品	系统服务、系统中的数据
响应	对用户身份进行认证；隐藏用户的身份；阻止对数据或服务的访问；允许访问数据或服务；授予或收回对访问数据或服务的许可；根据身份记录访问、修改或试图访问、修改数据、服务；以一种不可读的格式存储数据；识别无法解释的对服务的高需求；通知用户或另外一个系统，并限制服务的可用性
响应度量	用成功的概率表示，避开安全防范措施所需要的时间、努力、资源；检测到攻击的可能性；确定攻击或访问、修改数据或服务的个人的可能性；在拒绝服务攻击的情况下仍然获得服务的百分比；恢复数据、服务；被破坏的数据、服务和（或）被拒绝的合法访问的范围

典型真题

例：为了精确描述软件系统的质量属性，通常采用质量属性场景(Quality Attribute Scenario)作为描述质量属性的手段。质量属性场景是一个具体的质量属性需求，是利益相关者与系统的交互的简短陈述，它由刺激源、刺激、环境、制品、(1)六部分组成。其中，想要学习系统特性、有效使用系统、使错误的影响最低、适配系统、对系统满意属于(2)质量属性场景的刺激。

- (1) A.响应和响应度量. B.系统和系统响应 C.依赖和响应 D.响应和优先级
(2) A可用性 B.性能 C.易用性. D.安全性

参考答案：AC

目录

1

软件系统质量属性

2

系统架构评估

3

ATAM方法架构评估实践

系统架构评估的评估方法

- ◆系统架构评估是在对架构分析、评估的基础上，对架构策略的选取进行决策。它利用数学或逻辑分析技术，针对系统的一致性、正确性、质量属性、规划结果等不同方面，提供描述性、预测性和指令性的分析结果。
- ◆系统架构评估的方法通常可以分为3类：基于调查问卷或检查表的方式、基于场景的方式和基于度量的方式。
(★★★)
 - (1)基于调查问卷或检查表的方法。该方法的关键是要设计好问卷或检查表，充分利用系统相关人员的经验和知识，获得对架构的评估。该方法的缺点是在很大程度上依赖于评估人员的主观推断。
 - (2)基于场景的评估方法。基于场景的方式由卡耐基梅隆大学软件工程研究所首先提出并应用在架构权衡分析法(Architecture Tradeoff Analysis Method,ATAM)和软件架构分析方法(Software ArchitectureAnalysis Method,SAAM)中。它是通过分析软件架构对场景(也就是对系统的使用或修改活动)的支持程度，从而判断该架构对这一场景所代表的质量需求的满足程度。
 - (3)基于度量的评估方法。它是建立在软件架构度量的基础上的，涉及3个基本活动，首先需要建立质量属性和度量之间的映射原则，然后从软件架构文档中获取度量信息，最后根据映射原则分析推导出系统的质量属性。

表 13-1 三类评估方式比较表				
评 估 方 式	调查问卷或检查表		场 景	度 量
	调 查 问 卷	检 查 表		
通用性	通用	特定领域	特定系统	通用或特定领域
评估者对体系结构的了解程度	粗略了解	无限制	中等了解	精确了解
实施阶段	早	中	中	中
客观性	主观	主观	较主观	较客观

系统架构评估-概念

系统架构评估中的重要概念 (★★★)

◆敏感点(Sensitivity Point)和权衡点(Tradeoff Point)。

- ✓ 敏感点是一个或多个构件(和或构件之间的关系)的特性。研究敏感点可使设计师明确在搞清楚如何实现质量目标时应注意什么。
- ✓ 权衡点是影响多个质量属性的特性，是多个质量属性的敏感点。例如，改变加密级别可能会对安全性和性能产生非常重要的影响。提高加密级别可以提高安全性，但可能要耗费更多的处理时间，影响系统性能。如果某个机密消息的处理有严格的时间延迟要求，则加密级别可能就会成为一个权衡点。

◆风险承担者 (Stakeholders)或者称为利益相关人。系统的架构涉及很多人的利益，这些人都会对架构施加各种影响，以保证自己的目标能够实现。

◆场景

在进行架构评估时，一般首先要精确地得出具体的质量目标，并以之作为判定该架构优劣的标准。为得出这些目标而采用的机制称之为场景。

场景是从风险担者的角度对与系统的交互做的简短描述。在架构评估中，一般采用刺激、环境和响应三方面来对场景进行描述。(★★)

典型真题

在架构评估中，（1）是一个或多个构件（和 / 或构件之间的关系）的特性。改变加密级别的设计决策属于（2），因为它可能会对安全性和性能产生非常重要的影响。

A.敏感点 B. 非风险点 C. 权衡点 D. 风险点

A.敏感点 B. 非风险点 C. 权衡点 D. 风险点

参考答案：A C

识别风险、非风险、敏感点和权衡点是进行软件架构评估的重要过程。“改变业务数据编码方式会对系统的性能和安全性产生影响”是对（1）的描述，“假设用户请求的频率为每秒1个，业务处理时间小于30毫秒，则将请求响应时间设定为1秒钟是可以接受的”是对（2）的描述。

（1）A. 风险点 B.非风险 C.敏感点 D.权衡点.

（2）A. 风险点 B.非风险. C.敏感点 D.权衡点

参考答案：（1）D （2）B

系统架构评估的评估方法

◆三种典型架构评估方式又包括：(★★★)

- (1) 软件架构分析法-SAAM方法：最初关注可修改性，后扩充到可移植性、可扩充性等。
- (2) 架构权衡分析法-ATAM方法：由SAAM发展而来，主要针对性能、可用性、安全性、可修改性。
- (3) 成本效益分析法-CBAM方法，在ATAM基础上建立的软件的“经济”模型。

系统架构评估的评估方法-SAAM

◆SAAM方法是最早形成文档并得到广泛使用的软件体系结构分析方法，最初是用来分析体系结构的可修改性的，但实践证明，SAAM方法也可用于其他质量属性如可移植性、可扩充性等，最终发展成了评估一个系统架构的通用方法。

◆SAAM 分析评估架构的过程包括5 个步骤，即场景开发、架构描述、单个场景评估、场景交互和总体评估。
(★★★★)

通过各类风险承担者协商讨论，开发一些任务场景，体现系统所支持的各种活动。用一种易于理解的、合乎语法规则的架构描述软件架构，体现系统的计算构件、数据构件以及构件之间的关系(数据和控制)。对场景(直接场景和间接场景)生成一个关于特定架构的场景描述列表。通过对场景交互的分析，得出系统中所有场景对系统中的构件所产生影响的列表。最后，对场景和场景间的交互作一个总体的权衡和评价。

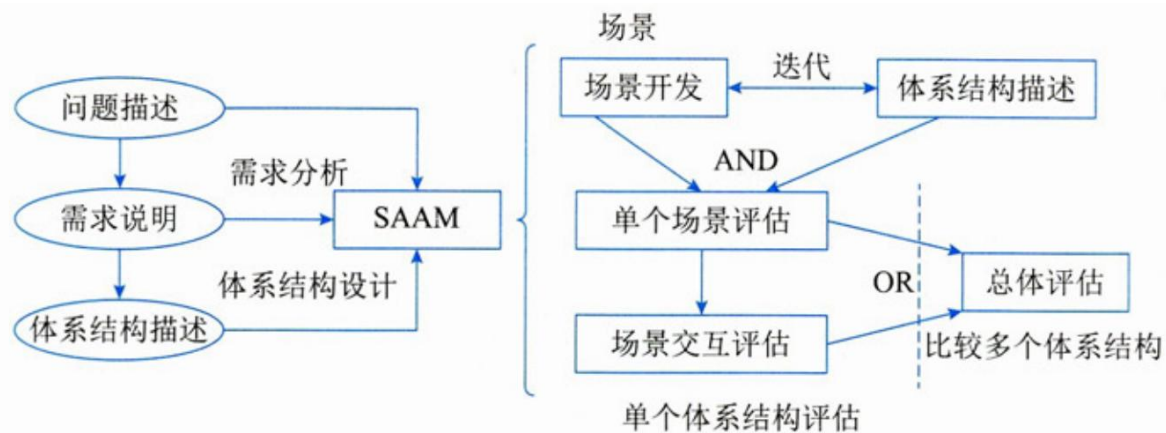


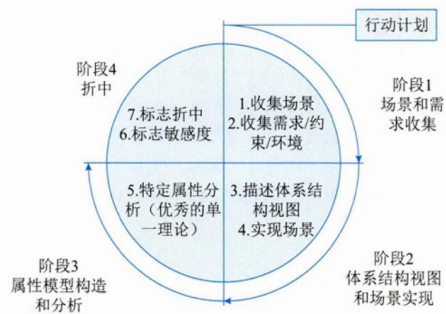
图 8-1 SAAM 输入与评估过程

系统架构评估的评估方法-ATAM

◆架构权衡分析方法(ATAM)是评价软件构架的一种综合全面的方法。主要针对性能、可用性、安全性和可修改性，在系统开发之前，可以使用ATAM方法在多个质量属性之间进行评价和折中。（★★★）

◆ATAM 分为4个主要的活动领域(阶段)

- ✓ 场景和需求收集
- ✓ 架构视图和场景实现
- ✓ 属性模型构造和分析
- ✓ 折中



◆ATAM方法采用效用树 (Utility tree) 这一工具来对质量属性进行分类和优先级排序。效用树的结构包括:树根→质量属性→属性分类→质量属性场景 (叶子节点)。（★★★）

◆ATAM 主要关注4 类质量属性：性能、安全性、可修改性和可用性，因为这4 个质量属性是利益相关者最为关心的。得到初始的效用树后，需要修剪这棵树，保留重要场景(不超过50个)，再对场景按重要性给定优先级(用H/M/L 的形式)，再按场景实现的难易度来确定优先级(用H/M/L的形式)，这样对所选定的每个场景就有一个优先级对(重要度、难易度)，如 (H、L)表示该场景重要且易实现。

系统架构评估的评估方法-CBAM

1、成本效益分析法 (CBAM)是在ATAM上构建，用来对架构设计决策的成本和收益进行建模，是优化此类决策的一种手段。CBAM 的思想就是架构策略影响系统的质量属性，反过来这些质量属性又会为系统的项目干系人带来一些收益(称为效用)，CBAM 协助项目干系人根据其投资回报 (ROI)选择架构策略。CBAM在ATAM结束时开始，它实际上使用了 ATAM评估的结果。

2、CBAM 方法分为以下8个步骤：

- (1)整理场景。
- (2)对场景进行求精。
- (3)确定场景的优先级。
- (4)分配效用。
- (5)架构策略涉及哪些质量属性及响应级别，形成"策略—场景—响应级别"的对应关系。
- (6)使用内插法确定"期望的"质量属性响应级别的效用。
- (7)计算各架构策略的总收益
- (8)根据受成本影响的ROI选择架构策略

典型真题

试题一

阅读以下关于软件架构评估的叙述，在答题纸上回答问题1和问题2.

【说明】

某单位为了建设健全的公路桥梁养护管理档案，拟开发一套公路桥梁在线管理系统。在系统的需求分析与架构设计阶段，用户提出的需求、质量属性描述和架构特性如下：

- (a) 系统用户分为高级管理员、数据管理员和数据维护员等三类；--功能
- (b) 系统应该具备完善的安全防护措施，能够对黑客的攻击行为进行检测与防御；--安全性
- (c) 正常负载情况下，系统必须在 0.5 秒内对用户的查询请求进行响应；--性能
- (d) 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；--敏感点
- (e) 系统的用户名不能为中文，要求必须以字母开头，长度不少于5个字符；--设计约束
- (f) 更改系统加密的级别将对安全性和性能产生影响；--权衡点
- (g) 网络失效后，系统需要在 10 秒内发现错误并启用备用系统；--可用性
- (h) 查询过程中涉及到的桥梁与公路的实时状态视频传输必须保证画面具有1024*768的分辨率， 40帧 /秒的速率；--性能
- (i) 在系统升级时，必须保证在 10 人月内可添加一个新的消息处理中间件；--可修改性
- (j) 系统主站点断电后，必须在 3 秒内将请求重定向到备用站点；--可用性

典型真题

试题一

阅读以下关于软件架构评估的叙述，在答题纸上回答问题1和问题2.

【说明】

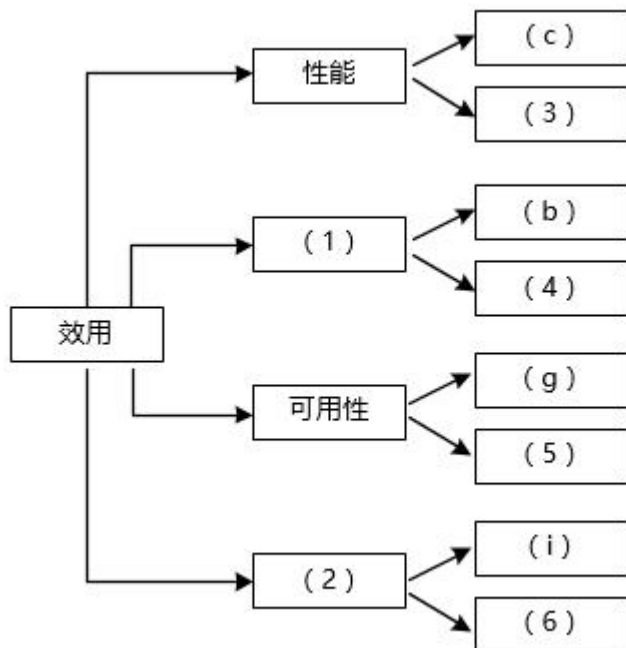
- (k) 如果每秒钟用户查询请求的数量是 10 个，处理单个请求的时间为 30 毫秒，则系统应保证在 1 秒内完成用户的查询请求；--性能
- (l) 对桥梁信息数据库的所有操作都必须进行完整记录；--安全性
- (m) 更改系统的 Web 界面接口必须在 4 人周内完成；--可修改性
- (n) 如果"养护报告生成"业务逻辑的描述尚未达成共识，可能导致部分业务功能模块规则的矛盾，影响系统的可修改性--风险点
- (O) 系统必须提供远程调试接口，并支持系统的远程调试。--可测试性

在对系统需求，质量属性描述和架构特性进行分析的基础上，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。

典型真题

【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；并选择题干描述的 (a)~(o)，填入(3)~(6) 空白处，完成该系统的效用树。



(1) 安全性

(2) 可修改性

(3) (h)

(4) (l)

(5) (j)

(6) (m)

典型真题

【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(a)~(o) 中分别选出 1 个对系统架构风险、敏感点和权衡点最为恰当的描述。

系统架构风险是指架构设计中潜在的、存在问题的架构决策所带来的隐患。

敏感点是指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。

权衡点是影响多个质量属性的特性，是多个质量属性的敏感点。

风险点：(n)

敏感点：(d)

权衡点：(f)

典型真题

效用树是采用架构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)进行架构评估的工具之一，其树形结构从根部到叶子节点依次为()。

- A.树根、属性分类、优先级，质量属性场景
- B.树根、质量属性、属性分类，质量属性场景
- C.树根、优先级、质量属性、质量属性场景
- D.树根、质量属性、属性分类，优先级

参考答案：B

典型真题

体系结构权衡分析方法（Architecture Tradeoff Analysis Method, ATAM）包含 4 个主要的活动领域，分别是 场景和需求收集、体系结构视图和场景实现、（1）、折中。基于场景的架构分析方法（Scenarios-based Architecture Analysis Method, SAAM）的主要输入是问题描述、需求声明和（2）。

- | | | | |
|---------|------------|---------------|---------|
| A. 架构设计 | B. 问题分析与建模 | C. 属性模型构造和分析. | D. 质量建模 |
| A. 问题说明 | B. 问题建模 | C. 体系结构描述. | D. 需求建模 |

参考答案：CC

架构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)是在基于场景的架构分析方法（Scenarios-based Architecture Analysis Method, SAAM）基础之上发展起来的，主要包括场景和需求收集、（1），属性模型构造和分析，属性模型折中等四个阶段。ATAM方法要求在系统开发之前，首先对这些质量属性进行（2）和折中。

- （1）
- | | |
|---------------|--------------|
| A. 架构视图和场景实现. | B. 架构风格和场景分析 |
| C. 架构设计和目标分析 | D. 架构描述和需求评估 |
- （2）
- | | | | |
|-------|-------|-------|--------|
| A. 设计 | B. 实现 | C. 测试 | D. 评价. |
|-------|-------|-------|--------|

参考答案：AD

目录

1

软件系统质量属性

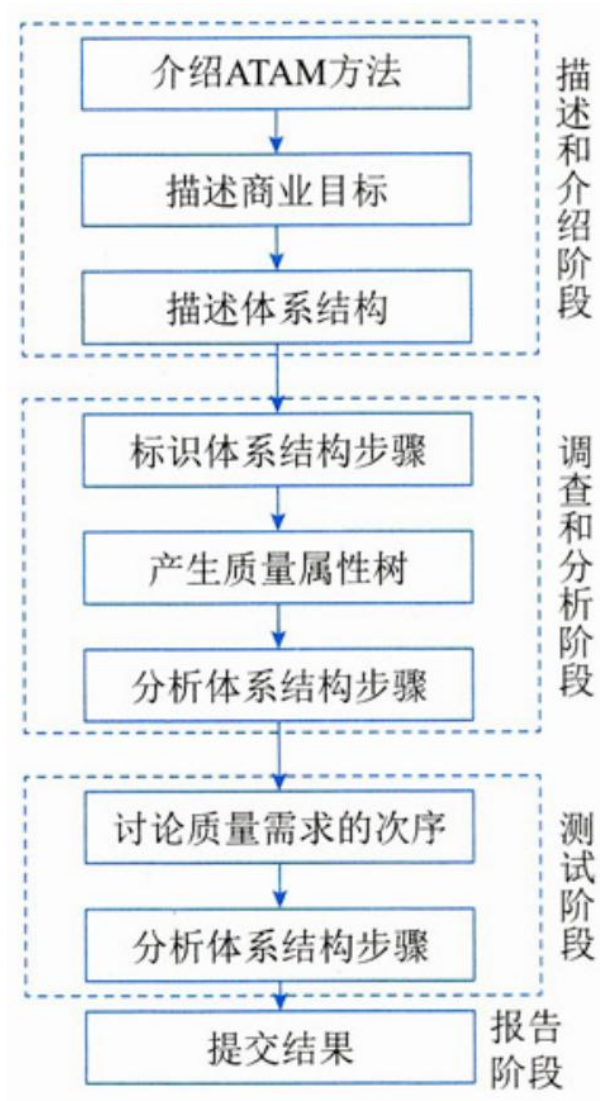
2

系统架构评估

3

ATAM方法架构评估实践

系统架构评估的评估方法-实践



◆ATAM方法评估软件体系结构，分为4个基本阶段：（★★★）

- ✓ 描述和介绍阶段(演示)
- ✓ 调查和分析阶段
- ✓ 测试阶段
- ✓ 报告阶段

系统架构评估的评估方法-实践

一、阶段1--演示

是ATAM 评估软件体系结构的初始阶段。此阶段有3个主要步骤。

第1步:介绍ATAM

涉及 ATAM 评估过程的描述。在此步骤中，评估负责人向所有相关参与者提供有关 ATAM过程的一般信息。领导者说明评估中使用的分析技术以及评估的预期结果。领导者解决小组成员的任何疑虑、期望或问题。

第2步:介绍业务驱动因素

在这一步中，提到了系统体系结构驱动程序的业务目标。这一步着重于系统的业务视角。它提供了有关系统功能、主要利益相关方、业务目标和系统其他限制的更多信息。主要利益相关方:最终用户、架构师、应用程序开发人员。

第3步:介绍要评估的体系结构

在这一步中，架构团队描述要评估的架构。它侧重于体系结构、时间可用性以及体系结构的质量要求。此步骤中的体系结构演示非常重要，因为它会影响分析的质量。这里涉及的关键问题包括技术约束，与正在评估的系统交互的其他系统，以及为满足质量属性而实施的架构方法。

系统架构评估的评估方法-实践

二、阶段2--调查和分析

在这个阶段，对评估期间需要重点关注的一些关键问题进行彻底调查。

这个阶段细分为3个步骤。

第4步:确定架构方法

这一步涉及能够理解系统关键需求的关键架构方法。在这一步中，架构团队解释架构的流程控制，并提供关于如何达成关键目标以及是否达到关键目标的适当解释。

第5步:生成质量属性效用树

在评估阶段，确定系统最重要的质量属性目标，并确定优先次序和完善。这一步至关重要，通过建立效用树，将所有利益相关方和评估人员的注意力集中在关系到体系成功至关重要的体系结构的不同方面。

第6步:分析体系结构方法

这是"调查和分析"阶段的最后一步。在这一步中，分析前一步生成的效用树的输出并进行彻底调查和分析，找出处理相应质量属性架构的方法。

这里还要确定每种架构方法的风险、非风险、敏感点和权衡点。

系统架构评估的评估方法-实践

三、阶段3--测试

第7步:头脑风暴和优先场景

这是 ATAM测试阶段的第一步。前者代表利益相关者的利益，用于理解质量属性要求。在效用树生成步骤中，主要结果是从架构师的角度来理解质量属性。在这一步中，目标是让更大的利益相关者参与其中。

第8步:分析架构方法

这是测试阶段的最后一步。在这一步中，我们分析上一步中高优先级的质量属性。找到了处理这些质量属性的架构设计方案，并检查相应的架构设计方案是否可支持满足这些属性。这一步重复“调查和分析”阶段的第6步。唯一的区别在于，在步骤6 中，高优先级质量属性来自效用树，而这一步需要考虑在头脑风暴投票中，高得票数的质量属性。

四、阶段4--报告ATAM

这是 ATAM 评估的最后阶段，其中提供了评估期间收集的所有信息。

ATAM 团队将他们的发现呈现给利益相关者。

ATAM团队的主要发现通常包括：

- ✓ 一种效用树；
- ✓ 一组生成的场景；
- ✓ 一组分析问题；
- ✓ 一套确定的风险和非风险；
- ✓ 确定的架构方法。

本章重点回顾

- 1、质量属性区分
- 2、质量属性效用树
- 3、ATAM架构评估方法

THANKS