

---

# 软件架构 (Software Architecture)

课时：5 天 深度：高级

## 课程描述

《软件架构》课程通过一个享有软件著作权的完整案例 -“岗位能力测评系统” (登记号 2009SRBJ3251)软件开发项目向**架构师 (Architect)**介绍了基于统一软件开发过程**(RUP)**有效运用 ISO/IEC 12207:2008 系统和软件工程 – 软件生命周期过程、**ISO/IEC/IEEE 42010:2011 系统和软件工程 - 架构描述**国际标准、由用例捕获的系统需求 (用例模型)、模型驱动开发**(MDD)**方法**构架 (Architecting)**模型驱动**架构 (MDA)**所必须的理(概)念和技巧。在这门含有讨论和实战的高端课程里,学员将学会如何贯穿软件项目生命周期进行构架,即构思、定义、表达、记录、交流、确认合适实施、维护、以及改进架构等活动。课程在一个用例驱动的、以架构为中心的以及迭代和增量的统一软件开发过程中,通过在整个软件项目生命期中充分运用架构描述语言(ADL)UML 和大量可重用的架构资产去捕获和交流软件架构决策,学员将掌握如何构架符合需求的大型的、复杂的和分布式的企业应用软件系统的软件架构以及如何基于架构框架使用**软件架构文档**记录软件架构。

## 课程目标

学完本课程后,学员将具备以下能力:

- 熟练掌握软件架构的起源、定义、范围、目的、风格、框架、描述、验证和评估等;
- 熟练掌握软件架构师的定义、职责、分类以及与系统/业务分析师、软件设计师和项目管理等角色的重要区别和内在联系等;
- 熟练掌握**创建逻辑架构**和**创建物理架构**的构架活动;
- 掌握设计出反映和平衡不同涉众需要的架构的方法;
- 掌握和涉众一起交流和论证架构满足涉众需要的方法;
- 使用架构框架和一组相关的视图文档化架构;
- 使用透视图确保架构满足系统的性能、伸缩性和安全等重要质量特性;
- 熟练掌握几种重要的架构风格和架构模式;
- 熟练掌握不同层级的架构机制,例如:分析机制、设计机制和实施机制;
- 熟练掌握 UML 模型驱动开发 (MDD) 过程;

## 课程内容

- |            |            |             |
|------------|------------|-------------|
| • 架构导论     | • 软件开发方法   | • 架构设计的基本原理 |
| • 需求       | • 可重用的架构资产 | • 架构实施      |
| • 架构描述语言   | • 架构分析     | • 架构描述      |
| • 架构集成开发环境 | • 用例分析     | • 架构评估      |

## 课程对象

此课程适合于有面向对象的软件系统开发经验的架构师、项目经理、系统分析师和设计师。

## 前提课程

- 统一建模语言(UML)和建模工具
- 统一软件开发过程
- 需求管理
- 面向对象的设计模式

## 后续课程

- Java EE 5 架构和程序设计
- 企业应用架构模式
- 软件项目管理

# 《软件架构》课程安排

此课程包括讲授和实验，共 5 天，具体内容安排如下表：

	上午	下午
第一天	<p>课程内容简介 软件架构导论 (1)</p> <p><b>核心目标：</b> 熟练掌握整个课程设计的核心思路。 深刻理解架构师、软件构架、软件架构各自的定义以及内在联系。</p>	<p>软件架构导论 (2) 架构沟通和描述实验 有关架构师工作经验的讨论题 实验——“岗位能力测评系统”案例介绍</p> <p><b>核心目标：</b> 熟练掌握整个课程的核心图像“<b>架构师构架架构</b>”的关键内容。熟练掌握交流架构的要点；</p>
第二天	<p>复习昨天所学内容 需求 架构描述语言 - UML 综述 架构集成开发环境 IBM Rational Software Architect for WebSphere (RSA4WS) V9.1</p> <p><b>核心目标：</b> 根据 ISO/IEC 29148:2011 系统和软件工程 - 需求工程、ISO/IEC 25010:2011 系统和软件工程 - 系统和软件质量要求和评价(SQuaRE) - 系统和软件质量模型等国际标准，熟练掌握分类软件需求的方法，为描述系统的功能、质量和约束打下基础； 熟练掌握 UML 中的类元和包的概念、4 种关系，这些是描述架构元素及</p>	<p>架构描述语言 - UML 元模型 软件开发方法 实验——架构决策过程模拟 实验——建立“岗位能力测评系统”的用例模型和领域模型</p> <p><b>核心目标：</b> 深刻理解 UML 的元模型，为描述和交流架构打下坚实的基础；基于 OMG 的 SPEM V2.2 标准理解几种常见的软件开发过程和方法，为软件架构设定项目语境和开发方法语境。 通过实战学会运用前面学到的知识；</p>

	其关系的基石；熟练掌握 UML 的包图、类图和交互图；	
第三天	<p>复习昨天所学内容 可重用的架构资产</p> <p><b>核心目标：</b> 理解从问题空间过渡到方案空间时，架构师可以重用的架构资产，这部分建立了描述方案空间架构的架构风格、架构模式、架构机制等核心概念。</p>	<p>架构描述 实验——创建“岗位能力测评系统”的<b>逻辑</b>架构</p> <p><b>核心目标：</b> 熟练掌握 ISO/IEC 42010 架构描述 V2 中介绍的架构框架和常用的 4 + 1 视图架构；熟练掌握软件架构文档的编写方法。</p>
第四天	<p>复习昨天所学内容 架构分析</p> <p><b>核心目标：</b> 结合前次实验结果，重新运用新学理论再一次迭代理解“岗位能力测评系统”的逻辑架构。</p>	<p>用例分析 实验——创建“岗位能力测评系统”的<b>物理</b>架构</p> <p><b>核心目标：</b> 掌握创建物理架构的方法。 掌握系统强壮性分析的系统方法。</p>
第五天	<p>复习昨天所学内容 架构设计的基本原理</p> <p><b>核心目标：</b> 深刻理解架构设计的基本原理，它是理解架构模式和设计模式核心的地方。 通过实战灵活运用常见的架构风格，特别是分层架构风格。</p>	<p>架构实施 实验——“岗位能力测评系统”的完整演示 课程总结和答疑</p> <p><b>核心目标：</b> 掌握架构实施中的关键注意事项。 通过使用真实的系统来重新体会所学知识。</p>