



第6章 软件体系结构编档



内容

- ■ **6.1 概述**
- 6.2 视图编档
- 6.3 制作文档包
- 6.4 架构编档案例



软件架构编档概述

- 软件架构视图文档
 - 描述主要构件和视图关系的**主要表示**
 - 说明和定义视图所展示的**构件**并列出其属性的**构件目录**
 - 构件的**接口**和行为规范
 - 对所有针对架构进行剪裁的现有内部机制进行说明的**可
变性指南**
 - **基本原理**和设计信息
- 软件架构**视图之外的信息**
 - 对**整个文档包**的介绍
 - 描述**视图之间关系**
 - 整体架构的**限制**和**基本原理**。



选择需要编档的视图

- 视图是一个架构文档的核心，至于创建哪些视图，需要根据设计过程中的编档决策来决定

编档需求一览表

视图 涉众	逻辑视图	开发视图	运行视图	部署视图	用例视图
项目经理	s	d		d	s
客户	o		s	d	d
最终用户	d		o	s	
架构师	d	d	d	d	d
开发人员	o	d	d	o	d
测试人员	s	d	o	s	s
集成人员	s	d	o	s	o
维护人员	s	d	d	s	s
分析人员	d	d	d	d	s
支持人员	s	s		d	o

图例：d=详细信息，s=某些细节，o=概括性信息



内容

- 6.1 概述
- ■ 6.2 视图编档
- 6.3 制作文档包
- 6.4 架构编档案例



一个视图包的标准结构

视图包结构

1主表示

2构件目录

A 构件和属性

B 关系和属性

C 构件接口

D 构件行为

3上下文图

4可变性指南

5原理



1、主表示

- 主表示能展示构件以及构件之间的关系
- 应该包含一些**主要构件和关系**（某些情况下无法包含所有这些构件和关系）
- 主表示通常采用**图形表示法**



2.构件目录

构件目录至少应该详细说明那些在主表示中描述的构件，可能还有其他一些构件

- (1) **构件及其特性**。这部分将命名视图包中的每一个构件，并列出它们的特性。
- (2) **关系**及其特性。
- (3) **构件接口**。接口就是构件间交互或通信的边界
- (4) **构件行为**。有些构件会与环境进行复杂的交互。为了理解或分析这一情况，架构师通常有责任规定构件行为。

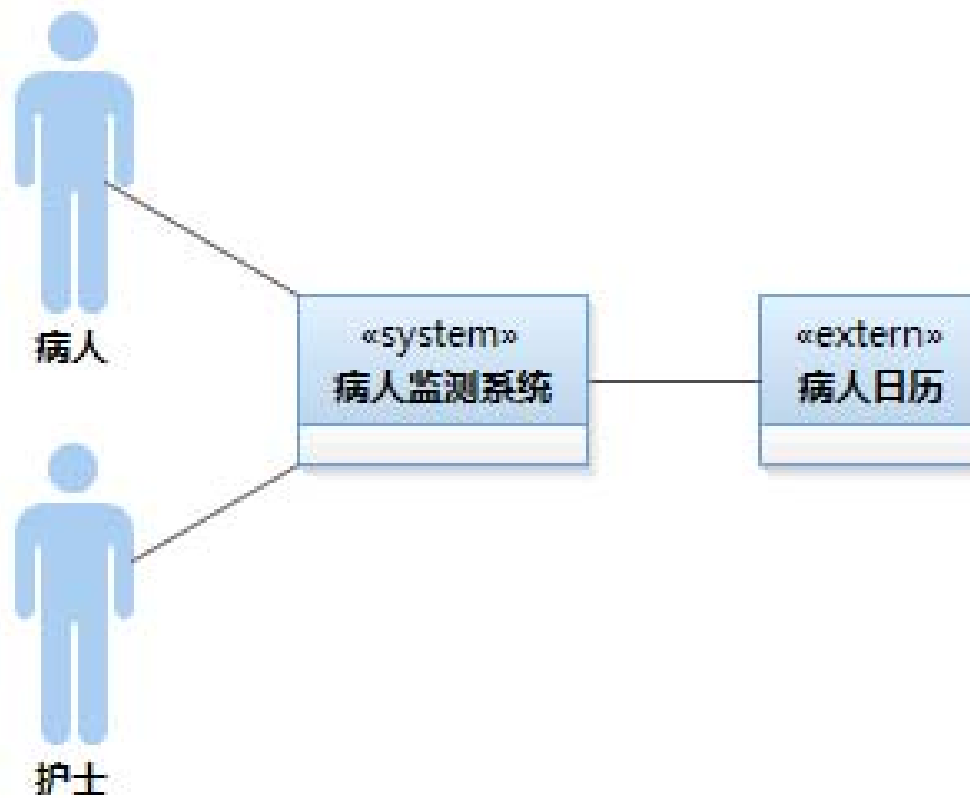


3.上下文图

- 上下文图用来展示视图中所描述的系统（或系统的一部分）与其环境的关系。上下文图一般包括：
 - 对**实体**的描述，如**构件**或**系统**
 - **数据源**和**数据目的地**，或由实体处理和产生的**触发**或**命令**，它们均出现在当前被描述实体符号的外部
 - 图例
 - 必须和当前编档实体进行交互的**其他实体**
- 上下文图的表示法可以分为非正式表示法和UML表示法。



上下文图举例





4.可变性指南

- 可变性是以**预先计划的方式**快速实现变化的能力
- 架构师往往通过在架构中设计**可变点**，来实现这种可变性。
- **可变点是架构中的某个位置**，在这一位置中，某个特定的决策已被限制为若干可选方案，但某个特定系统最终采用的那个**方案尚未确定**。
- 可以在可变点出现的地方，如主表示、构件目录、行为描述、接口描述等，对可变点进行编档。



6.原理

- 对**架构原理编档**，将说明架构师为什么会做出视图包中反映的设计决策
- 此外，**将没被采用的设计方案列一个清单**，还说明为何不采用这些方案，并解释其原因，**为什么在本视图使用该模式或风格是合理的**，这些信息能防止未来继任架构师在面对必要的变更时误入歧途



内容

- 6.1 概述
- 6.2 视图编档
- ■ 6.3 制作文档包
- 6.4 架构编档案例



软件架构文档模板

■ (1) ISO/IEC/IEEE 42010:2011 架构描述模板

第1章 引言

第2章 涉众和关注

第3章 视点+

第4章 视图+

第5章 一致性和通讯

附录（可选）架构决策和原理

ISO/IEC 42010:2007 (IEEE-Std-1471-2000)中规定：视点是一个有关单个视图的规格说明。视图是基于某一视点对整个系统的一种表达。



软件架构文档模板

■ (2) SEI “View and Beyond”架构文档模板

第1章 文档指南
第2章 架构背景
第3章 视图
第4章 视图之间关系
第5章 参考资料
第6章 字典
第7章 图表
附录



软件架构编档基本原则

通过以上两种软件体系结构文档标准的对比，可以发现它们都遵循了架构编档的基本原则：

- （1）涉众及其关注形成不同的**关注点**，**关注点是选择视图的依据**。关注点编档是架构编档的重要组成部分。但在实践当中，由于一些常见视图（如逻辑视图、开发视图）对应的关注点显而易见，这部分内容往往又会省略。
- （2）架构编档是编档**相关视图**和**视图之间的关系**，两种编档标准都有相关章节体现了编档视图信息。
- （3）添加了适用于多视图编档的视图外信息，包括架构**文档的组织**，**系统概要**，**术语和缩略语表**，以及**架构决策**和**基本原理**等。



1.文档指南

Documentation Roadmap

- (1) 文档管理和配置控制信息
- (2) 软件架构文档的目的和范围
- (3) 软件架构文档是如何组织的
- (4) 涉众代表
- (5) 视图是如何编档的
- (6) 和其他软件架构文档的关系



2.视点定义

Viewpoint Definition

- 摘要：视点的简要概述
- 涉众和它们强调的关注点
- 元素、关系、属性和约束：本节定义了元素的类型，它们之间的关系，它们表现出的重要属性，并它们遵守的约束
- 建模/表达符合视图的语言：本节列出了一种或者多种语言，用于建模或表达符合这一视图的视图，并引用定义文档
- 适用的评估/分析技术和一致性/完整性标准
- 视点来源：本节提供了这个视点定义来源的一个引用



3. 架构背景

Architecture Background

- (1) 系统背景
 - 系统概览
 - 目标和背景
 - 重要的需求驱动
- (2) 解决方案背景
 - **架构方法**:它描述了应用到软件体系结构的所有设计方法, 包括**使用的体系结构风格或设计模式**, 而这些方法的范围超越了任何单一的架构视图
 - 分析结果:本节描述所有定量或定性分析的结果已经被执行, 这些证据证明软件体系结构是适用的
 - 需求覆盖率



4.视图之间关系

- 一般说来，一个视图中元素的部分可以映射到另一个视图中元素的部分。
- 帮助读者理解视图间的关系能够使他洞察架构是如何作为一个统一的概念整体来发挥作用的。
- 可以通过提供视图间的映射来弄清视图间的关系，使读者加深对架构的理解、减少混淆。



6. 字典 Directory

- 元素列表(Index)
- 术语表(Glossary)
- 缩略语(abbreviation)



内容

- 6.1 概述
- 6.2 视图编档
- 6.3 制作文档包
- ■ 6.4 架构编档案例



案例 1

Software Architecture Document Online Catering Service 1.0

- Developing a J2EE Architecture with Rational Software Architect Using the Rational Unified Process
- https://www.ibm.com/developerworks/rational/library/05/0816_Louis/index.html



案例 2

Adventure Builder - Software Architecture Document (SAD)

- https://wiki.sei.cmu.edu/sad/index.php/The_Adventure_Builder_SAD
- https://wiki.sei.cmu.edu/sad/index.php/Reference_Documentation



案例 3

UbiEyes RTLS 3.0软件架构设计文档 (见教材)