

# 整理

## 第一章

### 互联网时代软件特点

- 业务与信息紧密融合
- 规模大、并发性高
- 周期短、更新快
- 开放、普适环境

### 互联网时代软件工程思想

软件工程方法从四方面体现：

需求获取、软件形态、开发过程、产品验证

覆盖全生命周期的需求动态获取及持续交付

需求主体的变化：确定的个体组织-->不固定的个体或群体

需求动态获取：静态的系统需求规约-->动态

需求满足：一次性测试提交-->全生命周期的多次迭代和持续交付

基于流程的网络服务动态集成交互

软件构件、组装方式、运行环境

基于敏捷和增量迭代的协同软件开发过程

过程推进：稳扎稳打、逐步推进-->完成比完美重要

开发方式：文档功能全部齐全-->小步快跑、增量迭代

团队协作：指责分工明确，接口清晰-->一专多能，协同工作

用户体验为中心的产品演进模式

验证目标：功能满足为核心-->客户体验为中心

验证内容：以软件上线的测试目标-->上线后的软件持续治理

验证方式：需求的覆盖性测试-->设计模型的一致性验证

### 软件工程方法框架——管理维度

从项目管理维度来说，互联网下的软件工程方法核心是群体协同开发。体现为提交的反复性，工作的协同性，过程推进的敏捷性。基于敏捷和增量迭代的项目管理成为核心特征。因此，Scrum+DevOps结合的软件项目管理模式逐渐成为主流

敏捷软件开发模式缩小了业务领域的需求与开发之间的第一个隔阂，而敏捷软件运维模式DevOps的出现缩小了开发与运营之间的第二个隔阂。

---

## 第二章

### 软件开发模式的特点和区别

三种开发模式：

- 定制开发模式
- 套件实施模式
- 模型驱动架构模式

开发模式	特点	优势	劣势
定制开发模式	含有高事务量或有特殊业务需求，不考虑时间成本	可购买大量经验和技术	人员、周期、费用需求高
套件实施模式	基于商业化平台通过业务模型配置搭建	周期短、成本低、质量稳定	特别需求难以完全满足
模型驱动架构模式	在构建的业务模型基础上，通过模型驱动力的转换，实现软件的架构	介于前两者之间，有一定灵活性又能保证软件质量	

### 单体到微服务架构变化

单体架构	微服务架构
整体部署	拆分部署
紧耦合	松耦合
基于整个系统的扩展	基于独立服务，按需扩展
集中式管理	分布式管理
应用无依赖关系管理	微服务间较强的依赖关系管理
局部修改，整体更新	局部修改，局部更新
故障全局性	故障隔离，非全局
代码不易理解，难维护	代码易于理解维护
开发效率低	开发效率搞
资源利用率低	资源利用率高
重，慢	轻，快

微服务架构的问题：

- 开发人员技能要求高
  - 接口的数目和匹配问题
  - 分布式系统复杂性
  - 可测性的挑战
  - 异步机制
  - 代码重复
-

# 第三章

## 业务流程管理生命周期四个阶段

四个阶段：

- 建模
- 装配
- 部署
- 管理

## 企业业务建模方法和发展，划分方式

基于业务建模的原子要素，可以划分以下建模方法：

- 基于功能
- 基于流程
  - 基于数据的业务流程建模（以数据流图为主，将活动按照数据流向连接）
  - 基于活动的业务流程建模（以活动图为主，将业务活动通过时序关系连接）
  - 基于事件的业务流程建模（以EPC为主，将业务活动通过消息事件连接）
  - 基于状态的业务流程建模（以PetriNet为主，将业务活动通过状态事件连接）
- 多视图建模（集成化建模）
- 基于服务
- 基于大数据

方法	主要组件	连接方式	主要代表模型
基于数据的业务流程建模	数据实体、数据活动	数据流	数据流图
基于活动的业务流程建模	时序活动	系列流	活动图
基于事件的业务流程建模	功能活动、事件	事件流	EPC
基于状态的业务流程建模	系统状态、状态转移事件	状态变迁	PetriNet

## 以过程为核心的

多视图建模：以过程模型为核心，其它视图为辅助

- 功能视图
- 数据视图
- 组织视图
- 控制视图

## 典型业务框架比较

		Zachman	CIM-OSA	PERA	ARIS
	维数	二维	三维	二维	三维
	覆盖的视图	完全	基本完全	不太完全	基本完全
结构	视图间的联系	松散	松散	松散	紧密
	覆盖的阶段	建立期	建立期	整个生命周期	建立期

	开放性	较强	强	弱	一般
应用	参考模型	多	较多	少	多
	可操作性	强	较弱	较弱	非常强

---

## 第四章

### 建模方法

- 基于任务的过程建模方法
- 基于数据的过程建模方法
- 基于状态的过程建模方法
- 基于消息事件的过程建模方法

---

## 第五章

### 企业资源管理思想

企业竞争力要素（TQCS）就是目标：

- 提供产品的时间（time）
- 产品的质量（quality）
- 产品的成本（cost）
- 产品的服务（service）

精益生产思想，定义为五个步骤的流程：

- 定义顾客的价值
- 定义价值流程
- 建立连续的作业流程
- 拉动式生产
- 持续改进

*lean production*的核心是减少浪费

### 四个生产类型

- 订货生产（Make To Order-MTO）
  - 绝大部分产品
- 备货生产（Make To Stock-MTS）
  - 家用电器、日常生活用品等
- 订货组装（Assemble To Order-ATO）
  - 办公家具、房屋门窗、计算机等
- 工程生产（Engineer To Order-ETO）
  - 船舶、飞机等

### 库存管理策略

独立需求：

- 定量库存控制模型：当库存数量下降到某个库存值时，进行补充库存，以此来保证库存供应
- 定期库存控制模型：按一定周期T检查库存，并进行库存补充，补充到规定的库存量S

相关需求：

- MRP库存控制方法

固定订货周期法使用范围：

- 消费金额高，需要实施严密管理的重要物品；
- 需要经常调整生产或采购数量的物品；
- 需求量变动幅度大且变动具有周期性的物品；
- 多种商品采购可以节省费用的情况；
- 需要定期知道的物品；
- 同品种物品分数保管、同品种物品向多家供货商订购、批量订货分批入库 等订货、保管、入库不规则物品等。

固定订货量系统的优点：

- 管理简便，订购时间和订购量不受人为判断的影响，保证库存管理的准确性；
- 由于订购量一定，便于安排仓库内的作业活动，节约理货费用；
- 便于按经济订购批量订购，节约库存总成本。

缺点：

- 不便于对库存进行严格的管理；
- 订购之前的各项计划比较复杂。

**RCCP和CRP比较**

项目	RCCP	CRP
计划阶段	MPS	MRP
计划对象	关键工作中心	全部工作中心
负荷计算	独立需求物料	相关需求物料
工作日历	工厂日历、关键工作中心日历	加工中心日历
提前期	以计划周期为最小单位	物料开始与完工时间

第六章

工作流建模概念

工作流是实现流程的组织管理和流程优化工作的最有效工具

按业务流程之间的协作方式可以分为：

- 单工作流模式
- 多工作流模式

流程定义

流程定义由以下模型组成：

- 流程模型（Process）： 描述了案例的执行方式

- 案例 (Case): 工作流的处理单元;
- 任务 (Task): 工作的逻辑单元
- 路由 (Router): 根据流程决定的任务执行的分支及次序
- 触发器 (Trigger): 工作项执行的触发条件

## 资源分类及管理 (结论)

资源分类是为资源的区分和选择:

- 指明资源应该归属的资源类, 分配原则同时指明角色和组织单元, 符合要求的资源必须属于这两个资源类的交集
- 职能分离的考虑
- 资源分配还可能依赖于一些特殊案例的属性(Attribute)

## 资源分配

总体来说, 资源分配包括了基本策略, 推拉模式选择, 排队原则确定, 以及资源选择策略等方面。

推拉模式:

- “推式驱动”: workflow引擎把工作项和资源进行匹配。通过预设定的条件, workflow引擎能够选择每个工作项由哪个资源执行, 资源本身不能做出选择。
- “拉式驱动”: 资源把工作项和自身进行匹配。资源考察它能够执行的工作项, 并从中选择一个。

一般采取介于推动和拉动之间的方法, 采用拉动的原则。

资源选择策略——如果一个工作项可以被多个资源执行, 那么就需要考虑以下事项:

- 让资源发挥自己的专长。
- 让一个资源连续做类似的任务。
- 为将来尽可能预留弹性资源。

## 流程检测 (指出是哪种类型的问题)

- 死锁
- 活锁 (死循环)
- 确定性 (最终指向一个唯一状态)
- 无终止
- 可达性 (可达图)
- 线性时序逻辑

## 合理性定义

每个合理 (Sound) 过程都必须满足以下要求:

- 流程不包含不必要的任务
- 每个提交的案例必须能够被完全完成。
- 完成后再没有被案例的引用。

## 性能分析方法

工作流的性能分析主要有以下几种方法:

- 马尔可夫分析方法
  - 优点: 对给定流程可自动构造, 将成本和时间要素引入可获得更多性能指标算法

- 缺点：并不是所有方面性能都能被分析出来，分析处理过程特别复杂非常耗时
- 排队论
  - 涉及到一个拥有大量的*相同资源*的*队列*的处理过程，可用排队论
  - 缺点：排队论中的很多假设对工作流程并不合适，特别在并行路由存在的情况下，排队论方法是不能直接应用的（计算值和真实值有波动）
- 计算机仿真方法
  - 优点：提供了可视化的流程执行过程展示，很容易为没有数学背景的人接受和理解
  - 缺点：模型的建立和分析非常耗时，仿真结果的深入处理也需要较为的概率统计知识背景

## 排队系统的Kendall表示

A/S/m/B/K/SD:

- A：顾客相继到达时间间隔的概率分布（通常用M泊松或负指数，D定长）
- S：服务时间分布（同A）
- m：服务台个数
- B：系统中顾客容量限额，或称等待空间容量（B=0，损失制不允许等待；B=无穷，等待制）
- K：顾客源限制（有限或无限）
- SD：服务规则（如FCFS，LCFS）

## 排队系统流程指标

- 平均WIP（队列长度）：等待服务顾客和接受服务的顾客数量之和
- 案例完成时间（平均流动时间，等待时间）
- 资源利用率（工作负荷，关键设备占用情况）：服务员服务强度
- 单位时间处理案例数量
- 预定标准时间完成案例的百分比
- 周转量和回退的数量

## 排队论计算过程

设单位时间内，有 $\lambda$ 个新案例到达，需要被1个资源处理。这个资源单位时间内能够完成 $\mu$ 个案例。

那么这个资源的能力利用率 $\rho$ ：

$$\rho = \lambda/\mu$$

假设处理时间和案例到达时间间隔都服从负指数分布，流程中的平均案例数量L:(WIP)

$$L = \rho/(1 - \rho)$$

平均等待时间W:

$$W = L/\mu = \rho/(\mu - \lambda)$$

平均系统时间S:

$$S = W + 1/\mu = 1/(\mu - \lambda)$$

## 流程优化方法

6种流程：

- Sequential
- Parallel：处理时间不变，等待时间缩短

- Composition：处理时间缩短，等待时间缩短
- Flexibilization：处理时间不变，等待时间缩短
- Triage（反而变慢）：处理时间不变，等待时间增加
- Priorization：处理时间不变，等待时间缩短

业务执行标准

流程标准	XPDL	BPEL	BPMN
维护组织	WfMC	OASIS	OMG
核心目标	数据共享及交换	服务自动化编排	业务流程设计到流程开发
针对用户	运维者	IT编程人员	业务人员
特点	缺少状态	无人工活动	多种交互方式
适用范围	简单的活动关系，如办公审批流程	程序的互操作实现	业务定义到组件映射
标准特点	大而全	可读性较弱	定义完备
针对生命周期阶段	设计到执行期	动态执行期	执行期