



凤凰公司 SOA 应用策划

Phoenix Corp. SOA Application Proposal

["IBM 杯" 高校 SOA 创新应用大赛]

[51Start 小组]

史广顺
卞晓青
杨巨峰
宫大伟
张 维



目 录

第一篇	前 言	4
第一章	SOA 解析与应用	4
1.1	SOA 逐本溯源	4
1.2	SOA 条分缕析	5
1.3	应用环境解析	7
1.4	SOA 落地切入点解析	8
1.5	项目设计中 SOA 的应用	9
第二章	思维与理论创新	10
2.1	核心创新点与立足点	10
2.2	分析与设计中的创新	11
第二篇	企业业务模型分析	14
第三章	业务模型分析与展望	14
第四章	业务模型对 IT 系统的挑战	16
4.1	SOA=VOA：面向商业价值的系统架构	16
4.2	大赛题目解析：超越系统的集成	18
第三篇	服务模型分析和设计	20
第五章	服务建模概念解析	20
5.1	Service：SOA 的理念核心	20
5.2	服务建模的过程与价值分析	22
第六章	凤凰公司服务模型分析设计	23
6.1	服务模型的原始定义	23
6.2	系统架构的层次分解	25
第七章	服务规约与技术实现	25
7.1	服务规约的作用和价值	25
7.2	服务实现技术方法浅析	27
第四篇	系统架构设计	29

第八章	用户需求分析.....	29
8.1	需求概述.....	29
8.2	凤凰公司业务环境描述.....	29
8.3	凤凰公司 IT 环境描述.....	34
第九章	用例模型分析.....	36
9.1	参与者列表.....	36
9.2	用例列表和规约.....	36
9.3	用例模型图.....	39
第十章	数据模型分析.....	39
10.1	数据汇总.....	39
10.2	数据模型构建.....	40
第十一章	关键技术架构决策.....	41
11.1	服务组件划分.....	41
11.2	ESB 设计.....	43
11.3	智能社区服务整合.....	44
第十二章	系统架构分析.....	45
12.1	系统结构图.....	45
第五篇	组件设计	48
第十三章	组件解析.....	48
13.1	什么是组件.....	48
13.2	组件的类型.....	48
13.3	组件定义方法.....	48
第十四章	组件设计原则与方法.....	49
14.1	组件层次化原则.....	49
14.2	粗组件粒度原则.....	49
14.3	定义合理组件范围原则.....	49
14.4	无状态组件原则.....	50
14.5	包装现有应用的方法.....	50

第十五章	组件设计	50
15.1	ERP 数据访问组件	51
15.2	CRM 数据访问组件	52
15.3	客户信息管理组件	52
15.4	业务机会管理组件	53
15.5	产品信息查询组件	53
15.6	销售订单管理组件	53
15.7	手机短信组件	54
15.8	日程表组件	54
15.9	日志组件	54
15.10	服务发现组件	55
第六篇	项目实施计划	56
第十六章	项目团队组成及分工	56
16.1	项目团队简介	56
16.2	项目团队组成	57
16.3	项目团队分工	57
第十七章	项目实施的环境及平台	58
第十八章	项目任务实施时间计划	59
第十九章	项目风险分析及规避方案	60
19.1	SOA 理解不深, 项目设计存在偏差	60
19.2	技术实现平台掌握不熟练, 影响项目开发进度	60
19.3	实施过程表明技术实施工作量很大, 进度可能需要拖延	60
19.4	技术人员和测试人员的主观失误	60
第二十章	项目验证和测试草案	61
20.1	项目验证工作详细说明	61
20.2	内部评测工作详细说明	61
第七篇	结语: 超越 SOA	63

第一篇 前言

第一章 SOA 解析与应用

1.1 SOA 逐本溯源

SOA 最初由世界著名的咨询公司 Gartner 在 1996 年提出，两位作者 Roy W. Schulte 和 Yefim V. Natis 在其具有前瞻性的文章《“Service Oriented” Architectures》中首次提出了面向服务架构的概念，在这篇文章中第一次给出了 SOA 的形象化定义：

SOA: A service-oriented architecture is a style of multitier computing that helps organizations share logic and data among multiple applications and usage modes.

在此之后 SOA 本身在长时间内并没有取得长足的发展，究其原因主要有以下两点：

第一、需求推动不足

任何理论上、技术上的概念或应用如果真的想取得实质性的发展，必须有客户或市场需求作为原始推动力，在当时的历史条件下企业信息化进程还没有达到瓶颈，包括 EAI 在内的各种技术还广泛应用于企业的各个信息处理流程及其整合之中，而电子政务化进程出于保密等因素考虑可以说还只是停留在所谓的桌面上而已；

第二、实现技术悬空

SOA 本身只是一种信息系统设计的基本组织原则（Principle），虽然呈现给了我们一幅组织松耦合、组件服务化、接口标准化的美好蓝图，但没有人能告诉我们究竟应该怎样实现 SOA，以趋利性为主要特征的企业当然不会在这样一个还仅仅是概念的新名词上花费太多的精力，他们需要的是能带来实际利益的变更而不是炒作。

不过任何金黄色的东西都会闪光的，更何况是真正的金子，随着时间的不断推移，SOA 越来越快地浮现在人们的视野中。企业信息化困境随着经济全球化的发展而逐渐凹陷出来，跨企业、跨部门、跨行业、跨国界的企业组织形式，高速、准确、复杂、多变的业务信息化标准，对企业信息化提出了能够“随需而变”地高效提升企业竞争力及价值的要求，这样使原有的复杂、僵化、零散、独立的信息系统整合成为了必然；同时随着 IBM、BEA、Oracle、SAP、Sybase、Sun、Microsoft 等多家跨国的 IT 行业旗舰厂商的不懈努力，各种各样的符合 SOA 原则的实现技术和标准也陆续出现并不断成熟和发展，其中 Web Services 作为目前相对最成熟完备的服务封装、传输、加密等一系列的接口标准集得到了越来越多厂商和行业从

业人员的推崇，SOAP、WSDL、UDDI 等名词也逐步映入人们的眼帘。

概括的讲，我们认为 SOA 就是一个原则或者讲是一种框架，它用于将业务流程和基础技术结构标准化，并且对松散耦合的可重用服务组件进行集成，它可以对服务进行组合、重用和调整以满足不断变化的业务优先级。



图 1 SOA 是衔接商务问题与 IT 解决方案的桥梁

总而言之，SOA，实际上是一座桥梁（如图 1 所示），衔接了业务管理与技术实现之间的万丈沟渠。从某种意义上讲，让 SOA 正常工作在很大程度上不是取决于技术问题，而是业务控制流和 IT 控制流最优分解、匹配的问题。了解了 SOA 这么多的故事，那么让我们再来看看 SOA 究竟拥有什么神奇的魔力吧。

1.2 SOA 条分缕析

先来看看什么是 SOA：SOA 是能够通过松散耦合关系组织交互软件模块（服务）的软件架构。那么什么是 Service 呢：Service 是服务提供者提供的能完成请求服务的用户所希望功能的处理集合。SOA 具有以下关键特点：

第一、 独立运行（standalone）

所谓的 service，它与组件（component）的根本不同首先在于 service 是独立于调用者自行运转的，即访问 service 的接口相对狭窄，我们只需要知道 service 如何满足我们的功能需求，而不需要管理它的生命周期，不需要理会那些维持 service 运行所需要考虑的种种细节。即我们对于 service 的了解只需要局限于功能接口即可，不需要理会它的那些管理接口，配置接口等。软件的发展趋势是智能化

（intelligence），而智能化的第一步是自治（autonomous），即软件要具有自己的独立性，这首先意味着软件的生命周期要脱离其他软件的控制（standalone）。服务无论调用者是否存在，它本身是独立存在，独立发展的。这与一般的对象是不同的，一般的对象是由调用者创建并控制其生命周期的。

第二、 异步调用（asynchronous）

异步特性是 SOA 包容真正的商业智能的关键所在，同样是一个函数调用，只有异步特性才能够包容真正的商业智能，才能真正模拟真实世界中各种各样的异步情

况，才能在函数这种最小的程序结构单元中引入最复杂的处理过程。现在 SOA 的宣传往往集中于机器之间的互相调用，强调异构系统之间的一种包容性，但事实上异步特性所能承载的内容要远远超越机器世界本身。当然，同步调用方式也是 SOA 架构的自然组成部分，就像我们既需要 email，也需要手机一样。

第三、 基于消息（message based）：

基于消息的调用方式是分布式系统的一种内在要求，消息是一种数据，它并不是远程对象指针，不需要维持其状态并将状态通知所有需要知道的调用者，远程对象这种基于 proxy-stub 的方案其要求是远程状态同步，即状态的一致性，而分布式系统是由众多独立的状态空间（进程空间）所构成的，这种内在的不协调是造成分布式的对象方案难以实现可扩展性的关键原因。

第四、 纯文本协议+元数据（Plaintext Meta）：

SOA 架构中基于纯文本协议是一个非常关键的技术抉择，当需要跨越众多硬件平台和软件系统的时候，各种二进制的远程调用方案总是存在着难以彻底解决的可理解性的问题。凭借纯文本的结构透明性加上元数据的自描述特性，我们希望实现一种语义透明性，使得各种中间层都能够以通用的方式传递经过的消息，并理解其中需要理解的部分。与 OOP 中的传统模式不同的是，SOA 架构中强调的是结构（structure）与内容（content）的分离，而不是数据与行为的耦合与封装。说到纯文本的元语言，XML 无疑是最好的选择，作为一种半结构化的文本表述，XML 天然就是人机共享的信道。

在 SOA 架构中，松散耦合是其最核心的特点，是延迟绑定（late binding），独立运行（standalone）和基于消息（message based）等多种技术策略综合作用之后所达到的一种效果，这为外部灵活的流程配置做好了铺垫。

SOA 并不能等同于 Web Service: Web Service 是一套技术体系标准集，包括 XML、SOAP、WSDL 和 UDDI，可以用来建立应用解决方案，解决特定的消息通信和应用集成问题。随着时间的推移，我们发现这些技术在不断发展、不断成熟，也会更好地帮助你实现 SOA。目前来讲它是实现 SOA 的最完善和优秀的解决方案，也最为大众所接受和了解。但是，Web 服务并不是 SOA。

SOA 是一种软件架构原则，而不局限于某个技术的组合（例如 Web Service），它超越了技术范畴。在一个商业环境中，纯粹的 SOA 是一种应用软件架构，其中所有的功能都是相互独立的服务模块，通过完备定义的接口相互联系起来。只要按照一定的顺序来请求这些功能模块所提供的服务，就可以形成完整的业务流程。所以 SOA 架构的核心体现在如何分解、协调服务，而不是如何规约、调用服务，理解这些才能真正理解 SOA

SOA 的优势是不言而喻的，从企业管理者的角度来讲，SOA 最大限度的保护了现有资产、拥有持续良好的可改进业务过程、减少成本和风险等等；从信息系统部署者的角度来讲，SOA 具有最大限度的维护了现有异构系统功能、良好的分解服务功能、可重用的信息系统组织模块等功能；从用户的角度来讲，SOA 能更快更好地满足他们多变的需求、使用起来更加简便。其明显的优势让人眼花缭乱、垂涎欲滴，现在让我们结合题目解析这个虚拟的现实环境。

1.3 应用环境解析

凤凰公司欲将原有的 ERP 系统与新引进的 On Demand CRM 系统(以下简称 CRM 系统)相结合，在提高销售与财务人员工作效率的前提下提升企业业务竞争力。这个题目本身只是现实应用中企业信息化系统集成的很小一部分，但足以让我们见微知著。

既然是系统整合就是为了要提升企业的核心竞争力进而获得更大更快的价值转化速度，所以在这里我们将重点放在目前信息系统无法解决的问题上，而不去关心那些现有系统已经可以解决的问题。我们认为需要解决以下重要问题才能满足大赛题目的要求：

第一、 用户操作的集成难题

ERP 系统与 CRM 系统在原有相互孤立的情况下，销售人员需要同时面对两套系统，最要命的问题不是系统使用习惯和熟练度的问题而是销售人员要随时负责自己同步两个系统中的信息，这样将极大地降低销售人员的工作效率，使公司在付出成倍的人力资源成本之后没有任何良好的收益；

第二、 价值创造的效率提升

ERP 系统与 CRM 系统相互孤立导致整个企业的价值链存在需要人为填补的断链，这种情况的存在使得企业的价值实现与增值过程相对缓慢甚至会出现丢失大量本来可以实现为企业价值的业务机会；

第三、 业务信息的同步更新

ERP 系统与 CRM 系统相互孤立导致无法从 CRM 系统中获取大量客户信息，无法在数据挖掘的基础之上进行客户分析，从而指导 ERP 系统中的生产管理部门进行某些预防性生产管理活动，更没有办法直接为公司决策层提供良好的决策支持，当然也就更提不上按需生产销售了；

第四、 企业管理的优化升级

企业业务变更困难，没有更细粒度的组件服务封装与重新设计的可变流程拼装，使

得企业可以承载的业务领域僵化单一，无法以最快的速度转移到“蓝海”之中，或深陷“红海”不能自拔或眼见“蓝海”却无能为力，企业业务领域的创新是一个企业生存和壮大的保证，不能让信息系统本身成为限制企业发展的瓶颈而应该想法设法将其转化为动力；

实际中的问题还有很多很多，如何解决这些问题呢？当然就是前面提到的 SOA 了，可是说归说，做起来就是另外一回事了，如何才能在 SOA “落地”的过程中既能“掷地有声”又能实现“软着陆”呢？

1.4 SOA 落地切入点解析

SOA 作为一种系统架构概念和观点，最终必然走向“落地”应用，即实际部署实施于现实企业系统之中。在 2006 年 IBM WebSphereLive!产品与方案大会上，IBM 软件集团 SOA 架构全球副总裁 Dan Power 先生指出：“SOA 在任何公司都扮演着战略性角色，因为它能帮助企业识别新的商机、进行创新并不断满足客户的需求。SOA 的成功首先取决于如何顺利‘进入’SOA，避免更多的开支，以及确保企业的 SOA 投资符合其发展战略，并能够经受住市场的波动与公司的变革。”

在随后的发言中，Power 先生详细指出：根据全球最佳实施经验、第三方研究结果、以及 IBM 与各行业、各种规模的企业开展的 1,800 余项 SOA 合作的心得，IBM 发现并指出了 5 个可以帮助客户更加容易地着手实施一个 SOA 项目的切入点。这些切入点包括：以人员(People)、流程(process)、信息(Information)为中心的方法、以及系统连接性(connectivity)和重用现有资产的能力(creating and reusing services)。

我们认为，人员是企业业务的执行者、流程是企业业务的架构者、信息是企业业务的支持者。这三者理应也必须成为 SOA 实际部署过程中值得关注的切入点。

以人员为中心的 SOA 切入点，能够为业务执行者们提供整合的综合信息，便于执行者们做出更加良好的决策和更加高效的执行，无论从决策力还是执行力角度来讲，SOA 以人员为切入点都切实符合提高企业效率的目的；

以流程为中心的 SOA 切入点，提供了一种借助重新利用和优化流程，快速部署创新的业务模式。在现代经济社会的企业竞争中，“蓝海”越来越成为企业角逐和竞争的焦点，如何让作为企业业务架构者的流程能够快速适应不断变化的业务需求是 SOA 要解决的核心问题，自然以它为切入点便于我们更加良好的架构随需而动的企业流程模型；

以信息为中心的 SOA 切入点，以嵌入式（in-line）或现场（in-context）的方法提供可靠信息服务，提高企业业务洞察力，从而降低企业运营风险。信息是企业业务的支持者，任何企业业务必将依据大量现实信息，通过对历史信息进行数据挖掘的从而提供决策支持的基础之上，决定企业业务行为，并产生新的信息供其它企业部门决策或执行之用，所以以信息为切入点实际上是希望 SOA 能够为企业决策、执行提供及时、准确、充足的信息。

用血液循环系统来作为企业隐喻恐怕能让更多人良好地理解人员、流程、信息在企业运营中的作用，人员相当于血红细胞、流程相当于血管组织结构、信息相当于血液中所携带的营养，三者共同构成了血液循环系统的核心。

我们认为将系统连接性和重用现有资产能力作为切入点并不太合适，它们作为企业 SOA 的评价机制恐怕更加好一些。系统连接性不足以影响整个系统的运行功能，只是在其性能尤其是易用性、安全性等角度提供更加良好的支持；重用现有资产能力实际上是企业成本降低的一种评价标准，重用率越高表明企业原始资产价值的转化率越高，使企业架构新型信息系统的成本降低，从某些角度上来讲它可以直接影响到企业决策层是否选择在企业内部部署 SOA 的决心和信心。

切入点是指导，它指导我们在实际部署 SOA 的过程中必须详细了解该企业的现有信息系统状况、现有业务状况、未来业务发展愿景和趋势，在这些基础之上考察人员、架构、信息，将他们分别独立地进行“采样”和分析，最终汇总并权衡给出符合该企业的 SOA 组织结构，这才是 SOA 实现“落地”的正途。换句话说，SOA 是一种原则的指导并不是实际的模型，任何企业都应该选择符合自身状况的不同的实际 SOA 模型，SOA 架构分析师的价值就体现在如何根据事实分析、抽取、提炼、升华出这样的模型的过程中。人本身的价值在于创造，SOA 的概念提出是一种创造，同样实际企业中信息系统的 SOA 模型的提出也是一种创造，创造本身并无不同，不同仅仅在于具体创造的内容和其价值而已。

1.5 项目设计中 SOA 的应用

首先，我们详细分析了凤凰公司的业务流程，在原有的销售部门和财务部门的基础之上添加生产、供应、研发、质管等部门，并给出了详尽的企业业务分析报告（详见业务模型分析部分），这是 SOA 现实部署的基础保证，合理的业务流程分析是准备进行服务划分的基础；

其次，我们针对企业价值链的实现过程提出了面向价值架构（VOA）的核心概念和立足点（详见 <http://nksoateam.blogbus.com>），将业务流程分为价值实现过程、价值增值过程两大部分，并分析了业务流程中可能出现的限制甚至阻碍企业价值实现的流程，分析了现有企

业信息系统环境中的弊端，并在此基础之上通过业务矩阵分析提出了合理的解决方案：

第三，我们对改进后的企业信息化系统流程进行了适当力度的分解，将其拆分成为可以独立部署的服务（详见服务模型分析部分），并对每个服务制定了服务规约，这些基本服务成为了 SOA 现实部署的基本组成部分，它也是实际系统能否达到松耦合的关键所在；

第四，我们针对 SOA 中处于核心地位的 ESB 进行了彻底分析，并通过与计算机体系结构中的总线进行比较（详见 <http://nksoateam.blogbus.com>）更加深入地理解它的设计理念，并以此为基础设计了系统架构，

第五，在对 SOA 和 ESB 的认知过程中，我们发现了目前 ESB 仍然存在的弊端（详见 <http://nksoateam.blogbus.com>），并分析了这种弊端可能给 SOA 带来的致命风险，进而提出了超越 SOA 的理论必然和方向指引；

第六，我们在组件设计的过程中遵从了普遍适用的 Web Service 设计原则，将每个的服务包装成为自组织、接口标准、异步传输、消息自解释的独立组件，这成为 SOA 现实部署中的关键与核心步骤；

第七，我们针对增值服务，提出了数据挖掘支持决策的机制，即通过对 CRM 及 ERP 系统中存在的用户资料、订单资料（包括成功生产并交付转化为价值的订单以及因为各种原因没能成功交付的订单）抽取并挖掘出合理的决策支持信息，为企业进行预防性生产管理、客户优先级及订单动态管理提供必要的支持，使企业在综合实力竞争及“蓝海”竞争中谋得先机，真正达到“随需而动”，这些都是在 SOA 架构的基础之上通过 IISC 进行松耦合部署的。

第二章 思维与理论创新

2.1 核心创新点与立足点

SOA 的简化及降低成本才是其普及的前提和基础，否则 SOA 永远只能停留在理论上，但这种简化不是指核心思想上的而是应该结合企业实际提出恰当复杂度的解决方案。SOA 归根结底是需要使企业价值增值最大化和生产、管理、维护成本最小化的，所以我们应该从企业价值的角度审视 SOA 而不仅仅是技术的角度，只有这样才有真正的指导实践的意义，也只有这样才能使 SOA 更加迅速的普及，成为企业手中的“利斧”而不是“口哨”。

这也是我们提出的核心创新点和立足点：**面向价值的架构（VOA）**。在学习和部署 SOA 的过程中，应该抓住其本质而不仅仅是表面上看到的技术层面的东西，那些只是“术”，SOA

的真正所谓“道”体现在价值上，我们提出了 VOA 的概念，即面向价值的体系架构（Value-Oriented Architecture），它才是 SOA 的本质和基础。企业价值链从采购部门到生产部门再到销售部门，这其中体现了价值创造到价值实现的过程，是一个循环往复的过程。价值链条中任何一部分出现问题都会影响企业经营的最终目的——价值增值。推而广之，这种价值链不仅仅存在于企业内部，同样存在于企业与企业甚至行业与行业之间，值得注意的是这种行业的范围早就已经超过了地域的限制而体现在全球一体化的经济体系之中。如何在保证价值链正常运作的前提下，使得它能合理、高效、方便的运行一直是企业管理者、行业专家、信息系统专家日思夜想要解决的问题。

不可否认的是，在企业信息化系统建设的过程中，在业务专家和信息系统专家之间存在着一种看似难以逾越的鸿沟，这种鸿沟是由于所处领域不同、理解事物的层次不同、认识事物的方法不同导致的，也就是说在这两种人之间难以存在一种通用语言，但是领域专家与信息系统专家的沟通从某些层面却又决定着整个系统的成败与否，业务流与信息流的统一成为了决定信息化建设成败的关键因素。如何让双方都能理解 VOA 与 SOA 中那种耦合性外化与价值封装的理念至关重要。我们尝试通过业务矩阵和功能矩阵的匹配去找到一种两部分人都能理解的架构原形系统的基础元素模式（详见业务模型分析部分）。

2.2 分析与设计中的创新

2.2.1 思想解读：SOA VS 冯·诺依曼体系结构

SOA 是面向服务的软件架构，它是一种服务组织方式。谈到组织方式或架构，任何学习过计算机发展历史的人都应该对冯·诺依曼体系结构耳熟能详，从早期的电子管计算机到现在的大规模并行计算机都采用了冯·诺依曼体系结构，它的最核心的思想如图 2 所示。在整个冯·诺依曼体系结构中，存储器是核心——输入输出设备访问的内容全部存储在存储器中；运算器运算的数据及运算的结果也是存储在存储器中；控制器需要访问的控制命令本身也来自于存储器。

仔细想来，冯·诺依曼体系结构特别像在企业战略管理中的价值链。价值链是指企业提供的产品及服务在价值产生（生产）和价值实现/增值（销售）过程中存在的环节或链条，价值链中的每项活动都对最终价值实现和价值增值起到至关重要的作用，即供应商、分销商、服务商和客户等构成了完整的价值链条，认识到价值链、价值实现过程和价值增值过程的存在是链条中每个实体实现合作与共赢的必然前提。

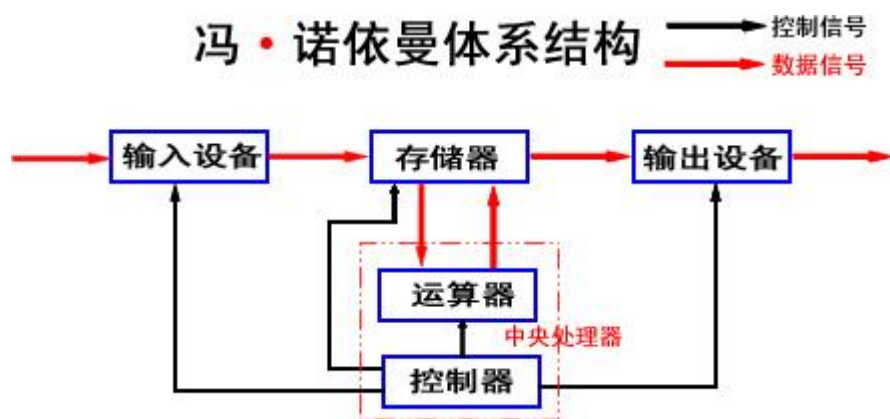


图 2 冯诺依曼体系结构

我们将生产企业从价值链中抽取出来，首先来分析一下它的基本价值链与冯•诺依曼体系结构的关系，这虽然从某些方面割裂了价值链的上下游，但是并不违反价值链本身的特点。因为价值链按照涵盖的范围可以分为三个层次——最广义上的是行业之间的价值链；其次是行业内部各企业之间的价值链；最后就是企业内部各部门之间的价值链——在这里我们首先讨论企业内部各部门之间的价值链体系，如图 3 所示。

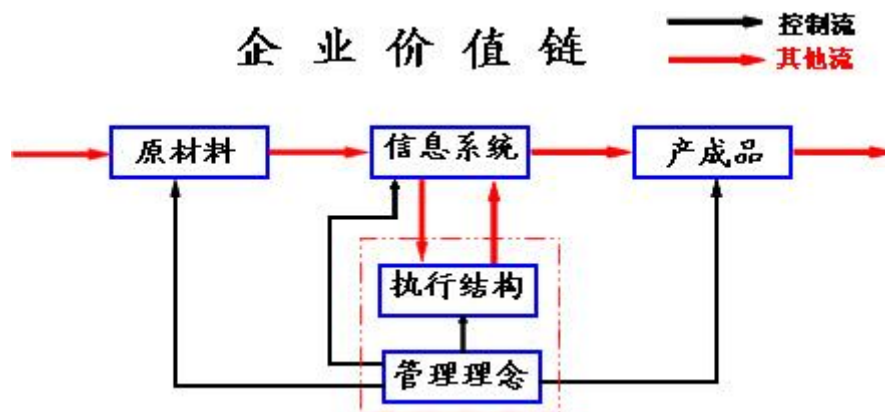


图 3 企业价值链

作为企业来讲，原材料（对于销售型企业就转化为了代售商品）和产成品（对于销售企业就转化为了货币）自然而然的作为企业系统结构中的输入输出部分；而管理理念是整个企业的中枢神经，它决定着如何控制各个部件之间的运行；各个独立的部门也就自然成为上行下效的具体执行机构。那么关键是冯•诺依曼体系结构中的核心——存储器在企业结构中到底体现为什么呢？按照我的理解，它是由各类信息系统组成的体现企业本身商业智能的集成信息系统环境。对于这样一个信息系统环境，“原材料”需要进入它进行处理并产生相应的“产成品”，企业经营和管理的理念也以一种智能化控制信息的方式“存储”在这样的信息系统环境中，作为执行结构的企业各个部门也都是通过这样一个智能的信息系统环境相互交流

并体现着各自的执行力的。所以无论从那个角度来讲在单个企业价值链中，智能的信息系统环境都起到了类似冯•诺依曼体系结构中存储器的核心作用，而它存储的恰恰就是我们所谓的“商业智能”。其实无论是行业之间的价值链还是行业内各个企业之间的价值链都从本质上体现着冯•诺依曼体系结构的基本思想，只是在这个时候它的价值流动不再是单向的、开链的，而是多向的、成环的，其中的信息系统环境涵盖的范围也会更加广泛、提供的功能也会越来越多、适应性和可重用性也会更高而已。

2.2.2 技术解读：ESB VS 计算机总线

ESB 是面向服务的体系结构中消息、数据、控制传递的媒介，是 SOA 的核心组成部分。通过与计算机总线的基本概念和原理的比较可以更加清楚地理解 ESB。

计算机系统结构中的总线是传送信息的公共通道，它由一组传输线和多个三态门组成，高电平、低电平、高阻抗构成了三态门的基本形态。计算机中的总线按照信息传送方向可以分为单向总线和双向总线，按照信息类型分可以分为数据总线、地址总线和控制总线，按信号传送方式可以分为并行总线和串行总线，按所处位置分可以分为芯片内部、（模板内部、机器内部）、外部总线。计算机总线对于计算机组成的意义在于减少连接线、方便系统构成、扩充和更新。

企业服务总线（ESB）是一个基于消息的通信模块，ESB 的功能在于服务集成而不是参与业务逻辑，它主要用于服务的注册命名与寻址功能，以及作为面向服务的中介提供透明的服务路由、定位同时支持多种消息传递机制和传输协议。它还可以消除不同应用之间的技术差异，让不同的应用服务协调运作，实现不同服务之间的通信与整合。

ESB 和计算机系统总线本身都不参与各自系统的逻辑处理部分，都是仅仅负责传输某些数据，ESB 传输的是标准格式封装的数据，计算机总线传输的是二进制数据流；ESB 上面的服务是靠事件驱动的，任何包装好的服务只有在实际应用发生时才会被调用，计算机总线上的设备实际上也是事件驱动的，依靠控制流对总线上多个部件进行状态控制，而控制流本身就是有其他部件产生并发起的；ESB 中存在有优先级决策模块，它负责权衡和协调 ESB 总线上各个服务部件之间的关系并决定其响应优先级，在计算机系统总线上同样也存在这样的优先级判决模块，其中最明显的就是外部中断响应判决器了……我们可以从两者之间总结出太多的相似，甚至不得不怀疑 ESB 的提出者在设计的时候多少都从计算机总线中得到了些许启发。由此我们也想到了在实际系统架构设计中，尤其是 ESB 的设计上能否借鉴到什么呢？（详见系统架构设计部分）

第二篇 企业业务模型分析

第三章 业务模型分析与展望

企业是通过创造价值以获取收益的经济组织。价值是产品（或服务）的一种综合属性：它是产品（或服务）所具有的，能够满足顾客需求而且顾客愿意为之支付货币的一组功能。图 4 是基于价值的企业业务模型。竞争优势归根结底取决于企业向顾客提供的价值，当企业能够以较低的成本向顾客提供同样的价值；或是以同样的成本向顾客提供更大的价值时；企业就拥有了竞争优势。

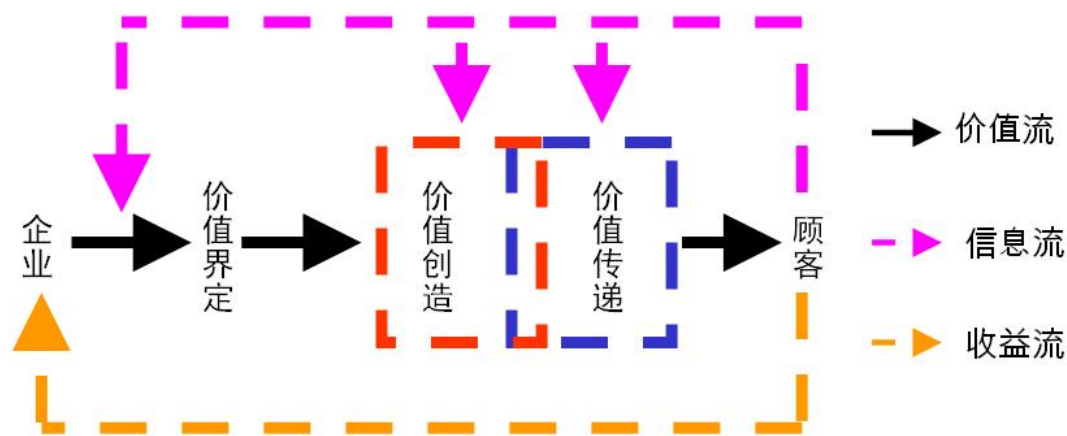


图 4 基于价值的企业业务模型

每个企业都试图设定一系列关于顾客需求的假设：顾客的价值诉求点是什么，顾客希望藉由何种方式得到产品，等等。这些假设是每种业务模式的基础。但是事实上，许多经理人员发现，有些假设只是企业自己想当然的，尽管已经存在了若干年，但是并不符合现实情况。以凤凰医疗设备有限公司为例，凤凰公司的销售人员挖掘并发现业务机会，然后通过采购部核对库存生成服务，最后再经由财务部审查与确认订单来完结服务，整个业务模型可以简化为市场发现、服务交互以及服务完成三层构面。具体的业务流程图如图 5 所示。显然，在市场发现构面，精干的销售队伍利用 CRM 系统处理事务，是典型的以营销手段为导向；而在服务交互构面，销售人员却不得不面对“有需求无库存”的尴尬，这恰恰是以内部约束为导向的。这就造成了在整个服务进程中，内部约束限制了营销手段，换句话说，是无米之炊难倒了巧妇。

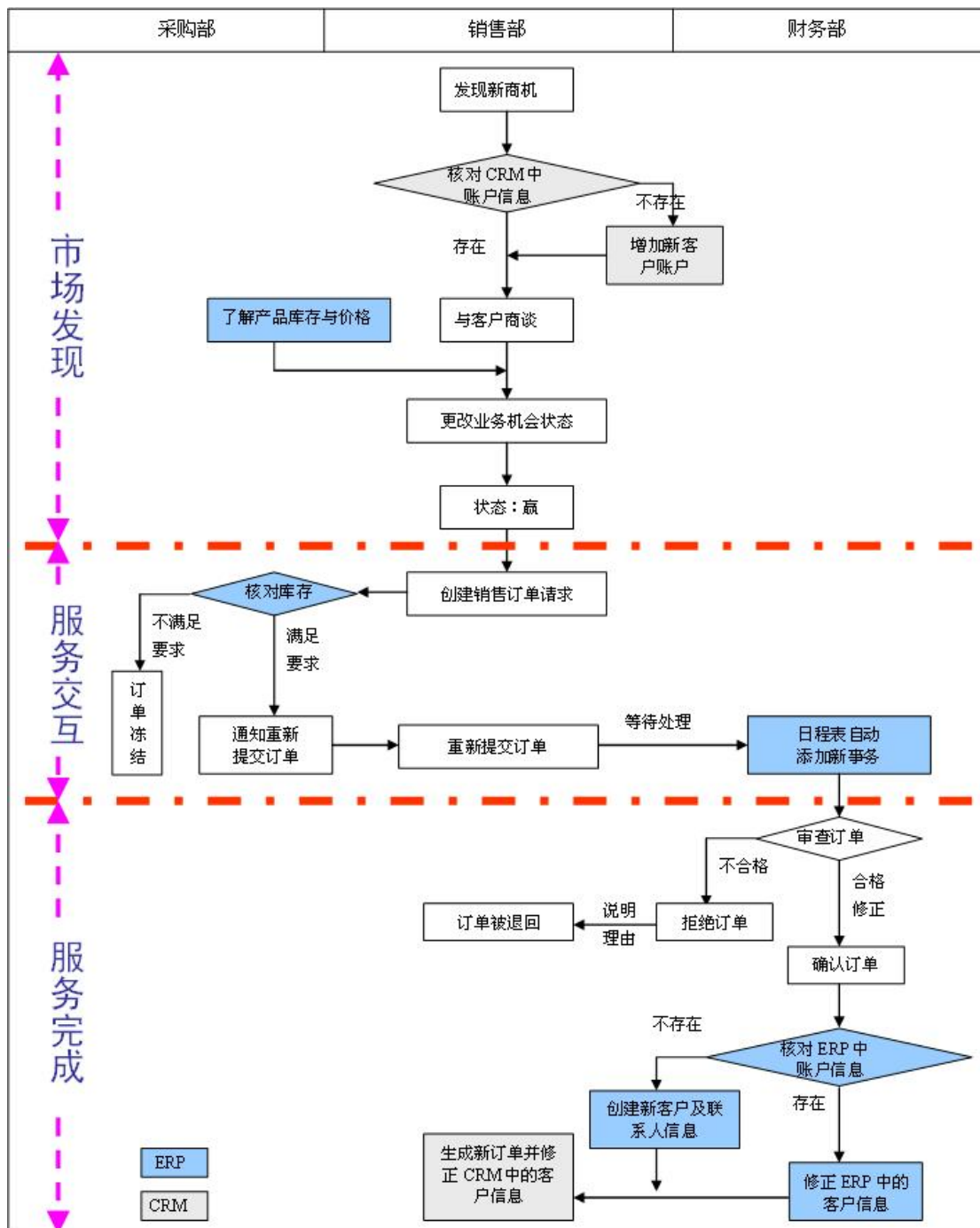


图 5 凤凰公司的业务流程分析

综合上述思考重新来审视凤凰设备有限公司的业务模型，不难发现其特点：

首先，业务规划性低。

缺乏对顾客价值诉求的深挖掘，而更多地依靠销售队伍各凭己力与臆想假设，结果是市场细分不明了，顾客层次不清晰。这时引进 SOA 的核心价值在于管理理念的更新和企业顾客战略的转变。SOA 旨在解决企业从最初细分市场和设定目标顾客

一直到赢得流失的顾客等一系列活动中可能存在的问题,它应该只致力于那些对企业核心竞争力至关重要的流程,如果不能为关键顾客带来关键价值,企业就难以获得预期的回报,自然称不上具有良好的业务规划战略意义。聪明的 SOAer 决不会满足于既有成就,他们会一丝不苟地分析系统所产生的数据,寻求新的价值缺口和业务机会,建立起有价值的互动关系。

其次,被动依赖性强。

在谈到业务流程和顾客导向时,总会有企业提出异议——“我们企业也有流程,也是在努力体现以顾客为中心的呀,为什么没有成效!”实则不然,他们的流程很多都是粗放型的,无协作型的,没能达到以顾客为导向的优化与细化。凤凰公司的内部约束强于外部营销即为典型的一例,正是因为这些内部约束导致的被动依赖性,深切影响了企业的整体绩效。以 SOA 思想来优化与细化的目标是标准化、精细化与可衡量化,实现的手段就是建立真正体现以顾客为导向的指标体系。正如 CRM 领域独立顾问叶开先生所讲:“你能衡量的,才是你能管理的;如果你不能测量,你就不可能控制它、管理它!”

第四章 业务模型对 IT 系统的挑战

4.1 SOA=VOA: 面向商业价值的系统架构

不能简单地将 SOA 理解为一个技术甚至是软件的引进,SOA 最核心的内容是价值:给企业带来的价值是什么,给顾客带来的价值又是什么。如果企业把 SOA 当作技术而不是管理实践来实施,SOA 必将走向夭折。要知道真正能给企业带来利润的并不是 SOA 项目中应用的 IT 技术,而是一种全新的发展理念与实践,技术不过是实现这一理念的手段,把 SOA 当作技术就是犯了把手段当作目的的错误。

以著名的战略大师迈克尔·波特(Michael Porter)的价值链模型为基础,当企业借助 IT 系统将产品更快、更好并以更低的成本推向市场时,我们尝试着构建基于信息技术的企业价值链,在企业的日常运营中更好地运用 SOA 思想,改进产品的价值增值过程,优化创造价值关系所形成的企业活动体系。如图 6 所示。

从顾客的角度来看,SOA 思想可以直接为顾客带来以下几方面的价值:更快的需求响应时间、更完备的服务体验、更稳定持久的服务流程。在以顾客为导向的营销过程中,如何发现客户只是服务的开始,如何留住客户才是价值实现的关键。SOA 将企业内部各个

业务流程进行了更为灵活高效的组织，对顾客的需求可以提供最快速的响应，超越 CRM 和 ERP 的系统集成又使得顾客的需求能够得到最大限度的满足。SOA 使得企业的市场营销流程更加完整和稳定，发现并留住客户将变得更加容易。



图 6 IT 系统与企业价值链

从企业业务流程运作的角度来看，SOA 实现对企业各个业务部门更加灵活、更加高效的组织，在 SOA 系统架构中，各种业务信息能够自由的流动，自动执行的业务规则保证了信息的准确的和完整，而灵活丰富的服务管理模式又使得企业内部业务流程可以从容的应对市场的变化。传统封闭的企业 workflow 管理在融入了松散耦合、自由绑定、面向服务的 SOA 理念之后，从根本上提升了企业的核心竞争能力。

从企业决策和管理者的角度来看，SOA 使得企业能够脱离 IT 基础建设的投资泥潭，以更加直观、更加廉价的方式来搭建更贴近企业核心价值的 IT 系统平台。建设具有自主生长功能的 IT 基础环境是很多企业管理者的梦想，SOA 使得这个梦想第一次变得真实可行。借助于 Web Service 的优势和 SOA 的先进理念，企业业务流程的调整不再需要复杂昂贵的 IT 系统重构，简洁廉价的服务定制、服务订购、服务重组使得 IT 系统能够为企业发展提供更加有力的支撑。

IT 指导下企业价值链的运作过程是非线性的——它是由一组可能的输出和输入活动的矩阵所构成，并有着多种分销渠道。IT 系统的运作使企业能够通过对顾客的深入了解，

全面地满足他们的需求，为他们创造出新的价值。

利用 IT 系统创造价值可以分为三个阶段：

第一阶段，清晰性，即企业利用信息获得一种更为有效地控制有形操作的能力；

这是企业应用 IT 系统的初级阶段，其主要目的是实现从手工操作向数字化、自动化的转移，形形色色的 OA 系统正是这一阶段的典型 IT 系统。

第二阶段，映射能力，企业用虚拟活动代替有形活动，在三维市场创造平行的价值链；

这是企业应用 IT 系统的中级阶段，其主要目的是通过建设或强化专用的 IT 系统，将企业价值的实现过程彻底转化为 IT 系统内部的信息流和控制流。在本次大赛的题目中，使用 ERP 和 CRM 正是这一阶段的真实写照。

第三阶段，新制式，企业利用 IT 技术建立新的价值创造体系。

这是企业应用 IT 系统的高级阶段，在这个阶段中，企业将在现有 IT 系统平台的基础上，搭建更为灵活、更符合“以顾客为中心、以市场为中心”原则、更具有主动性的 IT 环境。在本次大赛的题目中，凤凰公司提出的系统集成和扩展服务要求的核心目的就是建立更加高效、更加便捷的信息化企业价值创造体系。

立足于以上业务模型分析，我们能够对 SOA 有更为深入的理解：对企业的业务模型而言，SOA 就是面向价值的系统重用和集成了。

4.2 大赛题目解析：超越系统的集成

通过对 SOA 应用创新大赛题目中所描述的凤凰公司业务模型进行分析，我们不难发现凤凰公司正面临着 IT 系统集成带来的巨大挑战，同时又因为 SOA 的出现而带来巨大的发展良机。对凤凰公司而言，业务发展对 IT 系统提出了以下重要的挑战：

1. 业务规则的规约和集成

凤凰公司的现有业务模式中存在着外部价值链条与内部价值链条的脱离情况，市场营销人员开拓的客户资源和业务机会依赖于公司内部的库存管理和财务审计，而公司内部的业务规则却无法及时的响应市场情况的变化。如何通过 IT 系统的集成和升级实现企业内部不同部门之间的业务规则集成是凤凰公司面临的最大挑战。

2. 业务信息的低冗余和自动同步

在凤凰公司的现有 IT 系统中，存在着大量的信息冗余和同步困难，客户信息、库存信息、业务机会与订单信息分散在不同的 IT 系统中，信息的存储和管理异常复杂、信息的同步又必须靠大量的手工操作来实现，这必然导致大量的偏差和错误，

需要更加高昂的成本来弥补。将企业内部的 ERP 系统和 CRM 系统进行有效集成的关键目标在于保证企业内部信息流的低冗余存储和自动同步更新，这是在业务规则集成基础上需要实现的关键目标。

3. 业务模式的灵活调整和增值服务

在凤凰公司现有的业务模式中，市场营销流程与内部业务处理流程存在着严重的脱节和错位，市场变化风云莫测，唯有及时适当的调整丰富的业务模式才能保证企业的长久发展。在现有的 IT 系统环境中，凤凰公司的各种业务依托于不同的 IT 系统，业务模式的完整性只能靠员工的手工操作来保证，而任何新的业务模式都必然带来大量的 IT 系统技术开发升级、员工重新培训、内部业务流程调整等工作。要想方便的立足于企业内部信息平台使用 Internet 上的各类 Web 服务也非常的困难。

在现有 IT 系统的基础上，通过灵活的系统架构和关键技术开发以实现业务流程的自动化、智能化运转，实现业务模式的自由扩展和服务的方便设计，是凤凰公司面临的重要挑战。

4. 企业价值链的优化与升级

随着商务智能概念的普及，越来越多的企业期望在 IT 系统建设的过程中实现对企业核心价值链的把握和优化。对凤凰公司而言，着内部业务价值链的资源整合与优化是提升“内功”的关键，外部市场价值链的稳固和拓展是提升“外功”的关键。ERP 作为企业内部价值链的维护平台有着不可替代的作用，但是缺少与外部市场价值链的互动；CRM 作为企业外部市场价值链的实现平台同样有着不可替代的作用，但是脱离于 ERP 平台使其无法最大效率的发挥作用。借助 SOA 的先进理念将这两个 IT 系统进行有效的集成是凤凰公司为实现发展愿景的必然选择。

SOA 的核心理念并不仅限于技术创新，而是对企业业务模式的良好建模、对企业价值链的优化升级，从低级的数据流、信息流、控制流转向高级的消息流、服务流、业务流，由此实现的将是超越系统的集成。

SOA 赋予了企业“超越计算”的能力，业务、服务、操作的完美结合为企业价值的实现提供了充足的保障，为企业提供了更强大的商务智能平台。

正如我们团队对 51Start 这个名字的解读中所提到的，SOA 是一种服务、一种技术、一种架构、一种重组，更是一种我们无法逃避的趋势。选择 SOA，就是选择未来！

第三篇 服务模型分析和设计

第五章 服务建模概念解析

5.1 Service: SOA 的理念核心

SOA 是面向服务的系统架构过程,“服务”是 SOA 的理念核心,唯有清晰准确的理解并定义“服务”集合,才能实现对 SOA 的良好实现。

本文从以下几种角度对“服务”(Service)的概念进行阐述。

5.1.1 从企业价值的角度看“服务”

对渴望通过 SOA 实现内部 IT 系统集成和优化的众多企业而言,“服务”这个概念存在于企业的价值链条中,如表 1 所示。

表 1 企业价值体系中的“服务”概念

概念	内涵理解
业务 (Business)	企业在实现商业价值过程中所采用的业务模式和业务流程,业务是企业价值的核心,IT 系统集成的关键目标就是业务流程的优化和业务模式的升级
服务 (Service)	隶属于不同业务的事务集合,一项服务能够实现某种粒度的业务目标,多项服务可以组成完整的业务流程
操作 (Operation)	对服务的控制和使用,通过操作可以向服务输入各种信息、获取各种状态、控制服务运行的过程。所有被允许的操作都是服务提供的。
角色 (Role)	企业业务流程中不同分工所对应的人员角色,不同的角色对应着不同的操作权限,角色间的相互协作形成操作流程、进而形成服务流和业务流。
信息 (Information)	企业员工选择进行某种操作时所依赖的数据基础,它可以是状态、可以是详细的业务数据,IT 系统集成的基础目标就是实现对信息的良好管理和高效使用。

5.1.2 从系统架构的角度看“服务”

在遵循 SOA 理念进行系统架构的过程中，“服务”是一种广义的概念，它按照在系统架构中的层次和作用分为不同的类别，它代表的首先是一种“能力”，其次才是技术细节的具体方法和实现过程。表 2 描述了系统架构过程中服务类型的划分和作用。

表 1 系统架构过程中的“服务”概念

层次		服务类别	作用描述
低 	1	信息服务	负责从企业 IT 系统环境中提取各类必要的信息，并对信息进行必要的汇总、解析。
	2	事件服务	负责根据操作请求激活不同的事件，完成业务流程中必要的状态传递、同步和更新等操作。
	3	流程服务	负责根据业务模型的需求组织完整的服务流，将信息流和控制流进行有效的组织。
	4	传输服务	负责支撑 IT 系统内部不同类别服务间的协同工作。
高	5	协作与开放服务	负责接入并使用外部第三方提供的开放式服务，它可以采用同步模式，也可以采用异步模式。
	6	系统维护管理服务	负责保持 IT 系统的稳定高效运行，负责对服务进行登记和管理、负责记录系统运行过程信息。

5.1.3 从技术实现的角度看“服务”

对于很多 IT 技术开放商而言，广义的“服务”概念只是存在于系统架构之前的需求分析和系统设计中，一旦进入实现流程，“服务”的概念就变得更加具体和细节化，在技术实现的过程中，“服务”概念的实现需要注重以下几点：

✧ 服务规约与元数据定义

在 SOA 的概念体系中，服务规约是保证不同类型、不同层次的服务间能够松散耦合、灵活绑定的关键所在，服务规约就像是一种可供交流的语言，它限定了服务的存在形式和服务之间协作的方式。遵循服务规约实现良好的元数据定义是技术实现的基础所在。

✧ 组件设计与实现方法选择

抽象的“服务”概念是通过具体的编程实现的，在程序设计中，独立组件或者多个组件实现了一个服务的具体内容。组件设计的方法有很多，对 SOA 而言，组件设

计的关键在于保证通用性、封装性和独立性，不同组件之间既可以协作也可以独立运行，唯有如此才能实现松散耦合的目的。

在实现组件的过程中，实现方法的选择是系统架构的重要问题，选择通用的、相互兼容的实现方法是合理而明智的选择。

◇ 接口设计与可扩展考虑

为便于使用各种服务，在技术实现过程中必须注重各类服务的“接口”设计，我们认为在服务的接口设计过程中，最重要的是保证接口体系的“自我保持”能力，所谓自我保持，就是指一个服务的接口集合具有完整的管理、控制、运行功能，它只依赖于外界的信息和状态，而运行逻辑却完整内含，这样的设计思路可是保证每一个粒度级别的服务都具有强大的生命力和通用性。

除接口设计之外，可扩展性是服务设计时需要重点考虑的内容。任何企业都将随着市场的变化随时调整业务模式与工作流程，现有的服务在未来的市场情况下必将进行改变才能满足需要，可扩展性是通过“策略与机制”分离的方法来实现的。每一个服务中所遵循的业务规则都应该以规范完整的形式在外部进行定义，而服务的运行过程就是对业务规则进行解析和执行的过程，采用这样的方法，能够最大限度的保证服务的可扩展性。

对 SOA 而言，“服务”的概念更重要的体现在企业价值层面（业务流程）和系统架构层面，唯有在这两个层面中做好服务模型的分析设计、服务类型的划分定义，才能保证技术实现工作能够符合企业发展的需要。

5.2 服务建模的过程与价值分析

对任何一家企业而言，进行 SOA 应用实践是一个比较复杂而又艰难的过程，这一过程主要包括“需求分析——业务模型分析——服务模型分析设计——系统架构设计——技术实现——系统运行维护”等多个关键步骤，其中服务建模是保证 SOA 思想得到良好体现的关键步骤，在进行服务建模的过程中，我们遵循以下步骤展开工作。

5.2.1 业务流和信息流的分解与汇总

为实现良好的服务建模，应采用“自顶向下——自底向上”相结合的方式对企业现有的业务流程、IT 环境、信息流程进行分析。采用“自顶向下”的方式对业务流进行拆解，采用“自底向上”的方式对信息流进行汇集。最终提炼得到最佳的服务模型。

5.2.2 服务类型划分与层次设计

在实现对业务流和信息流的分析之后，最关键的服务模型已经定义完成，但是如果将其直接作为技术实现的目标，就无法保证系统结构的完整和灵活。我们将遵循表 1、表 2 中描述的内容，对服务的层次、类别进行划分，建立系统结构级的服务集合描述。

5.2.3 服务规约的定义与实现

完成服务集合的系统架构设计之后，我们将进一步实现对服务规约的设计，我们将采用“事件驱动”与“数据驱动”相结合的模式来规划服务之间的协作过程，“消息机制”是集合两种驱动模式的最佳选择，使用消息机制的另外一个重要目标是保证服务在协作过程中的“异步工作”模式，传统的服务实现方法常采用同步工作模式，当企业业务流程繁忙时，这必将造成服务消费者长时间的等候，降低 IT 系统的友好性。在 SOA 中采用异步工作模式，使得不同的服务消费者能够最大限度的发挥松散耦合的优势，最大限度的发挥系统性能。

采用以上工作步骤，就可以实现良好的服务模型设计，为 SOA 理念的实现奠定扎实的基础。

第六章 凤凰公司服务模型分析设计

6.1 服务模型的原始定义

6.1.1 服务类型的划分

通过对凤凰公司业务模型的分析以及对内部详细业务流程的拆解分析(详细业务流程分析请见 8.2.1)，可以定义以下不同类型的服务。

1. 信息服务——提取并集成 CRM 与 ERP 中的信息

在本项目中，销售人员需要 CRM 中的客户信息和 ERP 中的库存信息以确定一个“业务机会”。当“业务机会”得到满足时，将自动的生成“销售订单”，财务人员将根据库存情况、价格策略实现对“销售订单”的审核，审核结果将通知给销售人员。在这一过程中，“业务机会”和“销售订单”是立足于企业业务逻辑角度定义的信息类型，而 CRM 和 ERP 中的数据是立足于 IT 系统平台角度实现的数据存储，必须定义良好的信息服务以支持“业务机会”和“销售订单”对 CRM、ERP、以及其他数据信息的自动提取和汇集。

2. 业务流程服务——实现凤凰公司的核心业务过程

在本项目中，关键的业务流程是“销售订单”的审核过程，在这一过程中，销售人

员负责“销售订单”的生成，财务人员负责“销售订单”的审核，销售人员负责根据审核结果向客户提供反馈。这一核心业务流程可以被封装为一个独立的服务模型，通过良好的规划设计以实现松散耦合的自动化业务流程。

3. ESB 事件服务——实现松散耦合的事件分发

在业务流程服务中，财务人员需要及时了解“销售订单”的到达情况，每当销售人员生成一个“销售订单”时，系统应自动通知财务人员。当财务人员完成审核工作后，系统应自动将审核结果通知给销售人员，这样的协作过程必须依赖 EBS 中的事件服务来实现，一旦关键操作完成，事件服务将接受事件请求并将其分发给对应的服务组件。

4. 传输服务——负责实现运行流程的驱动和信息传输

对信息服务而言，需要从多个底层数据库中提取关键信息；对于业务流程服务而言，需要通过获取事件状态来实现对工作流的控制。为此必须设计功能完整性能稳定的传输服务，既要保证系统结构的完整，同时又要保证业务流程的正常运转。

5. 外部集成服务——集成并使用第三方的商务智能服务

在本项目中，凤凰公司提出了希望通过订阅的方式来使用 Web 上丰富的第三方智能服务，实际上凤凰公司目前使用的 CRM 服务就是通过订购方式实现的。对于类似于文档图象管理（DIM）和网上智能信息检索等第三方服务而言，在遵循服务规约的前提下，通过外部集成服务可方便快速的接入到现有的 SOA 体系结构中。

基于以上描述，表 3 列出了我们为凤凰公司定义的服务模型。

表 3 凤凰公司的服务模型设计

服务名称		服务类别	服务描述
信息聚合服务		信息服务	负责提取 CRM 和 ERP 中的信息，形成“业务机会”和“销售订单”的核心数据。
业务机会管理服务	检查业务机会	业务流程服务	根据业务机会的内容检查库存信息，查看库存是否满足业务机会的要求。
	更改业务状态	业务流程服务	手工操作，业务机会状态改为“赢”。
	生成订单	业务流程服务	手工操作，业务机会变为“赢”时，负责生成订单
销售订单	检查订单	业务流程服务	检查修订订单内容，增加客户信息

管理服务	审核订单	业务流程服务	审核订单是否满足业务规则，改变订单状态
ESB 事件服务	订单到达	事件分发服务	通知销售订单管理服务处理新订单
	审核结果到达	事件分发服务	通知业务机会管理服务接受订单审理状态，由销售人员向客户提供后续服务
信息传输服务		传输服务	负责实现业务机会管理服务和销售订单管理服务的信息传输过程
智能服务入口		外部集成服务	负责集成外部的第三方智能服务。

以上定义的服务内容覆盖了系统结构的各个层次，并遵循松散耦合、灵活绑定的原则进行了明确的分工。在以上的服务设计中，ESB 起到了承上启下的关键作用，它负责接受使用者在前台的操作请求，并根据请求激发不同的事件进行处理。它负责将数据从 CRM、ERP 以及其他基础数据库中提取并交付信息服务。

6.2 系统架构的层次分解

基于表 3 描述的服务模型定义，图 7 描述了我们设计的服务模型和服务系统架构。

图 7 中描述的高层系统架构立足于服务模型，而在本文的第四篇中，将提供更加详细、更加具体的系统结构图用以描述设计和实现过程。

第七章 服务规约与技术实现

7.1 服务规约的作用和价值

在上文中我们已经对服务规约的作用进行了必要解释，服务规约是 SOA 应用过程中保证服务模型设计准确完整的重要基础，站在技术实现的角度来看，服务规约的概念比较虚泛，但是站在系统结构和企业应用的角度来看，对服务规约的理解就更加直观和简单，IBM 也提供了一系列支持服务建模的工具，为服务规约和服务实现提供了良好的保障。

7.1.1 服务规约的作用

服务规约的主要作用在于以下几点：

1. 实现业务规则定义和执行的分离

在企业应用中，业务规则的变化非常频繁，但是业务规则的变化往往体现为数值计算、操作流程变化，其基本的计算集合并未发生变化。这就像企业中一个部门的业务流程一样，员工的分工和工作方式并未发生变化，但是当业务规则发生变化时，员工的协作关系和工资流程就会进行重组。

在 IT 系统的设计和实现中，传统的方式都是将业务规则固化到代码实现中，一旦业务规则发生变化，那么很多代码都需要进行必要的修改。在 SOA 的概念中，服务规约就是指使用规范的形式来定义业务规则，而在代码中只是对业务规则的内容进行解释和执行。这样就可以保证当业务规则发生改变时，各类服务中的代码并不需要修改，只需要进行必要的流程重组和接口参数修改就能够实现业务流程的快速更新。

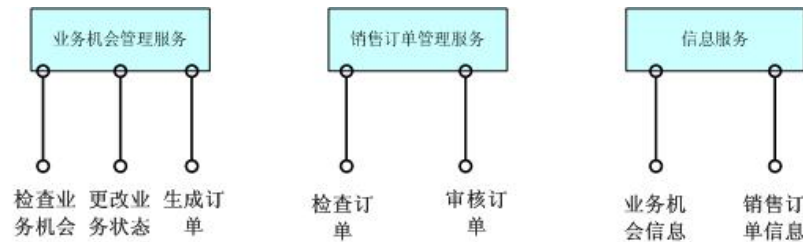


图 (a) 服务模型原型示意图

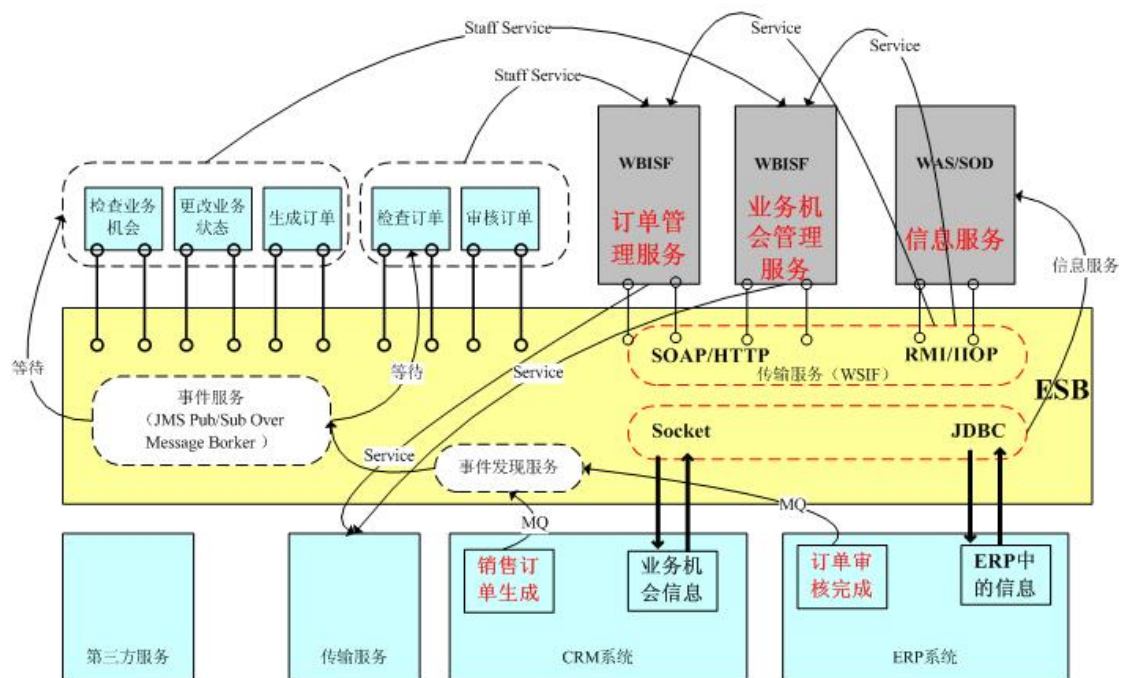


图 (b) 服务模型结构图

图 7 服务模型定义与系统架构示意图

2. 实现服务集合的灵活扩充和自由绑定

在企业应用环境中，随时会出现大量的、新颖的业务需求，IT 系统在设计过程中不可能对未来的需求变化进行完全精确的预测，因此必然需要根据企业发展的要求不断增加新的服务。对企业而言，选择不同 IT 服务提供商、对不同类型的服务进行组合是最合理的选择。但是如果没有统一、规范的服务定义和接口设计，新的服务

无法有效的集成到现有系统中。

服务规约的作用在于定义了一套标准通用的语言，这套语言限定了服务的接口形式、绑定方式和其他必要的系统策略，只要遵循服务规约，任何服务都可以方便的集成到现有系统中。

3. 对人工操作接口进行规范定义，最大限度保证人工操作接口的重用性

在传统的 Web Service 设计中，大量的人工操作接口都通过固定的、针对性极强的方式实现在页面中，这使得企业的 IT 系统中存在着大量重复的人工交互接口，通过良好的服务规约设计，可实现统一规范的人工操作接口定义，使得各类人工操作接口也可以封装为可重用的服务。进一步降低了 IT 系统的冗余程度。

7.1.2 服务规约的设计策略

为保证服务规约的实现效率，我们拟采用以下策略：

1. 使用 IBM Business Modeler 进行全面业务建模

手工进行业务建模的方式既不能保证业务建模的完整性，也无法准确的发现业务模型中的关键参数和性能瓶颈。使用 IBM Business Modeler 进行业务建模，可以最大限度的保证业务建模的完整性和准确性，并为服务规约的提取和设计实现奠定重要基础。

2. 使用 UML 2.0 Profile 进行模式设计，采用 XML 描述各类业务规则

在定义并实现各类服务时，我们将使用 IBM 提供的工具、借助 UML 2.0 Profile 进行模式设计，对各类业务规则的描述，我们拟借鉴各类 workflow 管理中使用的遵循 XML 规范描述机制，建立通用的业务规则描述方法。

3. 使用 WebSphere Integration Developer 和 WebSphere Process Server 作为实现平台和服务部署管理平台

IBM 系列软件的最大优势就是实现了业务建模、服务建模、系统设计、服务部署的无缝连接，符合服务规约的各类服务，均可通过 Service Registry 进行管理，并提供服务消费者获得必要的服务接口信息以供使用。

7.2 服务实现技术方法浅析

针对以上服务模型设计，在技术实现过程中，可以选择最合理的实现机制，现将我们拟针对各种服务选用的技术实现机制逐一介绍如下：

7.2.1 信息服务的实现方法

在本项目中，CRM 系统采用了第三方 Web 数据库、ERP 采用了本地数据库，除此之外信息服务还要提供其他一些可能的数据，我们拟采用 JDBC 或 J2C Connector 来获取本地 ERP 数据库中的信息，采用 Socket 或第三方提供的接入机制获取 CRM 中的数据。本项目中的信息服务将被实现为一个 EJB，外部访问通过 RMI/IIOP 绑定访问这个服务。在信息服务内部，“业务机会”和“销售订单”以 SDO 进行表示。

7.2.2 业务流程服务的实现方法

在本项目中，“销售订单管理服务”和“业务机会管理服务”将被实现为 Process Service，它将被 WBI SF 的 BPEL4WS 容器执行，这两个业务流程服务通过 RMI/IIOP 协议调用，我们拟使用 WSIF 作为协议调用服务的依据，它也是 ESB 中 Transport Service 的一部分。在凤凰公司的业务流程中，更改业务机会状态、生成销售订单、更改订单状态都需要员工的手动操作，我们将其设计为 Staff Service，通过 Portlet 实现，运行在 Websphere Portal Server 上。考虑到凤凰公司的业务流程中需要使用短信通知财务人员相关信息，我们拟设计实现 Delivery Service 来支持手持的智能 PDA 和手机设备，既可以向员工发送消息，也可以供员工通过手持设备访问系统。

7.2.3 ESB 事件服务的实现方法

在本项目中，我们一共定义了两个事件发现服务（Event Detect Service），第一个是当销售订单生成时负责通知“销售订单管理服务”，第二是当销售订单完成后负责通知“业务机会管理服务”。这两个事件发现服务将分别由 ERP 中的相关组件激发，通过 MQ 将事件传送给 Message Broker，通过 JMS 的 Pub/.Sub，这两个事件将分别传输给对应的业务流程服务。

7.2.4 传输服务和外部集成服务的实现方法

我们拟采用由 WSIF 实现的 Transport Service 以 SOAP/HTTP 调用业务管理流程服务，对于第三方智能服务，也可以通过这种方式予以实现，它最主要的作用在于实现了业务服务组件与 ESB 之间的松散耦合和标准调用，这就为 IT 系统架构带来了良好的可拓展性。

第四篇 系统架构设计

第八章 用户需求分析

8.1 需求概述

凤凰医疗设备有限公司是一家专门制造和营销专业医疗器械和实验仪器仪表等仪器的民营企业，其购销客户和网络遍布全国各地。凤凰成立于 2000 年，现有员工 750 名。公司领导一直非常重视企业的信息化建设并投入了大量的资金支持。公司内部普遍使用 Office 办公软件以及常规的工具软件。

目前凤凰公司引进并实施了 ERP 系统（主要用于凤凰公司的财务管理，其中包括产品库存及订单管理等）以及客户关系管理系统（CRM）。公司的财务和销售人员在 ERP 和 CRM 系统上工作，工作效率有很大提高。但是公司目前也面临挑战：一方面，ERP 和 CRM 中分别维护产品和客户信息，而公司规定 ERP 必须作为这些信息的主数据源，ERP 中的这些信息需要随时同步到 CRM 中去；另一方面，CRM 中维护的业务机会和 ERP 中维护的销售订单有着非常紧密的关系，凤凰公司希望能够把业务机会和销售订单有效地整合起来，而进一步提高业务运作的效率。

8.2 凤凰公司业务环境描述

8.2.1 部门流程分析

虽然在竞赛题目中重点提到了凤凰公司的销售和财务两个部门，但是出于完整性的考虑，我们还是愿意参照一般企业的组织结构和部门设置给出一个比较完整的流程分析。

表 4 凤凰公司部门工作流程

工作步骤	部门名称	工作描述	主要操作
1	销售部	客户信息管理	销售人员对客户信息的日常维护
2	销售部	业务机会发现	根据客户需求，在 CRM 系统中创建一个业务机会描述
3	销售部	产品库存查询	销售人员会实时查询 ERP 系统中的库存信息，如果库存能够满足业务机会的要求，则可以进一步创建“销售订单”

4	销售部	创建销售订单	将“业务机会”的状态改为“赢”，同时创建销售订单，并传送给财务人员
5	销售部	销售订单审核等待	销售人员向财务人员发送一个销售订单，财务人员根据库存和产品价格等情况对订单进行审核,有“冻结”和“确认”两种状态
6	销售部	销售订单提交	被审核通过的销售订单将通过 Email 或打印的方式交给客户，用作业务交易的凭证
7	销售部	客户服务跟踪	销售订单在执行过程中，销售人员一方面要监督生产部门做好服务工作，另一方面要配合财务部门做好收款工作，这是一个价值实现过程
8	销售部	订单完结	订单所涉及的所有服务均完成，客户已经付款，直接进入到“技术支持”阶段
9	销售部	市场开发	使用 CRM 中的各种统计功能查看业务进展、业务分布等情况，制定市场开发策略
10	财务部	审核销售订单	财务人员检查并修正销售订单中的某些数据项，如果满足要求则创建正式的销售订单于 ERP 系统中并同步更新到 CRM 系统，如果不符合需要给出退回来由并拒绝接受订单返回给销售人员
11	财务部	审核财务记录	根据正式销售订单的状态生成财务报表，类别包括已完成、生产中、付款中
12	财务部	正式销售订单监控	无人操作

13	生产部	订单拆解	生产部门人员获得优先级最高的订单数据，如所有原材料均满足则立刻开始生产，如有不满足则按需将其分派给供应或研发部门
14	生产部	内部订单监控	无人为操作
15	生产部	维护正式销售订单状态	生产部门人员根据实际情况手动更改销售订单状态为生产完毕
16	生产部	预生产	生产部门人员根据历史数据生成预生产列表，并以订单形式提交给财务部审核
17	供应部	订货	供应部人员收到订货单后选择三种方式：系统全自动处理、系统处理后人工修正、全人工处理三种方式生成电子订货单订货单，通知供应部人员及时发出订货请求
18	供应部	维护内部供应订单状态	供应部人员根据订货单的实际处理情况随时更新内部供应订单状态，使某些成产成为可能
19	研发部	维护内部研发订单状态	研发部人员根据实际处理情况随时更新内部研发订单状态，使某些成产成为可能

8.2.2 数据流分析

上节详细描述了凤凰公司内部工作流程及各步骤对应的职能部门，对于万合公司所要实施的 IT 资源整合任务，我们还需要了解在凤凰公司工作流程中隐含的数据流程，下表如实记录了流动在凤凰公司内部的各种数据。其中“工作步骤”一栏与“凤凰公司部门工作流程”表具有对应关系：

表 5 凤凰公司数据流程

工作步骤	依赖数据	产生数据	IT 系统操作
1	无具体形式	CRM 中的客户信息	增删改查 CRM 系统中的数据

2	客户需求	业务机会描述	向 CRM 中增加一个业务机会
3	业务机会描述、ERP	业务机会满足情况	根据业务机会中的详细需求，查询 ERP 中的产品库存信息，以决定“业务机会”是否能够转化为“销售订单”
4	业务机会满足情况	待审核销售订单	修改 CRM 系统中“业务机会”的状态，在 CRM 系统中增加“销售订单”
5	财务部门的审核意见	正式的销售订单	销售订单将从 CRM 系统向 ERP 系统创建
6	财务部门的审核结果	客户接收的销售订单	销售订单被销售人员交付给客户，作为正式服务的开始
7	生产部门和财务部门的配合	订单进展情况记录	CRM 中对销售订单状态的记录
8	订单完成	订单（状态修改后）	更新 CRM 中订单的状态
9	CRM 中的各类信息	CRM 信息统计结果	查看各种业务统计分析情况，制定市场开发策略
10	待审核的销售订单	正式的销售订单	ERP 系统中财务人员的工作列表中添加事务并通知财务人员审核下手订单
11	正式销售订单	财务报表	ERP 系统财务人员根据正式销售订单生成财务报表

12	正式销售订单	付款通知单	ERP 系统在监控正式销售订单为生产完毕之后自动生成付款通知单并通知财务人员
13	销售订单优先级列表	内部供应订单、内部研发订单	在销售部门提交前一个订单生产完毕之后，ERP 系统自动给出下一个优先级最好的订单
14	内部供应订单、内部研发订单	生产订单列表	生产部门人员通过系统自动检测发出的内部供应订单和内部研发订单是否完成
15	正式的销售订单	正式的销售订单	生产部门人员根据实际情况维护 ERP 中正式销售订单状态，通知销售部和财务部生产完毕
16	历史数据	预生产列表	根据 ERP 系统中的历史数据（主要是因生产能力或供货能力不足导致的交易取消的数据）进行数据挖掘，给出现阶段为了规避此等风险而应预生产的货物。优先级小于正式订单
17	内部供应订单	订货单	根据 ERP 系统中供货商列表及其供货内容，对内部供应订单进行分解成为不同的电子订货单

18	内部供应订单	内部供应订单	根据实际情况在 ERP 系统中维护内部供应订单的状态
19	内部研发订单	内部研发订单	根据实际情况在 ERP 系统中维护内部研发订单的状态

8.3 凤凰公司 IT 环境描述

目前，凤凰公司同时实施了 CRM 和 ERP，其主要功能有所不同：

8.3.1 CRM 分析

CRM 系统主要供给销售人员使用，其功能包括：

1. 客户信息管理

对于销售人员来说，管理一般的客户、伙伴、供应商的基本信息并不难，但是更重要的是动态信息（所有动态交往的记录），CRM 系统为销售人员提供了静态、动态信息管理手段。其中维护着客户名称、网站、股票代码、上级单位、电话传真、所属行业、收获和付费方式等信息。

2. 联系人管理

在当今激烈竞争的市场环境下，每一个企业都把“以客户为中心”做为提升客户关系管理的准则，但论起做好客户关系的本质，其实质上还是做好与人的关系，联系人成为了客户管理中非常重要的一条主线。CRM 系统提供了非常全面的联系人管理，通过客户、伙伴、供应商中联系人的管理可以方便的查询到与企业的联络状态。

3. 业务机会管理

同一个销售人员可能有多个业务机会同时都在进行。那么对这些业务机会很好的管理，方便的获得每个业务机会的信息以及当前的状态就显得很重要了。CRM 系统为销售人员量身定做了业务机会管理功能，使得销售人员在处理多个业务机会的时候得心应手，游刃有余。

4. 销售信息管理

销售人员的销售行为不能盲目进行，必须对当前的市场形势有清楚的认识。这就需要有大量的、完整的销售数据作为基础，尤其是采用销售信息分析图表等形式，可以使销售人员直观的获得目前市场的形势，进行更好的销售活动。

5. 营销活动

凤凰公司定期举行一定规模的营销活动，借以推广其主要产品或新上市产品。CRM 为销售人员提供公司营销活动的最新信息，销售人员通过 CRM 提交营销活动的实施报告。

6. 日程安排

与财务人员一样，CRM 也为销售人员提供手机短信和 ToDo-List 的日程安排服务。

7. 产品档案

ERP 中的产品信息定期同步到 CRM 中，通过查看产品目录及对应的库存和价格，销售人员可以合理的维护设置业务机会。

8.3.2 ERP 分析

ERP 系统主要提供给企业内部人员使用，比如：财务部门，生产部门等等。在下面的分析中只分析 ERP 提供给财务部门的功能。

1. 客户信息管理

ERP 是企业 IT 技术的核心部分，所有其他企业 IT 技术都要依赖于 ERP。像 CRM 的客户信息是由 ERP 来提供，这体现了企业稳定的客户信息保存在 ERP 中，而尚未成为客户的联系人等信息则保存在 CRM 系统中。若交易成功，联系人转化为客户，则信息更新到 ERP 中去。

2. 产品库存管理

企业中如何获取最新最准确的库存信息和价格信息是很重要的。只有最新的产品目录，销售人员才能向客户展示企业的技术；只有最准确的库存数量才能对客户需求做出合适的反应；只有最新最准确的价格信息，才能争取到客户。

ERP 系统中提供产品库存管理功能，该功能能够快速的获取当前最新最准确的库存信息和产品价格。

3. 订单管理

与客户建立订单后，由于各种原因，订单需要被审核、被冻结、被处理等。而这个过程是非常复杂的一个业务流程，需要有非常好的控制机制。否则，被埋在订单中的财务人员无法更好、更高效的进行工作。

ERP 系统有专门的订单管理功能，能够很好的控制订单状态的转移，订单的流向，以及订单被通过后的处理。

第九章 用例模型分析

用例模型是对凤凰公司 IT 系统既定功能及环境的定位和描述，并应作为凤凰公司和万合公司之间开发合作的契约。用例模型用作本案分析、设计和测试活动的基本输入。

9.1 参与者列表

销售人员：作为凤凰公司与外界的联络者，在业务流程中负责发现业务机会、提交和维护销售订单的工作。同时，销售人员还应及时维护和更新客户信息、采取一切方法扩展公司的业务渠道和业务范围、跟踪订单的进展情况、尽量满足客户的最新要求。与销售人员相关的企业数据包括：客户信息、业务机会描述、库存和价格、客户认同的销售订单等。

财务人员：是凤凰公司业务运转的中枢，有效连接着销售、生产、供应和研发部门。在一次业务活动中，财务人员应该根据公司的生产和库存状况，批准、延迟或否决销售人员提交的业务机会和销售订单。与财务人员相关的企业数据包括：业务机会、销售订单、库存状况、付款通知单和位于企业内部的各类财务数据。

企业主管：是凤凰公司发展战略和重大事务的决策者，也是企业实施 SOA 的发起者，理应从 SOA 的部署中获利。企业主管需要实时监控公司内部的生产、研发、销售和财务状况、同时对供应商和客户有比较详细的了解。因此，企业主管往往忽略细节数据，而比较关注各种经过汇总和统计的图表。

客户：是凤凰公司价值体现的来源，因而整合后的 IT 环境应该为客户提供相应的服务。客户可以从系统上了解凤凰公司实施的成功案例、主推或最新的产品和服务；并能通过这一渠道提交其个性化的需求。

9.2 用例列表和规约

9.2.1 查看产品和案例

1. 前置条件

客户存在与凤凰公司发生业务关系的动机。

2. 事件流

客户登陆凤凰公司的外部访问端口后，按照时间、地域、业务额（价格）等不同条件查看凤凰公司生产和销售的各种产品及以往的成功案例。

3. 特殊需求

该用例的实现应包含一个智能代理模块，它负责保存和分析用户的访问记录，从而有针

对性的提供产品目录和成功案例。

4. 后置条件

业务机会产生。

9.2.2 生成和维护业务机会

1. 前置条件

商机或其他来源导致产生一个业务机会。

2. 事件流

销售人员利用系统集成的信息智能社区服务（中文智能检索）查看和分析客户背景，进而在 CRM 中增加新客户的帐户（如该客户已存在，则此步骤略），之后新建业务机会并输入基本信息；通过查看产品目录、库存和价格信息及与客户进行交流，销售人员随时修改业务机会的信息并最终确定该业务机会的状态；制约条件（仅考虑库存）满足时，提醒销售人员激活被冻结的业务机会。

3. 后置条件

业务机会状态置为“赢”或“取消”。

4. 特殊需求

该用例要求系统后台运行着一个隐形用例，即定期将凤凰公司的产品目录、库存和价格信息从 ERP 同步更新到 CRM

9.2.3 提出生产建议

1. 前置条件

业务机会因库存不足延迟或取消；销售订单因库存不足被延迟或取消。

2. 事件流

业务机会或销售订单因库存不足被迫延迟或取消时，由销售人员或财务人员负责生成业务失败报告，详述该业务的产生、发展及失败原因。当多个业务因同一产品的库存不足而失败时，系统自动向企业主管提出生产建议。

3. 特殊需求

系统定期汇总业务进展状况，对于失败的业务，根据预置的业务数限额或预计业务总额决定是否向企业主管提出建议。

4. 后置条件

批准或搁置生产建议。

9.2.4 创建和批准销售订单

1. 前置条件

销售人员将业务机会置为“赢”，提请创建销售订单。

2. 事件流

销售人员提交创建销售订单的请求后，系统自动判断库存满足情况：如果库存不满足，销售订单被冻结；如果库存满足，这个请求被发送给财务人员。财务人员负责设置销售订单的最终状态。

3. 特殊需求

财务人员处理销售订单时，如果发现该客户在 ERP 中不存在，需要及时创建或更新客户信息；如果销售订单创建成功，其信息要同步到 CRM 中。

4. 后置条件

销售订单创建成功或失败。

9.2.5 审查业务数据

1. 前置条件

销售、财务人员提交原始业务数据。

2. 事件流

在凤凰公司的日常业务中，销售人员和财务人员按照“业务机会→销售订单”的流程开展工作。随着业务机会或销售订单的成功与失败，一系列业务状态被记录下来。企业主管可以实时监控当前业务机会报表、失败业务机会报表、执行中销售订单报表、成功销售订单报表以及若干阶段性报表中反映的业务信息，从而为凤凰公司制定经营策略和实施研发、供货、生产、销售、财务等环节具体工作提供有价值的参考。

3. 特殊需求

由于万合公司为凤凰公司实施的 IT 整合采用了 SOA 的思想，企业业务流程的调整变得简单易行，企业主管可以灵活的调整业务流中各项任务的内容和顺序。

4. 后置条件

由企业主管做出的各项决策。

9.3 用例模型图

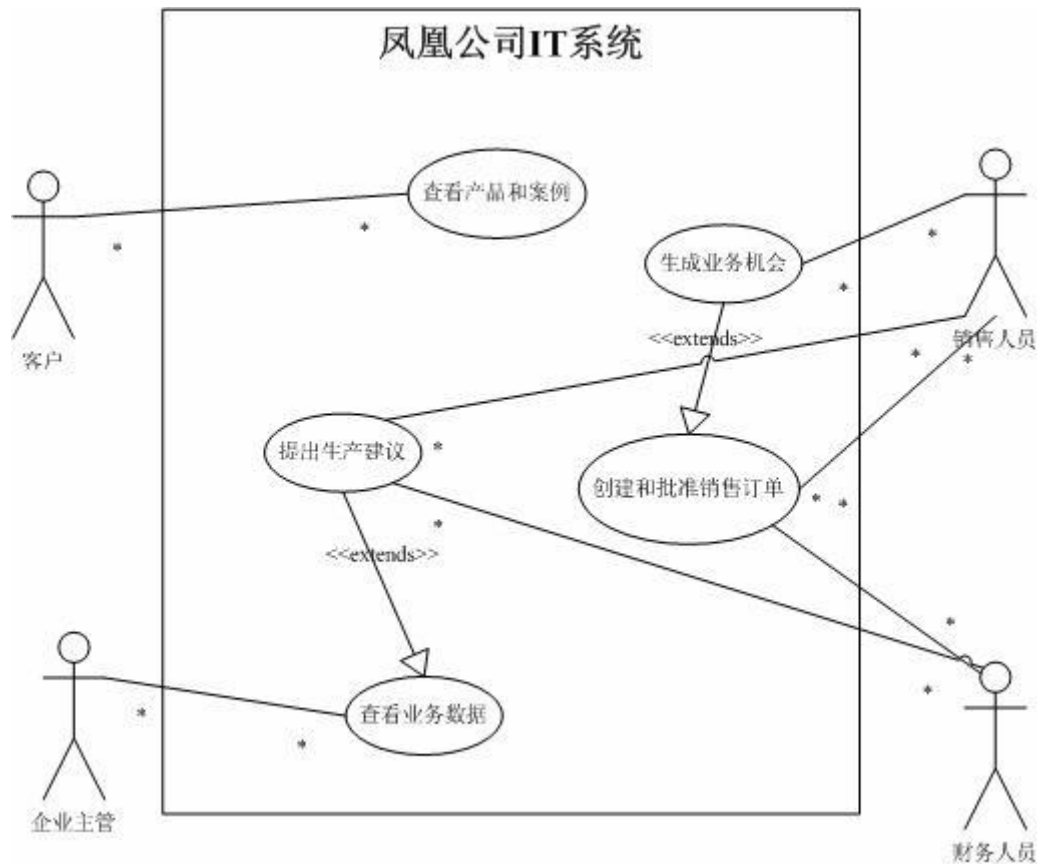


图 8 凤凰公司 IT 系统用例模型

第十章 数据模型分析

10.1 数据汇总

根据对凤凰公司业务流程的分析，IT 系统保存和处理的数据汇总如下：

表 6 凤凰公司数据汇总表

编号	名称	相关业务部门	所处 IT 子环境
D1	业务机会	销售	CRM
D2	客户	销售、财务	CRM \leftrightarrow ERP
D3	产品	销售、财务	ERP \rightarrow CRM
D4	销售订单创建请求	销售、财务	CRM \rightarrow ERP
D5	财务人员日程	财务	ERP
D6	销售订单	财务、销售	ERP \rightarrow CRM
D7	各种业务报表	销售、财务、企业主管	ERP

D8	案例	销售	ERP
D9	生产建议	销售、财务、企业主管	ERP
D10	公司员工	销售、财务、企业主管	ERP

10.2 数据模型构建

在凤凰公司现有的 IT 环境下，各种人员、产品和业务数据被分别孤立的保存在 CRM 和 ERP 中；由业务进展促生的数据流动只能由员工手动执行，因而优化数据流动路径、提高核心数据同步效率进而推动企业业务进展就成为凤凰公司实施 SOA 的主要目标之一。

下图简要的反映了凤凰公司 IT 系统中存在的主要数据及其依赖关系：

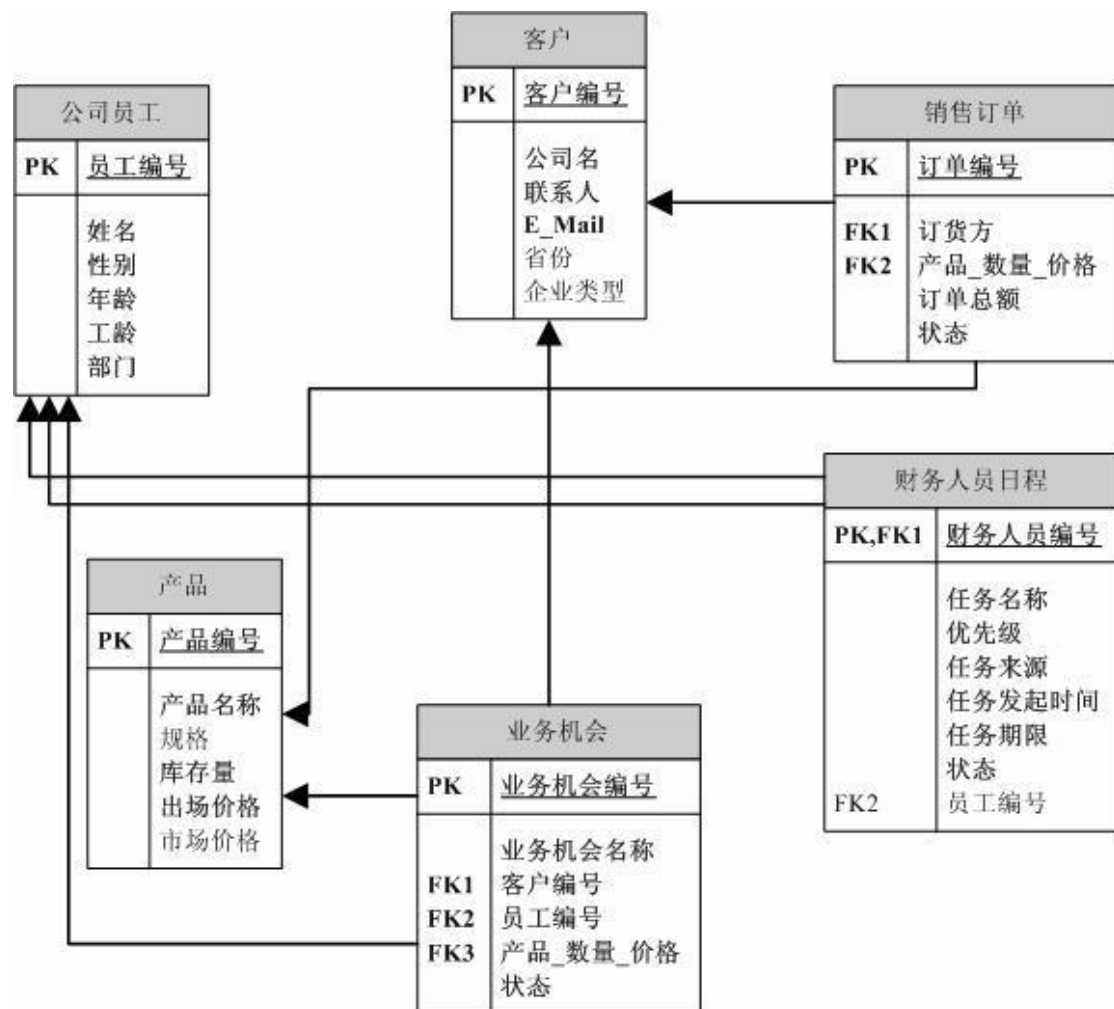


图 9 凤凰公司 IT 系统数据模型

在上图所列的凤凰公司数据集合中，各数据的说明如下：

公司员工——是企业基本组成单元，是企业生产和销售活动的执行者，也是凤凰公司企业价值的创造者。其容纳的信息包括唯一标识公司员工的编号、员工姓名和年龄、员工在凤

凰公司的工作年限以及该员工所属的部门。最简单的，目前我们假设凤凰公司只拥有销售、财务和管理三个部门。销售人员负责与外部环境产生交互；财务人员负责审核各种业务活动；企业管理者则着力于制定凤凰公司的整体战略、监控销售和财务部门的日常工作。

客户——是凤凰公司提供产品或服务的对象，也是实现企业价值必备的外部条件，在实际业务流程中，客户通过与凤凰公司销售人员的交互实现同凤凰公司的业务合作。客户有权获得最新产品信息以及合理化建议，其主要元素包括唯一标识其身份的客户编号、客户公司名称和联系人姓名等，联系人 E-Mail 地址是凤凰公司 IT 系统中辨识和使用客户信息的重要依据；最后，为了恰当的向客户推销产品或服务，记录其地理区域以及企业类型也是十分必要的。

产品——是凤凰公司实现企业价值的载体，产品名称和规格是几乎所有企业系统中标识产品的重要信息。在本案中，特别针对销售活动，产品记录还应包含当前库存量、出厂价格、市场参考价格等信息。

业务机会——是销售人员按照凤凰公司企业规范对其所遇商机的形式化描述，随着销售人员与客户的不断交流，业务机会包含的内容被及时更新，直至销售人员为其设置最终状态。在业务机会的“赢”、“延迟”、“取消”三种状态间销售人员有权进行修改。业务机会应该包含的信息包括：名称、机会发现者、业务对象以及详细的产品、需求量和建议价格列表。

销售订单——是凤凰公司与其客户确认合作关系的物质凭证，一般的它由销售人员发起、由财务人员审核和批准并由企业主管实时监控。对于每一个销售订单，以下信息必不可少：订单编号、客户、所涉产品明细（产品名、数量、价格等）以及最终的状态。

财务人员日程——是为了提高财务人员工作效率所设的功能。其内容包括任务的发起者和执行者、任务名称、任务时限等。优先级的设置便于财务人员在需要同时处理多个事务时可以有所侧重。

第十一章 关键技术架构决策

11.1 服务组件划分

在凤凰公司实施 SOA 战略的过程中，万合公司显然面临着凤凰公司企业主管和业务部门很高的期望值，这就迫切要求在 IT 整合时最大程度的重用凤凰公司现有的应用功能和数据资源，但这种重用面临着如何将高度异构、分布的各个应用（比如 CRM 和 ERP）集成起来的难题，将系统功能定制为若干个服务组件是解决这一难题的有效途径。

在我们的设计中，将凤凰公司内部的各种功能整合为九个相对独立的组件，其定义如下所示：

表 7 凤凰公司系统集成组件列表

组件序号	组件名称	功能描述
1	服务管理组件	该组件处于系统结构的最上层，负责检测、管理、调用和组合系统中的各类服务。下层组件提供的服务须经服务管理组件注册后才能被系统用户使用。
2	ERP 数据访问组件	该组件属于底层数据访问组件，主要负责提供上层组件对 ERP 系统中数据访问的服务。上层组件可以通过本组件访问 ERP 中的客户信息、产品信息和订单信息等。
3	CRM 数据访问组件	该组件属于底层数据访问组件，主要负责提供上层组件对 CRM 系统中数据访问的服务。上层组件可以通过本组件访问 CRM 中的客户信息、联系人信息和业务机会信息等。
4	客户信息管理组件	该组件主要负责管理客户信息。负责 CRM 中增加客户信息、查询客户信息和修改客户信息，同时也负责 ERP 中创建客户信息、核实客户信息（Email 地址核实），以及 CRM 和 ERP 中客户信息的同步。该组件为无状态组件。
5	业务机会管理组件	该组件主要负责管理销售人员发现的可能的业务机会。负责记录业务机会，修改业务机会状态，使销售人员能够同时处理多个业务机会。
6	产品信息查询组件	该组件主要负责提供给销售人员和财务人员关于产品的信息，包括产品清单、库存、报价等信息。对于销售人员需要掌握最近的产品清单和产品报价，便于客户进行沟通；对于财务人员需要有准确的库存数量，以便于对订单进

		行审核和处理。
7	销售订单管理组件	该组件主要负责提供给财务人员销售订单管理的功能，包括销售订单的创建、审查、修正、状态改变。
8	手机短信组件	该组件主要负责提供给销售人员和财务人员即时的体形功能，每次有一个新的任务可以采用手机短信的方式即时的进行通知。对于财务人员，每当有一个新的销售订单需要处理时，就可以发送一条手机短信进行提醒；对于销售人员，当提交的订单状态发生改变时，也会发送一条手机短信进行通知。
9	日程表组件	该组件主要负责对财务人员的事物进行备忘，可以对新任务进行自动添加也可以手动的添加工作列表。当有新的订单需要处理时，除了发送手机短信进行提醒外，会自动在其日程表中添加事务，用以提醒。
10	日志组件	该组件负责记录系统事件。

11.2 ESB 设计

企业服务总线（ESB）是一个基于消息的通信模块。ESB 实现企业内部信息的准确、高效和安全传递。它还可以消除不同应用之间的技术差异，让不同的应用服务协调运作，实现不同服务之间的通信与整合。ESB 是连接企业各种纷繁复杂应用的骨干神经系统，种种不同应用通过它整合在一起，以合理、符合业务需要的节奏连接应用程序和调节消息流量，从而达到提高效率和可靠性、加快响应速度的目的。

本案系统架构中 ESB 主要包含以下模块：

11.2.1 Channels

一个 sender 到 receiver 的虚拟管道。一个初始的服务总线架构不包含任何 channel；我们必须知道各个应用需要怎样通讯，然后才能建立 channel 来完成它。这是一种半集成化的消息传递思想，因为实质上它建立的还是点对点的调用接口，在我们的项目中不建议采用。

11.2.2 Messages

是在总线上传输的不可分割的包，是 ESB 乃至 SOA 的基础单元。为了传输数据，应用必须将数据打包成一个或多个 packets，将每个 packet 包装成一个 message，然后将其传输到一个总线上。同样的，一个 receiver 应用在接收到 message 后必须从 message 中提取出数据才能使用。总线应该能重复的传输 message，直到它成功为止。

11.2.3 Pipes and Filters

最简单的情况下，总线将一个消息直接从 sender 传送到 receiver。然而，通常在消息从 sender 中发出后，receiver 接受到之前，有一些动作需要对 message 执行。举例来说，message 也许需要验证或者转换。Pipes and Filters 架构使用 channel 将多个处理步骤连接起来，它是 Bus 内部模块通道。

11.2.4 Transformation

不同的应用的数据格式很可能不同。为了调节 sender 和 receiver 之间的数据格式不同的问题，message 必须经过一个中介的模块，它执行 Message Translation 的工作，将 message 从一个格式转换成另外一个格式，或转换成一个公共的格式。

11.2.5 Endpoints

大多数的既有应用程序没有内建的能力来同一个 message 系统交互。因此实现 SOA 或 ESB 时必须包含一个中间层，它知道应用系统如何工作，也知道服务总线如何工作，并桥接两个系统。在 Bus 的上下两端各有一组并列的 Endpoints，它能够使得应用发送和接受 message。

11.2.6 Event D&P

由于 SOA 实质上是时间驱动的系统架构，任何包装好的服务只有在实际应用发生时才会被调用，所以事件检测和处理模块是服务总线的核心处理模块。

11.2.7 System manager

作为一个大型的消息集成系统，其面向消息的、异步、低耦合的本质使得系统更加难以调试，运行期的状态也难以跟踪，所以，我们必须有强有力的手段进行系统的运行期管理和监控，同时最好能够在运行进行动态更新，以保障系统的强壮性。

11.3 智能社区服务整合

本节假设的两种智能社区服务分别服务于销售人员和所有企业员工，参赛团队 51START 本身就拥有这两种服务的开发经验；在复赛阶段，如果不能从 Internet 上找到符合

要求的服务，我们有能力独立完成相关开发工作。

11.3.1 中文智能检索

1. 服务说明

在确认和提交业务机会前，凤凰公司销售人员利用中文智能检索服务查询客户的背景信息，包括客户联系方式和地址、股票表现、主要业务案例、信誉评价等，从而为开展业务提供帮助。

2. 输入信息

客户公司名称或主要负责人姓名。

3. 服务结果

经过聚合的客户背景信息（不是零散的网页链接）。

11.3.2 文档图像管理

1. 服务说明

凤凰公司作为一个中等规模的医疗器械企业，其内部流通着多种类型的文件。作为企业信息化的重要标志，纸质办公必然被电子化办公所取代。在这一过程中，能否合理、快速的管理公司的各种文档图像文件，直接影响着企业各部门的沟通是否顺畅、企业管理是否高效。在实施 SOA 的过程中，我们使用成熟的文档图像管理系统（DIM）协助企业主管、销售人员和财务人员快捷的使用和管理各种电子文件。

2. 输入信息

流通在企业内部的电子文件。

3. 服务结果

个性化的 Cabinet→Folder 目录结构和灵活易用的索引机制。

第十二章 系统架构分析

12.1 系统结构图

经过对凤凰公司 IT 资源整合需求的分析，以及应用 SOA 思想设计和讨论必要的服务与组件，我们绘出凤凰公司整合后的 IT 系统结构如下图所示：

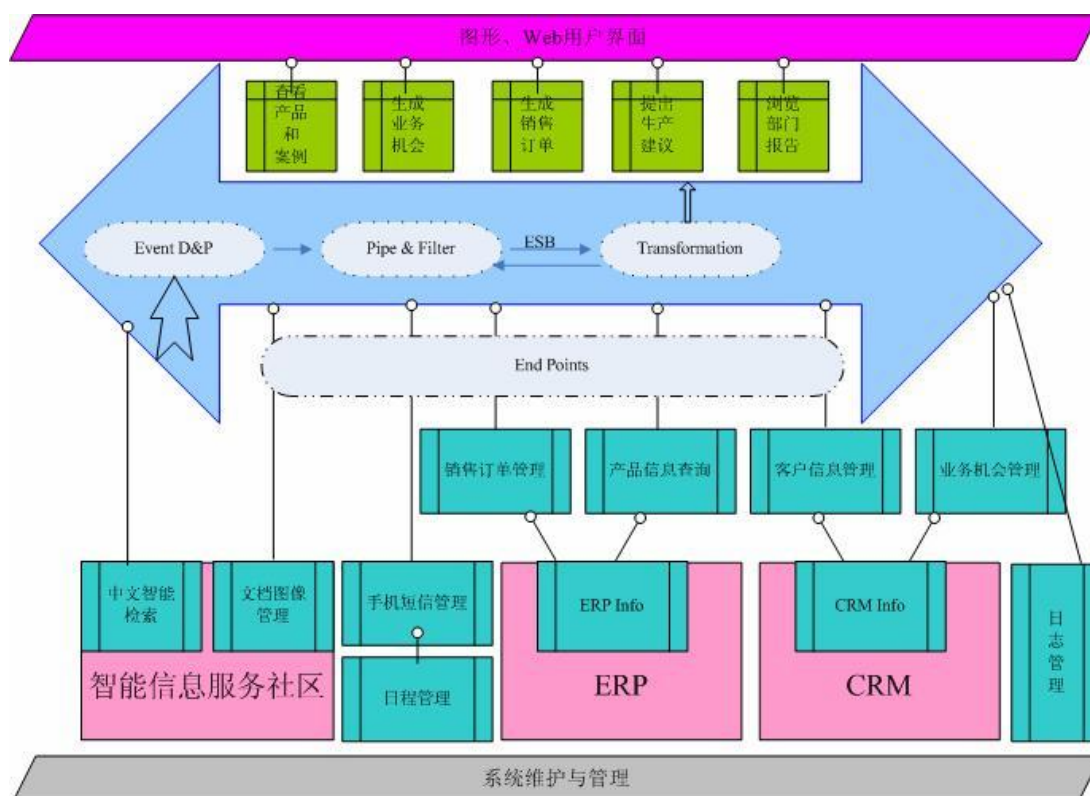


图 10 系统架构图

如下五点简要介绍了本案例中的主要架构元素以及他们之间的工作关系：

1. 系统维护与管理 ()

系统维护与管理模块位于系统结构的最低层，它负责维护上层各模块的组织与连接，并采用多种机制保证凤凰公司 IT 系统性能。

2. 现有 IT 资源 ()

现有 IT 资源包含运行在企业内部的 ERP、CRM 以及 Internet 上的智能信息服务社区。其中，ERP 是企业主管和财务人员的工作环境，而销售人员主要工作在 CRM 上。经本案整合后，ERP 与 CRM 中的信息均以统一的平台供凤凰公司员工使用。智能信息服务社区中提供中文智能检索和文档图像处理两个主要功能。

3. 各类组件 ()

这是本案的工作内容之一，这些组件大致分为四类：

ERP 信息抽取、CRM 信息抽取等组件直接访问和控制现有 IT 资源，是整合底层数据和上层应用的主要通道；

销售订单管理、产品信息查询、客户信息管理和业务机会管理等组件利用逻辑关系组织和调度基础数据，以符合业务流程的方式向上提供服务；

手机短信、日程、日志等组件则相对独立于凤凰公司的原有资源，它们为各部门员工高效工作提供保障；

服务管理组件位于 IT 系统的上层，负责检测、组织和协调松散耦合状态下的各类服务。

4. ESB ()

ESB 是系统内部信息传递的通道，采用“总线”这样一种模式来管理和简化应用之间的集成拓扑结构，以广为接受的开放标准为基础来支持应用之间在消息、事件和服务的级别上动态的互联互通。

在本案中，ESB 利用 End Points 封装和整合下层组件提供的服务为系统支持的统一格式，经事件检测和处理、管道和过滤、翻译之后以规范的格式提供给用户使用。

5. 灵活定制的服务及其界面展现 (+)

现有 IT 资源经底层组件的封装和 ESB 的传输后，以面向企业业务的服务形式提供给用户直接调用。服务之间以松散耦合关系互相连接，而且可以随意的扩充和调整调用的顺序；当凤凰公司业务流程发生变化时，企业主管有权新增服务或改变现有服务的顺序而不需对 IT 系统进行大规模修改。

通过对上述系统结构的分析可以发现，经“以服务为中心进行企业整合”后，我们可以为凤凰公司实现一种技术透明、结构灵活、高度重用和渐进式集成的新型企业 IT 系统。

第五篇 组件设计

第十三章 组件解析

13.1 什么是组件

组件是可执行软件的一个可分离的部分，作为一个单元来说也是有意义的，可以与其他组件实现互操作。只能通过组件的接口来访问它，在进行了必要的安装和配置过程之后，它就能以接口规定的方式来使用。为了能与其他组件一起工作，必须能够得到其接口的细节。

13.2 组件的类型

一般应用可以划分为三层。自顶向下分别是：表示层、业务逻辑层和数据访问逻辑。三层之间有着支撑和调用的关系。

根据组件在应用中所处的不同层次，可以将其分为以下几类：

1. 业务组件

该类组件包含那些属于业务逻辑层的逻辑。即包含核心业务相关的逻辑，代表最终用户认为真正有用的东西。它们负责确保任务规则得到遵守，负责维护业务数据的完整性。它又包括了“ workflow 或控制逻辑”和“业务对象”两种类型。

2. 应用组件

该类组件包含表示层逻辑。负责管理用户界面，与业务对象进行交互，从而提供满足用户需求的应用。

3. 数据访问组件

该类组件包含“业务逻辑层的数据视图与实际存储数据间的映射”和“以特定数据库或文件所要求的方式实现对物理数据访问”的功能。

4. 工具组件

该类组件被几个层次都用到，最好将它们作为软件基础设施来提供。比如说每个系统中都需要的日志组件。

13.3 组件定义方法

每种组件模型都有自己的组件定义方式。COM 采用 IDL，SCR 使用 WSDL 和 JAVA 接口。在本文档中，我们采用较为简单的，但是每种组件都必备的几个定义要素。

组件定义包括以下部分：

◇ 名称与简短描述

组件的名称以及对组件简短的描述，能够

◇ 提供的接口

说明组件提供的接口。接口中包含的服务将被列出。在有多个接口的情况下，每个接口都需要命名，并简单的进行描述，同时列出其服务。

◇ 所需的接口

指明该组件需要用到的一些接口，不必指明实现这些接口的组件。

第十四章 组件设计原则与方法

14.1 组件层次化原则

根据组件提供服务所属应用的层次，将组件也分为不同的层次。从而处在较高层次的组件可以使用较次层次组件所提供的服务。

采用这个原则，进行自底向上的开发。从现有的 IT 技术出发，首先构建较底层的、互相独立的、功能简单的、规模较小的组件。然后，基于已有的组件构建较高层的、功能复杂的、包含业务流程的、规模较大的组件。

14.2 粗组件粒度原则

组件粒度可以用组件所提供的功能数量来度量，特别是用功能点的数目来度量。这样对于使用不同的标称语言的代码都适用。

在设计组件的时候，组件的粒度是我们很关注的问题。因为复用大的组件更节省工作量，所以大都组建对业务来说具有更大的价值。小的组件更有可能被经常用到，但使用它节省的工作量较小。

SOA 提倡的使用粗粒度的组件，即大的组件。从组件设计角度来讲，符合了更好的重用性和节省工作量的原则。

14.3 定义合理组件范围原则

定义组件的范围是一个很困难的工作。每个组件的功能不能与其它组件相互覆盖，这样会造成冗余，同样更不能出现所有组件都覆盖不到的功能。

最简单的一个方法是把组件功能最小化，即每个组件实现的功能不能再细化。但是，基于组件粒度的原则，这种方法无法获得很好的组件复用性，而且也丧失了组件功能的完整性。

用合理的范围定义每个组件，能够达到没有功能冗余又有良好复用性的组件。

14.4 无状态组件原则

组件有两种类型：有状态和无状态。

在组件设计中，应该尽可能的将组件设计为无状态组件。这样不必保留已有的会话状态，每次交互必须提供完整的参数。可以有效的提高组件的重用性。

14.5 包装现有应用的方法

组件的开发可以基于现有的软件，使得包装后的组件具备原有软件功能的同时也具备组件的标准接口。

包装组件一方面提供了标准的组件接口，同时又以特定的方式与现有的应用代码实现接口。包装组件与它使用到的现存应用代码共同组成了一个虚拟的组件。

在 SOA 开发中，绝大多数情况都是包装现有的应用。像比赛题目中对 ERP 和 CRM 的包装就是一个很典型的例子。

第十五章 组件设计

组件设计基本按照第 2 节中的原则与方法。总体的组件结构图如下：

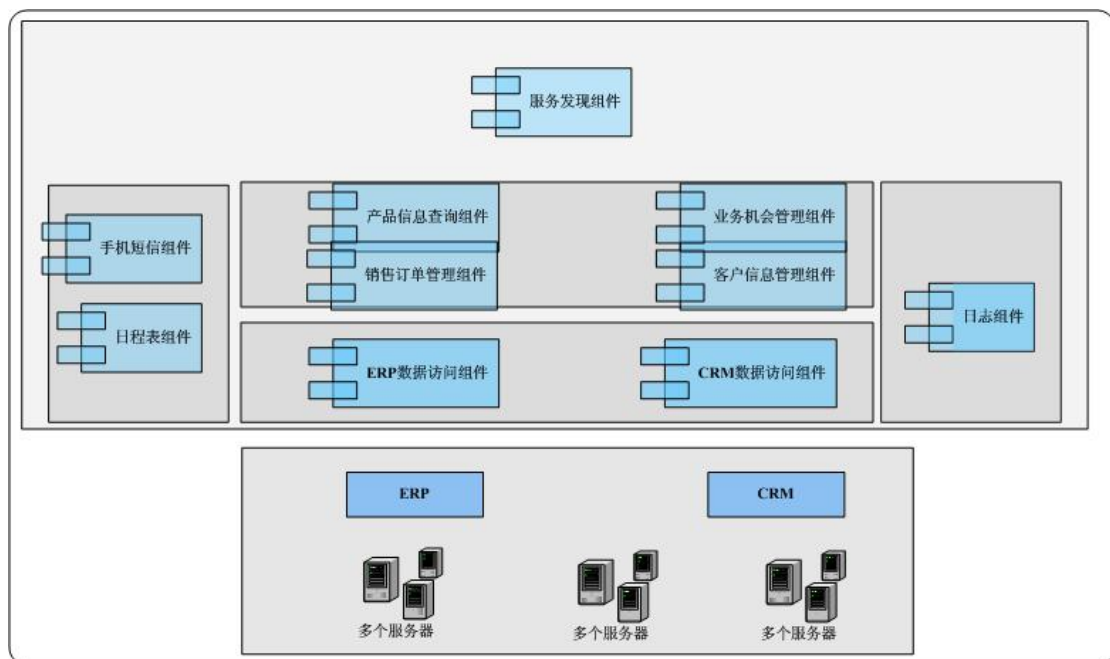


图 11 组件结构图

以第二篇图 7 的凤凰公司业务流程图为基础，组件在凤凰公司业务流程中的位置如下图：

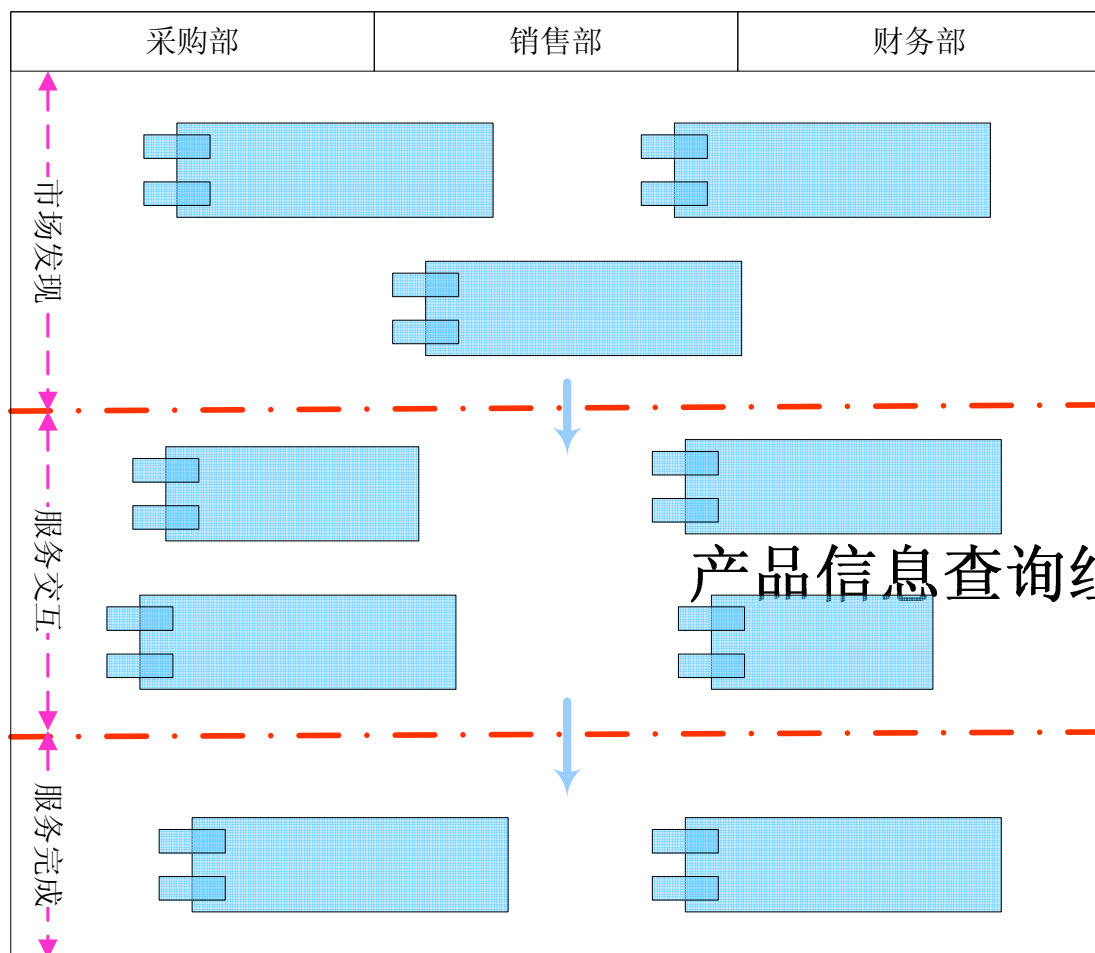


图 12 组件在凤凰公司流程中的位置

15.1 ERP 数据访问组件

手机短信组件

本组件属于底层数据访问组件，主要负责提供上层组件对 ERP 系统中数据访问的服务。

上层组件可以通过本组件访问 ERP 中的客户信息、产品信息和订单信息等。

15.1.1 提供接口

- A. 获取 ERP 客户信息
- B. 增加 ERP 客户信息
- C. 更新 ERP 客户信息
- D. 获取 ERP 产品信息
- E. 增加 ERP 产品信息
- F. 更新 ERP 产品信息
- G. 获取 ERP 订单信息

产品信息查询组件

销售订单管理组件

H. 增加 ERP 订单信息

I. 更新 ERP 订单信息

15.1.2 需要接口

A. 企业原有 ERP 系统对外提供的数据访问接口。

15.2 CRM 数据访问组件

本组件属于底层数据访问组件,主要负责提供上层组件对 CRM 系统中数据访问的服务。

上层组件可以通过本组件访问 CRM 中的客户信息、联系人信息和业务机会信息等。

15.2.1 提供接口

A. 获取企业客户信息

B. 获取 CRM 联系人信息

C. 增加 CRM 联系人信息

D. 更新 CRM 联系人信息

E. 获取 CRM 业务机会

F. 增加 CRM 业务机会

G. 更新 CRM 业务机会

15.2.2 需要接口

A. 获取企业 ERP 客户信息

15.3 客户信息管理组件

该组件主要负责管理客户信息。负责 CRM 中增加客户信息、查询客户信息和修改客户信息,同时也负责 ERP 中创建客户信息、核实客户信息 (Email 地址核实),以及 CRM 和 ERP 中客户信息的同步。该组件为无状态组件。

15.3.1 提供接口

A. 获取企业中现有客户信息

B. 在 CRM 中增加新客户

C. 更新客户信息

D. 同步 CRM 与 ERP 客户信息

15.3.2 需要接口

A. 数据访问层的获取企业客户信息

B. 数据访问层的增加 CRM 客户信息

C. 数据访问层的更新 CRM 客户信息

D. 数据访问层的增加 ERP 客户信息

15.4 业务机会管理组件

该组件主要负责管理销售人员发现的可能的业务机会。负责记录业务机会，修改业务机会状态，使销售人员能够同时处理多个业务机会。

15.4.1 提供接口

- A. 创建业务机会
- B. 修改业务机会状态

15.4.2 需要接口

- A. 数据访问层的增加 CRM 业务机会
- B. 数据访问层的更新 CRM 业务机会

15.5 产品信息查询组件

该组件主要负责提供给销售人员和财务人员关于产品的信息，包括产品清单、库存、报价等信息。对于销售人员需要掌握最近的产品清单和产品报价，方便于客户进行沟通；对于财务人员需要有准确的库存数量，以便于对订单进行审核和处理。

15.5.1 提供接口

- A. 查询产品清单
- B. 查询产品库存
- C. 查询产品报价
- D. 查询与需求匹配的产品

15.5.2 需要接口

- A. 底层数据访问层的获取 ERP 产品信息
- B. 底层数据访问层的获取 CRM 订单信息

15.6 销售订单管理组件

该组件主要负责提供给财务人员销售订单管理的功能，包括销售订单的创建、审查、修正、状态改变。

15.6.1 提供接口

- A. 创建销售订单
- B. 审查销售订单
- C. 修正销售订单
- D. 销售订单状态控制

15.6.2 需要接口

- A. 底层数据访问层的增加 ERP 订单信息
- B. 底层数据访问层的更新 ERP 订单信息

15.7 手机短信组件

该组件主要负责提供给销售人员和财务人员即时的体形功能,每次有一个新的任务可以采用手机短信的方式即时的进行通知。对于财务人员,每当有一个新的销售订单需要处理时,就可以发送一条手机短信进行提醒;对于销售人员,当提交的订单状态发生改变时,也会发送一条手机短信进行通知。

15.7.1 提供接口

- A. 发送手机短信

15.7.2 需要接口

- A. 电信服务提供商提供的短信发送服务接口

15.8 日程表组件

该组件主要负责对财务人员的事物进行备忘,可以对新任务进行自动添加也可以手动的添加工作列表。当有新的订单需要处理时,除了发送手机短信进行提醒外,会自动在其日程表中添加事务,用以提醒。

15.8.1 提供接口

- A. 添加事务
- B. 更改事务状态
- C. 删除事务

15.8.2 需要接口

- A. 销售订单控制组件中的销售订单控制接口

15.9 日志组件

该组件主要负责整个系统的日志记录工作。利用日志,可以对系统性能,功能进行评测,也可以对可能产生的错误进行恢复。

15.9.1 提供接口

- A. 记录日志接口
- B. 日志分析接口

15.9.2 需要接口

无

15.10 服务发现组件

该组件主要负责整个系统中所有服务的发现。利用该组件，各种组件和服务可以发现其他的组件提供的服务，从而可以使用其他组件提供的服务完成特定的功能。

15.10.1 提供接口

- A. 判断服务是否存在
- B. 获得服务调用接口

15.10.2 需要接口

无

第六篇 项目实施计划

第十六章 项目团队组成及分工

16.1 项目团队简介

51START 团队是来自南开大学，首先解读一下我们的团队名称 51START

第一重解读：Concept——对 SOA 概念的分解和重述

S、T、A、R、T 其实是我们这个团队对 SOA 概念的某种理解：

S——Service，它既可以理解为普通企业提供的各种业务服务，也可以理解为 IT 技术中的“服务”，我们生活在一个服务经济的时代，唯有服务才有价值。

T——Technology，它就是指技术，小到普通 OA、数据库，大到复杂的 ESB 和 EIP，技术可以提高工作的效率、优化工作的流程，技术优势是形成竞争优势的基础。

A——Architecture，它是指一种组织结构，既包括企业现有的业务流程架构，也包括对应的 IT 技术架构。

R——Reorganization，它是指一种为了优化和增值而做的重组，其实 SOA 以及 SOI 概念在某种程度上就是一种对现有的业务、技术、流程所进行的重组。

T——Trend，它是指趋势，SOA 正是一种发展的趋势，它代表着企业信息化的潮流所向，唯有把握趋势才能拥有先机。

同时，我们把 SOA 理解为：采用合理架构，对技术和服务进行有效重组，它代表企业信息化发展的趋势，而 51START 就可以理解为“我要 SOA”！

第二重解读：Team——对团队合作的解读

“51”两个数字，其实描述了我们团队的基本情况，我们的团队一共有 5 个人，但却有着 1 个共同的信念，那就是为了一个共同的目标而努力，这个共同的目标就是以参加比赛的方式领略 SOA 的奥妙，体会管理与技术相互结合的内涵。

第三重解读：Passion——对本次比赛以及参赛目标的解读

本次大赛的重要目的，是普及并弘扬 SOA 的概念，从而掀起对 SOA 概念、技术以及各项工作的推广。51START，其实用中文来说就是“我要开始”的意思，它是一种召唤、一种启迪，无论对技术厂商、还是对普通企业，以及所有即将走向社会的同学来说，面对新的应用潮流和技术发展趋势，我们都将踏上新的征程！51START，也是我们团队的一种宣言，昭

示着我们的存在，展现着我们的热情，START 能够给人以希望和力量，对我们团队而言，即将开始的并不是艰苦的比赛，而是充满智慧的头脑风暴、充满乐趣的探索旅程、充满信任的团队长征！

51START 是一个概念、是一个团队、是一个充满激情的召唤！

16.2 项目团队组成

指导老师：

史广顺，南开大学信息技术科学学院副教授，机器智能研究所科研工作负责人；

团队队长：

卞晓青，南开大学商学院企业管理系 2005 级博士生

团队成员：

杨巨峰，南开大学信息技术科学学院计算机软件专业 2006 级博士生

宫大伟，南开大学信息技术科学学院计算机软件专业 2005 级硕士生

张维，南开大学信息技术科学学院模式识别与智能系统专业 2005 级硕士生

16.3 项目团队分工

51START 中的五个字母，其实也代表着五个人的基本分工。

Service——卞晓青主要负责提炼抽象企业的业务流以及做好业务发展规划，业务最终会转化为各种各样的服务。

Technology——宫大伟主要负责学习并积累 SOA 相关的底层技术，分析并选择最佳的技术实现方式。

Architecture——杨巨峰主要负责参赛题目中需要做的系统架构。

Reorganization——史广顺老师主要负责把大家的经验和能力重新组合，从而满足参加大赛的各种要求。

Trend——张维对技术发展趋势有很深的把握，负责判断我们的具体工作是否符合 SOA 概念的内涵，同时提出创新的发展思路。

以上是题目解读及设计阶段的分工，项目实施阶段由史广顺老师任项目指导，由卞晓青和杨巨峰任联合项目经理，由杨巨峰、宫大伟和张维负责项目实施。

第十七章 项目实施的环境及平台

操作系统: Suse Linux 9.3

架构工具: Rational Software Architect

业务流程支持: IBM WebSphere Process Server V6 和 IBM WebSphere Integration Developer V6

开发工具: Rational Application Developer

应用服务器: IBM WebSphere Application Server V5.1

ERP 支持: 用友 ERP/NC

CRM 支持: TurboCRM

第十八章 项目任务实施时间计划

由于目前项目环境尚未完全确定，加之不能确定项目是否能获得实施，所以目前制定的项目实施计划粒度相对比较粗，在正式实施过程中还会加以细化和调整，以期达到较好的项目进度控制效果。项目实施计划甘特图如图 所示。



图 12 项目进度控制甘特图

第十九章 项目风险分析及规避方案

19.1 SOA 理解不深，项目设计存在偏差

风险可能性：小 风险影响力：大

由于本项目中设计的概念和方法相对较新，可能存在对题目解读不准确导致项目设计有误的情况。在题目解读和设计阶段，我们力求大量阅读和理解以 IBM 为主的行业领导厂商的 SOA 理念并通过多次头脑风暴讨论题目及设计方案力求准确切题。同时由于如果进入复赛在北京还有答辩阶段，所以相信这种风险发生的可能性在 IBM 公司、所有参赛团队及我们自身的多方努力下基本上会被规避。

19.2 技术实现平台掌握不熟练，影响项目开发进度

风险可能性：中 风险影响力：中

团队通过自我解剖发现目前存在对实现平台和工具掌握不够熟练的情况，一则现在就需要不断加强学习，二则如果进入决赛会在 IBM 公司技术人员的指导和帮助之下进行项目实施，相信凭借团队中高素质的人员的学习能力完全可以尽快掌握并迅速进入实施者的角色。

19.3 实施过程表明技术实施工作量很大，进度可能需要拖延

风险可能性：大 风险影响力：小

出于项目的需要，在设计阶段会尽量进行完整全面的设计及创新思维的引入，因此项目设计目标本身很可能存在实施工作量过大的风险，目前制定的实施方案基于倒计时的方式进行规划，技术实现的工作量并未细化考虑所有可能的细节，经过系统设计阶段的汇总分析，可能存在技术实现工作量超出预计的情况，这会直接导致后续时间进度有所拖延。针对此类问题，实施团队将对项目各个模块进行重要程度进行划分，保证高效保量完成核心模块，使系统基本稳定运行，对于其他附属模块给出实现方案，最大限度的保证该项目顺利完成。

19.4 技术人员和测试人员的主观失误

风险可能性：中 风险影响力：中

在技术实现阶段，由于技术人员未必完全了解需求或者缺少协作沟通，可能出现技术实现机制不正确、功能实现目标存在偏差等情况。针对此类风险，技术开发团队将在内部进行严格的项目实施管理手段，并通过向 IBM 公司专业工程师进行咨询，保证技术实现过程的准确和完整。

第二十章 项目验证和测试草案

20.1 项目验证工作详细说明

1. 建立完整统一的 Code Base，实现规范的源代码控制

本项目时间进度很紧张，为了保证技术实现的高效，必须首先建立规范完整的源代码控制机制，我们拟使用 CVS 建立完整的源代码控制环境，及时记录并保存各个阶段的代码版本。

2. 制定完整的代码注释和调试规范，实施内部 Code Review 机制

“写代码容易调代码难”，很多项目的进度拖延主要是卡在了代码调试上，为了降低代码中 Bug 出现的可能性，同时又保证源代码的可复用性，我们将制定规范完整的代码注释标准，所有技术人员都将遵守该标准进行技术实现工作。

同时我们将安排技术人员对他人的代码进行 Code Review 工作，在将编写的代码添加到 CVS 之前，我们将最大限度保证代码实现的准确和完整。

3. 遵循系统设计的结果，完整精确的实现设计结果所规定的技术工作内容

设计过程中清晰界定系统结构、内部外部接口、数据流程和控制流程，技术实现阶段必须严格遵循设计阶段的工作成果，使用最高效的实现手段以达到项目预期的各项目标。我们将通过内部 Code Review，单元评测与调试来保证技术实现的精确。

20.2 内部评测工作详细说明

1. 建立完整的评测发布环境，针对性的提供评测工具

所谓“评测发布环境”是指为了满足内部评测需要而搭建的软硬件平台，在这个评测发布环境中有很多用于测试和调试的辅助工具，用以实现评测过程的自动化，Bug 信息的管理和跟踪以及获取功能和性能的评测结果。该环境的建立是保证评测过程科学合理的重要基础，该环境的建立将由实施团队主要负责搭建并咨询和申请 IBM 公司的专业工程师的协助。

2. 评测方案的制定和评测样本的选择

完整的系统评测方案将在系统设计完成之后才能确定，评测方案经过开发团队与 IBM 专业工程师双方的交流讨论，评测方案中包括了评测案例的制作规范、评测方法的选择、评测参数的设定与计算标准、评测步骤的实施。

评测样本的选择包括评测样本分布情况的设定、典型评测样本的手动制作、海量评测数据的自动化制作等。

3. Bug 库的建立和 Bug 跟踪管理

在评测过程中发现了各种确认的 Bug 之后，需要将这些 Bug 保存在 Bug 库中，以供技术团队及时修改，测试人员又要及时的跟踪 Bug 变化情况，以保证 Bug 的最终解决。依靠 Word、Excel 等文档只能记录 Bug 信息，难以跟踪 Bug 的变化情况。实施团队将自行建立完整的 Bug 库，实现对 Bug 的科学管理。

第七篇 结语：超越 SOA

SOA 以前的时代属于社会主义初级阶段，讲究“按劳分配”，管理上侧重哪方面就加强哪方面，比如 ERP、OA、CRM 都是各有分工，企业要想获得更高效率的 IT 环境，那就需要多部署几个专用系统。SOA 以后的时代属于美好的共产主义，讲究“按需分配”，不再针对性的部署具体业务系统，而是把所有需要的都放在一块，想要啥就拿啥，构建企业信息化的“和谐社会”。

SOA 本身采用了松耦合关系的体系结构，每个服务组件之间的关系不再是紧密连接的了，其实相比较而言，SOA 在获得松耦合关系的同时，其通讯机制的设计却使得软件系统的设计与实现变得更加严格了，以前做 MIS 只需要严谨的需求分析就行了，现在做 SOA 可不一样，要是做需求分析，估计客户都不知道有啥需求；SOA 架构师需要的是深入现场了解企业，去做业务抽象，去做 workflow 封装，去做标准化通用接口，SOA 实质上需要的是企业管理现代化，IT 技术不过是手段而已。换句话说，从我们的观点来看，系统架构的变迁没有本质上改变系统的复杂程度变大的趋势，只是适应性地缩小了这种复杂度增加的幅度，只是逐渐把复杂度因子从内部向外部移了。从热力学第二定律的角度来思考这个问题，软件系统的熵（混乱度或复杂度）是在不断变大的，软件熵包括两个方面的内容，我们可以暂时把它们称为“内熵”和“外熵”，就目前软件系统架构的发展趋势来看，其“内熵”，即组件内部的关联度，并没有发生明显的变化；而“外熵”，即组件之间的关联度，是不断增大的。对于这个结论可以得到如下的解释，现在的每个组件其实就相当于原来紧耦合关系下的独立软件系统，因此其软件熵并无明显的变化；而组件之间的“外熵”却随着组件的不断独立和增多而越来越大；唯一可以使软件熵降低的因素来自于组件之间通讯协议的限制，但是这种限制只是使组件之间的通讯得到尽量统一，它虽然在一定程度上减弱了“外熵”但并不足以起到完全消除“外熵”增长的程度。由此可见，熵变化的时间箭头理论在软件系统架构发展过程中同样是有有效的。它也成为了我们合理地预测软件系统架构的未来发展趋势的一个参考依据。

所谓超越 SOA 就是研究未来 SOA 将向何处发展，如果想做到超越 SOA 就必须了解 SOA 对于未来发展趋势的不适应性。在 SOA 中系统服务总线 ESB 是其最核心的部分，所有的服务都是要挂接在其上，这使得 SOA 本身集中化、中心化。软件外熵就全部体现在了 ESB 的组织上，而软件外熵是不断增大的，这就必然会使 ESB 分崩离析成为分布式的服务结构组织方式。其实这种“单元简单、组织复杂”的系统于人类自身就有最明显的体现——神经系统，

人类的神经系统可以说超级强大和复杂的，但是每个单独的神经细胞却又是及其简单的，它强大的记忆、联想、推理、计算等能力都是由复杂的组织结构支撑和完成的，我们不得不承认人类的神经系统是经过数百万年进化得到的内熵及其小而外熵及其大的典型系统。目前软件系统尚不足以进化到如此程度，如真的如此，恐怕呈现在我们面前的就是如骇客帝国中的母体 Matrix 一般的庞然怪物了。

其实这种从紧到松进而形成“耦合性外化”的转化形式不仅仅体现在神经系统、软件体系架构上，在人类社会发展的历史进程中也可以找到蛛丝马迹。随着人类社会的发展，分工越来越细化，每个人所能熟练掌握的领域也越来越狭窄，这种趋势造就了每个人都逐步成为了社会发展的“螺丝钉”，即缺了你照样跑不会影响大局的变化。不得不承认英雄人物出现的可能性也越来越小了，因为英雄人物需要对整个社会层面的各个领域施加影响，这就需要他能对各个领域都有所涉及，这在目前来讲是及其困难甚至就是不可能的，这也是为什么几乎所有的机构都在强调“Team Spirits”，脱离了团队的人是无法干成任何事情的，就像这次大赛一样，我们必须团结如一人才能发挥 $1+1+1+1+1>5$ 的力量！

以上是从“模块”组织方式即软件外熵的角度考虑的问题，那么从软件内熵的角度考虑每个“模块”在未来会有什么样的变化呢？又是怎样超越 SOA 的呢？SOA 中的每个“模块”就是服务，现在要考虑的是这种划分是不是完全合理呢？明显在某些服务中是可能存在有交叉的，举个简单的例子，财务管理和库存管理中都存在并发控制的问题，不可否认这两种并发控制的对象是不同的——分别是资金和库存，但是明显其控制机制应该是一致的。如果分开实现部署的话，那么显然每个服务的重用性是会降低的，因为它只能在也需要并发控制的地方才可以复用，况且如果并发控制机制不同的话，所有模块中所有的并发控制功能就要全部都重新修改才能移植使用。那么如果将这种相应的 Aspect 从每个实体对象中剥离出来形成独立的 Aspect 对象，无疑是降低了实体对象（也就是我们现在所说的“模块”）的耦合度，同时将这种耦合外化至整个系统之中。

说了这么多，其实所谓超越不过是在现有事实基础之上通过对规律的总结进而对未来的的一种预测，这个世界上是没有人知道未来的，只能是预测未来。遗传还有变异呢，更何况这里预测的是飞速发展、变化多端的 IT 产业，预测需要事实来证明，让我们拭目以待，目前是要做好现实的工作，下面将正式进入我们的业务、服务、架构、组件分析和设计过程。