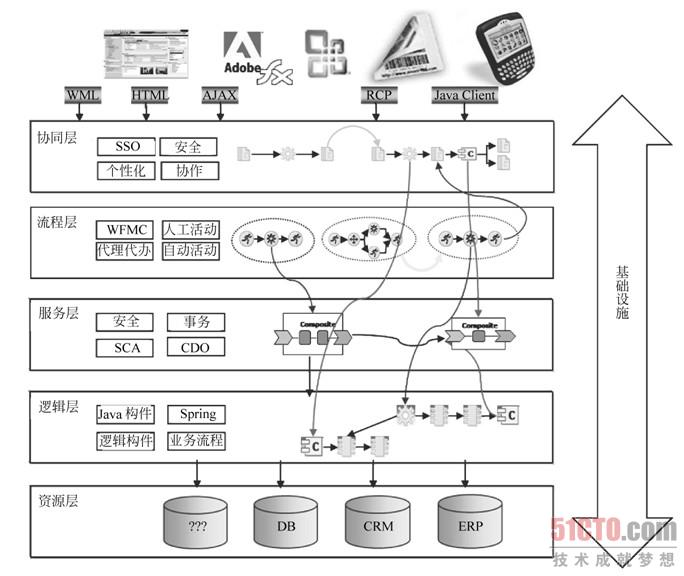
作为企业服务架构的一种体现形式，SOA技术参考架构自顶向下分为协同层、流程层、服务层、逻辑层和资源层等5个层次

[](http://images.51cto.com/files/uploadimg/20091228/222310770.jpg)

SOA技术分层架构

（1）资源层。包含现有的自定义构建的应用程序（遗留系统），现有的CRM和ERP打包应用程序，以及原有的面向对象的系统实现、业务智能应用程 序。SOA利用现有系统并且用基于服务的集成技术来集成它们。资源层解决如何整合数据的问题，需要通过一个统一的数据编程模式统一对不同数据源的访问。

（2）逻辑层。逻辑层实现了具体的业务逻辑，包括UI逻辑和后台逻辑。逻辑层由多个构件组成，这些构件将以可插拔的方式部署，使用AOP、依赖注入的方式编程，提供逻辑的编排能力。

（3）服务层。服务层将应用系统提供的逻辑以标准化的方式暴露出来，使开发者不需要关心逻辑的对外协议、逻辑的实现方式、逻辑的部署位置，并提供事件的方式降低逻辑间的耦合度，为非侵入式的操作提供基础。

（4）流程层。流程层维护跨系统之间的业务状态，企业应用的核心是业务流程，流程包括端到端流程和人工参与的流程，流程会产生任务，推送到工作平台。流程把企业中多个应用连接起来。

（5）协同层。协同层为用户提供了一个统一的交互门户和工作平台，通过RIA（Rich Internet Application）的方式在应用程序人机接口或者表现层来利用 Web 服务，以服务的形式进行界面的组装和重用，提升用户体验，用户通过协同层更容易与其他人进行协作，例如即时通信、查看任务列表、查看发布信息，也能够把已 有数据、服务或界面快速组合到新应用中。通过协同层，用户不再与多个孤立的系统进行交互，而是面对一个有机的整体。

基础设施包括企业服务平台、流程平台、业务工作台等内容，将在9.1.2节"基于SOA的企业应用基础设施"中详细介绍。下面将对技术架构中的5层进行较为详细的介绍。

**1．资源层**

基于SOA的应用在资源层很重要的一点是考虑如何整合遗留IT资源，遗留IT资源主要包括遗留数据和遗留系统两种类型。资源层的核心功能是整合这些 企业资产作为可复用的资源。对于遗留数据，平台通过SOA中的SDO标准将其封装成数据实体，对于遗留系统的集成，主要是通过系统功能进行服务抽取，生成 一系列的服务构件。

资源层的数据标准化提供统一的数据格式和统一的数据模型，有两层含义。

通过主数据管理提供统一的数据接口，以统一的数据视图展示数据；

通过统一数据模型SDO和XML Schema处理数据，包括数据的持久化、查找等功能，并屏蔽不同的数据源。

数据服务对象SDO以动态数据API和静态数据API提供数据访问服务，拥有元数据模型和复杂数据对象，可以记录变更历史。

**2．逻辑层**

逻辑层负责具体的业务逻辑实现，按照SOA中的SCA标准，允许多种不同的构件实现方式，如Java构件实现、Spring构件实现、逻辑流构件实 现等。逻辑流构件以图形化的方式构造服务构件，这样编写的代码严格受控，能够提升项目的可管理性和可维护性，并降低开发人员的成本，如图4-7所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222502930.jpg> |
| （点击查看大图）图4-7  逻辑流构件 |

由于构件的实现方式不受限制，可以利用丰富的开源构件和积累的构件库。构件库中存储着大量的展现构件、基础业务构件和运算构件，而且构件库中的构件 经过严格测试、并在大量项目中获得验证。因此，构件实现不需要从零开始，并且屏蔽了技术复杂度，能够迅速搭建高质量、高性能、稳定可靠的基础业务框架。

3．服务层  
服务层以一种中立的、语言无关的方式表达业务逻辑，屏蔽了业务逻辑的不同实现方式，使调用者不需要关心逻辑的具体实现技术和细 节。服务层用来将构件层编写的构件按照统一构件模型装配成服务的方式暴露给其他模块或者其他系统。服务层的服务含义不仅仅是指Web Service，在SOA体系中，服务是构件暴露出来的业务操作，在技术上可以体现为Web Service、Java、EJB、JMS、RMI等多种调用方式。服务与构件的逻辑模型如图4-8所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222611588.jpg> |
| （点击查看大图）图4-8  服务与构件的逻辑模型 |

在服务层，构件定义模型描述服务构件；构件装配模型将小粒度构件装配为组合构件；构件部署模型描述如何部署到运行环境。

使用SOA原则和统一构件模型提供了灵活性，通过基于接口编程和灵活的协议绑定机制，将新的或者遗留代码构建成服务构件，将小粒度构件组装为大粒度 构件，并在应用间共享服务构件以完成资产复用，提高可重用性；统一构件模型也容易获得主流平台厂商和应用厂商以及社区的广泛支持，达到厂商中立性；支持如 Java、C++、Ruby等多种编程语言，达到语言无关性；支持如Web Service、EJB、JMS、RMI等多种调用协议，达到协议无关性。SCA中构件和实现的关系如图4-9所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222652731.jpg> |
| （点击查看大图）图4-9  SCA实现类型 |

**4．流程层**

流程层提供对业务逻辑的编制功能，提供人工活动和自动活动两种活动的处理能力。在业务流程层使用流程对服务进行编制，组合成业务流程，以任务的形式 体现工作活动，并允许配置参与者、规则和过程来进行执行和监控，从而提升生产组织水平和工作效率。卡中心业务应用的流程示例如图4-10所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222745669.jpg> |
| （点击查看大图）图4-10  业务流程示例 |

**5．协同层**

协同层负责流程、服务、应用和人员间的协同，提供对HTTP、RCP、AJAX、Web Service、Java Client等多种客户接入方式的支持，并提供丰富的客户端展现方式，处理单点登录，解决多用户之间的协作问题。通过"协同桌面"提供如电子邮件、即时消 息、手机短信、智能移动终端、在线会议等各种网络交流手段，让员工之间，或员工与系统之间的交流与协同更加快速便捷，如图4-11所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222837234.jpg> |
| （点击查看大图）图4-11  流程、服务、应用和人员间的协同 |

为达到多个要素间的协同，需要使用Mash-up、Dashboard和页面流技术。实现显示与逻辑分离、流程的导航。目前流行的RIA技术包括Ext、Flex、Silverlight、JavaFX以及报表技术等。

建议通过上述5层SOA技术架构提供的能力，在SOA基础设施上使用SOA应用平台构建企业级的SOA应用，如图4-12所示。

|  |
| --- |
| <http://images.51cto.com/files/uploadimg/20091228/222914348.jpg> |
| （点击查看大图）图4-12  使用SOA构建企业级应用 |