



解析 SOA 架构与相关技术

面向服务架构 SOA 与相关技术

面向服务的体系结构（service-oriented architecture, SOA）是一个组件模型，它将应用程序的不同功能单元（称为服务）通过这些服务之间定义良好的接口和契约联系起来。接口是采用中立的方式进行定义的，它应该独立于实现服务的硬件平台、操作系统和编程语言。这使得构建在各种这样的系统中的服务可以以一种统一和通用的方式进行交互。

这种具有中立的接口定义（没有强制绑定到特定的实现上）的特征称为服务之间的松耦合。松耦合系统的好处有两点，一点是它的灵活性，另一点是，当组成整个应用程序的每个服务的内部结构和实现逐渐地发生改变时，它能够继续存在。而另一方面，紧耦合意味着应用程序的不同组件之间的接口与其功能和结构是紧密相连的，因而当需要对部分或整个应用程序进行某种形式的更改时，它们就显得非常脆弱。

SOA 与 BPM

BPM 阵营通常声称，SOA 对于实现 BPM 来说不是必需的。只需部署一个 BPM 套件，就可以更快地实现目标而不会带来多少复杂性。SOA 阵营则注重于如何从一般意义上解决企业 IT 的复杂性。该阵营通常声称 BPM 是 SOA 的一个特性，但是它是 SOA 解决方案的一部分，而不是一个单独的东西。当 SOA 领域的人士谈到 BPM 时，该术语通常与服务编排或流程整合同义，而不强调对业务分析人员友好的建模或人员交互，而后者对 BPM 阵营来说非常重要。

在 SOA 和 BPM 联合发展的浪潮下，我们首先要明确的是，BPM 与 SOA 的本质是截然不同的：SOA 是一种架构方法，BPM 则是一组流程协调管理理念。没有 SOA 之前，BPM 产品已经出现并成功应用。BPM 的主要应用场合有如下几点：1. 业务流程自动化。2.

整合应用系统，实现异构系统之间无缝交流。3.企业流程建模分析。4.监控企业活动，实现企业流程持续改进。

- ❖ **从 BPM 着手实施 SOA**
- ❖ **SOA 和 BPM 汇聚 推动企业并购**
- ❖ **SOA 前程似锦 BPM, BI 和 Web2.0 一个都不能少**

SOA 与 SCA/SDO

SOA 已经成为公认的 IT 基础架构发展的趋势，它为我们描绘了一幅美妙的 IT 系统和业务系统完美结合的图画。然而，即使是在各咨询机构推崇 SOA，各厂商大肆宣传推广 SOA，用户普遍认可 SOA 的今天，SOA 的美好未来依然给人一种不清晰、不踏实的感觉。

我们常常说 SOA 需要解决如何落地的问题。这个难题无法一蹴而就，必须花费很多时间才能逐步进行解决。但在目前，我们已经为 SOA 找到了一个着地的落脚点，这就是 SCA/SDO 规范。

- ❖ **SCA 和 SDO 标准**
- ❖ **服务于 PHP 的 SCA 和 SDO**
- ❖ **SCA/SDO 走向成熟 将正式成为 SOA 标准**

SOA 与 SaaS

随着 SaaS 的愈发火热，加之 SOA 的继续深入，这两种概念开始引出了一些新的混淆，市场上越来越多的人在谈论 SOA 产品在 SaaS 方面的能力。最近的一篇由对象管理组织(OMG) SOA 联盟所完成的，针对首席信息官和首席技术官的调查指出，市场上存在着这样一种期待，那就是 SOA 改变了软件厂商的市场，因此重要的软件可以通过 SaaS 使用 SOA 的方法提供给大家。

- ❖ SOA 与 SaaS 两者将在何处相遇？
- ❖ SOA 和 SaaS 将在 ERP 和 CRM 领域中结合

SOA 与 ESB

不久以前有一些比较聪明的做法，那就是脱离企业服务总线 (ESB) 来配置 SOA。你可以将 ESB 加入到强化现有的一系列已经存在的应用程序中去，从头建立一些服务，然后再将他们串连起来，这样你就完成了 SOA。

事实上，最初的 SOA 活动，就是这么进行的。企业要处理相关的优先数量的服务，配置给他们相关的有限的方法。IT 部门只是进行“SOA 试验”，花一些时间弄明白哪些是需要的而哪些是不需要的。经过一些试验，在级别分割和申请使用上，SOA 就被采纳了。这些很少会被斟酌。

- ❖ 让 ESB 与 SOA 同步
- ❖ 传统 ESB 与 SOA 架构融合
- ❖ 使用 ESB 简化 SOA 复杂性
- ❖ 超越 ESB：下一步 SOA 难题
- ❖ Forrester: 视 ESB 为 SOA 的本质
- ❖ 独家专访：如何看待开源 ESB 和基于 REST 的 SOA？

从 BPM 着手实施 SOA

BPM 和 SOA 正在融合，我们将先采用 BPM 方法，然后再渗透入 SOA 技术……

直到去年七月，半导体测试制造商 FormFactor Inc 对过去的工作方式进行了综合。旧的生产处理系统完成了手工信息输入，同时，将数据拷贝到了 ERP 系统中。

切削技术公司决定放弃原有的手工复式计帐方式，从事面向服务架构项目。在 FormFactor 公司的高级 IT 及商业工艺指导 Nilay Banker 还没有 ROI 成员来为两种系统作 Web 服务综合，但至少这可以让三名员工不用为手工复制数据而烦恼。

他表明，这是高技术制造制造商进入 SOA 化的业务流程管理 (BPM) 的第一步。

Banker 讲述在 FormFactor 的工作进程，FormFactor 是制造和设计用于电子测试的集成电路半导体探针板。Banker 说：“BPM 和 SOA 正在融合，我们将先采用 BPM 方法，然后再渗透入 SOA 技术。”

Banker 表明 BPM 和 SOA 化的某个项目将可以自动化。综合原有的制造处理系统，并将其融入 ERP 系统，可以称之为一个经典的自动化手工处理进程的案例，而其也蕴含了强大的商业生产力。

Banker 在对系统间联系的手工数据检索评论到，“随着生产量和复杂度的增长，那将无法平衡。”

为测试集成电路设计的半导体商提供的探针板的制造是一个很复杂的过程，因为每一个堆探针板都是唯一的对应于不同的制造方法。而且探针板不仅要完完全全的符合客户的说明，还要按期交货跟上生产新集成电路的步伐。

因此检查生产每个探针板的部门，如果发现在需要三个星期来完成的生产步骤中存在问题那时很麻烦的事情。是不是某一部分坏掉了，而所有的一切就要立刻重新开始呢？是不是就要被返工呢？这些问题可以设置或者打破生产进度表。

在众多的 IT 产品中，FormFactor 使用了 Oracle 电子商务组件作为 ERP 系统，其涵盖了生产计划，库存管理，航运和接受。其也包含了物料需求计划(MRP)系统。Oracle 的产品已经具备了 Web 服务的功能，但是在工厂里面的生产系统是没有的。

由在 11 月份被 Applied Materials Inc. 兼并的 Brooks Software 开发的 PROMIS 生产执行系统(MES)最初诞生于 19 世纪 70 年代，那时候 Fortran 问世。所有的生产执行、安排和对工人的培训都有 MES 完成。但是驱动计划进程的 MRP 系统关键数据的需求是很急切的。MES 数据是很耗时的，要三个工人在凌晨三点走有就像 MRP 系统内部进行拷贝，以备其在工作开始时使用。

前些年，点对点的整合两种系统是很困难得。但是去年 FormFactor 的 IT 和商业组织决定试着使用网络服务来解决这个问题，同时也作为 SOA 体系下层结构。

Banker 说：“我们使用面向服务架构体系结构的方法来使其有很好的适应性。”

Banker 的开发小组发现，当有像生产被推迟的事件发生时，MES 系统的 XML 准备数据需要有一组操作数据(ODS)需要被发送。这些信息需要被 MRP 系统提供给经理们关于此类问题的警告来检查生产进程。

Banker 解释说：“当我们建立自己的 Web 服务时，我们为操作数据仓库设置了触发器。在我们开发的 Web 服务中，我们组织和规划了每一个事件，并且为每一个事件设置了触发器，当事件发生时，来产生 ODS。”

在 ODS 中，建立 Web 服务需要的原始数据花了将近两个月，整个项目用了八个月。在去年 7 月投入试运行，这些为车间内的 MES 系统准备好的原始数据向管理员发出了警报，因此决策可以被及时的制定而对探针板的生产的也可以立刻进行。

Banker 表明，这是第一个为 BPM 作基础，由 SOA 驱动的项目。在今后的 12 月里，他还准备使用 SOA 整合 CRM 和 ERP，及生产数据管理和 MES。“我们将真正的使用面向服务架构体系用于商业服务仓库，从分发挥 SOA 的作用。”

(作者: Rich Seeley 来源: TT 中国)

SOA 和 BPM 汇聚 推动企业并购

当面向服务架构和业务流程管理之间的链接形成，供应商开始想法寻找一种工具，此工具用于搭建从高水平的计划、建模到应用开发的桥梁……

提供更全面的工具这一发展趋势，还能从本周 Metastorm 公司收购企业架构(EA)工具供应商——Proforma 公司的案例中窥见一斑，Metastorm 公司的重心向扩展超过它的 BPM 工具的方向转变。同样，IBM 近期的收购动机看来是 Telelogic。

这是大产业趋势的一部分，Bradley F. Shimmin——Current Analysis 有限责任公司的应用架构首席分析师说，正如供应商寻求各种方法销售企业架构领域的产品。

“观察到如 Progress、IBM、Oracle 和其他 SOA 平台供应商从真正的事件驱动观点达到事件最佳处理的目的，这是引人注目的。”他说。“这些供应商以上下文中译解的事件流为基础，寻求使他们的 BPM 产品具有运行时闭环最佳处理能力的方式，主要地用于 SOA/BPM 的复杂事件处理(CEP)。

同时，Shimmin 注意到 BPM 供应商转向了加强他们的商业智能(BI)供应。

“从一个真正的商业智能 (BI)角度来看 BPM 供应商正遇到问题，集中在更好地支持如热门的建模等事件的设计时间方面的阻碍。”Shimmin 说。“我猜想用这种方式 Metastorm 的组合解决方案能够在这个广阔的市场中取得一些进展，但在长远来看，这样做，公司还需要投资于运行时间方面的业务流程分析和优化。”

Neil Ward-Dutton, Macehiter Ward-Dutton 的研发主管，亦已注视这个市场的发展，并指出联合企业建模、事务处理建模和分析、流程执行/监测三者的性能的需要，是

IBM 近期收购 Telelogic 的推动力。他指出 Telelogic 已经从它早期收购 Popkin 软件公司中获得了这一范围内的一些工具。不出意外这将使得 IBM 变成此聚合市场的“纸上”玩家，Ward-Dutton 说。

“[IBM]拥有企业架构模型，一个完整的 BPM 系列，和同样完整的一系列其他模型以及相应的投资管理能力。许多其他 BPM 玩家与 IDS Scheer 公司的合作以求获得一些同样的能力。

Jason Bloomberg, ZapThink 有限责任公司的资深分析师，也观察到 Metastorm 本周的收购转向了增强其企业架构的份量，但是不确信它会在 SOA 领域中怎么操作。

“Metastorm 强烈聚焦 BPM 和 BPA(业务流程自动化)领域，但是它们没有大幅度覆盖 EA 领域和 BPA、EA 的重叠领域，这也是他们收购 Proforma 的原因，” Bloombeg 说。“这一收购将他们从运行时间流程管理和建模公司转变成流程设计和分析公司，并将潜在地增加的能力 例如需要更好的理解业务流程，以及它是如何与混合服务联系的 SOA。然而，两个公司在 SOA 领域的份量都比较轻，因此合并是否有所促进仍需拭目以待。

Metastorm 的首席技术官 Greg Carter 声称他公司扩大的产品供应不适合 SOA 情景。

“以 SOA 的观点，这所做的事情之一是它让我们向那些服务帮你完成的实际的合作、战略目标，进一步勘查了服务技术实现、ESBs、基础架构组件的商业价值。” Carter 说。

他说 Metastorm 工具将提供企业架构，这些架构拥有理解 SOA “变更的影响，无论是协同策略变更还是目标变更，并留心哪些进程和服务受到了变更的影响。”的关键成分之一的能力。工具还帮助架构师和商业计划者追踪服务成本，并将这些成本与总的商业决策和目标联系起来。

“所以我们想建立此更宽阔的情景不仅有助于帮我们的客户更好的理解他们的 IT 投资组合，更有助于理解他们更广泛的流程投资组合、服务投资组合，并以非常完整的方式进行。” Carter 说。

然而，Ward-Dutton 对工具在目前归入企业架构的不同领域中能发挥多少作用表示怀疑。

“这块的工作有两种趋势，”他说。“一种是公司在如企业架构等学科上花费更多的时间和金钱，以争取弄清他们的端到端 IT 前景的意义，以及他们如何能优化为支持事务优先的方法。另一个是在事务处理管理(BPM)领域，公司正在努力开创包含事务用户和软件解决方案设计者之间更积极对话的学科。收购 Proforma 后 Metastorm 看似能经受两种需要的冲击，但事实是他们需要通过不同的方式把他们的能力结合起来，以适应两种不同趋势的需要。能说‘Hey，看看我们背包的尺寸。我们这有你所要的一切。’是不够的。”

(作者: Rich Seeley 来源: TT 中国)

SOA 前程似锦 BPM, BI 和 Web2.0 一个都不能少

根据 Forrester 研究公司的报告，在未来的 5 年时间里，面向服务架构(service-oriented architecture, SOA)将会广泛地被架构和开发人员使用，开发新一代“为人定制，不断适应变化”的应用软件。仅仅依靠 SOA 是不可能单独做到这一切的。

Forrester 研究公司的分析师们为此还设想出了一个全新的类别：“动态业务应用”。这是根据 SOA 的发展原则，并在发展前程中涵盖了一种多个字母组合的服务技术。该研究公司整敦促架构师与开发人员即可开始着手制定未来 5 年内的业务策略和技术实现计划，从整体出发，考虑 SOA 与 B3 技术平台。这里所说的 B3 技术平台即是指 BPM(业务流程管理)、BI(商业智能)和业务规则。

除此之外，Forrester 研究公司所定义的“动态业务应用”将不仅仅在灵活性方面有了突出考虑，同时做到以人为本，它还提倡使用 Ajax 和其他的丰富互联网应用(rich Internet application, RIA)技术将 Web 2.0 式的社会关系网络方案合并其中。

根据 Forrester 长达二十八页，题为“动态业务应用势在必行-IT 行业和业务趋势预览”的报告我们可以看到，这些在 2012 年左右将广泛普及的动态业务应用系统在今天而言虽然还未成型但却是非常有效的应用。该报告的作者，Forrester 分析师 John R. Rymer 和 Connie Moore 对此指出，SOA 的发展前景是显而易见的，但是未来并不能单独依靠 SOA，伴随着 BPM、BI 以及 Web2.0 技术的结合与成熟，下一步的发展必将是动态业务应用系统完美的走上舞台。

报告的作者 Rymer 和 Moore 声称，之所以 SOA 以及其附随的设计方法和相关技术在当前的 IT 行业中没有达到普遍实施动态业务应用的目标，是因为在整个过程当中，他们

的使用往往只是从某一个或者某几个部分出发，单纯的针对某项需求，而不是从一个全局的整体进行考虑并采取应用。

对于目前发展状况，Forrester 研究公司在报告中指出现状，SOA 和 BPM 往往只是作为用于对应用系统集成的改进技术，Ajax 的价值仅仅只在创造性的改进用户界面，而社会关系网络不过是一个很酷的方式衔接起了用户和网站。

“尽管这些看法和带来的改进效果是明显的，且不容小视的，” Rymer 和 Moore 在报告中透露了 Forrester 研究公司的认识，“但是这并没有带来突破性的创造价值。而这些价值才是我们真正应该去挖掘的。SOA，B3 平台 (BPM、BI、业务规则) 以及社会关系计算，只有将这些都整合到一起来才能完全的实现动态业务应用目标。当然，这也是当前业务系统软件转型中的关键所在。”

Rymer 和 Moore 在报告中使用了“转型”一词，实际上只是很简单的描述了这个变化。基于 SOA 的动态业务应用将会完全的改变当前的业务活动方式，就如同 20 世纪 90 年代万维网的出现对整个世界的改变一样。“买卖交易，市场活动，软件开发(通过开放源代码形式)……还有许多 IT 行业内的其他各个方面都会因此而发生改变。而且这变化已经在进行当中。”

“应用程序的设计思路、使用，以及其他我们所能想象和不能想象到的相关技术将会在动态业务应用下创造更多的生产价值，提供更多的生产经验，从而也带来更多的商业价值。这将在自动化决策和业务流程中扮演起关键角色，满足企业各方面需求。这些都会是整个行业突破性的价值创新。” Rymer 和 Moore 还在书中补充道，“我们当前的应用是不可能达到这样的效果的，而我们的企业却又是非常需要这样的应用。”

就目前行业发展看来，有很多的企业其实已经高瞻的看到了这方面的好处，他们也在通过结合 SOA 与 B3 平台 (BPM、BI、业务规则) 的方法走在了通向动态业务应用的道路上，尽管这一路可能会有不少的曲折和坎坷，Forrester 研究公司分析师如是说。同时他还指

出了一个具体的例子，美国 Washington Mutual 公司的在线租赁应用系统就是使用了 BPM 实现了业务规则，逻辑过程的自动化，以软件的形式评估申请人及其所对应的国家制度规定，并通过 BI 系统完成了对客户的动态化处理。

另一个例子则是 Best Buy 公司，作为一个电子产品零售商，它成功地将业务规则与 BPM 结合，从产品需求性和地域性等角度考虑，提供了适当的本地营销计划。

Forrester 研究公司的报告还指出，诸如 IBM，微软和 Adobe 这样的公司已经开始在产品的用户界面上下功夫，这些都是从确定的社会关系网络中获取的经验，Web2.0 的思维形式在这方面有着绝对的指导意义，而对于动态业务应用而言这是最起码的工作。

Forrester 研究公司预测道，在接下来的业务策略中，大多数的组织“将会以信息基础作为业务发展的前提。像 IBM 的 Lotus Notes，IBM 的 WebSphere Portal 以及微软的 Office 套件都会在基础功能，大致框架，服务和流程，事件工具等一些最基本元素上投入大的精力和准备。”在这个基础上，大量的 Ajax 和 Web2.0 技术将会使用于其中，当然也会有包括 wikis，博客和 Mashup 一类技术。

在提供这些应用软件套件的厂商中，最显著的则是微软，甲骨文以及 SAP，他们已经在工作中将 SOA，B3 平台以及更多信息技术转化成为了它们的产品。Rymer 和 Moore 指出。

这个趋势必将成为现实，等待着在 5 年内看到这一切实现并不是当前最好的做法，Forrester 研究公司分析师仍旧在鼓励读者尽快的去开展这项计划，因为他们认为在当前的 IT 行业内遵循这个趋势是没有什么不可能实现的，而尽可能早的开始将会获益匪浅。

“在目前这个阶段而言，尝试去满足动态业务应用的需求远比创造出另一种新的实践方法更加可行，至少我们能够清晰的看见方向。” Rymer 和 Moore 强调道，“SOA 已经

开始在不断的深入发展，伴随着的 BPM，BI，还有 Web2.0 也在崭露出前所未有的美好未来。开始这趟光辉旅程的时间已经来到。”

(作者: Rich Seeley 译者: Shirley 来源: TT 中国)

SCA 和 SDO 标准



David Linthicum 是 ZapThink 执行合伙人。他主要从事 SOA 策略咨询，SOA 项目指导，风险投资顾问服务和 SaaS/Web 2.0 集成咨询。David 已发表八本著作，最新的是《Next Generation Application Integration》（下一代应用集成）。

问：在过去的几个月里，我注意到 IBM (fabric) 和 Tibco (Matrix) 都发布了几个产品，都是从服务角度利用这种“管理式集装箱”的方式。请问你对 SCA 和 SDO 的使用建议有何想法，以及怎么看待它们使 SOA 得以正常运作？

答：服务组件体系结构 (SCA) 描述了一个使用 SOA 的核心概念进行程序及系统构建的模型规范。SCA 支持一种业务应用程序代码组织，该组织是基于执行业务逻辑的组件，通过面向服务的接口发挥功能，并且使用其他组件通过面向服务的接口，即服务引用接口发挥的功能。

在构建 SCA 程序组件时，需要完成以下两个主要步骤。第一，服务组件的实施不仅要提供服务，而且也能使用其他服务。其次，通过服务引用的服务线路要装配成套组件，以构建业务应用程序。

SCA 的目的是，强调从基本的基础结构对服务执行和服务组装的解耦，以及如何访问服务的细节。因此，我们可以把 SCA 的组件看成是进程级执行，而不是侧重于使用许多基本的中间件服务。

SCA 支持多种语言编写服务执行语言，可以用任何一种编程语言，包括 JavaT、PHP、C++、COBOL、XML 为中心的语言如 BPEL 和 XSLT，以及说明性语言，如 SQL 和 XQuery。而且，SCA 也是一种独立的运作方式，可以用异步和/或同步进行编程。

SCA 也提倡使用服务数据对象（SDO），代表商业数据形成参数并返回服务值，对商业数据提供统一访问，以配合 SCA 本身提供的对商业服务的统一访问。SDO 的使用提供了一个数据抽象的基础结构，以最合乎逻辑的途径和方法获取信息。

现在市场上出现的十几个 SOA 标准中，出现了一种服务组件体系结构（SCA）规范。SCA 利用共同的数据抽象为定义共同事务提供了基础，从而促进标准服务的设计与复用。

但是，SCA 本身并不是一种解决问题的方案，选择利用 SCA 的技术也会有所差别。SOA 成功的关键是不仅包括使用 SCA 设计、部署和管理服务的能力，还需要提供一个运行环境，允许那些服务扩展到业务需求的交易量，以及提供企业级的可靠性和耐用性，通过各种消费支持复用。

这项工作的关键，是要了解自己的需求使用合适的案例，以及选择合适的技术。事实上，大部分企业需要的技术，不仅能够支持某个标准，而且该标准需达到业务需要的服务水平。

(作者: David Linthicum 译者: Eric 来源: TT 中国)

服务于 PHP 的 SCA 和 SDO

本文将探究 PHP 等流行脚本语言怎样包含 SCA、SDO 这些相同的 SOA 技术……

由于 OSOA 协作组织的引导，SCA 和 SDO 已经成为下面向服务架构的最新标准。企业在很广泛使用的程序环境中起先构思使用的语言，Java 和 C++，SCA 和 SDO 都已经能为他们提供全统的平台，在本文中我们将探究流行脚本语言怎样包含这些相同的 SOA 技术。

让我们从看看 SDO 能提供给你什么开始。在一个典型的 PHP 应用软件当中，数据将会很大程度上来自于一个相关的数据库，但是假如这些相同的应用软件稍后既需要访问这些资源的信息，又要访问一个平台或者 Web 服务的信息，那将会发生什么？最好的是它将成为一个很长的过程，最坏的它将会是一令人费解的任务，这样的原因是每个数据资源和他本身的一系列适当的古怪行为一起来的。

这样来判断，当处理应用软件数据资源时，SDO 为了 PHP 达到了 SOA。取代起先和每一种数据资源工作，SDO 提供了一统一的风格访问数据实体。完成这个过程的方法是通过 DAS，一间接建造于 SDO 结构的标准。看看以下举例说明 PHP 中 SDO 查询的清单。

Listing 1.1 SDO query in PHP

```
<?php
$providers = $company->shippingByGround;
foreach ($providers as $name => $value) {
echo "$name: $value\n";
}
?>
```

注意为何这最后询问有数据源不可知的特性。那就是说，你不能说说出下面的数据是从哪里获取的，你用 PHP 的语句简单的做一个研究并且复杂细节被处理在 DAS，PHP 当前支持 XML 和关系数据库来源

当 SDO 集中在数据，SCA 的决心是在达到这同样的透明度，但是以更加通用的类或者组成。什么将会成为另外一个典型的 PHP 情景，从任何 PHP 班强迫开发商限制一系列特别的假定设计访问现有的商业逻辑：这个逻辑是在另外一个 PHP 班？或者是它定位于穿过网络？甚至它是不是用 PHP 写的？

当每一个上述情景是可溶解的，每一个要求有不同的编码技术。

用 SCA 的方法，这样的逻辑位于的地点—— 横跨网络或当地—— 应该是需讨论的点，不要提及语言工具——PHP，JAVA 或者是 C++——都将不相关。这最后的陈述可能会让你问“嘿，这个听起来很像非选拔的网络服务，不同在哪里？”好好看看清单 1.2，包括一 PHP SCA 组件。

Listing 1.2 SCA component in PHP,

```
<?php

include "SCA/SCA.php";

/**
 * Calculate a shipment price for a given customer using a specific
provider
 *
 * @service
```

```
*/
class ShipmentQuote {

/**
 * The customer discount fee service to use.
 *
 * @reference
 * @binding.php ../DiscountFeeRate/DiscountFeeRate.php
 */
public $discountFee;

/**
 * The shipping service to use.
 *
 * @reference
 * @binding.wsdl ../Shipper/ShipperQuote.wsdl
 */
public $shipper;

/**
 * Get a quote for a given customer using a specific provider
 *
 * @param string $shipping The shipping company
 * @param string $customer The customer requiring shipment, in order to
obtain discount rate
 * @return float The quote for a given customer using a certain shipping
provider.
 */
```

```
function getQuote($shippingCo, $customer)
{
    $rateShip = $this->shipper->getShippingPrice($shippingCo);
    $rate = $this->discountFee->getDiscountRate($customer);
    return $rate * $rateShip;
}
}
?>
```

上次陈述的最重要的内容是那些 made in @statements, 自从每一个都提供特定的 SCA 行为。东西完成是 @service tag, which is charged with exposing the class in question as a service. 这特定的例子, getQuote 公式是唯一通过服务执行, 有些是通过使用 the @param and @return annotations 获得。

实际执行或采用这个服务代表 PHP SCA runtime -- 将让我们进入 point capable 制作一个 WSDL contract, 和你想获得其它 web 服务一样。除了简单的采用这个服务在 in this manner, 真正好处是让 SCA's model 成为更好证据, 当你测试 getQuote function code 时

陈述 \$this-<shipper->getShippingPrice(\$shippingCo) 和 \$this->discountFee->getDiscountRate(\$customer), 是 2 个基本的 references 出现在早些是候的课. 每个 references 都保留在 underlying services -- 感谢 SCA's @reference and @binding annotations -- 例 \$discountFee 这是 a PHP class and in shipper's case 成为标准的 WSDL backed service.

你也了解到, 大多数的能量在 SCA's programming model 是很简单, 你可以建立 chained calling sequences services 不需要 tainting the actual business logic, 可以简单得结论: 简单移去所有我们只欠描述的 SCA annotations . 若这样做了, 就只

留下 plain business logic 而没其他的依赖, 一个很高利益 characteristic which is highly beneficial to 不仅是为发展 SOA 服务, 同时在 software 方面的利益。

虽然 PHP 简单和用户群体广泛地的特点使得他成为人们喜欢使用的一种建构网络的应用软件, 可是到目前为止, 它的适用范围仅限于一小部分企业, 但根据 PHP 现在所支持的 sca 和 sdo, 以及雷达技术组织试图上显示的技术试图实现 soa 的情况来看, 珩磨你 PHP 的技能可能是明智的选择, 尤其在一些像其他主要语言的东方国家服务领域。

(作者: Daniel Rubio 来源: TT 中国)

SCA/SDO 走向成熟 将正式成为 SOA 标准

本系列的主要内容是展望 2007 年即将出现的面向服务架构的标准情况。本文将首先将介绍 SCA 和 SDO。2007 年 SCA 规范和 SDO 规范逐渐成熟，在新的一年里，SCA 规范和 SDO 规范将正式被归为 SOA 标准。

本系列的主要内容是展望 2007 年即将出现的面向服务架构的标准情况。本文将首先将介绍服务组件架构(Service component architecture, SCA)和服务数据对象(Service Data Object, SDO)。2007 年 SCA 规范和 SDO 规范逐渐成熟，在新的一年里，SCA 规范和 SDO 规范将正式被归为 SOA 标准。

SCA 规范和 SDO 规范将成为专门提供编程模型的标准，开发人员可以在创建 Web 服务时使用 SCA 和 SDO 规范，尽管此时 SCA 规范和 SDO 规范还没有完全成熟，达到标准水平，但是 2007 年他们一定能够成为 SOA 标准。

2006 年 7 月，希望曾到来过，有消息表明 SCA 规范和 SDO 规范有望在去年圣诞节时正式成为 SOA 标准。开放 SOA (OSOA) 组织，一个由多家提供商包括 IBM、BEA Systems Inc、和 Oracle Corp 等公司自发成立的组织，目前正在致力于 SCA 和 SDO 规范晋升成为 SOA 标准的工作。去年夏天 OSOA 组织预测 SCA 和 SDO 规范将于 2006 年底正式成为 SOA 标准。现在看来，这似乎会是 2007 年的某一天。

在回复询问关于 SCA 和 SDO 规范更新情况的电子邮件中，IBM 的 SOA 合作伙伴、项目经理、规范编写人之一 Graham J Barber 这样答道：“我们希望在 2007 年第一季度将 SCA 规范的主要部分作为第一版推出。之后，我们将两个规范与已发布的 V2.1 SDO 规范放到一起，申请成为 SOA 标准。”

不论 SCA 规范是否已经完善或是否能成为 SOA 标准，SCA 都是已被广大提供商应用于产品的 SOA 技术。甲骨文工具与中间件副总裁兼首席架构师 Ted Farrell 说，SCA 规范是一种实用的技术。目前甲骨文的 WebCenter Suite 就使用了 SCA 规范，使开发人员方便开发 SOA 和 Web 2.0 项目，他说。

Rogue Wave Software, Quovadx, Inc. 的一家分公司，本月初宣布开始使用 SDO 规范，并将 SDO 加入产品名称，公司的 SOA 工具套装将被称为 HydraSDO。

甲骨文的 Farrell 说，正式发布 SOA 标准是一件好事，他希望正式标准尽早出台，不过他真正关心的是在工具和应用程序开发中如何有效地使用 SCA/SDO 技术。

“我们对 SCA 何时成为标准十分感兴趣，现在我们叫它 SCA 伪标准，”他说。“我们不希望有所有权问题，因为采用开放结构会有许多好处。”

但是 Farrell 说，最重要的是，一个标准需要适用于应用程序，并有广泛的行业支持。

“SCA 规范更趋于点对点模式，”他说。“IBM 和 BEA 在推广一些标准的时候受到一些挫折，所以他们开始成立开放 SOA 组织。甲骨文以及其他一些提供商加入了该组织。尽管它不是以前的标准组织，但是它在不断发展，不断进步，其实我们最为企业软件提供商真的希望把这些规范变为一种软件标准，希望能够为 SOA 的发展献出一份力量。”

Farrell 说 SCA/SDO 规范似乎在走业务处理执行语言 (BPEL) 的发展道路。业务处理执行语言 (BPEL) 最初由 IBM 和微软共同努力开发出第一个版本，之后 BEA、SAP AG 和 Siebel Systems 先后使用 BPEL，现在甲骨文也开始使用 BPEL。2003 年，这些提供商正式将 BPEL 提交给开放标准组织 OASIS 标准化，2003 年 4 月 6 日，OASIS 组织用 WS-BPEL 的名字吸纳了 BPEL 标准。2003 年 5 月 3 日，SAP/SIEBEL 加入并共同推出 WS-BPEL1.1 版。2003 年 5 月 16 日，WS-BPEL2.0 的草案也在当时被纳入议事日程。

SOA 项目的开发人员和架构师应该从现在开始就接触这些规范，不应该等着它们成熟成为正式标准，Harte-Hanks (HHS) 公司 Aberdeen Group 企业集成副总裁 Peter S. Kastner 说。他认为 SOA 的工作人员都应该熟悉这些规范，并促进它们的发展。

“在未来几年，将迅速推出一批与 SOA 相关的标准，所以对用户来说，最好简要地使用这些标准，从某种意义上说，这是可能的，”他说。“基本上，变化是不可避免的，所以勇于面对变化，适应变化。如果要等到所有标准都完善，之后出台，我像那你会疯掉，这大概需要十年的时间。”

(作者: Rich Seeley 来源: TT 中国)

SOA 与 SaaS 两者将在何处相遇？

随着 SaaS 的愈发火热，加之 SOA 的继续深入，这两种概念开始引出了一些新的混淆，市场上越来越多的人在谈论 SOA 产品在 SaaS 方面的能力。最近的一篇由对象管理组织 (OMG) SOA 联盟所完成的，针对首席信息官和首席技术官的调查指出，市场上存在着这样一种期待，那就是 SOA 改变了软件厂商的市场，因此重要的软件可以通过 SaaS 使用 SOA 的方法提供给大家。

但是，拥有清晰的定义是十分重要的事情，Current Analysis 有限公司应用软件程序基础设施首席分析师 Bradley F. Shimmin 这样说道。

“我确实是把 SaaS 看作一个传递机制，这个传递机制指出单个实例/多个承租的应用软件程序，”他说。“而且我将 SOA 视为开发松散的耦合的软件的哲学框架。因此，SOA 包括了一切关于软件是如何被架构起来的東西，而 SaaS 是一切关于软件是如何被应用的。”

Shimmin 观察了有关 SOA 和 SaaS 之间的混淆的一部分，这些混淆是源于当我们谈及服务的时候我们没有清晰的指明我们的意思造成的。

“也许这个问题滋生于服务这个词语，”他解释说。“在 SaaS 当中，他表示应用程序可以像任何服务一样被传递，就像你家中电话的语音一样，看起来似乎就是为你的需求量体裁衣得到的，也是你可以一定层度上客户化的东西。而 SOA 的定义和这个无丝毫的联系。SOA 支持的服务，都是些离散的可以再使用的事务处理，这些事务处理合起来就组成了一个业务流程，是从基本的系统中提取出来的抽象代码。”

ZapThink 有限公司高级分析师 Jason Bloomberg，赞同有关 SOA 和 SaaS 的混淆是针对于两者的不同点没有清晰的定义以及在结合使用时就出现了的问题。

“在 SOA 和 SaaS 的关系之中有大量混淆的地方。” Bloomberg 说道。“SOA 是一个框架的方法，而 SaaS 是一种传递模型。服务通过 SaaS 传递模型传递也许可能也许不可能达到松散的耦合以及像我们在谈及 SOA 的时候谈到的类似于 Web 服务的订立了标准的服务。大体上来说，这些服务的种类是不同的，但是我们在市场上正在寻找能够通过 SaaS 的方法进行传递的合约化服务的汇合点。”

Bloomberg 还叙述说传统的用 SaaS 传递应用程序的方法的功能已经通过网络接口实现了。最近 SaaS 开始试图融合 Web 服务，使得它可以在没有用户接口的情况下通过网络服务进行传递，但是，这种传递的方法还不是 SOA。

“通过 SaaS 传递 Web 服务并不需要 SOA。” Bloomberg 是这样说的。

不过，Bloomberg 以及其他的本文采访过的分析人士都认为 SOA 的方法是对 SaaS 有好处的。

“SOA 带给 SaaS 的既有松散的耦合，也有约定化的、能够治理的服务。” Bloomberg 解释说。“这些服务都经过约定，而且都于政策的元数据相关联，这样可以对服务提供者和服务使用者之间的关系进行约束。举例来说，这些政策也许指明的是服务需求的质量，再使用的指南或者是版本政策。”

对于版本政策的需要更加凸显了 SOA 能够为 SaaS 软件厂商提供价值的重要。他说。

“我们假设，你通过 SaaS 提供一个 Web 服务，而且你有很多顾客在使用这项服务。现在，到了该将这项服务升级的时间了。对于所有的客户而言这会发生什么呢？让他们将

所有的工作停下来?他们需要手动升级他们的软件吗?每一个选项都代表着服务使用者和服务提供商之间的紧密联系——在这种情况下 SOA 能够解决的问题。”

提供给 SaaS 的 SOA 方法是能够解决这个问题的。Bloomberg 说道，因为 SOA 可以提供“一个适当的事先定义好的版本政策，这样会规定用户必须每个月都要用一些规定的步骤去保证他们都在使用最新的软件。例如，通过自动的下载一个升级，在用户下载的第二天，服务将自动更新版本。现在，对于客户而言，自动化保持一个或者所有的版本和服务服务器同步更新是可以实现的。这就实现了松散的耦合的运作，以及一个 SOA 有力运转的证明。”

SOA 对那些 SaaS 的软件厂商而言也是相当重要的。原因在于它能帮助他们更有效的在于它能帮助他们更有效的进行应用程序软件的传递，而且，和传统的打包的应用软件厂商相比，他们又从价格方面获得了竞争优势。Interarbor Solutions 有限公司首席分析师 Dana Gardner 说道。

“SOA 对软件而言是一个非常重要的方法，原因在于软件提供厂商是如何构造他们的结构以及如何使得他们的传递应用软件程序更有效率等方面。” Gardner 如是说。“因此，在某种意义上，他们提供了一个 SOA 的试验场。因为他们的业务常常是基于预订的而且他们和那些提前的软件许可的软件公司厂商进行竞争，他们想要降低自己的价格。因此他们需要不断的解决成本问题，使用再使用的方法以及在中有效的、灵活的运用他们运作节俭的特点。”

公司现在已经开始建立一个基于 SOA 的 SaaS 系统。Gardner 说道。包括 Google 公司和 Salesforce.com 等公司在内的大公司以及刚刚开始提供 SaaS 商用应用软件程序的 Workday 公司等都在做这件事。他还指出，微软正在开始从他的打包的软件工具像 SaaS 的方向进行转变。例如提供 Microsoft Office Live，这个服务是微软补充其桌面应用软件程序的一个范例，被他们称为“软件加服务。”

Gardner 说，在等式的企业 IT 这一侧，业务开始期待 SaaS 能够为他们提供一些 SOA 的执行。

“随着企业开始越来越多的使用 SOA 的方法，他们在业务运作和服务使用上应用的范围越来越广泛。”他解释说。“他们将拥有习惯于处理丰富互联网应用程序的，习惯进行 mashup 的以及习惯用组合的或集成的应用程序的方法使用服务的软件开发人员和商业分析人士。他们将走向可以使用 API 并可以从开放的市场上获得的 Web 服务，也可以内部使用或者跨供应链使用或者使用其他的延伸企业业务系统的方法这条路。我们有很多理由期待这一点。”

就像 Shimmin 的看法一样，让 SOA 和 SaaS 一起工作对于两者而言都是最好的结果。

“两种技术都是共生的，但是两者可以通过两种不同的方法一起工作。”Shimmin 说道。“首先，你的 SaaS 应用程序需要通过 SOA 的标准和 SOA 的观点——很有可能是个好的观点建立起来，其次，你的 SOA 基础设施将作为 SaaS 应用程序的一个集成点被使用。例如，一些公司正在扩展的 SaaS 应用程序和内部业务线应用程序之间雇用 SOA 企业服务总线，这样做可以提供需要的转变，并可以进行安排数据治理。”

Gardner 认为这将是 SOA 和 SaaS 的未来一个展望。

“因此你可以在不久的将来，在两者的交集处看见软件厂商不断增长的提供这样的产品：它能够像软件即服务的产品一样，也许不是整个的打包的应用程序，但是一定是组件和服务混合在一起的，和 SOA 和软件开发人员能够使用的内部业务服务组合相匹配的产品。”

(作者: Rich Seeley 来源: TT 中国)

SOA 和 SaaS 将在 ERP 和 CRM 领域中结合

如果面向服务架构(service-oriented architecture, SOA)和软件即服务(Software as a Service, SaaS)的结合是应用交付和集成的未来,那么 Workday 公司就是模范标兵

Cape Clear 软件公司的 CEO 和创始人, Annrai O' Toole 说, 当他看到 Workday 对他的公司的企业服务总线(enterprise service bus, 缩写为 ESB)进行了处理之后, 扩展性就从他眼前消失了, 并且他看到了主办的(hosted)、按需应变的 SOA 的未来。Workday 的 CTO, Stan Swete 表示从相反的方向, O' Toole 为他的 ESB 客户演示了 SOA 和 SaaS 如何改变了 ERP 被交付、集成和改变的方式。

无论是否有互利性, Swete 说, “一个新的商业模式已经出现了, 这就是所需应变的应用的交付即服务(on-demand delivery for applications as services)。

O' Toole 说如果你想要看到新的商业模型涌现, 看 Workday 就行了。Swete 警告所有这些都还是一个为期两年的启动项目的早期阶段, 该项目受到 PeopleSoft 的创始人及前 CEO, Dave Duffield, 他现在是 Workday 的 CEO, 和该创始人, PeopleSoft 公司另外一个高层 Aneel Bhusri, 他现在是 Workday 的总裁的资助。该公司发起了第一个按需应变的基于 Web service 的人力资源产品, 目前该产品已经有了 12 个客户。

唯恐所有人都陷入 O' Toole 的将 Workday 看成是潜在的 SOA 公司的观点, Swete 警告主办 SOA 并不是 Workday 的 SaaS 产品的最初的计划, 但是的确包含。

“Annrai 一直都在强调保持 SOA 简单化的需求, ” Swete 说。“我认为正是这种简单化的驱动让他看到了主办 SOA 的机会。它就是我们正在思考的东西。

但是绝对不是行动。Swete 指出 Workday 的很多潜在客户还没有使用 Web service。他们还在运行了基于遗留硬件和软件的人力资源管理系统，它们可以追溯到客户端/服务器时期，甚至是主机和文件数据库还是最新技术的时期。SOA 仍然是很神圣的东西。

“在我们和客户进行的交流中，我认为每个人都看到了 SOA 的力量和潜力，” Swete 说。“但是我认为每个人都承认它是一个新技术并且它不是他们掌握的技能。他们很高兴实验一些技术，但是他们并不打算真正的使用它，虽然它擅长构建这种集成。”

最先尝试者(early adopters)往往都站在相同的高度，比如 RightNow Technologies 公司，该公司是按需应变的 CRM 产品的供应商，在全球拥有 1800 名客户。“我们在和我们商业应用的 SaaS 厂商的合作中也非常的投入，” RightNow 的人力资源主管，Kevin Boylan 说。“在和我们自己的客户的合作中我们也看到了按需应变解决方案的好处，所以对我们来说购买一个满足我们需求的按需应变的供应商是容易的决定。Workday 就非常适合。”

那些不是很前沿的公司则采取一种比较谨慎的态度来对待 Workday 提供的产品，Swete 承认。有一些客户需要理解他们从 Workday 获得的基于 SaaS 人力资源应用中嵌入式 ESB 的价值。

“我们和客户进行交流有两种截然不同的方法，”他说。“我们必须解释通过嵌入 ESB 我们要做什么。它提供了什么好处。它带来了什么挑战。典型的交流是首先解释对我们产品的集成将通过 Web service 进行，而这会被很好的接受，因为即使是内部没有实现这种 ESB 类型的客户也会将其视为进行他们自己集成的一种方式。”

对于那些对主办 SOA 感兴趣、但是他们已经有了 ESB 的客户，交流将会转向互操作和 Web service 标准，Swete 说。

“有很多连接的方式。他们想要确保你有一个开放的方式连接到他们企业中所有的系统，”他解释。“对标准的支持，尤其是我们对 Web service 的支持，加上他们对 ESB 的了解确保了我們不会让他们失去任何机会。”

Swete 听到潜在的客户这么说：“你使用 Cape Clear，而我使用 IBM WebSphere。这是否会阻止我连接到你？”

他是这么解释的：因为 Cape Clear 和 IBM 都遵守了基本的 Web Service 标准，集成不是问题。Workday 可能不会开始通过 SaaS 的主办 SOA 计划，但是它的确承诺对于 Web service 标准的遵守，关于这一点 Swete 从 SOA 和 SaaS 的未来的基础来争论。

“我们的应用和外部世界的交流的唯一方法是通过基于 XML 的方法，”他说。“无论你和我们 UI 进行交流或者我们的产品和第三方的应用进行交流，都是这样的。我们将所有的东西都基于 XML，并且在第三方产品的情况中我们将所有的东西都基于 Web service。我们将让我们应用的所有数据进出和任何接触我们应用逻辑的操作都是基于 Web Service，而不是，比如直接调用底层的数据库。我们已经封装了我们的商业逻辑并且唯一将其暴露给外部世界的方法是通过 Web service。”

SaaS 和 SOA 的结合的希望是能够提供互操作和集成的能力，Swete 说。在需要和工资系统、福利应用和公司的大的帐务软件进行交互的人力资源应用的例子中，客户现在才刚开始认识到 SaaS 和 SOA 的结合所带的好处，他说。

“人们将发现 SaaS 结合 SOA 的力量，”他说。“作为 SaaS 厂商，开始的时候我们还没有获得 Web service 合适的粒度，我们开始了构建可互操作的应用的漫长的路程，但是当我们开始调用其他厂商和客户的更多的解决方案的时候，我们发现 Web service 的合适级别是和这些应用进行互操作。所以我认为 SaaS 和 SOA 的结合是通用互操作的重要概念，并且我认为你会看到它在业内将得到发展。”

(作者: Rich Seeley 来源: TT 中国)

让 ESB 与 SOA 同步

更多的 ESB 经销商仍在为找到多于一种的实现中间件连通的方法而匆匆地重新构造他们的产品。Fiorano Software Inc 已发出最新公告，他们将所有的产品标记为面向对象的架构 (SOA) 平台，而不是 ESB。

今年，当主要的经销商如 IBM 和 BEA Systems Inc 冲入企业服务总线这个市场时，更多的 ESB 经销商仍在为找到多于一种的实现中间件连通的方法而匆匆地重新构造他们的产品。实际上，Fiorano Software Inc 已发出最新公告，他们将所有的产品标记为面向对象的架构 (SOA) 平台，而不是 ESB。

根据 Fiorano 公司的首席执行官 Atul Saini 所说，它们之间的区别在于业务流程协作和重构 Java 连接器架构 (JCA) 构件。Fiorano 公司的 SOA 平台 2006 版看起来在连通性方面甚至在处理一些基础的商业构件方面提供了一些智能机制，这些机制可以让用户在创建 Web 服务的时候进行重用，以替代单一的连通性。

“ESB 本身的价值是有局限性的” Saini 说。“它增加了智能方向，这是我们从中间件的早期版本所看到的，而且这就是它出色的一面。”

但是 Saini 并不是把 ESB 投放在总线的使用中。Fiorano 公司的 ESB 处于 SOA 平台的核心。

“如果你使用不同的机器，那么通信已经完成了。”他说。“一旦你开始使用多终端，那么在这个过程中某些关键步骤已经完成了。”

ESB 为可视测图工具、Java 连接器架构(JCA)的构件以及构成 Fiorano 公司的 SOA 平台的业务流程执行语言(BPEL)服务模型提供神经系统。Fiorano 公司的竞争对手 Cape Clear Software Inc. 公司这个月将发出公告来推出他们的最新更新版本，同样，Fiorano 公司也打算通过发布这样的信息冲击市场，这信息表明了所有建立在 ESB 基础之上的联合功能将使 SOA 采用一种更加简单的，并且是低费用的建议。

ZapThink LLC 公司的分析师 Ron Schmelzer 说，ESB 市场并不能在第一时间吸引大量的资金，而且现在 IBM、BEA 和开放源码的 ESB 现在都正在进入这一领域，它为那些建立类别的公司甚至只留下了相当少的市场份额。他补充说，当 IT 公司只顾着紧紧抓住大量的关于建立一个面向服务的设计的争议时，ESB 看起来已经丢失了某些商机。

“ESB 其实是个真正的沼泽，” Schmelzer 说。“它有一点但不是完全地与 SOA 有关。实现 SOA 面临的真正挑战不是你需要更多的信息。顾客真正需要帮助的地方是处理他们创建服务时的所有元数据。”

当然，这也就是 Fiorano 公司使用 BPEL 服务器所获得的，但是 The 451 Group 的分析师 Dennis Callaghan 争辩道，它是没有资格作为一个完全的 SOA 平台的。

“Fiorano 公司已经为他们的企业总线增加了一个更好的 BPEL 引擎，并且现在称其为 SOA 平台，”他说。“那么这是否意味着他们已经拥有了一个完整的平台，来实现企业的面向服务架构了么？实际上并不是的。我想说的是，你仍然需要 Web 服务管理，一个使其生效的注册机制和安全机制。”

Callaghan 补充说，目前没有一个经销商达到这样的程度。他相信，除业务流程管理之外，ESB 将作为基础类别在业务活动监控、Web 服务设计以及 SOA 管理等这些领域中得到批量化传播。

同时，Schmelzer 争辩道，改革需要尽快地进行。

“经销商需要得到这样的证明:他们与面向服务的架构是息息相关的。”他说。“它不能仅仅只是新瓶装旧酒的产品。对于 Web 层而言,与最新的六种不同于以往的解决方案相比较,你的产品中有哪些不同?”

对 ESB 而言,围绕着如何更快地与新服务“整合在一起”,看起来将要实现与其相关的案例,正如 Saini 使 ESB 并入 SOA。这样看来,ESB 的未来就是让用户生活更方便的包中的一部分。

“顾客既想建立面向服务的架构,又想每一个步骤都不付费用给咨询公司,” Saini 说。“只有当你减纷繁的复杂性的时候,才有可能减少对咨询的依赖性。除此之外,你是不可能省时或者省钱的。”

(作者: Michael 来源: TT 中国)

传统 ESB 与 SOA 架构融合

在科技日新月异的大潮下，“传统”的企业服务总线(ESB)正经历着保持与面向服务架构需求同步的改革，从原来的综合/信息工具转化成为具备商业流程管理(BPM)工具，管理和分析功能的基础架构。

在近期 webMethods Fabric 7.0 介绍展现了 ESB 在其原有的综合支柱角色基础上成长：

AMR Research 副主席 Bill Swanton 先生说，我们看到 webMethods Fabric 7.0 是以 SOA 方式建立综合应用软件的一个很直接的发展环境。新的产品将所有历史上与集成有关联的工具加以整合，其中包括原有企业应用系统集成 EAI 技术和 ESB，将 BPM 和商业活动监控(BAM)囊括进来，并增加了注册，储存和管理的性能。

阿伯丁集团企业整合副主席 Peter S. Kastner 指出：像 EAI 一样过时的软件也可能出现在 SOA 世界中，对于支持遗留技术它还是很重要的，因为这些遗留技术还存在于许多企业当中。

他还说：“我们发现几乎没有人愿意舍弃他们在 EAI 上的投资只为了购买 ESB 然后宣称他们是唯一一个使用 ESB 的人。”他引用了 webMethods 为例，webMethods 是原 EAI 公司，现在已经成功转型进入 SOA 行业，同时提供与旧的整合技术相连的桥梁。今年他对于 IT 部门的研究推翻了他自己的观点，旧整合技术供应商并非注定失败。

Kastner 还说“今年我的假设是：EAI 公司将会受到重创。但事实上，他们绝大多数的客户很容易的将自己的 EAI 架构通过适配器与 SOA 连接。”

Kastner 和 Swanton 都赞同这一点：对于 webMethods 正在包装的旗下品牌 ESB 技术发展来说，引入更新的技术，特别是 BPM 是很重要的。

Swanton 说：许多 ESB 供应商都还在尝试仍在整理支持商业分析和软件开发者的综合工具包。整个 ESB 技术会将 BPM 纳入其中以便使软件开发者能与商业分析师在商业程序软件上进行合作。

Kastner 说， BPM 之所以重要， 是因为在其研究的基础上， 大综商业交易在很大程度上包含 BPM. 我们看到大概有 50%全球 5000 强积极投入 BPM 发展上。他还指出 webMethods 并不是唯一一个将 BPM 融入 ESB 技术的厂商。

他说：我们应该注意到 Tibco 在过去的几年中已经对 BPM 进行了大量投资， Fiorano 也在近半年中对其 ESB 产品在 BPM 领域进行了加强。

webMethods 技术总监 Marc Breissinger 说， ESB 的发展需要一个新的定义， 甚至是对它进行重新命名。他还指出这个产业已经引入了“Fabric”这个术语， 这在他的公司和其他公司的产品品牌上可以看到。

技术总监从 webMethod 角度来讲， ESB 的定义从一组直接对特征和功能的定义扩展到了一个大的解决商业问题的技术。

Breissinger 说：这周上市的 webMethods Fabric 7.0 重点在于 BPM 系统在 SOA 和经典整体环境下的发展。我们所看到的是一个直接自动的处理 BPM 类型与人为控制更多的 BPM(工序)类型的结合。他们与商业活动监控相连贯， 这也需要 BPM 系统和 UI 来实现。

“Fabric”这个术语已经成为了新增工具和技术的保护伞。Burton 集团在讨论使用 SOA 实现软件整合时也用到像中间件“Fabric”和 Web 服务“Fabric”这样的术语。

Breissinger 指出他并非对 Fabric 这个术语有过多偏爱，但它是描述 ESB 和 BPM 和其他技术的整体方案来实现测量、建模和实施 SOA 的最佳用词。

WebMethods 技术总监在简单介绍术语背景时说：“过去，我们还没有接触 BPM 领域的时候，我们说整合‘支柱’，之后我们又称服务母线。在一定程度上来说，Fabric 是更深层次的，囊括 BPM、体系、复合程序开发能力的概念，这样你可以有一个建立系统基础架构的整体概念，而不单单是一条线或是支柱。”

(作者: Rich Seeley 来源: TT 中国)

使用 ESB 简化 SOA 复杂性

为什么在面向服务的景观图中又加入了一个移动的部分?难道对面向服务的应用的管理还不够复杂?引入企业服务总线 (ESB) 的原因和许多年前选择企业应用集成策略的原因是一样的。

在 SOA 实现的早期阶段,当目录仅仅由一个或者两个基于项目的服务组成的时候,ESB 看起来是英雄无用武之地。但是幸运的是,如果在企业中采用 ESB,那么服务的部署会被加速。任何一项策略都被要求能够提供按需应变的扩展性,可靠性和足够的性能等特点。从构架的角度,使用一个合理的原则避免服务混乱不失为一个好的想法。

随着企业 SOA 的逐步成熟,业务功能会从各种源头被挖掘和发现出来。这些服务的提供者可能是遗留的应用,第三方软件包或者主要解决方案提供的功能。虽然理想状态是所有这些服务都使用相同的技术,但是现实情况证明这是不可能的。很有可能 Web Service 标准仅仅是使用的技术中的一个而已。

高级的 SOA 功能,比如服务编排 (orchestration),自动业务流程,事件驱动架构和复杂事件处理,都依赖于健壮的企业应用集成架构。

将这些注意事项牢记在心,判定带有 ESB 的完整功能的 integration broker 的标准就很清楚了。

可靠性和可用性

由于多层应用的出现,对分布式软件中所有移动部分的可见性是一项重大的挑战。由于消费者完全从应用面向对象开发中分开,服务的消费者必须能够发现服务并且预计服务的可用性、质量和性能特征。ESB 能够处理服务注册,同时有助于兼容 SLA (服务等级协

议, service level agreement)。为了监控服务的健康状态,就必须清楚该服务对其他服务的依赖,以及运行平台的状态。当问题被检测出来,就会发出警报和通告。许多 integration broker 的产品或者提供内建的监控功能,或者,更常见的情况,提供钩子 (hook)给监控工具以便获得更全面的可见性。这些工具可以提供审计,日志和异常处理的通用服务。

扩展性

基于总线的架构提供了对服务的真实的部署拓扑结构的一种抽象。增加 Web 服务器、应用服务器、第三方软件、遗留应用和数据库实例——服务描述中功能的真正执行者——的处理能力,将不会影响到服务的消费者。实际上,工作可以进一步委托给联合的总结,这样 ESB 本身可以按照最有效的利用硬件和网络资源的方式来部署。

安全

已经实现的服务不应该承担管理什么用户在什么情况下能够访问服务的重担。用户的角色可能随着时间而变化。这种横切 (cross-cutting) 的功能应该处于 ESB 中的各种服务之外。

延伸性和敏捷性

随着业务适应于变化的时代和新的机会,业务流程同样会变化。对于一个成功的 SOA 实现来说,对灵活性的规划非常关键。转换应当在尽可能的靠近往来于商业信息模型的集成端点处完成。这种方法可以便于流程的灵活自动化,组合服务的编排 (orchestration) 和将流程相关的商业规则从服务的商业功能中抽取出来。

总之,功能完善的 integration broker 会提供核心的 ESB 功能,以及其他的一些异构的服务拓扑中必须的功能。这类技术能够为复杂的结构提供一个“简化的”外壳,否则这种简化需要在每个应用或者服务里面取实现。

(作者: Maja Tibbling 来源: TT 中国)

超越 ESB：下一步 SOA 难题

不久以前有一些比较聪明的做法，那就是脱离企业服务总线 (ESB) 来配置 SOA。你可以将 ESB 加入到强化现有的一系列已经存在的应用程序中去，从头建立一些服务，然后再将他们串连起来，这样你就完成了 SOA。

事实上，最初的 SOA 活动，就是这么进行的。企业要处理相关的优先数量的服务，配置给他们相关的有限的方法。IT 部门只是进行“SOA 试验”，花一些时间弄明白哪些是需要的而哪些是不需要的。经过一些试验，在级别分割和申请使用上，SOA 就被采纳了。这些很少会被斟酌。

但是，SOA 也在不断的成熟和发展。企业从 SOA 中得出结论，认为 SOA 可以帮助企业带来新的动力和在现有的系统上创造新的价值。SOA 可以使服务的申请和复用变得简单，从而促进生产的发展，同时降低了成本。

但是，采用 SOA 虽然可以带来利润，但是企业需要摒弃以前的一些观点和看法，重新定义大多数早前的 SOA 配置，来适应在意识形态上的变化，而这种意识会影响企业和其流程。

如果缺乏适应 SOA 配置结构化的巨大转变的心理准备和认识，组织往往会陷入“SOA 恐慌”的危机之中。不幸的是，这样的认识往往可望而不可及。早期的 SOA 试验在各操作领域和商业单位中建立了“独立的 SOA”系统的环境。

随着 SOA 深入企业内部，将面对由被复制在不同分区中的服务对应得唯一的可能性，这是展示 SOA 主要益处的一个方面，即服务的重复利用。其他的方面，用离散的方法来对

SOA 进行配置，需建立针对处理现存的服务和所支持的 IT 资源的无效管理的成熟环境，即处理那些在 IT 部门内不需要存储的服务。

帮助引导 SOA 进入下一个阶段，使之不仅成熟，而且有效，需要以下方式：以企业方面定义的调配内涵来管制你的 SOA 发展，强调应当强调的流程和任务，以及处理新的 SOA 动力学。

管制看起来很好，但是当你静思该如何做的时候，就会发现那真的很迷茫。一些人会说很简单，“将这些工作注册，然后就行了。”可是，“简单”这个词好像是相反的意思，每一个从事过 SOA 开发工作的人都会这么说，告诉你那个东西很简单，但是这个词很少用于描述一个流程。

这并不是说注册工作不重要。明白你现在在此企业中所处理的服务和怎样完成这些服务是第一步工作，可以用来避免“SOA 恐慌”。但是那里有很多工作要去做，以明确你的管制工作。存储功能可以作为一种好的 SOA 管制来考虑。

然而，你的注册功能就像目录或电话簿一样，可以在其中找到服务的位置，而你的 SOA 存储功能是一个针对全企业的完全有效服务目录，那么代替对服务种类和服务位置的辨识，存储功能可以使企业对现存的有效服务进行适当的使用。筛选和存储对所有相关的服务的信息的存储，不仅仅是契约，还包括政策，实现的产物和依赖分析等。通过由组织内部定义的角色和职责对这些服务提供不同的显示，那么有效的存储就可以加大服务的重复使用率，而更好的控制调配和 SOA 策略中的各个不同部分。

什么可以使一个 SOA 存储唯一呢？即 SOA 存储包括 SOA 生命周期的所有方面，从设计和发展到发展和管制。在一个逐渐成熟的 SOA 中，这可以有效的为你提供所有需要的 Web 服务。

例如，某个存储应该保存服务的概念，基于应用的细节，包含服务使用的规则和更加详细的信息，如针对指定用户的每一个服务。简短的说，存储应该包含为整个 Web 服务所准备和重新调配的必要信息。通过手工获取这些功能，明确它们的作用，更简单的分析在 SOA 环境中的服务，检查出存储中与实际流程相悖的地方。

SOA 领域在成熟和变更，如果 SOA 成功地话，企业也要变更。ESB 足以运行的地方，SOA 就要做的更多。公司需要思考他们的 SOA 调配策略和帮助他们实现目标的技术。对于整个企业，正确的存储选择可以给客户带来加强已有的服务，调配和管理周期的方法。

手工完成这些功能，因审视被可以带来利润的 SOA 调配支持的商业灵活性和适应性使 SOA 变更时，企业最好可以重复利用已有的服务，理解现有的多种服务的关系，重新调配和准备服务。

(作者: Steph Bacon 来源: TT 中国)

Forrester:视 ESB 为 SOA 的本质

尽管关于如何把必要的企业服务总线转化为面向服务的架构存在激烈的争论，但是在最近 Forrester Research 公司在报告中说他们认为持续采用 SOA 能很好的体现 ESB 的思想，他们把它称为“SOA 的主要切入点”。

该报告的细节分析了两种独立 ESB 市场、处于领导地位的公司以及使 ESB 对有集成问题的公司价格上的吸引力。Forrester 公司总共调查了 116 家公司，报告显示 77% 的大企业、51% 的中型企业和 46% 的小企业将主动在今年年底前实现 SOA。评估认为三分之一的“基础设施决策者”在未来 12 个月中会增加他们的 ESB 部署。

Forrester 公司分析师 Mike Gilpin 说：“尽管人们还不十分确定如何构建出一个完整的 SOA，但他们已经知道要解决集成问题，而 ESB 正好能帮助他们解决该问题。”

报告把客户市场分成两个独立的群体。一个被称为“保持简单”群体，另一个被成为“即刻整体部署”群体。

“保持简单”群体想要一种低廉的、基于标准的 Web 服务编排工具并在此之上构建健壮的 SOA。Gilpin 认为价格正在成为 ESB 的一大卖点，它使得公司用相对少的投资就能启动 Web 服务，于是他们就可依靠必要的技术去构建完整的 SOA。

Cape Clear Software、Sonic Software 以及 Fiorano Software 等公司就是采用了以上这种做法。

他说：“实际上，ESB 的成功为导致集成软件方面费用的降低。因此尽管公司迁移了更多的单元，但费用却降低了。”

而“即刻整体部署”群体则希望能一次性解决集成方面的所有问题，希望得到在服务证明周期中负责细粒度控制和服务过程管理的工具。

Gilpin 相信这两种不同的市场会导致对两种不同类型 ESB 产品的需求，即一种轻量级的产品以及一种全面的 SOA 集成与控制产品。报告显示，所有 ESB 倡导者以及企业应用集成的推崇者都有目光投向了全面产品市场，于是把轻量级产品的市场留给了 IBM 不久将要发布的 WebSphere ESB。

Forrester 公司高兴的发现经常被分析师团体称为遗留中间件的 EAI 厂商正在发生转变。他们在 Web 服务基础设施的服务仲裁以及控制/变更方面变得很有竞争力。根据报告显示，EAI 的推崇者对扩展市场贡献巨大，但只有 Oracle Corp 和 webMethods 公司有能够在价格上取得有竞争力的地位。

处于领导地位的公司包括 Cape Clear、Sonic、Fiorano、Iona Technologies、Oracle、webMethods、Tibco Software 以及收购了 SeeBeyond 公司的 Sun Microsystems。今年夏天，BEA Systems 公司作出了两次领导者的举动，一次是发布 AquaLogic ESB，另一次则是推出 WebLogic Integration suite。IBM 由于最近才刚推出 ESB 产品所以没有参加调研。

Gilpin 并不认为像 BEA、Oracle 以及 IBM 这样的平台提供商能够引起市场两极分化，因为集成问题不应该与应用程序运行在 SOA 内部哪个地方紧密耦合在一起。

他确信与产品缺少结合会对市场造成挑战。

他说：“人们都认为 ESB 并没有被标准良好的定义。它还处于一种类似于 J2EE 之前的应用服务器市场的状态。”

他看好像 Apache Synapse 这样的项目，该项目被认为会为 Web 服务仲裁定义标准的方法，而那会是解决问题的潜在手段。

Gilpin 认为控制已经是 ESB 市场没有解决的最大问题。SOA 注册提供商早已注视着该领域，而他希望这两个领域能够通过融合或协作的方式走到一起，也可能是通过 XML 网络的方式。

他说：“SOA 太大，以致于已经不可能靠一个提供商来实现全部的内容，即使是 IBM。今后很可能会靠很多提供商的团队工作才能形成一种生态系统。”

(作者: Michael Meehan 来源: TT 中国)

独家专访：如何看待开源 ESB 和基于 REST 的 SOA？

开源 ESB（企业服务总线）的前景如何？是否有闭源 ESB 的应用空间或者能否采取某种开闭源混合的方式？在这次采访 Paul Fremantle 的第一部分中，这位 WS02 公司的共同创始人兼副总裁论述了这些问题，并介绍了 WS02 公司 ESB 产品的核心——Apache Synapse 开源 ESB。在合作建立 WS02 公司、开发基于 Web 服务标准的开源产品之前，Fremantle 曾担任 IBM 的高级技术人员，创建了 Web Services Gateway，而且带领团队开发并组装成 WebSphere 应用服务器的一部分。他作为团队的成员，也为 WebSphere Application Server 6. 开发了综合服务总线技术。目前，他是 OASIS Web 服务可靠交换技术委员会（WS-RX）的主席之一，该委员会研究使用 SOAP 进行可靠信息交换的标准。他第一次参与开源的研究可追溯到最初 Apache 的 SOAP 项目。Fremantle 获得过牛津大学的数学和哲学硕士，以及计算技术的理科硕士。



新推出的 WS02 Enterprise Service Bus 和你正着手进行的 Apache Synapse 项目有什么联系？

Paul Fremantle: Apache Synapse 是核心运行的动力，而且如果我们想提高核心运行能力，需要执行的代码都在 Apache Synapse 中。我们并非是贬抑其他的核心代码，但根本上，核心运行是以 Apache 项目为基础。

那么 WS02 公司的价值观是什么？

Fremantle: 我们为 ESB 提供支持, 无论其用于高质量的商业培训、支撑还是服务。我们有一个图形用户界面。这是一个完全基于网络的用户界面, 允许用户对基础位置的 Synapse 进行配置、监控和管理。

这是一个基于 Ajax 的 Web 界面吗?

Fremantle: 是的, 它是一个基于 Ajax 的 Web 界面。它的作用之一是公开所有的 API 管理及服务, 因此你可以将其与其他界面区别出来。

这么说来, 这是一个将管理内置的 ESB?

Fremantle: 正是, 但它的源程序是完全开放的, 包括管理控制程序也是。

还有没有其他的特点?

Fremantle: 我们有两样法宝。第一, 我们在常数存储器中编写信息, 而不是套用存储库中的大型信息树或信息模型。不过在某些情况下, 也必须在存储库中建立信息模型。信息不可能都被流化, 但只要是可流化的, 我们就可以做到。第二, 我们有一个完全无阻碍的运输模式。因此, 我们可以处理大量的通信连接, 而不会用光所有线程或是受到阻碍。我们非常重视提高运行能力时运行的稳定性, 同时也注重通过采用某种清洁、简单的模式提高其简易性。

一些厂商试图提供开源 ESB 和所谓的“闭源” ESB。你怎样看待?

Fremantle: 当我在 IBM 工作的时候, 我接触过开源和闭源共存的情况。我发现客户总是很难分辨两者之间哪一个更好, 特别是在过去几年中, 开源各项性能的品质不断上升。这是为什么我们没有推出企业版、标准版、免费版和共享版中任何一个版本的一个原因。我们只生产简单的开源产品, 客户可以直接购买。我们倒认为这对客户来说更为简

便。

将来会不会有闭源 ESBs 的应用空间？

Fremantle: 我认为，有一些产品会非常适合用闭源 ESB。举例来说，有一些金融机构有超高的通信要求性能。如果每秒需要处理上百万条信息，就需要非常复杂的高度调试软件产品。而这个产品的市场可能是 30 到 100 个客户，也就是说它不是一个开放的市场。所以，如果我要生产这么一个产品的话，我不会使用开源。

但是另一方面，ESB 正成为解决问题的不二之选。即使是小公司也看到了 ESB 的好处。因此，我认为开源正在完全接手这类市场。所以，既然两三个高质量的开源产品能够很好的解决问题，而且有更低的获取成本，为什么要使用专利产品，限定于某一个特定的供应商呢？

在过去的一年里，应用于 SOA 的开源软件是否取得了进步？

Fremantle: 是的。我认为我们的第一组软件很稳定。我们在 2007 年已经为 SOA 提供了一个更为坚实的平台。现在我们正在同更大的机构洽谈，比如说财富 500 强公司。他们表示正在认真考虑将开源用于 SOA。

除了 ESB，你觉得将开源应用于 SOA 方面，还有哪些新的项目或新技术特别能引起你的兴趣？

Fremantle: 我们最近刚启动了基于 REST 的 SOA 注册项目。现在有开源的 UDDI 项目和基于 ebXML 的项目，但是仍然有很多人买很昂贵的专利产品。我发现相对于我们的那个项目，UDDI 和 ebXML 两者都是过于沉重和复杂的解决方法。因此我们回头去找最初的原理，等我们再回来查看的时候，我们认识到，面对资料库/存储库时，Web 资源才是最根本、

最重要的。这才是真正的管理资源。所以，这个想法引领我们采用了 REST 模式。这就是为什么我们要建设一个完全基于 REST 的资料库/存储库。

(作者: Rich Seeley 译者: Eric 来源: TT 中国)