参考：<https://www.ibm.com/developerworks/cn/xml/x-sisoap/>

本文对 SOAP 作了一个初步介绍，给出几个简单示例；接着比较 CORBA，DCOM/COM 与 SOAP 的联系与区别；然后浅析 SOAP 简单的理解为 RPC+HTTP+XML 时的运行机制；最后展现 SOAP 的前景。



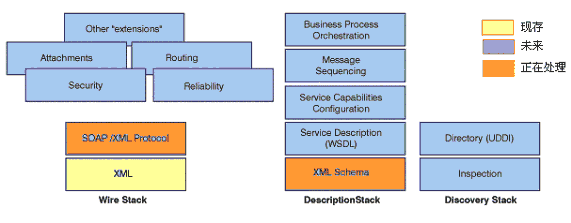
## 一：为什么需要SOAP？

随着计算机技术的不断发展，现代企业面临的环境越来越复杂，其信息系统大多数为多平台、多系统的复杂系统。这就要求今天的企业解决方案具有广泛的兼容能力，可以支持不同的系统平台、数据格式和多种连接方式，要求在Internet 环境下，实现系统是松散耦合的、跨平台的，与语言无关的，与特定接口无关的，而且要提供对Web 应用程序的可靠访问。

随着异种计算环境的不断增加，各种系统间的互操作性就愈显得必要，要求系统能够无缝地进行通信和共享数据，从而在 Internet 环境下，消除巨大的信息孤岛，实现信息共享、进行数据交换，达到信息的一致性。Web services 希望实现不同的系统之间能够用"软件-软件对话"的方式相互调用，打破了软件应用、网站和各种设备之间的格格不入的状态，实现"基于WEB无缝集成"的目标。

今年四月份的时候，W3C联盟召开了第一次 Web 服务专题研讨会，目的为探索 W3C应向哪个方向发展才能实现新兴的 Web 服务架构的标准化，期间提出了一个"Web 服务堆栈"的构想，如下图，从图中可以看出，SOAP在WEB服务堆栈中作为用于 XML 消息传递的一种非常普遍的协议，发挥着十分重要的作用。

##### 图 1：SOAP 在 WEB 服务堆栈中的作用



[**回页首**](https://www.ibm.com/developerworks/cn/xml/x-sisoap/#ibm-pcon)

## 二：什么是SOAP？

SOAP（Simple Object Access Protocol ）简单对象访问协议是在分散或分布式的环境中交换信息的简单的协议，是一个基于XML的协议，它包括四个部分：SOAP封装(envelop)，封装定义了一个描述消息中的内容是什么，是谁发送的，谁应当接受并处理它以及如何处理它们的框架；SOAP编码规则（encoding rules），用于表示应用程序需要使用的数据类型的实例; SOAP RPC表示(RPC representation)，表示远程过程调用和应答的协定;SOAP绑定（binding），使用底层协议交换信息。

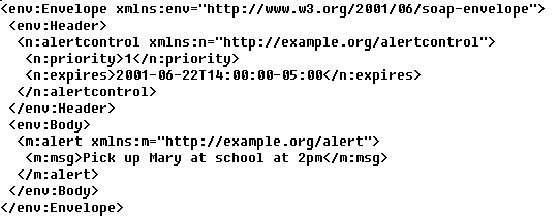
虽然这四个部分都作为SOAP的一部分，作为一个整体定义的，但他们在功能上是相交的、彼此独立的。特别的，信封和编码规则是被定义在不同的XML命名空间(namespace)中，这样使得定义更加简单。

SOAP的两个主要设计目标是简单性和可扩展性。这就意味着有一些传统消息系统或分布式对象系统中的某些性质将不是SOAP规范的一部分。比如：分布式垃圾收集 (Distributed garbage collection)、成批传送消息(Boxcarring or batching of messages)、对象引用 (Objects-by-reference（which requires distributed garbage collection）)、对象激活 (Activation（which requires objects-by-reference）)。

### SOAP消息举例：

1．第一个例子阐明了SOAP中一个简单的通信信息，包括了两个不是SOAP定义而是应用程序定义的元素：头块元素alertcontrol 和体块元素alert。头块元素包括两个参数：priority 和expires。体块元素包括的是实际传送的信息。

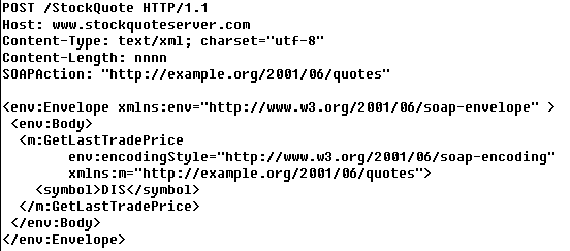
##### 图 2：SOAP 中一个简单的通信信息



（EXAMPLE 1）

2．SOAP通信与底层的不同协议和不同的交换格式有关，下面的例子SOAP使用HTTP作为底层通信协议，从而可以很好的使用request/response机制来传送信息。 SOAP/HTTP请求包括一个GetLastTradePrice的块元素，该请求携带一个字符串参数和ticker符号，在SOAP应答中返回一个浮点数。XML名域用来区分SOAP标志符和应用程序特定的标志符。

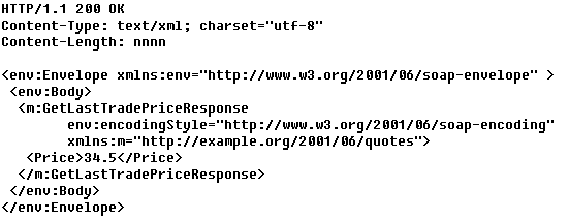
##### 图 3：使用 HTTP 作为底层通信协议



（EXAMPLE 2）

3. 例3 展示的是StockQuote SOAP服务信息，是对例2的请求作出的一条应答消息。

##### 图 4：对例 2 的请求作出的一条应答消息



（EXAMPLE 3）

## 三：SOAP与CORBA，COM/DCOM的区别？

在SOAP刚刚提出来的时候，许多人就提出了疑问：SOAP与CORBA和DCOM的区别何在？

1. CORBA（Common Object Request Broker Architecture）公共对象请求代理体系结构是由OMG组织制订的一种标准的面向对象应用程序体系规范。由对象请求代理ORB、对象服务、公共设施、域接口和应用接口这几个部分组成。其核心部分是对象请求代理ORB（Object Request Broker）。ORB提供了一种机制，通过这种机制，对象可以透明的发出请求和接收响应。分布的、可以互操作的对象可以利用ORB构造可以互操作的应用。ORB可看作是在对象之间建立客户/服务关系的一种中间件。基于ORB，客户可以透明的调用服务对象提供的方法，该服务对象可以与客户运行在同一台机器上，也可以运行在其他机器上通过网络与客户进行交互。ORB截取客户发送的请求，并负责在该软件总线上找到实现该请求的服务对象，然后完成参数、方法调用，并返回最终结果。CORBA 1.1 由对象管理组织在 1991 年发布。定义了接口定义语言（IDL）和应用编程接口（API），从而通过实现对象请求代理（ORB）来激活客户/服务器的交互。CORBA 2.0 于 1994 年的 12 月发布定义了如何跨越不同的 ORB 提供者而进行通讯。
2. COM/DCOM(Component Object Model / Distributed Component Object   
   Model )是微软公司提出的分布式组件对象模型标准，支持在局域网、广域网甚至Internet上不同计算机的对象之间的通讯。DCOM基于COM的应用程序、组件、工具等的基础之上，处理网络协议的低层次的细节问题，而不必关心太多的网络协议细节，从而使用户能够集中精力解决用户所要求的问题。DCOM位于应用程序的组件之间，将组件以不可见的方式胶合在一起组成具有完整功能的应用程序。
3. SOAP 与CORBA，DCOM/COM的比较。

3.1 首先指出的是SOAP不会取代CORBA，COM/DCOM，三者的概念有所区别。COM/DCOM是个组件模型标准，CORBA是分布式应用的服务标准。CORBA和DCOM为分布式应用程序建立服务，服务对象来执行客户端调用的服务。而SOAP是基于XML和HTTP的分布式对象的通信协议，是COM/DCOM和CORBA对象进行通讯的协议。实际上，利用SOAP的互操作性和CORBA强大的执行能力，两者可以很好的结合在一起。 OMG （Object Management Group responsible for the CORBA specification）正在关注这方面的发展。

3.2. CORBA应用程序和DCOM应用程序不能实现互操作，两者不能在一起协作。因为在ORPC(Object RPC)协议中，用ObjRef代表了一个正在运行对象的引用；在CORBA/IIOP(Internet Inter-Orb Protocol)中，用交换可互操作对象引用IOR（Interoperable Object Reference）代表一个服务器的对象引用。不幸的是，IOR 与 ObjRef不能够关联起来。然而，使用SOAP可以实现在垂直应用层面上CORBA ，DCOM技术的水平整合，能够更好的集成CORBA，DCOM为一个整体。

3.3 SOAP并没有定义信息的语义，服务质量，基于INTERNET的事务处理。而是采用 XML 进行消息编码，正确的处理需要服务器和客户端本身来执行，理解和执行彼此使用的信息格式（ONE-TO-ONE，REQUEST/REPLY，BROADCAST，ETC），应用程序本身在语义解析中扮演着十分重要的角色。而CORBA，DCOM表示了传送信息的语义，对参数和返回值使用二进制编码。可对诸如参数名称或类型的任何元信息都不编码，但使中介很难处理消息。又因为每个系统使用不同的二进制编码，系统间的互操作的很难实现。

3.4 尽管CORBA可以在不同的平台上执行，DCOM可以在微软的各种平台上运行，但是基于CORBA和DCOM的解决方案必须依赖于单一的应用程序。比如说，假如运行的是DCOM服务器程序，所有的分布式的客户端不得不运行于微软的操作平台上。CORBA 虽然可以运行于不同的平台，但CORBA的互操作性并没有在更高层的服务上进行扩展，如安全性和事务处理，在这种情况下，许多提供的服务没有得到很好的优化。DCOM和CORBA适合于服务器--服务器间的通讯，但是对于客户端--服务器的通讯十分脆弱，尤其当客户程序分布在INTERNET上更是如此。

3.5 SOAP不象DCOM一样试图定义分布式系统的所有元素，SOAP没有提供分布式类库，类型安全检查，版本控制等等，SOAP比它处于一个更低的层次，有点类似于IIOP在CORBA的作用，DCOM却提供了一些额外的协议功能，是IIOP 或者SOAP所不具备的。然而，许多. DCOM的额外功能只有在服务器��服务器间通信时才会用到，对于客户端��服务器之间的通信则是多余的。

[**回页首**](https://www.ibm.com/developerworks/cn/xml/x-sisoap/#ibm-pcon)

## 四：SOAP=RPC+HTTP+XML

SOAP简单的理解，就是这样的一个开放协议SOAP=RPC+HTTP+XML：采用HTTP作为底层通讯协议；RPC作为一致性的调用途径，ＸＭＬ作为数据传送的格式，允许服务提供者和服务客户经过防火墙在INTERNET进行通讯交互。RPC的描叙可能不大准确，因为SOAP一开始构思就是要实现平台与环境的无关性和独立性，每一个通过网络的远程调用都可以通过SOAP封装起来，包括DCE（Distributed Computing Environment ）　RPC CALLS，COM/DCOM CALLS, CORBA CALLS, JAVA CALLS，etc。

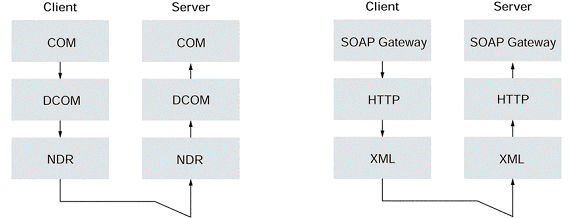
SOAP 使用 HTTP 传送 XML，尽管HTTP 不是有效率的通讯协议，而且 XML 还需要额外的文件解析（parse），两者使得交易的速度大大低于其它方案。但是XML 是一个开放、健全、有语义的讯息机制，而 HTTP 是一个广泛又能避免许多关于防火墙的问题，从而使SOAP得到了广泛的应用。但是如果效率（http的通信一般来讲有些慢，主要是因为http的三次握手保证连接的可靠性而消耗了很多的资源和时间，但是相比其他的协议可靠性高，如果对安全性要求高的话可以考虑）对你来说很重要，那么你应该多考虑其它的方式，而不要用 SOAP。

Me:理解起来，简单对象访问协议SOAP就是基于xml的http访问。之所以简单是指这些请求和响应的资源能够使用xml来很好的描述。

为了更好的理解SOAP,HTTP,XML如何工作的，不妨先考虑一下COM/DCOM的运行机制，DCOM处理网络协议的低层次的细节问题，如PROXY/STUB间的通讯，生命周期的管理，对象的标识。在客户端与服务器端进行交互的时候，DCOM采用NDR（Network Data Representation）作为数据表示，它是低层次的与平台无关的数据表现形式。

DCOM是有效的，灵活的，但也是很复杂的。而SOAP的一个主要优点就在于它的简单性，SOAP使用HTTP作为网络通讯协议，接受和传送数据参数时采用XML作为数据格式，从而代替了DCOM中的NDR格式，SOAP和 DCOM执行过程是类似的，如下图，但是用XML取代 NDR作为编码表现形式，提供了更高层次上的抽象，与平台和环境无关。

##### 图 5：用 XML 取代 NDR 作为编码表现形式



客户端发送请求时，不管客户端是什么平台的，首先把请求转换成XML格式，SOAP网关可自动执行这个转换。为了保证传送时参数，方法名，返回值的唯一性，SOAP协议使用了一个私有标记表，从而服务端的SOAP网关可以正确的解析，这有点类似于COM/DCOM中的桩（STUB）。转化成XML格式后，SOAP终端名（远程调用方法名）及其他的一些协议标识信息被封装成HTTP请求，然后发送给服务器。如果应用程序要求，服务器返回一个HTTP应答信息给客户端。与通常对HTML页面的HTTP GET请求不同的是，此请求设置了一些HTTP HEADER，标识着一个SOAP服务激发，和HTTP包一起传送。例如：对于一个询问股票价格的应用程序，服务器端具有组件提供某股票当前的价格，组件是COM或CORBA在服务器上建立的。客户端发送一个SOAP请求给服务器询问股票价格。服务器依赖于服务器上的SOAP网关，使用内嵌的HTML对象调用合适的方法， 然后把得到的价格通过SOAP应答传给客户端。

## 五．SOAP 的前景

W3C于2000年5月8日发表了Simple Object Access Protocol (SOAP) 1.1版本，具体规范发布在下列站点上（ <http://www.w3.org/TR/SOAP/>）。又与今年7月9号推出了SOAP Version 1.2版本的建议草案，具体规范发布在下列站点上（ <http://www.w3.org/TR/soap12/>）。编写SOAP Version 1.1版本的工作小组的成员包括： [DevelopMentor](http://www.develop.com/), [International Business Machines Corporation](http://www.ibm.com/), [Lotus Development Corporation](http://www.lotus.com/),[Microsoft](http://www.microsoft.com/), [UserLand Software](http://www.userland.com/)。

SOAP的推出是令人兴奋的。可以相信，随着网络服务的的不断发展，它将极大的改变我们的思考模式和开发模式。现在，已有许多大公司着手支持SOAP的开发，去年 IBM公司 和 Microsoft公司 都发行了实现 SOAP 的第一批版本。 IBM 公司启动了Apache SOAP 项目计划，微软最近又推出了SOAPtoolkit2.0的正式版，主要包括如下的一些特征：SOAP的高层接口和低层接口，消息对象接口，完全支持WSDL 1.1标准，支持用户自定义类型映射，并且提供了丰富和完整的开发文档以及应用实例。而且，两家公司正在互操作性方面努力研究。可以乐观的估计，不用多久，SOAP 互操作性的时代就将来临。

### 与SOAP相关的一些标准：

1. HTTP 1.0 or greater（ <http://www.w3.org/Protocols/HTTP/ietf-http-ext>）
2. the core W3C XML recommendation（ <http://www.w3.org/TR/1998/REC-xml-19980210>）
3. W3C XML namespace recommendation（ <http://www.w3.org/TR/REC-xml-names>）.
4. XML Schema（ <http://www.w3.org/TR/xmlschema-1/>）

##### 目前支持SOAP的一些公司产品：

| **Organization** | **Product** |
| --- | --- |
| Rogue Wave | Nouveau ORB |
| Iona | Orbix 2000 |
| ObjectSpace | Voyager |
| Digital Creations | Zope, the Python Application Server |
| UserLand | Frontier groupware product |
| Microsoft | Windows DNA 2000 |

SOAP是一个协议，与编程语言无关。实际上，许多语言已经开始支持SOAP，如：java,c/c++,vb,c#,perl,php.下面列出了在Java/C++/Perl/ADA/Python环境下SOAP的执行工具：

* **Java**: [Apache SOAP](http://www.soaprpc.com/faqs/), [DevelopMentor's implementation](http://www.develop.com/soap/soap.jar), [IdooXoap from ZVON](http://www.zvon.org/index.php?nav_id=30)
* **Python**: [PythonWare (client side only)](http://www.pythonware.com/products/soap)
* **C++**: [IdooXoap from ZVON](http://www.zvon.org/?nav_id=33)
* **Perl**: [SOAP::Lite](http://www.soaplite.com/)

鉴于SOAP是当前的新技术，国内资料贫乏，主要参考了国外的一些资料及IBM和MICROSOFT的相关资料，文章偏颇之处，请多指教！