# 一、Web 部分

122、说出Servlet 的生命周期，并说出Servlet 和CGI 的区别? 【基础】

答：Web 容器加载Servlet 并将其实例化后，Servlet 生命周期开始，容器运行其init 方法进行Servlet 的初始化，请求到达时运行其service 方法，service方法自动派遣运行与请求对应的doXXX 方法（doGet，doPost）等，当服务器决定将实例销毁的时候调用其destroy 方法。与cgi 的区别在于servlet 处于服务器进程中，它通过多线程方式运行其service 方法，一个实例可以服务于多个请求，并且其实例一般不会销毁，而CGI 对每个请求都产生新的进程，服务完成后就销毁，所以效率上低于servlet。

123、Servlet 的基本架构。【基础】

答：public class ServletName extends HttpServlet {

public void doPost(HttpServletRequest request,

HttpServletResponse response)

throws ServletException, IOException {

}

public void doGet(HttpServletRequest request,

HttpServletResponse response)

throws ServletException, IOException {

}

}

124、forward 和redirect 的区别? 【基础】

答：forward 是容器中控制权的转向，是服务器请求资源，服务器直接访问目标地址的URL，把那个URL 的响应内容读取过来，然后把这些内容再发给浏览器，浏览器根本不知道服务器发送的内容是从哪儿来的，所以它的地址栏中还是原来的地址。redirect 就是服务端根据逻辑,发送一个状态码,告诉浏览器重新去请求那个地址，一般来说浏览器会用刚才请求的所有参数重新请求，所以session,request 参数都可以获取，并且从浏览器的地址栏中可以看到跳转后的链接地址。前者更加高效，在前者可以满足需要时，尽量使用forward()方法，并且，这样也有助于隐藏实际的链接；在有些情况下，比如，需要跳转到一个其它服务器上的资源，则必须使用sendRedirect()方法。

125、JSP 中动态INCLUDE 与静态INCLUDE 的区别？【基础】

答：动态INCLUDE 用jsp:include 动作实现<jsp:include page="included.jsp"flush="true" />它总是会检查所含文件中的变化，适合用于包含动态页面，并且可以带参数；静态INCLUDE 用include 伪码实现,它不会检查所含文件的变化，适用于包含静态页面<%@ include file="included.htm" %>

126、说出数据连接池的工作机制是什么? 【基础】

答：J2EE 服务器启动时会建立一定数量的池连接，并一直维持不少于此数目的池连接。客户端程序需要连接时，池驱动程序会返回一个未使用的池连接并将其表记为忙。如果当前没有空闲连接，池驱动程序就新建一定数量的连接，新建连接的数量有配置参数决定。当使用的池连接调用完成后，池驱动程序将此连接表记为空闲，其他调用就可以使用这个连接。

127、JSP 的内置对象及方法？【基础】

答：request 表示HttpServletRequest 对象。它包含了有关浏览器请求的信息，并且提供了几个用于获取cookie, header 和session 数据的有用的方法。response 表示HttpServletResponse 对象，并提供了几个用于设置送回浏览器的响应的方法（如cookies,头信息等）。out 对象是javax.jsp.JspWriter 的一个实例，并提供了几个方法使你能用于向浏览器回送输出结果。pageContext 表示一个javax.servlet.jsp.PageContext 对象。它是用于方便存取各种范围的名字空间、servlet 相关的对象的API，并且包装了通用的servlet 相关功能的方法。session 表示一个请求的javax.servlet.http.HttpSession 对象。Session可以存贮用户的状态信息。application 表示一个javax.servle.ServletContext 对象。这有助于查找有关servlet 引擎和servlet 环境的信息。config 表示一个javax.servlet.ServletConfig 对象。该对象用于存取servlet 实例的初始化参数。page 表示从该页面产生的一个servlet 实例。

128、JSP 的常用指令？【基础】

答：<%@page language=”java” contenType=”text/html;charset=gb2312”

session=”true” buffer=”64kb” autoFlush=”true” isThreadSafe=”true”

info=”text” errorPage=”error.jsp” isErrorPage=”true” isELIgnored=”

true” pageEncoding=”gb2312” import=”java.sql.\*”%>

isErrorPage：是否能使用Exception 对象；isELIgnored：是否忽略EL 表达式；

<%@include file=”filename”%>

<%@taglib prefix=”c”uri=”http://……”%>

129、jsp 有哪些动作?作用分别是什么? 【基础】

答：JSP 共有以下6 种基本动作：

jsp:include：在页面被请求的时候引入一个文件；

jsp:useBean：寻找或者实例化一个JavaBean。；

jsp:setProperty：设置JavaBean 的属性。；

jsp:getProperty：输出某个JavaBean 的属性；

jsp:forward：把请求转到一个新的页面；

jsp:plugin：根据浏览器类型为Java 插件生成OBJECT 或EMBED 标记。

130、jsp 有哪些内置对象?作用分别是什么? 【基础】

答：JSP 共有以下9 种基本内置组件（可与ASP 的6 种内部组件相对应）：

request：用户端请求，此请求会包含来自GET/POST 请求的参数；

response：网页传回用户端的回应；

pageContext：网页的属性是在这里管理；

session：与请求有关的会话期；

application：servlet 正在执行的内容；

out：用来传送回应的输出；

config：servlet 的构架部件；

page：JSP 网页本身；

exception：针对错误网页，未捕捉的例外。

131、get 和post 的区别？【基础】

答：Form 中的get 和post 方法，在数据传输过程中分别对应了HTTP 协议中的GET 和POST 方法。二者主要区别如下：

1）Get 是用来从服务器上获得数据，而Post 是用来向服务器上传递数据；

2）Get 将表单中数据按照variable=value 的形式，添加到action 所指向的URL 后面，并且两者使用“?”连接，而各个变量之间使用“&”连接；Post 是将表单中的数据放在form 的数据体中，按照变量和值相对应的方式，传递到action所指向URL；

3）Get 是不安全的，因为在传输过程，数据被放在请求的URL 中；Post 的所有操作对用户来说都是不可见的；

4）Get 传输的数据量小，这主要是因为受URL 长度限制；而Post 可以传输大量的数据，所以在上传文件只能使用Post；

5）Get 限制Form 表单的数据集必须为ASCII 字符，而Post 支持整个ISO10646字符集；

6）Get 是Form 的默认方法。

132、什么情况下调用doGet()和doPost()？【基础】

答：Jsp 页面中的form 标签里的method 属性为get 时调用doGet()，为post时调用doPost()。

133、如何从form 表单中得取checkbox 的值；【基础】

答：可在页面把checkbox 的name 属性取同一个，value 属性取每个条目的id,后台用getParamter(“name”)能取到checkbox 的一组值。

134、页面中有一个命名为bank No 的下拉列表，写脚本获取当前选项的索引值。【基础】

答：用java 或javaScript 的处理方式分别如下：

Java： request.getParameter(“bank No”);

javaScript：

var selectItems = document.getElementsByName(“bank No”);

selectItems[0].value;

135、javascript 常用的方面；【基础】

答：常用于数据输入校验和页面特殊效果等。

136、常用的web 容器和开发工具；【基础】

答：最常用的容器包括：tomcat、weblogic；开发工具有：eclipse,jbuilder。

137、请画出Servlet 2.2 以上Web Application 的基本目录结构（2 分钟）【基础】

答：目录结构如下图所示：

webapps

|

Applocation

|

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| |

JSP 页面WEB-INF

|

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| | |

classes lib web.xml

138、JSP 和Servlet 有哪些相同点和不同点，他们之间的联系是什么？【基础】

答：JSP 是Servlet 技术的扩展，本质上是Servlet 的简易方式，更强调应用的外表表达。JSP 编译后是"类servlet"。Servlet 和JSP 最主要的不同点在于，Servlet 的应用逻辑是在Java 文件中，并且完全从表示层中的HTML 里分离开来。而JSP 的情况是Java 和HTML 可以组合成一个扩展名为.jsp 的文件。JSP 侧重于视图，Servlet 主要用于控制逻辑。

139、jsp 的四种范围？【基础】

答：a.page 是代表与一个页面相关的对象和属性。一个页面由一个编译好的Java servlet 类（可以带有任何的include 指令，但是没有include 动作）表示。这既包括servlet 又包括被编译成servlet 的JSP 页面b.request 是代表与Web 客户机发出的一个请求相关的对象和属性。一个请求可能跨越多个页面，涉及多个Web 组件（由于forward 指令和include 动作的关系）c.session 是代表与用于某个Web 客户机的一个用户体验相关的对象和属性。一个Web 会话可以也经常会跨越多个客户机请求d.application 是代表与整个Web 应用程序相关的对象和属性。这实质上是跨越整个Web 应用程序，包括多个页面、请求和会话的一个全局作用域。

140、Request 对象的主要方法? 【基础】

答：setAttribute(String name,Object)：设置名字为name 的属性值

getAttribute(String name)：返回由name 指定的属性值

getAttributeNames()：返回request 对象所有属性的名字集合(枚举)

getCookies()：返回客户端的所有Cookie 对象，结果是一个Cookie 数组

getCharacterEncoding()：返回请求中的字符编码方式

getContentLength()：返回请求的Body 的长度

getHeader(String name)：获得HTTP 协议定义的文件头信息

getHeaders(String name)：返回指定名的request Header 的所有值(枚举)

getHeaderNames()：返回所有request Header 的名字(枚举)

getInputStream()：返回请求的输入流，用于获得请求中的数据

getMethod()：获得客户端向服务器端传送数据的方法

getParameter(String name)：获得客户端请求中传送的name 指定的参数值

getParameterNames()：获得客户端传送给服务器端的所有参数的名字(枚

举)

getParameterValues(String name)：获得有name 指定的参数的所有值

getProtocol()：获取客户端向服务器端传送数据所依据的协议名称

getQueryString()：获得查询字符串

getRequestURI()：获取发出请求字符串的客户端地址

getRemoteAddr()：获取客户端的IP 地址

getRemoteHost()：获取客户端的名字

getSession([Boolean create])：返回和请求相关Session

getServerName()：获取服务器的名字

getServletPath()：获取客户端所请求的脚本文件的路径

getServerPort()：获取服务器的端口号

removeAttribute(String name)：删除请求中的一个属性

141、如何实现servlet 的单线程模式？【基础】

答：<%@page isThreadSafe=”false”%>

142、页面间对象传递的方法。【基础】

答：request,session,application,cookie 等。

143、详细描述MVC。【基础】

答：基于Java 的Web 应用系统采用MVC 架构模式，即model（模型）、view（视图）、control（控制）分离设计；这是目前WEB 应用服务系统的主流设计方向。

Model：即处理业务逻辑的模块，每一种处理一个模块；

View：负责页面显示，显示MODEL 处理结果给用户，主要实现数据到页面转换过程；

Control：负责每个请求的分发，把FORM 数据传递给MODEL 处理，把处理结果的数据传递给VIEW 显示。

144、MVC 的各个部分都有那些技术来实现?如何实现? 【基础】

答：MVC 是Model－View－Controller 的简写。"Model" 代表的是应用的业务逻辑（通过JavaBean，EJB 组件实现）， "View" 是应用的表示面（由JSP 页面产生），"Controller" 是提供应用的处理过程控制（一般是一个Servlet），通过这种设计模型把应用逻辑，处理过程和显示逻辑分成不同的组件实现。这些组件可以进行交互和重用。

145、应用服务器有那些？【基础】

答：BEA WebLogic Server, IBM WebSphere Application Server, Oracle9i Application Server, JBoss, Tomcat。

146、Servlet 执行时一般实现哪几个方法？【基础】

答：public void init(ServletConfig config)

public ServletConfig getServletConfig()

public String getServletInfo()

public void service(ServletRequest request,ServletResponse

response)

public void destroy()

147、struts 的入口类？【基础】

答：是ActionServlet，所有的struts 请求都经由该类转发处理。

148、STRUTS 的应用(如STRUTS 架构)？【基础】

答：Struts 是采用Java Servlet/JavaServer Pages 技术开发Web 应用程序的开放源码的framework。采用Struts 能开发出基于MVC(Model-View-Controller)设计模式的应用构架。Struts 有如下的主要功能：

1)包含一个controller servlet，能将用户的请求发送到相应的Action 对象；

2)JSP 自由tag 库，并且在controller servlet 中提供关联支持，帮助开发人员创建交互式表单应用；

3)提供了一系列实用对象：XML 处理、通过Java reflection APIs 自动处理JavaBeans 属性、国际化的提示和消息。

149、几种会话跟踪技术？【基础】

答：cookie、URL 重写、设置表单隐藏域。

150、BS 与CS 的联系与区别？【基础】

答：C/S 是Client/Server 的缩写，是客户机与服务器结构的应用程序，服务器通常采用高性能的PC、工作站或小型机，并采用大型数据库系统，如Oracle、Sybase、Informix 或SQL Server。客户端需要安装专用的客户端软件。B/Ｓ是Brower/Server 的缩写，是浏览器和服务器结构的应用程序，即Web 应用程序，客户机上只要安装一个浏览器（Browser），如Netscape Navigator 或InternetExplorer，服务器安装Oracle、Sybase、Informix 或SQL Server 等数据库。在这种结构下，用户界面完全通过WWW 浏览器实现，一部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现。浏览器通过Ｗeb Server 同数据库进行数据交互。

C/S 与B/S 区别：

1)硬件环境不同:

C/S 一般建立在专用的网络上, 小范围里的网络环境, 局域网之间再通过专门服务器提供连接和数据交换服务；B/S 建立在广域网之上的, 不必是专门的网络硬件环境,例与电话上网, 租用设备. 信息自己管理. 有比C/S 更强的适应范围, 一般只要有操作系统和浏览器就行；

2)对安全要求不同：

C/S 一般面向相对固定的用户群, 对信息安全的控制能力很强. 一般高度机密的信息系统采用C/S 结构适宜. 可以通过B/S 发布部分可公开信息；B/S 建立在广域网之上, 对安全的控制能力相对弱, 可能面向不可知的用户；

3)对程序架构不同：

C/S 程序可以更加注重流程, 可以对权限多层次校验, 对系统运行速度可以较少考虑；B/S 对安全以及访问速度的多重的考虑, 建立在需要更加优化的基础之上.比C/S 有更高的要求B/S 结构的程序架构是发展的趋势, 从MS 的.Net 系列的BizTalk 2000 Exchange 2000 等, 全面支持网络的构件搭建的系统. SUN 和IBM 推的JavaBean 构件技术等,使B/S 更加成熟；

4)软件重用不同：

C/S 程序可以不可避免的整体性考虑, 构件的重用性不如在B/S 要求下的构件的重用性好；B/S 对的多重结构,要求构件相对独立的功能. 能够相对较好的重用.就入买来的餐桌可以再利用,而不是做在墙上的石头桌子；

5)系统维护不同：

C/S 程序由于整体性, 必须整体考察, 处理出现的问题以及系统升级. 升级难. 可能是再做一个全新的系统；B/S 构件组成,方面构件个别的更换,实现系统的无缝升级. 系统维护开销减到最小.用户从网上自己下载安装就可以实现升级；

6)处理问题不同：

C/S 程序可以处理用户面固定, 并且在相同区域, 安全要求高需求, 与操作系统相关. 应该都是相同的系统；B/S 建立在广域网上, 面向不同的用户群, 分散地域, 这是C/S 无法作到的. 与操作系统平台关系最小；

7)用户接口不同：

C/S 多是建立的Window 平台上,表现方法有限,对程序员普遍要求较高；B/S 建立在浏览器上, 有更加丰富和生动的表现方式与用户交流. 并且大部分难度减低,减低开发成本；

8)信息流不同：

C/S 程序一般是典型的中央集权的机械式处理, 交互性相对低；B/S 信息流向可变化, B-B B-C B-G 等信息、流向的变化, 更像交易中心。

151、过滤器有哪些作用？【基础】

答：可以验证客户是否来自可信的网络，可以对客户提交的数据进行重新编码，可以从系统里获得配置的信息，可以过滤掉客户的某些不应该出现的词汇，可以验证用户是否登录，可以验证客户的浏览器是否支持当前的应用，可以记录系统的日志等等。

152、过滤器的用法？（对客户端的请求统一编码和对客户端进行认证）【基础】

答：首先要实现（implements）Filter 接口，同时覆盖Filter 接口的三个方法：

init(FilterConfig config) //用于获得FilterConfig 对象；

doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) //进行过滤处理一些业务；

destroy() //销毁Filter。

153、简述HttpSession 的作用、使用方法，可用代码说明。（3 分钟）【基础】

答：HttpSession 中可以跟踪并储存用户信息，把值设置到属性中，有2 个方法：setAttribute(),getAttrribute()；

例如：在一个方法中用session.setAttribute(“student”,student);在session 中设置一个属性名为student,值为一个名为student 的对象。而后可在同一session 范围内用getAttribute(“student”)取出该属性，得到student对象。

154、介绍在JSP 中如何使用JavaBeans？【基础】

答：在JSP 中使用JavaBean 常用的动作有：

1）<jsp:useBean />：用来创建和查找bean 对象；

2）<jsp:setProperty />：用来设置bean 的属性，即调用其setXxx()方法；

3）<jsp:getProperty />：用来获得bean 的属性，即调用其getXxx()方法。

155、JSP 和Servlet 中的请求转发分别如何实现？【基础】

答：JSP 中的请求转发可利用forward 动作实现：<jsp:forward />；Serlvet 中实现请求转发的方式为：

getServletContext().getRequestDispatcher(path).forward(req,res)。

156、Web.Xml 的作用？【基础】

答：用于配置web 应用的信息；如listener、filter 及servlet 的配置信息等。

157、写出熟悉的JSTL 标签。【基础】

答：<c:if>、<c:choose>、<c: when>、<c: otherwise>、<c:forEach>、<c:set>。

158、说出struts 中的标签。【基础】

答：<bean:message /> <html:errors />

<bean:include /> <html:messages />

<bean:define /> <html:html>

<bean:write /> <html:form>

<bean:resource /> <html:link>

<bean:cokkie /> <html:text>

<bean:heder /> <logic:present />

<bean:parameter /> <logic:equal />

<bean:size />

159、JSP 标签的作用？如何定义？【中等难度】

答：作用：分离jsp 页面的内容和逻辑；

业务逻辑开发者可以创建自定义标签；

封装业务逻辑；

可重用并且易维护；

易于手工修改、易于工具维护；

提供简洁的语法；

定义:

写标签处理器；

写tld 文件；

讲标签处理器和tld 文件放到同一个包里面；

把jsp 页面和标签库配置部署在一起。

160、写一个自定义标签；【中等难度】

答：代码如下：

import javax.servlet.jsp.tagext.\*;

import javax.servlet.jsp.\*;

import java.io.\*;

public class TimeTag extends SimpleTagSupport{

private boolean isServer = true;

public void setServer(boolean isServer){

this.isServer = isServer;

}

public void doTag() throws JspException, IOException{

JspWriter out = getJspContext().getOut();

if(isServer) {

out.println(new java.util.Date());

}else{

out.println("<script language=\"javascript\">");

out.println("document.write(new Date());");

out.println("</script>");

}

}

}

161、javascript 的优缺点和内置对象；【中等难度】

答：1）优点：简单易用，与Java 有类似的语法，可以使用任何文本编辑工具编写，只需要浏览器就可执行程序，并且事先不用编译，逐行执行，无需进行严格的变量声明，而且内置大量现成对象，编写少量程序可以完成目标；

2）缺点：不适合开发大型应用程序；

3）Javascript 有11 种内置对象：Array、String、Date、Math、Boolean、Number、Function、Global、Error、RegExp、Object。

# 二、WebLogic 及其它：（共13 题：附加部分，超出授课范围）

1、如何给weblogic 指定大小的内存?

答：在启动Weblogic 的脚本中（位于所在Domian 对应服务器目录下的startServerName），增加set MEM\_ARGS=-Xms32m -Xmx200m，可以调整最小内存为32M，最大200M。

2、如何设定的weblogic 的热启动模式(开发模式)与产品发布模式?

答：可以在管理控制台中修改对应服务器的启动模式为开发或产品模式之一，或者修改服务的启动文件或者commenv 文件，增加set PRODUCTION\_MODE=true。

3、如何启动时不需输入用户名与密码?

答：修改服务启动文件，增加WLS\_USER 和WLS\_PW 项；也可以在boot.properties文件中增加加密过的用户名和密码。

4、在weblogic 管理控制台中对一个应用域(或者说是一个网站,Domain)进行jms及ejb 或连接池等相关信息进行配置后,实际保存在什么文件中?

答：保存在此Domain 的config.xml 文件中，它是服务器的核心配置文件。

5、说说weblogic 中一个Domain 的缺省目录结构?比如要将一个简单的helloWorld.jsp 放入何目录下,然后在浏览器上打入http://主机:端口号//helloword.jsp 就可以看到运行结果了? 又比如这其中用到了一个自己写的javaBean 该如何办?

答：Domain 目录\服务器目录\applications，将应用目录放在此目录下将可以作为应用访问，如果是Web 应用，应用目录需要满足Web 应用目录要求，jsp 文件可以直接放在应用目录中，Javabean 需要放在应用目录的WEB-INF 目录的classes 目录中，设置服务器的缺省应用将可以实现在浏览器上无需输入应用名。

6、在weblogic 中发布ejb 需涉及到哪些配置文件？

答：不同类型的EJB 涉及的配置文件不同，都涉及到的配置文件包括ejbjar.xml,weblogic-ejb-jar.xml，CMP 实体Bean 一般还需要weblogic-cmprdbms-jar.xml

7、如何在weblogic 中进行ssl 配置与客户端的认证配置或说说j2ee(标准)进行ssl 的配置？

答：缺省安装中使用DemoIdentity.jks 和DemoTrust.jks KeyStore 实现SSL，需要配置服务器使用Enable SSL，配置其端口，在产品模式下需要从CA 获取私有密钥和数字证书，创建identity 和trust keystore，装载获得的密钥和数字证书。可以配置此SSL 连接是单向还是双向的。

8、如何查看在weblogic 中已经发布的EJB?

答：可以使用管理控制台，在它的Deployment 中可以查看所有已发布的EJB。

9、CORBA 是什么?用途是什么?

答：CORBA 标准是公共对象请求代理结构(Common Object Request BrokerArchitecture)，由对象管理组织(Object Management Group，缩写为OMG)标准化。它的组成是接口定义语言(IDL), 语言绑定(binding:也译为联编)和允许应用程序间互操作的协议。其目的为：用不同的程序设计语言书写在不同的进程中运行，为不同的操作系统开发。

10、在weblogic 中开发消息Bean 时的persistent 与non-persisten 的差别?

答：persistent 方式的MDB 可以保证消息传递的可靠性,也就是如果EJB 容器出现问题而JMS 服务器依然会将消息在此MDB 可用的时候发送过来，而nonpersistent方式的消息将被丢弃。

11、LINUX 下线程，GDI 类的解释？

答：LINUX 实现的就是基于核心轻量级进程的"一对一"线程模型，一个线程实体对应一个核心轻量级进程，而线程之间的管理在核外函数库中实现；GDI 类为图像设备编程接口类库。

12、Jdo 是什么?

答：JDO 是Java 对象持久化的新的规范，为java data object 的简称,也是一个用于存取某种数据仓库中的对象的标准化API。JDO 提供了透明的对象存储，因此对开发人员来说，存储数据对象完全不需要额外的代码（如JDBC API 的使用）。这些繁琐的例行工作已经转移到JDO 产品提供商身上，使开发人员解脱出来，从而集中时间和精力在业务逻辑上。另外，JDO 很灵活，因为它可以在任何数据底层上运行。JDBC 只是面向关系数据库（RDBMS）JDO 更通用，提供到任何数据底层的存储功能，比如关系数据库、文件、XML 以及对象数据库（ODBMS）等等，使得应用可移植性更强。

13、WEB SERVICE 名词解释；JAXP、JAXM 的解释；SOAP、UDDI,WSDL 解释？

答：Web Service 是基于网络的、分布式的模块化组件，它执行特定的任务，遵守具体的技术规范，这些规范使得Web Service 能与其他兼容的组件进行互操作；JAXP(Java API for XML Parsing)定义了在Java 中使用DOM, SAX, XSLT的通用的接口，这样在你的程序中你只要使用这些通用的接口，当你需要改变具体的实现时候也不需要修改代码；JAXM(Java API for XML Messaging)是为SOAP 通信提供访问方法和传输机制的API；WSDL 是一种XML 格式，用于将网络服务描述为一组端点，这些端点对包含面向文档信息或面向过程信息的消息进行操作。这种格式首先对操作和消息进行抽象描述，然后将其绑定到具体的网络协议和消息格式上以定义端点。相关的具体端点即组合成为抽象端点（服务）；SOAP 即简单对象访问协议(Simple Object Access Protocol)，它是用于交换XML 编码信息的轻量级协议；UDDI 的目的是为电子商务建立标准；UDDI 是一套基于Web 的、分布式的、为Web Service 提供的、信息注册中心的实现标准规范，同时也包含一组使企业能将自身提供的Web Service 注册，以使别的企业能够发现的访问协议的实现标准。