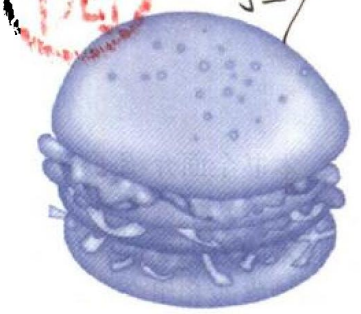
最新Web应用开发高级实例精解丛书

B/S 结构应用程序开发秘笈

【威比动感技术工作组】 陈 卫 编著

国防工业出版社

最新Web应用开发高级实例精解丛书



B/S结构应用程序开发秘笈

【威比动感技术工作组】陈卫　编著

国防工业出版社

•北京•

内容简介

本书全面介绍了 B/S（Browser/Server）结构应用程序的开发技术，并从实用的角度讲述了每一个读者都可能会碰到的问题及其解决方案。本书涵盖了基于 B/S 结构软件开发技术的方方面面，通过大量的示例，揭示了许多高难度的开发技巧。书中大量的源代码稍加修改和组合，即可应用到实际的开发项目当中，从而帮助你快速进入 B/S 结构应用程序开发的佳境！

B/S 结构是指分布式 Web 应用程序，它可以把应用逻辑放在服务器端集中进行控制，而客户端仅需要一个标准的浏览器。如果应用功能发生了变化，不再需要把新的程序分发到客户端，只需更新运行在服务器端的应用逻辑就可以了，用户则通过浏览器启动 Web 应用程序就会用上最新的版本。显然 B/S 结构应用程序相对于传统的 C/S 结构应用程序将是巨大的进步。

图书在版编目（CIP）数据

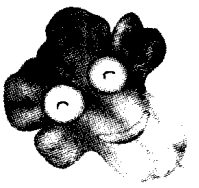
B/S 结构应用程序开发秘笈 / 陈卫编著. —北京：国防工业出版社，2001.10
（最新Web 应用开发高级实例精解丛书）
ISBN 7-118-02608-5

Ⅰ.B… Ⅱ.陈… Ⅲ.计算机网络—程序设计
Ⅳ.TP393

中国版本图书馆 CIP 数据核字（2001）第 046086 号

国防工业出版社出版发行
（北京市海淀区紫竹院南路 23 号）
（邮政编码 100044）
北京奥隆印刷厂印刷
新华书店经售
*
开本 787×1092 1/16 印张 13¾ 314 千字
2001 年 10 月第 1 版 2001 年 10 月北京第 1 次印刷
印数：1-4000 册 定价：29.80 元（含光盘）

前 言



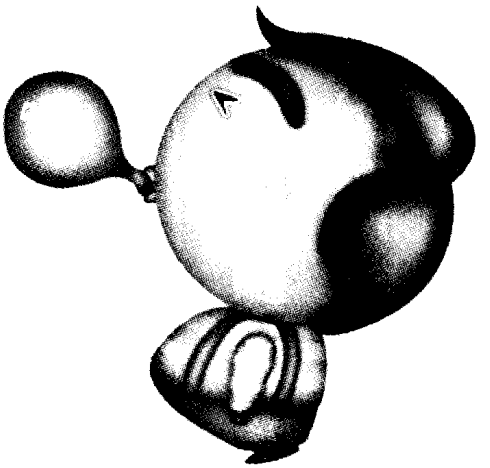
基于 B/S 结构的软件开发，是目前国内软件行业最热门的话题之一。但鉴于这方面的技术资料实在太少，普通的开发人员要想开发一个 B/S 结构的软件项目实在是困难重重。作者从事 B/S 结构软件开发已有两年多的时间，积累了相当丰富的实践经验，为了能让更多的人学会 B/S 结构软件开发技术，作者觉得有义务把多年来的编程心得和体会，以大量实例的形式展现给大家，以求共同推动中国 B/S 结构的软件开发水平。如果本书没有提供你所需要的内容，我们将会在下一版中尽力满足你的需求——在此欢迎你提出宝贵的建议。经过一版一版的不断完善，终有一天，我们会使这本书成为经典的 B/S 结构软件开发宝典。我们力图使本书每一次的再版能够引起大家浓厚的兴趣。

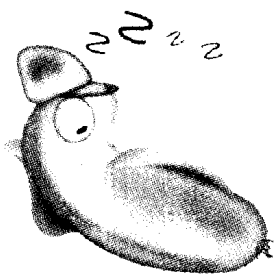
本书探讨的内容是在其他书中看不到的。在其他的书中大多是一些理论性的文章，忽略了实际开发的需求。本书则是针对实际开发的要求，提供了非常深入的程序代码。

要想通过本书获得尽量多的信息，读者应该具备一些 ASP、VBScript、JavaScript、SQLServer 等等的使用经验，但不必太多。在编写本书的时候，我们假设读者已具有 ASP 的一些基础知识，尽管要掌握这些必要的基础知识并不太难。如果在阅读本书第 1 章时确实有困难，应该在使用本书之前，参阅一下有关的入门书籍。

参加本书编著工作的有陈卫、杨前东、攀晓惠、江群、陈超、江显梅、张桂国、陈藕玲等同志。由于时间仓促，水平有限，本书难免存在疏漏或不妥之处，欢迎读者批评指正。

如果读者对本书有什么想法或建议，请与我们联系：VBBook@163.net





目 录

第1章

字符处理

1.1 客户端

1.1.1 删除字符串右侧空格符

JavaScript 脚本语言没有直接提供上述功能的字符串处理函数，因此必须自己编程。删除字符串右侧空格符常用的有两种方法。

1) 常规用法

```
function rTrim(s)
{
  var nPos=-1;
  if(s=="") return "";
  //从字符串右侧开始循环检查空格符
  for(var i=s.length-1;i>=0;i--)
   {
    //返回指定索引位置处的字符
    var c=s.charAt(i)
    //如果当前字符是空格或方格符，则指针变量加 1
    if(!(c==unescape('%20') || c==unescape('%A0'))) { nPos=i; break; }
   }
  //如果指针变量为-1，表示字符串右侧没有空格
  if(nPos==-1) return s;
  //返回去掉右侧空格符的字符串
  return s.substring(0,nPos+1);
}
```

示例：

```
var sVar = "vb.com   ";
alert("www."+rTrim(sVar)+".cn");  //显示："www.vb.com.cn"
```

2) 正则表达式

```
////////////////////////////////////////////////////////////////////////
//
//   采用正则表达式处理字符串的语法说明：
//   / /g   声明正则表达式，g 表示该表达式尽可能多的查找匹配
//   ( )    子表达式(简单的应用可以不用括号)
//   ^      匹配字符串的开始位置(左边)
//   $      匹配字符串的结束位置(右边)
//   \s     匹配任何空白字符
//   *      匹配前面的子表达式零次或多次
//
////////////////////////////////////////////////////////////////////////
String.prototype.rTrim = function()
{
    return this.replace(/(\s*$)/g, "");
}
```

1.1.2 删除字符串左侧空格符

删除字符串左侧空格符的方法和删除右侧空格符的方法类似。

1) 常规用法

```
function lTrim(s)
{
  var nPos=-1;
  if(s=="") return "";
  //从左侧往右循环检查空格，如果遇到的不是空格符则跳出循环
  for(var i=0;i<s.length;i++)
   {
    //返回指定字符下标(从 0 开始)的单个字符
    var c=s.charAt(i) ;
    //解码十六进制编码(%20：代表空格；%A0 代表方格符)
    //如果当前字符不是空格则记下它的位置，然后再中止循环
    if(!(c==unescape('%20') || c==unescape('%A0'))) { nPos=i; break; }
```

```
    }
    //如果左侧没有空格符，则原样返回结果
    if(nPos==-1) return s;
    //在循环中断的位置返回右侧的字符串
    return s.substring(nPos,s.length);
}
```

示例：

```
var sVar = "    vb.com";
alert("www."+lTrim(sVar)+".cn");  //显示："www.vb.com.cn"
```

2) 正则表达式

```
String.prototype.lTrim = function()
{
    return this.replace(/(^\s*)/g, "");
}
```

1.1.3 删除字符串左右两侧空格符

下面介绍的这个函数功能非常不错，不但可以去除字符串左右两侧的空格符，也可以完成删除左侧或右侧的空格符。

1) 常规用法

```
function allTrim(str){
    if(str==null || str=="")  return ""
    lIdx=0;rIdx=str.length;

  if (allTrim.arguments.length==2)
    {
      act=allTrim.arguments[1].toLowerCase()
    }else {
      act="all"
    }

    //循环检查字符串的两侧是否有空格，遇到空格符，则计数。
    //然后增量检查下一个字符
     for(var i=0;i<str.length;i++){
       //依次返回左侧一个字符，但条件是必须遇到空格符，
    //不然依旧返回上次的字符
     thelStr=str.substring(lIdx,lIdx+1)
```

```
    //依次返回右侧一个字符
      therStr=str.substring(rIdx,rIdx-1)
    //如果遇到空格,则左侧计数变量增加 1
    if ((act=="all" || act=="left") && thelStr==" "){
lIdx++
     }
    //如果遇到空格，则右侧计数变量减 1
    if ((act=="all" || act=="right") && therStr==" "){
            rIdx--
     }
  }
  //裁取指定开始和结束位置(中间)的字符串
   str=str.slice(lIdx,rIdx)
  return str
}
```

示例：

```
var sVar = "   vb.com   ";
alert("www."+allTrim(sVar)+".cn");          //显示："www.vb.com.cn"
alert("www."+allTrim(sVar,"left")+".cn");   //显示："www.vb.com   .cn"
alert("www."+allTrim(sVar,"right")+".cn");  //显示："www.   vb.com.cn"
```

2) 正则表达式

```
String.prototype.allTrim = function()
{
   return this.replace(/(^\s*)|(\s*$)/g, "");
}
```

1.1.4 读取字符串指定位置的字符(支持中文)

因为英文与中文字符字节数的不同，导至用 JavaScript 的 stringvar.substr（）函数不能精确地返回所需的内容。为了解决上述问题，特别设计了本函数。

```
function Substr(s,nStart,nEnd)
 {
   var i=0;
   //下标起始位置数
   var j=nStart;
   var nLen=s.length;
   //如果结束位置小于字符串长度，则用结束位置数赋值
   if(nEnd<nLen) nLen=nEnd
```

```
    //循环检查指定位置和长度的字符串中是否含有中文
    while(i<nLen)
      {
       //返回指定下标字符的编码，大于 256 表示是中文字符
       //如果是中文字符计数增量加 2，英文字符增 1
       if(s.charCodeAt(j)>256)
         {  i=i+2  }
       else
         {  i=i+1  }

       //下标变量增 1
       j=j+1
      }
    //如果计数变量大于字符串的长度，表示有中文字符，导致增量大过实际长度
    //因此下标变量要减 1
    if(i>nLen)
       j=j-1;
       //如果下标大于起始长度， 则返回有效的字符串
    if(j>nStart)
      return s.substring(nStart,j)
    else
      return ""
}
```

示例：

```
var sVar = "vb 威比动感技术工作组";
alert(Substr(sVar,0,2));   //显示："vb"
alert(Substr(sVar,0,3));   //显示："vb" (因为第 3 个字符是半个中文字符，顾舍去)
alert(Substr(sVar,0,4));  //显示："vb 威"
```

1.1.5 返回指定标签之间的字符

下面这个函数可以返回两个不同标签之间的字符。

```
function getTagValue(s,sStartTag,sEndTag)
{
  //获取开始标签的位置数
  var nPos1=s.indexOf(sStartTag);
```

```
  //检查是否找到标签
  if(nPos1<0) return "";

  //检查第二个标签的位置数
  var nPos2=s.indexOf(sEndTag,nPos1+2);

  //检查第二个标签位置是否有效
  if(nPos2<nPos1+sStartTag.length)
     return "";
  else
    //返回之间的字符
    return s.substring(nPos1+sStartTag.length,nPos2);
}
```

示例:

```
var sVar = "e-cw@yeah.net";
alert(getTagValue(sVar,"@","."));    //显示:“yeah”
```

1.1.6 获取当前网页主域名字符串

该函数可以接收不同的参数，而返回当前网页的主域名或当前网页的名称。

```
function getPath(nFlag)
 {
   var sP=""
   //注意：实际使用时请把下面这行代码的注解去掉
   //var sLoc=document.location.href
   //下面这行代码仅供测试用， 可删去
   var sLoc="http://www.vb.com.cn/default.asp"

   var nStart=sLoc.indexOf("http://")
   nStart=sLoc.indexOf("/",nStart+7)
   if(nStart>=0)
     { nEnd=sLoc.indexOf("/",nStart);
       if(nEnd>=0)
        {
          if(nFlag>0)
            sP=sLoc.substring(nStart)+"/";          //显示当前网页名称
          else
            sP=sLoc.substring(0,nEnd)+"/";          //显示网页主域名
```

```
        }
      }
     return sP
}
```

示例:

```
alert(getPath(0))  //显示："http://www.vb.com.cn/"
alert(getPath(1))  //显示："/default.asp/"
```

1.1.7 加入前导'0'字符

数值字符串前面要求加入“0”字符，这是在数值处理当中经常会碰到的问题。下面这个函数可以实现上述要求。

```
function getZero(theNum,nLen)
{
  var nZero="";
    //识别差值，据此循环加入前导空'0'
  for(var i=0;i<nLen-theNum.length;i++){ nZero+="0"; }
    //连接前导'0'字符串，返回数值字符串
  return nZero+theNum
}
```

示例:

```
var num="12345.1";
alert(getZero(num,10))  //显示："00012345.1"，含小数点共计10位长
```

1.1.8 加入后缀'0'字符

该函数可以指定小数的位数，如果位数不够则在小数后加入'0'数值字符补充。

```
function getFloat(theNum,nLen)
{
 var ss=theNum+""; var sR=""; var sZero=""
//累积后缀'0'字符
 for(i=0; i<nLen; i++) { sZero=sZero+"0"; }
//返回小数点在字符串中的位置数
 var Posd=ss.indexOf(".")
//加入浮点数
 if(Posd>=0)
   {
     //返回小数点左侧的整数部份
```

```
        sR=ss.substring(0,Posd);

        //用整数-连接-数字符串与后导'0'相接，然后截去多余
//的后导空'0'，得到实际的小数部份
        //注意：这里未用到四舍五入的方法
        sR=sR+"."+(ss+sZero).substring(Posd+1,Posd+nLen+1)
    } else {
      //如果没有小数， 则直接加入后导空'0'
      sR=ss+"."+sZero;
    }
  return sR
}
```

示例:

```
var num="12345.1";
alert(getFloat(num,4))     //显示："12345.1000"，小数点 4 位长
```

1.1.9 格式化日期字符

格式化日期字符是客户端基本的功能，下面这个函数可以让你指定不同的日期格式，从而完成不同的需求。后面章节中讲到的日历表单就会引用这个函数。

```
function formatDate(sDate)
  {
    if (sDate==null){ return ""; }
    var sDateSplit = "/";
    var nPos1 = sDate.indexOf("|"); if (nPos1==-1) { return ""; }
    //获取格式化字符
    var sFormat = sDate.substring(0,nPos1);

    //取得日期字符
    var sDate = sDate.substring(nPos1+1);

  //过滤出后面的分隔日期数据
  var nPos2 = sDate.indexOf("/");  if (nPos2==-1) { return ""; }
  var nPos3 = sDate.indexOf("/",nPos2+1);  if (nPos3==-1) { return ""; }

  var nX1    = parseInt(sDate.substring(0,nPos2));
  if (isNaN(nX1)) { nX1=0; }
  var nX2    = parseInt(sDate.substring(nPos2+1,nPos3));
  if (isNaN(nX2)) { nX2=0; }
```

```
var nX3    = parseInt(sDate.substring(nPos3+1));
if (isNaN(nX3)) { nX3=0; }
//根据日期格式检查日期字符串有无问题
switch (sFormat.toUpperCase())
 {
    case "DMY":  //日/月/年
       var D = nX1;     var M = nX2;     var Y = nX3;

      if (!checkDate(D,M,Y))
       {
         alert("(DD/MM/YYYY)\n Sample:31/12/2000 \n\n DateFormatError!");
               return "Error";
          } else {
               return D+sDateSplit+M+sDateSplit+Y;
          }
          break;

    case "YMD":  //年/月/日
         var Y    = nX1;     var M    = nX2;    var D    = nX3;

        //检查日期是否合法
     if (!checkDate(D,M,Y))
       {
        alert("(YYYY/MM/DD)\n Sample:2000/12/31 \n\n DateFormatError!");
           return "Error";
              } else {
                return Y+sDateSplit+M+sDateSplit+D;
              }
           break;
    }
 }
```

示例:

```
var sDate='YMD|1999/2/12';
alert(formatDate(sDate));     //显示:“1999/2/12”
var sDate='DMY|12/2/2000';
alert(formatDate(sDate));     //显示:“12/2/2000”
```

1.1.10 格式化日期字符合法性检查

这个函数主要是用来配合前面一个函数使用。

```
function checkDate(D,M,Y)
{
  var aMonthDays = new Array(31,28,31,30,31,30,31,31,30,31,30,31);
  var lLeap = false;

  if ((Y%4 ==0) && ((Y%100!=0) || (Y%400==0))) { lLeap = true; }

  if ((D<1) || (D>31) || (M<1) || (M>12) || (Y<1000)) {  return false; }

  if (D>aMonthDays[M-1] && !((M==2)&&(D>28))) { return false; }

  if (!lLeap && (M==2) && (D>28)) { return false; }

  if (lLeap && (M==2) && (D>29)) { return false; }
  return true;
}
```

1.1.11 JavaScript 语言调用 VBScript 内置函数

VBScript 中有很多非常好用的内置函数，例如：StrReverse()、Filter()、FormatNumber()、FormatDateTime()以及 FormatCurrency()等等。但是在 JavaScript 中却没有，那么能不能在一种脚本语言中调用另外一种脚本语言呢？答案是肯定的，但需要一个缓冲的中间函数，不能直接进行调用。除了跨语言实现函数调用外，实际上不同脚本语言中的变量，同样可以实现交互，具体怎么实现请看下面的代码。

```
<SCRIPT LANGUAGE="vbscript">
    '这个函数供 JavaScript 脚本语言调用，实现不同语言之间的交互
    function FormatValue(Value)
       FormatValue = FormatCurrency(Value,2)
       '显示 JavaScript 脚本中定义的变量值
       Msgbox  n
    end function
</SCRIPT>
<SCRIPT LANGUAGE=JavaScript>
<!--
    var n= 12355;
    //现在没有将 JScript 数组转换为 VBArray 的方法
    //var a = new Array("a","b");
```