**在输入中文条件进行翻页查询时乱码有关问题解决**

www.MyException.Cn   发布于：2012-11-08 08:48:11   浏览：10次

http://bdimg.share.baidu.com/static/images/type-button-1.jpg?cdnversion=20120831

在输入中文条件进行翻页查询时乱码问题解决  
问题描述：应用程序在输入中文条件后进行查询，当进行翻页和导出时，出现乱码并且不能导出数据。

问题处理：

      Tomcat服务器在处理请求时，对get和post两种请求方式采取了不同的处理方式，如果是post请求方式(也就是我们提交表单时常用的方式)则我们在web.xml中如果配置了用于处理字符编码的过滤器，此过滤器的行为将会对后续的请求处理产生影响，而如果是get方式(通常情况下是点击链接发送出的请求)过滤器的行为则不会对请求产生影响。  
  
     处理方法是：修改Tomcat服务器目录下%Tomcat%\conf\**server.xml**，增加一句

**URIEncoding="GB2312"**即可。  
原来配置：  
 <Connector port="8088"

maxHttpHeaderSize="8192"

maxThreads="150"

minSpareThreads="25"

maxSpareThreads="75"

enableLookups="false"

redirectPort="8443"

acceptCount="100"

connectionTimeout="20000"

disableUploadTimeout="true"/>

新配置：  
 <Connector port="8088"

maxHttpHeaderSize="8192"

maxThreads="150"

minSpareThreads="25"

maxSpareThreads="75"

enableLookups="false"

redirectPort="8443"

acceptCount="100"

connectionTimeout="20000"

disableUploadTimeout="true"

URIEncoding="GB2312"/>

重启服务器，乱码问题消失。

网友总结的POST和GET请求的处理方法：  
  
      表单提交时（post和Get方法），使用request.getParameter方法得到乱码，这是因为tomcat处理提交的参数时默认的是iso-8859-1，表单提交get和post处理乱码问题不同，下面分别说明。  
String strMenu =

**new** String(strMenu.getBytes("ISO-8859-1"),"UTF-8");  
  
  
(1)POST处理  
    对post提交的表单通过编写一个过滤器的方法来解决，过滤器在用户提交的数据被处理之前被调用，可以在这里改变参数的编码方式，过滤器的代码如下：

package example.util;

import java.io.IOException;

import javax.servlet.Filter;

import javax.servlet.FilterChain;

import javax.servlet.FilterConfig;

import javax.servlet.ServletException;

import javax.servlet.ServletRequest;

import javax.servlet.ServletResponse;

public class SetCharacterEncodingFilter implements Filter {

protected String encoding = null;

protected FilterConfig filterConfig = null;

protected boolean ignore = true;

public void destroy() {

this.encoding = null;

this.filterConfig = null;

}

public void doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) throws IOException, ServletException {

if (ignore || (request.getCharacterEncoding() == null)) {

String encoding = selectEncoding(request);

if (encoding != null) {

request.setCharacterEncoding(encoding);

}

}

// Pass control on to the next filter

chain.doFilter(request, response);

}

public void init(FilterConfig filterConfig) throws ServletException {

this.filterConfig = filterConfig;

this.encoding = filterConfig.getInitParameter("encoding");

String value = filterConfig.getInitParameter("ignore");

if (value == null) {

this.ignore = true;

} else if (value.equalsIgnoreCase("true")) {

this.ignore = true;

} else if (value.equalsIgnoreCase("yes")) {

this.ignore = true;

} else {

this.ignore = false;

}

}

protected String selectEncoding(ServletRequest request) {

return (this.encoding);

}

}

在web.xml中的配置：

<filter>

<filter-name>Encoding</filter-name>

<filter-class> example.util.SetCharacterEncodingFilter </filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>gbk</param-value>

<!--gbk或者gb2312或者utf-8-->

</init-param>

<init-param>

<param-name>ignore</param-name>

<param-value>true</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>Encoding</filter-name>

<servlet-name>/\*</servlet-name>

</filter-mapping>

(2) Get方法的处理  
tomcat对post和get的处理方法不一样，所以过滤器不能解决get的乱码问题，它需要在其他地方设置。  
打开<tomcat\_home>\conf目录下server.xml文件，找到对8080端口进行服务的Connector组件的设置部分，给这个组件添加一个属性：URIEncoding="GBK"。修改后的Connector设置为：

<Connector port="8088" maxHttpHeaderSize="8192"

maxThreads="150" minSpareThreads="25" maxSpareThreads="75"

enableLookups="false" redirectPort="8443" acceptCount="100"

connectionTimeout="20000" disableUploadTimeout="true" URIEncoding="GB2312"/>