5.6 Changing the Page According to How the User Got There

The Referer header designates the location of the page users were on when they clicked a link to get to the current page. If users simply type the address of a page, the browser sends no Referer at all and request.getHeader("Referer") returns null.

This header enables you to customize the page depending on how the user reached it. For example, you could use this header to do the following:

- Create a jobs/careers site that takes on the look and feel of the associated site that links to it.
- Change the content of a page depending on whether the link came from inside or outside the firewall. (Do not use this trick for secure applications, however; the Referer header, like all headers, is easily forged.)
- Supply links that take users back to the page they came from.
- Track the effectiveness of banner ads or record click-through rates from various different sites that display your ads.

Listing 5.5 shows a servlet that uses the Referer header to customize the image it displays. If the address of the referring page contains the string "JRun," the servlet displays the logo of Macromedia JRun. If the address contains the string "Resin," the servlet displays the logo of Caucho Resin. Otherwise, the servlet displays the logo of Apache Tomcat. The servlet also displays the address of the referring page.

Listing 5.6 shows the HTML pages used to link to the servlet. We created three *identical* pages named JRun-Referer.html, Resin-Referer.html, and Tomcat-Referer.html; the servlet uses the name of the referring page, not form data, to distinguish among the three. Recall that HTML pages are placed in the top-level directory of your Web application (or an arbitrary subdirectory thereof), whereas servlet code is placed in a subdirectory of WEB-INF/classes that matches the package name. So, for example, with Tomcat and the default Web application, the HTML pages are placed in *install_dir*/webapps/ROOT/request-headers/ and accessed with URLs of the form http://hostname/request-headers/Xxx-Referer.html.

Figures 5-6 through 5-9 show some representative results.

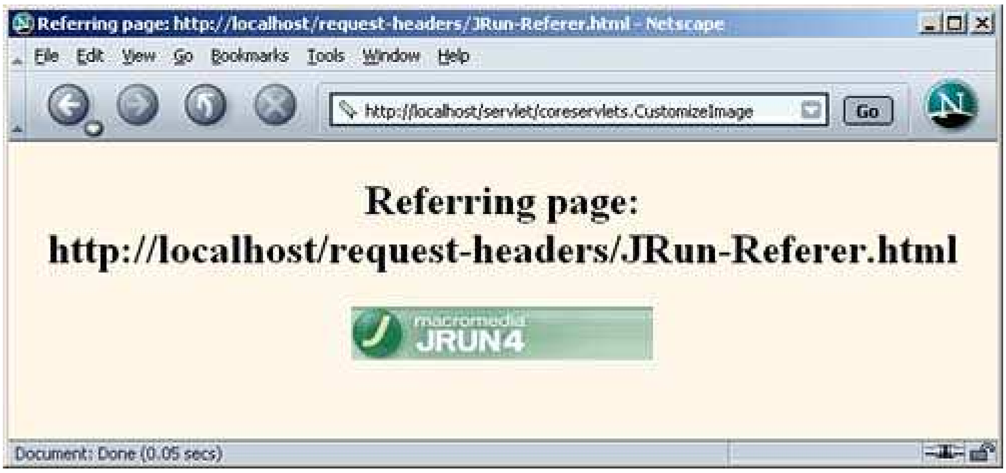
Listing 5.5 CustomizeImage.java

```
package coreservlets;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

/** Servlet that displays referer-specific image. */

public class CustomizeImage extends HttpServlet {
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String referer = request.getHeader("Referer");
    if (referer == null) {
      referer = "<I>none</I>";
    }
    String title = "Referring page: " + referer;
    String imageName;
    if (contains(referer, "JRun")) {
      imageName = "jrun-powered.gif";
    } else if (contains(referer, "Resin")) {
      imageName = "resin-powered.gif";
    } else {
      imageName = "tomcat-powered.gif";
    }
    String imagePath = "../request-headers/images/" + imageName;
    String docType =
      "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
      "Transitional//EN\">\n";
    out.println(docType +
                "<HTML>\n" +
                "<HEAD><TITLE>" + title + "</TITLE></HEAD>\n" +
                "<BODY BGCOLOR=\"#FDF5E6\">\n" +
                "<CENTER><H2>" + title + "</H2>\n" +
                "<IMG SRC=\"" + imagePath + "\">\n" +
                "</CENTER></BODY></HTML>");
  }

  private boolean contains(String mainString,
                           String subString) {
    return(mainString.indexOf(subString) != -1);
  }
}
```

Listing 5.6 JRun-Referer.html (identical to Tomcat-Referer.html and Resin-Referer.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD><TITLE>Referer Test</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H1 ALIGN="CENTER">Referer Test</H1>
Click <A HREF="/servlet/coreservlets.CustomizeImage">here</A>
to visit the servlet.
</BODY></HTML>
```

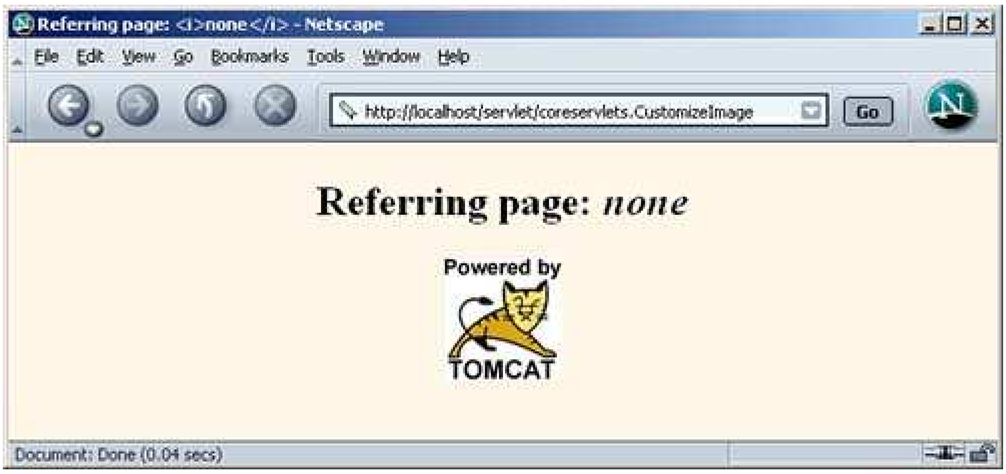
Figure 5-6. The `CustomizeImage` servlet when the address of the referring page contains the string "JRun."



Referring page: http://localhost/request-headers/JRun-Referer.html - Netscape
File Edit View Go Bookmarks Tools Window Help
http://localhost/servlet/coreservlets.CustomizeImage
Go
macromedia
JRUN4
Document: Done (0.05 secs)

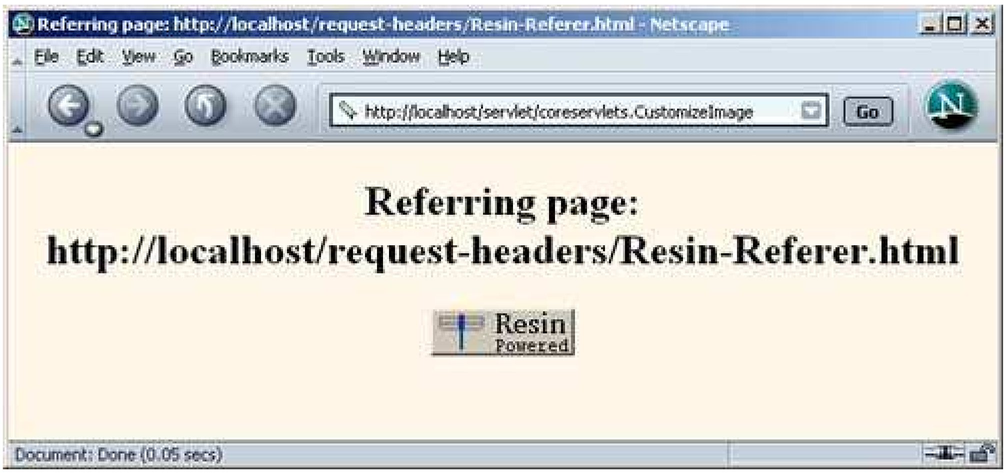
Referring page:
http://localhost/request-headers/JRun-Referer.html

Figure 5-9. The `CustomizeImage` servlet when the `Referer` header is missing. When using the `Referer` header, always handle the case in which the result of `getHeader` is `null`.


Referring page: <i>none</i> - Netscape
File Edit View Go Bookmarks Tools Window Help
http://localhost/servlet/coreservlets.CustomizeImage
Go
Powered by
TOMCAT
Document: Done (0.04 secs)

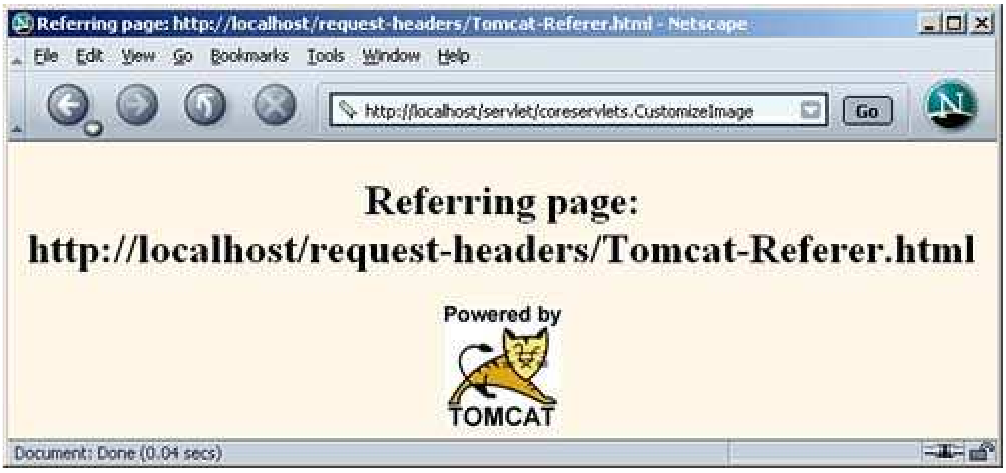
Referring page: *none*

Figure 5-7. The `CustomizeImage` servlet when the address of the referring page contains the string "Resin."


Referring page: http://localhost/request-headers/Resin-Referer.html - Netscape
File Edit View Go Bookmarks Tools Window Help
http://localhost/servlet/coreservlets.CustomizeImage
Go
Resin Powered
Document: Done (0.05 secs)

Referring page: http://localhost/request-headers/Resin-Referer.html

Figure 5-8. The `CustomizeImage` servlet when the address of the referring page contains neither "JRun" nor "Resin."


Referring page: http://localhost/request-headers/Tomcat-Referer.html - Netscape
File Edit View Go Bookmarks Tools Window Help
http://localhost/servlet/coreservlets.CustomizeImage
Go
Powered by
TOMCAT
Document: Done (0.04 secs)

Referring page: http://localhost/request-headers/Tomcat-Referer.html