Handling parameters and windows in Javascript

Philippe Martin
(philippe.martin@gu.edu.au)

Laboratory for Knowledge Visualisation and Ordering (KVO)
(http://www.int.gu.edu.au/kvo/)

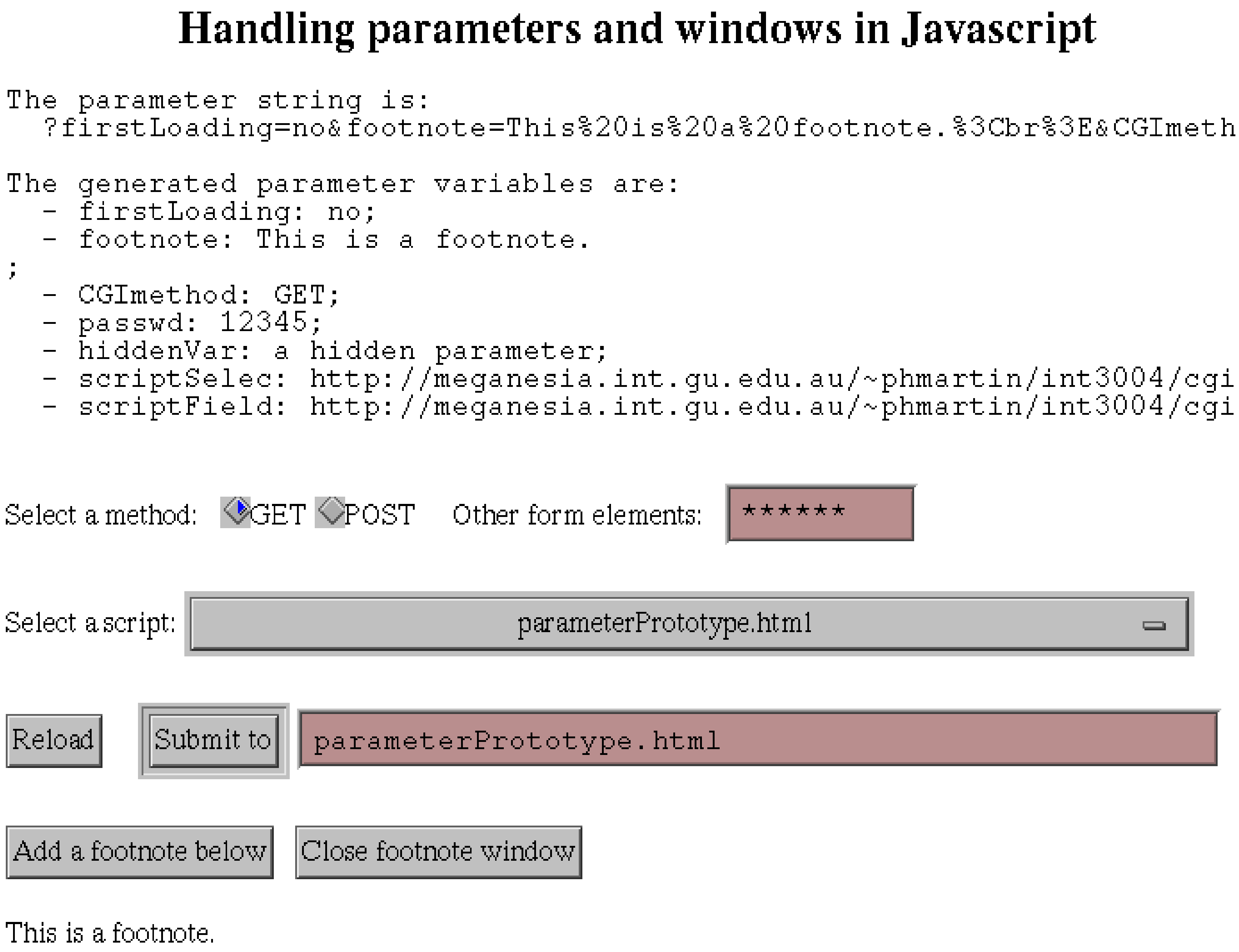
Tutorials
The Common Gateway Interface
The Common Gateway Interface: FORMS
Mosaic for X version 2.0 Fill-Out Form Support
The Basics

FAQs, newsgroup, and problems
CGI Programming OpenFAQ
CGI Security: Writing Secure CGI Programs
The WWW Security FAQ

Script prototypes
C Script storing its GET/POST parameters into an associative list
Shell script using a C program to decode the GET/POST parameters
Interface to test the previous scripts
Decoding function used in the first script
Directory for these scripts
Library to access files on the Net

Script repositories
CGI scripts examples
ScriptSearch (CGIs and other kinds of scripts in many languages)

Miscellaneous stuff related to the CGI
FastCGI
NCSA httpd Installation
An alternative to CGI: incorporating a Web server in the application
A counter



Handling parameters and windows in Javascript

```
The parameter string is:
  ?firstLoading=no&footnote=This%20is%20a%20footnote.%3Cbr%3E&CGImeth

The generated parameter variables are:
  - firstLoading: no;
  - footnote: This is a footnote.
;
  - CGImethod: GET;
  - passwd: 12345;
  - hiddenVar: a hidden parameter;
  - scriptSelec: http://meganesia.int.gu.edu.au/~phmartin/int3004/cgi
  - scriptField: http://meganesia.int.gu.edu.au/~phmartin/int3004/cgi
```

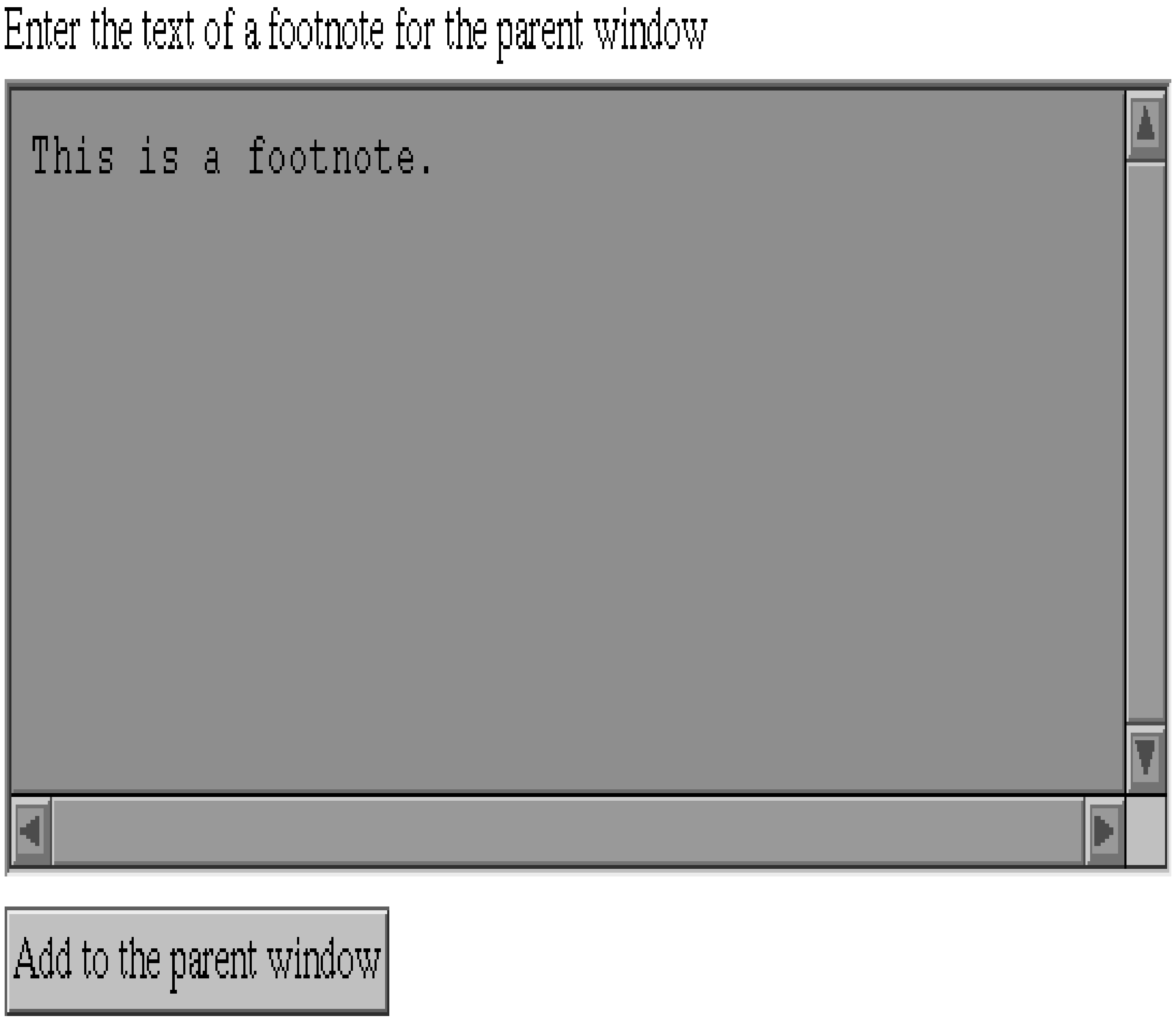
Select a method: ● GET ○ POST   Other form elements: ******

Select a script: parameterPrototype.html

Reload   Submit to   parameterPrototype.html

Add a footnote below   Close footnote window

This is a footnote.



Enter the text of a footnote for the parent window

This is a footnote.

```
?firstLoading=no&footnote=
    This%20is%20a%20footnote.%3Cbr%3E&   CGImethod=GET&
    passwd=12345&   hiddenVar=a%20hidden%20parameter&
    scriptSelec=http%3A//meganesia.int.gu.edu.au/%7Ephmartin/int3004/cgi/appli/printParameter
    scriptField=http%3A//meganesia.int.gu.edu.au/%7Ephmartin/int3004/cgi/appli/printParameter
```

```
<html>
<head><title>Interface to the CGI script examples</title>

<script language="javascript">
<!--

//*********** 1) GLOBAL VARIABLES ************

var Parameters= new AssociativeArray();
    Parameters.firstLoading="yes";
    Parameters.footnote="";
initParametersByDecoding(location.search);

var AnnotationWin=null;

function AssociativeArray ()  { return this; }
```

```
function /*void*/ initParametersByDecoding (paramStr)
{ if (!paramStr) return;           //paramStr:=location.search
  paramStr += ’&’; //so that the last parameter is ended by ’&’
  var l= paramStr.length;   var s="";  var attr="";

  for (var i=1; i<l; i++) //the 1st character is ’?’
  { var c= paramStr.charAt(i);
    if (c == ’+’)  { s += ’ ’; }
    else if (c == ’%’)   //escape sequence
    { s += unescape(paramStr.substring(i,i+3));  i+=2; }
    else if (c == ’=’)   //end of an attribute, the value follows
    { attr= s;   s=""; }
    else if (c == ’&’)   //end of an attribute-value pair
    { Parameters[attr]= (s) ? s:"";    s=""; }
    else s += c;
  }
}
```

```
<script language="javascript">
var doc=document; doc.write("<pre>\n");
if (!location.search)
{ doc.write("There are no parameter.\n\n");
  doc.write("By default, the parameter variables are:\n  ");
}
else
{ doc.write("The parameter string is:\n  " +
                  location.search + "\n\n");
  doc.write("The generated parameter variables are:\n  ");
}
doc.write(slotsOfObject(Parameters));
doc.write("</pre>\n");
</script>
```

```
<FORM name="myForm"
      onSubmit="this.action=this.scriptField.value; return true">
      <!-- target="resultWin" -->

<p>Select a method: &nbsp;

  <input type="RADIO" name="CGImethod" value="GET" checked
         onClick="this.form.method='GET';
                  Parameters.CGImethod='GET'">GET

  <input type="RADIO" name="CGImethod" value="POST"
         onClick="this.form.method='POST';
                  Parameters.CGImethod='POST'">POST
```

```
&nbsp; &nbsp; Other form elements:
&nbsp; <input type="PASSWORD" name="passwd"
              size=9 value="passwd"
              onChange='Parameters.passwd=this.value'>

&nbsp; <input type="HIDDEN" name="hiddenVar"
               value="a hidden parameter"
              onChange='Parameters.hiddenVar=this.value'>
```

```
<p>Select a script:
  <SELECT name="scriptSelec" onChange="
        Parameters.scriptSelec= selectedTextInSelectArray(this);
        Parameters.scriptField= Parameters.scriptSelec;
        this.form.scriptField.value= Parameters.scriptSelec;">
     <option>parameterPrototype.html
     <option>http://meganesia.int.gu.edu.au/~phmartin/int3004/cgi/appli/printParameters.cgi
     <option>http://meganesia.int.gu.edu.au/~phmartin/int3004/cgi/appli/echoParameters.cgi
   </SELECT>
```

```
<p><input type="BUTTON" value="Reload" onClick="reload()">
   <input type="SUBMIT" value="Submit to">
   <input type="TEXT" name="scriptField" size=50
          value="parameterPrototype.html"
          onChange="Parameters.scriptField=this.value;">

<p><input type="BUTTON" value="Add a footnote below"
          onClick="generateAnnotationWin()"> &nbsp;

   <input type="BUTTON" value="Close footnote window"
          onClick="AnnotationWin.close()">
```

```
<p><script language="javascript">
<!--
if (Parameters.footnote) document.write(Parameters.footnote);
//end of script -->
</script>

</FORM>
</body>
</html>
```

```
function /*void*/ generateAnnotationWin ()
{ AnnotationWin= window.open("","AnnotationWin");
  var doc= AnnotationWin.document;
  doc.write("<html>\n<head><title>Annotation</title></head>\n\n");
  doc.write("<body bgcolor='white'>\n<form>\n");
  doc.write('Enter the text of a footnote for the parent window\n');
  doc.write('<textarea name="noteArea" rows=10 cols=50></textarea>\n');
  doc.write('<input type="BUTTON" value="Add to the parent window"\n');
  doc.write('       onClick=\'window.opener.Parameters.footnote+=\n');
  doc.write('                     this.form.noteArea.value+"<br>";\n');
  doc.write('                window.opener.reload()\'\n');
  doc.write('>\n');
  doc.write("</form>\n</body>\n</html>\n");
  doc.close();  //AnnotationWin.opener= window;
}
```

```
function /*void*/ reload()
{ Parameters.firstLoading="no";
  location.href= mkNewHRefUsingParameters();
  //not "document.forms[0].submit()":
  //   opens a new window or increase the history list,
  //   and may cause problems if the window is resized
  //not "generateDocument()": removes variables and functions
}
```

```
function /*void*/ mkNewHRefUsingParameters()
  //HRef:= protocol//hostname:port pathname search hash
{ var s=""; var first=true;
  for (var attr in Parameters)
  { var value= Parameters[attr];
    if (first) first=false;  else s+=’&’;
    if (value) s += attr + ’=’ + escape(Parameters[attr]);
    else s += attr + ’=’;
  }
  var protocol= location.protocol;
  if (!protocol) protocol= "http://";
  else if (protocol.indexOf("//") < 0) protocol += "//";
  return protocol+location.host+location.pathname+’?’+s;
}
```

```
function /*bool*/ trr(obj)
  //recursively trace the object's slots and values
{ return confirm(recursContent(obj,""));
}
```

```
function /*String*/ recursContent(obj,indent)
{ var str=""; var s=String(obj);
  if ((s.charAt(0)==’[’) && (s.length>1))
  { for (var slot in obj)
    { str+= indent + "(\n" +
            recursContent(slot,indent+"   ") + ’\n’ +
            indent + ")\n";
      var value= obj[slot];
      if (!value)  str+=indent+"{undefined}\n";
      else str+= indent+"{\n"+
                 recursContent(obj[slot],indent+"   ")+
                 ’\n’ + indent + "}\n";
    }
    return str;
  }
  else return indent+s;
}
```

Security issues

- JavaScript holes: users privacy (files, e-mails, URLs)
- CGI installation: separate machine/directory/user
- Never allow the user to execute a command on the system
- Filter parameters
- Check robustness: use signals and dynamic buffers
- Encrypt: Java applets, SSL, SHTTP

`http://hostname/cgi-bin/perl.exe?-e+unlink+%3C*.*%3E%3B`