Parsing a URL Encoded String

If your HTML FORM has more than one input field, you have to parse CGI input to retrieve an individual field value.

Let's look at a more complex HTML document that is a front-end to another item inquiry application. This HTML FORM has various HTML elements such as checkboxes, radio buttons, and drop-down menus (see "2.1.1 How CGI Works" for the meaning of each HTML tag) to specify multiple search conditions. Each element has its own name to distinguish the input.

Figure 23 shows a sample FORM that uses many HTML FORM elements. Throughout this section, this form is used to pass parameters from the Web browser to CGI programs.
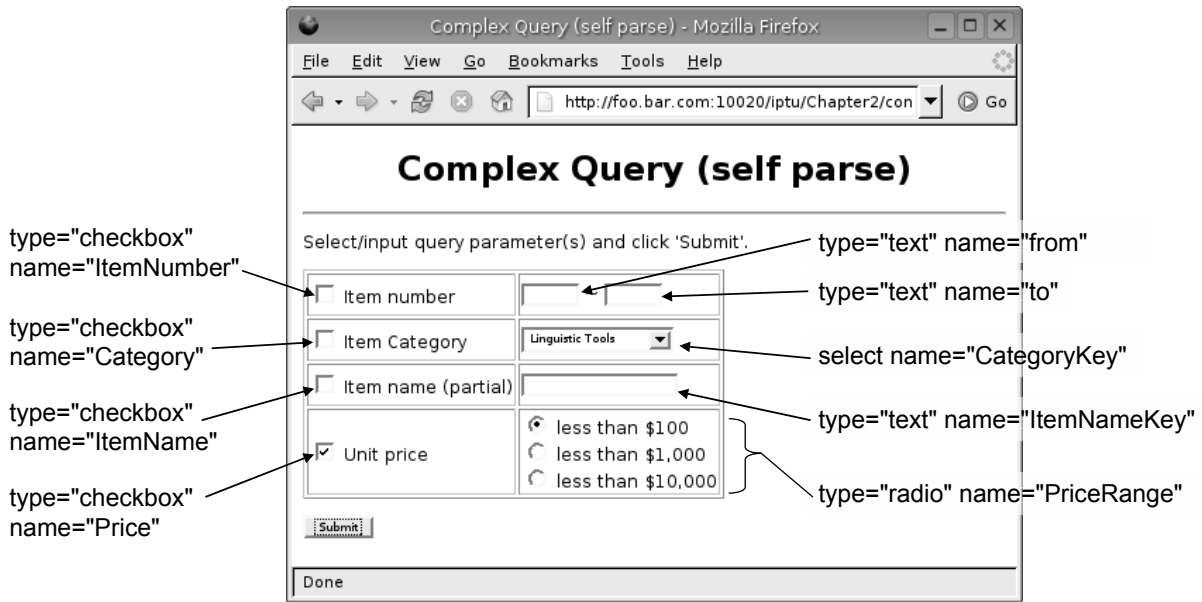

Figure 23 – A Complex FORM Example

The input to the CGI program (either standard input or environment variable QUERY_STRING) is a series of characters and may look like the following:

```
   > EVAL inbuff
     INBUFF =
              ....5...10...15...20...25...30...35...40...45...50...55...60
          1   'from=&to=&Category=on&CategoryKey=Linguistic+Tools&ItemNameK'
         61   'ey=&Price=on&PriceRange=1                                   '
```

For CGI programs to work with this string, you need to incorporate the logic that performs several tasks:

- Divide the string to individual fields by using the "&" character.

- Decode the triplet to representing characters and replace the "+" character with a space.

- Extract the variable value.

- Convert character code (for example, ASCII to EBCDIC) if necessary.

Multi-Occurrence Fields

You can use the same variable name for multiple fields in HTML specifications. Figure 24 shows a simple example of a multi-occurrence field.
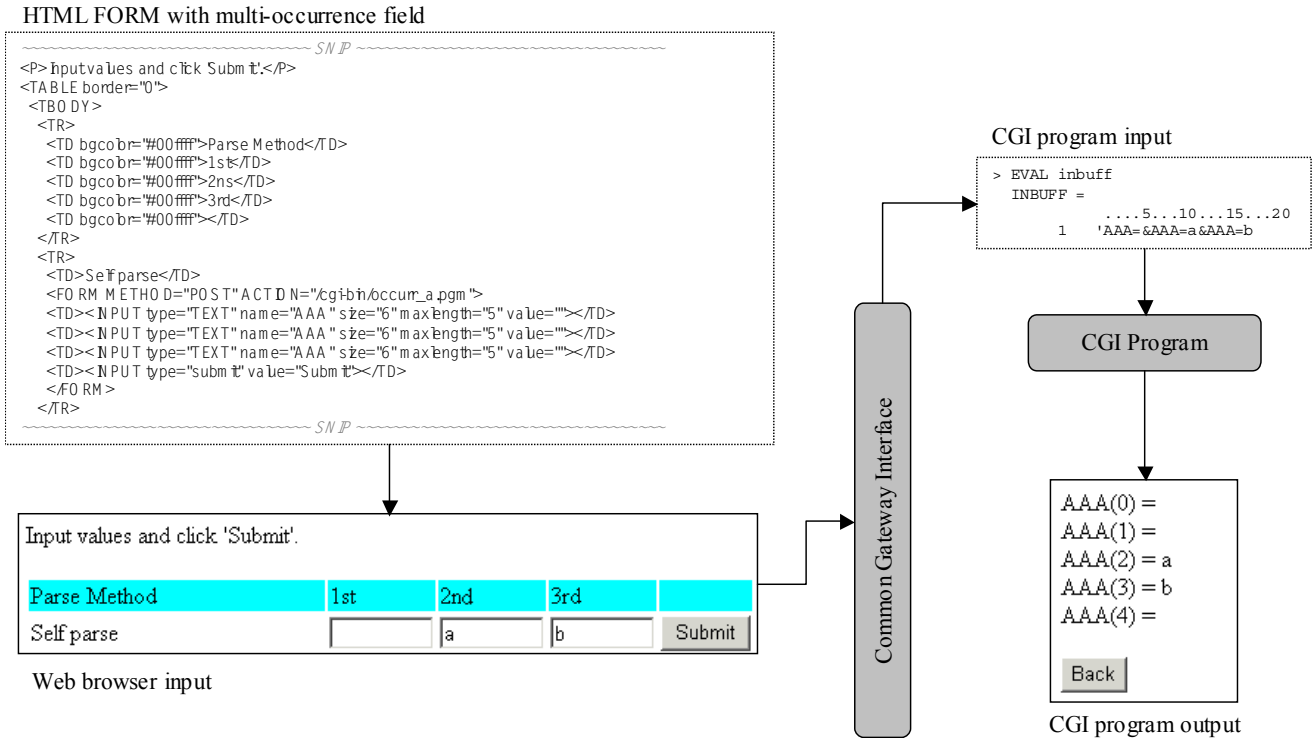
HTML FORM with multi-occurrence field

```
~~~~~~~~~~~~~~~~~~~~~~~~~~ SNIP ~~~~~~~~~~~~~~~~~~~~~~~~~~
<P>Input values and click Submit.</P>
<TABLE border="0">
 <TBODY>
  <TR>
   <TD bgcolor="#00ffff">Parse Method</TD>
   <TD bgcolor="#00ffff">1st</TD>
   <TD bgcolor="#00ffff">2ns</TD>
   <TD bgcolor="#00ffff">3rd</TD>
   <TD bgcolor="#00ffff"></TD>
  </TR>
  <TR>
   <TD>Self parse</TD>
   <FORM METHOD="POST" ACTION="/cgi-bin/occum_a.pgm">
   <TD><INPUT type="TEXT" name="AAA" size="6" maxlength="5" value=""></TD>
   <TD><INPUT type="TEXT" name="AAA" size="6" maxlength="5" value=""></TD>
   <TD><INPUT type="TEXT" name="AAA" size="6" maxlength="5" value=""></TD>
   <TD><INPUT type="submit" value="Submit"></TD>
   </FORM>
  </TR>
~~~~~~~~~~~~~~~~~~~~~~~~~~ SNIP ~~~~~~~~~~~~~~~~~~~~~~~~~~
```

CGI program input

```
> EVAL inbuff
  INBUFF =
          ....5...10...15...20
      1   'AAA=&AAA=a&AAA=b
```

CGI Program

Input values and click 'Submit'.

| Parse Method | 1st | 2nd | 3rd | |
| Self parse | | a | b | Submit |

Web browser input

Common Gateway Interface

```
AAA(0) =
AAA(1) =
AAA(2) = a
AAA(3) = b
AAA(4) =

Back
```

CGI program output

Figure 24 – Multi-Occurrence Field and CGI

To get a value of specific variable, CGI program usually passes occurrence number to parse routine.

An Introduction to Sample Programs

IBM provides two APIs, QtmhCvtDb and QzhbCgiParse, to parse the string. You can, of course, write a CGI program that parses the string by your own logic. Or you can adopt some kind of Web-programming framework. In the following sections, let's take a look at four ILE-RPG CGI programs that have different parsing routines.

Section	HTML	CGI Program	CGI Modules	Parsing API	CGI-mode	DBCS Support
2.3.2	complex_selfparse.html	COMPLEX_A	COMPLEX, UNQRYSTR	(Self parse from stdin)	%%MIXED%%	Need to modify parameters of QDCXLATE in procedure DECODE
2.3.3	complex_QzhbCgiParse.html	COMPLEX_B	COMPLEX, CGII0100	QzhbCgiParse (with format CGII0100)	%%EBCDIC%%	Yes
	complex_QzhbCgiParse2.html	COMPLEX_C	COMPLEX, CGII0200	QzhbCgiParse (with format CGII0200)		
2.3.4	complex_CGIDEV2.html	CGIDEV2	CGIDEV2	Wrapper APIs for IBM –supplied Qtmh* APIs		Need to modify subprocedures

Table 22 – A Complex CGI Program Examples