4th Dimension 2004 Allows New Freedoms in Form URLs

By David Adams

Technical Note 04-42

Overview

4th Dimension 2004 includes a significant change in how incoming form requests are handled in certain circumstances. For the first time, 4th Dimension developers have complete freedom in how URLs submitted to 4th Dimension are named. In the past, forms submitted to the 4th Dimension Web server with the HTTP `POST` method had to start with `4DACTION`, `4DCGI` or `4DMETHOD`, as in the examples below:

http://www.example.com/4DACTION/Web_HandleSearch/
http://www.example.com/4DCGI/ProcessApplication/

You may want to take control of the exact contents of the URL for policy, security, aesthetic, commercial, or usability reasons. Using 4th Dimension 2004, form requests can now consist of any valid URL, including the examples below:

http://www.example.com/cgi-bin/process_form.php
http://www.example.com/forms/search/
http://www.example.com/MyStore/WebObjects/MyStore.woa
http://www.example.com/forms/search.aspx

This technical note details the change in the 4th Dimension Web server.

URLs and GET WEB FORM VARIABLES

The *4th Dimension 2004 Upgrade* manual describes the new behavior in these words:

> **GET WEB FORM VARIABLES**
> The capacities of this command have been extended in 4th Dimension 2004: it is now possible to use it regardless of the type of URL sent to the Web server. In particular, it now operates with HTML forms sending POST data to all URLs.
>
> In previous versions of 4th Dimension, only requests beginning with /4DACTION, /4DMETHOD and /4DCGI or containing a request string could be parsed by this command.

To appreciate the change, it is necessary to understand the previous behavior. Form data submitted to the 4th Dimension Web server using the `POST` method is available in up to three locations:

- In $2 of the **On Web Authentication** database method.
- In $2 of the **On Web Connection** database method, if invoked.
- In arrays populated by a call to **GET WEB FORM VARIABLES**.

In earlier versions of 4th Dimension, the `POST` data was stripped out of all of these locations unless the form URL started with one of the special 4th Dimension URL keywords. There was no workaround to this limitation of the native Web server.

How Custom URLs are Handled

Each of 4th Dimension's special URL keywords is handled in a slightly different manner by the native Web server, as summarized below:

Keyword	**Behavior**
`4DACTION`	Run the *Compiler_Web* method, if defined. Run the **On Web Authentication** database method. Run the method named after `4DACTION`.
`4DCGI`	Run the *Compiler_Web* method, if defined. Run the **On Web Authentication** database method. Run the **On Web Connection** database method.
`4DMETHOD`	Run the **On Web Authentication** database method. Run the **On Web Connection** database method. Run the method named after `4DMETHOD` in contextual mode.

What happens in the case of a custom URL, such as the example shown below?

http://www.example.com/forms/search/

If the URL is the path to a document hosted by the 4th Dimension Web server, the document is served regularly. Otherwise, 4th Dimension treats the request as an unknown URL. Like all such requests, 4th Dimension follows the same steps as for `4DCGI` requests:

`Unknown URL`	Run the *Compiler_Web* method, if defined. Run the **On Web Authentication** database method. Run the **On Web Connection** database method.

This behavior is the same in 4th Dimension 2003 and 2004. The difference in 4th Dimension 2004 is that the `POST` data is now available. The easiest and most reliable way to extract the form values is with code as shown below:

```
ARRAY TEXT(web_form_field_names_at;0)
ARRAY TEXT(web_form_field_values_at;0)
GET WEB FORM VARIABLES(web_form_field_names_at;web_form_field_values_at)
```

The **GET WEB FORM VARIABLES** command may be called anywhere within the process handling the Web request. It is convenient to call it once in **the On Web Authentication** database method and store the resulting values in a pair of process arrays for use throughout the method.