Writing z/OS CGIs in COBOL

Course Summary

Objectives

By the end of this course, students will be able to:

- Code, compile, bind, debug, deploy, and maintain CGIs for the z/OS environment, written in COBOL
- Handle GET and POST requests: analyze and take action, as appropriate
- Parse and decode a QUERY_STRING value for GET
- Gather in the stdin data for POST
- Save a file as is or translated to EBCDIC on the mainframe, for POST
- Produce responses that are dynamically created HTML pages or redirection to existing pages
- Access environment variables
- Access DB2 data (optional: depends if DB2 installed and lab set up done)
- Access VSAM KSDS data by primary key or alternate index
- Put out HTML encoded in UTF-16, to provide a truly international aspect to your website
- Submit jobs to the batch from a CGI (optional; may not be appropriate in all environments).

Topics

- General Program Structure and Techniques
- Basic Processing
- Handling GET Requests
- The Data Connection - Part I: The Story
- The Data Connection - Part II: Working With VSAM Data
- The Data Connection - Part III: Working With DB2 Data
- Hidden Controls and cookies
- POST Requests
- Handling Files Sent by POST
- Working With Unicode Data
- Submitting jobs from a CGI

Prerequisite

Experience in coding / maintaining COBOL programs; current knowledge of COBOL, such as might be obtained from attending course D704: "Enterprse COBOL Update I - Essentials". Familiarity with z/OS UNIX such as might be obtained from attending U510: " Introduction to z/OS UNIX" and U520: "Developing Applications for z/OS UNIX". Familiarity with developing a website using the IBM z/OS HTTP server, such as might be obtained with attending U518: "You and z/OS and The World Wide Web"

Duration

Two Days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.

Writing z/OS CGIs in COBOL

Course Outline

I. General Program Structure and Techniques

A. General program structure
B. Redirect using Display
C. Redirect using printf
D. Redirect using bpx1wrt
E. Watching for errors
F. Deploying your CGI

- Computer Exercise: Setting up for labs:

II. Basic Processing

A. Emitting Headers
B. Emitting HTML
C. Accessing environment variables
D. Displaying environment variables
E. Stylesheets and CGIs

- Computer Exercise: Writing out HTML pages

III. Handling GET Requests

A. Some scenarios
B. Parsing QUERY_STRING content
C. Decoding QUERY_STRING content

- Computer Exercise: Handling incoming data

IV. The Data Connection - Part I: The Story

A. Working With Data on the Server

V. The Data Connection - Part II: Working With VSAM Data

A. Working with VSAM files

- Computer Exercise: Working with VSAM data

VI. The Data Connection - Part III: Working With DB2 Data

A. Working with DB2 data

- Computer Exercise: Working wth DB2 Data (optional)

VII. Hidden Controls and cookies

A. Session continuity
B. Hidden controls
C. Cookies
D. Modifying the previous CGI [to emit data]
E. Designing the invoked CGI [to catch data]
F. Coding the invoked CGI [to catch data]

- Computer Exercise: The Persistence of Memory

VIII.POST Requests

A. Finding needed storage size
B. Allocating storage
C. Reading from stdin
D. Breaking Apart Headers and Data
E. Our Sample POST CGI Logic
F. The TCBPSTD CGI code

- Computer Exercise: Handling POST Processing

IX. Handling Files Sent by POST

A. File Handling

- Computer Exercise: Saving and Linking to Files

X. Working With Unicode Data

A. The Role of Unicode
B. CGIs and Unicode

- Computer Exercise: Working With Unidcode

XI. Submitting jobs from a CGI

A. Set up
B. Logic
C. Computer Exercise: Submitting a job (optional)
D. Wrap up