

Accessing CA-IDMS from Internet

VG-IDMS/Web Server Overview

VEGASOFT



Contents

- ◆ From Browser to Web Server
- ◆ Server Application Functionality
- ◆ Building HTML Pages
- ◆ Structure of CA-ADS Dialog
- ◆ Included CA-ADS Functions
- ◆ Authentication

Contents

- ◆ Secured Connection
- ◆ Included Tools
- ◆ SQL Support
- ◆ XML Support
- ◆ Map Converter

NEW!

From Browser to Web Server

◆ URL

- `http://domain/cgi-bin/pgm`
- `http://domain/dir/page1.html`
- `http://www.vegasoft.com`

◆ HTTP request

- contains query variables

◆ HTTP response

◆ HTML

- Browser decides how to show HTML



Server Application Functionality

- ◆ Browser connects to TCP/IP Listener
- ◆ Listener starts Web Server
- ◆ Web Server starts Server Application
- ◆ Server Application reads query variables
- ◆ Server Application processes data

Server Application Functionality

- ◆ Application builds HTML page(s)
 - into scratch
 - into memory
- ◆ HTTP response header
 - built automatically by the Web Server
 - built by the application into scratch
- ◆ Server Application Ends
- ◆ Web Server sends HTTP and HTML to the Browser

Building HTML pages

- ◆ Pages may contain links to other HTML components
- ◆ Static HTML pages from dictionary
 - CA-IDD source modules (language HTML)
 - load modules (binary components)
- ◆ Dynamic HTML pages
 - application (dialog/program) builds on the fly
 - page layout in CA-IDD

Structure of CA-ADS Dialog

- ◆ New work record WWWWPARM
- ◆ Dialog can be mapless
- ◆ Dialog can have a map

```
IF (WWWWPTR EQ 0) THEN
  DISPLAY MSG TEXT 'ENTER DATA'.
ELSE
DO.
  MOVE 'ENTER DATA' TO WWWHTML-TEXT-MSG.
  CALL WWWPUT.
  RETURN.
END.
```

Included CA-ADS Functions

- ◆ HTTPHVAR
 - request header field
- ◆ HTTPDVAR
 - reads a given form variable
 - recognizes and verifies data type
 - edit and code tables are supported

Included CA-ADS Functions

◆ HTMLPAGE

- build an HTML page into scratch
- page layout in CA-IDD (similar to MAPC)
- replaceable parameters
- basic editing
- code tables are supported

Example: HTTP Request

```
GET
/WWWSEVT/DICTSD52/DWWWEX01.DIA?MSGID=DC001003&
SUB=SUBMIT HTTP/1.0..
USER-AGENT: MOZILLA/3.01WIN95; I; 16BI..
HOST: 192.83.13.14:1691..
ACCEPT: IMAGE/GIF, IMAGE/X-XBITMAP, IMAGE/PJPEG, */*..
FORWARDED: BY HTTP://PRO2.KTT.FI:8080 (NETSCAPE-
PROXY/2.52)
....
```

Example: Get Query Variable

◆ `<INPUT TYPE="TEXT" NAME="MSGID"
SIZE=8 >`

```
MOVE SPACE TO MSG-KEY-116.  
MOVE HTTPDVAR(WWWWPTR,'MSGID',1,MSG-KEY-116) TO MSG-KEY-116.  
IF WWWVIND-OK THEN      ! MSGID CORRECT  
    CALL MSGS.  
ELSE  
DO.  
    IF WWWVIND-TRUNC THEN  $ MSGID TO LONG  
        MOVE '<P> MESSAGE ID TOO LONG '  
        TO WWWHTML-TXT.  
    ELSE  
...  
END.
```

Example: HTML Field Tag

- ◆ ?html_field([Code Table][,BS])
 - ? (WWWFLD-MARK)
 - Code table
 - BS removes trailing spaces
- ◆ End of field tag
 - space, end of line
 - < , > , ” or &

Example: Build an HTML Page

- ◆ CA-IDD Module: hwwwex02
 - Msgid: <INPUT TYPE=TEXT NAME=MSGID VALUE="?MSGID">
- ◆ Msg-key-116='dc003001'
- ◆ Move HTMLPAGE (wwwptr, hwwwex02, 1,'MSGID',msg-key-116) to wwwhttp-status.
- ◆ Output in Response Scratch
 - Msgid:<INPUT TYPE=TEXT NAME=MSGID VALUE="dc003001">

Example: Build an HTML Page

```
!-----  
! WRITE HTML HEADER FROM  
! CA-IDD MODULE HWWWEX02  
MOVE HTMLPAGE(WWWWPTR,'HWWWEX02',1,  
'MSGID',MSG-KEY-116,  
'DATECR',DATE-CREATED-116,  
'DATELU',DATE-LU-116,  
'ERRMSG',WWWHTML-TXT)  
TO WWWWHHTTP-STATUS.
```

Example: Build an HTML Page

```
<HTML><HEAD> <TITLE> WEB EXAMPLE02 </TITLE>
</HEAD><BODY>
<BODY><FORM
ACTION="HTTP://192.83.13.14:1691/WWWSERV/DICTSD52/
DWWWEX02.DIA"
METHOD=GET>
<P><B>MESSAGE ID:</B>
<INPUT TYPE="TEXT" NAME="MSGID" SIZE=8
VALUE=?MSGID > CREATED: ?DATECR LAST UPDATED:
?DATELU
<P> <INPUT TYPE="SUBMIT" NAME="SUB" VALUE="SUBMIT">
<INPUT TYPE="RESET" VALUE="RESET">
<P>?ERRMSG
```

Authentication

- ◆ Basic Authentication Scheme
- ◆ Browser prompts for the user id and password
 - reply to authenticate request
 - only once during the session
- ◆ If load module name position 8 is 'A'
 - automatic authentication request

Authentication

- ◆ If load module name position 8 is 'S'
 - automatic authentication request
 - signon (in CA-IDMS R12/14)
- ◆ Applications can call authentication routines



Secured Connection

- ◆ SSL indirectly supported with firewalls (or proxies)
- ◆ Secured connection between browser and firewall
- ◆ Unsecured connection between firewall and server

Included Tools

- ◆ Web IDD, OCF
- ◆ Web Master Terminal (in CA-IDMS R12/14)
 - DCMT and DCUF commands
- ◆ Interactive SQL
- ◆ CA-IDMS Log browser
- ◆ Terminal Emulation
 - to test Web applications on 3270 terminals
 - debugger can be used
- ◆ Save binary files as CA-IDD load modules

SQL Support

- ◆ Browser can send SQL statements
- ◆ Supported databases
 - CA-IDMS SQL Option (w/o CA- IDMS Server)
 - DB2
 - ODBC
- ◆ Max. 99 SQL statements
 - total length less than 16KB
- ◆ Result sets as HTML tables

SQL Support

- ◆ Query variables
 - MAXSTMT
 - # of SQL statements
 - if missing, only SQLSTMT is used
 - MODULE01 - MODULE99 (optional)
 - contains HTML preceding or replacing SQL statement output
 - SERVER
 - Server Control Table entry (logical server name)

SQL Support

- ◆ Query variables (continues)
 - SQLSTMT
 - contains SQL statement
 - SQLSTMT01 - SSTMT99 (optional)
 - each contains an SQL statement

JavaScript example

```
<HTML>
<HEAD>
<TITLE>SQL QUERY FROM EMP TABLE</TITLE>
<SCRIPT>
function submitPage(form) {
    if (form.DEPTNO.value.length == 0) {
        alert ("Deptno missing!");
    }
    else {
        form.SQLSTMT.value =
        "SELECT * FROM DSN8410.EMP WHERE WORKDEPT = " +
        form.DEPTNO.value;
    }
}
</SCRIPT>
```

XML Support

- ◆ XML output supported
- ◆ Entity types xml, css, dtd and xsl
 - As IDD modules with LANGUAGE HTML
- ◆ How to send XML from a dialog
 - HTMLPAGE as usual
 - WWWHTTP-CTYP = 'TEXT/XML'

XML Support

◆ HTML module

- `<?xml version="1.0"?>`
 - HTMLPAGE without replacement parms
 - XML syntax
- `<??xml version="1.0"?>`
 - HTMLPAGE with replacement parms
 - can be viewed as static XML module

XML Support

- XML example

```
<?xml version="1.0" encoding="UTF-8"?>
<department>
  <employee id="J.D">
    <name>John Doe</name>
    <email>jdoe@ibm.com</email>
  </employee>
  <employee id="B.S">
    <name>Bob Smith</name>
    <email>Bob.Smith(@)foo.com</email>
  </employee>
</department>
```

XML/SQL Server

- ◆ DB2 and CA-IDMS SQL Option
- ◆ Query variables
 - SERVER
 - Server Control Table entry
 - SQLSTMT
 - SQL statement
 - STYLESHEET (optional)
- ◆ Output consists of
 - Inline DTD
 - Rowset & SQLCA

Map Converter

- ◆ Batch Utility with two phases
- ◆ Second phase also as PC version
- ◆ Input
 - reads dictionary definitions of given maps
 - reads dictionary definitions of related edit/code tables
- ◆ Output
 - maps as HTML forms in CA IDD modules
 - form input and validation as CA ADS routines
 - form output as CA ADS routines

Map Converter

◆ Features

- recognizes map type (pageable/non-pageable)
- recognizes field attributes (bright, protected, dark, required)
- supports OCCURS definitions
- recognizes field error text
- recognizes system variables (\$MESSAGE, \$PAGE, \$RESPONSE)
- generates list boxes from edit/code tables
- generates JavaScript from edit ranges
- either free form or "3270 look alike" HTML forms

Map Converter

- ◆ Example: Original Map

```
File Edit Transfer Fonts Options Macro View Window Help
VEGASOFT OY          VG-ADS/DB2          V1.2
LIST OF EMPLOYEES USING DYNAMIC SQL
-----
DEPARTMENT  A00          SPIFFY COMPUTER SERVICE DIV.

EMPNO      FIRST NAME  LAST NAME
000010     CHRISTINE   HAAS
000110     VINCENZO   LUCCHESI
000120     SEAN        O'CONNELL
200010     DIAN        HEMMINGER
200120     GREG        ORLANDO

SQLCODE      0

ENTER=EXECUTE

1 Sess-1          § 5/14
```

Map Converter

◆ Example: Generated HTML form

Screenshot of Microsoft Internet Explorer showing a web form titled "VEGASOFT OY VG-ADS/DB2 V1.2 LIST OF EMPLOYEES USING DYNAMIC SQL". The form includes a department input field with "A00", a table of employee data, and "SEND" and "RESET" buttons.

VEGASOFT OY VG-ADS/DB2 V1.2
LIST OF EMPLOYEES USING DYNAMIC SQL

DEPARTMENT SPIFFY COMPUTER SERVICE DIV.

EMPNO	FIRST NAME	LAST NAME
000010	CHRISTINE	HAAS
000110	VINCENZO	LUCCHESI
000120	SEAN	O'CONNELL
200010	DIAN	HEMMINGER
200120	GREG	ORLANDO

SQLCODE 0

ENTER=EXECUTE

Map Converter

- ◆ Generated CA-ADS routines
 - WWWGET
 - reads all unprotected fields with the HTTPDVAR built in function
 - WWWCHECK
 - checks the outcome of the HTTPDVAR function
 - this is an INCLUDE module
 - WWWPUT
 - sends all fields and literals of a non pageable map

Map Converter

- ◆ Generated CA-ADS routines
 - WWWPUTP
 - sends all literals and protected fields
 - used when sending input validation error messages
 - WWWPUTH (pageable map)
 - sends fields and literals preceding the detail area
 - WWWPUTD (pageable map)
 - sends fields and literals of the detail area
 - WWWPUTF (pageable map)
 - sends fields and literals following the detail area

Map Converter

- ◆ Generated CA-ADS routines (example)

```
DEFINE WWWGET.
```

```
!THIS ROUTINE READS ALL UNPROTECTED FIELDS
```

```
MOVE 'N' TO WWWWINP-EIND.
```

```
MOVE HTTPDVAR(WWWWPTR,'WEXD-WDB2DEPTNO',0001,  
WEXD-WDB2DEPTNO) TO WEXD-WDB2DEPTNO.
```

```
MOVE 'WEXD-WDB2DEPTNO' TO WWWHTML-TXT-WRK.
```

```
CALL WWWCHECK.
```

```
GOBACK.
```

```
INCLUDE WWWCHECK.
```

} Three
statements
for each
field.

Map Converter

- ◆ Generated CA-ADS routines (example)

```
DEFINE WWWPUTD.
```

```
!THIS ROUTINE SENDS DETAIL FIELDS
```

```
MOVE HTMLPAGE
```

```
(WWWPTR, 'SDB2EXD-C',0001,
```

```
'WEXD-WDB2EMPNO',WEXD-WDB2EMPNO,
```

```
'WEXD-WDB2FIRSTNME',WEXD-WDB2FIRSTNME,
```

```
'WEXD-WDB2LASTNAME',WEXD-WDB2LASTNAME)
```

```
TO WWWHTTP-STATUS.
```

```
IF NOT WWWHTTP-STATUS-SUCCESS THEN
```

```
DO.
```

```
RETURN.
```

```
END.
```

```
GOBACK.
```

This is the generated HTML module for the detail area.



End of Presentation

See
www.vegasoft.com
for
international
distributors.