Integrating EWD with VISTA

By: Sam Habiel

Sam.habiel@gmail.com

A biased history of User Interfaces to VISTA

- Traditional Roll and Scroll (CHUI) to execute routines.

- Any good Roll and Scroll will need a menu system.

Menus

And then...

Why the menu system?

- The menu system provides the foundation of VISTA security

- The menu system is very extensible.

- Power users can develop new applications using Fileman RAD (from... the menu system)

- A new application can be easily developed and dropped into the menu system.

Why again?

- The crux of the Problem
    - Power Users versus Programmers versus Sys Admins
    - Accountability
- EWD has the potential to bring back DHCP

What's needed?

- We need the following in order for this to happen
  - Has to look pretty.
  - Foundational Elements of VISTA
    - Sign-on
    - Division Selection/Change
    - Menu System

What's needed?

- Fileman Building Blocks
  - DIR
  - DIC
  - DIQ
  - DIE
- Automatic conversion of existing applications
  - Screenman
  - List Manage
- A good IDE for end users to build their own applications.
  - Text editor won't cut it

The devil is in the details

- Developing infrastructure tools for VISTA is hard
    - They need extensive testing
    - They need to be used a lot by end users to be called tested
  - A good example of a failure in development
    - Support for mixed case verify codes

Specific Issues

- Session Maintenance

    - Inject code like this in every page

        - r(sessid) ; Restore Symbol Table
        - d mergeArrayFromSession^%zewdAPI(.DUZ,"DUZ",sessid)
        - d mergeArrayFromSession^%zewdAPI(.IO,"IO",sessid)
        - s U=$$getSessionValue^%zewdAPI("U",sessid)
        - quit
        - ;
        - s(sessid) ; Save to Symbol Table
        - d mergeArrayToSession^%zewdAPI(.DUZ,"DUZ",sessid)
        - d mergeArrayToSession^%zewdAPI(.IO,"IO",sessid)
        - d setSessionValue^%zewdAPI("U","^",sessid)
        - quit

Session Maintenace

- Can wrap the above in a custom tag

- Or can use ewdTemplate.ewd

- Need to design mechanism on incrementing decrementing variables saved in the session.

- Very commonly VISTA keeps incrementing variables to use

    - Can do S X="%" F S X=$O(@X) Q:X="" to save all the variables in ST to session.

  - But is the session the new ST? The session does not scope variables.

Session Maintenance

- In order to prevent variables from accumulating

    - Use tmp.sessionname – but if the form doesn't submit, you lose your session values

    - Namespace the variables and delete them en masse

Configuring IO for EWD

- NULL ; Open Null Device

- s %ZIS="0H",IOP="NULL" d ^%ZIS

- i POP s $ec=",U1," ; this crashes everything... intended...

- q

-

Tracking EWD Errors

- Right now, EWD Errors (even VISTA related) are in ^%zewdError

- It would be better to use the far more robust VISTA Error Handler (robust, but it still can't handle transactions—FYI: Transactions were introduced in the '95 standard)

Menu System Support for EWD Applications

- The idea is that each separate "menu item" is going to be a separate EWD application.

- To switch between applications use:

  - d setRedirect^%zewdAPI

  - BUT: ewdLogout.ewd doesn't return you to the starting application.

RPC Wrapper

- A lot of existing and new functionality work over the RPC system

EWD Custom Tags to work with Fileman

- <fm:edit dd="200" fields="200.05;200.09;31.2;31.3" iens="#DUZ" />
  - EDIT(nodeOID,attrValues,docOID,technology) ; Fileman Field Edit tag
  - ; Get Attributes supplied to us: dd, fields, iens
  - ;
  - N attrs
  - D getAttributeValues^%zewdCustomTags(nodeOID,.attrs)
  - N DD S DD=$g(attrs("dd")) ; Data Dictionary
  - N FIELDS S FIELDS=$g(attrs("fields")) ; Fields to Edit
  - N IENS S IENS=$g(attrs("iens")) ; IENs to Edit
  - ;
  - ; Does our Form Tag Exist?
  - N fOID S fOID=$$getElementById^%zewdDOM("fmEdit",docOID)

Various other issues

- EWD SAC compliance
  - Is it important? EWD meets the spirit of SAC compliance, but not the letter
  - EWD Routines on Cache are closed source – may as well be Intersystems routines—Do we dictate what Intersystems can or cannot do?
  - EWD Source are our own routines + ewd/css/js
- Transmission of EWD applications via KIDS
  - Should be easy
  - EWD applications are mostly files on the fs with our own routines

Farewell to Monolithic Applications