FoxInCloud

What are adaptations for?

Filling the gap between a VFP app running on Desktop –vs– FoxInCloud Web Application Server.

AT20 Development Workshop – Day 1, Activity 2
# Agenda

<table>
<thead>
<tr>
<th>Section</th>
<th>Subject</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A different process flow</td>
<td>15'</td>
</tr>
<tr>
<td>2</td>
<td>Access to user’s peripherals</td>
<td>3'</td>
</tr>
<tr>
<td>3</td>
<td>State factors</td>
<td>3'</td>
</tr>
<tr>
<td>4</td>
<td>Where user event are processed</td>
<td>3'</td>
</tr>
<tr>
<td>5</td>
<td>dodefault()</td>
<td>3'</td>
</tr>
</tbody>
</table>
Desktop: one execution thread / stack per user

1- User event code runs in the same call stack as the initial application startup.

2- When user closes a form, it releases completely from memory and these events fire: form.unload(), form and members .destroy()

3- Variables assigned at app startup without LOCAL are ‘seen’ by all called procedures and methods
Web: Request – Response

For each request, Server **must deliver a response**

Server can’t stay waiting for requests from a single user.

Web: User events run in server context

Each user request runs “Stateless”, IOW, out of the context of startup and previous user action.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Adaptations

- Private variables must either:
  - Become global: `PUBLIC, _Screen.Property` or `_VFP.Property`,
  - Be passed to form as parameters (up to 20 supported).

- Minimum dependencies between forms:
  - Private datasession preferred
  - Restore Settings
  - Avoid mutual references

- Callback (…)

- App Startup & Exit (…)

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Adaptations (...)

Callback

In Web mode, calling form and modal child form run in different threads; Server can’t wait (suspend) until user replies.

In Web mode, calling form and modal child form run in different threads; Server can’t wait (suspend) until user replies.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Adaptations (...)

App Startup & Exit

Because of differences in process flow, app startup program should run differently in desktop and web mode:

<table>
<thead>
<tr>
<th></th>
<th>Desktop</th>
<th>Web</th>
</tr>
</thead>
<tbody>
<tr>
<td>* set up app environment</td>
<td>OK</td>
<td></td>
</tr>
<tr>
<td>do form login</td>
<td>KO</td>
<td>displaying a form out of a user context makes no sense</td>
</tr>
<tr>
<td>read events</td>
<td></td>
<td>can’t enter a modal state</td>
</tr>
<tr>
<td>* clean up app environment</td>
<td>KO – no prior modal wait state: this code won’t execute</td>
<td></td>
</tr>
</tbody>
</table>

Recommended: move app environment set up code into a standard env. class (xxxSets as awSets of awPublic.prg) that cleans up environment automatically.

<table>
<thead>
<tr>
<th></th>
<th>Desktop: main.prg</th>
<th>Web: xxxServer.prg</th>
</tr>
</thead>
<tbody>
<tr>
<td>o = NewObject('xxxSets', 'xxxSets.prg')</td>
<td>cAppSetsLib = 'xxxSets.prg'</td>
<td></td>
</tr>
<tr>
<td>do form login</td>
<td>cAppSets = 'xxxSets'</td>
<td></td>
</tr>
<tr>
<td>read events</td>
<td>lAppSetsClass = .T.</td>
<td></td>
</tr>
</tbody>
</table>
Stateless and Scalable: N Users <-> N Servers

Any user can hit any server at any time.

Each server (FAS) uses a single instance of each form for any user.

FAS saves/restore the state of form for each user.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Form instances stay alive until server stops

Because each server keeps an instance of each requested form, some events fire slightly differently.

<table>
<thead>
<tr>
<th>User</th>
<th>Desktop</th>
<th>Web</th>
<th>Adaptation</th>
</tr>
</thead>
<tbody>
<tr>
<td>opens a form</td>
<td><code>.Load()</code></td>
<td><code>.Load()</code>: once at first user request</td>
<td>Move user-dependent code from <code>.Load()</code> to <code>.Init()</code></td>
</tr>
<tr>
<td></td>
<td><code>.Init()</code></td>
<td><code>.Init()</code>:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Once at first request with</td>
<td>Move code within <code>.Init()</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>.wlInitFirst</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Once per user</td>
<td></td>
</tr>
<tr>
<td>exits a form</td>
<td><code>.Release()</code></td>
<td><code>.Release() only</code></td>
<td>Move user-dependent code to <code>.Release()</code></td>
</tr>
<tr>
<td></td>
<td><code>.Destroy()</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>.Unload()</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Commands acting on user’s peripherals

These commands require access to the user’s peripherals, which web server can’t access.

do form / form.show() / report form
MessageBox() / InputBox() / WAIT
LocFile / PutFile / GetFile / GetDir / GetColor()
Menu commands

To be replaced by procedures and methods supporting either desktop or web:

<table>
<thead>
<tr>
<th>Original</th>
<th>Adapted</th>
</tr>
</thead>
<tbody>
<tr>
<td>do form/form.show()</td>
<td>.wForm*()</td>
</tr>
<tr>
<td>report form</td>
<td>PDF generation</td>
</tr>
<tr>
<td>MessageBox()</td>
<td>.wMessageBox()</td>
</tr>
<tr>
<td>WAIT</td>
<td>wWait()</td>
</tr>
<tr>
<td>PutFile()</td>
<td>.wFileSaveAs()</td>
</tr>
<tr>
<td>Menu commands</td>
<td>wMenu()</td>
</tr>
</tbody>
</table>

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Adaptation:
Tell FoxInCloud the state factors

```
.wcPropSave
```
- Each object inherits a `wcPropSave` property holding a list of properties that user action can change (mostly automated)

```
.wContentDynamic
```
- Tells FoxInCloud that the members of a form/container/page can change at runtime

```
.wViewParmSet()
```
- Tells FoxInCloud the name and value of parameters when querying the views

To avoid saving properties that never change, FoxInCloud needs to know elements that user can change at runtime.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
Events: process on server, browser, both, or ignore

- Any user event method must begin with this code:
  ```
  if thisForm.wlHTMLgen
      return <some value>
  endif
  ```
  where `<some value>` tells FoxInCloud how to process the Event:
  - `.T.`: use existing VFP code on server;
  - `'string'`: JavaScript to be executed in browser; eg. `MouseMove();`
  - `thisForm.wcScriptEventClientServer()`: first execute on browser, then on server;
  - `.F.`: ignore event in web mode.

In most cases, events are processed on server using the existing event code.

You can also implement this process on the client browser using JavaScript.

Make sure to let FoxInCloud code run by calling dodefault() where appropriate (soon automated by FAA)

Add dodefault() to your code

- .Init(), .Destroy()
- .Load(), .UnLoad()
- .Release()
- .AddObject(), .NewObject(), .RemoveObject()
- .Requery()
- .SetFocus()
- ._assign()
- .AutoFit()

Look at the code inherited from aw.vcx!aw* to know where to add dodefault() in your code.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/
That’s about it

This presentation has covered the main differences between desktop and web modes, requiring developer’s understanding and attention.

**FAA** provides an educated, in-depth list of all adaptations you need to care about.

Once your application is adapted and you’ve practiced the adaptation process for a while, you’ll naturally develop ‘the FoxInCloud way’.

One last word about FAA FoxInCloud Adaptation Assistant

- Free: http://foxincloud.com/download.php
- Copy (test) mode / Source mode
- Assistant, not magician!
- Adapts 99% of your code (avg)
- Spots adaptation needing your attention
- Provides guidance, documentation and code samples, together with FoxInCloud Live Tutorial: http://foxincloud.com/tutotest/
- Helps you manage your adaptation project.

Any question? Post with screenshot in the ‘FoxInCloud’ section of https://support.west-wind.com/ or http://www.universalthread.com/