

1 What's New In Nine: Grab Bag

Rick Schummer
White Light Computing, Inc.
Copyright 2005

2 Who Am I?

- President - White Light Computing, Inc.
- Co-author: *What's New in Nine*
- Co-author: *Deploying Visual FoxPro Solutions*
- Co-author: *MegaFox: 1002 Things You Wanted to Know About Extending Visual FoxPro*
- Co-author: *1001 Things You Wanted to Know About Visual FoxPro* (KiloFox)
- VFP MVP and MCP
- Author of WLC HackCX & MenuDesigner
- Shedding Some Light
- raschummer@whitelightcomputing.com

3 Agenda

- Property Editors
- MenuHits
- Example of extending the VFP 9 DataExplorer
- Miscellaneous VFP 9 productivity tips.
- Free VFP 9 Tool under development
- Whitepaper for this session is available on Hentzenwerke.com

4 Property Editors

- Available via the Property Sheet ellipses button
- Driven by MemberData (local and encapsulated with inheritance)
-- AND/OR --
Defined in "E" records in FoxCode.dbf (global)
- Typically work with one property, can work with more

5 Property Editors

- Work with intrinsic properties, but even better for custom properties
- Team deployment issues
- Global naming limitations based on FoxCode (24 character property names).
- Local MemberData can break backwards compatibility if more than 255 characters, but easy to fix

6 Property Editors

- Example 1: VFP 9 native
 - Anchor (Anchor Editor)
 - Caption
 - Name
 - _MemberData (MemberData Editor, which is also a builder)
- Example 2: Logical Properties
 - Toggles the custom property between True and False, like intrinsic properties

7 **Property Editors**

- Example 3: BuilderX
 - Creates the settings for the custom builder property in the proper order by presenting a user interface
- Example 4: Naming convention enforcer
 - Automatically names the object based on the data the control is bound to

8 **Property Editors**

- Ideas for future
 - Framework implementations so now we can enforce valid settings for our custom framework properties
 - Doug Hennig is working on Property Editor Framework
 - Anything practical to your development

9 **MenuHits**

- Replacing VFP native dialogs with your own
- Naturally you need to build equivalent or better functionality
- Need IntelliSense turned on
- Does not get called from toolbar equivalent calls

10 **MenuHits**

- How does it work:
 - Searches FoxCode for MENUHIT record (type S), otherwise native dialog
 - Executes script in Data memo field
 - Examines return value from script (False or ValueType property not V or L gives native dialog, otherwise you get your behavior)

11 **MenuHits**

- Best implemented with menu hit handler code
- MenuHit example in Solution Sample (allows either FoxCode or PRG option for different menu hits)
- Parameter object
 - MenuItem, UserTyped, ValueType properties

12 **MenuHits**

- Example 1: New Property/Method (By Doug Hennig)
 - Memberdata integration
 - Non-Modal
 - Property default value defaults
 - Validation on the fly of the name
 - Add & Close single click
 - Dockable
 - Switch between methods and properties

13 

MenuHit

- Example 2: New Dialog
 - One click access to the new file
 - One click access to the Wizards
 - Non-modal
 - Organized in most used order (faster access with keyboard)

14 

MenuHits

- Ideas for future
 - Edit Property/Method Dialog
 - Project Info
 - DataSession
 - Compile

15 

Extending the DataExplorer

- DataExplorer runs as Task Pane or standalone
- Functions like SQL Server Enterprise manager, but with VFP, SQL Server, Oracle, or any OLE DB compliant database server
- Is very extendible

16 

Extending the DataExplorer

- DataExplorer.dbf stores metadata
- Add-in Manager
- Menu Manager
- Drag/Drop Manager
- Ability to restore everything to default if you disable the functionality or want more practice hacking away

17 

Extending the DataExplorer

- The DataExplorer allows you to design tables and local views
- Normally it brings up the VFP Table and View Designer
- We can customize the code to bring up our favorite View Designer replacement

18 

Miscellaneous Productivity Tips

- Build Messages
- Default values for New Property
- Select printed text
- BROWSE memo tips
- SET PATH ... ADDITIVE
- C++ operators
- View Definition

19 

Class Compare Utility

- **Class Compare Utility**

- Compares all the classes in two "identical" class libraries to see if there are differences (framework release check)
- Compare two classes to determine what I have changed in a subclass from the superclass. This way I can determine what I have changed in my Ilibs from the VFE framework classes.
- Compare two class libraries to each other to see new classes, missing classes and classes with different base classes

20 ☐ **Questions?**

- If there is time for questions that have not been asked during session, now is a great time to ask
- Feel free to ask me questions about this topic at any time
- Contact information is on my Web site

21 ☐ **Thank You**

Please remember to fill out your evaluation.