

## 1 Best Practices for Error Handling

Session Number

## 2 Who Am I?

- President of White Light Computing
- Co-author of *What's New in Nine, Deploying Visual FoxPro Solutions*, *MegaFox: 1002 Things You Wanted to Know About Extending Visual FoxPro* and *1001 Things You Wanted to Know About Visual FoxPro (KiloFox)*
- VFP MVP and MCP
- Authored WLC HackCX / MenuDesigner

## 3 Agenda

- Error handling fundamentals and overview in context of best practices
- Mixing the styles together
- One commonly used strategy
- Collecting the evidence
- Communicating and tracking problems revealed

## 4 Errors Happen

- We all have errors in our apps
- Programmer, predictable, and unpredictable
- The difference between professionals and pretenders is the pros know they happen and know how to respond

## 5 User Response

- "I didn't do it"
  - "I never saw the error dialog"
- OR
- "Your software is killing me"

## 6 Application Response

- Address without notification
- Address with notification
- Try again
- Avoid
- Abandon

## 7 Error Handling Fundamentals

- Determining cause of the error
- Notification
- Be assertive
- Error Log
- Clean shut down
- Turn off error handling inside error handling

- 8  **VFP Error Handling**
- None
  - Default
  - Global via ON ERROR
  - Object-Level via Error Event
  - Structured via TRY...CATCH
- Your best battle-tested code in your application
- 9  **Default**
- No other error handling
  - Raw message
  - Cancel, Suspend, Ignore, Help
  - Not recommended at all
- 10  **Global Error Handling**
- Largest legacy code base
  - Centralized and error handler of last resort
  - Code is removed from the code triggering the error
  - Pass in parameters to understand the state of the error
- 11  **Global Error Handling**
- Use AERROR()
  - Does not have to be a program, it can be a class
  - Address as many errors as practical
  - Capture the error details when nothing else can be done
- 12  **Object-Level Error Handling**
- Encapsulated
  - Easiest way to address errors in an object
  - Integrated with object with the error
  - Is triggered by code in the object and procedural code called by the object
- 13  **Structured Error Handling**
- TRY...CATCH...FINALLY
  - Code in TRY runs until it errors
  - CATCH evaluated until true
  - Unhandled exceptions are bad
  - Exception objects
  - FINALLY always runs
- 14  **Error Handling Priorities (single)**
- ON ERROR until you set it again
  - Error() event runs, rest handled by default handler
  - TRY block is only place you can trap, rest handled by default

- Minimally you should always have global error handler
- 15  **Error Handling Priorities (two)**
- Global and object-level
  - Global and structured
  - Object and structured
- 16  **Error Handling Priorities (all)**
- If one exists, the immediate TRY...CATCH in the same method/procedure that the error occurs. (Also remember this applies also to external procedures the code calls within the TRY)
  - The object's Error event, if code exists, for the object
- 17  **Error Handling Priorities (all)**
- TRY...CATCH at the next level up in the call stack or in a higher-level method
  - ON ERROR routine, if one exists
  - The standard VFP system error message
- 18  **Error Handling Strategy**
- There are a few out there, but it boils down to what works for you
  - It will depend on your framework, or lack of one
  - It will depend on the type of applications you write
- 19  **Error Handling Strategy**
- If you are looking for a battle tested strategy, shop no further
  - Doug Hennig
    - *Error Handling in VFP 8*
    - *Error Handling in Visual FoxPro*
- 20  **Error Handling Strategy**
- Three layers
    - Structured for localized errors
    - Object-level for encapsulation
    - Global for catch all
- 21  **Error Handling Strategy**
- Chain of Responsibility
    - Error events handle error or pass up to superclass
    - Once at the top of the object hierarchy you pass to the parent container (another object)
  - Global error handler is class that takes care of the rest
- 22  **Logging Errors**
- Who, when, where, why, and how
  - VFP LIST command
  - AERROR function
  - Environment settings
  - Call stack

- Datasessions
- Config.FPW

23  **Communicating Errors**

- Exact steps to reproduce the error
- What was observed
- What was expected
- Additional comments from user
- Details we logged when the program crashed

24  **Communicating Errors**

- Phone call
- Screen shots
- Screencast (video recording)
- Email a report or error log files
- Web service
- RSS feed

25  **Tracking Errors**

- Web based
  - FogBugz
  - BugCentral
  - Wiki
- Client based
  - Anomaly Tracking System
- Mix
  - IssueView

26  **Resources**

- Framework vendors sent me notes on how they address error handling
- The VFP Help file has several excellent topics
- Doug's whitepapers
- FoxPro Wiki

27  **Resources**

- *Structured Error Handling in VFP 8*, by Markus Egger, CoDe FOCUS: Visual Foxpro 8 (March 2003)
- *Exception Handling Dilemma in VFP*, Rick Strahl's WebLog
- *Debugging Visual FoxPro Applications*, by Nancy Folsom, Hentzenwerke Publishing

28  **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 on my Web site - whitelightcomputing.com

29  **GLGDW Web Update Page**

[www.hentzenwerke.com](http://www.hentzenwerke.com)  
(click on Downloads)

30  **Thank you!**

Please remember to fill out your evaluation.