(alternate title 1)
The Philosophy of DataFlex in the 21st Century

John Tuohy

(alternate title 2)
We may be come from different worlds, but
we are not so unlike, you and I

John Tuohy

As we go through these final slides…

- I hope you are thinking the following:

    o My goodness, we experience these exact same things

    o Thank goodness, they've done this so we don't have to

Does this describe your application?

- It's a business application
- It's geared towards very specific businesses - sometimes a single business
- It's a very high level solution
- Your real skill is knowing the business and not just knowing technology
- It's built to last
- The definition of success, is to have a legacy
- If successful, a large part of it becomes evolutionary

The DataFlex Way

- DataFlex a business software language
  - Its genesis was based supporting business rules in the language (autofind, required, range, etc.)
  - Although it gotten a lot bigger, it still does retains that orientation (DDOs, DEOs)
- It works best in the hands of someone who knows the rules of a business and not just IT
- It tries to keep things as simple as possible
- It values and respects continuity
- It is our responsibility to take the latest industry changes and to integrate them in the product - we do it so you don't have to

As the Data Access CTO I have the following roles

- Lead the group in deciding what to do next
- Assist in product design
- Code (mostly flex)
- Act as the developer advocate
  - Ok, so how would someone do the this…
- Know when to say "be careful"
- Know when to say "No"

How to decide what changes to make

- We have a long list
- We respond to changes in the industry
- We listen to our developers
- We have are own ideas as well
- We recognize our resource limitations
- We recognize that adding a feature goes way beyond just technical implementation
- We respect continuity and legacy
- We mix all this together to produce product – it is the magical art

How do we decide where to make changes?

Work in the product area that…

- gives the best technical result
- gives the most maintainable result
- is most cost effective
- you are most familiar with
- is the most fun/cool

Personal Experience

- Over the last view few years, I've worked in a lot of the code areas I've had not worked in before

- This has been an instructive experience as it has allowed me to get a better sense of what's out there

What are our development platforms?

- Main Runtime (C)
- Extension dlls (C, C++)
- Extension COM and MFC components (C++)
  - Extensions accessed at runtime and at flex level
- Add on technologies (Codejock MFC/COM, Boost, Xerces)
  - mostly we have source
  - some are no longer supported
- Flex development code
  - Mostly for high level and UI
  - Flex package Library
- JavaScript
- We also have to deal with HTML, XML (and xpath, xslt, etc.), JSON, CSS

Observations

- C
    - You sure have to careful.
    - Pointers and memory leaks, oh my!
    - This is expensive development
    - Low level
- C++
    - Powerful but complex - especially STL
    - It does amazing things
    - So many files
    - This is expensive development
    - Low level
- Extension libraries (for C and C++)
    - Are a requirement
    - But so much code (number of files, lines of code)
    - Can't live with them and can't live without them
- DataFlex
    - Higher level
    - Great for business processing and intensive UI work
    - Is not the tool for every application need
    - When applied to the right task, amazingly cost effective

More Observations

- You need a good development studio and debugger
- Development is the same everywhere - design first, code carefully, debug extensively
- Comments in code – we are all sinners
- Documentation
  - No one is great, many are horrible
  - DataFlex is not all that bad
  - How did we survive before goggle?
- It's not that hard to change gears (languages and development platform)
- Having smart developers who work across "gears" is invaluable

Conclusions

- Find the right tool for the task
- DataFlex is the tool for high level business and UI needs
- As needed, develop at other levels. It's not that hard (e.g., COM, dlls)
- There is a real advantage knowing how to work in multiple areas

A few quotes to end the show

- We are behind schedule because of delays caused by our industry (Peter Bragg)
- That's the way the whole world goes, so do we (Carsten Sorensen)
- So you say you want a revolution (John Lennon)

And Finally…

It's the singer not the song (Mick Jagger)

　　or

While we are proud of our product we never forget that it is our developers who have the required business knowledge and programming skills that allows our song to sing – we thank you!