

„Arctica“?



The Fox and the Polar Bear!

„First look into Xbase++ for FoxPro and Visual FoxPro Developers“

Overview

- Visual FoxPro and Xbase++ common roots
- Similarities and differences
- Commands, Functions, Objects and Codeblocks
- DatabaseEngines and SQL in Xbase++
- Intelligent multi-threading
- User interface concepts
- Where do we want to go!



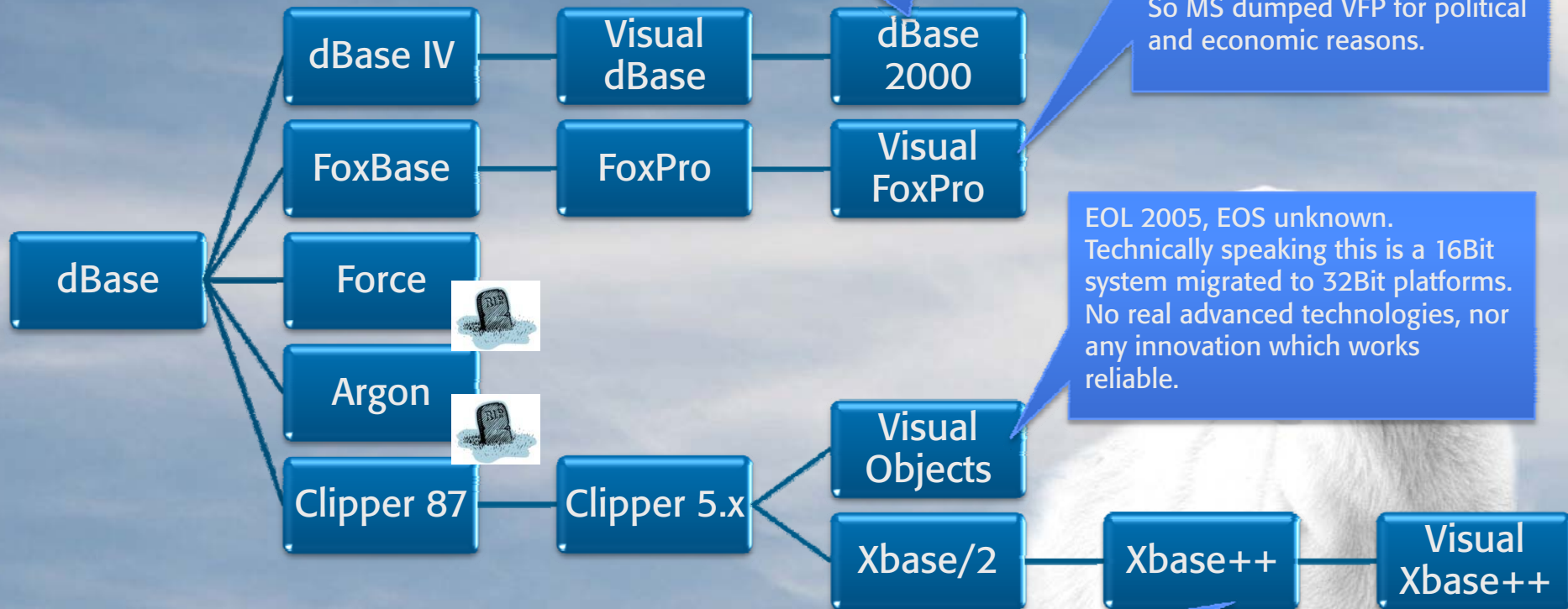
The roots!

EOL/EOS in 2000, However dead since Borland decided to dump dBase in favour of Paradox to compete with MS Access in the mass market! Also it always ate market share from Delphi so...

EOL 2007, EOS 2015. MS had too many language in their shop, too limited resources. VFP code base too old, needed a complete rewrite to conform with 32bit/64bit requirements. So MS dumped VFP for political and economic reasons.

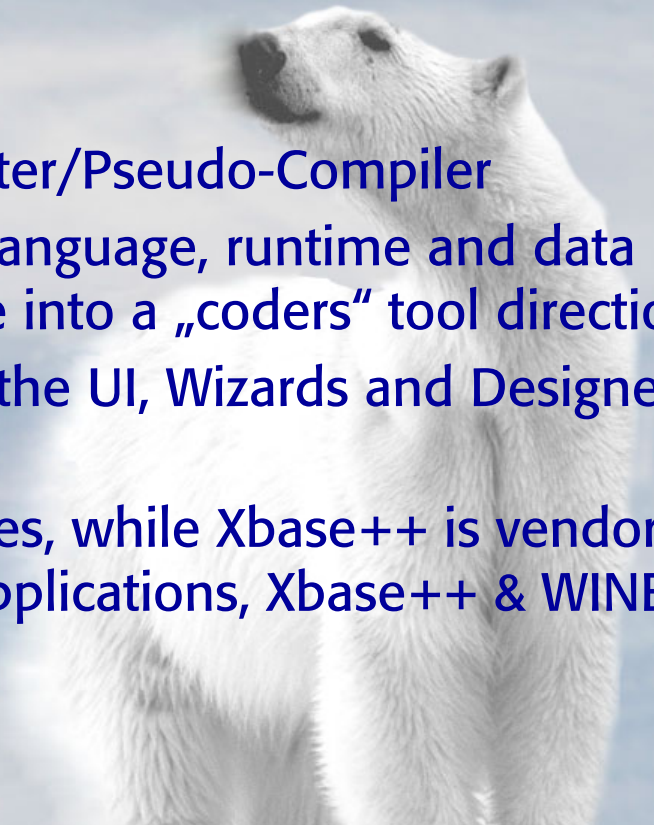
EOL 2005, EOS unknown. Technically speaking this is a 16Bit system migrated to 32Bit platforms. No real advanced technologies, nor any innovation which works reliable.

No EOL nor EOS, instead the product moves forward with a growing customer base. New technologies and product releases are under development. Service Level for Windows 7 is under development, release this year.



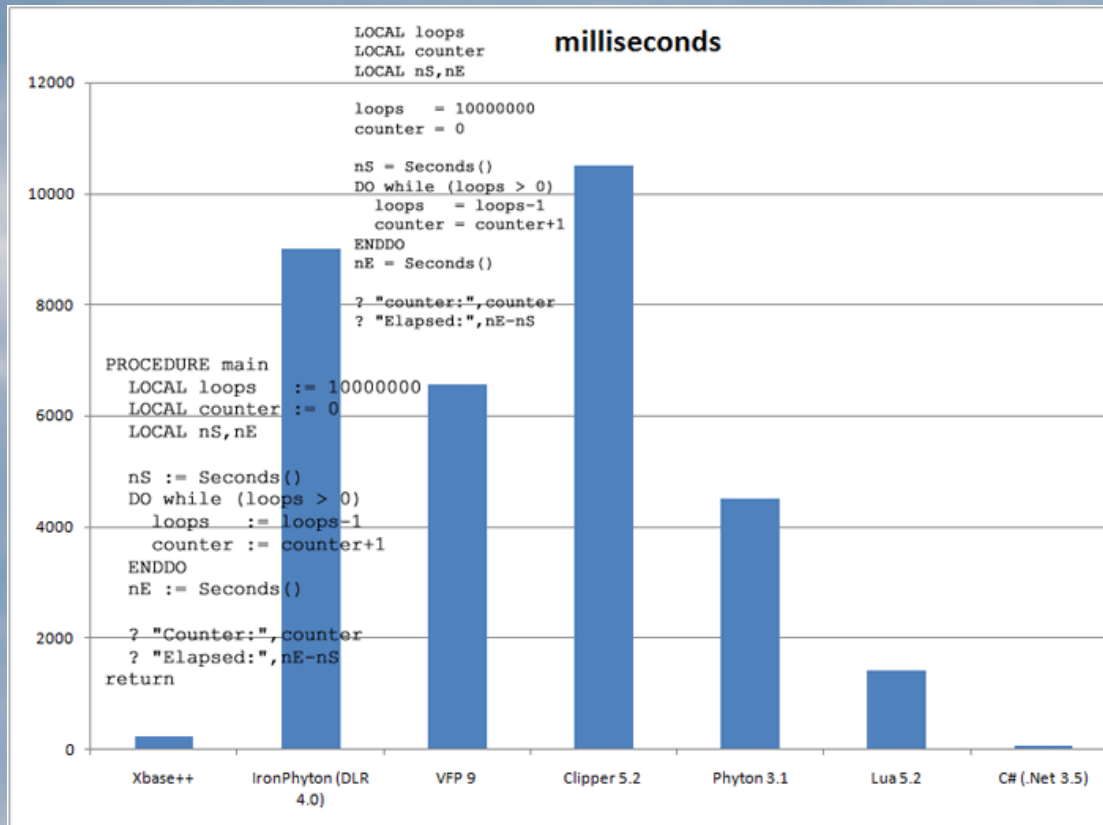
Xbase++ & Visual FoxPro

- In common:
 - are xBase language dialects
 - have developed over time into their own direction
 - are database & data access-centered
- Some differences:
 - Of course, different vendor -😊
 - Xbase++ is a compiler. VFP is a Interpreter/Pseudo-Compiler
 - Xbase++ evolution has focused on the language, runtime and data access-abstraction moving it much more into a „coders“ tool direction.
 - Visual FoxPro evolution has focused on the UI, Wizards and Designers making it a design-oriented tool.
 - Visual FoxPro focuses on MS technologies, while Xbase++ is vendor neutral. IIS & Apache support for web-applications, Xbase++ & WINE => Linux.



Commands, Functions, Objects and Codeblocks!

- Xbase++ is a hybrid compiler, generating native code and pcode.



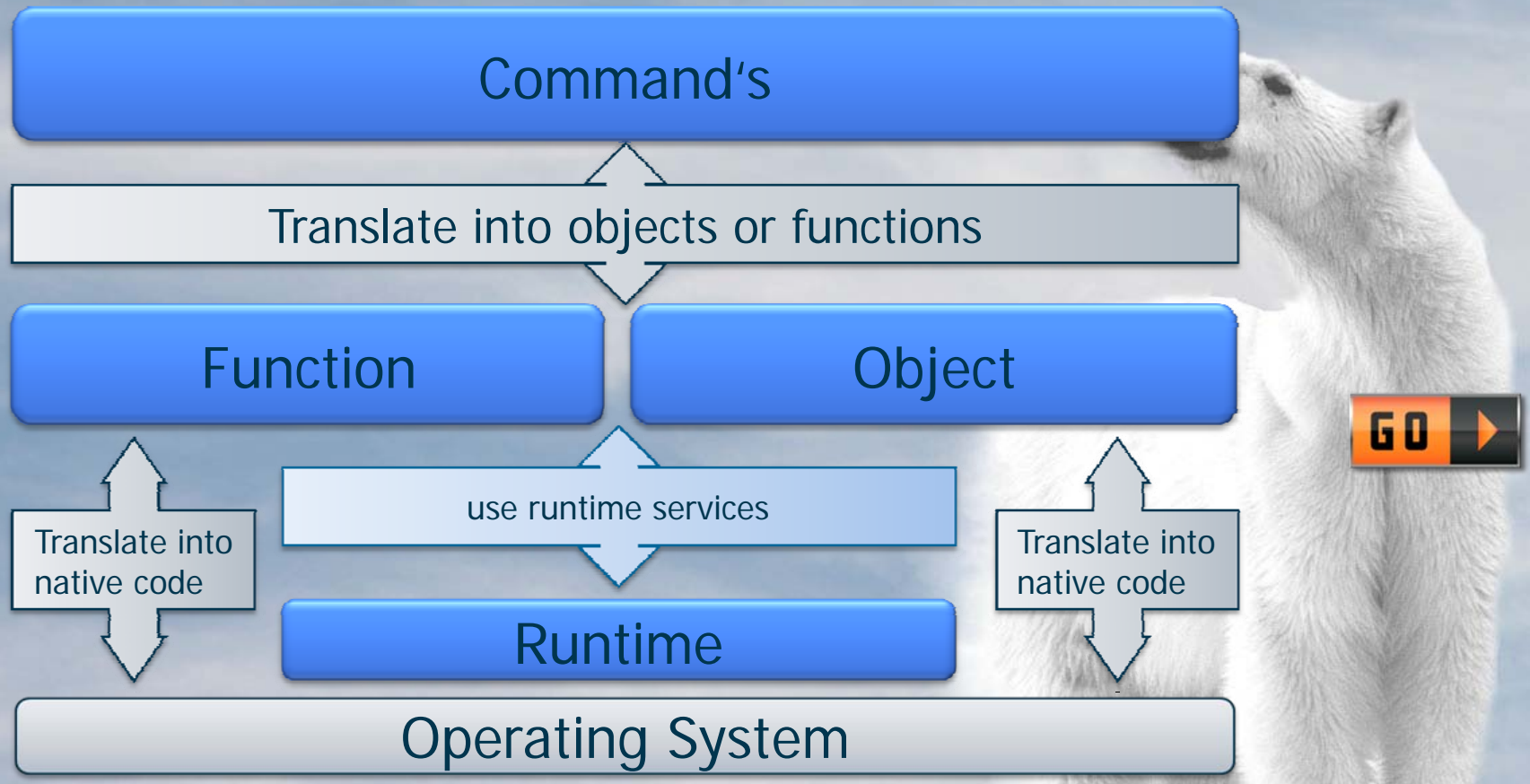
The grafik on the left compares the performance of different programming languages to illustrate the power of a modern hybrid compiler technology.

First, look at the difference between Xbase++ and Clipper. That's a ~45 times performance increase, while VFP is faster than Clipper it is still ~30 times behind Xbase++. Second, VFP is still able to outperform latest dynamic languages such as IronPhyton running on .NET DLR 4.0. This simple shows how bad the DLR performs.

And finally, Xbase++ is only ~5 times slower than a native C# 3.5 implementation. Taking into account the increased productivity of a 4GL such as Xbase++ compared to a system language such as C# is, the performance difference is rendered irrelevant in most if not all real world application scenarios.

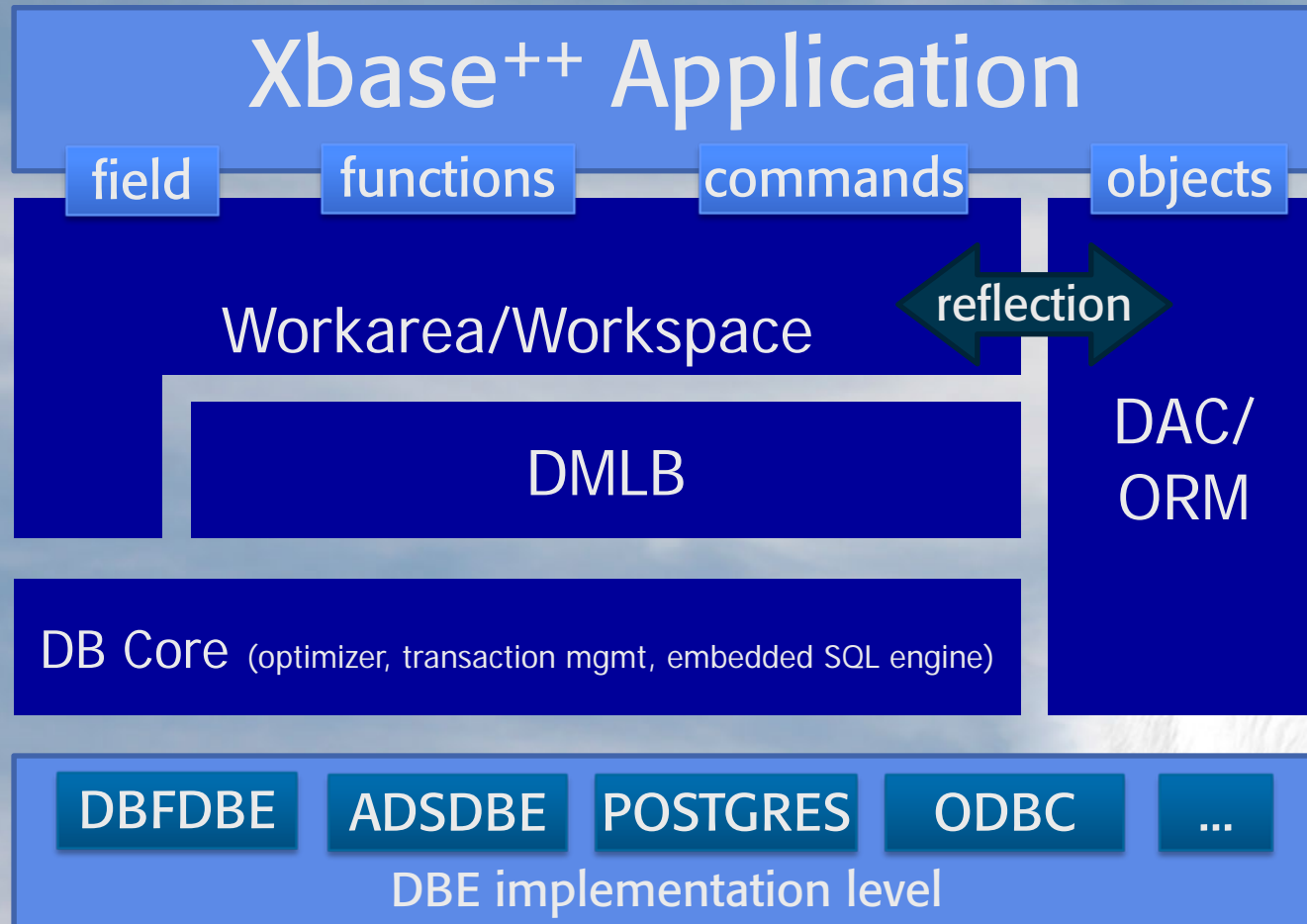
Commands, Functions, Objects and Codeblocks!

- Xbase++ is a hybrid compiler, generating native code and pcode.
- The Xbase++ language is multitiered



Database Engines

- The Xbase++ language is isolated from the underlying DBMS



But what about SQL?

- ODBC DatabaseEngine. SQL result-set becomes workarea
- ADO via ActiveX -☹
- Unified SQL*
 - Remote SQL for different data sources.
 - SQL statements are first class members of the language
 - Same SQL syntax for different backends, statements get rewritten by database engine on demand.
- Universal SQL*
 - Local SQL, embedded transactional in-memory SQL engine
 - SQL statements are first-class members of the language
 - Works with:
 - Workarea
 - DBF/NTX/CDX/DBT/FPT, CSV/TXT file
 - ODBC Datasource or any other remote SQL source
 - Objects, Arrays
 - XML tree or whatever you want it to be



*) Technologies currently under development/customer preview.

Multithreading

Automatic:

Garbage Collection
Event-Dispatching

Intelligent:

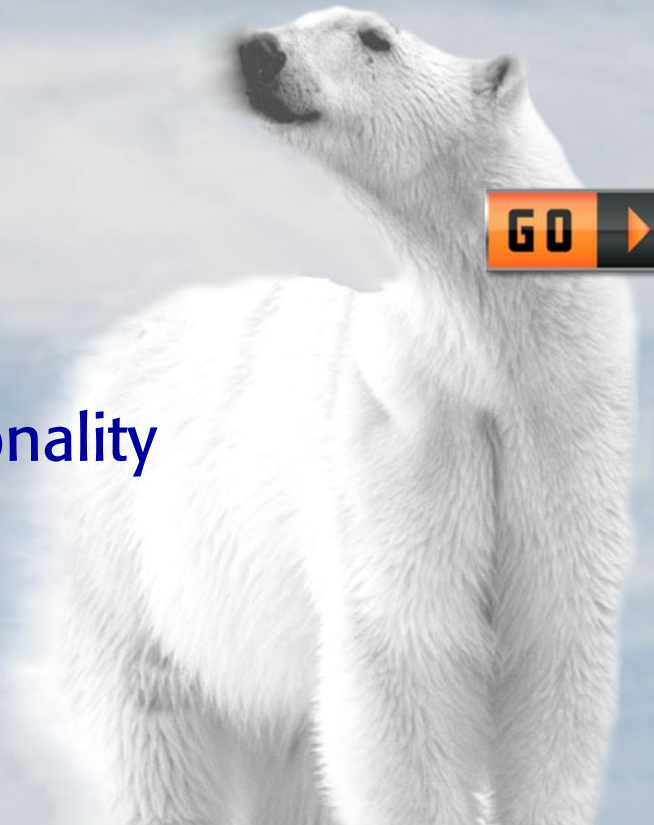
No locking required
No deadlocks
SYNC methods

- In practice
 - Xbase++ does multithreading for more than 10 years
 - it is widely used in our customers applications to
 - Perform work in the background
 - Make the UI more responsive
 - Increase performance
- Research efforts at our labs now focus on
 - Automatic scaling over n-cores ($8 \leq n \leq 256$)



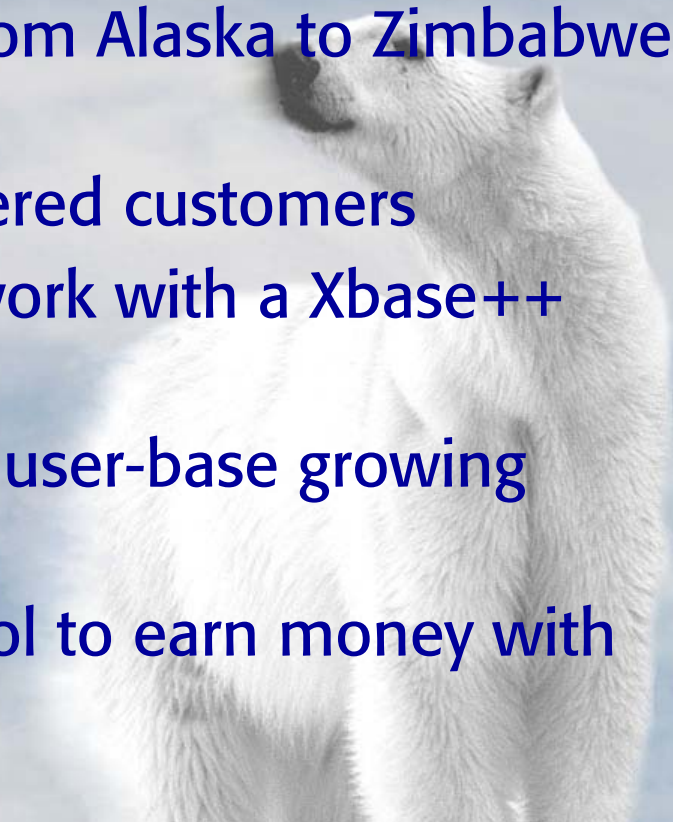
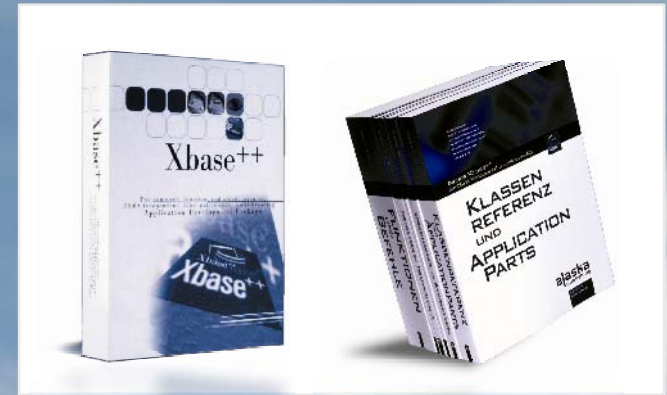
And the UI

- No UI, StdIn/StdOut or Console Textmode
 - command line tools...
- XbpCRT to mix GUI controls and Textmode
 - Seamless migration path into the GUI
- Xbase Parts Windows UI
 - More than 40 UI components
 - Owner drawing and visual styles
- ActiveX Controls
 - Extend visual appeal or UI functionality
- XAML (prototyping stage)
- Web using WAA and HRF



Who is using Xbase++

- >90% of the top-500 enterprises
- Thousands of mid-sized companies to develop in-house or vertical-market solutions in more than 120 countries.
- Tens of thousands of freelancers from Alaska to Zimbabwe in their project business
- There are more than 25.000 registered customers
- More than one million end-users work with a Xbase++ solution each day – world wide
- Around 300 new leads per month, user-base growing slowly, but steady
- Xbase++ is not a hype, its just a tool to earn money with



Why you should not buy Xbase++ today!

- Because Xbase++ today misses:
 - Wizards and Builders
 - Visual design tools for database, report and UI
- But
 - Visual Xbase++ is aimed to cover those „missings“
 - We have taken the good from VFP and designed a open and customizable application design environment
 - Of course, we are to add a bunch of new technologies and concepts



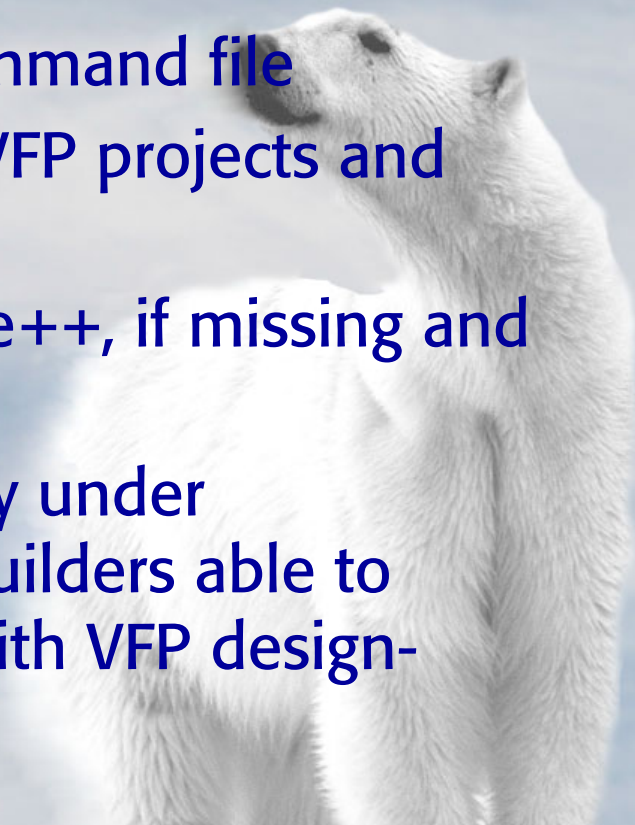
Roadmap for VFP developers!

- **Timetable:**

- 2008/2009 learning & prototyping
- 2010/2011 implementation & release

- **Action-Plan**

- Deliver a vfp.ch plus fox26.ch command file
- Deliver a transpiler to transform VFP projects and code to Xbase++
- Add VFP unique features to Xbase++, if missing and suitable.
- Enhance Visual Xbase++ currently under development with wizards and builders able to provide backward compatibility with VFP design-data.



Summary

- Xbase++ can easily adapt other xBase language dialects such as Visual FoxPro
- Xbase++ is a piece of advanced technology
- Xbase++ is feature complete and solid, in use for mission critical applications for a decade!
- Mission Statement:

Alaska Software aims to become for **Visual FoxPro** and **FoxPro** developers what we constitute for Clipper developers: **the natural successor!**



CU @ Southwest Fox

2010, 2011, 2012....

Thanks for your attention

Please visit us at
www.alaska-software.com

alaska
software