U/WIN* – UNIX** for Windows***

David Korn
AT&T Labs
dgk@research.att.com

August 15, 1997

* U/WIN is a registered trademark of Global Technologies Ltd., Inc.

** UNIX is a registered trademark of X/Open Ltd.

*** Windows is a registered trademark of Microsoft, Inc.

MOTIVATION

- Large collection of Windows software
- More familiar GUI
- Lots of existing UNIX applications
- Management requires Windows NT
- UNIX is simpler
- Enable UNIX development for Windows
- Bridge gap between UNIX and Windows

GOALS

- Minimal change to existing UNIX source
- Minimal performance overhead
- Ability to mix and match UNIX and Windows
- Windows NT and Windows 95
- Readily available – including source
- Learn about Windows NT and Windows 95

ADVANTAGES OF WINDOWS NT

- Low Cost
- Lots of Software
- Controlled by Microsoft
- Source Portability
- Similar to Windows 95
- Many Experts

DRAWBACKS TO WINDOWS/NT

- Complexity - >1000 calls
- Subsystems are separate
- Controlled by Microsoft
- Not very innovative
- Too much like VMS
- Not many experts

PROBLEMS

- Case sensitivity + namespace
- Pathname syntax
- `<nl>` vs. `<cr><nl>`
- Signal handling
- Incomplete interface
- Permission mappings
- Building shared libraries harder

PORTING CHOICES

- Rewrite applications
- Use middleware framework
- Use POSIX subsystem
- Microsoft C library
- Use `NuTCracker` from DataFocus
- Use `Portage` from Consensys
- Use `OpenNT` from Softway
- Use `Cygwin` from Cygnus
- Use `U/WIN` from AT&T
- Write UNIX/POSIX library interface

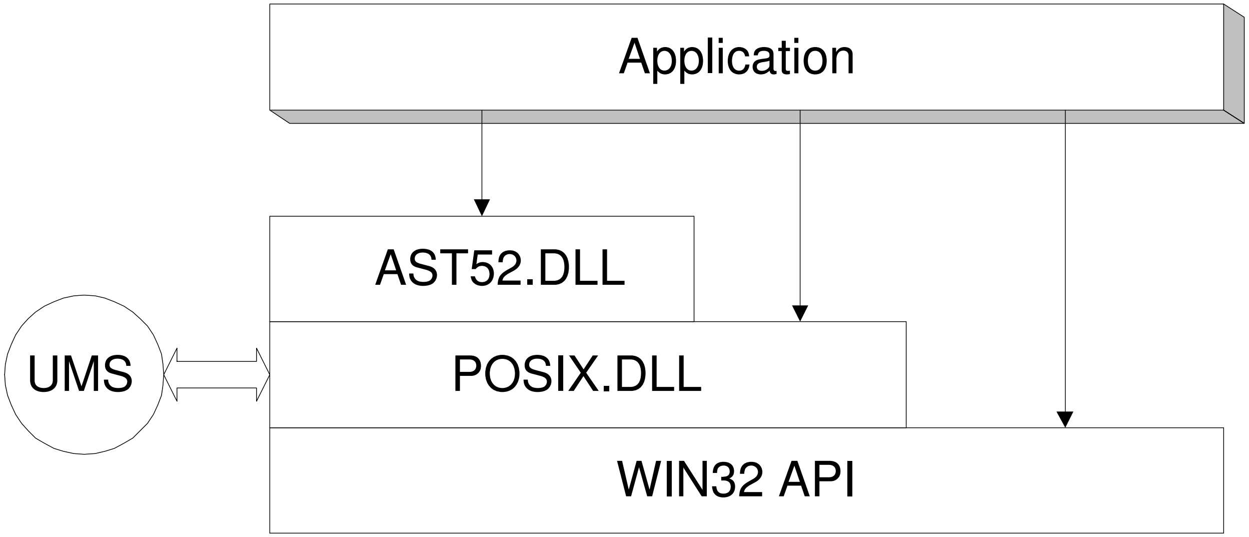
THE POSIX SUBSYSTEM

- Quick way port some UNIX application
- Can’t call NT application from POSIX
- Can’t make WIN32 calls from application
- No support for GUI applications
- No debugger support
- No support for tape or network naming
- Not supported on Windows 95

OUR POSIX LIBRARY - U/WIN

- Written using WIN32 API calls
- Virtually all UNIX system calls
- `stdio` library based on `sfio`
- Microsoft C library for remainder
- Supports hard links/symbolic links
- Memory mapping and dynamic linking
- System V shared memory, semaphores, fifos
- BSD sockets over Winsock
- Setuid/setgid programs on Windows NT

U/WIN Architecture



U/WIN Server Process for NT

- Starts at boot time
- Run as Administrator – `uid` 0
- Creates `/etc/passwd`, `/etc/group`
- Creates security tokens
- Runs `/etc/rc`

CURRENT STATUS

- Version 1.30 2Q97
- C compiler wrapper for Visual C/C++
- *Practical Reusable UNIX Software* tools
- Most X/Open interfaces
- `libast`, curses, X11R6 and other libs
- About 170 UNIX tools
- `ksh93` with editing and job control
- AT&T `nmake` and GNU `make`
- New BSD `vi`
- telnet, rlogin, and rsh daemons

PERFORMANCE

- No loss in I/O performance
- `fork/exec` factor of 3 slower
- `vfork/exec` ~ 30% slower
- `uwin_spawn()` ~ 25% slower
- File deletion ~ 2x slower

FUTURE

- Case sensitive file names
- Mount table
- Rest of inet daemons
- 64-bit file support
- POSIX pthreads API
- Registry file system
- `tksh` and `deet` debugger
- I18N based on UFT8
- *n*-DFS
- Multi-format Documentation
- Run SCO/Linux binaries

AVAILABILITY

- Available freely within AT&T
- 4Q96 release available to Lucent/NCR
- Used by several projects
- Freely available for non-commercial use
  `http://www.research.att.com/sw/tools/uwin`
- Commercial licenses from Global Technologies
  `http://www.gtlinc.com`
- Commercial licensing from AT&T