

Portiranje programske podrške sa Unix-a na Windows-e

Dobrica Pavlinušić

<dpavlin@foi.hr>

<http://www.foi.hr/~dpavlin/>

Fakultet organizacije i informatike Varaždin

Na početku...

port - to create a new version of (an application program) to run on a different hardware platform (sometimes fol. by over).

Random House Websters College Dictionary

- Portiranje izvornog koda
 - UNIX ⇒ Windows (Cygwin, Uwin, OpetNT, NuTCracker)
 - Windows ⇒ UNIX (Twin, Microsoft)
- Izvršavanje gotovih programa
 - Windows ⇒ UNIX (WINE, Wabi, SoftWindows...)

Sadržaj

- Zašto portirati?
- Kako portirati?
- Windows i Unix arhitektura
- Alati za portiranje
- CASE
- Zaključak

Zašto portirati?

- održavanje više platformi istovremeno (npr. UNIX i Windows)
- svijet postaje sve više Microsoft orijentiran \Rightarrow pritisak da se sve izvršava pod Windows operacijskim sustavima
- zaštita ulaganja u Unix aplikacije (napravljene vlastitim razvojem?)

Zašto portirati?

- korišćenje poznatih Unix alata na Windows-ima
- iskorištavanje obilja izvornog koda za Unix pod Windows-ima

Načini prijenosa programske podrške

- kompletno ponovno pisanje aplikacije za Win32 API
- emulacija UNIX poziva bibliotekom (bazirana na Win32 API-ju)
 - Cygwin, UWin
- korištenje zajedničkih biblioteka za Unix i Win32
- Microsoft POSIX podsustav (samo na Windows-ima NT)
 - OpenNT

Ciljevi portiranja

- mogućnost izvršavanja Unix programa koji su dostupni u izvornom kodu bez većih promjena na Windows (Win32) računalima (Windows 95/98, NT)

Windows arhitektura

- Windows NT arhitektura:
 - slojevita, bazirana na microkernelu (slično Mach-u) sa više različitih podsustava za izvršavanje aplikacija (trenutno postoje Win32, OS/2 i POSIX)
- Windows NT POSIX podsustav:
 - strogo prati POSIX 1003.1 specifikaciju iz 1990 godine i onemogućava pozivanje funkcija izvan tog podsustava (npr. Win32)

Windows arhitektura

- Windows 95/98 implementacija:
 - podržava veći dio funkcija koje postoje u Win32 API-ju
- Nekonzistentnost Win32 API-ja:
 - greška se vraća kao rezultat 0 ili -1 zavisno od funkcije
 - ponekad je rezultat 0 indikacija uspješnog završetka

sigurnost

uid, gid

ACL

datoteke

putanje sa /

svi diskovi su u hijerarhijskom stablu
direktorija

osjetljive na veličinu slova

symbolic i hard linkovi

LF za kraj reda

putanje sa \

diskovi imaju oznake u obliku c:\
neosjetljive na veličinu slova

symbolic linkovi na svim Win32

hard linkovi na NTFS-u

CR/LF par za kraj reda ⇒ razlika
između text i binary otvaranja
datoteka

handles i file descriptors

file descriptors u točno određenom redu unutar jednog procesa

handles bez određenog reda unutar procesa ⇒ zahtjeva tablicu za pamćenje redoslijeda

kreiranje procesa

fork/exec

CreateProcess (kao spawn)

socket-i

Berkeley sockets

WinSock - Microsoftova implementacija Berkeley socket-a (select radi samo na socketima)

UNIX na Windows migracija

- Cygwin (GNU-win32)
- UWin (UNIX for Windows)
- OpenNT

Da li sve patke lete u Redmond?

Cygwin

- Sučelje (API) za 32 bitne Windows-e namijenjeno portiranju Unix aplikacija
 - Cygwin.dll
 - POSIX.1/90 funkcije osim setuid i mkfifo
 - standardni ANSI C pozivi
 - uobičajeni BSD i SVR4 pozivi uključujući Berkely sockets
- Nastao iz potrebe portiranja development alata na Win32 platformu

Cygwin

način implementacije

- kod prvog pokretanja stvara se dijeljeni segment memorije koji koriste svi programi koji se izvršavaju pod Cygwin.dll-om
 - file descriptori, podrška za exec i fork
- svaki proces ima i per_process strukturu sa podacima:
 - pid, uid, maske signala itd.

Cygwin

način implementacije

- dll je implementiran korištenjem Win32 API-ja
- procesi mogu pristupati Win32 pozivima (npr. za izradu GUI-ja)
- ne teži se potpunom poštivanju specifikacija (npr. POSIX.1)
- emulacija uid i gid podataka iz ACL-a kod NT-a

Cygwin

način implementacije

- Datoteke
 - podrška za Win32 i POSIX putanje (\ i /)
 - podrška za UNC putanje (//nešto)
 - korištenjem mount komande moguće je odrediti disk koji će biti root (/)
- API za programe
 - koristi se Cygnus-ova newlib ANSI C biblioteka (izvedena iz BSD-a)
- mogućnost korištenja Win32 poziva

Cygwin

način implementacije

- Kreiranje procesa (fork/exec)
 - non-copy-on-write implementacija
 - fork/exec je ne-efikasan poziv za implementaciju na Win32 API-ju; poželjno ga je zamijeniti sa spawn-om (koji je sličniji Win32 pozivu CreateProcess)
 - Win32 poziv CreateProcess ne podržava overlay (potreban za exec)

Cygwin

način implementacije

- Signali
 - pri pokretanju DLL-a stvara se jedna nit (dretva) koja se brine za signale
 - podržana većina signala (job controll je podržan u shell-ovima)

Cygwin

način implementacije

- **Socketi**
 - pozivi istim funkcijama u WinSock-u
 - WinSock mora biti inicijaliziran prije prvog poziva socketima (Cygwin učitava i inicijalizira winsock.dll prema potrebi)
- **Select**
 - u Win32 samo na socket handle-ovima
 - u Cygwin na svim tipovima handle-ova

Cygwin

nedostaci

- implementacija nije optimalna što se tiče performansi
- ne podliježe niti jednom standardu u potpunosti
- dijeljeni memorijski segment je sigurnosna rupa
- ne podržava višenitno izvršavanje (npr. POSIX threads, iako sam dll koristi niti)
- slaba podrška Unix security modelu

UWIN - UNIX for Windows

ciljevi razvoja

- korištenje samo Win32 API-ja (izvršavanje na Windows-ima 95/98 i NT)
- IEEE POSIX.1 sučelje prema biblioteci
- "stvari" programi zahtijevaju i proširenja standarda kao što su socketi i/ili IPC
- emulacija tekstualnog terminala (curses)
- podrška za dinamičko povezivanje
- mogućnost root privilegija (setuid, setgid)

UWIN - UNIX for Windows

ciljevi razvoja

- kompletni alatima za razvoj (kompajliranje bez mijenjanja makefile datoteka)
- pristup resursima OS-a (npr. registry-ju, taskovima/procesima) preko UNIX API-ja
- robustan sustav, bez vidljivog usporenja u odnosu na programe za Win32 API
- integracija sa programima za Win32 API

⇒ potrebna je potpuna UNIX funkcionalnost bazirana na X/Open Release 4 standardu

UWIN - UNIX for Windows

način implementacije

- posix.dll i ast52.dll implementiraju funkcije
- kod Windows NT-a proces UMS
 - generiranje ACL-ova za setuid i setgid
 - održavanje datoteka /etc/group i /etc/passwd

UWIN - UNIX for Windows

način implementacije

- `posix.dll`
 - tablica otvorenih datoteka (u dijeljenoj mmap datoteci)
 - struktura sa podacima o procesima
 - `ppid`, `pgid`, maska signala, tablica procesa
 - implementira `malloc`
 - file descriptors \Leftrightarrow handles
 - Unix \Leftrightarrow Win32 putanje

UWIN - UNIX for Windows

način implementacije

- `ast52.dll`
 - oslanja se na Microsoft C biblioteku
 - `stdio` (pozivi `posix.dll`-u za mapiranje imena)
 - `spawn` (zamjena za `fork/exec`)

UWIN - UNIX for Windows

način implementacije

- signali
 - nit koja čeka na signale
- terminali (POSIX termios)
 - dvije niti (unos, ispis)
- socketi
 - pozivi WinSock-u

UWIN - UNIX for Windows

nedostaci

- nemogućnost kreiranja datoteka čiji nazivi završavaju sa točkom
- nema pravih crypt-iranih passworda nego se koriste security tokeni
- problemi sa istovremenošću
- problemi sa pravima pristupa

OpenNT

- proširenje Microsoft-ovog POSIX podsustava
- ne podržava Win32 pozive
 - bolje performanse
 - potpunije podržavanje standarda
 - ne mogu se koristiti Win32 i UNIX pozivi u istoj aplikaciji
- ne podržava Windows 95/98

CASE - portiranje

Cygnus Cygwin 20.0

UWin 1.33 (1.60)

mkisofs 1.12b4

boa 0.92q+

Cygwin - mkisofs

- configure ne radi
 - `#!/bin/sh` → `#! sh`
 - make bez problema
 - 486618 bytova mkisofs.exe
 - nakon strip-a 65024 bytova
- ⇒ *poznata gnu okolina sa poznatim alatima*

Cygwin - boa

- makefile
 - CC = gcc
 - boa.c
 - setrlimit (zakomentirano)
 - boa.conf
 - User nobody
- ⇒ nakon pokretanja ne šalje podatke

Uwin 1.60 \Rightarrow 1.33

- 1.60: (98)
 - vi ne radi (nedostaje curses10.dll)
 - egcs-2.91.57 se ruši (vfork?)
 - GNU configure problemi sa sed-om
- 1.51: (NT)
 - egcs-2.90.24 radi bez problema

Uwin 1.60 \Rightarrow 1.33

- 1.41: (98)
 - problemi kod instaliranja
- 1.33: (98)
 - pokretanje login.exe blokira računalno \Rightarrow pokrenuti ksh.exe
 - Microsoft Visual C 2.1

Uwin - mkisofs

- configure i make bez problema
- veliki broj warning-a (MSVC!)
- 108032 bytova mkisofs.exe

Uwin - boa

- problemi sa lex-om \Rightarrow korišten GNU flex
- setrlimit (kod linkanja)
- nema mijenjanja boa.conf - ne starta se kao root na Windows-ima 98
- javljala grešku 404

CASE - zaključak

- Cygwin

- ugodnija okolina
- kompatibilnost sa GNU alatima

- UWin

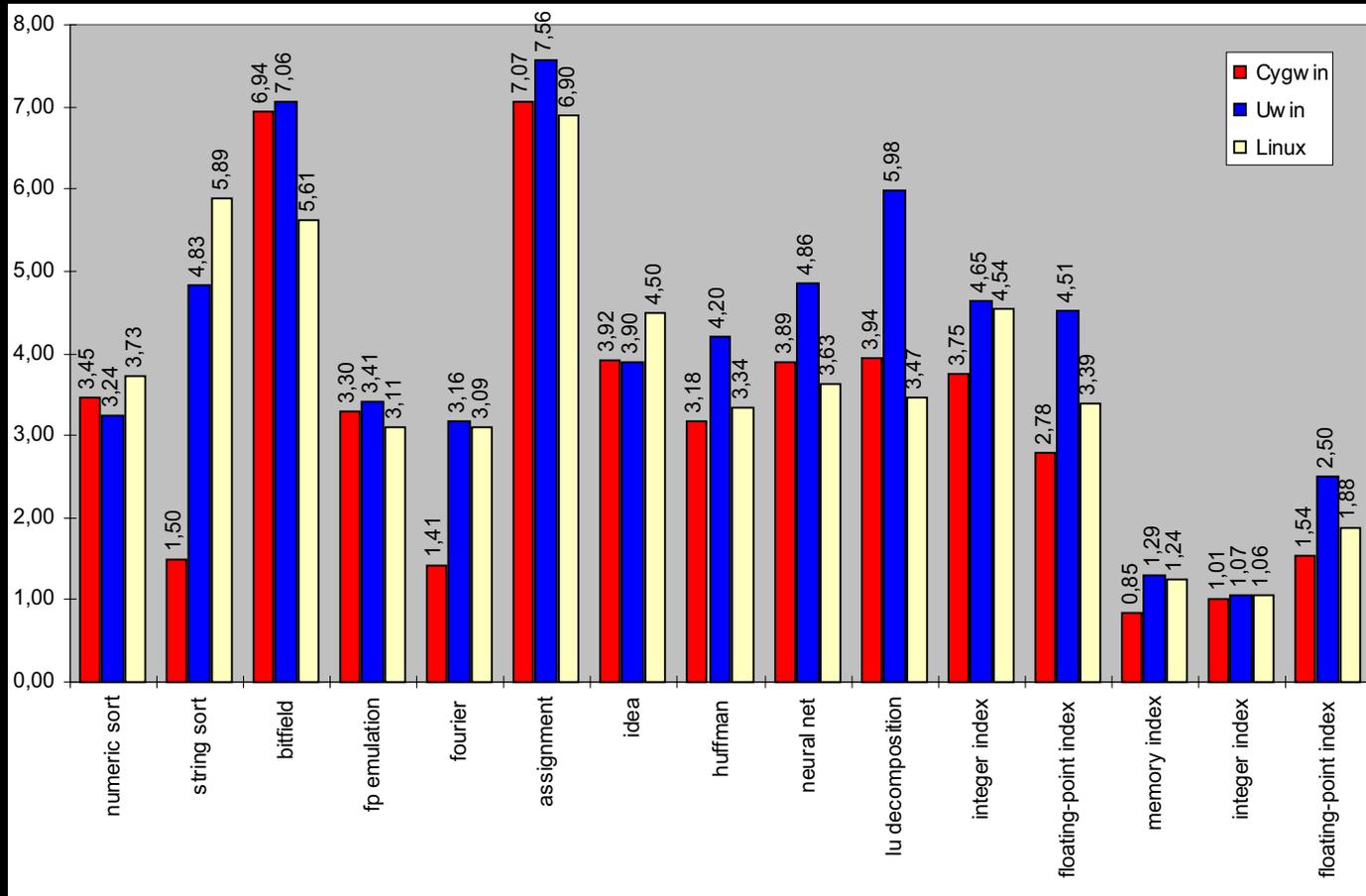
- brže kompajliranje (sa MSVC!)
- stabilnost

Ne omogućavaju sasvim jednostavno portiranje složenijih aplikacija bez bitnih modifikacija izvornog koda!

Uspješni portovi

- Apache web server
- perl programski jezik
- MIT X11 GUI sučelje
- GNU alati
 - fileutils, binutils, gcc, egcs, itd.

Usporedba brzine



Ostala rješenja

- ručno portiranje aplikacije na Win32 API
- što napraviti sa GUI-jem?
 - Win32 API (Cygwin, UWin)
 - Tcl/Tk (Cygwin, UWin, OpenNT)
- djgpp - da, DOS ali korisno
 - 32 bitni extender, DPMI

Portiranje programske podrške sa Unix-a na Windows-e

poziv za diskusiju