

Svobodný software

99 % všech desktopů na
platformě MS Windows
Monopol !!!

Jaké z toho plynou nevýhody?

Jaké z toho plynou výhody?

source code – zdrojový kód

- obsah souboru nebo souborů napsaný v určitém programovacím jazyce
- čitelná podoba – zápis programu je možno prohlédnout v libovolném textovém editoru

```
* Dialog and Message Boxes
```

```
*  
* Dialog widgets are used to pop up a transient window for user feedback.  
*/
```

```
#include <config.h>  
#include <gtk/gtk.h>
```

```
static GtkWidget *window = NULL;  
static GtkWidget *entry1 = NULL;  
static GtkWidget *entry2 = NULL;
```

```
static void
```

```
message_dialog_clicked (GtkButton *button,  
                        gpointer user_data)
```

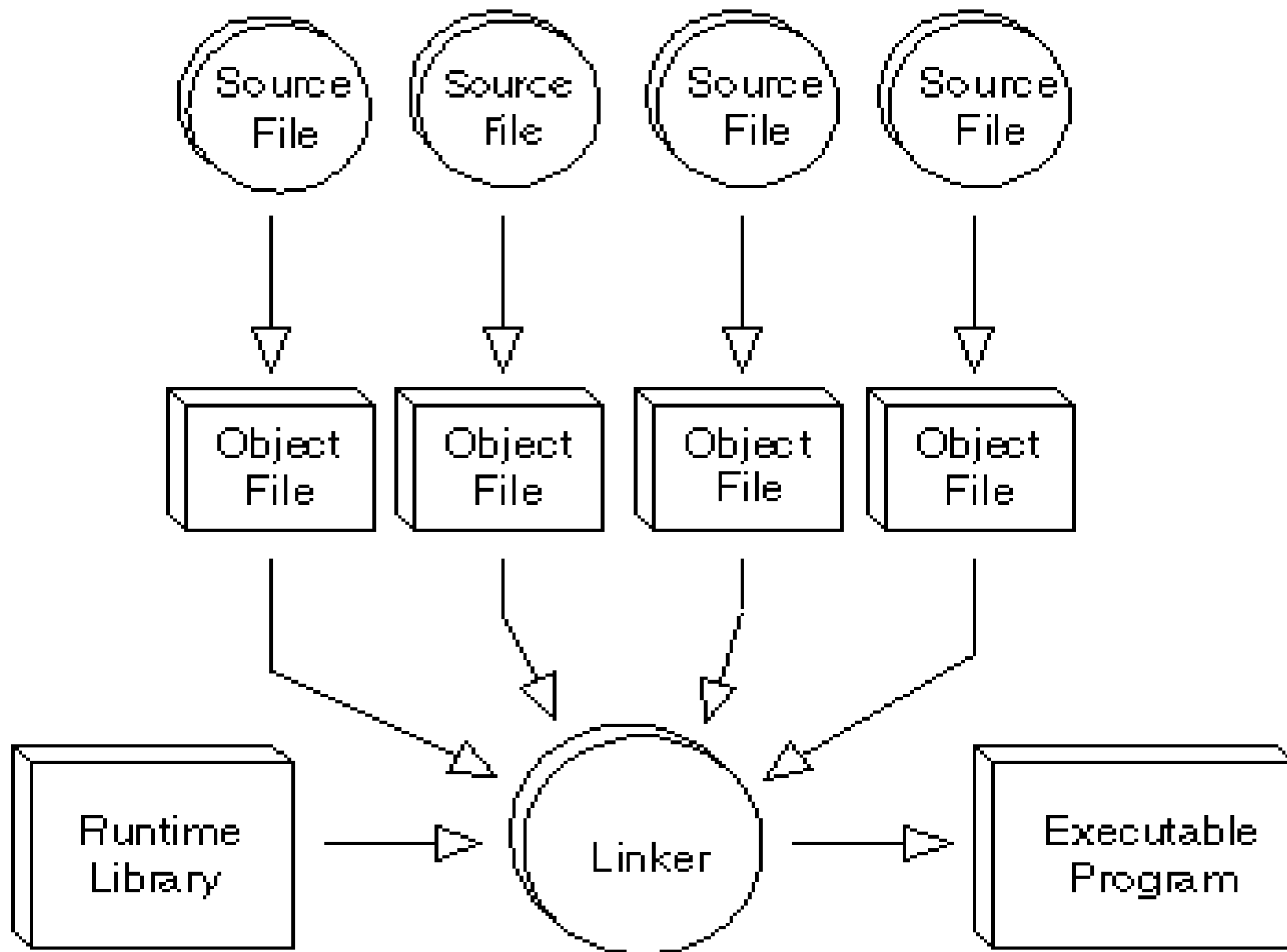
```
{  
    GtkWidget *dialog;  
    static gint i = 1;  
  
    dialog = gtk_message_dialog_new (GTK_WINDOW (window),  
                                    GTK_DIALOG_MODAL | GTK_DIALOG_DESTROY_WITH_PARENT,  
                                    GTK_MESSAGE_INFO,  
                                    GTK_BUTTONS_OK,  
                                    "This message box has been popped up the following\n"  
                                    "number of times:\n\n"  
                                    "%d", i);  
  
    gtk_dialog_run (GTK_DIALOG (dialog));  
    gtk_widget_destroy (dialog);  
    i++;  
}
```

```
static void
```

```
interactive_dialog_clicked (GtkButton *button,  
                            gpointer user_data)
```

```
{  
    GtkWidget *dialog;
```

kompilace – překlad



kompilace – překlad

proces převodu zdrojových kódů programu do spustitelné (binární) podoby programu za pomoci kompilátoru - překladače

binary code – binární kód

- spustitelná binární podoba programu
- vlastní program, který spouští a používá uživatel

reverzní inženýrství

- z binární spustitelné podoby programu, již nejsme schopni získat původní zdrojový kód
- **ALE** metodami reverzního inženýrství jsou *někteří* schopni zjistit, jak části programu fungují, případně jejich funkci pozměnit

hacker

- jedinec jehož/jejíž zálibou je zkoumání detailů programovatelných systémů a posouvání jejich možností
- programuje a hackuje nadšeně až návykově

proprietární software
close source

software s uzavřeným zdrojovým kódem

X

neproprietární software
open source

software s otevřeným zdrojovým kódem

u proprietárního software

získáváme pouze právo jej používat, licenci k užívání, za určitých podmínek:

- omezení na vlastníka licence**
- na určitý počet počítačů většinou jeden kromě multilicencí**
- případně i na konkrétní hardware, se kterým byl software zakoupen OEM licence**
- zákaz reverzního inženýrství, nesmíme nijak software upravovat či pokoušet se zjistit, jak funguje**
- nemáme přístup ke zdrojovým kódům**
- nesmíme kopírovat, max. jedna záložní kopie**
- nesmíme distribuovat dál**
- nesmíme zahrnout jako součást vlastního software**
- nemůžeme převést licenci na někoho jiného, ať již zadarmo či za úplatu**

freeware

Software, který je šířen zdarma, například na internetu nebo na různých CD. Program je možno provozovat zdarma po neomezenou dobu a je možno jej i zdarma šířit dále. Není však dovoleno šířit jej za úplatu.

Z původní definice freeware také plyne, že autorská práva k takovému programu drží jeho autor a není tedy dovoleno bez jeho souhlasu program jakkoliv měnit či upravovat pro komerční účely.

Nemusí být a také většinou nejsou dostupné zdrojové kódy programu.

shareware

Software šířený zdarma, kupříkladu na internetu či jinak. Obvykle je možno program provozovat na zkoušku po určitou omezenou dobu, po jejímž uplynutí přestane celý program nebo jeho klíčové funkce fungovat. Pro další používání programu je nutno jeho autorovi nebo distributorovi zaplatit. Po zaplacení obdrží uživatel údaje, kterými se program zplnohodnotní, nebo celý nový program v poslední verzi s neomezenou dobou užívání. Není možno jej dále šířit za úplatu a nejsou k němu dostupné zdrojové kódy.

free software – svobodný software

dva přístupy:

- BSD – Berkeley Software Distribution

Kalifornská univerzita v Berkeley (1975 získala licenci Unixu)

- GNU GPL

BSD licence

program pod touto licencí lze:

- libovolně používat
- kopírovat
- modifikovat
- modifikace šířit pod jinou licenci
tedy i takovou, která zakáže modifikace volně používat, kopírovat a šířit!

GNU

- Richard Stallman – programátor v bostonské Massachusetts Institute of Technology (MIT)
- v roce 1984 zahajuje projekt GNU, jehož cílem je vytvořit operační systém včetně uživatelských aplikací pouze ze svobodného software
- GNU's Not Unix – rekurzivní akronym
- sám začíná programovat některé komponenty (např. editor Emacs)
- v roce 1985 publikuje v Dr. Dobb's Journal svůj GNU Manifest
- www.gnu.org
www.gnu.cz

Richard Matthew Stallman



gnu [nu:] pakůň žíhaný/běloocasý



GNU GPL

- v roce 1985 zakládá nadaci Free Software Foundation (FSF)
- Obecná veřejná licence GNU – GNU General Public Licence
GNU GPL 1989, 1991
- <http://staff.cesnet.cz/~lhotka/gnugpl-cz.html>
- copyleft – autorizuje software, ale ve smyslu trvalého zajištění volného nakládání se software i jeho odvozeninami
- GNU GPL na rozdíl od BSD licence zajišťuje, že i všechny programy odvozené od svobodných programů, tedy všechny modifikované zdrojové kódy, zůstanou také svobodné

Se softwarem, který je distribuován pod licencí GNU GPL, musí být nakládáno takto:

- nikdo nesmí být jakkoli omezován v jeho užívání, distribuci, prodeji či darování
- software musí být distribuován se zdrojovým kódem nebo musí být zdrojový kód jinak veřejně dostupný např. na internetových stránkách projektu atd.
- zdrojový kód software nesmí být uveden v úmyslně zastřené nesrozumitelné formě
- každému musí být umožněna modifikace a další využití zdrojového kódu programu v softwaru jiných subjektů, avšak za stejných podmínek jako u původního software

Se softwarem, který je distribuován pod licencí GNU GPL, musí být nakládáno takto:

- musí být výslovně dovoleno volně distribuovat software vytvořený na základě jiného svobodného softwaru
- může být vyžadováno, aby odvozený software nesl jiné jméno nebo číslo verze odlišné od softwaru původního
- nesmí být nijak omezena jakákoli osoba či skupina osob v užívání softwaru
- nesmí být jakkoli omezováno využití software v jakékoli oblasti lidského počínání (např. v komerci či v genetickém výzkumu)

Se softwarem, který je distribuován pod licencí GNU GPL, musí být nakládáno takto:

- výše zmíněná ustanovení automaticky přecházejí na všechny, kteří svobodný software užívají bez nutnosti, dalšího licencování
- licence se vztahuje na všechny části programu respektive zdrojového kódu
- nesmí být omezována distribuce jiného softwaru společně se svobodným softwarem (např. není možno trvat na tom, že veškerý software distribuovaný na společné médium se svobodným softwarem, musí být také svobodný software)
- musí být zajištěna technologická neutralita (nesmí být např. bráněno použití softwaru na platformě MS Windows či hardware PowerPC atd.)

Operační systémy s otevřeným zdrojovým kódem

- **1990 počátek vývoje OS GNU HURD**
- **1993 FreeBSD, NetBSD, OpenBSD**
- **1991 počátek vývoje OS GNU Linux**
- **Linus Torvalds**
- **Minix**
- **Freax -> Linux**
- **kernel - jádro operačního systému**



From: torvalds@klaava.helsinki.fi (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Keywords: 386, preferences
Message-ID: <1991Aug25.205708.9541@klaava.helsinki.fi>
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki
Lines: 20

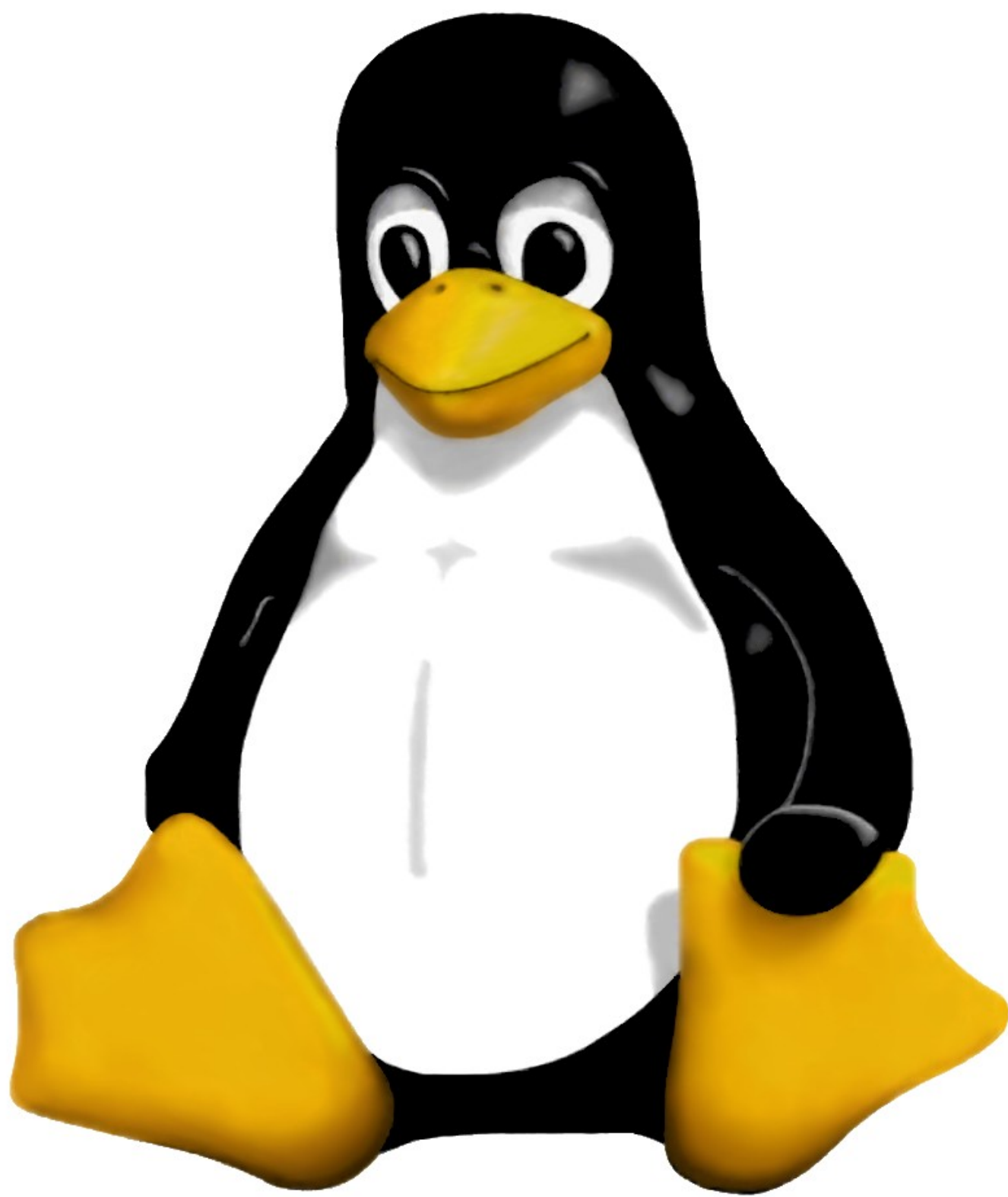
Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).



Co je tedy GNU Linux?

- jádro operačního systému – kernel
- souborový systém
- příkazový interpret – shell – příkazy
- nástroje – drobné aplikace
- textové editory
- archivery
- překladače
- grafická nádstavba – X-server
- okenní managery – Gnome, KDE
- aplikace - Openoffice, Gimp, Mozilla, Firefox, xpdf, gftp, hry
- servery - Apache, Sendmail, PostgreSQL, DNS, DHCP, FTP

Vývoj, distribuce a instalace GNU Linux

- **projekt** – webová stránka projektu
- **download z Internetu, CD, DVD média**
- **manuální instalace** – Linux from Scratch
- **pomocí instalátoru distribuce**
- **distribuce** – balíčky aplikací (packages)
 - Red Hat, Fedora, Mandrake, SuSELinux, Debian, Slackware, Gentoo, Danix, Knopix, Ubuntu atd.

výhody a nevýhody svobodného softwaru

Zdroje:

<http://en.wikipedia.org/>

<http://www.computerhope.com/history/>

<http://www.abclinuxu.cz/>

<http://www.root.cz/>

<http://linuxshop.ru/linuxbegin/win-lin-soft-en/table.shtml>