



QEMU beüzemelése és részletes ismertető

Név: Rehó Imre Béla

Tárgy neve: Virtualizációs technológiák és alkalmazásaik

Tárgy kódja: BMEVIMIAV89

Oktatók: Micskei Zoltán, Tóth Dániel

Dátum: 2009. december 11.

1 A QEMU

1.1 A virtualizációs technológiák

A virtualizáció fogalma: az erőforrások elvonatkoztatása az erőforrásokat nyújtó elemektől, valamint az erőforrások megosztása és finomítása. A számítógépes virtualizációs technológiák mára egy külön tudományterületté fejlődött ki, és az idő teltével egyre bővülni fog a létjogosultsága a mindennapi életben. A virtualizációs technológiákat az élet számos területén sikerrel alkalmazzák, főleg a nagyobb vállalkozásoknál tapasztalható érdeklődés. A számítógépek hálózatba kötésével és az Internet terjedésének hatására szükség volt egy professzionális megoldásra, amely széles körben ki tudja elégíteni a felhasználók igényeit, és egyben jól átlátható valamint karbantartható.

Többféle virtualizáció létezik, amik különböző szinteken valósulnak meg: platformvirtualizáció, operációs rendszer szintű virtualizáció, alkalmazás virtualizáció, erőforrás-virtualizáció, megjelenítés virtualizáció, stb. A sort lehetne még folytatni nagyon sok speciális fajtával is.

1.2 A QEMU bemutatása

A QEMU egy nyílt forráskódú processzoremulátor program és egyben dinamikus fordító. A nyílt forráskód nagy előny, mert ebből kifolyólag ingyenesen beszerezhető, változtatható, és platformfüggetlen, tehát nincs operációs rendszerhez kötve a futtathatósága. A program mérete is figyelemre méltó, ugyanis egészen kicsi a kategóriáját tekintve a többi ilyen programhoz képest. A QEMU-nak két fő funkciója van: komplett architektúrák emulálása, futtatási környezet emulálása. Nagyon sok fajta processzor-architektúra emulálására képes, itt egy kisebb felsorolás a teljesség igénye nélkül: x86, x86_64, PowerPC, Sparc, Coldfire, ARM, ARM-Cortex, OMAP, DEC Alpha, ETRAX CRIS, MIPS, MicroBlaze, stb.

Operációs rendszer, mint futtatási környezet emulálására a következő architektúrával, perifériákkal dolgozik:

i440FX host PCI bridge és PIIX3 PCI az ISA bridge-hez

Cirrus CLGD 5446 PCI VGA kártya vagy ál (látszólagos) VGA kártya, Bochs VESA bővítménnyel (hardver szint, beleépítve az összes nem standard móddal)

PS/2 egér és billentyűzet

PCI IDE csatoló, HDD és CD-ROM támogatással

Floppy lemez

NE2000 PCI network adapterek

Soros portok

Soundblaster 16 hangkártya

A QEMU mindezen kívül számtalan lehetőséget hordoz magában. Készíthetünk több virtuális gépet, attól függően mire van szükségünk. Lehetőségünk van kernel alapú virtuális gépeket is létrehozni, de ehhez megfelelő virtuális tulajdonságú processzorra van szükség. Az elkészített virtuális géphez lehet még floppy és CD-ROM image-t hozzáadni, valamint valódi merevlemezt is. Az USB és a VNC kezelése is a lehetőségek sorába tartozik.

2 QEMU telepítése, konfigurálása, futtatása

2.1 Windows környezetben

Első lépésként le kell töltenünk a QEMU valamelyik változatát, én az alábbi linkről szereztem be a 0.10.6.-os verziószámú példányt: <http://homepage3.nifty.com/takeda-toshiya/qemu/>
A .zip kiterjesztésű fájlt ki kell tömörítenünk a merevlemezünk egy általunk kreált és választott könyvtárba. Ez lesz az a könyvtár, ahonnan később a QEMU-t futtatni fogjuk, illetve itt célszerű tárolni a virtuális gép image-eket is. Ez a letöltött verzió tartalmaz egy linux.img nevű fájlt, ami a programunk tesztelésére ad lehetőséget. A programot nem kell külön installálni, mert a kitömörítés után azonnal futtatható formában el lehet indítani. A futtatást három különböző módon is megtehetjük: parancssorból, batch fájl használatával, és parancsikon segítségével is.

Parancssorból a következőképpen indítható el: **C:\bin\qemu> qemu -L . -hda linux.img**

Batch fájl használatával a következőt indítsuk el: **qemu-win.bat**

Parancsikonként az elérési út: "C:\Program Files\qemu-0.8.2-windows\qemu.exe" -L . -hda linux.img

Bármelyik módot is választjuk, a QEMU el fog indulni egy ablakban, és be fogja tölteni a linux.img által meghatározott Linux operációs rendszert. Ez az operációs rendszer egyszerű felépítésű, parancssorból kezelhető, nem vesz el számottevő időt az elindulása. A következő ábrán láthatjuk az eredményt.

```
QEMU
hdc: attached ide-cdrom driver.
hdc: ATAPI 4X CD-ROM drive, 512kB Cache
Uniform CD-ROM driver Revision: 3.12
Partition check:
  hda:
Soundblaster audio driver Copyright (C) by Hannu Savolainen 1993-1996
NET4: Linux TCP/IP 1.0 for NET4.0
IP Protocols: ICMP, UDP, TCP, IGMP
IP: routing cache hash table of 512 buckets, 4Kbytes
TCP: Hash tables configured (established 8192 bind 16384)
NET4: Unix domain sockets 1.0/SMP for Linux NET4.0.
VFS: Mounted root (ext2 filesystem) readonly.
Freeing unused kernel memory: 64k freed
EXT2-fs warning: mounting unchecked fs, running e2fsck is recommended

Linux version 2.4.21 (bellard@voyager.localdomain) (gcc version 3.2.2 20030222 (
Red Hat Linux 3.2.2-5)) #5 Tue Nov 11 18:18:53 CET 2003

QEMU Linux test distribution (based on Redhat 9)

Type 'exit' to halt the system

SIODSIFADDR: No such device
eth0: unknown interface: No such device
sh-2.05b# _
```

1. ábra

A következő két sor beírásával egy test-benchmark futtatható le:

sh-2.05b# cd nbench

sh-2.05b# ./nbench

```
QEMU
BITFIELD      :      4.2286e+07 :      7.25 :      1.52
FP EMULATION  :      7.8871 :      3.78 :      0.87
FOURIER       :      2749.7 :      3.13 :      1.76
ASSIGNMENT    :      1.8532 :      7.05 :      1.83
IDEA          :      319.38 :      4.88 :      1.45
HUFFMAN       :      76.812 :      2.13 :      0.68
NEURAL NET    :      1.0276 :      1.65 :      0.69
LU DECOMPOSITION :      43.296 :      2.24 :      1.62
=====ORIGINAL BYTEMARK RESULTS=====
INTEGER INDEX      : 4.252
FLOATING-POINT INDEX: 2.262
Baseline (MSDOS*)   : Pentium* 90, 256 KB L2-cache, Watcom* compiler 10.0
=====LINUX DATA BELOW=====
sh: line 1: uname: command not found
CPU                :
L2 Cache           :
OS                 :
C compiler          : gcc version 2.95.2 20000220 (Debian GNU/Linux)
libc                : ld-2.1.3.so
MEMORY INDEX       : 1.129
INTEGER INDEX      : 1.013
FLOATING-POINT INDEX: 1.255
Baseline (LINUX)    : AMD K6/233*, 512 KB L2-cache, gcc 2.7.2.3, libc-5.4.38
* Trademarks are property of their respective holder.
sh-2.05b# _
```

2. ábra

Most következik a vendég operációs rendszer installálása. Én személy szerint a parancssoros megoldást preferáltam, a következőt írtam be:

```
qemu-img.exe create -f qcow hda.img 3G
```

Ez egy „üres” image fájlt hoz létre, amibe majd a vendég operációs rendszerünk lesz installálva. A fájl neve jelen esetben hda.img, de ez szabadon változtatható, mint ahogy az is, hogy mekkora méretű legyen az image fájl, itt a 3G a 3GB-os méretet jelöli.

Az installálást kétféle módon lehet elvégezni, optikai meghajtóról és ISO image fájlból. A QEMU két meghajtót támogat alap esetben, egy Primary Master merevlemezt, és egy Secondary Master CD-ROM-ot, ami CD-ROM image betöltésére is alkalmas. Az optikai meghajtós módszer esetében be kell helyezni az operációs rendszer telepítő lemezét az optikai meghajtóba, majd a következőt kell megadni a parancssorban:

```
qemu.exe -L . -cdrom "D:" -hda hda.img -m 256 -boot d
```

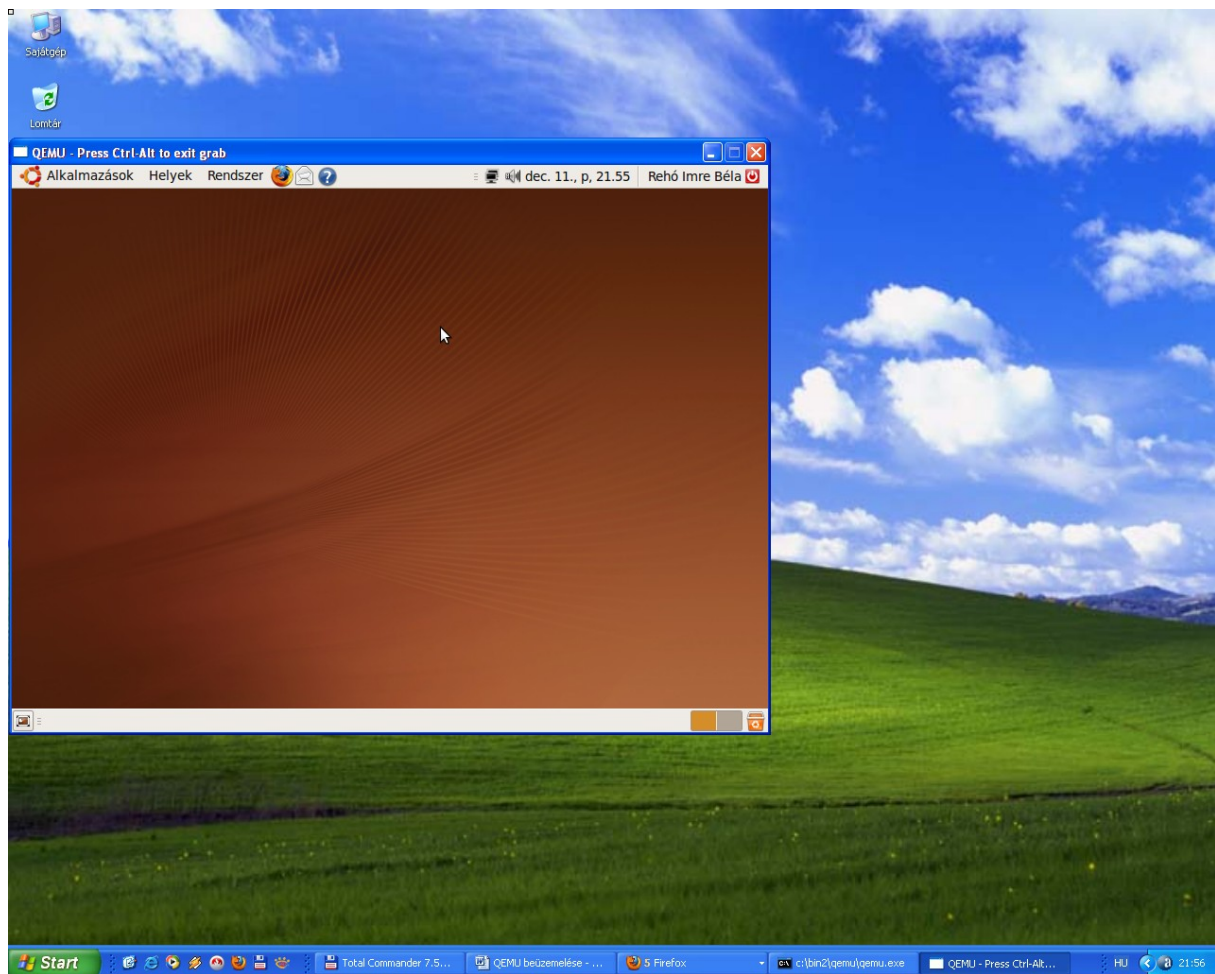
Feltételezzük, hogy az optikai meghajtó betűjele a D, a 256-os szám pedig a RAM méretére utal.

Az ISO image fájlból való installálásnál be kell másolni a QEMU könyvtárába az általunk telepíteni kívánt operációs rendszer ISO fájlját, majd a következőt kell megadni a parancssorban:

```
qemu.exe -L . -cdrom my_os_install.iso -hda hda.img -m 256 -boot d
```

Hasonlóan működik, mint az optikai meghajtós esetben. Itt a my_os_install.iso fájl az installálásra kijelölt ISO image fájl, értelemszerűen ez helyett azt kell megadni, ami a valódi neve.

A telepítés során úgy kell eljárni, mintha egy valódi gépre installálnánk fel az operációs rendszerünket. Furcsának tűnhet, hogy egy kisebb ablakban történik mindez. Sajnos én sem találtam meg a módját, hogyan lehetne teljes képernyőre állítani, de meg lehet szokni, számomra nem volt zavaró. Én egy Ubuntu Linux 9.04-es verziójú vendég operációs rendszert használtam. A telepítés és a használat során is azt vettem észre, hogy a QEMU mindent extrém lassúsággal végzett el. Ez azt jelenti, hogy a műveletek ideje többszörösére növekedett a szokásoshoz képest. Az is megfigyelhető volt, hogy a Linux-on belüli alkalmazások elindítása és futtatása is lassú volt, akadozott, úgy mint a billentyűzet általi karakterek bevitele és az egérmutató mozgása is. A következő képen látható, hogy egy Microsoft Windows XP SP2 rendszer alatt hogyan működik a QEMU, amiben éppen egy Ubuntu Linux 9.04-es operációs rendszerű virtuális gépet futtatunk.



3. ábra

Ezen a képen azért szerepel a Windows asztal teljes tartalma, mert így kívántam érzékeltetni vizuálisan azt a hatást, mikor azt mondjuk: „gép a gépben”. A másik oka pedig az, hogy ebben az esetben jól látható, hogy a QEMU ablakát sajnos nem lehet teljes képernyőre állítani. A lassúságra persze van orvosság is, amit egy `kqemu` nevű programmal oldottak meg, ami tulajdonképpen egy „QEMU accelerator”. Ez nagybán javít a sebességén a QEMU-nak, tapasztalható, hogy a működési sebesség majdnem eléri azt a fokozatot, mint normális esetben. A `kqemu` használatához egy azzal kompatibilis verziójú QEMU-t kell letölteni, és a `kqemu`-t a QEMU könyvtárába kell kitömöríteni. A `kqemu`-val való indításhoz ugyanúgy kell eljárni, mint anélkül, a különbség annyi, hogy a parancs végére egy „-kernel-kqemu” kiegészítést kell tenni.

Érdekességképpen megpróbáltam egy Microsoft Windows 7 terméket is QEMU-val emulálni, de az installálás során hiba történt, és a virtuális gép egy „kékhalál”-szerű képernyővel és némi hibaüzenettel kifagyott. Így sajnos ezt a példát nem tudom megmutatni, miként lehet egy Windows-t a Windows-ban emulálni. Fordított esetben is érdekes lehet a dolog, de a

Windows 7 már tartalmaz olyan részt, ahol a Windows XP virtualizálásával lehet futtatni programokat.

2.2 Linux környezetben

Linux környezetben nagyobb a támogatottsága a QEMU-nak, ugyanis léteznek olyan változatok, amelyek grafikus kezelőfelülettel rendelkeznek. A bemutatás az előző fejezetben szereplő Ubuntu Linux 9.04-es változatával fog történni, de természetesen ebben az esetben ez lesz a gazda operációs rendszer.

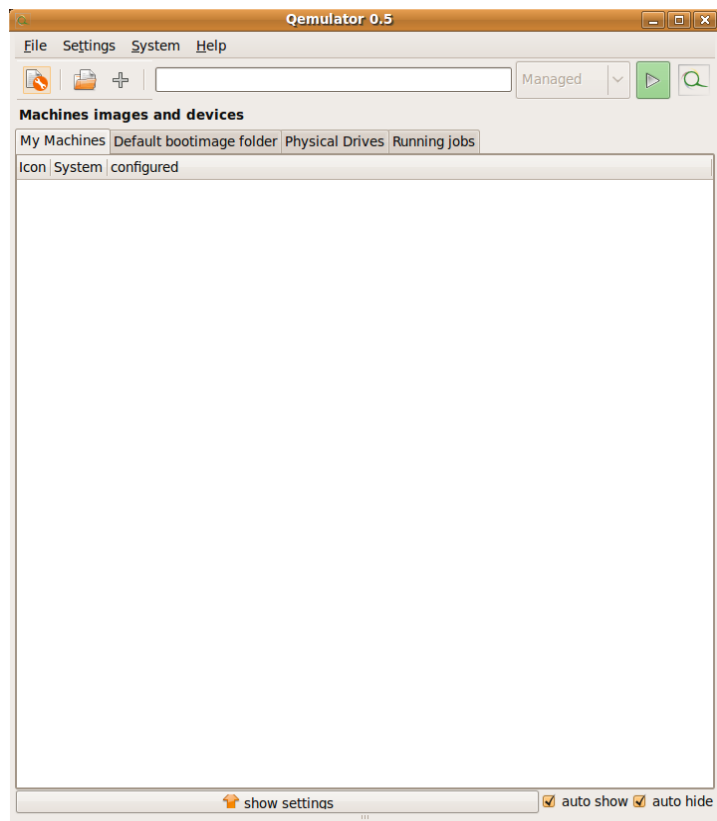
Ez esetben a QEMU felülete két részből tevődik össze: qemu launcher, qemulator.

A qemulator a könnyű telepítésekért és beállításokért felel, ezzel lehet képfájlokat és eszközöket használni, itt található a „gépeim listája”. A qemu launcher segítségével lehet létrehozni, menteni, futtatni több virtuális gépet egyaránt. Itt lehet létrehozni és konvertálni lemezképeket is.

A QEMU telepítése: `sudo apt-get install qemu qemulator`

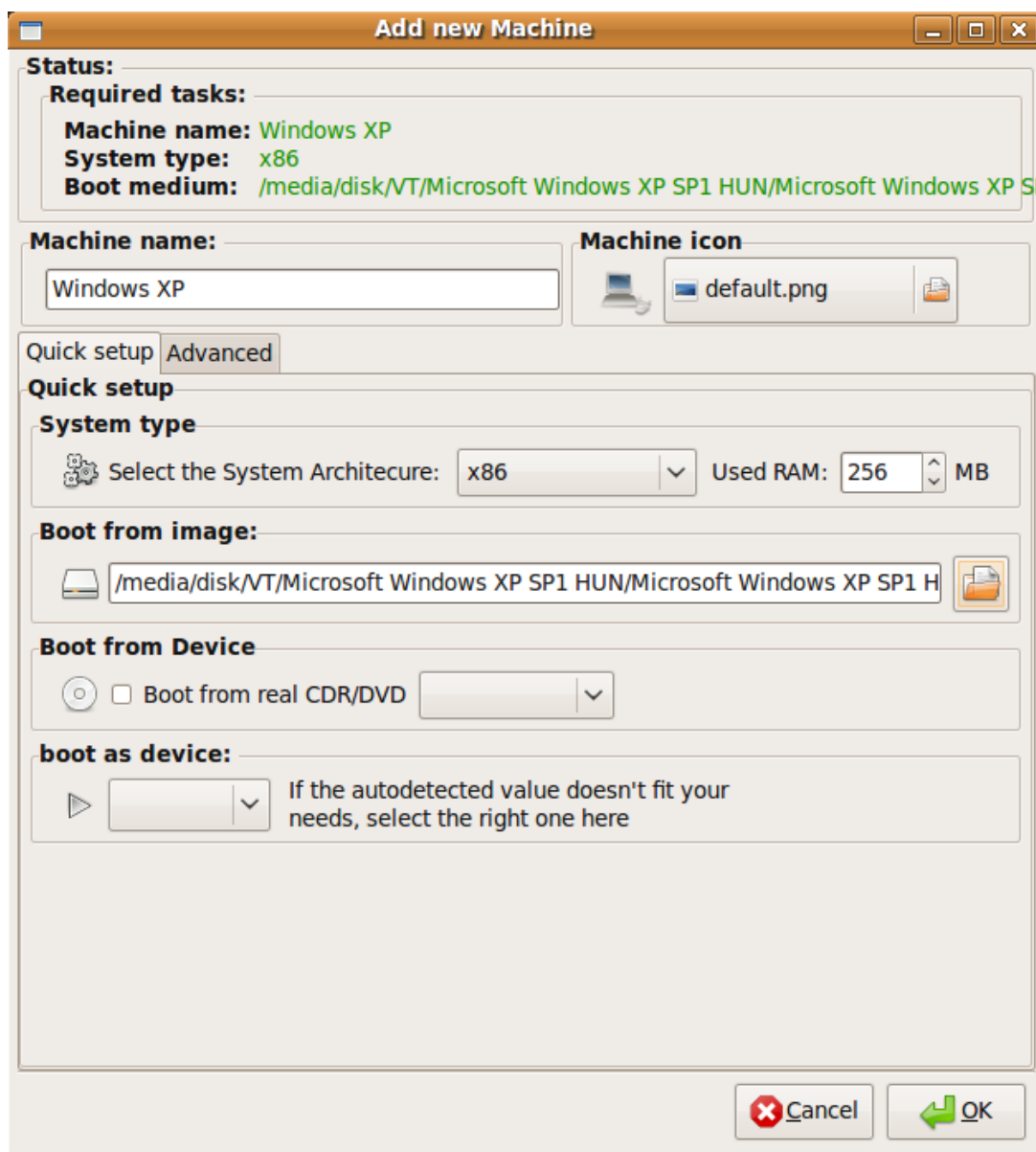
A qemu launcher telepítése: `sudo apt-get install qemu-launcher`

Természetesen a telepítésekhez internet-kapcsolatra van szükség. A qemulator-t az Alkalmazások-Eszközök-qemulator úton lehet elérni. Indítás után a következő ablak fog megjelenni:



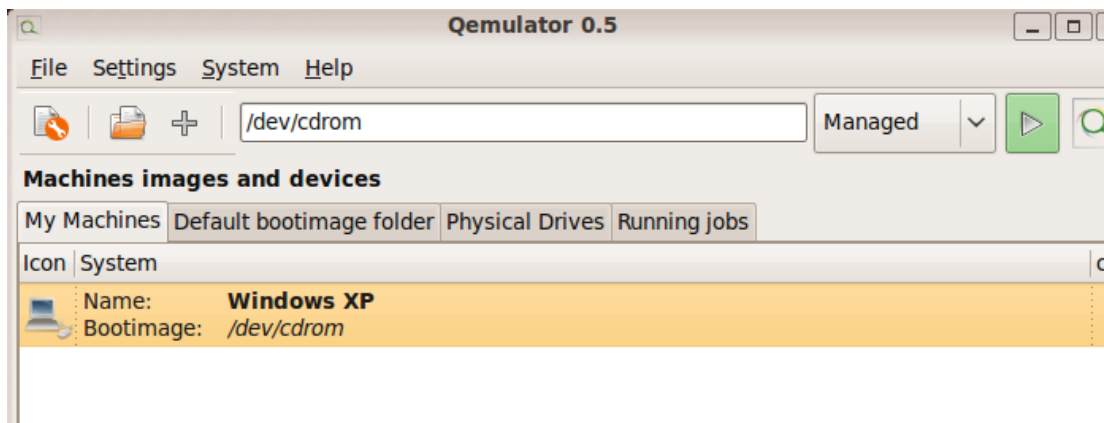
4. ábra

A Settings-Preferences menüben be kell állítani az elérési utat, ahol a QEMU képfájlok lesznek tárolva. Ha az optikai meghajtót is szeretnénk használni, azt is hozzá kell adni. Célszerű még az összes QEMU alkalmazást hozzáadni a különböző emulációs platformokból. A main window fülön a program néhány tulajdonságát lehet beállítani, főleg a megjelenítéssel kapcsolatosan. A control monitor fülön a control monitor viselkedése állítható be. A System-Create new machine menüpontban lehet új virtuális gépet hozzáadni a listához. Ezt meg lehet tenni a „+” jel kiválasztásával is. A megjelenő ablakban meg kell adjuk a virtuális gépünk nevét, a rendszer architektúrát és a felhasznált RAM méretét, valamint hogy a boot-olás honnan történjen, optikai meghajtóról vagy pedig image fájlból, utóbbinál a helyét is meg kell adni.



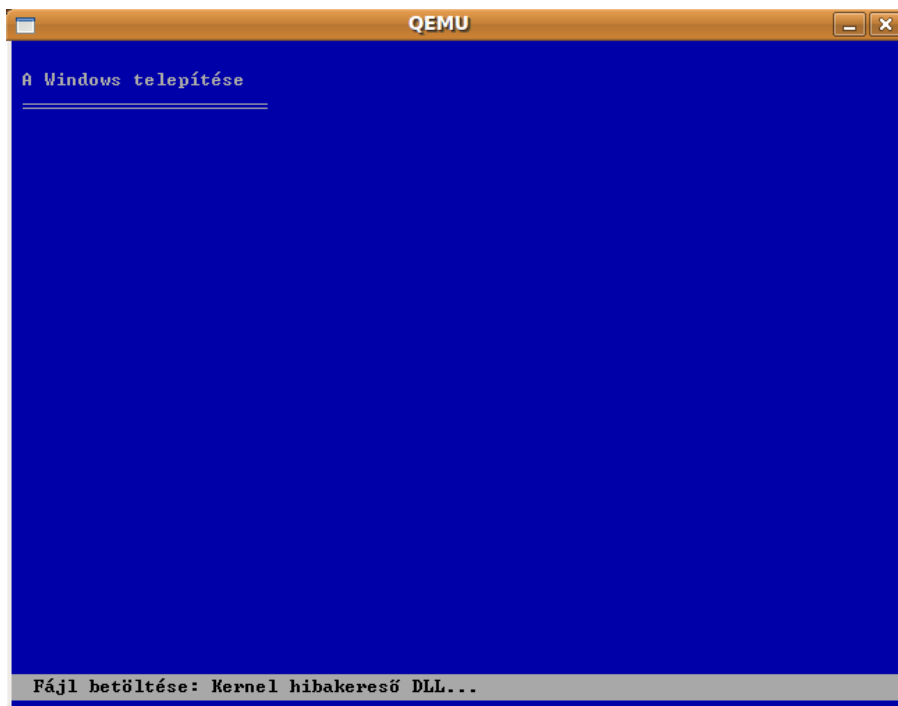
5. ábra

A paraméterek megadása és az OK gomb megnyomása után az elkészült virtuális gépünket viszontláthatjuk a listában, valamint a zöld nyíl kiválasztásával el tudjuk indítani.



6. ábra

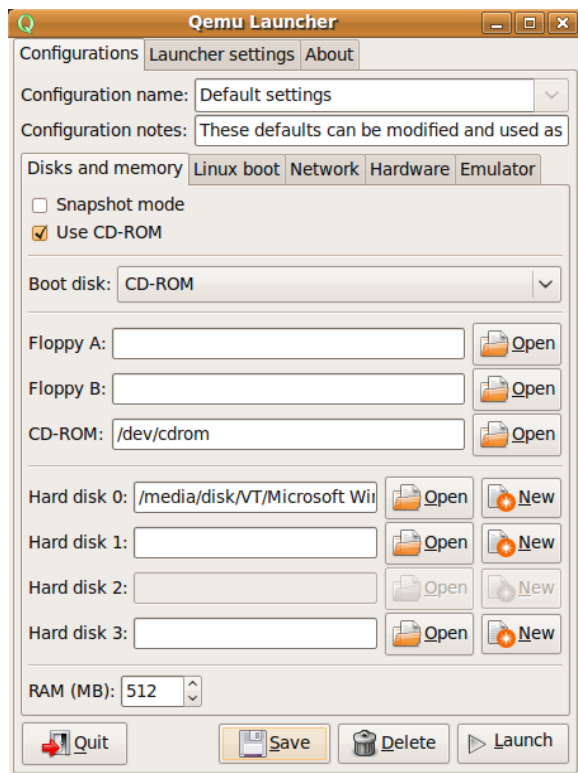
A virtuális gépünk egy külön ablakban fog megnyílni, amit az alábbi ábra szemléltet.



7. ábra

Mind minden QEMU-s esetben, itt is célszerű boot-olható képfájlokkal dolgozni. A legjobb, ha előre legyártottakat keresünk az interneten.

A qemu launcher hasonlóképpen működik, mint a qemuator, csak itt könnyebb a kezelhetőség.



8. ábra

3 Összefoglaló

A QEMU használatával kapcsolatban a következő összefüggések vonhatóak le:

Kis méretű

Könnyen kezelhető

Könnyen konfigurálható

Platformfüggetlen

Nyílt forráskódú

Viszonylag lassú, de gyorsítható

Egyszerű felépítése, de ebben rejlik az ereje

Más architektúráknál is használható

Néhány személyes tapasztalat a teljesítménnyel kapcsolatban:

Önmagában a QEMU nem egy hatékony virtualizációs program, de kiegészítőkkal sokkal tűrhetőbb. A kezelőfelülete nagyon egyszerű, könnyen áttekinthető, erre szokták mondani: „a kevesebb néha több”. Funkcióit tekintve számomra kielégítő volt, de ez egyéntől függ, kinek mire van szüksége. Teljesítmény szempontjából a sebessége nagyon is kifogásolható, hiszen egy Ubuntu Linux operációs rendszert feltelepítenie majdnem az örökkévalóságig tartott. Pontosan nem mértem, de kb. 12 óra időtartamba tellett neki. Ez sokszorososa a normálisnak, a hardverre nem tudtam gyanakodni, hiszen korszerű. Ezekhez a sebesség gondokhoz fűződik

az a gondolat, hogy kiegészítő szükséges a tűrhető használathoz. A qemu nagymértékben orvosolta a hibát, itt a sebesség már kb. csak 1,5-szerese a normális megszokotthoz képest. Operációs rendszert futtatva, működés közben is hasonló sebességet produkál pl. programok megnyitásánál, futtatásánál. Sokszor a billentyűzetnél a bevitt karakterek is késve jelennek meg a képernyőn, és az egér mutatója is tud akadozni, szaggatni, ami elég zavaró tud lenni.

4 Források

- [1.] <http://wiki.hup.hu/index.php/QEMU>
- [2.] <http://www.jox.hu/cikkek/167/12>
- [3.] <http://en.wikipedia.org/wiki/QEMU>
- [4.] <http://209.85.135.132/search?q=cache:8bGXUvguSGMJ:wiki.archlinux.org/index.php/QEMU+qemu+wiki&cd=2&hl=hu&ct=clnk&gl=hu>
- [5.] <http://qemu-buch.de/cgi-bin/moin.cgi/>
- [6.] <http://qemu-buch.de/cgi-bin/moin.cgi/QuickStartGuide>