

Colectia de drivere dispozitiv la Linux si Windows

Arama Dan, 432A

1. Generalitati

1.1 Notiuni introductive

Termenul driver denota un program care asigura functionarea corecta si controleaza un anumit tip de dispozitiv atasat la calculator. Rolul unui driver este de a realiza o legatura intre nivelul hardware si software. In alte cuvinte, driverul mediaza comunicarea dintre un dispozitiv anume si sistemul de operare sau aplicatiile care vor sa foloseasca dispozitivul. Driverul primeste comenzi de la programe si transmite aceste comenzi dispozitivelor, fiind in esenta o punte de legatura hardware-software. Aceasta functie permite programatorilor sa realizeze aplicatii de nivel inalt cu un caracter universal, care vor functiona indiferent de marca de hardware a fiecarui utilizator.

Exista drivere dispozitiv pentru o gama larga de dispozitive atasate calculatorului, datorita diversitatii componentelor hardware moderne si a sistemelor de operare. Dispozitivele care necesita drivere pot fi de intrare/iesire, precum mouse-ul, tastatura, imprimante, sau pot fi componente interne ale calculatorului, precum placa video sau placa de sunet. Exista si dispozitive care nu necesita ca un driver sa fie instalat de catre utilizator, deoarece folosesc un standard, integrat in sistemul de operare care sunt recunoscut si folosite in mod implicit. Astfel de componente sunt procesorul, RAM-ul sau majoritatea hard driver-urilor. Trebuie mentionat si mecanismul plug'n'play, care permite detectarea dispozitivelor noi introduse in calculator si activarea driverelor incluse in sistemul de operare pentru a permite utilizarea dispozitivelor, fara implicarea utilizatorului. Aceasta configurare automata a driverelor dispozitiv va avea loc doar daca BIOS-ul este compatibil cu plug'n'play.

Pentru majoritatea componentelor si dispozitivelor, inasa, este necesara instalarea unor drivere programate special pentru functionarea optima, ajungandu-se pana la o scadere in performante si chiar oprirea din functionare daca sunt folosite drivere inechite sau incompatibile. In cazul placilor video, de exemplu, desi cele ATI sau Nvidia indeplinesc aceeasi sarcina, ele o fac in moduri diferite, de aceea necesita drivere diferite. Driverele sunt specifice sistemului de operare, un driver scris pentru Linux nu poate fi folosit de catre Windows. Actualitatea driverelor este un alt factor de performanta, existand cateodata probleme in utilizarea unei componente al carei driver nu este actualizat de catre un program nou. Performanta si stabilitatea componentei hardware in cadrul sistemului de operarea sunt intr-o mare parte determinate de calitatea codului driverului, observandu-se diferentele dintre Linux si Windows: driverele Windows vor fi testate riguros, avand deci o fiabilitate crescuta, pe cand cele Linux vor fi disponibile mai repede.

In mediile Linux, driverele dispozitiv pot fi implementate ca parte a kernelului, dar pot de asemenea fi incarcate separat ca module. Fisierile .sys din Windows si modulele .ko din Linux detin drivere incarcabile, avantajul acestor tip de drivere fiind acela ca vor fi in memorie doar in perioada in care este necesar dispozitivul respectiv, in rest el nu va fi incarcat, eliberand astfel memoria in perioada de inactivitate.

1.2 Tipuri de drivere

Driverele generice sunt de doua tipuri, categorizate in functie de sursa lor, produse de catre firma producatoare de chip hardware, sau a sistemului de operare. Exista de asemenea si drivere virtuale, care simuleaza prezenta unei componente in cadrul sistemului de operare.

drivere de chip hardware

Drivererele codate de firma de chip hardware vor fi in general de o calitate superioara, acest lucru fiind datorat faptului ca programatorii firmei vor avea acces deplin la specificatiile tehnice. In principal, accesul se va pune pe performanta, stabilitate si compatibilitate cu sistemul de operare. Drivererele de acest tip vor functiona pentru orice componente care detin chipul hardware respectiv (de exemplu, drivererele Nvidia vor fi compatibile cu orice placa video avand chipset Geforce).

drivere de sistem de operare

Drivererele scrise de firma de sistem de operare vor permite functionarea tuturor dispozitivelor, si faciliteaza mecanismul de plug'n'play: in cazul in care un nou dispozitiv este introdus in calculator in timpul functionarii sau nu, sistemul de operare va cauta in colectia sa de drivere si va instala un driver corespunzator daca exista. Acest lucru va permite dispozitivul sa functioneze, chiar daca sub-optimal. Astfel, in programarea acestor drivere, accentul este pus nu pe performanta maxima, ci pe stabilitate si compatibilitate.

drivere virtuale

Un mediu de virtualizare, sau masina virtuala este un sistem de operare oaspete instalat in cadrul sistemului normal, numit sistem gazda. Drivererele virtuale sunt un tip particular de drivere. Ele sunt folosite pentru a emula anumite componente, de multe ori componente utilizate in cadrul unui mediu de virtualizare, ca de exemplu rulara unui program DOS in cadrul unui sistem pe care este instalat Windows. Drivererele virtuale au, paradoxal, un rol opus driverelor tipice, ele emuland o componenta reala hardware, astfel incat sistemul oaspete si toate driverele care functioneaza in cadrul masinii virtuale au doar iluzia ca acceseaza hardwareul real. Incercarile sistemului oaspete de a accesa hardwareul gazdei sunt rutate la driverele virtuale in cadrul masinii gazda ca subprograme.

Drivererele virtuale pot fi folosite si in afara masinilor virtuale. Doua exemplu de astfel de drivere virtuale sunt Hamachi, un simulator de retea privata, permitand realizarea unei retele LAN prin intermediu Internetului, si Daemon Tools, un program care permite montarea imaginilor de CD/DVD prin simularea unui CD/DVD drive.

2. Drivererele la Linux

Introducere

Linux este un sistem de operare open-source, distribuit gratuit, astfel ca unul dintre multele sale avantaje este ca mecanismul intern de functionare este vizibil publicului larg. Prin aceasta democratizare a sistemelor de operare Linux a permis unui numar mare de programatori, care detin cunostiintele necesare, sa poata examina, intelege si modifica codul sistemului. Astfel, actualizarea driverelor va fi realiza, in general, mai rapid decat in cazul sistemelor closed-sourced cum este Windows. Pe cand in cazul driverelor Windows vor exista teste riguroase si de durata, drivererele Linux vor fi disponibil utilizatorului mai frecvent si mai rapid, datorita ansamblului mare de programatori individuali care lucreaza in scopul realizarii lor. Dezavantajul acestei accesibilitati insa, este faptul ca drivererele realizate de astfel de programatori neafiliati oficial vor putea prezenta bug-uri, lacune in functionare, iar performanta in principal, dar si compatibilitatea si stabilitatea vor putea fi compromise.

In kernelul Linux, pe langa functia de interfatare software-hardware, drivererele de dispozitiv au si rolul de a actiona ca niste veritabile cutii negre. Ele faciliteaza transmiterea comenzilor de la Linux, sau programe din cadrul Linux, spre componenta, dar in acelasi timp mascheaza detaliile referitoare la functionarea dispozitivul fata de utilizator. Activitatea unui utilizator va fi gestionata printr-un set standardizat de apeluri (call-uri), independente de un driver anume. Driverul se va ocupa de maparea acestor apeluri la operatii specifice ale dispozitivului, legand astfel kernelul de dispozitivul propriu-zis.

Modularitate

In cazul dispozitivelor care nu au drivere instalate sau al caror driver nu este detectat corect, se pot folosi module kernel incarcabile. Un modul kernel este o bucata de cod care poate fi incarcata si descarcata din kernel, atunci cand este sau nu necesara, fara a fi necesara restartarea sistemului si implicit a reconstruirii kernelului, extinzand astfel functionalitatea Linux. Ele sunt create separat si apoi introduse in sistem, instalandu-se in directorul `/lib/modules/`. Aceasta modularitate face ca driverele Linux sa fie usor de scris, numarul driverelor disponibile va fi deci mai ridicat decat cel al driverelor Windows. Desi mii de dispozitive hardware sunt suportate de catre Linux, kernelul are o dimensiune mica deoarece doar un numar mic de drivere este incarcat in memorie, urmand ca alte module sa fie incarcate daca este nevoie de un anumit dispozitiv. Se pot viziona modulele care ruleaza in mod curent in cadrul sistemului de operare prin comanda `lsmod`, iar informatii despre un modul se afla prin comanda `modinfo`. Incarcarea si descarcarea modulelor se realizeaza prin comenzile `modprobe` si `insmod`, respectiv `rmmod`.

Drivere dispozitiv sunt privite in cadrul Linux ca niste procese independente, fiecare cu propriul spatiu de adresare intern, care poate fi extins pentru a cuprinde numeroase dispozitive simultan. Exista in fiserele de baza Linux, dll-uri (dynamic link libraries) care includ drivere pentru discheta, RAM si hard-disk, dar si alte dispozitive. Daca doua dispozitive incearca sa foloseasca acelasi tip de driver, acesta va fi incarcat din nou din dll-ul respectiv si va rule independenta fata de prima instanta.

Driverele ca fisiere

In cazul sistemelor bazate pe arhitectura Unix, accentul se pune intr-o mai mare masura decat la Windows pe gestionarea fisierelor. De aceea, perifericele vor fi considerate, prin abstractizare, tot niste fisiere. In acest sens, un exemplu este comanda `write()`. Aceasta comanda va scrie intr-un fisier, in mod normal, dar daca este scrisa in fisierul special `/dev/lp0`, comanda va fi trimisa la o imprimanta

la portul la care e asignat lp0. Un fisier de driver dispozitiv va avea 4 parametrii caracteristici, acestia fiind : numele, tipul (pentru dispozitive care lucreaza cu blocuri sau siruri de date), numarul major (care va identifica tipul dispozitivului controlat, cuprins intre 1 si 255) si numarul minor (diferentiaza dispozitivele care au aceeasi numar major). Acestia vor fi folositi ca parametrii pentru comanda mknod () pentru a genera un fisier device. Dev_t va fi un numar de 16 biti care va contine atat numarul major cat si cel minor, si va ajuta la identificarea cererilor facute de catre aplicatiile utilizatorului. Fisierele dispozitiv sunt incluse in directorul /dev in mod obisnuit, si pot fi asociate atat unui dispozitiv real, fizic, unei subunitatii al dispozitivului, asa cum /dev/hda1 este asignat primei partitii a unui hard-drive dar si unui dispozitiv existent doar in plan logic, precum /dev/null.

Denumirea dispozitivelor este tipizata, astfel, driverul unui hard-disk pe interfata IDE va fi /dev/hd, urmat de litere incepand cu a in functie de numar de ordine a detectarii dispozitivului (/dev/hda pentru primul dispozitiv). Un hard-disk pe interfata SCSI va avea driverul numit /dev/sd, urmat din nou de o litera in functie de ordine. Dispozitivele de stocare vor fi considerate hard-diskuri SCSI. Astfel va aparea deosebirea intre denumiri la Windows si Linux. De exemplu, la Windows, un CDROM s-ar vedea ca fiind I: si un hard disk C:, pe cand la Linux dispozitivele s-ar vedea /mnt/cdrom, respectiv /dev/hdb1/.

Porturile seriale

La Windows, porturile seriale au numele "COMx", x fiind numarul de ordine. La Linux, inasa, porturile seriale vor fi accesa prin intermediul driverelor dispozitiv, prin fisiere din interiorul directorului /dev, numite /dev/ttysx, x fiind numarul de ordine. Fisierele driver pentru porturi din /dev pot exista chiar daca nu exista porturi fizice, iar ele se pot lega virtual prin intermediul comenzii setserial. Pentru legatura fizica (domeniu de adrese si IRQ) se poate folosi un dispozitiv virtual /dev/modem. Spre deosebire de configurarea automata a porturilor de la Windows, la Linux, acest lucru va fi ceva mai complicat. Se va utiliza comanda dmesg pentru afisarea listelor IRQ si porturilor aferente. Controlul intreruperi (IRQ) va fi facut de sistemul de operare si nu de dispozitiv,

putand exista pana la 16 dispozitive care cer intreruperi simultan, fiecare cu prioritati diferite. Se pot viziona intreruperile curente in directorul `/proc/interrupts/`.

Driveri la Linux 2.6+

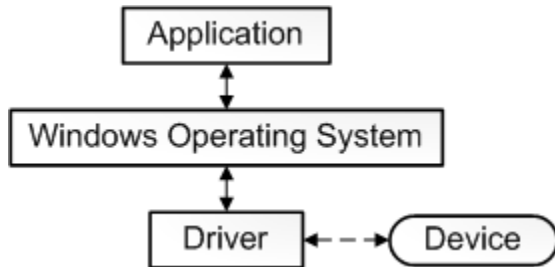
Incepand cu Linux 2.6, drivere dispozitiv incep sa depaseasca functiile de baza, primare ale versiunilor anterioare, incepand sa apara drivere cu rezultate similare celor de la Windows, iar programarea driverelor devine mai putin complicata, timpurile intre update-uri fiind reduse. Noile drivere introduc o multime de avantaje precum controlul asupra managementului puterii, mecanismul plug'n'play sau de hot-plugging. Driveri se gasesc nu doar in `/dev`, dar si in `/sys`, grupate in subdirectoare:

- `/sys/block` (lucreaza cu blocuri de date),
- `/sys/devices` (toate dispozitivele organizate dupa bus-ul la care se conecteaza),
- `/sys/bus` (busurile),
- `/sys/driver` (driverele inregistrate, incarcate momenta in memorie)
- `/sys/class` (clasele de dispozitive din sistem)
- `/sys/power` (dispozitive care executa managementul puterii)
- `/sys/firmware` (controleaza firmware-ul anumitor dispozitive)

Folosirea driverelor din directorul `/dev` este recomandata pentru o functionare optima a sistemului, deoarece directorul `/sys` nu trebuie modificat decat de utilizatori cu cunostiintele necesare in materie de Linux. Datorita numarului extins de dispozitive, nu se poate pretinde o compatibilitate maxima pentru orice combinatie driver/dispozitiv. Astfel se vor observa trei tipuri de nivel de compatibilitate: nesuportat, suport minim si suport extins, fiecare denotand nivel de interactiune posibil intre sistemul de operare si periferic. Linux are inclus, in mod implicit, un numar de drivere care controleaza anumite dispozitive utilizate uzual, acestea incluzand drivere pentru dispozitive Bluetooth, CD-ROM-uri, placi video pe AGP, joystick-uri, imprimanta HP300 si altele.

3. Drivererele la Windows

Introducere



Sistemele de operare Windows sunt sisteme closed-sourced, distribuite contra-cost. Avantajul major va fi deci fiabilitatea sistemului, acest avantaj extinzandu-se si asupra driverelor. Drivererele Windows sunt realizate si furnizate in mare masura de catre Microsoft, programatorii companiei fiind singurii cu acces la specificatiile complete ale sistemului. Astfel, drivererele Windows vor avea un termen mai lung de dezvoltare, perioada dintre lansari va fi mai mare, dar vor asigura un grad de stabilitate si compatibilitate mai ridicate.

O mare parte a driverelor necesare functionarii de baza a componentelor este inclusa in sistemele Windows, dar performanta obtinuta prin folosirea lor pentru interfatarea cu hardwareul nu va fi optima. Exista insa cazuri in care drivererele incluse in sistemul de operare sunt cea mai buna solutie, iar in cazul sistemelor Windows, pentru componentele cu o tehnica de functionare simpla, precum hard-diskul, mouseurile si tastaturile generice si altele, sunt de obicei folosite drivererele generice, fara fi necesara instalarea unui driver specific, acestea fiind suficiente pentru o functionarea corecta a componentei.

Sistemele Windows au inclus programul Windows Update. Acest program va cauta pe Internet drivere noi disponibile pentru componente calculatorului, si permite utilizatorului sa le downloadeze si instaleze intr-un mod foarte simplu. Exista de astfel si optiunea de actualizare automata a driverelor, eliminand necesitatea implicarii utilizatorului si asigurand o functionare optima a sistemului prin furnizarea celor mai recente drivere dispozitiv disponibile.

Creearea driverelor Windows

Windows Driver Foundation (WDF) este un model de dezvoltare a driverelor introdus de Windows, incluzand componente necesare pentru creerea si intretinerea driverelor modurilor kernel si utilizator. WDF faciliteaza creerea de obiecte, permitand programatorului de driver Windows sa se concentreze pe dispozitiv si nu pe sistemul de operare. Modelul WDF se bazeaza pe un model de driver incluzand scheletul driverelor modurilor utilizator si kernel, care va fi infrastructura de baza. Vor fi implementate astfel caracteristicile comune si de baza a driverului, care vor manevara operatiunile cu sistemul de operare.

Asemnator fisierelor driver de la Linux, Windows are un fisier numit `config.sys`, de configurare, care contine comenzi DOS, pe care le cauta in directorul radacina la incarcarea sistemului de operare. Acest fisier va efectua incarcarea driverelor la pornirea calculatorului si va schimba setarile sistemului de operare. O linie din fisier este de forma `DEVICE=C:\Windows\HIMEM.SYS`, aceasta avand o importanta deosebita, aceea de a incarca driverele in high memory, fara de care Windows nu ar incarca deloc driverele dispozitiv.

4. Concluzii

Driverele reprezinta modul de comunicare a softwareului cu componentele hardware. Ele pot fi scrise atat de producatorii componentelor hardware, cat si de producatorii sistemelor de operare si necesita un proces eficient de testare, datorita posibilitatii de provocare a unui crash de sistem in cazul erorii. Driverele sunt compatibile doar cu sistemul de operare pentru care au fost scrise, deci Windows si Linux prezinta colectii diferite de drivere. Atat pentru Linux, cat si pentru Windows, in cazul driverelor, se pune accentul pe stabilitate, performanta si compatibilitate cu sistemul pe care sunt instalate, toti acesti factori depinzand in buna parte de actualitatea driverului si calitatea codului.

Exista diferente intre driverele Windows si Linux, referitoare la modul de operare, modul de numire si de apelare, modularitate si programare, dar ele vor indeplini acelasi rol, de legatura intre software si hardware, chiar daca prin moduri diferite.

5. Bibliografie

1. http://simple.wikipedia.org/wiki/Device_driver
2. http://en.wikipedia.org/wiki/Device_driver
3. <http://ro.wikipedia.org/wiki/Driver>
4. http://en.wikipedia.org/wiki/Windows_Driver_Model
5. <http://lwn.net/Kernel/LDD3/>
6. <http://tldp.org/LDP/khg/HyperNews/get/devices/devices.html>
7. <http://msdn.microsoft.com/en-us/windows/hardware/gg581061>
8. <http://windows.microsoft.com/en-US/windows/>