

A Microwindows múltja, jelene és jövője



Eleinte csak szórakozásból terveztem, fejlesztettem a Microwindowst, mert imádom, amikor kis rendszerek nagy dolgokra képesek.

Greg Haerr a Century Software vezérigazgatója és a Microwindows Project alapítója. A projekt beágyazott rendszereken és eszközökön futó grafikus ablakkezelő rendszer fejlesztésével foglalkozik.

Rick: *Mikor találoztál először a Linuxszal, és mi késztetett arra, hogy grafikus felületet írj a Linuxhoz?*

Greg: A linuxos múltam 1993-ra nyúlik vissza, akkor használtam az egyik legelső i386-os terjesztést, az Yggdrisilt. Emlékszem, hogy a csomag egy hajlékonylemez és egy CD-t tartalmazott, ami újdonság volt abban az időben. UMSDOS fájlrendszerrel indult – ez lehetővé tette, hogy a meglévő DOS-os fájlrendszer szabad területét használva fusson a Linux –, elindította a grafikus felületet, megjelenített egy bejelentkezési képernyőt és lejátszotta a Star Trek zenéjét. A képernyő felbontását három beállítás között lehetett kapcsolgatni az ALT+ billentyűkombinációval... A Linux már a kezdetek kezdetétől megmutatta, hogy milyen jól tud alkalmazkodni a sokféle gépfelépítéshez, és azt, hogy a felhasználó segítségével (beállítások megváltoztatása) nélkül is meg tudja ezt tenni.

Több mint 25 éven át programoztam, vagyis jóval azelőtt kezdtem a programozást, mint hogy üzletember lettem. Annak ellenére, hogy hét éve követem a Linux fejlődését, csak az utóbbi két évben vettem részt néhány fontosabb fejlesztésben. Mindenképpen megérte.

Rick: *Miből gondolod, hogy a Linux jó választás a beágyazott rendszerekhez?*

Greg: Ennek több oka is van. Ezek alátámasztják azt a feltevést, hogy a Linux roppant gyorsan fog fejlődni a jövő évtizedben. Először is, a 32 bites processzorok ára végre elég alacsony, sebességük és áramfogyasztásuk is megfelelő ahhoz, hogy olcsó beágyazott rendszerekhez használhassuk őket, például tenyérgepekben, webpadokban és egyéb érintőképernyős eszközökben. A Linux szinte a kezdetek kezdetétől több processzortípust támogatott, például a fejlett RISC processzorokat is. Így a Linux és a beágyazott rendszerek közös útja szinte természetesnek vehető. Másodsorban a programfejlesztők szeretnék átültetni az asztali gépeken már nagy sikert aratott felületet hordozhatóbb eszközökre. A vezeték nélküli hálózatos eszközökben sem kell újra feltalálni a spanyolviaszt, amikor olyan sok közkedvelt alkalmazás is felhasználható már. Végül a Linux, mivel ingyenes és nyílt forrású, lehetővé teszi, hogy a gyártók tovább mérsékeljék a költségeket, a legjobb megvalósítást lehetővé tevő fejlesztések és eljárások megosztásával, használatával. Ez egyaránt nyereség a gyártó és a fogyasztó számára.

Rick: *Hogyan viszonyul a Linux a beágyazott rendszerekben használatos „hagyományos” operációs rendszerekhez? Olyanokra gondolok, mint a VxWorks, a pSOS, vagy a VRTX.*

Greg: A Linux legnagyobb előnye a fejlesztési szakaszban mutatkozik meg, ugyanis a fejlesztők ugyanazt az operációs rendszert használják az asztali gépükön a fejlesztés során, mint ami azon az eszközön van, amire az alkalmazást szánják. Ez azt jelenti, hogy annyira tisztában vannak a céleszköz lehetőségeivel, mint amennyire az asztali gépükével. Például a Century Software új, a Compaq iPAQ-hoz készített Microwindows operációs környezet és fejlesztői eszköz esetében teljes egészében elő tudtuk állítani a célkörnyezet pontos mását, ezért a program fejlesztése az eszköz fejlesztésével egy időben folytatható. Vannak természetesen más előnyök is, főleg azok számára, akik már befektettek a unixos megoldásokba, legyen Sun, DEC, HP vagy IBM a választás. Ha ugyanis fejlesztéssel valamilyen alkalmazást unixos környezethez, és ki akarsz használni az új, hordozható eszközök nyújtotta lehetőségeket, kimondhatatlanul könnyebb helyzetben vagy, ha az alkalmazásodat Linuxra kell átültetni, mintha ugyanazt Windows CE-re kellene megtenni. És természetesen, mivel a webkiszolgálók többsége Linuxot futtat, az ügyféloldalon is megjelennek a linuxos alkalmazások.

Rick: *Miért fogtál bele a Microwindows fejlesztésébe?*

Greg: Eleinte csak szórakozásból terveztem, fejlesztettem a Microwindowst, mert imádom, amikor kis rendszerek nagy dolgokra képesek. Az egész azonban gyorsan felfutott, a fejlesztést a felhasználói igényekhez kellett igazítani, számos olyan lehetőséget is beépítettem, melyeket addig csak a fejlettebb rendszereken lehetett elérni. A Microwindows támogatja például az élsimított szöveget, a TrueType és Adobe Type 1 betűtípusokat és a külön csatornára alapozott átlátszóságot (alpha blending), ezeket például az X is csak most kezdte támogatni. Ha a Microwindows méretét összehasonlítjuk az X és a hozzá tartozó programkönyvtárak méretével, a különbség óriási a Microwindows javára. Az X fejlesztői keményen dolgoznak azon, hogy a rendszer méretét csökkentésük. A két rendszer összetettsége között azonban mindig óriási lesz a különbség. A beágyazott rendszerek esetében mindig igyekszünk a lehető legjobban kihasználni a gépet, mivel elsősorban ezért hozzuk létre a készüléket. Ez azt jelenti, hogy amikor a grafika gyorsításához ki akarsz használni a gép lehetőségeit, vagy amikor a programnak le kell kezelnie valamilyen furcsa érintőképernyős külső egységet, egy nagyságrenddel könnyebb dolgod van a Microwindows esetében. Emellett a Microwindows Nano-X protokollja nagymértékben hasonlít az X-hez, a különbség a színkezelésben van, ezért a Microwindows alá könnyebb alkalmazást fejleszteni.

Rick: *Milyen más megoldásról tudsz, ami esetleg versenyezhet a Microwindows rendszerrel?*

Greg: Valószínűleg két nagy vetélytársunk van, már ha azokat a cégeket nézzük, amelyek valójában létrehozzák az újításokat ezen a területen. A TrollTech mellett ott van a Javában megvalósított PocketLinux is. A Century, a TrollTech és a Transvirtual tekinthetők vetélytársainknak is, bár merőben másképp próbálják az alkalmazásaikat a piacra dobni, mindegyikük más területre összpontosít.



A Transvirtual PocketLinuxa egy aranyos Javaás megvalósítás, ami fut framebufferen, viszonylag kevés alkalmazás érhető el hozzá, de ezek az alkalmazások minden felhasználási területet lefednek. Ez mind szép, ha van Javád. A PocketLinuxon nem futtatható olyan alkalmazás, amit nem Javában írtak meg, tehát nem alkalmas általános célokra. A TrollTech Qt/Embedded rendszere hasonló helyzetben van, de népebb felhasználótábor számára jelenthet megoldást. A Qt/Embedded projekt célja, a Qt könyvtárral egyetemben, egy Windowshoz hasonló keretrendszer létrehozása az alkalmazások számára. A Qt-s alkalmazásokat pillanatok alatt át lehet ültetni egy framebufferes beágyazott eszközre. Ez nagyon szép, de a Qt/Embedded alatt csak Qt-s alkalmazások futtathatók. Ez sok kódolást és egy összetett rendszer kezelését jelenti. A Microwindows a leggyorsabb mind közül, és kiválóan alkalmas általános alkalmazásfejlesztéshez, itt nem lehet arra számítani, hogy az alkalmazások kinézete megegyezik, hogy ugyanazt az elemkészletet használják, vagy ugyanazt a Java-megvalósítást. Ez az általános környezet jól jön, ha általános alkalmazásfejlesztésről van szó, ahol sok külső cég által készített terméket kell használni, vagy ha több eltérő eljárást kell alkalmazni együtt, az API vagy a fejlesztéshez felhasználható nyelvek korlátozása nélkül.

Rick: *Útnak indítottál egy böngésző projektet is, a ViewML-t. Milyen más lehetőségek vannak a ViewML mellett, és hogyan viszonyulnak ezek az általad fejlesztett programhoz?*

Greg: Természetesen elfogult vagyok, mert a ViewML a Century Software nyílt forráskódú beágyazott böngészője. A ViewML elfut 800 k ROM-mal és 2,1 MB RAM-mal, és kiválóan működik a kis eszközökkel. Eléggé korlátozottak a képességei, a kevés memória határokat szab. Nemrégiben, Ausztráliában egy lelkes csapat átültette a teljes Mozilla böngészőt Microwindowsra, és jól működik. Tehát az igényektől függően elérhető a kis méret és a teljes kiépítésű böngésző is, kinek mire van szüksége. Az Opera nagyon jól vizsgálja HTML-megfelelőségéből, és létezik már egy még ki nem adott változat Microwindowsra. A Microwindows munkakörnyezet szempontjából az volt a fontos, hogy a ViewML jól és gyorsan fusson a kis PDA eszközökön és a nagyobb gyorsabbakon egyaránt. Ha azonban nem felel meg az ügyfél számára és van elég memóriamodul, még mindig felteheti a Mozillát.

Rick: *Milyen kihívásokkal találkozottál a ViewML fejlesztése közben? Hogyan léptél túl a nehézségeken? Ki kellett hagynod bizonyos tulajdonságokat a ViewML-ből?*

Greg: A lehető legkisebb méret kulcsfontosságú volt, mint az is, hogy a böngésző minden lapot helyesen tudjon megjeleníteni. A KDE csapat kfm KHTML elemére esett a választás. A ViewML projekt létrehozott egy beágyazott böngészőt, úgy, hogy közben a KHTML kódjában egy sort sem változtattunk meg, így nem nagyon tudtuk elrontani. Írtunk egy QT→FLTK osztályátalkító réteget, ami lehetővé tette, hogy az egész grafikus felületet és az elemek megjelenő vezérlőket kevesebb, mint 100 k-ban valósítsuk meg. A végeredmény a ViewML. A legtöbb

munkát az átalakítórétég véglegesítése jelentette. Azt is mindig szem előtt kellett tartanunk, hogy a böngészőnek elég gyorsnak kell lennie a mindennapi használatához. Még mindig dolgozunk a TrueType betűtípusok megjelenítésének tökéletesítésén.

A jövőre nézve, azt tervezzük, hogy felhasználjuk a KDE Konqueror v2.0 html programelemet, ami támogatja a HTML 4.0-t és a JavaScript 1.4-et. Ennek a megvalósítását is egy átalakítórétégen keresztül tervezzük.

Rick: *Visszatekintve, milyen volt vezetni egy nyílt forráskódú projektet?*

Greg: A Microwindows Projekt biztosan nem lenne az, ami ma, ha nem használhatott volna fel számos nyílt fejlesztést, mint például a méretezhető betűmegjelenítés támogatását, ha nem lett volna lehetőség a kipróbálásra és a hibakeresésre számtalan rendszeren, és ha nem tudtuk volna megvitatni a kulcsfontosságú kérdéseket a levelezőlistán. Nagyon élveztem, hogy vezethettem ezt a fejlesztést. A Nyílt Forráskód Közössége sokat segített azzal, hogy a megjelent terméket, majd az ahhoz megjelent további alkalmazásokat azonnal kipróbálták. Hidd el, ha valami nem működne jól, arról tudnék...

Rick: *Milyen új bővítéseket vagy fejlesztéseket tervezel a Microwindowsba?*

Greg: Sokkal több, a piacon már jelen lévő vagy megjeleni készülő PDA-ra lesz elérhető a Microwindows ingyenes bináris terjesztése. Ez felkelti majd a fejlesztők érdeklődését, hiszen még több hasznot húzhatnak abból, hogy az alkalmazásuk több felületen futhat. Nagyon érdekesnek ígérkezik a megjelenésünk a webpadok területén. Egy olyan felépítés áll rendelkezésünkre, amely lehetővé teszi, hogy a fejlesztők és a felhasználók ugyanazokat a grafikus alkalmazásokat használják számos lapos képernyős eszközön. Az alkalmazásnak más lehet a kinézete, attól függően, hogy 240x320-as vagy 600x800-as felbontásban használják, vagy hogy két-, háromdimenziós vagy tévészerű vezérlőket valósítanak meg a fejlesztői könyvtárak.



Rick Lehrbaum

(rick@linuxdevices.com) hozta létre a LinuxDevices.com „beágyazott Linuxok portálját”, ez nemrég tagja lett a ZDNet Linux Resource Centernek. Rick 1979 óta beágyazott rendszerek fejlesztésével foglalkozik. Társalapítója az Ampro Computersnek, alapító tagja a PC/104 Consortiumnak, és fontos szerepet játszott abban, hogy elindulhatott az Embedded Linux Consortium.

Rick: *A Linuxszal ellentétben a Microwindowsra nem a GNU General Public License (GPL) vonatkozik. Szólnál pár szót a Microwindows felhasználási feltételeiről?*

Greg: A Microwindows első fejlesztőivel összhangban úgy döntöttünk, hogy a termék felhasználását az MPL fogja szabályozni, ami kevesebb megkötéssel jár, mint a GPL. Ez azt jelenti, hogy a Microwindowst fel lehet használni zárt forráskódú fejlesztésben is, vagy olyan esetben, ha a meghajtó forráskódja nem tehető közzé. Lehet, hogy ez ellenkezik a szigorú értelemben vett szabad program szellemével, de én inkább gyakorlati szempontból közelítem meg a dolgot. Az az igazi küldetésem, hogy elősegítsem a grafikus felületű beágyazott eszközök elterjedését, ezért olyan rendszert kellett fejleszteni, aminek engedékenyebb a felhasználási szerződése.