



Többfelületes programozás wxWindowszal

A jól bevált wxWindows a most készülő Chandler csoportmunkaprogram eszközkészlete.

A többfelületes fejlesztés az utóbbi években minden napos kifejezéssé vált – Redmond bugyraiban éppúgy, mint a nyílt forrású kezdeményezések háza táján. Használata szinte mindig a Java, .NET, a V vagy a Qt körül forog, de ne feledkezzünk meg a wxWindowsról sem. Akár MFC-kódot akarunk Linuxra átültetni, akár egyszerűen csak a lehető legszélesebb közönség számára akarunk egy programot elérhetővé tenni, a wxWindows megállja a helyét. Furcsa módon úgy tűnik, hogy még nem sokan hallottak róla, de írásunk remélhetőleg segít abban, hogy szélesebb körben is megismerjék.

Miért választanánk a wxWindowst a fenti fejlesztőeszközök helyett? Jogos a kérdés, és minden egyes eszköz esetében más és más a válasz.

Összehasonlítás más eszközökkel

Bár a KDE-programok készítéséhez a Qt a szabványos eszköz, a wxWindows szintén használható. A Qt windowsos változata jogdíjas programok készítéséhez nem használható fel szabadon, de a wxWindows igen, és a Qt az eseménykezelő rendszerhez különleges előfeldolgozót igényel.

A Microsoft .NET és az Ximian Mono rendszere éppúgy, mint a Java, nem annyira eszközkészlet, inkább technológia. Ezek a megoldások újabb réteget adnak a programokhoz, ami a teljesítmény rovására mehet. Erről a kérdésről a „fejlesztési idő vagy futásidő” ellentét alapján is vitát lehetne nyitni, amit a program életciklusa befolyásol a leginkább. Ahelyett, hogy belebonyolódnunk a kifejezések körüli csatározásokba, csak annyit mondunk: „mindenki másképp csinálja”.

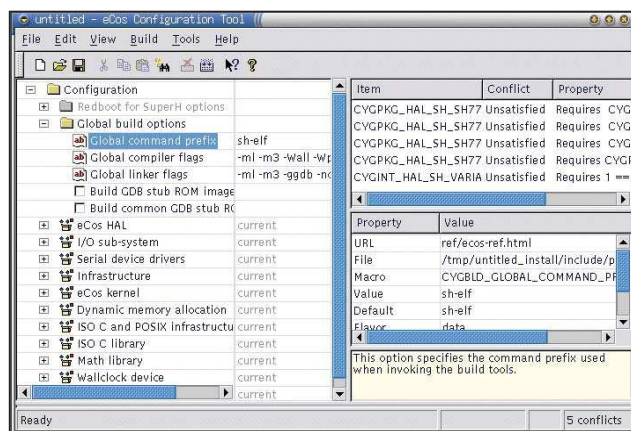
A V pedig egyszerűen nem olyan rugalmas, mint a wxWindows; nem támogat annyi fordítót. Még mielőtt a wxWindowst ezekkel a többfelületes fejlesztőeszközökkel igazán összevetnénk, fontos megemlítenünk, hogy a wxWindowsnak mindenféle összehasonlítás nélkül is megvannak a maga erősségei.

A wxWindowszal megkapjuk a nyílt forrás megszokott előnyeit – jogdíj fizetése nélkül felhasználható, szabadon módosítható, továbbá megszabadulunk attól a kockázattól, hogy a terjesztő cég megszűnik (vagy olyan módon frissíti a programkönyvtárait vagy a fejlesztőeszközöket, hogy kénytelenek legyünk megvenni az új változatot). Ezenkívül a programot a legkülönbözőbb operációs rendszereken is szabadon futtathatjuk, például Unix/Linux, Windows, Mac OS 9, Mac OS X és OS2 környezetben. Ráadásul minden rendszeren egységes megjelenést ad. A wxWindows felhasználási szerződése rugalmas, és a Nyílt Forrás Kezdeményezés (Open Source Initiative) is jóváhagyta.

Körülbelül 11 évi fejlesztéssel a háta mögött a wxWindows üzembiztos, kiforrott fejlesztői programkönyvtár, amihez a <http://wxWindows.org> honlapon elérhető levelezőlisták valós idejű támogatást nyújtanak. A több mint háromszáz osztály és ötezer függvény nagyszámú szolgáltatást tesz elérhetővé. Ezzel a könyvtárral az MFC programok átültetése is egyszerű feladat.

wxWindowst használó programok

Olyan vezető cégek is használják a wxWindowst, mint az AMD, az Intel Graphics Lab, a Compaq Alpha processor fejlesztőcsoportja, a Netscape és a Lockheed Martin. Az Open Source Applications Foundation az új, Chandler névre keresztelt, személyes adatok kezelésére és csoportmunka támogatására szolgáló programjában használja (wxPython segítségével).



A Helpblocks (<http://www.helpblocks.com>) programmal különböző felületekre készíthetők sűgőállományok, és mind Windows-, mind Linux-rendszere a wxWindows könyvtárral fejlesztik. Többek között wxWindowsos alkalmazás még a Vulcan 3D (modellezőprogram bányaipari felhasználásra, de még több is ennél), a SciTech Display Doctor programja, az Intuitive MX (egy többsávú hangkeverő, ami 3D-ben jeleníti meg a keverési folyamatot – az Intuitive Works fejleszté), valamint a Ground Control Station a Geneva Aerospace-től. Akár a te programod lehet a következő a listán!

Linuxos fejlesztés wxWindowszal

Szükségünk lesz a gcc-re, és olyan eszközöket használhatunk, mint a gdb, a ddd, az Emacs, valamint olyan egységes fejlesztőkörnyezetet, mint az Anjuta. Ha Windowsra is fordítunk, egy keresztfordítóra is szükség lehet. Legfőképpen pedig szükségünk lesz magára a *wxWindows* könyvtárra, ami szintén megtalálható a wxWindows-honlapon vagy megrendelhető CD-n is. A cikk írásának idején a legfrissebb változat a 2.4.0-s volt. (jelenleg a legfrissebb megbízható változat a 2.4.1-es, ami megtalálható az 51. CD Magazin/wxWindows könyvtárában). Sokunk jó szokásaival ellentétben nem árt elolvasni a csomagban található leírás linuxos fejlesztésre vonatkozó részét. Nem árt feliratkozni a *wx-user* levelezési listára sem (a <http://wxWindows.org> lapon a *Support* alatt találjuk), hátha nem megy minden könnyűszerrel. Legokosabb, ha a kérdéseinkre rákeresünk az archívumban, hátha föltette már őket más is.

Maga a könyvtár több mint hetven példaprogramot tartalmaz,

Egy wxApp osztály létrehozása

```
#include "wx/wx.h"
#include "mondrian.xpm"

// Új alkalmazásosztály (wxApp)
// meghatározása
class MyApp : public wxApp {
public:
    // kezdőérték-adás
    virtual bool OnInit();
};

// Új kerettípus meghatározása fő keretként
class MyFrame : public wxFrame {
public:
    MyFrame(const wxString& title,
            const wxPoint& pos,
            const wxSize& size,
            long style = wxDEFAULT_FRAME_STYLE);

// Eseménykezelők (nem virtuális!)
void OnQuit(wxCommandEvent& event);
void OnAbout(wxCommandEvent& event);

private:

    // Eseménytábla bevezetése
    DECLARE_EVENT_TABLE()
};

// Új alkalmazásobjektum létrehozása
IMPLEMENT_APP(MyApp)

// 'Main App' megfelelője
bool MyApp::OnInit()
{
    // Fő alkalmazásablak létrehozása
    MyFrame *frame =
        new MyFrame(_T("Example"),
                    wxPoint(50, 50), wxSize(450,
340));
    frame->Show(TRUE);
    return TRUE;
}
```

amelyek szinte mindent bemutatnak, amire a könyvtárat valaha is eszünkbe juthat használni – valószínűleg ez a legjobb leírás, mert így működő programokkal kísérletezgetve figyelhetjük meg, hogyan is rakták őket össze. Ha a könyvtárat beszereztük, és a megszokott fejlesztőeszközeinkkel már kezelni tudjuk, kezdődhet a programozás.

Egy-két példa

Egyszerű alkalmazásosztályt létrehozni nem nehéz. *Listánkon* egy wxApp osztály létrehozására láthatunk példát. Amint látható, könnyen hozhatunk létre új wxWindows alkalmazást, és az MFC-t közelebről ismerő olvasók talán elgondolkozva dörzsölik az állukat. Örömmel vehetik tudomásul, hogy nincs benne MFC, soha nem is lesz, de az osztályok egyszerűen átalakíthatók MFC-ből. Akad benne néhány dolog,

ami az MFC-ben nincs, és nem támogatja az OLE-t. *Listánkon* egy üres eseménykezelő is látható, ami olyan események kezelését teszi lehetővé, mint az egérgattintás, a billentyűleütések vagy a saját egyedi események. Ez a kód egy eléggé unalmas programot ír le, ami csak csücsül magában és meg se mukkan.

A könyvtár tele van működő példákkal – mintha minden osztályhoz lenne egy példadémon, ami bemutatja a működését. Az egyik az eseménykezelést bemutató példa, ami *Vadim Zeitlin* munkája. Azért választottam ezt, mert a legtöbb wxWindowszal ismerkedő fejlesztőnek gondja akad az eseményekkel – az olvasó tehát egy kis előnyre tehet szert. Kipróbálásához szükség lesz a *wxWindows* könyvtárra, és ha ez megvan, akkor a teljes forráskódja is megtalálható a *Samples* könyvtárban.

A honlapon még egy Wikire mutató hivatkozás is fellelhető, úgyhogy könnyen elérhető a legfrissebb leírás, és még bele is javíthat, akinek hozzáfűznivalója akad.

Adjuk meg a fejlesztőknek, ami jár!

A több ezer hozzájáruló mellett a fejlesztőcsapat magja külön említést és köszönetet érdemel:

- *Julian Smart* ➔ <http://www.anthemion.co.uk>
- *Robin Dunn* ➔ <http://www.python.org>
- *Vadim Zeitlin* ➔ <http://www.tt-solutions.com>
- *Stefan Csomor* ➔ <http://www.advanced.ch>
- *Vaclav Slavik* ➔ <http://sourceforge.net/users/vaclavslavik>
- *Robert Roebling* ➔ <http://www.roebling.de>

Ők a *xw-user* levelezőlistán gyakran napi, ha nem óránkénti rendszerességgel nyújtanak segítséget a wxWindows könyvtár felhasználóinak.

A wxWindows jövője

A 2.4.0-s változat kiadását követően a fejlesztőcsapat megadta azoknak a szolgáltatásoknak a listáját, amiket a 3.0-s változatban látni szeretnének. A támogatott felületek köre folyamatosan bővül, ez a következő kiadásban sem lesz másképp.

Az egyik új felület *Marco Cavallini* és Robert Roebling munkájának köszönhetően feltehetőleg a Windows CE lesz.

Készül a Winelib támogatás is a Winelib-csapat gondozásában, Julian Smart és Stefan Csomor pedig az MFC-ről wxWindows-ra átültetést segítő jogdíjas eszközöket fejleszt.

Mivel egyre több cég használ Linuxot, MFC-kódjukat szeretnék Linuxra átültetni, és ennek egyik legegyszerűbb eszköze a *wxWindows* könyvtár. Már csak a kedvező költségek miatt is megéri. Az átállással járó fejlesztési gondokat a könyvtár egyszerű felépítése a lehető legkevesebbre csökkenti, ezért a Windowsról Linuxra történő átállást tervező cégek kódjuk egyszerű átvitelére számíthatnak. Még wxNET felület fejlesztése is folyamatban van.

Lehet, hogy neked is van mit hozzátenned a wxWindowszal megvalósított programok hosszú sorához. A wxWindows programokról naprakész lista található a ➔ <http://wxWindows.org> honlapon.

Linux Journal 2003. július, 111. szám



Taran Rampersad (➔ <http://KnowProSE.com>)

14 éves tapasztalattal rendelkező programfejlesztő; jelenleg tanácsadással és fejlesztéssel foglalkozik. Trinidad és Tobagoról ír. Részt vesz a helyi számítógépes társaságokban, és segít a Caribbean FLOS Konferencia (➔ <http://floscaribbean.org>) szervezésében.

© Kiskapu Kft. Minden jog fenntartva