



Programmeren onder Unix met BASIC



door John Perr
<johnperr@linuxfocus.org>

Over de auteur:

Hij gebruikt Linux sinds 1994, en is één van de Franse LinuxFocus editoren.

Kort:

Is het mogelijk om programma's in BASIC te ontwikkelen met Linux, of een ander unixstelsel? Waarom niet? Verschillende vrije oplossingen laten ons toe om geïnterpreteerde of gecompileerde toepassingen te ontwikkelen door gebruik te maken van de BASIC taal.

*Vertaald naar het
Nederlands door:*
Cyberprophet
<cyberprophet@linux.be>

Inleiding

BASIC verspreidde zich - al leek het later op het computertoneel verschenen te zijn dan andere talen - snel over vele niet Unixsystemen als een vervanging voor de scriptttalen waarvan de oorsprong in unixomgevingen te vinden was. En dat is waarschijnlijk de hoofdreden waarom deze taal zo zelden door Unix-adepten gebruikt wordt. Unix heeft al vanaf dag één een veel krachtigere scripttaal. Net zoals andere scripttalen wordt BASIC over het algemeen geïnterpreteerd en heeft hij een simpele schrijfwijze, zonder datatypen, buiten een onderscheid tussen nummers en strings. Zijn naam is kenmerkend voor zijn eenvoud en daar het een taal betreft die zich uitstekend verleende om studenten de programmeerkunst aan te leren.

Helaas leidde het gebrek aan standaardisatie ertoe dat er vele verschillende versies opdoken, die in het algemeen niet met elkaar compatibel waren. We durven zelfs te zeggen dat er zoveel versies als interpreters zijn, wat BASIC zo goed als niet overdraagbaar maakt.

Ondanks deze en vele andere nadelen die door de "ware programmeurs" steeds weer ter herinnering naar voren worden geschoven, blijft BASIC een optie om in beschouwing te laten om snel kleine programma's te ontwikkelen. Dit is al sinds vele jaren zeker waar door de geïntegreerde ontwikkelingsomgeving die gevonden kon worden in Windowsversies, die het toelaten in enkele muisklikken een grafische interface te ontwerpen. Verder werden deze versies, die "Visual" genoemd werden, geïmplementeerd als macro talen in vele productiviteitstoepassingen, wat de verspreiding onder mensen die anders nooit met programmeren begonnen zouden zijn, vergrootte. We zullen hier de verschillende versies die voor Linux beschikbaar zijn, of toch tenminste, de meest gekende, bespreken, en een poging wagen ze met elkaar te vergelijken.

Een stukje geschiedenis

bron: Andrea M. Marconi's History, gevonden in de Kbasic documentatie:

De geboorte van BASIC

De Basic taal (Beginners All-Purpose Symbolic Instruction Code) werd in 1964 aan het Dartmouth College in New Hampshire (USA) geboren, waar het ontwikkeld werd door John G. Kemeney (1926-1993) en Thomas E. Kurtz (1928 -). Kemeney, werkte eerst in het Manhattan Project (1945) en werd later (1948-1949) de assistent van Albert Einstein. En dat is wanneer hij in Dartmouth in 1956 Kurtz ontmoette.

Beiden begonnen aan een nieuwe, vereenvoudigde programmeertaal te werken, en na de zogenaamde Darsimco en DOPE versies, begonnen ze naar een taal te zoeken met volgende specificaties:

1. Algemeen gebruik
2. Eenvoudig in gebruik
3. Uitbreidbaar
4. Interactief
5. Met een debugging helpsysteem
6. Efficient
7. Hardware onafhankelijk
8. Onafhankelijk van het besturingssysteem

Om dit te verwezenlijken gingen ze uit van Fortran en ALGOL. De eerste afgewerkte versie werd de "Dartmouth BASIC" genoemd, en verschafte 14 instructies. De "Dartmouth BASIC", was er één die compileerde en kan voor zijn tijd snel genoemd worden.

Op 1 mei 1964 om 4 uur in de morgen werden de inspanningen van Kemeney en Kurtz beloond, wanneer twee BASIC programma's simultaan liepen op een General Electric 225 UC van de Dartmouth College.

De groei van BASIC

Kemeney en Kurtz beschermden hun uitvinding niet met een patent en lieten het als publiek eigendom achter. Dat liet toe dat de taal, maar ook het aantal versies, groeide. Onder de eerste gebruikers vinden we General Electric die zijn GE-255 aan Dartmouth verkocht.

Gordon Eubanks (de CEO van Symantic) lag aan de basis van vele BASIC's, waaronder de E-basic die het daglicht zag in 1970. Het gebruikte een pseudo code zoals java dat vandaag de dag doet. Dan kwam CBASIC en vele andere versies die ertoe leidden dat de ANSI in 1974 standaarden begon te definiëren. Deze laatste werden nooit opgevolgd daar ze pas in 1978 verschenen, wanneer BASIC al wijdverbreid was.

Ondertussen werd in 1975 TinyBASIC door Bob Albrecht en Dennis Allison ontworpen. Het kan uitgevoerd worden met 2KB RAM. Ook verschenen de eerste geïnterpreteerde BASIC-versies. Het werd door William H. Gates III (1988-, aka Bill) en Paul Allen (1953-) ontworpen. De computermakers begonnen een kopie van BASIC in de ROM van hun machines mee te leveren. Aan het einde van de jaren '70 hadden de eerste Personal computers hun eigen versie van BASIC.

- Radio Shack Level 1 BASIC (TRS 80)
- Apple Integer BASIC (Apple II, 1977)
- Timex Sinclair 1000 BASIC (Sinclair ZX80, 1980)
- Sinclair ZX81BASIC (Sinclair ZX81, 1981)
- PET BASIC (Commodore PET, 1977)
- Atari BASIC (Atari 400/800, both 1978)
- Commodore BASIC (VIC 20 in 1981 & C64 in 1982)
- TI-BASIC (Texas TI-99)
- etc...

De Evolutie van BASIC

Sinds het begin van de jaren '80 is de geschiedenis van BASIC nauw verwant aan die van computers. Twee producten domineerden de markt. IBM leverde de BASIC A met PC DOS, geïnterpreteerd en in ROM, maar uitbreidbaar. MS-DOS leverde de Gee-Witz BASIC (of GW).

In 1984 verschijnt de Microsoft BASIC compiler, gevolgd door vele versies waaronder de QuickBASIC serie, die in 1985 startte en in 1990 met de komst van de Microsoft BASIC professional Development System 7.1 stopte.

Eens te meer is het het besturingssysteem die veranderingen aan de taal brengt wanneer hij de gebruiker laat kennismaken met de grafische interfaces. (GUI). Visual BASIC beweert een objectgeoriënteerde taal te zijn, wat door vele programmeurs betwist wordt. Een recent onderzoek echter wijst uit dat ongeveer 90 % van alle programma's die voor Windows 9x ontworpen zijn, geschreven zijn in de visual BASIC taal.

BASIC(s) vandaag

Als we een inventaris van het aantal BASICs die voor linux beschikbaar zijn, proberen te maken, kunnen we zo'n half dozijn projecten vinden, die de moeite van het vermelden waard zijn. Er is een "Basic Foundry" in sourceforge om een classificatie te geven van het aantal downloads:

TOP DOWNLOADS van Sourceforge.net zijn:

1. XBasic
2. SmallBASIC
3. wxBasic
4. GNU/Liberty Basic
5. YaBASIC
6. X11-Basic

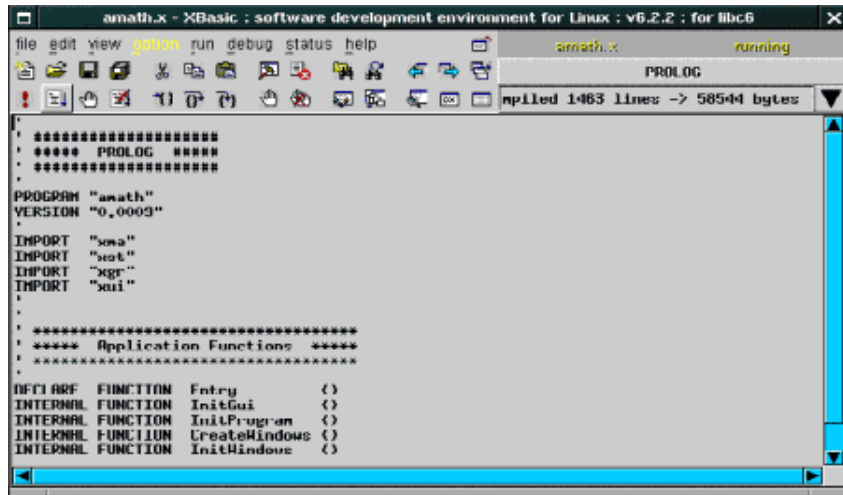
We komen ook te weten dat zowel GNOME als KDE een project hebben met de bedoeling Visual Basic te vervangen. Op 10 februari 2002 sprak Richard Stallman trouwens over de nood van een vrij alternatief voor VB in een brief aan "The register".

Anthony Liquori (ajl13-at-bellatantic.net), auteur van GLBCC (GNU/Liberty Basic Compiler Collection), het enige Basic Project met een GNU kaartje, vraagt dit ook op de GLBCC website (lbpp.sourceforge.net).

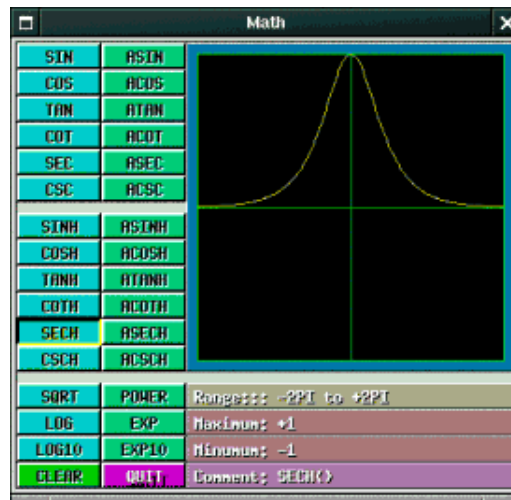
Laten we even de verschillende BASIC's onder de loep nemen om te kijken naar de specificaties van elke interface, en naar de mogelijkheden van de taal.

XBasic

XBasic is waarschijnlijk de ontwikkelingsomgeving met de meest geavanceerde gebruikersinterface. Het is een IDE, waarmee bedoelt wordt dat het over een bouwomgeving voor GUI, een debugger en een compiler beschikt. Een "punt-commando" systeem laat toe om in plaats van de muis het toetsenbord te gebruiken voor het bouwen van interfaces, door het schrijven van commando's die gevolgd worden door een punt in het schrijfgebied in de rechterbovenhoek van het hoofdvenster:



XBasic, Tekening 1 a: het hoofdvvenster



XBasic, tekening 1b: De wiskundige toepassing

XBasic heeft op het gebied van interessante kenmerken alle vereiste bibliotheken die nodig zijn om grafische interfaces en vele uitbreidingen te programmeren. Laten we ook even de mogelijkheid vermelden om functies - die in de C-taal geschreven zijn - aan te roepen. Vele kenmerken van de C taal zijn beschikbaar, zoals het declareren van types, de associatie van variabelen of het maken van bibliotheken.

Als slot vermelden we dat XBasic beschikbaar is onder de GPL voor Windows en Linux op <http://www.xbasic.org>.

SmallBASIC

SmallBASIC is een interpreter in consoleversie voor Win32, Linux en PalmOS. Het ontwikkelingsgedeelte is heel goed gedocumenteerd om het overzetten naar andere besturingssystemen

te bevorderen. De interpreter kan voor verschillende interfaces gecompileerd worden:

- SVGALIB
- Frame Buffer
- SDL

Het kan in commandoversie of in grafische versie uitgevoerd worden. Het volgende voorbeeld laat het programma System_info.bas uitvoeren:

Console mode

```
$ sbasic System_info.bas
SmallBASIC version 0.8.2, use -h for help
http://smallbasic.sourceforge.net
VMT Initialization...
File: System_info.bas
Section: Main
PASS1: Line 24; finished
SB-MemMgr: Maximum use of memory: 30KB
PASS2: Node 3/3
Creating byte-code...
Variables 20
Labels 0
Proc/Func 0
Code size 707
System Information
OS:      Unix/Linux version 2.4.18-6mdk (quintela @ bi.mandrakesoft.com)
        (gcc version 2.96 20000731 (Mandrake Linux 8.2 2.96-0.76mdk))
        #1 Fri Mar 15 02:59:08 CET 2002 204018
SB:      802
Display 99x58
Colors 16
Font:   1x1
Total free memory: 127728 KB
Stack size: 127728 KB
Largest free memory block: 127728 KB
* DONE *
SB-MemMgr: Maximum use of memory: 30KB
$
```

Grafische versie

```
$ sbasic -g System_info.bas
SmallBASIC version 0.8.2, use -h for help
http://smallbasic.sourceforge.net
VMT Initialization...
File: System_info.bas
Section: Main
PASS1: Line 24; finished
SB-MemMgr: Maximum use of memory: 30KB
PASS2: Node 3/3
Creating byte-code...
```

```
Variables 20
Labels 0
Proc/Func 0
Code size 707
SB-MemMgr: Maximum use of memory: 30KB
$
```



Tekening 2 : SmallBASIC SDL grafische mode

De SmallBASIC taal is eenvoudig en nadert de standaardfuncties die men van een BASIC mag verwachten. De grafische functies bieden niets nieuw en je kunt de klassieke rechthoeken en circels uitvoeren op elk van de hierbovengenoemde besturingssystemen. Er is geen typering van variabelen. SmallBASIC is echter wel compatibel met het oude TINYBasic en Quickbasic en is heel goed in PalmOS geïntegreerd.

Het is beschikbaar van <http://smallbasic.sourceforge.net>

wxBasic

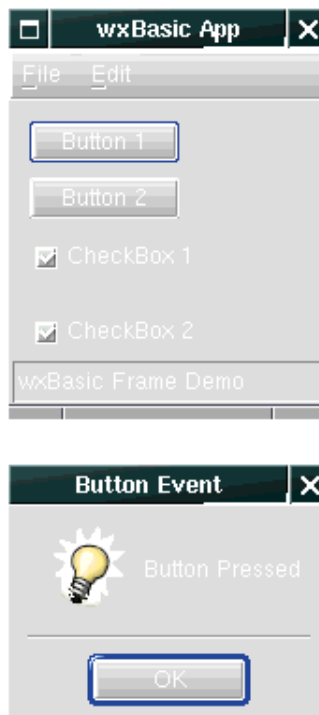
Van wxBasic wordt verondersteld dat het de kenmerken van Quick Basic heeft en enkele specificaties van Unix zoals de geassocieerde arrays die in gawk gevonden kunnen worden. Het is een kleine interpreter: het past op één diskette. De documentatie is compleet en verkrijgbaar als een pdf bestand van 138 pagina's. De taal heeft een grafische bibliotheek zodat je programma's voor X window of voor windows kunt schrijven. Deze taal leunt dicht aan bij objectgeoriënteerde talen zoals C++, althans wat het ontwerpen van GUI's betreft. De grafische interfaces dienen echter wel manueel ontworpen te worden. Er is geen geïntegreerde ontwikkelingsomgeving voor deze BASIC beschikbaar.

```
// My first wxBasic demo...
option explicit
// create the window
dim frame=new wxFrame(0,-1,"wxBasic App",wxPoint(10,10),wxSize(300,200))
frame.Centre()
// place a panel in the window
dim panel = new wxPanel(frame, -1)
// add a status bar
dim sBar = frame.CreateStatusBar( 1 )
sBar.SetStatusText("wxBasic Frame Demo")
// attach a menubar to the window
```

```

dim mBar = new wxMenuBar()
frame.SetMenuBar(mBar)
// build the "File" dropdown menu
dim mFile = new wxMenu()
mBar.Append(mFile, "&File")
// populate it
mFile.Append(wxID_NEW, "&New", "Creates a new file")
mFile.Append(wxID_OPEN, "&Open", "Loads an existing file from disk")
mFile.Append(wxID_SAVE, "&Save", "Saves current file")
mFile.Append(wxID_SAVEAS, "Save &As", "Saves current file with new name")
mFile.AppendSeparator()
mFile.Append(wxID_EXIT, "&Exit", "Exit Application")
// build the "Edit" dropdown menu
etc.....

```



Tekening 3: wxBasic: één van de demo programma's

Website: <http://wxbasic.sourceforge.net>

GNU/Liberty Basic

Wordt ook GLBCC genoemd (GNU Liberty Basic Compiler Collection), het is een compiler, of nauwkeuriger een C gnu gcc compiler interface die BASIC naar C converteert, zoals sommige C++ compilers die voordat ze compileren c++ naar C overzetten. Zoals de naam doet vermoeden verplicht deze compiler zich ertoe om 100% compatibel te zijn met Liberty BASIC. De Liberty BASIC was één van die BASIC's die in de jaren '90 op de windowsplatformen verscheen, en die door zijn vrije beschikbaarheid (vandaar zijn naam) succesvol was. Kijk om hier meer over te weten eens op deze website die lof zingt over zijn hoge kwaliteit. Deze taal is niet vrij, maar een Windows versie van

Liberty Basic kan kosteloos van de website gedownload worden.

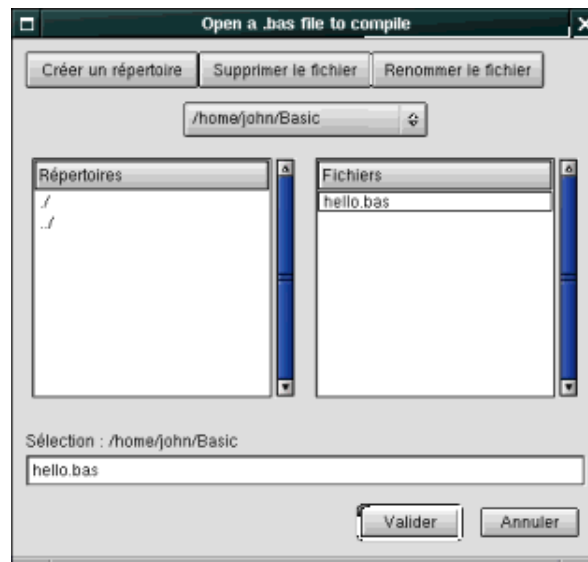
De GLBCC compiler is voor zowel windows als linux beschikbaar en kan onafhankelijk uitvoerbare bestanden maken, waarvan gezegd wordt dat ze net zo snel zijn al waren ze in welke andere taal dan ook geprogrammeerd. De auteurs roepen zelfs uit dat Liberty BASIC code die met GLBCC gecompileerd werd Visual BASIC belachelijk kan doen lijken wat snelheid betreft.

Het installeren van GLBCC onder Linux is eenvoudig en vereist enkel het decomprimeren van het klassieke "tar"bestand en een "make install".

In normale toestand, wordt het programma in een shell gebruik en door het typen van "glbcc hello.bas" zal een uitvoerbaar bestand, zoals in onderstaand voorbeeld, gemaakt worden:

```
$ glbcc
/usr/local/bin/lbpc -I/usr/local/lib/glbcc-lib/0.0.7/include -o out.c hello.bas
gcc -g -I/usr/local/lib/glbcc-lib/0.0.7/include `gnome-config --cflags gnomeui`
-o hello out.c /usr/local/lib/glbcc-lib/0.0.7/lib/lbcrct0.o
-L/usr/local/lib/glbcc-lib/0.0.7/lib -lLB
-lm `gnome-config --libs gnomeui`
$ ls -l hello*
-rwxr-xr-x 1 john john 339671 oct 13 21:55 hello
-rw-r--r-- 1 john john 22 avr 14 17:41 hello.bas
$ cat hello.bas
print "Hello, world!"
$ ./hello
Hello, world!
```

Wanneer het zonder parameters gebruikt wordt, opent GLBCC een grafisch dialoogvenster en vraagt de gebruiker om de naam van het BASIC bestand en de naam van het uitvoerbaar bestand die aangemaakt dient te worden. Standaard wordt de ingegeven naam van het programma gebruikt voor het uitvoerbaar bestand, met een .exe uitbreiding voor Windows, en geen uitbreiding voor Linux.



Tekening 4: GNU/Liberty Basic

Als taal is deze BASIC compleet en de GUI is gebaseerd op de GTK bibliotheek. Het interessante aan GLBCC is dat het zelf in BASIC geschreven werd.

YaBASIC

Yet another Basic voor Unix en Windows

Zelfs al gedraagt het zich dan als een interpreter, YaBasic is geen interpreter. Het is eerder een compiler: wanneer je het broncode om te verwerken geeft, zal het dit compileren, d.w.z. het omzetten naar machinetaal en het meteen uitvoeren. YaBasic wordt vanuit de shell gebruikt. Met de bestandsnaam als argument, zal het dat bestand uitvoeren zoals we net beschreven hebben. Zonder argumenten zal het de interpretermode, zoals in onderstaand voorbeeld, binnengaan:

```
$ yabasic
Enter your program, type RETURN twice when done.
for i=1 to 10
  print i,"hello, world"
next i
1hello, world
2hello, world
3hello, world
4hello, world
5hello, world
6hello, world
7hello, world
8hello, world
9hello, world
10hello, world
$
```

Deze manier van werken laat je toe om Unix scripts of zelfs CGI scripts met YaBasic te schrijven, zolang je maar de regel "#!/bin/yabasic" als eerste regel van het bestand opneemt. Als taal is Yabasic standaard (d.w.z. dicht tegen Quick BASIC) zonder variabeletypen.

X11-BASIC

X11-BASIC is een gestructureerde BASIC interpreter met grafische mogelijkheden uit X11. Het gebruikt de kenmerken van de GFA BASIC die op de Atari ST gevonden werd. Het is de poort van BASIC die gebruikt werd om vanuit de Atari ST GEM/TOS systeem (einde van de jaren '80) naar Unix te gaan. Het kan als interpreter gebruikt worden of om scripts of CGI's te schrijven. Er is een pseudo compiler beschikbaar die ofwel de code meteen omzet naar onafhankelijke uitvoerbare bestanden (statische versie), of gelinkt naar de X11 basic bibliotheek die zo'n 200 kB bedraagt (dynamische versie). Het wordt een pseudo-compiler genoemd, omdat het bekomen bestand geen machinetaal bevat die meteen door de CPU wordt uigevoerd, maar een gecomprimeerde vorm van de BASIC taal met zijn interpreter. Deze pseudo-compiler is zelf in X11-BASIC geschreven.

De taal van deze BASIC is heel rijk, gestructureerd en met getypeerde variabelen (integer, floating, text, array, boolean). Instructies zijn beschikbaar om toegang tot het geheugen te krijgen zoals de C malloc() of de vermenigvuldigingsmatrix die toepasbaar is op arrays.

De grafische instructies zijn dezelfde als die van de Atari ST GFA BASIC, maar hebben een

gelijkaardig resultaat onder X. De MENU instructie bijvoorbeeld maakt een menu in een grafisch venster. Deze interpreter wordt geleverd met documentatie en vele voorbeelden, op sommige plaatsen nog steeds in de Duitse taal. De schaduwzijde is dat de overzetting nog niet geheel af is, en daardoor bugs kunnen optreden, zelfs met de geleverde voorbeelden. Laten we aannemen dat deze interpreter zich hoofdzakelijk als een beta versie gedraagt en gemakkelijk gebruikt kan worden door de categorie van gebruikers waarvoor de BASIC- taal in eerste instantie ontworpen was.

```
$ xbasic
*****
*                xbasic                V. 1.07                *
*                by Markus Hoffmann 1997-2002 (c)            *
*                                                                 *
* version date:           Wed Sep 25 10:26:29 CEST 2002      *
* library V. 1.07 date:   Wed Sep 25 10:26:29 CEST 2002      *
*****
```

X11 Basic: de thuishaven van de interpreter

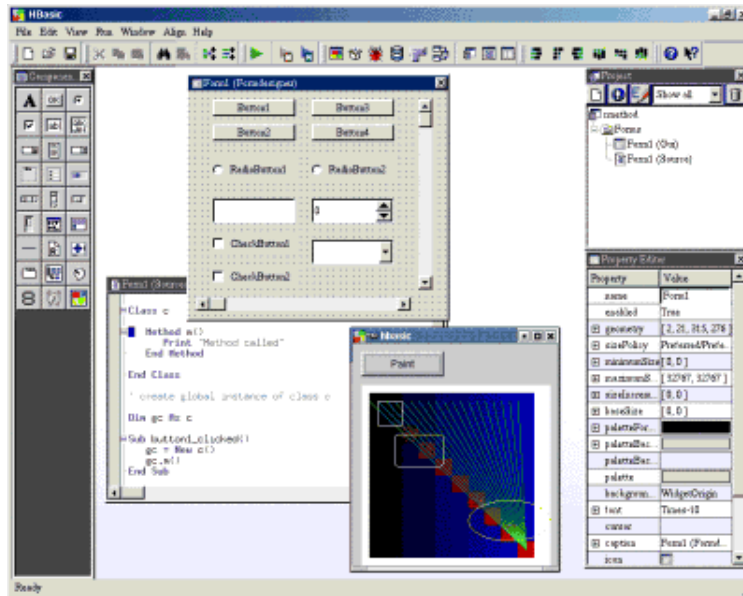
Website: <http://www-cip.physik.uni-bonn.de/~hoffmann/X11-Basic/>

HBasic

Dit is een basic die een uitstekende eerste indruk maakt, enerzijds door zijn mogelijkheden en anderszijds door de kwaliteit van zijn documentatie die 7,7mb bedraagt. Qt-3 is voor de installatie vereist, wat van de trolltech website (<http://www.troll.no>) beschikbaar is, als je geen nieuwe linuxdistributie draait. Het is een complete ontwikkelingsomgeving die alle kenmerken bezit waarvan een programmeur kan dromen (ondanks zijn jonge leeftijd: versie 0.8):

- Geïntegreerde ontwikkelingsinterface, GUI ontwerper, en eigenschappenvenster
- Het laden en gebruiken van voorgedefinieerde pakketten voor het invoegen van formulieren en grafische objecten in programma's.
- Broncode editor met woordoplichting, het completeren van woorden en modules wrapping
- Levert een interpreter om compilatie voor het testen te vermijden
- De compiler is geïntegreerd en maakt echte uitvoerbare bestanden.
- Een .NET code compiler kan in een .NET omgeving uitgevoerd worden.
- Debugger: beschikt over breakpoints, een viewer voor de inhoud van variabelen (ofwel tijdens het uitvoeren, of wanneer je over de naam van een variabele gaat in de code editor).
- De mogelijkheid om objecten in C++ aan te maken om HBasic programma's of de ontwikkelingsinterface tijdens de uitvoering uit te breiden.
- Objectgeoriënteerde taal voor de klassen die in de HBasic broncode zijn gedefinieerd of voor de componenten die als pakketten worden geladen.
- Gereedschappen voor de ontwikkeling van geïntegreerde databases die toelaten om toegang te krijgen tot de data ofwel vanuit de IDE ofwel vanuit het programma
- Het maken en gebruiken van instanties van Qt klassen
- .NET ondersteuning: wat toelaat om het gehele bereik van .NET bibliotheken en hun methoden, eigenschappen en velden te gebruiken.
- De mogelijkheid om c# code te bewerken en te compileren vanuit de HBasic IDE.
- Rekenbladen en grafieken (nog in alfa versie).

De auteurs van HBasic waarschuwen: "De huidige versie van HBasic is niet stabiel genoeg om door BASIC ontwikkelaars gebruikt te worden. Zij zullen moeten wachten op de vrijgeving van de eerste 1.0 stabiele versie."



Tekening 5: HBasic

GNOME Basic

GNOME Basic is een project die als doel heeft 100% compatibel te zijn met Visual BASIC, VBA, VBscript en andere gelijkaardige talen. In het begin had het tot doel om er voor te zorgen dat VBA Macro's in Gnumeric konden worden uitgevoerd, maar door onverwachte veranderingen voerde het maar enkele formulieren uit, controleerde het het grootste deel van de VBcode en voerde het enkelvoudige ASP's uit. Met een betere integratie in Bonobo kan een omgeving die volledig compatibel is met MS verwacht worden.

Momenteel is Gnome Basic een poging om kenmerken die VB compatibel zijn te leveren voor het Gnome project, en vooral voor productiviteitsapplicaties (VBA).

Het project is nog niet eens in alfa-staat en dient nog gereserveerd te blijven voor de ontwikkelaars van het Gnome Project.

Website: <http://www.gnome.org/gb/>

KBasic

KBasic is een andere poging om een BASIC compatibiliteit met Visual basic te krijgen. De ontwikkelaars hopen de eerste 1.0 stabiele versie voor de zomer van 2003 klaar te hebben. Momenteel is er enkel een onstabiele versie die gereserveerd is voor ontwikkeling. KBasic zou de Kdevelop omgeving

moeten gebruiken.

Dit is hoe de downloadbare versie er op het moment uitziet:

```
$ kbasic1 --help
Usage: kbasic1 [OPTION]... FILE
    --help                display this help and exit
    -V, --version          print version information and exit
    -c, --copyright        print copyright information and exit
$ kbasic1 -V
KBasic version 0.8
Copyright (C) 2000, 2001, 2002 The KBasic Team
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE,
to the extent permitted by law.
Visit us on http://www.kbasic.org/
$ kbasic1 hello.bas
-- scan --
LINE 00001 PRINT STRING "Hello, world!" EOL
LINE 00002 END
-- parse --
STRING in line 1
EOL in line 1
parsed PRINT
parsed END
-- interpret --
0000: KBASIC    KBASIC
0006: VER      1.000000
0011: OPTION R OFF
0013: JMP      36
0036: DIM_STAT "Hello, world!", 1
0055: JMP      18
0018: PUSHS   1 , " Hello, world! "
0023: PRINT
Hello, world!
0034: END      execution ended
```

Zoals je kunt zien werkt KBasic, maar het is verre van operationeel, op het ogenblik zijn de interpreter debugging boodschappen en de gegenereerde assembleercode onvermijdelijk.

Site: <http://www.kbasic.org>

Besluit

Dit overzicht van BASIC interpreters en compilers verschaffen ons wat informatie. Eerst en vooral blijkt verrassend genoeg deze zo verguisde taal nog steeds te leven, zelfs op Unixsystemen waar zijn gebruik eerder gereserveerd blijft door zijn vele concurrenten (perl, python, Tcl/Tk, php...), die ook in geïnterpreteerde versies uitgevoerd kunnen worden. De activiteit die deze taal onder ontwikkelaars voortbrengt is veelzeggend. Het aantal projecten zijn nogal hoog en ze zijn allen zeer actief. De vraag is waarschijnlijk nogal groot en dit zou de ontwikkeling van vrije software ten goede kunnen komen, daar BASIC programmeurs door Unix zouden worden aangetrokken.

VisualBASIC is wijdverbreid op Windowsplatformen door zijn integratie met het Microsoft OS, de

ASP's en zijn productiviteitsformules. Maar, laten we opmerken dat de BASIC's die we getest hebben op beide platformen werken, sommiges zelfs op nog enkele andere.

BASIC heeft echter nog steeds enkele nadelen. Het gebrek aan standaarden leidt tot vele versies waarvan de één meestal niet compatibel is met de andere. De wens van sommige projecten zoals Gnome Basic of KBasic om de standaard van VisualBasic te vergezellen zou een goed idee kunnen zijn moest VB vrij zijn, wat het duidelijk niet is. Monopolie zou in dit geval handiger zijn...

Voor de BASIC ontwikkelaar is er één ware keuze en enkele gereedschappen komen dicht in de omgeving. Als geïntegreerde omgeving is HBasic waarschijnlijk de meest belovende. Terwijl we wachten op de stabiele versie kan er het meest met XBasic bereikt worden. Smallbasic en Yabasic laten degene die net naar Unix zijn overgestapt echter toe om scripts of CGI's te schrijven zonder verplicht te zijn een keuze te maken uit de vele oplossingen die dit systeem biedt. Ook GLBCC laat dit toe, maar het heeft het voordeel dat het programma's creëert die gecompileerd worden met de grootse GCC. Door zijn gebrek aan een IDE en dus ook van een gebruiksvriendelijke debugger, zal het moeilijk zijn (blijven) om grote programma's met dit gereedschap te maken en te onderhouden. Als laatste vermelden we dat wx-Basic tot geen enkele categorie behoort, maar dat het toch enkele voordelen heeft, zoals zijn mogelijkheid om grafische interfaces te creëren.

Een ander voordelig punt van deze BASIC's is zijn gemakkelijke migratie van één OS naar een ander zonder recompilering. In feite zijn de meeste zowel beschikbaar voor win32 als voor unix omgevingen. De BASIC ontwikkelaar staat dan voor een zware taak: het juiste gereedschap voor zijn doel kiezen.

Site onderhouden door het LinuxFocus editors team © John Perr "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Vertaling info: fr --> -- : John Perr < johnperr(at)Linuxfocus.org > fr --> en: Georges Tarbouriech < gt%28at%29Linuxfocus.org > en --> nl: Cyberprophet < cyberprophet/at/linux.be >
---	---